RESEARCH ON COMPUTING SCIENCE

Avances en Control, Instrumentación Virtual, Sistemas Digitales, Arquitectura de Computadoras, Robótica y Procesamiento de Señales

Sergio Suárez Guerra
Carlos Aguilar Ibáñez
Jesús Figueroa Nazuno
(Eds.)

RCS
Research on Computing Science

Research on Computing Science

Avances en Control, Instrumentación Virtual, Sistemas Digitales, Arquitectura de Computadoras, Robótica y Procesamiento de Señales

Vol. 9

Research on Computing Science

Comité Editorial de la Serie / Series Editorial Board

Editores en Jefe / Chief Editors:

Juan Luis Díaz de León Santiago (México)
Gerhard Ritter (Estados Unidos)
Jean Serra (Francia)
Ulises Cortés (España)

Editores Asociados / Assoaciate Editors:

Jesús Angulo (Francia)
Oscar Camacho (México)
Jihad El-Sana (Israel)
Jesús Figueroa (México)
Alexander Gelbukh (Rusia)
Ioannis Kakadiaris (Estados Unidos)
Serguei Levachkine (Rusia)
Petros Maragos (Grecia)
Julian Padget (Reino Unido)
Miguel Torres (México)
Mateo Valero (España)
Cornelio Yáñez (México)

Diseño y Formación: / Design and Format

Luis Hernández Lara
Miguel R. Silva Millán
Ma. del Carmen Rodríguez Aparicio

Diseño de Portada: / Design Front

Ignacio García Araoz

Research on Computing Science es una publicación trimestral, de circulación internacional, editada por el Centro de Investigación en Computación del IPN, para dar a conocer los avances de investigación científica y desarrollo tecnológico de la comunidad científica iberoamericana. **Volumen 9**, septiembre, 2004. Tiraje: 500 ejemplares. Registro ante la Dirección General de Derechos de Autor en trámite. Certificado de Licitud de Título y Contenidos por la Comisión Calificadora de Publicaciones y Revistas Ilustradas de la Secretaría de Gobernación en trámite. El contenido de los artículos es responsabilidad exclusiva de sus respectivos autores. Queda prohibida la reproducción total o parcial, por cualquier medio, sin el permiso expreso del editor, excepto para uso personal o de estudio haciendo cita explícita en la primera página de cada documento. Impreso en la Ciudad de México, en los Talleres Gráficos del IPN – Dirección de Publicaciones.
Editor Responsable: *Juan Luis Díaz de León Santiago, DISJ 690604*

Volumen 9

Avances en Control, Instrumentación Virtual, Sistemas Digitales, Arquitectura de Computadoras, Robótica y Procesamiento de Señales

Editores del Volumen:

Sergio Suárez Guerra
Carlos Aguilar Ibáñez
Jesús Figueroa Nazuno

Instituto Politécnico Nacional
Centro de Investigación en Computación
México 2004

ISBN: 970-36-0180-4
ISSN: 1665-9899

Copyright © Instituto Politécnico Nacional 2004

Instituto Politécnico Nacional (IPN)
Centro de Investigación en Computación (CIC)
Av. Juan de Dios Bátiz s/n esq. M. Othón de Mendizábal
Unidad Profesional "Adolfo López Mateos", Zacatenco
07738, México D.F., México

http://www.ipn.mx
http://www.cic.ipn.mx

Impresiones: 500
Impreso en México

Prólogo

La presente edición contiene una selección de los trabajos presentados en el **5to. Congreso Internacional en Control, Instrumentación Virtual y Sistemas Digitales CICINDI'2004**, con sede en el Centro de Investigación en Computación del Instituto Politécnico Nacional.

El **5to. Congreso Internacional en Control, Instrumentación Virtual y Sistemas Digitales CICINDI'2004**, ha servido de foro para el encuentro de los especialistas en las disciplinas de cómputo asociadas a las aplicaciones, enfatizándose en esta ocasión por gran cantidad de trabajos en Control y Procesamiento de Señales.

El Centro de Investigación en Computación, a la vanguardia del Instituto Politécnico Nacional en tareas de aplicación de cómputo, ha mantenido su empuje en la formación de especialistas en las diferentes áreas que el país demanda, así como propiciar encuentros de intercambio entre las múltiples empresas e instituciones del gobierno que requieren la aplicación y solución de problemas con técnica de cómputo.

Queremos agradecer a todos los trabajadores del Centro su colaboración y apoyo para poder llevar a cabo esta obra y a todos los miembros del Comité Científico y de Revisión que aportaron sus criterios y sugerencias.

Dr. Juan Luis Díaz de León Santiago
Director del Centro de Investigación en Computación

NOTA ACLARATORIA

El día 31 de agosto del 2004 fue autorizado el registro de **ISSN** a la publicación "**Research on Computing Science**"; por lo que a partir de este volumen 9 (*Avances en Control. Instrumentación Virtual, Sistemas Digitales, Arquitectura de Computadoras, Robótica y Procesamiento de Señales)*, se hace constar que el registro correspondiente es: **ISSN 1665-9899.**

Unidad de Publicaciones e Informes Técnicos
Centro de Investigación en Computación
Instituto Politécnico Nacional

Contenido

Pág.

Arquitectura de Computadoras y Sistemas Digitales

Diseño de bloques básicos para simulación de sistemas asincrónicos en Matlab 3
Guadalupe Ayala Lomeli, Gonzalo Isaac Duchén Sánchez

Propuesta para la construcción de generadores de secuencias pseudoaleatorias basados en LFSR 13
Francisco I. Peralta, Gonzalo I. Duchén, Rubén Vázquez

Space qualified computer server developed for long life microsatellite applications 25
E. Vicente Vivas, J. F. Mendienta J., A. Calvillo T., M. Tapia R.

Evaluation of a space qualified long life flight computer server 37
E. Vicente Vivas, J. F. Mendienta J., A. CalvilloT., R. M. Tapia

Robótica e Instrumentación Virtual

Bauhaus: Development of a walking robot 53
Francisco Padilla, Javier Porras, Israel Martínez, Luis Ávila, Carlos Soberanes, Juan Navarro, Jorge Barrera, Leonardo Téllez

Un ambiente de simulación y de realidad virtual para asistir el diseño de redes hidráulicas 63
Jaime Muñoz Arteaga, Jannet Cruz Pérez, Gustavo Rodríguez Gómez

Sistema de supervisión y adquisición de datos en un ambiente distribuido 73
Juan Francisco Novoa Colín, Luis Pastor Sánchez Fernández

Instrumento virtual para la adquisición y evaluación del patrón de radiación de una antena 85
Víctor Manuel Medina Meneses

Instrumentos virtuales para laboratorios de oleajes utilizados en el diseño de obras costeras y marítimas 91
Fernando Berrelleza, Luis P. Sánchez

Correlation of gradients and the algorithm SUSAN for motion estimation and tracking 101
R. Aburto G., A.C. González G., J.H. Sossa A.

Control y Tiempo Real

Diseño vía LMIs del generador difuso de residuales para detección de fallas en un sistema de nivel de tres tanques 111
Efraín Alcorta García, Salvador Saucedo Flores

Diseño robusto del estabilizador de un sistema de potencia con ubicación de polos 127
Salvador Saucedo Flores, José A. Ortega Herrera

Pág

Desarrollo de un sistema de control de posición de un servomecanismo 141
I. I. Siller Alcalá, R. Acántara Ramirez, J. Jaimes Ponce, R. Rodríguez Ruiz, J. Cinco Galicia, E. Ruiz May

Chattering control of DC-DC PWM boost converter 151
Eduardo Barba, Leonardo Acho

Configuración de un sistema de control distribuido delta V para un sistema de calentamiento 157
J. C. Ravelo Hernández, J. Martínez Ham, I. I. Siller Alcalá, R. Acántara Ramirez, J. Jaimes Ponce, M. Cabrera Jiménez

Modelo estocástico para tiempos de arribo en tareas de tiempo real 167
Daniel Cruz Pérez, José de Jesús Medel Juárez, Pedro Guevara López

Reducción del error en la medición de velocidad en máquinas rotatorias 177
José de Jesús Medel Juárez, Pedro Guevara López, Alberto Flores Rueda, Asdrúbal López Chau

Desarrollo e implantación de un control de velocidad para motores de corriente continua usando el microcontrolador LM629 195
Alberto Flores Rueda, José de Jesús Medel Juárez, Pedro Guevara López

Visual servo control for an underactuated system 207
A. Espejel Rivera, L. E. Ramos Velasco

Estación de desarrollo de control difuso (EDCD) 217
Haydée Patricia Martinez Hernández, Germán A. Muñoz Hernández, José Federico Ramírez Cruz,

Diseño del subsistema de distribución de potencia para el microsatélite experimental mexicano SATEX 229
Andrés Calvillo Téllez, Esaú Vicente Vivas

Chattering adaptive control for first order systems 243
Ricardo Guerra, Leonardo Acho

Stability analysis for switched systems 251
E. Rubio, F. Rios

Integral high gain observers: Theory and experimental evaluation 261
Rubén Garrido Moctezuma, Ernesto Flores Garcia, Rafael Martínez Guerra

Sistema de control difuso usando IVs 273
Romeo Urbieta Parrazales, Luis Pastor Sánchez Fernández, Diego Ramírez Martínez

Análisis y Procesamiento de Señales

Estudio de señales electroencefalográficas para la detección de movimientos voluntarios 287
Aldo Escalona Delgado, Evelyn Peredo Fuentes

Pág

Reconocimiento de objetos traslapados usando memorias asociativas 297
Roberto A. Vázquez, Humberto Sossa, Ricardo Barrón

Digital processing of a ECG using a DSP: Wavelet approach 307
S. Razo Zapata, L. E. Ramos Velasco

Identificación de objetos por radio frecuencia 319
Sergio Sandoval Reyes

A single sensor active noise control with a hybrid structure using FxLMS and FxRLS algorithms 331
Miguel A Bravo Parra, Héctor Pérez Meana, Mariko Nakano Miyatake

Análisis de desempeño de un filtro para el mejoramiento de imágenes de huellas dactilares 341
Victor Hugo Garcia Ortega, Osvaldo Espinosa Sosa

Diseño de un sistema encriptador para mensajes telefónicos 351
René Rodriguez Ávila, Alfonso Gutiérrez

Reconocimiento de patrones por medio de redes neuronales en sistemas de cultivo acuícola automatizados 361
José Ángel Alcaraz Vega, Luis Pastor Sánchez Fernández

La medición de perturbaciones de frecuencia como indicador de la aspereza de la voz" 373
C. Ferrer, M. E. Hernández Diaz

A quantitative and qualitative comparison between the empirical mode decomposition and the wavelet analysis 381
H. Solis Estrella, A. Angeles Yreta, V. Landassuri Moreno, J. Figueroa Nazuno

Creación de un sistema experto clasificador de estructuras silábicas para tareas de reconocimiento de voz mediante sílabas 391
José Luis Oropeza Rodriguez, Sergio Suárez Guerra, Ricardo Barrón Fernández

Images retrieval using wavelets, histograms and sub images 403
A. C. González G., J. H. Sossa A., E. M. Felipe R.

Detección de rasgos y control de calidad en la manzana 413
Verónica Barzola L., Moisés Bonilla E., J. Humberto Sossa A., Ricardo Barron F.

Organización 425

Índice de Autores 426

Arquitectura de Computadoras y Sistemas Digitales

Diseño de bloques básicos para simulación de sistemas asincrónicos en Matlab

Guadalupe Ayala Lomeli, Gonzalo Isaac Duchén Sánchez.

Sección de Estudios de Posgrado e Investigación (SEPI)
Escuela Superior de Ingeniería y Mecánica y Eléctrica (ESIME)
Instituto Politécnico Nacional (IPN)
Unidad Profesional Culhuacán
Av. Santa. Ana No.1000., Col. San Francisco Culhuacán, C.P. 04430,
México, DF.

lomeli79@calmecac.esimecu.ipn.mx

Abstract. The use of a central clock has begun to have some fundamental limitations like: overheat of the circuit, inefficiency in the use of the energy, emission of electromagnetic noise and synchronization failures[1,2]. One solution to this problem is the asynchronous design, reducing the limitations of the synchronous circuits. So, it is necessary to begin to handle these circuits, but the problem now is that there are no tools for the asynchronous design as in the case of the synchronous design, for this reason, in this paper is described the development of a library that contains a set of asynchronous blocks designed in Matlab to make simulations of totally asynchronous circuits.

Resumen. El uso de un reloj central ha comenzado a tener algunas limitaciones fundamentales como: sobrecalentamiento del circuito, ineficiencia en el consumo de la energía, emisión de ruido electromagnético y fallas de sincronía [1,2]. Una solución a este problema es el diseño asincrónico, que reduce en gran medida las limitaciones de los circuitos sincrónicos. Como se ha observado es necesario comenzar a manejar estos circuitos, pero existe el problema de que no hay herramientas para el diseño asincrónico como es el caso del diseño sincrónico, por tal motivo, en este artículo se describe el desarrollo de una librería que contiene un conjunto de bloques asincrónicos diseñados en Matlab para poder realizar simulaciones de circuitos totalmente asincrónicos.

Palabras clave: handshaking, circuito asincrónico, comunicación, latch, elemento muller c, simulink.

Introducción

En el diseño de circuitos asincrónicos no es necesario esperar un cambio en la señal de reloj para ejecutar alguna acción, en vez de esto, se emplea un protocolo (handshaking) entre sus componentes para llevar a cabo la sincronización necesaria, la comunicación y secuencia de operaciones, evitando el sobrecalentamiento del circuito y consumo de energía, ya que con este protocolo, los componentes solo trabajan cuando se les requiere y además elimina la emisión de ruido electromagnético debido a que no existe una señal de reloj para controlar acciones, como es el caso del diseño sincrónico.

La mayoría de los circuitos asincrónicos están basados en un protocolo que está conformado por una señal de petición y una señal de reconocimiento (Fig.1), en donde la señal de petición la genera el emisor y es la principal iniciadora de cualquier transferencia, mientras que la señal de reconocimiento la emite el receptor e indica la terminación de la transferencia [3]. Estas señales proporcionan el control de secuencia necesario para llevar a cabo el correcto funcionamiento del sistema asincrónico.

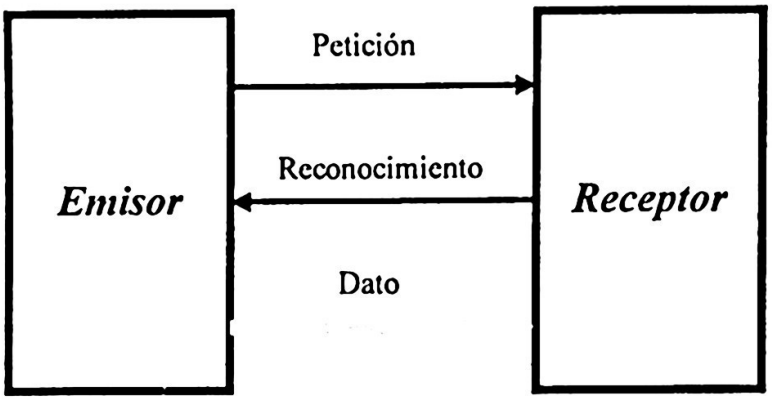


Fig. 1. Interfaz asincrónica

La codificación más popular para las señales de handshaking (petición y reconocimiento) son: el protocolo de 2 fases y 4 fases bundled data, también conocido como NRZ (sin retorno a cero) y esta se caracteriza por una señal alta (generalmente +5 ó +3,3 voltios) para un 1 binario y una señal baja (0 voltios) para un 0. El protocolo de 4 fases bundled data también conocido como RTZ (con retorno a cero) [4]. En el protocolo de 2 fases los eventos están indicados por solo 2 transiciones, ya sea de 1→0 ó de 0→1, en la figura 2 se observa el esquema de este protocolo.

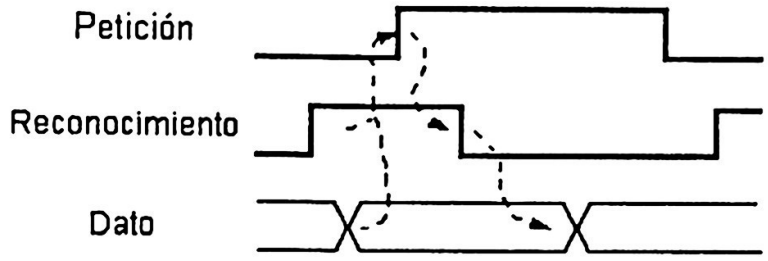


Fig. 2. Protocolo de 2 fases Bundled Data

En cambio, en el protocolo de 4 fases, los eventos están indicados por 4 transiciones, en donde la primera transición se lleva a cabo cuando el emisor necesita enviar un dato y la señal de petición se coloca en 1, la segunda transición es cuando el receptor recibe el dato y la señal de reconocimiento se coloca en 1, la tercera transición es cuando el emisor responde colocando la señal de petición en 0 y por ultimo la cuarta transición es cuando el receptor reconoce la acción realizada colocando la señal de reconocimiento en 0. En este momento, el emisor puede iniciar el siguiente ciclo de comunicación. En la figura 3 se observa el esquema de este protocolo de comunicación

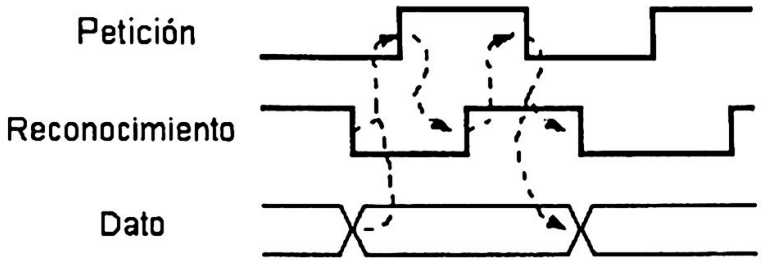


Fig. 3. Protocolo de 4 fases Bundled Data

Como se explicó arriba, en los circuitos asincrónicos la sincronización no es por medio de una señal de reloj, en vez de está, se emplean las señales de handshaking entre registros vecinos; ahora vamos a considerar que los registros son un conjunto de latches.

Las latches son los únicos componentes que inician y toman parte activa en el handshaking, mientras que los demás componentes son "transparentes" al handshaking. Para representar las latches, emplearemos el símbolo de una caja con doble línea vertical.

Para comenzar a familiarizarnos con el funcionamiento de las latches, en la figura 4 se muestra un pipeline compuesto por cinco latches. Los arreglos de cajas representan enlaces que consisten de petición, reconocimiento y señales de dato. El valor valido en L1 tiene que copiarse en L2 y el valor vacío en L3 tiene que copiarse en L4. Esto significa que L1 y L3 contienen ahora los duplicados viejos de los valores que han

sido almacenados en L2 y L4. Los duplicados viejos son llamados "bubbles" y los nuevos valores válidos y vacíos se llaman "tokens" [5].

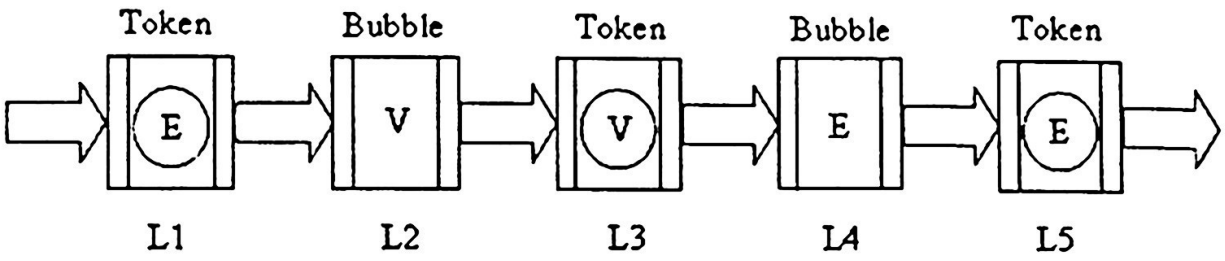


Fig. 4. Estado pipeline de 5 etapas

Ahora, para poder realizar la implementación de circuitos asincrónicos, es necesario contar con un conjunto mínimo de componentes, los cuales se pueden agrupar en cuatro categorías: Latch, Bloque de función, Control de flujo condicional y Control de flujo incondicional. (Fig.5).

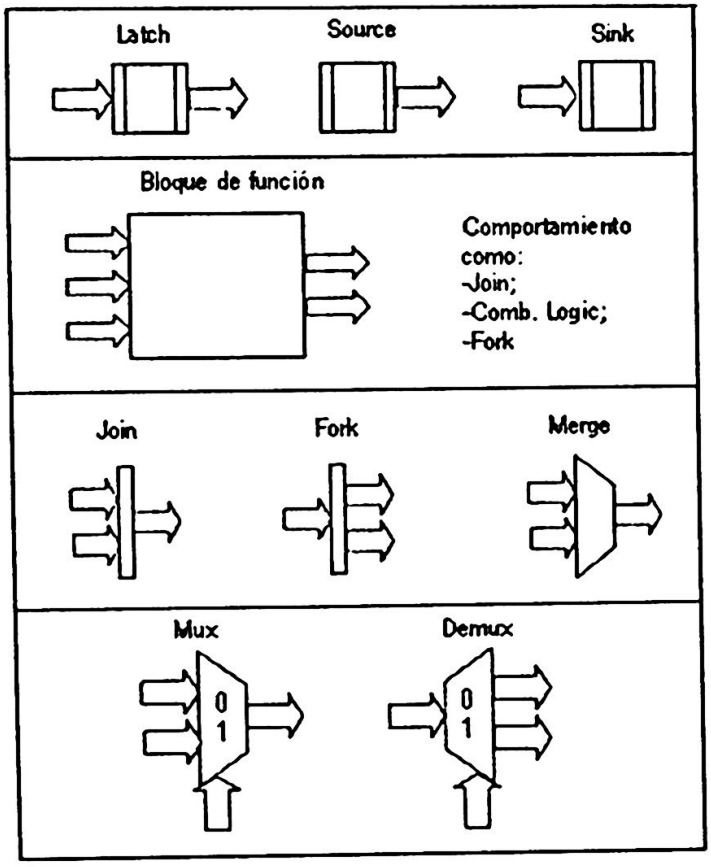


Fig. 5. Conjunto de componentes asincrónicos

El componente latch nos proporciona el almacenamiento de variables e implementa el handshaking. Una latch con solo un canal de salida es una fuente (*source)* que produce tokens (con el mismo valor constante) y una latch con solo un canal de entrada es un consumidor (*sink)* que consume tokens.

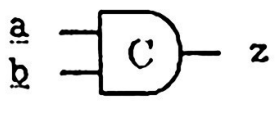
El bloque de función realiza lo siguiente: (1) Espera tokens en sus entradas (un join implícito), (2) realiza la función combinacional necesaria y (3) pone en circulación los tokens en sus salidas.

Los componentes de control de flujo incondicional son: fork, join y merge, donde un fork se emplea cuando la salida de un componente es la entrada para más componentes; un join se emplea cuando el dato de varios canales independientes necesita ser sincronizado, normalmente por que son entradas (independientes) para un circuito. El salir en todas direcciones de un canal implica un fork, y el entrar en todas direcciones de varios canales implica un join. El componente merge tiene dos o más canales de entrada y un canal de salida., el handshake en la entrada del canal es mutuamente exclusivo.

Los componentes de control de flujo condicional son el mux y demux, los cuales realizan la función usual de seleccionar entre varias entradas para una salida y varias salidas para una entrada. Un mux sincronizará el control del canal. Similarmente el demux sincronizará el control y los canales de entrada de datos y dirige la entrada al canal seleccionado de salida.

Implementación de bloques

La implementación de estos componentes se realizó con la ayuda de S-Function Builder, que es una utilería de simulink que permite construir bloques por medio de código C [6]. Para comenzar la implementación de los circuitos mencionados arriba, primero fue necesario analizar y programar el elemento Muller C, ya que es un componente fundamental para la mayoría de los circuitos asincrónicos y permite sincronizar los procesos. Este elemento consta de 2 entradas y 1 salida, en donde la salida será 1 si las entradas son 1, la salida será 0 si la entrada es 0 y en los demás casos la salida no cambia. En la figura 6 se muestra la tabla de verdad y la simbología del elemento muller C.



a	b	z
0	0	0
0	1	no cambia
1	0	no cambia
1	1	1

Fig. 6. Elemento Muller C

El siguiente paso fue realizar la implementación del conjunto latch (source, sink y latch) conforme a la figura 7. El funcionamiento del conjunto latch es el siguiente: para que el dato se copie de una latch a otra, es necesario que la petición sea igual a un 1 lógico y el reconocimiento sea un 0 lógico, implicando esto que la siguiente latch está lista para copiar el dato, y así sucesivamente. Como se observa, por cuestiones de

diseño del elemento Muller C, es necesario invertir la señal de reconocimiento, ya que el elemento Muller C permite pasar el dato solo si la petición y el reconocimiento es igual a 1.

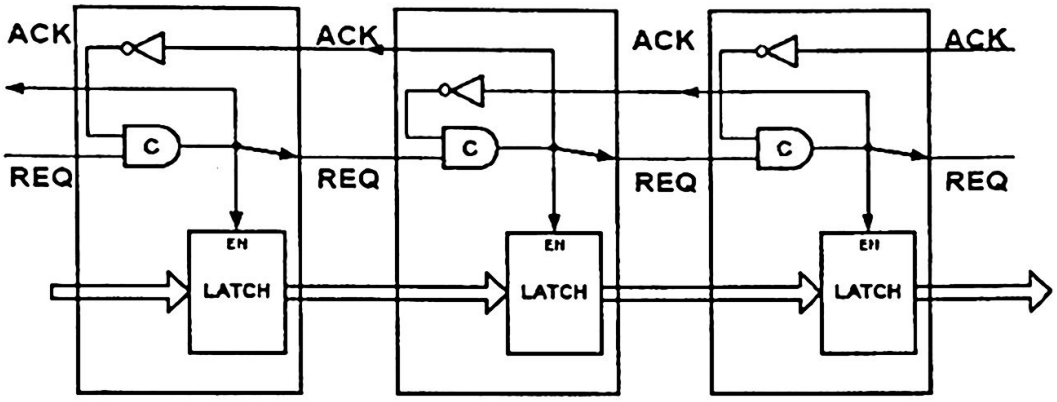


Fig. 7. Implementación de latch

La implementación del componente fork se llevó a cabo conforme a la figura 8. El fork simplemente duplica el dato de entrada y está compuesto por un elemento Muller C que controla las señales de reconocimiento.

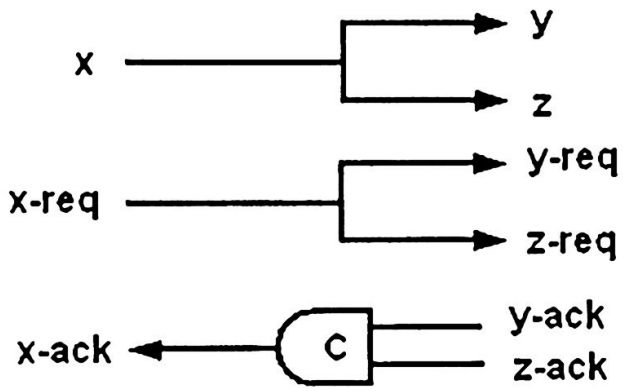


Fig. 8. Implementación de fork.

La implementación del componente merge, se realizó conforme a la figura 9. Este componente es un poco más complicado, ya que a diferencia de los demás, en los canales de entrada las señales de handshake son mutuamente exclusivas. En general la función del merge es sincronizar la transmisión de la entrada activa a los canales de salida.

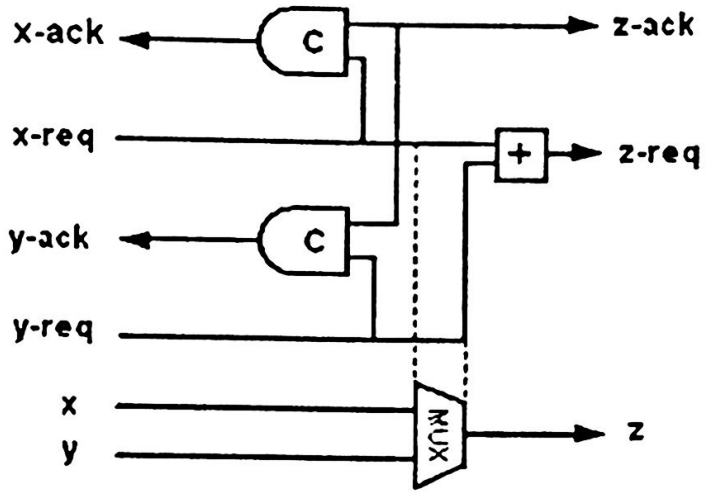


Fig. 9 . Implementación de merge

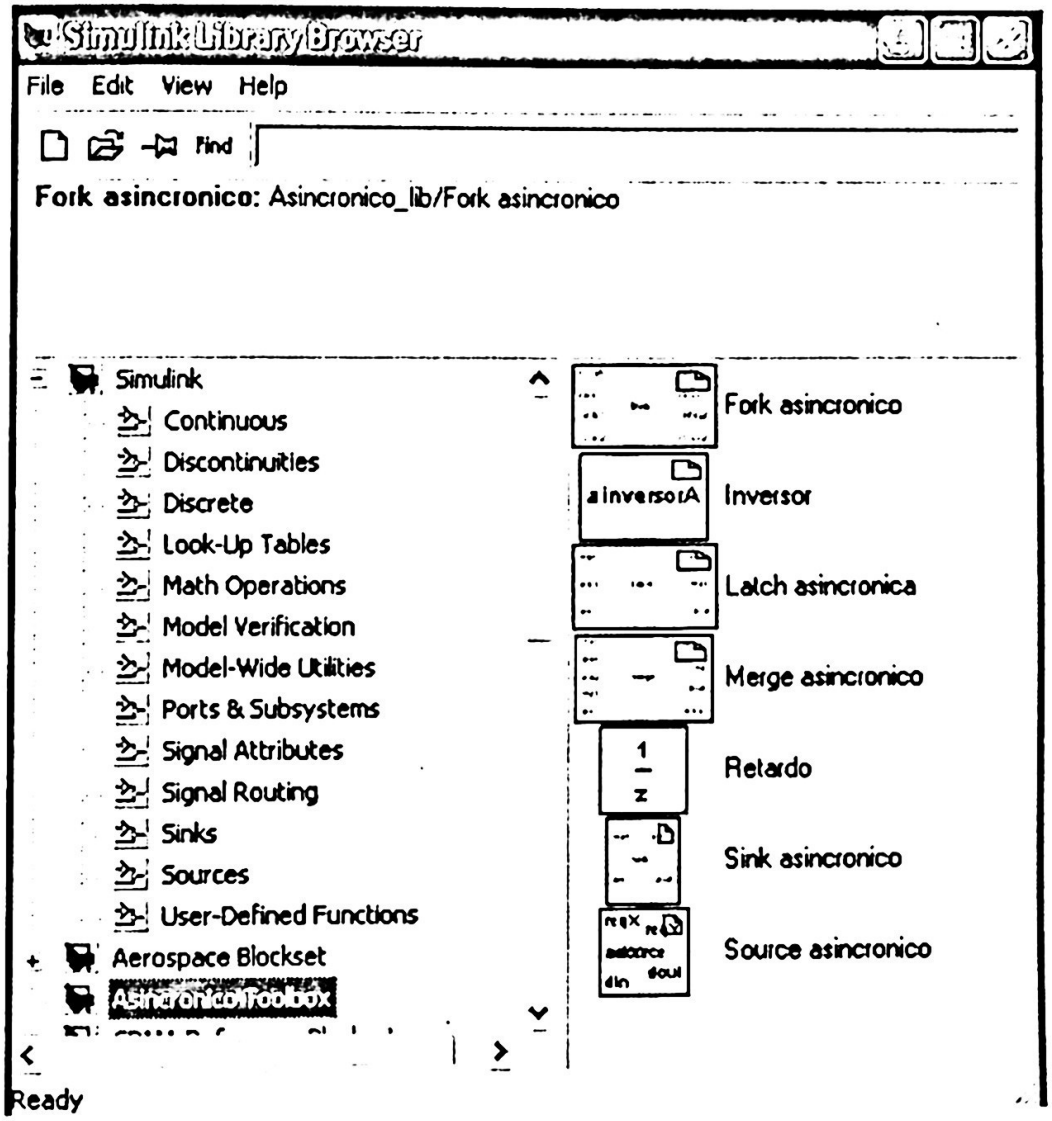


Fig. 10. Librería Asincrónica en Simulink

Ejemplos y resultados

Una vez que los bloques fueron implementados, el siguiente paso fue crear una librería dentro de Simulink, que se realizó con la ayuda de la función slblocks de matlab. La librería lleva el nombre de Asincrónico Toolbox y contiene los siguientes elementos: fork, latch, merge, retardo, inversor, sink y source. En la figura 10 se observa la librería y sus componentes.

En la figura 11 se muestra el ejemplo del funcionamiento de latch, sink y source, en el cual se realiza la transferencia del dato 999 en modo asincrónico. Primero es necesario realizar la petición por medio de un valor constante para comenzar la comunicación entre las latches. Como se observa, es necesario colocar un retardo por cada uno de los componentes, ya que este tipo de circuitos requieren retroalimentar la señal de reconocimiento y por cuestiones de simulación es necesario colocar el retardo para poderse llevar a cabo correctamente.

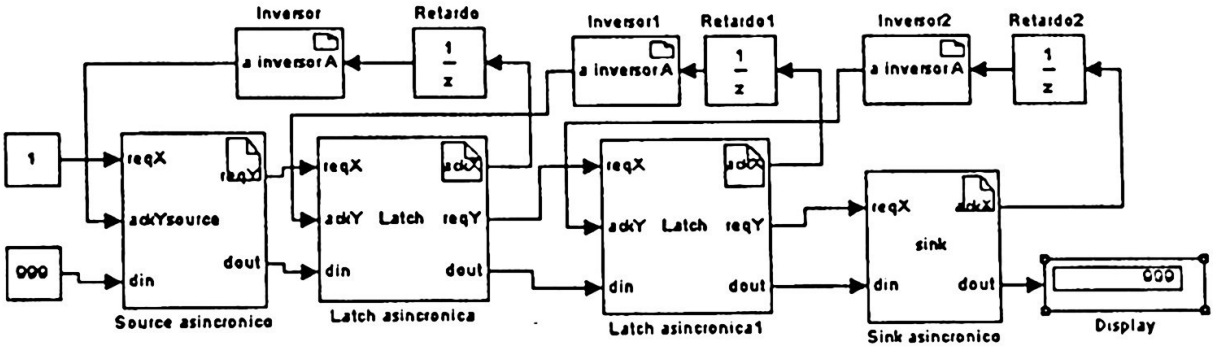


Fig. 11. Transferencia de datos asincrónico.

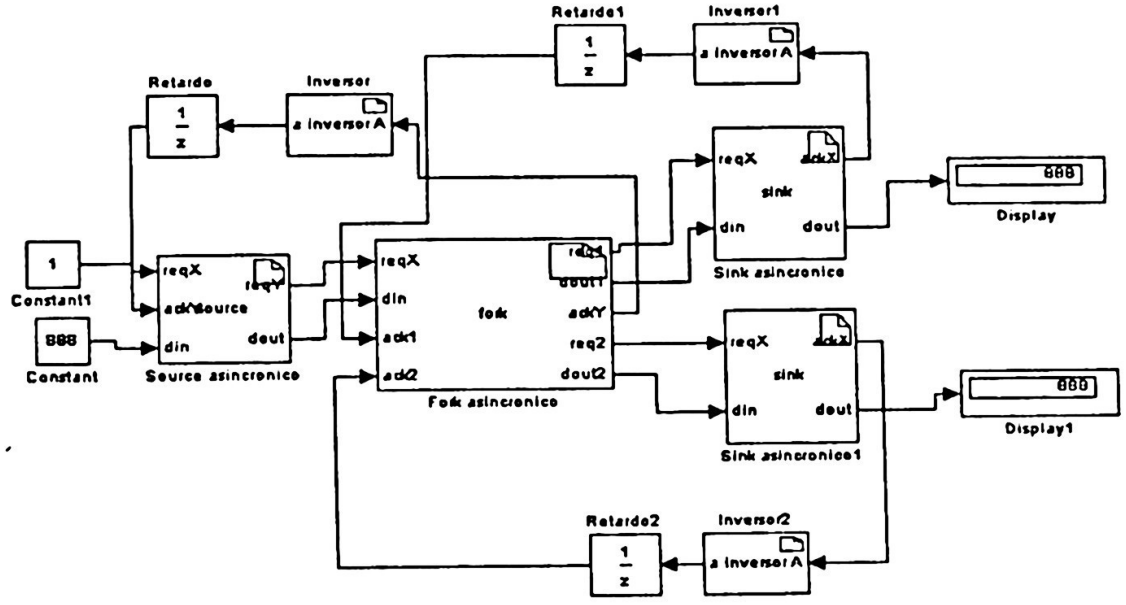


Fig. 12. Ejemplo del funcionamiento de fork

En la figura 12 se muestra el ejemplo del funcionamiento de sink, fork y source, en donde la función del fork es duplicar el dato 888 que le proporciona el source y ambos sinks consumen los datos duplicados.

Conclusiones

Los circuitos sincrónicos hoy en día tienen muchas limitaciones y es necesario comenzar a cambiar de visión y retomar el diseño asincrónico para contrarrestar esas limitaciones.

Un buen comienzo hacia el diseño asincrónico es la implementación de este conjunto de componentes que proporciona una herramienta muy útil para el usuario interesado en simular y diseñar circuitos asincrónicos.

Referencias

1.Bradley J. Benschneider, et. al., A 300-MHz, 64-b Quad-Issue CMOS RISC Microprocessor, IEEE Journal of Solid-State Circuits, vol. 30, no. 11, November 1995, pp. 1203-1214.

2.D.W. Dopperpuhl et. al., A 200-MHz 64-b dual-issue CMOS microprocessor, IEEE Journal of Solid-State Circuits, vol. 27, no. 11, November 1992, pp. 1555-1565.

3.A. Davis and S.M Nowick, "An introduction to asynchronous circuit design", Technical Report UUCS-97-013, Deparment of Computer Science, University of Utah, septiember 1997.

4.D.W. Lloyd and J.D. Garside, "A practical comparison of Asynchronous Design Styles", IEEE Computer Society Press, 2001.

5. J. Sparsø and S. Furber, "Principles of a synchronous circuit design" -A systems perspective, Kluwer Academic Publishers, 2001.

6 .Building S-Functions Automatically (Construcción automática de funciones S), Ayuda en línea de Matlab.

Propuesta para la construcción de generadores de secuencias pseudoaleatorias basados en LFSR

Francisco I. Peralta, Gonzalo I. Duchén, Rubén Vázquez.

Instituto Politécnico Nacional
Escuela Superior Mecánica y Eléctrica Unidad Culhuacan
Sección de Estudios de Posgrado e Investigación
Av. Santa Ana, 1000, Col. San Francisco Culhuacan, C. P. 04430,
Coyoacan, México D. F.
franciscoipc@calmecac.esimecu.ipn.mx

Abstract. This paper shows the results of using the Berlekamp-Massey algorithm to construct an SLFSR (Shortest Linear Feedback Shift Register) that can help to reconstruct a plaintext that was encrypted using an LFSR. It is also used to produce polynomials using sequences of bits. Additionally, the design of a function that represents an LFSR (Linear Feedback Shift Register) and the implementation into an FPGA are made with a combination between hardware description language like VHDL, and the result that produce the algorithm of Berlekamp-Massey. This function, without being built from a primitive polinomial, permits the hardware implementation of stream ciphers based on LFSRs.

Keywords: LFSR, VHDL, Linear Complexity, Berlekamp-Massey Algorithm, Massey-Rueppel Generator, Primitive Polynomials.

Resumen: Este artículo presenta los resultados de utilizar el algoritmo de Berlekamp-Massey para romper un cifrado hecho a partir de un LFSR, así como también para crear polinomios a partir de una secuencia de bits. Además con la ayuda de un lenguaje de descripción de hardware combinado con el resultado que produce el algoritmo de Berlekamp-Massey, se puede crear en un FPGA la función de un Registro de Desplazamiento con Retroalimentación Lineal (LFSR: Lineal Feedback Shift Register). Esta función, sin ser construida a partir de un polinomio primitivo, permite crear en hardware arreglos de LFSR, que generan secuencias pseudoaleatorias con mejores propiedades criptográficas que las secuencias generadas por la función original.

Palabras Clave: LFSR, VHDL, Complejidad Lineal, Algoritmo de Berlekamp-Massey, Generador Multivelocidad, Polinomio Primitivo.

Introducción

En la actualidad la protección de información por medio de técnicas criptográficas es una práctica muy importante para la transferencia segura de datos. La criptografía se puede aplicar por ejemplo, para dar confidencialidad a un mensaje o garantizar la autenticidad de criptogramas y/o remitentes y destinatarios. Actualmente existen distintos tipos de cifrado que ayudan a otorgar atributos de seguridad a una comunicación, entre ellos el cifrado de flujo que cifra bit a bit los datos de un texto claro, utilizando una secuencia cifrante y una función booleana. Este cifrado, realizado con LFSR posee las siguientes propiedades importantes:

. Los LFSR se pueden realizar fácilmente en hardware.
2. Pueden generar secuencias de periodo T grande, con $T = 2^L-1$ y L la longitud del LFSR.
3. Producen secuencias con buenas propiedades estadísticas, que pueden evaluarse con los postulados de Golomb [7]:
4. Debido a su estructura, pueden analizarse usando técnicas algebraicas, una de estas la complejidad lineal.

En telecomunicaciones es muy utilizado con las siguientes consideraciones:
Cuando el espacio de almacenamiento es limitado.
Cuando los caracteres deben procesarse individualmente y recibirse de igual forma.
Cuando la propagación de errores está limitada.
Cuando el error de transmisión tiene una probabilidad alta.

2 LFSR (Registro de Desplazamiento con Retroalimentación Lineal)

Un LFSR de longitud L consiste en L etapas (o unidades de almacenamiento) numerados desde 0 hasta L-1, cada una capaz de almacenar un bit; tiene una entrada, una salida y un reloj que controla el flujo de datos entre las distintas etapas.
Durante cada unidad de tiempo se realizan las siguientes operaciones [1]:
1. El contenido de la etapa 0 es la que otorga el flujo de salida.
2. El contenido de la etapa i se traslada a la etapa i-1 considerando que, $1 \le i \le L-1$; y
3. El nuevo contenido de la etapa L-1 es el bit de retroalimentación s_j, que se calcula por medio de una función booleana que depende de la combinación de los contenidos de ciertas etapas predefinidas.

La figura 1 muestra la estructura general de un LFSR de longitud L. Cada c_i es 0 ó 1; el bit s_j es una combinación lógica no lineal del contenido de las etapas L, según sea el valor del vector {c}. Un LFSR puede generar secuencias pseudo-aleatorias de tamaño máximo 2^L-1. La secuencia resultante se denomina m-secuencia. Para que un LFSR tenga el periodo máximo (2^L-1), el polinomio formado por {c} más 1 debe ser un polinomio primitivo módulo 2 [3]. A continuación se dan algunas definiciones para conocer las características que deben cumplir los polinomios primitivos.

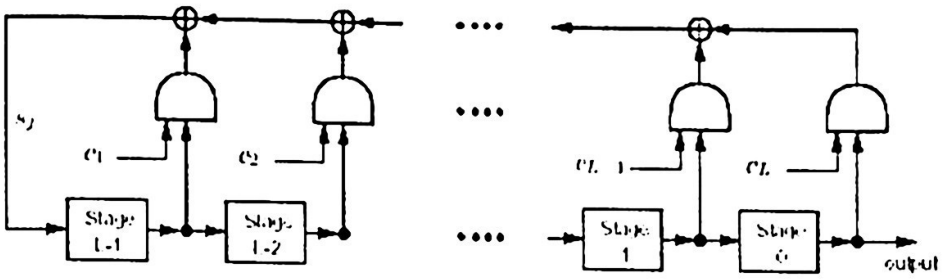


Fig. 1. LFSR de longitud L.

Definición. Si hay un polinomio p(x) tal que, r(x)a(x)=s(x), se dice que el polinomio s(x) es divisible por el polinomio r(x) o que r(x) divide a s(x). Un polinomio diferente de cero p(x) que solo es divisible por p(x) se dice irreducible ya que no puede ser escrito como el producto de dos polinomios, cada uno de grado positivo [6].

Definición. Un polinomio irreducible p(x) de grado L se dice que es primitivo solo si
1) p(x) divide a $X^n + 1$, con $n = 2^L - 1$, y
2) p(x) no divide a $X^n + 1$, cuando $n < 2^L - 1$.

3 Complejidad Lineal

Que un LFSR genere una secuencia con periodo máximo no implica que el cifrar alguna información con ella sea seguro. La complejidad lineal es una métrica importante para analizar a los LFSR y los generadores basados en LFSR, ya que mide la robustez o seguridad de un generador de secuencias pseudoaleatorias o cifrador de flujo. Concretamente, la complejidad lineal se define como la longitud L, del LFSR más pequeño o SLFSR (Shortest Linear Feedback Shift Register) que puede imitar la salida del generador. A continuación se darán las definiciones formales relacionadas con la complejidad lineal.

Definición. Sea $s = s_0, s_1, s_2, \ldots$ una secuencia infinita. La subsecuencia que consiste en los primeros n términos de s se denota por $s^n = s_0, s_1, s_2, \ldots, s_{n-1}$.

Definición. Un LFSR genera una secuencia s si hay un estado inicial para el cual secuencia de salida del LFSR es s. De forma similar, un LFSR genera una secuencia finita s^n si hay algún estado inicial para el cual la secuencia de salida del LFSR tiene s^n como sus primeros n términos.

Definición. La complejidad lineal L(s), de una secuencia binaria infinita s, se define como:
1) Si s es la secuencia cero s = 0, 0, 0,..., entonces L(s)=0;
2) Si ningún LFSR genera s, entonces $L(s)=\infty$;
3) De otra forma, L(s) es la longitud del LFSR más pequeño que genera s.

3.1 Propiedades de la Complejidad Lineal

Sean s y t secuencias binarias.
1) Para cualquier $n \geq 1$, la complejidad lineal de la subsecuencia s^n satisface $0 \leq L(s^n) \leq n$.
2) $L(s^n) = 0$ si y solo si s^n es la secuencia cero de longitud n.
3) $L(s^n) = n$ si y solo si $s^n = 0, 0, 0, ..., 0, 1$.
4) Si s es periódica con periodo N, entonces $L(s^n) \leq N$.
5) $L(s \oplus t) \leq L(s) + L(t)$, donde $s \oplus t$ denota la operación XOR de los bits de s y t [1].
Se sabe que el SLFSR generado a partir de una secuencia tiene una gran importancia práctica, en especial en criptografía y teoría de códigos. El algoritmo de Berlekamp-Massey es eficiente para obtener el SLFSR de una secuencia y se desarrolla a continuación.

4 Algoritmo de Berlekamp-Massey

El algoritmo de Berlekamp-Massey (ABM) hace uso de la complejidad lineal de una secuencia binaria finita s^n de longitud n para medir la robustez de un cifrador de flujo [4]. El algoritmo toma k iteraciones, con la k-ésima iteración escribe la complejidad lineal de la subsecuencia s^k que consiste en los primeros k términos de s^n [1]. Además, puede generar el SLFSR después de examinar solo 2L bits de la secuencia de salida. Donde L es la longitud del LFSR. Una vez generado, se podrá romper el cifrador ya que se conoce el polinomio con el cual se construye el SLFSR [3].

Definición. Considere la secuencia binaria finita $s^N = s_0, s_1, ..., s_{N-1}, s_N$. Para un polinomio de longitud L $C(D) = 1 + c_1 D + ... + c_L D^L$, sea $\langle L, C(D) \rangle$ un LFSR que genera la subsecuencia $s^n = s_0, s_1, ..., s_{n-1}$. Donde N es la longitud de la secuencia total generada por el LFSR y n es la longitud de una subsecuencia de la secuencia total. La discrepancia d_n es la diferencia entre S_n y el último bit generado por el LFSR [1]:

$$d_n = (s_n + \sum_{i=1}^{L} c_i s_{n-i}) \bmod 2 \qquad (1)$$

La demostración de que este algoritmo es correcto se puede consultar en The Stability Theory of Stream Ciphers C. Ding et al. (1991). EL ABM es un algoritmo iterativo que analiza una secuencia $s^n = s_0, s_1, s_2, \ldots, s_{n-1}$.
En el primer paso se inicializan las variables operativas del programa y las que representan al polinomio de combinación y la complejidad lineal.
La Inicialización de variables se realiza de la siguiente forma:
$C(x) \leftarrow 1$, $L \leftarrow 0$, $m \leftarrow -1$, $B(x) \leftarrow 1$ $k \leftarrow 0$ y $T(x) \leftarrow 0$
Para calcular la discrepancia se usa la ecuación (1). Y para calcular el vector C(x) se hace uso de las variables T(x), B(x), k y m.
$T(x) \leftarrow (x)$; $C(x) \leftarrow C(x) + B(x) \cdot x^{k-m}$.

En la figura 2 se muestra el diagrama de flujo del algoritmo de Berlekamp-Massey. A continuación se muestra un ejemplo de cómo el ABM genera un polinomio a partir de una secuencia de bits con las cuales se alimenta a este algoritmo.

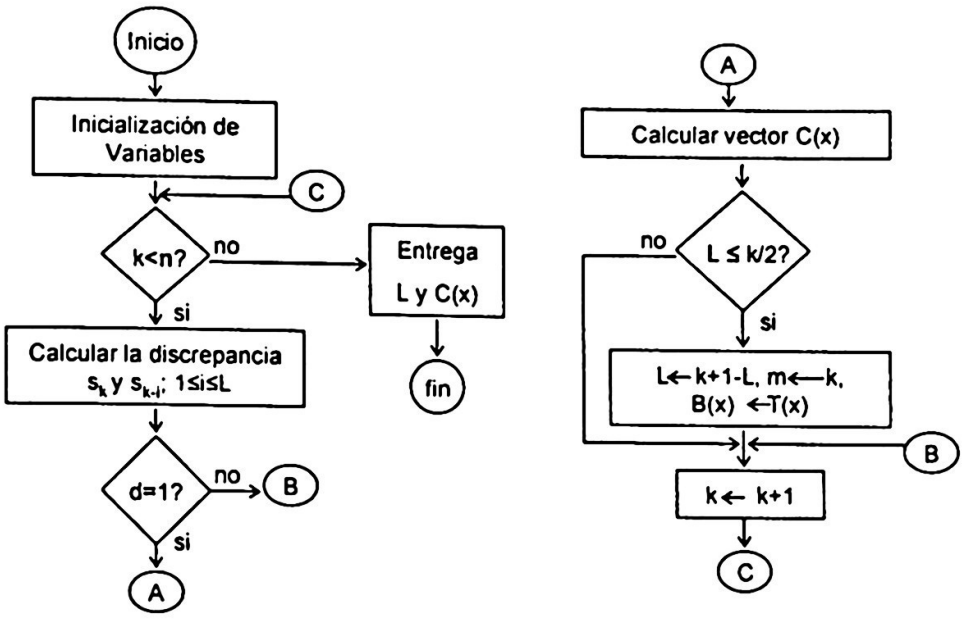


Fig. 2. Diagrama de flujo del algoritmo de Berlekamp – Massey.

Al introducirle al ABM una secuencia de bits como la siguiente: s^n: 011101101001.

El resultado que se obtiene es $P(X)=X^6+X+1$. Este es el polinomio con el que se puede construir un LFSR que consistirá de 6 etapas y se realizará una operación XOR con las salidas de la etapa 6 y 1. Este polinomio es primitivo ya que X^m+1 es divisible solo por P(X), con $m = 2^L-1$ y $L = 6$, pero no lo es cuando $m < 2^L-1$. Por tanto, la secuencia de salida total generada tiene un periodo máximo $(2^L-1) = 63$.

El diseño del Generador Multivelocidad que utiliza LFSR y se trata más adelante ha sido simulado en un FPGA de la familia MAX7000S (EPM7128SLC84-7) con las herramientas que proporciona ALTERA™ y ocupa el 27% de los recursos del chip. Es importante mencionar que no se ocupó ninguna directriz especial de síntesis para optimizar el diseño. Cuando se hace un diseño usando VHDL, herramienta de ALTERA™ le asigna por omisión un retardo a cada uno de los dispositivos que corresponden al FPGA (Field Programmable Gate Array) que se esté utilizando.

A continuación se muestra la secuencia total generada por el LFSR construido a partir del polinomio X^6+X+1 y la semilla 011101, siendo el primer bit de la izquierda más significativo.

Secuencia:011101101001001110001011110010100011000010000011111101010110
011...
Con el ABM y tomando solo 2L bits de cualquier porción de esta secuencia (en este caso L = 6), es decir con solo 12 bits, se puede encontrar que el LFSR que genera esa secuencia tiene el polinomio como ya sabemos X^6+X+1. Esto quiere decir que el cifrado usando la secuencia de salida de un LFSR, no puede asegurar que la información esté protegida, pues es relativamente sencillo obtener esta información usando criptoanálisis si se conocen 2L bits de la secuencia que se usó para cifrarla.

5 Generador Multivelocidad en VHDL

Si combinamos el resultado que ofrece el algoritmo de Berlekamp-Massey con el uso de un lenguaje de descripción de hardware como VHDL (VHSIC Hardware Description Language), se puede desarrollar un programa que genere un LFSR a partir del polinomio que entregue el algoritmo. El diseño de un LFSR en VHDL es muy sencillo, además se puede aprovechar esta herramienta para construir generadores de secuencias pseudoaleatorias que usan combinaciones de LFSR, como el Generador Multivelocidad de este artículo.

Este generador utiliza 2 LFSR que funcionan distintas frecuencias de reloj como se muestra en la figura 3 y el código en VHDL en la tabla 1. El LFSR de L etapas trabaja con un reloj $d \geq 2$ veces más que el de M etapas con $L \geq M$. El factor d es variable y se usa como parte de la clave. Según Rainer A. Rueppel (1986) la diferencia de frecuencia entre los LFSR proporciona flexibilidad y seguridad en aplicaciones criptográficas. Se pueden simular diferentes conexiones de retroalimentación sin cambiar físicamente los estados de retroalimentación, es decir, el polinomio primitivo que indica la retroalimentación puede permanecer constante y lo único que se modifica es la frecuencia, que es más sencillo que cambiar los polinomios de retroalimentación.

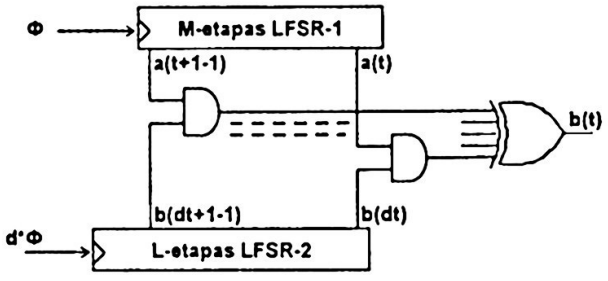


Fig. 3. Generador multivelocidad de Massey-Rueppel.

La incertidumbre acerca de la recursión utilizada en un generador de secuencias pseudoaleatorias hace más difícil el trabajo de un criptoanalista. En particular, la diferencia de frecuencia entre los LFSR disuade al criptoanalista de intentar un ataque de correlación sobre una combinación no lineal de LFSR, pues intentaría buscar a través de todas las recursiones posibles.

Table 1. Código en VHDL del Generador Multivelocidad.

```
library ieee; use ieee.std_logic_1164.all; use ieee.std_logic_arith.all;
library gmr; use gmr.comps_gmr.all;
entity gmr is
generic(M : integer := 7; L:integer:=10; dist : integer := 3; PM : integer := 121;
        INIM : integer := 41;  PL : integer := 911; INIL : integer := 887);
port(clock,ci: in std_logic; salM, salL: out std_logic_vector(M-1 downto 0);
     sal: out std_logic);
end gmr;
Architecture vhdl of gmr is
        signal QA:  std_logic_vector(1 downto 0);
        signal TAP_A,sal_and: std_logic_vector(M-1 downto 0);
        signal TAP_B:  std_logic_vector(L-dist-1 downto 0);
        signal sal_xor:  std_logic;
begin
DIV :    contador port map(clock,QA);
LM:      lfsrm generic map (ancho=>M, polinomio=>PM, inicializacion=>INIM)
         port map(QA(1),TAP_A,ci);
LL:      lfsrl generic map (ancho=>L, polinomio=>PL, inicializacion=>INIL)
         port map(QA(0),TAP_B,ci,QA(1)) ;
AG:      andg generic map(ancho=>M) port map(TAP_A,TAP_B,sal_and);
XORG:  orexg generic map(ancho=>M) port map (sal_and,sal_xor);
salM<=TAP_A;
salL<=TAP_B;
sal<=sal_xor;
end vhdl;
```

5.1 Propiedades del generador multivelocidad

Si los LFSR empleados en el generador se forman con polinomios irreducibles, sus grados L y M son primos relativos y los factores $d_1 \neq d_2$ se escogen tal que:

$$\gcd(d_1, T_1) = \gcd(d_2, T_2) = 1 \tag{2}$$

T_1 y T_2 son los periodos de los LFSR, entonces b(t) tendrá complejidad lineal:

$$CL = LM \tag{3}$$

Con periodo:

$$T \geq \frac{(T1)(T2)}{(q-1)} \qquad ($$

donde q denota el campo de Galois (GF(q)), en este caso q=2 que denota el campo binario [7]. Cuando el generador está restringido a emplear LFSR de longitud L y

(L > M) con polinomios primitivos sobre GF(2), entonces el número de unos y ceros dentro de un periodo $l=(2^M-1)(2^L-1)$ de la secuencia de salida b(t) son:

$$U(1) = (2^M - 1)2^{L-1} \tag{5}$$

$$U(0) = (2^M - 1)(2^{L-1} - 1) \tag{6}$$

y la diferencia de U(1) y U(0) dividida entre la longitud l del periodo es

$$\frac{U(1)-U(0)}{l} = \frac{1}{(2^L-1)} \tag{7}$$

Debe mencionarse que para q>2, los resultados estadísticos no son tan fáciles de obtener [7].

6 Pruebas y Resultados

Se introdujo secuencias de bits al ABM para obtener polinomios que sirven como entrada al código en VHDL para generar los LFSR. A partir de estos, se construye el Generador Multivelocidad. Las secuencias de bits utilizadas son las siguientes:
1) Para el LFSR de M etapas: S = 101010001011110000.
2) Para el LFSR de L etapas: S = 10011101111101110010001.
El ABM proporcionó para el primer caso el siguiente polinomio:
$x^7+x^4+x^3+x^2+x+1$; M = 7. Tiene un periodo de 21 con la siguiente semilla: 1001010.
En el segundo caso, el polinomio obtenido es $x^{10}+x^9+x^8+x^7+x^3+x^2+x+1$; L=10. El periodo de la secuencia de este polinomio es 4 con la siguiente semilla: 1110111011
El código en VHDL para este generador se muestra en la tabla 2 con M = 7 y L = 10 y la simulación se observa en las figuras 4 y 5.
La secuencia de salida del Generador Multivelocidad, usando un factor de frecuencia d=2, es la siguiente. Secuencia: 11111111111111111111111111101111111111111

Este generador proporciona una secuencia con periodo T = 42. El instante en que el generador comienza a entregar los bits de salida es en t = 49.5 ns, y cuando entrega la secuencia completa es en t = 3.41 µs. Como se observa, esta secuencia no es aleatoria, las razones por las cuales este generador no obtuvo el periodo máximo dado por $(2^M-1)(2^L-1)$ y no tiene aleatoriedad se debe a que:

1) los polinomios que se obtuvieron por medio del ABM no son primitivos,
2) la condición $\gcd(d_2, T_2) = 1$, no se cumple debido a que $\gcd(2,4) \neq 1$, y
3) debido a estas razones las ecuaciones (5), (6) y (7), no se cumplen.
Ahora, se construirá el mismo generador pero usando polinomios primitivos.
1) El LFSR de M-etapas es: x^6+x+1 y su semilla es: 011101.
2) El LFSR de L-etapas es: $x^{15}+x+1$ y su semilla es: 110000000000001.

El factor de frecuencia continuará siendo d=2.

Como se están utilizando LFSR con polinomios primitivos se debe cumplir que:
El periodo de la secuencia de salida debe ser: $T = (2^6 - 1)(2^{15} - 1) = 2{,}064{,}321$ bits
El máximo común divisor es: $\gcd(1,63) = 1$ y $\gcd(2,32767) = 1$
y el número de 0's y 1's como se establece en (5) y (6) es:
$U(1) = (2^6 - 1)2^{15-1} = 1{,}032{,}192$; $U(0) = (2^6 - 1)(2^{15-1} - 1) = 1{,}032{,}129$

En las figuras 6 y 7 se muestran el instante de inicio en t= 45.2 ns y final en t=165.1457252 ms de la simulación.

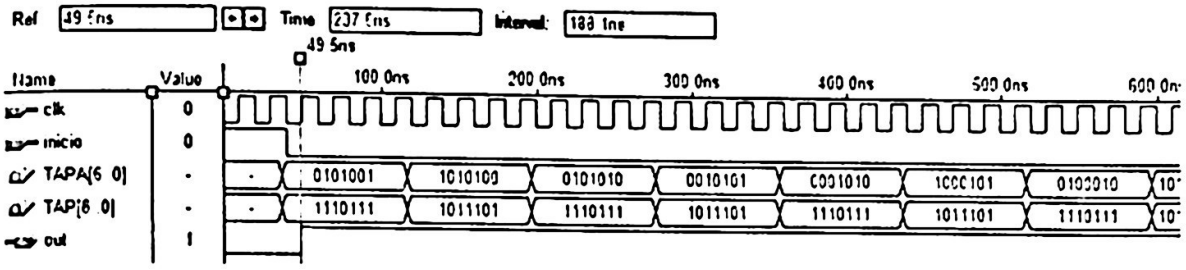


Fig. 4. Inicio de la secuencia pseudoaleatoria en *t = 49.5 ns.*

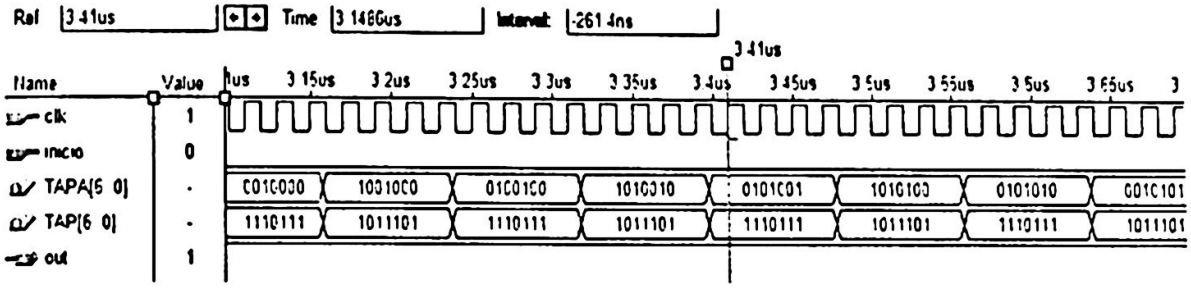


Fig. 5. Final de la secuencia pseudoaleatoria en *t = 3.41μs.*

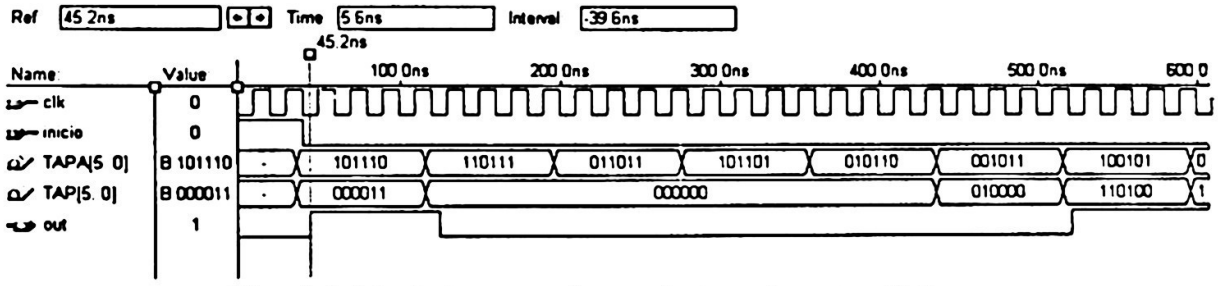


Fig. 6. Inicio de la secuencia pseudoaleatoria en *t = 45.2 ns.*

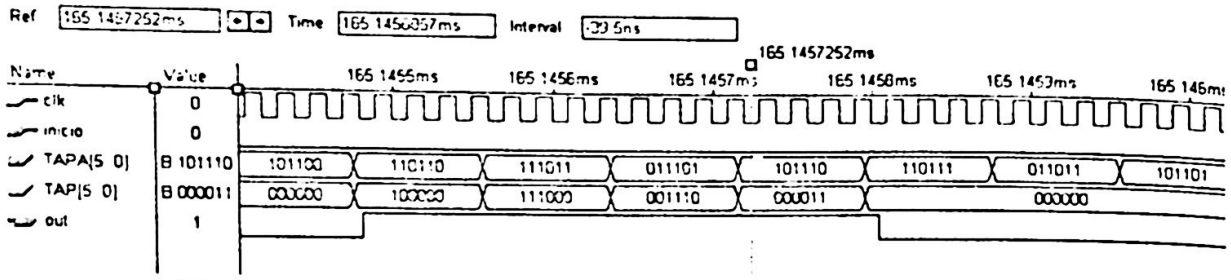


Fig. 7. Final de la secuencia pseudo aleatoria en $t = 165.1457252\ ms$.

7 Conclusiones

Se presentó el algoritmo de Berlekamp-Massey como una herramienta muy eficaz para analizar la seguridad de los LFSR. Se mostró que con 2L bits de una secuencia producida por un LFSR para cifrar información, ésta no se protege, pues se puede encontrar el polinomio con el cual se construye el SLFSR que genera la secuencia con que se cifró la información.

También se presentó el Generador Multivelocidad de Massey-Rueppel. Como el polinomio generado por el ABM no es un polinomio primitivo no se garantiza que pueda ofrecer un periodo máximo al implementarlo en un LFSR. Por tanto, al construir generadores más robustos, como el Generador Multivelocidad con estos polinomios que no cumplen con las propiedades de construcción de este generador en particular, no se podrá obtener una secuencia con periodo máximo.

Al usar polinomios primitivos y cumplir las propiedades del Generador Multivelocidad se pueden obtener secuencias con periodos máximos y buenas propiedades criptográficas. Además haciendo uso del lenguaje de descripción de Hardware como VHDL la construcción de generadores de secuencias pseudoaleatorias se vuelve una tarea sencilla.

Referencias

1. A. Menezes, P. Van, Oorschot, and S. Vanstone: Handbook of Applied Cryptography, CRC Press (1996).
2. A. Fuster, D. de la Guia, L. Hernandez, F. Montoya y J. Muñoz: Técnicas Criptográficas de protección de datos, 2a Edición, Alfaomega (2001).
3. Bruce Schneier, Applied Cryptography: Protocols Algorithms and Source Code in C, Second Edition, John Wiley & Sons, INC (1996).
4. C. Ding, G. Xiao, W. Shan: The Stability Theory of Stream Ciphers; Springer-Verlag Berlin Heidelberg (1991).
5. J. L. Massey y S. Serconek, Linear Complexity of Periodic Sequences: A General Theory, Advances in Cryptology CRYPTO'96, Lecture Notes in Computer Science No. 1109. New York, Springer, pp. 358-371 (1996).
6. Man Young Rhee: Error Correcting Coding Theory, Mc Graw-Hill Communications Series (1989).7. Rainer A. Rueppel: Analysis and Desing of Stream Ciphers, Springer-Verlag (1986).

8. Zeng, K. C.; Yang, C. H. y Rao, T. R.: On the Linear Consistency Test (LTC) in Crypto-analysis with Applications; Proc Crypto'89, Lecture Notes in Computer Science, 435,pp. 164-174 (1989).

Space qualified computer server developed for long life microsatellite applications

Vicente-Vivas E.[1], Mendieta J.F.J.[2], Calvillo T.A.[3], Tapia R.M.[4]

[1] Instituto de Ingeniería, UNAM, DF, México, evv@servidor.unam.mx
[2] CICESE, Ensenada BC., México, jmendiet@cicese.mx
[3] CITEDI, IPN, Tijuana BC., México, calvillo@citedi.mx
[4] CIMAT, Gto., Gto., México, max@fractal.cimat.mx

Abstract. Several research institutions from all over the country work together in the development of a 55 Kg low earth orbit (LEO) microsatellite aiming the development of in house technology in the space field, [1]. As known LEO operation implies a dephased orbital dynamics among satellite and our planet, achieving time limited communications either to download telemetry or to upload command and/or vehicle missions. By these reasons long life space computer architectures constitutes an important research field oriented to preserve communications among satellite and its control earth station (ES). Under this scenario few years of Mexican research efforts have been dedicated towards the development of a reconfigurable space qualified computer server (SQCS) which integrates cold spare redundancies in critical points of the architecture. This paper shows the SQCS hardware architecture specially developed for the Mexican Satex microsatellite and underlines the integrated features in the hardware to withstand the harsh space environment.
Keywords: Reconfigurable computers, computer architecture, flight computers and microsatellites.

Resumen. Varias instituciones de investigación Mexicanas colaboran en el desarrollo de un microsatélite de 55 Kgs, de órbita baja, con el objetivo de desarrollar tecnología nacional en el campo espacial, [1]. El desplazamiento de órbita baja conlleva un defasamiento entre la dinámica de nuestro planeta y la del satélite, lo cual implica restricciones en las comunicaciones ya sea para bajar telemetría, o bien, para enviar comandos o misiones al satélite. Por tales razones el tema de arquitecturas de cómputo espacial de larga vida constituye un importante campo de investigación que persigue preservar o mejorar las comunicaciones satélite-estación terrena. Ante este escenario, se han dedicado varios años de investigación en México para desarrollar un servidor de cómputo reconfigurable y de calificación espacial (SCCE) que cuenta con redundancias de refacción en frío en puntos críticos de su arquitectura. Este trabajo describe la arquitectura del SCCE que fue diseñada especialmente para el microsatélite Mexicano Satex; de igual forma detalla sus características de hardware que le permiten operar en el medio espacial que se caracteriza por ser un ambiente altamente agresivo para equipos electrónicos.

Palabras llave: Computadoras reconfigurables, arquitectura de computadoras, computadoras de vuelo y microsatélites.

1 Introduction

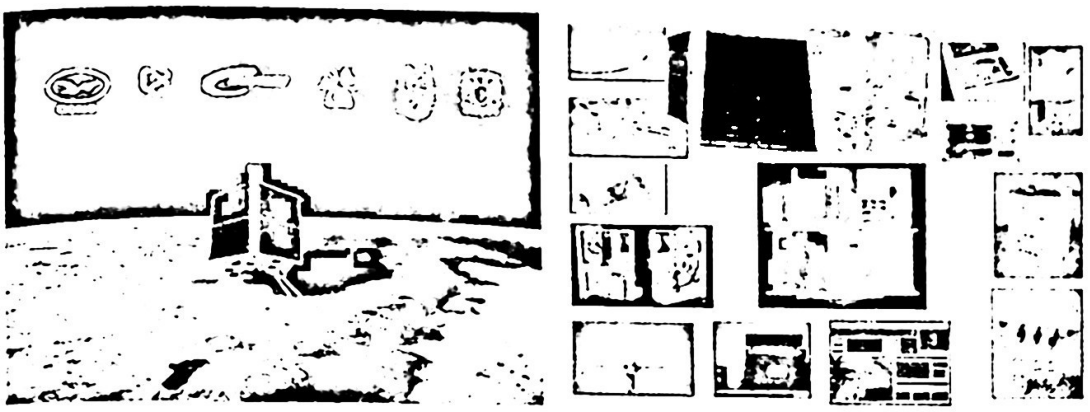
Electronic equipment developed for space applications needs to accomplish a set of fundamental requirements in order to withstand the harsh space environment in terms of vibration during the launching phase as well as in terms of radiation, extreme temperatures, and vacuum during orbital operation. For this purpose space projects have employed approaches both for failure avoidance as well as for fault tolerance, [2]. The former includes the selection of qualified components, enforcing design rules and the periodical review of designs. The last handle hardware failures and software errors when they occur, through the help of redundant hardware, plus fault diagnosis, fault detection and reconfiguration techniques, [3].

Electronic equipment for space applications, and specially those projected for small space vehicles have strong limitations in terms of weight, volume, and electrical power. However, the small satellite field is also characterized by the adoption of "faster, cheaper and probably better" approaches to easy the access to space. In this sense the use of commercial-off-the-shelf (COTS) components is becoming a common practice, [4], which in turn has enabled the launching of state-of-the-art electronics into LEO orbits. This picture is represented by the successful missions developed by the University of Surrey, UK, a worldwide recognized institution both in the small satellite field as well as in the use of COTS components in space platforms, [5].

In addition, as the budget for worldwide space projects is clearly diminishing some important space institutions, such as the Jet Propulsion Laboratory from NASA have started research efforts towards reusable avionics computer architectures to be used in multiple missions, aiming to reduce the development and production costs of flight projects [6].
On the other hand, the Satex Mexican project, figure 1.a, aims the integration of microsatellite generic subsystems, figure 1.b, with capabilities to adapt them to future missions. In this sense the project demanded the development of a redundant and reusable flight computer with capabilities to apply maintenance to it after the detection of failures. It is important to highlight that all over the world very few microsatellite missions make use of redundant flight computers [7] and [8]. In other words, must of them employ centralized computer architectures to automate operations and payloads [9] and [10].

Details about Satex microsatellite subsystems, figure 1.b, are given in [1]. The project contains five payloads, few of them with dedicated control requirements. This goal demanded the development of a satellite distributed control system where satellite operations are governed by the flight computer and some payloads contain a dedicated microcomputer for dedicated control purposes. In addition communications inside the vehicle are accomplished by a fault tolerant redundant local area network

(FTRLAN) that achieves safe communications through real time accomplishment of fault diagnosis, fault detection and fault reconfiguration processes, [11].



(a) (b)

Fig. 1. (a) Satex Microsatellite project. (b) Subsystems already available for the Mexican Microsatellite.

The operations delegated to the SQCS were the followings:

- Starting operations for the satellite after liberation in space.
- Telemetry acquisition and telemetry packaging.
- Communications with the Earth Station.
- Computer server functions among satellite and ES.
- Communications with payload microcomputers.
- First stabilization stage to allow gravity gradient deployment.
- Second stabilization stage to allow payload pointing to Earth.

As seen from the previous list the SQCS represents a single point of failure whose malfunction leads to a failure of the whole satellite system. To overcome this harmful effect a reconfigurable SQCS architecture was proposed, [12], and developed. The design integrates up to three single board microcomputers (SBM) each one with enough hardware capabilities to fulfil the satellite instrumentation requirements. SBM full characteristics are given in a later paragraph.

In order to generate a cost effective design it was proposed for the architecture to use three SBM with identical printed circuit boards (PCB). This lead to the design of a single PCB which employs jumpers to program the SBM identity as well as to provide separate energization paths for three different processor configurations (main, first backup and second backup).

The SQCS architecture also demanded the design of a compact digital switching unit to cross strap any one of the SBM to the satellite instrumentation. Besides, quad digital switches were employed to provide energization for every SBM. By these means SQCS reconfiguration (maintenance) can be commanded from external sources, in this case from any microcomputer payload controlled either in automated

fashion or remotely operated from the ES. This topology also allowed the definition of fault containment regions in the SQCS architecture, composed by every SBM to avoid damage propagation when failures take place. In [13] full details are given about SQCS evaluation.
In addition, the following interface requirements were settled for the SQCS:

- A multiplexing, conditioning and filtering module (MCFM) to allow the acquisition of up to 48 signals from satellite sensors (fine sun sensors, magnetometers, current, voltage, light presence, etcetera).
- Electronics for both main and redundant LANs.
- Connectors to interface the SQCS I/O signals to the satellite instrumentation as well as the required line drivers for signal coupling.

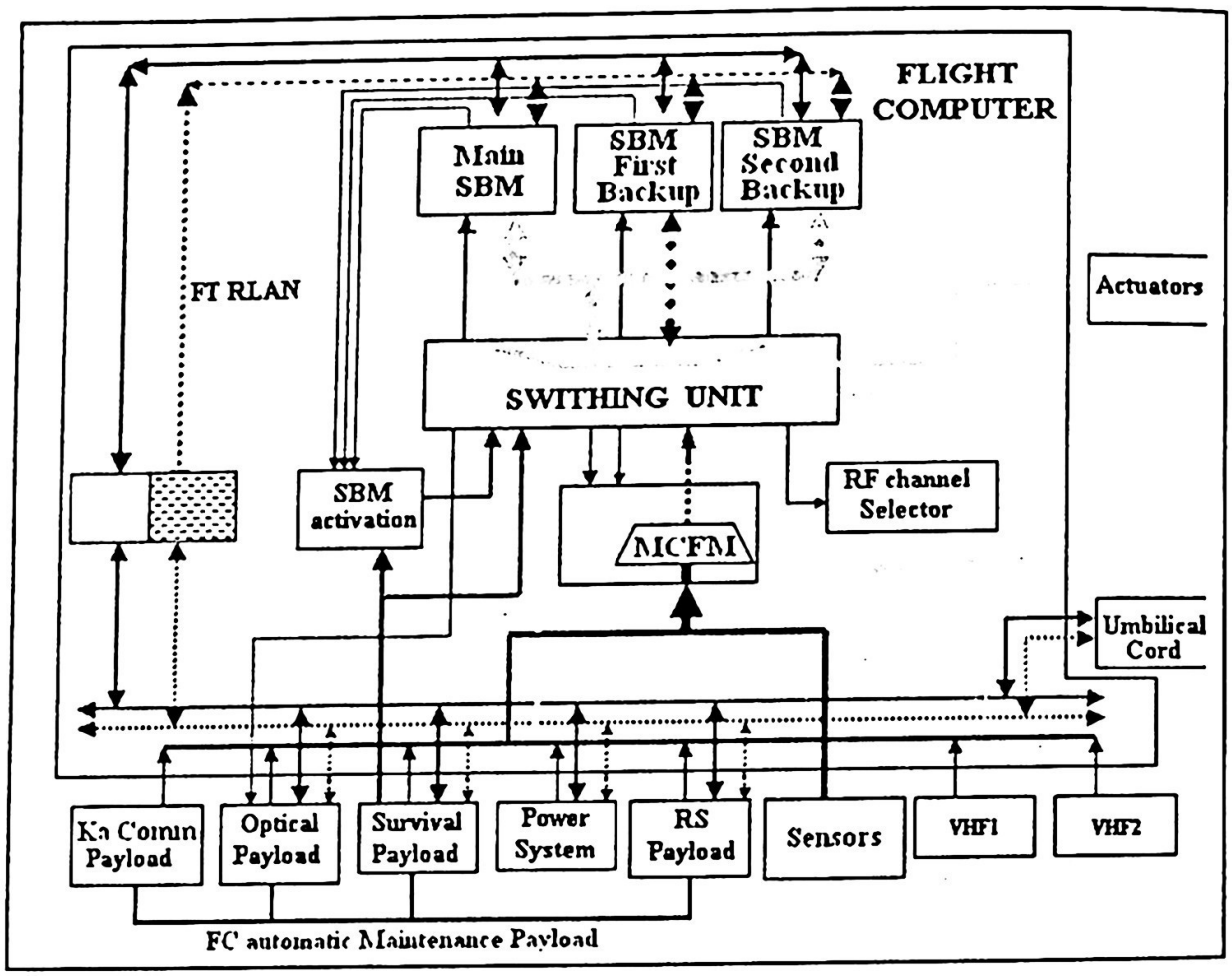


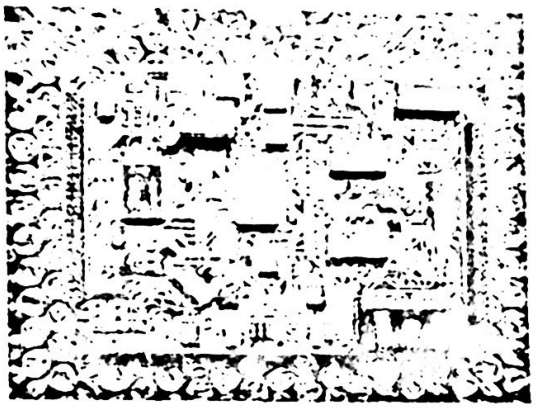
Fig. 2. Flight computer architecture developed for Satex microsatellite.

The hardware architecture developed for the Satex mission's computer server is shown in figure 2. As can be seen the microsatellite contains five payloads, they will treated with details in other publications.

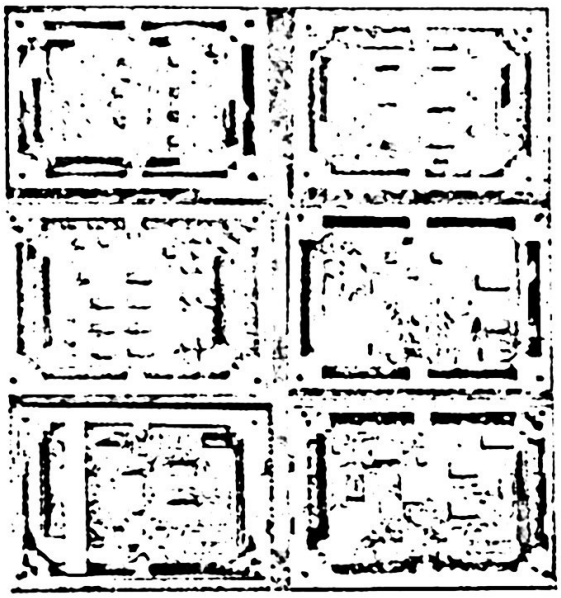
2 Single Board Microcomputers

In order to increase satellite availability the SQCS architecture was conceived and developed to admit automatic maintenance through SBM reconfiguration facilities. However strong limitations in weight, volume and available power forced the automatic maintenance process to take advantage of onboard microcomputers to implant an N modular redundancy computer architecture. This architecture along with utilized schemes for fault diagnosis, fault detection and automatic maintenance are described in [3].

By these reasons the SQCS can be assembled with up to three SBM, each one, figure 3.a, capable of being activated through digital signals coming from an external source.



(a)



(b)

Fig. 3. (a) SBM designed and fabricated for the space qualified computer server. (b) Computer server boards attached to aluminium containers.

Under this scope, one SBM is activated at a time and remaining SBMs comprise cold standby spares that can be employed when failures are detected at main SBM. The exposed scheme enhances the relation among availability and redundancies projecting a computer with power consumption demands very likely to that of a simplex system. Each SBM contains:

- A 40 Mhz 16 bit RISC microcontroller (RISCM) from Siemens with 76 I/O lines, 16-priority-level interrupt controller, ten-channel 10-bit A/D converter, seven 16-bit timers and a programmable watchdog timer.
- 64 Kb PROM containing the basic satellite software,
- 1.2 Mb SRAM for data management, for upgraded software uploading, as well as for image storage,
- EDAC protection for whole static RAM in order to prevent single-event-upsets,
- 1.2 Mb SRAM for EDAC syndrome storage,
- Three serial channels, one assigned for communications with ES and the two other for FTRLAN operation,
- Local temperature sensor with associated conditioning electronics,
- Lateral connectors that allow the interconnection of up to three identical SBM in a piggyback manner, each one rotated 180 degrees against each other. In this case every SBM is jumper programmed for identification purposes as well as to provide independent activation.

With the purpose of saving weight and space in the SQCS the PCBs were designed to hold electronic components on both of its faces. This was achieved with the use of surface mount technology for both active and passive electronic parts, and avoiding the use of electrolytic capacitors which are not recommended for use in space.

The replacement of SBMs offers redundancy support for the RISCM and its external peripherals, such as: program memory, data memory, I/O lines, A/D converter, network ports, timers, interrupt controller, oscillator, etc. In addition the basic satellite software stored in PROM is also replaced whenever a SBM reconfiguration takes place. In this way, the substitution of SBMs allows the extension of life for the flight computer and consequently the satellite availability as well as its useful life when permanent faults occur.

3 Fault protections for single board microcomputers

The semiconductor chips from microsatellites are exposed to space radiation, whose effects are usually a function of the total radiation dose seen by each component. The radiation is originated by anyone of the following phenomena:

- Particles associated with magnetic planetary fields, as in the case of Van Allen rings that affect satellites orbiting the Earth.
- Cosmic rays from deep space, such as Gamma rays, X rays, etc.,
- High energy protons generated during solar explosions.

Charged particles can pass through the electronic devices and generate a cloud of electrical charge. This charge can induce single-event-upsets (SEU) that may generate serious consequences to the integrated circuit, i.e., the lost of stored bits from important software variables. The permanent exposition to radiation generates charge accumulation, which modify the semiconductors properties, i.e., transistor beta goes down, threshold voltages change, leakage currents increase, and so on. This means that digital logic slows down, operational amplifiers offset voltages change, power dissipation goes up, etc. The undesirable effect of radiation is measured in rads; in general, a rad is the quantity of any type of ionizing radiation that adds 100 ergs from energy to a gram of material, [14]. Some times SEUs will be followed by a phenomenon called latch-up characterized by very large currents flowing in the device. The large currents will usually destroy the electronic part within a few tens of milliseconds. Although SEU events can occur in various integrated circuits technologies, such as CMOS and TTL bipolar devices, latch-up phenomenon has almost exclusively been observed in CMOS, [15] and [16].

4 Protections against SEU events

SRAM from SQCS will always be exposed to radiation and consequently will present random data errors, in average 1 every 15 hours according with experimental data reported by [20] and [17] when using COTS memory with EDAC protection. In addition published reports show that errors are more often generated in the South America area where relatively much radiation was indirectly detected, [18].
To counteract error generation in memories each SBM integrates a 100 pins military qualified 16-bit flow-trough EDAC device placed between the microcontroller and SRAM memory. The 29C516E EDAC allows the monitoring and correction of erroneous data values coming from data memory. It detects and corrects 100% of all single-bit errors and detects all double bit-errors. However, EDAC utilization introduces a penalty because additional syndrome SRAM has to be included of the same capacity as that from the memory to be protected. In our case 1.2 Mb of SRAM were added for this purpose.

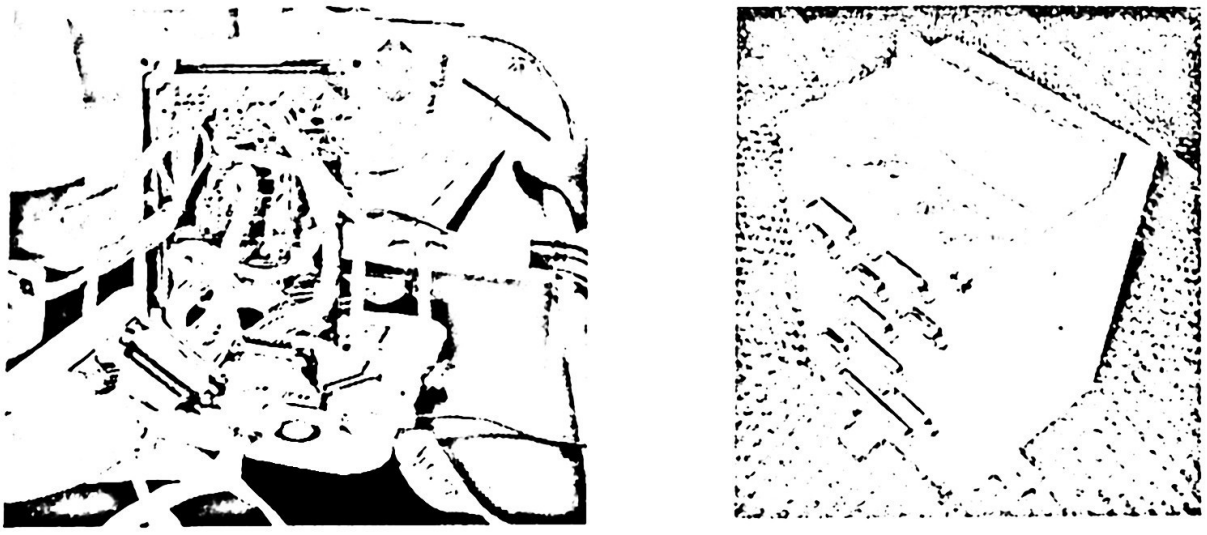
In addition, dedicated software is required for "memory wash" purposes which takes place every ten minutes when the satellite is operative. In this way when single errors are detected and corrected an interrupt request is sent to the RISCM in order to asses the number of errors. This information is attached to telemetry to notify the ES personnel about the operative state of memory. Besides, when multiple errors are detected the hardware automatically forces the SQCS to be temporally switched off taking into consideration that software status can be seriously damaged. In addition as it is considered that multiple errors might be related with the start of a latch-up phenomenon, the only safe process to overcome a latch-up event is to remove energy from the affected component [14], however for safety reasons the whole SQCS is switched off.

5 Latch-up protection

The integrated circuits employed at every SBM, excluding the RISCM, are all military qualified components (MQC) according with MIL-STD-883. The MQCs contain electronic protection to avoid the latch-up phenomenon. However COTS parts as the RISCM are not latch-up free. To overcome this risk two different types of protections were implemented. The first one consists of an EDAC based latch-up protection for SBMs, which was described in above paragraph. The second one is a dedicated protection for the RISCM integrated at each SBM. This is a sensor that permanently measures the microcontroller current looking for measurements whose value falls above from the nominal current driven by the microcontroller. When this event is logged an electrical pulse of 16 seconds is generated to automatically interrupt power application to the whole SQCS. After the programmed time gets over energy is applied again to the SQCS. However if the latch-up process resumes its operation the sensor protection works in iterative fashion.

6 Other protections

In addition, COTS critical components from SQCS will be protected for space flight with thin sheet shields either from Tungsten or Tantalum, as recommended by [19] and [15] preventing charged particles from interacting with those devices [4].



(a) (b)

Fig. 4. (a) Computer server boards under laboratory testing. (b) Space qualified computer server developed for Satex microsatellite.

7 Computer server validation

The space qualified computer server was integrated and validated in several working stages. At the beginning operative validation was performed at board level. This process took to the installation of every single board in to space qualified aluminium containers built at university workshops, figure 3.b. Afterwards the validation among adjacent boards took place, figure 4.a. Finally the computer server, figure 4.b and figure 5, was validated and evaluated with the help of special tools developed for the project as described in [13].

8 Concluding Remarks

The space qualified architecture for a computer server specially developed for the Satex microsatellite has been depicted. The architecture takes in to account the strong limitations associated with small space vehicles in terms of weight, volume and available power. It also considers electronic and physical protections to cope the harsh environment expected at altitudes of 800 Km, in particular that related with the effects of radiation in electronic parts.

The SQCS consists of a maximum of three single board microcomputers (main and cold spares), each one with enough resources to fully control the satellite operations. In addition three separate boards contain an electronic switching unit that allows any one of the SBM to be energized and connected to the satellite instrumentation; a multiplexing and conditioning unit to read up to 48 electronic signals from satellite sensors; hardware for a redundant LAN employed to interconnect the satellite microcomputers; as well as protection against SEUs and latch-up phenomenon.

The flight model for the SQCS has been fully tested. Details about this as well as information related with the special tools developed in the project for testing purposes can be found in [13].



Fig. 5. Space qualified server under validation with the help of special tools developed for the project.

9 Acknowledgements

The authors would like to acknowledge to federal agencies from México, TELECOM and COFETEL who supported the work described in this paper. Besides, acknowledges are given to the Mexican Research Institutions which collaborate in the development of this project, in alphabetic order: CICESE, CIMAT, CITEDI, INAOE, IPN and UNAM.

References

1. Vicente-Vivas E., Roch-Soto J. y Mendieta-Jiménez J., "La convergencia de un Proyecto multi-institucional para el desarrollo de un microsatélite con tecnología Mexicana", Revista INGENIERÍA Investigación y Tecnología, ISSN 1405-7743, pp. 157-168, Vol. III - No. 4 - octubre-diciembre 2002.
2. Johnson B. W., "Design and Analysis of Fault Tolerant Systems, Addison Wesley, 1989.
3. Vicente Vivas Esaú, "Arquitectura de Cómputo Semivirtual Tolerante a Fallas con Capacidad de Mantenimiento Automático Aplicada a un Microsatélite de Órbita Baja", Tesis Doctoral en Ingeniería Eléctrica, DEPFI, UNAM, 21 de Mayo de 2004, México, DF.
4. AstroExpo.com e-newsletter, "COTS in space, Space Electronics and Processors", March 2004, http://www. astroexpo. com/news/articlesdetail.asp?ID=233.
5. Day M., "30 Years of Commercial Components In Space: Selection Techniques Without Formal Qualification, 13th Annual AIAA/USU Conference on Small Satellites, Utah, USA, 1999.

6. Chau S., K. Reh, B. Cox – NASA/JPL, J. Barfield, W. Lockhart, M. McLelland - SouthWest Research Inst. "A Multi-Mission Space Avionics Architecture", AstroExpo.com e-newsletter, March, 2004, http://www.astroexpo.com/reference/techpapersdetail.asp?ID=229.
7. Frank Sperber, "Amsat-Phase 3-D a 400 Kgs International Communication and Experimental Satellite in a High-Elliptical Orbit", 3rd Intl Symposium on Small Satellite Systems and Services, Annecy, France, 1996.
8. Allery M., Sweeting M., and Ward J., "University of Surrey Small Satellite Systems: In orbit capabilities and development programme", 1997.
9. Robert Zee y Peter Stibrany, "Canada's First Microsatellite – An Enabling Low- Cost Technology for Future Space Science and Technology Missions", 11th CASI Conference on Astronautics, November, 2000.
10. Christopher Kitts, et al., "Experiments in Distributed Microsatellite Space Systems", Proceedings of AIAAA Space Technology Conference and Exhibit, Albuquerque, NM, USA, Sept., 1999.
11. Vicente Vivas Esaú, "Red de Área Local Tolerante a Fallas Aplicada a Sistemas de Tiempo Real con Altos Requisitos de Confiabilidad", Revista Científica de la ESIME, Num.15, pp. 13-24, Mayo-Junio de 1999.
12. Vicente Vivas E. et al., "Fault-Detection and Reconfiguration Capabilities for the Distributed Computer Architecture on board the Satex Microsatellite", Revista: Seminars of the United Nations Programme on Space Applications, Selected papers on Space Education, Remote Sensing and Small satellites, No.8, pp. 176-182, United Nations, Vienna, 1997.
13 Vicente Vivas Esaú et all., "Evaluation of a Space Qualified Long Life Flight Computer Server", 5th International Conference on Control, Virtual Instrumentation, and Digital Systems CICINDI 2004, México, DF, September 2004.
14. Vincent L. Pisacane and Robert C. Moore, The Johns Hopkins University Applied Physics Laboratory, "Fundamentals of Space Systems", Oxford University Press, 1994.
15. Dino B. Milanil, John Pokoski, "Design of a Low-Cost Single Board Computer System for use in Low-Earth Orbit Small Satellite Missions", Proceedings 11 AAIA/USU Small Satellite Conference, Utah, August 1997.
16. Ash M., and Messenger G., "The effects of radiation on electronic systems", Van Nostrand Reinhold, 1992.
17. Laurence J.F., et all., " A General-Purpose MIL-STD-1750A Spacecraft Computer System", American Institute of Aeronautics and Astronautics, 1993.
18. Impyeong Lee et al., "Experimental Multimission Microsatellites-Kitsat Series", Proceedings of 6th Annual USU/AIAA Conference on Small Satellites, Logan, Utah, September, 1992.
19. Winokur P.S., Fleetwood D.M., and Sexton F.W., "Radiation-Hardened Microelectronics for Space Applications", Radiation Phys. Chem. Vol 43., No. ½., 1994, Great Britain.
20. Shaneyfelt M.R, Winokur P.S., Sexton F.W., Roeske S.B. and Knoll M.G., Hardness Variability in Commercial Technologies", IEEE Transactions on Nuclear Science, Vol. 41, No. 6, December 1994.

Evaluation of a space qualified long life flight computer server

Vicente-Vivas E.[1], Mendieta J.F.J.[2], Calvillo T.A.[3], Tapia R.M.[4]

[1] Instituto de Ingeniería, UNAM, DF, México, evv@servidor.unam.mx
[2] CICESE, Ensenada BC., México, jmendiet@cicese.mx
[3] CITEDI, IPN, Tijuana BC., México, calvillo@citedi.mx
[4] CIMAT, Gto., Gto., México, max@fractal.cimat.mx

Abstract. The first ever fully designed Mexican microsatellite is to be assembled and tested by several federal research institutions from the country. Some satellite subsystems are already done, few other are about to get finished and a couple of them are still under development stage. One of the already available subsystems is the space qualified computer server (SQCS) along with its operations software for both the space and ground segments, as well as associated tools that enabled the full subsystem validation regardless the physical presence of other complementary satellite subsystems. The single points of failure from the SQCS architecture were developed with cold standby redundant modules to improve hardware reliability. In addition reconfiguration capabilities commanded by digital signals were also considered, thereby allowing the extension of satellite life in the presence of important SQCS failures. Under this scenario, this paper outlines the hardware and software tools specially created for SQCS validation purposes, as well as results of a reliability study to predict the server behaviour based on the exponential failure law, the military standard MIL-HDBK217f notice 2 and MatLab software.
Key words: computer architecture, flight computer, hardware reliability, computer evaluation, Markov models, microsatellite.

Resumen. El primer microsatélite completamente diseñado en México está por ensamblarse y certificarse con la participación de diversos centros de investigación nacionales. Algunos de sus subsistemas ya se encuentran concluidos, otros están a punto de terminarse y solo un par de ellos se encuentran aun bajo desarrollo. Uno de los subsistemas terminado es el servidor de cómputo de calificación espacial (SCCE) junto con su software operativo tanto del segmento espacial como del segmento terrestre, así como sus herramientas que permitieron validar operaciones del subsistema sin necesidad de contar con la presencia física de los subsistemas satelitales con los que interactúa. Los puntos más vulnerables del SCCE fueron diseñados con partes de refacción para mejorar su confiabilidad operativa. Adicionalmente se le agregaron capacidades de reconfiguración por medio de señales digitales, para permitir de esta forma extender la vida útil del satélite ante fallas del SCCE. Ante este panorama este trabajo describe las herramientas desarrolladas en hardware y software para propósitos de validación del

SCCE, así como los resultados de un estudio de confiabilidad que predicen el comportamiento del servidor de cómputo con base en la ley exponencial de fallas, la norma militar MIL-HDBK217f y software escrito en Matlab.
Palabras clave: arquitectura de cómputo, computadora de vuelo, confiabilidad de hardware, evaluación de computadora, modelos de Markov, microsatélite.

1 Introduction

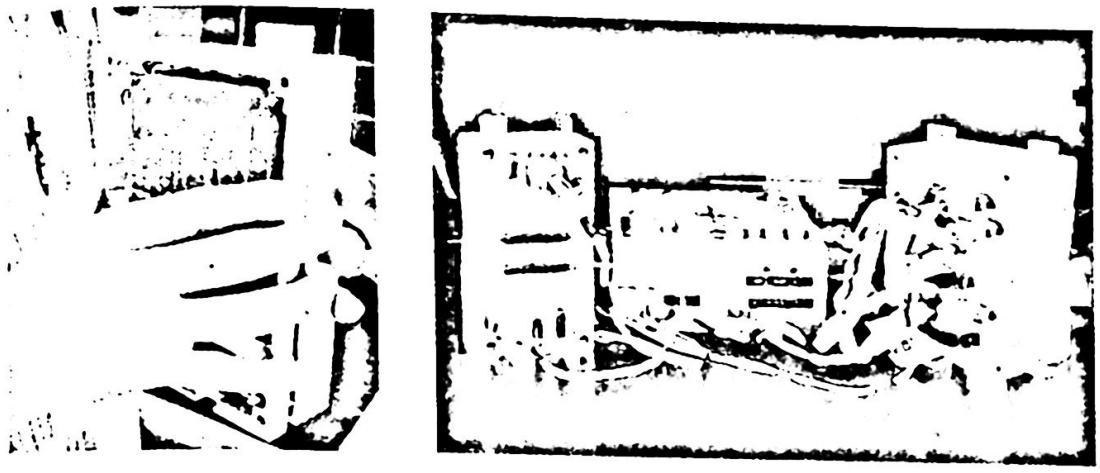
Satex microsatellite project is a multiinstitutional Mexican effort to develop domestic technology in the space field. The microsatellite, [1], will weight about 55 Kgs, with cubic shape, solar panels on four of its faces, five experiments, and 5 microcomputers interconnected through a redundant LAN. The payloads will consist of an optical communications downlink, a Ka band communications uplink, a minimal survival system to control satellite elementary operations, a digital camera for remote sensing studies and a computer architecture experiment planned to apply automatic maintenance to the SQCS, the most critical microsatellite computer, [2].

The microsatellite project constitutes an excellent opportunity for hands on training of human resources from all over the country in order to generate a strategic infrastructure considering the feasibility to develop more sophisticated satellites in the medium term. Moreover, assuming continuous activities in the area and considering some kind of successes in the medium term, possibilities of commercial implications exists for products and technology developed under the project. Therefore the space subsystems fabricated for Satex were designed and built according with the strong limitations of space vehicles in terms of weight, volume, and electrical power. In addition, hardware reliability enhancement through the use of redundancies in critical parts of the instrumentation was envisioned. In particular, the SQCS was designed and constructed with reconfiguration capabilities and cold standby spares to withstand important hardware failures. Besides considering that the use of commercial-off-the-shelf (COTS) components is becoming an important practice to launch state-of-the-art electronics into LEO orbits, the Satex mission also considered the viability to experiment the utilization of COTS parts protected for space flight.

From the beginning of the project steps were traced to develop a reusable fault-tolerant flight computer server and use it as the automation backbone for the space vehicle, [3]. A complimentary goal was to build up a reusable SQCS to reduce the development and production costs for future space projects.

In this way the SQCS was developed with: a reconfigurable architecture that contains up to three single board microcomputers (SBC); one digital switching unit that enables any one of the SBC to be connected to the satellite instrumentation; multiplexing unit to capture 48 electrical signal from satellite sensors; hardware for redundant LAN that combined with dedicated software admits fault-tolerant communications among satellite computers, [4]; latch-up protection for COTS parts

as well as I/O connectors. By financial reasons the Satex's SQCS was integrated with just two SBC into a space qualified aluminium cabinet which integrates five printed circuit boards, each one with electronic components on both of its faces and mostly of them of surface mount type, figure 1.a. Information regarding SQCS design and protections for use in space environment can be found in [5].



(a) (b)

Figure 1. Space qualified Computer Server: (a) Stackable printed circuit boards; (b) Flight model under testing with CESSM module at the left, satellite simulator at the centre and SQCS at the right.

2 Hardware and software tools developed for SQCS validation

To allow the validation of SQCS hardware as well as its progressive software development must of the onboard satellite equipments were needed in continuous working stages. However the different satellite subsystems were all together grown in parallel fashion and therefore were unable for combined testing purposes. By this reason, since the starting of the project provisions were made to develop hardware and software tools for SQCS validation purposes. The following hardware and software tools were developed throughout the last years: a satellite simulator (SIMSAT), an emulation software for satellite payloads called SOFDEVO and the earth station software (ESS). These tools allowed the progressive development of both SQCS hardware and software.

3 Satellite simulator

The satellite simulator shown in figure 1.b was constructed to allow the SQCS to interact with satellite peripherals such as sensors, actuators, payloads and other subsystems. The SIMSAT contains: 48 passive sensors to provide real telemetry collection; electronic representation for UHF radios and magnetic coils; gravity gradient release emulation and 10 I/O connectors of different size for interconnection

with SQCS. Furthermore contains several analog switches to manually control the SQCS reconfiguration as well as push buttons to command the execution of programs either from PROM or from SRAM memory. In addition SIMSAT allocates LAN ports to attach up to 4 personal computers, each one running the software SOFDEVO, for payload emulation purposes and consequently for SQCS operations validation.

4 SOFDEVO software

SOFDEVO allows the depuration, emulation and operative validation of software for all or any one of the satellite computers, figure 2.a. In consequence, the software allowed the depuration and operative validation of SQCS hardware and software. For this purpose SOFDEVO was planned to grow in parallel with all other satellite subsystems. Its operation is based on the fact that all computers on board the space vehicle are connected through a redundant LAN where communications traffic generated by computing nodes is found. In this way SOFDEVO runs dedicated software for every emulated payload, establishing therefore the payload operating behaviour. The software manages network protocols, fault-tolerant network attributes [4], and also contains all kind of programmed answers for network queries related the payload. However, some software features are allowed to be programmed advance by the user through the help of menus to generate automated answers according with specific needs. Whenever the last process is required it has to be programmed before starting an emulation process in such a way that SOFDEVO real time performance is achieved.

Moreover SOFDEVO contains two data windows, one shows decoded messages collected from the network, allowing the user to be informed about real time satellite network operations. The second window shows the answers generated by SOFDEVO when network enquires are directed to anyone of the emulated payloads.

On the other hand SOFDEVO allows the programming of fault insertion processes which are emulated in combination with LAN protocols whenever payload diagnostic results are generated in order to validate satellite software under the presence failures. By these reasons the software was invaluable for SQCS hardware validation as well as for SQCS software developing process.

5 Earth station software

The earth station software (ESS) was also created in parallel fashion along with other satellite subsystems according with needs and characteristics from the satellite flight software. This approach allowed the design and the implementation of data frames, commands and communications protocols among satellite and ESS. From the beginning of the project the initial progresses in satellite automation software enabled the validation of operations among satellite instrumentation and ESS. In this way the

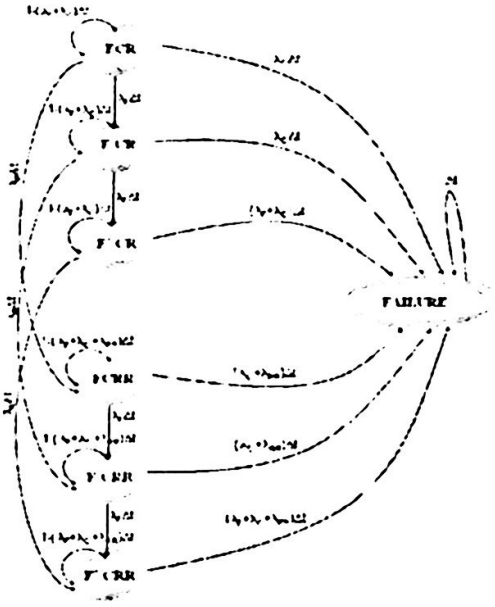
continuous verification of automatic processes took to the progressive development of the whole SQCS hardware and software.

Moreover the project offered enough available time to generate three updated versions of ESS from which the last one contains a highly visual environment where satellite telemetry is exhibited with digital animations and virtual instruments, figure 2.a. In addition the ESS contains a set of menus from which following actions can be controlled:

- Communication speed among satellite and earth station.
- Automatic satellite search.
- Programming of alarm thresholds for anyone of the satellite sensors.
- Transmission of new programs to the SQCS.
- Definition of a satellite telemetry mission (Normal or Special type).
- Telemetry downloading.
- Statistics of initialization procedures performed by the SQCS.
- Telemetry displaying with virtual instruments for up to 64 satellite sensors, 48 of them obtained by the SQCS.
- Animated displaying of satellite equipment status.
- Numerical telemetry displaying.
- Image acquisition programming.
- Image downloading and fast local view.



(a)



(b)

Figure 2. (a) SQCS validation in laboratory; SOFDEVO software at the left and ESS software at the right. (b) Markov model describing SQCS operating states.

6 SQCS validation

The SQCS hardware was progressively validated in several development stages. At the commencement of the project must of the electronic designs were validated in laboratory with the help of breadboards and COTS components. While waiting the arrival of military qualified components some PCBs were designed and sent for production. Later the PCBs were carefully checked and then parts assembling took place with the application of continuous testing to guarantee PCB operation. The overall process conducted to the identification and elimination of errors as well as to the development of PCB updated versions for every one of the five electronic cards that comprise the SQCS architecture. At the same time preliminary software testing procedures were achieved through the direct connection of a single SBC either to SOFDEVO or to ESS. Later on testing was carried out among ESS, SBC and SOFDEVO. The progressive development of referred subsystems took to the validation approach seen in figure 1.b and figure 2.a. Where satellite sensors emulation was replaced with the module called "conditioning electronics for satellite sensors and maintenance" (CESSM). By these means the SQCS was continuously validated with the CESSM flight model and updated versions of ESS and SOFDEVO. Successfully testing involved following actions:

- Transmission of new programs from ESS to the SQCS.
- Programming and transmission of satellite missions at ESS.
- Execution of either new programs or new missions at SQCS with SIMSA and SOFDEVO support.
- Telemetry transmission from SQCS to the ESS.
- Telemetry downloading and consequently telemetry displaying at ESS.

7 Reliability study of SQCS hardware

In order to know the reliability gains introduced by the use of cold standby SBC's and redundant LAN under space environment an analysis based on the exponential failure law, the military standard MIL-HBK217f notice 2 and Markov models was elaborated.

To obtain the failure rates for each one of the electronic components contained in SQCS the exponential failure law was employed as well as the calculation procedures for failure rates suggested by the military norm "MIL-HDBK-217f Notice 2", see table 1.a. Full information about the procedures to calculate reliability rates for any kind of electronic component was directly obtained from the Department of Defence of the USA.

(a)

Description	Acronym	Failure rate
16 bit RISCM	SAB80	8.36297E-07
16 bit EDAC	EDAC	9.12533E-08
SRAM memory (128K x 8)	RAM128	1.01393E-07
SRAM memory (512K x 8)	RAM512	3.11022E-07
EPROM memory (32K por 8)	PROM	1.30852E-08
Comparator LM139FK	COMP	3.08985E-08
Nand gate SN54HC00FK	NAND	1.0771E-08
Or gate SN54HC32FK	OR	1.0771E-08
And gate SN54HC08FK	AND	1.0771E-08
Operational amplifier TLC2201FK	AMP2201	3.54674E-08
Operational amplifier TLC271FK	AMP271	3.64674E-08
Digital multiplexer SN54HC157FK	MUXDIG	1.1571E-08
Line driver SN55189FK	LD189	3.86674E-08
Line driver SN55188FK	LD188	3.86674E-08
Analog multiplexer CD4053BM	MUX4053	2.52931E-08
Monostable MM54HC221A	MONO	3.78951E-08
40MHz oscilator	XTAL	0.000000015
Hexfet (MOSFET) IRFF130	IRFF	1.16E-08

(b)

Mission	Sucess Reliability
1 year	0.99
3 years	0.97
5 years	0.95
10 years	0.9

Table 1. (a) Calculated failure rates for SQCSs' electronic components. (b) Reliability values given by Hoffman for electronic space equipment

Once the failure rates were obtained, the SQCS was divided into modules: SBCs, switching unit, and so on. Afterwards each module was analyzed to check for serial or parallel arrangements. In this case, every module was represented with a failure rate whose value depends on the determination of the failure rates for every one of the grouped electronic components. Particularly, the failure rate was obtained by applying combinatorial techniques and the exponential failure law [6]. Then, in association with components shown in table 1.a the following failure rates for SQCS modules were obtained:

$$\text{lamda_2_MONO_en_P} = -\log(1-(1-\exp(-(1/2)*\text{MONO}))^2)\,.$$

$$\text{lamda_2_HEXFET_en_P} = -\log(1-(1-\exp(-\text{IRFF}))^2)\,.$$

$$\text{lamda_2_MUX4053_en_P} = -\log(1-(1-\exp(-1/3)*\text{MUX4053}))^2)\,.$$

$$R = \text{LD189} + \text{AND} + (3/4)*\text{AND} + \text{LD188} + (1/4)*\text{OR} + (1/4)*\text{AND}\,.$$

$$RR = \text{LD189} + \text{AND} + (1/2)*\text{NAND} + (1/4)*\text{AND} + \text{LD188} + (1/4)*\text{OR}\,.$$

$$C = (1/4)*\text{MUXDIG} + (2/3)*\text{LD188} + (3/4)*\text{MUXDIG} + (2/3)*\text{MUX4053} + \text{MUXDIG} + (2/3)*\text{MUX4053} + (1/3)*\text{MUX4053} + 2*\text{lamda_2_MUX4053_en_P} + (2/3)*\text{MUX4053} + (1/4)*\text{OR} + (1/2)*\text{LD188} + 2*\text{lamda_2_HEXFET_en_P} + 2*\text{lamda_2_HEXFET_en_P} + (1/4)*\text{OR} + (1/4)*\text{AND} + \text{AND} + (1/2)*\text{OR} + \text{MUX4053}\,.$$

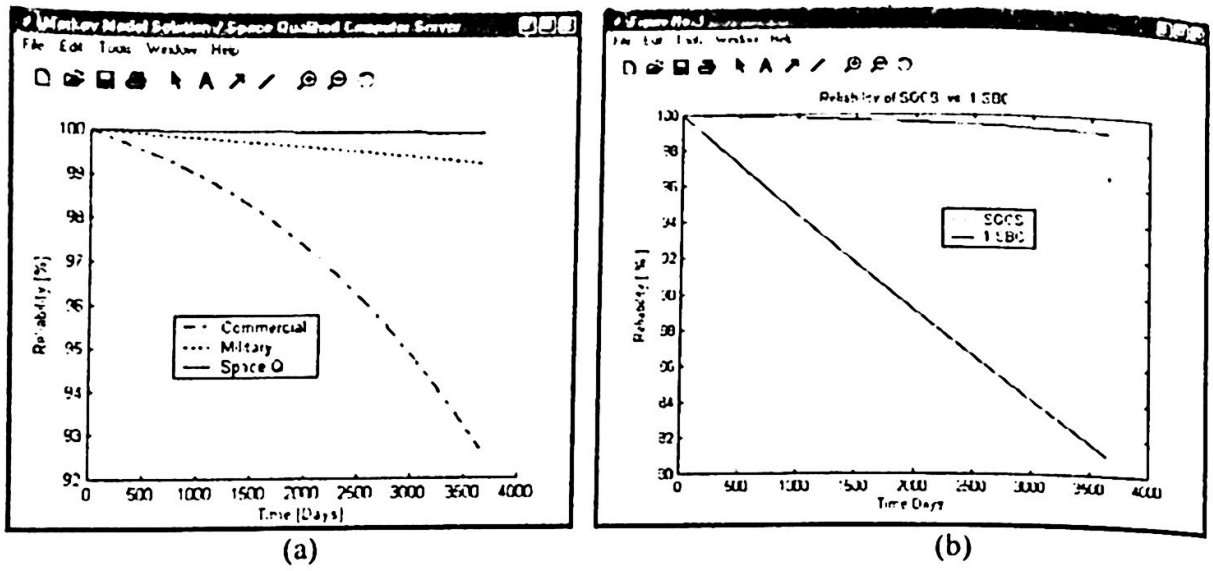
$$F = \text{XTAL} + \text{SAB80} + 2*\text{PROM} + 2*\text{RAM128} + 2*\text{RAM512} + \text{RAM512} + \text{RAM128} + \text{EDAC} + (1/2)*\text{AND} + (3/4)*\text{OR} + (1/2)*\text{NAND} + (1/2)*\text{OR} + \text{NAND} + (3/4)*\text{OR} + (3/4)*\text{AND} + \text{AMP2201} + \text{AMP2201} + \text{AMP271} + (1/4)*\text{LM139} + \text{LM136} + (1/4)*\text{OR} + (1/4)*\text{OR} + \text{lamda_2_MONO_en_P} + (1/4)*\text{OR} + (1/4)*\text{OR} + (1/4)*\text{LD188} + 2*\text{lamda_2_HEXFET_en_P}\,.$$

From these modules the participation of spares was analysed to generate the Markov model for SQCS architecture, where λ_F stands for SBC failure rate, λC is the failure rate for the switching unit, λ_R is the failure rate for main LAN, and λ_{RR} is the failure rate for redundant LAN, figure 2.b. In this figure the state titled as FCR denotes that SQCS operates with main SBC symbolised with letter F (from three alternatives, F, F′ and F′′), the switching unit designated with letter C, and main LAN indicated with letter R (from two alternatives, R and RR). Once generated the Markov probabilistic model, the following Markov equations were generated.

$$p_{FCR}(t+\Delta t) = p_{FCR}(t)(1-(\lambda_F+\lambda_C)\Delta t)$$
$$p_{F'CR}(t+\Delta t) = p_{F'CR}(t)(1-(\lambda_F+\lambda_C)\Delta t) + p_{FCR}(t)(\lambda_F\Delta t)$$
$$p_{F''CR}(t+\Delta t) = p_{F''CR}(t)(1-(\lambda_F+\lambda_C)\Delta t) + p_{F'CR}(t)(\lambda_F\Delta t)$$
$$p_{FCRR}(t+\Delta t) = p_{FCRR}(t)(1-(\lambda_F+\lambda_C+\lambda_{RR})\Delta t) + p_{FCR}(t)(\lambda_R\Delta t)$$
$$p_{F'CRR}(t+\Delta t) = p_{F'CRR}(t)(1-(\lambda_F+\lambda_C+\lambda_{RR})\Delta t) + p_{FCRR}(t)(\lambda_F\Delta t) + p_{F'CR}(t)(\lambda_R\Delta t)$$
$$p_{F''CRR}(t+\Delta t) = p_{F''CRR}(t)(1-(\lambda_F+\lambda_C+\lambda_{RR})\Delta t) + p_{F'CRR}(t)(\lambda_F\Delta t) + p_{F''CR}(t)(\lambda_R\Delta t)$$
$$p_{FALLA}(t+\Delta t) = p_{FALLA}(t)(\Delta t) + p_{FCR}(t)(\lambda_C\Delta t) + p_{F'CR}(t)(\lambda_C\Delta t) + p_{F''CR}(t)((\lambda_F+\lambda_C)\Delta t)$$
$$+ p_{FCRR}(t)((\lambda_C+\lambda_{RR})\Delta t) + p_{F'CRR}(t)((\lambda_C+\lambda_{RR})\Delta t) + p_{F''CRR}(t)((\lambda_F+\lambda_C+\lambda_{RR})\Delta t)$$

The obtained equations can be represented by $P(t+\Delta t) = [A]\ P(t)$. Afterwards, the vector of probabilities to stay at each one of the model states was extracted, in such a way that grouping the SQCS success states the survival reliability was determined. To make the calculation of such a vectors an interactive software in Matlab was generated to plot reliability curves for variable times.

To highlight the overall benefits of SQCS hardware (assembled with 3 SBCs) a reliability comparison among three different implementations was performed, one assembled with COTS electronic components, another assembled with military components and a last one assembled with space qualified electronic parts. Figure 3.a shows the reliability results for three different SQCS server versions. The highest reliability is obtained when the computer is assembled with space qualified electronic components and the lowest one is achieved with COTS parts. However the space components are expensive and require longer arrival times. Consequently the best reliability-cost relation is obtained with military components.



(a) (b)

Figure 3. Reliability results for SQCS (assembled with 3 SBC) in a 10 year time period: (a) For three SQCS versions; (b) SQCS and a single SBC both of them with military parts.

The reliability analysis shows the advantages of using military qualified electronic parts as well as the improvement obtained from using spares to increase the expected operational life for the SQCS and therefore for the satellite.

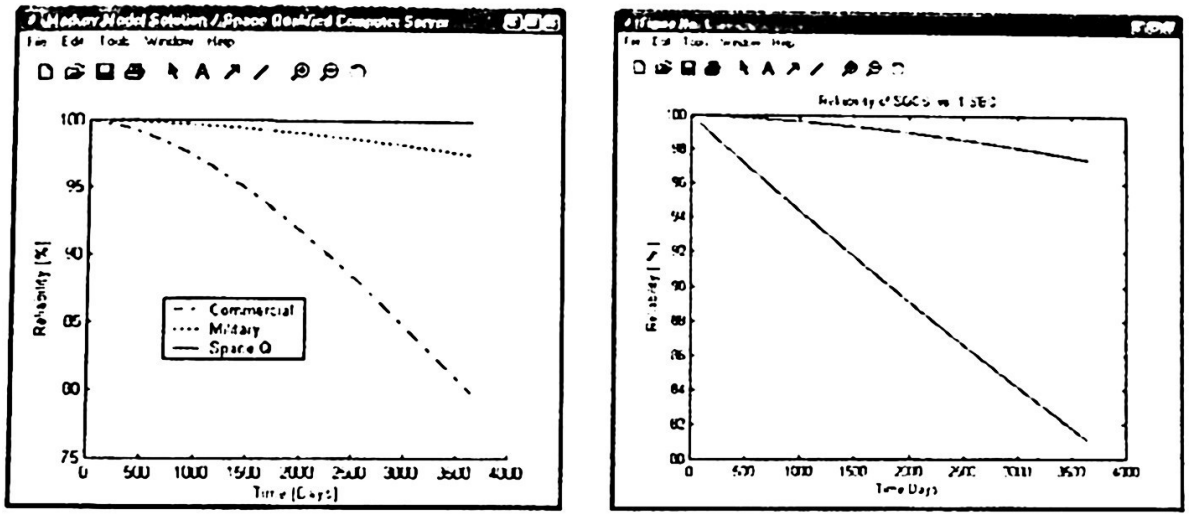
In figure 3.a the reliability obtained for SQCS assembled with 3 SBC in a one year time operation is important because represents the minimal expected life for the satellite. A reliability of 99.94 was obtained for this time, showing a very good forecast because of the spares included for the single points of failure from the architecture. In the paper from Hoffman [7] some reliability values that space equipments have to accomplish are given, see table 1.b. As seen, the SQCS assembled with military parts fulfills the Hoffman's reliability requirements.

In figure 3.b the reliability curves for a SQCS (assembled with 3 SBC) and a single SBC are shown. From this figure the reliabilities for a single SBC are as follows: 97.9, 93.8, 89.9 and 81.2 for 1, 3, 5 and 10 years of operation, respectively. By comparing last reliability values with reliability data given by Hoffman [7] we may conclude that a non redundant SBC assembled with military qualified parts does not accomplishes the reliability indexes required by electronic space equipment. This is the quantitative justification to develop a SQCS with cold standby spares in order to increase satellite survival possibilities.

As also shown in figure 3.b an expected reliability of 0.95 for a single SBC is reached in 2.44 years time period. On the other hand, a reliability of 0.95 is important in aeronautical applications [8], because once an electronic equipment reaches this value preventive maintenance has to be programmed on it in order to avoid unexpected failures. Besides, electronic equipment utilized in long life space applications must achieve a minimal reliability of 0.95 for a ten year time period, [6]. According with this information, the single SBC assembled with military parts must be maintained before 2.44 years, hence does not qualify either as a long life space equipment.

In addition, a reliability study was made for a SQCS architecture assembled with just 2 SBCs. For this architecture a Markov model was obtained, similar to that shown in figure 2.b. For this model, the states F''CR and F''CRR were eliminated. Then the respective Markov equations were obtained, generating a matrix [A] of 5x5 elements. Afterwards, the vector of probabilities to stay at each one of the model states was extracted. Then grouping the SQCS success states the survival reliability was determined. Again, to make the calculation of such a vectors it was elaborated another interactive software in Matlab to generate reliability curves.

Figure 4.a shows the reliability results for three different types of SQCS (each one with 2 SBC) assembled with commercial, military and space qualified components. As can be seen, reliability results for the military version are above 0.95 for 1, 3, 5 and 10 years of operation. This means the equipment accomplishes Hoffman's rule as well as the requirements for long life space equipment. In the case of the commercial version reliability data are: 0.992, 0.966, 0.927 and 0.798 for 1, 3, 5 and 10 years of operation, respectively. In this case only the reliability for 1 year of operation satisfies the Hoffman's criteria. In other words, a commercial SQCS containing two SBC may be used in space for a mission lasting less than one year, if and only if the commercial parts are protected for latch-up phenomena.



(a) (b)

Figure 4. Reliability data for SQCS assembled with two SBC: (a) Three versions of SQCS; (b) SQCS and a single SBC, both of them assembled with military parts.

In figure 4.b reliability results are given for a SQCS (containing two SBCs) as well as for a single SBC, both of them assembled with military parts. The reliability values for the SQCS for 1, 3, 5 and 10 years are all over the Hoffman's criteria. In addition, they all accomplish the reliability indexes for long life space equipments. On the other hand reliability values for the single SBC are the same as those discussed for figure 3.b.

In this way, the obtained results show that SQCS reliability forecast results are encouraging and better than those obtained from centralized computer systems without redundancies.

8 Concluding remarks

This paper has presented the evaluation procedure undergone by the SQCS which was specially designed and manufactured in México for the Satex microsatellite. Taking in to account that SQCS is in charged of critical microsatellite operations, it deserved an architecture design with cold standby spares to avoid single points of failures. In addition, its validation required the availability of complementary satellite subsystems for hardware and software debugging. However all of the satellite subsystems were designed and fabricated in parallel fashion, by this reason the validation of the computer server flight model required the development of special tools. In this sense a satellite simulator, an emulation software for payloads, as well as the earth station software were created to enable SQCS validation without the need of complementary hardware satellite subsystems. In addition, the ESS provided virtual animations facilities to monitor the computer server behavior at a glance. With referred resources the SQCS was debugged in hardware and software, conducting to a successful validation.

In addition, to know the reliability gains obtained from SQCS hardware spares a reliability prediction study was performed. For this purpose, combinatorial techniques, the exponential failure law and the military standard MIL-HDBK217f notice 2 were employed to reduce the SQCS hardware architecture into modules. From these modules the participation of hardware spares were represented by Markov models, and then the Markov equations were obtained. To obtain reliability forecast data dedicated Matlab software was elaborated. The obtained reliability curves show that aggregated redundancies in critical parts of the architecture generate good reliability expectations for the SQCS and therefore for the microsatellite. In particular, the SQCS's reliability results fulfill the reliability requirements established for long life space equipment.

In future papers additional SQCS fault detection facilities will be taken in consideration to generate new Markov models for better understanding of the SQCS behavior under the presence of failures and automatic maintenance. In the mean while reliability data should be considered as optimistic because the SQCS reconfiguration process involves additional hardware neglected in the exposed analysis.

9 Acknowledgements

The authors would like to acknowledge to federal agencies from México, TELECOM and COFETEL who supported the work described in this paper. Besides, acknowledges are given to the Mexican Research Institutions which collaborate in the

development of this project, in alphabetic order: CICESE, CIMAT, CITEDI, INAOE, IPN and UNAM.

References

1. Vicente-Vivas E., Roch-Soto J. y Mendieta-Jiménez J., "La convergencia de un Proyecto multi-institucional para el desarrollo de un microsatélite con tecnología Mexicana", Revista INGENIERÍA Investigación y Tecnología, ISSN 1405-7743, pp. 157-168, Vol. III - No. 4 - octubre-diciembre 2002.
2. Vicente Vivas Esaú, "Arquitectura de cómputo semivirtual tolerante a fallas con capacidad de mantenimiento automático aplicada a un microsatélite de órbita baja", Tesis Doctoral en Ingeniería Eléctrica, DEPFI, UNAM, 21 de mayo de 2004, México, DF.
3. Vicente Vivas E. et al., "Fault-Detection and Reconfiguration Capabilities for the Distributed Computer Architecture on board the Satex Microsatellite", Revista: Seminars of the United Nations Programme on Space Applications, Selected papers on Space Education, Remote Sensing and Small satellites, No.8, pp. 176-182, United Nations, Vienna, 1997.
4. Vicente Vivas Esaú, "Red de Área Local Tolerante a Fallas Aplicada a Sistemas de Tiempo Real con Altos Requisitos de Confiabilidad", Revista Científica de la ESIME, Num.15, Mayo-Junio de 1999.
5. Vicente Vivas E., et all., Space Qualified Computer Server Developed for Long Life Microsatellite Applications, 5th International Conference on Control, Virtual Instrumentation, and Digital Systems CICINDI 2004, México, DF, September 2004.
6. Johnson B. W., "Design and Analysis of Fault Tolerant Systems, Addison Wesley, 1989.
7. E.J. Hoffman y R.C. Moore, "Spacecraft Tracking, Telemetry and Control, Notas de curso, "The Johns Hopkins University, Applied Physics Laboratory Space Department", EU, 1990.
8. Hills, A.D. and Mirsa, N.A., "Fault Tolerant Avionics", *AIAA*, 1988.

Robótica e Instrumentación Virtual

Bauhaus: Development of a walking robot

Francisco Padilla[1], Javier Porras[1], Israel Martínez[1], Luis Ávila[1],
Carlos Soberanes[1], Juan Navarro[1], Jorge Barrera[1] and Leonardo Téllez[1]

[1] Universidad Panamericana
Augusto Rodin #498 Col. Insurgentes Mixcoac
C.P. 02920, México D.F.
agonzah@mx.up.mx

Abstract. Human being has been trying to create mechanisms that resemble living things all along his history in order to use the versatility they have. Within nature there are animals that develop a harmonic locomotion that allows them to overcome the obstacles in an apparently simple way. Walking machines try to get closer to these creatures' technological development, to have access to places where ordinary transportation would be very difficult.

Engineering students from the Universidad Panamericana, located in Mexico City, developed a walking robot that is capable to walk over any kind of surface.

1 Introduction

Flying machines as well as walking machines have a lot in common as both try to copy natural movements. As flying machines were seen by hundreds of people a few years ago, walking machines haven't caused much enthusiasm even today. Perhaps it's due to availability for millenniums, of the wheel. If there were a paved path between two points, vehicles with wheels would be faster than those with legs. Any way, there are many places over the earth where vehicles over wheels can't walk, approximately half of the earth, according to the U.S. Army [1].

Some advantages of walking machines:

- Obstacles.
- Can go up and down stairs.
- Can walk over irregular fields using a effective variation in the leg's length in order to equalize the surface undulations.

Investigations on walking robots have been focused on aspects that interrelate robot's control, the step analysis and the robot's mechanical design. These aspects are divided in two groups:

- Legs geometry.
- Performance and movement transmission power.

The way to lean of a walking machine is the figure of the polygon formed by the points where the foot of the robot touches the floor. More rigorously the support

form is two points dimensionally placed in a plane that consists of a convex helmet of the vertical projection of all the points of the foot when this one is supported in the floor. So that a robot is statically stable, the vertical projection of the gravity centre must be within the support. Robots with static stability have at least three legs.

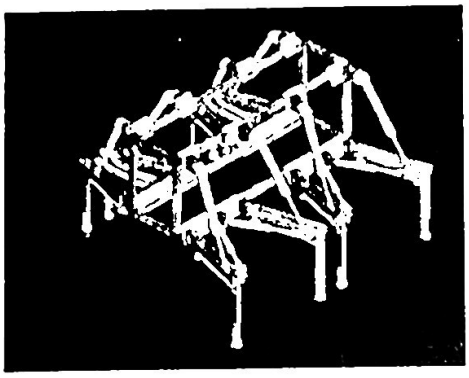


Fig.1. Bauhaus Robot

Project Bauhaus consists in build an octopod robot completely autonomous, mechanically animated with pneumatic cylinders that are sequentially activated by electronic valves. This robot is a direct evolution of Yolcatl and PeKe2:TB robots, built by our university in 2002 and 2003 respectively, this as a stronger development of the project in order to achieve a higher efficiency and use of the machine's resources.

Comparing with previous robots, we consider as primordial objectives:

- Weight reduction.
- Reduce air consumption.
- Simplify electronic circuits.
- Increase autonomy level.
- Simplify construction processes.
- Capability for future upgradeable

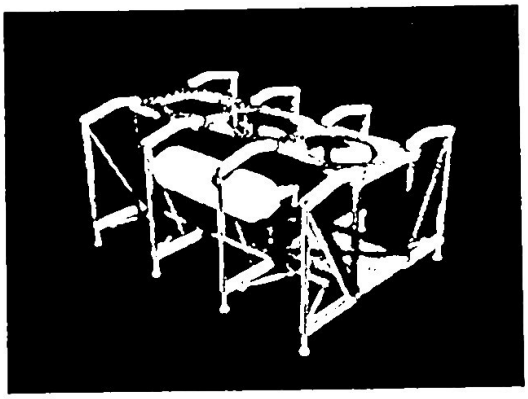


Fig.2. PeKe2:TB Robot

2. Mechanical Description

The Mechanical process design was established with the purpose to create a robust, fast and intelligent walking robot. In order to accomplish this objective, it was necessary to use several static and dynamic simulations, also a material analysis and a study of the finite element (F.E.M.).

The main objective in designing an insect-like robot is the fact that we want to obtain stability while walking in any type of surface. Nevertheless, it is difficult that a robot can obtain an absolute stability while walking through irregular surfaces, mainly if it is desired to have his legs in several heights.

2.1 Gait Analysis

One of the most important characteristics in this type of design is the versatility and the variety of gaits. Taking into account the step length, the stability and the air consumption, it was decided to have only one optimal walk type, although adjustments can be had bye changing the action sequence of the pneumatic actuators.

The robot's step length in each cycle depends on the angle the leg is moved, what depends as well of the horizontal actuators length move. A significant improvement from Yolcatl was that this robot has an adequate actuator position, managing to reduce the race of this cylinders and the air consumption by cycle.

Using the Working Model® software, it was simulated the robot movement, determining the most by this way the most adequate configuration. The configuration seeks to have a perfect symmetry between legs with angular equivalent movements, as much towards the front as backwards, with respect to the spin center of each leg, which is a fundamental affair if we want to have an optimal control mass area.

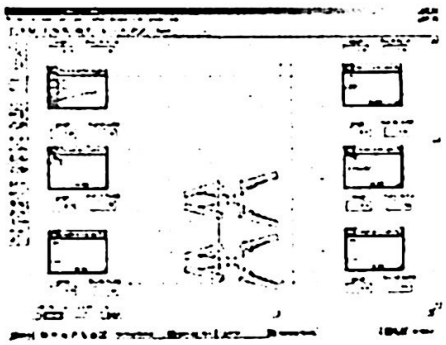


Fig.3. Working Model® display with Bauhaus' mechanical design

Nevertheless, given each leg configuration and the walking type, the robot has a little drag in the moment of driving the body forwards, which causes a little bit of inaccuracy in the total displacement. This point could be understood better if we solve the point on the ground as if they describe an overdeterminated four-legged mechanism

that needs an additional degree of freedom. The problems carried by this drag may be reduced by the flux air regulation that gets through the leg's actuators, thus their consequences can be considered of little importance.

2.2 Lateral Scuff

The fact of walking using angular movements in the legs carries a problem of adjustment on the support position. When one leg is moved forwards and it is put at a certain distance without varying the spin radius, the distance between the leg support and the body is reduced. Now, if it is wanted to displace with one spin point in the new support, it is necessary a spin radius reduction. This change is adjusted by the passive degree of freedom and the ankles that are in the legs, being able the robot to walk without problems of deviations, excessive efforts in the legs or flexing momentums in the pistons rod.

2.3 Mass Distribution

In order to obtain a safe stability in a robot, three alternate strong points are necessary that distribute the weight properly. However, when a robot must walk through complicated surfaces, there is a risk of support lackness, which may cause the robot to fall. The fact that a single leg loads almost with half of the weight of the robot causes that the legs (at least the central ones) must be more robust and that the walk do not cause inertial forces that may be out of the movement axes of the robot. Nevertheless, the robot has movement in its mass center, which intermittently oscillates towards the right or the left but that when finishing the cycle of a complete step arrives at its starting point.

2.4 Turning System and Rotating Method

The robot has two rotating methods. The precision rotating system placed in the inferior part of the robot, has the purpose of making adjustments in the robot trajectory in case of unexpected deviations. A rotating actuator that permits 230° rotations, with torque of 10Nm, enough to beat the robot static inertia, composes the system. This Rotrik® is united to the robots body using bearings, to reduce the effects of friction. The base surface has been simulated, so that it would be the smallest possible, and to reduce the effects of inertia due to the robot's weight. In order to position the robot in the rotating system, it is not necessary to lay it down, but the robot sits down on it when the vertical actuators are closed. With this, the necessary height of the robot to rotate is minimal, having a major stability that enables us to rotate faster. The rotating actuator is in the base of the robot, not inside of it, so we are able to have more space inside the robot, also the base has more weight, gaining with this stability and reducing the falling possibilities.

On the other side, for turning purposes, four legs are raised and set each one at their corresponding position; left legs are set to the front while right legs are set to the back position. The other legs are set higher than the rest of them and external legs are

moved to the opposite side. Several iterations of this method while varying cylinder displacement allow robot to rotate 360° clockwise (CW) or counterclockwise (CCW). A great advantage of this procedure is that robot rotates around an imaginary axis located at the center of body; this allowed robot to make turns in reduced areas.

2.5 Leg Design

Each leg has three degrees of freedom (up-down, left-right, front-back) that permit to have an optimal stability statically or in movement.

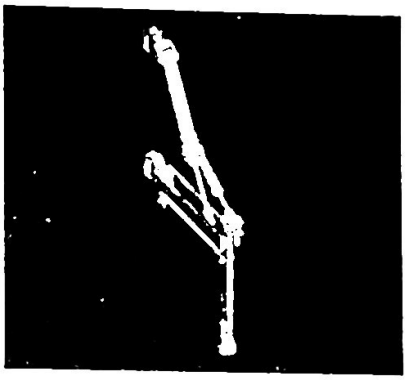


Fig.4. Bauhaus' leg mechanism

The horizontal portion on the leg is made of two parallel rectangular pieces of carbon fiber. A cold rolled iron round joints them and allows articulation with the rest of the leg and body. A detailed resistance analysis was performed in order to select material properties and specifications.

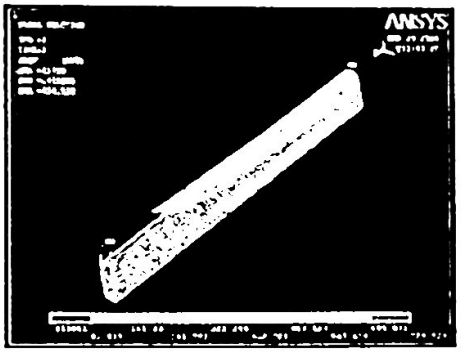


Fig.5. Horizontal bar stress simulation

The vertical leg portion was designed to raise the robot high enough to overcome obstacles that can be found along the track. They also allow the robot to walk at different heights.

By a set of parallelograms, legs are always maintained vertical and thus preventing robot to reduce its stability while ensuring that leg will support the same force anytime. However, when robot walks forward, vertical legs are pulled back due to angle variations at each step of walking. The variations mentioned above produce different

sorts of stresses at inferior rounds of parallelogram. Those stresses created are absorbed by an iron spring, which also, allows leg to return to its perpendicular position against the floor when robot raises it back. Even that spring coefficient was calculated and tested, the leg is provided with two nuts and washers that allow adjustment of this system in case it has lost its elasticity and stiffness.

3. Pneumatic System

The pneumatic circuits provide to the robot the power needed to move him without an external power source, and give it the force to carry a bigger weight than its own. The system was kept very simple and compact, especially because of the AS valves which provide modularized support for pneumatic system.

Pneumatic circuit of robot consists of two independent sets of valves, both supplied by the same air tank. Each pneumatic circuit moves both horizontal and vertical cylinders, which are responsible of providing motion force to walking machine. Both circuits are controlled by an element connected in consistent series to a fast-escape valve activated directly by the emergency stop button. The reason of this fast escape valve is to fully free the air contained in system, except from air inside pneumatic cylinders. A third circuit provides the necessary air to power the circular actuator that permits the rotatory movement of the rotation system.

Finally, before air is sent to valves, a basic maintenance circuitry was added in order to eliminate possible humidity from air supply. This device is not required in spite of the air supply used. However, it was added for testing and maintenance situations in which air supply might be different.

4. Control System

Several factors were considered before choosing the controlling devices, including price, adaptability, performance, reliability and ease of use. Other important prerequisite was the multitasking feature, available on some controllers. Not only this allows getting better controlling but allows obtaining a smoother performance while walking. Although most single-board computers provide several built-in programming languages, just a few can be easily integrated without dealing with assembly-debugging or separated compiling. The time-critical support of the controller we selected, allowed us to mix assembly code and C code without precompiling and using far-calling conventions.

4.1 Programming Schemes

Programming system was based on a layered model similar to TCP, which provides reliable flow of data from entities. Also optimized code based on preprocessor usage reduces CPU overhead and allow mixed multitask to use spare cycles for making

arithmetic calculations for next step. New features include a self-orientation system involving the auto leveler system.

4.1.1 Layered Programming Model

A layered model has several advantages against traditional programming model in which the code is just divided into functions that has no specific rules or that perform low and high level operations. Layered model is capable of being partially updated or upgraded while preserving its compatibility between each function. In addition, major modifications or improvements are fully supported by making just small modifications to concerning layer.

Another important advantage of this model is that as soon as a layer is operational and bug free, the rest of the code is built over this stable platform, ensuring that future development problems will be independent from lower level layers. Finally, layers make group-working easy cause each developer can be responsible of one layer without having to take in consideration lots of previously defined variable names or rules that involve the whole code.

Our model is divided in 4 layers as TCP is. OSI model involves a physical layer similarly structured as our model. The main reason of this is because our model was based on TCP. The layers used to design code were the following:

- Physical: Responsible of reading sensors and moving each cylinder to the indicated position.
- Walking and Positioning: Responsible of calculating robot position through various ways and detecting possible navigational errors and drifts.
- Path and Obstacle Identifier: Responsible of finding the shortest path and keeping track of obstacles or objects.
- Principal Task: Responsible of taking decisions and executing the different steps required to achieve the specific goal.

4.1.2 Mixed Multitasking

Microcontroller offered the possibility to implement both cooperative and preemptive multitasking without any serious problem or performance impact. Preemptive multitasking was used for fast or permanently updated tasks without releasing control to system or consuming full processor cycles for a period of time, for instance, for sensors. Cooperative multitasking provides a much more efficient and precise way of controlling high priority tasks or tasks that involves activating a specific mechanical device (in which most of the time is spent losing cycles).

Multitasking was established within four priorities to allow extremely important tasks to be running in foreground while slow tasks or low priority tasks are running in backgro and or whenever they are required for taking decisions. This scheme allowed the

robot to be thinking while its moving reducing time before each decision is taken or then next step is initiated and while keeping code unmixed and modularized.

4.1.3 Extensive Pre-processor Usage

C Pre-processor was extensively used to reduce the number of functions and to reduce the size and number of parameters passed to functions. Also, the pre-processor was used to reduce code complexity while keeping its size small and high performance. At high level programming explicit directions like FRONT, LEFT, HIGH, or LOW are passed to functions making code readable and understandable. In the pre-processor, exactly before compiling, words are transformed into short integer constants that are passed much faster after compilation. Also, conditional compilation took an important part in developing code and testing process, allowing code to just include statements required at that time. The main function contains the multitasking code and robot initialization process, which is capable of setting the robot at home position and perform a quick self test. After that process the robot checks whether a specific trial was selected and prepares to execute the "Principal" task once. While the Principal Task is running a monitoring task called Sensors checks robot status and sends it to a global memory area, which was based on DMI concept. The whole set of tasks are being executed while the robot is trying to accomplish an specific duty, making it fully automated and creating a true "Act/React" system.

5. VISION SYSTEM

The vision system was created to fulfill the requirements of a versatile positioning system that could give us a real autonomy. The general idea was to recognize the lines and numbers on the stage in order to calculate the absolute position and direction over the stage. This recognition is made during 3 main stages, in the first we recognize the lines, next we search for a relative position and finally we get the absolute position.

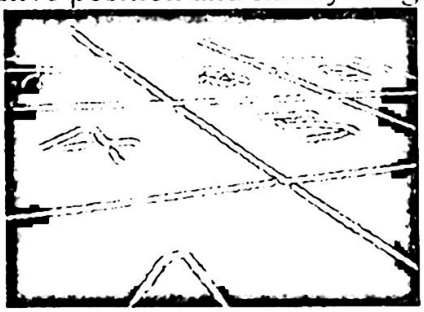


Fig.6. Track picture taken by Bauhaus camera

1st recognition phase

We start by taking a frame from the camera and then we process it by applying the Roberts Operator to enhance the edges from the image. We also calculate *a priori* some tables to optimize the next process, which is the Hough Transforms, this transform searches all over the bitmap using an accumulator array and lines represented in

their (ρ,θ) form. Finally we throw away lines that can't form a perspective and then we calculate the intersection points between these lines.

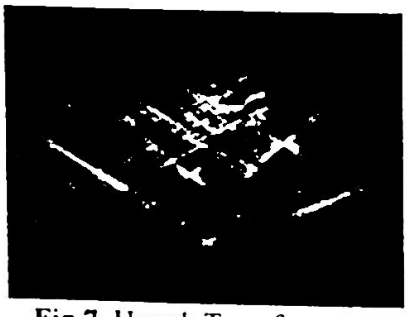


Fig.7. Hough Transforms

2nd recognition phase

The main idea behind this process is to apply an inverse rendering algorithm using the intersections points and the fact that the points that form the block we are looking for in the space must be equally distant and with 90° angles. The approach we take to solve this nonlinear problem was:

(a) Reduce the search range by using obvious facts about where the block we are looking for is.
(b) Scan each point on a line intersecting our range of search from beginning to end and looking if there is an optimal block that contains it.

After a successful search we made a coordinate system transformation, so we obtained the robot position relative to a block that is unknown by now.

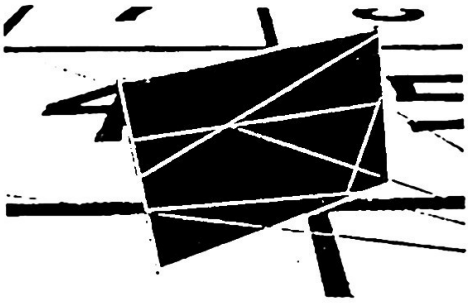


Fig.8. 2nd recognition phase

3rd recognition phase

The objective was to identify the block that the robot was seeing in order to transform the relative position into an absolute position. The approach we took was to compare the block identified with the mapping of a photo taken from each of the numbers. It was also needed to compute the mass center of each captured number to reduce the

error from the Hough transform, then we assign a probability of being the correct number to each photo in the library and finally from the one with greatest probability we calculate the absolute position.

6. Conclusion

This robot presents the following advantages regarding its predecessor: Light weight; better stability during walking due to legs geometry; piston displacement reduced by 50%; air flow consumption was reduced by 40%; turning sequence optimized to a minimum amount of steps; there was developed a completely Re-design and improved vision system; programming code was reduced to its minimum.

Also we have many innovative approaches: modular design simplifies transportation and assembly; materials as carbon fiber and Epopast® used on robot's structure (Legs build with two different materials to enhance properties); and the use of finite element analysis simulation for validate the design of critical parts.

Then we have many troubles designing the robots that give us new ways of improvement and investigation: eight legged configuration generates a statically overdetermined system, which leads toe small pulling in the legs; horizontal piston's position creates a certain waste in its race; horizontal piston's force is not equivalent in all race moments; there has to be certain angle compensation due to the angular movement of legs; used camera requires a great range of vision, because of that we need to put it in a high position respect the floor; and a great amount of light made difficult to distinguish specifies objects.

References

1. ANON (1967), Logistical Vehicle Off-Road Movility, Project TCCO 62-5, U.S. Army Transportation Developments Agency, Fort Eustis, Va.
2. BASUALDO, Raul, JACOBUS, Roberto; Designing a Walking Robot Through Computer Simulation, III Conference about Computer Simulation, Mexico City, 195pp 138-144.
3. Da Fortuna, L.; Shape Analysis and Clsification. Theory and Practice; CRC Press; 2001.
4. Watt A.; Advanced Animation and Rendering Techniques, Addison-Wesley; 1992
5. Glaeser, G.; Fast Algorithms for 3D-Graphics; Springer-Verlag; 1994

Un ambiente de simulación y de realidad virtual para asistir el diseño de redes hidráulicas

Jaime Muñoz Arteaga[1], Jannet Cruz Pérez[2] y Gustavo Rodríguez Gómez[3]

[1]Departamento de Sistemas de Información de la Universidad Autónoma de Aguascalientes. Av. Universidad # 940. Bosques. Aguascalientes, México. C.P. 20100.
jmunozar@correo.uaa.mx

[2]Universidad Autónoma de Tlaxcala. Departamento de Ciencias Básicas Ingeniería y tecnología. Calzada Apizaquito s/n. Apizaco, Tlax., México. C.P. 90300
jcrufperez@hotmail.com

[3]Instituto Nacional de Astrofísica Óptica y Electrónica (INAOE). Coordinación de ciencias computacionales. Luis Enrique Erro #1, Sta. Ma. Tonantzintla, Puebla México
grodrig@inaoep.mx

Resumen. El diseño de redes hidráulicas es reconocido como una tarea difícil ya que es necesario plantear propuestas en términos de modelos y requiere un alto nivel de abstracción para optimizar el desempeño del traslado del flujo de un medio a otro. A pesar de que los ambientes de simulación facilitan el trabajo del diseñador en la creación, mantenimiento y ejecución de los modelos de redes hidráulicas, estos ambientes solo pueden ser usados por diseñadores expertos pues dificultan el aprendizaje hacia diseñadores novatos e inclusive llegan a complicar la capacitación. Con el fin de mitigar este problema, el presente trabajo propone integrar las facilidades de la realidad virtual en los ambientes de modelado y simulación. El resultado de dicha integración es el sistema VRASE el cual asiste al usuario en el análisis, la verificación, el aprendizaje y la exploración de los modelos de redes hidráulicas como se observan en el mundo real.

Palabras clave: Redes hidráulicas, Modelado, Simulación y Realidad Virtual

Abstract. The modeling of hydraulics nets is considered as an important task for the development of simulators training. This work proposes a simulation environment called for design hydraulics nets represented in three dimensions inside a virtual world in order to help the user to analyze, verify and explore the models of hydraulics nets with elements represented as they are in the real world. Using this environment makes easy the user task, saving cost, time and effort.

Keywords: Hydraulic nets design, Modeling, Simulation and Virtual reality.

1 Introducción

La realidad virtual es usada como una herramienta de análisis y exploración de modelos. Aún si la expresión "realidad virtual" en si mismo es paradójico ya que se compone de dos términos totalmente opuestos "Real" y "Virtual", es difícil tener una definición exacta, sino mas bien variantes tal como la siguiente: "Disciplina que permite la visualización, manipulación e interacción con ordenadores y datos extremadamente complejos"[1], dentro de esta disciplina existen clasificaciones por lo que es importante aclarar que se usara la realidad virtual de escritorio. Un sistema de simulación de realidad virtual cuenta con tres grandes características a saber, *la simulación, la interacción y la percepción* [2]. La simulación es el proceso de diseñar un modelo de un sistema real y simular dicho modelo con la finalidad de comprender el comportamiento del sistema o de evaluar nuevas estrategias - dentro de los límites impuestos por un criterio o conjunto de ellos, para el funcionamiento del sistema.

El área de la hidráulica es "la rama de la mecánica que tiene por objeto, por medio del análisis y la experimentación unidos, estudiar el movimiento y el equilibrio de los fluidos, y especialmente del agua" [3]. Así el diseño de redes hidráulicas "es la conexión de elementos que permiten el transporte del fluido con cierta velocidad y cierta cantidad según los requerimientos establecidos o acordados".

El propósito de este trabajo es el diseño y experimentación de redes hidráulicas asistido por un ambiente de simulación de realidad virtual, manipulando los modelos gráficos correspondiente a equipos tales como, bombas, tubos, depósitos, reactores etc, así como fluidos, estos fluidos van desde agua hasta sustancias que ponen en riesgo vidas humanas, experimentando, modificando variables, tantas veces como se desee hasta obtener el resultado esperado.

2 Problemática en el Diseño de Redes Hidráulicas

Es necesario tener conocimiento y experiencia para implantar y mantener en operación eficiente una red hidráulica, ya que si algún elemento falla (por ejemplo una tubería), las consecuencias representa una gran perdida de recursos. Por otro lado una red hidráulica transporta fluidos que pueden ser desde agua hasta sustancias que representan algún riesgo para el bienestar humano, por lo que es necesario estar muy seguro del tipo de material y equipo que deberá transportar los fluidos, la cantidad necesaria, presión, velocidad etc., con el fin de evitar perdidas materiales así como humanas.

El diseño de redes hidráulicas es reconocido como una tarea difícil ya que es necesario de la intervención de una serie de conocimiento y acciones a realizar por el diseñador. Una gran mayoría de las fallas se debe a la falta de experiencia por parte del diseñador, por lo que conviene facilitar el diseño de modelos para visualizar en cada etapa un panorama en donde puede haber posibles fallas y actuar anticipándose a estas. Enseguida es mostrado el proceso general que involucra el diseño de redes hidráulicas tradicional. (ver Fig. 1)

Ahora bien, ya que todo este desarrollo se mueve en función del tiempo y costos, entre mayor sea el tiempo mayor es la inversión. Una falla en el cálculo de diseño en cualquiera de sus puntos en las etapas, afecta fuertemente el desempeño de la red, además puede generar consecuencias catastróficas según la naturaleza del fenómeno a tratar.

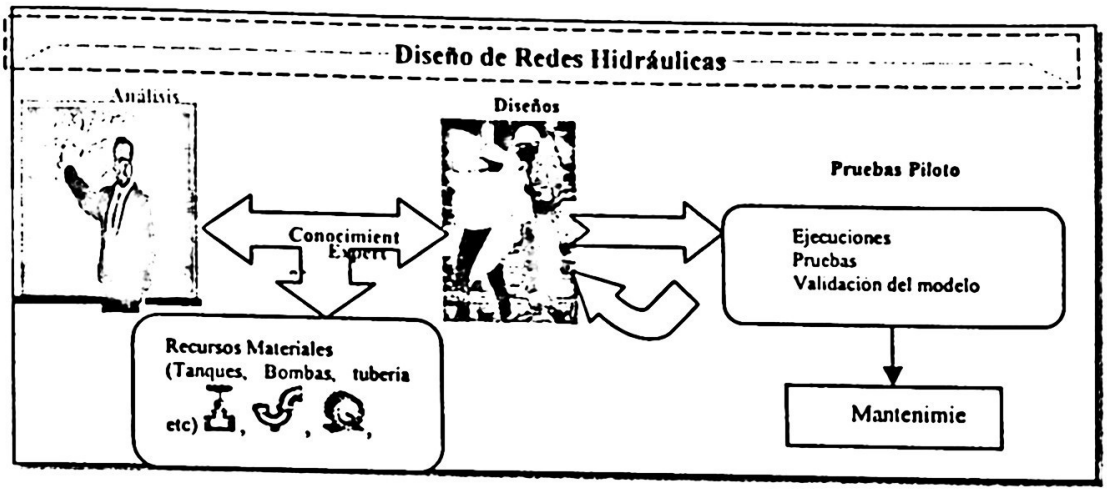


Fig. 1. Proceso tradicional para diseñar redes hidráulicas.

Para el diseño de una red es necesario el conocimiento y experiencia del (los) diseñador (es) experto(s) para lograr resultados satisfactorios, y para expresar el conocimiento se lleva a cabo una serie de procedimientos, o ciclo de vida de la red hidráulica, en la que se inicia con el análisis, diseño, experimentación y mantenimiento, en cada una de estas etapas podríamos encontrar dificultades, puntos vulnerables que se describen en a continuación en la Tabla 1

Tabla 1. Puntos vulnerables en las etapas del diseño de redes hidráulicas

Etapas	Posibles puntos en los que se pueden encontrar problemas
Análisis	-Formulación problema -Recolección de datos -Análisis de requerimientos (características del medio y equipo a instalar) -Estimación de Tiempos
Diseño	-Cálculos Matemáticos -Diseño y Elaboración de Planos
Experimentación	-Pruebas de los modelos propuestos -Análisis de resultados
Mantenimiento	-Cambio de equipo -Calibración de equipo

Durante el diseño pueden encontrarse puntos vulnerables en el *análisis*, como el no tomar en consideración aspectos del fenómeno a analizar, lo cual puede traer como consecuencia datos erróneos que afectan el planteamiento del problema. Al mismo tiempo repercute en el modelo del *diseño* en la definición de ecuaciones matemáticas erróneas y evita dar una solución adecuada para representarlas en planos. Dando

como resultado una *experimentación* que arroja resultados inconsistentes o no deseados. Todo este proceso genera perdidas y consumo de tiempo. Además, el diseño de redes hidráulicas exige el trabajo de un conjunto de expertos por lo que es necesario que estos estén capacitados, organizados y tomen en consideración aspectos como los antes mencionados en cada una de las etapas para un rendimiento eficaz y eficiente de la conexión de las tuberías a instalar.

3 El Diseño de Redes Hidráulicas Asistido por Computadora

En este trabajo proponemos encapsular el conocimiento, experiencia y fundamentos del diseñador experto en redes, ocultar la complejidad del modelado matemático, y presentar una solución representada en un ambiente de simulación y de realidad virtual. (Fig. 2).

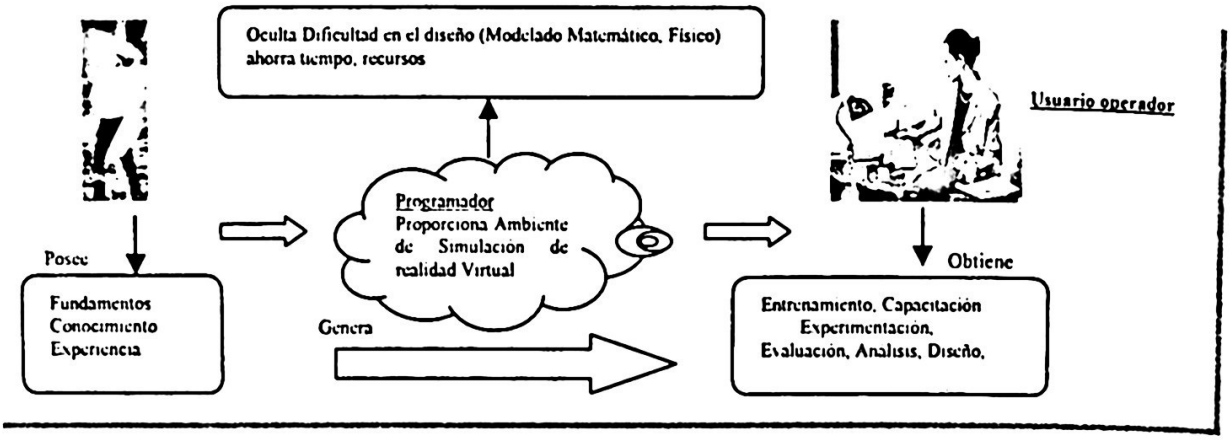


Fig. 2. Diseño de redes hidráulicas asistido por ambientes de simulación y de realidad virtual

Nuestro objetivo es integrar realidad virtual con el fin de ofrecer una herramienta de diseño para redes hidráulicas que facilite la tarea difícil de diseño, ahorrando tiempo, ofreciendo capacitación y experimentación al usuario común o inexperto. Es importante indicar que el objetivo de integrar la realidad virtual a los ambientes de modelado y simulación no es con el fin de atenuar las ventajas que ya se ofrecen al usuario. Las ventajas de un ambiente de simulación y de realidad virtual aquí se pueden clasificar conforme a los criterios establecidos por [2] aspectos de *interactividad, de simulación y de percepción.*

Tabla 2. Ventajas de un ambiente de diseño interactivo, simulación y con realidad virtual

Interactividad	Simulación	Percepción
-Agiliza el diseño, análisis - Selección de equipo - Conexión de Equipos - Facilidad de uso -Transparencia	- Experimentación - Facilidades de análisis - Exploración - Animación - Estimación - Inspección	- Representación real - Vistas de observación - Translación, rotación - Sonidos 3D - Desplazamiento 3D - Inmersión

- *La simulación* representa la actividad del fenómeno físico y llevar acabo experiencias, con el fin de comprender su comportamiento.
- *La interacción* se da mediante la modificación de estado de las variables de equipo a actuar, modificando y experimentando en algún punto o situación del fenómeno
- *La percepción* ofrece un cierto grado de percepción al usuario a través de los sentidos (vista, oído y tacto), y se recibe viendo la simulación del comportamiento del fenómeno y escuchando algún efecto de sonido en la animación y manipulando mediante el ratón modificando el estado del equipo.

La importancia de integrar la realidad virtual a los ambientes de modelado y simulación es el de incrementar las facilidades de diseño de tal manera que resuelvan la problemática planteada en la sección anterior en lo que corresponde al diseño de redes hidráulicas. La realidad virtual en ambiente de simulación beneficia resolviendo los problemas de diseño de redes hidráulicas como es mostrado en la Tabla 3.

Tabla 3. Solución a la problemática por medio de un ambiente de simulación de realidad virtual.

Etapas	Solución propuesta para el usuario en el diseño de redes hidráulicas mediante realidad virtual
Análisis	•Especificación de estados y variables del equipo a actuar •Selección de equipo a interactuar mediante un menú
Diseño	•Oculta la complejidad del modelo matemático •Visualización del diseño 3D (como en los planos) •Colocación de cada elemento dentro del ambiente •Desplazamiento dentro del ambiente •Conexión de equipos que intervienen en la simulación
Experimentación	•Apoyo al experto •Capacitación del usuario diseñador inexperto •Transparencia en la experimentación •Visualización del fenómeno 3D •Interacción en forma directa con el equipo •Navegación 3D (rotación, traslación, escalas)

	•Sonidos de equipos como si fuera real •Evaluación de la forma de conexión y especificación de estados •Análisis de Resultados •Repetición de Experimento •Modificación de Experimento •Validación de hipótesis
Mantenimiento	•Sustitución de componentes (equipo) •Evitar riesgos antes de hacer modificación en equipo real

- En la etapa de análisis de acuerdo a las condiciones del problema podemos establecer los parámetros iniciales y configuraciones de cada equipo a intervenir, seleccionándolo según las necesidades, sin que generen recursos perdidos en caso de equivocación (ver tabla 3).
- En el diseño, hacia cualquiera de los dos tipos de usuario, oculta la complejidad de los cálculos, pero no de los resultados, de esta forma el usuario puede saber si es que es lo deseado o esperado, en otro caso vuelve a intentar (ver tabla 3). Además de poder colocar el equipo en una posición (x,y,z), desplazándose de un lado a otro, insertando, retirando o conectando componentes dentro del área de trabajo.
- En la experimentación es la evaluación de lo que se espera un mayor realismo, repetición del proceso, modificación de variables, etc.
- En el mantenimiento para un modelo ya existente pues suele haber cambios, es decir colocar equipo con características diferentes, entonces regresamos a la experimentación, pero la ventaja es que no se arriesgan recursos materiales ni humanos al menos sin estar seguros de que va ha funcionar (ver Tabla 3).
- En la capacitación ya que los simuladores de realidad virtual representan y facilitan el manejo de los objetos tal como aparecen en el mundo real así los refleja la computadora, llegando a convertirse una herramienta para entrenamiento y aprendizaje.

4 VRASE Un Ambiente de Simulación de Realidad Virtual

En esta sección se da a conocer el ambiente de simulación de realidad virtual VRASE ("Virtual Reality and Simulation Environment"), propone una solución para la capacitación y diseño de redes hidráulicas mediante un ambiente que favorecerá el entrenamiento de los futuros expertos en redes hidráulicas.

La ventaja de tener una herramienta para el diseño de redes hidráulicas con espacios de realidad virtual es que nos permite manipulación directa sobre el ambiente virtual, nos permite proponer y obtener actividades y resultados siempre diferentes durante la experimentación. VRASE cumple con tres aspectos fundamentales: simulación, interacción, percepción. Lo que da como resultado una herramienta factible para el diseño de redes hidráulicas y capacitación al usuario.

La arquitectura de VRASE esta orientada a objetos, donde cada elemento es independiente en cuanto su funcionalidad, lo que facilita al usuario la elección y manipulación de los elementos a usar, un tiempo de ejecución mas corto, etc.
En la creación del ambiente virtual se desarrolla en VRML (Lenguaje de modelación de mundos Virtuales), este lenguaje por si solo permite la creación de animaciones estáticas entre el usuario y el ambiente, sin embargo para el diseño de redes hidráulicas se requiere de animaciones dinámicas, donde el usuario necesita equipo a insertar, modificar o borrar, además de la modificación del estado de las variables para ver el comportamiento de la simulación en un lapso de tiempo, y todo esto para la evaluación de los resultados. JAVA permite dar el funcionamiento dinámico del sistema y para poder otorgar estas características al ambiente de simulación es necesario hacer uso de EIA (External Autoring Interface), esto significa tener los módulos dinámicos fuera del VRML, comunicándose con scripts mediante eventos de entrada y salida.

De esta manera puede realizarse la ejecución más rápida además de aligerar el tamaño de los archivos. El acceso a VRASE es mediante una conexión al servido por Internet, lo que permite al usuario accesarlo desde cualquier lugar, con la única condición de disponer de algún visualizador para archivos *.wrl como CosmoPlayer, que se encuentra en Internet disponible. En la figura 3 representa la comunicación entre VRML y Java mediante EIA.

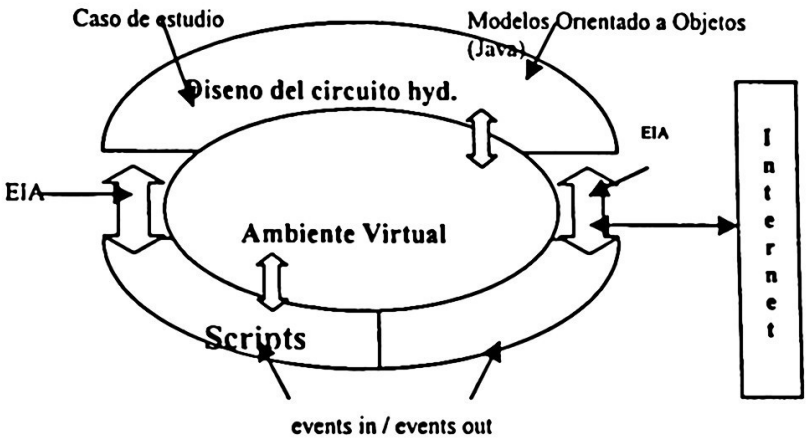


Fig.3 Comunicación de JAVA, VRML, y EIA para simulación dinámica.

El modelo arquitectónico del sistema esta basado en el conocimiento del experto y pretende cubrir las necesidades que surgen en el usuario inexperto con el fin de dar una solución de un fenómeno. El programador es el mediador entre el conocimiento y la experimentación del usuario dentro del ambiente virtual. En la (Fig 4) se establece la arquitectura del sistema compuesta por tres grandes módulos: Los modelos y el controlador de la interacción y la Interfaz de Usuario", puntualizaremos cada parte del sistema según el modelo conceptual arquitectónico.

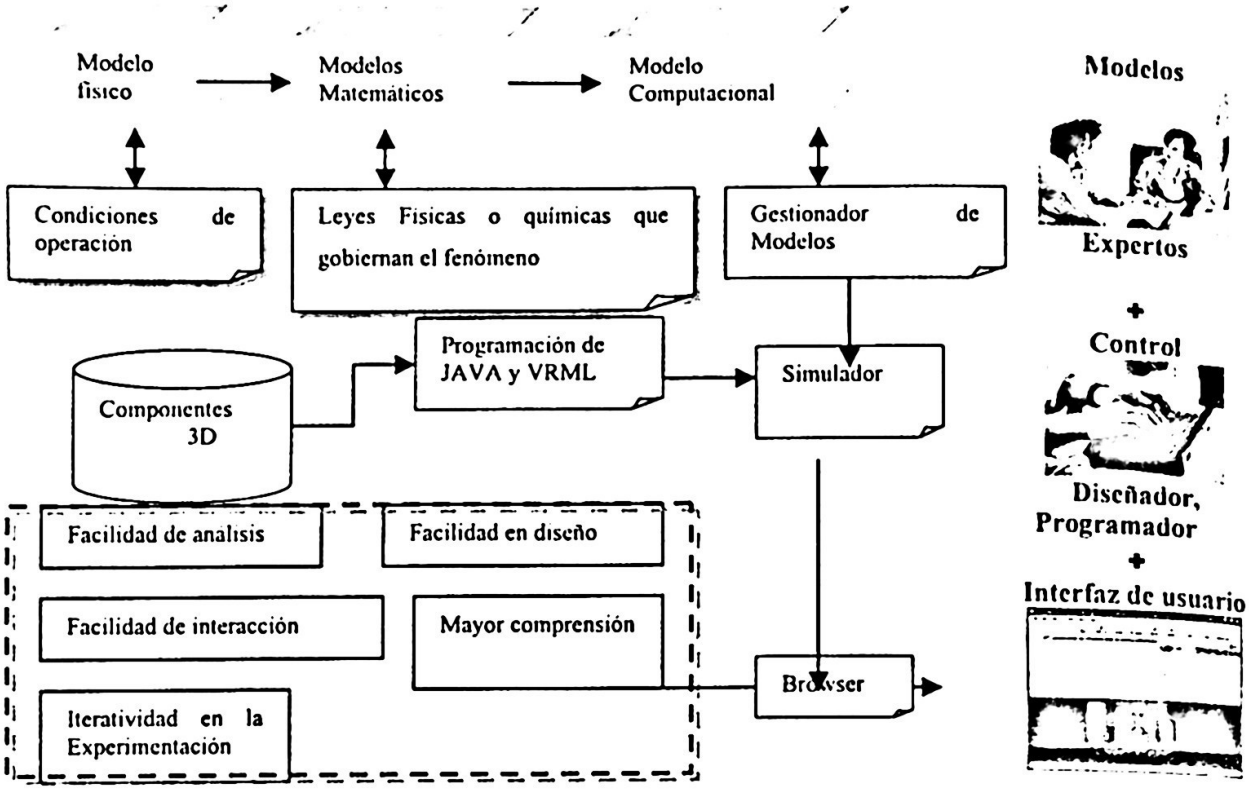


Fig. 4. Modelo arquitectónico del ambiente de realidad virtual y simulación.

Los Modelos están basados en los conocimientos del experto, básicamente interviene el modelado Físico y Matemático es donde se definen las leyes físicas que gobiernan el fenómeno así como los estados de cada componente dentro de la red (paro, inicio, cambio).

Controlador de la interacción es dado por el trabajo del programador, en este punto entra la simulación mediante el modelo computacional, esto significa la codificación del modelo matemático y un método de solución para resolverlo mediante el uso de la computadora. Esto requiere la declaración de variables, definición de funciones, cláusulas condicionales, restricciones que permitirán al usuario interactuar con el ambiente.

Interfaz de usuario Permite al usuario tener un mejor análisis, de lo que ve, una mayor interacción o comunicación con el sistema, interactuando y experimentado cuantas veces lo desee obteniendo resultados de la experimentación, además de la facilidad de diseño y conectar los equipos a actuar, observando el resultado del diseño.

Por la otra parte, favorece al usuario diseñador con modelos mas reales, con la capacidad de colocarlos en alguna posición de tres dimensiones (x, y, z) y distribuirlos según su criterio de modelado a fin de obtener resultados esperados o aun los inesperados. Disponiendo de un conjunto de equipo para redes hidráulicas de los que puede colocar 1,..., n según los necesite, para la experimentación, modificando el estado de equipo, posición, etc. Aprendiendo y capacitando al usuario.

5 Trabajos Relacionados

El ambiente de modelado y simulación VRASE es puesto en comparación con otros ambientes de simulación en la Tabla 4. Los criterios de comparación son en base a un conjunto de criterios generales para los ambientes de simulación y de realidad virtual. En función del grado de aportación que ofrece un trabajo a un criterio se cuantifica de la manera siguiente con un 3 de buen nivel de aportación, 2 de Regular, 1 de Suficiente y 0 de nula aportación.

Tabla 4 Comparación de Ambientes para el Diseño de Modelos.

	LabView [4]	TI [5]	AF [6]	MGFP [7]	EVEC [8]	CIPRES [9]	SIMU2 [10]	VRASE
Simulación	3	3	3	3	3	3	3	3
interacción,	3	3	3	3	3	3	3	3
Percepción	1	1	1	1	2	3	3	2
Gráficos 3D,	0	0	0	3	3	3	1	3
Navegación	1	1	1	1	3	3	2	3

Conforme a la tabla LabView[4], TI[5], AF[6], y MGPF[7] asisten efectivamente al diseñador en la tarea de realización de modelos de redes hidráulicas ofreciendo un ambiente de modelado y simulación en solo dos dimensiones sin ofrece las facilidades de realidad virtual que el presente trabajo preconiza. En cambio los ambientes EVEC[8], CIPRES[9], SUMU2[10], efectivamente utilizan escenarios tridimensionales que permiten desarrollar experimentos virtuales interactivos pero no en el dominio de los modelos de redes hidráulica. Ahora bien, el sistema VRASE aquí propuesto toma ventaja sobre los sistemas anteriores ya que toma aspectos de simulación, así como simulación con realidad virtual para asistir eficientemente al diseñador en su tarea con modelos lo mas cercanos a la realidad y apoyándolo en la validación y verificación de dichos modelos con las facilidades de la realidad virtual.

6 Conclusiones

El presente documento preconiza la integración de la realidad virtual en los ambientes de modelado y simulación de redes hidráulicas. Una de las principales ventajas a ofrecer es la facilidad de interacción con un modelo de la red hidráulica el cuales es representado y manipulado de forma mas real de manera que el usuario se vea inmerso en un ambiente mas familiar, ofreciendo así un ambiente capaz de satisfacer al diseñador experto como al diseñador iniciado. Además el ambiente de simulación y de realidad virtual VRASE es propuesto como una herramienta que asiste al usuario en el análisis, el mantenimiento, el aprendizaje y la exploración de los modelos de redes hidráulicas. El objetivo a alcanzar con el ambiente VRASE es encapsular el conocimiento, experiencia y fundamentos del diseñador experto en redes, ocultar la complejidad de los modelos, y presentar una solución representada en un ambiente de simulación y de realidad virtual.

Reconocimientos

El presente trabajo es parte del proyecto de investigación de ciencias básicas con clave 2002-C01-40022-Y. Los autores agradecen el apoyo económico recibido por parte de CONACYT.

Referencias

1. Ledesma Ma. Jesús, Introducción a la Realidad Virtual, Ingeniería Neuro sensorial, Departamento de ingeniería Electrónica (2000)
2. Ortega, J.A., Burgos, M. A., Vera G.: La Realidad Virtual y sus Posibilidades Didácticas, Universidad de Granada. Universidad de Malanga (2001)
3. Sotelo, G.: Hidráulica General. (ed.) : Limusa, Vol. 1. Fundamentos, (1996) 111-1134.
4. Nacional Instrument LabView 6i, copyright Nacional Instrument July (2000)
5. Morales, E., Sucar, E.: Tutores Inteligentes, ITESM Campus Morelos, Temixco, Morelos, México, European Conference on Artificial Intelligence, Wokshop. Model based reasoning for intelligent education environments, (1998) 58-83.
6. Applied Flow Technology, AFT Fathom V5.0, USA, website: www. aft.com, (2002)
7 Roldan, E., Tavira, J.: Automatización del modelado de Redes Hidráulicas con aplicación de Simuladores para Entrenamiento, Boletín IIE (1997)
8 Torres, G.A., Nuñez, G.: Espacios Virtuales de Experimentación Cooperativa Laboratorio Virtual de Cinemática, Universidad Autónoma del Estado de Hidalgo, Centro de Investigación en Tecnologías de Información y Sistemas, Pachuca Hidalgo(2001)
9. A. Felipe, F. Sarti, J. Rodenas.: Entornos de Realidad Virtual a la Protección Radiológica. Departamento de Generación Nuclear. Madrid España Y Universidad Politécnica deValencia Departamento de Ingeniería Química y Nuclear, Valencia(2001)
10. Gómez, B., Salve, R., Burgos, E., Herrero, P., Quiros F., Largo, P.: Simulación Virtual para lanificación de Interve nciones en Entornos. TECNATOM, CIEMAT, UPMCNA (2002) lleve

Sistema de supervisión y adquisición de datos en un ambiente distribuido

Juan Francisco Novoa Colín, Luis Pastor Sánchez Fernández

Centro de Investigación en Computación del IPN
Av. Juan de Dios Bátiz s/n esquina con Miguel othón de Mendizábal
Unidad Profesional Adolfo López Mateos. México, D.F. 07738, México
Juan Francisco Novoa Colín jfnovoac@correo.cic.ipn.mx,
Luis Pastor Sánchez Fernández lsanchez@cic.ipn.mx

Abstract. A project of a virtual Data Acquisition and Supervisory Distributed System (SCADA) for industrial processes, it was developed on a communication protocol for industrial networks called OPC; this was implemented in the programming language G, on the LabVIEW development enviroment. The system counts on several frontal panels to watch the behavior of the process variables, three types of input data, kept from data to a text or binary file, an algorithm that allows the flexibility to have so many sampling periods as process variables supervised, as well as digital filters, ifnecessary

Resumen. Se presenta un proyecto de un Sistema Distribuido de Supervisión y Adquisición de Datos (SCADA) virtual para procesos industriales, se desarrolla sobre un protocolo de comunicaciones para redes industriales llamado OPC; esta implementado en el lenguaje de programación G, en el ambiente de desarrollo LabVIEW. El sistema cuenta con varios páneles frontales para observar el comportamiento de las variables a monitorear, tres tipos de entrada de datos, guardado de datos a un archivo de texto o binario, un algoritmo que permite la flexibilidad de tener tantos períodos de muestreo como variables del o de los procesos supervisados, así como filtrado digital, en caso de ser necesario.

Palabras clave. SCADA, distribuido, OPC, periodo, muestreo, LabVIEW, variable, virtual.

1 Introducción

Un sistema SCADA permite conocer los diferentes estados de las variables que son parte de un proceso supervisado; su configuración se realiza mediante las aplicaciones HMI/SCADA, que son programas orientados a objetos y eventos con un amplio manejo de información. Al usar sistemas SCADA comerciales, lo único que se hace es configurar el sistema, sin realizar ningún tipo de desarrollo, y generalmente hay una falta de flexibilidad o un exceso de opciones; aunque por otro lado no se requiere de ningún conocimiento de programación para utilizarlos, además de tener un costo muy elevado. Por otro lado, al desarrollar un sistema SCADA en un entorno de

programación, se tiene la flexibilidad necesaria dependiendo de las opciones y los algoritmos que se requieran dentro del sistema sin tener exceso de opciones [1], [2], [7].

LabVIEW (Laboratory Virtual Instrumentation Engineering Workbench) es un ambiente de programación gráfico basado en el concepto de programación de flujo de datos. Este tipo de programación ha sido ampliamente adoptado por la industria, en ambientes universitarios y laboratorios de investigación [3], [6].

Ahora, la ventaja de utilizar un entorno de desarrollo grafico como LabVIEW, en lugar de un ambiente de programación visual, es un ahorro considerable de tiempo, ya que, aunque en los ambientes visuales ya hay objetos predefinidos, muchas veces, es difícil configurarlos; mas cuando no se tiene un gran conocimiento en programación. En LabVIEW, debido a cuenta con numerosas herramientas como los son: algoritmos matemáticos, de procesamiento digital de señales, protocolos de comunicación, tanto para ambientes distribuidos como para comunicar con dispositivos de campo, manejo de tarjetas de adquisición de datos, así como la posibilidad de poder manejar código en C y archivos tipo DLL, entre otras características; hacen que este entorno de desarrollo sea muy potente, productivo y fácil en el desarrollo de sistemas SCADA [4].

2 Muestreo de señales y teorema de muestreo de Nyquist

Para usar técnicas de procesamiento digital de señales, primero se debe representar una señal analógica en forma digital. En la práctica, esto se hace con un convertidor analógico digital (A/D). Considere una señal analógica $x(t)$ que esta muestreada cada Δt segundos. El intervalo de tiempo Δt es conocido como "intervalo o período de muestreo". El recíproco de este intervalo, $1/\Delta t$, es conocido como "frecuencia de muestreo", con las unidades de muestras/segundo (s^{-1}). Cada uno de los valores discretos de $x(t)$ cuando $t=0, \Delta t, 2\Delta t, 3\Delta t, \ldots$, es conocido como muestra. Entonces la señal $x(t)$ puede ser representada por el conjunto discreto de muestras:

$$\{x(0), x(\Delta t), x(2\Delta t), x(3\Delta t), \ldots, x(k\Delta t), \ldots\}$$

Otro parámetro importante es la taza o frecuencia de muestreo con que se adquiere la señal. Una taza de muestreo rápida adquiere más puntos en un tiempo dado y puede dar una mejor representación de la señal original. Una taza de muestreo demasiado lenta, tiene como consecuencia una representación muy pobre de la misma señal, a este efecto se le llama *aliasing*.
De acuerdo al teorema de Nyquist, para evitar el *aliasing*, "la taza de muestreo debe ser por lo menos el doble de la frecuencia de la componente más alta en frecuencia de la señal a muestrear".

Para el diseño del sistema se propuso que cada variable tenga su propia taza de muestreo, esto proporciona flexibilidad al sistema para poder adquirir señales con diversos rangos de frecuencia independientes, además de aligerar al sistema, ya que

no es necesario adquirir una muestra de todas las variables cada período de muestreo, aunque para cada señal, el operador debe elegir un período congruente que cumpla con lo dicho anteriormente y evitar el *aliasing*

3 Filtrado digital

El filtrado es el proceso mediante el cual el contenido de frecuencia de una señal es alterado. Los filtros alteran o remueven frecuencias que no son deseadas. Dependiendo del rango de frecuencia que los filtros dejen pasar o atenúen, se clasifican de la siguiente manera:

- Filtro Pasa Bajas. Deja pasar bajas frecuencias pero atenúa las altas frecuencias.
- Filtro Pasa Altas. Deja pasar las altas frecuencias pero atenúa las bajas frecuencias.
- Filtro Pasa Banda. Deja pasar una cierta banda de frecuencias.
- Filtro Supresor de Banda. Atenúa una cierta banda de frecuencias.
-

La ganancia de un filtro ideal es uno (cero decibeles) en la banda de paso, o sea que la amplitud de la señal ni se incrementa ni decrementa, y cero en la banda de paro (-∞ decibeles). Pero en la práctica siempre hay una región de transición finita entre las bandas de paso y transición, la ganancia del filtro en esta región cambia gradualmente de uno (cero decibeles) en la banda de paso, a cero (-∞ decibeles) en la región de paro [11].

En algunas aplicaciones, es aceptable que en la banda de paso la ganancia varíe muy suavemente de la unidad, a esta variación se le llama "rizo de pasa banda". "La atenuación en el supresor de banda", en la práctica, no puede ser infinita, por lo que se debe especificar un valor aceptable de supresión. Ambos, el rizo de pasa banda y la atenuación del supresor de banda, se miden en decibeles (dB), y están definidos por la ecuación 1.

$$dB = 20 \log_{10} [A_0(f)/A_i(f)] \qquad (1)$$

Donde $A_i(f)$ y $A_0(f)$ son las amplitudes de una frecuencia en particular antes y después del filtrado, respectivamente.

4 Diseño del sistema SCADA

Tomando en cuenta lo anterior se hizo el diseño del sistema, en cual contempla varios instrumentos virtuales, tanto para la adquisición de datos y la parte de filtrado. Este sistema consta de los siguientes módulos de interfaz principales:

- Panel general del sistema.
- Panel de cambio de parámetros.
- Gráfico para visualización de señales.

- Módulo de recolección de datos.

Los instrumentos virtuales anteriores mandan llamar otros, necesarios para el funcionamiento del sistema.

4.1 Panel general del sistema

En la figura 1 se muestra este panel, el cual cuenta solo con tres botones, el primero es para apagar el sistema, el segundo se utiliza para mandar llamar al panel de cambio de parámetros y el tercero es para mandar llamar el programa de visualización general de todo el sistema: los gráficos de visualización de señales, cambio de parámetros de filtrado, cambio de parámetros para la alarma predictiva y señalizaciones de alarma [6].

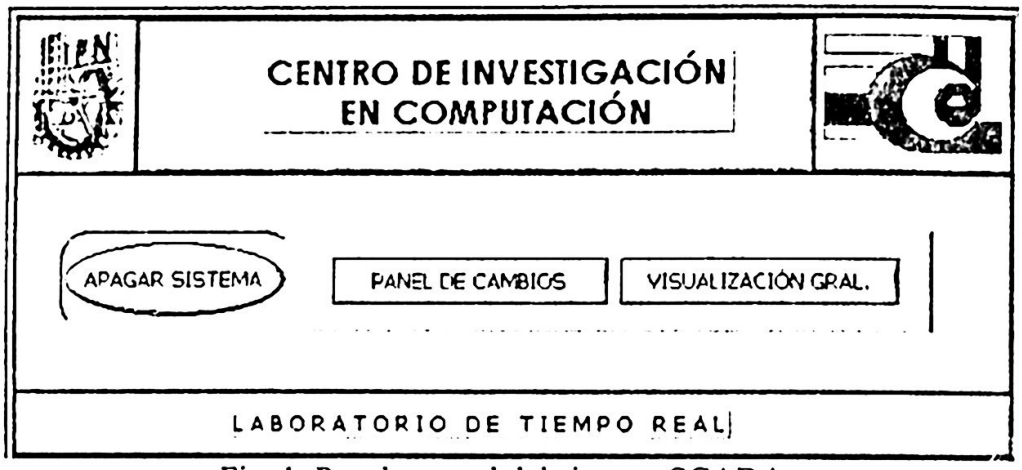


Fig. 1. Panel general del sistema SCADA.

4.2 Panel de cambio de parámetros

Este panel que se encuentra en la Fig, 2, muestra el panel donde se encuentran los parámetros generales, por los que se va a regir todo el sistema. El panel esta dividido en dos secciones principales.

Debido a que el número de variables que puede adquirir el sistema diseñado es en teoría ilimitado; con la única limitante del sistema de cómputo y el tipo de dato del campo (entero sin signo) que define el número de variables que el sistema manejará en total. La primera sección corresponde a las "*VARIABLES CONECTADAS*", aquí se define el número total de variables que manejará el sistema, para esto se tiene el control numérico de la parte superior izquierda, este caso, se muestra el número "6". En la parte inferior a este control numérico se encuentran un arreglo que contiene una serie de conectores, lo cual permite conectar o desconectar dicha o dichas variables, los campos se habilitan dependiendo del número de variables establecidos. A la derecha de estos conectores se puede definir el nombre o tag de la variable correspondiente. Por último, en la parte inferior de esta sección se puede observar un indicador que señala la última variable que se muestra en el panel, por ejemplo, si se llena el panel que se visualiza, de la variable 0 a la 11, se mostrará de la variable 12 a la 23.

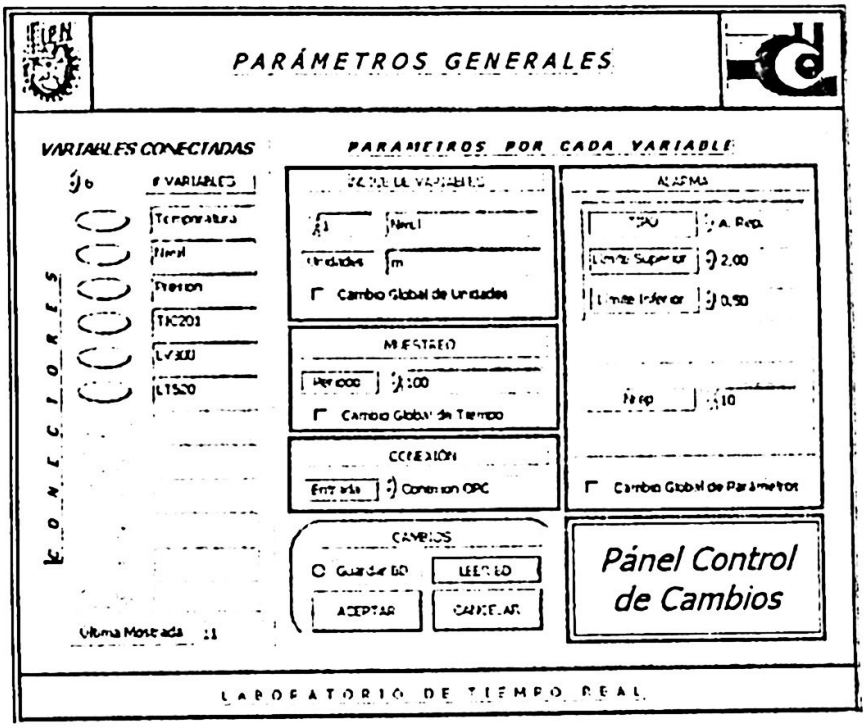


Fig. 2. Panel de cambio de parámetros.

En la segunda sección se muestran los "*PARÁMETROS GENERALES POR CADA VARIABLE*", y se enuncian como sigue:

4.2.1 ÍNDICE DE VARIBLES"

En la sección anterior se muestran todas las variables conectadas, éste como su nombre lo dice, es un índice que muestra la variable actual, nombre y unidades que representa. Por otro lado, hay una opción que hace que las unidades sean las mismas para todas las variables.

4.2.2 "MUESTREO"

En este apartado se puede determinar el período a la que la variable que corresponde al número de índice, va a ser muestreada, esto es independiente por cada variable, pero también se tiene la opción de tener el mismo período de muestreo para todas las variables existentes en el sistema.

4.2.3 "CONEXIÓN"

Este sistema selecciona tres tipos de fuente de datos, puede ser vía conexión ethernet por OPC, mediante reproducción de datos grabados en un archivo de texto o binario, o por la generación de señales; se pueden generar 5 tipos de funciones: senoidal, exponencial de 1^{cr} orden, exponencial de 2do orden en sus tres tipos: subamortiguado, críticamente amortiguado y sobreamortiguado. La fuente más importante de datos es por red, mediante OPC; la opción de reproducción por un archivo binario o de texto sirve como datos históricos para ver sucesos importantes en un determinado tiempo. Y la opción de generación de señales es para hacer pruebas de adquisición de datos y como trabajo futuro.

4.2.4 "CAMBIOS"

Aquí se tienen tres botones y un campo de selección. Con el botón "LeerBD" se pueden leer los parámetros almacenados previamente en un archivo de texto o binario. Con el botón "Aceptar" se aceptan los cambios realizados con redundancia, ya que, el sistema pregunta si realmente se quieren hacer los cambios. Con el botón "Cancelar" se aborta la operación de cambio de parámetros, y con el campo de selección "Guardar BD" se almacenan los parámetros de los cambios realizados, después de dar click en el botón "Aceptar"; aparece un cuadro de diálogo para crear el archivo donde se van a almacenar los datos de la base, y se almacenan en un archivo de texto o binario.

Por otro lado cuando se trabaja con generación de señales, aparece un cuadro de diálogo para seleccionar el tipo de función que se va a generar. Esto se hace por cada variable.

También, se tienen varios subVI adicionales que permiten guardar en un archivo binario o de texto todos los psrámetros contenidos en la base de datos, asi como, poder cargar esos mismos datos guardados.

4.2.5 "ALARMA"

Con los algoritmos de alarma se pueden detectar cuando las variables de un proceso han tomado valores extremos que pueden resultar peligrosos o que deben ser señalizados pues resultan importantes en el funcionamiento del sistema.
En un proceso tecnológico las posibles tendencias que pueden derivarse a partir del estado inicial de la variable y llegar a un límite superior o inferior de alarma (LSA, LIA respectivamente). Por arriba del LSA la región se denomina GAMA ALTA; por abajo del LIA la región se denomina GAMA BAJA, y a la región entre estos dos límites se le nombra GAMA NORMAL. Y a su vez, la variable puede tomar una tendencia con una pendiente positiva, negativa o tener pendiente cero.

Se implementan tres tipos de alarma: por límite simple, por límite con repeticiones y por límite con histéresis.

4.2.6 Algoritmo para obtener el máximo común divisor (MCD)

Como se comento anteriormente, para evitar que el sistema se cargue innecesariamente, cada variable puede tener un período de muestreo diferente, por lo que se tiene que tener un criterio para saber cuando tiene que ser leída cada variable; es por esto que se necesita obtener el máximo común divisor (MCD), y una vez que se han hecho los cambios pertinentes en el Panel de Cambios y se da click en el botón "ACEPTAR", se calcula el MCD del arreglo donde se encuentran los períodos de muestreo de todas las variables. Una vez que se obtiene, se almacena en una base de datos global para que pueda ser utilizado por el módulo de recolección de datos. El algoritmo que se sigue para obtener el MCD es el llamado algoritmo de Euclides, el cual consiste el hacer divisiones sucesivas como se enuncia en las ecuaciones (2) a (7).

Sean **a** y **b** los números de los que se quiere calcular el MCD. Se hacen las siguientes divisiones hasta que el residuo sea cero:

$$\begin{array}{r|l} b & a \\ & q \\ \hline & r \end{array} \qquad (2)$$

$$a = bq_1 + r_1 \qquad (3)$$

$$b = r_1q_2 + r_2 \qquad (4)$$

$$r_1 = r_2q_3 + r_3 \qquad (5)$$

$$r_2 = r_3q_4 + r_4 \qquad (6)$$

$$r_{n-1} = r_n q_{n+1} + 0 \tag{7}$$

Donde q_1, q_2, ... son los cocientes y r_1, r_2, ... son los residuos. El MCD es r_n.

Para calcular el MCD de varios números (a_1, a_2,...,a_n) con el algoritmo de Euclides se procede de la siguiente forma: se calcula el MCD de los números a_1 y a_2. Sea este número M_1. Después calculamos el MCD de los números M_1 y a_3. Sea este número M_2. Repetimos el proceso hasta el último elemento. El MCD de los números M_{n-1}, a_n, será el MCD de a_1, a_2,...,a_n.

4.2.7 Gráfico para visualización de señales

En este gráfico se puede apreciar el comportamiento de las variables que se están adquiriendo, la apariencia del diseño asemeja a un osciloscopio.

El control numérico con el nombre de "Índice" permite cambiar la variable que se esta visualizando. El siguiente control numérico permite variar la velocidad de visualización en el gráfico, este toma por nombre "Actualización". Y por último el botón que dice "Parar" detiene el modulo de visualización de variables.

Se encuentra otro control en forma de pestañas con los nombres "Rango" y "Generación". En "Rango", que son los controles parecidos a las perillas de un osciloscopio, se ajustan las escalas de los ejes X y Y del gráfico. El botón "Guardar" da la posibilidad de almacenar los datos adquiridos en un archivo binario o de texto. Y por último se tienen los gráficos que muestra el comportamiento de la variable, en la Fig. 3 se muestra la opción de filtrado de señales, se usan dos tipos de filtro digital, un IIR y uno de Media con todos sus parametros. Dentro del gráfico se pueden comparara las dos señales, la filtrada y la no filtrada, y también la respuesta del filtro, tanto en magnitud como en fase en el dominio de la frecuencia.

La pestaña de "Generación" se usa solo cuando se ha escogido la opción de generar las señales a visualizar y no se están adquiriendo; las funciones que se pueden generar son: senoidal, exponencial de 1er orden, exponencial de 2do orden en sus tres casos principales: subamortiguado, críticamente amortiguado y sobreamortiguado.

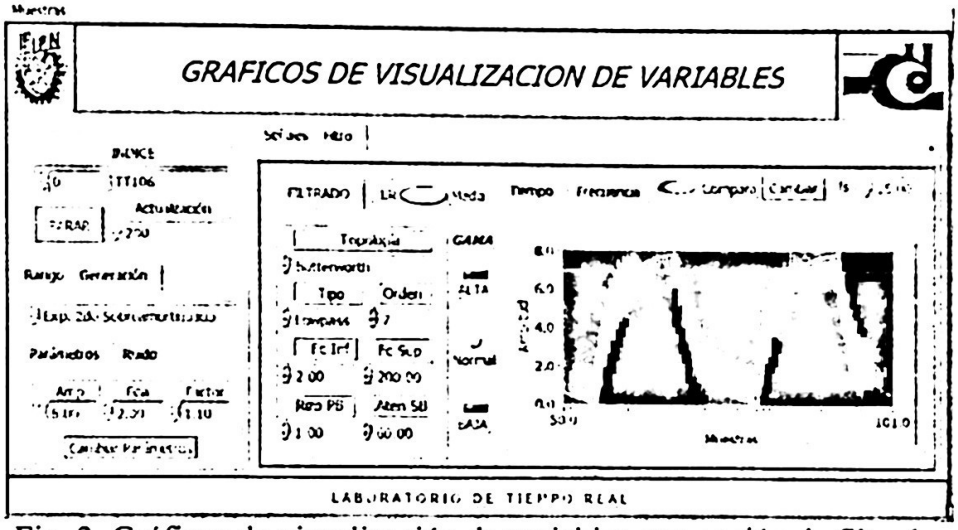


Fig. 3. Gráficos de visualización de variables con opción de filtrado.

Por otro lado tambien se pueden guardar las muestras que se van adquiriendo en un archivo binario o de texto, y a su vez, se pueden cargar los datos guardados previamente.

4.2.8 Recolección de datos

Este módulo se puede decir que es el más importante del sistema, ya que es el que se encarga de leer los datos, ya sea, por una conexión OPC, por reproducción de datos de un archivo binario o de texto, o por la generación de los mismos.

El funcionamiento de este módulo consiste en obtener las muestras de las variables conectadas en los períodos de tiempo especificados; como ya se había comentado, este sistema no requiere que la frecuencia de muestreo sea igual para todas las variables, por lo que se requiere de un valor que sea común para todas las frecuencias de muestreo, este valor es el MCD (máximo común divisor) que se calculó en el módulo del panel de cambios. Lo que hace el algoritmo, es obtener el índice de las variables que se van a leer en ese instante, y esto se hace restando el MCD del período de cada variable constantemente, cada que llega a cero es cuando se manda a leer dicha variable. Por último, una vez que se adquiere el dato solo resta pasarlo por un filtro digital, en caso de ser necesario.

Cabe señalar que en todos los accesos a la base de datos global se usaron semáforos para evitar que en algún momento se pierdan datos por conflictos de sincronización.

4.3 Protocolo de comunicaciones para redes industriales OPC

"OLE para control de procesos" (OPC) esta diseñado para permitir a aplicaciones cliente tener acceso a los datos del piso de una planta de una manera consistente. Con la amplia aceptación de la industria, OPC dará varios beneficios [9], [10], [12]:

- Fabricantes de hardware sólo tienen que hacer un conjunto de componentes de software, para sus compradores y utilizarlos en sus aplicaciones.

- Los diseñadores de software no tienen que reescribir drivers a partir de cambios de características y adiciones a nuevo hardware.
- Compradores tendrán más opciones para desarrollar sistemas de manufactura integrados.

4.3.1 Condiciones necesarias para la puesta en marcha del modelo cliente-servidor mediante OPC

Para que un modelo cliente servidor OPC pueda funcinar se tienen que tomar ciertas medidas, las que se enumeran a continuación [8]:

- Asegurarse que tanto las máquinas cliente como los servidores esten bajo un mismo dominio.
- Configurar adecuadamente de la tecnología DCOM en los servidores como en maquinas cliente.
- Configuración del acceso al registro de windows en caso de tener sistemas operativos windows 2000 o XP, o instalacion del acceso al registro de windows para los sistemas operativos windows 95/98/ME..

4.3.2 DataSocket e implementacion de la comunicación

Dentro del modelo OSI, la capa 6 o de transporte, no solo es una capa mas. Es el corazón de la jerarquía completa de protocolos, la tarea de esta capa es proporcionar un transporte de datos confiable y económico de la máquina de origen a la máquina destino, independientemente de la red o redes físicas en uso. DataSocket funciona en esta capa, y esta construido usando tecnología TCP/IP y activeX/COM [6].

En el caso de este proyecto, como la conexión OPC se basa en la operación de DataSockets. Dentro de LabVIEW ya existen subVI's prediseñados para utilizar estas herramientas.

Estos VI's abren o cierran conexiones. La función <<Open>>, abrirá la comunicación; que está basada en una entrada de URL. Esta entrada puede tener los protocolos HTTP, FTP DSTP u OPC, una vez que esta URL entra, esta función es una referencia DataSocket, la cual se usa de la misma manera que una conexión refnum. Los demás VI's usan esta referencia para realizar otras operaciones.

Para leer datos del servidor, es necesario determinar si el servidor es local o remoto. Una vez que el nombre del servidor esta resuelto y se tiene la conexión abierta con el atributo de leer, se necesitará usar el VI <<Update>> si se quieren leer nuevos datos.

5 Conclusiones

1. La supervisión de procesos mediante técnicas digitales es una tecnología muy difundida actualmente.

2. El técnica de adquisición de datos que permite tener un período de muestreo diferente para cada variable, permite al sistema ser flexible y robusto.
3. En la actualidad en el control industrial el uso de redes de comunicaciones se han vuelto una parte fundamental para su desempeño, es por eso que se han estandarizado un gran número de protocolos de redes industriales basados en TCP/IP

Referencias

1. Jacob, M. (1993). *Industrial Control Electronics*. USA, New Jersey, Prentice-Hall International, Inc., 584p.
2. Johnson, C. (1993). *Process Control Instrumentation Technology*, Fourth Edition, USA, New Jersey, Prentice Hall Inc., 592p.
3. Franklin, G.F., J.D. Powell and M.L. Workman, (1990.). Digital Control of Dynamic Systems, USA, New Jersey, Prentice-Hall.
4. Aström, K.J., and B. Wittenmark. (1984), Computer Controlled System, Theory and design, USA, New Jersey, Prentice Hall Inc., Englewood Cliffs.
5. PACHECO Xavier; *Borland DELPHI 4 Developer's Guide*; Ed. SAMS; 1998; p.p. 653-659.
6. National Instruments; *LabVIEW QuickStart Guide*; 1999.
7. SAUCEDO Flores S; *Sistemas de Control Distribuido*; Escuela Superior de Ingeniería Mecánica y Eléctrica IPN; 1998.
8. SIMMONS Curt; *Guía Avanzada Microsoft Windows Server Configuración;* Ed. Prentice Hall Iberia; Madrid; 2000; p.p. 149-162.
9. NI Developer Zone; *General DCOM Configurations for Windows 95/98/Me.*
10. NI Developer Zone; *Setting Proper Access for Remote OPC Server When Using National Instruments OPC Clients*
11. OPPENHEIM A.V, R.W. Schafer y J.R Buck, *Tratamiento de señales en tiempo discreto*; 2ª edición; Ed. Prentice-Hall, Inc.,Madrid, España, 2000, pp. 873.
12. OPC Technical Overview. Parts of this OPC Overview were estracted from a special OPC Supplement of Startmagazin published in their January/February 1998. The entire OPC Supplement can be viewed at: http://www.startmagazine.com/v2n1opc0.html

Instrumento virtual para la adquisición y evaluación del patrón de radiación de una antena

Víctor Manuel Medina Meneses[1]
victor.medina@satmex.com

Resumen. En el presente trabajo se presenta un instrumento virtual para resolver el problema de la adquisición y evaluación del patrón de radiación de una antena. Se desarrollo un sistema cliente-servidor en donde el servidor, implementado sobre la plataforma RT-Linux, encapsula las funciones para adquirir el patrón de radiación de un analizador de espectros y el cliente, desarrollado en Java, encapsula la GUI. Ambos se comunican a través de TCP/IP por medio de un conjunto pequeño de instrucciones tipo SCPI. Se tiene como modalidad de que en el servidor puede hacer un muestreo en tiempo real de los valores de potencia del analizador de espectros para con estos valores se genere el patrón de radiación.

1 Introducción

Los satélites geoestacionarios por su naturaleza solo pueden ser colocados sobre el ecuador por lo que el número de posiciones orbitales para colocarlos es limitado. En un principio se colocaban satélites geoestacionarios con una separación de 3 grados. Pero con la escasez de posiciones orbitales útiles se han colocado satélites con una separación de 1.5 grados e incluso se han co-ubicado. Debido a lo cerca que conviven los satélites geoestacionarios es necesario tomar precauciones para que una estación terrena que suba señales a un satélite no interfiera con las señales de un satélite contiguo. Por lo anterior las compañías que prestan el servició de capacidad satelital tienen la necesidad de medir el patrón de radiación de las antenas que hacen uso de su segmento espacial, para asegurarse que cumplen con las recomendación S.465-5 de la UIT y la regla 25.209 de la US FCC [1][4], las cuales recomiendan que el patrón de radiación no sobrepase una cierta envolvente.

1.1 Envolvente de evaluación

La definición de la envolvente se expresa en las ecuación (1).

Ganancia-[29- 25LOG (ø)dB	$1 \leq ø \leq 7$	$-1 \leq ø \leq -7$	(1)
Ganancia-[+8]dB	$7 < ø \leq 9.2$	$-7 < ø \leq -9.2$	
Ganancia-[32- 25LOG (ø)] dB	$9.2 < ø \leq 48$	$-9.2 < ø \leq -48$	
Ganancia-[-10]dB	$48 < ø \leq 180$	$-48 < ø \leq -180$	

[1] Estudiante de maestría en ciencias de la computación del Centro de Investigación en Computación del IPN

- Pulsar en el sistema el botón de inicio de adquisición [Get AP] al mismo tiempo que se empieza el movimiento de la antena.
- Una vez adquirido el patrón se debe manipular para que se normalice, centre y si es un patrón de azimut se haga la corrección.
- Se procede a evaluar el patrón con la opción del menú.
- Se imprime un reporte.
- FIN.

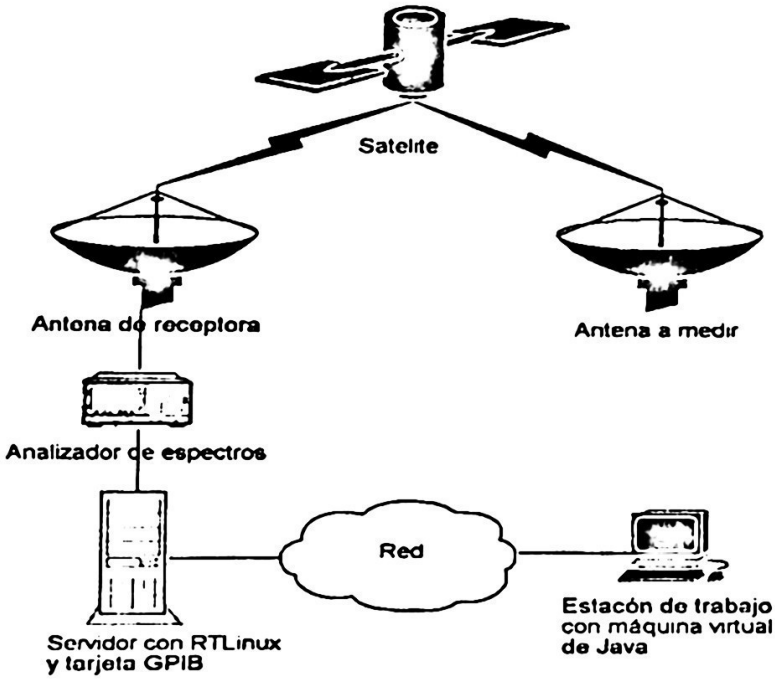


Fig. 2. Interacción entre los diversos equipos para medir el patrón de radiación de una antena

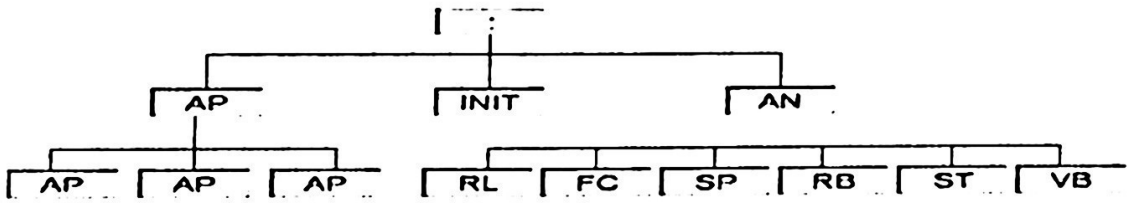


Fig. 3. Jerarquía de comandos SCPI diseñados para la comunicación entre el cliente y el servidor.

2.1 Comunicación entre el cliente y el servidor

La comunicación entre cliente servidor se hace a través de sockets TCP/IP, se diseñaron comandos del tipo SCPI para establecer la comunicación. En la fig. 2 se muestra la jerarquía de estos comandos.

- Pulsar en el sistema el botón de inicio de adquisición [Get AP] al mismo tiempo que se empieza el movimiento de la antena.
- Una vez adquirido el patrón se debe manipular para que se normalice, centre y si es un patrón de azimut se haga la corrección.
- Se procede a evaluar el patrón con la opción del menú.
- Se imprime un reporte.
- FIN.

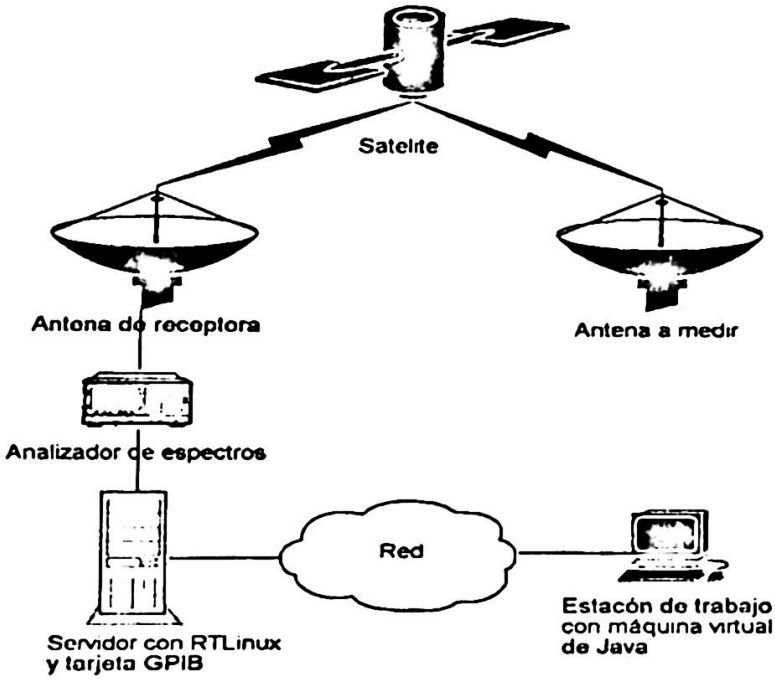


Fig. 2. Interacción entre los diversos equipos para medir el patrón de radiación de una antena

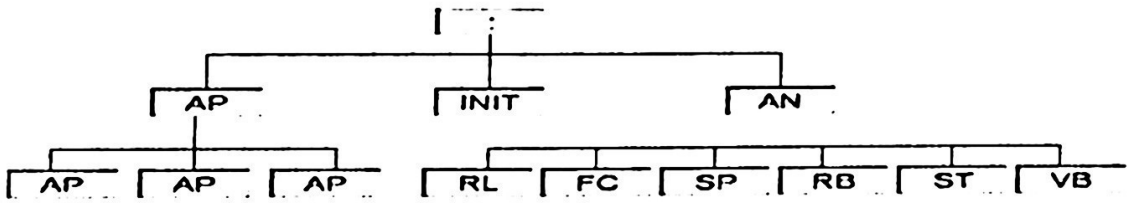


Fig. 3. Jerarquía de comandos SCPI diseñados para la comunicación entre el cliente y el servidor.

2.1 Comunicación entre el cliente y el servidor

La comunicación entre cliente servidor se hace a través de sockets TCP/IP, se diseñaron comandos del tipo SCPI para establecer la comunicación. En la fig. 2 se muestra la jerarquía de estos comandos.

2.2 Bloqueo entre el STR y la graficación en línea (SL)

Para evitar el bloqueo entre la interacción entre el STR y el sistema de graficación en línea se utilizó la siguiente técnica: En la máquina servidora están corriendo dos procesos AP_Server que es un proceso que no es de tiempo real por lo que tiene la prioridad más baja y AP_Muestreo que tiene la prioridad más alta por ser un proceso de tiempo real. AP_Server permanece inactivo hasta que AP_Muestreo coloca en una tubería de tiempo real la duración del muestreo. Entonces AP_Muestreo empieza a tomar de manera periódica una muestra de potencia del analizador de espectros y la coloca en un área de memoria compartida que hace las veces de una cola "queue". Debido a que la escritura en memoria es no bloqueante no corre riesgos de que se AP_Muestreo se bloquee. Durante la diferencia de tiempo entre el plazo "deadline" D y el tiempo de ejecución C de AP_Muestreo la tarea AP_Server que no es de tiempo real va sacando de la memoria compartida "queue" los datos y los envía por medio de sockets TCPIP al cliente (SL) para su graficación, si esta tarea se bloquea debido a cargas en la red o al desempeño del SL, no importa, porque debido a que tiene prioridad baja la tarea de tiempo real continuará su ejecución, así que AP_Server seguirá enviando datos al SL aún después que el muestreo hecho por AP_Muestreo haya terminado. Este sistema de comunicación entre un STR y un SL se puede decir que es basado en una cola "queue" en el STR y mensajes bloqueantes con ayuda de una tarea de baja prioridad. En la fig 4 se muestra un cronograma de los procesos del servidor para el muestreo en tiempo real, dado que el muestreo es una tarea periódica la definen su tiempo de ejecución C_i y su periodo T_i [3].

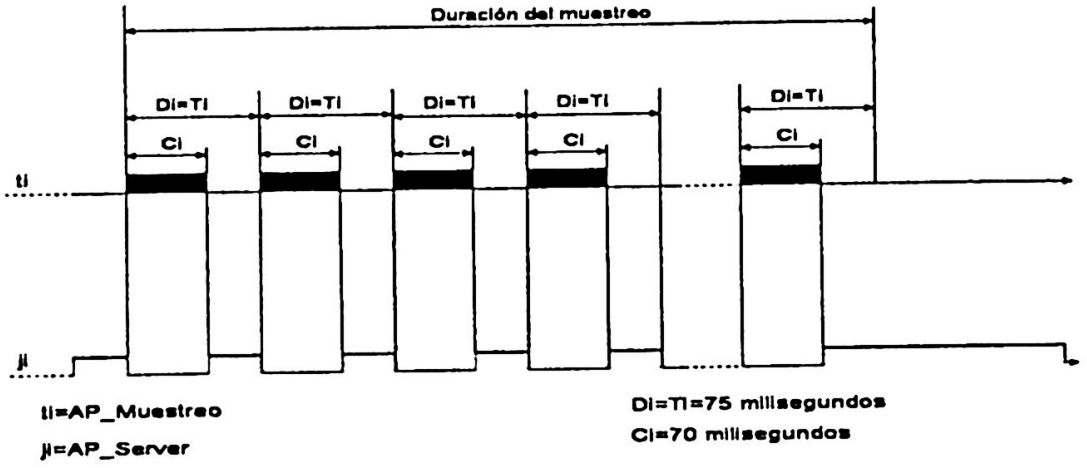


Fig. 4. Cronograma de los procesos en el servidor, el muestreo en tiempo real lo ejecuta la tarea AP_Muestreo la cuál esta dormida hasta que AP_Server le indica la duración del muestreo.

3 Adquisición del patrón de radiación

Una vez sintonizado el analizador de espectros se procede a utilizar la sección de adquisición, en ésta se coloca el tiempo que dura la antena para recorrer el ángulo para el cual se medirá el patrón de radiación. Se selecciona quien hará el muestreo si el analizador de espectros o el servidor (PC). También se selecciona que tipo de

patrón de radiación será si de azimut o de elevación. Una vez que se configura esta información se procede a obtener el patrón de radiación pulsando el botón de [GetAP]. En la fig 5 se observa el resultado obtenido al medir un patrón de radiación de una antena marca Vertex de 7 m de diámetro. Se observa que el patrón no esta centrado, esto es debido a que el movimiento no se empezó al mismo tiempo que el se empezó a tomar muestras, lo anterior no importa debido a que el patrón puede ser manipulado y centrado al máximo.

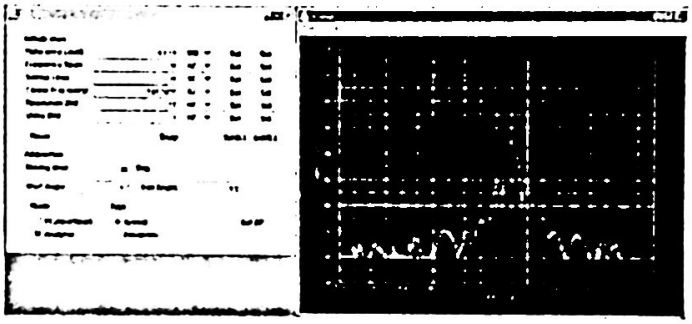


Fig. 5. Patrón de radiación obtenido con el sistema. En la parte izquierda se muestra la GUI inicial y en la derecha la gráfica del patrón de radiación.

4 Evaluación del patrón de radiación

Previo a la evaluación el patrón se debió centrar y corregir (sólo necesario corregir para patrones de azimut) se procede a evaluarlo, en la fig 6 se muestra la evaluación del patrón de radiación. En esta figura se muestran varios cálculos que el instrumento virtual realiza automáticamente y se agregó un nuevo cálculo que es el área del patrón, este nuevo cálculo aunque las recomendaciones no lo piden ahora es posible hacerlo debido a que se cuenta con el poder de computo de las computadoras, anteriormente se evaluaba de manera visual.

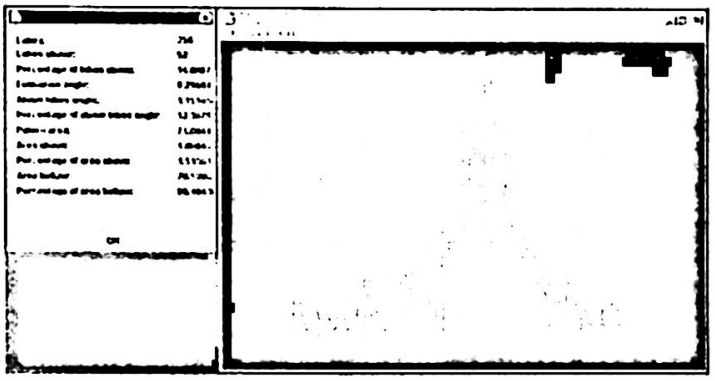


Fig. 6. Evaluación del patrón de radiación con el instrumento virtual. A la izquierda se muestra la evaluación del patrón, el sistema proporciona los lóbulos por arriba de la curva de evaluación, el área por arriba de la curva de evaluación y la suma de los intervalos de ángulos de los lóbulos que se encuentran por arriba de la curva de evaluación.

5 Conclusiones

Se tiene un instrumento virtual para la medición y evaluación del patrón de radiación de una antena, el cuál además automatiza las acciones que se dan en las recomendaciones [1][4].

En la sección 2.2 se utilizó una técnica de comunicación entre un sistema de tiempo real y un sistema en línea que no tiene pérdida de información.

References

[1]Andew Corporation, (1994). "Summary of FCC Rules for Testing Satellite Earth Station Transmit Antennas", Andrew Corporation, pp. 1-4.

[2]Butazzo Guorgio, (1997). "Hard Real-Time Computing Systems Predictable Scheduling Algorithms and Applications". Kluwer Academic Publisers.

[3]Morgan Michael A. and Burt Dennis, (1988). "Field Testing An Earth Station for Two Degree Compliance", Andrew Corporation, pp. 1-7.

[4]UIT, (1993).RECOMENDACIÓN UIT-R S.465-5 "Diagrama de radiación de referencia de estación terrena para utilizar en la coordinación y evaluación de las interferencias, en la gama de frecuencias comprendidas entre 2 y unos 30 GHz", pp. 1-2.

Instrumentos virtuales para laboratorios de oleajes utilizados en el diseño de obras costeras y marítimas

Ing. Fernando Berrelleza[1], Dr. Luis P. Sánchez[1]

[1] Laboratorio de Modelado y Simulación de Sistemas,
Centro de Investigación en Computación - IPN
Av. Juan de Dios Batís S/N, México, D.F. C.P. 07738.
fberrellea@sagitario.cic.ipn.mx
[2] Laboratorio de Modelado y Simulación de Sistemas,
Centro de Investigación en Computación - IPN
Av. Juan de Dios Batís S/N, México, D.F. C.P. 07738.
lsanchez@cic.ipn.mx

Resumen. En este trabajo se presenta el desarrollo de instrumentos virtuales (IV) utilizados en laboratorios de generación de oleaje unidireccional donde se diseñan obras costeras y marítimas tales como: plataformas petroleras, predaplenes, playas artificiales, puentes, obras de protección contra penetraciones de mar y arrastres de arena, entre otras. Es un proyecto en desarrollo y hasta el momento se ha realizado la programación preliminar de IV para adquisición de datos, acondicionamiento digital de la información, visualización gráfica y cálculo de valores estadísticos de oleaje. Se ha usado LabView como lenguaje de programación fundamental.

1 Introducción

Muchas veces la realización de una medida requiere de la intervención de varios instrumentos, unos generan estímulos sobre el dispositivo que se pretende medir, y otros recogen la respuesta a estos estímulos. Este conjunto de instrumentos que hace posible la realización de la medida, recibe el nombre de sistema de instrumentación. La utilización manual de instrumentos para realizar medidas es prácticamente un hecho aislado, sólo en los procesos de investigación y desarrollo de nuevos prototipos o en entornos docentes es una práctica habitual. A nivel industrial las medidas para el control de un determinado proceso, las pruebas funcionales sobre un equipo o el control de la calidad de la producción se realizan de manera automática. Hablar de instrumentos virtuales es hablar de un software que se ejecuta sobre el controlador, que permite independizarnos de los instrumentos reales y de la forma de interconectarlos. El usuario final del sistema de instrumentación solo ve los instrumentos virtuales en el ordenador. La forma habitual para construir un sistema de instrumentación virtual a partir de los instrumentos controlables que tenemos disponibles y de tarjetas de adquisición conectadas directamente al bus del controlador es utilizar un software comercial que facilite esta tarea, en este caso empleamos LabView.

Nuestro sistema de instrumentación está enfocado a los laboratorios de oleaje para el diseño de obras costeras y marítimas, donde el problema en la creación de las mismas, esta dado en la adquisición y procesamiento de datos que resulta difícil y extremadamente limitada. Este sistema de IV's permite la recolección y procesamiento computarizado de la información a través de sensores continuos. Visualización gráfica dinámica y estática.

2 Diagrama de bloques para la adquisición de datos

Podemos adquirir información simultanea de 16 sensores, desde la opción de configuración podemos seleccionar la frecuencia de muestreo, el número de sensores conectados, entre otras funciones. Tenemos tres tipos de dato de entrada: generado, tiempo real y reproducción.

La entrada de datos generados se emplea cuando se requiere tener una respuesta de tipo exponencial o senosoidal, la entrada de datos en tiempo real cuando recolectamos la información desde una tarjeta de adquisición y la entrada de reproducción cuando tenemos ficheros de tipo binario y txt.

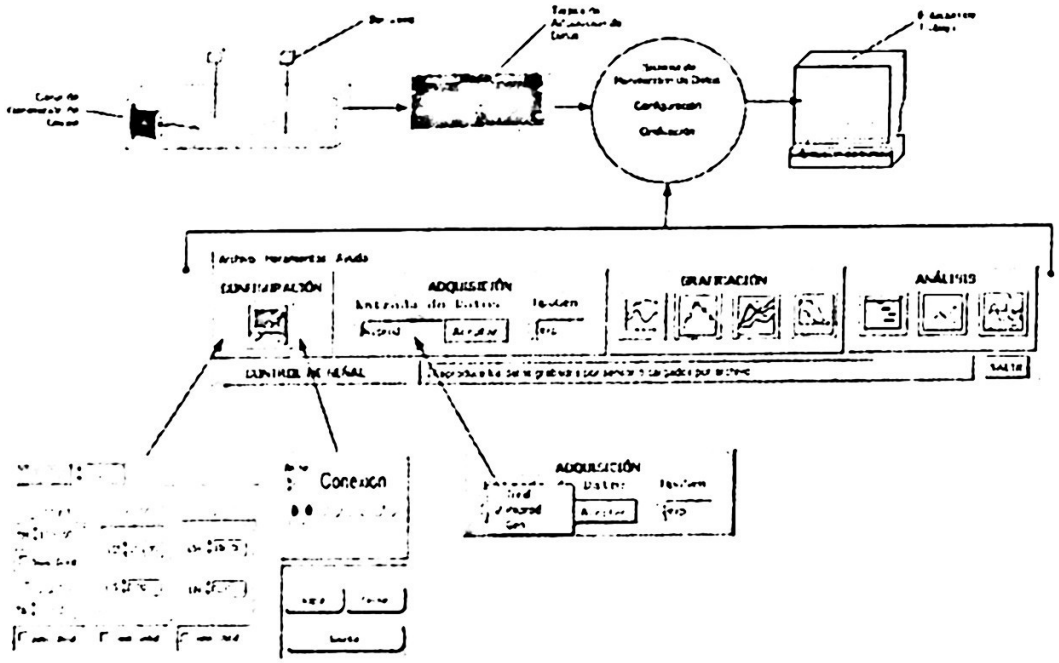


Fig. 1.- Diagrama de adquisición de datos.

3 Acondicionamiento digital de la información

El acondicionamiento digital de las señales es un procesamiento digital primario de las señales muestreadas a fin de dejarlas en condiciones óptimas para su uso por los algoritmos de aplicación tales como: regulación, optimización, análisis estadístico, gráfico, etc.

3.1 Filtrado digital

Con los filtros digitales sólo es posible filtrar ruidos que tengan frecuencias inferiores a la mitad de frecuencia de muestreo. A continuación se plantean las expresiones discretas de tres filtros simples que pueden resultar útiles para la operación en tiempo real y cuando no se requiera una alta selectividad.

Filtro exponencial de primer orden.

$$Y(k) = Y(k-1) + a[X(k) - Y(k-1)] \tag{1}$$

Filtro exponencial de segundo orden.

$$Y(k) = Y(k-1) + A[Y(k-1) - Y(k-2)] + B[X(k) - Y(k-1)] \tag{2}$$

Filtro promedio simple.

$$Y(k) = Y(k-1) + \frac{1}{n}[X(k) - X(k-n)] \tag{3}$$

Donde:

Y(k): salida del filtro en el instante de muestreo *k*.
Y(k-1) : salida del filtro en el instante de muestreo *k-1*.
X(k) : entrada al filtro en el instante de muestreo *k*.
a : parámetro de ajuste del filtro exponencial de primer orden.

$$a = \frac{WcT(-WcT + \sqrt{4 + Wc^2T^2})}{2} \tag{4}$$

Siendo:

Wc : frecuencia de corte de filtro en radianes/segundos.
T : período de muestreo de la señal del sensor en segundos.
A, B : parámetros de ajuste del filtro exponencial de segundo orden.
Y(k-2) : salida del filtro en el instante de muestreo *k-2*.
n : cantidad de muestras a promediar en el filtro promedio.
X(k-n) : entrada al filtro promedio en el instante de muestreo *k-n*.

Como puede apreciarse estos filtros digitales pueden implementarse en sistemas que operan en tiempo real, púes sólo se requiere información en el instante de muestreo actual (*k*) y en instantes anteriores. Estos filtros deben de ser inicializados así como guardar en la base de datos los parámetros y variables anteriores.

3.2 Conversión de unidades de ingeniería

Esto es un procesamiento matemático que permite obtener, a partir de las lecturas del conversor análogo/digital, sus equivalentes en unidades de ingeniería, es decir, en las

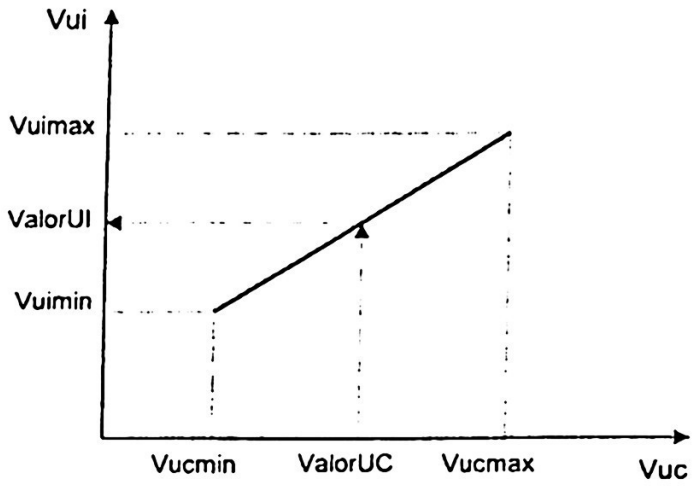
unidades de medición de la variable continua que está siendo medida. De forma matemática se podría plantear que:

ValorUI = F (ValorUC)

Siendo:

ValorUI : el valor de la variable en unidades de ingeniería. Por ejemplo: presión en Kg/cm^2, temperatura en °c, nivel en cm.
ValorUC : es el valor leído en unidades del conversor.
F : es una función lineal.

En la gráfica siguiente se ilustra la relación anterior:



$$\frac{Vui\max - Vui\min}{Vic\max - Vuc\min} = \frac{ValorUI - Vui\min}{ValorUC - Vuc\min} \tag{5}$$

Tomando *Vucmin* = 0 obtenemos:

$$ValorUI = \frac{1}{G} ValorUC + Vui\min \tag{6}$$

Donde:

$$G = \frac{Vuc\max - Vuc\min}{Vui\max - Vui\min} \tag{7}$$

3.3 Interpolación

Se realiza el análisis del efecto de la interpolación de puntos en la serie temporal, compatibles con su vector complejo en el dominio de la frecuencia. Con este método

se añaden puntos a un registro de oleaje, obteniéndose un efecto cercano a lo que se produciría de haberlo muestreado a mayor frecuencia. Puede ser seleccionado el número de puntos a añadir, llamándole factor de relleno (*Fr*), definiéndose de la forma siguiente:

$$Fr = \frac{NPSR}{NPSO} \tag{8}$$

Donde:
NPSR : número de puntos de la serie llenada.
NPSO : número de puntos de la serie original.
Fr : debe ser potencia de base 2 para que pueda ser aplicado el algoritmo de la *Transformada Rápida de Fourier.*

La figura 2 muestra parte de la gráfica del nivel de agua contra el tiempo para la señal de oleaje. Se presenta la serie original muestreada a *3 Hz* y la serie rellenada con factor *Fr = 8.*

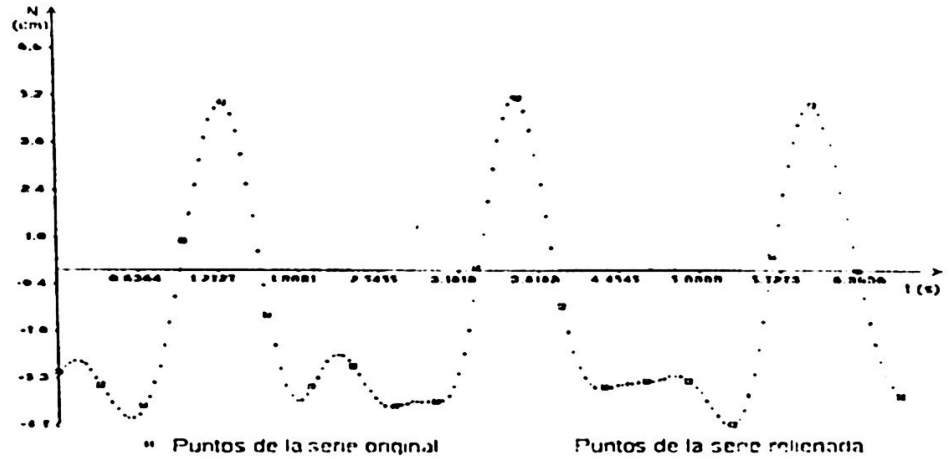


Fig. 2.- Ejemplo de Interpolación Compleja

Con la aplicación del método, se ha reconstruido la serie original, contrarrestando en gran medida la distorsión provocada por una baja frecuencia de muestreo. Se observa que los picos de la señal han sido conformados añadiendo, entre los valores muestreados, puntos los cuales mantienen la compatibilidad con el vector complejo de la serie original (en este caso se añaden 7 puntos entre cada dos muestras). Se nota la conformación de los valores máximos y mínimos, con independencia o no de que exista en dicha posición un punto de la serie original.

En la tabla 1 se presenta un resumen de la media y la varianza de los errores relativos en %, respecto a la señal muestreada a *48 Hz*, de las series interpoladas con diferentes factores de relleno (*Fr*).

Tabla 1. Resumen de la media y la varianza de los errores relativos en %,

	24 Hz	12 Hz	6 Hz	3 Hz
Series Originales	0.48 ±0.08	1.64±1.52	2.61±3.63	3.1±9.02
Series rellenadas 1	(Fr = 2) 0.48±0.08 No hubo mejoria	(Fr = 4) 0.37±0.13 Mejoró 4,4 veces	(Fr = 8) 1.16±0.52 Mejoró 2.3 veces	(Fr = 16) 2.28±2.49 Mejoró 1.4 veces
Series rellenadas 2	(Fr = 4) 0.04±0.002 Mejoro 11 veces	(Fr = 8) 0.03±0.001 Mejoró 55 veces	(Fr = 16) 0.97±0.76 Mejoró 2.8 veces	(Fr = 32) 1.92±3.09 Mejoró 1.6 veces

Se les llamó series rellanadas 1, a aquellas, en las que se utilizaron factores de relleno (*Fr*), de 2, 4, 8 y 16 para las señales muestreadas con 24, 12, 6 y 3 *Hz*, respectivamente. Para las 4 señales el número total de puntos de las series interpoladas fue de 4096, esto significa que la serie original muestreada a 24 *Hz*, tenía 2048 puntos y que la interpolación sólo ha añadido un punto intermedio (entre dos muestras) lo que hace suponer la limitación de poder reconfigurar la serie adecuadamente, para esta señal la interpolación no produjo disminución de los errores. Los errores disminuyeron 4.4, 2.3 y 1.4 veces para las señales muestreadas con 12, 6 y 3 *Hz*, respectivamente.

Se les denominó series rellenadas 2 a las que utilizaron factores (*Fr*) de 4, 8, 16 y 32, para las señales muestreadas con 24, 12, 6 y 3 *Hz* respectivamente. El número total de puntos de las series interpoladas fue de 8192. Para 24 *Hz*, la interpolación adicionó 3 puntos intermedios. Se añadieron 7, 15 y 31 puntos intermedios a 12, 6 y 3 *Hz*, respectivamente. En los dos primeros casos (muestreo a 24 y 12 *Hz*, respectivamente) se aprecian mejoras considerables, con disminución de los errores en 11 y 55 veces y resultados más modestos para las dos últimas señales, con frecuencia de muestreo muy bajas de 6 y 3 *Hz*, respectivamente.

Se observa que una vez interpoladas las series muestreadas a 24 *Hz* y a 12 *Hz*, en ambos casos la media y la varianza de los errores relativos son similares, quedando demostrado que la interpolación hace innecesario para este ejemplo muestrear la señal continua a 24 *Hz*, púes resulta suficiente hacerlo a 12 *Hz* e interpolar con un factor de relleno *Fr = 8*. Se aprecian valores de media y varianza ligeramente menores para 12 *Hz*, se considera que en tal comportamiento influyen factores aleatorios, que dependen fundamentalmente de la forma en que queden distribuidos los puntos muestreados respecto a la señal continua en el tiempo, la cual al ser irregular hace impredecible tal distribución. En pruebas realizadas que no se muestran en este trabajo, adicionando la misma cantidad de puntos intermedios en ambas series, se mantienen las características de los resultados mostrados.

Resulta evidente que el método de interpolación utilizado siempre disminuye los errores y su efecto es significativamente importante para mejorar la exactitud de las mediciones, cuando éstas han sido realizadas con un muestreo cuya frecuencia ha sido bien elegida. Resultan satisfactorios factores de relleno de 8 y 16, para canales donde se

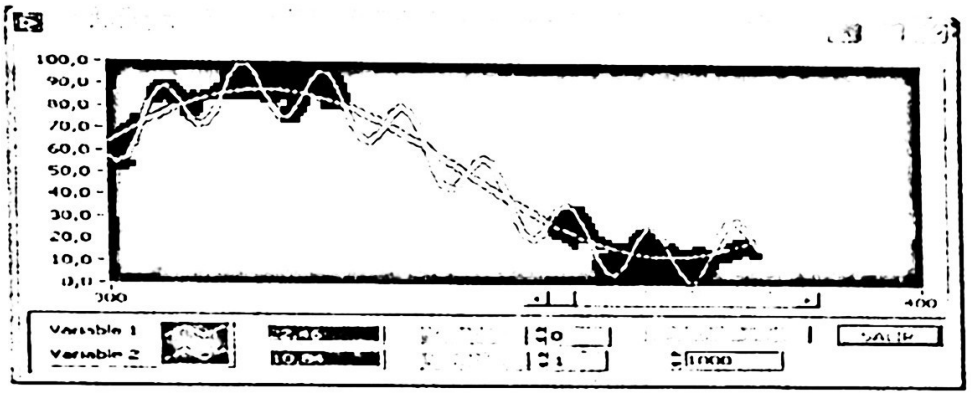
generan oleaje con características similares al analizado y cuyo espectro fue presentado en la ilustración x.

En todas las pruebas realizadas se han mantenido constante el número de olas presentes en el registro de oleaje (35 olas) para que en los errores de los cálculos estadísticos sólo influya la frecuencia de muestreo.

4 Visualización gráfica

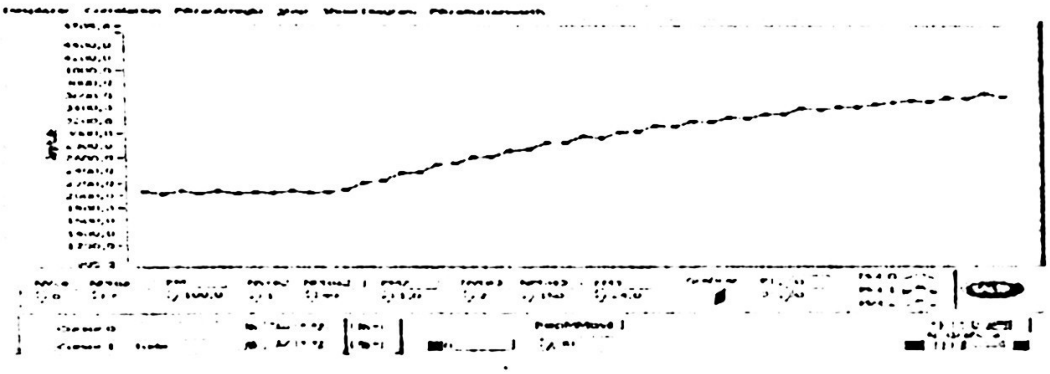
La visualización gráfica nos permite analizar el proceso de recolección de datos en cualquiera de sus entradas, contamos con dos tipos de graficación: dinámica y estática.

4.1 Gráficos dinámicos



La graficación dinámica nos permite visualizar el estado de los sensores ya sea en reproducción, generado o tiempo real, para describir el análisis de datos, permite la selección de dos variables y la actualización de la misma.

4.2 Gráficos estáticos



La graficación estática permite visualizar el efecto de la interpolación donde se realiza el análisis de puntos. Permite la selección del número de puntos y la frecuencia de muestreo de la variable.

5 Análisis estadístico de oleaje

Considérese como punto de partida que se tiene un registro temporal de la superficie del mar en un punto dado. Se tendrá, una muestra del oleaje y se admitirá que este registro representa un estado de mar, donde las variaciones en el tiempo con respecto a su nivel medio vienen dadas por *n(t)*.

5.1 Corrección del nivel medio

Los registros del oleaje contienen la influencia de la variación de la marea u ondas largas y, normalmente, están referidos al fondo, por tanto, es necesario realizar una corrección del nivel medio. Esta corrección bien puede ser la resta aritmética del promedio de la señal o una línea recta o una parábola, siendo estas dos últimas las más convenientes.

En este trabajo se decidió utilizar la corrección parabólica la cual ha demostrado tener mejores resultados para el caso de oleaje.

La corrección de nivel medio del agua haciendo uso de una parábola se deduce también a través del método de mínimos cuadrados, y se puede expresar de la siguiente manera:

$$\bar{\eta} = B_0 + B_1(\Delta t \circ n) + B_2(\Delta t \circ n)^2 \tag{9}$$

5.2 Método de pasos ascendentes por cero

Las ondas individuales pueden ser identificadas usando el criterio de pasos ascendentes por cero o pasos descendentes por cero, entre otros, aunque el primero es el más utilizado en la actualidad. Este método define que una onda tiene su inicio cuando la superficie libre del agua cruza el nivel medio de forma ascendente, y termina en el siguiente cruce ascendente.

El punto exacto donde la superficie libre cruza el nivel medio es localizado por interpolación lineal entre dos puntos más cercanos al cero. Matemáticamente el cruce se define mediante las siguientes ecuaciones:

$$h_i \cdot h_{i+1} > 0 \quad \text{y} \quad h_{i+1} > 0 \tag{10}$$

Donde h_i es el *i-ésimo* punto de la elevación de la superficie libre.

Una ola individual es definida usando dos puntos adyacentes por cero. El período de ola de una ola individual, T_i, es la duración entre dichos pasos ascendentes. El valor

más alto de los puntos en una ola individual es encontrado mediante una comparación entre los valores de h_i incluidos.

El valor mínimo de la elevación de una ola individual, h_{min}, puede ser encontrada de forma similar. La correspondiente altura de ola, h_i, de una ola individual, esta dada por:

$$H_i = \eta_{max} - \eta_{min} \tag{11}$$

Este procedimiento se realiza con cada una de las olas del registro, obteniéndose una secuencia de parejas de datos de altura y período de ola. Estos datos pueden ser analizados por medio de técnicas estadísticas.

La altura de ola media, $\overline{H}$, y el período de ola medio, $\overline{T}$, pueden ser calculados por medio de las siguientes ecuaciones:

$$\overline{H} = \frac{1}{N_0}\sum_{1}^{N} H_i \tag{12}$$

$$\overline{T} = \frac{1}{N_0}\sum_{1}^{N} T_i \tag{13}$$

Donde N_0, es el número de olas individuales con período de ola y altura de ola identificados por T_i y H_i respectivamente.

La altura de ola cuadrática media, *Hrms*, esta definida por:

$$H_{rms} = \sqrt{\frac{1}{N_0}\sum_{i=1}^{N_0} H_i^2} \tag{14}$$

Para determinar la altura de ola significante, H_s, y el período de la ola significante, T_s, se ordenan los datos (H_i ó T_i) de forma descendente. La altura de ola significante, H_s, se define como la media aritmética del tercio de las olas más altas.

$$H_s = \frac{3}{N_0}\sum_{r=1}^{N_0/3} Hr \tag{15}$$

Donde H_r, es la altura de ola del rango *r-ésimo* (ordenados de mayor a menor). De manera similar el período de ola significante, T_s, está dado por:

$$T = \frac{3}{N_0}\sum_{r=1}^{N_0/3} T_r \tag{16}$$

Adicionalmente el promedio de la altura y el promedio de ola de un décimo de las olas más altas H_{10} y T_{10}, respectivamente, esta dado por:

$$H_s = \frac{10}{N_0} \sum_{r=1}^{N_0/10} Hr \quad \text{y} \quad T = \frac{10}{N_0} \sum_{r=1}^{N_0/10} T_r \tag{17}$$

6 Conclusiones

En las pruebas realizadas se ha mantenido constante el número de olas presentes en el registro de oleaje, para que en los errores de los cálculos estadísticos sólo influya la frecuencia de muestreo. Es evidente que a un mayor número de olas, es decir mayor tiempo de toma de muestras, será superior la probabilidad de reducir el error en los cálculos estadísticos, pues éstos se basan en utilizar las olas mayores, las cuales pueden ser detectadas con mayor probabilidad cuando se toma un elevado número de muestras.

Referencias

1. Sánchez Fernández, L. Pastor., Error de Cuantificación y Frecuencia de Muestreo. Centro de Investigación en Computación – IPN, México
2. Goda, Y.: Random Seas And Design Of Maritime Structures, Segunda Edición. World Scientific Publishing Co. Singapore (2000).

Correlation of gradients and the algorithm susan for motion estimation and tracking

R. Aburto G.[1,2], A.C. González G.[2,3], J.H. Sossa A.[3]

[1]Centro de Estudios Tecnológicos Industrial y de Servicios # 34, Cd. Lázaro Cárdenas, Michoacán México.
raburto@sct.gob.mx
[2]Instituto Tecnológico de Toluca, Depto. de Ingría. Eléctrica y Electrónica
Av. Instituto Tecnológico S/N Ex Rancho La Virgen, Metepec México, C.P. 52140
alaing@sagitario.cic.ipn.mx alaing@ittoluca.edu.mx
[3]Instituto Politécnico Nacional, Centro de Investigación en Computación
Av. Juan de Dios Batíz esq. Miguel Othón de Mendizábal, C.P. 07000 México, D.F.
hsossa@cic.ipn.mx

Abstract. The motion estimation problem is important when the basic objective is to determine the spatial localization of mobile objects inside images, the motion detection is basically based on brightness changes in an images sequences. In this work present a correlation of space-temporal gradient and the algorithm SUSAN for determine the motion estimation and the direction.

1. Introduction

The motion analysis in an image sequences is growing in many application. Some of them are: Mobile robotics, Satellite image application, Objects tracking, Autonomous navigation, Medical image processing, Virtual reality and Surveillance applications.

Tracking the objects in a images sequences is a common procedure, when we wants to know if an objects inside of the image sequences is moving and the direction they follow; if there are moving some objects, there will be changes in the image pixels brightness.

2. Motion estimation

There are many techniques available for motion estimation, the main groups are: Gradients, Correlation, and Frequency based methods.

2.1 Gradients

Motion estimation based on gradient is the technique most using actually, this is due to a good estimation of the motion field. Methods based on gradients usually presents some restriction, normally before start the processing they require filtering the images. One of the most gradients based method is Horn Shunk [1], this methods applied a global smoothness to the motion field, while the Lucas and Kanade [2] applied

a local smoothness to the motion field, while the common disadvantage of both gradients methods is the change in brightness.

2.2 Correlation

One way to avoid limitations of the gradients based methods is to consider image regions instead of motion estimation pixel by pixel. In general, correlation based methods are less sensible to noise; due to they take more information from images in motion estimation process.

The techniques based on correlation minimize the difference between two block of pixels located in images I_t and I_{t+1} belong to an images sequence [3]. Basically the correlation based algorithm divide the image in series of block with same dimensions. For every blocks in the image I_t, applied a search in the image I_{t+1}, looking for a correlation that minimize a match function.

If block size is fixed constant, the motion estimation field is limited due to be not possible to search for blocks that contents various movements. Some researches still working for solving this problem in multiresolution block correlation [11]. Others are using the Hough Transformer for motion estimation [4][5].

One disadvantage that shares the correlations methods is that they depend to the translation model. The blocks must be small enough in order to consider the translational model valid.

2.3 Frecuency domain

One of the must used methods in the frequency domain are the methods based on the phase. This takes de advantage that a change in the spatial domain produces a change in the phase in the frequency domain [6].

Techniques based in the phases are in use in order to estimate the motion [7]; in this methods a gradients based technique is apply to the phase component of the filters output, sintonized to different velocities.

There is also a technique in the frequency domain based in the correlation [8].
The techniques based in the frequency domain have some advantages over methods in space domain, due to they are less sensible to the global changes in brightness and they have good response against the noise.

3. SUSAN

The algorithm SUSAN is a filter that can be used as corner and edge detector of the objects contain in a digital images, it was development by Smith in 1997 [12]. The acronym of SUSAN is "Smallest Univalue Segment Assimilating Nucleus". The algorithm SUSAN can be used as a filter to reduce the noise in the images, one important

application of this algorithm is in searching of corners and edges of the objects contain in the images.

The SUSAN principle is based on the circular mask, having a center pixel, which shall be known as the nucleus. The brightness of each pixel within a mask is compared with the brightness of that mask's nucleus.

This area of the mask shall be known as the USAN (Univalue Segment Assimilating Nucleus), this concept of each image point having associated with it a local area of similar brightness is the basis for the SUSAN principle. The local area or USAN contains much information about the structure of the image. It is effectively region finding on a small scale. From the size, centroid and second moments of the USAN two dimensional features and edges can be detected.

The area of an USAN conveys the most important information about the structure of the image in the region around any point in question. The USAN area is at a maximum when the nucleus lies in a flat region of the image surface, its falls to half of this maximum very near a straight edge, and falls even further when inside a corner. It is this property of USAN's area, which is used as the main determinant of the presence of edges and corners in two-dimensional features.

This algorithm use two thresholds g and t, where g is call geometric threshold and t is the contrast threshold, the geometric threshold clearly affects the quality of the corners detected, and t affects the numbers of corners found.
Due to the fact that SUSAN does not use image derivatives in the search of corners and edges it has good results in presence of noise.

SUSAN analyses different region separately and made local measurement finding the place where the boundaries of two regions has the function in his minimum value, and found the corner between this two regions. There are not conditions about the image structure for using the algorithm SUSAN

4. Implementation

The system is basically formed for three processing image modules.

The first module is the algorithm in charge of calculating the special-temporal gradients of one image sequence [9], the output of this module is the objects edges that move from image I_t to image I_{t+1}, both belonging to an image sequence.

Second module is the algorithm SUSAN, applied as a corner and edges detector, it find the objects corners and edges contained in image I_t and image I_{t+1} and determine the point that change the position between both images.

The third module find the correlation between the space-temporal gradients and the corners and-edges found after applied the algorithm SUSAN.

After found the correlation of both algorithms, the global center of gravity is calculate for the space-temporal gradients and also for algorithm SUSAN and a comparison is made in order to determine the direction of the motion estimation, figure 1.

4.1 Detection of moving objects edges based on space-temporal gradients

One-way of detecting the moving objects edges in an images sequence or built the optical flow is trough the space-temporal gradients [3], equation 1.

$$E(x,y,t)=dF(x,y,t)/ds*dF(x,y,t)/dt \quad (1)$$

Where dF(x,y,t)/ds is the spatial gradient and dF(x,y,t)/dt is the temporal gradient and the symbol * is the AND operation between them.

For detecting the edges of the objects that move from image I_t to image I_{t+1}, the gradient based Sobel filter is used, equation 2.

$$\Delta f=[Gx\ Gy]^T \quad (2)$$

The gradient absolute value is calculated with equation 3.

$$|\Delta f|=|Gx|+|Gy| \quad (3)$$

The angle of the gradient vector is calculated with equation 4.

$$A(x,y)=\tan^{-1}(Gx/Gy) \quad (4)$$

The gradient in an image is determined calculating the partial derivative in the x (df/dx) and in the y (df/dy) direction in every pixel position. The form to implement the partials derivatives in digital image is using the mask of the Sobel filter in both directions, figure 2.

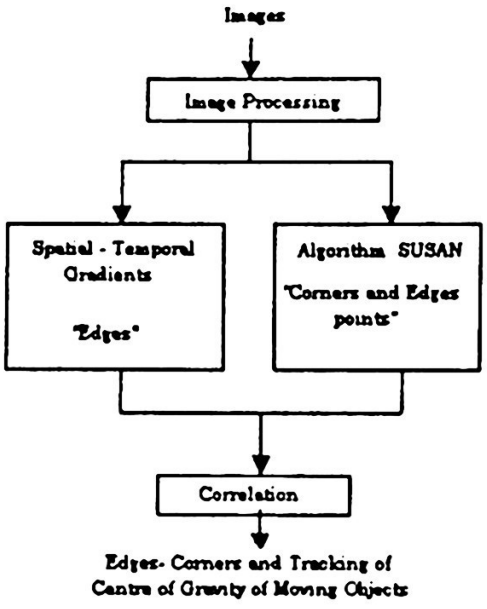


Fig. 1. Algorithm.

$$\text{Direction x} \quad \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

$$\text{Direction y} \quad \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

Fig. 2. Mask of Sobel filter.

After calculation of gradients magnitude and it's angle, the pixel in the center of the mask takes these values, this process is repeated for all the pixels of the images, and the gradients results are storage in a vector. This procedure is used in all image sequence.

For determining the objects inside the images, that move from image I_t to image I_{t+1} is necessary the following procedure:

- Calculate the difference between the spatial gradients of images I_t and image I_{t+1} in order to obtain the temporal gradient.
- The temporal gradient obtained in the prior step is multiply (AND operation) by the spatial gradient of image I_t
- The values obtained of gradients AND operation should be normalized from 0 to 255, after that a threshold has to be used in order to show only the objects edges that experiment a motion from the image I_t to image I_{t+1}.

4.2 Algorithm SUSAN

The algorithm SUSAN is implemented through the following procedure:

1. - Places a circular mask around the pixel in question.
2. - Using equation 4.5 calculate the number of pixels within the circular mask, which have similar brightness to the nucleus. These pixels define the area USAN.

$$\begin{aligned} C(r,r_o) &= 1 \text{ if } |I(r) - I(r_0)| \leq t \\ &= 0 \text{ if } |I(r) - I(r_0)| > t \end{aligned} \tag{5}$$

Equation 5 gives good results, but a much more stable and sensible equation to use for C is equation 6.

$$C(r,r_o)=e^{-((I(r)-I(ro)/t)6} \tag{6}$$

Where:

$I(r)$ is the brightness of the nucleus

$I(r_o)$ is the brightness of any other pixel inside the mask

t determines the minimum contrast of features, which will be detected, and also the maximum amount of noise, which will be ignored.

3. - Obtain the difference between USAN and the geometric threshold.

$$R(r_o)= g - n(r_o) \text{ si } n(r_o) \\ = 0 \text{ otherwise} \tag{7}$$

Where $n(r_o) = \Sigma C(r,r_o)$.

4. - Storage the corners found in a vector.

5. - Test for false corners detection. This can occur with real data where blurring of boundaries between regions occurs. Calculating the center of gravity of the USAN and the distance from the center of gravity to the nucleus can eliminate this problem. Clearly an USAN corresponding to a proper corner will have a center of gravity that is not near the nucleus while a thin line passing through the nucleus will have a short corresponding distance from center of gravity to the nucleus.

4.3. Correlation

If M represents a matching function which return a value proportional to the match of two given features, such as the absolute difference between the two pixels intensity values (E_1 and E_2), then the match strength M(x,y;u,w) for a point (x,y) and displacement (u,w) is calculated by taking the sum of the match values between each pixel in the displaced region in the image I_t and corresponding pixel in the in the actual region of the image I_{t+1}, the match function is calculating using equation 8.

$$M(x,y;u,w) = \sum\Phi(E_1(i,j) - E_2(i+u,j+w)) \tag{8}$$

The actual motion of the pixel is taken to be that of the particular displacement with the maximum neighborhood match strength, that is equivalent to the minimum region difference.

In this particular work, a correlation was applied to the results of space-temporal gradients and the corners found by the algorithm SUSAN. A 5x5 mask center in the position of the corners is displaced for all corner's vector, searching in the same position of the space-temporal gradients image, giving as a results only the corners that match with the edges of the moving objects found by the space-temporal gradients.

5. Results

Figures 3 and 6 show two images belonging to a monocular image sequence, after applied the space-temporal gradients algorithm, the image in figure 4 was obtained. Figure 5 show the image of the corners and point of the edges found by the application of the algorithm SUSAN.

Figure 6 show the image that combine both algorithms and figure 7 and 8 shows the trajectory that follows the global center of gravity, calculated for around 20 images belonging to the sequences.



Fig. 3. 1st. Image



Fig. 4. Gradient



Fig. 5. Corners

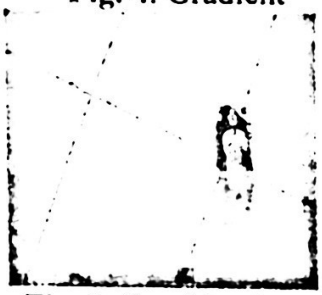


Fig. 6. Combination



Fig. 7. Trajectory



Fig. 8. Trajectory

Both centers of gravity were calculated after application of region correlation to the resulting of both algorithms, and projected over one of the images.
The application of correlation to both algorithms belonging to the same image sequence, reinforce the motion estimation, and the following of the moving objects trajectory is more accurate, during the time interval that the images were taken.}

The application of both algorithms gives good results and can be used in tasks like the surveillance.

References

[1] Horn B. K. P. & Schunck B. G. "Determining optical flow"; Artificial Intelligence, 17, 1981, pp. 185-203.

[2] Lucas B. & Kanade T. " An iterative image registration technique with an application to stereo vision", proceeding DARPA, Image Understanding Workshop, 1981, pp. 121-130.

[3] Jain J. R. & Jain A. K. "Displacement measurement and its application in interframe image coding", IEEE Transactions on Communications, vol. 29, No. 12, 1981, pp. 1799-1808.

[4] Adiv G., "Determining tree-directional motion and structure from optical flow generated by several moving objects", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 7, no. 4, 1985, pp. 384-401.
[5] Baber & Kitter J. , "Robust motion analysis" Proceeding CVPR'94, Seattle, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1994, pp. 947-952.
[6] Popoulis A. Signal Analysis, Mc Graw Hill, 1984.
[7] Fleet D. J. & Jepson A. D. "Computation of components image velocity from local fase information", International Journal of Computer Vision, 5, 1990, pp. 77-104.
[8] Calway A. D., Knutsson H. Wilson R., "Multiresolution estimation of 2-D disparity using a frequency domain approach", Proceeding of British Machine Vision Conference, 1992.
[9] Jain, Ramesh, Rangachar Kasturi y Brian G. Schunck, Machine Vision, Singapore; Mc Graw-Hill Book Co., 1995.
[10] Kearney J. K. , Thompson W. B. & Boley D. L., " Optical flow estimation: an error analysis of gradient-based methods with local optimization", IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 9, No. 2, 1987, pp. 229-244.
[11] Burt P. J. & Adelson E. H., "The Laplacian pyramid as a compact image code", IEEE Transaction on Communication, vol. 31 no. 4, 1983.
[12] Smith, Stephen M. "SUSAN", Low Level Image Processing, 1997.
[13] H. Liu, T. Hong, M. Herman, R. Chellapa, "A general Motion Model and Spatio-Temporal Filters for Computing Optical Flow". International Journal of Computer Vision, 1996.
[14] G. D. Hager and P. N. Belhumeur. Real-Time Tracking of Image Regions with Changes in Geometry and Illumination. IEEE Proc.. of CVPR, 1996.
[15] J. M. Letang, V. Rebuffel & P. Bouthemy, "Motion detection robust to perturbations" A statistical regularization and temporal integration framework". In Proc. ICCV, 21-23, 1993.
[16] L. Wixson, "Illumination assessment for vision-based traffic monitoring," In Proc. ICPR, 57-60, 1996.
[17] F. Bremond & M. Thonnat, "Tracking multiple non-rigid objects in a cluttered scene," In Proc. SCIA, 1997.
[18] S. P. Liou & R. C. Jain, "Motion detection in spatio-temporal space," CVGIP 45: 227-250, 1989.

Control y Tiempo Real

Diseño vía LMIs del generador difuso de residuales para detección de fallas en un sistema de nivel de tres tanques

Efraín Alcorta García[1] y Salvador Saucedo Flores[2]

[1] Posgrado FIME-UANL; C.U. San Nicolás de los Garza, N. L. ealcorta@iecc.org
[2] ESIME Z.-IPN Edif. 5 3er piso, UPALM, Deleg. GAM, D. F. ssaucedo@iecc.org

Resumen. Una de las dificultades para detectar y aislar fallas en sistemas no lineales a través de métodos basados en observadores es el diseño del generador de residuales. En este trabajo se propone un procedimiento para resolver el problema de la detección y el aislamiento de fallas en sistemas no lineales. El resultado se obtiene al derivar varios subsistemas, cada uno de los cuales contiene información acerca de fallas específicas, de la planta en consideración. El modelado Takagi-Sugeno se usa para diseñar los observadores difusos y generar los residuales. Una de las ventajas del método es que el diseño es laxo. Además, se refuerza el uso potencial de técnicas de desacoplo. La metodología del diseño se basa en desigualdades lineales de matrices (LMIs) y se ilustra usando un sistema de laboratorio de tres tanques.

Abstract. One of the difficulties to detect and isolate faults in nonlinear systems via observer based methods is the design of a residual generator. In this work a procedure to approach the fault detection and isolation problem in nonlinear systems is proposed. The result is obtained by deriving several subsystems, each of them containing information about specific faults, from the plant under consideration. Takagi-Sugeno modeling is used to design fuzzy observers and to generate residuals. The detection and isolation tasks is solved by evaluating these residuals. The advantage of the proposed approach is the relaxed condition design for the observers. In addition, the potential use of decoupling techniques is reinforced. Intense use of linear matrix inequalities (LMI) was necessary for the design. A three tank level system illustrates the methodology.

Keywords: Fault detection, nonlinear systems, fuzzy logic, Takagi-Sugeno, linear matrix inequalities (LMI).

Introducción

Los componentes de un sistema físico (incluyendo transductores y actuadores) están sujetos a cambios inesperados, o fallas, las que producen desviaciones desde las condiciones nominales. Para mantener confiable al sistema y económicamente rentable se ha realizado una gran cantidad de investigación en el campo de diagnóstico de fallas, como se puede ver en [3]. Muchos de los métodos realizan la detección de fallas y el aislamiento de las mismas en dos pasos: primero, señales que

sólo dependen de las fallas son generadas. Dichas señales se les llama residuales; y el segundo paso, que consiste en la evaluación de los residuales, esto es, obtener información acerca de las fallas.

Uno de los métodos más efectivos para la generación de residuales se basa en observadores [4]. La idea básica para obtener el residual es comparar la salida medida actual del sistema y el valor nominal (calculada con la ayuda de observadores). Notemos que si el sistema en consideración es no lineal, el observador requerido pudiera ser también no lineal.

En general, el diseño de observadores no lineales no es tarea fácil, incluso si el sistema no lineal es completamente conocido. Para evitar las dificultades relativas al diseño del observador, se puede usar un método no basado en observadores [2], o bien usar otro tipo de observadores [5]. El método en [2] tampoco requiere un modelo matemático de la planta, en cambio, sub modelos basados en identificación se obtienen a través de datos experimentales. Si un modelo de la planta está disponible, el método de [5] se puede usar. Empero, el método considerado en [5] usa observadores Takagi-Sugeno y evaluación difusa. Un tratamiento diferente se aplica en este trabajo. Aquí también se emplean observadores Takagi-Sugeno para obtener los residuales, sin embargo, los residuales generados son procesados en la manera clásica, ie, como los residuales generados con observadores no lineales. El esquema propuesto permite usar el método para desacoplamiento no lineal en conexión con la generación de residuales. Ello representa una ventaja, porque normalmente el método de desacoplamiento produce subsistemas muy complicados y un observador no lineal no es fácil de diseñar.

Este trabajo se presenta como sigue: En la sección 2 se explican algunos antecedentes; el método propuesto se presenta en la sección 3; Un ejemplo de aplicación se halla en la sección 4 y las conclusiones se dan en la sección 5. El apéndice A revela detalles del diseño.

1. Antecedentes

Se revisarán algunos conceptos relevantes para este trabajo. Primero, se presentan los conceptos de detección de fallas y de aislamiento de fallas. Luego se muestra el modelado Takagi-Sugeno de sistemas no lineales y el diseño del observador en dicho contexto.

1.1 Generación de Residuales para Detectar y Aislar Fallas

Como se dijo, la detección y el aislamiento de fallas se llevan al cabo en dos pasos [3]:

1. **Generación de residuales.** El primer paso consiste en la obtención de señales que son sensibles sólo a las fallas. Estas señales son llamadas *residuales.* En forma ideal, los residuales deben ser cero en el caso libre de fallas y diferentes

de cero en otro caso. Un residual puede ser formado como la diferencia de la medición actual de la salida y de un estimado del valor nominal de la salida. Ver por ejemplo la figura 1. En tal figura el residual es generado usando un observador no lineal. Este observador se diseña en base al modelo nominal, esto es, para el sistema libre de fallas.

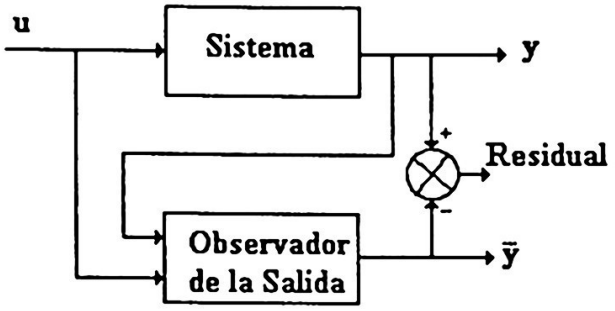


Fig. 1. Generación del residual.

2. **Evaluación de residuales.** En este paso, la información de las fallas contenida en el residual es extraída. Esta operación se realiza generalmente usando normas de las señales y un umbral que permita evitar falsas alarmas.

1.1. Desacoplamiento de fallas

La idea fundamental es encontrar una transformación (quizá no lineal) de modo que se logre sensitividad a una falla específica. Ver la figura 2. El procedimiento usado fue propuesto en [6], ver también [1].

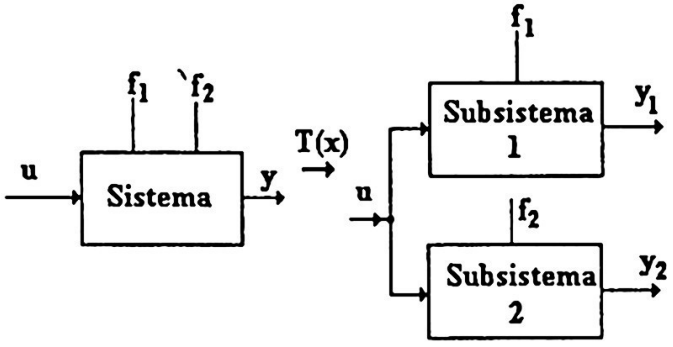


Fig. 2. Desacoplamiento de fallas

Considerar un sistema no lineal

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}) + g(\mathbf{x}) + E_1(\mathbf{x})f_1 + E_2(\mathbf{x})f_2 \qquad (1)$$

$$\mathbf{y} = \mathbf{h}(\mathbf{x}) \qquad (2)$$

donde $\mathbf{x} \in \Re^n$ es el vector de estado del sistema; $\mathbf{y} \in \Re^m$ es el vector de salida; $\mathbf{u} \in \Re^p$ es el vector de entrada $\mathbf{f}_1 \in \Re^{s1}$ es el vector de fallas a ser detectadas; $\mathbf{f}_2 \in \Re^{s2}$ es el

vector de fallas sin interés; $\mathbf{f(x)}$ y $\mathbf{h(x)}$ son funciones vectoriales de dimensión adecuada; $\mathbf{g(x)}$, $\mathbf{E_1(x)}$ y $\mathbf{E_2(x)}$ son funciones matriciales de dimensión apropiada. La idea básica del desacoplamiento es hallar una transformación $\mathbf{T(x)}$ tal que

$$\frac{\partial T(\mathbf{x})}{\partial \mathbf{x}} E_1(x) = \mathbf{0} \qquad (3.a)$$

$$\frac{\partial T(\mathbf{x})}{\partial \mathbf{x}} E_2(x) \neq \mathbf{0} \qquad (3.b)$$

Para satisfacer las condiciones (3) algunos elementos del vector de estado $\mathbf{x}$ deberían ser eliminados. Los estados eliminados deben ser recuperados a partir de la salida del sistema, es decir, la condición

$$rango\left\{\frac{\partial}{\partial \mathbf{x}}\begin{bmatrix}\mathbf{T(x)}\\ \mathbf{h(x)}\end{bmatrix}\right\} = \dim \mathbf{x} \qquad (4)$$

debe ser satisfecha. La existencia de una transformación que satisfaga las condiciones (3) y (4) es garantizada por el teorema de Frobenius.

2.3 Modelado Takagi-Sugeno de Sistemas

Los métodos para detectar fallas basados en observadores dependen del modelo. Si el modelo del sistema en consideración es complejo, el diseño de un buen generador de residuales pudiera ser también difícil, aun si las no linealidades son perfectamente conocidas. Una manera de superar las dificultades del diseño es el uso del así llamado modelo Takagi-Sugeno. Las trayectorias no lineales son en cierto modo linealizadas alrededor de algunos puntos de operación. El conjunto de modelos lineales son usados para realizar una interpolación difusa para conseguir una aproximación del comportamiento del sistema no lineal [7].

En cierto punto de operación se define una regla **IF THEN**, la cual representa una relación local de salida/entrada del sistema no lineal. La regla para describir la situación puede ser como:

Regla i (i = 1, 2, ...r)
IF z(t) es $\mathbf{M_i}$

$$\textbf{THEN} \begin{cases} \dot{\mathbf{x}} = \mathbf{A}_i\mathbf{x}(t) + \mathbf{B}_i\mathbf{u}(t) \\ \mathbf{y}(t) = \mathbf{C}_i\mathbf{x}(t) \end{cases}$$

Donde $\mathbf{x}(t) \in \Re^n$, $\mathbf{y} \in \Re^m$, $\mathbf{u} \in \Re^p$ y $\mathbf{A_i}$, $\mathbf{B_i}$ y $\mathbf{C_i}$ son matrices de dimensiones compatibles. El vector $\mathbf{z}(t)$ es llamado la variable premisa, que se supone disponible, $\mathbf{M_i}$ es un conjunto difuso y r es el número de reglas **IF-THEN** para linealizar.

El modelo Takagi-Sugeno del sistema no lineal (1) viene dado por las siguientes fórmulas:

$$\dot{\mathbf{x}}(t) = \sum_{i=1}^{r} \omega_i[\mathbf{z}(t)]\{\mathbf{A}_i\mathbf{x}(t) + \mathbf{B}_i\mathbf{u}(t)\} \quad (5.a)$$

$$\mathbf{y}(t) = \sum_{j=1}^{r} \omega_j[\mathbf{z}(t)]\mathbf{C}_i\mathbf{x}(t) \quad (5.b)$$

donde $\omega_i[z(t)]$ es el grado de pertenencia de la variable premisa z(t) o el producto T-norma de grados de pertenencia (si z(t) tiene dos o más componentes). Notar que el grado de pertenencia debe cumplir con las siguientes restricciones:

$$\sum_{i=1}^{r} \omega_i[z(t)] = 1 \quad (6)$$

$$0 \le \omega_i[z(t)] \le 1 \quad i = 1, \ldots r \quad (7)$$

2.4 Observadores difusos

Existen varios procedimientos para el diseño de observadores difusos para sistemas Takagi-Sugeno. La técnica de [5] será la considerada aquí. Considerar un sistema no lineal (1) y su modelo Takagi-Sugeno correspondiente dado por (5), la estimación del estado es una combinación no lineal difusa de las salidas de observadores lineales locales. La dinámica completa del observador es una suma ponderada de los observadores individuales:

$$\dot{\hat{x}} = \sum_{i=1}^{r} \omega_i[z(t)]\{A_i\hat{x} + B_iu + L_i(y - \hat{y})\} \quad (8.a)$$

$$\hat{y} = \sum_{j=1}^{r} \omega_i[z(t)]C_j\hat{x}(t) \quad (8.b)$$

El observador dado arriba se basa en la *compensación paralela distribuida* (PDC) [10]. Ello implica que los pesos $\omega_i[z(t)]$ son los mismos que aplica el modelo Takagi-Sugeno del sistema no lineal. Un observador difuso debe satisfacer:

$$\mathbf{x}(t) - \hat{\mathbf{x}}(t) \rightarrow \mathbf{0} \quad \textit{cuando} \quad t \rightarrow \infty$$

donde $\hat{\mathbf{x}}(t)$ denota el estimado del vector de estado mediante el observador difuso. La convergencia del error de estimación se analiza usando el siguiente resultado:

Teorema 1 [9]. El observador difuso (8) es estable de manera asintótica si existen dos matrices positivas definidas **P**, **Q** y r matrices $\mathbf{N}_1, \ldots \mathbf{N}_r$, tales que:

$$-\mathbf{A}_i^T\mathbf{P} - \mathbf{P}\mathbf{A}_i + \mathbf{C}_i^T\mathbf{N}_i^T + \mathbf{N}_i\mathbf{C}_i - (s-1)\mathbf{Q} - 2\alpha\mathbf{W} > \mathbf{0} \quad i = 1, \ldots r \quad (9)$$

$$-\mathbf{A}_i^T\mathbf{P}-\mathbf{P}\mathbf{A}_i-\mathbf{A}_j^T\mathbf{P}-\mathbf{P}\mathbf{A}_j+\mathbf{C}_i^T\mathbf{N}_j^T+\mathbf{N}_j\mathbf{C}_i$$
$$+\mathbf{C}_j^T\mathbf{N}_i^T+\mathbf{N}_i\mathbf{C}_j+2\mathbf{P}-4\alpha\mathbf{Q}\ \geq\ \mathbf{0} \qquad (i<j) \qquad (10)$$

para todas las i y j exceptuando los pares (i, j) tales que $\omega_i[z(t)]\ \omega_j[z(t)] = 0$ para todo valor de tiempo, $s > 1$ es el número máximo de reglas que se disparan en todo valor de t y α es la tasa de decaimiento deseada.

El diseño del observador consiste en calcular las ganancias $\mathbf{L}_i$ de las partes consecuentes de las reglas para el observador. Dichas ganancias se calculan, tras resolver de modo iterativo las LMIs (9) y (10), mediante:

$$\mathbf{L}_i \ =\ \mathbf{N}_i\mathbf{P}^{-1} \qquad (11)$$

0

1 Acotamiento de la salida del observador

Si suponemos que la condición inicial x(0) es conocida, la restricción

$$\|\hat{x}(t)\|_2 \ \leq\ \upsilon$$

es forzada en todo momento $t \geq 0$ si las LMIs

$$\begin{bmatrix} 1 & \mathbf{x}(0)^T \\ \mathbf{x}(0) & X \end{bmatrix} \geq \mathbf{0} \qquad (12)$$

$$\begin{bmatrix} X & \mathbf{N}_i \\ \mathbf{N}_i^T & \upsilon^2\mathbf{I}_p \end{bmatrix} \geq \mathbf{0}, (i = 1, 2, ..r) \qquad (13)$$

son satisfechas.

3. Procedimiento propuesto

La solución al problema de detección y aislamiento de fallas es frecuentemente hecha en dos pasos, como se comentó en la sección anterior. La generación de residuales con sensitividad a fallas específicas requiere primero una etapa de desacoplamiento. Un problema habitual con las técnicas de desacoplamiento es que los subsistemas resultantes son muy no lineales y un observador para estos subsistemas no siempre es fácil de diseñar.

La observación y el modelado Takagi-Sugeno es un método versátil para resolver el problema de la generación de residuales de sistemas muy no lineales. Si el modelo Takagi-Sugeno de un sistema no lineal existe, se puede diseñar un observador para tal

sistema. La estabilidad de la dinámica del error de estimación y la tasa de convergencia a cero de dicho error se garantizan mediante la selección adecuada de las matrices de ganancia del observador ($\mathbf{L}_i$). Las técnicas postmodernas de programación no lineal permiten resolver las LMIs (9) y (10) [agregando si así se desea las (12) y 13) para acotar la salida] de modo iterativo, para luego calcular tales ganancias con (11).

El esquema presentado consiste en la aplicación de las ideas clásicas de detección de fallas a sistemas no lineales. La diferencia fundamental con las técnicas tradicionales es la forma de obtener los residuales, pues en este caso los residuales se obtienen usando observaciones construidas con modelos Takagi-Sugeno del sistema no lineal. El procedimiento se resume como:

1. Seleccionar un grupo de fallas de interés. Las fallas que no interesan se pueden tomar como señales indeseables o fallas ligeras del modelo.
2. Diseñar una transformación no lineal de desacoplamiento para eliminar las fallas que no interesan.
3. Obtener el modelo Takagi-Sugeno del subsistema resultante.
4. Diseñar un observador para el modelo Takagi-Sugeno del paso anterior.
5. Con los estimados obtenidos formar el generador de residuales.
6. Definir una rutina que permita evaluar los residuales y decidir si alguna de las fallas está activa.

4. Ejemplo de Aplicación

El sistemas que se considera es un prototipo de laboratorio de tres tanques. El sistema consiste de tres tanques acoplados por un tubo, Sólo dos tanques (en los extremos del arreglo) reciben flujo. La salida del sistema está compuesta por los niveles de los tres tanques. Sólo uno de los tanques cuenta con descarga. Ver la figura 3.

Las ecuaciones que corresponden al modelo se deducen del principio de conservación de la masa:

$$a_1 \frac{dh_1}{dt} = (1 - f_{A1})q_1 - Q_{12} - Q_{L1} \qquad (14.a)$$

$$a_2 \frac{dh_2}{dt} = Q_{12} - Q_{23} - Q_{L2} \qquad (14.b)$$

$$a_3 \frac{dh_3}{dt} = (1 - f_{A2})q_2 + Q_{23} - Q_{30} - Q_{L3} \qquad (14.c)$$

Se usan transductores piezorresistivos para realizar las mediciones cuyas ecuaciones son:

$$y_1(t) = (1 + f_{s1})h_1(t) \qquad (15)$$

$$y_2(t) = (1 + f_{s2})h_2(t) \quad (16)$$
$$y_3(t) = (1 + f_{s3})h_3(t) \quad (17)$$

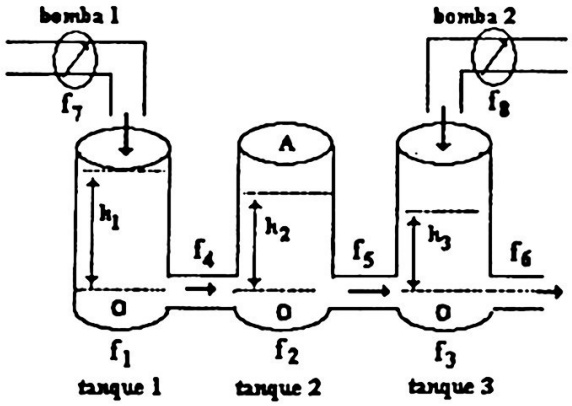


Fig. 3. Sistema de tres tanques

Al aplicar la ley de Torricelli:

$$Q_{12} = (1 - f_{B12})\mu_1 S_n \, signo(h_1(t) - h_2(t)) \bullet \sqrt{2g|h_1(t) - h_2(t)|}$$
$$Q_{23} = (1 - f_{B23})\mu_2 S_n \, signo(h_2(t) - h_3(t)) \bullet \sqrt{2g|h_2(t) - h_3(t)|}$$
$$Q_{30} = (1 - f_{B30})\mu_3 S_n \sqrt{2gh_3(t)} \qquad Q_{L1} = f_{L1} u_L S_L \sqrt{2gh_1(t)}$$
$$Q_{L2} = f_{L2} u_L S_L \sqrt{2gh_2(t)} \qquad Q_{L3} = f_{L3} u_L S_L \sqrt{2gh_3(t)}$$

donde $a_i = a$ es el área transversal de cada tanque; Q_{ij} es el flujo efectivo del tanque i al tanque j; Q_{Li} representa el flujo de fuga en cada tanque; g es la aceleración de la gravedad; S_n es el área de las conexiones; S_L es el área de las descargas de fuga; μi es el coeficiente de descarga de cada tanque y μ_L es el coeficiente de descarga de las fugas. Los valores numéricos de los parámetros son:

$\mu_1 = 0.5$ $\mu_2 = 0.5$ $a = 0.01554\ m^2$ $S_L = 8x10^{-5}\ m^2$
$\mu_3 = 0.6$ $\mu_L = 0.7$ $g = 9.8 m/s^2$ $S_n = 5x10^{-5}\ m^2$

4.1 Subsistemas desacoplados

Para obtener residuales con sensitividad selectiva a las fallas (residuales estructurados) se aplicara el método dado en [6]. Como resultado, un conjunto de subsistemas no lineales deberá ser obtenido. Cada subsistema contendrá información

acerca de un conjunto específico de fallas, como se señaló en el punto 2.2. Ver el apéndice A para el diseño del observador 2.

El primer subsistema se obtiene directamente de la ecuación diferencial para el primer tanque

$$\begin{aligned} \dot{h}_1 &= \frac{1}{a}\left[(1-f_{A1})u_1 - (1-f_{B12})\mu_1 S_n \cdot \right. \\ &\quad signo(h_1 - h_2)\cdot \\ &\quad \left.\sqrt{2g|h_1 - h_2|} - f_{L1}\mu_L S_L \sqrt{2gh_1}\right] \\ y_1 &= (1+f_{S1})h_1 \end{aligned} \tag{18}$$

Notar que las fallas f_{A1}, f_{B12}, f_{S1}, f_{S2} y f_{L1} están involucradas en el subsistema, esto es, este subsistema es sensible a dichas faltas. La variable de estado se obtiene de la segunda salida. Esa es la razón para tener al subsistema 1 sensible también a fallas del transductor 2, f_{S2}.

De igual modo los otros dos subsistemas han sido seleccionados como la segunda y la tercera ecuaciones diferenciales del modelo original (14). La tabla 1 muestra la incidencia de cada una de las once fallas posibles en cada uno de los tres subsistemas.

Tabla 1. Sensitividad a fallas de los subsistemas

	Subsistemas		
Fallas	1	2	3
f_{A1}	1	0	0
f_{A2}	0	0	1
f_{B12}	1	1	0
f_{B23}	0	1	1
f_{B30}	0	0	1
f_{L1}	1	0	0
f_{L2}	0	1	0
f_{L3}	0	0	1
f_{S1}	1	1	0
f_{S2}	1	1	1
f_{S3}	0	1	1

4.2 Modelo Takagi-Sugeno del sistema

Para cada uno de los tres subsistemas se definen seis reglas Takagi-Sugeno. Se eligieron seis puntos de operación para definir cada una de las reglas, que se resumen en la tabla 2.

Tabla 2. Reglas para obtener el modelo Takagi-Sugeno (los valores de las celdas representan fracciones de Hmax)

Regla	h_1	h_2	h_3
1	0.90	0.80	0.70
2	0.90	0.80	0.40
3	0.90	0.50	0.40
4	0.60	0.50	0.40
5	0.60	0.20	0.10
6	0.30	0.20	0.10

Para definir los puntos de operación se impusieron tres condiciones:

$h_1 > h_2$ (1); $h_2 > h_3$ (2) y $h_2 \leq 2h_3$ (3)

Las funciones de pertenencia fueron obtenidas aplicando un método analítico. Por ejemplo, el segundo término no lineal de (14) puede representarse de manera exacta usando funciones de pertenencia F_{21} y F_{22}:

$$\sqrt{h_2 - h_3} = (h_2 - h_3)\cdot[F_{21}\alpha_{21} + F_{22}\alpha_{22}] \qquad (19)$$

donde

$$F_{21}, F_{22} \in [0\ 1] \quad F_{21} + F_{22} = 1 \qquad (20)$$

Resolviendo para (19) y (20)

$$F_{21} = \frac{\delta_{22} - \sqrt{h_2 - h_3}}{\sqrt{h_2 - h_3}} \cdot \frac{\delta_{21}}{\delta_{22} - \delta_{21}}; \qquad \alpha_{21} = \frac{1}{\delta_{21}}, \quad \alpha_{22} = \frac{1}{\delta_{22}}$$

$$\delta_{21} = \sqrt{(0.8 - 0.7)H_{MAX}}$$

$$\delta_{22} = \sqrt{(0.8 - 0.4)H_{MAX}} \text{ donde } H_{MAX} = 0.64\text{m}.$$

Notar que las ecuaciones recién vistas son la representación exacta de la no linealidad porque se siguió el procedimiento de [8]. La representación grafica de estas dos funciones de pertenencia se muestra en la figura 4.

Las matrices de estado del sistema se calculan como sigue (se muestra sólo la matriz de estado del subsistema 3):

$$\mathbf{A}_3 = \begin{bmatrix} -\mu_1 S_n \alpha_{12} & \mu_1 S_n \alpha_{12} & 0 \\ \mu_1 S_n \alpha_{12} & \mu_1 S_n \alpha_{12} - \mu_2 S_n \alpha_{21} & \mu_2 S_n \alpha_{21} \\ 0 & \mu_2 S_n \alpha_{21} & \mu_2 S_n \alpha_{21} - \mu_3 S_n \alpha_{32} \end{bmatrix}$$

$$\frac{\sqrt{2g}}{a}$$

$$\mathbf{A}_3 = \begin{bmatrix} -0.0142 & 0.0142 & 0 \\ 0.0142 & -0.0426 & 0.0284 \\ 0 & 0.0284 & -0.0455 \end{bmatrix} \qquad (21)$$

Matriz de salida: $\mathbf{C}_i = \mathbf{C} = \mathbf{I}_3$; $(i = 1,\ldots 6)$

Matriz de control: $\mathbf{B}_i = \mathbf{B} = \begin{bmatrix} \frac{1}{a} & 0 & 0 \\ 0 & 0 & \frac{1}{a} \end{bmatrix}^T$; $(i = 1,\ldots 6)$

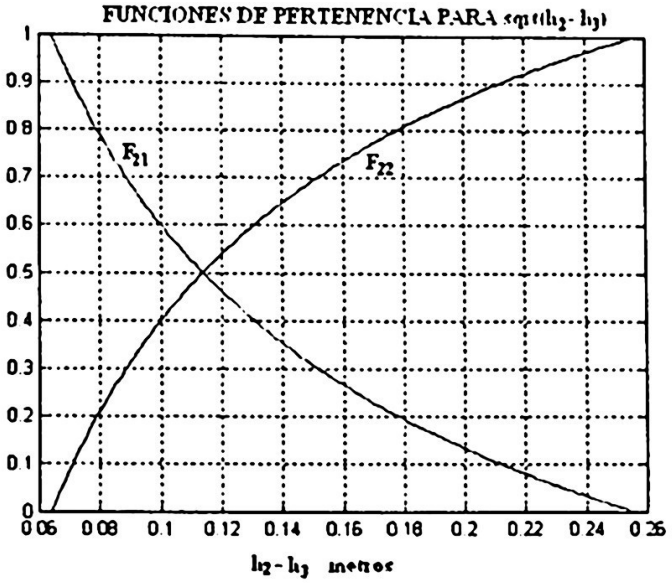


Fig. 4. Funciones de pertenencia para la variable premisa $z_2 = h_2 - h_3$.

Tabla 3. La verdad de cada regla es el mínimo de sus funciones dividido por la suma de todos los mínimos.

Regla	Funciones de pertenencia		
1	F_{11}	F_{21}	F_{33}
2	F_{11}	F_{22}	F_{32}
3	F_{12}	F_{21}	F_{32}
4	F_{11}	F_{21}	F_{32}
5	F_{12}	F_{21}	F_{31}
6	F_{11}	F_{21}	F_{31}

4.3 Resultados

El modelo del sistema de los tres tanques usado fue además controlado de modo difuso empleando el algoritmo difuso (PDC). El controlador y los observadores usan las mismas reglas del modelo. Las condiciones iniciales para el sistema ya descrito fue $x(0) = [0.24 \;\; 0.20 \;\; 0.10]^T$. Las corridas se hicieron llevando al sistema al punto de operación $x_r = [0.48 \;\; 0.40 \;\; 0.32]^T$. Notar que la elección de los puntos de operación inicial y final acentúa la naturaleza no lineal del sistema.

Para probar el método propuesto diferentes escenarios fueron considerados. Primero, el caso libre de fallas se usó para probar los valores de los residuales. En este caso los residuales tienen muy poca magnitud, casi cero, y el valor de estado permanente se aprecia en la tabla 4

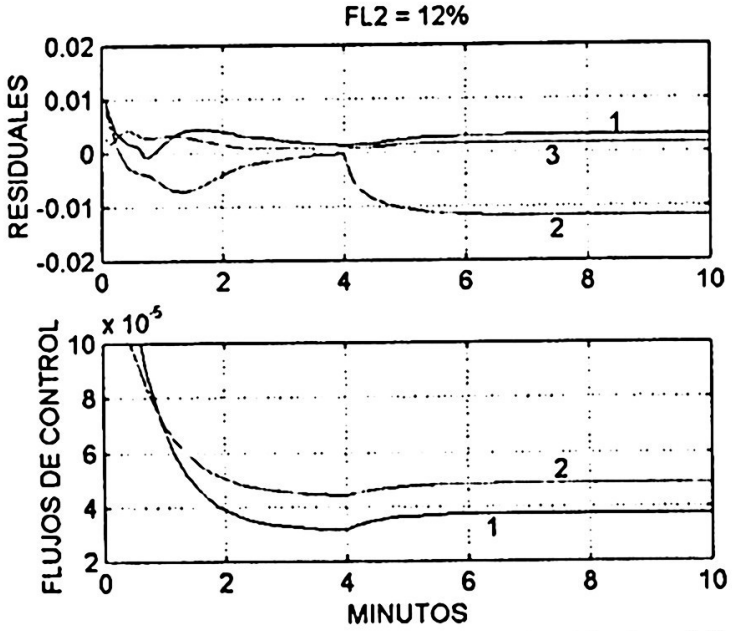


Fig. 5. Respuesta de residuales para F_{L2} de 12%. Las fallas fueron introducidas en el minuto 4, una a la vez.

En la figura 5 se ilustra la respuesta de los residuales para una fuga en el tanque 2. La magnitud de la falla fue de 12%. El residual 2 es activado, mostrando efecto de la falla, de acuerdo con la tabla 1.

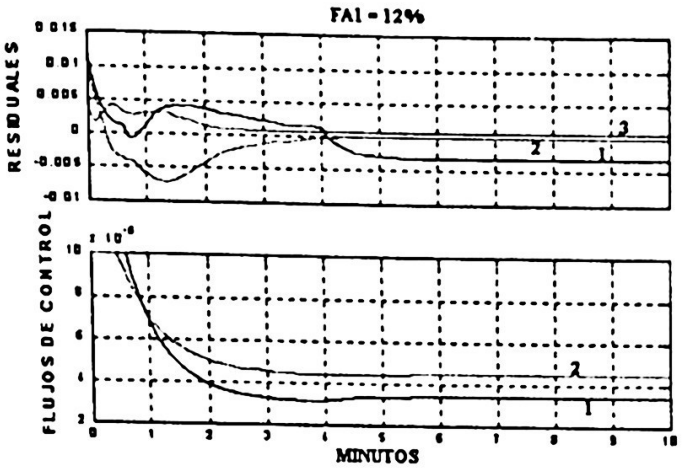


Fig. 6. Respuesta de residuales para F_{A1} de 12%.

La figura 6 exhibe la respuesta de los residuales a una falla en la bomba 1. Al igual que el caso anterior la magnitud de la falla es de 12%. El residual 1 es el único activado, de acuerdo con la teoría de la tabla 1.

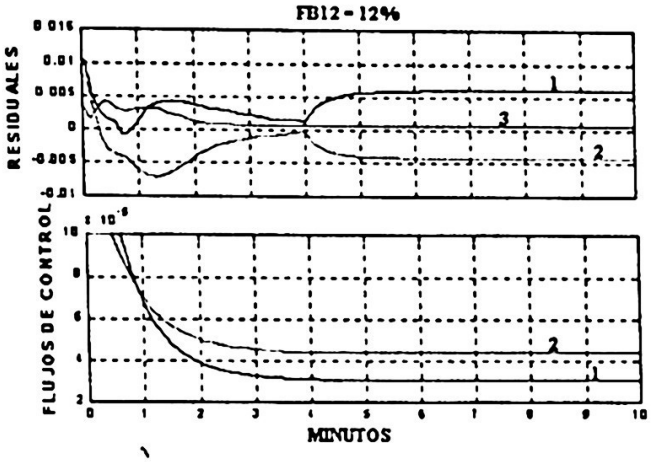


Fig. 7. Respuesta de residuales para F_{B12} de 12%.

Un último caso se muestra en la figura 7. Una falla de bloqueo entre los tanques 1 y 2 de 12 % activa los residuales 1 y 2, también consecuente con la tabla 1. El transitorio inicial que se aprecia en las gráficas para cada caso se debe a que el estimado inicial de cada variable de estado se puso como 95% del valor inicial de cada variable de estado.

Tabla 4: Valores de estado permanente de los residuales para diversas fallas

Falla	r_1	r_2	r_3
F_{L2}	0.0032	**-0.0118**	0.0018
F_{A1}	**-0.0030**	0.0000	0.0007
F_{S3}	0.0008	**-0.0086**	**0.0081**
F_{B12}	**0.0061**	**-0.0042**	0.0009
BIEN	0.0009	-0.0000	0.0007

La tabla 4 resume el valor permanente de los residuales para las cinco corridas efectuadas; cuatro de ellas con una falla y una corrida con el sistema libre de fallas. Los residuales activados se muestran en negrita para las corridas con una falla.

5. Conclusiones

En el presente trabajo el problema de la detección y el aislamiento de fallas usando desacoplamiento no lineal y observadores difusos Takagi-Sugeno ha sido abordado. El desacoplamiento no lineal se utiliza para garantizar una sensitividad específica para fallas determinadas. Los subsistemas desacoplados son a menudo marcadamente no lineales.

Los observadores difusos (basados en un modelo Takagi-Sugeno) son una buena manera para construir un observador para tales casos. Con el problema de la observación resuelto el diseño del esquema de detección y aislamiento de fallas deja de ser una tarea complicada. El esquema propuesto está siendo probado en un sistema de laboratorio de tres tanques.

El control y la estimación de sistemas no lineales mediante el diseño con LMIs representa el nuevo paradigma a aplicar en sistemas marcadamente no lineales.

Referencias

[1] **E. Alcorta-García y P. M. Frank** "Multiplicative fault isolation in linear systems", Proc. Of the Conference on Decision and Control, Phoenix, Arizona, USA, DEC. 1999.

[2] **P. Balle** "Fuzzy Model based parity equations for fault isolation", Control Engineering Practice, 7:261-270, 1999.

[3] **P. M. Frank** "Fault diagnosis in dynamic systems using analytical and knowledge based redundancy- a survey", Automatica, 26:459-474, 1990

[4] **P. M. Frank.** Analysis and qualitative model-based fault diagnosis – a survey and some new results. European Journal of Control, 2(1):6–28, 1996.

[5] **C. J. Lopez-Toribio, R. J. Patton, C. Batt, and J. Chen.** Fault diagnosis of the three tank system using fuzzy multiple inference modelling. In European Control Conference, page CD, September 1999.

[6] **R. Seliger and P. M. Frank.** Fault diagnosis by disturbance decoupled nonlinear observers. In Proc. of the CDC '91, Brighton, England, pages 2248–2253, Dec. 1991.

[7] **M. Takagi and M. Sugeno.** Fuzzy identification of systems and its application to modelling and control. IEEE Transactions on Systems Man and Cybernetics, 15(1):116–132, 1988.

[8] **K. Tanaka, T. Ikeda, and H. Wang.** "Robust stabilization of a class of uncertain non-linear system via fuzzy control". IEEE Transactions on Fuzzy Systems, 4:1–13, Feb. 1996.

[9] **K. Tanaka, T. Ikeda, and H.Wang.** "Fuzzy regulators and fuzzy observers: relaxed stability conditions and LMI-based designs". IEEE Transactions on Fuzzy Systems, 6(2):250–265, May. 1998.

[10]**H. O. Wang, K. Tanaka, and M. F. Griffin.** "Parallel distributed compensation of a nonlinear system by Takagi-Sugeno models". In IEEE-IFES, pages 531–538, 1995.

Apéndice A: Observador del 2° subsistema

Como ya se dijo, los tres observadores usan las mismas reglas que el modelo difuso y el regulador difuso. Para el segundo subsistema la iésima regla se puede expresar como:

Regla i del observador 2 (i = 1, ... 6)

$$\textbf{IF z(t) es } \mathbf{M_i} \textbf{ THEN } \begin{cases} \dot{\bar{x}}_2 = a_{2i}\bar{x}_2(t) + \mathbf{b}_{2i}\begin{bmatrix} y_1 \\ y_3 \end{bmatrix} + L_{2i}(y_2 - \bar{y}_2) \\ \bar{y}_2(t) = \bar{x}_2(t) \end{cases}$$

Notar que y_1 y y_3 entran como señales de control. Las ecuaciones para el observador se derivan de la suma ponderada dada en (8) y son:

$$\dot{\bar{x}}_2 = \sum_{i=1}^{6} \omega_i[z(t)]\left\{a_{2i}\bar{x}_2 + \mathbf{b}_{2i}\begin{bmatrix} y_1 \\ y_3 \end{bmatrix} + L_{2i}(y_2 - \bar{y}_2)\right\} \quad \bar{y}_2 = \bar{x}_2$$

Se diseñó el observador con el toolbox LMI de MATLAB, formando el conjunto de 47 LMIs para (9), (10), (12) y (13) con $\upsilon = 1.2H_{max}$, que se resolvió con el comando de eigenvalores generalizados:

```
[to,xo]=gevp(LOPEZ,19,[0 190 2e11 25 0],[],[],-3.25);
```

que tras 15 iteraciones produjo P = 23.14, Q = 2.210 y un valor de tasa de decaimiento de 4.843min^{-1}, quedando las ganancias, calculadas con (11), para las seis reglas como sigue:

Regla	1	2	3	4	5	6
L_{2i}	2.219	2.547	2.547	2.491	2.448	2.371

Además: $a_{2i} = A_{i(2,2)}$ y $\mathbf{b}_{21} = [A_{i(2,1)} \quad A_{i(2,3)}]$ y la premisa es: $z(t) = [h_1 - h_2 \quad h_2 - h_3 \quad h_3]^T$

Notar que la regla 1 y la regla 5 nunca se disparan al mismo tiempo pues si F_{31} es diferente de cero F_{33} será cero y viceversa; y que lo mismo sucede para las reglas 1 y 6, por lo que estas dos parejas no se incluyen en los LMIs de (10), por lo que éstos son sólo trece.

Diseño robusto del estabilizador de un sistema de potencia con ubicación de polos

Salvador Saucedo Flores[1] y José A. Ortega Herrera[2]

[1] SEPI-ESIME Zacatenco, IPN Correo ssaucedo@ieee.org
[2] SEPI-ESIME-IPN Correo oeha430210@hotmail.com

Resumen: Este trabajo presenta un método postmoderno para la síntesis del estabilizador de un generador conectado a un bus infinito. Se usa la transformación fraccional lineal para formar un modelo que incluya la incertidumbre de los parámetros del conjunto excitador-estabilizador. Se propone un objetivo mixto, H_2/H_∞, mismo que se minimiza mediante la técnica de las LMIs, incluyendo además ubicación de polos en cierta región de interés. Al resolver el problema de la optimización dicha, se obtienen los parámetros del regulador que hace que exista estabilidad en una gama muy amplia de puntos de operación. Se presenta un ejemplo de diseño con el toolbox LMI de MATLAB con la simulación respectiva con SIMULINK.

Palabras clave: Desigualdades lineales de matrices (LMIs), control robusto, sistemas de potencia, diseño H_2/H_∞ , sistemas inciertos, ubicación de polos, PSS.

Abstract: A postmodern design method is used to synthesize the stabilizer of a generator connected to an infinite bus. It uses the linear fractional transformation to include , in a compact way, the parameters uncertainty for an exciter-generator set. A mixed criterion H_2/H_∞ is then minimized by LMI techniques. Aside, a zone in the complex plane is chosen to place the closed loop poles for the system, in order to meet some performance specifications. When the optimization problem is solved, the controller parameters to achieve stability over many operating points are obtained. A design example using MATLAB is presented followed by its simulation.

Introducción

El requisito más fuerte para obtener un servicio confiable de transmisión de la energía eléctrica es mantener a los generadores sincrónicos trabajando en paralelo y con la capacidad adecuada para responder a las demandas de carga. Si en cierto instante un generador sale de sincronía con respecto a los demás pueden ocurrir variaciones importantes tanto en voltaje como en corriente y se pueden generar cortes por

protección en ciertas líneas de transmisión. Si una máquina tiende a ir más rápido más lento, las condiciones de sincronismo la hacen mantener el ritmo adecuado. Sin embargo, tales condiciones pueden no ser adecuadas para mantener el sincronismo éste se puede perder de manera transitoria. Es entonces el momento en que estabilizador del sistema local entra en juego para regresar al punto de operación previo, [1].

Uno de los problemas cruciales para lograr la estabilidad de los sistemas de potencia es el control de señales de poca variación alrededor de un punto de operación. oscilaciones de pequeña magnitud y baja frecuencia si persisten por periodos prolongados pueden ocasionar pérdidas en la capacidad de transferencia de potencia. Ya que nuestra preocupación es la desconexión de líneas, las fluctuaciones potencia que pueden ser toleradas dependen de muchas condiciones, entre las que está el nivel de carga y los impactos a los cuales el sistema está sujeto.

Los estabilizadores de sistemas de potencias (PSS = Power System Stabilizer) se considerado desde hace más de sesenta años como una herramienta útil para amortiguar las oscilaciones electromecánicas en sistemas de potencia [2]. El modelo linealizado puede ser empleado para este estudio y un PSS convencional se puede considerar como un controlador de una entrada y una salida, montado en la máquina generadora. La entrada al PSS que aquí usamos es la velocidad de la flecha de máquina, pero también se pueden usar como entradas al PSS la frecuencia del bus alterna o la segunda derivada de la señal de potencia. La salida del PSS se aplica en sumador de entrada del regulador de voltaje, como una señal suplementaria al control del excitador [3]. Se han empleado otros esquemas de regulación, ver [4] para control difuso y [1] para el diseño clásico de Bode del PSS.

Al experimentar cambios en las condiciones de operación se da una variación grande correspondiente en el comportamiento dinámico en el modelo de pequeñas señales un sistema de potencia. Esto puede interpretarse como incertidumbre paramétrica el modelo linealizado de señales pequeñas. Por ello, el objetivo de este trabajo es diseño robusto del PSS de modo que las oscilaciones sean amortiguadas en una gama muy amplia de puntos de operación. El trabajo fue motivado por dos contribuciones recientes [5]-[6], que usan ambas programación no lineal para el diseño. Se siguió metodología de [5], con tres cambios significativos: primero, se usó un modelo sexto orden en lugar del de cuarto orden de ellos; segundo, se hicieron menos simplificaciones para calcular el modelo; y por último se hizo truncamiento par- reducir el orden del controlador y se agregó la red de lavado "wash-out", que en no hicieron.

Se usa el toolbox LMI para aplicar esta técnica relativamente nueva al diseño compensador. Es bien sabido que las especificaciones para el diseño de PSSs son coeficiente de amortiguamiento y la velocidad de respuesta y que ambas se traduce en una región adecuada para la operación en lazo cerrado del conjunto. Además, sabe que obtener estabilidad y rendimiento sobre una gama muy amplia del valor los parámetros del sistema se logra al fijar una cota superior a la norma H_∞ de función de transferencia en lazo cerrado. Una restricción dada por una región dond

ubicar los polos y una cota razonable sobre la norma H_∞ se puede formular como una optimización convexa, la cual se puede resolver con técnicas LMI. Además, si se desea limitar el esfuerzo de la señal de control conviene adicionar a lo anterior una función objetivo cuadrática H_2, cosa que aquí se ilustra [7].

Control Óptimo Mixto H_2/H_∞ con Ubicación de Polos

Dado el modelo LTI

$$\left.\begin{aligned} dx/dt &= \mathbf{A}x + \mathbf{B}_1\omega + \mathbf{B}_2\mathbf{u} \\ \mathbf{z}_1 &= \mathbf{C}_1x + \mathbf{D}_{11}\omega + \mathbf{D}_{12}\mathbf{u} \\ \mathbf{z}_2 &= \mathbf{C}_2x + \mathbf{D}_{21}\omega + \mathbf{D}_{22}\mathbf{u} \\ \mathbf{y} &= \mathbf{C}_y x + \mathbf{D}_{y1}\omega + \mathbf{D}_{y2}\mathbf{u} \end{aligned}\right\} \qquad (1)$$

donde $\mathbf{x}$ es el estado del sistema, $\mathbf{u}$ es el vector de control, $\mathbf{y}$ es el vector de salida y $\mathbf{z}_1$ y $\mathbf{z}_2$ son salidas ficticias propuestas para poder definir los objetivos, $\mathbf{w}$ es una entrada real o ficticia, según se ilustra en la figura 1. Se desea diseñar un regulador que se aplique sobre el estado $\mathbf{x}$, o la salida $\mathbf{y}$, de modo que se satisfagan tres criterios: primero, que la ganancia RMS de la entrada $\mathbf{w}_1$ hacia la salida $\mathbf{z}_1$ sea igual o menor que cierto límite prefijado; segundo, que la ganancia H_2 norma de la entrada $\mathbf{w}_2$ hacia la salida $\mathbf{z}_2$ también sea igual o inferior a otro límite prefijado; y tercero, que la velocidad de respuesta y tasa de amortiguamiento sean apropiados, lo que se especifica como que los polos de lazo cerrado del sistema queden ubicados dentro de cierta región en el plano complejo que asegure estabilidad y el rendimiento ya dicho. En particular aquí se usará un regulador al que se le aplica la señal escalar de salida, con función de transferencia que en el dominio de Laplace, y en estado espacio, queda:

$$\left.\begin{aligned} \mathbf{U}(s) &= \mathbf{K}(s)\mathbf{Y}(s) \\ \mathbf{K}(s): &\begin{cases} \dot{\mathbf{x}}_K = \mathbf{A}_K\mathbf{x}_K + \mathbf{B}_K y \\ \mathbf{u} = \mathbf{C}_K x_K + \mathbf{D}_K y \end{cases} \end{aligned}\right\} \qquad (2)$$

En los últimos años se han generado rutinas muy efectivas que resuelven de modo iterativo el problema enunciado arriba [8]. El criterio cuadrático H_2, para el caso SISO (una entrada, una salida), si D_{21} es cero es igual a

$$J = \int_0^\infty \left(\mathbf{C}_2 y^2 \mathbf{C}_2^T + \mathbf{D}_{22}u^2\mathbf{D}_{22}^T\right)dt \qquad (3)$$

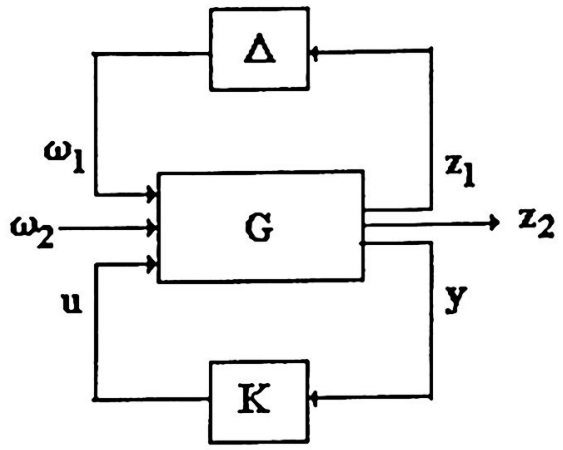


Figura 1. Diagrama que muestra la transformación fraccional lineal para representar la incertidumbre y la realimentación para estabilizar el sistema, [9].

Se emplea la técnica LMI para resolver el problema enunciado, definiendo primero región donde deben quedar los eigenvalores de lazo cerrado. Una región LMI en plano complejo se define mediante dos matrices **L** y **M** de modo que

$$D = \left\{ z \in \mathbf{C} : \mathbf{L} + z\mathbf{M} + \bar{z}\mathbf{M}^T < 0 \right\} \qquad (4)$$

donde **L** y **M** son matrices reales tales que $\mathbf{L} = \mathbf{L}^T$. La función característica de una región LMI viene dada por

$$f_D(z) = \mathbf{L} + z\mathbf{M} + \bar{z}\mathbf{M}^T \qquad (5)$$

Por ejemplo, para un sector cónico con vértice en el origen y ángulo interior 2θ función característica viene dada por [8]:

$$f_D(z) = \begin{bmatrix} sen\theta\left(z + \bar{z}\right) & \cos\theta\left(z - \bar{z}\right) \\ \cos\theta\left(z - \bar{z}\right) & sen\theta\left(z + \bar{z}\right) \end{bmatrix}$$

Las intersecciones de regiones LMI también son regiones LMI y cualquier región convexa simétrica al eje real se puede aproximar mediante una región LMI. En sección 4 se abundará en este tópico.

Modelo Incierto del Sistema de Potencia

La representación unifilar de una máquina conectada a un bus infinito se muestra la figura 2. El generador está dotado de un excitador estático tipo 1 y un regulador automático de voltaje (AVR), que cuentan con estabilizador local para lograr avance de fase en lazo abierto. La tabla 1 ofrece el valor de los parámetros del sistema. Para

el análisis de pequeñas oscilaciones se puede despreciar los transitorios rápidos de los devanados. Ver la simbología al final.

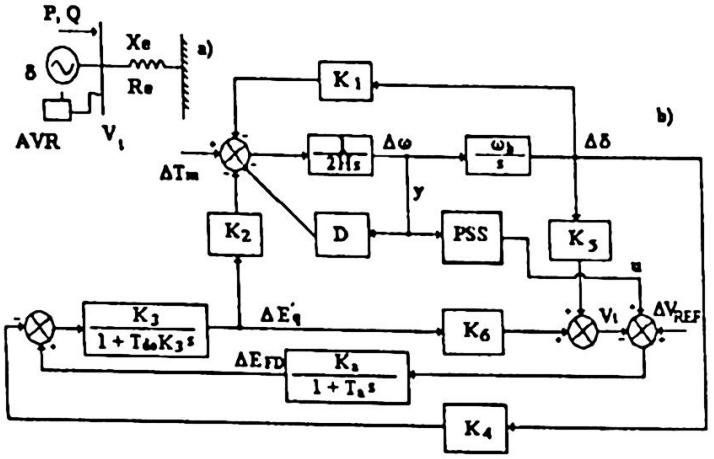


Figura 2. Modelo de la máquina de potencia con bus infinito. (a) Representación unifilar y (b) Diagrama a bloques del sistema linealizado con el estabilizador.

Las condiciones de operación del sistema en cuestión está completamente definido por los valores de la potencia real P y la potencia reactiva Q en las terminales del generador y la impedancia de la línea de transmisión X_e. Se calculó el modelo lineal asumiendo que P, Q y X_e varían independientemente sobre los rangos (0.4 a 1.0), (-0.2 a 0.4) y (0.2 a 0.6) respectivamente, en pasos de valor 0.2, lo que arroja 48 modelos, de sexto orden cada uno, para reflejar la incertidumbre de los parámetros. En el estado espacio estos modelos vienen dados por:

$$d\mathbf{x}/dt = \mathbf{A}_i\mathbf{x} + \mathbf{B}u \quad i=1,2,\cdots,48 \tag{6}$$

donde la matriz de estado, de acuerdo con la figura 2, viene dada por:

$$\mathbf{A}_i = \begin{bmatrix} 0 & \omega_b & 0 & 0 & 0 & 0 \\ -\frac{K_1}{2H} & -\frac{D}{2H} & -\frac{K_2}{2H} & 0 & 0 & 0 \\ -\frac{K_4}{T_{do}} & 0 & -\frac{1}{K_3T_{do}} & \frac{1}{T_{do}} & 0 & 0 \\ 0 & 0 & 0 & -\frac{K_7}{T_E} & \frac{1}{T_E} & 0 \\ -\frac{K_1K_5}{T_A} & 0 & -\frac{K_1K_6}{T_A} & 0 & -\frac{1}{T_A} & -\frac{K_A}{T_A} \\ 0 & 0 & 0 & -\frac{K_7K_F}{T_ET_F} & \frac{K_F}{T_ET_F} & -\frac{1}{T_F} \end{bmatrix} \tag{7}$$

$\mathbf{x} = \begin{bmatrix} \Delta\delta & \Delta\omega & \Delta E'_q & \Delta E_{FD} & \Delta V_R & \Delta V_{ST} \end{bmatrix}^T$ es el vector de estado.

$\mathbf{B}_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & K_A/T_A & 0 \end{bmatrix}^T$ es la matriz de control y

$\mathbf{C}_y = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$ es la matriz de salida, donde [10]:

$$K_1 = \frac{E_{qo}V_{io}}{\Delta z}\left(r_e sen\delta + (x_e + x_d')(x_e + x_q)\cos\delta\right) + \frac{I_{qo}V_{io}}{\Delta z}\left((x_q - x_d')(x_e + x_q)sen\delta - r_e(x_e - x_d')\cos\delta\right) \quad (8)$$

$$K_2 = \frac{r_e E_{qo}}{\Delta z} + I_{qo}\left(1 + \frac{(x_d - x_d')(x_q + x_e)}{\Delta z}\right) \quad (9)$$

$$K_3 = \frac{1}{1 + \frac{(x_d - x_d')(x_q + x_e)}{\Delta z}} \quad (10)$$

$$K_4 = \frac{V_{io}(x_d - x_d')}{\Delta z}\left((x_e + x_q)sen\delta - r_e\cos\delta\right) \quad (11)$$

$$K_5 = \frac{V_{qo}}{V_{io}}\frac{V_{io}x_d'}{\Delta z}\left(r_e\cos\delta - (x_e + x_q)sen\delta\right) + \frac{V_{do}}{V_{io}}\frac{V_{io}x_q}{\Delta z}\left(r_e sen\delta + (x_e + x_d')\cos\delta\right) \quad (12)$$

$$K_6 = \frac{V_{qo}}{V_{io}}\left(1 - X_d'(x_e + x_q)\Delta z\right) + \frac{V_{do}x_q r_e}{V_{io}\Delta z} \quad (13)$$

$$\Delta z = (r_e + r_s)^2 + (x_e + x_q)(x_e + x_d') \quad (14)$$

Donde la corriente entregada por el generador es

$$I_e = conj\left(\frac{P + jQ}{V_{io}}\right) \quad (15)$$

Los voltajes generado y en terminales son

$$E_{qe} = V_{io} + \left((r_s + r_e) + (x_q + x_e)j\right)I_e$$

$$V_t = V_{io} + (r_e + jx_e)I_e \quad (16)$$

con

$$\delta = ang(E_{qe}), \qquad E_{qo} = |E_{qe}|, \qquad V_{io} = \frac{\overline{E_{qe}}}{E_{qo}}V_t = V_{qo} - jV_{do}, \qquad [10].$$

$$K_7 = K_E + A_{Ex}B_{Ex}e^{B_{Ex}E_{FD}}$$

x_d	x_d'	x_q	T_{do}	V_{io}	H
1.790pu	0.355pu	1.660pu	7.9s	1.01pu	3.77s
ω_b	K_A	T_A	r_e	r_s	D
377rad/s	50	0.06s	0.027pu	0.0048pu	2.0s
K_1	K_2	K_3	K_4	K_5	K_6
1.092	0.920	0.345	1.221	0.0385	0.480

Tabla 1. Datos del sistema: 3600 RPM, 18KV, 920 MVA, 0.9PF. Los valores de las Ks son para el caso específico P = 0.6; Q = -0.2 y X_e = 0.4. Proyecto 10.4 de [10]. Cuenta con

estabilizador del AVR con $K_F = 0.0832$ y $T_F = 1.0$ y el excitador es tipo 1 con $T_E = 0.0520$ y $K_E = -0.0465$.

Representación del Modelo Incierto

Dado el modelo lineal

$$\left.\begin{aligned} dx/dt &= \mathbf{A}x + \mathbf{B}_2 u \\ y &= \mathbf{C}_y x \end{aligned}\right\} \qquad (17)$$

La incertidumbre del modelo se puede representar como una transformación fraccional lineal como se explica a continuación. Se agrega al modelo (17) una entrada ficticia ω_1 y una salida ficticia $\mathbf{z}_1$.

$$dx/dt = \mathbf{A}x + \mathbf{B}_1\omega + \mathbf{B}_2 u,\ z_1 = \mathbf{C}_1 x \qquad (18)$$

y apliquemos una realimentación $\omega_1 = \Delta x$, donde Δ es una matriz real que satisface

$$\|\Delta\| \leq 1 \qquad (19)$$

La dinámica de (18) viene dictada por

$$dx/dt = (\mathbf{A} + \mathbf{B}_1\Delta\mathbf{C}_1)x + \mathbf{B}_2 u = \mathbf{A}_i x + \mathbf{B}_2 u \qquad (20)$$

y con la elección adecuada de las matrices $\mathbf{B}_1$ y $\mathbf{C}_1$, todos los modelos que representan al modelo incierto (3) pueden ser cubiertos por (18) en el sentido de que cualquier matriz posible del sistema $\mathbf{A}_i$ puede ser generada al elegir una matriz Δ cuya norma será menor que la unidad.

Usando el procedimiento S de [7] es fácil demostrar que el sistema (18) es estable para todas las Δ adecuadas si

$$\|\mathbf{C}_1(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}_1\|_\infty < 1 \qquad (21)$$

Este resultado se puede aplicar al metasistema de la figura 1, donde el modelo físico de la planta ha sido aumentado con la entrada ficticia ω_1 y la salida ficticia $\mathbf{z}_1$ para modelar las incertidumbres de la planta y la salida z_2 para incluir un criterio cuadrático de rendimiento que se discutió en la primera sección. El sistema aumentado tiene la representación dada en (1). La formulación del problema se dio en la sección inicial y sólo se aclara que en este caso se resolvió usando el comando **hinfmix** del toolbox LMI de MATLAB para diseñar el bloque PSS del sistema dado en la figura 2, con una constante de tiempo dominante de 3s y una tasa de amortiguamiento ξ mayor que 0.14. La ganancia estática del PSS debe ser cero, por lo que se debe agregar a la solución dada por **hinfmix** una red de lavado ("wash-out").

Al aplicar el procedimiento recién descrito al diseño del estabilizador del sistema de potencia, el modelo incierto, representado por 48 modelos lineales, debe ser expresado en términos de las matrices $\mathbf{B}_1$ y $\mathbf{C}_1$ en (1). Para tal fin, escribir la matriz del sistema en cierto punto de operación como:

$$\mathbf{A}_i = \mathbf{A}_o + \bar{\mathbf{A}}_i \; ; \qquad \bar{\mathbf{A}}_i = \mathbf{B}_1 \Delta_i \mathbf{C}_1 \tag{22}$$

donde $\mathbf{A}_o$ representa la condición nominal de operación. Las matrices $\mathbf{B}_1$ y $\mathbf{C}_1$ deben elegirse de modo que la matriz Δ_i satisfaga (22) y (19). El primer requisito es que

$$\bar{\mathbf{A}}_i \in \mathrm{Im}\,\mathbf{B}_1 \; , \bar{\mathbf{A}}_i^T \in \mathrm{Im}\,\mathbf{C}_1 \; ; \; i = 1, 2, \ldots, N.$$

Se definen ahora dos matrices Φ_B y Φ_C de orden n x N·n, donde n = 6 es el orden del modelo y N = 48 es el número de modelos linealizados usados, como

$$\left.\begin{aligned} \Phi_B &= \begin{bmatrix} \mathbf{A}_1 & \bar{\mathbf{A}}_2 & \cdots & \bar{\mathbf{A}}_N \end{bmatrix} \\ \Phi_C &= \begin{bmatrix} \mathbf{A}_1^T & \bar{\mathbf{A}}_2^T & \cdots & \bar{\mathbf{A}}_N^T \end{bmatrix} \end{aligned}\right\} \tag{23}$$

Al aplicar la descomposición de valores singulares de tales matrices

$$\Phi_B = \mathbf{U}_B \Lambda_B \mathbf{V}_B^T ; \qquad \Phi_C = \mathbf{U}_C \Lambda_C \mathbf{V}_C^T \tag{24}$$

Si ahora se calculan los rangos de tales matrices r_B = rango(Φ_B) y r_C = rango(Φ_C), entonces existen Δ_i que satisfacen (22) para i = 1, 2, ..., N, si las matrices $\mathbf{B}_1$ y $\mathbf{C}_1$ se eligen, en primera instancia, como

$$\mathbf{B}_1 = \mathbf{B}_1' \quad y \quad \mathbf{C}_1 = \mathbf{C}_1', \text{ donde}$$

$$\mathbf{B}_1' = \begin{bmatrix} u_1 & u_2 & \cdots & u_{r_B} \end{bmatrix} \quad \mathbf{C}_1' = \begin{bmatrix} v_1 & v_2 & \cdots & v_{r_C} \end{bmatrix}'$$

y u_i y v_i son las columnas de $\mathbf{U}_B$ y $\mathbf{U}_C$, respectivamente. Las descomposiciones se realizaron con el comando **svd** de MATLAB. Los elementos de las matrices

$$\Lambda_B = diag(\lambda_1^b, \lambda_2^b, \cdots) \qquad \Lambda_C = diag(\lambda_1^c, \lambda_2^c, \cdots)$$

son los valores principales, ordenados en magnitud decreciente, de Φ_B y Φ_C, respectivamente. Si desde algún cierto valor principal, éste resulta muy pequeño relativamente, se puede reducir el rango y por tanto el orden de la incertidumbre.

Para satisfacer (22) y (19) las columnas de $\mathbf{B}_1$ y las hileras de $\mathbf{C}_1$ requieren de ser escaladas dos veces. Esto se logra como sigue.

$$\left.\begin{aligned}\Lambda_B &= diag(\lambda_1^b, \lambda_2^b, \cdots \lambda_{r_B}^b)\\ \Lambda_C &= diag(\lambda_1^c, \lambda_2^c, \cdots, \lambda_{r_C}^c)\end{aligned}\right\} \qquad (25)$$

y hacer $\mathbf{B}_1^* = \mathbf{B}_1 \Lambda_B$ $\mathbf{C}_1^* = \Lambda_C \mathbf{C}_1$, como un primer escalamiento. En (22) sustituir $\mathbf{B}_1$ por $\mathbf{B}_1^*$ y $\mathbf{C}_1$ por $\mathbf{C}_1^*$ y resolver para Δ_i en todos los puntos de operación y calcular la norma máxima $\gamma = \max_i \Delta_i$. Entonces existirá una Δ_i que satisfará a (22) y a (5) para cualquier punto de operación si las matrices $\mathbf{B}_1$ y $\mathbf{C}_1$ se eligen , ver [9], como

$\mathbf{B}_1 = \gamma^{1/2} \mathbf{B}_1^*$ $\mathbf{C}_1 = \gamma^{1/2} \mathbf{C}_1^*$ lo que constituye un segundo escalamiento.

Diseño Robusto del PSS

Para implementar el modelo incierto se debe elegir una matriz nominal del sistema $\mathbf{A}_o$ y calcular a continuación las matrices $\mathbf{B}_1$ y $\mathbf{C}_1$. La figura 3 exhibe los polos más lentos de lazo abierto de los 48 puntos de operación elegidos. El sistema nominal se eligió como

$$\mathbf{A}_o = \frac{\mathbf{A}_{min} + \mathbf{A}_{max}}{2}$$

$$= \begin{bmatrix} 0 & 377.0 & 0 & 0 & 0 & 0 \\ -0.1547 & -0.2653 & -0.1642 & 0 & 0 & 0 \\ -0.2112 & 0 & -0.3851 & 0.1266 & 0 & 0 \\ 0 & 0 & 0 & 0.3477 & 19.231 & 0 \\ 43.407 & 0 & -321.32 & 0 & -16.667 & -833.33 \\ 0 & 0 & 0 & 0.0289 & 1.60 & -1.00 \end{bmatrix}$$

donde $\mathbf{A}_{max}$ y $\mathbf{A}_{min}$ son matrices cuyos elementos son los valores máximo y mínimo, de los correspondientes elementos de las matrices $\mathbf{A}_i$ para i = 1, 2, ..,48. Esta representación incierta está aproximadamente en el centro de los 48 modelos lineales. En la figura 3 se muestran también los polos de $\mathbf{A}_o$. El valor de γ quedó en $3.81 * 10^{-1}$, y los valores principales de la descomposición quedaron

$$\Lambda_B = diag(825.6, 2.04, 0.55, 0.23, 0, 0)$$
$$\Lambda_C = diag(686.9, 457.9, 2.04, 0, 0, 0)$$

esto es, considerando el decrecimiento rápido de Λ_B su rango numérico se tomó como $r_B = 1$, y de manera semejante $r_C = 2$. Al realizar en sucesión los dos escalamientos del punto anterior, queda:

$$\mathbf{B}_1 = \begin{bmatrix} 0 & 0.0042 & 0.0051 & 0 & -16.11 & 0 \end{bmatrix}^T$$

$$\mathbf{C}_1 = \begin{bmatrix} 0.9197 & 0 & -13.370 & 0 & 0 & 0 \\ 8.9132 & 0 & 0.6131 & 0 & 0 & 0 \end{bmatrix}$$

que en combinación con (19) representa el rango aceptable de la variación de parámetros del modelo de la planta. La figura 4 ilustra que las 48 matrices satisfacen (19).

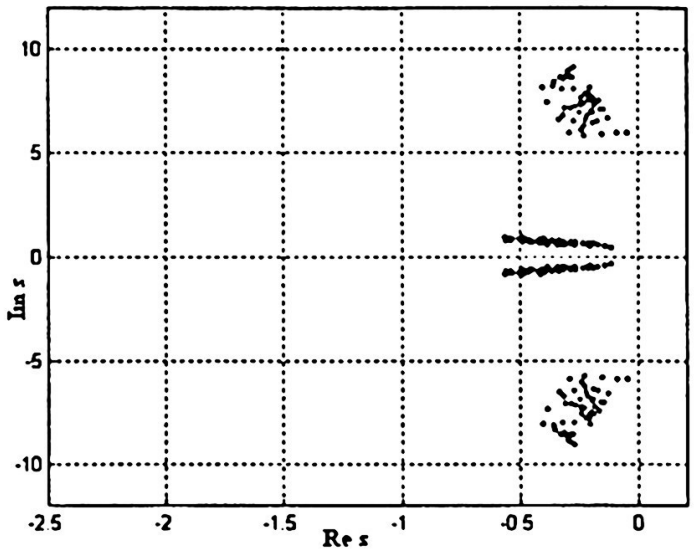


Figura 3. Polos dominantes de lazo abierto del generador en todos los puntos aceptados de operación. Los polos de la matriz nominal $\mathbf{A}_o$ se representan con círculos de mayor área.

A continuación se usó la función **hinfmix** del toolbox LMI de MATLAB para diseñar el controlador robusto del sistema (1) con $\mathbf{A} = \mathbf{A}_o$, y $\mathbf{B}_1$ y $\mathbf{C}_1$ como se calcularon antes. Los parámetros de ajuste son: a) Los pesos para la función cuadrática de rendimiento dada en (3) que se fijó como

$$\mathbf{C}_2 = \begin{bmatrix} \mathbf{C}_y \\ \mathbf{0} \end{bmatrix} \quad , \mathbf{D}_{22} = \begin{bmatrix} 0 \\ q \end{bmatrix}$$

donde $\mathbf{C}_y$ es la matriz de salida en (17) y q es un valor positivo para acotar magnitud del control; b) los parámetros (σ_1, σ_0, β) de las restricciones para ubicación de polos que se exhibe en la figura 5. Se selecciona así una región LMI definida por un sector cónico, con σ_0 = -0.5, β = 8°, para satisfacer especificaciones de velocidad de respuesta y de tasa de amortiguamiento; la segund- región fue el semiplano a la derecha de σ_1 = -9 para evitar dinámicas muy rápidas. se usó la barra horizontal. El valor de q se fijó en 1.

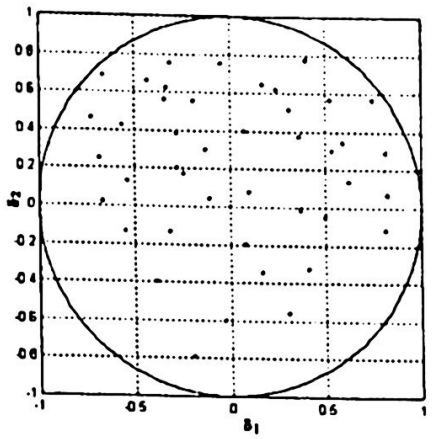


Figura 4. Dispersión de las matrices $\Delta_i = [\delta_{1i}\ \delta_{2i}]$ calculadas con (22). Notar que todas quedan dentro del círculo unitario.

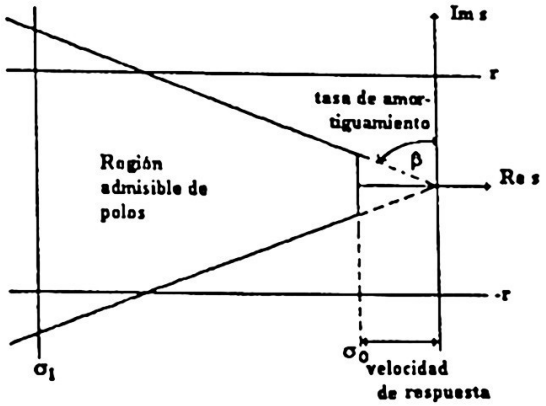


Figura 5. Región LMI definida por tres regiones: un sector cónico, a la derecha de una abscisa (semiplano) y una barra horizontal entre –r y r.

A continuación se empleó el comando **schmr** , de truncamiento balanceado, para reducir el orden del controlador quedando éste de tercer grado. Al afectar ese resultado con la función de transferencia de la red de lavado queda:

$$K(s) = 10.30\frac{s^4 + 31.00s^3 - 610.6s^2 + 269.9s}{s^4 + 17.75s^3 + 133.4s^2 + 136.2s + 25.99}$$

Los polos de lazo cerrado se dan en la figura 6, aunque algunos de ellos no quedan estrictamente en la región LMI, debido al efecto de la reducción del orden del controlador y de la red de lavado, cuya función de transferencia es: 2.75s/(4s + 1).

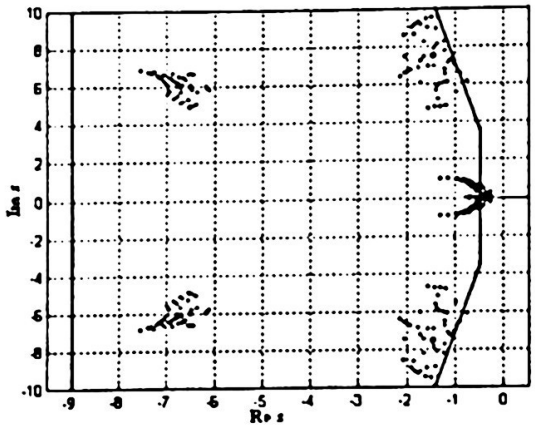


Figura 6. Valores principales más lentos de lazo cerrado. La región LMI viene dada por la intersección de dos regiones LMI, en sólido. El lazo cerrado para la matriz nominal se da con círculos de mayor área.

Resultados de la Simulación

La ubicación de los parámetros de la región LMI (σ_1, σ_0, β) se puede usar como un ajuste fino para mejorar las especificaciones originales incluyendo también el pico de la respuesta. Por ejemplo moviendo a la izquierda el valor de σ_0 tenemos más esfuerzo en el control pero menor pico en la respuesta, y viceversa. La figura 7 ilustra el resultado de la simulación con un 10% de escalón unitario en la variable de perturbación ΔV_{REF}, para tres casos diferentes. Se usaron modelos lineales.

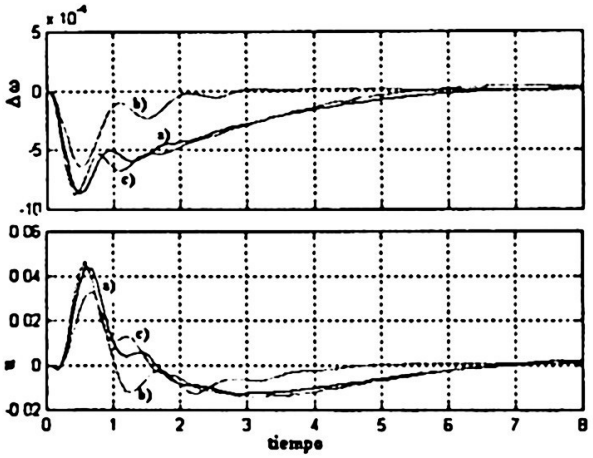


Figura 7. Respuesta a un cambio de 10% en escalón en la perturbación. (a) P = 0.4, Q = 0.4, X_e = 0.2; (b) P = 0.6, Q = 0.0, X_e = 0.6. (c) P = 0.8, Q = 0.4,X_e = .0.2.

Las figuras 8 y 9 ilustran la respuesta a cambios en el par mecánico de +10% en instante 1.666 y luego de menos 20% en el instante 8.333, para la condición inicial = 0.6, Q = 0.4 y X_e = 0.2; para el PSS encontrado aquí en trazo sólido, para el PSS propuesto por Ong, capítulo 10 de [10] en guión; y el lazo abierto en punteado. Se usó para esta corrida el modelo no lineal, dado en el CD que acompaña al texto de [10].

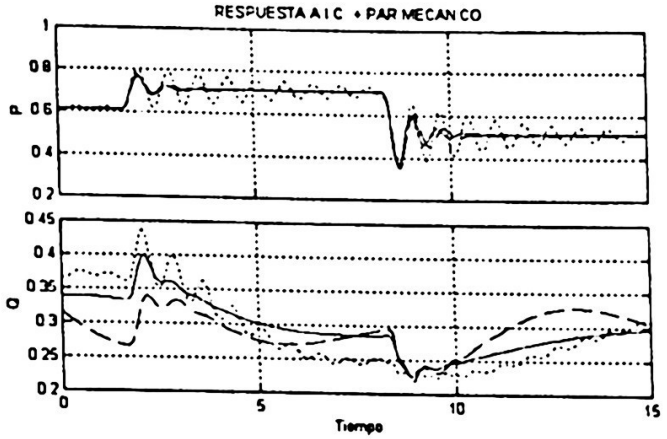


Figura 8. Potencia real y reactiva generadas con cambios en el par mecánico ΔT_m.

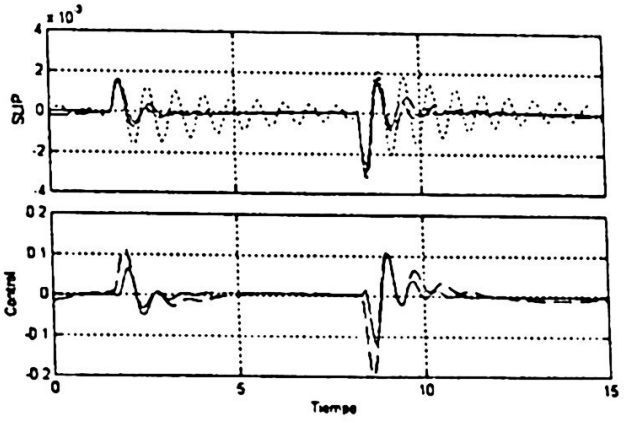


Figura 9. Cambio en la sincronía para cambios en el par mecánico y la señal de control de ambos PSSs.

Conclusiones

El método seguido es una solución automática para el diseño de sistemas lineales inciertos, brindando buenas características dinámicas en la respuesta a la perturbación. El mantener la sincronía de equipo tan delicado es un magnífico motivo para el desarrollo del ejemplo mostrado.

Se puso especial énfasis en usar un modelo matemático del generador sin hacer muchas simplificaciones con la idea de que el alumnado pueda simular sus propios ejemplos. Los datos se pueden cambiar fácilmente para diseñar el PSS de otras máquinas de forma expedita. Los files para ejecutar con MATLAB el diseño y la simulación para el ejemplo (aunque usando un modelo sin estabilizador local, de cuarto orden) se pueden descargar de [11] y modificarlas fácilmente para tener el caso presentado.

Simbología

Δ	Desviación pequeña,	x_e	Reactancia de la línea,
V_R	Voltaje del AVR	r_s	Resistencia estator
V_{ST}	Voltaje de estabilización	H	Constante de inercia del rotor
x'_d	Reactancia transitoria en eje d	D	Fricción viscosa del rotor
r_e	Resistencia en la línea,	K_A	Ganancia estática del AVR
E_{FD}	Voltaje de campo	T_A	Constante de retardo del AVR
x_q	Reactancia sincronía del eje q	T'do	Cte. de retardo en eje d
δ	V_{io} Voltaje de bus infinito	Angulo del rotor,	
x_d	Reactancia sincronía del eje d	$\Delta\omega$	deslizamiento
V_t	Voltaje en terminales,	AVR	Regulador automático de voltaje
w_b	Velocidad base		

Referencias

[1]*Anderson, P. M. y A. A. Fouad* Power System Control and Stability, IEEE PRESS. 1994.

[2]*Schleif, F. R., H. D. Hunkins, E. E. Hattan y W. B. Gish* "Control of Rotating Exciters for power system damping", IEEE Trans PAS 88. pp. 1259-66 1969.

[3]*L. Rouco, F.L. Pagola*, "An Eigenvalue Sensitivity Approach to Location and Controller Design Controllable Series Capacitors for Damping Power Systems Electromechanical Oscillations". IEEE Trans. on Power S., Vol. PWRS-12, No. 4, Nov.1997, pp. 1660-1666.

[4]*P. Hoang, K. Tomsovic, Pullman. WA*, "Design and Analysis of an Adaptive fuzzy power system stabilizer", IEEE Trans. on E. Conversion, Vol. 11, No. 2, Junio 1996, pp. 455-461.

[5]*Werner, H. , P. Korba y T. C. Yang* "Robust Tuning of Power System Stabilizers Using LMI-Techniques,", IEEE Trans. CST, vol. 11, no. 1, pp. 147-151, enero 2003.

[6]*Rao, P. S. e I. Sen* "Robust Tuning of Power System Stabilizers Using QFT", IEEE Trans. CST, vol. 7, no. 4, pp. 478-489, Julio, 1999.

[7]*Akparian, P. G. Becker, P. Gahinet y H. Kajiwara* "LMI Techniques in Control Engineering- From theory to practice", Proc. IEEE Conf. DCW, 1996.

[8]*Chilali, M. , P. Gahinet y P. Apkarian* "Robust pole placement in LMI regions" IEEE Trans on AC, vol. 44, no. 12 pp. 2257-2270, Dic. 1999.

[9]*B. R. Barmish* New Tools for Robustness of linear Systems, New York McMillan, 1993.

[10]*Ong, C. M.* Dynamic Simulation of Electric Machinery, Prentice-Hall, 1998.

[11] www.prodigyweb.net.mx/saucedo8 (Descargar PSS).

Desarrollo de un sistema de control de posición de un Servomecanismo

I. I. Siller Alcalá [1], R. Acántara Ramírez[1], J. Jaimes Ponce[1], R. Rodríguez Ruiz[1], J. Cinco Galicia[1] E. Ruiz May[1]

[1] UAM Azcapotzalco, Div. de CBI, Depto. de Electrónica, Grupo de Control de Procesos, Av. San Pablo No. 180, Col. Reynosa Tamaulipas, Del. Acapotzalco, C. P. 02200 México, D. F., México

{sai,raar,jjp}@correo.azc.uam.mx

Abstract. In this paper the development of a position control system of a servomechanism is presented. The servomechanism consists of a cart which slides on a ground stainless steel shaft. The control system is composed a data acquisition card AD/DA, with resolution of 12 bits and transmission velocity of 3500 samples/second, a power stage composed of two power supply used in order to provide the necessary energy to the data acquisition card, position sensor and the cart motor and a power amplifier and finally graphic interface developed in Microsoft Visual Basic 6.0, which allows the variables visualization and the changes of set point and controller parameters.

Key words. Automatic Control, Real Time, Data Acquisition, Virtual Control

Resumen. En este trabajo se presenta el desarrollo de un sistema de control de posición de un carrito que se desliza sobre un riel dentado. El sistema de control está compuesto por una tarjeta de adquisición de datos AD/DA con una resolución de 12 bits y velocidad de transmisión de 3500 muestras/ segundo, una etapa de potencia para la alimentación de la tarjeta de adquisición de datos, del sensor de posición y del motor del carrito y un amplificador de potencia y por último una interfaz gráfica desarrollada en Microsoft Visual Basic 6.0, con la cual será posible la visualización de las variables y el cambio de set point y parámetros del controlador.

1 Introducción

El control automático tiene como objetivo, construir un sistema, tal que las variables de interés (variables controladas) se comporten en la forma deseada, aún bajo perturbaciones externas e internas. Esto es llevado a cabo, mediante la manipulación de sus variables de entrada (variable de control ó manipulada). El primer requisito de un sistema de control es que tenga un comportamiento estable. Un segundo requisito es que tenga un comportamiento aceptable, es decir, que el error entre la salida y la en

trada de referencia sea aceptablemente pequeño con una velocidad de respuesta adecuada, así como una presencia reducida de oscilaciones. En el laboratorio de control de procesos, el control virtual como Matlab (Simulink, Real Time Workshop y el

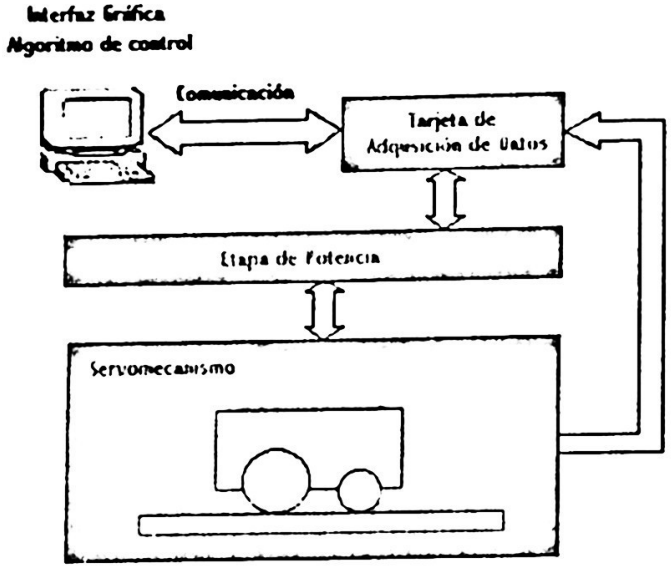


Fig. 1. Sistema de Control de posición para un servomecanismo.

Windows Tarjet) [1] y el Genie, han facilitado la implantación del control moderno y clásico [2], [3], [4] y [5], la amigable interacción entre el usuario y el proceso, permiten, la concentración del usuario en el análisis y diseño del control. El Grupo de Control de Procesos, se ha caracterizado por el desarrollo de prototipos de gran utilidad para el enlace de la teoría de control y la práctica. Este proyecto, es la primera etapa para la construcción de un sistema péndulo invertido, en esta etapa se ha logrado desarrollar la parte de adquisición de datos, la etapa de potencia, la implantación del algoritmo de control de posición y la interfaz gráfica para el carrito. El servomecanismo usado en este trabajo es de la marca Quanser Consulting [6], este consiste de un carrito, el cual se desliza sobre una barra guía lisa, por medio de un motor. El objetivo es desplazarlo a la posición especificada, cumpliendo con los requisitos mencionados arriba. Para lograr este objetivo es necesario lo siguiente: desarrollar una Etapa de Potencia que alimentará el motor, adquirir los datos como la posición y transferirlos a la PC, para que esta los procese a través del algoritmo de control y la señal de control (voltaje) sea recibida por el motor (ver Fig. 1), este artículo presenta el desarrollo de las partes del sistema de control de posición para el servomecanismo.

2 Descripción del Servomecanismo

El servomecanismo consiste en un carro deslizable sobre un riel dentado el cual esta equipado con un motor de corriente directa y un potenciómetro, acoplados al riel mediante engranajes tal como lo muestra la figura 2. El motor proporciona la fuerza necesaria para el movimiento del carro, mientras que el potenciómetro proporciona la coordenada de posición del carro sobre el riel. La posición del carro sobre el riel es la única variable que se necesita medir para la implementación del algoritmo de control.

El eje del motor está acoplado a un engrane de 0.5" de diámetro, mientras que el eje del potenciómetro se encuentra acoplado a un engrane de 1.25" de diámetro. Cuando el motor gira, el par creado en el eje de salida se transforma en una fuerza lineal, la cual origina un desplazamiento del carrito. Cuando el carrito se mueve, el eje del potenciómetro gira y el voltaje medido en sus terminales puede ser calibrado para obtener la posición del carrito dentro del riel dentado. El voltaje aplicado del motor está determinado por el control programado en la PC.

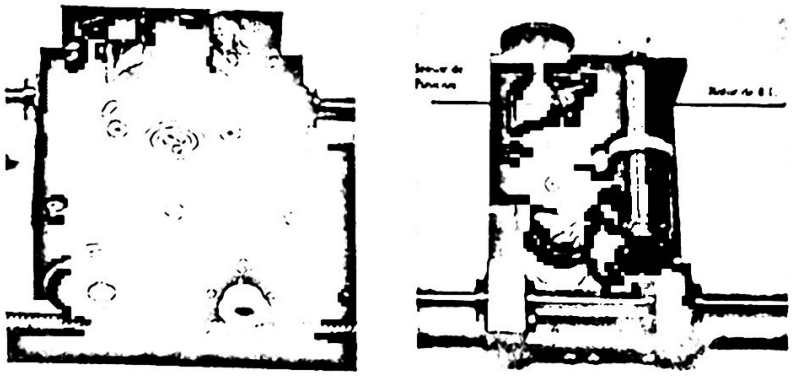


Fig. 2. Vista superior e inferior del servomecanismo.

3 Modelo Matemático

La función de transferencia del servomecanismo está dada por [7]:

$$\frac{x}{V} = \frac{1}{s(0.26s + 4.47)} \,. \tag{1}$$

Siendo V el voltaje de alimentación del motor y la ecuación que describe el control de posición es la siguiente:

$$V = K_p(x_d - x) - K_d\, x \,. \tag{2}$$

resultando así una función de transferencia de lazo cerrado de la siguiente forma:

$$\frac{x}{x_d} = \frac{K_p}{0.26s^2 + (4.47 + K_d)s + K_p} \,. \tag{3}$$

La respuesta deseada tiene un tiempo pico de 0.5 segundos con $\varsigma = 0.707$ de tal manera que: $K_p = 0.83$ V/cm y $K_d = 0.02$ V seg/cm. Para llevar a cabo la derivada de la posición, se utilizó una aproximación

$$\frac{d}{dt}x(t) \approx \frac{x(kT) - x(kT - T)}{T} \tag{5}$$

4 Tarjeta de Adquisición de Datos

El bloque de adquisición de datos consta de un microcontrolador, dos convertidores analógico-digital y un convertidor digital-analógico. Este bloque recibe la señal analógica (posición del carro) del sensor, para calcular la señal de control en la PC y enviarla al servomecanismo. La digitalización de las señales analógicas (posición y ángulo, esta última para un futuro) se realiza mediante los convertidores A/D y la conversión de la palabra digital de control a voltaje se realiza por medio del convertidor D/A. El microcontrolador se encarga de transmitir a la PC los datos obtenidos de los convertidores A/D y escribir la palabra de control al convertidor D/A. En la figura 3 se muestra un diagrama a bloques del sistema de adquisición de datos. Debido a que la señal analógica entregada por el sensor de posición varía en un rango de ±12V y para lograr una mayor precisión en las mediciones se optó por convertidores analógico/digital con entrada bipolar AD674B del fabricante Analog Devices [10], el cual tiene un tiempo de conversión de 15 μs, es compatible con el procesador y una resolución de 122 bits. Obteniéndose para una longitud de 89 cm. una resolución de 217.28

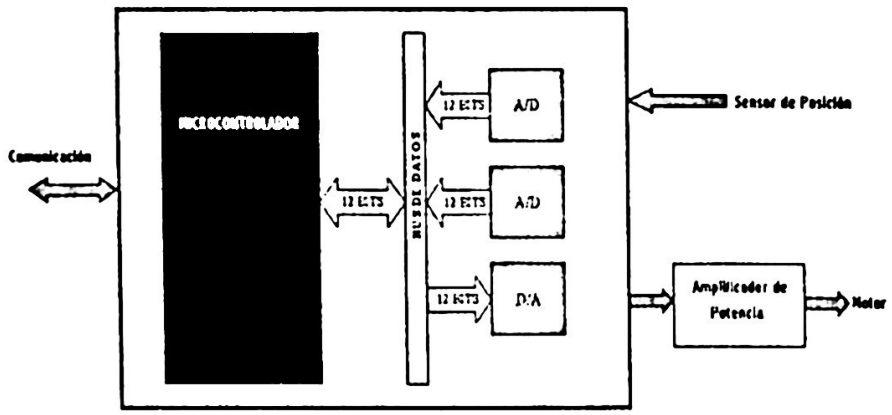


Fig. 3. Tarjeta de adquisición de datos

μm/bit. De igual manera, para lograr el movimiento del servomecanismo en ambas direcciones, la tarjeta desarrollada contiene un convertidor digital/analógico AD667 fabricado también por Analog Devices [10] con voltaje de salida bipolar de ±5V que garantiza el movimiento en ambas direcciones del motor DC cuyo voltaje máximo es de ±6V. El convertidor tiene una resolución de 12 bits, un tiempo de conversión de μs y es compatible con el procesador. Como cerebro, la tarjeta contiene el microcontrolador AVR AT90S8535 fabricado por ATMEL [9] cuyas características más importantes son: 32 líneas de Entrada/Salida programables divididas en 4 puertos (A,B,C y D) de 8 bits, 8KB de memoria Flash programable y una arquitectura RISC capaz de ejecutar hasta 8 millones de instrucciones por segundo con un reloj de 8MHz. Este microcontrolador se eligió, tomando en cuenta el número de puertos necesarios para poder manejar los dos convertidores A/D, el convertidor D/A, las líneas de control para cada uno de ellos, el bus de datos de la computadora y las líneas de protocolo en la comunicación con la PC. Para optimizar el uso de los puertos del microcontrolador, se maneja un bus de datos de 12 bits común para los dos convertidores A/D y el convertidor D/A. El manejo de los tres convertidores se realiza por medio de un multi-

plexaje en tiempo del bus, es decir, únicamente se habilita un convertidor a la vez, evitando así colisiones de datos en el bus y daños en los dispositivos. La figura 4 muestra un ejemplo del manejo del bus de datos con respecto al tiempo.

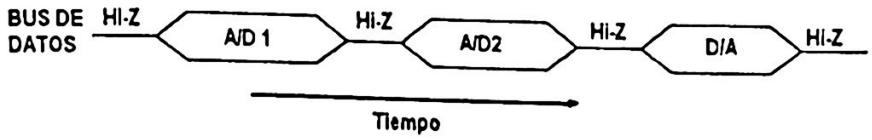


Fig. 4. Distribución de tiempo en el manejo del bus de datos.

El momento de habilitación de cada convertidor esta definido por el flujo del programa del microcontrolador al adquirir los datos. El programa del microcontrolador se encarga de controlar los ciclos de lectura y conversión tanto de los convertidores A/D como del convertidor D/A y las rutinas de comunicación con la PC por medio de ciclos de adquisición. La figura 5 muestra el diagrama de flujo de la estructura de un ciclo de adquisición programado en el microcontrolador. El primer paso que la tarjeta realiza es un ciclo de conversión y lectura del convertidor A/D para obtener la posición del carro, seguido por la transmisión de los datos adquiridos a la PC. La PC realiza el cálculo del control con los datos recibidos y lo envía a la tarjeta. El microcontrolador recibe el control y lo aplica al motor por medio de un ciclo de escritura al convertidor D/A.

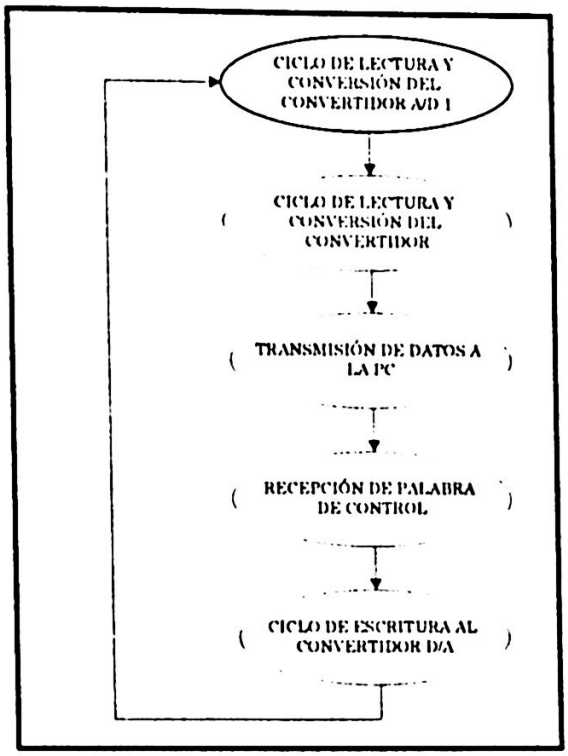


Fig. 5. Ciclo de adquisición del microcontrolador.

La comunicación con la PC del sistema de adquisición de datos se realizó por medio del puerto paralelo en modo bidireccional. La ejecución de un programa en la PC es mucho más rápida que en el microcontrolador. El protocolo de comunicación que el

sistema tiene, está estructurado de tal manera que se puedan asegurar tanto la coherencia de los datos como la dirección correcta del bus de datos.

4.1 La Interfaz Gráfica

Para la implementación del control el sistema cuenta con una interfaz gráfica en donde el usuario puede cambiar el valor de los parámetros de control y visualizar el comportamiento del sistema en forma gráfica, esta se desarrolló en *Microsoft Visual Basic 6.0* [8]. La interfaz gráfica permite visualizar la posición del carrito, la magnitud del voltaje de la señal de control, el error en la posición y las gráficas de posición y velocidad; también hace posible que el usuario pueda establecer la referencia a alcanzar, las constantes de la ecuación de control Kp y Kd y el intervalo de tiempo en que se quiere visualizar la gráfica. La figura 6 muestra la interfaz gráfica.

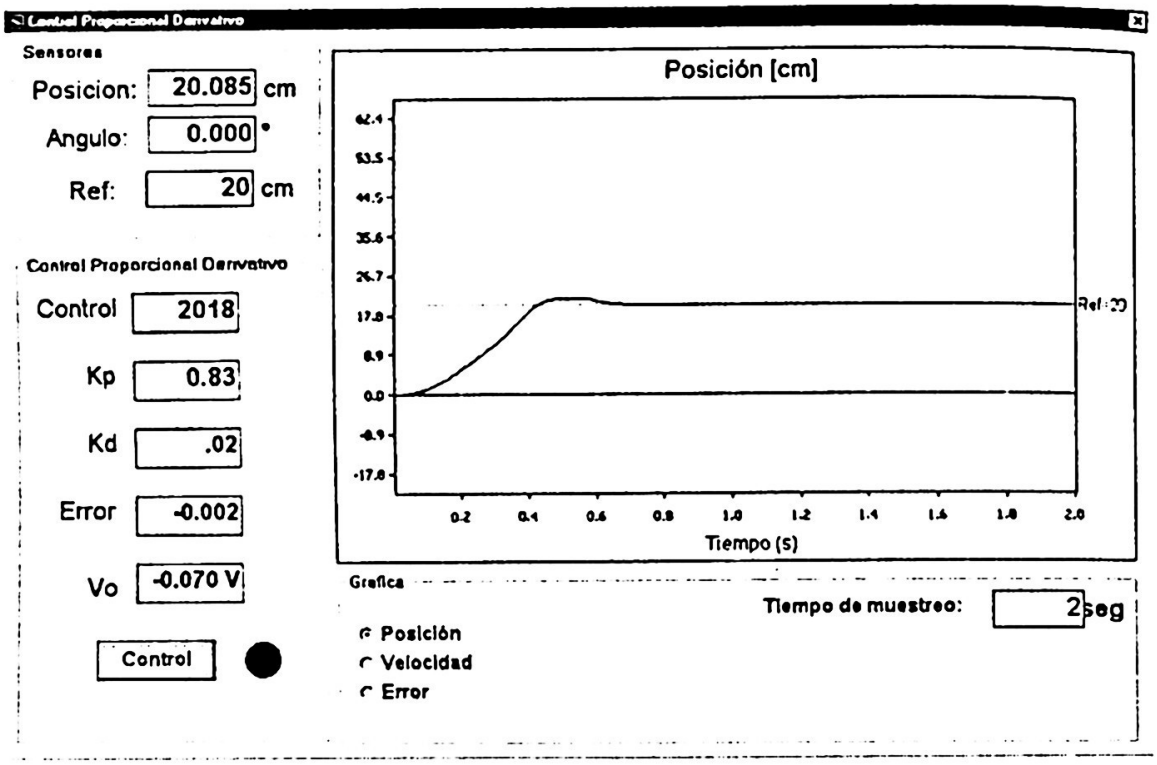


Fig. 6. Interfaz gráfica y gráfica de la posición del carrito siendo la referencia 20 cm.

Elementos de la Interfaz Gráfica:
Posición. Indica la posición del carrito.
Ángulo. Indica el ángulo de la barra del péndulo (En esta primera etapa de control este campo no se considera).
Referencia. Posición deseada.
Señal de Control. Indica la magnitud de la señal de control en código binario.
Constante de proporcionalidad. Campo para establecer el valor de la constante de proporcionalidad (0.83 por defecto) Kp.
Constante derivativa. Campo para establecer el valor de la constante derivativa Kd.
Error. Muestra el error de la posición respecto a la referencia.
Voltaje de salida. Indica la magnitud del voltaje de la señal de control.
Inicio/Paro del control. Activa o detiene el algoritmo de control.

Indicador de actividad. Este indicador cambia a color verde intermitente cuando es sistema está activo y a rojo cuando está detenido
Selección de gráfica. Selecciona la variable que se desea visualizar: posición, velocidad o error.
Tiempo de muestreo. Establece el tiempo que será mostrada la variable indicada en el campo 'Selección de gráfica'.
Área de gráfica. Al término del proceso, muestra la gráfica de la variable seleccionada.

4.2 Implementación del control

El diagrama de flujo del control de posición del servomecanismo se muestra en la figura 7. Este sensor de posición devuelve una palabra digital proporcional a la posición del carro y el bloque de polarzación BIAS establece la posición 0 (origen) del sistema al momento de que este se enciende. Cuando el sistema se enciende, las lecturas del sensor se almacenan en memoria por un lapso de tiempo mayor a 10 ms mientras se estabiliza la lectura, transcurrido dicho intervalo, la última lectura queda guardada en memoria y se restará a las siguientes lecturas del sensor dando como resultado que todas las mediciones estén referenciadas al valor guardado en memoria. Podemos ver de la figura 6 que la respuesta es buena y que coincide con lo que se esperaba según la teoría de control, tiene un tiempo pico de .5 seg., como se mencionó en la sección 3. También es posible analizar la acción de la parte proporcional y derivativa.

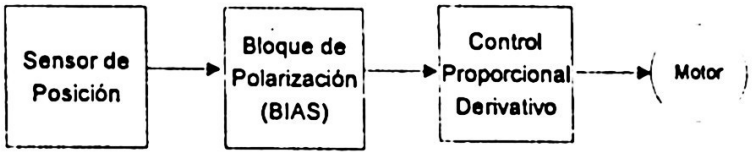


Fig. 7. Diagrama a bloques del control de posición del carrito.

5 Etapa de Potencia

El sistema cuenta además con una etapa de potencia, la cual consta de dos etapas distintas: dos fuentes de alimentación y un amplificador de potencia. La primera es una fuente de 5V a 1A, cuya función es la de alimentar todos los circuitos integrados contenidos dentro de la tarjeta de adquisición de datos. La segunda es una fuente bipolar de ±12V a 3A que proporciona, el voltaje necesario al sensor de posición y al motor del carro. Otra función de esta fuente es alimentar a los convertidores A/D y D/A. La segunda etapa consiste en un amplificador de voltaje con ganancia unitaria y con una salida de corriente de hasta 2A suficiente para garantizar el funcionamiento adecuado del motor de DC. El motor necesita una alimentación bipolar para ir de un sentido a otro; por este motivo la tarjeta cuenta con un controlador de motor implementado con una configuración de transistores, como lo muestra la siguiente figura.

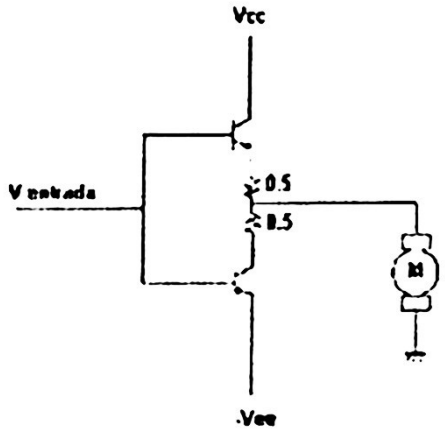


Fig. 8. Controlador del sentido del motor.

6 Conclusiones

En este trabajo se presenta el desarrollo de un sistema de control de posición de un carrito, el cual consiste una tarjeta de adquisición de datos, la etapa de potencia que alimentará el motor del carrito, la implantación del algoritmo de control de posición la interfaz gráfica con el usuario, la cual es muy amigable, ya que permite cambiar los valores de los parámetros del controlador y la referencia en una forma muy sencilla. Además de permitir la visualización del error entre la posición del carrito y la referencia, así como también la posición y la velocidad del carrito. Este trabajo, es la primera etapa para la construcción de un sistema de control de posición de un péndulo invertido, el objetivo de este control es llevar al carrito, a una posición deseada, manteniendo en posición vertical la varilla (péndulo invertido) montada en el cochecito. péndulo invertido es un sistema no lineal muy interesante, ya que presenta las características de inestabilidad y ser de fase no mínima. Pero esta primera etapa podrá servir de apoyo a los estudiantes, para encontrar un equilibrio entre la teoría y la práctica, así también encontrar acceso a la tecnología actual, como es el control virtual. Consideramos que desde el punto de vista pedagógico, será mas estimulante y valioso para estudiante, controlar un proceso en tiempo real, el cual brindará valiosas experiencias, como conocer el efecto del ruido, efectos no lineales, visualizar la retroalimentación, etc., en cambio llevar a cabo simulaciones, su aportación será limitada, ya que solo reforzará los conceptos teóricos.

Referencias

1. The MathWorks Inc, http://www.mathworks.com.
2. Siller-Alcalá I. I. ,Jaimes Ponce J., Alcántara R., Fajardo V. Bernal G.: Sistema de adquisición de datos y software de control de sistemas, en simulación y tiempo real. SOMI XVII Congreso de Instrumentación, Mérida, Yucatán Octubre 2002.
3. Siller-Alcalá I. I. ,Jaimes Ponce J., Alcántara R., Fajardo Pérez O. : Control y estudio de un módulo de transferencia de calor usando el Real Time Worksho de Matlab., XIV Reunión de Verano de Potencia, Aplicaciones Industriales y Exposiciones Industriales IEEE Sección México, Acapulco Guerrero 2001.

4. Siller-Alcalá I. I., Jaimes Ponce J., Ocampo A., Sánchez E.: Control de Posición en Tiempo Real de un Péndulo Invertido, SOMI XVI Congreso de Instrumentación, Quéretaro, Octubre 2001.
5. Siller-Alcalá I. I. ,Jaimes Ponce J., Alcántara R., Fajardo V. Bernal G.: Controles Predictivos de temperatura en simulación y tiempo real. SOMI XVIII Congreso de Instrumentación, México, D. F. Octubre 2003.
6. Quanser Consulting Inc. http://www.quanser.com
7. A comprehensive and modular laboratory for Control System Design and Implementation, Experimental Workstations for Feedback Control Education, Quanser Consulting Inc.
8. Microsoft Visual Studio 6.0 Professional Edition http://msdn.microsoft.com/.
9. Atmel Company . http://www.atmel.com
10. Analog Devices http://www.analog.com

Anexo

Se presentan los diagramas electrónicos de las etapas que conforman este trabajo.

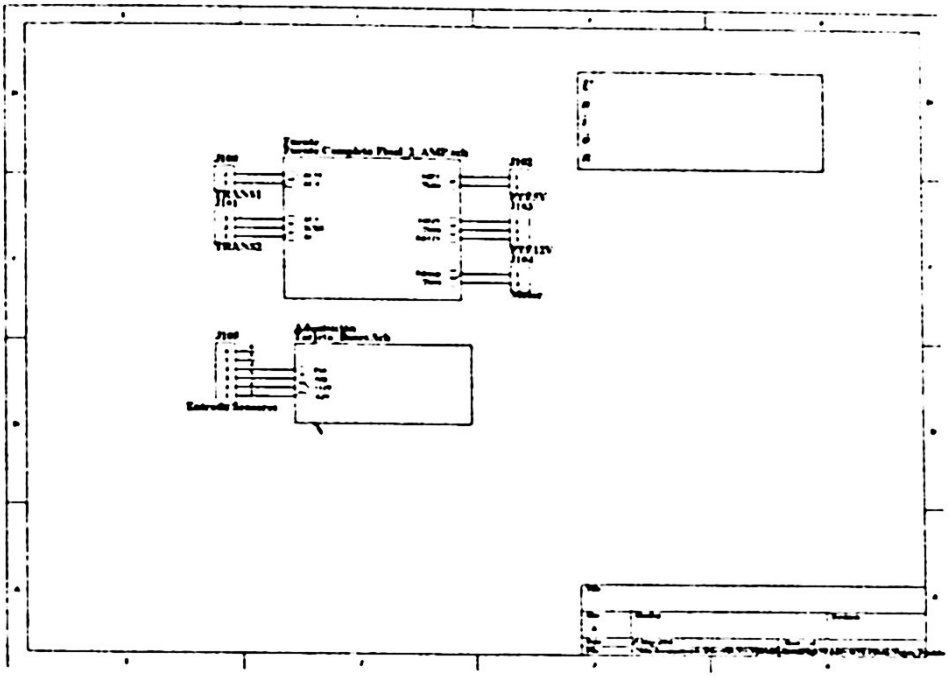


Fig. 9. Unión de la tarjeta de adquisición de datos y etapa de potencia

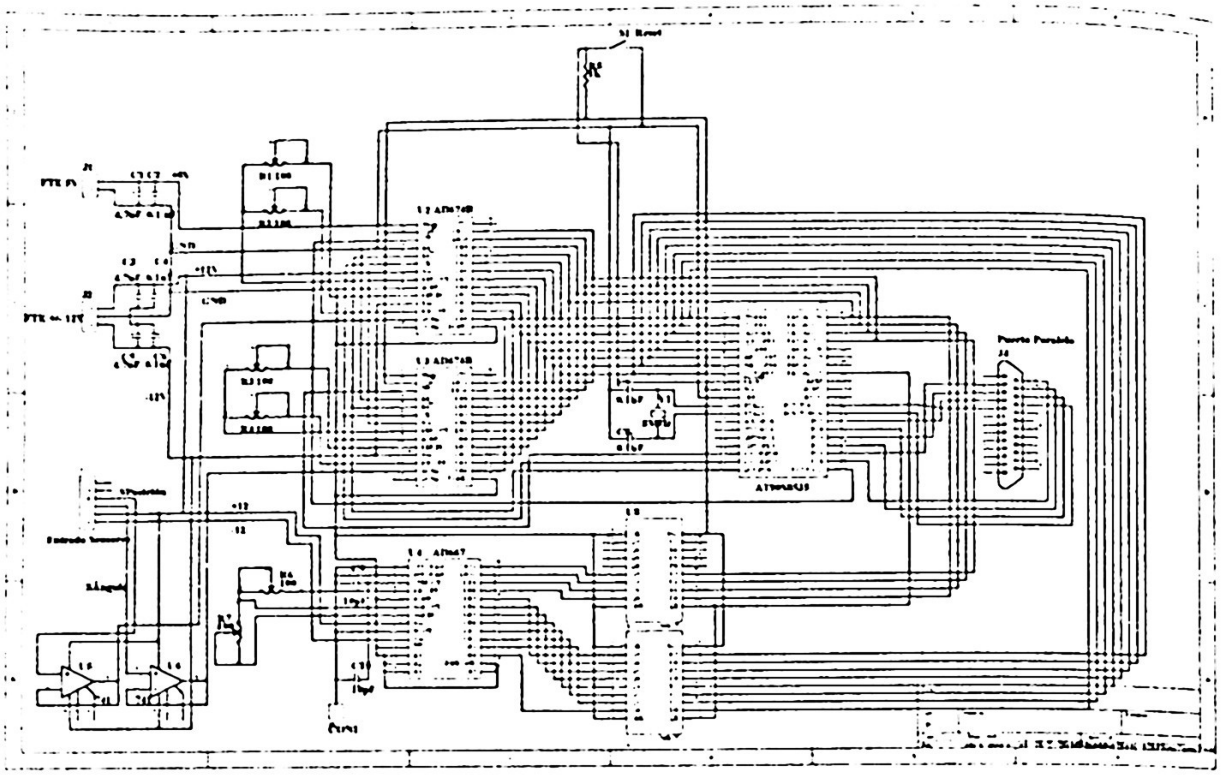


Fig. 10. Tarjeta de adquisición de datos.

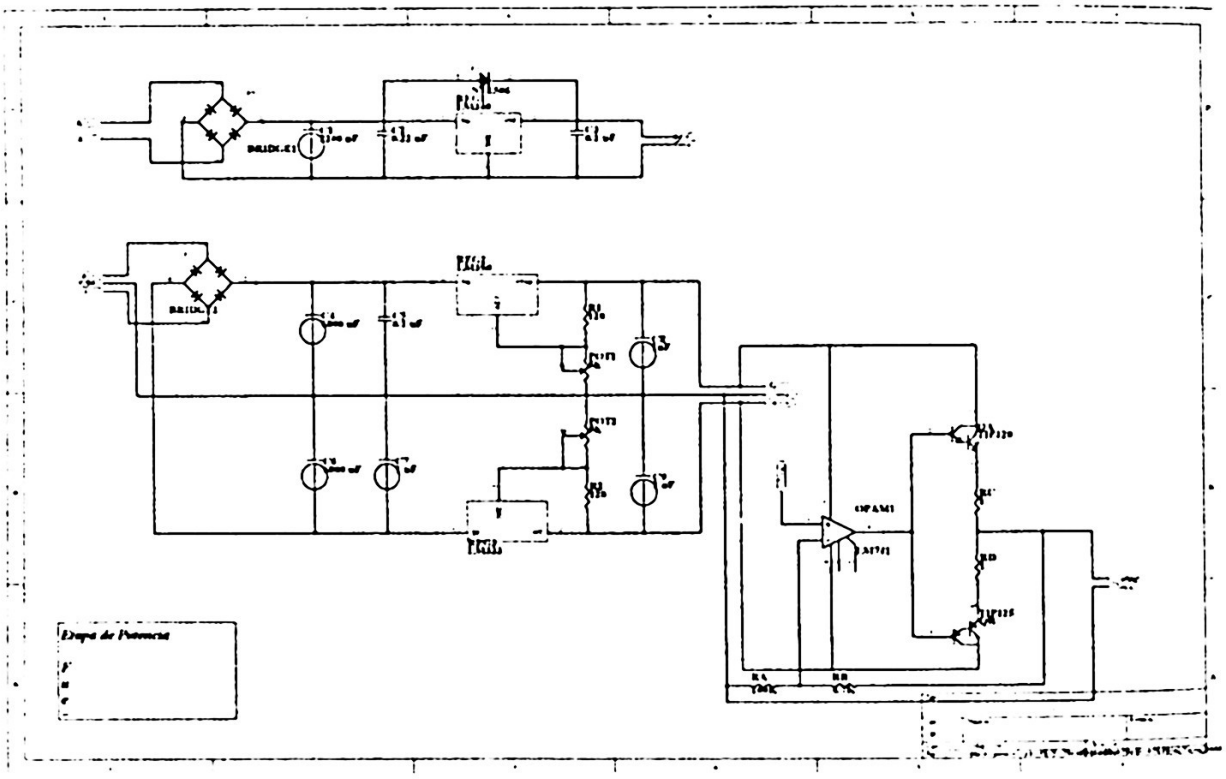


Fig. 11. Etapa de potencia.

Chattering control of DC-DC PWM boost converter

Eduardo Barba[1] and Leonardo Acho[1]

[1]CITEDI-IPN, Av. del Parque No. 131°, Mesa de Otay, Tijuana, Baja California, México C.P. 22510
{ebarba,leonardo}@citedi.mx

Abstract: In the present paper we developed a chattering PWM signal to control a DC-DC PWM Boost Converter. By chattering control we means chattering in the control law that control the high frequency switching in the relay part of the Boost converter. This chattering control produces fast change in the duty ratio from one value to another instead of smooth variation as is traditionally done. The novelty of our controller is the implementation of this chattering control law. We show experimental results.

Resumen: En el presente artículo nosotros desarrollamos un controlador de conmutación rápida en la señal del PWM para regular un convertidor DC-DC del tipo *Boost*. A este sentido, la señal PWM conmuta de un valor de ciclo de trabajo a otro en forma rápida en vez de hacerlo suavemente. La novedad de nuestra propuesta es la implementación de este control conmutado. Se muestra experimentalmente la robustez del controlador propuesto ante variaciones en la carga.

I. Introduction

High frequency switching (or chattering) controllers have been used to improve the performance of a given controller, see for instance [1], [2], [3], [4], and [5]. In [1], the idea to introduce a commutation term (or chattering term) helped to reduce the effects of friction forces on mechanical systems. This idea was bring here to show experimentally that switching controller can control DC-DC PWM Boost converter and can reduce external perturbations, as is shown in our experimental results. This chattering control was implemented to generate a chattering PWM signal producing fast changes in the duty ratio from one value to another instead of smooth variation as is traditionally done. Implementation for the perturbed case, or load variation, was realized using a parallel load, which was connected and disconnected to our load at $1Hz$. So, we have developed a new chattering control law to control the high frequency switching relay of the Boost converter. So, in these words, chattering control law is a discontinuous controller thatgenerates a chattering signal in the PWM applied to the relay term that control the Boost converter.

Roughly speaking, consider the linear system

$$\dot{y} = -ay + bu \tag{1}$$

and assume that

$$u = -\mathrm{sgn}(y) \tag{2}$$

where *sgn(.)* is the *signum* function. u can represent a signal control input. Because the *signum* function, this is called chattering term. Equation (2) into (1) produces

$$\dot{y} = -ay - b\,\mathrm{sgn}(y) \tag{3}$$

Using the next Lyapunov equation

$$V(y) = |y|, \tag{4}$$

its time derivative along the trajectory of (3) yields

$$\dot{V}(y) = \mathrm{sgn}(y)\dot{y} = -a|y| - b \leq -b$$

which, according to [6], produce finite-time convergence of *y(t)* to zero, that is,

$$\lim_{t \to t_s} y(t) = 0$$

being t_s the settling time. This finite time convergence, and due to chattering presence in (3) (the *signum* term), the system can face uncertainties or/and external perturbations as is shown in [6] and [3]. Basically, the general idea here is addition of the chattering to face with uncertainties or external perturbations. We used this, for the experimental realization to control a DC-DC PWM Boost converter.

2. Experimental Results

The proposed DC-DC Boost converter is shown in Fig. 1, where *R2* is used to model load variation that is activated in parallel with the load (*R1*) using a square signal with *f=1Hz*, and *Vcc=12Volts*. The Objective is to find a chattering PWM signal injected in "*a*" (Fig. 1), such that the dc output *Vo* (Fig. 1) be under regulation. The PWM circuit is shown in Fig. 2, where the IC TL494 is used and programmed to have sampling rate of 50kHz.

The experimental designed controller is illustrated in Fig. 3. The input to this controller is the output voltage to be regulated (*Vo*) and its output is the signal to be injected to the PWM circuit. Observe that after the error signal between the *Vo* and the *set point* appears the chattering term represented with the Operational amplifier in open loop format (see Fig. 3). The other circuits are used to adequate the signal with appropriate scale.

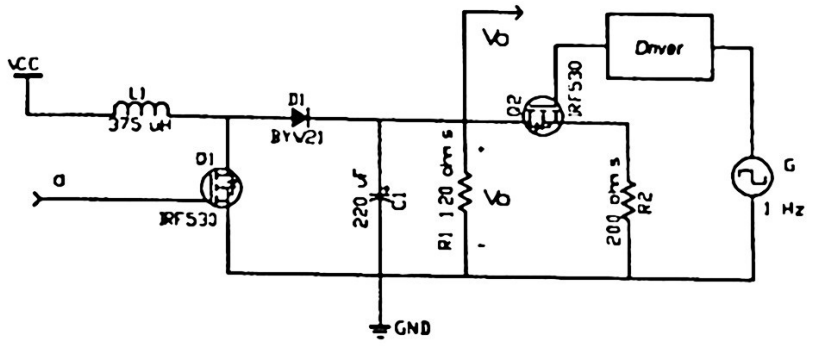


Fig.1 Proposed DC-DC Boost converter.

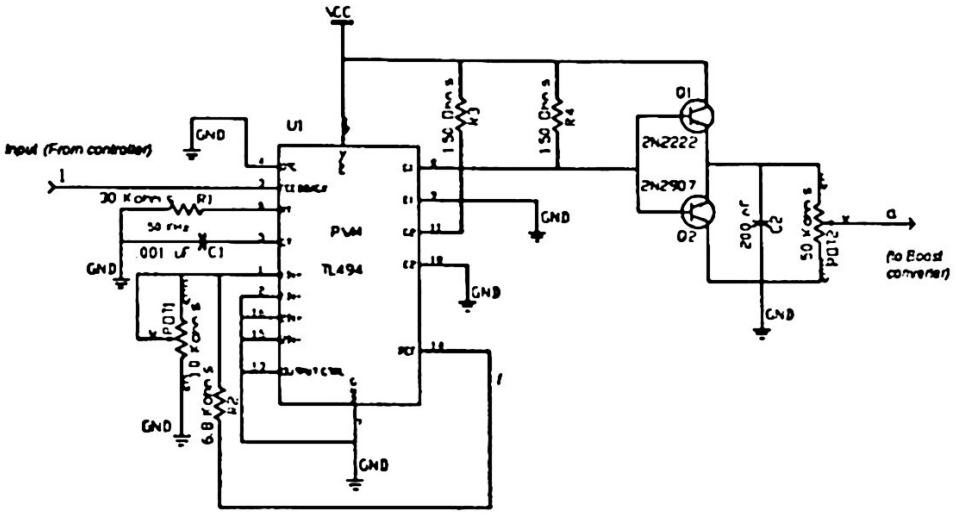


Fig. 2 PWM circuit realization.

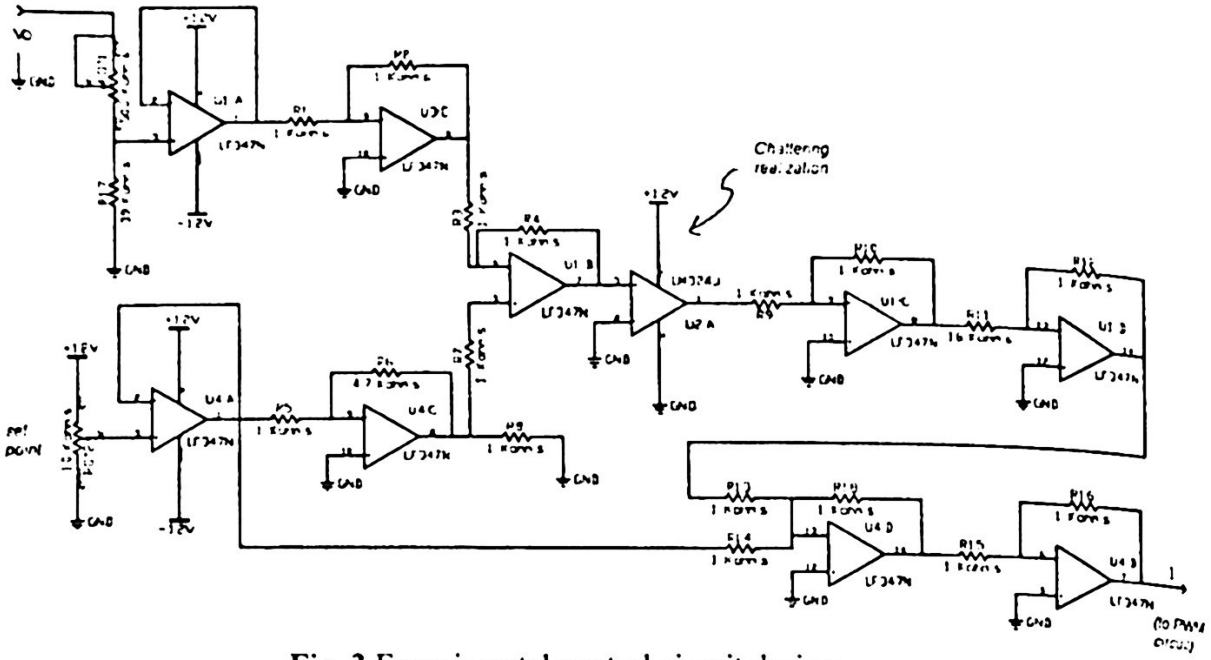


Fig. 3 Experimental control circuit design.

We programmed the *set point* to regulated *Vo* at *21 Vdc*. Fig. 4 shows *Vo* for the unperturbed case. Fig. 5 shows the PWM signal inject in "*a*". Figures 6 and 7 show the experimental results for the perturbed case (i.e., when the perturbation load connected and disconnected in parallel with the load at *1Hz*). From these experimental results we can appreciate the robustness of the chattering controller in the same manner as been illustrated for mechanical applications ([1], [2], [3] and [4]).

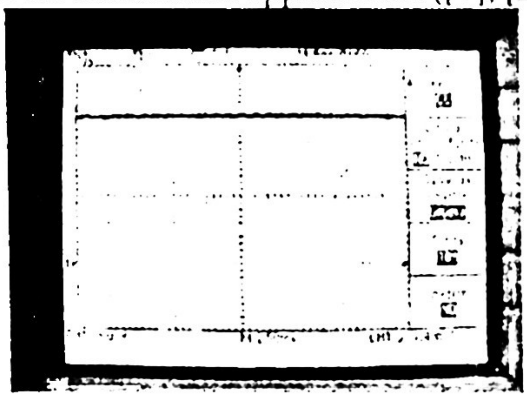


Fig. 4 Experimental result showing *Vo* (approx. 21.8 Vdc) for the unperturbed case.

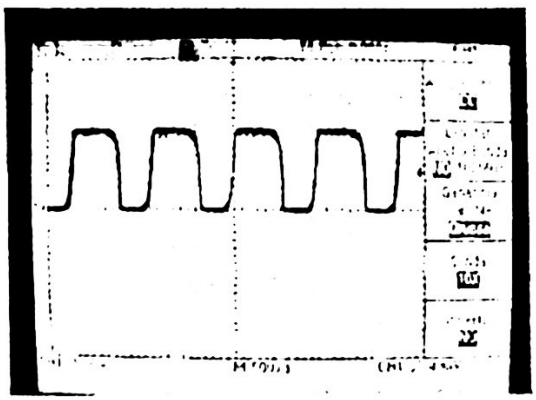


Fig. 5 PWM signal (unperturbed case), 5vdc per division.

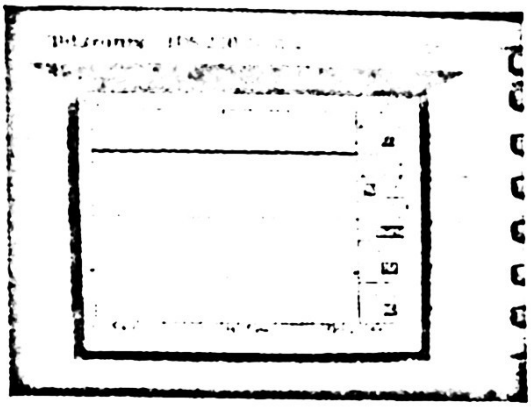


Fig. 6 Experimental result showing *Vo* approx. 21.8 Vdc (perturbed case).

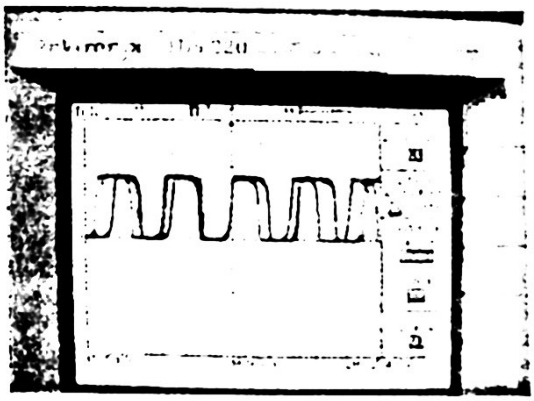


Fig. 7 chattering PWM signal (perturbed case), 5vdc per division.

References

[1] Orlov Y., Alvarez J., Acho L., and Aguilar L., "Global position regulation of friction manipulators via switched chattering control", *International Journal of Control*, Vol. 76, No. 14, 2003, 1146-1452.

[2] Rodríguez L.F. and Acho L., "Chattering Control Design for the Inverted Pendulum" *4th International Conference on Control, Virtual Instrumentation and Digital Systems*, Agosto 2002, México.

[3] Acho L., "On PID chattering control law to cope with friction forces" *IASTED Int. Conference*, 2003, Cancún, México.

[4] Rodríguez L.F., Acho L., y Moreno J., "Controlador de estructura variable con regulación adaptiva para el péndulo invertido", *IV Jornadas Iberoamericanas de Robótica*, 2003, Panamá, Panamá.

[5] Corradini M., L., Orlando G., "Transient improvement of variable structre controlled systems via multi-model swiching control", *J. of Dynamic Systems, Measurement and Control*, Vol. 124, 2002, pp. 321-326.

[6] Paden B. E., and Sastry S. S., "A Calculus for computing Filippov's Differential inclusion with application to variable structure control of robot manipulators", *IEEE Trans. on Circuits and Systems*, Vol. CAS-34, No.1, January 1987, pp. 73-82.

Configuración de un sistema de control distribuido Delta V para un sistema de calentamiento

J. C. Ravelo Hernández[2], J. Martínez Ham[1], I. I. Siller Alcalá[1], R. Acántara Ramírez[1], J. Jaimes Ponce[1], M. Cabrera Jiménez[1]

[1] UAM Azcapotzalco, Div. de CBI, Depto. de Electrónica, Grupo de Control de Procesos, Av. San Pablo No. 180, Col. Reynosa Tamaulipas, Del. Acapotzalco, C. P. 02200 México, D. F., México
{sai, raar, jjp, }@correo.azc.uam.mx

[2]Compañía Sherwin Williams, S. A. de C. V.
Poniente 140 No. 595 Col. Industrial Vallejo
C. P. 0230 México, D. F., México

Abstract: In this paper the development of the configuration of a Distributed Control System Delta V for Heating Process of resins of the Company Sherwin Williams S. A. De C. V. Obtaining with it, visualization in real time of the main process variables, showing the actual process situation, as well as it is possible to manipulate the elements that control the process. Visual warnings of events or alarms that are produced in the process are shown to give information about danger or warning situation and it manages the necessary mechanisms to guarantee an immediate response. Also it is possible to observe the process evolution through historical data, allowing to take decisions in order to improve the production.

Key words. Automatic Control, Real Time, Data Acquisition, Virtual Control.

Resumen: En este trabajo se presenta el desarrollo de la configuración de un sistema de control distribuido Delta V para un Sistema de calentamiento de resinas de la Cía. Sherwin Williams S. A. De C. V. Lográndose con ello la visualización de las variables principales del proceso en tiempo real, mostrando la situación actual del proceso, así como también es posible manipular los elementos que controlan el proceso. Avisos visuales de alarma ó eventos que se produzcan en el proceso, son mostrados para dar información sobre la situación de peligro o advertencia y gestiona los mecanismos necesarios para garantizar una respuesta inmediata. También es posible observar la evolución del proceso mediante históricos, permitiendo tomar decisiones para mejorar de la producción.

1. Introducción

La mayoría de los procesos industriales actuales, desarrollan su actividad mediante la utilización de PLC (Programmable Logical Controller) en sus cadenas de producción y/o control de fabricación. Un aspecto de vital importancia para cualquier empresa es el esfuerzo empleado en el control de calidad de sus productos, actualmente las industrias que buscan la competitividad en el mercado, utilizan sistemas automatizados basados en una computadora personal como el Sistema de Control Distribuido o SCD, los cuales permiten representar en tiempo real el estado de una planta industrial e interactuar con ella enviando valores de consigna y órdenes a los equipos de control. Gestiona las alarmas que se puedan producir, emite avisos de alarmas críticas y las muestra en todo momento, contando con una HMI (Interfaz Hombre Máquina) que es la parte visual para el operador de un proceso industrial automatizado.

Estos sistemas representan grandes ventajas operativas, tales como: Interfaz hombre-máquina muy amigable, programación gráfica orientada a control de procesos, comunicación bidireccional con equipos de campo o con otros sistemas de control, adquisición y tratamiento de datos para múltiples variables del proceso, disminución de costos y de espacio requerido para su instalación, facilidad para el mantenimiento y localización de fallas, posibilidades de expansión para un crecimiento de la planta o del proceso actual. Se pueden utilizar en aplicaciones de control lógico secuencial, control continuo y control por lotes o tipo batch, además de poder monitorear el proceso realizar diagnósticos del sistema desde un lugar remoto a través de Internet. En este artículo se presenta el desarrollo de la configuración de un el Sistema de Control Distribuido Delta V de Fisher Rosemount [3], el cual consiste en la programación para la adquisición de datos y creación de un ambiente de control gráfico en el sistema Delta V, para tener una interfase visual gráfica que permita el monitoreo permanente y en tiempo real del nivel y la temperatura de la resina en cada tanque, el arranque y paro de bombas, además de poder encender o apagar las trazas eléctricas, también se contaría con un registro o histórico de temperaturas de la resina, un indicador de la posición de las válvulas de control que regulan el flujo de aceite térmico al interior de cada intercambiador de calor, además de programar un registro de alarmas visibles audibles para la supervisión de las variables temperatura y nivel de cada tanque.

2. Área de Planta y Módulos en el Sistema de Calentamiento

Las áreas de planta, son divisiones lógicas dentro del programa Delta V para la organización de un sistema de control, las cuales pueden representar localizaciones físicas de la planta o funciones principales del procesamiento. Las áreas de planta contienen los módulos para hacer la estrategia de control. En este trabajo se creó el área planta SIS_CAL sistema de calentamiento. El control en el sistema Delta V está basado en módulos, los cuales son la entidad de lógica de control más pequeña dentro del sistema. Un módulo puede contener algoritmos, condiciones, alarmas, desplegados, información histórica y otras características que definen al equipo del proceso. En esta sección se definen los módulos de control, basado en la instrumentación y el equipo

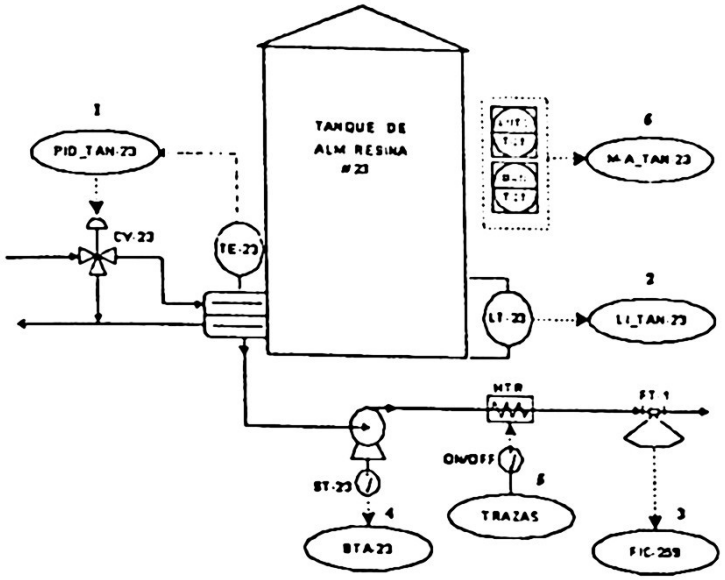


Fig. 1. Diagrama de la estrategia de control del sistema de calentamiento

de proceso. La figura 1, muestra el equipo de instrumentación y control para el proceso de calentamiento y transferencia de resina en uno de los cinco tanques, en este caso en el tanque 23. Este diagrama servirá de base para explicar como se configuró la estrategia de control de este proceso, así como el ambiente grafico de operación en el sistema Delta V. Es importante señalar que se siguió la misma filosofía o estrategia de

Tabla 1. Módulos de control y dispositivos de entrada-salida asociados

No.	Módulo de control	Descripción	*Propósito*	Tags de Entrada	Tags de Salida
1	PID_TAN-23	Lazo de control de Temperatura PID	Medir y controlar la temperatura de la resina que está en contacto con el intercambiador de calor.	TE-23	CV-23
2	LI_TAN-23	Medidor de Nivel	Medir el contenido de resina (en kg) que hay en el tanque.	LT-23	
3	FIC-259	Medidor de Flujo	Medir el flujo y la temperatura de la resina que llega al proceso.	FT-1	
4	BTA-23	Bomba de proceso	Monitorear el estado de la bomba de transferencia. (encendido/apagado)	MTR-23	
5	TRAZAS	Resistencias de Calentamiento	Mantener la Temp. de la resina en las tuberías que van al proceso.		TRAZAS
6	M-A_TAN-23	Indicación de Modo Manual o Automático	Indicar el modo de operación en el tablero local.	MAN_T-23 AUTO_T-23	

control en cada uno de los cinco tanques, por lo que solamente se explica con detalle el proceso para el tanque 23. La Tabla 1 contiene información acerca de los módulos de control y dispositivos de entrada-salida, que se utilizaron para la estrategia de control del tanque 23.

2.1 Descripción del algoritmo para el módulo de control PID_TAN-23

Para tener una visión general del algoritmo para el lazo de control de temperatura PID en el tanque 23, respecto a la relación que guarda con los registros en el PLC SLC 5/03 [1] y [2] y del Sistema Delta V, así como con la comunicación entre ambos sistemas, nos podemos apoyar en la figura 2. En este esquema se puede observar que la señal del sensor de temperatura TE-23 se conecta directamente al PLC, donde se utiliza un bloque de entrada analógica con escalamiento de la señal en un rango de –200 a 850°C. Este valor, se almacena en un registro de datos enteros de salida; luego se transmite vía comunicación serial RS-485 utilizando el protocolo MODBUS RTU a la tarjeta serial del Delta V; el valor de la señal de temperatura se almacena en un registro del dataset # 3, que es para datos de entrada, y de ahí es tomado para utilizarse en el algoritmo de control del módulo PID_TAN-23; el dato es leído por el bloque de entrada analógica AI1 que escala la señal de temperatura con un rango de 0 a 100°C para servir como señal de entrada o **PV** para el bloque PID. Una vez que el bloque PID calcula el valor de salida de control o **CV** en un rango de 0 a 100%, escribe este dato en un registro del dataset # 6, que es para datos de salida. Para poder cerrar el lazo de control, ahora se transmite este valor de salida con la tarjeta serial del Delta V hacia el PLC, escribiendo este valor en un registro para datos de entrada del PLC. Finalmente este dato es leído por un bloque de salida analógica y escalado apropiadamente. De esta manera, el PLC envía a través de un canal de la tarjeta de salidas analógicas, la señal correspondiente en un rango de 4-20 mA para manipular la válvula de control del tanque 23, a través del convertidor de corriente a presión (I/P), que escala la señal de 4 a 20 mA a un rango de salida de 3-15 psi.

2.2 Descripción del algoritmo para el módulo de nivel LI_TAN-23

La salida de este módulo indica el peso en la interfaz gráfica de operación y también es enviado al PLC a través de un registro de salida de la tarjeta de comunicación serial. En el PLC se utiliza este valor en su lógica de control, para condicionar el arranque de la bomba de transferencia en el modo automático. En este proceso, se requiere determinar el peso en kilogramos de la resina contenida en el tanque a partir de la medición de nivel, debido a que en la empresa se controlan los inventarios de materia prima de todo los tanques de almacenamiento, el peso se determina con la siguiente expresión:

$$W = \rho HA = P_h A \,. \tag{1}$$

donde W es el Peso del líquido contenido en el tanque [kg]. H es laAltura del nivel [cm]. p es el Peso específico del material [g/cm^3]. A es el área del tanque cilíndrico,

montado verticalmente = πr^2 [cm^2]. P_h es la Presión hidrostática debida a la columna de nivel del líquido = ρH [g/cm^2]

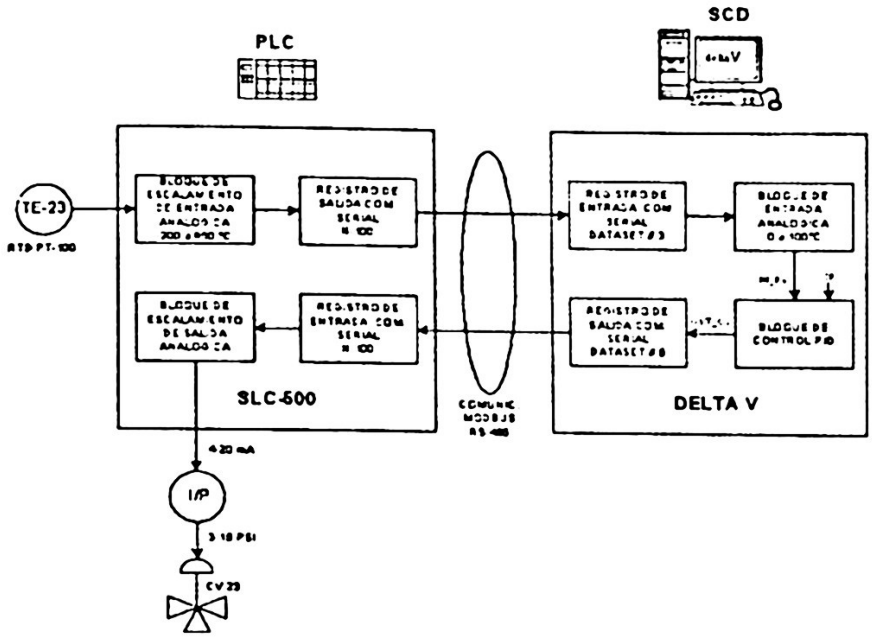


Fig. 2 Transferencia entre los registros del PLC y el sistema Delta V.

La tabla 2 contiene los valores que se establecieron para las alarmas correspondientes al 20%, 50%, 70% y 90% de la capacidad máxima del tanque en kilogramos. Estos valores se definieron en base a los criterios del control de inventarios para materia prima específicos de la empresa. Cada vez que los niveles alcancen estos puntos, se activará una alarma visual y audible en la pantalla de operación, alertando al operador del cuarto de control, quien a su vez informará al departamento de materia prima la existencia de material dentro del tanque. Cuando el nivel alcanza un 90% o más de la capacidad máxima del tanque, se activará una alarma de nivel alto-alto. Al llegar a este punto, se debe interrumpir de inmediato el proceso de llenado o recarga del tanque, debido a que existe un riesgo inminente de derrame de material, con las consecuentes pérdidas económicas y los riesgos en materia de seguridad y ecología que esto podría ocasionar.

2.3 Descripción del algoritmo para el módulo de flujo FIC-259

La señal de entrada al módulo, proviene de un medidor-controlador de flujo másico FE-1 tipo Coriolis ubicado en el área de proceso para fabricación de pintura. Este controlador, permite medir y controlar el flujo de resina que es transferida a los tanques de proceso en forma muy exacta; además de desplegar la medición de flujo localmente, también cuenta con una tarjeta de salida que retransmite la señal en un rango de 4-20mA, proporcional al flujo. Esta señal, se conecta directamente a una tarjeta de entrada analógica del Delta V. En el controlador de flujo se realiza la configuración para las unidades de ingeniería, rango de escalamiento de la señal y la variable a transmitir, ya que además de medir flujo másico, también tiene la posibilidad de medir flujo volumétrico, temperatura y densidad del producto que está fluyendo a través del

Tabla 2. Definición de valores límites para las alarmas de los cinco

Tanque	ALARMA BAJA-BAJA [kg] Inventario crítico 20%	ALARMA BAJA [kg] Punto de reorde-namiento 30%	ALARMA ALTA [kg] Preparar reorde-namiento 70%	ALARMA ALTA-ALTA [kg] Exceso de nivel 90%
23	4,546	11,364	15,910	20,546
24	4,546	11,364	15,910	20,546
25	4,769	11,924	16,693	21,462
26	4,769	11,924	16,693	21,462
27	1959	5,000	12,000	20,200

sensor. Para nuestra aplicación utilizamos la variable de flujo volumétrico en unidades de lts/min y la de temperatura en unidades de °C. Existen dos medidores de flujo para la resina en el área de producción, el FIC-258 y el FIC-259. El FIC-259 se utiliza para medir el flujo de la resina proveniente de los tanques 23 y 24, el FIC-258 para los tanques 25, 26 y 27. En la figura 3, se muestra el arreglo del medidor de flujo másico FIC-259, su interconexión mecánica con el equipo de proceso y su conexión eléctrica con el sistema de medición y control.

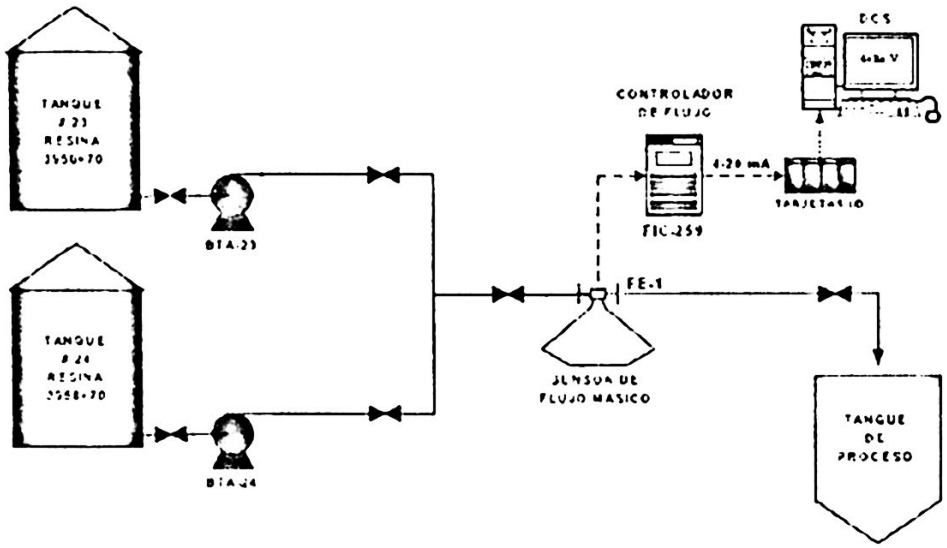


Fig. 3. Arreglo de tuberías e instrumentación

2.4 Descripción del algoritmo para el módulo BTA-23

La función de este módulo es únicamente la de monitorear el estado de la bomba de transferencia del tanque 23, es decir, si se encuentra operando o no (on/off). El bloque de entrada discreta se encarga de leer un valor discreto, es decir, un "1" ó un "0" de un canal físico asociado con una tarjeta de entradas discretas del sistema Delta V. La

señal discreta proviene de un "contacto seco" (libre de voltaje) instalado de manera auxiliar en el arrancador eléctrico de la bomba, ver figura 4. En el diagrama esquemático de la figura 4, se observa un arreglo para controlar el motor de la bomba del tanque 23, utilizando un arrancador electromagnético *trifásico*. El circuito de fuerza del motor, consta de un interruptor termomagnético para protección del circuito derivado, tres contactos principales M para conexión o desconexión y un relevador con dos elementos térmicos OL para la protección de cada fase, en caso de sobrecarga del motor. El circuito de control, consta de dos juegos de botones pulsadores para el arranque y paro del motor desde dos puntos diferentes; en este caso el motor puede accionarse desde el área de producción o donde se encuentra ubicado el tanque de almacenamiento. También se observa en el circuito de control, la bobina M del arrancador y una lámpara piloto que indica al operador de producción, si el motor de la bomba está operando y al final los contactos del relevador de sobrecarga para protección del motor.

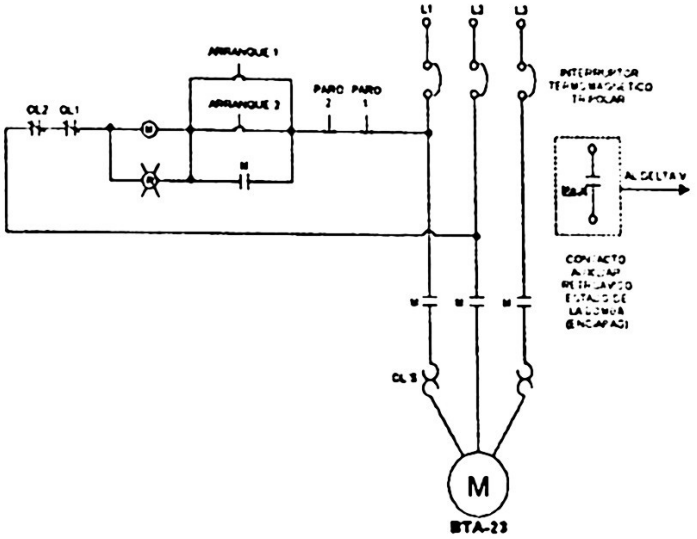


Fig. 4. Diagrama eléctrico de fuerza y control para el motor de la bomba del tanque 23.

2.5 Descripción del algoritmo para el módulo MAN-AUTO_TAN-23

El módulo MAN-AUTO_TAN-23, se creó con la finalidad de monitorear desde el cuarto de control el modo de operación automático ó manual de la bomba de transferencia de resina del tanque 23. El tablero de control local cuenta con cinco selectores en la parte frontal, para escoger el modo de operación de la bomba. En modo manual, se puede poner en marcha sin depender de los permisivos de nivel y temperatura mínimos; mientras que en el modo automático si se deben cumplir esas condiciones para que la bomba pueda funcionar. La figura 5 muestra las etapas descritas.

2.6 Descripción del algoritmo para el módulo TRAZAS

Este módulo se creó para poder controlar la operación de las trazas eléctricas. Estas son una red de resistencias instaladas en las paredes externas de las tuberías, con el fin de mantener la temperatura de la resina, mientras es transportada hasta el área de producción. En el diagrama de control de la figura 6, se puede observar la forma en que se controlan las bobinas de los contactores electromagnéticos C1, C2, y C3. Cada

contactor se controla de manera independiente por medio de un termostato, que regula la temperatura de una sección de trazas. Estos termostatos no reportan al PLC ni al Delta V, sino que se conectan directamente al circuito de control de las trazas. Sin embargo, se puede observar que la alimentación de las tres bobinas se puede interrumpir por medio del contacto CR1 que físicamente se encuentra en un canal de salida discreta del PLC; a su vez, este contacto está gobernado desde el Sistema de control Delta V, por medio de un registro de salida discreta y utilizando el canal de comunicación serial MODBUS. En el diagrama de la figura 6 se muestra el circuito de fuerza de las resistencias calefactoras (1.5 KW c/u), que forman una conexión trifásica en Estrella. La resistencia de cada fase en realidad representa un grupo de trazas distribuidas en la red de tuberías como sigue: R1 trazado para los tanques 23 y 24, R2 trazado para los tanques 25 y 26 y R3 trazado para el tanque 27. Con la finalidad de proporcionar una mayor flexibilidad y eficiencia en la operación del sistema de control para la red de trazas, deben conectarse y desconectarse desde el cuarto de control, usando el sistema Delta V. También es importante tomar en cuenta que el consumo de energía eléctrica de esta red de trazas es relativamente considerable (4.5 kilowatts), por lo que deberán operar, solo durante los días laborables de la fábrica.

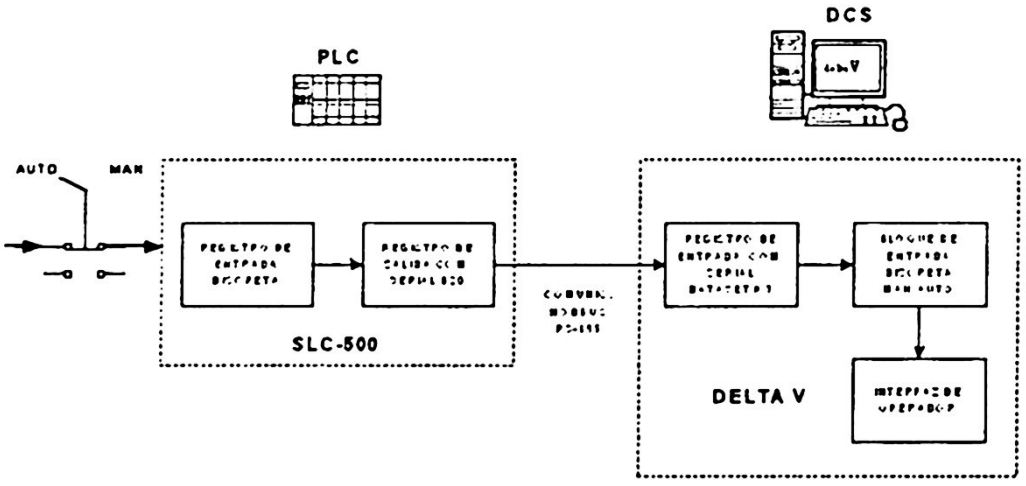


Fig. 5. Transferencia de una señal discreta del PLC hacia el DCS.

3. Desarrollo de la interfaz gráfica de operación

Teniendo como base la figura 1, en donde se muestran los módulos de control empleados en este proceso, es posible crear la interfaz gráfica de operación con la herramienta Graphic Studio del Sistema de Control Distribuido Delta V. Para la construcción del ambiente gráfico del proceso, el Delta V cuenta con las herramientas necesarias (botones para rotar, alinear un trazo, colores, plantillas que contienen una gran variedad de dibujos industriales como son motores, bombas, válvulas, tanques y tuberías), lográndose la interfaz gráfica mostrada en la figura 7. También se crearon ligas de datos, las cuales son vínculos que pueden crearse entre el ambiente gráfico y las

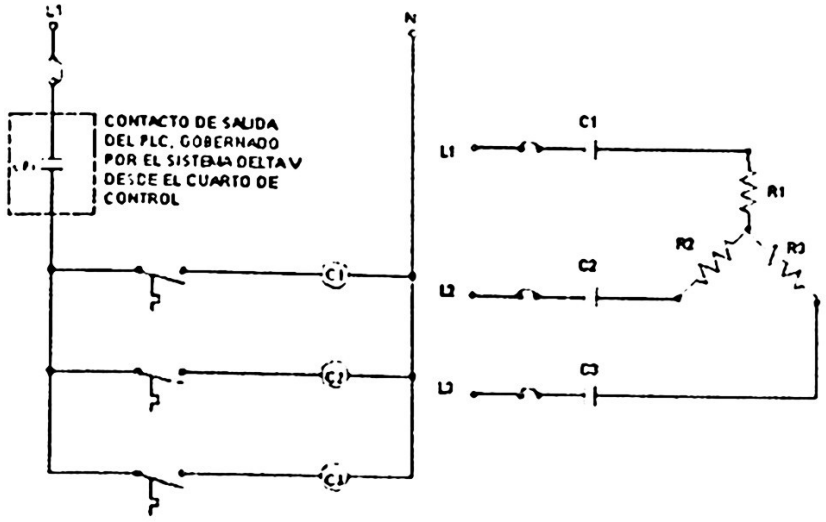


Fig. 6. Diagrama de control y fuerza de las trazas eléctricas.

variables de proceso tales como: porcentaje de apertura de la válvula, cantidad de resina en el interior del tanque, temperatura de la resina en el interior del intercambiador de calor y cantidad de resina enviada a la línea de producción. La finalidad es desplegar en la pantalla, el valor de estas variables en tiempo real. La interfaz construida en este trabajo cuenta con una serie de efectos visuales sobre un dibujo o gráfico, con la finalidad de que el operador perciba que ha ocurrido una variación en el proceso de producción que está controlando. Estos efectos pueden ser un cambio de color, el parpadeo o intermitencia, o bien aparecer y desaparecer de la pantalla.

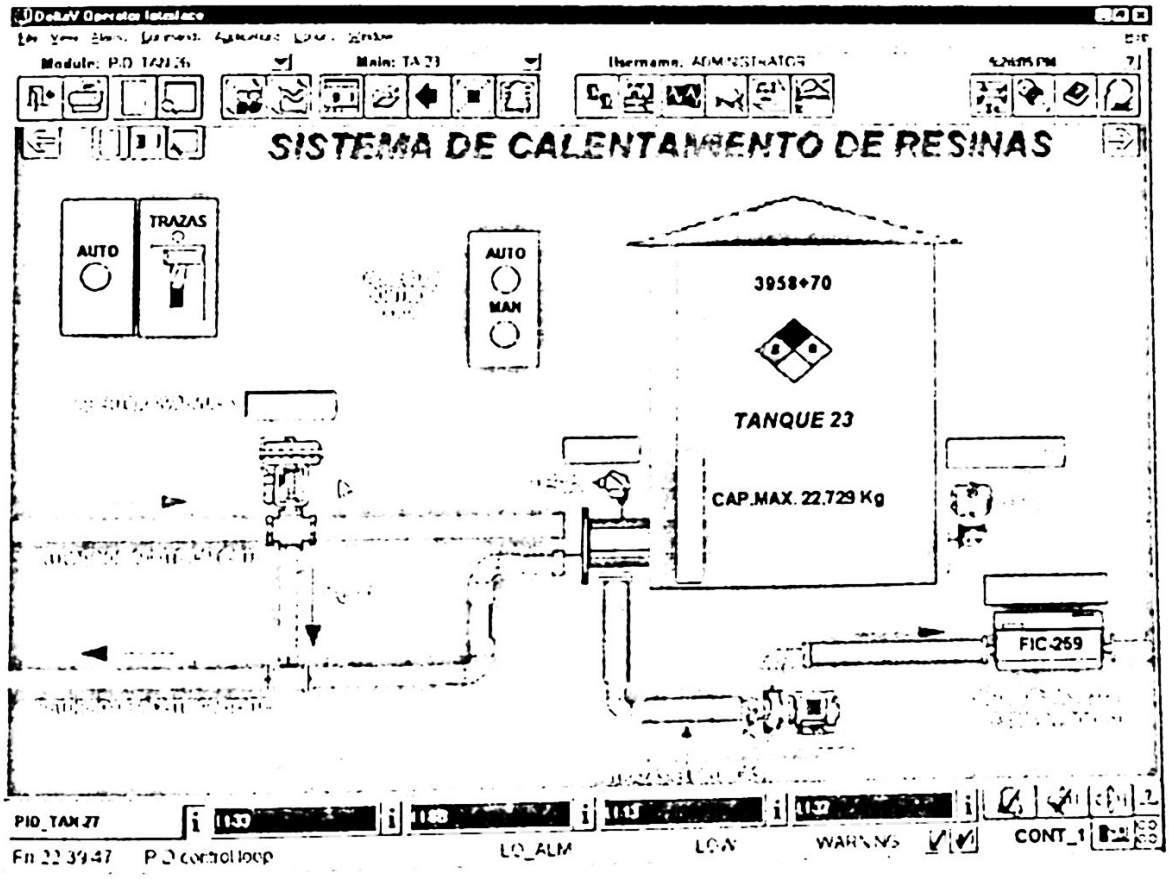


Fig. 7. Interfaz Gráfica de Operación

Además algunos dibujos pueden mostrar carátulas de control (faceplates) en donde el operador puede modificar el estado o condición de la variable de su interés, además de poder visualizarla. Esta interfaz tiene propiedades dinámicas en los siguientes dibujos: en el modo de operación e interruptor de trazas eléctricas, en el modo de operación del sistema desde el tablero de control local, en la carátula de control (Faceplate), en la barra indicadora de nivel en cada tanque, en el indicador visual de funcionamiento de trazas y bomba. Además la interfaz gráfica contiene históricos que muestran el comportamiento de variables del proceso durante un intervalo de tiempo. Estas gráficas permiten a los operadores o ingenieros tomar decisiones para mejorar la producción y realizar los ajustes necesarios a los elementos que controlan el proceso. Para este proyecto, se considera necesario, tener un registro permanente del comportamiento de cuatro variables en cada tanque de almacenamiento de resina: la temperatura de la resina al salir del intercambiador de calor, la temperatura de la resina cuando llega al área de producción, la apertura de la válvula de control, el encendido y apagado de las bombas.

4. Conclusiones

En este artículo se ha presentado el desarrollo de la configuración de un el Sistema de Control Distribuido Delta V, el cual consiste en la programación para la adquisición de datos y creación de un ambiente de control gráfico en el sistema Delta V. Lográndose tener una interfase visual gráfica que permita el monitoreo permanente y en tiempo real del nivel y la temperatura de la resina en cada tanque, el arranque y paro de bombas de acuerdo a las especificaciones señaladas. La red de trazas eléctricas se encenderá automáticamente solo en períodos de producción. Su control lo ejercerá el Delta V, sin tener relación con el PLC. También se cuenta con un registro o histórico de temperaturas de la resina, un indicador de la posición de las válvulas de control que regulan el flujo de aceite térmico al interior de cada intercambiador de calor, además de programar un registro de alarmas visibles y audibles para la supervisión de las variables temperatura y nivel de cada tanque. Existen ahora alarmas visuales y auditivas para alertar al operador de alguna condición anormal, así como un registro de las mismas.

Referencias

1. Allern Bradley. Instruction set, Reference Manual EUA Rockwell Software INC. 1999.
2. Allen Bradley SLC 500 family of small programmable controllers, EUA Rockwell Software INC. 1998.
3. Fisher Rosemount. Getting started with your Delta V software, Emerson Process Management 1999.

Modelo estocástico para tiempos de arribo en tareas de tiempo real

Daniel Cruz Pérez [1], José de Jesús Medel Juárez [2], Pedro Guevara López [3]

[1, 2] Centro de Investigación en Computación
Instituto Politécnico Nacional
Av. Juan de Dios Bátiz s/n C. P. 07738 México D. F., Tel. 57296000 Ext. 56523.
[3]pguevara@df1.telmex.net.mx, [2]jjmedel@pollux.cic.ipn.mx, [1]dcruzp@hotmail.com

Resumen: En este trabajo se presenta una discusión de los modelos de tareas (incluyendo sus arribos) en tiempo real (TTR) realizados por diversos investigadores y se propone un nuevo modelo de tiempos de arribo para TTR del tipo: a) periódicas, b) esporádicas y c) no-periódicas., al considerar a [2]. A través de ésta caracterización se obtiene dinámicamente el tiempo de arribo relativo π_k y el tiempo de arribo absoluto l_k de cada una de los tres tipos de tareas.

Palabras clave: Arribo, plazo, restricción de tiempo, tarea, tiempo real.

1. Introducción

Las TTR Periódicas son aquellas que se activan regularmente con *periodo* T_i y tiempo de cálculo c_i, *ambos* conocidos y constantes. La restricción principal es el *plazo* de ejecución (d_i) donde $d_i \leq T_i$. Las tareas periódicas son encontradas comúnmente en aplicaciones como aviones y procesos de control donde se requiere un monitoreo uniforme así como un procesamiento de datos con restricciones de tiempo y con respuestas de excelente calidad (ver: [3] y [16]).

Las Tareas No-periódicas son activadas irregularmente con un periodo desconocido. La restricción de tiempo es generalmente el *plazo* d_i [3].

Las tareas Esporádicas, de acuerdo a [4], [1], [6], [7], se caracterizan por un *tiempo de ejecución* C_i y un *tiempo mínimo de inter-arribo* T_i entre tiempos de activación. Las tareas esporádicas son asociadas con procesamiento de eventos que responden a entradas de dispositivos no periódicos; esos eventos ocurren repetidamente, pero el intervalo de tiempo entre ocurrencias consecutivas varía y puede ser arbitrariamente largo.

Liu y Layland en [6] dan las bases para el modelado de STR partiendo de conceptos diferentes a los utilizados por Martín en [8]. En su artículo se hace una mención a las TTR en donde se asumen ciertas características en el ambiente. Las tareas en tiempo real se restringen a un conjunto de condiciones: 1) se considera que las tareas críticas son las periódicas con tiempos de arribo constantes entre solicitudes, 2) el plazo consiste solamente en la restricción de capacidad de ejecución, esto es, cada tarea debe ser completada antes de que ocurra la próxima solicitud. 3) las tareas son

independientes en cuanto a que los tiempos de arribo para ciertas tareas no dependen de la iniciación o terminación de solicitudes de otras tares, 4) el tiempo de ejecución de cada tarea es constante para esa tarea y no varia con el tiempo. Este tiempo de ejecución se refiere al tiempo que se tome el procesador para ejecutar la tarea sin interrupción, 5) cualquier tarea no-periódica en el sistema es especial; estas son rutinas de inicialización o recuperación de fallas; éstas tareas desplazan tareas periódicas mientras ellas mismas son ejecutadas y no tienen restricciones de tiempo críticas o tipo "hard".

2. Modelos de tareas en tiempo real

En [6] se presenta un modelo sobre los tiempos de arribo y los plazos de las tareas, tomando como base el trabajo desarrollado por Martin en [8] quien, aunque menciona las causas por las que un sistema de esta clase puede ser lento tomando en cuenta desde los discos duros hasta los sistemas de comunicación con un modelo de petición de servicio usando teoría de colas y confiabilidad de sistemas, presenta varias deficiencias y omisiones (como los plazos) que ya han sido superadas.

En [9] se define una tarea en tiempo real como la pareja (Φ, P), donde Φ es un arreglo de tiempos de ejecución $(\phi_1, \phi_2, ...)$, y P es el tiempo mínimo de separación, el plazo de cada periodo es P después de su tiempo de arribo. También en [9] se propone un modelo "multicuadros" donde las tareas en tiempo real tipo quedan definidas con pareja (Γ, P) donde Γ es un arreglo de N tiempos de ejecución $(C^1, C^2, ... C^{N-1})$, para $1 \leq N$ y P es el tiempo mínimo de separación. El tiempo de ejecución del *i-ésimo* elemento es $C^{((i-1) \bmod(N))}$, donde $1 \leq i$. El plazo para cada instancia es P después su tiempo de arribo. Por desgracia, estos modelos son determinísticos y estáticos y debe conocer de antemano todos los tiempos de cada instancia y cada tarea.

En [10] se define un modelo de tareas esporádicas, en donde los tiempos de ejecución tienen una variación estocástica y la función de distribución probabilística de variación es conocida para el sistema. El autor hace un análisis fuera de línea estático para planificar las tareas; cada tarea esporádica es asociada con dos costos, y K_i. El primero es el costo que implica no aceptar la tarea y el segundo es el costo que implica que la tarea no cumpla con su plazo.

En [11] los autores proponen que las tareas no-periódicas τ_i (del conjunto de tareas no-periódicas $T=\{ \tau_1, \tau_2, ..., \tau_N \}$) se representan con lo siguiente

- Tiempo de arribo R_i
- Plazo absoluto D_i
- Peor tiempo de ejecución C_i
- Variable de ejecución e_i denotando el tiempo de procesamiento ya hecho para τ_i en cualquier instante de tiempo
- Variable de ejecución w_i denotando el último de comienzo o de τ_i, que es una función del tiempo actual "t" y el valor de e_i
- Tiempo de arribo menor est(i)

- Tiempo de arribo mayor lst(i)

Para las tareas esporádicas de tipo crítico se asume que el tiempo de inicio es igual al tiempo de arribo. Se define un conjunto $S=\{S_1, S_2,...,S_m\}$ como el conjunto de tareas esporádicas que tienen que ser planificadas con T. Este modelo tampoco es dinámico y es usado para planificación.

En [1] se dice que una tarea periódica se invoca con periodos regulares mientras que una tarea esporádica se invoca con períodos arbitrarios de tiempo pero este periodo tiene un valor mínimo conocido. Una tarea T es una pareja (c, p) donde: c es el tiempo de ejecución máximo de terminación de la tarea, p es el período o intervalo mínimo entre invocaciones de T. Si T es periódico, p especifica un intervalo constante entre invocaciones. Si T es esporádico, p especifica el intervalo mínimo entre invocaciones.

Para las tareas periódicas si t_k es le momento de la *k-ésima* invocación de la tarea T entonces:

- La *(k+1)-ésima* invocación ocurrirá en $t_{k+1}=t_k+p$.
- La *k-ésima* ejecución de la tarea T debe comenzar después de t_k y debe ser completada antes de t_k+p.

Para las tareas esporádicas, si t_k es el tiempo de la k-ésima de la tarea T entonces:

- La *(k+1)-ésima* invocación ocurrirá no antes de t_k+p, por lo tanto $t_{k+1} \geq t_k+p$.
- La *k-ésima* ejecución de la tarea T debe comenzar después de t_k y debe ser completada antes de t_k+p.

Se asume que las invocaciones de las tareas esporádicas son independientes y dependen solamente del momento de la última invocación.

En [12] se explica que un sistema de tareas esporádicas τ es una colección de tareas $\{T_1, T_2,...\}$. Una tarea esporádica T_i está caracterizada por tres elementos: tiempo de ejecución e_i, un plazo crítico d_i, y un periodo mínimo de separación p_i, con $e_i \leq d_i$ y $e_i \leq p_i$, de tal forma que $T_i=(e_i, d_i, p_i)$, $i \leq 1 \leq n$. En este caso, las tareas se representan con tres valores. No presenta ninguna relación temporal con las otras tareas.

Para el modelo propuesto en [13] se aclara que los tiempos de arribo y plazos son arbitrarios. El tiempo de arribo de una tarea T_i se denota con A_i, su tiempo de ejecución (posiblemente desconocido) se denota con C_i y su tiempo de respuesta máximo deseado se denota con D_i. Una tarea cumple con su plazo si termina antes de $A_i + D_i$. En el documento $A_i + D_i$ representa el plazo absoluto de la tarea y D_i su plazo relativo. El modelo no es general ya que solo toma en cuenta las tareas no-periódicas y además no propone un patrón para el arribo de tareas.

El modelo propuesto en [14] da una relación entre arribos contiguos de dos instancias en una TTR sin llegar a ser un modelo dinámico. Las consideraciones son:

- Todas la tareas τ_i: $i=1,...,n$ tienen plazos críticos;
- Todas las tareas *no-periódicas* J_i: $i=1,...,m$ no tienen plazos;
- Cada tarea periódica τ_i tiene un periodo constante T_i y un tiempo máximo de ejecución C_i, que se considera conocido y puede ser derivado por un análisis estático del código fuente;
- Todas las tareas periódicas son activada simultáneamente al tiempo $t=0$; por ejemplo, la primer instancia de cada tarea periódica tiene un tiempo de arribo $r_i(0)=0$;
- El tiempo de arribo de la k-ésima instancia periódica está dado por $r_i(k)=r_i(k-1)+T_i$;
- El plazo de la *k-ésima* instancia periódica está dado por $d_i(k)=r_i(k)+T_i$;
- El tiempo de arribo de cada tarea aperiódica es desconocido;
- El peor tiempo de ejecución de cada tarea aperiódica se considera conocido en su tiempo de arribo.

El modelo propuesto en [15] (un primer intento por parte de los autores de este artículo) se desarrollo para el análisis dinámico de predictibilidad. Consiste en el desarrollo de un algoritmo que brinde mayor información sobre el arribo y comportamiento de las TTR a través de un FDTR (Filtro Digital en Tiempo Real (ver por ejemplo: [16] y [17])) que permita reconstruir, seguir y predecir el desempeño de cada tarea, todo esto con el fin de hacer estudios de y confiabilidad de los STR y obtener mejores algoritmos de planificación en línea [3]. El modelo propuesto es:

$$x_{k+1}=a_k x_k+\omega^1_k \tag{1}$$

$$y_k=x_k+\omega^2_k \tag{2}$$

Donde los ruidos ω^1_k y ω^2_k corresponden a perturbaciones internas y externas al procesador, no están correlacionados entre si, pero si con la señal generadora de tareas y_k.

3. Modelo Estocástico para Tiempos de Arribo en Tareas de Tiempo Real

Como respuesta a los problemas expuestos en las secciones anteriores, en este trabajo se presenta un Modelo General para Tiempos de Arribo en las TTR basado en un modelo tipo recursivo que usa promedios móviles para caracterizar a las *TTR, de acuerdo a la dinámica de sus parámetros en*: a) periódicas, b) no-periódicas y c) esporádicas.

El modelo está representado por una ecuación de diferencias estocástica, de primer orden y primero grado, variante en el tiempo y no estacionario; considerando que las perturbaciones externas al procesador no están correlacionados y que obedecen a una función de distribución normal. El modelo propuesto es monovariable, esto quiere decir que solo se caracteriza el tiempo de arribo de una tarea. Se considera que los

tiempos de ejecución máximos; los plazos mínimo y máximos son conocidos; las tareas son críticas; el "jitter" está acotado dentro de un intervalo conocido.
Este modelo servirá para desarrollar un algoritmo que brinde mayor información sobre el arribo y comportamiento de las TTR a través de un Filtro Digital en Tiempo Real (ver: [16], [17]) que permita reconstruir, seguir y predecir el desempeño de cada tarea; todo esto con el fin de hacer estudios de predictibilidad y confiabilidad de los STR y obtener mejores algoritmos de planificación en línea de acuerdo a [3].

Definición 1. (Tiempo de arribo absoluto de una TTR). El tiempo de arribo absoluto l_k de una instancia con índice k de una TTR J_i está definido como el tiempo en que la instancia pide atención al procesador en relación al origen temporal de referencia.

Definición 2. (Tiempo de inter-arribo de una TTR). El tiempo de inter-arribo o tiempo de arribo relativo π_k de una instancia con índice k de una TTR J_i está definido como el tiempo en que la instancia pide atención al procesador en relación al arribo absoluto de la instancia con índice k-1.

Definición 3. (Fase de una TTR). La fase ϕ_i de una TTR está definida como el tiempo de arribo absoluto o el tiempo de inter-arribo de su primera instancia, esto es: $\phi_i = l_1 = \pi_1$.

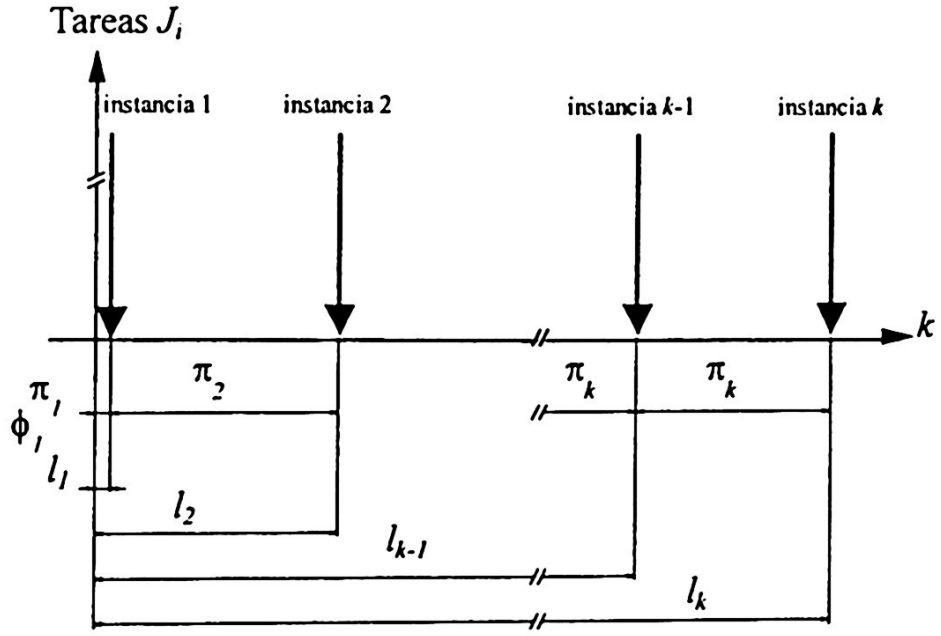


Fig. 1. Tiempos de arribo absoluto y tiempos de inter-arribo para k instancias para una tarea de TR.

Definición 4. (Tiempo de inicio de una TTR). El tiempo de inicio s_k dé una instancia con índice k de una TTR J_i está definido como el tiempo en que la instancia es atendida por el procesador en relación al origen temporal de referencia. Esto es s_k=

l_k+v_k , donde v_k es una variable aleatoria positiva denominada "jitter" y que depende del software y hardware de la computadora.

Propuesta 1. (Tiempo de arribo absoluto de una TTR). El tiempo de arribo absoluto l_k de una instancia k de una TTR J_i está descrito por:

$$l_k = l_{k-1} + \pi_k . \tag{3}$$

Propuesta 2. (Tiempo de inter-arribo en una TTR). El tiempo de inter-arribo π_k está definido por:

$$\pi_k = a_k(\pi_{k-1} - w_{k-1}) + u_k + w_k \tag{4}$$

Donde:

a_k	es el parámetro del sistema,
π_{k-1}	Tiempo de inter-arribo de la instancia con índice $k-1$,
w_k	son las perturbaciones externas al procesador, representadas a través de una variable aleatoria con distribución gaussiana,
u_k	es el tiempo de inter-arribo de referencia.

4. Caracterización del Tiempo de Inter-Arribo de Tareas Periódicas, Esporádicas y No-Periódicas

La caracterización de los diferentes tipos de TTR según su arribo se realizó utilizando el modelo de las ecuaciones (1) y (5); se utilizó el software Matlab 5.0 y se presentan las gráficas obtenidas.

Caso 1. TTR Periódica

Se hicieron las siguientes consideraciones para el modelo de una TTR periódica usando el modelo de las ecuaciones (1) y (5):

a_k	=	$0.1 \; \forall \; k$,
u_k	=	$T_k - a_k T$,
T_k	=	2 unidades de tiempo. (Período de referencia y es constante para todo k.),
w_k	=	Variable aleatoria con amplitud máxima de 0.1.

Para 100 intervalos con índice k se aprecia en la figura 4.

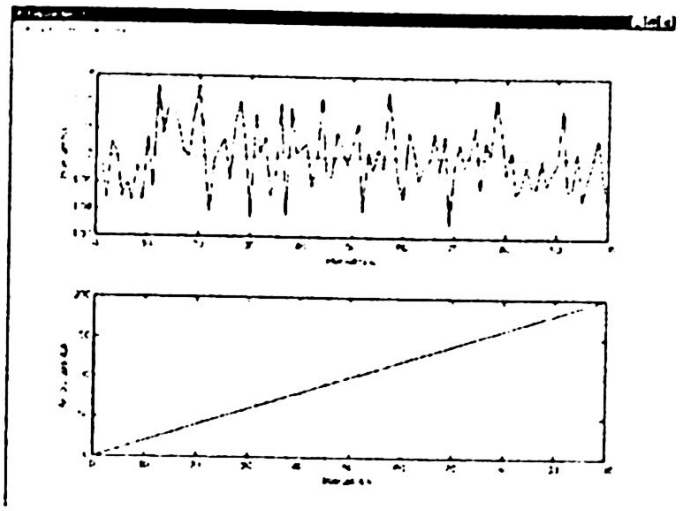


Fig. 4. Tiempos de inter-arribo (π_k) y arribo absoluto (l_k) para 100 instancias de una tarea periódica (La línea de referencia es T_k.).

Caso 2. TTR Esporádica

Se hicieron las siguientes consideraciones para el modelo de una TTR periódica usando el modelo de las ecuaciones (1) y (5):

a_k = 0.95 sin (k),
u_k = 5, es el ínfimo tiempo de arribo,
w_k = Variable aleatoria con amplitud máxima de 0.95.

Para 100 intervalos con índice k se aprecia en la figura 5.

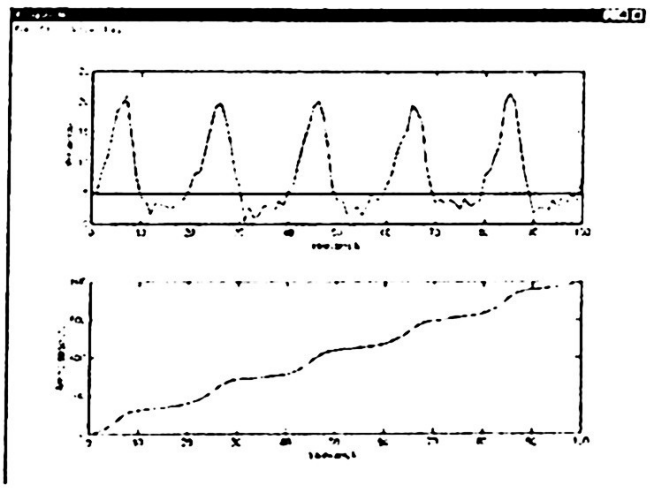


Fig. 5. Tiempos de inter-arribo (π_k) y arribo absoluto (l_k) para 100 instancias de una tarea esporádica. La línea de referencia es u_k.

Caso 3. TTR No periódica

Se hicieron las siguientes consideraciones para el modelo de una TTR periódica usando el modelo de las ecuaciones (1) y (5):

a_k = Variable aleatoria positiva con amplitud máxima de 0.95.
u_k = 5, es el mínimo tiempo de arribo,
w_k = Variable aleatoria positiva con amplitud máxima de 0.95.

Para 100 intervalos con índice k se aprecia en la figura 6.

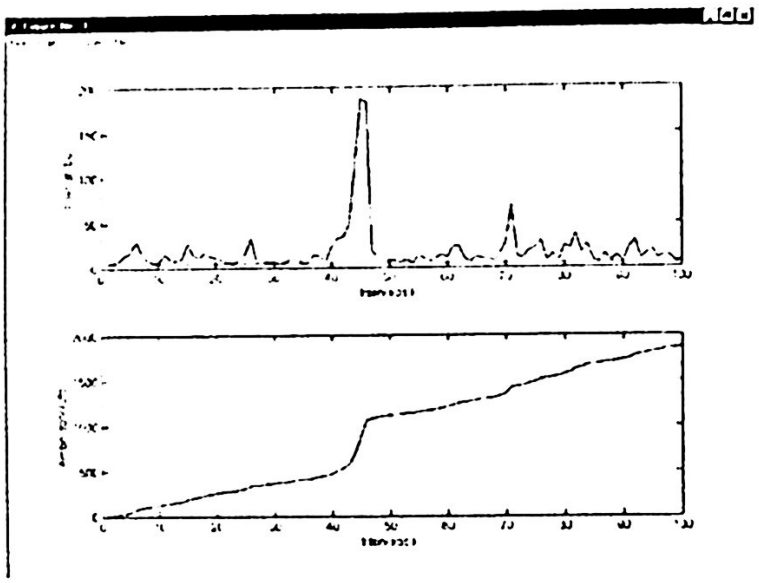


Fig. 6. Tiempos de inter-arribo (π_k) y arribo absoluto (l_k) para 100 instancias de una tarea no-periódica. La línea de referencia es u_k.

Conclusiones

La mayoría de los modelos de tiempos de arribo en Tareas de Tiempo Real son estáticos y no consideran la dinámicas internas que pueden intervenir en los arribos; por tal motivo este trabajo es el resultado de una investigación que involucra diferentes factores (errores de medición y comportamiento aleatorio generado en el medio ambiente, entre otros.) que afectan directamente al arribo de una tarea, ya sea periódica, esporádica o no-periódica. El modelo propuesto se basa en un sistema, recursivo y dinámico que consume pocos recursos y permite dar un seguimiento en línea de las tareas para posibles análisis de predictibilidad y planificación tanto en tiempos relativos como absolutos.

Bibliografía

[1] Jeffay K., Stanat D., Martel C. (1991). *On non-preemptive scheduling of periodic and sporadic tasks.* Proceedings of the Twelfth IEEE Real-Time Systems Symposium, San Antonio Texas.

[2] Liu C., Layland J. (1973). *Scheduling algorithms for multiprogramming in hard-real-time environment.* Journal of the ACM, Vol. 20, No. 4, (1982), pp273-250.

[3] Buttazzo G. (1997) *Hard real-time computing systems.* Scuola Superiore S. Anna, Kluwer Academic Publishers.

[4] Migge J. (1999). *Theoretical background.* Sophia Antipolis. http://www.sop.inria.fr/mistral/personnel/Jorn.Migge/manual/node1.html

[5] Tia T., Deng Z., Shankar M., Storch M., Sun J., Wu L., Liu J. (1995). *Probabilistic performance guarantee for real-time tasks with varying computation times.* IEEE Real-time technology applications symposium.

[6] Liu J. "*Real-time Systems*" Ed. Prentice Hall USA 2000.

[7] Rajkumar R. (1991).*Synchronization in real-time systems, a priority inheritance approach.* Kluwer Academic Publishers.

[8] Martin J. (1980). *Diseño de sistemas de computadores en Tiempo Real.* Ed. Diana 1980.

[9] Mok A., Chen D. (1997). *A general model for real-time tasks.* Technical report, University of Texas at Austin.

[10] Ramanathan P., Kang D. (1994). *A generalized guarantee model for servicing sporadic tasks with firm deadlines.* Real-time Systems Journal May 1994.

[11] Choi s., Agrawala A (1997). *Scheduling aperiodic and sporadic tasks in hard real-time systems.* Technical report University of Maryland

[12] Burns A., Wellings A. *"Real-time systems and programming languages".* University of York, Addison Wesley, 1997.

[13] Abeni L., Buttazzo G. (1999). *QoS Guarantee using probabilistic deadlines.* Proceedings of the IEEE Euromicro Conference on Real-Time Systems, York, UK, June 1999.

[14] Spuri M. & Buttazzo G. (1996). *Scheduling aperiodic tasks in dynamic priority systems.* Journal of real-time systems.

[15] Guevara P, Medel J. J. (2002). *Modelo ARMA para caracterización de tareas en tiempo real*". Taller Internacional de Instrumentación Virtual CIC-INDI 2002, Pachuca agosto de 2002.

[16] Medel J.J., Guevara P. (2002). *Caracterización de Filtros Digitales en Tiempo Real para Computadoras Digitales.* Sometido a revisión en la Revista Computación y sistemas, mayo de 2002, México D.F.

[17] Medel J.J., Guevara P. (2002). *Análisis Restrictivo para Filtros Digitales en Tiempo Real Presentando un Ejemplo para un SLIT tipo SISO.* IBERAMIA 2002, Sevilla España.

Reducción del error en la medición de velocidad en máquinas rotatorias

J. J. Medel [1], Pedro Guevara López [2], Alberto Flores Rueda [3], Asdrúbal López Chau [4]

[1,3,4] Centro de Investigación en Computación
Instituto Politécnico Nacional
Av. Juan de Dios Batiz s/n C. P. 07738 México D. F., Tel. 57296000 Ext. 56523.
[4]internal@terra.com, [1,3] (jjmedel, aflores) @pollux.cic.ipn.mx, [2]pguevara@df1.telmex.net.mx

Resumen: El medir la velocidad real de una máquina rotatoria es un problema que tiene que valorase, ya que de ello depende la eficiencia del sistema de control. Esta medición puede ser realizada utilizando un decodificador incremental: en la introducción de este trabajo se detallan dos métodos para realizar dicho cálculo. Los errores de lectura generados por los métodos seleccionados dependen de la precisión de lectura de los pulsos, de la decimación de los pulsos, de las oscilaciones de la flecha, de la exactitud de lectura de los orificios del disco montado sobre la flecha rotativa de la máquina, entre otras propiedades. Por tales motivos, se proponer dos propuestas para reducirlo al mínimo. En este trabajo se presentan en forma comparativa ambos métodos de decimación, quedando pendiente el análisis de optimalidad entre calidad de respuesta y tiempo de respuesta entre los métodos seleccionados.

Palabras clave: Tiempo de muestreo, decodificación, codificación, sistema digital, velocidad.

1. Introducción

Las medidas de velocidad de máquinas rotatorias, pueden realizarse con la ayuda de un codificador incremental. Dos de los métodos básicos de medición de velocidad angular, usando un sensor de este tipo son los siguientes:

Método I. Medir durante un intervalo de tiempo T, cuantos pulsos ha generado el codificador incremental [1].

Método II. Medir el periodo de los pulsos del codificador incremental [1].

El método I, que se discute en este trabajo, requiere más elementos de hardware, pero las lecturas son entregadas cada $T_{máx}+T_c$ segundos exactamente, donde $T_{máx}$ es el tiempo máximo entre dos impulsos del codificador incremental y T_c es el tiempo de cálculo para convertir a R. P. M. (revoluciones por minuto).

El método II, tiene la ventaja de que para ser implantado, requiere pocos elementos de hardware. La principal desventaja de este método es que el tiempo en que se producen las lecturas no es constante.

En este trabajo, se demuestra que existe un error que se genera al tratar de implantar en un microcontrolador, el método I de medición de velocidad. Se propone un algoritmo que entrega lecturas de velocidad de una manera más rápida y que puede reducir tal error de medición.

Los impulsos entregados por un codificador incremental, en general están contaminados por ruido, además, dependiendo del tipo de elemento que utilice el codificador (magnético u óptico), los impulsos pueden tener formas de señales cuadradas, triangulares o casi-cuadradas. Se considerará en este trabajo, que las señales provenientes del codificador están libres de ruido (jitter y glitch) y que son cuadradas (ciclo útil de 50%).

2. Error en la medición de velocidad

En esta parte del trabajo se describe el error producido al utilizar temporizadores o contadores de cualquier microcontrolador, este error se debe a que solamente puede ser contado un número entero de pulsos en un intervalo de tiempo, por lo que las fracciones de pulso son omitidas.

Dado un codificador incremental, es posible medir la velocidad angular de una máquina rotatoria empleando cualquiera de los dos métodos descritos anteriormente. De la formula para calcular la velocidad en R. P. M:

$$R.P.M = \frac{60 * N}{m * T} \quad . \tag{1}$$

Donde:

R. P. M Revoluciones por minuto.

N Número de pulsos contados en el intervalo T.

m Número de ranuras del codificador incremental.

T Periodo de tiempo en el que se cuentan los pulsos generados por el decodificador incremental.

Para determinar el intervalo de tiempo que el codificador incremental tarda en producir un pulso, se hace N = 1 en la ecuación (1) y se despeja T, obteniéndose la siguiente ecuación:

$$T = \frac{60}{R.P.M * m} \quad . \tag{2}$$

Lo anterior permite conocer un $T_{máx}$ y un $T_{mín}$. Estos tiempos son respectivamente el máximo y el mínimo que se necesita esperar, antes de que ocurra al menos un pulso de codificador incremental.

En la Figura (1) se muestran gráficas de la ecuación (2) para diferentes valores de m, estos últimos fueron tomados de codificadores comerciales.

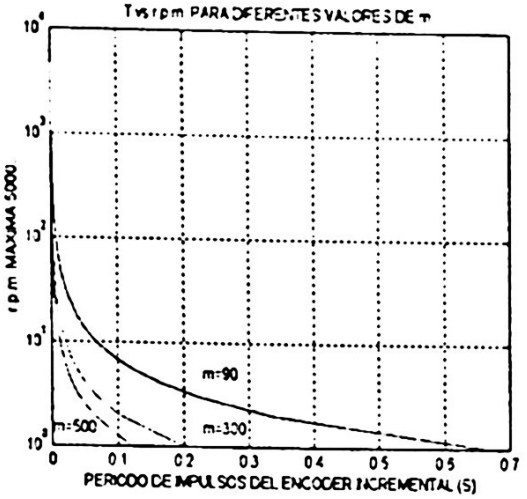


Figura (1). Gráfica de la ecuación (2) para diferentes valores de m. Donde $(R.\ P.\ M)_{min}$ Velocidad mínima que el tacómetro digital es capaz de medir. $(R.\ P.\ M.)_{máx}$ Velocidad máxima que el tacómetro digital es capaz de medir.

Se observa en la Figura (1) que T es mínimo para $(R.\ P.\ M)_{máx}$ y máximo para $(R.\ P.\ M)_{min}$, es decir:

$$T_{min} = \frac{60}{(R.P.M)_{máx} * m}, \tag{3}$$

$$T_{máx} = \frac{60}{(R.P.M)_{min} * m} \tag{4}$$

Las ecuaciones (3) y (4) son importantes, porque permiten conocer $T_{máx}$ y T_{min} antes de cualquier implantación, y por consecuencia, saber si un microcontrolador puede ser utilizado o no para la medición de velocidad.

Utilizando el método I anteriormente citado, para poder medir todo el rango de velocidades desde $(R.\ P.\ M)_{min}$ hasta $(R.\ P.\ M)_{máx}$, se requiere que el periodo del temporizador sea como mínimo $T_{máx}$.
Para microcontroladores, se puede expresar el tiempo en que un temporizador tarda en generar un evento de la siguiente manera:

$$T = T_{ck} * P * C \tag{5}$$

Donde:

T Tiempo en que el temporizador tarda en generar un evento.
T_{ck} Periodo de la señal de reloj que alimenta al temporizador.

P Pre-escalamiento que recibe la señal T_{ck}. Depende del microcontrolador a utilizar.

C Número de pulsos de duración $P*T_{ck}$ que se producen por cada evento (interrupción) que genera el temporizador.

De la ecuación (5) se tiene:

$$C = \frac{T}{T_{ck} * P} \tag{6}$$

Sustituyendo (4) en (6)

$$C = \frac{60}{m * (\mathrm{R.P.M})_{min} * T_{ck} * P} \tag{7}$$

De manera general, se puede decir que la ecuación (7) da como resultado un número real.

Por otra parte, como es bien sabido, para un temporizador con un archivo o registro contador de una longitud de n bits, el número máximo de pulsos que suceden antes de generar un evento son $P*(2^n - 1)$ (ver: Ec. (5)). Lo anterior obliga a que se cumpla la ecuación (8). Si ésta última no se puede satisfacer para cualquiera de los valores permitidos de T_{ck} y P, significa que es necesario utilizar un temporizador con un contador de longitud mayor.

$$(2^n - C) > 0 \tag{8}$$

Donde C es un número entero, lo que puede generar un pequeño error.

3. Problemas derivados de la implantación

Para darle un sentido real a las anteriores ecuaciones y mostrar el error por conteo incompleto, se utilizarán valores de un codificador incremental comercial y serán aplicadas las ecuaciones en un microcontrolador RISC AVR AT90S8515-4 de ATMEL®. Sin embargo, los resultados son *válidos ara cualquier microcontrolador.* Dado un codificador incremental con $m=500$ y si $(\mathrm{R.\ P.\ M})_{min} = 30\ R.\ P.\ M$ mientras que $(\mathrm{R.\ P.\ M})_{máx} = 5000\ R.\ P.\ M.$ Considerando que: $T_{ck} = 0.25x10^{-6}$ *seg.* (Microcontrolador AVR AT90S8515-4 con cristal a *4 Mhz.*). Sustituyendo valores en la ecuación (3), se tiene:

$$T_{máx} = 4.0 \times 10^{-3}\ seg.$$

De las hojas de especificaciones del microcontrolador [2], los valores posibles de P se muestran en la Tabla (1):

PRE - ESCALAMIENTO	
T_{ck}	$P = 1$
$T_{ck/8}$	$P=8$
$T_{ck/64}$	$P=64$
$T_{ck/256}$	$P=256$
$T_{ck/1024}$	$P=1024$

Tabla (1). Pre-escalamientos permitidos para el Timer0 del microcontrolador AT90S85185-4.

Usando la ecuación (7), se tienen los siguientes resultados.

P	C
1	16000
8	2000
64	250
256	62.5
1024	15.625

Tabla (2). Resultados de la ecuación (6).

El contador de registro TCNT0 del temporizador Timer0 del microcontrolador, tiene una longitud de 8 bits. Observando la ecuación (8), nos damos cuenta que los únicos valores permitidos son $C = 250$, $C = 62.5$ y $C = 15.625$.

Para poder cumplir con $T_{máx}$, hay que seleccionar $C=63$ y $C=16$ en los últimos dos casos.

Ahora bien, para medir la velocidad, se utilizará el **Método I**, en este caso se tienen los siguientes datos: $m=500$, $C=16$, $P=1024$, $T_{ck} = 0.25x10^{-6}seg$.

Usando la ecuación (5) se encuentra que T= 0.004096 seg.

Sustituyendo los datos anteriores en la ecuación (1) y despejando para N, se obtiene la Figura (2). En ella se puede observar que el número máximo de pulsos generados por el codificador incremental no sobrepasa 200.

Para el conteo de estos pulsos puede utilizarse ya sea un registro de propósito general del microcontrolador (incrementándolo en uno, cada vez que un pulso proveniente del codificador produce una interrupción, por ejemplo) o también se puede utilizar un temporizador en modo contador [5].

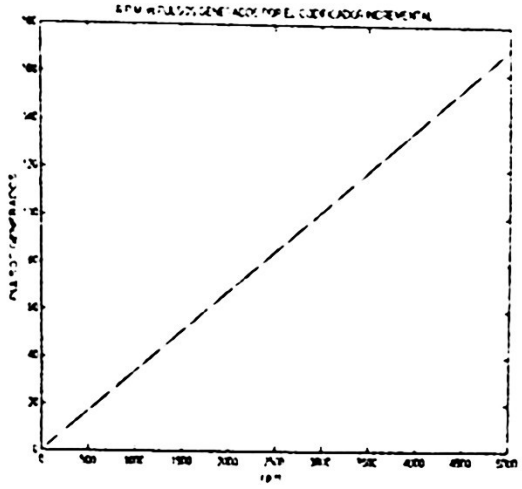


Figura (2) (Gráfica de las R.P.M. vs. pulsos generados por el codificador incremental.).

Cualquier método que se utilice para el conteo de los pulsos, tiene en común con los demás, que solo puede contar pulsos completos.

En la Figura (3), se muestra el número entero de pulsos que se contarían con el microcontrolador. Observe que la gráfica no es una línea recta como en el caso de la Figura (2), sino que ésta es escalonada.

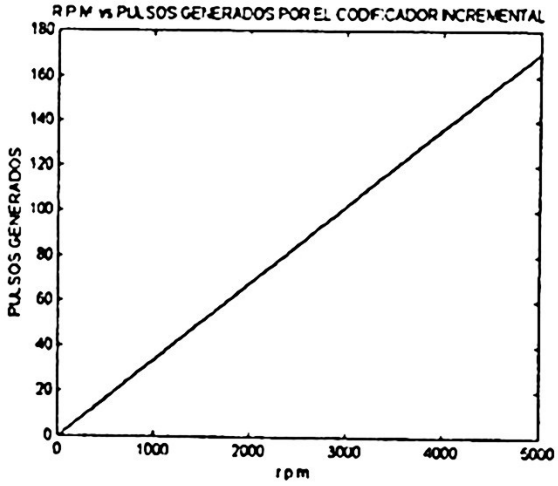


Figura 3. Pulsos contados por el microcontrolador para el rango de velocidades completo.

En la Figura (4) se muestra un acercamiento de la Figura (3) para ver el escalonamiento en el rango de velocidades de 30 a 100 R. P. M.

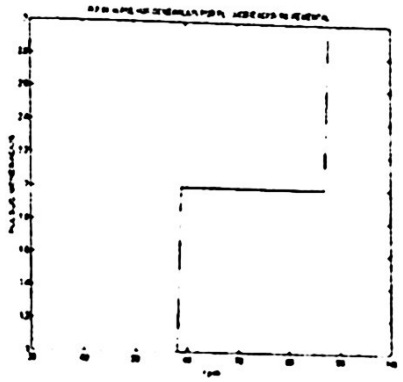


Figura 4.(Acercamiento de la Figura (3) en el intervalo de 30 a 100 R. P. M.).

Este escalonamiento se produce de manera análoga para todo el rango de velocidades. Como se observará más adelante.
La diferencia del número de pulsos reales y los valores medidos por medio del microcontrolador (Figuras (2) y (3)) se presenta la Figura (5).

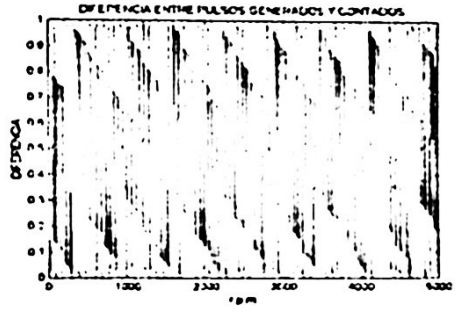


Figura 5. (Diferencia entre Número de Pulsos reales y Pulsos contados por el microcontrolador en todo el rango de velocidades a medir.).

Aunque la diferencia del error nunca sobrepasa la unidad, esto introduce un error final muy grande, como se puede observar claramente en las Figuras (6).

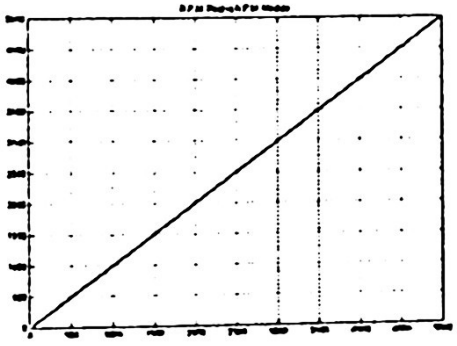


Figura 6. (a) (Gráfica de Velocidad Real vs. Velocidad Medida.).

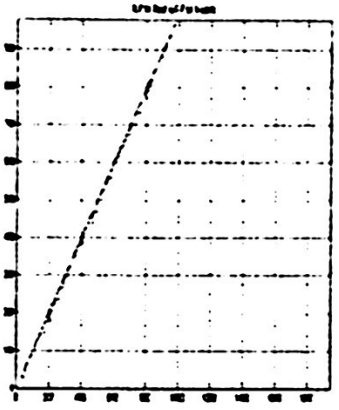


Figura 6 (b). (Acercamiento de Figura (6a) en el rango de velocidades desde 30R. P. M a 1000 R. P. M)

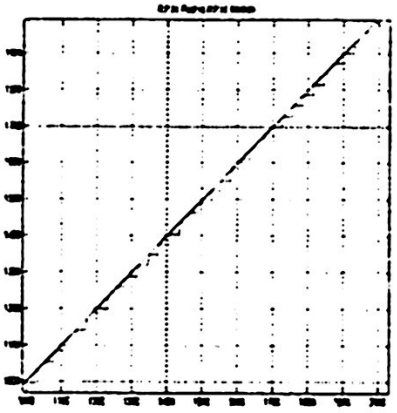


Figura 6 (c). (Acercamiento de Figura (6a) en el rango de velocidades desde 100R. P. M a 2000 R. P. M)

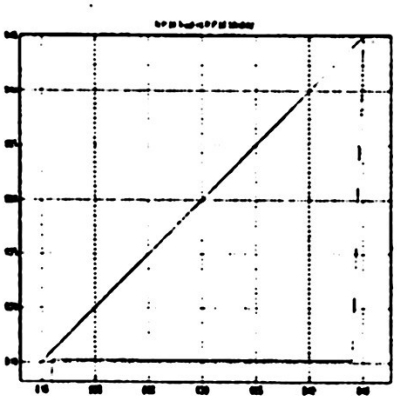


Figura 6 (d). (Acercamiento de Figura (6a) en el rango de velocidades desde 615 R. P. M a 645R.P.M.)

La Figura 6 (d) puede interpretarse de la manera siguiente:

De una máquina rotatoria que gira a una velocidad entre 620 y 640 R. P. M, junto con el codificador incremental de m=500, y si el tiempo de muestreo seleccionado es T= 0.004096 seg., ***no se podrá conocer exactamente la velocidad real*** a la que gira la máquina. En realidad, el microcontrolador dará una lectura de 616 R. P. M aproximadamente en todo ese intervalo. Esto es similar al error de cuantificación de amplitud en los muestreadores [3][4].

Lo anterior se cumple de manera similar para todo el rango de velocidades.

Con la interpretación anterior y la implantación real en el microcontrolador, hemos comprobado que existe un error en la medición de velocidad. Repetimos: los *resultados son similares* para cualquier microcontrolador o microprocesador que se utilice. Esto se debe a la *naturaleza digital de la solución.*

En la sección siguiente se describen la forma en las que se puede reducir el error mencionado. Además, se propone un algoritmo con el que se pueden obtener lecturas de velocidad de manera rápida y de precisión ajustable.

4. Propuestas para la reducción del error

La precisión requerida en las medidas de velocidad de una máquina rotatoria, es dictada por la aplicación en particular, y estará en función de los recursos de hardware con los que se cuente, además de las restricciones de tiempo que imponga el sistema (de control, por ejemplo) al que el esté conectado el medidor de velocidad.
La forma más simple de reducir el error mostrado en la sección anterior, es incrementar el valor de T de la ecuación (1). A continuación se describe una manera de poder determinar un T adecuado.

4.1 Propuesta de reducción de error con t constante

Obsérvese en la ecuación (1), que desde el punto de vista de la implantación, T es el único parámetro que puede ser modificado al intentar medir la velocidad.

De la misma ecuación (1), podemos obtener el gradiente con respecto al parámetro de interés.

$$\frac{\partial rpm}{\partial T} = -\frac{N*60}{m*T^2} \qquad (9)$$

Con la última ecuación se puede tomar un criterio para decidir la resolución en las mediciones de las R. P. M, que el microcontrolador puede medir. En otras palabras, se puede encontrar una T tal que haga que en la Figura 6 (d), la línea horizontal sea más pequeña (escalonamiento más fino, y por lo tanto, mejor resolución).

Si en la ecuación (9) se hace que los cambios sean menores que la unidad, es decir, si se obliga a que las mediciones tengan una granularidad de 1 R. P. M, obtenemos:

$$\left|\frac{\partial rpm}{\partial T}\right| \leq 1 \tag{10}$$

Sustituyendo (10) en (9)

$$\left|-\frac{N*60}{m*T^2}\right| \leq 1 \tag{11}$$

Con lo finalmente se encuentra:

$$T \geq \left| \sqrt{\frac{N*60}{m}} \right| \tag{12}$$

En la práctica hay que ajustar T, a los valores del microcontrolador, haciendo que el valor real generado, satisfaga la ecuación (13) y simultáneamente la ecuación (8).

$$T_{real} \geq T \tag{13}$$

Para los datos que se están utilizando desde el inicio de este trabajo se tiene

$$T \geq \sqrt{\frac{1*60}{500}} = 0.346 \text{ seg.}$$

En este caso se ha elegido N =1, para hacer que se cuente un pulso del codificador incremental y el proceso sea lo más rápido posible.

La Figura 7(a) muestra el valor de la velocidad real y el valor medido en el intervalo de 30 a 50 R. P. M aproximadamente. La Figura 7(b) muestra el error generado al hacer que T tome el último valor encontrado. Observe que el error se redujo notablemente.

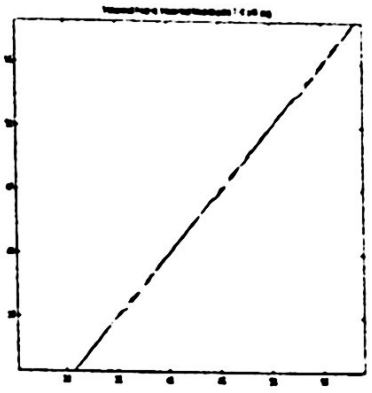


Fig. 7 (a). Gráfica de velocidad real vs velocidad medida para $T=0.346$ seg.

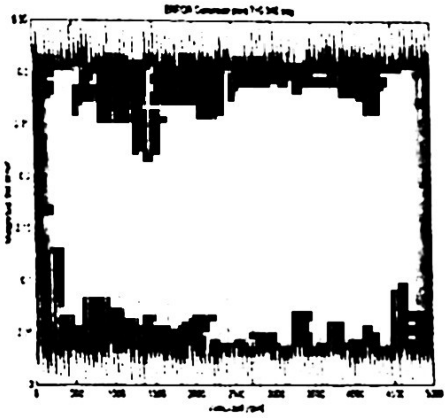


Fig. 7 (b). Error generado con $T= 0.346seg.$

Lo atractivo de este método para la reducción del error, es que **no requiere** gran poder **computacional** por parte del hardware, sin embargo, los tiempos para entregar las mediciones se hacen mayores, llegando incluso a las centésimas o décimas de segundo.

El tiempo total para la generación de lecturas puede ser reducido en un factor de $\frac{1}{\sqrt{2}}$ utilizando una compuerta XOR, con dos salidas desfasadas 90^0 eléctricos de un codificador incremental.

Si la medición de velocidad es utilizada solamente para visualización o para realimentar a un sistema de control con frecuencia de muestreo de unas centésimas de segundo, este sencillo método puede ser empleado sin mayor problema. Sin embargo, cuando se requiere que las lecturas de velocidad sean entregadas lo más rápidamente posible, es necesario recurrir a otros métodos.

Se propone a continuación un método para la obtención de velocidad con reducción del error, con la ventaja substancial de que es más rápido que el presentado anteriormente.

4.2 Propuesta de reducción de error con t variable: Algoritmo rápido de obtención de velocidad

El algoritmo para la medición de velocidad de una máquina rotatoria propuesto en este trabajo, utiliza un codificador incremental de **m** pulsos por revolución. Básicamente consiste de cinco pasos que son:

1. Proponer un T_0 inicial.
2. Contar cuantos pulsos (N_0) ha generado el codificador incremental en T_0.
3. Con los datos del paso (2), se propone un T^+_1.

$$T^+_1 = \frac{T_0}{N_0} * K_1 . \qquad (14)$$

   Donde K_1 es un número entero positivo mayor a 1.
4. Contar cuantos pulsos (N_1) ha generado el codificador incremental en T^+_1.
5. Calcular la velocidad con los últimos datos, usando la siguiente ecuación:

$$R.P.M = \frac{60 * N_1}{m * T^+_1} . \qquad (15)$$

La principal ventaja de este método sobre el presentado anteriormente, es que ***produce resultados*** de una manera **más *rápida*** y además la ***precisión puede ser ajustada variando T_0 y K_1.***

Los tiempos en los que el algoritmo puede encontrar la velocidad de una maquina rotatoria, disminuyen a medida en que esta última gira más rápido.

La desventaja de este algoritmo es que se requiere más operaciones aritméticas de punto flotante. Esta desventaja se presenta solamente en microcontroladores que no soportan operaciones de ese tipo, por lo que tienen que realizarla por software. Sin embargo, esta situación puede ser resuelta utilizando un procesador que realice operaciones de punto flotante por hardware. Observe que en el primer algoritmo presentado, T es constante (en general grande) y la utilización de un procesador más poderoso no contribuiría a reducir en tiempo de respuesta del algoritmo. Algunos resultados que arroja el algoritmo son presentados sección siguiente.

5. Algunos resultados arrojados por el algoritmo de obtención de velocidad con t variable

La simulación del último algoritmo propuesto, muestra que en general, éste entrega mediciones de velocidad en intervalos de tiempo menores que el algoritmo que utiliza un T fijo.

Por otra parte, la precisión puede ser variada ajustando T_0 y K_1. Lógicamente, entre mayor sea K_1, el tiempo en que se obtienen las velocidades también crecerá (pudiendo

ser incluso mayor al del primer algoritmo), pero la precisión mejora. En las figuras siguientes se presentan algunos resultados de la simulación.

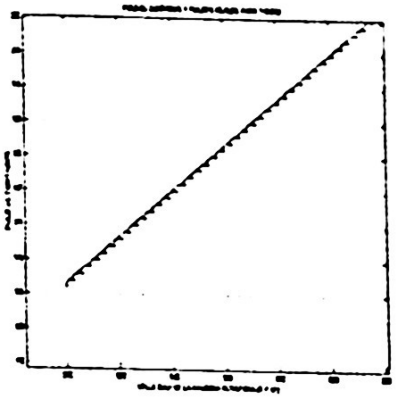


Figura 8. Pulsos reales (en rojo) generados por el codificador incremental (*m=500*). Pulsos contados (en azul) por el microcontrolador. Con T_0=*0.04seg*.

Paso 1 del algoritmo.
Se elige arbitrariamente un T_0 =0.04seg.

Paso 2 del algoritmo.
En la Figura (8) se observan los pulsos reales que genera el codificador incremental en el intervalo de 500 a 1500 rmp. También muestran los pulsos que el microcontrolador mide con T_0 = 0.04 seg. En general, existe una discrepancia entre los pulsos reales y los medidos.

Paso 3 del algoritmo.
Para cada una de las velocidades en el rango de 500 a1500 R. P. M. Se encuentra un T^+_1, que disminuye a medida que la maquina rotatoria incrementa su velocidad.
Observe en la Figura (9), que este T^+_1 es mucho más pequeño que el T encontrado en el algoritmo que utiliza T=0.346 seg. El T^+_1 servirá después como base para contar pulsos provenientes del codificador incremental.

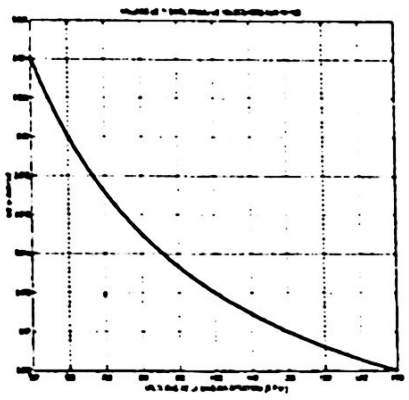


Figura 9. Valores de T^+_1 para cada una de las velocidades en el rango 500 a 1500 R. P. M. Con K_1=100.

Paso 3 del algoritmo.

Se cuentan los pulsos que genera el codificador incremental en el intervalo T^{+}_{1}. La Figura (10) muestra los pulsos generados por el codificador incremental y los pulsos medidos. Obsérvese que la discrepancia es ahora menor que la del paso 1.

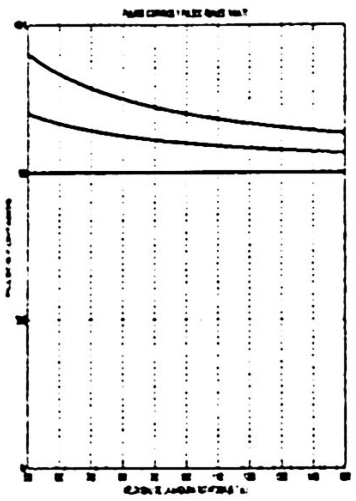


Figura 8. Pulsos reales generados (en rojo) por el codificador incremental (*m=500*). Pulsos contados por el microcontrolador (en azul).

Paso 5 del algoritmo.

Finalmente, se calcula la velocidad con los datos obtenidos.
La velocidad encontrada es muy parecida a la velocidad real a la que gira la máquina rotatoria.

En la Figura (11) pueden verse las velocidades reales y las velocidades medidas. En los casos en que el algoritmo entrega velocidades que divergen un poco de la real, siempre se localizan por debajo de la misma.

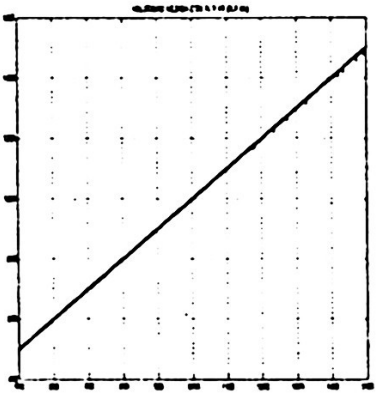


Figura 11. Velocidad real (en rojo). Velocidad medida con K_1=100 (en azul).

Un acercamiento de la Figura (11) permite verificar que las mediciones entregadas tienen una resolución bastante buena. Esto se muestra en la Figura (12).

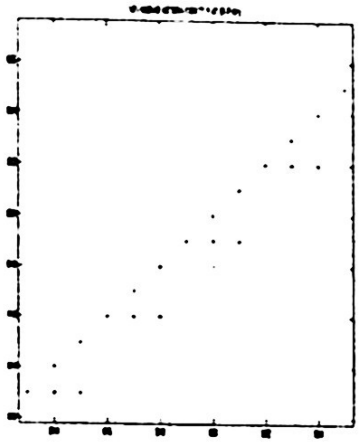


Figura 12. Acercamiento de la Figura (11) en el intervalo 814 a 824 R. P. M. Velocidad real (En rojo). Velocidad medida con K_1=100 (En azul).

La diferencia entre la velocidad real y la velocidad medida con el algoritmo, se presenta a continuación en la Figura (13).

Para la mayor parte del rango de velocidades, existe una diferencia de una o dos R. P. M con respecto a la velocidad real. Mediciones que pueden utilizadas en un sistema de control para maquinas rotatorias.

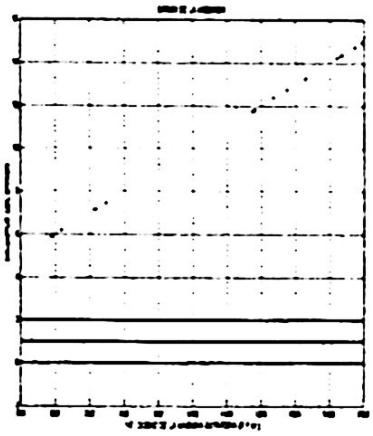


Figura 13. Diferencia entre Velocidad real y velocidad medida. Con K_1=100.

Como muestra del cambio en la resolución cuando se modifica cualquiera de los parámetros K_1 o T_0 del algoritmo, se presenta a continuación el error al hacer K_1 =200.

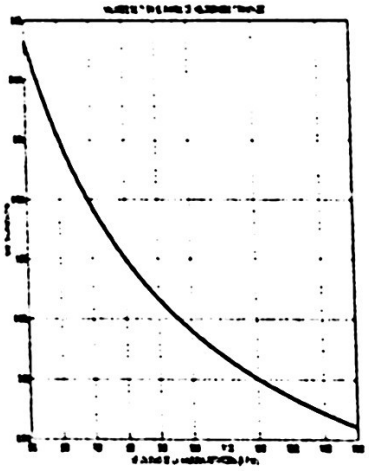


Figura 14. Valores de T^+_1 para cada una de las velocidades en el rango 500 a 1500 R. P. M. Con K_1=200.

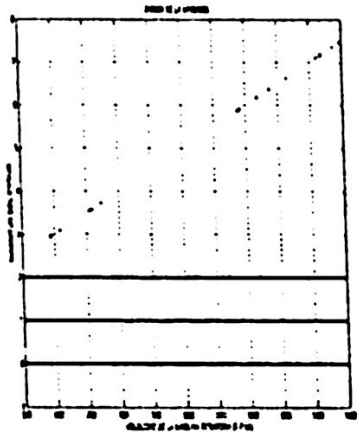


Figura 15. Diferencia entre Velocidad real y velocidad medida. Con K_1=200.

Los casos en los que existe diferencias entre la velocidad real y la medida, continúa siendo de una o dos R. P. M para la mayoría de los casos. Sin embargo, se ha disminuido en casi la mitad, el error para los otros casos.

Hay que hacer notar que T^+_1 se ha incrementado solamente el doble para K=200.
Si se modifica T_0 a 0.1 s, y se mantiene K_1=100, los resultados son los siguientes.

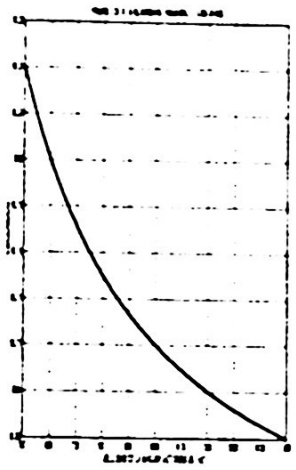


Figura 16. Valores de T^{+}_{1} para cada una de las velocidades en el rango 500 a 1500 R. P. M. Con K_1=100.



Figura 17. Diferencia entre velocidad real y velocidad medida. Con K_1=100.

Ahora la mayor parte de las mediciones se encuentran a distancias menores o iguales a 1 R. P. M. Mientras que T^{+}_{1} se ha mantenido sin cambio.

6. Conclusiones

La medición de velocidad de una máquina rotatoria puede ser realizada utilizando un codificador incremental. Dos métodos básicos para lograr esto se describen en la primera parte de este trabajo.

Al utilizar cualquier microcontrolador para implantar el método I, puede existir un error en las mediciones, debido a que solamente pueden ser contados pulsos completos. Dos propuestas para reducir este error son presentadas aquí.

La primera propuesta tiene la ventaja de no requerir muchas operaciones de punto flotante y que puede lograrse que el error sea menor a 1 R. P. M, pero el tiempo en que se entregan las mediciones es muy largo.

La segunda propuesta, reduce el error de medición y entrega resultados en tiempos cortos. Además, la resolución puede ser variada de acuerdo a las necesidades, modificando los parámetros T_0 y K_1.

El parámetro T_0 hace que la mayor parte de las mediciones se acerquen más a la velocidad real. Por otra parte, K_1 hace que las pocas mediciones en las que el algoritmo entrega resultados divergentes, no se separen mucho de la velocidad real.

7. Referencias

[1] "Sensores y Acondicionadores de Señal" 3ª edición. Ramón Payas Areny. Alfa Omega Marcombo. pag.455-456.

[2] "Avr RISC Microcontroller" DATA BOOK. AUGUST 1999. ATMEL.

[3] "Sistemas Modernos de Control, Teoria y Práctica". 2ª edicion. Richard C. dorf. Addison Wesley Iberoamenicana. pag. 508

[4] "Sistemas de Control en Tiempo Discreto" 2ª Edición. Katsuhiko Ogata. pag 8.

[5] "Microcontroladores PIC, Diseño practico de aplicaciones" José Ma. Angulo Usategui. Susana Romero Yesa. Ignacio Angulo Martínez. Mc Graw Hill

Desarrollo e implantación de un control de velocidad para motores de corriente continua usando el microcontrolador LM629

Alberto Flores Rueda[1], José de Jesús Medel Juárez[2], Pedro Guevara López[3]

[1,2] Centro de Investigación en Computación
Instituto Politécnico Nacional
Av. Juan de Dios Batiz s/n C. P. 07738 México D. F., Tel. 57296000 Ext. 56523.
[1]aflores@pollux.cic.ipn.mx, [2]jjmedel@pollux.cic.ipn.mx, [3]pguevara@dfl.telmex.net.mx

Abstract: En este artículo describe en forma resumida el diseño y construcción de un sistema de control de velocidad aplicado a 2 motores de C. D (corriente directa). El proceso del control consta: *a) Un procesador anfitrión: Una PC, que maneja un sistema de control re-entrante para dos motores de C. D., independientes., b) Sistema de control que usa al dispositivo LM629, a través del cual se realiza el ajuste de la velocidad en cada motor hasta estabilizarlos, c) Un* puente "H" formado por el dispositivo LMD18201 para control de la corriente aplicada al motor seleccionado. En este trabajo se presenta: 1) Descripción del sistema desde dos puntos de vista: de hardware y del firmware, 2) La descripción de ambas estructuras se presenta a través de módulos. 3) La prueba de control de velocidad para dos motores por medio de la implantación de una ley de control ya dispuesta en el MPID,Los resultados permiten usar la tecnología re-entrante para controlar n máquinas con características similares pero con dinámicas acotadas dentro de vecindades de operatibilidad.

Palabras clave: Microcontrolador, implantación, sistemas de control, sistemas digitales.

1. Introducción

La automatización de **n** máquinas eléctricas rotativas requiere controlar la aceleración, la velocidad, y la posición de sus flechas [3]. El controlar estos tres estados dentro de esta clase de sistemas es algo que a través del uso de circuitos integrados tales como el LM629 y el LM18201 y un algoritmo de control preseleccionado en ellos, permitirá el desarrollo de un sistema de control re-entrante para **n** máquinas con dinámicas que se encontrarán en entornos cerrados y definidos. Para ello es necesaria la descripción del sistema de control (ver: [1]-[16]).

2. Descripción del Sistema

El proceso de control consta de un procesador anfitrión (una PC) que maneja una función re-entrante controladora de dos máquinas eléctricas. El sistema en hardware cuenta con dos subsistemas en que utilizan al dispositivo LM629, y al puente "H" LMD18201 para realizar el ajuste de la velocidad en cada motor a través del control de la corriente de sus devanados.

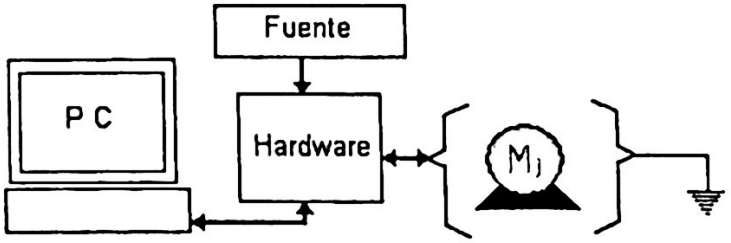


Fig. 1. Diagrama de un sistema de control para n máquinas a través de un solo sistema de hardware

2.1 Desde un punto de vista de Hardware

2.1.1 Descripción del sistema de control de velocidad

El sistema de control está definido por el procesador anfitrión estima y manda los comandos al microcontrolador LM629, en donde llega la señal observable del motor j-esimo y es decodificada dentro del MPID para actuar directamente en el filtro PID quien se encarga de corregir los valores que llegan al dispositivo de puente H y al circuito LM18201 (PH), que a su vez corrige la alimentación de corriente del motor.

El MPID, es un dispositivo dedicado para el control de motores de CD y servomecanismos usando las señales de retroalimentación de cuadratura incremental (ver la Fig. 2.) que da la flecha de la j-ésima máquina.

El LM629 tiene un bus periférico, y puede ser programado por una computadora personal, pero tiene preprogramado al filtro PID como compensador de lazo [2].

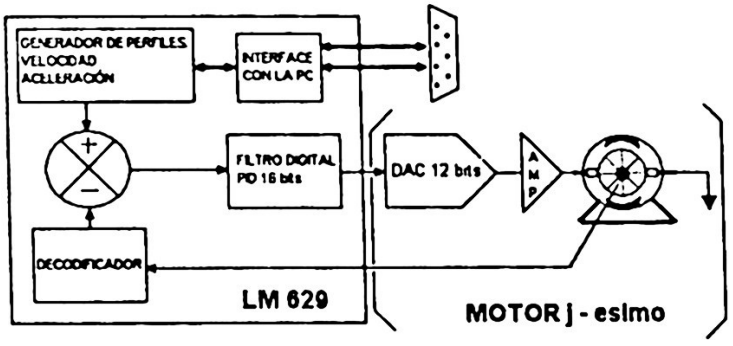


Fig.2 Sistema de control para el j – ésimo motor, basado en el LM629.

2.1.2 Descripción de la arquitectura del dispositivo MPID

En esta sección se detallan cada uno de los componentes y su uso dentro de la arquitectura del microcontrolador:

Se tiene una *ROM* de 1k x 16 bits que usa instrucciones de 16 bits, donde el algoritmo de control es almacenado. Se cuenta con un generador de secuencias *PLA* que decodifica esas instrucciones y da transferencia de señales de datos cronometradas para un bus de datos de 16 bits. Tiene una RAM donde son almacenados los parámetros de trayectoria de longitud doble (32 bits). El cual también es usado para valores de velocidad y aceleración. Y por último, tiene un ALU de 32 bits que se usa para soportar multiplicaciones de 16 x 16 bits para valores del error y coeficiente del filtro PID .

2.1.3 Descripción de la arquitectura del dispositivo puente H

El dispositivo de puente H [10][12], es de 3 amperes para aplicaciones de control de movimiento de motores de D.C. El puente H está construido en una estructura monolítica que incluye la multi-tecnología: *CMOS* para los circuitos de control, y *DMOS* para circuitos de potencia. La configuración de puente H es ideal para el manejo de motores de D.C. El puente H soporta picos de corriente de hasta 6 amperes.

2.2 Desde un punto de vista de Firmware

2.2.1 Módulos de programación

El microcontrolador maneja comandos e Instrucciones de dos bytes, un byte de 8 bits para la parte baja (desde el bit 0 hasta el bit 7) y un byte de 8 bits para la parte alta (desde el bit 8 hasta el bit 15), o sea una palabra de 16 bits[9].

El primer módulo que se requiere para una programación satisfactoria del MPID, es el módulo de prueba del *busy-Bit* : Coloca inmediatamente el bit cero en la parte baja del *byte de estado*, lo cual asegura satisfactoriamente la comunicación entre la computadora y el MPID después de que el procesador escribe un byte de comando, o lee o escribe el segundo byte del dato de la palabra (ver figura 3.). Cuando el *busy-Bit* es colocado, el MPID ignora cualquiera de los comandos o intentara transmitir un dato.

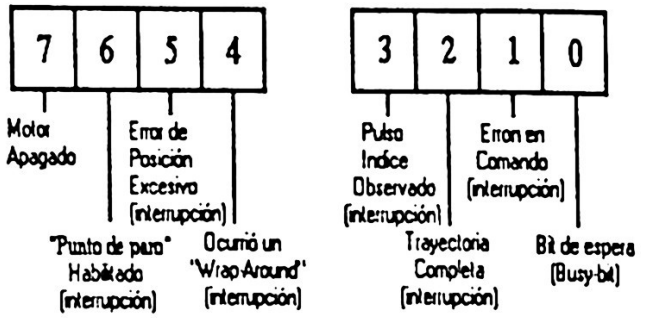


Fig. 3. Distribución de bits para interrupción del reset.

Leyendo el *Byte de estado* se cumple y se ejecuta un comando *RDSTAT*, el *RDSTAT* que es directamente soportado por el hardware del MPID, siendo este ejecutado para llevar /CS, /PS y /RD a un nivel lógico bajo.

2.2.2 Módulo de iniciación

En general, un módulo de iniciación contiene: Un comando de puesta a cero, otro de iniciación, un control de interrupción, y comandos de reportes de datos. Puede llevar el bloque de inicio para hardware y un comando PORT12.

Inmediatamente que entre la potencia, el inicio para hardware puede ser ejecutado, la puesta a cero del Busy-bit para hardware es iniciada para llevar la conexión */AST* (terminal 27), a un nivel lógico bajo. La rutina de inicio comienza después que */RST* es regresado a un nivel lógico alto, durante el tiempo de ejecución de la iniciación que es de *1.5 ms* máximo, el MPID ignora cualquier comando, o intentará transferir un dato.

2.2.3 Modulo de interrupciones

El comando *RSTI,* permite al usuario poner a cero los bits de la interrupción de bandera, unos seis bits de la palabra del byte de estado, como se puede ver en la Fig. 3., que contiene un comando *RSTI*, y una palabra de datos. El comando *RSTI,* inicia la puesta acero de los bits de la bandera de interrupción, y el comando RSTI también pone a cero el terminal de salida del interruptor de la computadora (terminal 17), enseguida del comando *RSTI*, una sola palabra de datos es escrita, el primer byte no es usado, y se usa el cambio de lógica cero en seis de los bits con lógica uno del segundo byte, iniciando las correspondientes interrupciones, como se ve en la Fig. 4. Cualquier combinación de los bits de la interrupción de bandera, serán puestos a cero en una sola secuencia del comando. Estas características permiten interrupciones que darán servicio de acuerdo a la prioridad de uso programado.

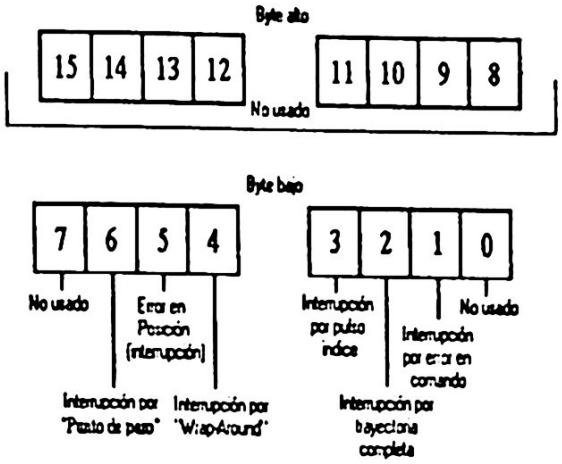


Fig. 4. Distribución de bits para interrupción del restaurador.

Consideraciones del restaurador del software.

El restaurador de software y hardware ejecutan los mismos trabajos, requiriendo el mismo tiempo de ejecución de *1.5 ms* como máximo. Durante la ejecución del restaurador del software, el MPID ignora cualquier comando, o intenta transferir un dato.

Módulo de programación del filtro

Una secuencia de un comando LFIL (carga de filtro) incluye: un comando LFIL, un control de filtro de la palabra, y un número variable de palabras de datos.

El comando LFIL inicia la carga del coeficiente del filtro, dentro de las entradas de los puertos

Los dos bytes de datos se escriben inmediatamente después del comando LFIL, que comprime la palabra del filtro de control. El primer byte del programa da el coeficiente de la muestra derivativa d_s (selección del intervalo de la muestra derivativa), y el segundo byte indica, con lógica de *unos* en las respectivas posiciones de los bits, en cuales está el remanente de cuatro coeficientes del filtro que pueden ser cargados.

Enseguida de la palabra del filtro de control, el coeficiente del filtro es escrito, cada coeficiente es escrito como un par de bytes de datos en una palabra, porque cualquier combinación de los cuatro coeficientes, puede ser cargada dentro de una sola secuencia del comando LFIL. El número de datos de las palabras, seguido de la palabra del control del filtro, puede variar en el rango de cero a cuatro.

Módulo de seguimiento de trayectoria

Una secuencia de un comando LTRJ (carga de trayectoria) incluye: El comando LTRJ, un control de trayectoria de la palabra, y una variable con un número de palabras de datos.
El comando LTRJ inicia cargando los parámetros de trayectoria en las entradas del buffer, y los 2 bytes de datos se escriben inmediatamente después del LRTJ, en los que se incluyen: la palabra de control de trayectoria, el primer byte del programa con lógica de unos en las respectivas posiciones del bit, el modo de la trayectoria (velocidad o posición), el modo de la dirección de la velocidad, y el modo de paro (ver módulo de paro), el segundo byte indica, con lógica de unos, las respectivas posiciones de cada bit, entonces los tres parámetros de trayectoria pueden ser cargados, también indica si los parámetros son absolutos o relativos, cualquier combinación de los tres parámetros puede ser cargada en una sola secuencia del comando LTRJ.

Seguido de la palabra de control de la trayectoria, el parámetro de la trayectoria es escrito, y cada parámetro es escrito como un par de palabras de datos (para bytes de datos). Para cualquier combinación, los tres parámetros podrán ser cargados en una sola secuencia del comando LTRJ.

Módulo de paro

Este módulo representa la forma requerida de pasar del estado de movimiento al estado de paro, porque el MPID opera en el modo de posición, normalmente el paro es siempre con desaceleración uniforme y ocurre automáticamente al final de una trayectoria especificada y en ocasiones es usado para desaceleración no uniforme para condiciones especiales, tales como el paro prematuro, etc.

Cuando el MPID opera en modo velocidad (considerando la desaceleración constante), el paro se realiza siempre por medio de este módulo.

Modulo de lectura para describir, posición, velocidad y aceleración

Este módulo consta de un programa maestro que utiliza los programas de los módulos de: inicialización, el del filtro, la secuencia del comando LTRJ y el comando STT describiendo los tres estados durante la trayectoria de la flecha.

Los factores que influyen el desarrollo de este programa son:
Un decodificador incremental de cuadratura, que codifica el eje de rotación en forma de pulsos eléctricos, en la Fig. 5., se detalla las señales generadas por un decodificador incremental de cuadratura de 3 canales M1, M2 e IN. El contador del MPID, decodifica la señal incremental de cuadratura para determinar la posición absoluta del eje.

La resolución de un decodificador de cuadratura integral [6], es usualmente especificado como el número de *líneas (éste* número indica el número de ciclos de la señal de salida) para cada revolución completa del eje de la flecha de la máquina eléctrica. Por ejemplo, un decodificador de *N líneas*, genera *N* ciclos de señales de salida, durante cada revolución.

Por definición, dos señales de dos canales A y B respectivamente, que están en cuadratura a 90°, o desfasadas 90° entre sí, se colocan juntas una de otra como se ve en la Fig. 5., se atraviesa por cuatro estados digitales distintos durante cada ciclo completo de cada canal. Cada transición de estado representa una *cuenta* de movimiento del eje, y el canal principal indica la dirección de rotación del eje.

Cada línea, sin embargo, representa un ciclo de las señales de salida, y cada ciclo representa *cuatro cuentas*, que se representan:

$$\left(N\frac{CICLOS}{REVOLUCION}\right)x\left(4\frac{CUENTAS}{CICLO}\right)=4N\frac{CUENTAS}{REVOLUCION} \qquad (1)$$

Si el sistema de referencia del codificador utiliza 1000 *líneas*, entonces:

$$\left(1000\frac{CICLOS}{REVOLUCION}\right)x\left(4\frac{CUENTAS}{CICLO}\right) = 4000\frac{CUENTAS}{REVOLUCION} \qquad (2)$$

Período de muestreo.

Considerando que el motor eléctrico gira a una velocidad constante, el muestreo que sucede de la posición actual del eje a una frecuencia fija, y es el recíproco del período de la muestra del sistema. Y el período de la muestra del sistema:

$$T_S = (2048)x\left(\frac{1}{T_{CLOCK}}\right) \qquad (3)$$

Período muestral del sistema.

Si el sistema de referencia usa un reloj de 8 MHz, entonces el período de la muestra del sistema de referencia que se sigue directamente al considerar a (3) es:

$$T_S = (2048)X\left(\frac{1}{8x10^6\ Hz}\right) = 256x10^{-6}\frac{s}{MUESTRA} \qquad (4)$$

Cálculo de parámetros de la trayectoria.

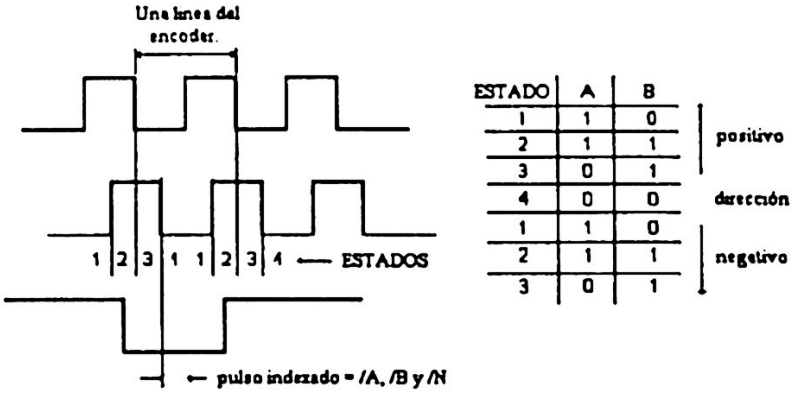


Figura 5. Señales generadas por un decodificador incremental de cuadratura de 3 canales.

Si el eje es acelerado a 0.1 rev/seg^2, hasta que alcance una velocidad máxima de 0.2 rev/seg, y entonces, desacelera hasta parar exactamente a dos revoluciones de la posición inicial Los cálculos de parámetros de la trayectoria para este movimiento son detallados enseguida.

NOTA: Una muestra en el tiempo se obtiene a través de pulsos en múltiplos de π radianes respecto al giro de la flecha (ver Fig. 6)

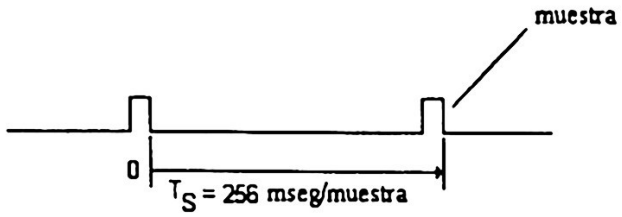


Fig. 6. Muestras obtenidas por impulsos.

Cálculo de velocidad, aceleración y posición.

A través de las expresiones (4) a (1), tenemos respecto a la aceleración

$$A = 00\ 00\ 00\ 02 \text{ hex} \qquad \frac{CUENTAS}{MUESTRA^2}$$

Ya que:

$$A = \left(4000 \frac{CUENTAS}{REVOLUCION}\right) x \left(256x10^{-6} \frac{SEGUNDOS}{MUESTRA}\right)^2 x \left(0.1 \frac{REVOLUCIONES}{SEGUNDO^2}\right) =$$

ada

$$A = 2.62x10^{-5} \frac{CUENTAS}{MUESTRA^2}$$

$$A = \left(2.62X10^{-5} \frac{CUENTAS}{MUESTRA^2}\right) x (65536) = 1.718 \frac{CUENTAS}{MUESTRA^2}$$

$$A = 2 \frac{CUENTAS}{MUESTRA^2}$$

De igual forma para calcular la velocidad

$$V = 00\ 00\ 34\ 6E \ \ hex \frac{CUENTAS}{MUESTRA}$$

Ya que:

$$V = \left(4000 \frac{CUENTAS}{REVOLUCION}\right) x \left(256x10^{-6} \frac{SEGUNDOS}{MUESTRA}\right) x \left(0.2 \frac{REVOLUCIONES}{SEGUNDO}\right)$$

$$V = 0.2048 \frac{CUENTAS}{MUESTRA}$$

$$V = \left(0.2048 \frac{CUENTAS}{MUESTRA}\right) x (65,536) = 13,421.77 \frac{CUENTAS}{MUESTRA} \quad Velocidad\ Escalada$$

$$V = 13,422 \frac{CUENTAS}{MUESTRA} \quad Velocidad\ redondeada$$

Y por último el cálculo de la posición lleva al siguiente resultado:

$$P = 00\ 00\ 1F\ 40 \ \ hex \ \ CUENTAS$$

Ya que:

$$P = \left(4000 \frac{CUENTAS}{REVOLUCION}\right) x (2\ 0\ REVOLUCIONES) = 8000 \ \ CUENTAS$$

3. Prueba del Sistema.

Prueba de motores bajo el programa de control 2.

Describimos a continuación el comportamiento de los motores para el perfil de velocidad que se muestra en la gráfica de la Fig. 7:

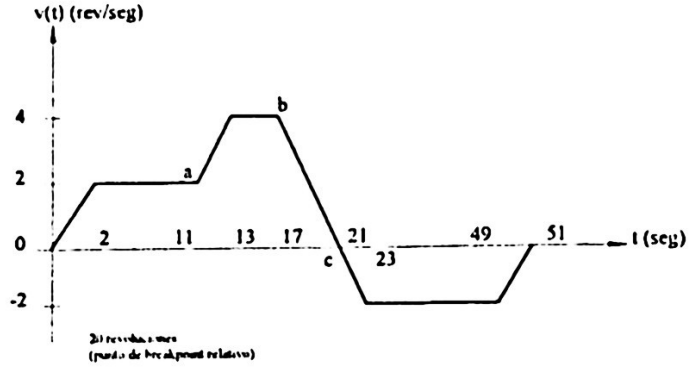


Fig. 7. (Perfil de velocidad para el control de trayectoria para ambos motores)

La flecha del motor acelera a 1rev/s^2 en una dirección hasta que alcance una velocidad constante máxima de 2 rev/sg. Después de completar 20 revoluciones en la misma dirección (incluyendo las revoluciones durante la aceleración) llega al punto ***a*** en la Fig. 7., entonces la flecha del motor se vuelve a acelerar a 1 rev/seg^2 para alcanzar una velocidad de 4 rev/seg. Después de completar 20 revoluciones en esa dirección (incluyendo las revoluciones durante la aceleración) y llegar al punto ***b*** en la Fig. 7., el motor desacelera (a 1 rev/seg^2) hasta parar. Finalmente, a partir del punto ***c*** de la Fig. 7., la flecha del motor acelera a 1 rev/seg^2 para alcanzar una velocidad de 2 rev/seg, y después de completar un número de revoluciones desacelera (a 1 rev/seg^2) y para, pero esta vez en dirección inversa.
La programación del MPID para lograr el perfil de velocidad de la gráfica de la Fig. 7., se describe a continuación:

Se inicia programando al MPID en *modo velocidad.* Se habilita un punto de paro relativo en el punto ***a*** de la Fig. 7. Esto representa un movimiento de 20 revoluciones en una dirección con una velocidad máxima de 2 rev/seg, esto es, tardará 10 segundos en cubrirlas. Cuando esta posición es alcanzada, el MPID manda una interrupción a la PC, y el ordenador anfitrión ejecuta una secuencia de comandos que incrementan la velocidad, además deshabilita la bandera de interrupción para el punto de paro, y carga un punto de paro absoluto. Es importante notar que en esta parte el MPID se programa para recibir un valor de velocidad relativo. El valor de velocidad que se carga es de 2 rev/seg, con lo cual el valor final de velocidad será de 4 rev/seg.

El punto de paro absoluto se representa como el punto ***b*** en la Fig 7. Cuando esta posición es alcanzada, el MPID interrumpe al ordenador anfitrión, y éste ejecuta un módulo de paro paulatino o desaceleración exponencial.

En el punto ***c*** de la gráfica el MPID es reprogramado, ahora en *modo de posición relativa.* Con este modo se asegura que, independientemente de la posición restante de la flecha, ésta completará treinta revoluciones en la dirección inversa, es decir en 15 seg.

En la Fig. 7., se observa que podemos tener diferentes velocidades de acuerdo a nuestro control, y también tener diferentes aceleraciones para poder alcanzar en un cierto tiempo esas velocidades. Para poder observar el cambio de velocidad y aceleración, se manejan los motores a baja velocidad y aceleración; para precisar estos cambios, nuestra velocidad fluctúa de 1 a 15 rev/seg en los valores de velocidad y de 1 a 15 rev/seg2 en los valores de aceleración.

4. Conclusiones

En este artículo describo en forma resumida el diseño y construcción de un sistema de control de velocidad aplicado a 2 motores de C. D (corriente directa). El proceso del control esta formado por:

a) Un procesador anfitrión: Una PC, que maneja un sistema de control re-entrante para dos motores de C. D., independientes.,
b) Sistema de control que usa al dispositivo LM629, a través del cual se realizó el ajuste de la velocidad en cada motor hasta estabilizarlos,
c) Un puente "H" formado por el dispositivo LMD18201 a través del cual se controló la corriente aplicada al motor seleccionado.

En este trabajo se presentó:
1) Descripción del sistema desde dos puntos de vista: de hardware y del firmware,
2) La descripción de ambas estructuras a través de módulos.
3) La prueba de control de velocidad para dos motores por medio de la implantación de una ley de control ya dispuesta en el MPID,

Los resultados muestran que el usar la tecnología re-entrante para controlar **n** máquinas con características similares es factible con la combinación PC – MPID, considerando dinámicas acotadas dentro de vecindades de control.

Referencias

[1] **Jhon Webb & Kevin Greshock.** Industrial Control Electronic Ed. Merrill 1997.

[2] **Curtis D. Johnson.** Process Control Instrumentation Technology. Ed. John Wilex & Sons. 1995

[3] **W.F. Stoccker & P.A. Stoccker.** Microcomputer Control of Thermal and Mechanical System. University of Illinois at Urbana-Champaign. Hewlett-Packard Company. Ed. Van Nostrand Reinhold New York. 1989.

[4] **M.Gopal.** Digital Control Enginnering. Electrical Engineering, Department Indian Institute of Technology Delhi India. Ed. Jhon Wilex and Sons.1992

[5] **Kenneth** Microcontroler Ed. Mc Graw-Hill International Editions 1996.

[6] **National Semiconductor.** Aplications Hand Book, AN-706. 1995 y 1993.

[7] **National Semiconductor.** LM628/LM629 Precision Motion Controler February 1995

[8] **Robert L. Hoekstra.** Robotics and Automated System. Ed. South-Western Publishing Co.1997

[9] **National Semiconductor.** LM628/LM629 Programming Guide, (nota 693).1998

[10] **National Semiconductor.** LMD18201 April 1998

[11] **Denis o´Kelly.** Performance and control of Electrical Machines. Ed Mc Graw Hill 1992

[12] **Philip T. Krein.** Elements of Power Electronics Ed. Oxford University Press. 1998.

[13] **Milos D. Ercegovac & Tomás Lang.** Digital Sistem and Hardware/Firmware Algorithms

[14] **Paul M. Embree &Bruce Kimble.** C Language Algoritms for Digital Signal Processing. Ed. Prentice Hall. 1991.

[15] **Texas Instrument.** TTL Logic Data Book. 1988.

[16] **CRC PRESS & IEEE PRESS.** The Control handBook. Ed. Williams Levine. 1996.

Visual servo control for an underactuated system

A. Espejel-Rivera and L. E. Ramos-Velasco

Centro de Investigación en Tecnologías de Información y Sistemas
Universidad Autónoma del Estado de Hidalgo
Carr. Pachuca-Tulancingo, Km. 4.5 C.P. 42084, Pachuca, Hgo.
MEXICO
aespejel@hotmail.com , lramos@uach.reduach.mx

Abstract: This paper discusses the visual servo control of an underactuated mechanisms under fixed-camera configuration. The control goal is to stabilize the system over a desired static target by using a vision system equipped with a fixed camera to observer the system and target. We present a control scheme based on the combination of a state observer and the visual feedback is applied to an underactuated system, the so-called *Pendubot*, consisting in a double pendulum actuated only at the first joint. The paper ends with the presentation of several simulation results and some guidelines future work are drawn in the conclusion.1

1 Introduction

This paper presents the application of modern linear systems theory with visual sensor to control of underactuacted mechanical system. In the eighties, the control of robot manipulators was extensively studied. Several control strategies based on passivity, Lyapunov theory, feedback linearization, output regulation, etc. have been developed for the fully actuated case, i.e. systems with the same number of actuators as degree of freedom [6, 10, 19]. The techniques developed for fully actuated robots do not apply directly to the case of underactuated mechanical systems [3, 7, 12{21]. Underactuated mechanical systems or vehicles are systems with fewer independent control actuators than degrees of freedom to be controlled. The use of visual sensors in feedback control loops with robot manipulators represents an attractive solution to position and motion control [1, 2, 4, 5]. Most existing generalizations of classical visual servoing techniques exploit a high gain or computed torque feedback to make a

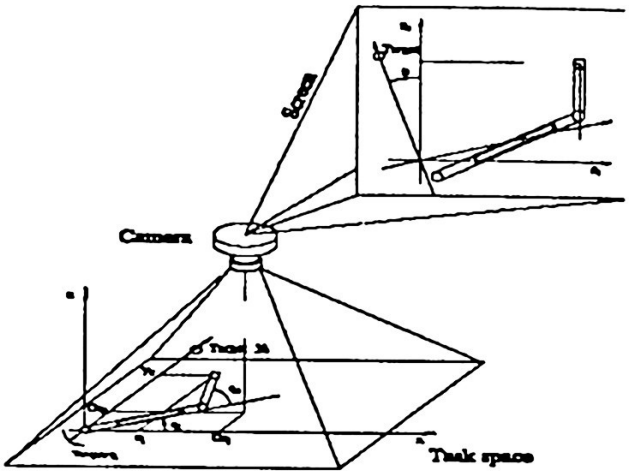


Fig. 1. Schematic representation of the Robot-camera system

dynamic reduction of the system to a controllable kinematic model for which the visual servoing task may be solved directly [5]. The dynamics model of a system is commonly ignored in the design of visual servo systems and closed-loop performance may be severely limited to ensure that the dynamic reduction is valid. Recently in [11] has explored a more nonlinear aspect of the system dynamics, and presented an asymptotically stable method for position regulation for fixed-camera visual servoing. The dificulties associated with controlling an underactuated system have received even less attention. In [22] has been working on the visual servoing problem using a Lagrangian representation of the system dynamics and consider underactuated and nonholonomic systems. In [8] proposed a new imagebased control strategy for visual servoing which is applicable to a class of underactuated dynamic systems.

In this paper we develop a dynamic controller using visual sensor for an underactuated dynamic system which operate with accurate target information as shown in Figure 1. The proposed approach is motivated by a theoretical analysis of the dynamic equation of motion of a rigid body and exploits structural linear properties of these dynamics to derive a linear control algorithm.
The paper is organized in the following manner. Section 2 describes the robot manipulator system model while section 3 is devoted to the design of the controller structure, whose performance is shown in Section 4 by presenting some simulation results. Finally, concluding remarks are given in Section 5.

2. The Pendubot Model

The Pendubot, which is the underactuated system considered here, it is shown schematically in Figure 2. For the p of this work, we assume that it has a planar motion without friction.

Table 1. PARAMETERS OF THE PENDUBOT

	Notation	Value	unit
Mass of link 1	m_1	0.5289	kg
Mass of link 2	m_2	0.3316	kg
Length of link 1	l_1	0.26987	m
Length of link 2	l_2	0.38117	m
Distance to the center of mass of link 1	l_{c_1}	0.13191	m
Distance to the center of mass of link 2	l_{c_2}	0.19208	m
Moment of inertia of link 1 about its centroid	I_1	0.013863	Kgm^2
Moment of inertia of link 2 about its centroid	I_2	0.016719	Kgm^2
Acceleration due to gravity	g	9.81	m/sec^2
Angle that link 1 makes with the horizontal	q_1		rad
Angle that link 2 makes with the link 1	q_2		rad
Torque applied on link 1	τ_1		$Nt \cdot m$

The equation of motion for the PENDUBOT can be described by the standard equation [19]

$$D(q)\ddot{q} + C(q,\dot{q}) + G(q) = \tau \qquad (1)$$

where q is the vector of joint variables (generalized coordinates), D(q) is the nxn inertia matrix, $C(q,\dot{q})$ is the vector of Coriolis and centripetal torques, G(q) are the gravitational terms, and τ is the vector of input torques. If only m joints are actuated, vector

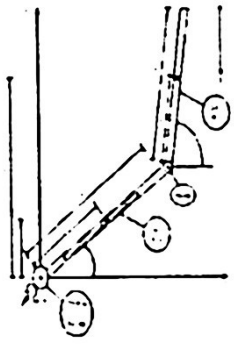


Fig. 2. The PENDUBOT system.

q can be partitioned, without loss of generality as (q_1, q_2); where $q_1 \in R^m$ represents the actuated joints, and $q_1 \in R^{(n-m)}$ represents the unactuated ones. For the Pendubot system, the dynamic model (1) is particularized as

$$\begin{bmatrix} D_{11} & D_{12} \\ D_{12} & D_{22} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} + \begin{bmatrix} G_1 \\ G_2 \end{bmatrix} = \begin{bmatrix} \tau_1 \\ 0 \end{bmatrix} \qquad (2)$$

where

$$\begin{aligned}
D_{11} &\quad m_1 l_{c1}^2 + m_2 (l_1^2 + l_{c2}^2 + 2 l_1 l_{c2} \cos(q_2)) + I_1 + I_2 \\
D_{12} &\quad m_2 (l_{c2}^2 + l_1 l_{c2} \cos(q_2)) + I_2 \\
D_{22} &\quad m_2 l_{c2}^2 + I_2 \\
C_1 &\quad -2 m_2 l_1 l_{c2} \dot{q}_1 \dot{q}_2 \sin(q_2) - m_2 l_1 l_{c2} \dot{q}_2^2 \sin(q_2) \\
C_2 &\quad m_2 l_1 l_{c2} \dot{q}_1^2 \sin(q_2) \\
G_1 &\quad m_1 g l_{c1} \cos(q_1) + m_2 g l_1 \cos(q_1) + m_2 g l_{c2} \cos(q_1 + q_2) \\
G_2 &\quad m_2 g l_{c2} \cos(q_1 + q_2).
\end{aligned} \tag{3}$$

Choosing as state vector $x = (x_1 x_2 x_3 x_4)^T := (q_1 q_2 \dot{q}_1 \dot{q}_2)^T$, as input $u = \tau_1$; and q_2 as the output, the description of the system can be given in state space form as:

$$\dot{x}(t) \quad f(x) + g(x) u(t) \tag{4}$$

$$y(t) \quad h(x) \tag{5}$$

where

$$f(x) \quad \begin{bmatrix} x_3 \\ x_4 \\ \frac{D_{22}}{D_{11} D_{22} - D_{12}^2} \left\{ \frac{D_{12} C_2}{D_{22}} + \frac{D_{12} G_2}{D_{22}} - C_1 - G_1 \right\} \\ \frac{D_{12}}{D_{11} D_{22} - D_{12}^2} \left\{ \frac{D_{12} C_2}{D_{22}} + \frac{D_{12} G_2}{D_{22}} - C_1 - G_1 \right\} - \frac{C_2}{D_{22}} - \frac{G_2}{D_{22}} \end{bmatrix},$$

$$g(x) \quad \begin{bmatrix} 0 \\ 0 \\ \frac{D_{22}}{D_{11} D_{22} - D_{12}^2} \\ \frac{D_{12}}{D_{11} D_{22} - D_{12}^2} \end{bmatrix}, \quad h(x) = x_2. \tag{6}$$

The control objective is to stabilize the system around its unstable equilibrium point $x^* = (x_1^* x_2^* x_3^* x_4^*)^T = \left(\frac{\pi}{2}, 0, 0, 0\right)^T$ i.e. to bring the second pendulum to its upper position and the first angle q_1 to zero simultaneously.

2.1 Controllability and Observability of the Linearized Model

When the Pendubot is in a neighborhood of its top unstable equilibrium position, a linear controller can stabilize the pendulum quite adequately. In order to implement a visual servo linear controller, the general non-linear differential equations (4)-(5) are linearized about the top equilibrium position. Provided that the linearized system is observable and controllable, we can design a dynamic linear control. Let us therefore

compute the rank of the controllability and observability matrices. Linearizing the non-linear equations (4)-(5) about the top unstable equilibrium point, we have

$$\delta\dot{x}(t) = A\delta x(t) + B\delta u(t) \tag{7}$$
$$\delta y(t) = C\delta x(t)$$

Where

$$A = \frac{\partial f(x)}{\partial x}\Big|_{x=x^*} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 32.2199 & -10.01720 & 0 & 0 \\ -20.8052 & 37.65097 & 0 & 0 \end{bmatrix},$$

$$B = g(x^*) = \begin{bmatrix} 0 \\ 0 \\ 26.65017 \\ -12.53857 \end{bmatrix},$$

$$C = \frac{\partial h}{\partial x}\Big|_{x=x^*} = [0 \; 1 \; 0 \; 0]$$

This model will be used to compute the control law necessary for achieving the tracking of a reference signal. Then, we have

$$det[B \mid AB \mid A^2B \mid A^3B] = -888.3171 \tag{8}$$

$$det[C \mid CA \mid CA^2 \mid CA^3]^T = -8.1198 \times 10^7 \tag{9}$$

The linearized system is controllable and observable. Therefore, a complete order observer and a full state feedback $u = -Kx$ with an appropriate gain vector K are able to successfully stabilize the system in a neighborhood of its unstable equilibrium point.

2.2 Discrete-Time State Space Equation

To obtain a discrete-time state-space equation from a continuoustime state-space equation (7)-(8), we use the following MATLAB command [Φ, Γ] = c2d(A,B,T) where T is the sampling period and it is equal to 0:005 sec. The state equation is

$$x(k+1) = \Phi x(k) + \Gamma u(k) \tag{10}$$
$$y(k) = Cx(k) \tag{11}$$

Where

$$\Phi = \begin{bmatrix} 1 & -0.0001252 & 0.005 & 0 \\ -0.00037326 & 1 & 0 & 0.0005001 \\ 0.1613 & -0.0501 & 1 & -0.0001252 \\ -0.1191 & 0.1883 & -0.00037326 & 1 \end{bmatrix},$$

$$\Gamma = \begin{bmatrix} 0.00033332 \\ -0.00053318 \\ 0.1333 \\ -0.2127 \end{bmatrix},$$

$$C = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix}.$$

Here, $x(k)$ is the state vector (n-vector) at kth sampling instant, $u(k)$ control signal (scalar) at kth sampling and $y(k)$ is the output at kth sampling.

3 Control Scheme

In this section it is shown how a visual servoing control may be derived based on estimation techniques for an fix camera with underactuated rigid body dynamics. Visual servoing systems incorporate the visual sensors in the feedback. Figure 3 depicts a block diagram of the closed-loop control system, this is a block diagram of one degree of freedom (1-DOF). The camera lens is modelled as a simple gain, *Klens*, which, due to perspective, is a function of target distance. The system is multi-rate due to the different sample intervals for the Pendubot's servo system and the camera's field rate. Pendubot structural dynamic effects are reprenented by $H(s)$, which is knowed. We shall first discuss the full-order state observer and then the state feedback controller.

It is important to note that, in the present analysis, state $x(k)$ is not available for direct measurement. Since the output $y(k)=Cx(k)$ can be measured by the fix camera. We have already checked that the system is completely observable and controllable. Therefore, we can design a state observer

$$\hat{x}(k+1) = \Phi\hat{x}(k) + \Gamma u(k) + K_e[y(k) - \hat{y}(k)] \qquad (12)$$

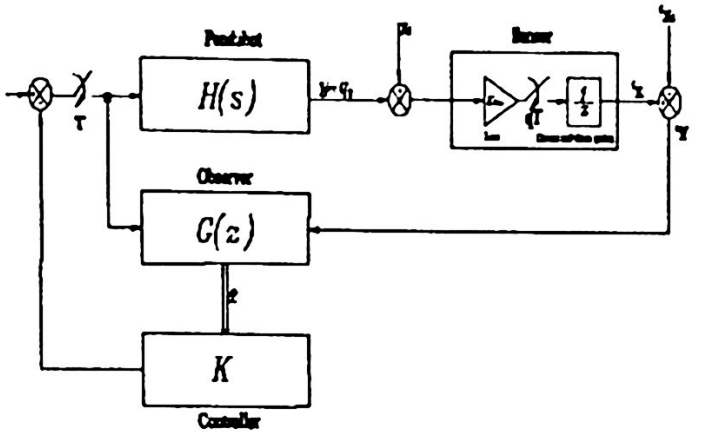


Fig. 3. Structure of visual servo control for the Pendubot. Here, x_t is the world coordinate location of the target, ${}^{i}x_d$ is the desired location of the target on the image plane, and ${}^{i}\tilde{x}={}^{i}x-{}^{i}x_d$ is the image plane error.

and the observed state $\hat{x}(k)$ is used to form the vector control $u(k)$, or

$$u(k) \quad -K\hat{x}(k) \tag{13}$$

where K_e and K are the observer feedback gain matrix and the state feedback gain matrix, respectively. The problem of stability analysis with multi-rate can be handled by the modified z-transform.

4 Simulation Result

In all the simulations, we consider that the initial condition of the system is near the equilibrium point x^* and the gain K that stabilizes

$$K \quad [-22.1131 \; -21.2982 \; -6.2282 \; -1.1932], \tag{14}$$

the linear system (10) was obtained by solving a LQR problem

$$K_e \quad \begin{bmatrix} -2.9168 \\ 0.3185 \\ -18.6506 \\ 7.7213 \end{bmatrix} \tag{15}$$

and the lens gain

$$K_{lens} \quad 1000. \tag{16}$$

We have used Simulink™ and MATLAB™ to simulate the full dynamic motion of the Pendubot. Figure 4 shows the trajectory of the target in the image plane with ${}^{i}x_{d} = 0$, for convenience.

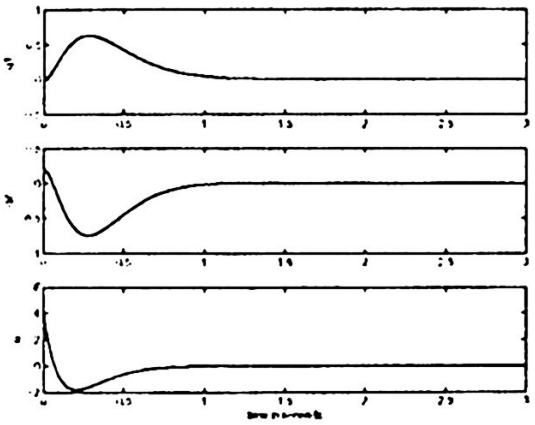


Fig. 4. Simulation results. Positioning with respect to the target Xt = (0; 90_).

5 Conclusions

The control of the Pendubot is specially dificult since it is an underactuated mechanism (two degrees of freedom and only one input). In this work, we have presented a linear position tracking controller for a fixed camera vision-based, Pendubot system. Specifically, by assuming exact knowledge of the mechanical parameters, and by considering an accepted camera model together with the robot linear dynamics, we have proposed an image-based visual servoing scheme derived from based on the combination of an estimator and the visual feedback, the design shows that this controller provides a good performance when balancing the links about the unstable vertical position. Numerical simulations assuming a discrete-time implementation of the visual controller showed the performance of the closed loop system. Preliminary results indicate that visual servoing is potentially attractive alternative for underactuated systems. We are currently implementing the algorithm on a Texas Instruments TMS320C6711 digital signal processor based system.

References

[1].Andersen, N. A., Ravn, O. & Sorensen, A. T.; *Real-time vision based control of servomechanical systems*; 2nd International Symposium Experimental Robotics, June 1991.

[2].Andersson, R. L.; Alow-latency 60 hz stero vision system for real-time visual control ; 5th International Symposium on Intelligent Control, 1990.

[3].Chung, C. C. & Hauser, J.; Nonlinear control of a swinging pendulum; Automatica, Vol. 31, No. 6, pp. 851-862, 1995.

[4].Corke, P. I., Hager, G. D. & Hutchinson, S.; Visual control of robots: highperformance servoing; John Wiley, 1996.

[5].Corke, P. I., Hager, G. D. & Hutchinson, S.; A Tutorial on Visual Servo Control ; IEEE Transactions on Robotics and Automation, Vol 12, No. 5, pp. 651-670, October 1996.
[6].Craig, J. J.; Introduction to Robotics: mechanics and control ; Addison-Wesley, U. S. A. 1989.
[7].Fantoni, I. & Lozano R.; Non-linear Control for Underactuated Mechanical System; Springer Verlang, London 2002.
[8].Hamel T. & Mahony R.; Visual servoing of a class of under-actuated dynamic rigidbody system; 39th IEEE Conference on Decision and Control; Sydney, Australia, Diciembre 2000.
[9].Hill, J. & Park, W. T.; Real time control of a robot with a mobile camara; 9th International Symposium on Industrial Robots, U. S. A., March 1979.
[10].Isidori, A. Nonlinear Control Systems, 3rd ed., New York: Springer-Verlag, 1995.
[11].Kelly, R.; Robust Asymptotically Stable Visual Servoing of Planar Robots; IEEE Transactions on Robotics and Automation, Vol. 12, No. 5, pp. 759-766, October 1996.
[12].Lozano, R. & Fantoni, I.; Stabilization of the inverted pendulum around its homoclinic orbit; System and Control Letters, Vol. 40, No. 3, pp. 197-204, 2002.
[13].Ramos, L. E., Castillo-Toledo, B. & Negrete, S.; Nonlinear regulation of a seesawinverted pendulum; Proceedings of the 1998 IEEE ICCA, Trieste, Italy 1-4, September 1998.
[14].Ramos, L. E., Castillo-Toledo, B. & Ivarez, J.; Nonlinear regulation of an underactueted system; Proceeding of the 1997 IEEE ICRA, pp. 3288-3293, Albuquerque, New Mexico, April, 1997.
[15].Ramos-Velasco, L. E., Ruz-Len, J. J. & Celikovsk, S.; Rotary Invertd Pendulum: Trayectory tracking via nonlinear control techniques; Kybernetika, Vol. 38, No.2, pp. 217-232, 2002.
[16].Ramos-Velasco, L. E., Celikovsk, S. & Kucera, V.; Nonlinear regulation for a laboratory helicopter model using error feedback; Asian Journal Control, Sometido, 2004.
[17].Reyhanoglu, M., van der Schaft, A. J., McClamroch, N. H. & Kolmanovsky, I.; Dynamics and control of a class of underactuated mechanical systems; IEEE Transactions on Automatic Control, pp. 1663-1671, 1999.
[18].Shim, H., Koo, T., Ho_mann, F. & Sastry, S.; A comprehensive study of control design for an autonomous helicopter; In Proceedings of the 37th IEEE Conference on Decision and control CDC'98, 1998.
[19].Spong, M. W, Vidyasagar, M., Robot Dynamics and Control, John Wiley and Sons, Inc., New York, 1989.
[20].Spong, M. W.; The swing up control of the acrobot; In IEEE International Conference on Robotics and Automation, San Diego, CA, 1994.
[21] Spong, M. W. & Block, D. J.; The pendubot: A mechatronic system for control research and education; In Poceedings of the 34th IEEE Conference on Decision and Control, 1995.
[22] Hang, H. & Ostrowski J. P.;Visual Servoing with Dynamics: Control of an Unmanned Blimp; IEEE International Conference on Robotics & Automation, Detroit, Michgan, May 1999."

Estación de desarrollo de control difuso (EDCD)

Haydée Patricia. Martínez Hernández[1], Germán A. Muñoz Hernández[2],
José Federico Ramírez Cruz[3].

[1,3]·Instituto Tecnológico de Apizaco.
Av. I. Tecnológico de Apizaco s/n, Apizaco, Tlaxcala, Tel 241-4172010
[2]· Instituto Tecnológico De Puebla.
Av.Tecnológico S/N, Puebla, Puebla , Tel. (22) 29-88-24
[1]·hpmartinez@itapizaco.edu.mx, [3]·framirez@itapizaco.edu.mx

Resumen: En este trabajo se describe el diseño y la implementación de una Estación de Desarrollo de Control con Lógica Difusa mediante el Microcontrolador MC68HC912B32. La interfaz de usuario reside en una computadora PC con un ambiente visual amigable al usuario. El software proporciona la capacidad de variar los parámetros de control y el punto de ajuste, esto es, permite cambiar la base de conocimiento; es decir, las reglas lingüísticas del control y las funciones de pertenencia difusas, las cuales ajustan y sintonizan el sistema. Además, realiza la comunicación vía serial hacia el microcontrolador, el cual ejecuta las acciones necesarias para mantener el sistema controlado. El sistema se probó diseñando un controlador PD difuso para controlar la velocidad de un Motor de C.D.

Keywords: Sistemas de Control Difuso, Estación de desarrollo, Prototipos didácticos, Microcontradores,

1 Introduction

En años recientes los sistemas de control han asumido un papel cada vez más importante en el desarrollo y avance de la civilización moderna y la tecnológica. Así que se han desarrollado numerosos sistemas de control, los cuales han sido clasificados de acuerdo a diferentes parámetros como a la presencia o ausencia de retroalimentación (Lazo Abierto o Lazo Cerrado), al proceso (continuo o discontinuo y manual o automático), a la naturaleza de los componentes de control (neumáticos y eléctricos o electrónicos y digitales), en esta última clasificación de controladores electrónicos y digitales, se vuelven a reclasificar y encontramos al Control Digital Directo (DDC), Control supervisor, Control distribuido, Control avanzado, Control con sistemas expertos, Control por Redes Neuronales y Control por Lógica difusa [1].

Haciendo referencia a la lógica difusa, en 1965, el Dr Lofti A. Zadeh, profesor de Ingeniería Eléctrica y Ciencias Computacionales de la Universidad de Berkeley,

basado en el un trabajo de Lukasiewicz; propone que es posible conectar el lenguaje matemático con la inteligencia humana, argumentando que muchos conceptos son mejor definidos por palabras que sólo por signos matemáticos, construyendo así mejores modelos de una realidad, de esta manera es como Zadeh define el concepto de conjunto difuso y crea a partir del mismo la teoría de subconjuntos difusos.[2].

Una de las razones principales del uso de la lógica difusa es que el algoritmo por naturaleza permite procesar información de forma semejante al cerebro humano; esta característica desde el punto de vista de control difuso, tiene varias ventajas:

- La primera y principal es que no requiere del modelo matemático del sistema a controlar.
- La segunda es que el diseño del control es simple, es decir, se requiere de menores herramientas matemáticas para el diseño de un controlador que por técnicas clásicas (esta característica es importante debido a que diseñar un controlador que sea robusto y adaptable puede llegar a ser muy complejo) y,
- La tercera consiste en que es posible conocer exactamente su funcionamiento interno ya que la acción que ha de realizar el sistema difuso se encuentra descrita por un conjunto de reglas de inferencia, de manera que puede determinarse el estado del sistema en cada instante.

En la mayoría de las instituciones de educación superior en el área de control electrónico en nuestro país existe una necesidad de contar con prototipos didácticos para la enseñanza de las nuevas técnicas de control inteligente; sin embargo, estas técnicas han sido poco difundidas pues prevalece la enseñanza en control clásico. Además, la enseñanza sobre estos temas es en su mayoría teórica, por tal motivo se observa la necesidad de contar con herramientas prácticas que nos permitan aprender a desarrollar sistemas de control inteligente. Existen propuestas de herramientas didácticas y de desarrollo de sistemas de control inteligente para simular estos sistemas mediante una computadora personal, pero no proporcionan la capacidad de llevar a término el control de un sistema real sensando las variables físicas y actuando con dispositivos electromecánicos. En este trabajo presentamos una implementación de una Estación de Desarrollo con Lógica Difusa, la cual tiene una interfaz de usuario mediante una computadora personal misma que permite interactuar de manera amigable con el diseñador de sistemas de control difuso. La computadora personal se comunica con un microcontrolador MC68HC912B32 a través del puerto serie, y actúa como controlador del sistema. La elección de este microcontrolador se debe a que posee internamente las instrucciones de control difuso como el proceso de fusificación, evaluación de reglas y defusificación. Para probar el sistema EDCD se realizó un control PD difuso para controlar la velocidad de un motor de c.d. utilizado como planta piloto, con el cuál pueden hacerse diversas pruebas en tiempo real, fijando un punto de consigna. Se diseñó un control difuso que tiene como entradas el error en la velocidad del motor y el cambio del error, y como salida la variación de corriente que llega al motor lo cual se ve reflejado en una variación de su velocidad. Los parámetros de control fueron establecidos a un conjunto de 5 funciones de pertenencia difusas del tipo trapezoidal por cada variable de entrada, 5 funciones de pertenencia difusas del tipo singlenton para la variable de salida y 25 reglas de control.

2. Descripción general de un Controlador Difuso.

Un sistema de control es un arreglo de componentes diseñado para alterar, regular y manipular un sistema físico a través de la acción del control, y los podemos encontrar de dos tipos: *sistema de control de lazo abierto*; en el cual la acción de control es independiente de la salida del sistema físico, y los *sistemas de control de lazo cerrado* en el cual la acción de control depende de la salida del sistema físico. Este último será mostrado en la Fig. 1.

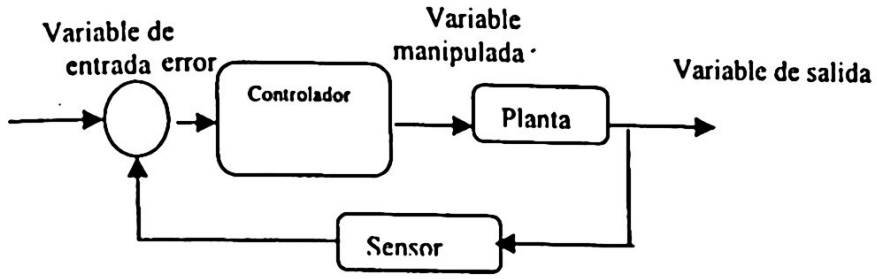


Fig. 1 Sistema de control de lazo cerrado

El problema del diseño de un sistema de control PID clásico, es cómo obtener la ganancia proporcional (k_p,) el tiempo integral (T_i)y el tiempo derivativo (T_d) para poder desarrollar la siguiente ecuación:

$$u(t) = K_p e(t) + \frac{K_p}{T_i}\int_0^t e(t)dt + K_p T_d \frac{de(t)}{dt}$$

El paradigma de los sistemas difusos es una disciplina desarrollada para controlar sistemas no lineales. Estas describen los algoritmos de procesos de control como un conjunto de relaciones difusas proporcionadas por un experto humano entre la información acerca del funcionamiento de los procesos a ser controlados, y las acciones a efectuar para llevar a cabo el control.

Una característica principal de los sistemas de control difuso es la existencia de dos niveles distintos de conocimiento tales como: 1) Reglas cualitativas o expresiones condicionales de la forma: *Si* "antecedente" *entonces* "consecuente"; y 2) Términos difusos o imprecisos de las variables de entrada y salida tales como muy pequeña, pequeña, mediana, grande, muy grande. Las reglas difusas son expresadas en términos lingüísticos de los conjuntos difusos correspondientes a los valores de entrada y salida pueden escribirse de la siguiente manera:

Si **a** es pequeña y **b** es grande, entonces **c** es mediana.

Si **a** es grande y **b** es mediana, entonces **c** es grande.

donde **a** y **b** son las entradas y **c** es la salida,. Los controladores difusos son construidos por estas reglas de control, y cuando una entrada es dada, la salida es calculada por medio de la *inferencia difusa.*

Un controlador lógico difuso es un sistema que nos provee de un algoritmo de control automático basado en una estrategia de control lingüístico implementado con ayuda de reglas de inferencia y funciones de pertenencia. Para realizar un sistema de control

difuso deben considerarse cuatro elementos principales tales como: 1) Un módulo de *fuzificación*, el cual mide el valor de las variables de entrada, realiza una normalización que mapea el rango de los valores de las variables de entrada en una escala de valores preestablecidos, y convierte la información obtenida a valores de pertenencia. 2) La *Base de conocimientos*, la cual consiste en un conjunto de datos y un conjunto de reglas que son el conocimiento del control en la aplicación; el conjunto de datos contiene las definiciones de los términos que serán usados en las variables lingüísticas o de pertenencia, y el conjunto de reglas difusas contiene la caracterización de las metas y políticas de control manejados por un humano experto. 3) *Lógica de toma de decisiones o inferencia difusa,* esta es la parte medular del control y tienen la capacidad de simular la toma de decisiones del razonamiento humano, ayudándose de los términos difusos, de álgebra de conjuntos difusos y de algunos métodos matemáticos. 4) El módulo de *Defuzificación*, la cual convierte los valores difusos emitidos por la inferencia difusa en información precisa (numérica), realiza una equivalencia que mapea los valores proporcionados por el control en un rango preestablecido a valores apropiados para las variables de salida. La figura 2 muestra el diagrama a bloques del control lógico difuso simple.

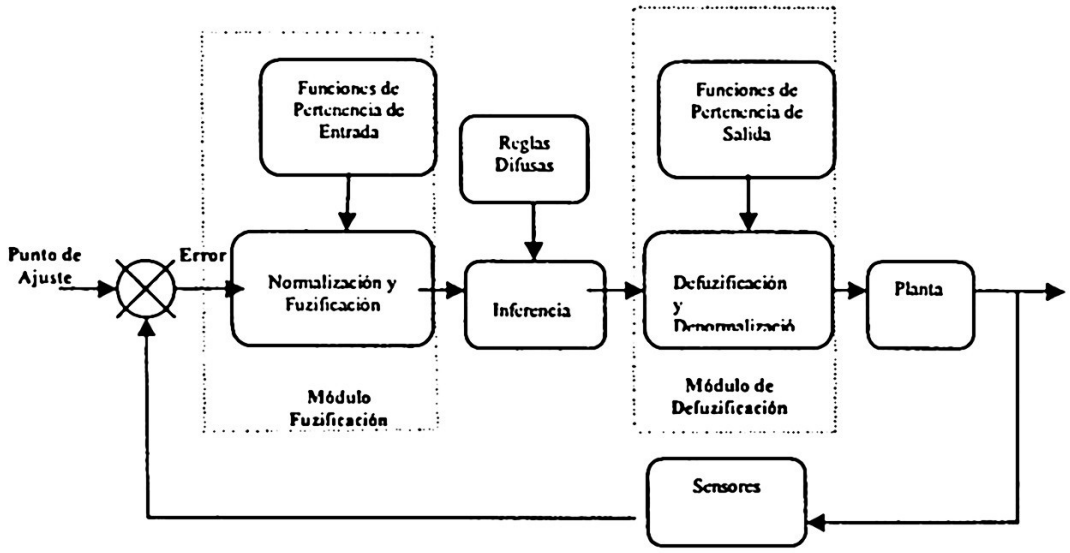


Fig. 2 Diagrama de bloques de control lógico difuso simple.

3. Implantación del Sistema

La EDCD es un sistema basado en el microcontrolador M68HC912B32, con una interfaz contenida en una computadora personal. La estación tiene la función de controlar un sistema mediante un controlador difuso. El microcontrolador monitorea las señales eléctricas recibidas del sistema a controlar y ejecuta las acciones de control necesarias en tiempo real. El sistema proporciona la capacidad para que el usuario pueda diseñar su control lógico difuso desde la PC modificando los parámetros de control. La computadora interactúa con el microcontrolador enviándole la información de las actualizaciones de dichos parámetros. El microcontrolador actúa también como

una terminal de monitoreo que manda la información del estado de la variable a controlar a la PC cuando se lo solicita.

Para llevar a cabo la implantación del sistema nosotros utilizamos la tarjeta de evaluación MC68HC912B32 de motorola, la cual posee las siguientes características: una memoria flash EEPROM de 32 k, una memoria EEPROM de 768 bytes, una memoria RAM de 1k, 1 convertidor analógico digital de 8 canales, 1 temporizador y acumulador de pulsos de 8 canales, 1 modulador-demodulador de pulso de 4 canales, una interfaz de comunicación serial asíncrona, 1 interfaz de periférico serial síncrono y un bus multiplexado de 16 bits. La figura 3 muestra el esquema general de la EDCD.

Para probar el sistema EDCD se realizó la implementación de un sistema de control difuso para regular la velocidad de un motor de c.d. mediante un modulador de ancho de pulso (PWM). Para variar la velocidad del motor de c.d. se utilizó la relación existente entre la tensión aplicada al motor y la velocidad de giro; a este motor se le acopló un codificador óptico incremental (encoder) para detectar el número de revoluciones por minuto cerrando el lazo de control. En la figura 4 se muestra el diagrama a bloques del control de velocidad del Motor de CD.

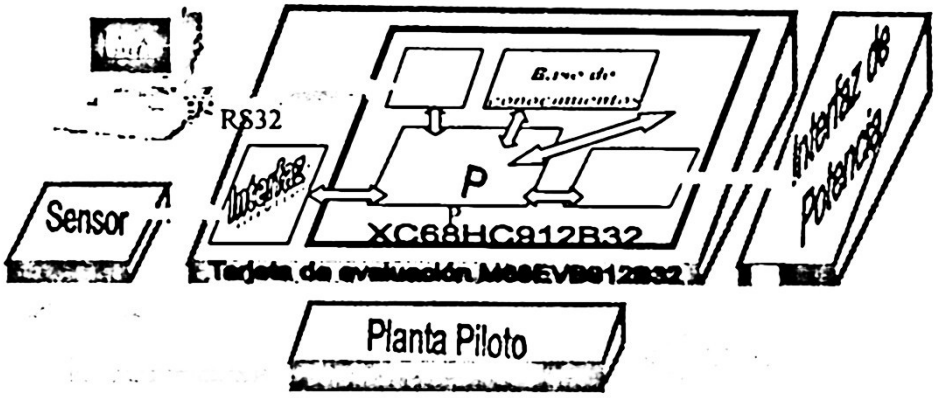


Fig. 3 Esquema de la Estación de Desarrollo de Control Difuso.

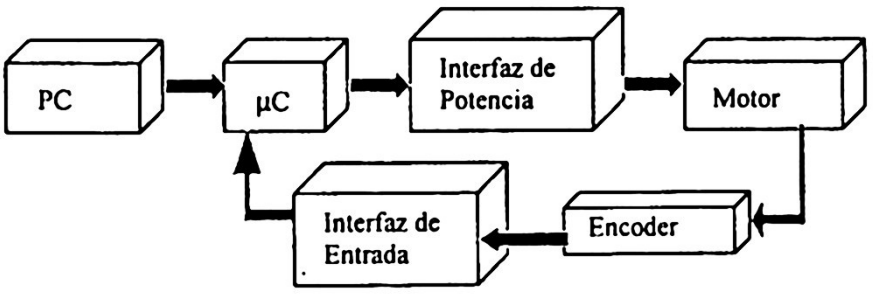


Fig. 4 Diagrama a bloques del control de velocidad de un motor de c.d.

3.1 Interfaz de salida

Para el variar el voltaje suministrado al motor se utilizó la técnica de control por interrupción de voltaje, la cual se basa en la conmutación del voltaje promedio suministrado por una fuente fija de c.d. [10]. La conmutación es un método Encendido-Apagado, también denominado *conmutación de corriente* [11]. Existen tres técnicas comúnmente usadas para variar la proporción de la conmutación en el tiempo de encendido (t_{ON}) y la conmutación en tiempo de apagado (t_{OFF}) de la forma de onda de salida "interrumpida" durante el tiempo total de trabajo (Duty Cycle), estas son: 1 modulación de ancho de pulso; 2) modulación por velocidad de pulso, y 3) combinación de ancho de pulso y velocidad de pulso. En este trabajo utilizamos la primera (Fig. 5). *El Porcentaje Duty Cycle* es definido como la proporción de la longitud de la forma de onda en el tiempo de encendido del periodo contra el periodo total de un ciclo de la señal (tiempo de encendido más el tiempo de apagado) [12], como se puede observar en la siguiente fórmula:

$$\text{Duty Cycle (\%)} = \frac{t_{ON}}{t_{ON} + t_{OFF}} x100$$

El porcentaje total de trabajo corresponde al voltaje de carga promedio que circula sobre el motor y que se visualiza en la disminución o aumento de flujo de corriente sobre el dispositivo.

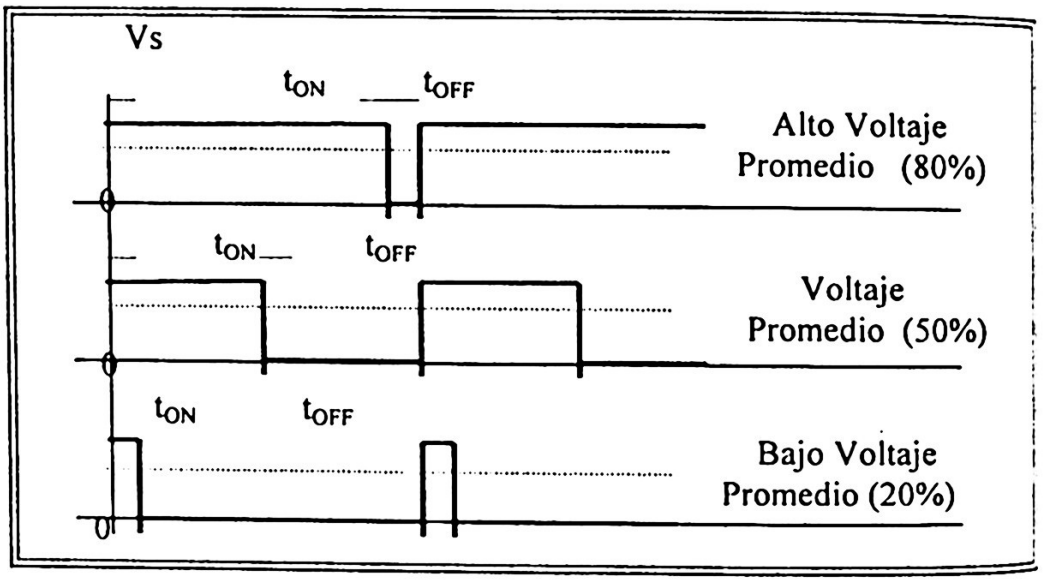


Fig. 5 Formas de onda por modulación de ancho de pulso.

La figura 6 muestra la interfaz de potencia que se utilizó en este trabajo la cual se encarga de proporcionar el voltaje y la corriente de c.d. necesaria para hacer funcionar el motor. Esta interfaz consta de un transistor utilizado como un interruptor en circuitos electrónicos digitales, que permite el paso de la corriente de c.d al motor cuando se tiene un pulso de salida en la compuerta "and" lógica e inhibe el paso de la corriente cuando se no existe señal a la salida de dicha compuerta[11].

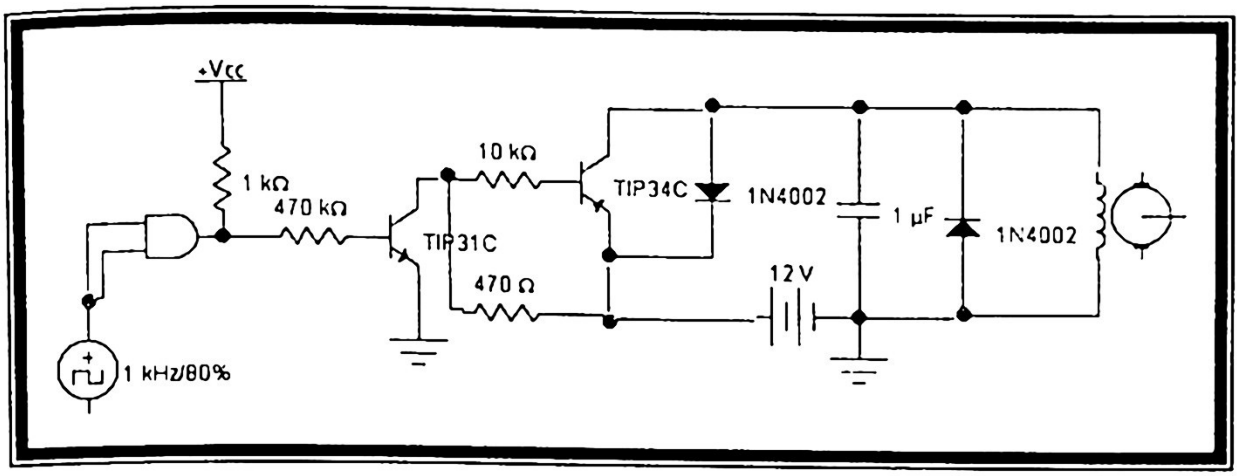


Fig. 6 Diagrama de la interfaz de potencia para el motor de CD

3.2 Interfaz de entrada

Para obtener la medición de la velocidad del motor se utilizó un codificador óptico incremental que consta de un disco metálico con una serie de perforaciones colocadas radialmente y equidistantes entre sí. La funcionalidad de este dispositivo consiste en detectar un tren de pulsos proporcionados por un sistema de transmisión óptica formado por un fototransmisor y un fotorreceptor como se observa en la figura 7.

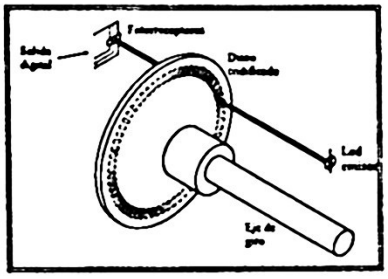


Fig. 7 Diagrama de la disposición de un codificador óptico (encoder) incremental.

La interfaz de entrada al microcontrolador por el puerto A consta de dos contadores binarios de ocho bits y dos buffers para manipular 16 bits multiplexados en 8 bits como se muestra en la figura 8. Estos dispositivos son controlados por el microcontrolador a través del puerto B.

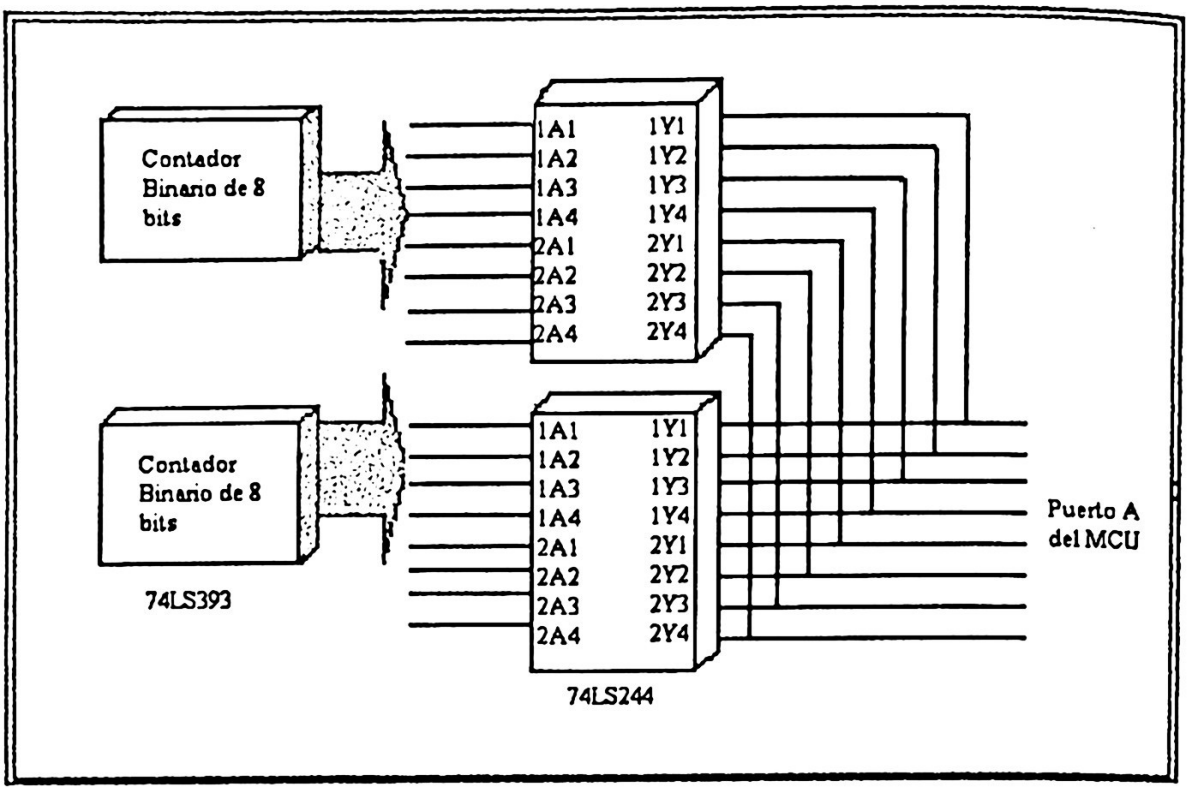


Fig. 8 Contador binario que entrega al Puerto A del MCU.

La secuencia del pseudo código del control se basa en la interrelación de todos los elementos que se han mencionado en esta sección. El temporizador del microcontrolador es programado para que cada 50 ms se realice un escrutinio del estado de la variable de control y se ajusta el ancho de pulso de acuerdo a la decisión tomada por el controlador difuso. Las actividades realizadas para modificar el error y encontrar el punto de ajuste son: 1) el conteo en la interfaz de entrada le proporciona un dato al controlador; 2) el controlador lo recibe y lo normaliza en un rango prestablecido; 3) se calcula la velocidad del motor y se obtiene el error; 4) se calcula el cambio en error con la muestra anterior y se normalizan para ser asignados al control; 4) el controlador calcula el ancho del pulso y se denormaliza en el rango establecido por el modulador de ancho de pulso (PWM); 5) se ajusta el ancho de pulso para corregir el error.

3.3 Sistema de comunicación

El sistema de comunicación que se desarrollo para el EDCD, se basa en el intercambio de información entre la PC y el MCU [19], las funciones que se llevan a cabo son las siguientes: 1) Consulta y modificación de los datos de las variables de entrada y salida difusas, las funciones de pertenencia y el conjunto de reglas; 2) Transmisión y recepción del las modificaciones realizadas en la interfaz de usuario a la tarjeta MC68HC912B32; 3) Recepción del punto de ajuste.

El enlace físico utilizado en este proyecto fue el estándar RS-232C desde la computadora hacia la interfaz de comunicación en serie (SCI) del tipo NRZ, el cual es compatible con la norma RS-232.

El protocolo de enlace de datos corresponde a una combinación entre BYSINC y el SDLC de IBM (protocolos de comunicación entre computadoras), considerando que el protocolo está orientado a caracteres. La relación entre los elementos de comunicación está definida de acuerdo al protocolo SDLC donde se describe un elemento maestro que controla la comunicación y un elemento esclavo que responde a las indicaciones del maestro [19].

El protocolo de enlace de datos es responsable de la encapsulación de mensajes en tramas y de controlar la transferencia de las tramas sobre el enlace físico, también la detección y corrección de errores cuando es factible.

4 Resultados experimentales preliminares

Para probar el sistema EDCD, se realizó la implementación de un sistema de control difuso para regular la velocidad de un motor de c.d. como se mencionó en la sección anterior. En la figura 9 se muestran las pruebas preliminares de nuestro sistema. Para tener una mejor respuesta consideramos realizar una optimización en el sistema de control.

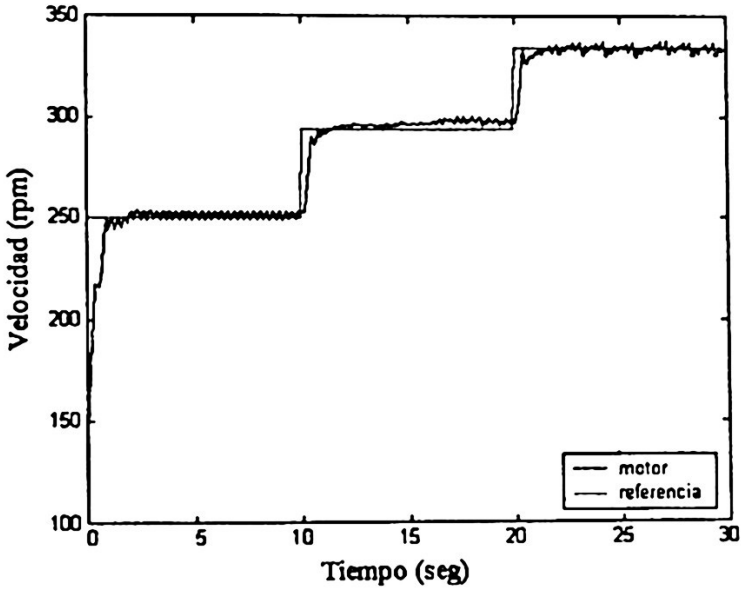


Fig. 9 Respuesta del EDCD para controlar un motor de c.d.

5 Conclusiones y perspectivas de desarrollo

Para el estudio de sistemas de control convencional, es necesario desarrollar prácticas, simularlas y posteriormente implementarlas físicamente para entender su comportamiento, así mismo, es importante continuar con esta metodología de aprendizaje ahora con el control con lógica difusa. Es por eso que la estación de desarrollo de control difuso EDCD mediante la EVBMC68HC912B32, ofrece la interacción física entre el usuario y el sistema a controlar en tiempo real.

El propósito de esta estación de desarrollo es ofrecer un control automático basado en una estrategia de control lingüístico, con algunas características sobresalientes con respecto a las técnicas de control convencional, es decir, para generar un esquema de control difuso únicamente se requiere tener conocimientos del proceso en sí y no de la matemática asociada a él, ni de la programación en bajo o alto nivel; pero sobre todo hace fácil la tárea de expresar el razonamiento natural de comprender las políticas y las metas del control difuso.

Con el uso de este sistema podemos mejorar el aprendizaje de los sistemas de control difuso, así como diseñar nuevos sistemas con solo cambiar los elementos de sensado y el tipo de planta a controlar.

Al momento nos encontramos en la experimentación de diferentes sistemas de control y la automatización de los parámetros de control mediante algoritmos de aprendizaje automático.

Referencias

[1] Antonio Creus Solé. "Instrumentación Industral." Alfaomega marcombo 6a. edición 1998.pp 487-554, 574-586.

[2] José R. Hilera; Víctor J. Martínez, "Redes Neuronales Artificiales fundamentos, modelos y aplicaciones" ADDISON-WESLEY IBEROAMERICANA Ed. ra-ma. pp 322-369.

[3] Dimitri Driankov, Hans Hellendoorn and Michel Reinfrank, "An Introduction to Fuzzy Control", Springer-Verlag Berlin Heidelberg 1993.

[4] Toshiro Terano, Kiyoji Asai, and Michio Sugeno, "Fuzzy Systems Theory and its Applications", English edition copyringht 1992, by Academic Press,

[5] Inc. Chuen Chien Lee. "Fuzzy Logic in Control Systems: Fuzzy Logic Controller". IEEE Transactions, man and cybernetics, Vol. 20 No. 2. March/April 1990.

[6] R. Haber. Julio Antonio Mella "Apuntes de Control Borroso." Instituto Superior Politécnico

[7] Bart Kosko. "Neural Networks and fuzzy Systems". Prentice Hall, 1992

[8] Dubois, D., and Prade, H., "Fuzzy Sets and Systems: Theory and Applications" Orlando, FL, Academic Press, 1980.

[9] Timothy J. Ross, " Fuzzy Logic with Engineering Applications" Mc Graw Hill 1995.

[10] Antonio Barrientos. Luis Felipe Peñin. Carlos Balaguer. Rafael Aracil. "Fundamentos de robótica", Mc Graw Hill 1997.

[11] Charles A. Schuler and William L. McNamee. "Industrial Electronics and Robotics". Mc Graw Hill Book Company Unites States of America, 1986.

[12] William Kleitz. "Digital Electronics a Practical Approach". Second Edition Prentice Hall, Inc. USA, 1990.

[13] Joseph Vitlayathil. "Power Electronics Principles and Aplications." McGraw-Hill series in Electrical and Computer Enginering. United States of America, 1995.

[14] Motorola. Fuzzy Microcontroller Development System Technical Summary 16-Bit Microcontroller.

[15] M68EVB912B32 Evaluaction Board USER'S MANUAL MOTOROLA

[16] *Luis Suárez Bernaldo VISUAL BASIC - Guía del Estudiante. Junio de 1998*

[17] www.guille.costasol.net

[18] www.ciudadfutura.com/ *Luis Suárez Bernaldo/ VISUAL BASIC*

[19] Andrew S. Tanenbaum,"Redes de computadoras", Pearson Education, Abril 1998

Diseño del subsistema de distribución de potencia para el microsatélite experimental mexicano Satex

Andrés Calvillo Téllez[1] y Esaú Vicente Vivas[2]

[1]I P N, Centro de Investigación y Desarrollo de Tecnología Digital, Av. del Parque 1310 Garita de Otay,
Tijuana, B. C., Tel. 01 (664) 6 -23-13-44, E-mail: calvillo@citedi.mx,
[2]Instituto de Ingeniería, UNAM, E-mail: evv@servidor.unam.mx

Resumen

Este artículo describe el diseño de un sistema digital que sensa, procesa y controla el recurso energético para el suministro y distribución de potencia de los sistemas que integran el satélite SATEX. Se presenta el diseño del subsistema de distribución de potencia con alta fiabilidad para alimentación los subsistemas del satélite. Cada subsistema posee una demanda de requerimientos energéticos diferente, la cámara digital, consume 1W durante 5 minutos, el transmisor de VHF consume 20W durante 10 minutos, el transmisor de UHF consume 10W durante 10 minutos, y el transmisor láser consume de 24W durante 15 minutos. El módulo de distribución de potencia diseñado mantiene un equilibrio entre el suministro de la demanda de potencia y la fiabilidad de los subsistemas. Posee habilitación independiente para cada carga útil y doble alimentación para el modulo de plataforma, con esto se incrementa la fiabilidad en los sistema más importantes, asegurando un suministro de energía a cada subsistema de hasta un 50% extra de la demanda solicitada.

1. Introducción

Para fortalecer los conocimientos científicos y tecnológicos, además de ampliar los recursos nacionales de conocimiento, algunas instituciones de educación superior e investigación se han integrado para desarrollar el primer microsatélite experimental mexicano [1]. Para proveer energía eléctrica a un microsatélite se requieren dos sistemas, uno que proporcione la energía y otro que la almacene como se ilustra en la figura 1. Para estimar la demanda energética del satélite se consideran los consumos de energía en misiones típicas durante una orbita, tomando en consideración que se realizarán experimentos cuando el satélite esté expuesto al sol y durante la fase de eclipse. El proceso implica el análisis de las características del sistema de generación y almacenamiento de energía considerando el área efectiva de captura de rayos solares así como la eficiencia del arreglo de paneles solares [2].



Figura 1. Diagrama de distribución de potencia.

2. El microsatélite satex

Las consideraciones acerca de las misiones que se realizarán, permitirán diseñar la distribución de potencia atendiendo prioridades de suministro energético y programando los experimentos con base en la capacidad de energía almacenada. Cada carga del satélite posee una tarea específica y por seguridad uno de los tópicos de diseño considera que cada carga posee una alimentación de potencia y tensión distinta. Por ejemplo, los sensores requieren una alimentación de +/- 5V mientras que el sistema de comunicación demanda 12V. Otro punto importante que se consideró es la prioridad en la fiabilidad, esto se puede ver desde el punto de vista de ofrecer una alimentación más segura a algunos subsistemas satelitales, como el caso del suministro energético para la computadora principal lo cual implica el incremento de la redundancia de alimentación. Por otro lado, el sistema de transmisión de RF no posee redundancia en alimentación sin embargo se tiene un segundo sistema de radio de respaldo.

El microsatélite experimental mexicano SATEX se ubicará en una órbita polar a 98° de inclinación y una altitud aproximada de 780 km sobre la Tierra. El periodo de recorrido de la órbita tendrá una duración de 101 minutos, con un periodo de sol y de eclipse de 67 y 34 minutos respectivamente. Sus experimentos fueron propuestos y desarrollados por diversas instituciones de todo el país [2]. El SATEX tiene forma cúbica de 50 cm por lado con una estructura de aluminio basada en un cilindro central y tres placas que forman dos pisos en donde se instalan equipos y experimentos, con una masa total de 55 Kg. Tiene cuatro paneles solares montados en sus caras laterales y será inicialmente estabilizado de forma pasiva mediante un mástil desplegable ubicado en su parte superior. Adicionalmente, cuenta con un sistema de estabilización dinámica basado en bobinas de torque magnético.

El subsistema de potencia está compuesto por los módulos de generación y almacenamiento de energía eléctrica. El dispositivo de generación se compone de cuatro tableros solares, cada uno de ellos compuesto por 84 celdas que producirán al inicio de vida un promedio de energía de 24W [3]. A su vez, el módulo de almacenamiento de energía está compuesto por 52 celdas de NiCd que ofrecen una

capacidad de almacenamiento total de 31.2V@3.4AH por medio de dos módulos independientes y conectados en paralelo. Respecto al módulo de distribución de energía, está constituido por convertidores de corriente directa que energizan a los equipos electrónicos del satélite, y que a su vez adquiere la energía de un *bus* no regulado de 31.2V, figura 2.

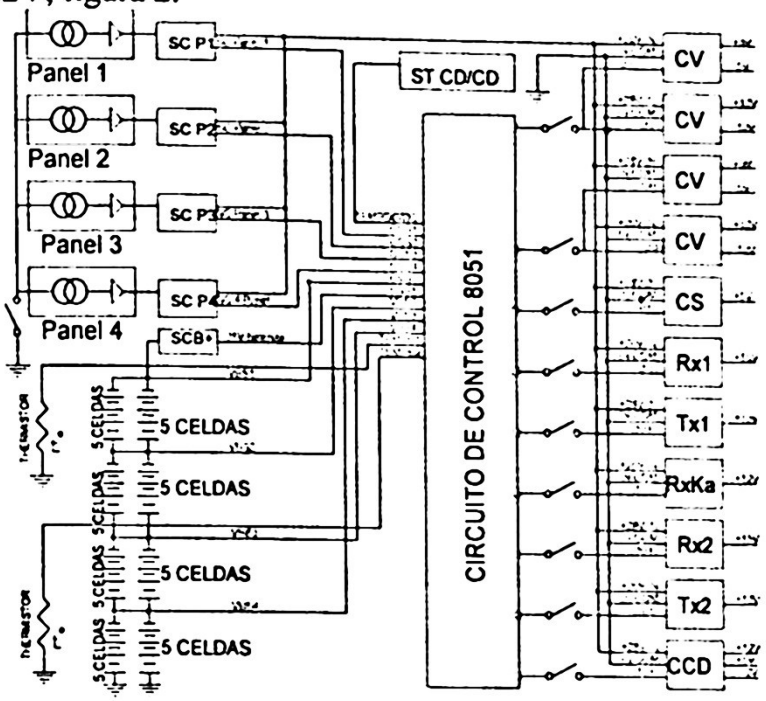


Figura 2. Subsistema de Potencia del SATEX I, con distribución de potencia para alimentación de las cargas.

3. Dimensionamiento del consumo de energía para el diseño de distribución de potencia

Para determinar que tipo de arreglo y componentes se van a utilizar en el diseño del modulo de distribución de potencia, primeramente se analizarán los requerimientos energéticos por subsistema, el consumo que presentarán y los niveles de alimentación que necesitan. Posteriormente se determinará si es prioritaria su función, si requiere módulos redundantes o si necesita alimentación redundante.

3.1. Consumo de potencia por subsistema

Se a concentrado la información de las necesidades energéticas de los subsistemas y se reportan a continuación los requerimientos de los niveles de alimentación de cada componente considerando el tiempo de operación por orbita, los niveles de tensión, corriente y potencia, todo esto se muestra en la tabla 1.

Tabla 1. Niveles de alimentación requeridos por cada subsistema del SATEX I

Subsistema	T	V	I	P
Acondicionamiento + Multiplexaje	∞	+5,+12, -12	.200, .050, .005	1.0, 0.6
Bobinas Magnéticas	10	10-12	0.3	
Carga Útil Óptica	10	+ 5, +12, -12	0.4, 0.3, 0.2	2.0, 3.6, 2.4
Cámara CCD	3	12	0.5	
Computadora Vuelo	∞	5	.250	1.25
Conmutadores de potencia	∞	5	0.100	0.5
Despliegue de Antenas	0.15	12	3	36
Magnetómetros	∞	5 +/- 0.5		0.040
Mástil.	0.1			
Receptor KA		+12	1/6	2.0
Rx UHF principal	∞	+5, 12, -12	.08, .120, .020	0.4, 1.44, 0.24
Rx UHF Secundario	∞	28	.080	2.24
Sensores de Corriente	∞	5	0.030	0.15
Tx VHF principal	10	+5, +12, -12	.08, .120, .020	0.4, 1.44, 0.24
Tx VHF secundario		+12	.08	2.24

Con el concentrado de los datos de potencia, se determina cual será la capacidad máxima en potencia que requiere cada subsistema y de esta forma se selecciona convertidor óptimo. Esto permitirá reducir espacio y peso, siempre y cuando mantenga la operación del convertidor dentro de los rangos de seguridad. El arreglo de convertidores se emplea de acuerdo con la alimentación que requieran: simple, doble, o triple voltaje.

Tabla 2. Demanda de potencia de los subsistemas que conforman a SATEX I

Subsistema	Componente	P_W	I_{mA}
Computadora de Vuelo	Computadora de vuelo	6.17	1066
Procesador de Sobrevivencia	Procesador de Sobrevivencia	0.65	130
	Sensores de sol	0.96	80
	Sensores de Corriente		
Sensores	Magnetómetros	0.04	8
	Sensores de Corriente y Temp.	1.56	130
Bobinas	6 bobinas	1.5	240
Mástil	Mástil		
Separador de Antena	DSA	0.027	2.3
Transmisores Principal y	Tx1	7.71	514
Redundante	Tx2	8.16	682
Receptores Principal y Redundante	Rx1	3.0	200
	Rx2	1.71	143
Banda Ka	Ka	9.36	780
Carga útil óptiça	CUO	21.6	900
Cámara Digital	CCD	1	200
	Batería (NiCd)	40	1800
Fuente de Poder	CD/CD		
	Microcontrolador	1.0	
	Paneles solares		

Los datos de potencia utilizada por subsistema se presentan en la tabla 2, los cuales permiten visualizar la información del tipo de alimentación que necesitan. Como

observa, se tienen múltiples voltajes, potencias y cargas. No existen niveles de tensión semejantes y se tienen cargas ruidosas y otras sensibles al ruido [4].

3.2. Presupuestos de potencia de experimentos de acuerdo con el modo operativo

De estas tablas también se pueden desprender algunos datos importantes, como conocer qué equipos se energizan de acuerdo con cada uno de los experimentos que se tiene planeado realizar en la misión. Por ejemplo, se puede realizar un presupuesto de gasto energético para determinar cual será el consumo que demandará la ejecución de cada experimento. Este presupuesto se obtiene para cada uno de los experimentos propuestos, y de esta forma se determina qué subsistemas se requiere energizar y por tanto se puede conocer también cuál es el gasto energético que generan.

En la tabla 3 se presenta el presupuesto de potencia que demandan los subsistemas agrupados como plataforma. También se muestra la demanda que presenta un experimento en operación, el de carga útil óptica, así como la demanda durante el envío de datos por el transmisor principal y la demanda que presenta el transmisor de respaldo, entre otros.

Tabla 3. Demanda de potencia y tiempo de operación de los subsistemas de SATEX I por modo operativo

	Modos Operativos del microsatélite			
M1 Modo de recuperación de energía. Solo permanece energizada la computadora del subsistema de potencia (SP), la cual realiza el monitoreo y la administración de los recursos energéticos del vehículo. El satélite inicia su vida operando en este modo y lo abandona cuando existe suficiente carga para proceder a ejecutar uno de los modos operativos subsecuentes.	Equipo	P [W]	min	Wmin
	Plataforma	13.5	101	1363
	Recepción en banda Ka	34.3	10	343
	Transmisor Principal VHF	27.0	10	270
	Experimento de la cámara digital	23.1	10	231
	Comunicaciones Ópticas	33.7	15	505
M2 Modo de monitoreo con la computadora de vuelo. El SP energiza al sistema mínimo de sobrevivencia (SIMSOS), el cual permanecerá en estado de mínimo consumo energético la mayor parte de tiempo, y a la computadora de vuelo (CV) para que ésta realice las primeras operaciones y los respectivos reportes a la estación terrena de control satelital (ETCS).	Componente	P. [W]		Wminuto
	Computadora de vuelo	6.17		61.7
	Procesador de Sobrevivencia	0.65		6.5
	Sensores de sol	0.96		9.6
	Sensores CT	1.56		15.6
	Fuente de Poder	7.9		79.0
	Transmisor Principal	7.71		77.0
	Ka Receptor	9.36		93.6

M3 Modo de Monitoreo Completo, permite realizar el modo 2 con la adición de diagnósticos de las computadoras de experimentos. Para recabar estado de computadoras de experimentos solicita vía red interna al SP las indicaciones sobre las computadoras que se pueden energizar para recabarles su estado operativo, éstas se activan por un tiempo menor a 2.5 minutos y posteriormente se desactivan. Con este modo operativo la ETCS recibe mayor información sobre el estado operativo de los equipos del satélite.

Componente	P. [W]	Wminuto
Computadora de vuelo	6.17	61.7
Procesador de Sobrevivencia	0.65	6.5
Sensores de sol	0.96	9.6
Sensores CT	1.56	15.6
Fuente de Poder	7.01	70.1
Transmisor Principal	7.71	77.1
Rx1 Principal	3.0	30

M4 Modo de envío de programas al satélite. Cuando los reportes de TLMN del satélite indican suficiente carga de las baterías, se puede subir un programa más complejo a la CV. Este programa se requiere para ejecutar aquellos experimentos que no están en el software básico del satélite.

Componente	P[W]	Wminuto
Computadora de vuelo	6.17	61.7
Procesador de Sobrevivencia	0.65	6.5
Sensores de sol	0.96	9.6
Sensores CT	1.56	15.6
Fuente de Poder	9.30	93.0
Transmisor Principal	7.71	77.1
Rx1 Principal	3.00	30
CUO	21.6	216.0

M5 Modo de Monitoreo Detallado. Una vez ejecutado con éxito el modo M4. el satélite puede realizar nuevas tareas o bien nuevos experimentos, entre ellas la captura de telemetría especial que se orienta a la recopilación de datos detallados sobre cualquiera de los sensores del satélite. Como en todos los modos, con excepción de M1 y M2, la solicitud de ejecución de este modo se efectúa desde la ETCS, en donde a su vez se especifican los sensores de telemetría especial (TLME), así como su frecuencia de muestreo. Una vez enviada la solicitud al satélite, la captura de TLME se realiza en la siguiente visita del vehículo a su ETCS.

Componente	P[W]	Wminuto
Computadora de vuelo	6.17	61.7
Procesador de Sobrevivencia	0.65	6.5
Sensores de sol	0.96	9.6
Sensores CT	1.56	15.6
Fuente de Poder	8.8	88.0
Cámara Digital	20	200.0

M6 Modo de estabilización en primera fase.
Para ejecutar este modo operativo se requiere que las baterías se encuentren en estado de carga alto, pues emplea la inyección continua de pequeños pulsos de corriente a las bobinas de torque magnético para reducir la complejidad dinámica del satélite después de la liberación de su vehículo lanzador. De igual forma esta fase de estabilización reduce las RPMs del satélite para proceder posteriormente a la liberación segura del gradiente gravitacional que se usará como medio pasivo de estabilización

Componente	P[W]	Wminuto
Computadora de vuelo	6.17	61.7
Procesador de Sobrevivencia	0.65	6.5
Sensores de sol	0.96	9.6
Sensores CT	1.56	15.6
Fuente de Poder	8.8	88.0
Cámara Digital	20	200.0

M7 Modo de liberación del gradiente gravitacional. Requiere la ejecución previa y exitosa del modo M6 para realizar la liberación segura del gradiente gravitacional. El modo se programa y envía desde el SET.

Componente	P [W]	Wminuto
Computadora de vuelo	6.17	61.7
Procesador de Sobrevivencia	0.65	6.5
Sensores de sol	0.96	9.6
Sensores CT	1.56	15.6
Fuente de Poder	8.8	88.0
Cámara Digital	20	200.0

M8 Modo de estabilización fina. Para utilizar este modo se requiere la ejecución previa y exitosa del modo M7. Este permitirá estabilizar al satélite para que realice los experimentos que demandan apuntamiento fino entre el satélite y la ETCS.

Componente	P [W]	Wminuto
Computadora de vuelo	6.17	61.7
Procesador de Sobrevivencia	0.65	6.5
Sensores de sol	0.96	9.6
Sensores CT	1.56	15.6
Fuente de Poder	8.8	88.0
Cámara Digital	20	200.0

M9 Modo experimento del sistema de sobrevivencia. Este experimento está planeado para ejecutarse en caso de que se llegaran a presentar problemas operativos drásticos tanto en la CV como en los equipos de RF.

Componente	P [W]	Wminuto
Computadora de vuelo	6.17	61.7
Procesador de Sobrevivencia	0.65	6.5
Sensores de sol	0.96	9.6
Sensores CT	1.56	15.6
Fuente de Poder	8.8	88.0
Cámara Digital	20	200.0

M10 Modo arquitectura de computadoras. Este experimento consume recursos energéticos equivalentes al modo M3, sin embargo permite supervisar el estado operativo de la CV y aplicarle mantenimiento automatizado en caso de requerirlo. Durante su operación se asegura la continuidad de las comunicaciones Tierra – Satélite aun después de que se presenten fallas graves en la computadora de vuelo. Este experimento no requiere de ningún tipo de estabilización satelital para llevarse a cabo.

Componente	P [W]	Wminuto
Computadora de vuelo	6.17	61.7
Procesador de Sobrevivencia	0.65	6.5
Sensores de sol	0.96	9.6
Sensores CT	1.56	15.6
Fuente de Poder	8.8	88.0
Cámara Digital	20	200.0

M11 Modo experimento de captura de imágenes. Este modo operativo se puede utilizar de dos formas, para propósitos de validación satelital como del sistema en particular. Este caso no requiere de ningún tipo de estabilización, por lo cual, debido a que consume poca energía, se puede realizar desde las fases operativas iniciales del satélite.

Componente	P [W]	Wminuto
Computadora de vuelo	6.17	61.7
Procesador de Sobrevivencia	0.65	6.5
Sensores de sol	0.96	9.6
Sensores CT	1.56	15.6
Fuente de Poder	8.8	88.0
Cámara Digital	20	200.0

M12 Modo de ejecución de experimento en banda Ka. Este modo requiere la ejecución previa y exitosa del modo M8 para que se puedan realizar comunicaciones en Banda Ka con línea de vista entre satélite y Tierra. Además, el modo solo se puede utilizar durante las visitas del satélite a su ETCS.

Componente	P[W]	Wminuto
Computadora de vuelo	6.17	61.7
Procesador de Sobrevivencia	0.65	6.5
Sensores de sol	0.96	9.6
Sensores CT	1.56	15.6
Fuente de Poder	8.8	88.0
Cámara Digital	20	200.0

M13 Modo experimento de comunicaciones ópticas. Este es el modo operativo más complejo que demanda además mayores recursos energéticos en el satélite. Su realización requiere la ejecución previa y exitosa del modo satelital M8 para reunir los requisitos indispensables para iniciar comunicaciones ópticas de bajada en el cercano infrarrojo.

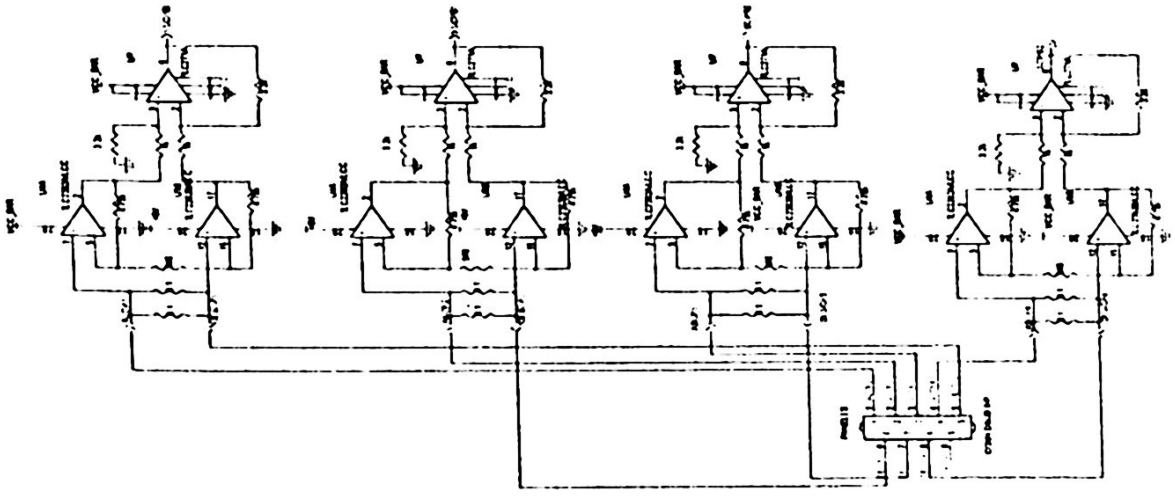
Componente	P [W]	Wminuto
Computadora de vuelo	6.17	61.7
Procesador de Sobrevivencia	0.65	6.5
Sensores de sol	0.96	9.6
Sensores CT	1.56	15.6
Fuente de Poder	8.8	88.0
Cámara Digital	20	200.0

El tiempo máximo permitido por experimento es aproximadamente de 10 minutos, esto es, el tiempo que el satélite estará en línea de vista con la estación terrena cuando sobrevuele el cenit de ésta. Se debe señalar que en un día terrestre el satélite describirá 14 vueltas a la Tierra, sin embargo, solo en tres de ellas tendrá línea de vista para realizar experimentos.

4. Resultados

En los últimos años el subsistema de potencia del satélite Satex ha pasado de utilizar una arquitectura de potencia centralizada (un convertidor único que alimenta a todos los subsistemas) a una de potencia distribuida. Bajo esta última arquitectura cada subsistema dispone de un convertidor dedicado. En este diseño se buscó que la topología de alimentación distribuida suministre el universo de voltajes y corrientes para cada demanda que presentan los subsistemas de SATEX. Esto es, no se utilizan convertidores de alta demanda (superiores a 25Watts) en donde cualquiera de las cargas presente una demanda máxima que no exceda 5 Watts. Para satisfacer los nuevos requerimientos se buscó una clara mejora de las características considerando la demanda y el tamaño de los componentes. En un primer acercamiento se buscó un diseño distribuido como el mostrado en la figura 4 empleando un bus común de 28V y señalando los niveles de tensión por subsistema, los módulos que requieren dos o tres niveles emplearan también dos o tres convertidores.

El arreglo de distribución de potencia que se logró se ilustra en la figura 4. La alimentación redundante que se asignó obedece a que los elementos de mayor prioridad en la operación del satélite son la computadora de vuelo y todo el sistema de sensado. Debido a que los transmisores pueden inducir ruido en la alimentación en el momento de entrar en operación, con la alimentación distribuida se protege a los demás subsistemas del fenómeno de ruido inducido.
En las figura 3-7 se presenta la conexión del sistema de distribución de potencia con el de control y sensado de la fuente de alimentación así como las tarjetas del circuito.



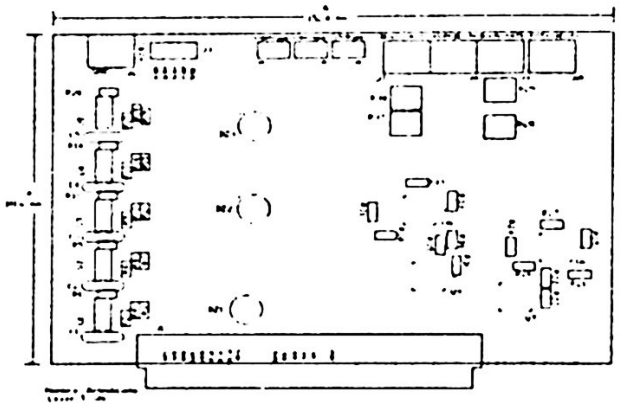


Figura 3. Sensores y acondicionamiento.

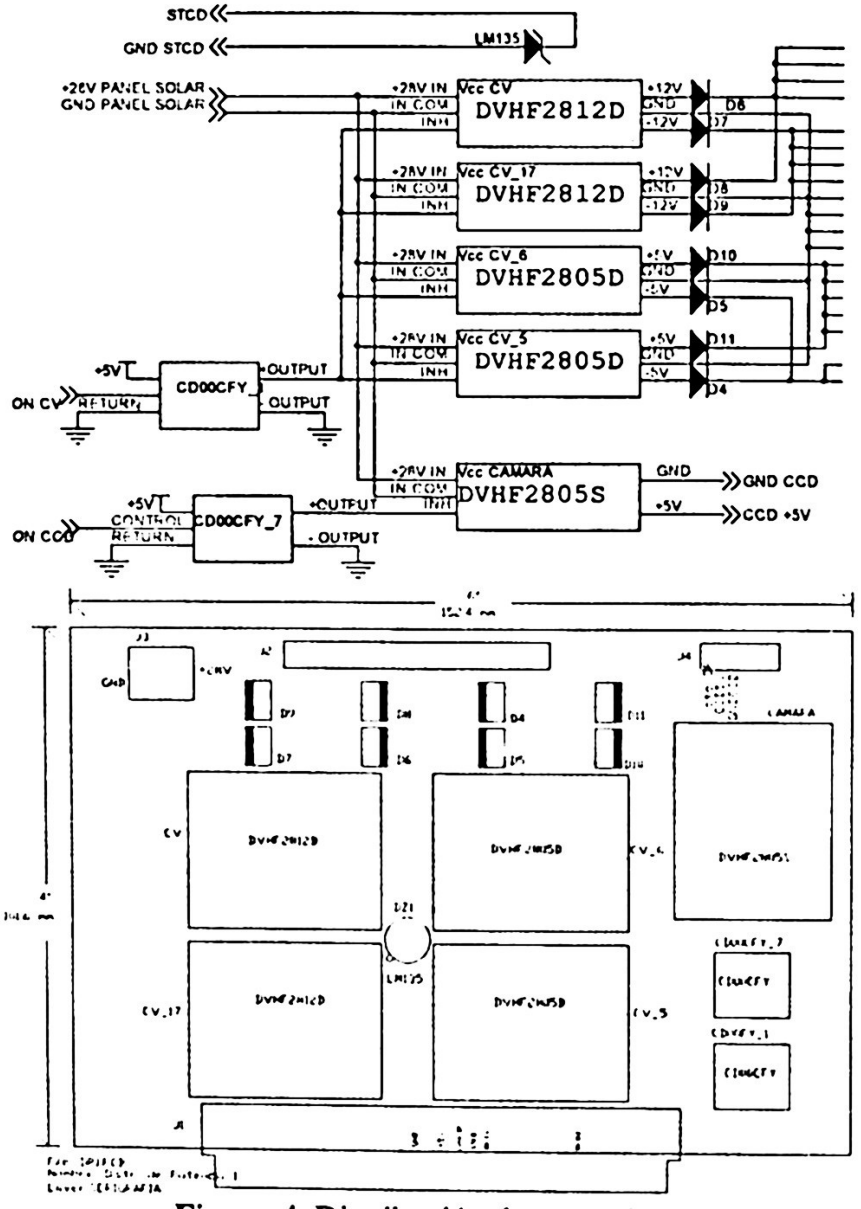


Figura 4. Distribución de potencia I.

5. Conclusiones

Debido a que durante el periodo orbital del microsatélite Satex se presentan se presentan condiciones operativas que demandan consumos energéticos hasta 300% superiores a lo que proporcionan sus paneles solares y a que éstos no pueden proporcionar energía durante periodos de eclipse, se requiere una fuente capaz de almacenar y retornar esa energía. Los dispositivos capaces de desarrollar esta función son arreglos de celdas de Níquel Cadmio, los cuales se utilizan para conformar la batería del satélite.

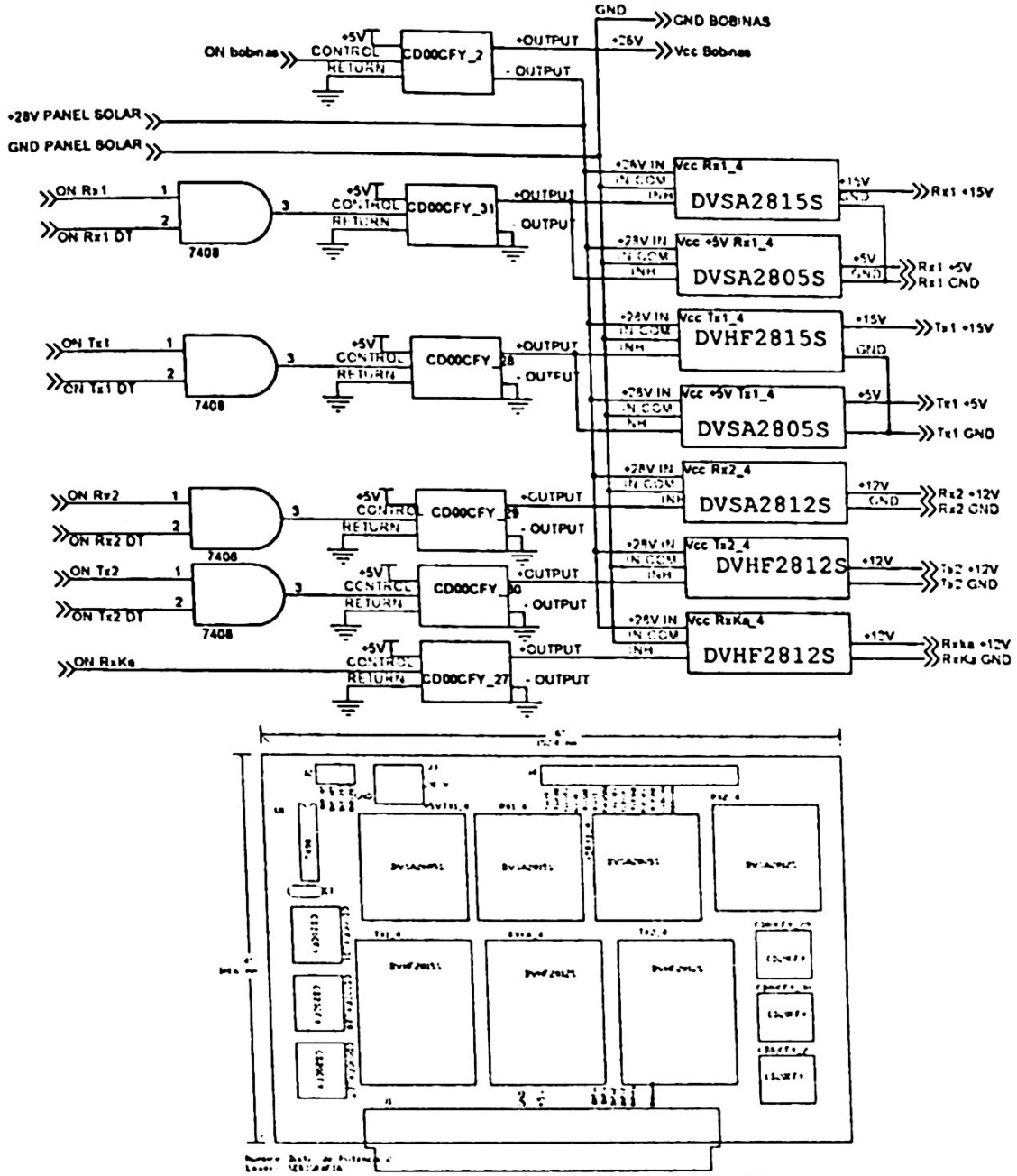


Figura 5. Distribución de potencia II.

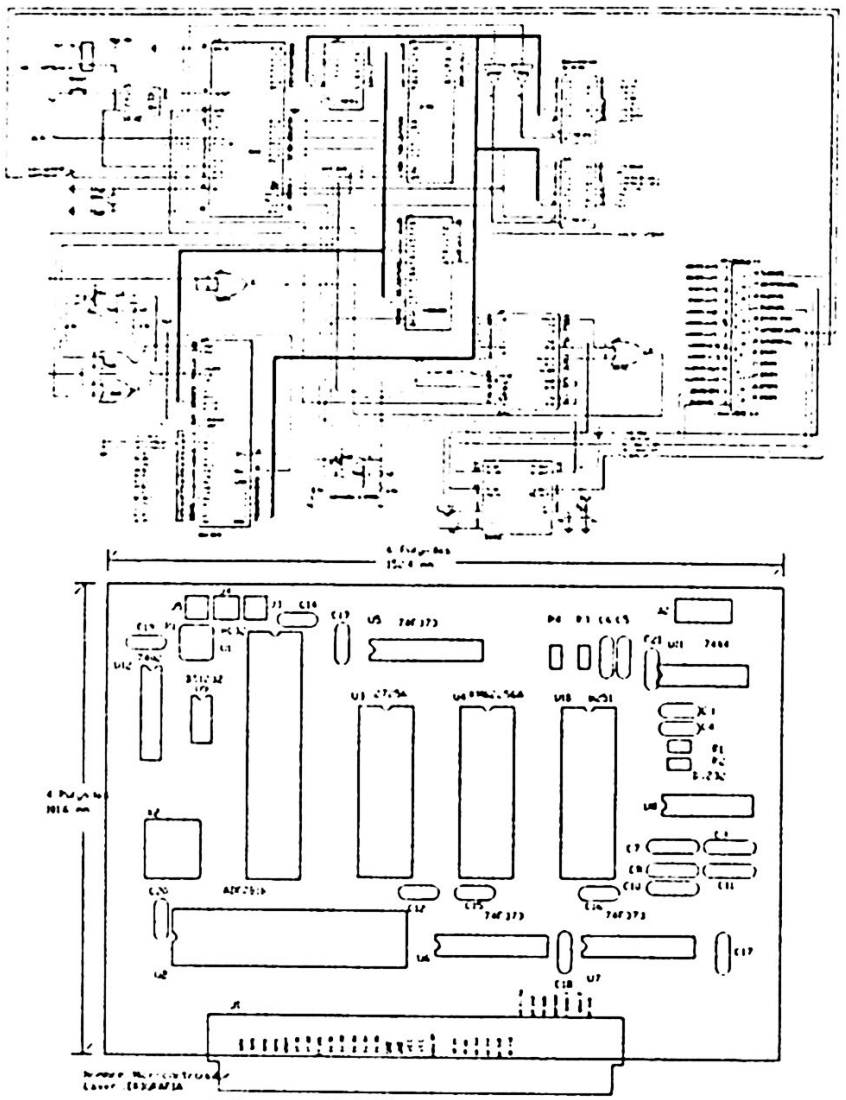
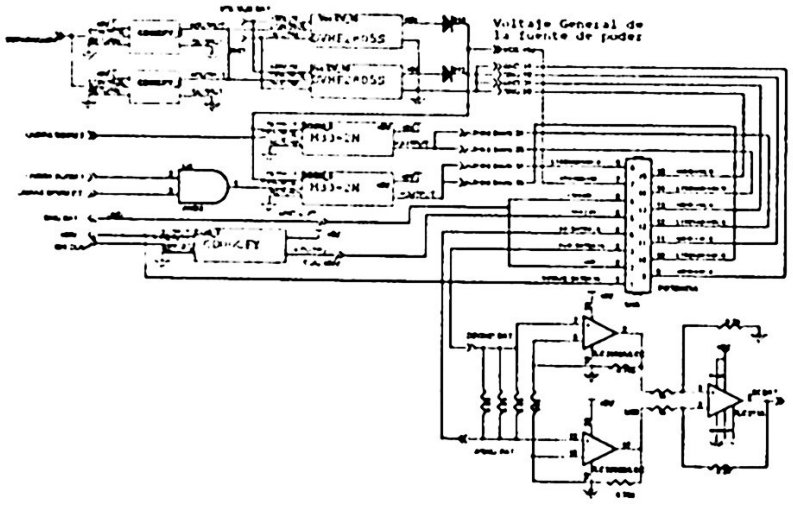


Figura 6. Microcontrolador.



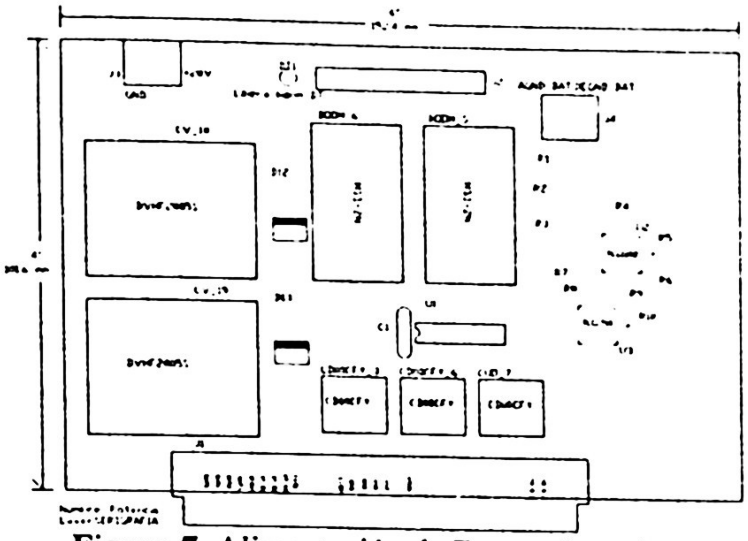


Figura 7. Alimentación de Fuente de poder.

La batería del Satex está compuesta por dos módulos de 26 pilas. Cabe subrayar, que una demanda repentina de corriente en exceso del 10% de la máxima capacidad de conversión de las celdas solares puede dañar severamente las mismas. Por esta razón se utilizan baterías recargables. Las baterías también sirven de fuente de energía en períodos de eclipse y se pueden recargar en períodos de sol.

El manejo eficiente del recurso energético del microsatélite SATEX cumple satisfactoriamente con los tipos de misiones que se realizarán en órbita, esto es dotar de energía suficiente para la operación del satélite y sus experimentos.

Sin embargo uno de los puntos más importantes a considerar es la extensión de la vida útil de las baterías, por lo que la filosofía originalmente planteada para la utilización de los experimentos se modificó de acuerdo con las limitaciones energéticas. De este modo, se concluye que las cargas útiles se habilitarán sólo cuando se cuente con energía suficiente en las baterías para su mejor desempeño.

El compromiso que existe para dotar de energía a cada una de las cargas, requiere realizar un buen presupuesto del consumo energético considerando la energía capturada por los paneles solares y el control oportuno del encendido de las cargas cuando se detecte que las baterías tienen suficiente energía almacenada.

El control eficiente de corriente y voltaje para cargar las baterías permitirá que la vida útil de las mismas se prolongue durante más tiempo, por lo que al emplear baterías recargables de carga rápida, se aprovecha su capacidad para convertir el exceso de carga y reciclarlo cuando se cargan a una velocidad entre 0.5 y 1.5 C (capacidad energética de la batería 3.4AH). Este rango de operación soportará los ciclos de luz y sombra que experimentarán los tableros solares durante la vida útil del vehículo espacial.

Referencias

1. www.satex.org.mx
2. Andrés Calvillo, Homero Ramos, Javier Roch. Estimación de la energía captada en el microsatélite SATEX - I. 3er Congreso Internacional de Ingeniería Electromecánica y de Sistemas ICESE-2002 del 25-29 Noviembre; D. F. México.
3. Andrés Calvillo Téllez y Esaú Vicente Vivas. Diseño y estimación de los sistemas de generación y almacenamiento de energía para el microsatélite experimental mexicano SATEX I. 3er Congreso Internacional de Ingeniería Electromecánica y de Sistemas ICESE-2002 del 25-29 Noviembre; D. F México.
4. Calvillo-Téllez, E. Barba Cervantes y E. Vicente-Vivas. Manejo eficiente del recurso energético del satélite experimental "satex" Memorias de la Cuarta Conferencia Internacional en Control, Instrumentación Virtual y Sistemas Digitales CIC-INDI2002, 26-30 Agosto: Pachuca Hidalgo México.

Chattering adaptive control for first order systems

Ricardo Guerra[1] and Leonardo Acho[1]

[1]CITEDI-IPN, Av. del Parque No. 131°, Mesa de Otay, Tijuana, Baja California, México C.P. 22510
{rguerra,leonardo}@citedi.mx

Abstract.- A new fast switching adaptive algorithm is presented as an alternative solution to the Reference Model Adaptive Control (RMAC) problem for first order linear time-invariant systems. As shown through numeric examples, the proposed control scheme deals with the problem of parameter variance that arises when the nominal system presents un-modeled dynamics y added noise al the output of the plant to be controlled, that in many cases is destructive to the closed loop system. The closed loop Lyapunov stability analysis is presented.

Key Words: Adaptive Control, linear system, reference model.

Resumen.- Se presenta un nuevo algoritmo de adaptación por conmutación rápida como una solución alterna al problema de control adaptable por el método de modelo de referencia (CAMR) para sistemas lineales de primer orden invariantes en el tiempo. Como se muestra a través de experimentos numéricos, nuestra propuesta puede hacer frente al problema de deriva parámetrica que aparece cuando el sistema nominal presenta dinámicas no modeladas y ruido aditivo en la salida de la planta a controlar, que en muchos casos, esta deriva parámetrica resulta destructiva para el sistema en lazo cerrado. Análisis de estabilidad del sistema en lazo cerrado es hecho por medio de la teoría de estabilidad en el sentido de Lyapunov.

Palabras clave: Control adaptable, sistemas lineales, modelo de referencia.

1. Introduction

Reference Model Adaptive Control techniques have been widely used in linear systems, especially first order linear systems (Slotine & Li, 1999). This is because in engineering there are a number of systems of interest that can be modeled by first order systems. For

instance, some thermal systems (Smith & Corripio, Chap. 3), the break system used automobiles and some electronic systems (Slotine & Li, 1999, Page 326) can be modeled by first order systems. One of the main problems of Reference Model Adaptive Control techniques is the parameter variance that that in most cases proves to be destructive to system to be controlled (Slotine & Li, 1999, Page 354). In this sense, as is shown through numeric examples, a new adaptation law is presented, that is capable of overcoming the problem of parametric variance. This proposal is based on the introduction of a fast switching term by means of the sign function into the control law, which makes the closed loop system more robust. The addition of fast switching terms has been effective in the control of robot manipulators with friction (Orlov et al., 2003). Lyapunov analysis is used to prove stability of the undisturbed closed loop system. This paper is distributed follows: Section II presents the modeling and solution of the reference model adaptive control giving a numeric example of the problem of parameter variance. Section III introduces the new algorithm and it is numerically shown that the parameter variance not present in section II. Finally, section IV presents the conclusions.

2. Reference Model Adaptive Control

Consider the nominal system to be approximately the following differential equation:

$$\dot{y} = -a_p y + b_p u \qquad (1$$

where y is the plan output, u is the control input and a_p and b_p are constant system parameters. From the adaptive control viewpoint, a_p and b_p are unknown, but the sign of b_p ($sgn(b_p)$) is known.

Consider the following reference model:

$$\dot{y}_m = -a_m y_m + b_m r(t) \qquad (2)$$

where a_m and b_m are constant parameter and $r(t)$ is an external bounded reference input. The input parameter a_m must be strictly positive to ensure the stability of the reference model.

Outline of the RMAC problem.

The reference model adaptive control problem consists of finding a control law $u(t)$ such that:

$$u(t) = \hat{a}_r(t) r + \hat{a}_y(t) y \qquad (3)$$

where $\hat{a}_r(t)$ and $\hat{a}_y(t)$ are adaptive gains that adjust dynamically, i. e., $\hat{a}_r(t)$ and $\hat{a}_y(t)$ are solutions to:

$$\dot{\hat{a}}_r(t) = f_1(\text{sgn}(b_p), \gamma, e, r) \tag{4}$$

$$\dot{\hat{a}}_y(t) = f_2(\text{sgn}(b_p), \gamma, e, y) \tag{5}$$

where γ is a positive constant that represents the adaptation gain and $e=y-y_m$, so that:

$$A) \lim_{t\to\infty} e(t) = 0 \tag{6}$$

$$B) \left|\hat{a}_r(t)\right| < \infty \quad \text{and} \quad \left|\hat{a}_y(t)\right| < \infty \quad \forall t \geq 0 \tag{7}$$

Commentary 1. Under the assumption of persistent input, reference model adaptive control schemes can be used as parameter identification schemes, explicitly, the values of a_p and b_p may be estimated by $\hat{a}_r(t)$ and $\hat{a}_y(t)$ (Slotine & Li, 1999). ♦

Using:

$$\dot{\hat{a}}_r(t) = -\text{sgn}(b_p)\gamma e r \tag{8}$$

$$\dot{\hat{a}}_y(t) = -\text{sgn}(b_p)\gamma e y \tag{9}$$

solves the RMAC problem (Slotine & Li, 1999, Page 328). The problem of parameter variance may appear when the reference signal $r(t)$ is constant and the nominal model contains un-modeled dynamics and added external noise (Slotine & Li, 1999, Pages 354-356). Consider the example shown in Fig. 1 (Slotine y Li, 1999). By using $\gamma=1$, $r(t)=2$, $n(t)=0.5\sin(16.1t)$, $y_m(0)=0$, $y(0)=-1$, $\hat{a}_r(0)=1$, $\hat{a}_y(0)=-1$ and with the un-modeled dynamic at rest, the result shown in Fig. 2 are obtained. Note that the adaptation parameters $\hat{a}_r(t)$ and $\hat{a}_y(t)$ vary resulting in a variance of $y(t)$.

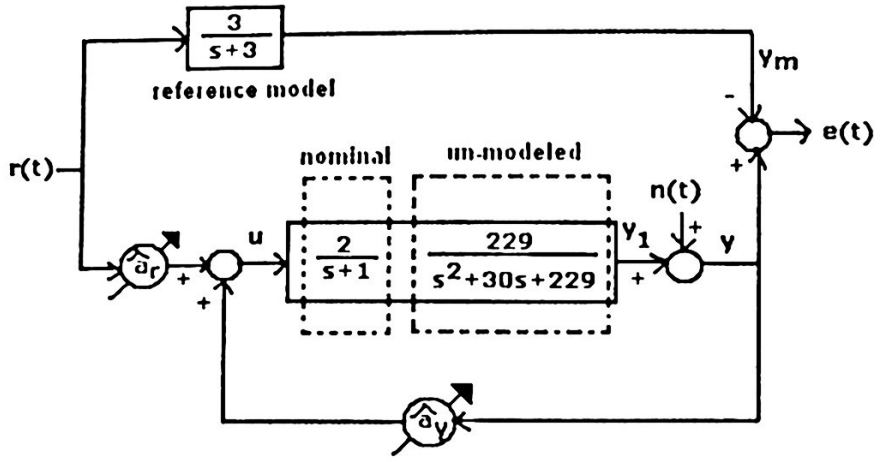


Fig. 1 Adaptive control scheme with un-modeled dynamic and added noise.

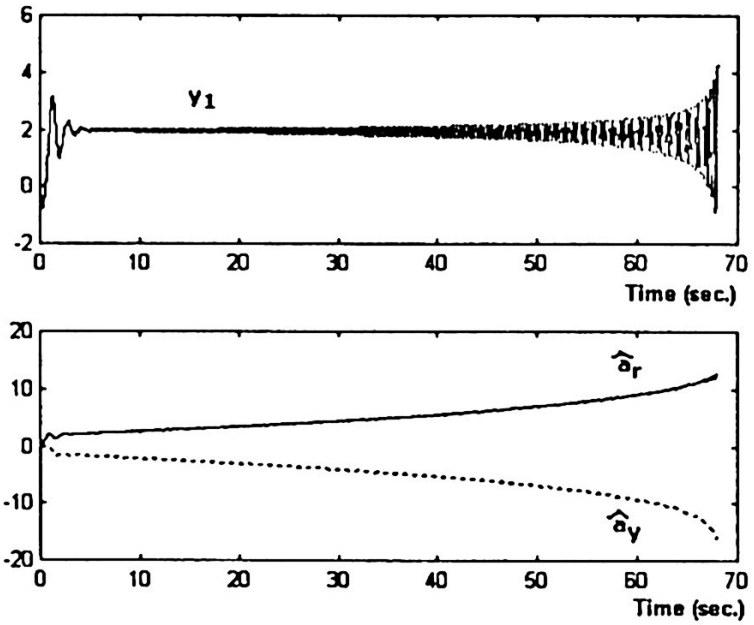


Fig. 2 Parameter variance.

3. RMAC via fast switching

In (Orlov et al., 2003) it is shown that the proper introduction of fast switching robustifies the closed loop system. In this sense, the following solution to the RMA' problem is proposed:

Theorem 1. Given the nominal system (1) and the reference model (2) with *u(t)* in (3)

$$\dot{\hat{a}}_r(t) = -\text{sgn}(b_p)\gamma e r \tag{10}$$

$$\dot{\hat{a}}_y(t) = -\text{sgn}(b_p)\gamma e\,\text{sgn}(y) \tag{11}$$

the RMAC problem is solved.

The addition of the fast switching term is is shown in (11) through the use of the sign function.

Proof.
Let

$$\tilde{a}_r = \hat{a}_r - a_r^* \tag{12}$$

$$\tilde{a}_y = \hat{a}_y - a_y^* \tag{13}$$

where a_r^* and a_y^* are positive constants that would yield perfect model estimation, i.e., if

$$\hat{a}_r(t) = a_r^* = \frac{b_m}{b_p} \tag{14}$$

$$\hat{a}_y(t) = a_y^* = \frac{a_p - a_m}{b_p} \tag{15}$$

the closed loop system (3) in (1) would be:

$$\dot{y} = -a_m y + b_m r \tag{16}$$

For this, knowledge of a_p and b_p is obviously required. Given $e=y-y_m$, we obtain

$$\begin{aligned}\dot{e} = \dot{y} - \dot{y}_m &\underset{(1),(2),\text{and}(3)}{=} -a_m(y - y_m) + (a_m - a_p + b_p\hat{a}_y)y + (b_p\hat{a}_r - b_m)r \\ &= -a_m e + b_p(\tilde{a}_r r + \tilde{a}_y y)\end{aligned} \tag{17}$$

Defining the Lyapunov function as:

$$V(e,\tilde{a}_r,\tilde{a}_y)=\frac{1}{2}e^2+\frac{1}{2\gamma}|b_p|\tilde{a}_r^2+\frac{k_1}{2\gamma}|b_p|\tilde{a}_y^2$$

where k_1 is a positive constant. The time derivative of (18) yields:

$$\dot{V}(e,\tilde{a}_r,\tilde{a}_y)=e\dot{e}+\frac{1}{\gamma}|b_p|\tilde{a}_r\dot{\hat{a}}_r+\frac{k_1}{\gamma}|b_p|\tilde{a}_y\dot{\hat{a}}_y$$
$$\underset{(10),(11),y(17)}{=}-a_m e^2+b_p\tilde{a}_r ye-k_1 b_p\tilde{a}_y e\,\mathrm{sgn}(y)$$

By using the following approximation:

$$\mathrm{sgn}(y)=\lim_{\varepsilon\to 0}\frac{y}{|y|+\varepsilon},$$

from (19) we have:

$$\dot{V}(e,\tilde{a}_r,\tilde{a}_y)=\lim_{\varepsilon\to 0}[-a_m e^2+b_p\tilde{a}_r ye-k_1 b_p\tilde{a}_y e(\frac{y}{|y|+\varepsilon})]-\frac{k_1 y}{|y|+\varepsilon}\le -y$$
$$\Rightarrow |y|+\varepsilon\le k_1$$

and thus from (20) we obtain:

$$\dot{V}(e,\tilde{a}_r,\tilde{a}_y)\le[-a_m e^2+b_p\tilde{a}_r ye-b_p\tilde{a}_y ey]=-a_m e^2$$

or:

$$\dot{V}(e,\tilde{a}_r,\tilde{a}_y)\le -a_m e^2 \qquad (21$$

which implies that $e(t), \tilde{a}_r(t), \tilde{a}_y(t) \in L_\infty$, and from the integration of (21) we conclude that $e(t) \in L_2$. From the previous in (17) we conclude that $\dot{e}(t) \in L_\infty$ (since y_m is bounded), and using Barbalat's lemma we conclude that $\lim_{t\to\infty} e(t) = 0$. ♦

To prove the effectiveness of the algorithm presented in *Theorem* 1, the same experiment shown in Fig. 1 is repeated with (10), (11) and (3). The results are shown in Fig. 3.

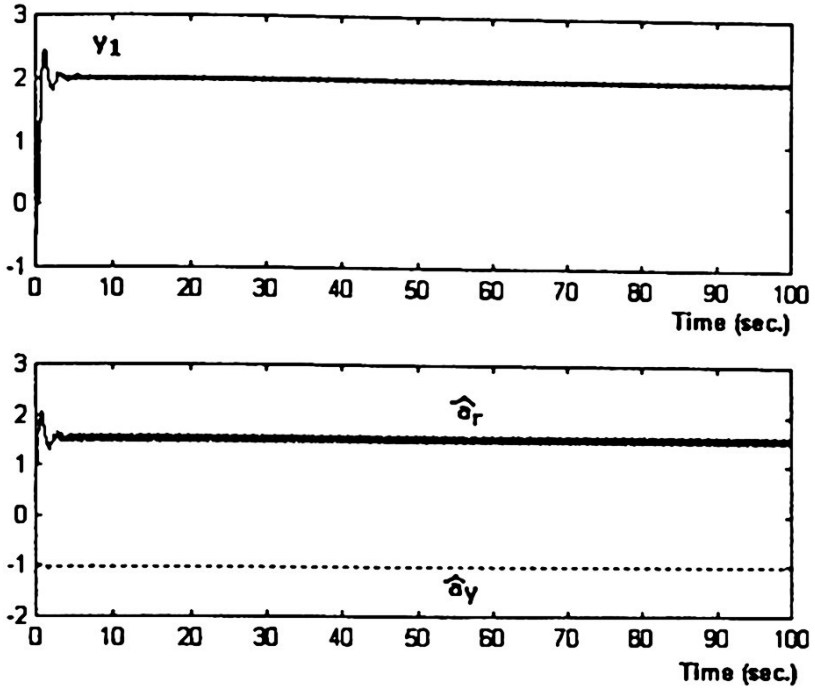


Fig. 3 Numeric results.

4. Conclusions

A new reference model adaptive control algorithm that uses a fast switching term in one of the adaptation laws, proving, from a numeric point of view, the efficacy of the algorithm when facing the problem of parametric variance. And while it is true that the inclusion of fast switching terms requires control signals that are rich in frequency content, said signals can be approximated by smoother functions (Christopher E., and Spurgeon S. K., 1998, Page 15), which is common in automatic control

References

Christopher E., y Spurgeon S. K., *Sliding Mode Control: Theory and Applications*, Taylor and Francis, British, 1998.

Orlov I., Alvarez J., Acho L., y Aguilar L., "Chattering Control of Coulomb Friction Systems", *5th. IFAC Symposium, NOLCOS*, St. Petersburgh, Russia, 2001.

Slotine J.-J. E., y Li W., *Applied Nonlinear Control*, Prentice Hall, USA, 1991.

Smith C.A. y Corripio A.B., *Principles and Practice of Automatic Process Control*, John Wiley and Sons, Canada, 1985.

Stability analysis for switched systems*

E. Rubio, F. Rios

Centro de Investigación en Computación. CIC-IPN.
U.P. Adolfo López Mateos. Zacatenco.
México, D.F. MÉXICO.
e-mail: erubio, frios@cic.ipn.mx

Abstract. In this paper, we treat the Hurwitz stability criterion for a convex fuzzy set of matrices represented by autonomous linear time invariants systems. We give necessary and sufficient conditions to assure the Hurwitz conditions in order to attest the existence of a common quadratic Lyapunov function in an arbitrary set of matrices, associated by a fuzzy set.

Keywords: Switched systems, Lyapunov functions, convex fuzzy sets, Hurwitz matrix, pencil of matrices

1 Introduction

This paper discusses the problem of find the negative roots of a set of matrices, in order to assure the stability conditions for a stable performance of a switched system ΣA_z by a pair of linear subsystems ΣA_i and ΣA_j. We consider the problem of verifying the stability of a convex set K of matrices, more specifically in an interval matrix. These results are motivated by the fact of obtain a stable performance of the fuzzy switched systems, minimize the energy dissipated and to preserve the stability properties in all the switched actions [12], [2], [9], [4]. We assure the stable performance for the convex set of matrices in its pencil. The matrices treated in the pencil, belong to linear time-invariant systems $\dot{x}(t) = Ax(t)$ where the parametric matrix $A \in K$, and the system resultant after the switched actions, is considered a linear time-variant system $\dot{x}(t) = A_z x(t)$. In this paper, we give necessary and sufficient conditions in K.

By a switched system, we mean a hybrid dynamical system consisting of a family of continuous-time subsystems and rules that orchestrates the switching between them.

Switching systems have numerous applications in control of mechanical systems, the automotive industry, aircraft and air traffic control, switching power converters and many others fields. In the last few years, every major control conference has had several regular and invited sessions on switching systems and control [7], [10].

* This work was supported by the Instituto Politécnico Nacional – México under the program SUPERA-ANUIES, COFAA-IPN and COTEPABE-IPN.

We assume that the states of the Switched Fuzzy System (SFS), does not jump at the switching instant, i.e., the solution $x(\cdot)$ is every where continuous. The systems ΣA_i and ΣA_j are continuous, and in the Switching Action (SA) there are interrupted by switching signals then, the switched system resulting is every where continuous.

The stability conditions for the pencil of matrices, are not in terms of the Lyapunov equation, there are exposed by Hurwitz stability criterion. This criterion is useful to establish fundaments to demonstrate the existence of a common quadratic Lyapunov function (CQLF) in an arbitrary compact convex sets, including the interval of matrices. A general result is given for matrices of n order.

Here, we do not treat extensively with the continuous fuzzy systems, we only utilize some notation and ideas. For more information about fuzzy systems, see [16], [12].

This paper is organized as follows, in section 2, we give some mathematical preliminaries and definitions. In section 3 , we present the principal results by a set of theorems and, finally the section 4 is about the conclusions.

2 Preliminaries

Consider the dynamical system

$$\dot{x} = Ax, \qquad A \in \mathcal{K} \triangleq \{A_1, A_2, ..., A_m\} \tag{1}$$

where A_i, $i=\{1,2,...,m\}$, are constant matrices in R^{nxn}. The matrices A_i, are assumed Hurwitz[1]. An important problem is to determine necessary and sufficient conditions for assure that a pencil of Hurwitz matrices nxn is Hurwitz in its interval (the SFS ΣA_z resultant are for an arbitrary switching sequence Hurwitz). This result is showed in the theorems 5 and 6 for the case 2x2 and nxn respectively. Hence, the existence of such conditions are sufficient to guarantee the uniform asymptotic (exponential) stability of the switching system, compose by elements of the set A.

Definition 1. The switched matrix is:

$$A_z(t) = \left(\frac{w_i A_i + w_j A_j}{w_i + w_j} \right) \tag{2}$$

where $i \neq j$ and $i,j \in \{1,...,m\}$, m is the number of systems in the phase plane, $x \in R^{nx1}$ is the state vector, and the matrices $A_i \in R^{nxn}$ each linear component is called subsystem, furthermore $w_i=(1-w_i)$. We can see from (2) that it is possible to represent a switched fuzzy system like Takagi-Sugeno a fuzzy model [5], [16], then

---

[1] The eigenvalues of each A_i matrix lies in the open left half of the complex plane, denoted C^-; at times, for clarity, we also refer to this as an asymptotically stable matrix.

$$\dot{x}(t) = \left(\frac{\sum_{z=i,(j)}^{j,(i)} w_z \, A_z(t)x(t)}{\sum_{z=i,(j)}^{j,(i)} w_z} \right) \tag{3}$$

where i,j∈ { 1,2,...,m}, and each linear component of $A_z(t)x(t)$ is called subsystem. Thus, if $w_z \in [0,1]$, then we can write (2) as

$$A_z(t) = (w_i A_i + (1 - w_i)A_j) \tag{4}$$

thence, the pencil of matrices is

$$\sigma_w [A_i, A_j] = A_z(t) \tag{5}$$

Definition 2 *A switched fuzzy system (SFS) Σ_{A_z} is an autonomous time-varying system defined by*

$$\dot{x}(t) = A_z(t)x(t) \tag{6}$$

where $A_z(t)$ will be is considered continuous for all t.

Definition 3 *A fuzzy switching action (FSA) is defined like the change of dynamic between two linear autonomous time-invariant systems Σ_{A_i} and Σ_{A_j} who are components of Σ_{A_z}.*

With the FSA, we can jump between the fuzzy sets of Σ_{A_i} , $i = \{1, 2, ..., m\}$ in the phase plan $\mathbb{R}^{n \times n}$. Then a FSA can be represented by the equation (2).

Definition 4 *A fuzzy subset $\mathcal{FC}$ of $\mathbb{R}^n$ is said to be convex if*

$$\left(\frac{w_i A_i x + w_j A_j}{w_i + w_j} \right) \in \mathcal{FC} \tag{7}$$

whenever $\{A_i, A_j\} \in \mathcal{FC}$, $\{w_i, w_j\} \subset w$, and $w \in [0,1]$. The convex fuzzy sets have to contain, along with any two distinct matrices A_i or A_j, a certain portion of the line trough A_i and A_j, thus

$$\left\{ \frac{w_i A_i + w_j A_j}{w_i + w_j} \;\middle|\; w \in [0,1] \right\}$$

3 Stability conditions

We will consider under the assumption that all the individual subsystems are asymptotically stables. Basically, we will find that the stability is ensured if the FSA is well proposed. These conditions are in terms of convex linear combinations of matrices A_i and A_j.

An important problem is to determine necessary and sufficient conditions for the existence of a quadratic Lyapunov function $V(x) = x^T Px$, $P = P^T > 0$, such that its time derivative along any trajectory of the system (2) is negative definite, or alternatively that,

$$A_i^T P + PA_i = -Q_i \tag{8}$$

where the matrices Q_i are symmetric and positive definite. The existence of such Lyapunov function is sufficient to guarantee the uniform asymptotic stability of switching system (2).

However, the conclusions we reached about the stable equilibrium of the system can also be reached by using energy concepts.

In the following discussion we attempt to determine a common Lyapunov function of the form

$$V(x) = x^T Px \tag{9}$$

where $x \in \mathbb{R}^{n\times n}$.

In this section, necessary and sufficient conditions are derived in order to assure the existence of a common quadratic Lyapunov function for a finite number of n order linear time-invariant systems.

Lemma 1 *If P is a positive definite common matrix, such that*

$$A_i^T P + PA_i < 0 \quad \text{and} \quad A_j^T P + PA_j < 0, \tag{10}$$

where $A_i, A_j, P \in R^{n\times n}$, then from (2), we can write

$$\left(\frac{w_i A_i + w_j A_j}{w_i + w_j}\right)^T P + P\left(\frac{w_i A_i + w_j A_j}{w_i + w_j}\right) < 0 \tag{11}$$

$\forall w \in [0, 1]$.

Proof: since P is a positive definite matrix, and $\left(\frac{w_i A_i + w_j A_j}{w_i + w_j}\right) < 0$, $\forall w \in [0, 1]$, so (11) is fulfilled.

Is well know that if the Lyapunov's condition

$$A_z^T P + PA_z = -Q \tag{12}$$

is fulfilled, the switched system $\sum_{A_z}$ is stable for some $P = P^T > 0$.

Theorem 2 *A matrix A_z is Hurwitz if and only if for every nonzero vector x there exist $P = P^T > 0$ such that*

$$x^T \left(A_z^T P + P A_z\right) x < 0 \tag{13}$$

Proof: If A_z is Hurwitz $\forall w \in [0, 1]$, then (12)⇒(13) otherwise (13) is violated if x is an eigenvector corresponding to an unstable eigenvalue of A_z.

Theorem 3 *The equilibrium of a fuzzy switched system*

$$\dot{x}(t) = \left(\frac{w_i A_i + w_j A_j}{w_i + w_j}\right) x(t) \tag{14}$$

is globally asymptotically stable if there exists a common positive definite matrix P for all the subsystems such that

$$A_z^T P + P A_z < 0 \tag{15}$$

Proof: Consider the scalar function $V(x(t))$ such that $V(x(t)) = x^T(t) P x(t)$, where P is a positive definite matrix, then

$$\begin{aligned}
\dot{V}(x(t)) &= \dot{x}(t)^T P x(t) + x(t)^T P \dot{x}(t) \\
&= x(t)^T \left(A_z^T P + P A_z\right) x(t) \\
&= x(t)^T \left(\left(\tfrac{w_i A_i + w_j A_j}{w_i + w_j}\right)^T P + P \left(\tfrac{w_i A_i + w_j A_j}{w_i + w_j}\right) \right) x(t) \\
&= x(t)^T \left(\tfrac{1}{w_i + w_j} \left(w_i A_i^T P + w_j A_j^T P + w_i P A_i + w_j P A_j\right) \right) x(t) \\
&= \tfrac{1}{w_i + w_j} \left(x(t)^T \left(w_i \left(A_i^T P + P A_i\right) + w_j \left(A_j^T P + P A_j\right) \right) x(t)\right)
\end{aligned}$$

where $w_i, w_j \in [0, 1]$, for $i, j \in \{1, 2, ..., m\}$.

From Lemma 1 and equation (15), we obtain

$$\dot{V}(x(t)) < 0$$

Then, $V(x(t))$ is a Lyapunov function and the fuzzy switched system (14) is globally asymptotically stable.

Theorem 4 *Let $\Sigma_A = \{\Sigma_{A_1}, \Sigma_{A_2}, ..., \Sigma_{A_n}\}$ be a set of stable systems with a CQLF. Since the matrix A_z is composed by pairs of elements of Σ_A, then A_z is Hurwitz and Σ_{A_z} is stable $\forall w_k \in [0, 1]$ for any fuzzy switching sequence.*

Proof: Since $V(x)$ is a QLF for Σ_{A_1}, $\Sigma_{A_2}$and Σ_{A_n}, then $V(x)$ will be a QLF for Σ_{A_z}. Then Σ_{A_z} is stable, and the matrix A_z is Hurwitz $\forall w \in [0,1]$ then, $V(x)$ is a CQLF. If A_z is not Hurwitz for any w, then a CQLF can not exists for Σ_{A_1}, $\Sigma_{A_2}$and Σ_{A_n}, thus a necessary condition for the existence of a CQLF for Σ_{A_1}, $\Sigma_{A_2}$and Σ_{A_n}, is that the matrix A_z be Hurwitz $\forall w$.

Theorem 5 *Let A_i and A_j be two Hurwitz matrices in $\mathbb{R}^{2\times 2}$. A necessary and sufficient condition for the dynamic systems Σ_{A_i} and Σ_{A_j} to have a quadratic Lyapunov function that the matrices A_iA_j and $A_iA_j^{-1}$ do not have real negative eigenvalues. An equivalent condition is that the pencils $\sigma_w[A_i, A_j]$, $\sigma_w[A_i, A_j^{-1}]$ are both Hurwitz for $w \in [0,1]$.*

Proof: The necessity can be directly obtained by theorem 3. The implication is showing that a sufficient condition for the matrix pencil $\sigma_w[A_i, A_j]$ to be Hurwitz is that both A_i and A_j are Hurwitz, and that the matrix $A_iA_j^{-1}$ has no real negative eigenvalues [17], [14].

The matrix pencil $\sigma_w[A_i, A_j]$ has the next characteristic polynomial

$$\det(\lambda I - \sigma_w[A_i, A_j]) = \lambda^2 - \lambda(trace(\sigma_w[A_i, A_j])) + \det(\sigma_w[A_i, A_j]) = 0$$

The pencil fulfill the Hurwitz conditions for all $w \in [0,1]$ if and only if:

a) $trace(\sigma_w[A_i, A_j]) < 0$ and

b) $det(\sigma_w[A_i, A_j]) > 0$.

It follows that the condition *a)* is satisfied if both A_i and A_j are Hurwitz. Thus for *b)*, we have that the equation for the determinant can be written as

$$\begin{aligned}\det(\sigma_w[A_i, A_j]) &= \det(w_iA_i + (1-w_i)A_j) \\ &= w_i^2 \det(\gamma I + A_iA_j^{-1}) \det(A_j) \qquad (16)\end{aligned}$$

with $\gamma = (1-w_i)/w_i$ for $w_i \in [0,1]$. Due to A_j is Hurwitz, then the eigenvalues of $\det(A_j)$ are negative so, $\det(A_j)$ is positive. In order to know the eigenvalues for $(\gamma I + A_iA_j^{-1})$, we set $\vartheta = A_iA_j^{-1}$,

$$\det(\gamma I + \vartheta) = \gamma^2 + \gamma(trace(\vartheta)) + \det(\vartheta) = 0$$

The conditions to accomplish for the latter equation is that the eigenvalues have no real negative parts, then

$$trace(\vartheta) > 0 \quad \text{and} \quad det(\vartheta) > 0 \qquad (17)$$

by determinants, we will analyze the conditions for ϑ

$$\det(\lambda I - \vartheta) = \lambda^2 - \lambda(trace(\vartheta)) + \det(\vartheta) = 0 \qquad (18)$$

therefore by Routh theorem [6], we can see that the equation given above only accept positive eigenvalues, in fact we can establish differents conditions by Hurwitz theorem, Descartes' sign rule, etc., [6]. Then from (18), we have $trace(\vartheta) > 0$ and $det(\vartheta) > 0$, then (18) is satisfied and $\det(\gamma I + \vartheta) > 0$. Finally, the eigenvalues of $\det\left(\gamma I + A_i A_j^{-1}\right)$ are positive, then are positive also for (16), so the condition *b*) is accomplished; in other words the matrix $A_i A_j^{-1}$, has no real negative eigenvalues.

Remark 1 *In the last proof, we utilise the condition (17), but is easy to see that we can analyse some others differents cases, for example:*

- $trace(\vartheta) > 0$ *and* $\det(\vartheta) > 0$ *and* $|\det(\vartheta)| > |trace(\vartheta)|$,
- $trace(\vartheta) > 0$ *and* $\det(\vartheta) < 0$ *and* $|\det(\vartheta)| < |trace(\vartheta)|$,
- $trace(\vartheta) < 0$ *and* $\det(\vartheta) < 0$ *and* $|\det(\vartheta)| + |trace(\vartheta)| < \gamma$,

  etc...

The next theorem is an extension of the above result, where the matrices are in $\mathbb{R}^{n\times n}$.

Theorem 6 *Let A_i and A_j be two Hurwitz matrices in $\mathbb{R}^{n\times n}$. A necessary and sufficient condition for the dynamic systems Σ_{A_i} and Σ_{A_j} to have a quadratic Lyapunov function is that the matrices $A_i A_j$ and $A_i A_j^{-1}$ do not have real negative eigenvalues. An equivalent condition is that the pencils $\sigma_w[A_i, A_j]$, $\sigma_w\left[A_i, A_j^{-1}\right]$ are both Hurwitz for $w \in [0, 1]$.*

Proof: In this proof, the necessity is in a similar way as the latter. So, for the necessity, we have that the conditions of equation (16), show that A_j is Hurwitz, then $sign(\det(A_j)) = sign(\sigma_w[A_i, A_j]) = (-1)^n$, if

$$\det(\gamma I + \vartheta) > 0 \tag{19}$$

$$\begin{aligned}\det(\gamma I + \vartheta) \;=\; & \gamma^n \\ & + \gamma^{n-1} \sum_{i=1}^{n} g_{ii} \\ & + \gamma^{n-2} \sum_{i=1}^{n} \sum_{i<i} \begin{vmatrix} g_{ii} & g_{ij} \\ g_{ji} & g_{jj} \end{vmatrix}\end{aligned}$$

$$+\gamma^{n-3}\sum_{i=1}^{n}\sum_{i<j}\sum_{i<j<k}\begin{vmatrix} g_{ii} & g_{ij} & g_{ik} \\ g_{ji} & g_{jj} & g_{jk} \\ g_{ki} & g_{kj} & g_{kk} \end{vmatrix}$$

$$\vdots$$

$$+|\vartheta|$$

where $\vartheta = \begin{bmatrix} g_{11} & \cdots & g_{1n} \\ \vdots & \ddots & \vdots \\ g_{n1} & \cdots & g_{nn} \end{bmatrix}$. We know that $\det(\lambda I + \vartheta) = \prod_{i=1}^{n}(\lambda - \lambda_i)$,

then we can replace the determinants by the product of eigenvalues, so

$$\begin{aligned} &\det(\lambda I - \vartheta) = \lambda^n && \lambda^n \\ &-\lambda^{n-1}\sum_{i=1}^{n} g_{ii} && -\lambda^{n-1}\sum_{i=1}^{n}\lambda i \\ &+\lambda^{n-2}\sum_{i=1}^{n}\sum_{j<i}\begin{vmatrix} g_{ii} & g_{ij} \\ g_{ji} & g_{jj} \end{vmatrix} && = \quad +\lambda^{n-2}\sum_{i_1<i_2<\ldots<i_{n-2}}\lambda_{i_1}\lambda_{i_2}\ldots\lambda_{i_{n-2}} \\ &-\lambda^{n-3}\sum_{i=1}^{n}\sum_{i<j}\sum_{i<j<k}\begin{vmatrix} g_{ii} & g_{ij} & g_{ik} \\ g_{ji} & g_{jj} & g_{jk} \\ g_{ki} & g_{kj} & g_{kk} \end{vmatrix} && -\lambda^{n-3}\sum_{i_1<i_2<\ldots<i_{n-3}}\lambda_{i_1}\lambda_{i_2}\ldots\lambda_{i_{n-3}} \\ &\vdots && \vdots \\ &\pm|\vartheta| && \pm\lambda_1\lambda_2\ldots\lambda_n \end{aligned} \tag{20}$$

then, if we apply the next equality

$$\det\left(\sigma_w[A_i, A_j]^{*i}\right) = \det\left(w_i A_i^{*i} + (1-w_i)A_j^{*i}\right) = \det\left(A_j^{*i}\right)\det\left(\gamma I + \vartheta^{*i}\right) \tag{21}$$

where, $i = \{1, 2, ..., n\}$ and, $*i$ is the $i-th$ cofactor of $\det(\cdot)$. We can visualize the eigenvalues by cofactors, thus if all the cofactors of $\det\left(A_i A_j^{-1}\right)^{*i} > 0$, so $\det\left(\gamma I + \vartheta^{*i}\right) > 0$ therefore *sign* $\left(\det\left(A_j^{*i}\right)\right)$ is equal to *sign* $\left(\det\left(\sigma_w[A_i, A_j]^{*i}\right)\right)$. Finally as A_j is Hurwitz, then the pencil $\left(\sigma_w[A_i, A_j]^{*i}\right)$ is Hurwitz.

Remark 2 *As well as we seen in remark (1), in this proof the possibility of combination of cases to analyse, is augmented due to the quantity of implied variables.*

Corollary 7 *If A_i and A_j are two Hurwitz matrices then A_z is Hurwitz $\forall w_k \in [0, 1]$*

Proof: By equation (2), we have a displacement of eigenvalues governed by the values of w. We have a movement from the eigenvalues of A_i to them of A_j and viceversa. For example and with loss of generality, if w has a variation

from $w_i = 1$, $w_j = 0$ to $w_i = 0$, $w_j = 1$, the system A_z has a displacement form the eigenvalues of A_i to them of A_j, therefore A_z keep up the eigenvalues with negative real parts. Hence A_z is also Hurwitz $\forall w_k \in [0,1]$.

Theorem 8 *A necessary and sufficient condition for the existence of a CQLF for the systems A_i and A_j, is that A_z be Hurwitz $\forall w_k \in [0,1]$.*

Proof:

($\Rightarrow$) Let a $V(x)$ be a CQLF for A_i and A_j then, there exists a matrix $P = P^T > 0$ such that:

$$x^T \left[\left(\frac{w_i A_i + w_j A_j}{w_i + w_j} \right)^T P + P \left(\frac{w_i A_i + w_j A_j}{w_i + w_j} \right) \right] x < 0$$

for all $x \in \mathbb{R}^n$ and $w \in [0,1]$. However we know from definition 3, that a FSA is a change of the dynamic of Σ_{A_i} to the dynamic of Σ_{A_j} and, in addition if a CQLF exists, the stability conditions in sense Lyapunov are preserved in the switching action. So, for the switching sequences where Σ_{A_i} and Σ_{A_j} are involved, the stability conditions are preserved. Therefore we can say that Σ_{A_z}, is stable for all switching sequence with Σ_{A_i} and Σ_{A_j}, as components.

($\Leftarrow$) Let A_z be not Hurwitz then, all FSA carries the Σ_{A_z} to a unstable dynamique. So that, either A_i or A_j or both are unstables then, a P matrix where:

$$\left(\frac{w_i A_i + w_j A_j}{w_i + w_j} \right)^T P + P \left(\frac{w_i A_i + w_j A_j}{w_i + w_j} \right) < 0$$

$P = P^T > 0$ does not exists, hence Σ_{A_z} is unstable, therefore $V_z(x)$ does not exists, so a common P does not exists and, finally a CQLF for the dual switching system where A_i and A_j are involved, does not exists.

4 Conclusions

In this paper, we give necessary and sufficient conditions for assure a convex solution in order to find a CQLF in a set A (see equation (1)). This convex solution is established for Hurwitz matrices and eigenvalues. A stable solution is guarantee for a switched dynamic who belong to an interval established by Hurwitz matrices in its extremes.

The above arguments formally show, that the domain into convex solutions where we can find a quadratic Lyapunov function, has a derivative negative definite along the trajectories of the system, $\dot{V}(x) < 0$.

We have seen that a switched system might become stable for fuzzy switching signals. One way to address this problem is to make sure that the interval between fuzzy switching actions is convex.

References

[1] S. Bialas. A necessary and sufficient condition for the stability of convex combinations of stable polynomials and matrices. Bull. Polish Acad. Sci. Tech. Sci., vol.33, pp.473-480, 1995.

[2] M. Branicky. Multiple Lyapunov functions and other analysis tools for switched and hybrid systems.. IEEE. TAC. vol. 43, no. 4, pp. 475-482, April 1998.

[3] N. Cohen, I. Lewkowicz. A Necessary and Sufficient Criterion for the Stability of a Convex Set of Matrices. IEEE. TAC. vol.38, No. 4, 1993.

[4] R. DeCarlo, M. Branicky, et.al. Perspectives and results on the stability and stabilizability of Hybrid Systems. Proceeding of the IEEE. vol. 88, no. 7, pp. 1069-1082. July 2000.

[5] S. Farinwata, et.al, Editors. Fuzzy Control, Synthesis and Analysis. pp. 239-252. John Wiley & Sons, 2002.

[6] F. R. Gantmacher. The theory of matrices. vol. I and II, Chesla Publ. Co. 1977.

[7] IEEE. Transaction on Automatic Control, Special Issue on Hybrid Control Systems, vol. 43, April 1998.

[8] S. H. Lee, T. H Kim, J. T. Lim. A new stability analysis of switched systems. Automatica, 36, 917-922, 2002.

[9] D. Liberzon, A. S. Morse. Basic problems in stability and design of switched systems. IEEE. Control Systems Magazine. 19(5); pp. 59-70.

[10] Proceeding of the IEEE, Special Issue on Hybrid Systems: Theory and Applications, vol. 88. no. 7, 2000.

[11] T. Rockafellar. Convex Analysis. Princeton Landmarks, 1970.

[12] E. Rubio, A. Titli. Stability analysis in fuzzy switched systems. (submitted).

[13] R. Shorten, Narendra K, Mason O. A result on common quadratic Lyapunov functions. IEEE. TAC. vol. 48, no. 1, January pp. 110-113, 2003.

[14] R. Shorten, Narendra K. Necessary and sufficient conditions for the existence of a common quadratic Lyapunov function for a finite number of stable second order linear time-invariant systems. Int. J. Adapt. Control Signal Process, 16:709-728, 2002.

[15] R. Shorten, K. Narendra. A sufficient condition for the existence of a common Lyapunov function for two second order linear systems. Proceedings of the 36th. CDC, San Diego Calif. USA. December pp. 3521-3522, 1997.

[16] K. Tanaka, H. O. Wang. Fuzzy Control Systems Design and Analysis: A Linear Matrix Inequality Approach. John Wiley \& Sons Inc, 2001.

[17] M. Vidyasagar. Nonlinear Systems Analysis. Prentice-Hall, 1993.

Integral high gain observers: Theory and experimental evaluation

Rubén Garrido-Moctezuma[1], Ernesto Flores-García[1], Rafael Martínez-Guerra[1]

[1]CINVESTAV-IPN, Departamento de Control Automático, A.P. 14-740,
México, D.F. 07360, México
{garrido, eflores, rguerra}@ctrl.cinvestav.mx

Abstract. Integral High Gain Observer evaluation is performed for the simultaneous estimation of the state variables and of an uncertain term for a class of nonlinear systems. It is assumed that the measured output presents additive noise. The proposed observer contains an integral-type contribution of the measured output, which provides robustness against noise. The observer is applied to a 1 degree of freedom (DOF) robot. Experimental results in real time confirm the validity and good performance of the observer.

1 Introduction

Thanks to the advances in the automatic control field, it has been possible to develop methods and techniques to estimate the unknown variables governing the behaviour of a dynamical system. A technique commonly employed for this purpose is based on the reconstruction of the required variables by means of an *observer*, which is an auxiliary dynamical system that attempts to *estimate* and *reconstruct* the state variables by using the model of the original system, its input and output. The first developments in estimation theory were obtained by Wiener [8] about temporal series analysis, based on statistical assumptions related to the nature of the noise present in the signal to be filtered and developing the so-called *Wiener filter*. A major thrust in the development of observers came from the introduction of state-space methods by R.E. *Kalman* in 1960 [7]. In 1963 D.G. *Luenberger* [7] showed that for observable linear systems, an observer can be designed with the property that the observation error, generated by the difference between the original system measured output and the output estimated by the observer, goes exponentially to zero [2], [7]. In order to implement a Luenberger observer it is necessary to know exactly the structure and the parameters of the system under observation. In real-life systems, it is unlikely to exactly known the above information then precluding the use of Luenberger observers at least in its theoretical form. Several modifications to the original Luenberger observer have been proposed to cope with the above problem [1], [2], [3], [4].

The integral observers proposed in [1] are particularly interesting for several reasons. They are high gain designs in which the observer gain is tuned using a single parameter. Moreover, their structure allows estimating certain classes of uncertainties in the same way as it is showing in [6].

Another interesting feature of integral observers is the fact that they are driven by the integral of the system output whereas in standard observers the observer is drivendirectly by the system output. As a consequence, measurement noise is filtered by the integral action then allowing using high gain in practice. Unfortunately, the integral-type observers have been evaluated using only numerical simulations and no study of its implementation in real-time has not been conducted.

The aim of this work is to experimentally evaluate a class of Integral-type high gain observers employing an electromechanical laboratory prototype consisting of a 1 DOF robot driven by a Direct Current (DC) motor. Two cases are considered. Firstly, it is assumed that all the prototype parameters are known, secondly, an uncertainty on the robot model is considered. The paper is presented as follows. Section 2 is devoted to the integral-type high-gain observers. Application of the observer presented in Section 2 to a 1 DOF robot is shown in Section 3. Section 5 presents the experimental results for the two cases discussed above, namely, design of the observer with and without uncertainties. The paper ends with some concluding remarks.

2 Integral High Gain Observers

The system studied in next section belongs to a class of partially known nonlinear systems described by

$$\begin{aligned} \dot{X} &= \mathrm{L}(X,U)+\mathrm{J}(X) \\ Y &= CX+d \end{aligned} \tag{1}$$

where $X \in \mathbb{U}^n$ is the state vector, $U \in \mathbb{U}^m$ is the control input vector, $\mathrm{J}(\cdot)$ is a nonlinear partially known vector, $\mathrm{L}(\cdot)$ is a linear vector of its arguments, $d \in \mathbb{U}$ is a bounded measurement additive noise and $Y \in \mathbb{R}$ is the system output.

Now, consider the following assumptions
A1. $d \in \mathbb{U}$ represents an external bounded additive noise

$$|d| \leq \alpha, \qquad 0 < \alpha < \infty \ .$$

A2. The system given by (1) is locally uniformly observable [5].

The following representation of (1) is proposed [1]

$$\begin{aligned} \dot{X}_0 &= CX+d \\ \dot{X} &= \mathrm{L}(X,U)+\mathrm{J}(X) \\ \dot{\mathrm{J}} &= \Theta(X) \\ Y_0 &= X_0 \end{aligned} \tag{2}$$

A key observation about (2) is the fact that the uncertain term J is considered as a new state and $\Theta(X)$ is a nonlinear unknown function that describes the dynamics of J, and satisfies the following assumption

A3. The dynamics $\Theta(X)$ of the uncertain term J is bounded, therefore

$$\|\Theta(X)\| \le \beta, \qquad \beta > 0 .$$

Another important remark is that another state x_0 is added by integrating the system output.
Then, the following dynamic system is an integral observer for (2) [1]

$$\begin{aligned} \dot{\hat{X}}_0 &= C\hat{X} + k_1\left(Y_0 - \hat{Y}_0\right) \\ \dot{\hat{X}} &= \mathrm{L}\left(\hat{X}, U\right) + \mathrm{J}\left(\hat{X}\right) + k_2\left(Y_0 - \hat{Y}_0\right) \\ \dot{\hat{\mathrm{J}}} &= k_3\left(Y_0 - \hat{Y}_0\right) \\ \hat{Y}_0 &= \hat{X}_0 \end{aligned} \tag{3}$$

The observer gain matrix $K = [k_1, k_2, k_3]^T$ is given as

$$K = S_\theta^{-1} C^T . \tag{4}$$

S_θ is a symmetric positive definite solution of the algebraic Lyapunov equation [5]

$$S_\theta\left(A + \frac{\theta}{2} I\right) + \left(A^T + \frac{\theta}{2} I\right) S_\theta = C^T C . \tag{5}$$

Where

$$C = [1 \quad 0 \cdots\cdots 0] \text{ and } A_{ij} = \begin{cases} 1, & \text{if } \ j = i+1 \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

For some parameter $\theta > 0$, determining the desired convergence speed. Coefficients of S_θ are given by $(S_\theta)_{ij} = \frac{s_{ij}}{\theta^{i+j-1}}$, where s_{ij} are entries of a symmetric positive definite matrix not depending on θ.

3 System Under Study

Let the single input-single output nonlinear system

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -k \sin(x_1) - a x_2 + bu \\ y = x_1 + d \end{cases} \tag{7}$$

corresponding to a 1 DOF robot model affected by additive measurement noise d.

3.1 Totally Known System

According to the results presented in Section 2, a new representation of the system (7) is proposed. Set $x_0(t)=\int_0^t y(\tau)d\tau$ so that $\dot{x}_0(t)=y(t)=x_1(t)+d$. Then, the following augmented system is obtained

$$\begin{cases}\dot{x}_0 = x_1 + d \\ \dot{x}_1 = x_2 \\ \dot{x}_2 = -k\sin(x_1) - ax_2 + bu \\ y_0 = x_0\end{cases} \tag{8}$$

(8) can be written in matrix form as follows

$$\begin{aligned}\begin{bmatrix}\dot{x}_0\\ \dot{x}_1\\ \dot{x}_2\end{bmatrix} &= \begin{bmatrix}0 & 1 & 0\\ 0 & 0 & 1\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}x_0\\ x_1\\ x_2\end{bmatrix} + \begin{bmatrix}0\\ 0\\ b\end{bmatrix}u + \begin{bmatrix}0\\ 0\\ -(k\sin(x_1)+ax_2)\end{bmatrix} + \begin{bmatrix}1\\ 0\\ 0\end{bmatrix}d \\ y_0 &= \begin{bmatrix}1 & 0 & 0\end{bmatrix}\begin{bmatrix}x_0\\ x_1\\ x_2\end{bmatrix}\end{aligned} \tag{9}$$

Or in a compact form

$$\begin{cases}\dot{x} = Ax + Bu + \Phi(x) + Ed \\ y_0 = Cx\end{cases} \tag{10}$$

Where

$$A=\begin{bmatrix}0 & 1 & 0\\ 0 & 0 & 1\\ 0 & 0 & 0\end{bmatrix};\ B=\begin{bmatrix}0\\ 0\\ b\end{bmatrix};\ \Phi(x)=\begin{bmatrix}0\\ 0\\ -(k\sin(x_1)+ax_2)\end{bmatrix};\ E=\begin{bmatrix}1\\ 0\\ 0\end{bmatrix};\ C=\begin{bmatrix}1 & 0 & 0\end{bmatrix}.$$

Notice that $E=C^T$ and the matrix A and C of the new augmented system (8) retain the properties required in (6), as well as system (10) is observable.
An integral high-gain observer design for (10) results in

$$\begin{cases}\dot{\hat{x}} = A\hat{x} + Bu + \Phi(\hat{x}) + K(y_0 - \hat{y}_0) \\ \hat{y}_0 = C\hat{x}\end{cases} \tag{11}$$

Combining (10) and (11) the estimation error dynamics $e=x-\hat{x}$ is given as

$$\begin{aligned}\dot{e} = \dot{x} - \dot{\hat{x}} &= (A-KC)e + \Phi(x) - \Phi(\hat{x}) + Ed \\ \dot{e} &= A_0 e + \Phi + C^T d\end{aligned} \tag{12}$$

With K defined as in (4), $A_\theta = A - KC$ and $\Phi = \Phi(x) - \Phi(\hat{x})$.

From (12) can be seen that the noise is not amplified by the observer gain. S_θ , S_θ^{-1} and A_θ matrices for (11) and (12) are given by

$$S_\theta = \begin{bmatrix} \frac{1}{\theta} & -\frac{1}{\theta^2} & \frac{1}{\theta^3} \\ -\frac{1}{\theta^2} & \frac{2}{\theta^3} & -\frac{3}{\theta^4} \\ \frac{1}{\theta^3} & -\frac{3}{\theta^4} & \frac{6}{\theta^5} \end{bmatrix}; \quad S_\theta^{-1} = \begin{bmatrix} 3\theta & 3\theta^2 & \theta^3 \\ 3\theta^2 & 5\theta^3 & 2\theta^4 \\ \theta^3 & 2\theta^4 & \theta^5 \end{bmatrix}; \quad A_\theta = \begin{bmatrix} -3\theta & 1 & 0 \\ -3\theta^2 & 0 & 1 \\ -\theta^3 & 0 & 0 \end{bmatrix}.$$

Observer (11) for system (10) has the following structure

$$\begin{cases} \dot{\hat{x}}_0 = \hat{x}_1 + 3\theta(x_0 - \hat{x}_0) \\ \dot{\hat{x}}_1 = \hat{x}_2 + 3\theta^2(x_0 - \hat{x}_0) \\ \dot{\hat{x}}_2 = -k\sin(\hat{x}_1) - a\hat{x}_2 + bu + \theta^3(x_0 - \hat{x}_0) \end{cases} \tag{13}$$

3.2 Uncertainty Estimation

Based on (7), several terms were grouped in a single term and considered as uncertain, i.e. $-(k\sin(x_1) + ax_2) = \varphi(x,u)$ is the uncertainty to be estimated. Then, system (8) can be rewritten as

$$\begin{cases} \dot{x}_0 = x_1 + d \\ \dot{x}_1 = x_2 \\ \dot{x}_2 = \varphi(x,u) + bu \\ y_0 = x_0 \end{cases} \tag{14}$$

Incorporating the uncertainty to the system as a new state the following augmented system is obtained

$$\begin{cases} \dot{x}_0 = x_1 + d \\ \dot{x}_1 = x_2 \\ \dot{x}_2 = x_3 + bu \\ \dot{x}_3 = \Theta(x,u) \\ y_0 = x_0 \end{cases} \tag{15}$$

Where $\Theta(x,u)$ is an unknown nonlinear function describing the dynamics of $x_3 = \varphi(x,u)$. The matrix form of (15) is given by

$$\begin{bmatrix}\dot{x}_0\\ \dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\end{bmatrix}=\begin{bmatrix}0&1&0&0\\0&0&1&0\\0&0&0&1\\0&0&0&0\end{bmatrix}\begin{bmatrix}x_0\\ x_1\\ x_2\\ x_3\end{bmatrix}+\begin{bmatrix}0\\0\\b\\0\end{bmatrix}u+\begin{bmatrix}0\\0\\0\\\Theta(x)\end{bmatrix}+\begin{bmatrix}1\\0\\0\\0\end{bmatrix}d$$
$$y_0=\begin{bmatrix}1&0&0&0\end{bmatrix}\begin{bmatrix}x_0&x_1&x_2&x_3\end{bmatrix}^T \tag{16}$$

Which in a compact form is represented as

$$\begin{cases}\dot{x}=Ax+Bu+\Psi(x)+Ed\\ y_0=Cx\end{cases} \tag{17}$$

Where

$$A=\begin{bmatrix}0&1&0&0\\0&0&1&0\\0&0&0&1\\0&0&0&0\end{bmatrix};\quad B=\begin{bmatrix}0\\0\\b\\0\end{bmatrix};\quad \Psi(x)=\begin{bmatrix}0\\0\\0\\\Theta(x)\end{bmatrix};\quad E=\begin{bmatrix}1\\0\\0\\0\end{bmatrix};\quad C=\begin{bmatrix}1&0&0&0\end{bmatrix}.$$

Again $E=C^T$ and the matrix A and C of the new augmented system (15) conserve the properties required in (6). An integral high-gain observer for (17) is given by

$$\begin{cases}\dot{\hat{x}}=A\hat{x}+Bu+K(y_0-\hat{y}_0)\\ \hat{y}_0=C\hat{x}\end{cases} \tag{18}$$

Using (17) and (18), the estimation error dynamics $e=x-\hat{x}$ is given as follows

$$\begin{aligned}\dot{e}=\dot{x}-\dot{\hat{x}}&=(A-KC)e+\Psi(x)+Ed\\ \dot{e}&=A_\theta e+\Psi(x)+C^T d\end{aligned} \tag{19}$$

With K defined as in (4). Matrices S_θ, S_θ^{-1} and A_θ for (18) and (19) are given by

$$S_\theta=\begin{bmatrix}\frac{1}{\theta}&-\frac{1}{\theta^2}&\frac{1}{\theta^3}&-\frac{1}{\theta^4}\\ -\frac{1}{\theta^2}&\frac{2}{\theta^3}&-\frac{3}{\theta^4}&\frac{4}{\theta^5}\\ \frac{1}{\theta^3}&-\frac{3}{\theta^4}&\frac{6}{\theta^5}&-\frac{10}{\theta^6}\\ -\frac{1}{\theta^4}&\frac{4}{\theta^5}&-\frac{10}{\theta^6}&\frac{20}{\theta^7}\end{bmatrix}\quad S_\theta^{-1}=\begin{bmatrix}4\theta&6\theta^2&4\theta^3&\theta^4\\ 6\theta^2&14\theta^3&11\theta^4&3\theta^5\\ 4\theta^3&11\theta^4&10\theta^5&3\theta^6\\ \theta^4&3\theta^5&3\theta^6&\theta^7\end{bmatrix}\quad A_0=\begin{bmatrix}-4\theta&1&0&0\\ -6\theta^2&0&1&0\\ -4\theta^3&0&0&1\\ -\theta^4&0&0&0\end{bmatrix}$$

The observer (18) for the system (17) has the following structure

$$\begin{cases} \dot{\hat{x}}_0 = \hat{x}_1 + 4\theta(x_0 - \hat{x}_0) \\ \dot{\hat{x}}_1 = \hat{x}_2 + 6\theta^2(x_0 - \hat{x}_0) \\ \dot{\hat{x}}_2 = \hat{x}_3 + bu + 4\theta^3(x_0 - \hat{x}_0) \\ \dot{\hat{x}}_3 = \theta^4(x_0 - \hat{x}_0) \end{cases} \tag{20}$$

4 Stability Analysis

Let the following error dynamics

$$\dot{e} = A_\theta e + \Omega(X, U, d) \ . \tag{21}$$

Where

$$e = \begin{bmatrix} X_0 - \hat{X}_0 \\ X - \hat{X} \\ \mathsf{J} - \hat{\mathsf{J}} \end{bmatrix}; \quad \Omega = \begin{bmatrix} d \\ \Delta\mathsf{L} \\ \Theta(X) \end{bmatrix}; \quad A_\theta = \left(A - S_\theta^{-1} C^T C\right) .$$

And consider the following assumptions

A4. The function $\Delta\mathsf{L} = \mathsf{L}(X,U) - \mathsf{L}\left(\hat{X},U\right)$ is bounded, i.e.

$$\left\| \mathsf{L}(X,U) - \mathsf{L}\left(\hat{X},U\right) \right\| \le \delta, \quad \delta > 0 \ .$$

A5. There exist two positive constants $\gamma > 0$ and $\lambda > 0$ satisfying

$$\left\| \exp(A_\theta t) e \right\| \le \gamma \exp(-\lambda t) \|e\| \ .$$

A6. The vector function Ω is bounded, so that

$$\|\Omega\| \le \mu, \quad \mu > 0 \ .$$

Now, by solving (21), the following expression is obtained

$$e = \exp(A_\theta t) e(0) + \int_0^t \exp\left\{A_\theta (t-\tau)\right\} \Omega d\tau \ . \tag{22}$$

Considering assumptions *A1* to *A6* and taking norms for both sides of (22), the following inequality is obtained

$$\|e\| \le \gamma \exp(-\lambda t) \left[\|e(0)\| - \frac{\mu}{\lambda} \right] + \frac{\gamma\mu}{\lambda} \ . \tag{23}$$

Taking the limit when $t \to \infty$, the following bound on e is obtained

$$\|e\| \leq \frac{\eta\mu}{\lambda} . \tag{24}$$

Then, the estimation error remains bounded inside a ball of radius $\eta\mu/\lambda$, i.e. $e \in B_{\frac{\eta\mu}{\lambda}}(0)$.

5 Experimental Results

In order to perform the experiments, a 1 DOF robot was employed and it is shown in Fig. 1.



Fig. 1. Laboratory prototype employed in the experiments

A DC-motor drives the link through a belt and angular position is measured using an optical encoder with 2500 pulses per turn. The encoder is directly attached to the link. The motor is driven by a Copley Controls, model 413, power amplifier configured in torque mode. Data acquisition is performed using the MultiQ 3 card from Quanser Consulting with optical encoder inputs. These inputs multiply by 4 the encoder resolution, then, the number of pulses per turn is 10000 . This output was further scaled down by a factor of 10000 corresponding to one link turn. The card also has 12 bits Digital to Analog converters with an output voltage range of $\pm 5v$. All the programming including the observers and data acquisition was implemented using the MatLab-Simulink software running under the WinCon program from Quanser Consulting. The WinCon environment was used in the client and the server running on different computers. The server is installed in a Pentium based computer running at $200\,Mhz$. The client is allocated in another Pentium based computer running at $350\,Mhz$. Sampling rate was set to $5Khz$.

Since the 1 DOF robot may get unstable in open loop, a loop was closed around the robot using a relay programmed in the MatLab-Simulink environment. In this way,

the robot performed a stable limit cycle. The velocity estimated from the observers was compared with another velocity estimate obtained from position measurements through numerical differentiation. Parameters for model (7) are $a = 0.7$, $b = 28$ and $k = 12.04$.

First Case: Robot without Uncertainty.
In this case the nonlinear terms in (7) are assumed known. Observer gain was set to $\theta = 120$. Fig. 2 and Fig. 3 show position and velocity from the real system and from the observer. It is noted that measured and estimated position do not show noticeable differences. On the other hand, velocity estimates show some variation. The estimate obtained from the integral observer shows a smoother behaviour compared with the velocity estimated obtained from numerical differentiation of the position measurement, a technique of broad use in industrial servomotors.

Second Case: Robot with Uncertainty.
In this case observer gain was set to $\theta = 80$. Fig. 4 and Fig. 5 display the position and velocity estimates. Behaviour in both states is similar to the behaviour observed in Fig. 2 and Fig. 3. Fig. 6 show the real and estimated uncertainty, the later exhibiting a smoother behaviour. The uncertainty estimate does not match perfectly the real uncertainty, however, it should be pointed out that the so called real uncertainty is also an estimate since it was computed using estimated parameters, position measurements possibly contaminated with noise and velocity estimates obtained from position measurements through numerical differentiation. Moreover, it is worth remarking that the integral observer does not need any a priory information regarding the uncertainty structure and parameters. Note also that even with very high observer gains, measurement noise does not affect the estimates. In the particular case of uncertainty estimation, the corresponding gain was $\theta^4 = 40960000$.

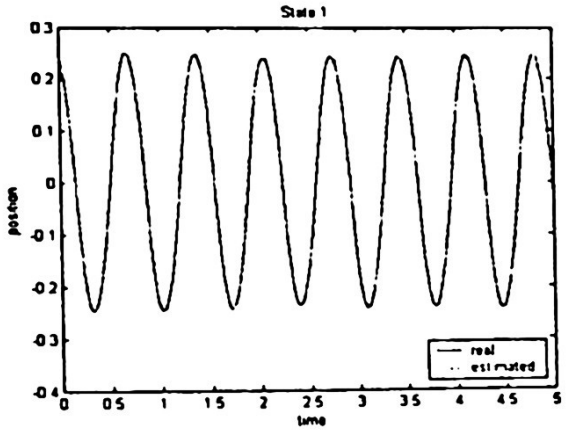


Fig. 2. Position

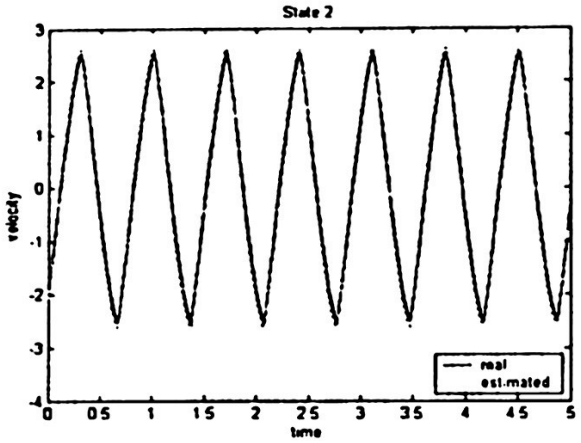


Fig. 3. Velocity

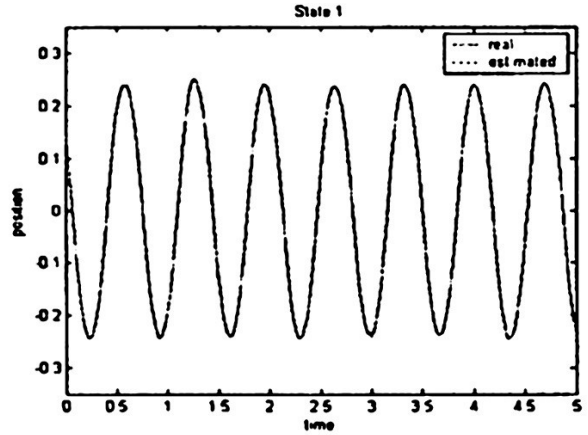


Fig. 4. Position

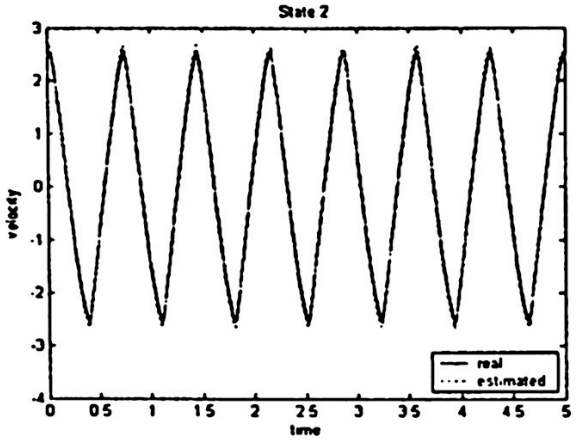


Fig. 5. Velocity

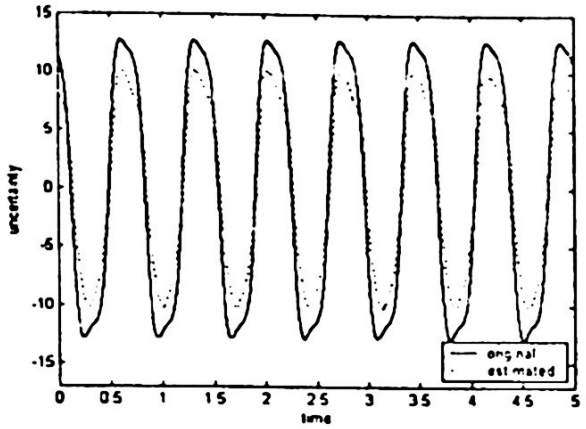


Fig. 6. Uncertainty

6 Conclusions

Few works regarding experimental evaluation of high gain observers have been published in the open literature. In this work and for the first time, an Integral-type high gain observer was experimentally evaluated. A feature of this observer is the fact that it is driven by the integral of the system output, an approach contrasting with the standard method in which the system output drives directly the observer. As a consequence, measurement noise is highly attenuated then allowing using very high observer gains. Performance of the integral observer was evaluated through experiment using a 1 DOF robot. Two cases were studied. In the first one all the robot parameters are assumed known. In the second case, the robot nonlinear term and viscous friction were supposed unknown.

In both cases the observer showed good performance in spite of the high gain employed.

Future work includes using the Integral observer in closed loop with a Proportional-Derivative control law applied to the same prototype.

References

[1]. Aguilar-López, R. Martínez-Guerra, and R. Maya-Yescas: "State Estimation for Partially Unknown Nonlinear Systems: a class of integral high gain observers". IEE Proc. Contr. Theory and Applications, 2003, Vol. 150, No. 3, pp.240-244.

[2]. R. Aguilar López: *Diseño y análisis de observadores robustos para monitoreo de incertidumbres en sistemas no lineales*. Ph.D. thesis, Cinvestav, México D.F., 2003.

[3]. S. Beale and B. Shafai: "Robust control system design with a proportional-integral observer". Int. J. Contr., 1989, Vol. 50, No. 1, pp. 97-111.

[4]. K.K. Busawon and P. Kabore: "On the design of integral and proportional-integral observers". Proc. of the American Contr. Conf., Chicago, Illinois, June 2000, pp. 3725-3729.
[5]. J.P. Gauthier, H. Hammouri and S. Othnman: "A simple observer for nonlinear systems, application to bioreactors". IEEE Trans. on Aut. Contr., 1992, Vol. 37, pp. 875-880.
[6]. G.C. Goodwin, S.F. Graebe, M.E. Salgado: *Control Systems Design.* Prentice-Hall, Upper Saddle River, New Jersey, 2001.
[7]. John O'Reilly: *Observers for Linear Systems.* Academic Press, New York, 1983.
[8]. N. Wiener: "The extrapolation, interpolation and smoothing of stationary time series". Wiley, N.Y., 1949.

Sistema de control difuso usando IVs

[1]Romeo Urbieta Parrazales, [2]Luis Pastor Sánchez Fernández, [3]Diego Ramírez Martínez

[1.2.3] Centro de Investigación en Computación
Instituto Politécnico Nacional
Av. Juan de Dios Bátiz s/n C. P. 07738 México D. F., Tel. 57296000 Ext. 56579.
[1]rurbieta@cic.ipn.mx, [2]lsanchez@cic.ipn.mx, [3]diegoramirez46@hotmail.com

Resumen. Este trabajo presenta la implementación de un sistema de control difuso aplicado a la variable de velocidad de un motor de corriente continua usando *Instrumentos Virtuales* para controlar y monitorear dicha variable. Los parámetros del controlador difuso están basados en pruebas realizadas. El hardware del sistema consiste en: una Computadora Personal Pentium tres, una interfaz de conversión analógica a digital y digital analógica de alta resolución instalada en un bus PCI, y el proceso en si. El software fue de dos clases: básico y de aplicación. El software básico fue Windows noventa y ocho como plataforma de trabajo y como software de aplicación, *Instrumentos Virtuales*. Se presentan diferentes esquemas de desarrollo del control difuso así como resultados del mismo en instrumento virtual.

Palabra Clave: Instrumentos Virtuales, Control Difuso.

1. Introducción

Muchos sistemas reales son difíciles o imposible de modelar matemáticamente, lo cual es requerido para el diseño de muchos algoritmos de control tradicional. Otros procesos que pueden o no pueden ser modelados matemáticamente son demasiado complejos o no lineales para ser controlados con estrategias tradicionales. Para esto, un controlador difuso puede ser implementado rápida y fácilmente. La lógica difusa es un método de decisión basado en reglas y algunas de sus aplicaciones es en sistemas expertos y en control de procesos. Todo proceso que una persona puede controlar manualmente de acuerdo a su experiencia, es susceptible de ser controlado con control difuso ya que las reglas lingüísticas que una persona experta puede describir de una forma intuitiva y general puede ser directamente trasladada a una base de reglas[1]. La base de la lógica difusa es la teoría de los conjuntos difusos que fue desarrollada por Lotfi Zadeh en la década de los 1960s.[2]

El módulo de construcción primaria de los sistemas de lógica difusa es la *Variable Lingüística.* Cuando se crea una variable lingüística, primero se debe determinar cuantos términos definen a la variable lingüística. En muchas aplicaciones, entre tres y siete términos son suficientes para definir una variable lingüística. No tiene sentido usar menos de tres términos, porque muchos conceptos lingüísticos tienen por lo menos dos términos extremos y un término medio entre ellos. Por otro lado, los sistemas lingüísticos que usan más de siete términos son difíciles de manejar porque los humanos usan su memoria de corto término para interpretar cantidades técnicas y

nuestra memoria de corto término solo puede procesar hasta siete símbolos simultáneamente.

Las variables lingüísticas usualmente tienen un número impar de términos porque ellos son definidos simétricamente e incluyen un término medio entre los extremos. Como un punto de arranque, usar por lo menos tres o cinco términos en las variables de entrada y cinco o siete términos en las variables de salida.

Si no hay información disponible a priori, empezar con términos igualmente espaciados dentro del rango de la variable asociada, con cada término completamente traslapado con sus términos adyacentes. Se debe cubrir la región estable deseada del sistema con más términos lingüísticos que tengan un pequeño intervalo de influencia en lugar de tratar de cubrir las regiones de los bordes que tienen solo unos pocos términos lingüísticos con un gran intervalo de influencia. Una distribución de términos hecha de esta forma, hace al controlador más sensible dentro de la región de estado estable del sistema.

Los efectos de disturbios en los valores de entrada tales como ruidos, deben ser tomados en cuenta. No crear funciones de pertenencia con un intervalo de influencia que sea más pequeño que la amplitud de la señal de ruido.

Para construir una base de reglas, definir una regla para cada combinación de términos de las variables de entrada usadas en la parte IF de la regla. Entonces seleccionar el término más conveniente de la variable de salida para especificar la parte THEN de cada regla. Para un sistema difuso, el número total de reglas posibles es $N= p^m$, donde: p = número de términos para cada variable de entrada, m = número de variables de entrada y N = número de reglas. Se puede cambiar el comportamiento de un controlador para una cierta situación de entrada, modificando una regla particular. Modificar una regla tiene mayor influencia en la característica del controlador que las modificaciones a las funciones de pertenencia. Un controlador difuso tiene una estructura estática y determinística.

2. Instrumentos Virtuales (IVs).

El lenguaje de programación gráfica LabVIEW es conocido como una buena herramienta para la construcción de sistemas adquisición de datos, de instrumentación y de control. Después de la primera versión, aparecida para Macintosh en 1986 y para Windows y Unix en 1992, en las sucesivas ha ido ofreciendo a los usuarios una mayor facilidad y rapidez en la programación, disminuyendo radicalmente los tiempos para la elaboración de soluciones.

Los sistemas operativos de tiempo real son muy caros y exigen especialistas para trabajar con los mismos. De aquí el interés de la solución (software y hardware) para control en tiempo real que ofrece ahora National Instruments.

Con una relación precio/prestaciones realmente bueno, permite ampliar un sistema basado en PC Windows con un sistema de control en tiempo real, que se integra con

la instrumentación, la adquisición de datos, accionamientos, etc. creando una solución completa [3].

3. Implementación del Controlador con Instrumentos Virtuales.

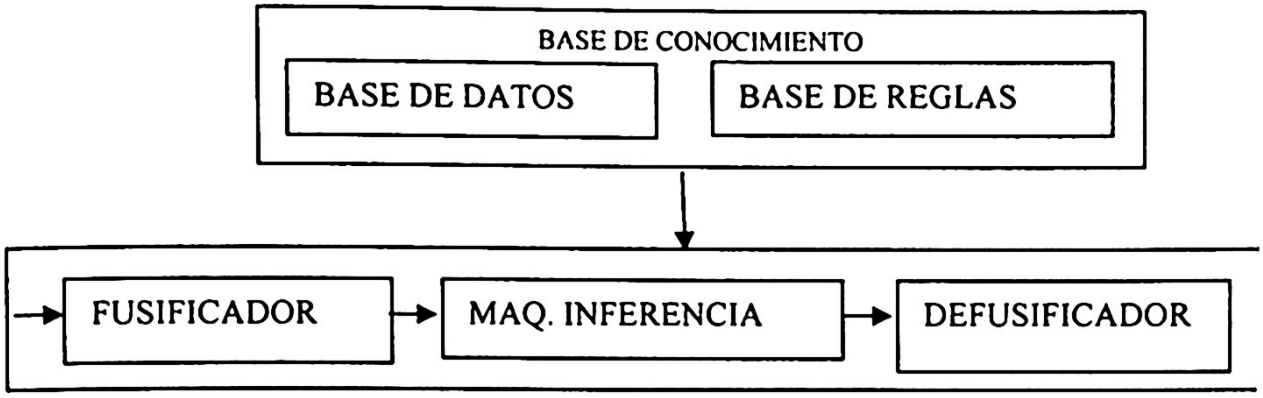


Fig. 1. Diagrama de bloques de un CLD.

El controlador difuso es un sistema que consta de cuatro componentes básicos: fusificador, máquina de inferencia, defusificador y la base de conocimiento. Se observa que el primer paso es convertir las señales de los sensores a variables lingüísticas (fusificación). En seguida se ejecuta el paso de Inferencia difusa para derivar conclusiones de la base de reglas como valores lingüísticos y por último este valor lingüístico se convierte a un valor real en el paso de defusificación que es la salida de control analógica.

La inferencia difusa consiste de dos componentes:

Agregación – es la evaluación de la parte IF de cada regla.

Composición – es la evaluación de la parte THEN de cada regla.

Para implementar el Controlador Difuso con instrumentos virtuales se usa el kit de lógica difusa que tiene National Instruments. Este kit contiene un editor de conjuntos difusos y un editor de base de reglas. Con el editor de conjuntos difusos se crea las variables lingüísticas, los términos lingüísticos y se establecen los rangos, con el editor de la base de reglas se crean las reglas, se asignan los factores de peso, se selecciona el método de defusificación y el valor tomado por la variable de salida cuando ninguna regla está activa.

Estos parámetros se almacenan en un archivo con extensión xx.fc. En seguida se crea el programa gráfico orientado a flujo de datos para realizar los cálculos necesarios en cada paso y para recuperar los parámetros contenidos en el archivo .fc. [6],[7]

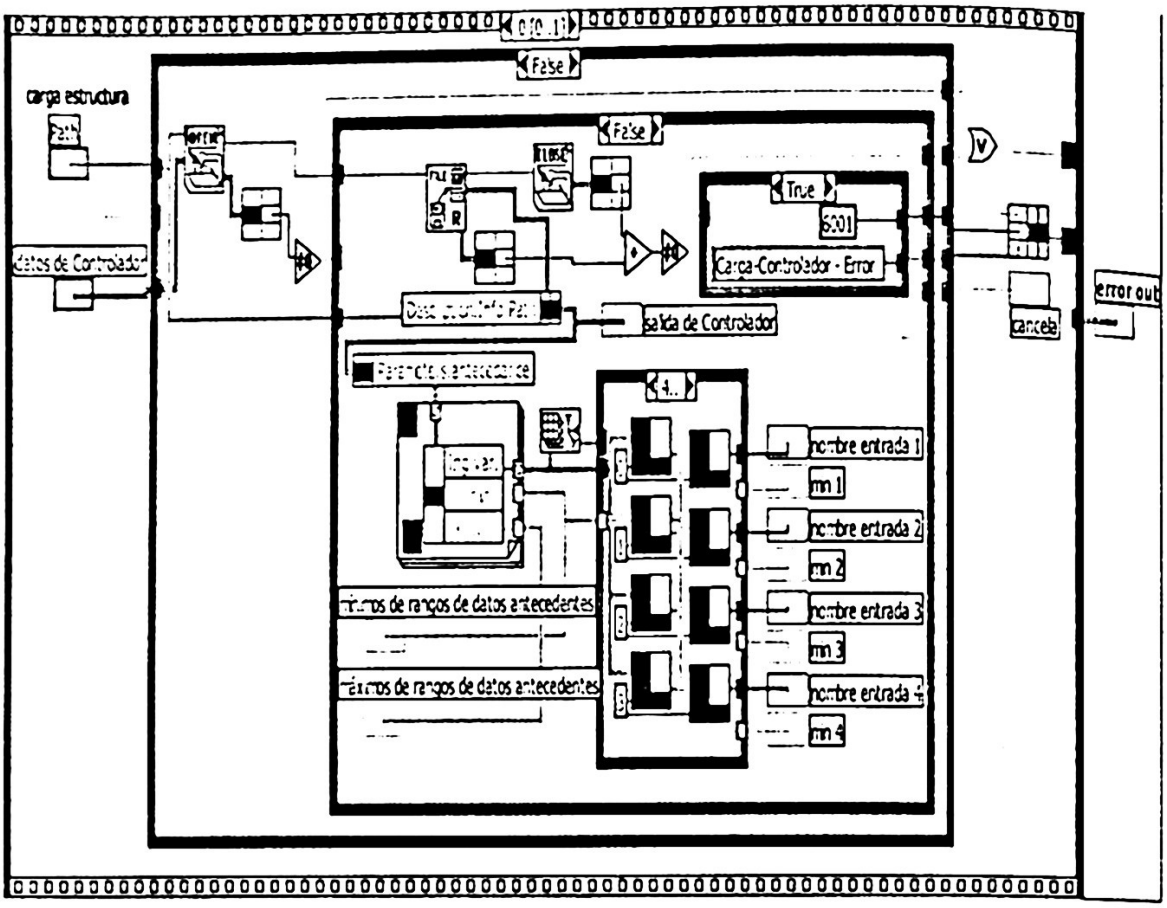


Fig. 2. Carga de parámetros del controlador

Para este sistema se crearon dos variables lingüísticas antecedentes (error y derivada del error) con tres funciones de pertenencia triangulares cada una y una variable de salida también con tres términos lingüísticos.

Se creó una base de 9 reglas con un factor de peso de 1.0 para todas. Se usó el método de inferencia max-min de Mamdani y el centro de máximos como defusificador. Estos parámetros se cargan al controlador usando el IV mostrado en la figura 2. La adquisición de datos se hizo con el IV de la figura 3.

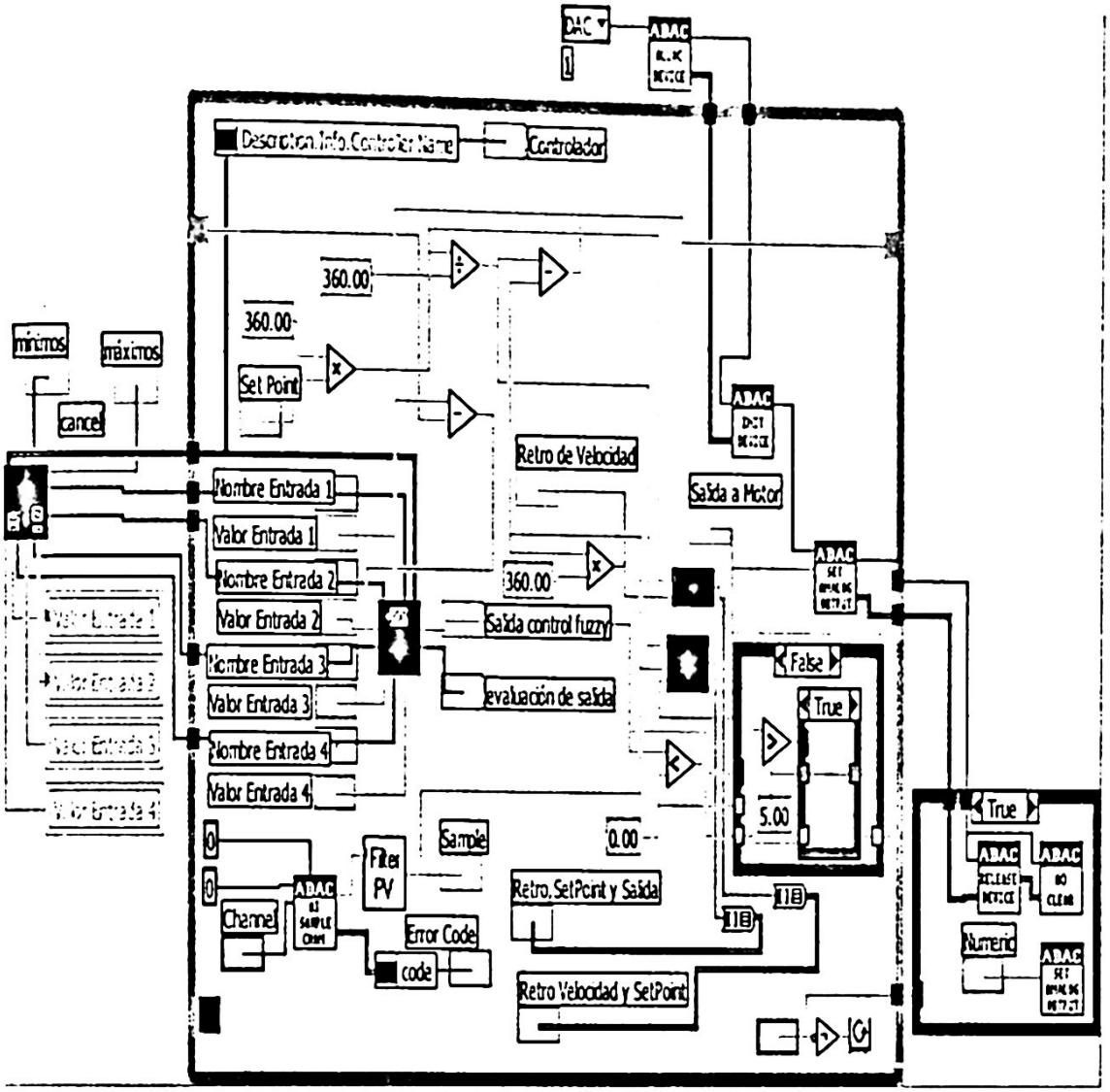


Fig. 3. Adquisición de datos

Por lo que respecta a la función Control Difuso, que es propiamente el algoritmo de control difuso, tenemos los tres pasos del controlador; la fusificación, la inferencia difusa y el defusificador.

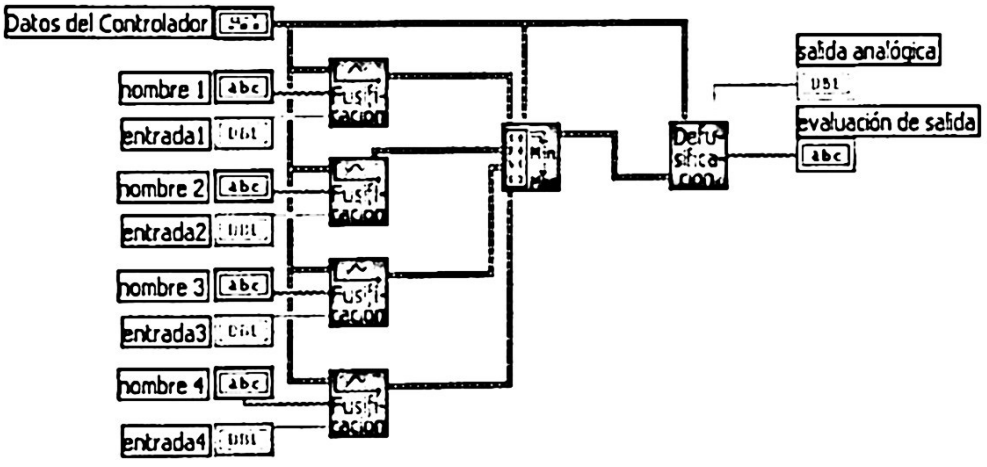


Fig. 4. Arquitectura del instrumento virtual del Controlador Difuso.

Como entrada 1, se tiene el valor del error obtenido con la diferencia entre el "Set point" y el valor de 0 a 5 voltios del sensor de velocidad, en la entrada analógica 1 de la tarjeta de adquisición de datos y la derivada del error en la entrada 2. Los nombres de cada entrada son las variables lingüísticas que vienen del archivo xx.fc (en nuestro caso es el Controlmotor1.fc) Los restantes datos del controlador como nombres de términos, rangos, tipos, valores de los puntos a,b,c,d normalizados de cada término de entrada y de salida, los datos de la base de reglas, del método de defusificación y de inferencia difusa usada, igualmente vienen del archivo .fc [6],[7].

El IV que realiza la fusificación está en la figura 3, y se usa una subIV defusificadora para cada entrada.

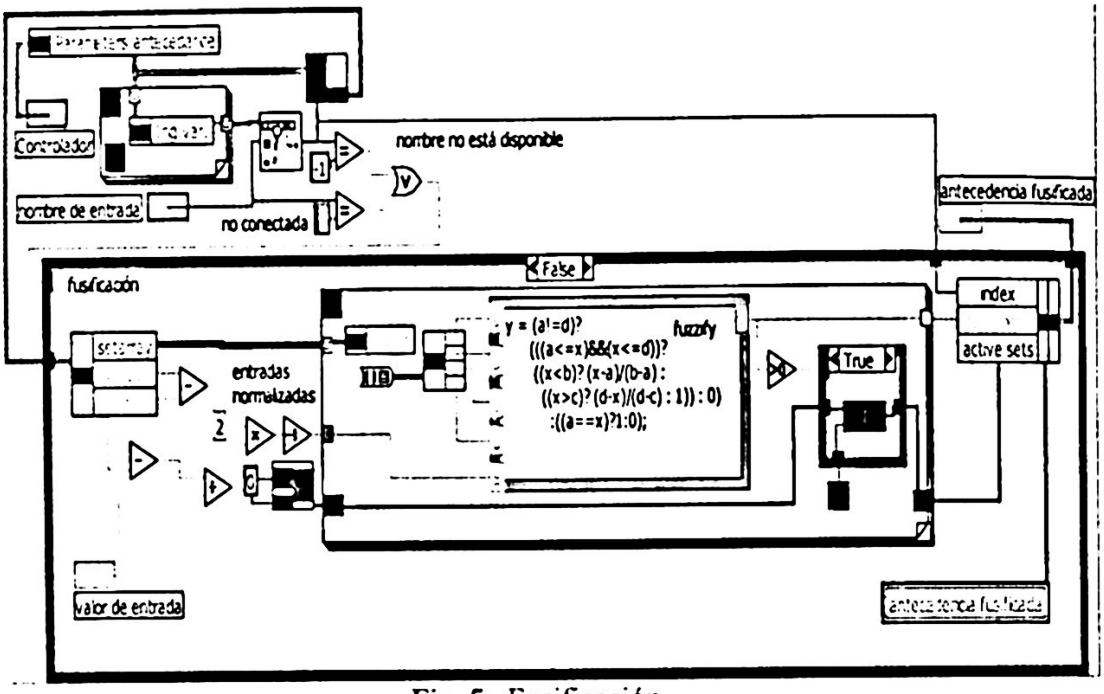


Fig. 5. Fusificación

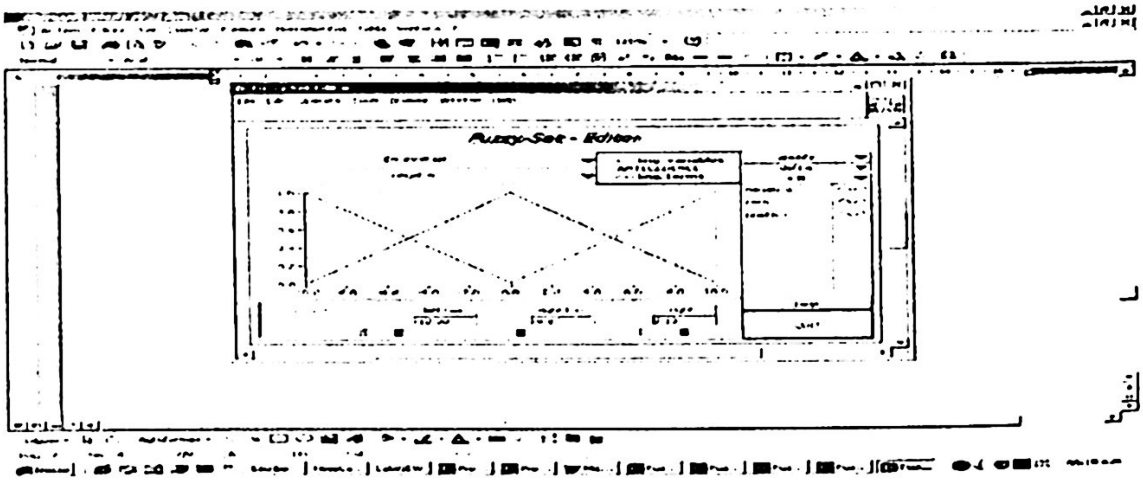


Fig. 6. Editor de conjuntos difusos

En la figura 6 tenemos las funciones de pertenencia de la variable lingüística Error y en la figura 7 se presenta un ejemplo de los puntos de las funciones de pertenencia que toma el editor difuso y los pasa al controlador.

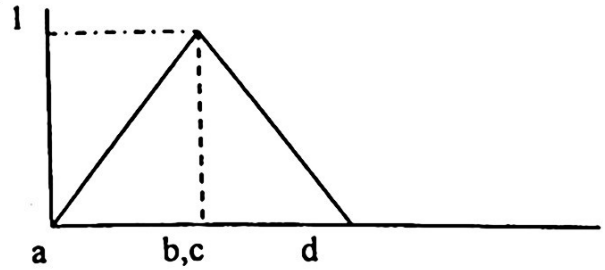


Fig. 7. Puntos de la función de pertenencia.

En la figura 5 se puede observar que el VI de fusificación toma los puntos a, b, c, d normalizados de cada uno de los términos lingüísticos y los pasa en un arreglo hasta un lazo **for** para hacer su auto indexación. Dentro de la estructura case general se lee el valor de entrada real y se normaliza. Un nodo de fórmulas toma los puntos a, b, c, d y hace los cálculos para obtener los grados de pertenencia en cada función. Las fórmulas de las rectas que forman las funciones de pertenencia están en el siguiente algoritmo:

El grado de pertenencia "y" = (x-a)/(b-a) si "a" es diferente de "d" y "a" es menor o igual a "x" y "x" es menor o igual a "d" y "x" es menor a "b", de lo contrario si "x" es mayor a "c" y=(d-x)/(d-c) si no y=1.
Si "x" es menor que "a" ó mayor que "d" y=0.
Si "a" es igual a "d" y "a" es igual a "x" y=1, si no y= 0

```
y = (a!=d)?
    (((a<=x)&&(x<=d))?
     ((x<b)? (x-a)/(b-a) :                                    (1)
       ((x>c)? (d-x)/(d-c) : 1)) : 0)
        :((a==x)?1:0);
```

La salida de la fusificación da los índices de las variables lingüísticas, los grados de pertenencia para las funciones activas y los conjuntos activos y estas pasan al VI de la Inferencia difusa que se muestra en seguida:

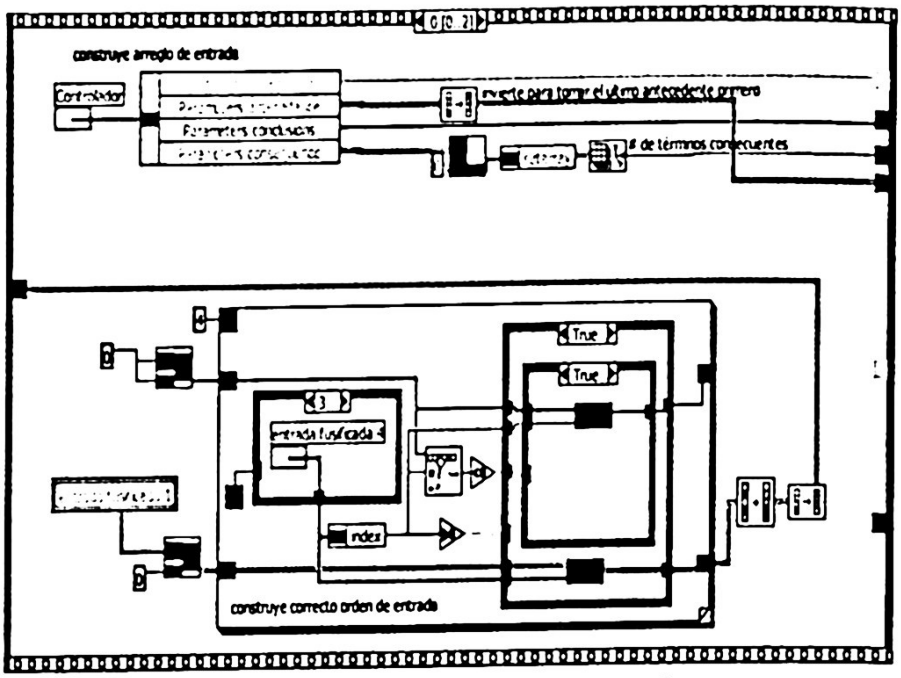


Fig. 8. IV de Inferencia difusa-marco 0

En el IV de la figura 8 se muestra el primer armazón de la secuencia en la que se toman los grados de pertenencia de los conjuntos activos y se colocan en un arreglo de una dimensión. Se recupera del archivo ControlMotor1.fc, el # de términos consecuentes, los factores de peso y los antecedentes. Todo ello se pasa al segundo marco de la secuencia mostrada en la figura 9. En el segundo marco se lee el arreglo de los grados de pertenencia y los conjuntos fuzzy activos y se efectúa con ellos la implicación de Mamdani que dice que:

$$\mu_{Rk}(x,y) = \min(\mu_A(x), (\mu_B(y)) \qquad (2)$$

Obteniéndose los mínimos para las reglas encendidas y las reglas activas.

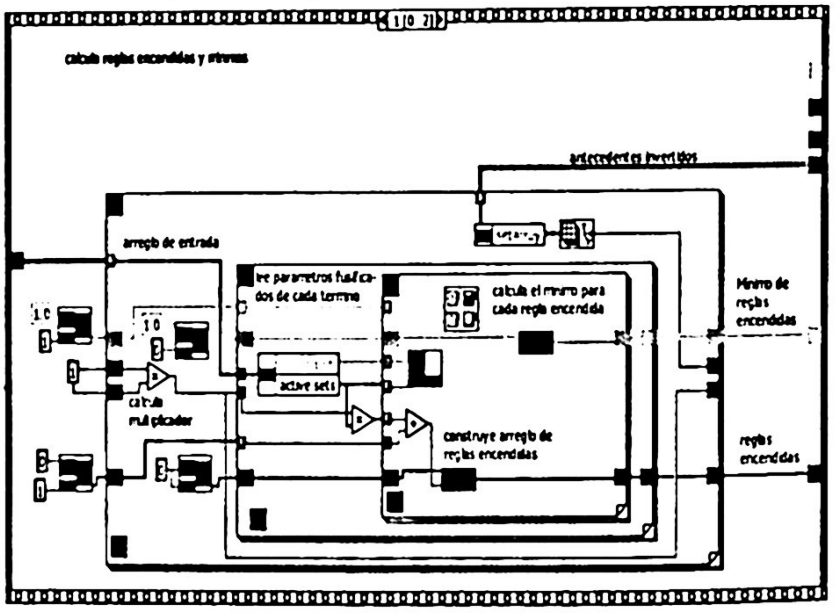


Fig. 9. VI de Inferencia difusa-marco 1.

Tabla 1. Base de reglas del controlador

	Error	Derivada error	Salida	DoS
1	Negativo	Negativo	Negativo	1.0
2	Negativo	Cero	Negativo	1.0
3	Negativo	Positivo	Cero	1.0
4	Cero	Negativo	Negativo	1.0
5	Cero	Cero	Cero	1.0
6	Cero	Positivo	Positivo	1.0
7	Positivo	Negativo	Cero	1.0
8	Positivo	Cero	Positivo	1.0
9	Positivo	Positivo	Positivo	1.0

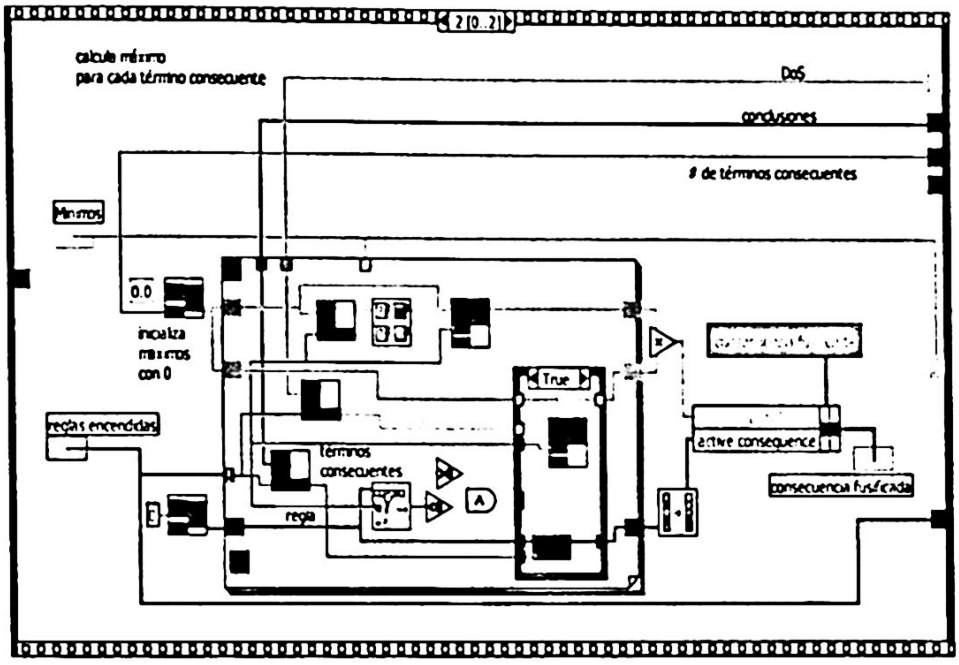


Fig. 10. IV de Inferencia difusa-marco 2.

En el tercer marco de la secuencia del IV de inferencia, figura 10, se aplica la relación:

$$R = \cup_{k=1}^{n} R_k \quad y \quad \mu_R(x,y) = \bigvee_{k=1}^{n} \mu_R(x, y) \tag{3}$$

a traves del operador de agregación max a la implicación de Mamdani se tiene:

$$\mu_R(x,y) = \max \mu_{Rk}(x, y)$$
$$\mu_R(x,y) = \max (\min (\mu_A(x), \mu_B(y))) \tag{4}$$

Corrigiendo la salida con el factor de peso de cada regla, tenemos los grados de pertenencia de los conjuntos difusos de salida y las consecuencias activas.

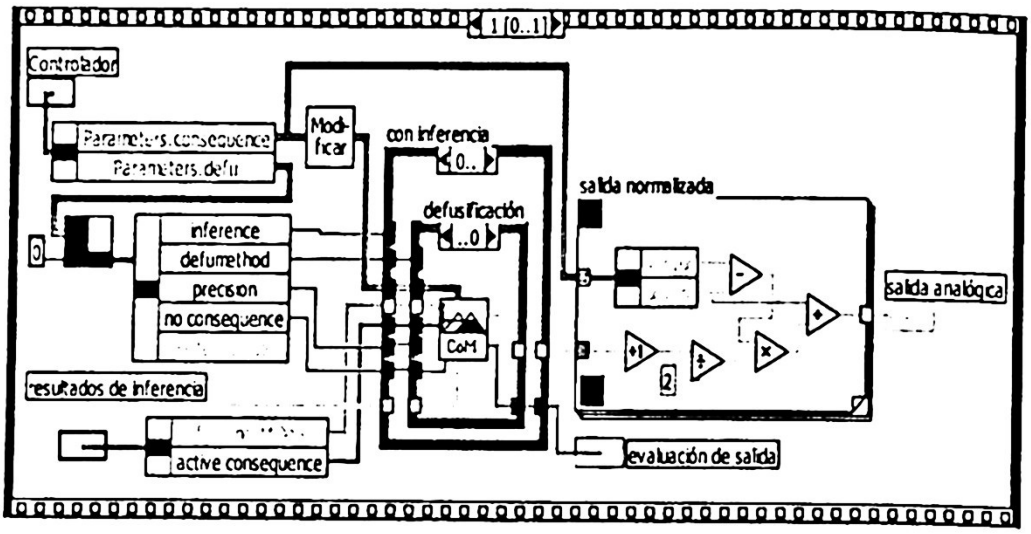


Fig. 11. IV de defusificación

Antes de realizar la defusificación se debe modificar los parámetros consecuentes configurados en la base de datos. Esto se hace con el IV de la figura 12, [6].

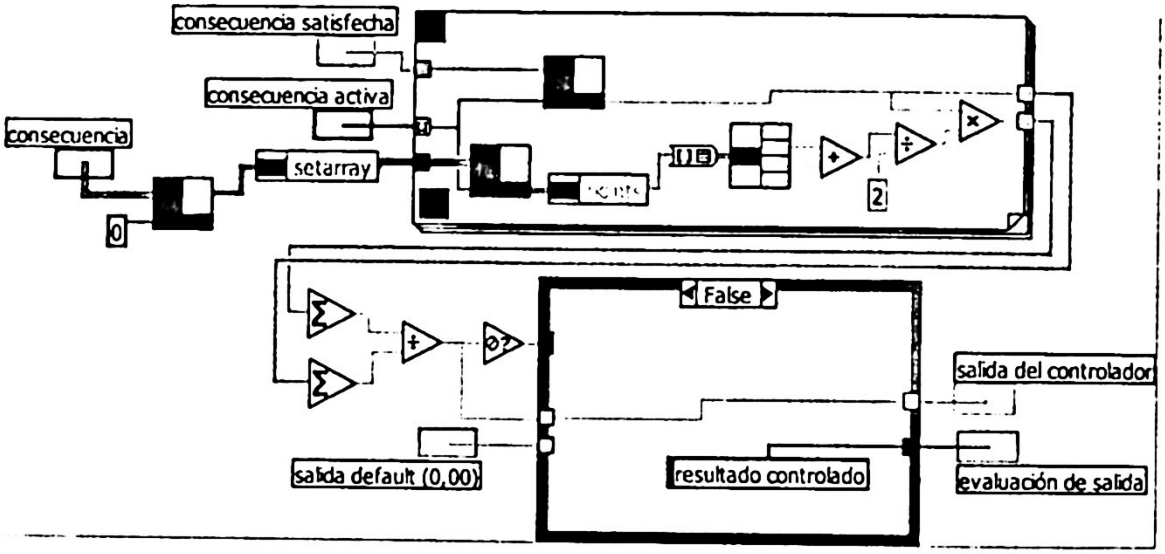


Fig. 12. IV de Modificación

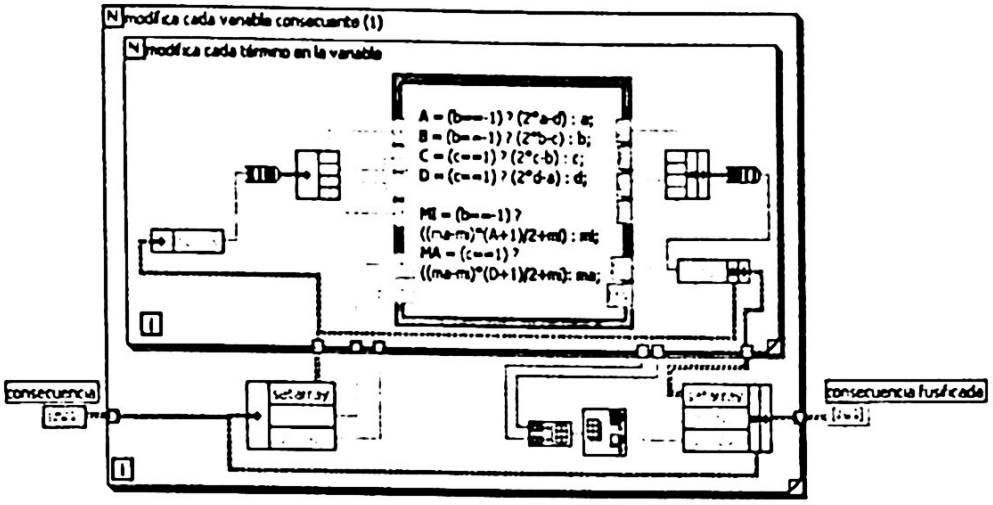


Fig. 13. IV de defusificación

En la figura 13 tenemos el IV de defusificación que recibe los grados de pertenencia de los conjuntos difusos de salida de la inferencia, las consecuencias activas y los puntos a, b, c, d modificados de los términos consecuentes y realiza la siguiente operación para utilizar el método de Centro de Máximos, también llamado Método de Pesos:

$$Z = \frac{\sum_{k=1}^{m} w_k z_k}{\sum_{k=1}^{m} w_k} \qquad (5)$$

El IV de la figura 13 da una salida abrupta y el IV de la figura 11 termina aplicando los factores de escalamiento para obtener la salida analógica final. Se desarrollaron las pruebas y ajustes de parámetros, obteniéndose los resultados mostrados en las figuras 14 y 15.

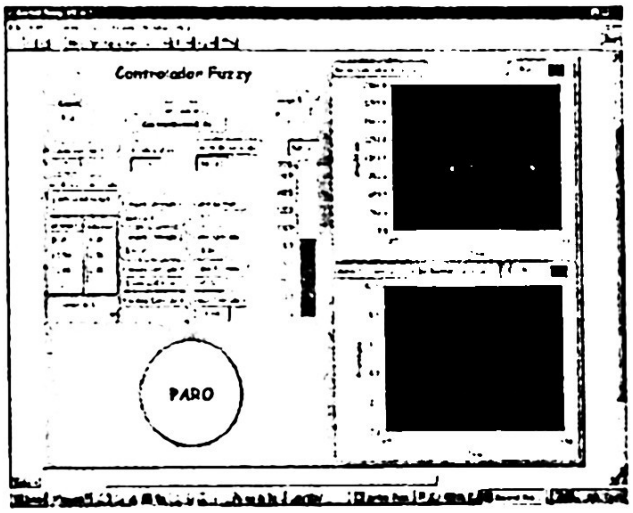


Fig. 14. Monitoreo del Control Difuso

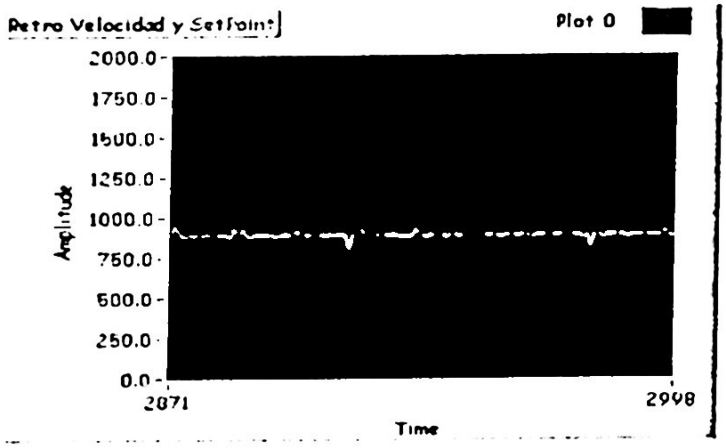


Fig. 15. Monitoreo del Set Point y la velocidad

Conclusiones

Se desarrollo un control interactivo para regular la velocidad de un motor de CD con lógica difusa, presentando la facilidad de poderse usar para cualquier otro tipo de control, con pequeñas modificaciones, manejando hasta 4 señales de entrada. La utilidad del sistema logrado es evidente.

Por el lado de la lógica difusa aplicada al control de procesos es claro que sin ser la panacea que va a resolver todos los problemas del mundo, si puede ser una herramienta muy útil sola o combinada con redes neuronales y algoritmos genéticos, para resolver problemas importantes donde los controles tradicionales no lo han hecho o que presentan desventajas en su aplicación.
Específicamente para este caso, con el control difuso se ve mejorado el funcionamiento del motor, sobre todo durante disturbios grandes, comparado con un control PID

Bibliografía

[1] George and Maria Bojadziev. Fuzzy Set, Fuzzy Logic, Applications. World Scientific.1995.
[2] W.A.R.P. Application note. Fuzzy Control of a DC Motor. SGS Thompson. Microelectronics. June 1994.
[3] Bruce Mihura. LabView for Data Acquisition. Virtul Instrument Series. National Instrument. Prentice Hall, 2001.
[4] Diego J. Ramírez Martinez & Romeo Urbieta P. Regulación de Velocidad de un motor de CD con Control Difuso. Implementación a través de un Algoritmo con LabView. Reporte 2003.
[5] Jelena Godjevac. Clarifying Ideas on Programing Lenguages and Fuzzy Control Theory by Using Mobile Robots. EPFL Microcomputing Laboratory IN-F Ecublens CH-1015 Lausane.
[6] Jelena Godjevac. Neuro-Fuzzy Controllers. Design and Application. Collection META Presses Polytechniques et Universitaires Romandes.
[7] National Instruments. Fuzzy Logic for G Toolkit Reference Manual. March 1997 Edition. Part Number 321511A-01

Análisis y Procesamiento de Señales

Estudio de señales electroencefalográficas para la detección de movimientos voluntarios

Aldo Escalona Delgado[1], Evelyn Peredo Fuentes[1].

[1] Unidad Profesional Interdiscliplinaria de Ingenieria y Tecnologias Avanzadas del Instituto Politecnico Nacional, Av. Instituto Politecnico Nacional #2580, Col. La Laguna Ticomán, C.P. 07034
aldonicoh@hotmail.com, epefe@hotmail.com

Abstract. This is a study of electroencefalografic signals measured in the motor cortex, by mean non invasive electrodes of silver-silver chloride placed in the positions C3,C4,T3,T4, according to the international system 10-20, to detecte voluntaire movementes of the right and left index fingers, in order to classify them by networks neurals (Support Vector Machine and Learning Vector Quantization). Besides was found in experiments that signals measured in the positions FP1-F3, FP2-F4 in the same system, give a significative change of amplitude when is made a blink, so its proposed a simple application program where a round element in the screen is controlled by two secuences of blinks. Similar studies are called Brain Computer Interfaces, these are a growing field of interest of human computer interaction with divers medical applications, that try to give independence to people with motors problems.

1 Introducción

Las investigaciones de la relación del hombre con la máquina, avanzan en la experimentación de lo que podría significar una nueva concepción del cuerpo humano, con un sistema regido voluntariamente por impulsos cerebrales controlados y capaces de salvar las limitaciones espaciales o la pérdida de funciones motoras y sensoriales.

Es posible medir potenciales en respuesta a un movimiento voluntario, ya que el tejido nervioso presenta como una de sus funciones básicas la capacidad de generar potenciales eléctricos que son la base de la excitabilidad del organismo. Actualmente existen sistemas que desempeñan una comunicación entre el cerebro humano y la computadora, a estos sistemas se ha llamado Interfases Cerebro Computadora o BCI (Brain Computer Interfas).

El presente trabajo propone un BCI, que utiliza potenciales relacionados con movimientos voluntariamente desencadenados en la corteza motora para establecer una comunicación con una computadora, mediante algoritmos de reconocimiento de patrones. Se comienza desde el diseño y construcción de un EEG, implementación de redes neuronales tipo Maquina de Soporte Vectorial (SVM) y Cuantización de Vectores de Aprendizaje (LVQ). Posteriormente dados los resultados poco favorables se

plantea una disyuntiva que propone un canal de comunicación mediante secuencias de parpadeos para traducirlas en comandos simples.

2 Arreglo Experimental

El arreglo experimental que se propone en este trabajo consiste en hacer ensayos en 2 personas sanas con electrodos no invasivos sobre el cuero cabelludo colocados cerca de la corteza motora, es decir, sobre T3, C3, C4 y T4 del sistema internacional 10-20 y además se utilizan dos interruptores sobre ambos dedos índices como lo muestra la figura 1, al indicarle a la persona el inicio del experimento, debe oprimir los interruptores aleatoriamente durante un periódo de tiempo definido a intervalos cortos.

En la figura 1 la primera etapa es la adquisición de datos a través de la adaptación del EEG de cuatro canales, estas señales se digitalizan y junto con las señales de los interruptores se transmiten a una computadora.

Tanto las ondas cerebrales como las señales del los interruptores, serán almacenadas en memoria física de una computadora, en donde, se tomarán características y mediante un entrenamiento de redes neuronales artificiales se intenta discriminar las señales. Uno de los patrones a buscar es cuando el sujeto haya presionado el interruptor de lado izquierdo, otro para el derecho.

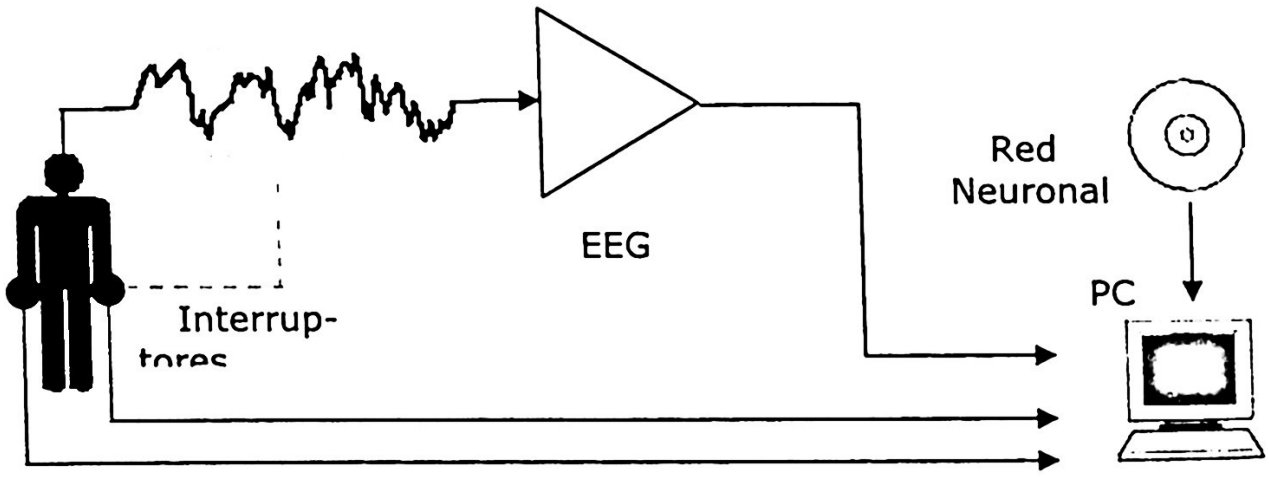


Figura 1. Diagrama a bloques del arreglo experimental con un EEG de cuatro canales y dos interruptores.

3 Procesamiento de la señal

Las señales bioeléctricas en general tienen un ancho de banda reducido. En el caso de una señal de EEG la amplitud esta comprendida entre 10 y 100 μV y la banda de frecuencias oscila entre 0.1 y 50 HZ. El amplificador usado en el diseño del EEG tiene un factor de rechazo en modo común mayor a 100 dB, rechaza banda a 60Hz, filtro pasa bajas de bessel 6° orden, banda de frecuencia de 0.5 a 75Hz, En señales

EEG esta frecuencia es de 50Hz por tanto la frecuencia mínima de muestreo es de 100Hz. Para mayor calidad de la señal se usan 500Hz con una resolución de 10 bits.

Los vectores de entrenamiento son formados tomando los fragmentos de señal de los cuatro canales que coinciden con la activación de los interruptores. Se toman 64ms antes y 64ms después de que se genera el movimiento. Los fragmentos se ordenan en un vector colocando un fragmento tras otro, de tal manera que se forma un único vector de 128*2*4=512 (figura 2), a este vector le llamaremos Vector de Comportamiento Global (VCG).

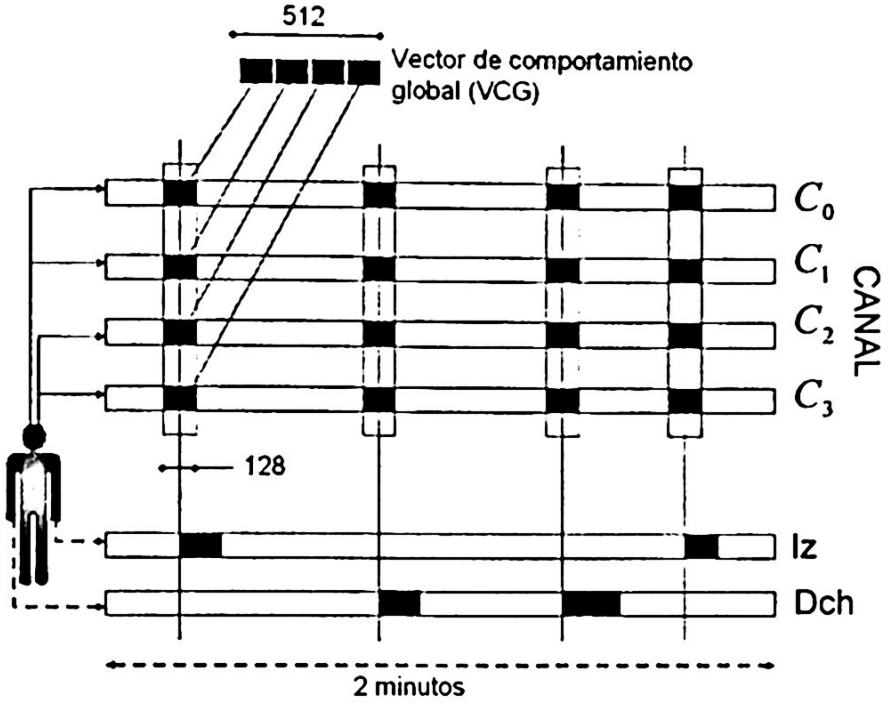


Figura 2. Formación de Vectores de Comportamiento Global.

4 Redes neuronales

4.1 LVQ (Learning Vector Quantization)

Si w_c es el vector de referencia más próximo al patrón de entrada x^μ presentado y denominamos C_{wj} a la clase definida por dicho vector de referencia, se tendrá la siguiente actualización para la neurona c:

si $C_{wc} = C_{x\mu}$ (clasificación correcta), el vector de referencia se modifica de la forma

$$w_c(t+1) = w_c(t) + \alpha_t \left[x^\mu - w_c(t)\right] \tag{1}$$

con $0 < \alpha_t << 1$, y normalmente decreciente con el tiempo t (por ejemplo, partiendo de 0.1). En este caso, los pesos rotan hacia las entradas, aproximándose (se puede considerar equivalente a un premio).

Si por el contrario $C_{wc} \neq C_{x\mu}$ (clasificación incorrecta), la modificación se realiza justo en sentido contrario, viniendo dada por

$$w_c(t+1) = w_c(t) - \alpha_t \left[x^\mu - w_c(t)\right] \tag{2}$$

de forma que los pesos tienden a alejarse de las entradas (castigo).

El resto de los vectores de referencia no se modifican. El proceso se repetirá iterativamente con todos los ejemplos de entrenamiento, un numero de épocas finita.

4.2 Máquina de Soporte Vectorial

La SVM se puede escribir como:

$$f(\mathbf{x}) = sign\left(\sum_{vectores\ soporte} \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i) + b \right) \tag{3}$$

y la arquitectura de la Maquina de Soporte Vectorial queda como lo muestra la figura 3.

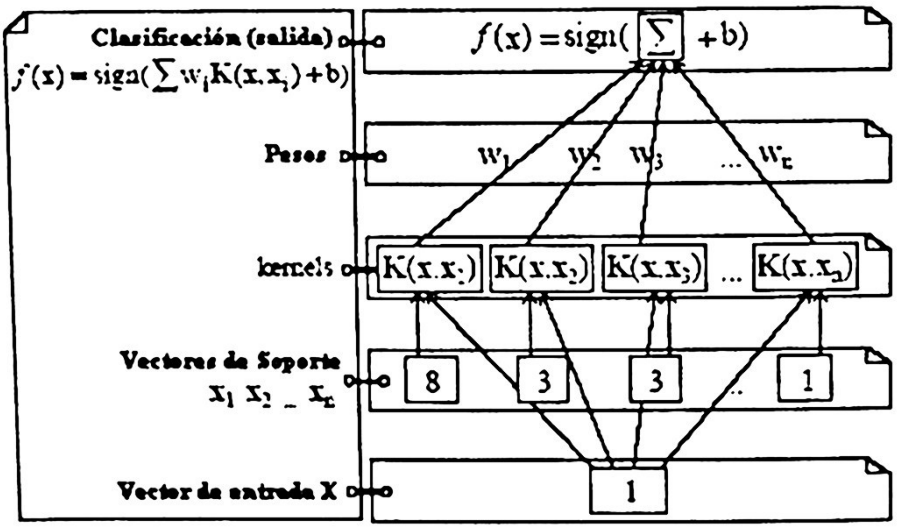


Figura 3. Arquitectura de la Maquina de Soporte Vectorial

Para obtener los multiplicadores de Lagrange α_i, nuevamente sólo es necesario conocer el kernel.

Minimizar:

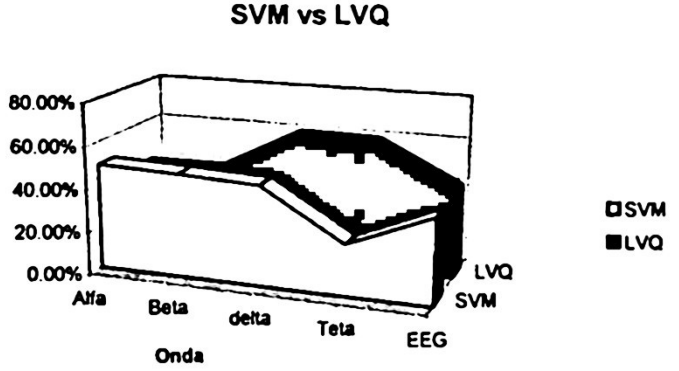
$$W(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \tag{4}$$

sujeto a:

$$0 \leq \alpha_i \leq C, \quad i = 1, \ldots, N \tag{5}$$

5 Resultados

El total es de 15 muestras por cada una de las dos personas. Para una cuantificación de resultados se realizó la etapa de aprendizaje o entrenamiento con 10 muestras, esto representa alrededor de 1000 VCGs (vectores de comportamiento global). Las pruebas se realizan con las 5 muestras restantes que representan alrededor de 500 VCGs. Los resultados están reflejados en las graficas de la figura 4.



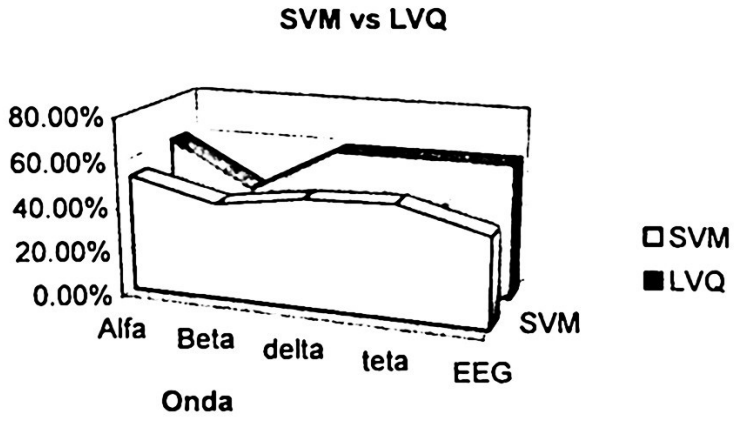


Figura 4. Gráficas comparativas para las personas A y B.

Ya que con los datos obtenidos, el porcentaje de acierto no sobrepasa el 60%, en una aplicación en tiempo real traería como consecuencia que el error aumente y que el control de un BCI no sea confiable al 100%, se plantea ahora como alternativa realizar un programa de simulación que este controlado pero por medio de otra señal que se encontró en las pruebas, que si es visible.

5.1 Disyuntiva

Dentro de los experimentos realizados se observó que cuando se lleva a cabo un parpadeo, se forma una cresta visiblemente apreciable en la señal electroencefalográfica (Figura 5).

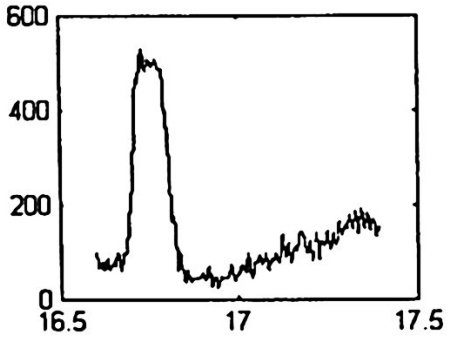


Figura 5. Ritmo de Parpadeo

Este comportamiento fue investigado en bases de datos y se comprobó que éste se hace presente en las señales electroencefalográficas, y pertenece a las onda teta.

Por lo anterior se propone como aplicación realizar una simulación gráfica simple en la que se mueve un elemento circular en la pantalla, controlando su desplazamiento, ya sea izquierdo o derecho, por medio de secuencias de parpadeos, en un tiempo determinado.

5.2 Caracterización

En la figura 6 se observan tres ritmos de mayor amplitud en la señal electroencefalográfica, que corresponden a tres parpadeos voluntarios. En dicha figura se presentan las señales de dos canales frontales, es decir de las señales obtenidas colocando los electrodos en las posiciones FP1-F3 y FP2-F4 del montaje longitudinal bipolar (LB), según las tablas de combinaciones de posicionamiento de electrodos recomendados del sistema internacional 10-20.

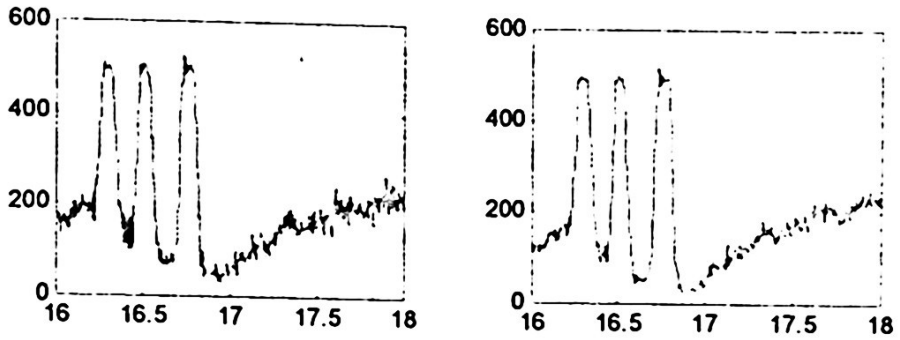


Figura 6. Secuencia de 3 parpadeos.

La caracterización de estos ritmos aparecen en la figura 7-a y 7-b para dos y tres parpadeos respectivamente.

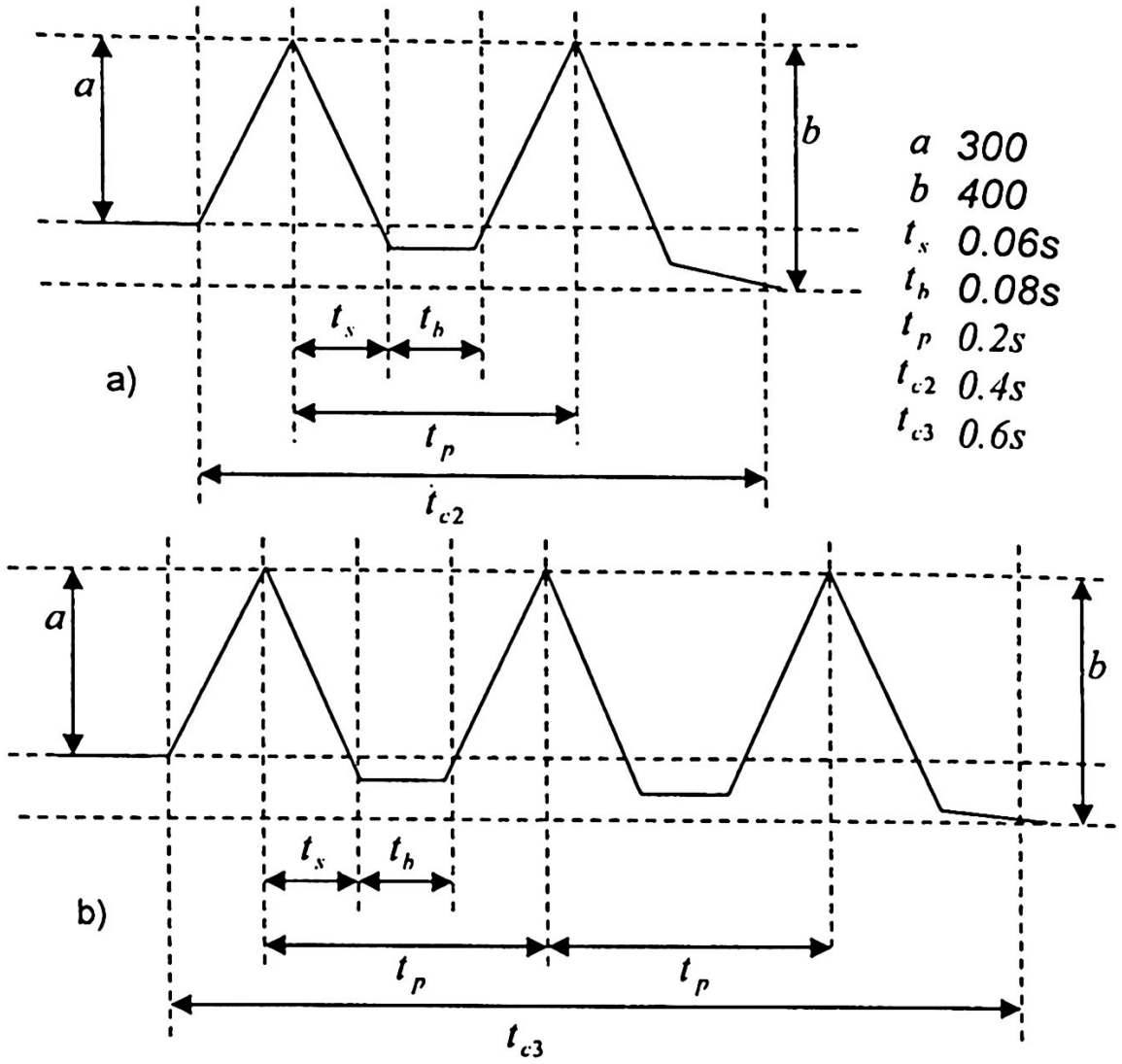


Figura 7. Caracterización de ritmos. a) Dos parpadeos. b) Tres parpadeos

Esto permite establecer como patrón, secuencias de tres parpadeos para generar el comando de dirección a la derecha y secuencias de dos para generar el comando de dirección a la izquierda. Cuando se realice un solo parpadeo no se el sistema no debe responder con acción alguna, ya que se podría confundir éste parpadeo, con uno involuntario que se realiza cuando se lubrica el ojo.

6 Conclusiones

Una de los avances que podemos citar en la realización de este trabajo, es que se encontró que se genera un patrón visible cuando se realiza un parpadeo, de aquí se propuso una aplicación simple. Ésta aplicación consiste en un programa gráfico, que simula un camino en forma de cruz, donde un punto móvil se mueve en la pantalla, controlado por medio de diferentes secuencias de parpadeos traducidos a comandos simples. Las pruebas se realizaron con tres personas mentalmente sanas, 2 hombres y una mujer, de 22 años de edad en promedio.

Aplicación desarrollada permite reducir enormemente el costo computacional ya que las señales mencionadas anteriormente se pueden obtener hasta una frecuencia de

muestreo mayor o igual a 15Hz., en comparación con la frecuencia implementada que debía ser mayor o igual a 100Hz. Con esto se reduce el espacio de memoria utilizado para guardar estos datos, el procesamiento computacional y hasta el número de canales del amplificador. Además, el entrenamiento se facilita ya que la caracterización de un parpadeo es sencilla. Por si fuera poco, un sistema controlado con parpadeos no es vulnerable a ruidos por distracción de las personas, es decir no es indispensable un escenario controlado.

Los resultados obtenidos son satisfactorios, y que el control de dicho móvil, incrementa su eficiencia en la manera en la que la persona adquiere cierto entrenamiento. Para lograr una mejoría en esta aplicación se necesitan realizar mucho mayor número de pruebas y con un número mayor de personas para poder cuantificar mejor los resultados.

Referencias

1 Bonifacio Martín del Brio et. al. "Redes neuronales y sistemas difusos", Editorial Alfaomega. 2ª Edición, 2002. México.

2 Yonas B. Dibike, Slavco Velickov, Dimitri Solomatine, Michael B. Abbott. "Model Induction with Support Vector Machines: Introduction and Applications", ASCE Journal of Computing in Civil Engineering, July 2001, vol. 15, No. 3.

1

!

Reconocimiento de Objetos Traslapados usando Memorias Asociativas

Roberto A. Vázquez, Humberto Sossa & Ricardo Barrón
Centro de Investigación en Computación – IPN
Av. Juan de Dios Batíz, esquina con Miguel Otón de Mendizábal
Ciudad de México, 07738. México
Contacto: robertov@sagitario.cic.ipn.mx

Resumen. El problema del reconocimiento de patrones presenta algunas dificultades que dependen del tipo de objetos a reconocer y del contexto donde se hayan dichos objetos. El problema se incrementa cuando los objetos aparecen traslapados, debido a que para el reconocimiento se cuenta con una parte del objeto. Ésto dificulta su identificación debido a que varios de los objetos tienen partes similares. La importancia práctica del reconocimiento de objetos en presencia de traslapes se aprecia en aplicaciones como conteo de células en tejidos dañados, detección de objetos intrusos, conteo de objetos, etc. En éste trabajo se presenta una metodología para resolver el problema del reconocimiento de objetos traslapados, basado en un algoritmo de *Barrido Bloqueante* usando una máscara para detectar un objeto a partir de sus partes más representativas y haciendo uso de las memorias asociativas.

Palabras clave. Reconocimiento de objetos traslapados, memorias asociativas

1 Introducción

El uso intensivo de computadoras u otros dispositivos electrónicos en los últimos años, ha impulsado el estudio y aplicación de técnicas de reconocimiento de patrones para resolver algunos problemas de clasificación o reconocimiento.

Existen cuatro enfoques para resolver el problema de reconocimiento de patrones:

- Reconocimiento estadístico de patrones.
- Reconocimiento sintáctico de patrones.
- Redes neuronales.
- Reconocimiento lógico combinatorio de patrones.

Se han desarrollado infinidad de trabajos que utilizan estos enfoques para resolver un problema específico de reconocimiento [8] [9] [10], pero ninguno de éstos trabajos permite que sus patrones estén traslapados o se toquen.

Conforme avanza la tecnología y dependiendo del tipo de problema que se está resolviendo, es necesario que las restricciones para que el problema sea resuelto sean

las mínimas. Una restricción que ha sido atacada por varios investigadores, es la que involucra el traslape de los objetos. En varios trabajos se han propuesto metodologías para resolver el problema de reconocimiento de patrones sin importar que los objetos estén traslapados [11][12].

En algunos trabajos [6][7] se hace el uso de las memorias asociativas para resolver el problema del reconocimiento de patrones, pero no se permite que los objetos estén traslapados o se toquen.

Para hacer el reconocimiento de un objeto se tiene que entrenar una memoria asociativa con los diferentes objetos que se quiere reconocer. La memoria asociativa utiliza para su entrenamiento los rasgos descriptivos de cada uno de los objetos.

Cuando se trata de reconocer objetos que no están traslapados se pueden usar descriptores por región como los invariantes de Hu [4]. Los invariantes de Hu son calculados a partir de los momentos centrales normalizados de segundo y tercer orden. Estos momentos son invariantes a rotación, traslación y escalamiento.

$$M_{H1} = \eta_{20} + \eta_{02}$$
$$M_{H2} = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}$$
$$M_{H3} = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2$$
$$M_{H4} = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2$$

Una manera sencilla para obtener los rasgos descriptivos de un objeto y poder entrenar la memoria asociativa es:

1. *Aplicar una binarización [2] a la imagen donde se encuentra el objeto.*
2. *A la imagen binaria obtenida aplicarle el algoritmo de etiquetado de componentes conexas [1].*
3. *Eliminar el ruido generado por la binarización y el etiquetado. Una solución es un filtro de tamaño mínimo [3].*
4. *Calcular los invariantes de Hu a la componente obtenida.*
5. *Entrenar la memoria asociativa con los rasgos calculados.*

Una vez entrenada la memoria asociativa con los rasgos descriptivos, se podrá hacer el reconocimiento de un objeto de la siguiente forma:

1. *Aplicar los pasos 1 a 5 a la imagen que tiene el objeto a reconocer.*
2. *Alimentar la memoria asociativa con los rasgos descriptivos del objeto a reconocer.*
3. *Finalmente, operar la memoria con los rasgos descriptivos del objeto para obtener la clase a la que pertenece.*

Ya vimos que el problema del reconocimiento de objetos se incrementa cuando los objetos están traslapados. Si usamos la memoria construida anteriormente y queremos reconocer un objeto traslapado, sencillamente la memoria no clasificará correctamente al objeto. Esto se debe a que el objeto que está traslapado con otro objeto va a ser analizado de manera global, es decir, se van a tomar como si fueran un solo objeto y este objeto no pertenece en lo más mínimo al conjunto de objetos con el que se entrenó a la memoria asociativa.

En este trabajo se presenta una metodología que da solución al problema de reconocimiento de objetos traslapados. La metodología aplica los mismos pasos descritos anteriormente para obtener los rasgos descriptivos, con la diferencia de que los rasgos obtenidos no van a ser de todo el objeto, sino de una parte representativa del objeto. El método utiliza una memoria asociativa y aplica el algoritmo de *Barrido Bloqueante* para reconocer un objeto dentro de una imagen sin importar que esté traslapado o no.

2 Memorias asociativas

El propósito fundamental de una memoria asociativa [5] es recuperar correctamente patrones completos a partir de patrones de entrada, los cuales pueden aparecer alterados con ruido aditivo, sustractivo o combinado. De acuerdo con ésta afirmación, una memoria asociativa **M** puede formularse como un sistema de entrada y salida, donde las entradas van a ser los rasgos descriptivos del objeto a clasificar y la salida la clase a la que pertenece el objeto.

La solución de un problema usando una memoria asociativa es dividida en dos fases:

1. Fase de aprendizaje (generación de la memoria asociativa).
2. Fase de recuperación (operación de la memoria asociativa).

La memoria asociativa **M** se representa mediante una matriz cuya componente *ij*-ésima es m_{ij}. La matriz M se genera a partir de un conjunto finito de asociaciones conocidas de antemano: éste es el *conjunto fundamental de asociaciones* o simplemente *conjunto fundamental*. A los patrones que conforman las asociaciones del conjunto fundamental, se les llama *patrones fundamentales*.

La memoria asociativa utilizada para el reconocimiento de objetos traslapados en este trabajo es la presentada en el artículo [6]. Ésta memoria es una memoria asociativa morfológica que utiliza la media de los rasgos descriptivos de cada objeto para construir la matriz **M**, determinando la clase a la que pertenece un objeto por medio de la distancia de Chebyshev.

A continuación se presenta un ejemplo que ejemplifica la manera de implementar la fase de aprendizaje y de recuperación de ésta memoria.

Dadas las siguientes asociaciones de patrones:

((1,1,1), 1)	((4,4,4), 2)	((10,9,10), 3)
((1,2,1), 1)	((4,4,5), 2)	((9,9,10), 3)
((2,1,1), 1)	((4,5,5), 2)	((10,10,10), 3)
((1,1,2), 1)	((5,4,4), 2)	((10,11,11), 3)
((2,2,2), 1)	((5,4,5), 2)	((10,9,11), 3)

Entonces, calculando la media de los patrones de cada clase obtenemos:

$\phi_1 = (1.4, 1.4, 1.4)$, $\phi_2 = (4.4, 4.2, 4.5)$, y $\phi_3 = (9.8, 9.6, 10.4)$.

Quedando así

$$M = \begin{bmatrix} 1.4 & 1.4 & 1.4 \\ 4.4 & 4.2 & 4.5 \\ 9.8 & 9.6 & 10.4 \end{bmatrix}$$

Para la fase de recuperación utilizamos la siguiente ecuación:

$$i = \arg\left[\bigwedge_{l=1}^{m} \bigvee_{j=1}^{n} \left| m_{lj} - x_j \right| \right] \qquad (1)$$

Donde $\vee \equiv \max$ y $\wedge \equiv \min$

Suponga que se quiere recuperar la clase a la que pertenece el patrón $x = (1,1,1)$

Entonces

$$\vee(1.4-1, 1.4-1, 1.4-1) = 0.4$$
$$\vee(4.4-1, 4.2-1, 4.5-1) = 3.5$$
$$\vee(9.8-1, 9.6-1, 10.4-1) = 9.4$$

$$\arg(\wedge(0.4, 3.5, 9.4)) = 1$$

Hemos visto cómo se puede clasificar un patrón usando ésta memoria asociativa. Ahora, para resolver nuestro problema de reconocimiento de objetos se tiene que entrenar a la memoria con los rasgos descriptivos de cada objeto.

3 Método para reconocer objetos traslapados

Como se dijo anteriormente, para entrenar la memoria asociativa se tienen que obtener los rasgos descriptivos, no del objeto sino de una parte o más partes del objeto. Un objeto puede ser descompuesto en varias partes, pero solo algunas de ellas van a caracterizar de manera única al objeto.

Para la selección de alguna parte del objeto se utilizó una ventana circular. Una vez seleccionada la parte del objeto le aplicamos los pasos 1 a 5, descritos en la sección 1, para la obtención de los rasgos descriptivos de la parte del objeto. Estos rasgos van a ser usados para el entrenamiento de la memoria asociativa.

A cada uno de los objetos con los que se quiere trabajar se le va a asignar una memoria asociativa. Cada memoria asociativa va a reconocer cinco partes de un objeto. La primera clase de parte del objeto va a ser la parte más descriptiva del objeto y la última clase de parte del objeto va a ser la parte menos descriptiva del objeto.

Finalmente se van a utilizar éstas memorias asociativas combinadas con el algoritmo de *Barrido Bloqueante* para determinar la existencia de algún objeto dentro de una imagen.

3.1 Fase de entrenamiento

Como se estableció anteriormente, a cada objeto se le va a asignar una memoria asociativa. Para poder entrenar estas memorias, primeramente hay que determinar cuáles son las partes del objeto que lo representan mejor. Una primer solución, para escoger las partes más descriptivas de los objetos, sería basándonos en las características que nosotros utilizamos para reconocer un objeto. Por ejemplo, a un tornillo lo reconocemos por la cabeza y a una rondana la reconocemos por el orificio que lo caracteriza. Si utilizamos los invariantes de Hu para describir las partes más descriptivas, estas partes de objetos son casi idénticas para la computadora.

Ésta primera solución como se puede ver en la tabla 1 es incorrecta, debido a que los rasgos descriptivos que se obtienen de éstas partes son muy cercanos. Estas partes no pueden ser consideradas como representativas y no pueden ser usadas para discriminar un objeto.

Una segunda alternativa es seleccionar todas las posibles partes de cada uno de los objetos con los que se va a entrenar a la memoria asociativa. Una vez que se tienen todas las partes de los objetos se tendrá que hacer una búsqueda de los rasgos más representativas de cada objeto. De éstos rasgos, finalmente escogemos los cinco más representativos de cada objeto para poder entrenar su memoria asociativa.

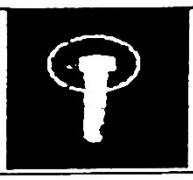
Objeto		
Rasgos descriptivos	0.187008390255792 0.000828434034786876 0.002073528381953 5.33516085908603E-5	0.198022513948688 1.01942318662163E-8 6.46010546977373E-10 1.80497146412438E-6

Tabla 1. Seleccion de parte usando la forma mas representativa.

Ya se ha establecido que la memoria asociativa de un objeto va a reconocer cinco partes del objeto. Se van a utilizar 20 imágenes de cada objeto en diferentes posiciones. Utilizando la ventana circular se van a obtener los rasgos descriptivos de las cinco partes del objeto más representativas y se van a promediar para obtener el vector de rasgos descriptivos promedio de cada una de las cinco partes más representativas del objeto. De ese grupo de partes, el más representativo va a ser la clase 1 (de las partes del objeto que la memoria asociativa va a reconocer) y el menos representativo va a ser la clase 5 (de las partes del objeto que la memoria asociativa va a reconocer).

3.2 Fase de recuperación

Para poder hacer el reconocimiento de los objetos que hay en una imagen, tenemos que seleccionar partes del objeto con la ventana circular que se utilizó para el entrenamiento de la memoria asociativa. Ésta ventana la tenemos que recorrer por toda la imagen para poder seleccionar partes de objeto y poder clasificarlas.

El algoritmo de B*arrido Bloqueante* nos permite clasificar un objeto que está traslapado por medio de sus partes más representativas. Para hacer la selección de partes del objeto se utiliza una máscara del tamaño de la imagen que está llena de ventanas circulares contiguas. Ésta máscara es la que va a permitir seleccionar o no seleccionar una parte del objeto. La máscara se auxilia de una tabla de bloqueados para decidir si coloca una ventana o no dentro de la máscara. La máscara sólo va a poder agregar una ventana si el región de píxeles donde se quiere colocar no está registrada en la tabla de bloqueados. Es decir, si la parte seleccionada del objeto no ha sido reconocida por una memoria asociativa en su clase uno, entonces se dibuja una ventana para poder seleccionar esa parte. La tabla de bloqueados va a registrar los rangos de ventana donde la parte seleccionada del objeto clasificó en la clase uno de la memoria asociativa.

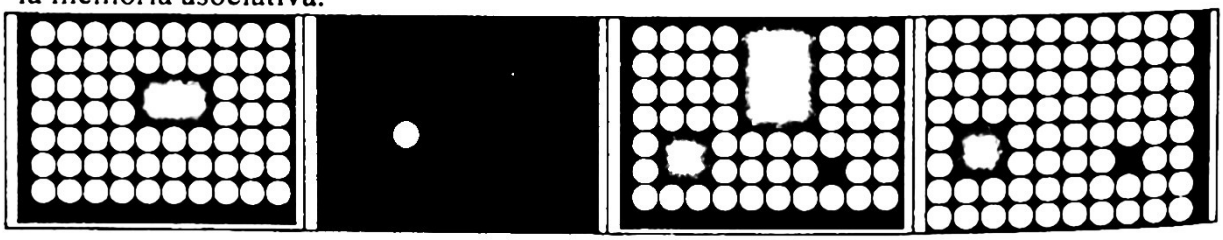


Fig. 1. Máscara con ventanas

A continuación se lista el algoritmo de *barrido bloqueante*:

1. *Inicializar tabla de bloqueados con ceros.*
2. *Dibujar máscara.*
3. *Hacer un AND de la máscara y la imagen a analizar.*
4. *Procesar la imagen para obtener los rasgos descriptivos de cada parte seleccionada del objeto.*
5. *Para cada parte seleccionada del objeto*
    a. *Cargar memoria asociativa del objeto*
        i. *Si los rasgos descriptivos de la parte del objeto son clasificados en la clase 1*
            1. *Registrar la región donde se localiza esa parte en la tabla de bloqueados*
        ii. *En caso contrario*
            1. *Continuar con la siguiente parte de objeto*
6. *Actualizar máscara e ir a 2*

4 Resultados Experimentales

En ésta sección el método propuesto para el reconocimiento de objetos traslapados es evaluado. En la figura 2 se muestran los objetos con los que las memorias asociativas se entrenaron.

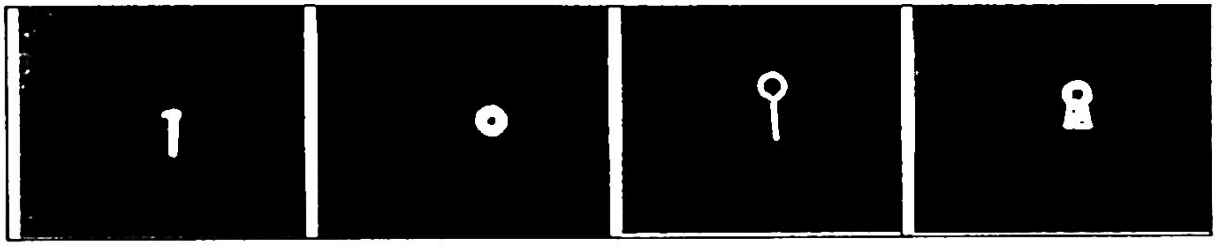


Fig. 2. Objetos usados en el entrenamiento de las memorias asociativas

Etapa de entrenamiento

Después de aplicar los métodos descritos en sección 3.1 se obtuvieron las memorias asociativas de cada uno de los objetos. En éste experimento se utilizaron los primeros cuatro momentos de Hu. A continuación se muestran algunas memorias asociativas obtenidas.

$$M_1 = \begin{bmatrix} 0.1742 & 0.0021 & 0.00052 & 4.97E-5 \\ 0.2331 & 0.0197 & 0.0025 & 0.0002 \\ 0.2912 & 0.0569 & 4.96E-5 & 9.60E-6 \\ 0.2683 & 0.0378 & 0.0038 & 0.0004 \\ 0.2220 & 0.0219 & 4.05E-5 & 5.35E-6 \end{bmatrix}$$

$$M_2 = \begin{bmatrix} 0.2094 & 0.0096 & 0.0003 & 8.88E-5 \\ 0.1956 & 0.0008 & 3.33E-7 & 2.25E-6 \\ 0.1954 & 1.61E-5 & 1.47E-6 & 2.90E-6 \\ 0.1952 & 8.00E-6 & 1.12E-6 & 3.88E-6 \\ 0.2436 & 0.0328 & 7.79E-5 & 2.60E-5 \end{bmatrix}$$

$$M_3 = \begin{bmatrix} 0.4405 & 0.0592 & 0.0205 & 0.0051 \\ 0.3721 & 0.0794 & 0.0126 & 0.0017 \\ 0.4622 & 0.1858 & 0.00018 & 0.00011 \\ 0.3749 & 0.1142 & 0.00011 & 4.90E-5 \\ 0.3152 & 0.0123 & 0.0320 & 0.0002 \end{bmatrix}$$

Etapa de recuperación

Un conjunto de imágenes se usó para evaluar el desempeño del sistema. Éste grupo consiste de 20 imágenes que están traslapadas, algunas imágenes de este grupo son mostradas en la figura 3.

El experimento consiste en determinar la existencia de algún objeto dentro de una imagen. En cada imagen se pide verificar si existe cada uno de los objetos con los que se trabajó. Dada una imagen, se verifica que la memoria asociativa de un objeto funcione de manera correcta. Es decir, si en el conjunto de imágenes de la figura 3 queremos encontrar los tornillos, al aplicar la memoria asociativa del tornillo nos tendría que indicar que reconoció tornillos; ahora, si queremos encontrar armellas, utilizamos la memoria asociativa de la armella esperaríamos que nos diera como resultado que no encontró ninguna armella en las primeras cuatro imágenes y que encontró una armella en la ultima imagen.

En total se hicieron cien pruebas en este experimento.

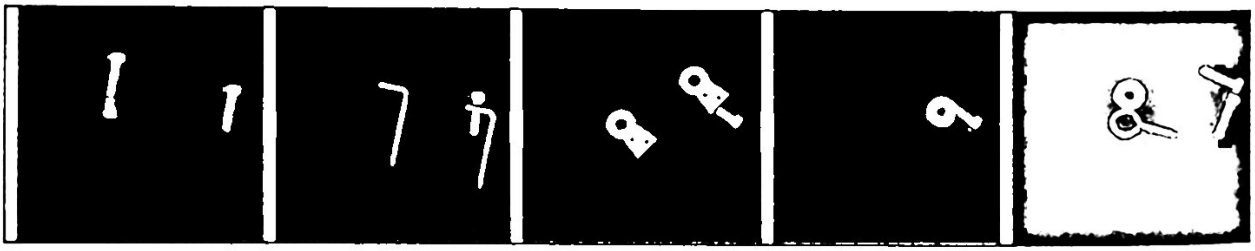


Fig. 3. Grupo de imágenes con objetos traslapados

En la tabla 2 se muestran los porcentajes de clasificación de cada objeto

	Tornillo	Rondana	Armella	Cola de pato
Porcentaje de clasificación	100%	60%	80%	80%

Tabla 2. Selección de parte usando la forma mas representativa.

Como se puede observar, el porcentaje de efectividad en la clasificación es del 80%.

5 Conclusiones e investigaciones futuras

El reconocimiento de objetos traslapados es una tarea difícil si se quieren usar descriptores por región para representar a un objeto. El método propuesto, a pesar de que entregó una eficiencia de clasificación del ochenta por ciento tiene sus deficiencias. Un primer problema que se presenta es que si la parte más representativa de un objeto está oculta, nunca se va a detectar el objeto. Otro problema que surge es cuando una objeto aumenta su tamaño, éste problema se debe a que ocupamos una ventana un tamaño definido y si un objeto cambia de tamaño no sabríamos en que proporción se tendría que aumentar la ventana.

Como próximo trabajo se tiene que lograr que éste método sea invariante a escalamientos. Otro posible trabajo sería la propuesta de una nueva metodología en la cual no exista la restricción de que la parte más representativa del objeto tenga que estar visible.

Finalmente se podría implementar alguna otra técnica para el reconocimiento de objetos traslapados que no involucre descriptores por región o memorias asociativas.

Referencias

[1] R. C. Gonzalez and R. E. Woods, Digital Image processing. Second edition. Prentice hall, Inc 2002

[2] N. Otsu, A threshold selection meted from gray-level histograms, IEEE transactions on SMC, 9(1):62-66, 1979.

[3] R. Jain et al. Machine vision, Mcgraw Hill, 1995

[4] M. K. Hu, Visual Pattern recognition by moment invariants, IRE Transactions on information theory, 8:179-187, 1962

[5] Cornelio Yánez Marquez, Diaz de Leon Santiago, Introducción a las memorias asociativas, IPN-CIC, 2003

[6] Ricardo Barrón, Humberto Sossa,Roberto A. Vazquez, A new associative model for the classification of real-value pattern, CIC-IPN, 2004

[7] Humberto Sossa, Ricardo Barron, Roberto A. Vázquez, New Associative Memories for the Recall of Real-Valued Patterns, CIC-IPN, 2004

[8] Karla P. Xicotencatl Aguilar, Tesis - Conteo de objetos en escenas ante deformaciones afines y proyectivas, CIC-IPN, 2001.

[9]Enrique López Paredes. Tesis - Seguimiento de objetos usando una cámara sobre una plataforma motorizada, CIC-IPN, 2001.

[10]Roberto A. Vazquez, Tesis - Localización de objetos 2-D a través de una plataforma motorizada usando análisis de imágenes, ESCOM-IPN, 2003

[11] Chanda, B., Application of binary mathematical morphology to separate overlapped objects, pp. 639-645.

[12] Stanley RJ, Keller J, Gader P, Caldwell CW. Homologue matching applications: recognition of overlapped chromosomes. *Pattern Analysis and Applications* 1(4): 206-217, 1998.

Digital processing of a ECG using a DSP: Wavelet approach[1]

S. Razo Zapata[2] and L.E. Ramos Velasco

ri096373@uaeh.reduaeh.mx, lramos@uaeh.reduaeh.mx

Centro de Investigación en Tecnologías de Información y Sistemas
Universidad Autónoma del Estado de Hidalgo
Carr. Pachuca – Tulancingo, km. 4.5 C.P. 42084, Pachuca, Hgo. México

Abstract. This work presents the digital processing of an electrocardiogram with the object of extracting useful information for its comprehension. The Wavelet technique is used for this porpouse. In order to perform the test a digital signal processor (DSP) TMS320C6711 Texas Instruments, was used.

Key words: Digital Processing, Electrocardiogram, Wavelets, DSP.

1 Introduction

Electrocardiogram (ECG) is a graphical register of the electrical potentials produced by cardiac tissue, which ones are located between a frequency range from 1 to 100 Hz. And the synchronization mechanism is shown on figure 1.

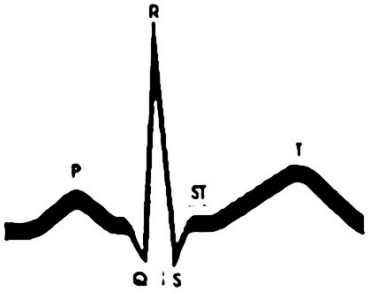


Fig. 1. Electrocardiogram elemental signal.

The bents presents in the drawing of an ECG represent different states of the heart during a beat. The *P wave* and the *QRS complex* indicates contraction of the atriums and ventricles respectively, the *ST segment* indicate the time that transcurse since a ventricle contraction ends to a starting rest period previous to the ventricles start to

[1] Supported by PROMEP with project number 103.5/03/1130

[2] Student of the Bachelor's Degree Program in Computer Sciences, at the Universidad Autónoma del Estado de Hidalgo, Mexico.

contract themselves for the next beat, and finally the *T wave* indicate the rest period of the ventricles.

The microelectrical signal analysis has been strongly used to determine different kinds of medical malaises and diseases such as muscular fatigue, cardiac disfunctions, etc. Y thanks to the advance of this area, actually it is possible to realize medical diagnostics.

The most used methods for the study of microelectrical signals are those that make use of Fourier Transform, but this ones have certain limitations, e. g. The fact that are only appropriated for stationary signals, which means that are not adapted in the signal analysis with transitory components in time, such as microelectrical signals.

Recently, time-scale methods (based on the Wavelet Transform) were proposed in an effort to superate the limitations of the traditionals time-frequency based ones. This kind of methods act as a math microscope that can be used to observ diferent components by only adjusting the wavelet focus [12].

This allows component discover of short duration in a signal. Between this methods are the generalizations of the Wavelet Transform, the banks or Wavelets paquets (WP) that allows a better adapted analysis of a signal.

Wavelet Transform actually is beeing used in an important manner in several medical applications [1], besides that new techniques that allows an adecuation in front of several applications cases are continuously introduced. Inside medical applications where Wavelet Transform is used as analysis technique, are:

- Echocardiogram generated data compression.
- Magnetic resonances.
- Electrocardiograms.
- Others.

The work is organized in the following way. Section 2 presents the math tools used to realize the digital processing of the signal of the ECG, section 3 shows the filter design used in the noise elimination from the signal to study, section 4 shows the results obtained in real time, and finally on section 5 are located the accomplished conclusions from results shown during work development.

2 Wavelet Transfrom (WT)

Wavelet theory provides a unified framework for a number of techniques which had been developed independently for various signal processing applications. For example, multiresolution: used in computer vision, subband coding: developed for speech and image compression, and wavelets series expansion, developed in applied mathematics, have been recently recognized as different views of a single theory [12]. In particular, the Wavelet Transform (WT) is of interest for the analysis of non-stationary signals, because it provides an alternative to the classical Short-Time

Fourier Transform (STFT). The basic difference is as follows. In contrast to the STFT, which uses a single analysis window, the WT uses short windows at high frequencies and long windows at low frequencies, as one is in the figure 3.

2.1 What Is a Wavelet?

The term Wavelet means little wave. In the most general context, a wavelet is a function with special properties and that satisfies the main (time domain) conditions:

1. it has a small concentrated burst of finite energy in the time domain; and
2. it exhibits some oscillation in time.

In addition a wavelet have a limited duration and average value equal to zero, it is possible to be seen in figure 2.

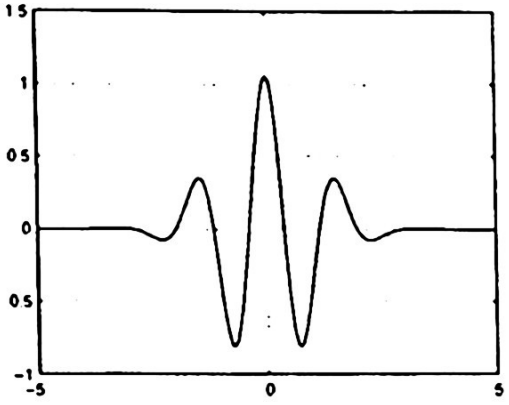


Fig. 2. Example of Wavelet.

2.2 Continuous Wavelet Transform

When a function *f(t)* is scaled:

$$f(t) \rightarrow f(at); \text{ donde } a > 0$$

for the constant a > 1 the function is contracted, and expanded if a < 1, using this basic idea of calculus, next the formal definition of the *Wavelet Continuous Transform* is showed.

Definition 1

$$CWT_f(\tau, \alpha) = \frac{1}{\sqrt{\alpha}} \int f(t)\psi(\frac{t-\tau}{\alpha})dt \qquad (1)$$

In this part a function $\psi(\tau, \alpha)$ is generated, this function depends of the parameters τ and α, where both are used to transfer and to climb respectively.
Changing t by αt in (1), yields

$$CWT_f(\tau, \alpha) = \sqrt{\alpha} \int f(\alpha t)\psi(t - \frac{\tau}{\alpha})dt \tag{2}$$

The interpretation of (2) is that as the scale grows, an increasingly contracted version of the signal is seen through a constant length filter. From (1) and (2) is possible to be concluded that the Wavelet Transform is the sum over all time of the signal multiplied by scaled, shifted versions of the wavelet ψ. Multiplying each coefficient by the correct scaled and shifted wavelet we obtain wavelets that compose the original signal. The process implied in the Transformed Wavelet is in figure 3.

It's important to understand that the scale is related to the size of the signal, while resolution is linked to the amount of detail present in the signal.

For many signals, the low-frequency content is the most important part, like EKG's. It is what gives the signal its identity. The high-frequency content, on the other hand, imparts flavor or nuance.

In wavelet analysis, we often speak of approximations and details. The approximations are the high-scale, low-frequency components of the signal. The details are the low-scale and high-frequency components.

2.3 Discrete Wavelet Transform (DWT)

In general, discrete wavelet transforms are generated by samplings (in the time-scale plane) of a corresponding continuous wavelet transform. Such a discrete wavelet transform is specified by the choice of items:

1. a time-scale sampling set(a countable set of points), and

2. an analyzing wavelet, in this case was used the Daubechies wavelet.

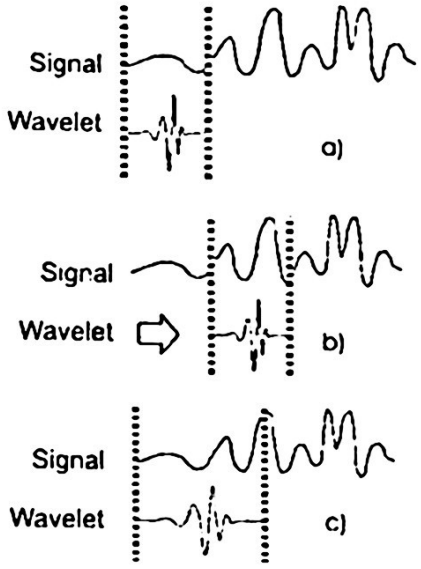


Fig. 3. Wavelet Transform a) Contraction $\alpha > 1$ b) Shifting with parameter τ c) expansion $\alpha < 1$.

Daubechies Wavelet. This family includes the Haar wavelet, written db1, the simplest wavelet imaginable and certainly the earliest. The names of the Daubechies family wavelets are written *db*N, where N is the order, and *db* the "surname" of the wavelet. In (3) one can see the definition of the family.

$$\psi = 2^{j/2}\varphi(2^j x - k) \quad j, k \epsilon Z \tag{3}$$

In (3), 2^j is the scaling factor and k the function's localization or the shifting factor; figure 4 is the graphic representation of the Daubechies wavelet.

In the discrete time case, two methods broadly used in the digital implementation were developed, these methods are:

- Subband coding.
- Pyramidal coding or multiresolution signal analisys.

2.4 Subband Codding

Calculating wavelet coefficients at every possible scale is a fair amount of work, and it generates an awful lot of data. And for this reason the Subband coding schema choose scales and positions based on powers of two – so called dyadic scales and positions–.

An efficient way to implement this scheme using filters was developed in 1988 by Mallat [13]. The Mallat algorithm is in fact a classical scheme known in the signal processing community as a two channel subband coder. This filtering algorithm yields a fast wavelet transform.

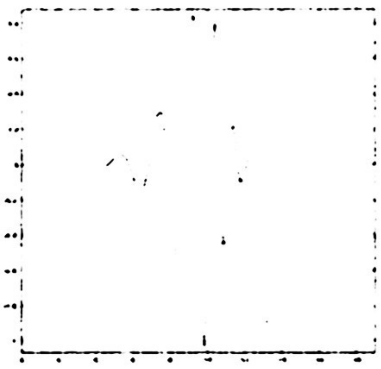


Fig. 4. Example of Daubechies Wavelet

The filtering process, at its most basic level, is showed in the figure 5. The original signal, S, passes through two complementary filters and emerges as two signals A y D. Unfortunately, if we actually perform this operation on a real digital signal, we wind up with twice as much data as we started with.

However, exists a more subtle way to perform the decomposition using wavelets. The Downsamplig allows keep only one point out of two to get the complete information. So are produce two sequences called cA and cD figure 6.

Example. To gain a better appreciation of this process, in the figure 7 our signal will be a pure sinusoid with highfrequency noise added to it, by means of the filtering process with downsampling an approach the natural form of the signal and the components in high frequency (noise) are observed as well as the reduction in the amount of data.

The decomposition process can be iterated, with successive approximations being decomposed in turn, so that one signal is broken down into many lower resolution components. This is called the Wavelet Decomposition Tree showed in the figure 8.

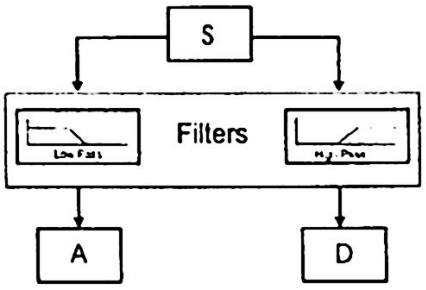


Fig. 5. Filtering Process.

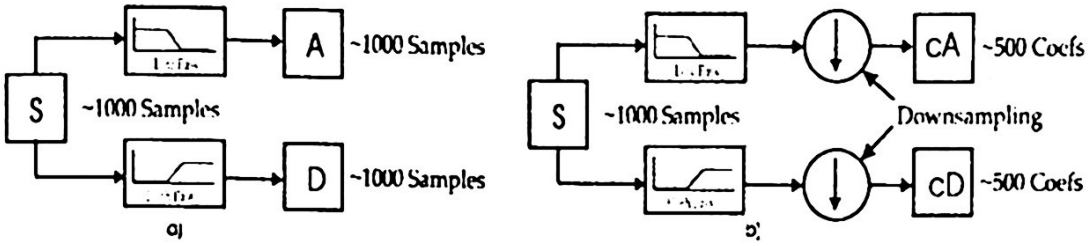


Fig. 6. a) Process without downsampling b) Process which includes downsampling.

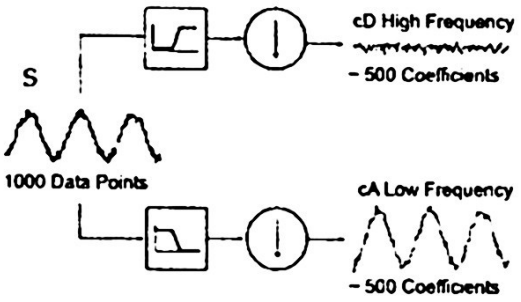


Fig. 7. Example of filtering process with downsampling.

2.5 Pyramidal Codding

This scheme was introduced by Burt and Adelson [15] and was recognized by the wavelet community to have a strong connection to multiresolution analysis, It consists of deriving a low-resolution version of the original, then predicting the original based on the coarse version, and finally taking the difference between the original and the prediction (figure 9). At the reconstruction, the prediction is added back to the difference, guaranteeing perfect reconstruction.

Obviously, the scheme can be iterated, decomposing the coarse version repeatedly, to obtain a coarse version at level J plus J detailed versions.

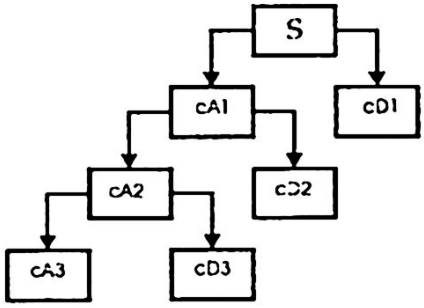


Fig. 8. Wavelet Decomposition Tree.

3 Filters Design

The processing of the ECG signals, was made in two steps, first of them consisted in filtrate stage with the intention of eliminating the possible presence of noise in the signal. The later stage consisted of applying the Transformed Discreet Wavelet to the filtered signal using the Tree of Wavelet Decomposition like analysis scheme, wavelet of analysis was the wavelet of Daubechies 10.

For the case of the signal of an Electrocardiogram, the rank of frequencies in where is the information of importance for its study is located between 1 and 100 Hz.

It is so the designed filter is a low pass Butterworth with cut frequency of 100 Hertz, in the equation (4) is observed the function of transference in s of the filter, in addition in the equation (5) its corresponding function of transference in z can be seen.

$$H(s) = \frac{248050000}{s^3 + 1256.6s^2 + 789570s + 248050000} \tag{4}$$

$$H(z) = \frac{0.17843z^3 + 0.53529z^2 + 0.53529z + 0.17843}{z^3 + 0.084523z^2 + 0.33522z + 0.0076958} \tag{5}$$

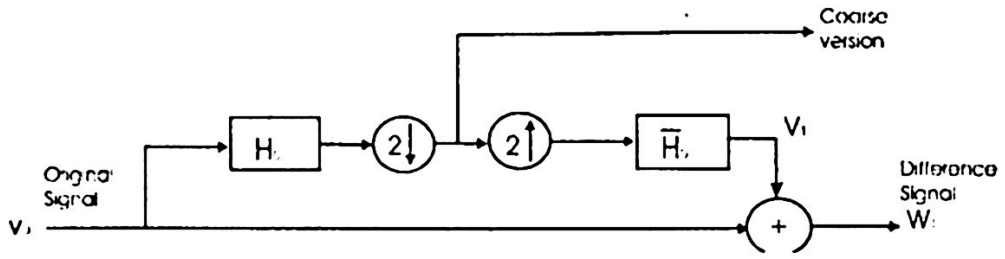


Fig. 9. Pyramid Scheme involving a coarse lowpass approximation and a difference between the coarse approximation and the original.

Finally the frequency response by Bodes Diagram of the filter is observed in figure 10. This filter was designed with commands that MATLAB has for that purpose. The used code is the following one:

```
% -- Analog Filter
[NumB,DenB] = butter(3,2*pi*[100],'s');
Bode(NumB,DenB);
% -- Digital Filter
Fs = 300; % -- Sampling frequency
[NumBd,DenBd] = bilinear(NumB,DenB,Fs);
```

4 Experimental results

For the real time implementation of this algorithms was used a DSP of the family of TMS320C67x devices [14], which permitted to process the data of an ECG that previously was obtained from the PhysioNet's network database[3].

The test results made to the filter with the signal of an Electrocardiogram to which has added him noise, is in figure 11.

For the Analysis schema was determined by empirical tests made in simulation to arrive at a decomposition level equal to four (4); the electrocardiograms are produced by the cardiac synchronization mechanism figure 1, that is tried to be located in this process.

The final result produced by the analysis schema is showed in figure 12, which consists in the ECG signal and the last two branches of decomposition tree. The figure 12.a corresponds to the signal previously filtered of the ECG; the figure 12.b talks about to the approach of this signal to a level 4, in this figure can be observed which is the natural shape of the signal, the part corresponding to the low frequencies (in great scales).

On the other hand the figure 11.c corresponds to the details of the signal (also at a level 4), information in high frequencies, in this part are forms of the signal that are not appreciable at simple view (in short scales).

5 Conclusions

The digital signal processing is a scientific field that can be explored from different perspectives, in this article has been presented a modern technology of digital processing used indifferently in images and signals. Also the algorithm of the Transformed Wavelet has been implemented on a DSP TMS320C6711 for the processing of the signal of an Electrocardiogram.

---

[3] www.physionet.org

When taking advantage of the multiresolution capacity of this transformed was extracted information of the signal that can be of utility in other applications.

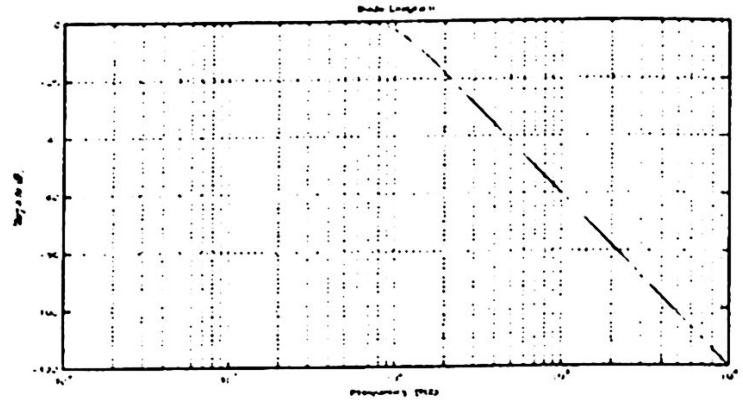


Fig. 10. Bode's diagram for low pass Butterworth filter, with cut frequency w_c = 100 Hz.

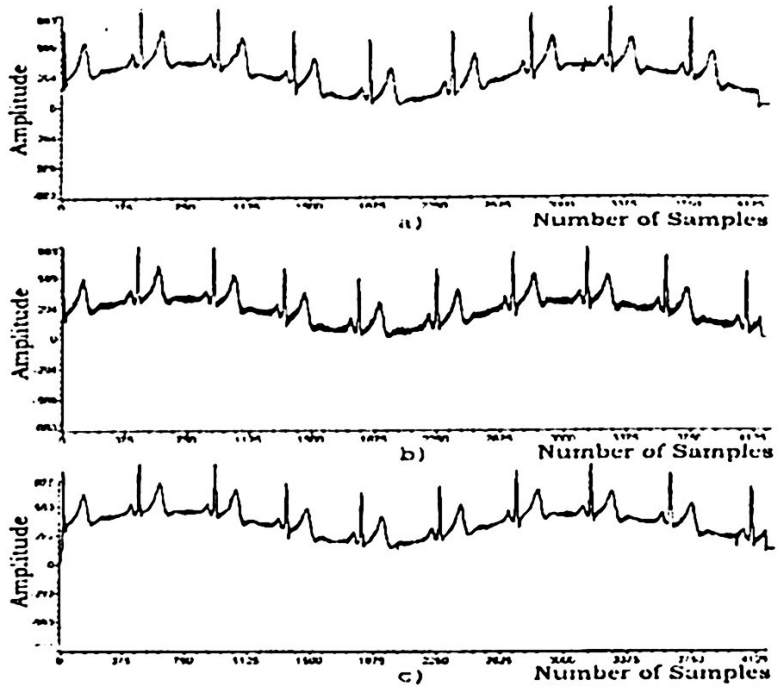


Fig. 11. a) ECG signal, b) ECG signal with noise, c) ECG signal filtered with the Buttertworth low pass filter.

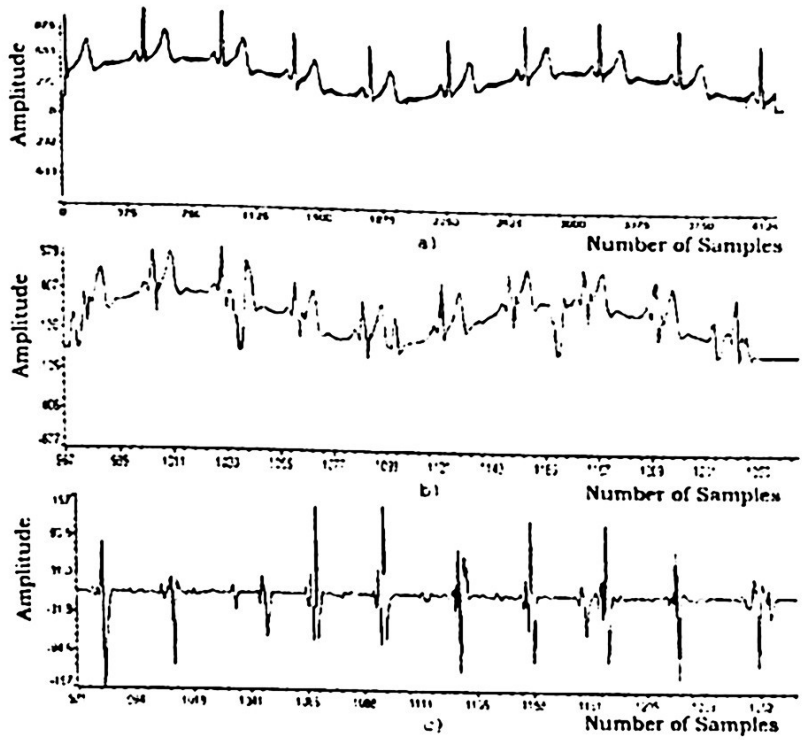


Fig. 12. a) ECG filtered signal, b) Approach of the ECG signal to a level 4 with the Wavelet of Daubechies 10, c) Derivate at a level 4.

References

1 Akay Metin, Mello Claudia, *Wavelets for Biomedical Signal Processing*, 19th International Conference, Chicago, 1997.

2 Ashok Ambadar, *Procesamiento de Señales Analogicas y Digitales*, Thomson Learning, 2002.

3 Bracewell, R.N., *The Fourier Transform and Its Applications*, McGraw Hill 1978.

4 Chassaing Rulph, *Digital Signal Processing, Laboratory Experiments Using and the TMS320C31 DSK*, John Wiley and Sons, Inc.

5 Gabel Robert A., Roberts Richard A., *Señales y Sistemas Lineales*, 1975.

6 Hsu Hwei, *Análisis de Fourier*, Addison Wesley Iberoamericana, 1987.

7 Kehtarnavaz Nasse, Simsek Burc, *C6X Based Digital Signal Processing*, 1977.

8 McMIlan Claude, González Richard F., *Análisis de Sistemas*, 1977.

9 Oppenheim Alan y Willsky Alan, *Señales y Sistemas*, Prentice Hall, 1983.

10 Proakis John y Manolakis Dimitris, *Tratamiento Digital de Señales: Principios, algoritmos y aplicaciones*. Prentice Hall 1998.

11 Papoulis A., *The Fourier Integral and Its Applications*, McGraw Hill, 1962.

12 Rioul Olivier, Vetterli Martin, *Wavelets and Signal Proccessing, IEEE SP Magazine*, pp. 14-38, 1991.

Identificación de objetos por radio frecuencia

Sergio Sandoval Reyes

Centro de Investigación en Computación
Instituto Politécnico Nacional
Av. Juan de Dios Bátiz s/n C. P. 07738, México, D. F.
Tel. 5729-6000 Ext. 56502, e-mail sersand@cic.ipn.mx

Abstract: This paper describes the technology of radio frequency identification (RFID), and its applications. Using a set of objects--each one with a small tag carrying a transmitter-receiver and a processor with identification data in memory--and a radio-frequency reader, when the tag is under the read range of the reader, it responds transmitting the identification data to a data base which shows the object description.

Resumen: El presente artículo describe la tecnología y una aplicación de identificación de objetos por radio frecuencia (RFID) A cada objeto se le adhiere una pequeña etiqueta conteniendo un radio transmisor, un procesador, y memoria donde se almacena información de identificación; cuando la etiqueta es excitada por ondas de radio-frecuencia provenientes de una lectora, los datos de la etiqueta son transmitidos vía la lectora, a una base de datos la cual muestra la descripción del objeto.

Palabras Clave: RFID, Tag-it, tag-ISO, Reader S6350.

1. Introducción.

La identificación por radio frecuencia es una de las tecnologías de mayor crecimiento reciente, para la recolección automática de datos de identificación.

Algunos expertos consideran a la RFID como un complemento de la tecnología de código de barras—en muchos casos como en el almacenaje de cajas, se emplean ambas tecnologías. La tecnología RFID, de hecho, supera ciertas limitaciones del código de barras. Una de ellas es la de no requerir línea de vista entre la lectora y el objeto etiquetado con RFID. Esto es porque RFID transmite datos en forma inalámbrica y además puede actualizar o cambiar los datos codificados en la etiqueta.

En este artículo se describe brevemente la tecnología RFID, las características de sus componentes, así como una aplicación de dicha tecnología para la identificación de objetos usando una base de datos.

2. Tecnología RFID.

La tecnología RFID se compone de etiquetas y lectoras. Las etiquetas RFID consisten de un diminuto circuito integrado (CI) adherido a una antena—formada por material conductor gravado en una hoja de plástico delgada. Los datos de identificación son almacenados en el CI y transmitidos a través de la antena. Las etiquetas RFID pueden ser pasivas (sin batería) o activas (energizadas por batería). La velocidad y alcance de la transmisión de datos dependen de la frecuencia usada, tamaño de la antena, potencia de salida, e interferencia. Las etiquetas pueden ser de solo lectura, lectura-escritura, o combinadas, en la que cierta información (tal como el número de serie) es almacenada permanentemente, mientras que otra se le deja en memoria para una codificación posterior, o para ser actualizada durante su uso [1], [2]. La información es enviada y leída de etiquetas RFID mediante señales de radio-frecuencia. En sistemas pasivos, una lectora de RFID transmite un campo de energía que activa a la etiqueta y le proporciona alimentación al CI, habilitándolo para transmitir o almacenar datos. Por contraste, las etiquetas activas pueden transmitir periódicamente una señal-faro, de tal forma que los datos pueden ser capturados por lectoras distribuidas a lo largo de una instalación. Para proteger las transmisiones y proporcionar seguridad e integridad de los datos viajando entre la etiqueta y la lectora, se emplea encriptado.

Las lectoras pueden ser integradas en PDAs (tipo Palm), o permanecer fijas y posicionadas en puntos estratégicos, tales como entradas, puertas, o en una línea de ensamble. Las lectoras pueden incluir una o más antenas para enviar y recibir señales de las etiquetas, así como un procesador para decodificar señales y datos recibidos. Dichos datos son luego pasados vía cable, o mediante una red local inalámbrica, a una computadora (Fig. 1).

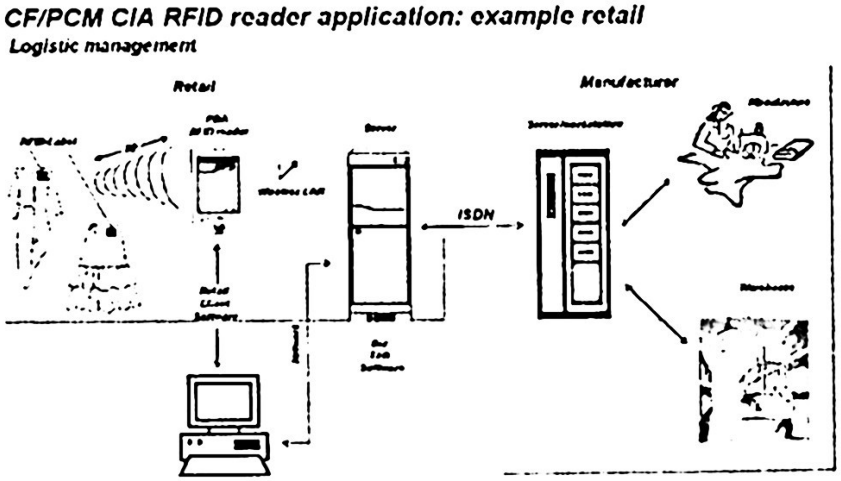


Fig. 1 Componentes de una aplicación RFID

Sin embargo, la tecnología RFID también presenta varios problemas técnicos. Los más importantes se indican a continuación [3]:

- Las ondas de radio de radio de un dispositivo podrían interferir con las de otro; los métodos de prevención incluyen el uso de algoritmos anti colisión, que aseguran que las etiquetas o las lectoras no transmitan al mismo tiempo.
- Una lectora podría encontrar datos de la etiqueta corruptos o faltantes, resultando en falsas identificaciones; para contrarrestar esto las etiquetas incorporan códigos de corrección de errores.
- Alcance de lectura. Este problema se puede superar seleccionando etiquetas de la frecuencia apropiada: las de baja frecuencia (125 KHz y hasta un 1 m de alcance) son las menos sujetas a interferencia; las de alta frecuencia (13.56 MHz y hasta 3 m de alcance) leen más rápido pero consumen más energía; las de ultra alta frecuencia (866-930 MHz y 5.8 GHz) ofrecen más alcance y altas tasas de transferencia, pero a estas frecuencias objetos con alto contenido acuoso, podrían bloquear las transmisiones.
- Multitud de estándares: Texas Instruments *Tag-it*, Phillips *I-CODE*, Inside Technologies *PicoTag*, etc.; a este respecto la ISO ha creado un conjunto de estándares tales como la ISO 15693 para 13.56 MHz, y la ISO 18000 para la banda 860-930 MHz.

2.1 Aplicaciones de la RFID

Aunque la tecnología RFID todavía es muy nueva, actualmente se están desarrollando aplicaciones en diversas áreas. A continuación algunos ejemplos:
Logística--para leer el contenido de productos en plataformas de carga, durante su carga o descarga.
Almacenaje—para detectar la ubicación de almacenaje de productos.
Farmacéutica—para que personas invidentes puedan "leer" el contenido de frascos de medicina.
Hospitales—mediante pulseras para identificación de pacientes así como de recien nacidos.
Manufactura—para automáticamente enrutar productos a través de los procesos de ensamble.
Librerías y video-tiendas—para detección de robo, inventarios y localización de productos desubicados
Seguridad—para identificación y control de acceso
En las escuelas---en las escuelas la identificación mediante RFID se está empleando para el manejo de libros, revistas, y otros materiales tales como videos, así como para la reubicación de dichos materiales fuera de lugar [3].

3. Descripción de la Aplicación

La aplicación que a continuación se describe está orientada a detectar e identificar diversos tipos de objetos, los cuales incorporan una etiqueta RFID. La aplicación se dividió en tres partes: 1) Características del lector y etiquetas disponibles; 2) Comunicación entre la lectora y la computadora; y 3) Implementación de comandos para leer y escribir información en la etiqueta, así como para accesar una base de datos con la descripción del objeto detectado.

3.1 Características del lector y etiquetas RFID

Disponemos del kit RI-K10-001A de Texas Instruments el cual incluye una lectora *S6350* (con interfaz RS232) con un alcance de lectura de hasta 18cms, operando a una frecuencia de 13.56 MHz. El kit incluye además, un conjunto de etiquetas *tag-it*, e *ISO 15693* de varias medidas (Fig. 2).

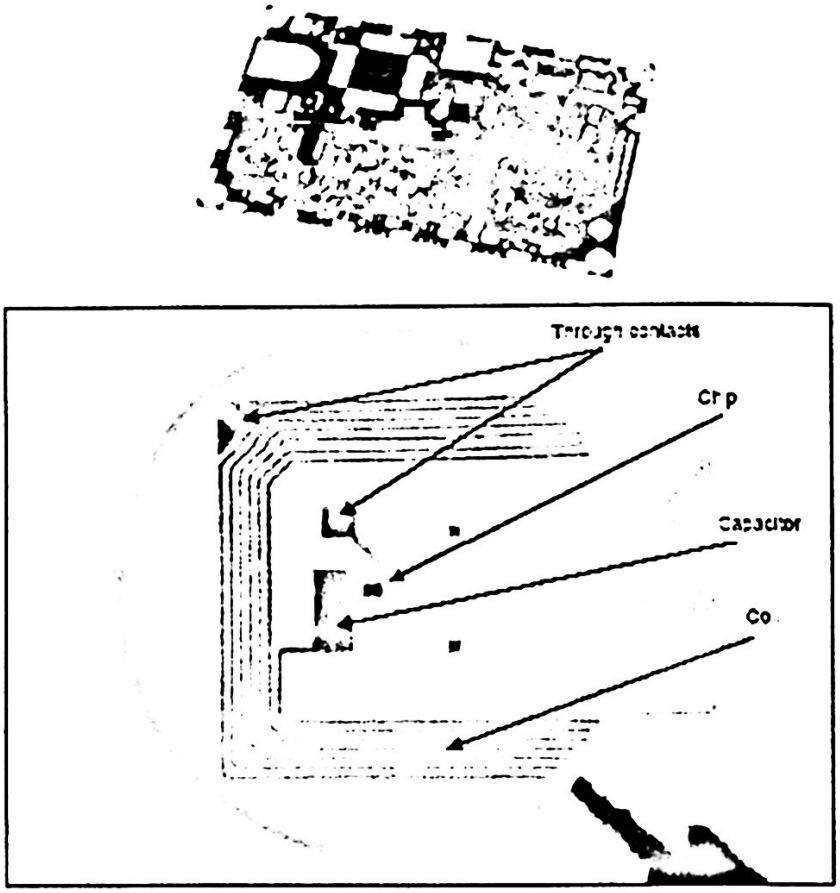


Fig. 2 Lectora S6350 y etiqueta Tag-it.

Las etiquetas vienen pregrabadas con un número de identificación de solo lectura SID, así como 256 bits de memoria no-volátil, arregladas en 8 bloques de 32 bits (Fig. 3). Los bloques pueden ser programados en forma temporal (modo unlock), permanente (modo lock) [4].

Aunque el kit incluye un programa de demostración para evaluar las capacidades de lectura y escritura de las etiquetas, desafortunadamente no incluye los archivos fuente para experimentar. Sin embargo se encontró un programa manejador de comunicaciones para la lectora TI 6000 desarrollado por Liu [5], el cual se utilizó con modificaciones.

...

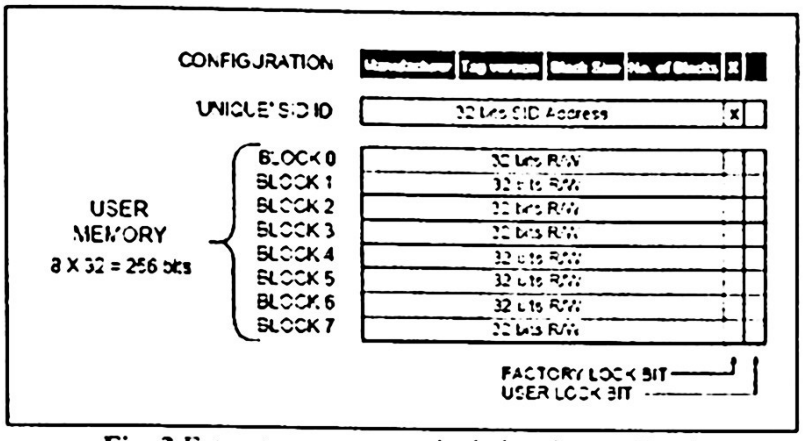


Fig. 3 Estructura en memoria de la etiqueta Tag-it

3.2 Comandos de la lectora

Los paquetes enviados de computadora a lectora siguen el siguiente formato [6]:

01_{hex}	Inicio de Trama
$0A_{hex}$	Byte menos significativo de longitud-paquete
00_{hex}	Byte más significativo de long. del paquete
00_{hex}	Byte menos significativo de dirección-nodo
00_{hex}	Byte más significativo de dirección-nodo
00_{hex}	Banderas de comando(sin direccionado)
02_{hex}	Comando (ejemplo: Leer bloque Tag-it)
01_{hex}	Datos (ejemplo: bloque número 1)
08_{hex}	Byte menos signif. de verific. de suma(XOR)
$F7_{hex}$	Byte más signif. complemento de suma(NXOR)

Los paquetes de respuesta siguen el mismo formato, excepto en el número de bytes (de 0 a 23) en el campo de datos, dependiendo del tipo de comando.

Definición de Comandos para etiquetas Tag-it

Lectura de un bloque	02_{hex}
Escritura no permanente (unlock) de un bloque	03
Escritura permanente (lock) de un bloque	04
Lectura de detalles de la etiqueta	05
Lectura de múltiples bloques	0F
Lectura de versión firmware de lectora	F0

Ejemplos:
1) Versión del firmware de la lectora
Paquete de solicitud:
01 0900 0000 00 **F0** $F807_{hex}$
Paquete respuesta:
01 0C00 0000 00 F0 40 01 07 $BB44_{hex}$
Explicación: La versión es la 1.4 cargada exitosamente (07).

2) Lectura de los detalles de una etiqueta Tag-it no direccionada (es decir, sin especificar el SID)

Paquete solicitud:
01 0900 0000 00 **05** $0DF2_{hex}$
Paquete respuesta:
01 1200 0000 00 05 A4340100 01 0500 08 04 $8F70_{hex}$
Explicación (ver campos de la figura 3):

SID de etiqueta	$000134A4_{hex}$
Identificación de fabricante	01
Número de versión	0005
Número de bloques	08
Número de bytes por bloque	04

3.4 Base de datos

Como se mencionó anteriormente, las etiquetas de RFID incorporan una identificación SID la cual no puede ser modificada, así como 8 bloques de memoria de 32 bits programadas por usuario en código hexadecimal. Como dicho código no permite representar otro tipo de caracteres, se decidió en la primera versión de esta aplicación, emplear el código SID para la identificación de objetos con etiquetas RFID. Con dicha identificación y empleando una base de datos, se puede incorporar la descripción del objeto en cuestión. La base de datos incorporara la siguiente información:

	AutoNumero	ID	Hex	descripcion	inventario	dentro
▶		GDO	11C473A	libro	☑	☑
	182	AjxD	1786AC4	revista	☑	☑
*	(Autonumérico)				☐	☐

Fig. 5 Tabla de la base de datos

4. Implementación

El sistema de detección de objetos aparece en la figura 6, la cual muestra el arreglo de la lectora y la etiqueta de RFID, un menú de *Configuración*, *Versión* y *Ayuda*, así como los botones *Iniciar*, *Detectar etiqueta*, *Agregar etiqueta* a la base de datos y *Salir*. Al seleccionarse *Versión* en el menú, se ejecuta el siguiente código:

```
While (Len (stròut) <> 12)
MSComm1.Output = Chr(1) & Chr(9) & Chr(0) & Chr(0) & Chr(0) & Chr(0) &
Chr(240) & Chr(248) & Chr(7)
Strout =MSComm1.Input
Wend
```

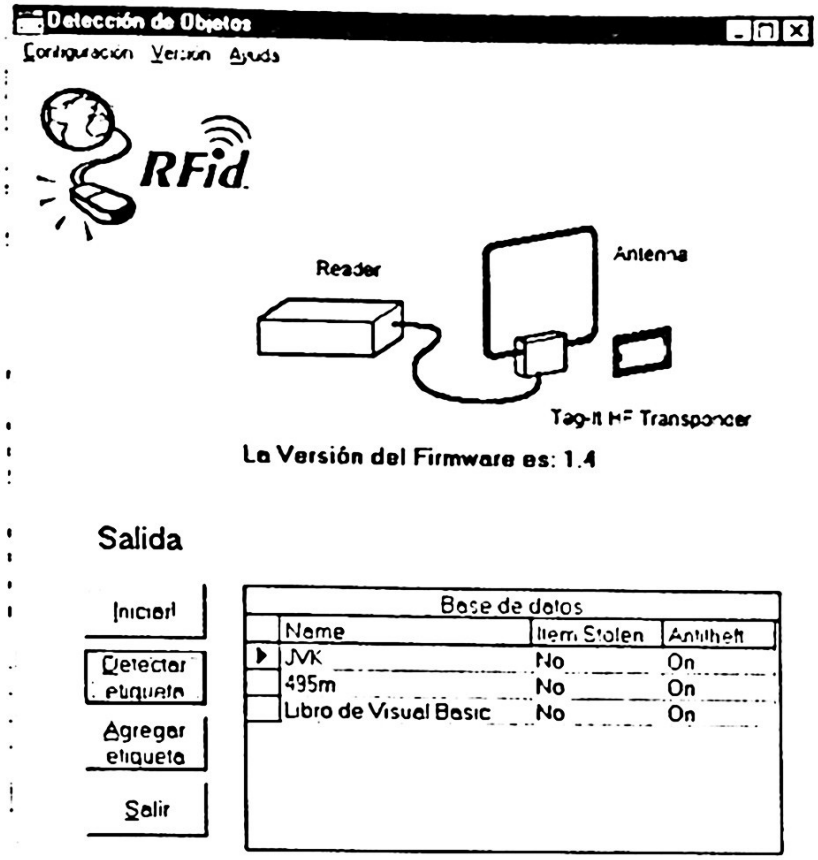


Fig. 6 Sistema de detección de objetos por RFID.

En dicho código los caracteres Chr (1), Chr(9), Chr(0)..., Chr(240), Chr(248) y Chr(7), son equivalentes al paquete de solicitud (ejemplo 1 de la sección 3.3), para obtener la versión 1.4 del firmware de la lectora (figura 6).

De manera similar al oprimir el botón *Detectar etiqueta* para conocer el SID de la misma, se ejecutaría el siguiente código:

MSComm1.Output = Chr(1) & Chr(9) & Chr(0) & Chr(0) & Chr(0) & Chr(0) & Chr(05) & Chr(13) & Chr(242)
strout =MSComm1.Input

En dicho código los caracteres Chr (1), Chr(9), Chr(0)..., Chr(05), Chr(13) y Chr(242), son equivalentes al paquete de solicitud (ejemplo 2 de la sección 3.3), para obtener el SID de la etiqueta.

Una vez detectado el objeto y si a continuación se oprime el botón *Agregar etiqueta*, para agregar el mismo a una base de datos, se ejecuta el siguiente código:

```
Private Sub Write_Button_Click()
Dim conn As New ADODB.Connection
Dim rs As New ADODB.Recordset
Dim strquery As String
Dim entrada_usuario As String
```

```
Entrada_usuario=InputBox("Por favor de el nombre del objeto")
If Len(entrada_usuario) > 0 Then
   conn.Open StrConnection
   conn.Execute "INSERT INTO base_de_datos(ID, Hex, descripcion, inventario,
dentro) VALUES (' " &tag1 & tag2 & "', '" & tag22 & tag11 & "', & "', '" &
entrada_usuario & "' True,True")
status.Caption=entrada_usuario & "Agregado a objetos detectados"
Conn.Close
End If
   Adodc1.Refresh
End Sub
```

5. Pruebas y Resultados

Para validar los resultados se empleó el programa de demostración de la lectora S6350, que como se indicó anteriormente, permite evaluar las capacidades de lectura y escritura, así como conocer la versión del firmware. Asimismo dicho demo trae integrado un "husmeador" de red, el cual muestra los paquetes de solicitud y de respuesta (figura 7).

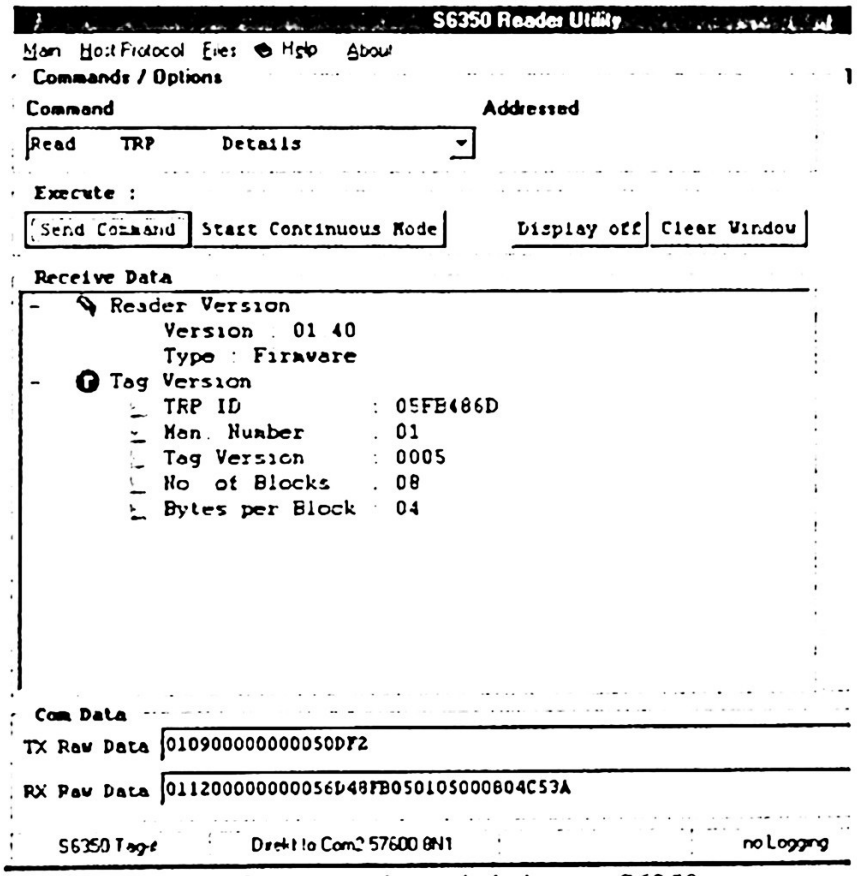


Fig. 7 Programa demo de la lectora S6350

En la figura 7 se puede observar que la versión del firmware cargado es la 1.4, y que la SID de la etiqueta Tag-it bajo lectura es *5FB486D*. Asimismo se observan los paquetes de solicitud, como el de respuesta. La misma identificación SID puede

apreciarse en la figura 8, cuando se hace clic en el botón *Detectar etiqueta,* en el programa Detección de Objetos.

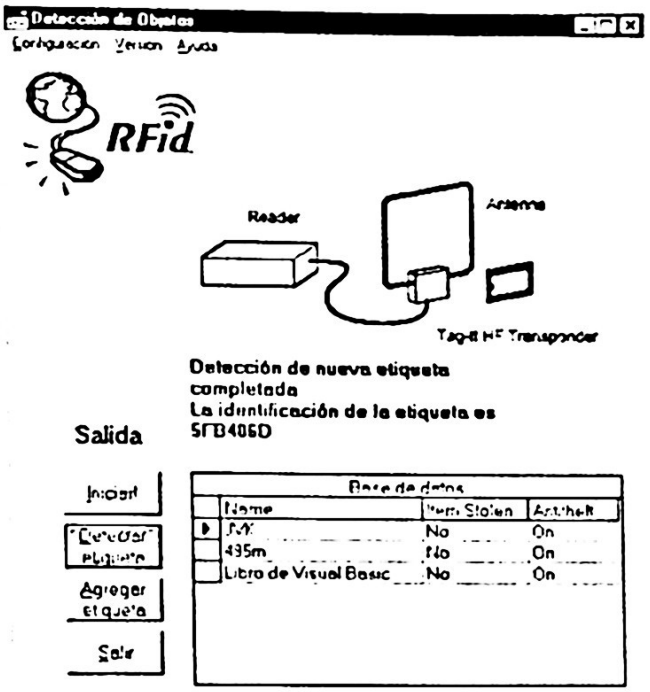


Fig. 8 Detección del SID de una etiqueta Tag-it.

Una vez detectada la etiqueta SID del objeto, éste puede incorporarse a una base de datos *Access*, al oprimir el botón *Agregar etiqueta*. Al hacerlo, aparecerá una ventana en la que se pide al usuario que de una descripción del mismo (figura 9).

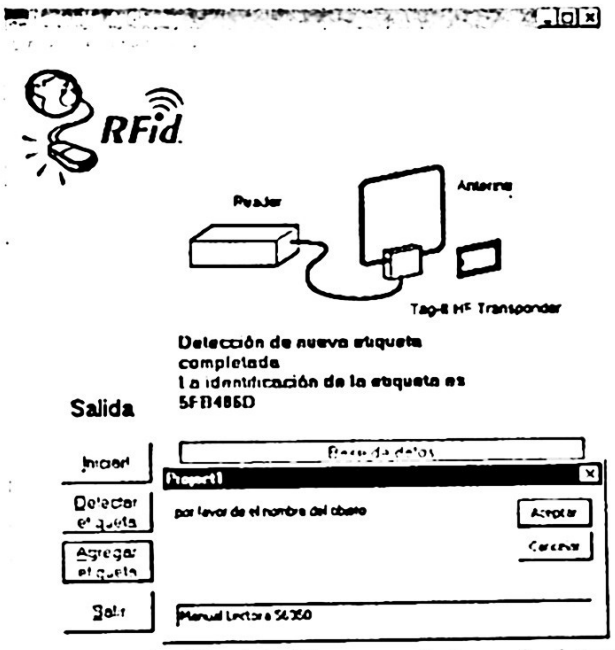


Fig. 9 Descripción del objeto para la base de datos.

Al hacer clic en Aceptar, el objeto quedará incorporado a la base de datos (figura 10).

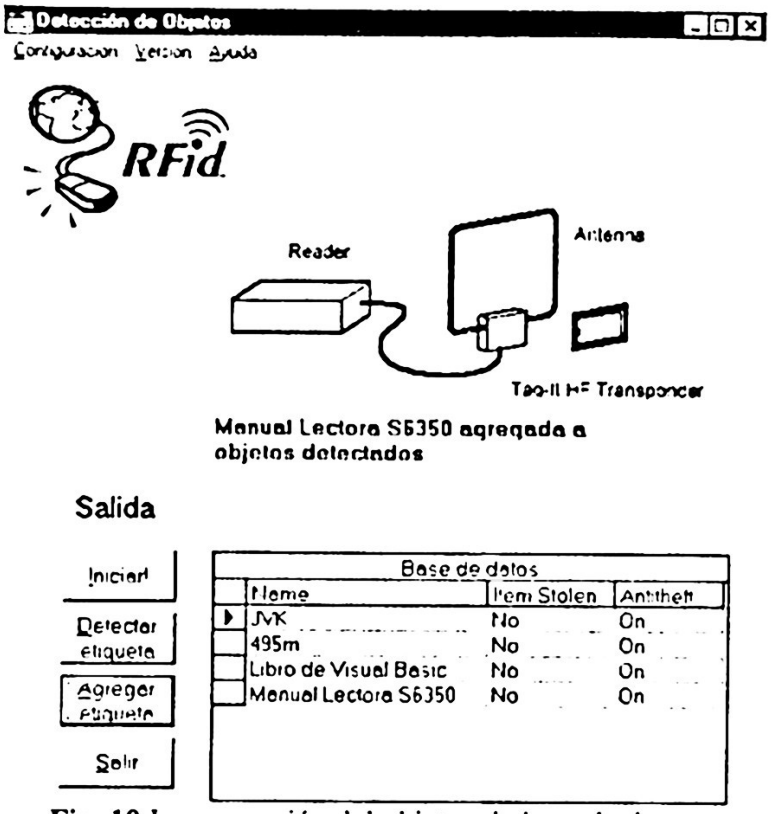


Fig. 10 Incorporación del objeto a la base de datos.

6. Conclusiones

En el presente artículo se mostró la aplicación de la tecnología de identificación de objetos por radio frecuencia, mediante un sencillo ejemplo. Los resultados obtenidos fueron satisfactorios en cuanto a funcionalidad y tiempo de respuesta de la aplicación, aunque el área de identificación de la lectora (18 cm.) requiere que el objeto esté muy cercano. Además, un fenómeno que se observó cuando a la lectora se le aproximaban dos objetos con sus respectivas etiquetas Tag-it, fue que la lectora no podía leer simultáneamente las SIDs de las mismas. Es decir, se producía un efecto de colisión de paquetes. Al parecer solo las etiquetas que siguen el protocolo *ISO 15693* pueden detectar esto y corregirlo. En la siguiente versión del programa incluiremos este tipo de etiquetas.

Actualmente, este tipo de tecnología está empezando ha ser explotado por el sector empresarial en México. Tal es el caso de la empresa *Mario Millar* que controla las entradas y salidas del personal, incrustando una etiqueta RFID en la suela de un zapato, para identificar nombre y número de empleado, así como el área al que corresponde [7]. Mientras más aplicaciones se desarrollen como la anteriormente mencionada, la tecnología de RFID tendrá en el mercado un impacto similar o superior a su antecesora, la identificación por código de barras.

Referencias

[1] Zebra Technologies, "The Next Generation of RFID". Artículo de Aplicación, 2004. http://www.rfid.zebra.com/.

[2] Zebra Technologies, "Zebra's RFID Readiness Guide: Complying with RFID Tagging Mandates". Artículo de Aplicación, 2004, http://www.rfid.zebra.com/.

[3] Graeme Daniel y Kevin Cox, "Identifying Things in Schools: Bar Codes and RFID", Mayo 4, 2003, http://webtools.cityu.edu.hk/news/newslett/RFID.htm.

[4] TI-RFid, "Tag-it HF Transponder Inlays Reference Guide", octubre 2001.

[5] Michael Liu, Adrian Tan, Dimas Brataadiredja y Pranav Kapra, "Driver Program for TI 6000 series reader", 2003. http://shay/ecn.purdue.edu/~mobility

[6] TI-RFid, "HF Reader System Series 6000, S6350 Midrange Reader Module RI-STU-TRDC-02", enero 2002.

[7] Icela Lagunas, "Reconocimiento en Alemania a los Zapatos Vigilantes", Periódico **El Universal**, página C4, 16 de mayo, 2004.

A single sensor active noise control with a hybrid structure using FxLMS and FxRLS algorithms

Miguel A. Bravo-Parra, Héctor Pérez-Meana, Mariko Nakano-Miyatake

Electrical and Mechanical Engineering School
National Polytechnic Institute of Mexico
Av. Santa Ana, no 1000, col. San Francisco Culhuacan, México D. F.
hmpm@prodigy.net.mx, mbravo@calmecac.esimecu.ipn.mx

Abstract. A hybrid structure for Active Noise Control is proposed which uses a system identification configuration, trained using the FxRLS algorithm, to reduce de acoustic noise, and a predictive structure, trained using the FxLMS algorithm, to further reduce the noise power. Here the predictive structure estimates the feedback interference introduced in the system by the two microphones used in the identification stage. In this form a fast convergence is obtained with high noise level cancellation and a relatively low computational complexity. Simulation results are provided to show the actual performance of proposed approach.

Keywords: Active noise control, Adaptive filters, Hybrid structure, FxRLS algorithm, FxLMS algorithm.

1. Introduction

Acoustic noise problems become more and more evident as increased number of industrial equipment such as engines, blowers, fans are in use [1]. This noise often is reduced using passive techniques such As enclosures, barriers and silencers to attenuate the undesired noise. These passive silencers are relatively large, costly and ineffective at low frequencies. Other technique which has a large attenuation is the Active Noise Control, based on the principle of superposition, ANC generate an "anti-noise" of equal amplitude and opposite phase which is combined with the primary source of noise, thus resulting in the cancellation of both noises [2]. ANC involves electro-acoustic or electromechanical system in order to cancel the undesired noise. Active noise control is defined as "a method for reducing the unwanted disturbances by the introduction of controllable 'secondary' sources, whose outputs are arranged to interfere destructively with the disturbance produced by the original 'primary' source" [3]. The Active Noise Control systems may have one several sensors that receive the signal of the primary source and the residual error signal. In this case we use the basic scheme of an acoustic duct in order to have just one reference sensor and one error sensor.

Figure 1 shows the block diagram of a single sensor active noise control used to cancel an acoustic noise signal traveling along a duct. Here the signal is picked up by a microphone at the reference position. This signal is then used as the input of the

ANC system whose parameters are adapted so that its output signal, after an electrical to acoustic conversion becomes the best estimated of the negative value of the duct noise in the error microphone position, resulting in such way in a cancellation of the acoustic noise in that point. The residual error, picked up by an error microphone, is then used to adapt the ANC system parameters.

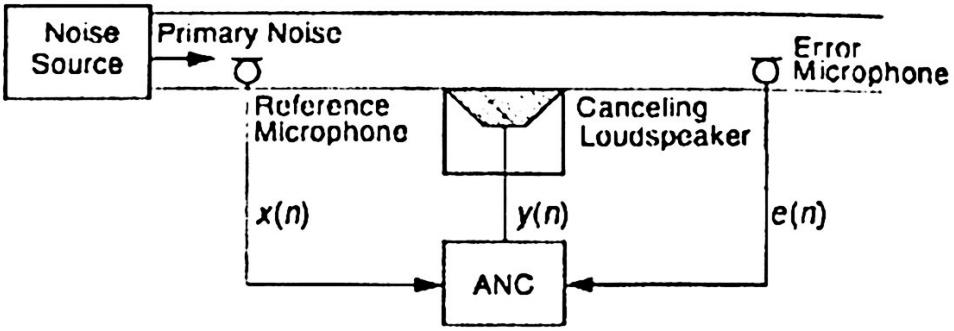


Fig. 1. . Active Noise Control on an acoustic duct.

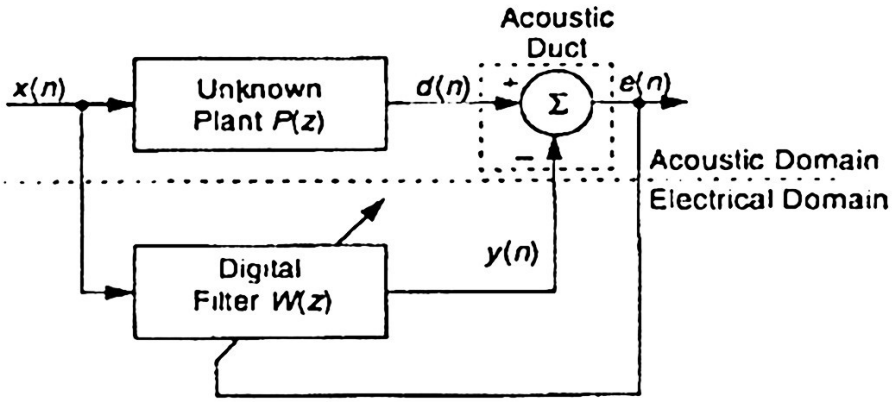


Fig. 2. Active Noise Control as an identification system.

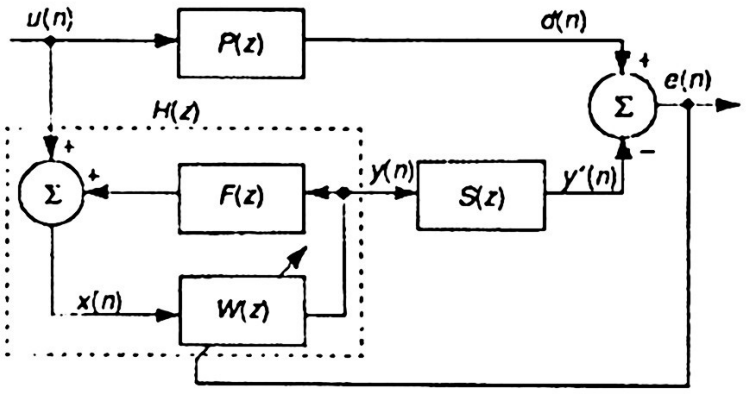


Fig. 3. Simplified diagram of ANC with the secondary path

Most ANC systems use an identifier structure, as shown in Fig. 2, in which an adaptive filter W(z) is used to estimate an unknown plant P(z). The plant and the adaptive filter are assumed to be driven by the same input x(n). Here the plant output is the desired signal which will be used for adaptive filter adaptation, which generates a replica of the plant output. Thus, if the adaptive filter can properly estimate the plant transfer function, P(z), the filter output will be very close the plant output and then the residual error will be close to zero.

The use of an adaptive filter for this application however is complicated due to the summing junction, because the noise cancellation must be done in the acoustic domain by superposition, at the error microphone, of the acoustic signal coming from de canceling speaker and the noise wave. To do that it is necessary to compensate the modifications that suffers the signal through the secondary a path S(z) which includes the digital–to–analog conversion, the reconstruction filter, power amplifier, loudspeaker, acoustic path, error microphone, preamplifier, ant–aliasing filter and the analog–to–digital converter (ADC).

Besides the secondary path explained above and denoted by S(z), the loudspeaker on a duct wall, shown in Fig.1, will generate plane waves that propagates to both, the error and reference microphones. Therefore the anti–noise not only cancels the primary noise source, but also radiates upstream to the reference microphone resulting on a corrupted reference signal x(n). This effect is called acoustic feedback.

Several approaches have been proposed to solve this problem, among them the feedback ANC with a predictive structure, which only uses an error microphone, appears to be a desirable alternative when a narrowband noise must be cancelled. However the performance of this structure degrades when the frequency band of noise widens because in this situation the signal can not be properly predicted.

To solve this problem this paper develops a single sensor ANC system in which an identification configuration is used in order to cancel the noise source together with a predictive structure which is used to reduce the feedback interference. Because the characteristics of the acoustic noise source and the environment may be time varying, and the frequency content, amplitude, phase and sound velocity of the undesired noise are non-stationary; the ANC must be adaptive in order to cope with these variations.

2. Hybrid ANC Structure

A combination of the feedforward and feedback control structures is called a hybrid ANC system, whereby the canceling signal is generated based on the outputs of both the reference sensor and the error sensor. Significant performance improvements occurred when the secondary – to – reference feedback is present. The figure 4 shows the classical block diagram of a hybrid ANC system.

In the proposed ANC system, the reference microphone is situated near the primary source in order to provide a coherent reference signal, where the error microphone is placed downstream and senses the residual noise. It is then used to synthesize the

reference signal for the adaptive feedback ANC filter, as well as to adapt the coefficients of both the feedforward and the feedback ANC filters.

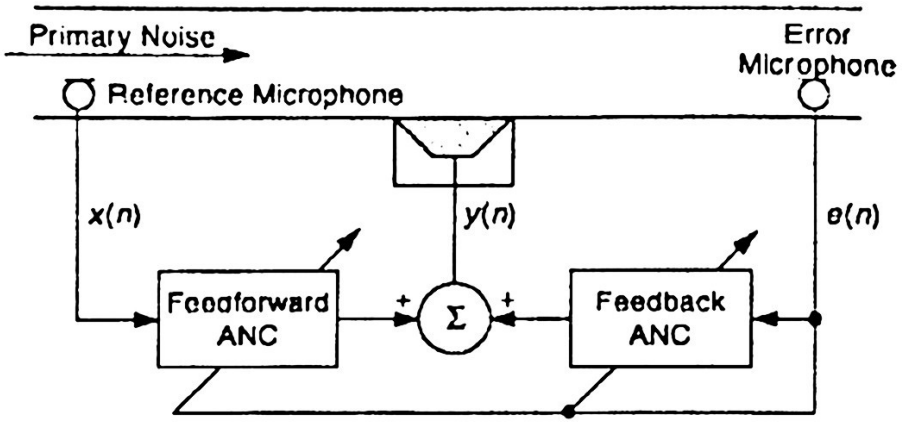


Fig. 4. Hybrid ANC system with a combination of feedforward and feedback ANC

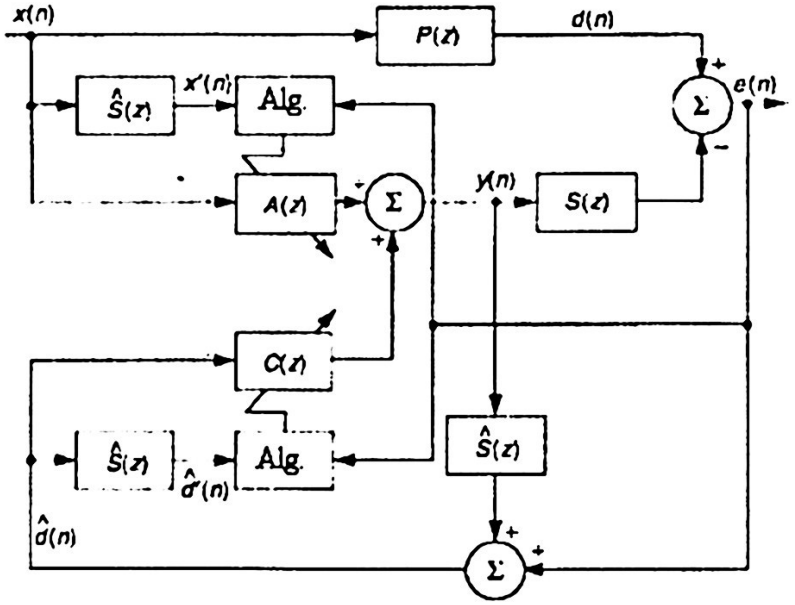


Fig. 5. Proposed hybrid ANC structure

The feedforward ANC attenuates the primary noise that is correlated with the reference signal, while the feedback ANC cancels the narrowband components and the secondary – to – reference feedback effect. The figure 5 shows the proposed hybrid structure, which increases the level of cancellation and reduces the feedback effects.

The proposed structure can be split into two main parts, the identifier part and the predictive part, where the identifier part, A(z), try to cancel the primary noise based on a ANC feedforward scheme of using the filtered input signal x'(n) to adapt the system, together with the residual error picked up by the reference sensor. Using these signals, the feedforward filter is updated using FxRLS algorithm. The second part of the ANC system is the predictive structure, in which the filter output and the output error are used to synthesize the feedback filter input signal, whose output signal will be used to predict and cancel in such way the secondary – to – reference

error and the remaining narrow band noise. In this part the algorithm used to update the filter coefficients is the FxLMS. Here the FxLMS is used instead the FxRLS to reduce the computational complexity of the prediction stage.

Then our proposed structure uses both algorithms, the FXRLS and the FXLMS, in order to obtain a higher level of cancellation and reducing the computational effort. Other hybrid ANC applications adapt both filters at the same time using the FxLMS algorithm or FxRLS algorithm. However using the FxLMS algorithm to adapt both filter a lower level of cancellation level is achieved even if the feedforward scheme with the secondary – to – reference effect is not present, and using the FxRLS algorithm the computational effort could be so high, because both filters must be updated simultaneously. On the other hand the proposed system uses a separate adaptation of the filters with both algorithms, achieving a high cancellation level with a moderate computational effort.

3. Filtered X Least Mean Square Algorithm (FxLMS)

The main problem present in the adaptation of W(z) that do not appear in the conventional noise canceling schemes is the introduction of the secondary path that may caused cause instability if it is not properly compensated, because the error signal is not correctly "aligned" in time with the reference signal, due to the presence of S(z). There are several schemes that can be used to compensate the secondary path effect. First, we can place an inverse filter, 1/S(z), in series with S(z) to remove this effect. This is possible only in the cases where the inverse of the transfer function of secondary path exists, but this is not always happens. In order to solve this problem a filter of the same characteristics of S(z) is placed on the path of the reference signal of the adaptive algorithm, that is the cause of the name "Filtered – X". The figure 3 will be used to derivate the algorithm, thus the residual error is given by:

$$\begin{aligned} e(n) &= d(n) - s(n) * y(n) \\ &= d(n) - s(n) * [w^{t}(n)x(n)] \end{aligned} \tag{1}$$

where n is the time index, s(n) is the impulse response of secondary path S(z), * denotes linear convolution, the coefficients vector W(n) at the time n is given by

$$w(n) = \left[w_0(n), w_1(n), \ldots, w_{N-1}(n)\right] \tag{2}$$

the input signal vector is at time n is given by equation (2), and N is the order of filter W(n). Here the impulse response of filter W(n) must be of sufficient order to accurate the model of the response of physical system.

$$x(n) = [x(n), x(n-1), \ldots, x(n-N+1)]^{T} \tag{3}$$

Thus the expected mean squared error is: $\xi(n) = E[e^{2}(n)]$, therefore the instantaneous error is given by:

$$\xi(n) = e^2(n) \tag{4}$$

Based on classic LMS algorithm, we have

$$w(n+1) = w(n) - \frac{\mu}{2} \nabla \xi(n) \tag{5}$$

Where $\nabla \xi(n)$ can be expressed as:

$$\nabla \xi(n) = \nabla e^2(n) = 2[\nabla e(n)]e(n) \tag{6}$$

Where $\nabla e(n) = -s(n) * x(n) = -x'(n)$, thus the FXLMS algorithm results:

$$w(n+1) = w(n) - \mu x'(n) e(n) \tag{7}$$

We can not forget that the secondary path is inherent to the characteristics of the physical system, thus this factor is unknown, and then the secondary path $\hat{S}(z)$ should be estimated from the input data with an "on line" technique or with a "off line" technique.

$$x'(n) = \hat{s}(n) * x(n) \tag{8}$$

4. Filtered X Recursive Least Square Algorithm (FXRLS)

To update the coefficients of the adaptive filter **W**(z), the FxRLS algorithm uses an approach similar to the FxLMS to derive the adaptation equation, because the secondary path is presented also in the system, but it uses a gain matrix including the filtered reference signal instead of directly using the filtered reference signal. The FxRLS algorithm combines a correlation of the filtered reference signal and a minimization of a weighted sum of the passed squared errors. Thus the FxRLS algorithm is given by.

$$\mathbf{W}(n) = \mathbf{W}(n-1) + \mathbf{K}(n)e(n) \tag{9}$$

The equation (9) shows the way how coefficient vector **W**(n) is updated in the FxRLS algorithm, where the vector **K**(n) is defined by:

$$\mathbf{K}(n) = \frac{\mathbf{Q}(n-1)\mathbf{X}(n)}{\lambda + \mathbf{X}^T(n)\mathbf{Q}(n-1)\mathbf{X}(n)} \tag{10}$$

where **X**(n) is the vector of the filtered reference signal, the vector length is equal to the adaptive filter order, λ is the "forgetting factor" that permits RLS type algorithms track the statistics of the input signal, and the matrix **Q**(n) is defined by equation (11), and it represents the inverse of the input signal autocorrelation matrix.

$$\mathbf{Q}(n) = \frac{1}{\lambda}[\mathbf{Q}(n-1) - \mathbf{K}(n)\mathbf{X}^T(n)\mathbf{Q}(n-1)] \tag{11}$$

Although the RLS algorithm represents a computational effort of the L^2 where L is the order of the filter, our proposed structure reduce this effort by a combination of the two algorithms and reducing the updating matrix.

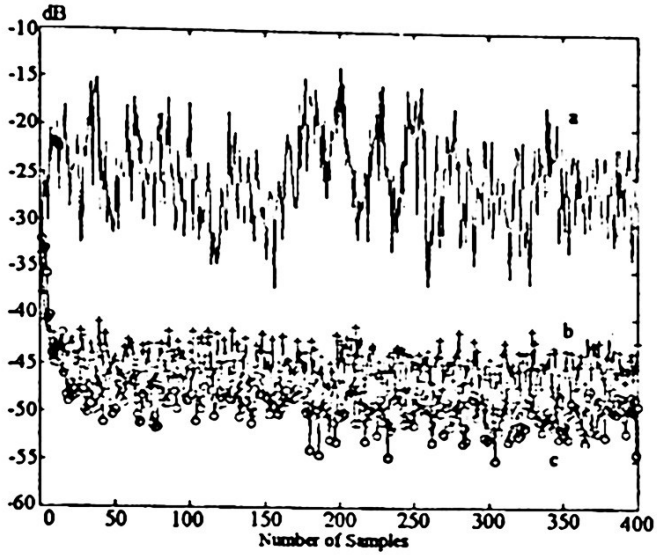


Fig. 6. Signal Power for the hybrid and identifier ANC with the sound of an airplane: (a) Original signal power, (b) Cancellation performance with out predictions stage. (c) Cancellation performance with prediction stage.

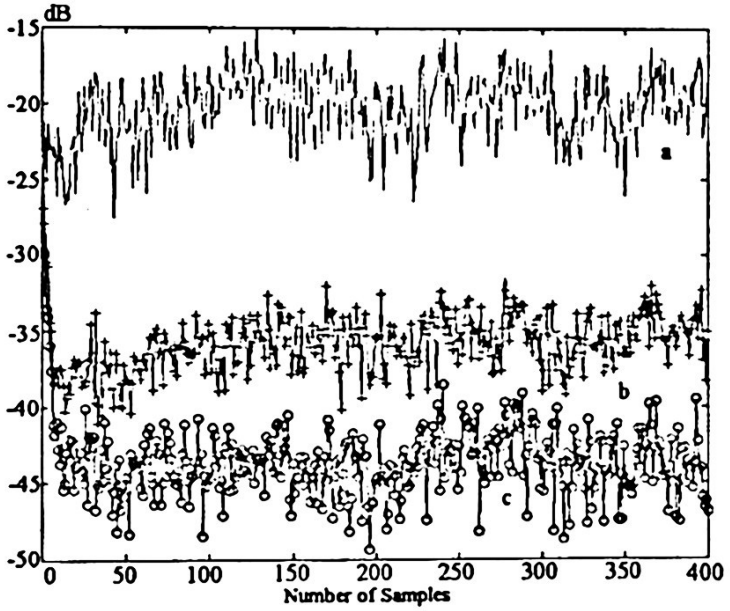


Fig. 7. Signal power for the hybrid and the identifier ANC with the sound of a bike: (a) Original signal power, (b) Cancellation performance with out predictions stage. (c) Cancellation performance with prediction stage.

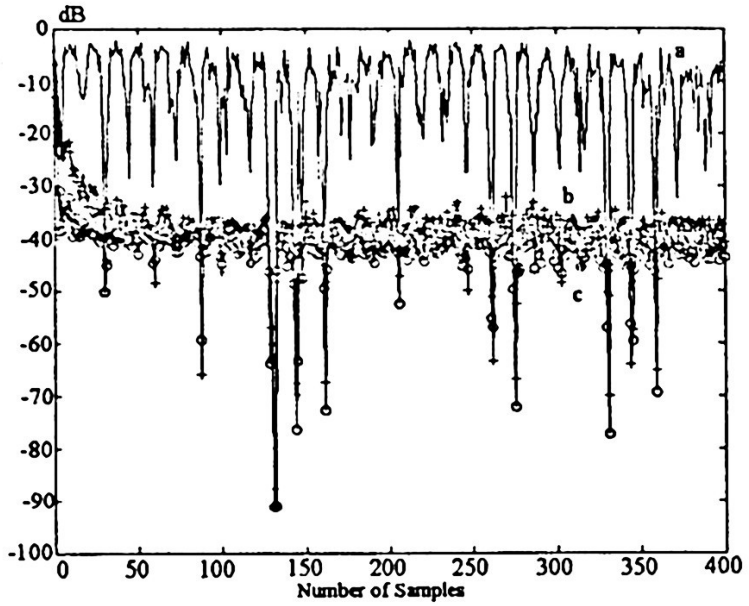


Fig. 8. Signal Power for the hybrid and identifier ANC with the sound if an engine: (a) Original signal power, (b) Cancellation performance with out predictions stage. (c) Cancellation performance with prediction stage.

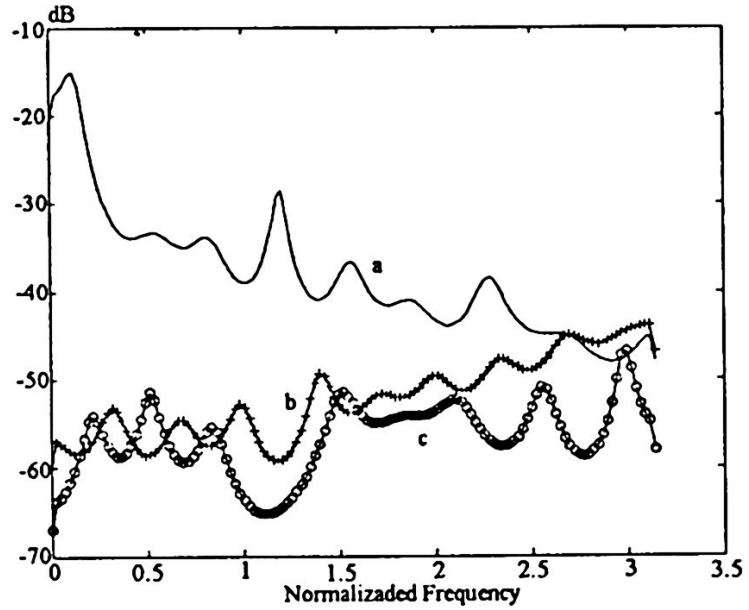


Fig. 9. Power spectral density for the hybrid and the identifier structures for the airplane sound: (a) Original signal power spectrum, (b) error signal power spectrum with out predictions stage. (c) Error signal power spectrum with prediction stage.

5. Simulation Results

The results that we present are from the simulation of the proposed structure and for the identifier structure, with both algorithms, the simulation result of the predictive structure is omitted because its performance is lower than even the identifier [6]. The

proposed structured was evaluated with the implementation of the FXLMS algorithms in both adaptive filters, and with the combination previously explained. The estimation of the secondary path was made with the "off line" technique, which consists on estimated the response of the secondary path when the ANC system is turned off yet. And when the estimation of S(z) converges the ANC system is turned on. The estimation scheme used was the "adaptive system identification problem", whose was made by the help of a white noise generator [6]. This can be possible because the characteristics of the physical system (length, distance between sensors, material, etc.) are assumed time invariant. The figures 6 to 8 show us the Power signal for the proposed scheme and the basic scheme without the prediction stage. The figures 9 and 10 present the power spectral density of the different structure.

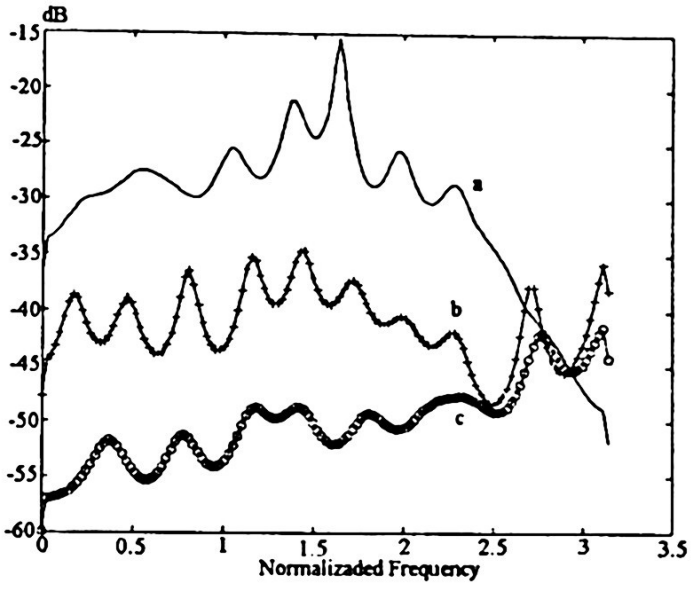


Fig. 10. Power spectral density for the hybrid and the identifier structures for the bike sound: (a) Original signal power spectrum, (b) error signal power spectrum with out predictions stage. (c) Error signal power spectrum with prediction stage

6. Conclusions

In this paper we have presented alternative structure for the hybrid ANC systems, based on the combination of the both the FXRLS algorithm and the FXLMS algorithm which allows to increase the performance by a higher level of cancellation and a moderate computational effort, because de updating matrix, for the FXRLS algorithm, is smaller than the one we should use if the filters were updated at the same time. This algorithm improve the performance in a narrow band noise situation, this noise is very common in the industrial environment, which give us a large implementation field. However with a broadband noise the scheme give us a similar level of cancellation than the identifier structure, thus the system can be implemented in a several bunch of applications without decreasing the performance.

Acknowledgments

The authors specially thank to the Consejo Nacional de Ciencia y Tecnologia (CONACYT) for the support to the investigation that gives origin to this paper.

References

[1]Sen M. Kuo, Dennis R. Morgan, "Active Noise Control Systems –Algorithms and DSP implementations", John Wiley & Son, New York, 1996.

[2]Daniel Tapia Sanchez, Hector M. Perez Meana, Mariko Nakano Miyatake, "Orthogonal Methods for Filtered – X Algorithms applied to Active Noise Control using Discrete Cosine Transform ", Zarubezhnaya Radioelektronika, Vol. 2003, No. 6, pages. 59 – 67, June 2003.

[3]Akira Omoto, "Active Noise Control: Adaptive Signal Processing and Algorithms", IECE Trans. Fundamentals, Vol. E85-A, No. 3, 2002, pages. 548 – 557.

[4]Sen M. Kuo, Dennis R. Morgan, "Active Noise Control: A Tutorial Review", Proceedings of IEEE, Vol. 87, No. 6, 1999, pages. 943 – 973.

[5]Orlando J. Tobias, José Carlos M. Bermudez, "Mean Weight Behavior of the Filtered-X LMS Algorithm", IEEE Transactions on Signal Processing, Vol. 48, No. 4, 2000, pages. 1061 – 1075.

[6]Miguel A. Bravo Parra, Héctor M. Pérez Meana, Mariko Nakano Miyatake. "Algoritmos para la cancelación activa de ruido" ,14 Reunión de Otoño de Comunicaciones Computación y Electrónica, ROC&C'03, Acapulco, Guerrero, Mexico, November 2003

Análisis de desempeño de un filtro para el mejoramiento de imágenes de huellas dactilares

Victor Hugo García Ortega, Osvaldo Espinosa Sosa
Laboratorio de Sistemas Digitales
Centro de Investigación en Computación-IPN, Av. Juan de Dios Batíz, esquina con Miguel Otón de Mendizábal
07738 México, D.F
Tel. 57596000 ext. 56519 v.garciaortega@lycos.com, espinosa@cic.ipn.mx

Resumen. Este trabajo presenta los resultados de un algoritmo basado en la Transformada de Fourier para el filtrado y mejoramiento de imágenes de huellas dactilares, dicho algoritmo es parte de un sistema AFAS(Automatic Fingerprint Authentification System)[5][6], el cual esta basado en la detección de minucias en huellas dactilares para el control de acceso a recintos. Este algoritmo fue probado en imágenes de buena y mala calidad obteniendo resultados muy satisfactorios. El algoritmo esta sustentado en el trabajo descrito en [7] y presenta variantes con respecto a éste, además, fue probado en una computadora personal Pentium III y un Procesador Digital de Señales(DSP) a fin de poder medir y comparar el tiempo de ejecución del algoritmo en este procesador de arquitectura específica para la ejecución de algoritmos como el que aquí se plantea.

1 Introducción

Básicamente los sistemas biométricos[12] basados en huellas dactilares son de dos tipos[8]:

Automatic Fingerprint Authentification System(AFAS).
Automatic Fingerprint Identification System(AFIS).

En un AFAS la entrada es la identidad de la persona y la imagen de la huella dactilar de esa persona; y la salida es una respuesta de SI ó NO, indicando si la imagen de entrada pertenece a la persona cuya identidad es proporcionada.

En un AFIS la entrada es sólo la imagen de la huella dactilar y la salida es una lista de identidades de personas que pueden tener la huella dada, además de una puntuación de cada identidad indicando el grado de similaridad entre ésta y la huella dada.

Ambos sistemas utilizan los detalles formados en las huellas dactilares. Estos detalles llamados "*ridges*" son definidos como un segmento de curva simple. La combinación de varios *ridges* forman un patrón de huella dactilar. Las pequeñas características formadas por el cruce y terminación de *ridges* son llamadas *minucias*. La finalidad del filtro propuesto es el poder eliminar todo el ruido posible de la imagen a fin de poder extraer las minucias de la imagen[5][6][2].

Cabe mencionar que para la adquisición de la imagen se uso el *5th Sense Parallel Evaluation Kit*, el cuál es el Kit de desarrollo del sensor de huella dactilar FPS110. Dicho sensor es fabricado por la compañía Veridicom. El sensor es del tipo capacitivo, el cuál cuenta con una resolución de 500 dpi y con un tamaño de arreglo de píxeles de 300x300[4].

Existen muchas técnicas específicas para realizar un mejoramiento de huellas dactilares. Las técnicas descritas en [13] y [11] son de las más efectivas, pero estas técnicas están diseñadas para usarse en los sistemas del tipo AFIS(como en sistemas policíacos del FBI); y por lo tanto requieren demasiado tiempo de procesamiento en comparación con sistemas mas pequeños como son los sistemas del tipo AFAS.

El principio básico de operación en estos métodos es dividir a la imagen en bloques, creando un filtro específico para cada bloque basado en la dirección promedio del bloque(imagen direccional)[5][6][9] y entonces filtrar cada bloque. En [13] este proceso se hace en el dominio del tiempo y en [11] se hace en el dominio de la frecuencia.

Por lo tanto, se requiere un sistema el cual cree un filtro adaptado a la dirección exacta del bloque. La información direccional de un bloque específico de la imagen esta contenido en la magnitud de la Transformada de Fourier del bloque, por lo que sí se considera un bloque el cual contenga dos o tres *ridges* aproximadamente paralelos y se toma su Transformada de Fourier, entonces se supone que las frecuencias dominantes de ese bloque corresponden a los *ridges* en el bloque y a la respuesta en Frecuencia del filtro para ese bloque.

2 Algoritmo

Es posible mejorar un bloque específico por sus frecuencias dominantes multiplicando la Transformada de Fourier del bloque por su magnitud un cierto número de veces, entonces cuando se calcula la Transformada Inversa de Fourier se observa que el bloque ha sido mejorado.

Si multiplicamos por el cuadrado o el cubo de la magnitud de la transformada de Fourier se obtendrían mejores resultados, sin embargo las regiones donde se localizan las minucias suelen ser alteradas, por lo que en vez de multiplicar por factores enteros a la magnitud de la transformada de Fourier se debe utilizar una fracción[7].

Para el cálculo de la transformada de Fourier se implementó el algoritmo de la transformada rápida de Fourier(FFT– Fast Fourier Transform) descomposición en tiempo(Decimation in Time) en la PC y en frecuencia(Decimation in Frequency) en el DSP, ambos base 2(radix 2).

El algoritmo propuesto consiste básicamente en realizar dos filtrados a la imagen(ver figura 1.), donde el segundo filtrado sólo se hace a las regiones que lo necesiten(las mas ruidosas). En este trabajo se llegó a la conclusión de que este filtro basado en la

Transformada de Fourier trabaja mucho mejor sobre la imagen binaria que sobre la imagen en niveles de gris como se describe en [7]. Por esta razón primero se binariza la imagen con un umbral dinámico y después se filtra. Después de realizar el primer filtrado, en algunas ocasiones suelen haber regiones todavía con demasiado ruido, por lo que un segundo filtrado es necesario a fin de poder eliminar dicho ruido. El umbral compuesto determina las regiones ruidosas que necesitan el segundo filtrado, las cuales se les aplica el filtro nuevamente.

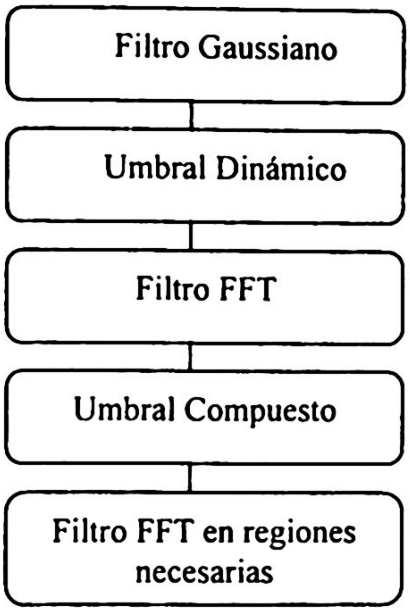


Fig. 1. Algoritmo propuesto para el mejoramiento

2.1 Filtro Gaussiano

Este filtro es un filtro pasabajas, cuyo efecto es el de realizar un suavizado en la imagen, en el cuál el núcleo es una Gaussiana 2-D, de media cero y desviación estándar σ, de la forma:

$$G(x,y) = e^{-(x^2+y^2)/2\sigma^2} \tag{1}$$

La transformada de Fourier de la Gaussiana es todavía Gaussiana y prácticamente no posee lóbulos laterales, por lo que su respuesta en frecuencia es muy buena.

2.2 Umbral Dinámico

Para poder separar claramente las regiones que pertenecen al fondo de la imagen, de los ridges, se requiere aplicar alguna técnica de segmentación, esto se hace binarizando la imagen a través de un proceso de umbralización[10].

Existen diferentes tipos de umbrales, global, local y dinámico[10], este último consiste en dividir la imagen en bloques B_{KL} de 15x15 píxeles, de los cuales se obtiene su media μ_{KL}. Entonces la media μ_{KL} de un bloque B_{KL}, esta definida por:

$$\mu_{KL} = \left(\sum_{j=LB}^{(L+1)(B-1)} \sum_{i=KA}^{(K+1)(A-1)} I(i,j) \right) / (AB) \tag{2}$$

Donde A, B es el tamaño del bloque y K, L es el número de bloques obtenidos de la imagen. Entonces μ_{KL} es el umbral del bloque B_{KL}, de tal forma que cada bloque tendrá un umbral diferente. Este método funciona muy bien sobre imágenes de mala calidad, a diferencia del umbral global que funciona bien en imágenes de buena calidad[1]. La imagen binarizada usando los métodos de umbralización dinámico y global se muestran en la figura 2.



Fig. 2. Huella binarizada con umbral global y dinámico respectivamente

2.3 Filtro FFT

Para mejorar a la imagen a través de la magnitud de su transformada de Fourier, ésta es dividida en bloques de 32x32 píxeles, a los cuales su transformada de Fourier F(u,v) es calculada de acuerdo a:

$$F(u,v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) \exp\left\{ -j2\pi \left(\frac{ux}{M} + \frac{vy}{N} \right) \right\} \tag{3}$$

La imagen mejorada g(x,y) en cada bloque es obtenida por:

$$g(x,y) = F^{-1}\left\{ F(u,v) x \, F(u,v)^{k} \right\} \tag{4}$$

Donde F(u,v) es la transformada de Fourier del bloque y k es la fracción por la cual se multiplica a la magnitud de la transformada de Fourier, y F^{-1} es la transformada Inversa de Fourier obtenida de acuerdo a:

$$f(x,y) = \sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F(u,v)\exp\left\{j2\pi\left(\frac{ux}{M} + \frac{vy}{N}\right)\right\} \tag{5}$$

2.4 Umbral Compuesto

Puesto que la umbralización global funciona bien en una imagen de buena calidad, y el umbral dinámico en imágenes de mala calidad, en este trabajo se combinaron ambos métodos para obtener un algoritmo al cual se le denominó *umbralización compuesta.* Este algoritmo se muestra en la figura 3. y se utiliza después de realizar el primer filtrado basado en la FFT, en donde a veces, se tienen todavía regiones de mala calidad. La determinación de estas regiones de mala calidad se hace usando el método de segmentación por varianza[14], en este método la imagen se divide en bloques B_{KL} de 15x15 píxeles de los cuales se extrae su varianza v_{KL} dada por:

$$v_{KL} = \left(\sum_{j=LB}^{(L+1)(B-1)}\sum_{i=KA}^{(K+1)(A-1)}\left[I(i,j) - \mu_{KL}\right]^2\right)/(AB) \tag{6}$$

Donde A, B, K, L tienen el mismo significado que en la ecuación 2. La varianza obtenida de los niveles de gris correspondientes al fondo y/o a regiones uniformes es muy pequeña (cero en regiones perfectamente uniformes), y la varianza obtenida en regiones donde se encuentra la imagen de la huella dactilar en buenas condiciones es alta, con este método los bloques de mala calidad se marcan para que posteriormente éstos sean nuevamente filtrados.

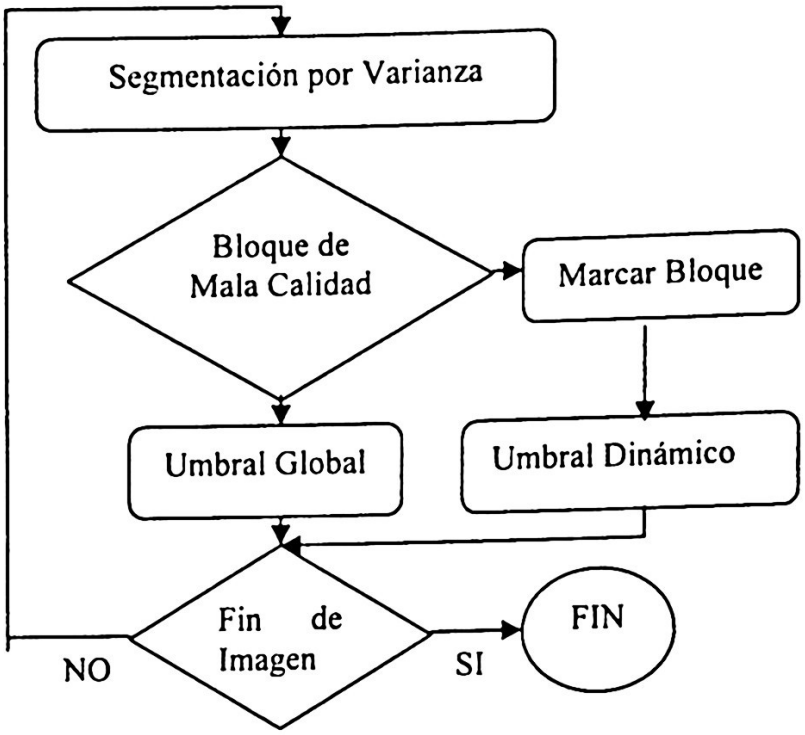


Fig. 3. Algoritmo propuesto denominado umbralización compuesta

3 Plataformas usadas para la ejecución del algoritmo

3.1 Computadora personal

Este algoritmo se programó en una computadora personal(PC) en lenguaje "c" usando el ambiente de programación visual Builder C++ versión 5.0 y en Matlab versión 5.0. Los requerimientos de la PC donde se hicieron las pruebas es el siguiente:

- Computadora personal Pentium III a 550Mhz
- 128Mb de memoria RAM

3.2 DSP

El DSP usado pertenece a la famlia TMS320C3x, de Texas Instruments. La arquitectura de la familia 'C3x' tiene un alto desempeño debido a su amplio y preciso rango dinámico de unidades de punto flotante, gran capacidad de memoria, alto grado de paralelismo(al tener una arquitectura Harvard) y el controlador de DMA[3].

Específicamente se uso el DSP TMS320C31-50, el cuál realiza el procesamiento de la imagen. Este DSP es un procesador con 32 bits en el bus de datos y 24 bits en el bus de direcciones, así mismo trabaja con una frecuencia de reloj de 25 MHz, al dividir internamente su frecuencia de oscilador de 50 MHz en un factor de dos. Por lo tanto su ciclo de instrucción es 40ns y realiza 275 Millones de operaciones de punto fijo por segundo(MOPS), 50 Millones de operaciones de punto flotante por segundo(MFLOPS), 25 Millones de instrucciones por segundo(MIPS).

Además estos procesadores tienen un modo de direccionamiento especializado en realizar el proceso conocido como reversión de bits para el cálculo del algoritmo de la FFT. Con esto la ejecución de dicho algoritmo es muy rápida.

4 Resultados y conclusiones

Experimentalmente se encontró que un valor de k=1.1 en el filtro fue el óptimo para el mejoramiento en imágenes de buena calidad, y para imágenes de mala calidad k=1.3 presenta mejores resultados[5][6][2]. Este método presenta un problema en los bordes de cada bloque por lo se necesito un traslape de 26 píxeles entre cada bloque para eliminar este efecto[5][6].

El filtro basado en la FFT se probó con 5 imágenes de diferentes personas, las cuales tienen diferentes tamaños y diferentes tipos de calidad. Los resultados del algoritmo en 4 de esas imágenes se muestran en la figura 4.

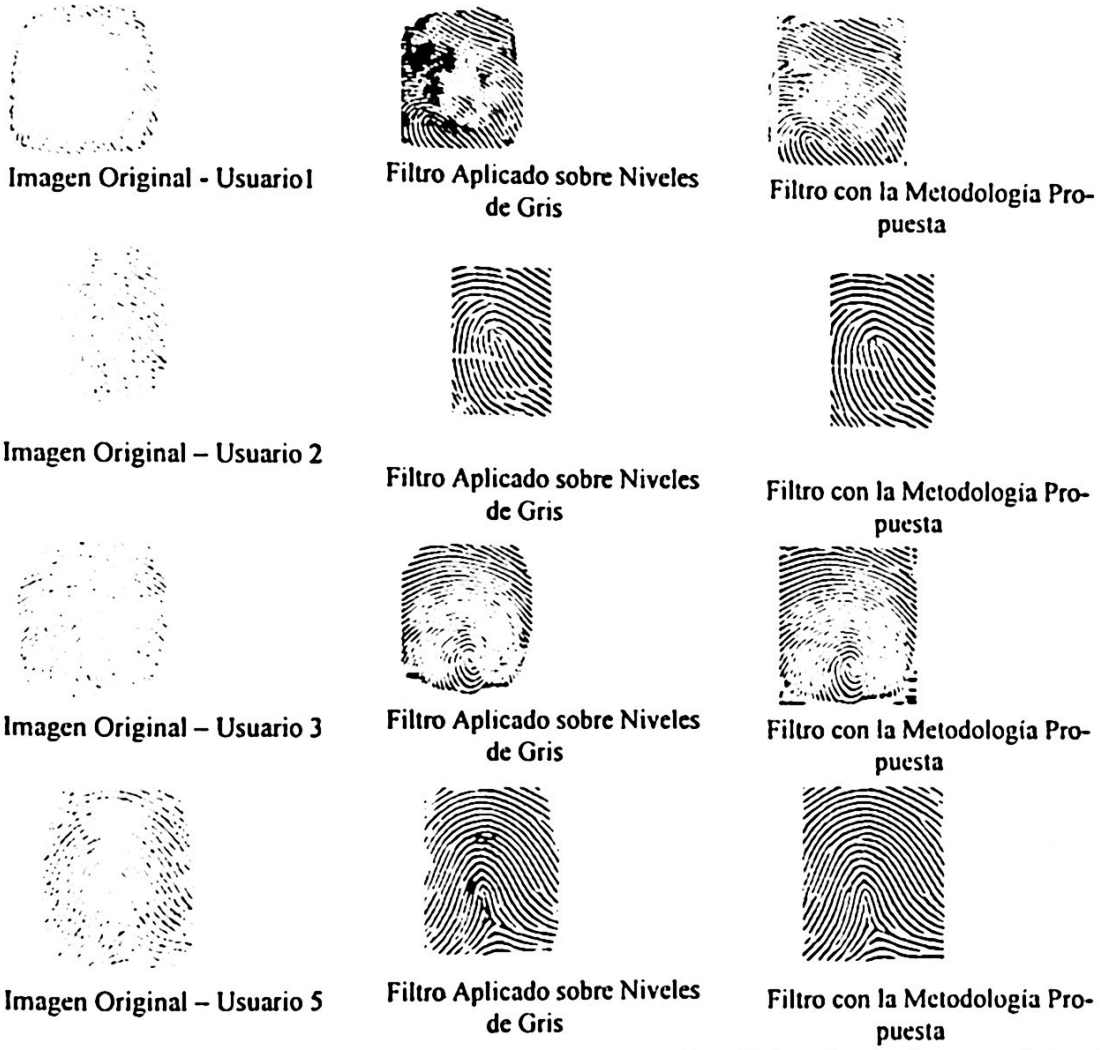


Fig. 4. Imágenes de huellas dactilares

De los resultados mostrados se puede concluir que la eficiencia de la metodología propuesta para el filtrado y mejora de imágenes de huellas dactilares es grande. Se puede observar que para las imágenes de mala calidad los resultados de la metodolo-

gía propuesta son muy buenos, en cambio con las imágenes de buena calidad los resultados son muy similares a los descritos en [7].

Con estos resultados el porcentaje de eficiencia obtenido en el AFAS es muy grande[2][1].

Los resultados de los diferentes tiempos de ejecución de este algoritmo para las 5 personas en los diferentes lenguajes de programación y plataformas, se muestran en la tabla 1.

Tabla 1. Tiempos de ejecución del filtro basado en la FFT

Imagen	Algoritmo en "C"	Algoritmo en el DSP	Algoritmo en Matlab
Usuario1	10 seg.	42 seg.	54 seg.
Usuario2	4 seg.	14 seg.	26 seg.
Usuario3	10 seg.	39 seg.	51 seg.
Usuario4	5 seg.	21 seg.	33 seg.
Usuario5	7 seg.	31 seg.	38 seg.

Cabe mencionar que para realizar la implementación del filtro en el ensamblador del DSP se implementaron las funciones de logaritmo natural y exponencial en el DSP, para esto se utilizó el código de las funciones que proporciona Texas Instruments[3].

En cuanto a los tiempos de ejecución del algoritmo en la PC se tiene la ejecución más rápida al programarse el algoritmo en lenguaje "C", en esta misma plataforma al programarse el algoritmo sobre Matlab los tiempos se incrementan considerablemente.

En el DSP los tiempos son mayores que en la PC, a pesar de la arquitectura de este procesador, lo posibles factores que ocasionan esto, son sin duda primeramente la velocidad de 25Mhz del DSP contra 550 Mhz de la PC y el uso de métodos numéricos para el cálculo y evaluación del logaritmo natural y la exponencial en el ensamblador del DSP, en la PC se hace uso del coprocesador matemático para el cálculo de estas funciones.

Referencias

[1]V. H. Garcia, *"Tesis de Maestría: Sistema de Reconocimiento de Huellas Dactilares para el Control de Acceso a Recintos"*,CIC-IPN, México D.F., 2004.
[2]V. H. Garcia, O. Espinosa, *"Estrategias para la Autentificación de Huellas Dactilares"*,Congreso Internacional ANIEI, Zacatecas, Zacatecas, 2003.
[3] Texas Instruments, TMS320C3X User's Guide. 2003.
[4] Veridicom, FPS110 Data Sheet. 2003
[5] V. H. Garcia, O. Espinosa, *"Sistema de Reconocimiento de Huellas Dactilares para el Control de Acceso a Recintos"*,Congreso Internacional CIC-INDI, México DF, 2001.
[6] V. H. Garcia, *"Sistema de Reconocimiento de Huellas Dactilares para el Control de Acceso a Recintos Basado en el DSP TMS320C31"*, Congreso Internacional ELECTRO-2001, Chihuahua, Chihuahua, 2001
[7] A.J. Willis and L. Myers, *"A cost-effective fingerprint recognition system for use with low-quality prints and damaged fingertips"*, Pattern Recognition, vol. 34, No 2, pp. 255-270, February 2001.
[8] L. C. Jain, U. Halici, I. Hayashi, S. B. Lee, Intelligent biometric techniques in fingerprint and face recognition, 1999.
[9] Mardia, K.V., Baczkowski, A. J., Feng, *"Statistical methods for automatic interpretation of digitally scanned finger prints"*, Pattern Recognition Letters. pp 1197-1203,1997.
[10] R. González, R. Woods, Digital Image Processing, 1996.
[11] Sherlock, B.G., Monro, D.M., and Millard, K., *"Fingerprint Enhancement by Directional Fourier Filtering"*, IEEE Proceedings Vision, Image and Signal Processing, Vol. 141, No 2. pp 87-94. 1994.
[12] B. Miller, *"Vital signs of identity"*, IEEE Spectrum, 31(2), pp. 22-30, 1994.
[13] O' Gorman, L. And Nickerson, J.V., *"An approach to fingerprint filter design"*, Pattern Recognition, Vol. 22, No. 1, pp 29-38. 1989.
[14] B.M. Mehtre and B. Chatterjee, *"Segmentation of fingerprint images A composite method"*, Pattern Recognition, vol. 22, pp. 381-385, 1989.



Diseño de un sistema encriptador para mensajes telefónicos

René Rodríguez-Ávila, Alfonso Gutiérrez.

Instituto Politécnico Nacional, Centro de Investigación en Computación
Av. Juan de Dios Batíz s/n, casi Esq. Miguel Othón de Mendizábal,
Unidad Profesional "Adolfo López Mateos", Edificio CIC.
Col. Nueva Industrial Vallejo 07738, México D.F.
citvrene@hotmail.com

Resumen. Este sistema pretende proteger conversaciones realizadas a través de líneas telefónicas (señales analógicas de hasta 4 KHz de frecuencia) a partir de la implementación de un algoritmo de encriptación/descencriptación RSA en un DSP; Primeramente la señal entra al interfase de comunicación, pasa por un filtro antialias con una frecuencia de corte de 3.5 KHz, después se convierte de analógico a digital con un DAC de 8 bits después se procesa por el DSP de la 5ª generación de Texas Instruments TMS32050, conectado a un módem de 56 Kbps para realizar la llamada al receptor, la señal se hace pasar por un filtro de reconstrucción con una frecuencia de corte de 3.5 KHz para obtener la señal original y sea escuchada por el receptor.

1 Introducción

La Criptografía según el Diccionario de la Real Academia, proviene del griego kriptos, que significa oculto, y grafos, que significa escritura, y su definición es: "Arte de escribir con clave secreta o de un modo enigmático".

Hace años que la Criptografía dejó de ser un arte para convertirse en una técnica, o más bien un conglomerado de técnicas, que tratan sobre la protección u ocultamiento frente a observadores no autorizados de la información. Entre las disciplinas que engloba cabe destacar la Teoría de la Información, la Teoría de Números o Matemática Discreta, que estudia las propiedades de los números enteros, y la Complejidad Algorítmica [1].

Definiremos un criptosistema como una quíntupla (M, C, K, E, D), donde:

- M representa el conjunto de todos los mensajes sin cifrar que pueden ser enviados (lo que se denomina texto claro, o plaintext).
- C representa el conjunto de todos los posibles mensajes cifrados, o criptogramas.
- K representa el conjunto de claves que se pueden emplear en el criptosistema.

- E es el conjunto de transformaciones de cifrado o familia de funciones que aplica a cada elemento de M para obtener un elemento de C. Existe una transformación diferente Ek para cada valor posible de la clave k.
- D es el conjunto de transformaciones de descifrado, análogo a E.

Todo criptosistema ha de cumplir la siguiente condición:

$$Dk\ (Ek\ (m)) = m,$$

es decir, que si tenemos un mensaje m, lo ciframos empleando la clave k y luego desciframos empleando la misma clave, obtenemos de nuevo el mensaje original m.

Existen dos tipos fundamentales de criptosistemas:

Criptosistemas simétricos o de clave privada. Son aquellos que emplean la misma clave k tanto para cifrar como para descifrar. Presentan el inconveniente de que para ser empleados en comunicaciones la clave k debe estar tanto en el emisor como en receptor, lo cual nos lleva preguntarnos cómo transmitir la clave de forma segura.

Criptosistemas asimétricos o de llave pública, emplean una doble clave (kp, kP) donde kp se conoce como clave privada y kP se conoce como clave pública. Una de ellas sirve para la transformación E de cifrado y la otra para la transformación D de descifrado. En muchos casos son intercambiables, esto es, si empleamos una para cifrar la otra sirve para descifrar y viceversa, y en general se basan en plantear atacante problemas matemáticos difíciles de resolver. En la práctica muy pocos algoritmos son realmente útiles. El más popular por su sencillez es RSA, que ha sobrevivido a multitud de ataques [2].

El Algoritmo RSA de entre todos los algoritmos asimétricos, quizá sea el más sencillo de comprender e implementar, sus claves sirven tanto para codificar como para autentificar. Debe su nombre a sus tres inventores: Ronald Rivest, Adi Shamir Leonard Adleman, y estuvo bajo patente de los Laboratorios RSA hasta el 20 de septiembre de 2000. Sujeto a múltiples controversias, desde su nacimiento nadie ha conseguido probar o rebatir su seguridad, pero se le tiene como uno de los algoritmos asimétricos más seguros.

RSA se basa en la Teoría de Números, que entre otras técnicas, estudia la dificultad para factorizar números grandes. Las claves pública y privada se calculan a partir un número que se obtiene como producto de dos primos grandes. El atacante enfrentará, si quiere recuperar un texto claro a partir del criptograma y la llave pública, a un problema de factorización[3].

Para poder implementar este algoritmo nos remontamos a la Teoría o Aritmética Modular.

La *Aritmética Modular* es una parte de las matemáticas extremadamente útil criptografía, ya que permite realizar cálculos complejos y plantear problemas

interesantes, manteniendo siempre una representación numérica compacta y definida, puesto que sólo maneja un conjunto finito de números enteros [4].

El *Algoritmo de Euclides* que es utilizado para desarrollar el algoritmo RSA permite obtener de forma eficiente el máximo común divisor de dos números [5] y se basa en el Teorema de él mismo que dice:

Teorema de Euclides. Si un número primo divide a un producto, divide a uno de los factores[6].

El *Algoritmo Extendido de Euclides* también puede ser empleado para calcular inversas las cuales se requieren para elaborar el algoritmo de encriptación y es una ampliación del de Euclides que posee el mismo orden de complejidad, y que se obtiene simplemente al tener en cuenta los cocientes además de los restos en cada paso [7].

El *Algoritmo de Exponenciación Rápida* se utiliza para calcular el módulo de dos números enteros largos, y es una herramienta de la aritmética modular muy poderosa ya que si tuviéramos que evaluar el módulo de dos enteros largos usando el método convencional que es multiplicar n veces el número al que se desea obtener el módulo, implicaría realizar demasiadas operaciones, además de obtener números que quizá ni una computadora podría soportar, por tanto este algoritmo simplifica dichas operaciones [8].

A continuación se muestra el algoritmo de la exponención rápida:

```
int exp_rapida(int a, int b) {
int z,x,resul;
z=b;
x=a;
resul=1;
while (z>0) {
    if (z%2==1)
          resul=resul*x; }
x=x*x;
z=z/2;
}
```

Un ejemplo de cómo aplicar el algoritmo es el siguiente:

Calcular el valor de 210368 mod 187.

Tabla 1. Cálculos realizados por el algoritmo para calcular 210368 mod 187.

No. de iteraciones	resultado	z	x
1	r = 1	z = 10368	x = 2
2	r = 1	z = 5184	x = 4
3	r = 1	z = 2592	x = 8
4	r = 1	z = 1296	x = 16
5	r = 1	z = 648	x = 32
6	r = 1	z = 324	x = 64
7	r = 1	z = 162	x = 128
8	r = 1	z = 81	x = 256 (mod 187) = 69
9	r = 69	z = 40	x = 4761 (mod 187) = 86
10	r = 69	z = 20	x = 7396 (mod 187) = 103
11	r = 69	z = 10	x = 10609 (mod 187) = 137
12	r = 69	z = 5	x = 18769 (mod 187) = 69
13	r = 4761 (mod 187) = 86	z = 2	x = 7396 (mod 187) = 86
14	r = 86	z = 1	x = 10609 (mod 187) = 103
15	r = 8858 (mod 187) = 69		

El sistema consta de 5 partes: interfase comunicación que consta de un micrófono una bocina por la que el usuario mandara el mensaje y escuchara la respuesta, la etapa de filtrado para el evitar el efecto aliansing y para reconstruir la señal analógica, el DAC que es la etapa de conversión analógica a digital, un Procesador Digital de Señales (Digital Signal Processor o DSP) que hará el procesamiento de la señal aplicando el algoritmo de encriptación RSA, y por último el módem el cual efectuara la llamada telefónica y hará el acoplamiento hacia la línea telefónica, el sistema se muestra en la figura 1.

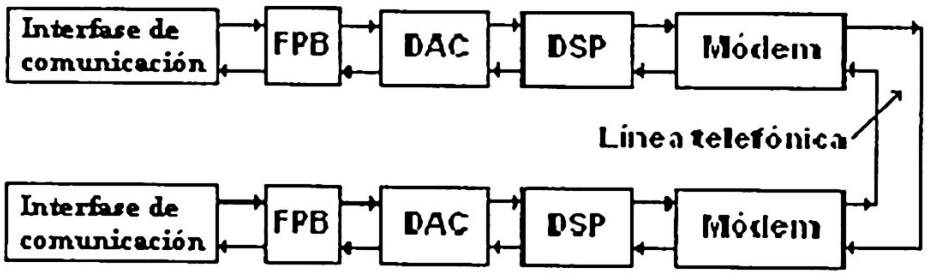


Fig. 1. Diagrama del sistema transmisor (encriptador)/receptor (descencriptador).

2 Metodología

En sustitución de la señal de voz (o telefónica) se utilizaron archivos de audio tipo WAV, con el uso de Matlab hemos pasado los datos de estos archivos de audio muestreados a 8 KHz, con 8 bits, monoaural con una duración de 60 s a archivos de tipo texto para probar el algoritmo de encriptación y reconstrucción de la señal (descencriptación).

Basados en la investigación anterior hemos desarrollado e implementado este algoritmo en un leguaje de programación de alto nivel "Delphi" ver. 4.0 para aplicar el sistema de criptografía RSA a los valores de la señal, el programa desarrollado permite introducir un archivo de texto con los datos de la señal que se desean encriptar o descencriptar, estos datos están en forma fraccionaria por lo que se trataron como datos tipo flotante.

El programa incluye una sección de acondicionamiento para la señal ya que el algoritmo utiliza enteros y nuestra información es de tipo fraccionario.

El acondicionamiento consistió en:

1. Obtener una señal positiva, esto debido a que la señal esta en el orden de ± 1Volt, se le sumo un 1 (1 Volts) teniendo ahora una señal de 0 a 2 Volts,
2. Convertir los datos fraccionarios a enteros, esto se hizo simulando la conversión digital; es decir; dependiendo el número de bits seleccionados para la conversión sé multiplico por la mitad de 2^{bits}, esto es similar a proceso de cuantificación del convertidor pues al realizar esta operación se les asigno valores dentro de los niveles de cuantificación que van de 0 a 2^{bits}. Por ejemplo: para 8 bits se tiene 256 niveles de cuantificación.

Prácticamente esto sería equivalente a tener una señal y procesarla por el DSP teniendo al final un archivo encriptado.

Para seleccionar el DAC óptimo requerimos de hacer pruebas con el programa para verificar la calidad de la señal reconstruida y que esta fuera comprendida; esta información se obtuvo usando nuevamente Matlab para convertir el archivo de tipo texto descencriptado a un archivo de audio tipo WAV y escuchar su contenido.

Posteriormente se seleccionó el DSP dependiendo del número de operaciones requeridas por el algoritmo RSA; otro factor a considerar es el tipo de operaciones que realiza, es decir, si son de punto fijo o flotante, como ya dijimos el algoritmo utiliza enteros por lo que queda claro que la familia del DSP debía ser de punto fijo.

3 Resultados

Al ejecutar el programa en Delphi, es decir, aplicar el algoritmo RSA a la información del archivo obtenemos como resultado un archivo de salida con los datos procesados o encriptados.

El programa RSA se muestra en la figura 3 que se muestra a continuación:

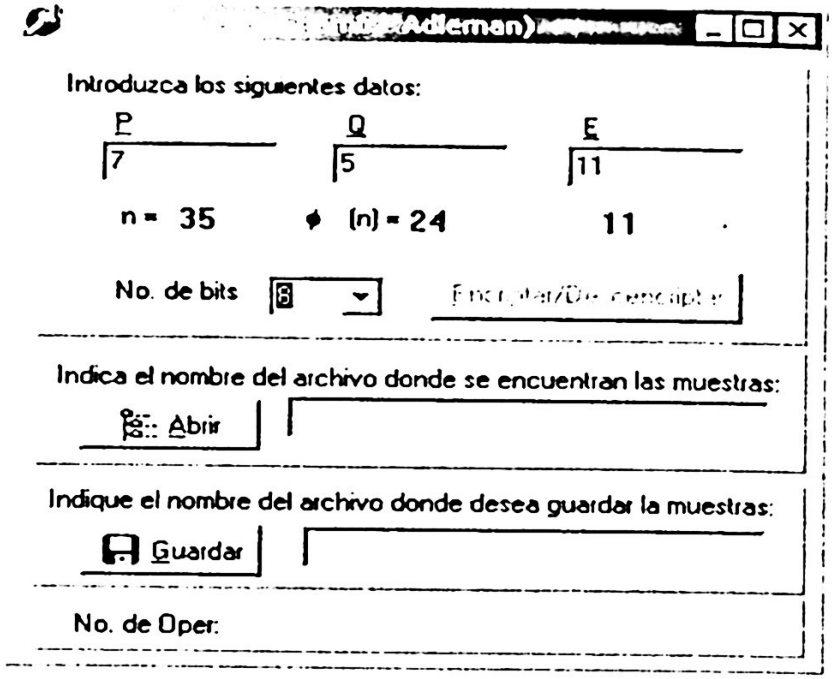


Fig. 3. Programa en Delphi para encriptar/descerncriptar archivos por medio del algoritmo RSA.

Los datos requeridos por el programa para realizar una encriptación son: P = primer número primo, Q =segundo numero primo, E = clave pública y el número de bits para la conversión digital; Por último se indica el archivo fuente que es donde se tienen los datos del contenido la señal y la dirección de la carpeta donde se guardará el resultado de la encriptación.

Debido a que el algoritmo sirve tanto para encriptar como para descencriptar, podemos después de encriptar la información introducir el archivo encriptado a entrada del programa y aplicar el algoritmo nuevamente para obtener la señal original.

Se hizo la prueba con el archivo de texto obtenido con Matlab con la información del archivo WAV que tenía una duración 60 s.

La gráfica de 500 muestras de la señal original y reconstruida se muestra en la figura

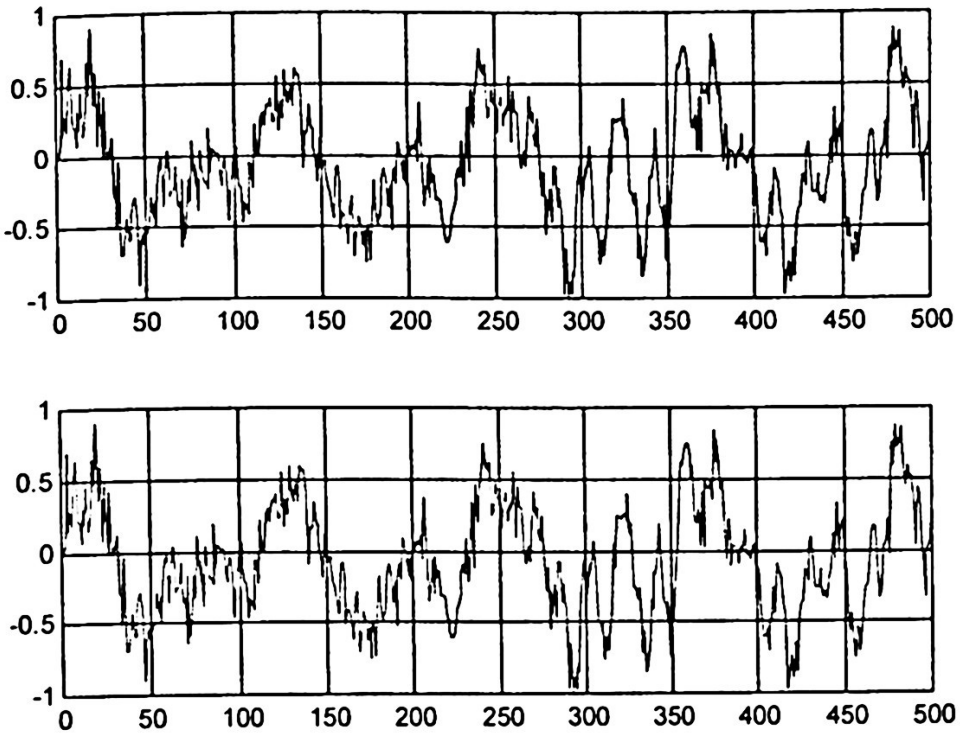


Fig. 4. Gráfica de la señal original y reconstruida.

Señal original.

Señal reconstruida.

La gráfica del error producido por el redondeo en las 500 muestras se observa en la figura 5.

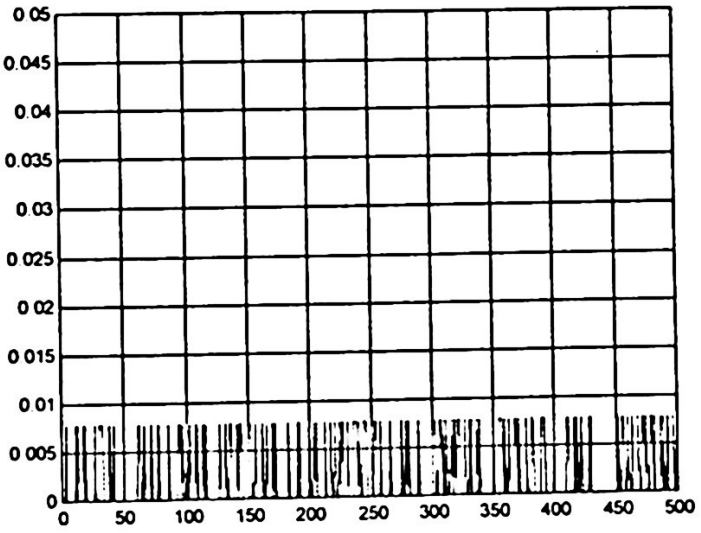


Fig. 5. Gráfica de las muestras con error de redondeo.

En las 500 muestras se presentaron 196 errores, esto quiere decir que 39.2% de los datos tuvieron errores de redondeo en promedio, este error es de 0.007843 que con respecto a la señal es poco significante respecto a sus valores reales.

La calidad de la señal reconstruida con los DAC's de 8 y 16 bits para un archivo de audio de 60 s de duración es casi la misma, es decir, la señal reconstruida en la que se uso el DAC de 8 bits y muy similar a la obtenida cuando se uso el DAC de 16 bits, de hecho, la diferencia es poco perceptible por lo que bastaría con utilizar un DAC de bits.

El hecho de usar un DAC de 16 bits reduciría el ancho de banda que necesita el módem para enviar los datos ya que por su naturaleza el módem utiliza 8 bits y si enviamos un dato 16 bits necesitaríamos ocupar el DAC 2 veces para enviar un dato .

Debido a lo anterior el DSP seleccionado debía ser uno que ejecutara 500,000 operaciones/s o más, el procesador TMS32050 cumple con esta especificación, este DSP de la marca Texas Instrumentes de la 5ª generación ejecuta una instrucción en 50 ns, procesa datos de punto fijo mismo que nos servirá porque los datos que procesaremos son enteros.

Discusión

Las pruebas indican que el desarrollo es viable por lo que se ha iniciado la implementación del algoritmo en el DSP y para ello estamos utilizando el software Code Compuser Studio de TI mostrado en la figura 2, este software permite programar los procesos utilizando Lenguaje "C" el cual es muy usado en la ingeniera en lugar del lenguaje ensamblador, el compilador del software traduce del lenguaje "C" a "ensamblador" las rutinas implementadas minimizando el tiempo para programación e implementación de rutinas o algoritmos.

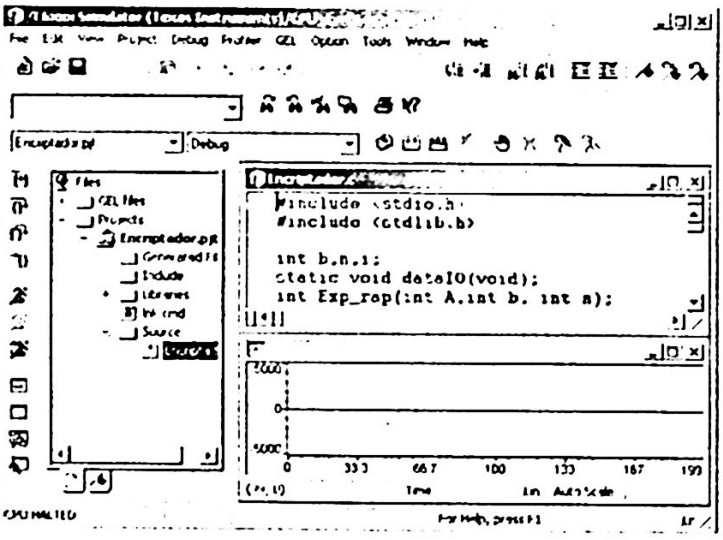


Fig. 2. Code Compuser Studio de TI última versión para ambiente Windows 2000 / XP.

Una vez implementado el algoritmo se harán pruebas con señales digitales (ya muestreadas) y posteriormente se adaptara el DAC el cual se encargara de convertir la señal de Analógica a Digital (DAC) y viceversa con los filtros antialias y de reconstrucción respectivamente, y por último el marcado telefónico por medio del módem y el aparato telefónico.

Como se dijo habrá una etapa de filtrado donde se implementará un filtro antialias que se construirá con la configuración de un filtro Butterworth pasabajas 60 db/dec con una frecuencia corte 3.5 KHz para obtener una señal lo más limpia posible de ruido, esto, basándonos en la frecuencia de la señal original, además de un filtro de reconstrucción de la señal analógica, con la configuración de un filtro Butterworth pasabajas de 60 db/dec con frecuencia de corte 3.5 KHz para obtener una señal analógica similar a la original o de entrada. Estos filtros se implementarán con elementos activos como Amplificadores Operacionales (OPAM's).

Referencias

1. Manuel José Lucena López
   Criptografía y Seguridad en Computadores
   Tercera Edición (Versión 1.14). Marzo de 2002
   e-mail: mlucena@ujaen.es
   Capítulo. 2.1, 29
2. Capítulo 2.2, 30
3. Capítulo 12.2, 143 - 144
4. Capítulo 5.1, 57
5. Capítulo 5.1.1, 58
6. Amparo Fúster, Dolores de la Guía
   Técnicas Criptográficas de protección de datos
   2ª. Edición Actualizada
   Alfaomega & Ra-Ma
   Apéndice A, 1, 240.
7. Manuel José Lucena López
   Criptografía y Seguridad en Computadores
   Tercera Edición (Versión 1.14). Marzo de 2002
   e-mail: mlucena@ujaen.es
   Capítulo 5.2.3, 63
8. Capítulo 5.4.1, 64
9. Apéndice C, Capítulo 5, problema 5.

Reconocimiento de patrones por medio de redes neuronales en sistemas de cultivo acuícolas automatizados

José Ángel Alcaraz Vega[1], Dr. Luís Pastor Sánchez Fernández[2]

Centro de Investigación en Computación CIC-IPN
Av. Juan de Dios Bátiz esq. M. Othón de Mendizábal S/N
Col. Nueva Industrial Vallejo.
C.P. 07738. México DF.
Teléfonos: 5729-6000 ext: 56501 y 56537
http: /www.cic.ipn.mx
[1]alcaraz@sagitario.cic.ipn.mx [2]lsanchez@cic.ipn.mx

Resumen. El artículo describe la realización de un proyecto de medición y predicción de estados de cultivo en granjas acuícolas utilizando redes neuronales para el reconocimiento de patrones basado en variables medioambientales en granjas camaroneras. Con la adaptación de la backpropagation al sistema de recolección de datos, se evalúa la situación del hábitat para la producción de camarón en los estanques de cultivo y se hace un análisis de tendencia para prever enfermedades bacteriológicas y virales que en la actualidad son la principal causa de mortandad en los crustáceos.

1 Introducción

En la actualidad la tendencia de los sistemas de control industrial está dirigida al campo de la inteligencia artificial. El diseñar sistemas que asemejen el funcionamiento del cerebro humano es la tarea fundamental de los investigadores de hoy en día. Las redes neuronales son la herramienta que más se asemeja al funcionamiento neuronal de nuestro cerebro; son neuronas que almacenan información y que se comunican entre ellas por medio de conectores utilizando diferentes arquitecturas de conexión, que permiten evaluar el envío y/o bloqueo de la información con el fin de realizar una modificación general de la misma. Se han podido realizar máquinas inteligentes por medio de software y hardware haciendo uso de la arquitectura neuronal en algoritmos computacionales y en chips electrónicos[9].

Por medio de las redes neuronales se tiene la facilidad de procesar información de forma inteligente, con esto se quiere decir que, no solamente se hace una evaluación estadística de la información si no también se hace una evaluación cuantitativa y cualitativa[8].

El proyecto es un sistema basado en el reconocimiento de patrones por medio de redes neuronales, aplicado para solucionar problemas referentes a la producción de camarón en granjas acuícolas. Haciendo uso de principios fundamentales de un sistema

SCADA incorporando herramientas matemáticas y electrónicas en conjunción con software de programación de alto nivel para control industrial, se ha llevado a cabo realización del sistema de control, monitoreo y supervisión. Por medio de las técnicas de programación del lenguaje G, LabVIEW™ de National Instrument™ se tienen resultados favorables en el empleo de instrumentación virtual para el control del sistema[6].

LabVIEW™ es un lenguaje de programación de alto nivel compuesto de herramientas y funciones programables que permiten el análisis profundo y detallado de variables patrones generados por fenómenos físicos y químicos. Con la creación rápida y sencilla de programas individuales para una tarea específica (también llamados VI's), se puede tener una arquitectura de programación de VI's más organizada, evitando errores causados por exceso de código y por conexiones extralimitadas entre subrutinas utilizadas en otros lenguajes de programación[7].

2 Análisis de campo

Por medio del estudio de campo realizado en las granjas camaroneras, se ha podido obtener una panorámica de los métodos que actualmente se utilizan en el cultivo del camarón y del nivel tecnológico con el que se cuenta para medir los parámetros en los estanques. Todos son medidos manualmente y con metodologías de análisis químicos. La frecuencia con la que éstos son tomados, varía dependiendo del parámetro. Con uso de estas técnicas se agrava la posibilidad de error, ya sea por errores humanos de equipo en la medición[2].

Con el sistema en cuestión, se realiza un análisis más detallado de cada uno de los parámetros implicados en el cultivo del camarón haciendo uso de tecnología electrónica implementada en sensores y computadores como también, el uso de la PC ordinaria.

El manejo de la calidad del agua en los cultivos acuícolas es de suma importancia, es por ello, que mantener los niveles adecuados de cada parámetro de cultivo, es fundamental para alcanzar niveles de producción globales.

Los parámetros se dividen en tres grupos[1]:

A) Parámetros Físicos

- Temperatura
- Salinidad
- Sólidos
  - Composición (orgánica e inorgánica)
  - Tamaño
  - Concentración
- Color

- Luz
  - Artificial o Natural
  - Total anual de energía inherente
  - Intensidad de la energía por radiación
  - Calidad de la luz
  - Período de luz

B) Parámetros Químicos

- PH y alcalinidad
- Gases
- Nutrientes
- Compuestos orgánicos
  - Biodegradables
  - No biodegradables
- Compuestos Tóxicos
  - Metales pesados

C) Parámetros biológicos

- Bacterias
- Virus
- Hongos

Todos son de suma importancia en el cultivo del camarón, ya que la variación de cada uno de ellos y/o en conjunto, provoca niveles de inestabilidad en el hábitat requerido para los estanques de cultivo llevando a la población del cultivo a una perdida total. De igual forma se ha comprobado científicamente que estos cambios bruscos en el hábitat alteran el comportamiento del camarón, provocándole estrés. Al estar estresado el camarón es más propenso a tener enfermedades, entre las cuales están las más peligrosas, las virales[4].

El virus WSSV, mejor conocido como virus de la mancha blanca, es hasta ahora el más mortífero de todos provocando una mortandad al 100% de la producción en los cultivos acuícolas. Por esto, los acuicultores, biólogos y especialistas en el área están luchando contra esta amenaza utilizando las técnicas más innovadoras[5].

Con el sistema se tiene una mejor perspectiva de como actúa el virus WSSV al modificarse el hábitat del camarón. Haciendo una predicción de la tendencia del hábitat; se puede observar como la variación de los parámetros de cultivo influyen en el camarón en el momento de ser infectado por el virus. Actualmente no se ha podido combatir el virus por medio de químicos y técnicas genéticas, por lo cual, es más conveniente evitar la infección haciendo uso de técnicas de predicción. Para llegar a estas técnicas fue necesario hacer un estudio detallado del comportamiento del virus al ser modificados cada uno de los parámetros de cultivo

3 Clasificación de Estados

Los estados de cultivo son clasificados en tres diferentes, *bajo*, *normal* y *alto*; cada uno de ellos fue clasificado dependiendo de los valores obtenidos en la medición de los parámetros correspondientes al hábitat del camarón en las granjas acuícolas. Los parámetros utilizados para la lectura de valores en el sistema son:

Tabla 1. Parámetros utilizados por el sistema de adquisición de datos empleados en el cultivo del camarón.

Parámetros de cultivo	Temperatura Ph Salinidad Oxígeno disuelto Amonia Nitritos Nitratos Salinidad

Estos parámetros son los de más importancia, debido a que la variación a gran escala de cada uno o en dado caso en conjunto, propician un desequilibro mortal en el hábitat de cultivo.

En cada parámetro hay un rango específico de valores donde el camarón tiene un crecimiento saludable y óptimo. Este rango óptimo se clasifico como *estado normal.* Para los rangos de valores que están bajo el rango de normales se clasifico como *estado bajo* y los que se encuentran arriba, como *estado alto.*

Tabla 2. Clasificación de los rangos de parámetros en estados de alarma.

Parámetros y Estados	Temp (°c)	PH	OD	Amonia	Nitritos	Nitratos
			(mg/l)			
Bajo	0-25	3-7.9	1.5-3	-	-	0-25
Normal	25-33	7.9-8.5	3-8.5	0-15	0-10	25-35
Extremo	33-45	8.5-9.0	8.5-10	15-25	10-21	35-45

Con los valores comprendidos entre los rangos correspondientes a cada uno de los parámetros, se realiza el entrenamiento de la red neuronal. La cual esta compuesta por tres neuronas de entrada, cada una con relación a uno de los estados (bajo, normal alto), teniendo a la vez tres neuronas de salida, donde cada una de ellas representa valor cualitativo dependiente del valor de entrada dado.

4 Arquitectura del sistema

El sistema en general es un SCADA, adaptado especialmente para la adquisición, procesamiento y presentación de datos, como también para la predicción de fallas y pronosticación de estados referentes al hábitat de cultivo que establecen la calidad de la producción.

Por medio de sensores instalados estratégicamente en los estanques que conforman la granja el sistema recopila información del medio (granja camaronera) y la envía por medio de sistemas de transmisión hacia el computador que contiene el sistema de predicción de fallas en lenguaje G, con esto se hace referencia a la posible mortandad de camarones en los estanques.

Los sistemas de transmisión de información pueden ser diferentes; si se necesita transmisión en área local, se usa los sistemas de bus de campo; para transmisiones remotas a larga distancia, se utiliza la radio frecuencia o la internet. Las restricciones de tiempo de lectura de parámetros en la granja no son estrictas; con esto se puede hacer uso de tecnología de transmisión lenta, como hasta ahora lo es el internet.

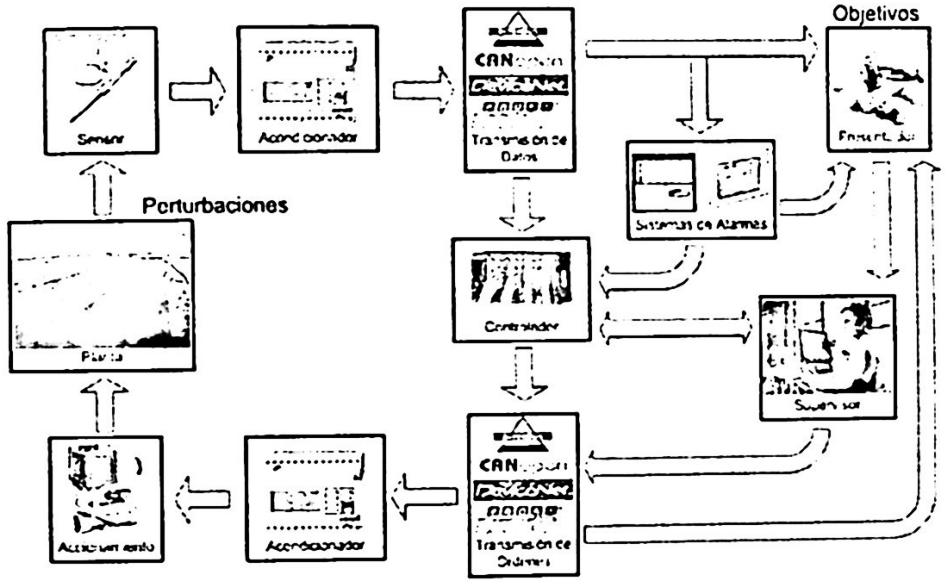


Fig. 1. Diagrama a bloques del sistema SCADA de control, monitoreo y control de granjas camaroneras.

El sistema es el encargado de pedir la información a los sensores, de procesarla para evaluar la situación del hábitat, de activar señales de alarma y de activar actuadores para estabilizar el entorno.

Cabe señalar que esta contemplada la transmisión de video a larga distancia, con el fin de que el operador esté monitoreando la granja y sus estanques desde otro lugar.

4.1 Adquisición y control de datos

La adquisición y el control de datos es la parte del sistema encargada de proporcionar la información de cada uno de los parámetros de los estanques. Por medio de sensores se adquiere la lectura de los valores correspondientes a cada parámetro, esto, puede ser utilizado un sensor para cada uno o bien, utilizando un sensor que tenga la capacidad de leer todos[10].

El sensor adquiere el estado actual del parámetro y lo trasforma en un estimulo señal análoga por medio de niveles de amplitud, la cual es trasformada en señal digital en el acondicionador de señal. Al ser transformada de análoga a digital, se puede enviar al computador para ser procesada.

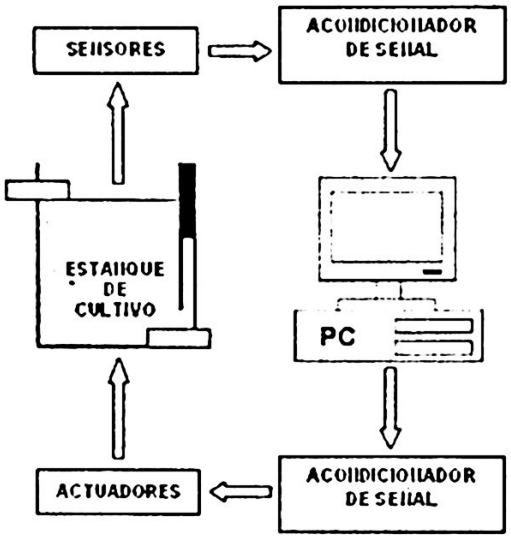


Fig. 2. Diagrama a bloques de adquisición de información y control.

En el computador se encuentra el sistema de control, monitoreo y predicción, que es encargado de pedir la adquisición de datos en determinados intervalos de tiempo, según lo requiera. Al procesar y evaluar las condiciones del estanque muestra los estados referentes al hábitat actual, dependiendo de éstos, se estima si es necesaria su modificación por medio de los actuadores.

Los actuadores son dispositivos eléctricos y neumáticos o bien, electroneumáticos que son activados por medio de un pulso eléctrico enviado del computador por órdenes del sistema al acondicionador de señal para que sea convertida de señal análoga digital[11]. Por medio de los actuadores se pueden modificar los parámetros para establecer el hábitat a un estado óptimo.

4.2 Funcionalidad del SCADA

La funcionalidad del sistema es básica para un sistema SCADA, consiste de cuatro etapas principales

- Etapa de supervisión, monitoreo y control.
- Etapa de adquisición de datos.
- Etapa de control directa.
- Etapa de acceso remoto de información.

La etapa de supervisión, monitoreo y control es la principal de las cuatro, es la que contiene el sistema en lenguaje G, por medio del cual se tiene la plataforma de control, monitoreo y adquisición de información. De igual manera se encuentran los algoritmos de clasificación de información en los que se utilizan redes neuronales. Se tiene la interfase usuario-maquina, que permite al operador monitorear y llevar un reporte detallado de las actividades realizadas.

La segunda, consiste en le software y los dispositivos encargados de adquirir información de los estanques por medio de sensores. El software utilizado aquí, es dedicado a la lectura de parámetros. En cambio, en la etapa de control directa se tienen dispositivos y software en cargado de accionarlos por medio de órdenes de la etapa principal.

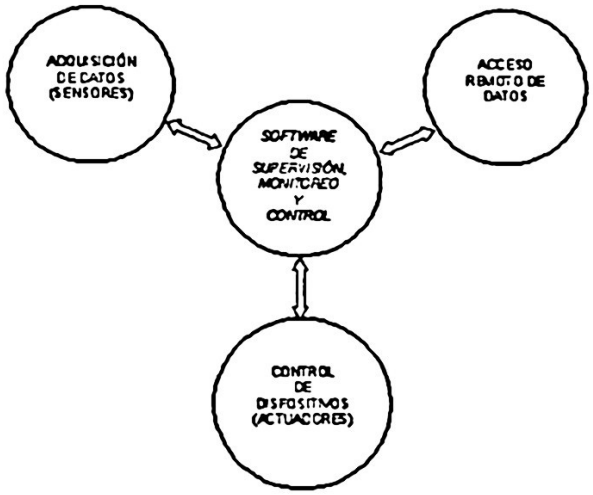


Fig. 3. Diagrama funcional del sistema.

En la etapa de acceso remoto a datos tenemos las terminales que se comunican por medio de internet y/o radio frecuencias con la etapa principal del proyecto. También llamadas laboratorios virtuales, se tiene la opción de monitoreo a distancia y de control, siempre y cuando se tenga autorización.

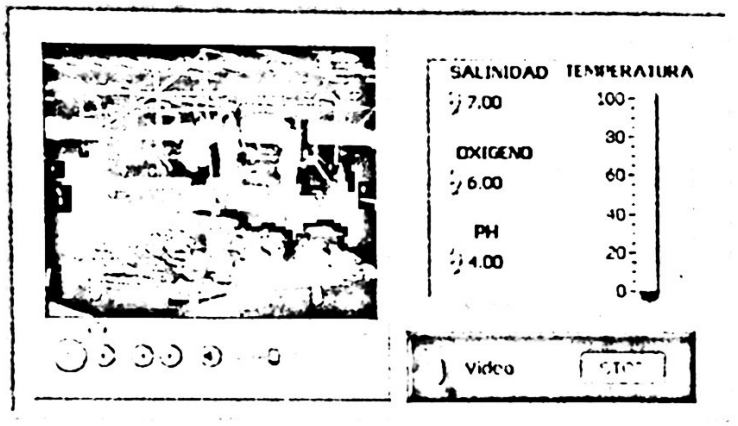


Fig. 4. Panel de visualización virtual

El lenguaje de programación G de LabVIEW™ nos permite tener interfases a distancia por medio de internet donde se puede visualizar lo que sucede en las granjas camaroneras y se puede tener un reporte de los parámetros.

5 Aplicación de la red neuronal

La aplicación de la red neuronal en el sistema es de suma importancia para el reconocimiento de patrones. La red es la encargada de clasificar los valores de entrada en estados. El entrenamiento de la red es basado en la catalogación de estados de comportamiento en los estanques de cultivo.

La red que se emplea es la backpropagation, ya que este tipo de red es recomendable para el reconocimiento y clasificación de patrones; en este caso se esta utilizando con los parámetros de cultivo.

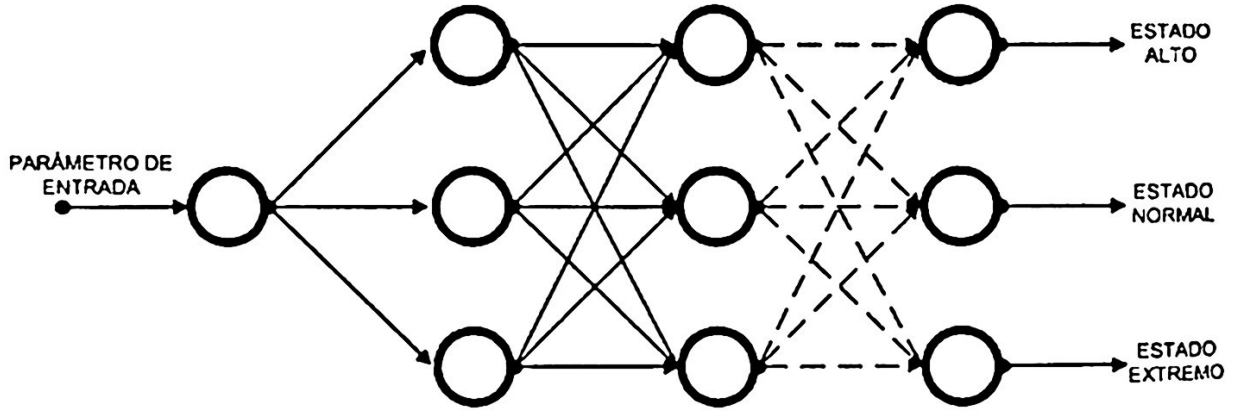


Fig. 4. Estructura de la red neuronal para la clasificación de estados.

La estructura de red mostrada en la figura 4, es aplicada en cada parámetro con el propósito de obtener la evaluación de su estado. En cambio, en la figura 5, se muestra la arquitectura generalizada de toda la red, donde tomando en cuenta la evaluación del estado de cada parámetro, se hace la evaluación general del hábitat. La evaluación del hábitat ayuda a establecer la tendencia del cultivo en los estanques y a prever enfermedades bacteriológicas y virales que atacan al camarón.

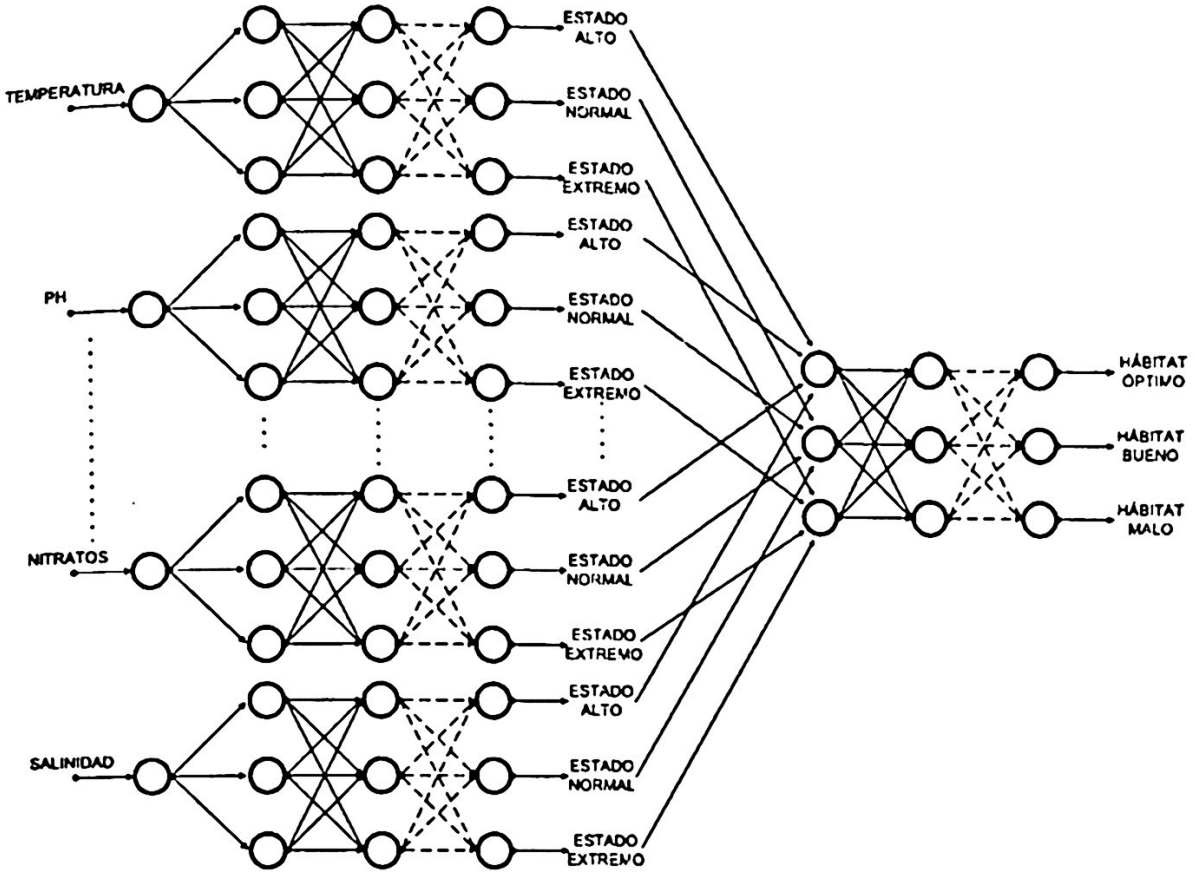


Fig. 5. Estructura general de la red neuronal para la predicción del hábitat.

Como se puede observar la backpropagation utilizada en le sistema es multicapa, con más de tres capas, con el fin de poder hacer la clasificación correcta de los valores de entrada con respeto a los esperados en la salida.

6 Plataforma SCADA

El sistema esta basado en el software de adquisición de información, en otras palabras, adquisición de datos. Por medio del programa de adquisición de datos se obtienen los valores correspondientes a cada uno de los estados especificados anteriormente y son guardados en una base de datos que posteriormente es utilizada en la red neuronal.

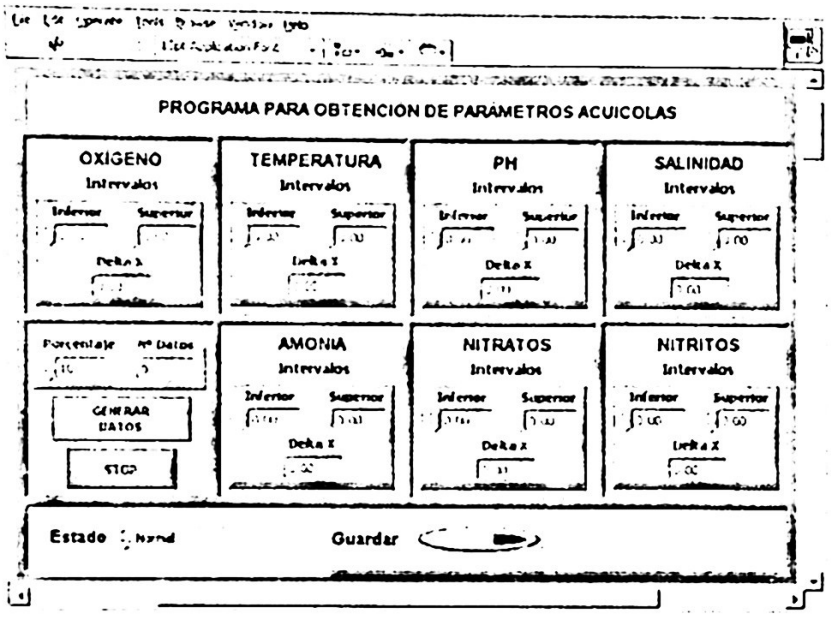


Fig. 6. Panel frontal del sistema de adquisición y clasificación de parámetros.

Como se puede ver en la figura 6, se tienen los límites inferior y superior que delimitan el rango de cada parámetro. Los límites indican los rangos de medición de los parámetros.

7. Conclusiones

La realización del sistema ha abierto una brecha en la solución de problemas enfocados a la producción acuícola, ayudando a los especialistas en el área a llevar un control más formal y detallado de los problemas causados por cambios en los parámetros del cultivo de camarón que producen una inestabilidad en el hábitat. Consecuentemente a estimar mortandad por enfermedades derivadas de cambios bruscos en el hábitat como la infección por bacterias y virus que provocan una mortandad global en el cultivo de camarón.

Se tiene la oportunidad de un laboratorio virtual a distancia para analizar y monitorear las instalaciones de las granjas camaroneras con el fin de que estudiantes y profesionistas en el ramo acuícola conozcan y evalúen más a fondo las operaciones y actividades realizadas en ellas. Por otro lado, una de las aportaciones del proyecto es al medio ambiente, llevando el control de los recursos naturales que han y siguen siendo materia prima en el cultivo de camarones en el mundo, como el agua y manglares ubicados en las costas de nuestro país. Esto es posible con la automatización de las granjas por medio del sistema de monitoreo y control del proyecto ayudando a reutilizar los recursos naturales y a no contaminar las costas de nuestros mares.

Referencias

[1] Bancomex, *Camarón Mexicano – Cultivo en Granjas*, Ed. Banco Nacional del Comercio Exterior, S.N.C. y el Centro de Investigaciones Biológicas del Noroeste, S.C. con el apoyo de la Federación de Acuicultores de México, 1999, pp. 27-51.

[2] Bancomex, *Camarón Mexicano – Resuelva Problemas en el Cultivo*, Ed. Banco Nacional del Comercio Exterior, S.N.C. y el Centro de Investigaciones Biológicas del Noroeste, S.C. con el apoyo de la Federación de Acuicultores de México, A.C., 2000, pp. 13-67.

[3] J.E. Bardach, J. Ryther, and W. McLarney, *Aquaculture – The Farming and Husbandry of Freshwater and Marine Organisms*, Ed. Science Editions, 1988, pp. 587-633.

[4] D. Brune, J. Tomasso, *Acuacultura y calidad del Agua*. Ed. The Word Aquaculture Society, 1991, pp. 11-70.

[5] J. Juárez, G. Palomo, *Acuicultura - bases biológicas del cultivo de Organismos Acuáticos*, Ed. Compañía Editorial Continental, S.A. de C.V. 1988, pp. 19-64.

[6] G. W. Johnson, *LabVIEW Graphical Programming – Practical Applications in Instrumentation and Control*. 2nd ed, Ed. McGraw-Hill, 1997, pp. 10-53.

[7] L. K. Wells, J. Travis, *LabVIEW For Everyone – Graphical Programming Made Even Easier*, Ed. Prentice Hall PTR, 1997, pp. 30-75.

[8] B. M. del Brío, A. Sanz Molina. *Redes Neuronales y Sistemas Difusos*. 2da Ed., Alfaomega & RA-MA, México 2000, pp. 10-85.

[9] J. R. Hilera, V. J. Martínez. *Redes Neuronales Artificiales - Fundamentos, Modelos y Aplicaciones*. Alfaomega & RA-MA, México 2000, pp. 20-73.

[10] R. Pallás Areny. *Sensores y Acondicionadores de Señal*. 3ra Ed, Alfaomega & Marcombo. México 2001, pp. 22-37.

[11] W. A. Rosenblith. *Sensory Communication – Contributions to the symposium on Principles of Sensory Communication*. Edicott House, pp. 22-41.

La medición de perturbaciones de frecuencia como indicador de la aspereza de la voz

C. Ferrer, M.E. Hernández-Diaz

CEETI, Facultad de Ingeniería Eléctrica, UCLV
Carretera a Camajuaní, Km 5½, cferrer@uclv.etecsa.cu

This paper deals with the quantification of voice harshness in patients with voice disorders by means of digital signal processing. A previously reported index (the JF) is used to this end. It is obtained from the variability of the glottal pulse intervals found with a recently developed method. The results of the JF are compared with the subjective ratings made by two specialists of 48 recordings of the phonation of the sustained vowel 'a' by dysarthric patients. The results reveal that the JF can be a reliable index of harshness in pathological voice, in the same manner that the perceptual ratings are used.

Resumen. En el presente trabajo se aborda la cuantificación, mediante técnicas de procesamiento digital de señales, del grado de aspereza de la voz de pacientes con trastornos del habla. Para este fin se emplea un índice objetivo reportado en la literatura (conocido como JF, de *Jitter Factor*), obtenido a partir de la determinación de la posición de los pulsos glotales mediante un método desarrollado recientemente [13]. Se comparan los resultados del índice objetivo con las valoraciones subjetivas, emitidas por dos especialistas, de 48 grabaciones de la fonación por pacientes disártricos de una vocal 'a' sostenida. Los resultados obtenidos permiten afirmar que el *Jitter Factor* puede ser un indicador confiable de aspereza en voces patológicas en la misma medida en que lo son las valoraciones subjetivas.

Palabras clave: *procesamiento de voz, disartria, jitter, shimmer, aspereza.*

1. Introducción

La obtención de índices objetivos que reflejen el grado de perceptibilidad de características subjetivas del habla es una rama del procesamiento de voz al que se le dedica especial interés en el mundo. Este interés esta dado por las aplicaciones que pueden tener estos índices en el diagnóstico, seguimiento y rehabilitación de personas con trastornos del habla.

Varios trabajos [3], [4], [6], [7], [8], [9], [10], [12] han mostrado que existe una correspondencia entre el tipo de trastorno y las características perceptuales del habla del enfermo. Dichas características perceptuales pueden agruparse en referidas al período fundamental, a la calidad, a la intensidad, a la articulación o a la prosodia.

Dentro de las referidas a la calidad se pueden mencionar la nasalidad, la aspereza, la ronquera y el jadeo.

La aspereza de la voz ha sido relacionada con las perturbaciones de la periodicidad de los pulsos glotales [5], [11], [16], [17] y pueden dividirse en perturbaciones de frecuencia (*jitter*) y de amplitud (*shimmer*). Estas perturbaciones de amplitud y frecuencia son provocadas por la vibración irregular de las cuerdas vocales, y rara vez parecen de forma independiente una de otra.

Como medidas de estas perturbaciones se han propuesto en la literatura [2] varios índices, entre los que se puede mencionar el *APQ* para el *shimmer*, y el *PPQ* y el *JF* (*Jitter Factor*) para el *jitter*. Todas estas medidas parten de la obtención previa de las posiciones y amplitudes de cada uno de los pulsos glotales en un sonido vocálico (preferentemente una vocal sostenida). La determinación de la posición de estos instantes de excitación en voces patológicas es poco confiable, pues las alteraciones de la periodicidad pueden ser de una magnitud tal que no se logre determinar correctamente el periodo fundamental en algunos segmentos de la vocalización. Esta falta de confiabilidad de los detectores de período fundamental en voces patológicas provoca que, aunque se acepte generalmente la relación entre la aspereza de la voz y el *jitter/shimmer*, no se hayan llegado a resultados definitivos en aplicaciones médicas.

En este trabajo se explora la correspondencia entre valoraciones subjetivas de aspereza en la voz de pacientes disártricos y el *jitter* obtenido a través de la utilización de un método de detección de período fundamental desarrollado recientemente.

2. Metodología

Se dispuso de 48 grabaciones [1] de pacientes disártricos efectuando la fonación de una vocal 'a' sostenida. Dos especialistas escucharon estas grabaciones y emitieron valoraciones subjetivas del grado de aspereza de cada una en una escala de 0 a 6, con 0 no perceptible y 6 aspereza extrema.

Como índice objetivo de aspereza se utilizó el JF, calculado según la expresión (1) donde con *F(i)* se denota el inverso de la duración del pulso *i*-ésimo y *N* es la cantidad total de pulsos glotales en el segmento de voz bajo estudio.

$$JF = \frac{\frac{1}{N-1}\sum_{i=2}^{N}\left|F(i)-F(i-1)\right|}{\frac{1}{N}\sum_{i=1}^{N}F(i)} * 100 \qquad (1)$$

La determinación de la posición de los pulsos glotales se efectuó mediante el método de detección del período fundamental pulso a pulso descrito en [13], que muestra mejores resultados en voces patológicas que los tradicionales. Este detector requiere de un estimado del período fundamental en un margen de ±25% del real. Dicho estimado fue obtenido mediante un detector básico de AMDF [15] para segmentos de

30 milisegundos. De todos los detectores de período fundamental por bloques el AMDF fue el que mostró un mejor seguimiento de la trayectoria del período fundamental en voces patológicas.

La confrontación entre los resultados del índice objetivo y las valoraciones subjetivas se efectuó siguiendo en lo fundamental el trabajo de Fukazawa *et al* [14] donde se desarrolla una tarea similar a la aquí descrita. En el mencionado trabajo se propone un índice objetivo para cuantificar el jadeo, que a su vez no estuviera significativamente correlacionado con la aspereza. Para ello usaron 24 grabaciones de vocales sostenidas de sujetos sanos y 34 de pacientes con afecciones del habla. Dos especialistas emitieron valoraciones subjetivas en una escala de 0 a 3, y a partir de estas se determinaron los coeficientes de correlación entre especialistas (0.714 en jadeo y 0.589 en aspereza), y entre el valor del índice propuesto y el promedio de las valoraciones de jadeo y aspereza de los especialistas (0.734 y 0.211 respectivamente). De aquí concluyen que el índice puede ser empleado como medida del jadeo de la voz en la misma forma en que se usa la percepción psicoacústica, ya que los resultados están en el mismo orden (0.734 y 0.714).

En el presente trabajo se decidió efectuar además un análisis de regresión lineal entre el índice objetivo y las valoraciones subjetivas, así como del histograma del coeficiente de correlación entre ambos. Estas pruebas brindan información adicional sobre la interrelación entre ambas variables.

Todos los algoritmos fueron programados en el paquete de software MatLab 5.2, incluyendo los de detección de período fundamental y el análisis estadístico.

La regresión lineal se efectuó entre el *JF* y el promedio de las valoraciones subjetivas (*VS*) de los dos especialistas. Se empleó la función **regress** del MatLab, que halla los valores b_1 y b_2 de la recta dada por:

$$JF = b_2 * VS + b_1, \tag{2}$$

además de brindar algunos datos estadísticos de interés, que se tratan con mayor detalle en las secciones 3 y 4 de este trabajo.

3. Resultados

En la Tabla 1 se muestran los coeficientes de correlación (ρ) obtenidos entre los especialistas (denominados 1 y 2 en la Tabla I) y entre el *JF* y el promedio de las valoraciones de ambos.

Tabla I

Coeficientes de Correlación Obtenidos

	1 vs. 2	***JF* vs. promedio(1,2)**
ρ	**0.7453**	**0.5996**

Para tener una mejor idea de las dependencias manifestadas por el coeficiente correlación se obtuvo un histograma del mismo. Este histograma fue hallado mediante el cálculo del coeficiente de correlación entre mil parejas de vectores (aleatoriamente conformados a partir del remuestreo de los originales) de *JF* y las valoraciones subjetivas. Los histogramas de las correlaciones entre los especialistas y de las correlaciones entre el *JF* y el promedio de los especialistas se muestran en la Figura

Los resultados de la regresión lineal se muestran en la Tabla II. La columna **b** muestra los valores de b_1 y b_2, mientras la denominada **bint** muestra los intervalos del 95% confianza de b_1 y b_2. En **stats** se muestran (de izquierda a derecha) el cuadrado del coeficiente de correlación y la probabilidad de haber obtenido la pareja de vectores *JF* y *VS* dado que ambos no estuvieran correlacionados.

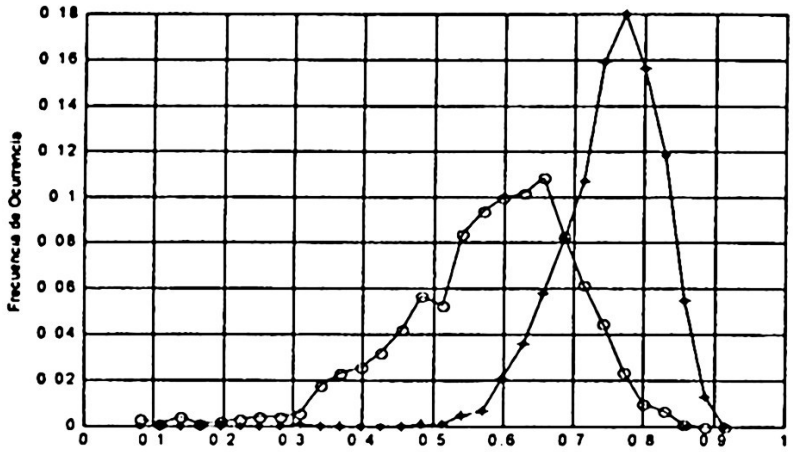


Fig. 1: Histogramas de las correlaciones entre los especialistas (-•-) y entre JF y el promedio de las valoraciones de los especialistas (-o-)

Tabla II

Resultados de la regresión lineal

	b	bint		stats	
1	0.6595	-0.253	1.572	0.3595	$6.67*10^{-6}$
2	0.9946	0.6	1.3886		

4. Discusión

El valor de correlación entre los especialistas en la valoración subjetiva de la aspereza (0.745) es algo superior al reportado por Fukazawa (0.589) pero similar al obtenido por este último (0.714) en cuanto a la valoración del jadeo. La falta de un número mayor de especialistas impide emitir un criterio definitivo, pero estos resultados hacen

pensar que el intervalo de correlación de las valoraciones subjetivas oscile entre 0.6 y 0.7.

Por otra parte, la correlación entre el índice *JF* y el promedio de las valoraciones subjetivas (0.599) es realmente más bajo que el obtenido entre especialistas en este trabajo, pero similar al reportado por Fukazawa entre especialistas. Siguiendo el razonamiento planteado por este último, pudiera considerarse el *JF* como una opción de índice de aspereza en la misma medida que una valoración subjetiva, pues su correlación con estas últimas (0.599) está en el intervalo descrito (0.589-0.745) para estas.

Los histogramas mostrados en la figura 1 ratifican que la correlación entre los especialistas resultó superior. Determinar que este hecho haya sido casual o no requeriría, como se mencionó, de la existencia de un mayor número de especialistas. De hecho, es poco probable que un único índice describa a plenitud el nivel de perceptibilidad de determinada característica, debido a que en la percepción muchas veces influye más de un rasgo. En el mencionado artículo de Fukazawa se deseaba separar precisamente aspereza de jadeo, dos aspectos que tienen cierta relación en cuanto a su origen atribuido a la falta de periodicidad de los pulsos glotales. De cualquier forma, a partir de los resultados obtenidos se puede afirmar que las perturbaciones de frecuencia, cuantificadas a través del *JF*, constituyen un factor de la mayor relevancia en la percepción de la aspereza de la voz.

A esta afirmación contribuyen los resultados de la regresión lineal, pues puede apreciarse que el intervalo de confianza del factor b_2, correspondiente a la pendiente de la recta que relaciona al *JF* con las valoraciones subjetivas, se encuentra dentro de un de ±40% del determinado. Además, la probabilidad de haber obtenido el par de vectores *JF* y *VS* bajo la condición de que no existiera correlación es inferior a 10^{-5}, lo que denota una fuerte interrelación.

5. Conclusiones

Los resultados obtenidos ratifican la relación reportada en la literatura entre aspereza y perturbaciones de la periodicidad. El *JF* muestra una correlación con las valoraciones subjetivas de aspereza comparable a la existente entre especialistas.

Todo lo anterior indica que el método de detección de pulsos glotales empleado permite obtener resultados confiables en cuanto a la correlación entre las perturbaciones de la periodicidad (*jitter* y *shimmer*) y la percepción subjetiva de la aspereza en voces patológicas.

Referencias

[1] Aronson, A.E. "Dysarthria: Differential Diagnosis". Mentor Seminars. Rochester. Michigan. (4 cassettes). 1993.
[2] Baken, R.J. "Clinical Measurement of Speech and Voice". Singular Publishing Group, Inc. San Diego. 1996.
[3] Berry, W.R.; Aronson, A.E.; Darley, F.L. & Goldstein, N.P. "Effects penicillamine therapy and low-copper diet on dysarthria in Wilson's disease (hepatolenticular degenerative)". Mayo Clinic Proceedings. 49, pp 405-408. 1974.
[4] Berry, W.R.; Darley, F.L.; Aronson, A.E. & Goldstein, N.P. "Dysarthria Wilson's disease". Journal of Speech & Hearing Research. 17, pp 169-183. 1974.
[5] Coleman, R.F. "Effect of waveform changes upon roughness perception". Folia Phoniatrica. No 23. pp 314-322. 1971.
[6] Darley, F.L.; Aronson, A.E. & Brown, J.R. "Clusters of deviant speech dimensions in the dysarthria". Journal of Speech & Hearing Research. 12, pp 462-496, 1969.
[7] Darley, F.L.; Aronson, A.E. & Brown, J.R. "Differential diagnostic patterns dysarthria". Journal of Speech & Hearing Research. 12, pp 246-269, 1969.
[8] Darley, F.L.; Aronson, A.E. & Brown, J.R. "Motor Speech Disorders" Philadelphia. Saunders. 1975.
[9] Darley, F.L.; Aronson, A.E. & Brown, J.R. "Motor Speech Signs in Neurologic Disease". Medical Clinics of North America, 52, 835-844. 1968.
[10] Darley, F.L.; Brown, J.R. & Goldstein, N. "Dysarthria in multiple sclerosis" Journal of Speech & Hearing Research. 15, pp 229-245, 1972.
[11] Deal, R.E. & Emanuel, F.W. "Some waveform changes upon roughness perception". Journal of Speech & Hearing Research. 21, pp 250-264, 1978.
[12] Enderby, P. "Frenchay Dysarthria Assessment". California. College Hill. 1983.
[13] Ferrer, C., "Cuantificación de parámetros subjetivos de la voz para el diagnostico de Disartrias". Tesis de Master, UCLV, 2000.
[14] Fukazawa, T.; El-Assuooty, A. & Honjo, I. "A new index for evaluation of the turbulent noise in pathological voice". Journal of the Acoustical Society America. Vol. 83, No 3. pp 1189-1193. March. 1988.
[15] Hernández-Díaz M.E., "Algoritmos para la extracción del período fundamental de la voz. Desarrollo y evaluación". Tesis doctoral. UCLV. Cuba 1996.
[16] Wendall, R.W. "Approach to the objective evaluation of hoarseness". Folia Phoniatrica. No 18. pp 98-108. 1966.
[17] Wendall, R.W. "Laringeal Analog Synthesis of Harsh voice Quality". Folia Phoniatrica. No 15. pp 251-255. 1963.

Carlos Ariel Ferrer Riesgo; Centro de Estudios de Electrónica y Tecnologías de Información (CEETI); UCLV; Carretera a Camajuaní km 5½, Santa Clara; Villa Clara; Cuba; email: cferrer@uclv.etecsa.cu ; Fax: (53) (422)-281608; Tel: (53) (422)-281157

María Esperanza Hernández-Díaz Huici; Centro de Estudios de Electrónica Tecnologías de la Información (CEETI); UCLV; Carretera a Camajuaní km 5½, Santa

Clara; Villa Clara; Cuba; email: mariae@uclv.etecsa.cu ; Fax: (53) (422)-281608; Tel: (53) (422)-281157

Biografía corta de cada autor

Carlos Ferrer Riesgo, nació el 7 de Octubre de 1973. Graduado de Ingeniería en Telecomunicaciones y Electrónica en la UCLV en 1996. Graduado de Master en Telecomunicaciones en la UCLV en 2000. Se desempeña como investigador en el CEETI, UCLV desde 1996. Su área de interés es el procesamiento de voz para aplicaciones médicas.

María Esperanza Hernández-Díaz Huici, nació el 18 de Octubre de 1958. Graduada de Ingeniería en Telecomunicaciones en el ISPJAE en 1981. Obtiene el Título de Doctor en Ciencias en la UCLV en 1997. Labora como profesor asistente en el Centro de Estudios de Electrónica y Tecnologías de la Información (CEETI) de la UCLV desde 1992. Sus intereses principales están en el área del tratamiento digital de señales biomédicas, en particular la voz y la compresión de datos.

A quantitative and qualitative comparison between the empirical mode decomposition and the wavelet analysis

H. Solís-Estrella, A. Angeles-Yreta, V. Landassuri-Moreno
and J. Figueroa-Nazuno
Centro de Investigación en Computación, Instituto Politécnico Nacional
UPALM 07738, Mexico D.F.
{habacuc, malberto, victorm}@sagitario.cic.ipn.mx, jfn@cic.ipn.mx

Abstract. In this paper we compare two techniques developed in recent years to overcome the limitations of conventional approaches, the wavelet transform and the empirical mode decomposition. The wavelet analysis decompose data through multiple scales using a previously selected basis from many available wavelet function families, moreover it has a fast discrete algorithm which is computationally less complex than the Fast Fourier Transform. In the empirical mode decomposition the basis is extracted from the data itself, thus it is optimal for that given data set and provides a more detailed view of the internal dynamics of the process, however it is computationally intensive. First we present a brief theoretical background of both techniques. Then we review some experimental results with complex and chaotic time series.

1. Introduction

In order to obtain useful information from complex data is often desirable to separate the data into its basic components such that those components are easier to examine and give a more meaningful representation of the phenomenon.

Unfortunately the traditional approach – the Fourier analysis and its derivatives – assumes that the data is linear and stationary, this assumption leads to a very poor resolution of the embedded structures of the phenomenon in this new representation, since most of the significant phenomena are non-stationary and present non-linear effects to some degree.

Wavelet analysis is now a well established theory based in the multiresolution concept where a prototype function with compact support is dilated and contracted to see the general behavior and the fine detail in the data

With this approach we can overcome the limitations with non-stationary data that Fourier analysis presents and obtain a time-scale-energy representation that has been very useful in many areas.

The Empirical Mode Decomposition (EMD) is based on the concept of extracting the intrinsic mode functions (IMF) from the data. Such functions have a meaningful instantaneous frequency and in consequence a well behaved Hilbert transform.

The EMD is defined by an algorithm rather than an analytical function, however given that the basis is extracted from the data itself, it is adaptive and in addition to overcome the non stationary limitations, it provides more detail for the non-linear effects in the data set because it extracts the oscillatory modes inherent to the phenomenon.

2. Wavelet analysis

The Continuous Wavelet Transform (CWT) is defined as

$$F_W(a,b) = \langle f, \psi_{a,b} \rangle = \frac{1}{a} \int_{-\infty}^{\infty} f(t) \psi^* \left(\frac{t-b}{a} \right) dt \tag{1}$$

Where ψ is a prototype function called "mother wavelet", b is the location of the analysis window and a is the scale. Such function must fulfill certain requirements such as having finite length and energy, and it has to be an oscillatory function that is well localized both in time and frequency. A wavelet prototype is often called a time-frequency atom.

When we deal with discrete time functions, we can calculate the inner product of the wavelet basis and the data using numerical methods, however it is computationally intensive and it yields redundant data, for that reason there was a need to found a way to construct a discrete transform with a fast algorithm.

Fortunately, a close relationship between wavelets and filterbanks was found. If we define two functions such that

$$\phi(x) = \sum_{n=-\infty}^{\infty} c_n \phi(2x-n) \tag{2}$$

$$\psi(x) = \sum_{n=-\infty}^{\infty} (-1)^n c_{-n+1} \phi(2x-n) \tag{3}$$

a low-pass, perfect reconstruction FIR filter h_0 can be constructed as described in [1].

Once the low pass filter is constructed, using (4) we obtain a quadrature mirror filterbank (QMF), such filterbank is used for analysis and synthesis of the data.

$$h_1(n) = (-1)^n h_0(L-n-1) \tag{4}$$

The Discrete Wavelet Transform (DWT) is then defined as

$$y_0(k) = \sum_{k=-\infty}^{\infty} x(n) \cdot h_0(2k-n) \tag{5}$$

$$y_l(k) = \sum_{k=-\infty}^{\infty} x(n) \cdot h_l(2k - n) \tag{6}$$

The function $y_h(k)$ is called the detail coefficients and it is obtained filtering the original signal with the high pass filter and then downsampling in a dyadic fashion. The function $y_l(k)$ is called the approximation coefficients, it is obtained with the low pass filter, each one of them can be further decomposed with the same filterbank in octave bands or in an arbitrary tree.

This result is of great importance because it means that every orthogonal wavelet basis has an associated filter and thus, a fast discrete transform, furthermore it is also shown that wavelets can be constructed from FIR filters although not every filter can result in a wavelet.

The main drawback in the wavelet analysis is that once a basis is selected, it is used during the whole process, and we cannot assume that a single basis is the best for all the data – especially in non-linear cases – and while the resolution is acceptable, sometimes a better resolution is desirable.

3. Empirical mode decomposition

The empirical mode decomposition was introduced in [3] by Huang et al to obtain an analysis method that was complete, orthogonal, local and adaptive from where time-frequency scales were extracted. The principle of this technique is to decompose a signal x(t) into a sum of intrinsic mode functions such that:

$$x(t) = \sum_{i=1}^{n} imf_i(t) + r_n(t) \tag{7}$$

An intrinsic mode function is a function that satisfies two conditions (1) have the same numbers of zero crossings and extrema; and (2) are symmetric with respect to the local mean. The first condition is similar to the traditional narrow band requirements for a stationary Gaussian process. The second condition modifies the classical global requirement to a local one; it is necessary to eliminate unwanted fluctuations from the instantaneous frequency that asymmetric wave forms introduce.

With this definition, an IMF is not restricted to a narrow band signal, and it can be both amplitude and frequency modulated. This provides more flexibility as one embedded oscillation mode is allowed to have more than one component.

The algorithm for the decomposition is as follows

1. Initialize r(t) = x(t), i = 1
2. Extract the i-th IMF:
   (a) Initialize h0(t) = ri(t), j = 1
   (b) Extract the local minima and maxima of hj-1(t)

(c) Interpolate the local minima and maxima by a cubic spline to form upper and lower envelopes of hj-1(t)
(d) Calculate the mean mj-1(t) of the upper and lower envelopes of hj-1(t)
(e) hj(t) = hj-1(t) – mj-1(t)
(f) If stopping criterion is satisfied then
set imfi(t) = hj(t)
else go to (b) with j = j + 1

3. ri(t) = ri-1(t) – imfi(t)
4. If ri(t) still has two extrema, then
go to 2 with i = i + 1
else the decomposition has finished and ri(t) is the residue.

Step 2 is known as the sifting process, and has two effects: eliminates riding waves; and smoothes uneven amplitudes. Unfortunately, the second effect, when carried to the extreme, could destroy the physically meaningful amplitude fluctuations. To guarantee that the IMF components retain enough physical sense of both amplitude and frequency modulations, in the stopping criterion we have to limit the size of the standard deviation between two consecutive sifting results, the suggested value in the literature is 0.3.

4. Experimental results

We analyzed a set of 30 time series with both techniques, here we present detailed results for one of them, the time series are measurements from El Niño phenomenon of complex nature (Figure 1). All time series consist of one thousand samples.

The complete list of the employed time series and their classification according to [6] and [7] is shown in Table 1.

Table 1. Time series used for the experiments

Time Series	*Nature*	*Time Series*	*Nature*
Sine	periodic	Dow Jones	complex
Vanderpol	periodic	Kobe	complex
Qperiodic2	quasiperiodic	ECG	complex
Qperiodic3	quasiperiodic	EEG	complex
Mackey-Glass	chaotic	ASCII	complex
Logistic	chaotic	El niño	complex
Lorenz	chaotic	HIV DNA	complex
Rossler	chaotic	Human DNA	complex
Ikeda	chaotic	Lovaina	complex
Henon	chaotic	Plasma	complex
Cantor	chaotic	Primes	complex
Tent	chaotic	SP500	complex
A1	complex	Star	complex
D1	complex	Brownian motion	complex
Laser	complex	White Noise	complex

The DWT was computed obtaining six decomposition levels using the 'DB8' wavelet. The EMD was computed until no extrema were found in the residual.

Two interesting cases were the prime number and the Lovaina time series. The prime number time series consists in the distance from one prime number to the next, and the Lovaina time series was generated from ASCII data for the Lovaina University contest.

In the prime number case, there were contiguous identical maxima, consequently the spline interpolation algorithm returned with error.

In the Lovaina case, during the sifting process, the extracted functions never fulfilled the 0.3 standard deviation stopping criterion, so it iterated indefinitely, when the criterion was relaxed to 0.35, the EMD went without trouble and the original time series were reconstructed from the IMFs with no significant distortion

In Figures 2 and 3 we can appreciate how the EMD and the DWT decompose the data going from fine details to a coarse approximation, but the oscillations in each level of decomposition are more contoured in the IMFs.

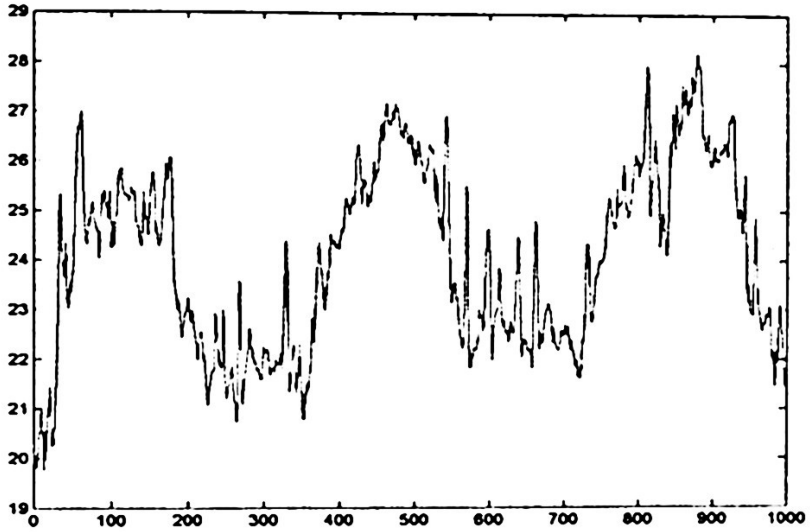


Fig. 1. El Niño time series

Also we can notice how the number of IMFs increases with the intricacy of the data. In Table 2 lists the number of extracted IMFs using the EMD algorithm with each time series.

Unlike the discrete wavelet transform which is dependant on the amount of samples in the time series, the number of intrinsic mode functions extracted from the data with the empirical mode decomposition, is dependant only from the dynamics of the phenomenon. Even the most intricate time series resulted in no more than 16 intrinsic mode functions, and the average number of IMFs is 12.

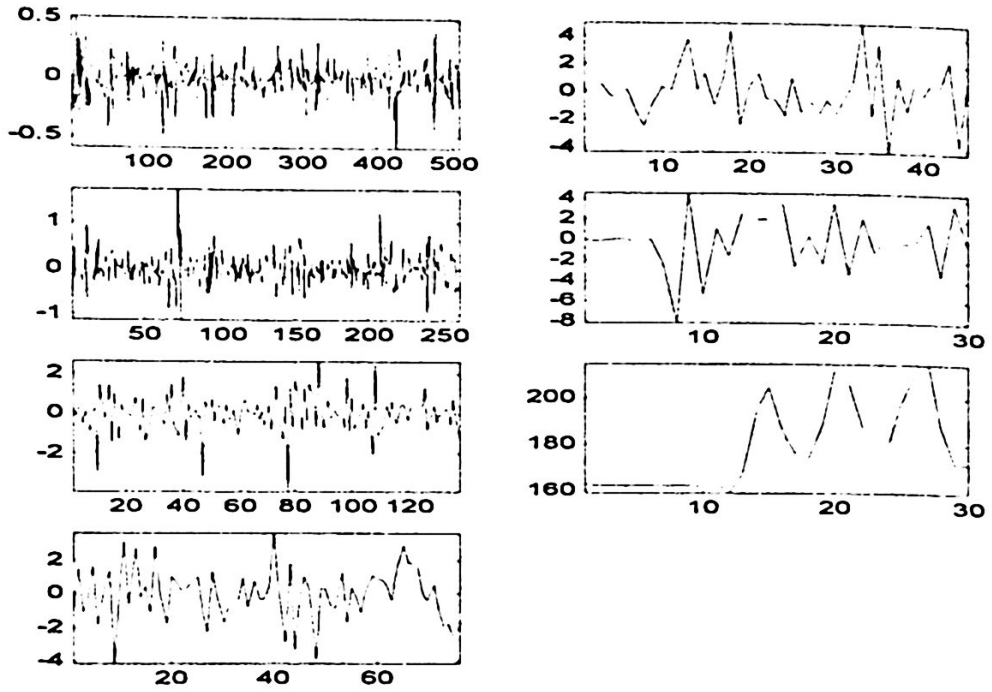


Fig. 2. El Niño DWT detail and approximation coefficients

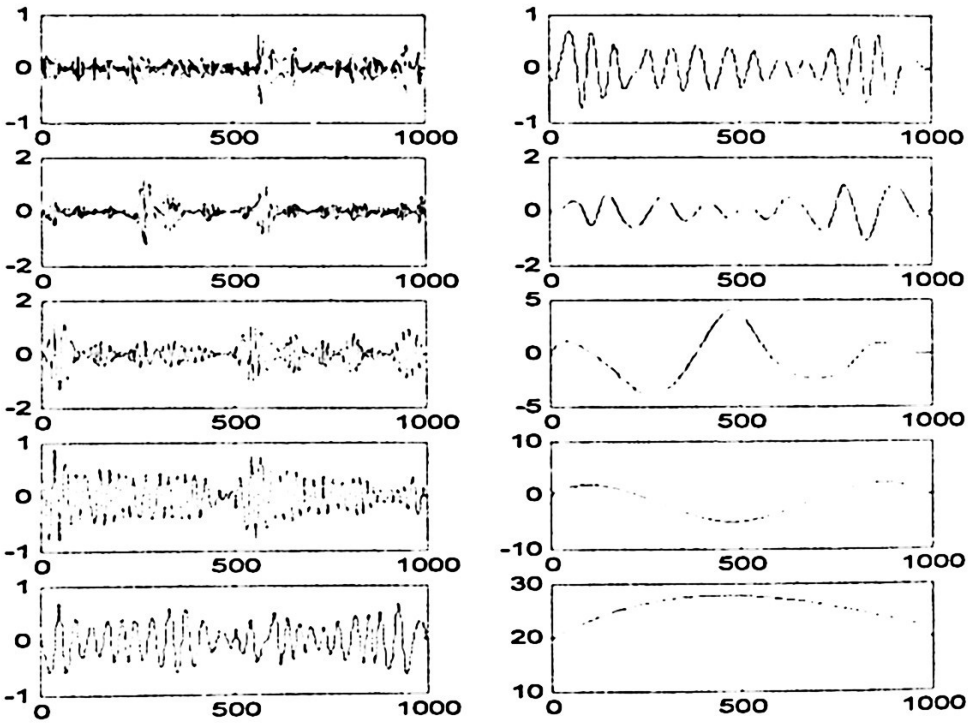


Fig. 3. El Niño IMFs and residual resulting from the EMD

Table 2. Number of extracted IMFs

Time Series	*IMFs*	*Time Series*	*IMFs*
Asciitxt	15	logistic	14
brownian	10	lorentz	7
Cantor	14	lovaina	10
D1	12	mackey	14
Djones	12	plasma	16
Ecg	9	qperiodic2	9
Eeg	13	qperiodic3	12
Elnino	10	rossler	8
henon	16	SP500	15
hivdna	13	Sine	8
humandna	9	Star	13
ikeda	15	Tent	16
kobe	15	vanderpol	6
laser	15	whitenoise	15

Another interesting property of the EMD is that the oscillatory modes it finds may have certain physical meaning, not jus a band of frequencies than comprise the signal and the residual can be interpreted as a trend of the time series.

In the Hilbert spectrum in Figure 4 it is also shown how the energy is more spread in the wavelet analysis while with the IMFs the instantaneous frequencies are very accurate and thus is easier to understand the behavior of the phenomenon as the frequency components at a given instant are evident.

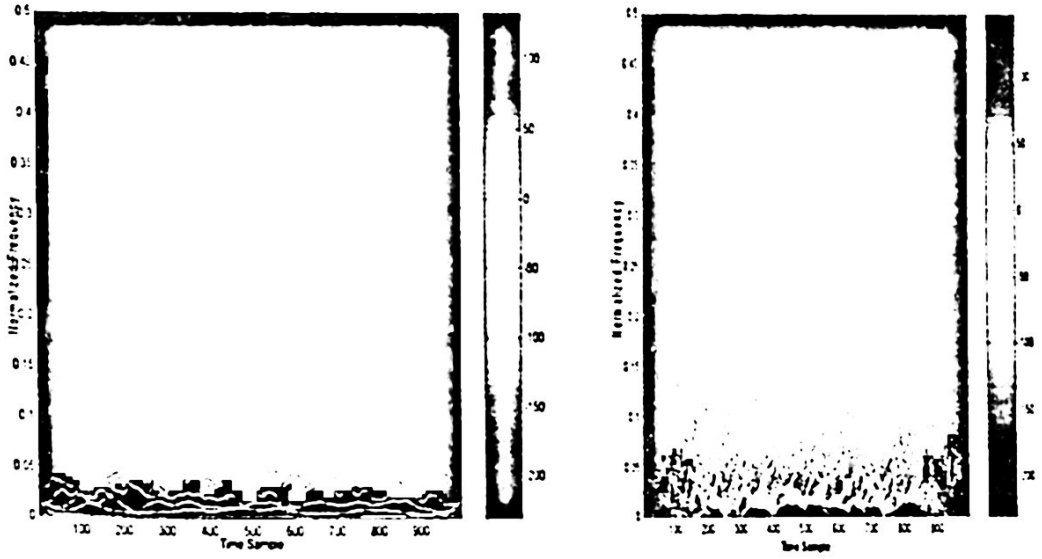


Fig. 4. (a)El Niño IMFs Hilbert Spectrum (b) El niño CWT Hilbert Spectrum

In Figure 5 we can observe that both techniques can be useful to remove certain components from data. First we added Gaussian noise to the ECG time series, to reduce the noise components we disregarded the first five detail levels in the DWT

case, and the first six IMFs of the EMD. After reconstruction, similar results were obtained, the EMD obtained better results reconstructing the main peak of the ECG (the QRS complex) but in the last part, the signal reconstructed from the DWT is less distorted.

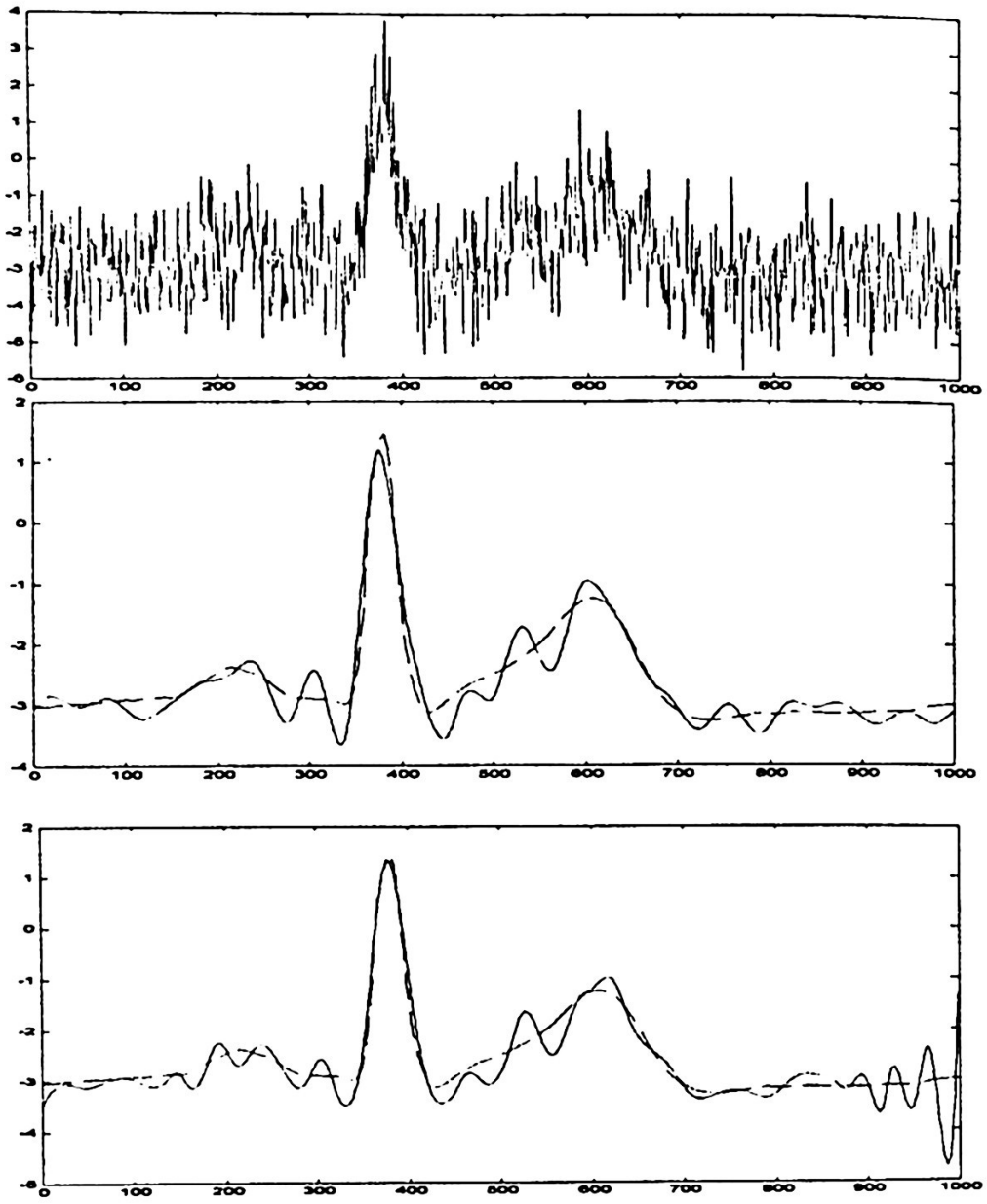


Fig. 5. (a)ECG with additive Gaussian noise, parameters μ=0, σ=1 (b) ECG denoised discarding detail levels (c) ECG denoised discarding IMFs

Finally in Table 3 we can compare the signal to noise ratio (SNR) obtained from the original data and the reconstructed data with both techniques, as we can see, both offer excellent reconstruction and the error could be attributed to the floating point operations but the EMD reconstruction offers much lower distortion probably because the DWT losses energy in the downsampling and upsampling stages

Table 3. SNR for reconstructed data.

Series	*EMD*	*DWT*	*Series*	*EMD*	*DWT*
Asciitxt	315.1785	236.3873	logistic	313.2426	235.9292
brownian	315.6049	225.7096	lorentz	317.2882	222.7929
Cantor	315.414	229.9274	lovaina	315.6988	221.3103
D1	312.3989	230.0365	mackey	315.4807	230.1314
Djones	315.6676	237.8043	plasma	312.5775	223.6647
Ecg	317.1134	232.7189	qperiodic2	318.302	239.3443
Eeg	312.8058	220.5422	qperiodic3	317.4853	243.7349
Elnino	317.4214	245.1368	rossler	316.6886	220.322
henon	314.5757	227.7612	SP500	315.6374	237.043
hivdna	315.3482	232.6521	Sine	317.4877	219.0768
humandna	316.203	247.3116	Star	312.5734	219.2526
ikeda	313.8854	230.1964	Tent	312.8422	251.1903
kobe	314.9108	225.7457	vanderpol	310.5083	219.0846
laser	313.1771	231.8879	whitenoise	313.7753	230.8806

5. Conclusions

We have analyzed several complex and chaotic time series with two relatively recent analysis techniques, the wavelets and the empirical mode decomposition. We observed results in a time-amplitude domain and in a frequency-time-energy domain to compare their behavior resolution, and then synthesized the data from the components to measure the accuracy of the reconstruction they offer.

The EMD is a step further in the classical signal analysis and the resulting decomposition helps to identify underlying structures and trends however with large data sets its complexity and the fact that it needs from ten to fifteen times the size of the original data set can pose a problem. The wavelet analysis is better suited for large amount of streaming data as it has a fast algorithm and only needs a buffer with the same size of the filters and the resulting coefficients have the same length of the original data.

On the other hand, if one wants to synthesize a signal after some processing in the components, the reconstruction is much simpler with the EMD as one must only sum all the IMFs while the inverse discrete wavelet transform (IDWT) involves filtering and dyadic upsampling in each level.
The EMD is another example where an intuitive computational solution provides better results than an analytical one. Regrettably as of today, the EMD is defined only for one-dimensional data.

Although the EMD has no mathematical proof of orthogonality, the reconstructed time series had a better signal to noise ratio than the ones reconstructed from DWT.

Perhaps it is not important which technique is better suited for a given problem but that we have another useful representation of data which gives more chances to obtain interpretations of such data and to understand phenomena.

References

1. M. Vetterli and C. Herley, "Wavelets and Filterbanks: Theory and design", IEEE Transactions on Signal Processing, Vol. 40, No. 9, (1992) 2207-2232.
2. "Special Issue on Wavelets", Proceedings of the IEEE, Vol. 84, No. 4, (1996) 507-687
3. N. E. Huang, et al., "The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis", Proceedings of the Royal Society Lond. A 454, (1998), 903-995.
4. I. Magrin-Chagnolleau and R. G. Baraniuk "Empirical mode decomposition based time-frequency attributes"
5. Patrick Flandrin, "Empirical Mode Decomposition as a Filter Bank", IEEE Signal Processing Letters, (2003)
6. A. Espinosa-Contreras and J. Figueroa-Nazuno, "Análisis del comportamiento de la pérdida de paquetes en la red Internet con técnicas de la dinámica no-lineal" Memorias del Congreso Internacional de Computación CIC2000, (2000), 529-535
7. J. C. Sport "Chaos and time series análisis" Oxford University Press, (2004)

Creación de un sistema experto clasificador de estructuras silábicas para tareas de reconocimiento de voz mediante sílabas

M. en C. José Luis Oropeza Rodríguez, Dr. Sergio Suárez Guerra, M. en C. Ricardo Barrón Fernández,.

Centro de Investigación en Computación.
Departamento de Sistemas Digitales.
Instituto Politécnico Nacional.
Av. Juan de Dios Batís esq. Miguel Otón de Mendizábal, s/n,
j_oropc2002@hotmail.com, ssuarez@cic.ipn.mx, rbarron@cic.ipn.mx.

Resumen .El presente artículo presenta los resultados obtenidos al realizar la programación de un sistema experto para realizar la división de un conjunto de palabras, frases y un archivo de texto en Español en unidades silábicas de acuerdo a las reglas que el idioma tiene para la generación de sílabas dentro de este lenguaje.

La tarea principal del sistema experto es tomar la información proveniente de palabras o frases escritas. con tal información cual se encarga de aplicar las reglas de división silábica para realizar la búsqueda de las diferentes sílabas en el medio de entrada[1]. Dentro del motor de inferencia del sistema experto se encuentran programadas las reglas. La base de conocimiento en este caso es dinámica al igual que el mecanismo de inferencia.

El sistema fue creado con el fin de servir como material de apoyo a las aplicaciones de reconocimiento de voz que utilizan a la sílaba como unidad básica de reconocimiento[2,3,4,5,6,7].

La finalidad es analizar el corpus de entrada y crear todo el conjunto de información de las características de tales unidades, permitiendo con ello establecer el número de elementos de éstas y con ello realizar una conexión con la base de datos del sistema para descargar en ella la información necesaria de cada uno de estos elementos, tanto de los tiempos de inicio y de etiquetado del mismo.

Los valores alcanzados al hacer uso de este sistema experto son altísimos del orden del 98 al 100% dependiendo del tipo de entrada al mismo.

Palabras clave: sílabas, Reconocimiento de voz, sistema experto, reglas de división silábica,

1. Introducción

Como parte esencial del desarrollo del presente trabajo se comenta la elaboración de un sistema experto dedicado a realizar las tareas de clasificación de sílabas en la etapa de entrenamiento del sistema. En este caso, el uso de un sistema experto es con el fin de poder hacer uso de las reglas establecidas para el Español y tratar de acoplar las necesidades de reconocimiento a tales funciones[2,3,4].

Bajo el término de Sistemas Expertos se entiende un nuevo tipo de software que imita el comportamiento de un experto humano en la solución de un problema. Pueden almacenar conocimientos de expertos para un campo determinado y solucionar un problema mediante deducción lógica de conclusiones. En este caso se trata de facilitar la labor de un experto clasificador de sílabas, que recibe la información de un texto de entrada y extrae de él un análisis cuantitativo y cualitativo del mismo.

2. Metodología

CARACTERÍSTICAS DE LOS SISTEMAS EXPERTOS.

Los Sistemas Expertos son una expresión de los sistemas basados en el conocimiento. Con la aplicación de técnicas de Inteligencia Artificial finaliza la transición del procesamiento de datos al procesamiento de conocimientos.

Los sistemas expertos se aplican por norma general en problemas que implican un procedimiento basado en el conocimiento. Un procedimiento de solución basado en el conocimiento comprende las siguientes capacidades:

- Utilización de normas o estructuras que contengan conocimientos y experiencias de expertos especializados.
- Deducción lógica de conclusiones.
- Capaz de interpretar datos ambiguos.
- Manipulación de conocimientos afectados por valores de probabilidad.

LA FUNCIÓN DE UN SISTEMA EXPERTO

La función de un Sistema Experto es la de aportar soluciones a problemas, como si de humanos se tratara, es decir, capaz de mostrar soluciones inteligentes. Esto es posible gracias a que el sistema se crea por expertos (humanos), que intentan estructurar y formalizar conocimientos poniéndolos a disposición del sistema, para que éste pueda resolver una función dentro del ámbito del problema, de igual forma que lo hubiera hecho un experto[8,9,10].

Poder acceder a los conocimientos adquiridos por experiencia es lo más difícil, ya que los expertos, al igual que otras personas, apenas los reconocen como tales. Son buscados con mucho esfuerzo y cuidado siendo descubiertos de uno en uno, poco a poco.

COMPONENTES DE UN SISTEMA EXPERTO

Una característica decisiva de los Sistemas Expertos es la separación entre conocimiento (reglas, hechos) por un lado y su procesamiento por el otro. A ello se añade una interfaz de usuario y un componente explicativo.

- La Base de Conocimientos de un Sistema Experto contiene el conocimiento de los hechos y de las experiencias de los expertos en un dominio determinado.
- El Mecanismo de Inferencia de un Sistema Experto puede simular la estrategia de solución de un experto.
- El Componente Explicativo explica al usuario la estrategia de solución encontrada y el por qué de las decisiones tomadas.
- La Interfaz de Usuario sirve para que éste pueda realizar una consulta en un lenguaje lo más natural posible.
- El Componente de Adquisición ofrece ayuda a la estructuración e implementación del conocimiento en la base de conocimientos.

LA BASE DE CONOCIMIENTOS

La Base de conocimientos contiene todos los hechos, las reglas y los procedimientos del dominio de aplicación que son importantes para la solución del problema[8,9,10].

La base de conocimientos dispone de reglas. Estas reglas se representan en forma de:
Si *premisas* Entonces *Conclusión y/o Acción.*

En la zona de las premisas se solicitan vinculaciones lógicas referentes a las cualidades de los objetos.

En la zona de la conclusión se añaden nuevos hechos y cualidades a la base de conocimientos y/o se ejecutan acciones. Esto se define a menudo como programación orientada a reglas.

En nuestro caso la base de conocimientos se encuentra constituida por todas las reglas de clasificación de sílabas del lenguaje Español que se analizan a continuación:

LAS REGLAS DE LA SÍLABA.

Palabras monosilábicas:

Son las que están formadas por una sílaba: luz, mar.

Palabras bisilábicas:

Son las que están formadas por dos sílabas: silla, mesa.

Palabras trisilábicas:
Son las que están formadas por tres sílabas: ventana, cabeza.

Palabras polisilábicas:

Son las que están formadas por cuatro o más sílabas: Argentina, Polideportivo.

En [12] se menciona que en el Español existen 29 letras, las cuales están clasificadas de acuerdo a su pronunciación en dos grupos: vocales y consonantes. En la figura 1 se muestra un esquema de la clasificación de las letras para una mejor comprensión. El grupo de las vocales está formado por cinco letras, su pronunciación no dificulta la salida del aire. La boca actúa como una caja de resonancia abierta en menor o mayor grado y de acuerdo a esto, las vocales se clasifican en abiertas, semiabiertas y cerradas[2][3].

El otro grupo de letras, las consonantes, está formado por veinticuatro letras, de las cuales tres son consonantes compuestas, llamadas así, por ser letras simples en su pronunciación y dobles en su escritura. Las letras restantes son llamadas consonantes simples, por ser simples en su pronunciación y en su escritura.

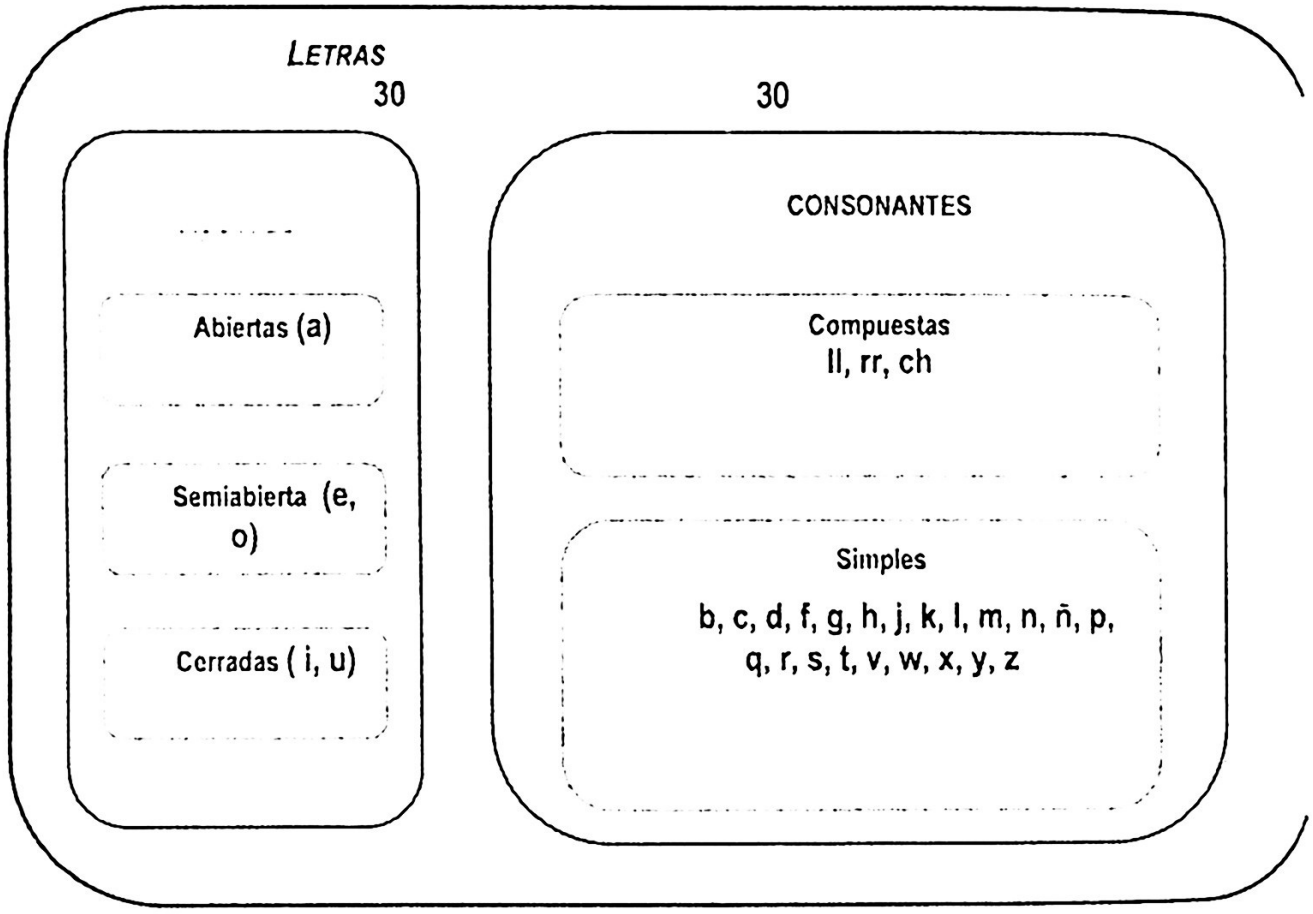


Fig. 1 Esquema de la clasificación de las letras.

Las sílabas

En el idioma español existen once reglas, las cuales determinan la separación de las sílabas de una palabra. Estas reglas son listadas a continuación mostrando enseguida excepciones a la misma.

REGLA 1

En las sílabas, por lo menos, siempre tiene que haber una vocal. Sin vocal no hay sílaba.

REGLA 2

Cada elemento del grupo de consonantes inseparables, mostrado en la figura 3.6, no puede ser separado al dividir una palabra en sílabas.

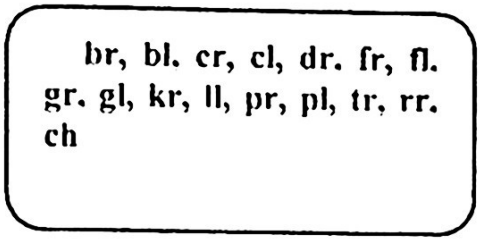


Fig. 3.6. Grupo de consonantes inseparables.

REGLA 3

Cuando una consonante se encuentra entre dos vocales, se une a la segunda vocal.

REGLA 4

Cuando hay dos consonantes entre dos vocales, cada vocal se une a una consonante.

Excepción: Esto no ocurre en el grupo de consonantes inseparables (regla 2).

REGLA 5

Si son tres las consonantes colocadas entre dos vocales, las dos primeras consonantes se asociarán con la primera vocal y la tercera consonante con la segunda vocal.

Excepción. Esta regla no se cumple cuando la segunda y tercera consonante, forman parte del grupo de consonantes inseparables.

REGLA 6

Las palabras que contienen una h precedida o seguida de otra consonante, se dividen separando ambas letras.

REGLA 7

El diptongo es la unión inseparable de dos vocales. Se pueden presentar tres tipos de diptongos posibles:

- Una vocal abierta + Una vocal cerrada.
- Una vocal cerrada + Una vocal abierta.
- Una vocal cerrada + Una vocal cerrada.

Son diptongos sólo las siguientes parejas de vocales: ai, au, ei, eu, io, ou, ia, ua, ie, ue, oi, uo, ui, iu, ay, ey, oy .

La unión de dos vocales abiertas o semiabiertas no forma diptongo, es decir, deben separarse en la segmentación silábica. Pueden quedar solas o unidas a una consonante.

REGLA 8

La h entre dos vocales, no destruye un diptongo. *Ejemplo:*

Palabra	sílabas
ahuyentar	ahu + yen + tar

REGLA 9

La acentuación sobre la vocal cerrada de un diptongo provoca su destrucción.

Palabra	sílabas
María	Ma + rí + a

REGLA 10

La unión de tres vocales forma un triptongo. La única disposición posible para la formación de triptongos es la siguiente:

Vocal cerrada + (vocal abierta | vocal semiabierta) + vocal cerrada

Sólo las siguientes combinaciones de vocales, forman un triptongo: iai, iei, uai, uei, uau, iau, uay, uey.

3. El mecanismo de inferencia.

El mecanismo de inferencia es la unidad lógica con la que se extraen conclusiones de la base de conocimientos, según un método fijo de solución de problemas que está configurado imitando el procedimiento humano de los expertos para solucionar problemas. Una conclusión se produce mediante aplicación de las reglas sobre los hechos presentes[8,9,10].

Ejemplo.

Una Regla es: Si p y q entonces r
p y q son justo aquellos hechos que se mencionan en la cláusula "si" de la regla, es decir, las condiciones para la aplicabilidad de la regla. Aplicar la regla es: deducir de los hechos p y q el hecho r.

En un Sistema Experto existirá un hecho sólo cuando esté contenido en la base de conocimientos.

Los hechos que constan en la cláusula "si" se llaman *premisas*, y el contenido en la cláusula "entonces", se llama *conclusión*. Cuando se aplica una regla sobre algunos hechos cualesquiera se dice que se *dispara*. El disparo de una regla provoca la inserción del nuevo hecho en la base de conocimientos.

Las funciones del mecanismo de inferencia son:

- Determinación de las acciones que tendrán lugar, el orden en que lo harán y cómo lo harán entre las diferentes partes del Sistema Experto.
- Determinar cómo y cuándo se procesarán las reglas, y dado el caso también la elección de qué reglas deberán procesarse.
- Control del diálogo con el usuario.

La decisión sobre los mecanismos de procesamiento de reglas, es decir, qué estrategias de búsqueda se implementarán, es de vital importancia para la efectividad del sistema en su conjunto.

4. Resultados

La implantación del sistema experto para el presente trabajo tiene como entrada el conjunto de frases o palabras que conforman un vocabulario determinado a reconocer, tras la aplicación de las reglas pertinentes, se procede a corroborar cada uno de los elementos de entrada, con lo cual se puede lograr establecer los inicios y finales de una sílaba y la otra.

Lo anterior cumple con el objetivo de que en el reconocimiento se extraigan la cantidad y tipos de sílabas que conforman el corpus y asimismo provee la cantidad de bloques que serán usados en la etapa de segmentación y etiquetado silábico de las señales de voz. Todo lo anterior, permite encontrar e indicar un parámetro de referencia en los siguientes puntos:

Cuando se utiliza en el entrenamiento, se procede a concordar el número de sílabas extraídas del experto con el número obtenido de las estructuras energéticas que se analizan para realizar tal segmentación y clasificación. En caso de no coincidir se desecha la muestra analizada y se retoma una nueva.

Cuando se lleva a cabo el reconocimiento del habla discontinua, se hace uso de tal segmentación, para extraer cada uno de los componentes de las frases o de las sílabas para ser analizados de forma independiente por cada uno de los algoritmos de reconocimiento y establecer por tanto la decisión de cual será la decisión a tomar por el sistema de reconocimiento.

IMPLANTACIÓN DEL SISTEMA EXPERTO.

Básicamente y debido a que la finalidad es crear el sistema experto que se acople a las necesidades del tipo de reconocimiento que estamos tratando de realizar, se procedió a realizar el diseño y la programación del mismo usando lenguaje C para ello. Las principales características de tal programa se describen a continuación[8,9,10]:

- Posee una toma de decisión de tipo cualitativo y cuantitativo del tipo de sílabas que comprenden la entrada que se le está otorgando.
- La entrada al sistema se produce a través del software diseñado para las aplicaciones referentes a los fines de investigación precisos.
- En caso de ser frases, el sistema se encarga de realizar su separación en palabras para después separarlas por sílabas.
- La base de conocimiento es usada para poder realizar tal tarea de segmentación, se tiene como referente las reglas del Español para la obtención de sílabas. EL uso de las sentencias if proposición then, son el estándar usado en esta capa del sistema.
- La base de programación está realizada en estructuras dinámicas de datos del tipo lista enlazada. Las cuales nos permiten ir accediendo a cada uno de los elementos de la palabra para posteriormente realizar la segmentación adecuado de la palabra según sea necesario.

A continuación se muestran dos segmentos del código del sistema experto en cuestión:

```
void CDivide_elemento::inicializa (void)
{
posiciones_corte[0]=elemento;
elemento++;
  for (i=0;i<20;i++)      reglas[i]=false;
cons_insepa="brblcrcldrfrflgrglkrllprpltrrrchtl";
vocal_diptongo="aiaueieuioouiauaieueoiuouiiuayeyoy";
vocal_abierta="íaíoíeíuúaúeúoúiooaeaoeaeooaoeee";
vocales="aeiouíúáéó";
}
```

Este método se encarga de inicializar los elementos necesarios para analizar dentro de una palabra la existencia de elementos tales como: las consonantes inseparables, las vocales diptongo, las vocales abiertas, las vocales, etcétera.

/*aquí se comienza a analizar la regla 2, después de ella se identificará si existen consonantes inseparables en la cadena de texto que se está analizando*/

```
   r=inicio
  do
  {
    ptr1=*r->fonema;
   if (ptr1==*"C")
    {
      p2=r->next;
      ptr2=*p2->fonema;                                    if(ptr1==ptr2)
        {
      for (i=0;i<int(strlen(cons_insepa));i+=2)
        {
          if(cons_insepa[i]==palabra[contador]                              &&
cons_insepa[i+1]==palabra[contador+1])
            {
  conta_conso_insepa[cuenta_insepa]=contador;
            cuenta_insepa++;
            reglas[1]=true;
  if(verifica_existencia(posiciones_corte,elemento,contador))
   {
        posiciones_corte[elemento]=contador;
         elemento++;
                                                        }
            }
          }
        }
      }
  r=r->next;
  contador+=1;
  }while (r->next!=NULL);
```

Las sentencias anteriores permiten observar la forma en la cual se trabaja sobre el conjunto determinado de elementos de entrada para poder realizar las tomas de decisiones adecuadas, el uso de banderas que representan las reglas que se activan conforman los resultados de la aplicación de los mecanismos de inferencia que internamente se han aplicado. Cuando estas reglas satisfechas se contrarrestan, se procede a analizarlas de forma posterior en otra etapa posterior del mismo sistema.

Los resultados obtenidos en la división silábica al hace uso de este sistema en los diferentes corpus usados a lo largo del presente trabajo se resumen en la siguiente tabla, manifestándose en la misma su porcentaje de efectividad alcanzado.

TIPO DE CORPUS	PORCENTAJE DE EFECTIVIDAD
Sílabas independientes	100%
Dígitos	100%
Corpus de palabras aisladas	100%
Corpus de Latino40	100%
Corpus de frases simples	100%

Tabla 5.1. porcentaje de efectividad en la aplicación del sistema experto a diferentes corpus de voces.

5. Discusión

La inclusión de un sistema experto a un sistema de reconocimiento de voz no es nuevo, técnicas empleadas con anterioridad de la IA (Inteligencia Artificial) demuestran este hecho, lo importante en estos momentos es la oportunidad de intercalar las reglas de las sílabas a los sistemas basados en los mismos.

Los resultados obtenidos en el presente reporte servirán para poder analizarlos e incrustarlos en tareas de reconocimiento.

6. Conclusiones

La incrustación de los sistemas expertos para las tareas de reconocimiento de voz por computadora, representa una alternativa para conseguir obtener mejores rendimientos de su uso. En este trabajo hicimos uso de las características y beneficios del sistema experto para clasificar un conjunto de elementos de entrada (frases, palabras, etc.,) en unidades silábicas.

La tarea principal fue programar al experto con la base de conocimiento de las reglas de las sílabas y sobre la base de este parámetro, realizar la clasificación adecuada. Como se observa en los resultados, el experto clasificador alcanzó altos rendimientos en cuanto a identificar de forma adecuada el número de sílabas total del elemento de entrada y además asignarlas a cada una de las estructuras silábicas correspondientes.

Dado lo anterior se tiene la posibilidad de encontrar todos estos parámetros una vez que se ha alimentado al sistema de forma adecuada.

Referencias

[1]Speech Recognition Using Syllable-Like Units, Zhihong Hu, Johan Schalkwyk, Etienne Barnard, Ronald Cole. Center for Spoken Language Understanfing. Oregon Graduate Institute of Science and Technology. 1998.

[2]Análisis del Español Mexicano, para la construcción de un sistema de reconocimiento de dicho lenguaje, Ben Serridge, GRUPO TLATOA, UDLA, Puebla México 1998.

[3]Syllable Structure Constraints, A C/D Model Perspective, Osamu Fujimura. 1998.

[4]Sobre el uso de la sílaba como unidad de síntesis en el Español. Feal L. Pinto, Informe técnico, Departamento de Informática, Universidad de Valladolid, 2000.

[5]Continuos speech recognition using automatically segmented data at syllable units. V. Kamakshi Prasad, T. Nagarajan and Hema A. Murphy. Department of Computer Science and Engineering, Indian Institute of Technology, Madras, Chennai, ICSP, 2002.

[6]Integrating Syllable Boundary Information Into Speech Recognition. Su-Lin Wu, Michael .

[7]Pruebas y validación de un sistema de reconocimiento del habla basado en sílabas con un vocabulario pequeño. Sergio Suárez Guerra, Karen Suso, Mariana del Villar, Congreso Internacional de Computación CIC2003. México, D.F.

[8]Jesus Savage Carmona, A Hybrid Systems with Symbolic AI and Statical Methods for Speech Recognition, University of Washington, 1995.

[9]Stuart Russell, Peter Norvig, Inteligencia Artificial un enfoque moderno, Prentice Hall, 1996.

[10Joseph Giarratano y Gary Riley, International Thompson Editores, Sistemas expertos, principios y programación 2001.

José Luis Oropeza Rodríguez; Centro de Investigación en Computación (CIC); IPN; Av. Juan de Dios Batís esq. Miguel Otón de Mendizábal, s/n, México; email: j_orope2002@hotmail.com; Tel: 57296000 ext. 56519

Sergio Suárez Guerra; Centro de Investigación en Computación (CIC); IPN; Av. Juan de Dios Batís esq. Miguel Otón de Mendizábal, s/n, México; email: ssuarez@cic.ipn.mx; Tel: 57296000 ext. 56519

Ricardo Barrón Fernández; Centro de Investigación en Computación (CIC); IPN; Av. Juan de Dios Batís esq. Miguel Otón de Mendizábal, s/n, México; email: rbarron@cic.ipn.mx; Tel: 57296000 ext. 56519

Biografía corta de cada autor

José Luis Oropeza Rodríguez, nació el 30 de Noviembre de 1972. Graduado de Ingeniería en Telecomunicaciones y Electrónica en la ESIME UC, en 1994. Graduado de Master en Ingeniería de Cómputo en el CIC en 2000. Se desempeña como alumno de doctorado en el CIC, IPN desde 2000. Su área de interés es el reconocimiento de voz.

Dr. Sergio Suárez Guerra. Graduado de Ingeniero Electricista en el Instituto Politécnico José Antonio Echeverría en La habana, Cuba en 1972, Doctor en Ciencias en el Centro de Cálculo de la Academia de Ciencias de la URSS en 1979. Profesor Investigador del Instituto Politécnico Nacional desde 1998. Su área de interés: Procesamiento digital de señales, en especial Voz.

M. en C. Ricardo Barrón Fernández. Graduado de Matemático en la Facultad de Ciencias de La Universidad Nacional Autónoma de México en 1985, Maestro en Ciencias en el Centro de Investigación en Computación del Instituto Politécnico Nacional (IPN) en 1998. Profesor Investigador del IPN desde 1986. Su área de interés: Procesamiento digital de señales, imágenes y voz.



Images retrieval using wavelets, histograms and sub images

A. C. González G.[1,2], J. H. Sossa A.[2], E. M. Felipe R.[2]
[1] Instituto Tecnológico de Toluca, Depto. de Ingeniería Eléctrica y Electrónica
Av. Instituto Tecnológico S/N Ex Rancho La Virgen, Metepec, México, C.P. 52140
alaing@ittoluca.edu.mx, alaing@sagitario.cic.ipn.mx
[2] Instituto Politécnico Nacional, Centro de Investigación en Computación
Av. Juan de Dios Bátiz esq. Miguel Othón de Mendizábal, C.P. 07000, México, D.F.
hsossa@cic.ipn.mx, edgardo@cic.ipn.mx

Abstract. In this paper are presented three methods of images retrieval through the Daubechies 4 Wavelet Transform in the color space RGB; two of global characteristics, which they are: 1. - Extraction Wavelet Coefficients of the all images, 2. - Extraction Wavelet Coefficients of the histogram of a circular window circumscribed in the image; and one of local characteristics, 3. - Extraction Wavelet Coefficients of sub images histograms disjoints distributed of uniform way in the entire image. It is presented the system that contains the three methods and the results upon proving with different data bases of images.

1. Introduction

The information processing often involves the recognition, storage and visual information retrieval.

It is conceived that an image contains visual information and that what is important within such information retrieval process is to return the original image or a group of images with similar information [5][8]. The images retrieval is referred to seek and recover visual information in form of images, within a collection of data bases of images, being one of their investigation areas the organization and retrieval based on the content, in color terms.

The organization and retrieval based on the content in color terms is used as a means of codification through the organization of the information of an image using the Wavelets coefficients and as technical of retrieval spatial relationships through the Euclidian distance.

1.1 Justification

In the article "Next generation Web searches for visual contents", of the magazine Computer [3], in their pages 46-53, is appointment that, though the objective principal of a search machine is to facilitate to find it text in the Web, in digital libraries, Intranets, on the image data bases, etc., these have few or no capacity of finding images or video.

Many user of the Web, newspaper publishers, or professional, designers of site Web need to find images using few global characteristics. With hundred of million of sites to seek and with 73 percent of the Web devoted to images, to find it the necessary exact image can be a task very complex.

As is observed, this investigation line is important, since the relevancy of the problem bases in the great quantity of existing visual information in the Web, the which is not structured.

For the foregoing, in the development of a methodology for the images retrieval in banks of image data bases, it must be made emphasize, first in the estructuration of the images space, and second in the information technologies to use in the retrieval process given a query image.

2. Methodology

Due to the fact that the images in a data base of images can have different formats and different sizes, it should be to normalize the scale so that the comparison between images will be possible. The only one purpose to normalize images scales is to make possible the use of the Wavelet transform and with this to normalize the characteristic vectors. The images can be normalized to any size, where the length of each side is a power of two [9], for example 256.

In the process of normalizing images scales, is made a sampling of the entry image overlapping a grid, for example of 256 x 256 points. This gives a point of the grid by each pixel in the exit image. The entry image is sampled in each point of the grid to determine the colors of the pixel in the exit image. When the points of the grid lie between centers of pixels, the color values of the point of the grid are determined by bilinear interpolation between the colors of adjacent pixels (both vertical and horizontally).

Is expressed the general methodology to use for the images retrieval; first is opened the size image n x m in some graphic format from among the following: BMP, JPG, PNG and ICO; immediately it is used bilinear interpolation to normalize the picture size to 256 x 256 pixels, these common steps, are observed in the three methods, in addition to that the three use the discreet algorithm of the Wavelet transform.

2.1 Extraction of Wavelet Coefficients of the images (EWCI)

It is intended indexer images by color distribution. The information of color distribution is an alternative to describe the content of an image. If is required to recognize objects as content in an image, is proposed to use Wavelet transform, to decompose the image in characteristic vectors of color distributions and to detect through these the presence of certain objects in the images [7][10].

The process of indexed applies a Daubechies 4 Wavelet transform for each one of the three-color components of the RGB space.

The Wavelet coefficients of the frequency bands more lowers, they are stored as characteristic vectors.

In the Wavelet transform matrix, the sub matrixes of the left upper corner, represent the frequency bands more lowers of the image 2-D, and the sub matrixes of the straight inferior corner represent the highest frequency bands.

Generally the frequency bands more lowers into the Wavelet transform represent configurations of objects in the images, and the highest frequency bands represent texture variations.

It is developed a indexed for color distribution using the Daubechies 4 Wavelet transform, which represents the semantics of an image, to know, the configuration of the objects and the local color variation, both represented by Daubechies 4 Wavelet coefficients.

An image file in graphic format contains samples in the dominance of the time; therefore to filter them (dominance of the frequency), is accomplished an operation of convolution. Since the Wavelet transform is a linear transformation, the transformation of a two-dimensional sign can be calculated to the convolution first in horizontal address and then in vertical address, or conversely.

It is part of a image of size 256 x 256, is applied in the horizontal address the discreet algorithm of the Wavelet transform, such be has a image size of 128 x 256, now the discreet algorithm of the Wavelet transform is applied in the vertical address to obtain a size image 128 x 128, thus from now on to obtain images of sizes 64 x 64, 32 x 32, 16 x 16, 8 x 8 and 4 x 4. This is made in the flat of three colors RGB; to obtain 16 coefficients by each color plan RGB [11].

2.2 Extraction of Wavelet coefficients of the histogram of a circular window (EWCHCW)

In this method are analyzed the images in global form, seeking invariance to the rotation [1]. First is opened a window in circular form registered in the image of size 256 x 256, or rather, the diameter of the circle is of 256.

Below it is proceeded to calculate the histograms of the three colors in the flat RGB of the image in the circular window, are normalized the values of the three histograms, to proceed to apply the discreet algorithm of the Daubechies 4 Wavelet transform of 256 original values of the histograms, first of 256 to 128, of 128 to 64, of 64 to 32 and of 32 to 16, for thus to count finally on 16 wavelet coefficients in each color in the flat RGB.

2.3 Extraction of Wavelet Coefficients of histograms of sub images (EWCHS)

In this method is analyzed the image through local characteristics, for something which the image of size 256 x 256 is split into sub images of size 64 x 64, covering to

all the image with 16 sub images disjoints [4][6]; of these sub images are calculated their histograms, and such be have 16 histograms of 256 values; immediately it is applied the discreet algorithm of the Daubechies 4 Wavelet transform, to obtain 256 original values, first from 256 to 128, from 128 to 64, from 64 to 32, from 32 to 16, from 16 to 8 of 8 to 4 and from 4 to 1. Finally they have to be by each sub image a wavelet coefficient; in total 16 wavelet coefficients in each color plan RGB.

2.4 To store processing or to consult

To store the processing are applied the three methods described in the points 3.2, 3.3 and 3.4, for each one of the flat three of color RGB. The low-frequency coefficients of the Wavelet transform are stored as characteristic vectors, depending on the specific method [2].

To consult, the Euclidian distance between the characteristic vectors of the images previously processed and that of the query image are compared and the images with distances smaller are selected and shown as images of matching to the query image.

3. Tests, results and conclusions

The tests consist of making queries through an image. It is considered to seek in the images data base 8 images that more are seemed to her visually, the user of the system determines the degree of seemed between the query image and 8 images that shows the system for each one of the three methods, determining thus the percentage of efficiency.

The three methods, sections 3.2, 3.3 and 3.4, were proven with the following bases of images data

3.1 Images Library of objects of the University of Columbia [12]

It is considered 7000 images of different real objects isolated in different positions against contrasting homogeneous fund, such as those, which are observed in the figure 2.



Fig. 1. Some objects of the Library of the University of Columbia

Description of the test: They were accomplished 25 experiments of 10 images at random, each one of the same images data base, or rather, that the query images are found previously processed together with the rest of the images of the data base. Therefore they were analyzed 250 images obtaining the averages from the efficiency results from 25 experiments.

The final efficiency results of the three methods are shown in table No. 1, and in table No. 2 is shown the average of an experiment of 10 images of the 25; the entire value expresses how many images of the images data base are seemed of a total of 8 to the query image; table No. 3 expresses the average of 25 experiments.

Table 1. Final efficiency results

Extraction of Wavelet Coefficients of the images (EWCI)	0.355
Extraction of Wavelet Coefficients of the histogram of a circular window (EWCHCW)	0.575
Extraction of Wavelet Coefficients of histograms of sub images (EWCHS)	0.75

Table 2. Results of an experiment of the 25, in each one of the three methods

Query	EWCI	EWCHCW	EWCHS
1	1	3	5
2	2	5	6
3	1	4	5
4	2	4	6
5	3	5	5
6	2	3	6
7	3	4	6
8	1	3	7
9	2	5	6
10	3	4	8
Averages	0.25	0.5	0.75

Table 3. Results of 25 experiments of 10 queries in each one, of the three methods

Experiment	EWCI	%	EWCHCW	%	EWCHS	%
1	2	0.25	4	0.5	5	0.625
2	1	0.125	4	0.5	5	0.625
3	3	0.375	4	0.5	5	0.625
4	2	0.25	5	0.625	5	0.625
5	2	0.25	5	0.625	4	0.5
6	1	0.125	4	0.5	6	0.75
7	3	0.375	3	0.375	7	0.875
8	3	0.375	4	0.5	7	0.875
9	3	0.375	5	0.625	7	0.875
10	3	0.375	3	0.375	8	1.0
11	3	0.375	5	0.625	4	0.5
12	2	0.25	4	0.5	4	0.5
13	2	0.25	4	0.5	4	0.5
14	3	0.375	4	0.5	5	0.625
15	1	0.125	4	0.5	5	0.625
16	2	0.25	5	0.625	4	0.5
17	3	0.375	5	0.625	5	0.625
18	3	0.375	7	0.875	7	0.875
19	4	0.5	7	0.875	7	0.875
20	5	0.625	6	0.75	8	1.0
21	4	0.5	4	0.5	8	1.0
22	4	0.5	4	0.5	8	1.0
23	5	0.625	7	0.875	6	0.75
24	4	0.5	3	0.375	8	1.0
25	3	0.375	5	0.625	8	1.0
Averages	0.355	0.355	0.575	0.575	0.75	0.75

3.2 Library of planes images of Aircraft -Images.co.uk [13]

Consist of 1068 real planes images, all different mutually, as those, which are observed, in the figure 3.

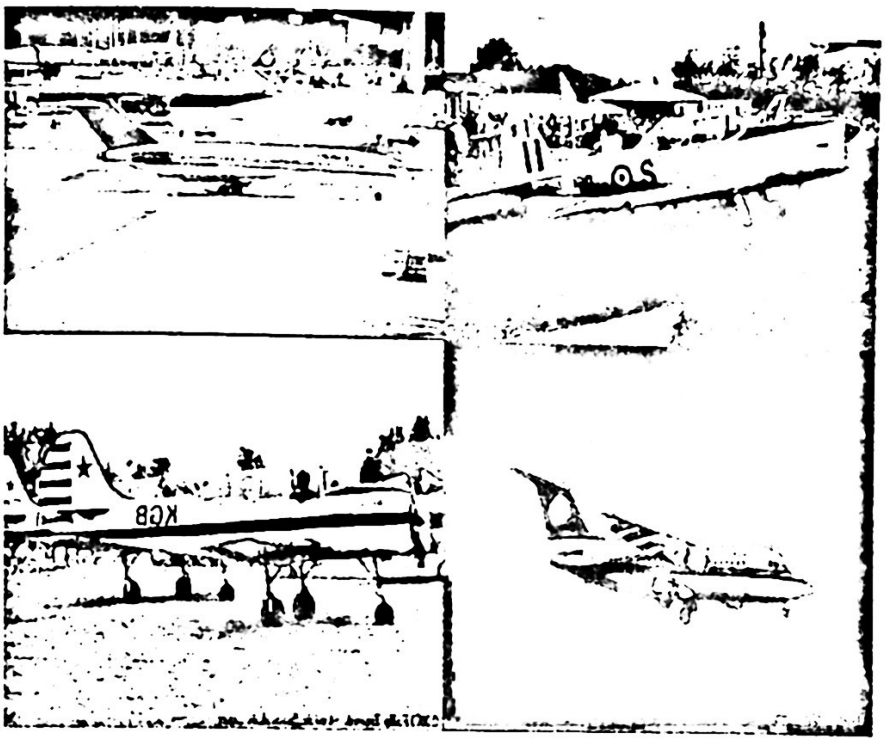


Fig. 2. Some planes of the Library Aircraft-Images.co.uk

Description of the test: They were accomplished 30 experiments of 10 images at random each one, or rather, that the query images are not found previously processed together with the rest of the images data base. Therefore they were analyzed 300 images obtaining the averages from those of the efficiency results from 30 experiments. The final efficiency results of the three methods are shown in table No. 4, and in table No. 5 is shown the average of an experiment of 10 images of 30, the entire value expresses how many images of the images data base are seemed of a total of 8 to the query image; table No. 3 expresses the average of 30 experiments.

Table 4. Final efficiency results

Extraction of Wavelet Coefficients of the images (EWCI)	0.44583
Extraction of Wavelet Coefficients of the histogram of a circular window (EWCHCW)	0.48333
Extraction of Wavelet Coefficients of histograms of sub images (EWCHS)	0.5125

Table 5. Results of an experiment of 30, in each one of the three methods

Query	EWCI	EWCHCW	EWCHS
1	2	2	5
2	3	5	6
3	4	5	5
4	5	2	6
5	2	5	4
6	3	4	5
7	2	5	5
8	3	4	5
9	4	5	5
10	2	3	4
Averages	0.375	0.5	0.625

Table 6. Results of 30 experiments of 10 queries in each one, of the three methods

Experiments	EWCI	%	EWCHCW	%	EWCHS	%
1	1	0.125	0	0.0	3	0.375
2	4	0.5	5	0.625	4	0.5
3	3	0.375	5	0.625	4	0.5
4	6	0.75	1	0.125	4	0.5
5	1	0.125	6	0.75	4	0.5
6	3	0.375	5	0.625	4	0.5
7	2	0.25	4	0.5	1	0.125
8	4	0.5	5	0.625	6	0.75
9	3	0.375	5	0.625	4	0.5
10	2	0.5	1	0.125	2	0.25
11	1	0.125	2	0.25	0	0.0
12	1	0.125	2	0.25	2	0.25
13	1	0.125	1	0.125	0	0.0
14	0	0.0	2	0.25	3	0.375
15	3	0.375	4	0.5	4	0.5
16	4	0.5	5	0.625	4	0.5
17	2	0.25	3	0.375	1	0.125
18	3	0.375	2	0.25	4	0.5
19	4	0.5	1	0.125	5	0.625
20	5	0.625	4	0.5	6	0.75
21	8	1.0	8	1.0	8	1.0
22	8	1.0	7	0.875	7	0.875
23	4	0.5	8	1.0	4	0.5
24	1	0.125	1	0.125	3	0.375
25	6	0.75	6	0.75	8	1.0
26	4	0.5	3	0.375	4	0.5
27	8	1.0	7	0.875	8	1.0

28	4	0.5	3	0.375	5	0.625
29	8	1.0	8	1.0	8	1.0
30	3	0.375	2	0.25	3	0.375
Averages	0.44583	0.44583	0.48333	0.48333	0.5125	0.5125

References

[1] Chi-Man Pun, "Rotation-invariant texture feature for image retrieval", Computer Vision and Image Understanding,Vol. 89, pp. 24-43, Elsevier, 2003.

[2] Talagala N., Asumi S., Patterson D., Futernick B. and Hart D., "The Art of Massive Storage", Computer, Information Retrieval, Volume 33, Number 11, pp. 22-28, Computer Society IEEE, November 2000.

[3] Lew S. M., "Next-Generation Web Searches for Visual Content", Computer, Information Retrieval, Volume 33, Number 11, pp. 46-53, Computer Society IEEE, November 2000.

[4] ByoungChul Ko, Haesung Lee and Hyeran Byun, "Flexible subblocks for visual information retrieval", Electronics Letters, Vol. 36, No. 1, pp 254-25, IEE 2000.

[5] Del Bimbo Alberto, Visual Information Retrieval, Morgan Kaufmann Publishers, 1999.

[6] J. Puzicha, Th. Hofmann and J. M. Buhmann, "Histogram clustering for unsupervised segmentation an image retrieval", Pattern Recognition Letters Vol. 20, pp 899-909, Elsevier 1999.

[7] Yu Zhong and Anil K. Jain, "Object localization using color, texture and shape", Pattern Recognition Vol. 33, pp 671-684, Elsevier 2000.

[8] Leung Clement, Visual Information Systems, Springer, 1997.

[9] Chun S. Lu, Pau C. Chung and Chih F. Chen, "Unsupervised texture segmentation via wavelet transform", Pattern Recognition, Vol. 30, No. 5, pp. 729-742, Elsevier 1997

[10] Jain A. K. and Vailaya A., "Image retrieval using color and shape", Pattern Recognition, Vol. 29, No. 8, pp 1233-1244, Pergamon Elsevier, 1996.

[11] D.G. Sampson, E.A.B. da Silva and M. Ghanbari, "Wavelet transform image coding using lattice vector quantisation", Electronics Letters, Vol. 30, No. 18, pp. 1477-1478, IEE 1994.

[12]http://www.cs.columbia.edu/CAVE/research/softlib/coil-100.html

[13]http.//www.aircraft-images.co.uk

Detección de rasgos y control de calidad en la manzana

Verónica Barzola L.[1], Moisés Bonilla E.[1],. J. Humberto Sossa A.[2], Ricardo Barron F.[2]

[1]Centro de Investigación y de Estudios Avanzados –IPN, Departamento de Control Automático, vbarzola@aol.com
[2]Centro de Investigación en Computación –IPN, hsossa@cic.ipn.mx , rbarron@cic.ipn.mx

Abstract. El control de calidad en frutos es un tema desafiante, cada fruto tiene diferentes cualidades y grados de calidad, los grados de calidad son dados por las normas nacionales e internacionales en donde se especifica que debe tener y no el fruto. En este trabajo se estudiaron solo 2 rasgos de calidad de la manzana golden delicius amarilla mexicana los cuales son: golpes y pedúnculo, estos rasgos son una parte importante para determinar el grado de calidad de la manzana, primeramente se realiza el procesamiento de la imagen digital para obtener los golpes y el pedúnculo si es que hay en la manzana, segundo se utilizan las técnicas de los sistemas difusos para realizar una clasificación de calidad de este fruto conforme a la norma mexicana NMX-FF-061-SCFI-2003.

Keywords: control de calida, métodos no destructivos, sistemas de lógica difusa

1. Introducción

En horticultura, muchos procesos requieren de toma de decisiones basadas en información visual. Esto es particularmente cierto en tareas de corte de fruto, en la cosecha y en el *control de calidad.*

El avance de las aplicaciones practicas del análisis de imágenes en la agricultura y la horticultura ha dependido crucialmente en 2 puntos clave de la revolución computacional: decrecimiento del costo de los sistemas e incremento en la velocidad de procesamiento y de capacidad de memoria.

Hay varios ejemplos prácticos en donde se utilizan las técnicas del procesamiento digital de imágenes en agricultura y horticultura, unos ejemplos son: el proceso de selección basado en tareas repetitivas, obtención de volumen realizando interpretaciones en imágenes de 3D. En los últimos años, el procesamiento digital de imágenes en agronomía se ha incrementado considerablemente, dentro de la literatura se puede encontrar [1] donde, mediante técnicas de procesamiento de imágenes determina la relación entre cobertura de tierra relativa por el dosel de cosecha y el peso seco, y con ello obtener el crecimiento de la lechuga, otro ejemplo es: [2] en

donde por medio de la técnicas del procesamiento digital de imágenes se obtiene el área de hoja de la espinaca, [3] determina la cantidad de agua consumida por una hoja de la planta, también es posible detectar enfermedades, mejorar el control de calidad en alimentos frescos destinados al consumo humano.

Uno de los objetivos primordiales es que mediante *métodos no destructivos* se realice la medición de crecimiento y el grado de calidad tanto de frutos como vegetales, para que estos no se vean afectados fisicamente.

Una de las técnicas más favorables es el análisis de imágenes digitales por computadora(Visión Artificial). Las técnicas de este procesamiento digital aplicadas a ciertas áreas de la agronomía permiten el reconocimiento de enfermedades, maduración de los frutos, calidad de los frutos, mejoramiento de la siembra.

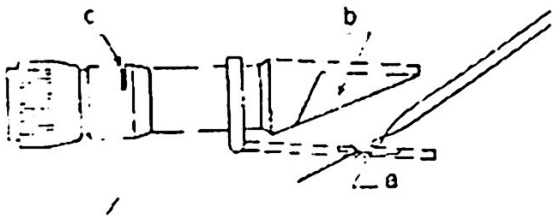
El control de calidad de los frutos ha resultado ser uno de los puntos más importantes en la agronomía, dado a que es necesario la selección del fruto para ser llevado ya sea a consumo humano o para algún proceso. Un ejemplo es:

La manzana [4] (Malus domenstica Borkh), cual es uno de los frutos más consumidos a nivel mundial, la manzana golden amarilla es la más consumida en México.

La calidad de la manzana esta dada por varias características fisicas y químicas dentro de la fisicas se encuentran: firmeza, golpes, pedúnculo, rozaduras, roseteado (área de color café que se forma alrededor del pedúnculo), forma, tamaño, grietas. Dentro de las características químicas esta el color, olor, y contenido de sólido soluble(azúcar).

La gran mayoría de las técnicas usadas para determinar la calidad de los frutos son métodos destructivos un ejemplo es:

Para determinar el contenido de sólido soluble (contenidos de azúcar) se utiliza un refractómetro para tomar una muestra del jugo de manzana, para ello es necesario cortar una rebanada de manzana y obtener su jugo y así medir el contenido de sólido soluble en grados Brix, Figura 1.



a) base para la muestra
b) prisma
c) tornillo de ajuste de la escala
d) muestra de jugo de la manzana

Fig. 1. Refractario

Otra método destructivo es para obtener la firmeza o resistencia de la manzana, para ello se usa un penetró metro, Figura 2, el cual es introducido con fuerza a la manzana y así medir resistencia a la cual se opone.

La cantidad de golpes y la existencia de un pedúnculo son evaluados por personas a la hora de ser seleccionados. Mucho de los otros rasgos que determinan la calidad de la manzana son tomados por personas, por lo que se lleva un largo tiempo de decisión en calidad del los frutos.

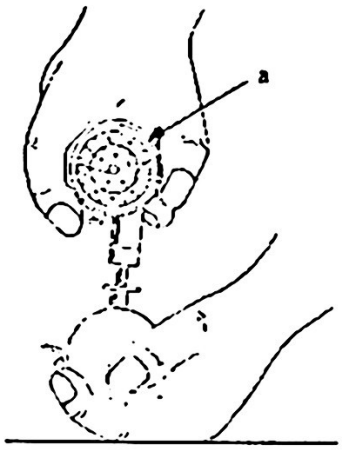


Fig. 2. Penetrómetro

La manzana en estado fresco se clasifica en 3 grados de calidad [5]: México extra(Mex-extra), México 1(Mex-1), México 2(Mex-2). Mex-extra es la manzana de mejor calidad en esta no se permite ningún grado de defecto o anomalía. Mex-1 y Mex-2 son de menor calidad cada uno de ellos permite un cierto grado de imperfecciones en cada una de las características que se describen en la Norma Mexicana de calidad de la manzana (NMX-FF-061-SCFI-2003), Figura (3).

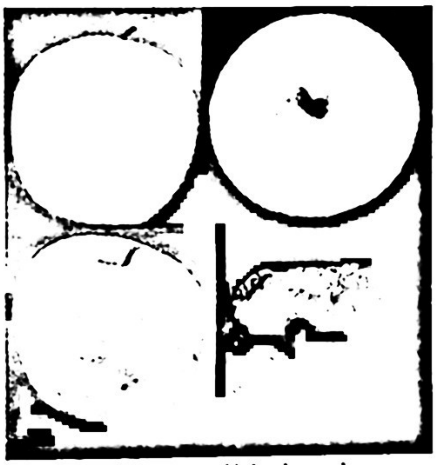


Fig. 3. Grados de calidad en la manzana

2. Material y Modelo

Para realizar el análisis de calidad en la manzana golden amarilla, se usaron 20 manzanas de Chihuahua, México, una cámara digital de 1024 x 768 píxeles, resolución horizontal de 300, resolución vertical de 300, profundidad de 24 y representación en color verdadero RGB. La cámara se fijo a una altura de 50cm arriba de la manzana a un ángulo de 55° a la normal. El fondo fue negro, blanco y rosa. La iluminación fue luz azul de 500W de forma continua, con una lámpara semiesférica para restar sombras. La operación de la manzana fue controlada a través de una computadora personal, Figura 4. Las imágenes obtenidas fueron procesadas con el toolbox Image Processing de Matlab y la toma de decisión del grado de calidad con el toolbox Fuzzy Logic de Matlab.

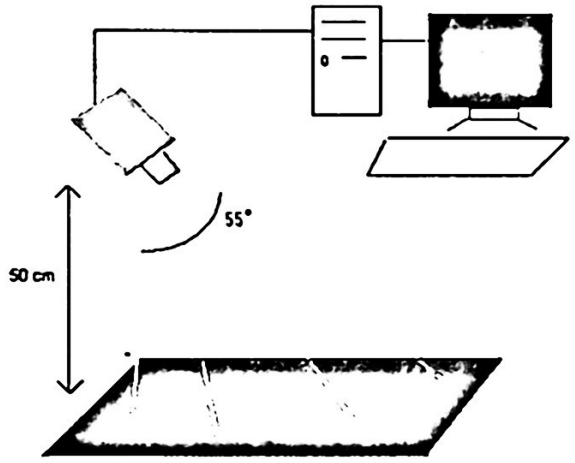


Fig. 4. Sistema de Visión

3. Sistemas de Lógica Difusa

La lógica difusa [6] empieza con el concepto de un conjunto difuso, donde un conjunto difuso es un conjunto que no es claramente definido acotado, el cual puede contener elementos con un solo grado parcial de membresía. Sea F un conjunto definido sobre *U* (universo de discurso), tal que un conjunto difuso es descrito por la función de membresía (μf), que mapea *U* al intervalo real [0,1]. Por cada elemento u ε *U*, la función μf(u) produce el grado de membresía de u al conjunto difuso F ($0 \leq \mu f(u) \leq 1$), donde el 1 denota la membresía completa y cero denota la no membresía.

Una función de membresía, (μf), es una curva que define como cada punto en el espacio entrada es mapeado a un valor membresía (o grado de membresía entre el valor 0 y 1). Una función de membresía entrada a un valor de membresía apropiado. Las operaciones básicas de conjuntos difusos son: intersección (∧), unión (∨) y negación (not).

Las reglas difusas permiten expresar un procesamiento estratégico en la forma en que toma decisión el ser humano. Las reglas ***If-Then*** son usadas para formular la declaración condicional que forma la lógica difusa. Ejemplo: ***If*** *x es A* ***then*** *y es B*, donde A y *B* son valores lingüísticos definidos por conjuntos difusos sobre los rasgos (universo de discurso) *'x'* y *'y'* respectivamente. El ***If*** parte de la regla *"x es A"* llamado antecedente o la causa, mientras que la parte de la regla *"y es B"* es llamado consecuente o efecto. El conjunto difuso *A* esta representado como un número entre 0 y 1, así el antecedente es una interpretación que regresa un número simple entre 0 y 1 El conjunto difuso *B* asigna todo el conjunto *B* a la variable de salida *'y'*. La salida de cada regla es un conjunto difuso. Básicamente un sistema difuso es una colección de reglas difusas que son numéricamente procesadas por medio de un mecanismo de inferencia.

La inferencia difusa es el proceso de formulación de un mapeo de entrada y salida usando lógica difusa de tal forma que se describa completamente el proceso utilizando las funciones membresía, operadores lógicos y reglas ***If-then***. Hay dos tipos de sistemas de inferencia difusa tipo [7] Mamdani y tipo [8] Sugeno. Cada uno determina de diferente forma la salida del sistema.

4. Análisis del golpe, pedúnculo y clasificación

Para la determinación de los rasgos en la manzana y la clasificación de está en los grados de calidad, Mex-extra, Mex-1 y Mex-2, se procesaron 20 imágenes de manzanas amarillas utilizando algunas técnicas prácticas del procesamiento digital de imágenes, los datos obtenidos fueron introducidos en un sistema difuso de tipo Mamdani, utilizando 18 reglas para su clasificación. Las imágenes se analizaron con el programa Matlab, Toolbox Image Processing , primeramente se convirtieron las imágenes en color a imágenes en niveles de gris, para determinar el rango de valores de gris que describen a los golpes y al pedúnculo, después se paso cada imagen por un filtro de mediana para eliminar el poco ruido que tienen las imágenes. Ya convertidas las imágenes en valores de girs y eliminado el ruido, para determinar cual es el rango con que se pueda distinguir tanto los golpes como el pedúnculo se utilizo el mouse, el cual al estar en una posición *(xi,yi)* de la imagen muestra el nivel de gris de esta posición, y así obtener un rango que encierra todo un rasgo de la manzana como es el golpe y el pedúnculo, este procedimiento se hizo para cada una de las 20 manzanas, de esta manera se obtuvo el rango en nivel de gris en que se encuentra tanto los golpes como el pedúnculo. Con estos valores se hace un corrimiento dentro de la imagen y si *(xi,yi)* = *R*, (*R* = rango de niveles de gris que encierran cada rasgo de la manzana a estudiar), entonces *(xi,yi)* =*1* sino *(xi,yi)* = *0*, con lo que se obtiene una imagen de 2 valores *0 (blanco)* y *1 (negro)*, el uno representa el objeto a estudiar y el cero lo que no pertenece al análisis o no es de gran interés. Para el primer estudio de los valores de gris de golpes, se posiciona el mouse en el área del golpe para obtener el valor del píxel en ese punto, se consideraron alrededor de 10 valores de píxel de tal forma que se rodeara bien el golpe, el rango que se encontró en promedio entre todas las manzanas que si tenían golpe fue [80-140], con este valor se realizó el corrimiento en la imagen y si *(xi,yi)* = *(80-140)*, se marca con valor de 1, si no entonces *(xi,yi)* = *0*, de tal forma que se resalta el golpe en la imagen, Figura 5. Los resultados fueron

satisfactorios para 13 manzanas de las 20 imágenes. De esta misma forma se estudia la existencia del pedúnculo, rasgo importante para determinar la calidad de la manzana según las normas mexicanas e internacionales, el rango de niveles de gris para el pedúnculo en promedio es de *35 a 100*. Si *(xi,yi) = (35-100)* se le asigna el valor de 1 (negro), sino si *(xi,yi) ≠ (35-100)* se asigna el valor de 0 (blanco), de esta forma se resalta el pedúnculo y lo demás no importa, Figura 6, los resultados fueron satisfactorios para 15 manzanas de las 20 imágenes.

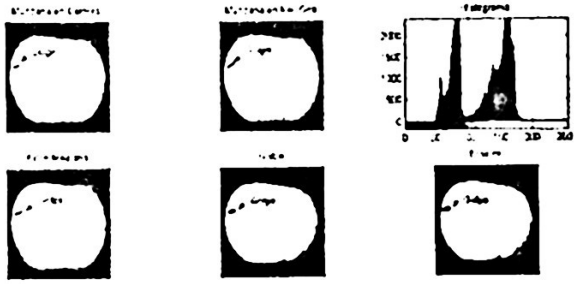


Fig. 5. Golpe

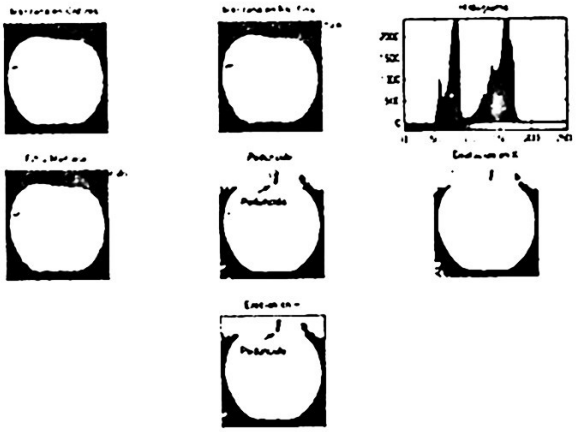


Fig. 6. Pedúnculo

Al realizar la separación de golpe y pedúnculo debido a que los rangos de separación se traslapan, esto es, el rango promedio del golpe es de *80-140* y el rango del pedúnculo es de *35 -100* en algunas imágenes salieron a relucir estos 2 rasgos, Figura 7. Es notorio que hay un traslape de niveles de gris para determinar la existencia tanto de golpes como de pedúnculo. La intersección de valores de gris entre los rasgos estudiados no ayudan a lograr una separación entre golpes como de pedúnculo, debido a que cuando se quiera detectar de una manera automática el número de golpes que tiene una manzana puede llegar a confundirse con el pedúnculo y ser considerado como golpe.

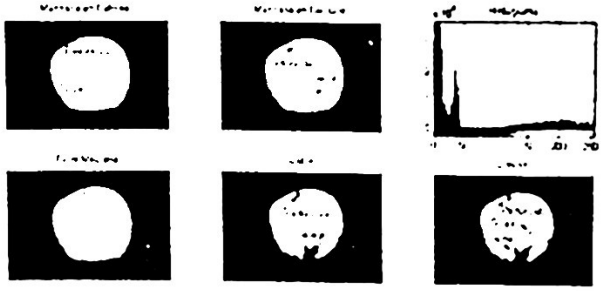


Fig. 7. Golpe y pedúnculo en una misma imagen

La clasificación del grado de calidad de las 20 manzanas analizadas fue por medio de sistemas de lógica difusa, en donde se usaron funciones de membresía tipo triangular estas funciones variaron dependiendo el grado de importancia o pertenencia que tiene cada rasgo como es golpe, pedúnculo y color. Se introdujeron 18 reglas ***If-Then***, para determinar el grado de calidad que son México-extra, México-1 y México-2 según la norma mexicana. Dentro de las 18 reglas ***If-Then*** se considera la cantidad de golpes en una manzana, si hay ó no pedúnculo y el color de la manzana, como operador lógico se uso el "and", es decir el valor mínimo entre los conjuntos, Tabla 1.

La clasificación se llevo a cabo por el método de Mamdani [6] en el toolbox Fuzzy logic de Matlab, haciendo uso de la herramienta del ruleedit el cual permite usar método Mamdani o método Sugeno, así como introducir las membresías correspondientes a cada una de las entradas considerando los diferentes grados de golpe que son: *nada de golpes, pocos golpes y muchos golpes*, Figura 9, las membresías del pedúnculo dentro del cual se consideraron 2 partes en la que si hay pedúnculo y en la que no lo hay, y por último el color de la manzana amarilla dependiendo la cantidad de golpes (amarilla, semi-amarillo y semi-cafe). Las 18 reglas ***If-Then*** que se muestran a continuación, ayudan a determinar el grado de calidad de la manzana por ejemplo ***If*** no hay nada de golpes *y* si hay pedúnculo *y* el color es amarillo ***then*** la manzana es de calidad Mex-extra, esto indica,

And $\mu A \wedge B = \min(\mu A, \mu B)$		
A	B	min(A,B)
0	0	0
0	1	0
1	0	0
1	1	1

Tabla 1. AND

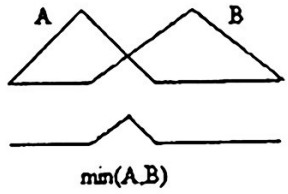


Fig. 8. El mínimo entre 2 funciones

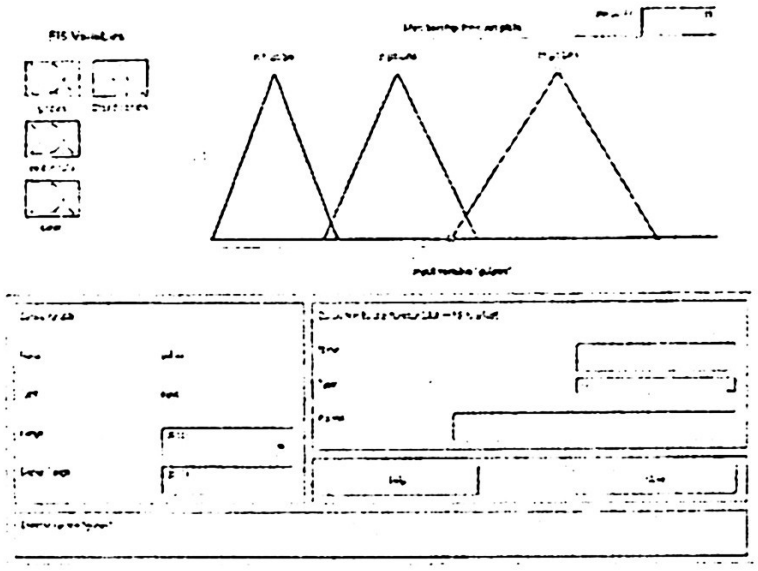


Fig. 9. Funciones de Membresía

que la manzana es de la mejor calidad ya que no tiene nada de golpes, tiene el pedúnculo que indica que no hay bacterias que lleguen a afectar al producto interno y además el color es amarillo.

If no hay nada de golpe ***y*** hay pedúnculo ***y*** el color es amarillo ***then*** la manzana es Mex-extra.

If no hay nada de golpe ***y*** hay pedúnculo ***y*** el color es semi-amarillo ***then*** la manzana es Mex-1.

If no hay nada de golpe ***y*** hay pedúnculo ***y*** el color es semi-cafe ***then*** la manzana es Mex-2.

If hay pocos golpes ***y*** hay pedúnculo ***y*** el color es amarillo ***then*** la manzana es Mex-1.

If hay pocos golpes ***y*** hay pedúnculo ***y*** el color es semi-amarillo ***then*** la manzana es Mex-1.

If hay pocos golpes *y* hay pedúnculo *y* el color es semi cafe *then* la manzana es Mex-2.

If hay muchos golpes *y* hay pedúnculo *y* el color es amarillo *then* la manzana es Mex-1.

If hay muchos golpes *y* hay pedúnculo *y* el color es semi amarillo *then* la manzana es Mex-1.

If hay muchos golpes *y* hay pedúnculo *y* el color es semi cafe *then* la manzana es Mex-2.

If no hay golpes *y* no hay pedúnculo *y* el color es amarillo *then* la manzana es Mex-1.

If no hay golpes *y* no hay pedúnculo *y* el color es semi amarillo *then* la manzana es Mex-2.

If no hay golpes *y* no hay pedúnculo *y* el color es semi cafe *then* la manzana es Mex-2.

If hay pocos golpes *y* no hay pedúnculo *y* el color es amarillo *then* la manzana es Mex-1.

If hay pocos golpes *y* no hay pedúnculo *y* el color es semi-amarillo *then* la manzana es Mex-2.

If hay pocos golpes *y* no hay pedúnculo *y* el color es semi-cafe *then* la manzana es Mex-2.

If hay muchos golpes *y* no hay pedúnculo *y* el color es amarillo *then* la manzana es Mex-2.

If hay muchos golpes *y* no hay pedúnculo *y* el color es semi-amarillo *then* la manzana es Mex-2.

If hay muchos golpes *y* no hay pedúnculo *y* el color es semi-cafe *then* la manzana es Mex-2.

La Figura 10 muestra los resultados obtenidos de las simulaciones, se tiene una columna de golpes en donde las membresías del lado izquierdo denotan que no hay golpes, las siguientes membresías en medio denotan que hay pocos golpes y por ultimo las membresías que están en el extremo derecho denotan que hay mucho

golpe, la siguiente columna es de las membresías de si hay ó no pedúnculo, en el cual las primeras membresías en el extremo izquierdo denotan que si hay pedúnculo y las siguientes membresía denotan la falta de pedúnculo, en la siguiente columna está el color, las primeras membresías al lado izquierdo denotan amarillo total en la manzana, la siguiente membresía denota que la manzana es semi-amarilla y la última membresía denota que la manzana es semi-café, la ultima columna denota la calidad del fruto, primeras membresías pertenecen la manzana Mex-extra, las segunda membresía pertenece a la manzana Mex-1 y por ultimo la manzana Mex-2.

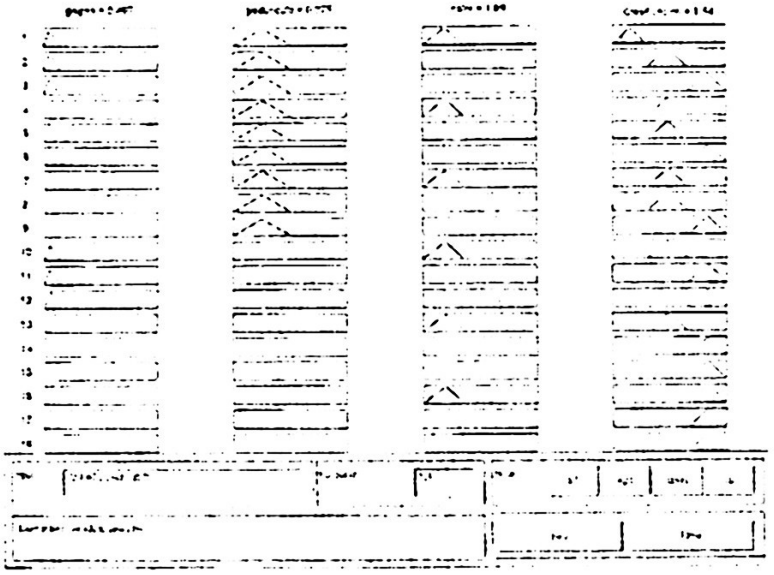


Fig. 10. Membresías de pertenencia

5. Resultados

La determinación de los golpes tomando como referencia el rango [80-140] de valor de gris permitió en 13 manzanas de las 20 analizadas encontrar el golpe más sobresaliente, en el resto de las otras manzanas no fueron detectados los golpes debido a su baja intensidad tanto de golpe como de valor de gris. En el estudio del pedúnculo con el rango de [35-100] de valor de gris de las 20 manzanas solo en 15 de ellas se pudo detectar la existencia del pedúnculo, debido a que en algunos casos el pedúnculo presenta un color cafe claro semejante al color de la manzana. En otros casos donde la manzana presenta varios golpes y todos ellos de diferente intensidad no se logro la separación de todos, solo de aquellos donde el golpe es bastante fuerte, un color café obscuro. Cuando el color del pedúnculo es semejante al color del golpe al realizar la separación de estos se obtienen los 2 rasgos en una misma imagen. Observando estos resultados de las 20 imágenes se crearon 18 reglas **if-then** para un sistema difuso tipo Mamdani, en donde se incremento una variable de entrada que es el color, con lo que permite poder realizar una clasificación de mayor aproximación al grado de calidad de la manzana dependiendo de los defectos que se tengan.

6. Conclusiones y Perspectivas

La determinación de los rasgos en las manzanas no es una tarea fácil de realizar debido a que cada manzana manifiesta diferentes formas de crecimiento aun siendo de la misma cosecha. Uno de los problemas principales es la detección de golpes debido a la intensidad del golpe y la intensidad del valor de gris. Otro problema es intersección de los rangos entre golpe y pedúnculo ya que no permiten hacer una separación adecuada para el estudio de cada uno de ellos. Una de las herramientas que más se acopla a la clasificación de la manzana según el número defectos son los sistemas de lógica difusa, debido a la complejidad de toma de decisión, para determinar el grado de calidad.

Para trabajo futuro es incrementar en un vector de rasgos más características que ayuden a determinar tanto los golpes como el pedúnculo de una manera automática sin que haya problemas de distinción entre ellos, y pasar estos datos a un sistema difuso que permita la clasificación de manera directa. Además de incrementar los rasgos que ayuden a dar una mejor aproximación al grado de calidad del fruto, dentro de estos rasgos pueden estar el porcentaje de color, el tamaño ó la forma.

Referencias

1. Van Henten, E.J., y Bontsema J.,"Non-Destructive crop measurements by image processing for crop growth control", Control Journal of Agricultural Engineering Research, 1995,
2. Barzola V, "Determinación del Área de hoja de la Espinaca por medio del procesamiento Digital de Imágenes", CINVESTAV, 2002,
3. H. Murase, Y. Nishiura, K. Mitani, "*Enviromental control strategies base don plant responses using intelligent machina vision technique*" , Elseivier, Computer and Electronics in Agricultura, 18(1997) pp.137-148,
4. Brown, A. G. Apples, pp 3-37. In: Advances in Fruit Breeding. J . Janick and J. Moore (eds). Purdue Indiana Press. West Lafayette, Indiana, USA, 1975,
5. Secretaría de Economía, Productos Agrícolas no industrializados para consumo humano – Fruta fresca - Manzana *(Malus pumila* Mill*)* - (*Malus domestica* Borkh) – especificaciones (cancela a la NMX-FF-061-1993-SCFI) , NMX-FF-061-SCFI-2003.
6. Vojislav Kecman, "*Learning and Soft Computing*" MIT Press,2001
7. Mamdani, E.H., "Application of Fuzzy algorithms for control of simple dynamic plant" Proceedings IEE 121(12):1585-1588,1974,
8. Sugeno M., "*An Introductory Survey of Fuzzy Control*", Information Sciences, 36, 59-83(1985)

Organización / Organization

Comité Científico

Dr. Sergio Suárez Guerra. (CIC-IPN), México e (ICIMAF), Cuba.
Dr. Klaus Michael Lindig Bos. (CIDETEC-IPN), México.
Dr. Mateo Valero. (UPC), España.
Dr. Antonio González. (UPC), España.
Dr. Ramón Beivide. Universidad de Cantabria, España.
Dra. María E. Hernández Díaz Huici. (UCLV), Cuba.
Dr. Oscar Camacho Nieto. (CIC-IPN), México.
Dr. Luis Villa. (IMP), México.
Dr. Alexander V. Vaindembaun. University of California, Irving, USA.
Dr. Jesús Savage. (UNAM), México.
Dr. Luis Pastor Sánchez. (CIC-IPN), México.
Dr. Manuel Guzmán Renteria. (CINVESTAV), México.
Dr. Oleksy Progrenbyak. (CIC-IPN), México.
Dr. Virgilio López Morales. UAEH (CITIS-UAEH), México.
Dr. Carlos Aguilar Ibáñez. (CIC-IPN), México.
Dr. José de Jesús Medel Juárez. (CIC-IPN), México.
Dr. Juan Humberto Sossa Azuela. (CIC-IPN), México.
Dr. Jesús Figueroa Nazuno. (CIC-IPN), México.
Dr. Rafael Martínez Guerra. (CINVESTAV), México.
Dr. Rubén Garrido Moctezuma. (CINVESTAV), México.
Dr. Juan Carlos Martínez García. (CINVESTAV), México.
Dr. Marco Iván Ramírez Sosa. (ITESM), México.
Dr. Marco de Alba. Intel Barcelona.
Profr. Joseph Lluis Larriba Pey. (UPC), España.

Índice de Autor

	Pág.
A	
Aburto G. R.	101
Acántara Ramírez R.	141, 157
Acho Leonardo	151, 243
Alcaraz Vega José Ángel	361
Alcorta García Efraín	111
Angeles Yreta A.	381
Ávila Luis	53
Ayala Lomeli Guadalupe	3
B	
Barba Eduardo	151
Barrera Jorge	53
Barron F. Ricardo	297, 391, 413
Barzola L. Verónica	413
Berrelleza Fernando	91
Bonilla E. Moisés	413
Bravo Parra Miguel	331
C	
Cabrera Jiménez M.	157
Calvillo T. A.	25, 37, 229
Cinco Galicia J.	141
Cruz Pérez Daniel	167
Cruz Pérez Jannet	63
D	
Duchén Sánchez Gonzalo Isaac	3, 13
E	
Escalona Delgado	287
Espejel Rivera A.	207
Espinosa Sosa Osvaldo	341
F	
Felipe R. E. M.	403
Ferrer C.	373
Figuero Nazuno J.	381
Flores García Ernesto	261
Flores Rueda Alberto	177, 195

	Pág
G	
García Ortega Víctor Hugo	341
Garrido Moctezuma Rubén	261
González G. A. C.	101, 403
Guerra Ricardo	243
Guevara López Pedro	167, 177, 195
Gutiérrez Alfonso	351
H	
Hernández Huici María E.	373
J	
Jaimes Ponce J	141, 157
L	
Landassuri Moreno V.	381
López Chau Asdrúbal	177
M	
Martínez Ham J.	157
Martínez Guerra Rafael	261
Martínez Hernández Haydée Patricia	217
Martínez Israel	53
Medel Juárez José de Jesús	167, 177, 195
Medina Meneses Víctor Manuel	85
Mendieta J. F. J.	25, 37
Muñoz Arteaga Jaime	63
Muñoz Hernández Germán A.	217
N	
Nakano Miyatake Mariko	331
Navarro Juan	53
Novoa Colín Juan Francisco	73

	Pág.
O	
Oropeza Rodríguez José L.	391
Ortega Herrera José A.	127
P	
Padilla Francisco	53
Peralta Francisco I.	13
Peredo Fuentes Evelyn	287
Pérez Meana Héctor	331
Porras Javier	53
R	
Ramírez Cruz José Federico	217
Ramírez Martínez Diego	273
Ramos Velasco Luis Enrique	207, 307
Ravelo Hernández J. C.	157
Razo Zapata S.	307
Rios F.	251
Rodríguez Ávila René	351
Rodríguez Gómez Gustavo	63
Rodríguez Ruiz R.	141
Rubio E.	251
Ruiz May E.	141
S	
Sánchez Fernández Luis Pastor	73, 91, 273, 361
Sandoval Reyes Sergio	319
Saucedo Flores Salvador	111, 127
Siller Alcalá I. I.	141, 157
Soberanes Carlos	53
Solís Estrella H.	381
Sossa A. J. H	101, 297, 403, 413
Suárez Guerra Sergio	391
T	
Tapia R. M.	25, 37
Téllez Leonardo	53
U	
Urbieta Parrazales Romeo	273
V	
Vázquez Roberto A.	297
Vázquez Rubén	13
Vicente Vivas Esaú	25, 37, 229

Impreso en los Talleres Gráficos
de la Dirección de Publicaciones del
Instituto Politécnico Nacional
Tresguerras 27, Centro Histórico, México, D.F.
Septiembre 2004. Edición: 500 ejemplares.

La publicación de este volumen de la Serie: Research on Computing Science, representa un serio esfuerzo en dar a conocer los desarrollos científico técnicos y aplicados que en México y en la comunidad internacional relacionada, se tienen en la Ciencia de la Computación y la Ingeniería de Cómputo.

Este Volumen 9, contiene la selección de los más recientes avances en Control, Instrumentación Virtual, Sistemas Digitales, Arquitectura de Computadoras, Robótica y Procesamiento de Señales, presentados en el **5to. Congreso Internacional en Control, Instrumentación Virtual y Sistemas Digitales CICINDI'2004**, con sede en el Centro de Investigación en Computación del Instituto Politécnico Nacional.

Especialistas nacionales e internacionales han contribuido de forma destacada en la elaboración y presentación de estos trabajos. Se destaca el avance en la especialidad de Control y Procesamiento de Señales, áreas de amplio impacto en las aplicaciones en la actualidad.

Con toda pasión le deseamos a los lectores de esta obra su valoración y crítica a la dinámica de la introducción de los resultados en esta temática de cambio continuo, la Computación.

Queremos agradecer a todos los trabajadores del Centro de Investigación en Computación su colaboración y apoyo para poder llevar acabo esta obra y a todos los miembros del Comité Científico y de Revisión que aportaron sus criterios y sugerencias.

ISBN:970-36-0180-4
ISSN: EN TRÁMITE