

## **Sistema multi-agente para el control de tráfico urbano para vehículos de prioridad**

Guillermo Fernández Estrella<sup>1</sup>, Joel Antonio Trejo Sánchez<sup>2</sup>,  
Mario Renán Moreno Sabido<sup>1</sup>

<sup>1</sup> Tecnológico Nacional de México,  
IT de Mérida,  
Departamento de Sistemas y Computación,  
México

<sup>2</sup> CONACYT-Centro de Investigación en Matemáticas,  
México

gfernandezestrella@gmail.com, joel.trejo@cimat.mx,  
mario.ms@merida.tecnm.mx

**Resumen.** El control de tráfico urbano se encarga de controlar el tráfico urbano con el fin de reducir los tiempos de espera de los vehículos involucrados en la red de tráfico. Particularmente, para los vehículos de prioridad, los tiempos de espera son de suma importancia, ya que una mayor espera en dichos vehículos podría tener consecuencias catastróficas. Los sistemas multi-agentes permiten representar el sistema de tráfico urbano y dar capacidades independientes a todas las entidades involucradas en la red de tráfico. En este trabajo se presenta una arquitectura basada en agentes que modela los componentes involucrados en un sistema de tráfico urbano, y se enfoca en reducir el tiempo de espera en los vehículos de prioridad. Experimentalmente, se muestra que la arquitectura propuesta mejora significativamente las estrategias predefinidas en la red de tráfico.

**Palabras clave:** Control de tráfico urbano, sistemas multi-agentes, vehículos de prioridad.

### **Multi-Agent Urban Traffic Control for Priority Vehicles**

**Abstract.** Urban traffic control deals with the efficient control of signal traffic lights to reduce the time delays of the involved vehicles in the traffic networks. Particularly, the delay in priority vehicles is very important, because a largest delay could have catastrophic consequences. Multi-Agent systems are a tool to represent an urban traffic system and provide with independent capacities to the involved entities in the urban traffic system. This paper shows a multi-agent system architecture to urban traffic control focus on priority vehicles. Experimentally, this paper shows that the architecture outperforms the predefined strategies in the urban traffic system.

**Keywords:** Urban traffic control, multi-agent systems, priority vehicles.

## 1. Introducción

Hoy en día, las ciudades van incrementando su tamaño, lo que genera mayor cantidad de tráfico, principalmente en las grandes ciudades. Está demostrado que la congestión de tráfico urbano produce estrés [1], contaminación [2], repercute directamente en la calidad de vida [3], entre otras cosas.

Con el fin de reducir la congestión de tráfico urbano, existen algunas estrategias que consisten en realizar una asignación predefinida a los semáforos, con el fin de minimizar la espera de los vehículos. Este tipo de estrategia se conoce como control pre-establecido (*pre-timed urban traffic control*); por otro lado, aquellas estrategias que buscan asignar dinámicamente los tiempos en los semáforos, se conocen como control dinámico de tráfico urbano [3].

Existen diversas estrategias que pretenden establecer de forma inteligente los tiempos de los semáforos en un sistema de control de tráfico urbano al tomar en consideración la información referente al ambiente en el que está inmerso el sistema de tráfico urbano. El control de tráfico urbano consiste en establecer ciclos, repartos y desfases en una vía o red, de manera tal que los vehículos puedan desplazarse a una cierta velocidad, procurando que las interrupciones generadas por luz roja sean mínimas [4].

Los sistemas multi-agentes están englobados en el campo de la inteligencia artificial, y su propósito es la construcción de sistemas complejos que interactúan entre sí para lograr un objetivo. Los agentes son autónomos y capaces de trabajar en conjunto para la resolución de problemas [5].

## 2. Trabajos relacionados

Se han propuesto diversas estrategias basadas en agentes para el control de tráfico urbano. En [6], Balaji y Srinivasan proponen dos arquitecturas de agentes. Dichas estrategias incluyen políticas de control dinámicas, en la que los agentes se deben adaptar a las condiciones de tráfico. Experimentalmente, los autores muestran que dichas arquitecturas tienen un mejor rendimiento en comparación con otras estrategias, incluyendo estrategias de control de tráfico urbano con tiempos predefinidos.

El dar prioridad a vehículos de prioridad no es nuevo. Existen algunos trabajos que se enfocan en minimizar el tiempo de espera en vehículos de prioridad. En [7] Tlig y Bhouri definen un sistema multi-agentes para dar prioridad a sistemas de transporte público. A diferencia del enfoque que se propone en este trabajo, ellos se enfocan principalmente a vehículos en los cuáles es posible conocer el tiempo aproximado en que pueden pasar por la intersección de tráfico. En el esquema propuesto en esta investigación, los vehículos de prioridad pueden pasar por la red de tráfico en cualquier momento. Recientemente, [8] propone un esquema que da prioridad en la asignación de los tiempos a los sistemas de servicios de transporte urbano. A diferencia de la propuesta que se presenta en este trabajo, en [8], los vehículos de prioridad tienen un calendario estimado del arribo de los vehículos de prioridad en las intersecciones de la red de tráfico urbano.

Este artículo se basa parcialmente en el trabajo realizado por [9 y 10]. Trejo-Sánchez y Fernández-Estrella presentan un controlador de sistemas multi-agentes en la cual se emplea una arquitectura multi-agente cuyo modelo de intersección esté representado por redes de Petri temporizadas, permitiendo analizar un modelo de acuerdo con sus propiedades, tal es el caso del análisis de invariantes, bloqueos, vivacidad, acotamiento, entre otras cosas. Algunos autores han utilizado sistemas de eventos discretos con redes de Petri para el control de tráfico urbano. Los sistemas de eventos discretos [11] son aquellos que están compuestos por elementos que manejan entidades discretas, es decir, numerables y diferenciables entre sí. Su funcionamiento está caracterizado por una sucesión finita o infinita de estados, delimitados por eventos.

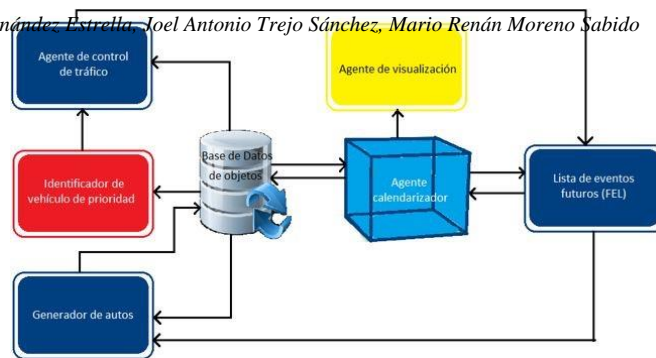
Di Febraro [12] et al. utilizan redes de Petri para modelar la red de tráfico que utiliza una red de Petri híbrida para el control de tráfico urbano. Para una revisión más detallada de cómo se han utilizado las redes de Petri en el control y simulación de tráfico urbano, el lector puede referirse a [13]. Más adelante, cuando se describe la arquitectura propuesta se presenta el módulo encargado del establecer las políticas de control de tráfico urbano. Dicho módulo está basado en sistemas de eventos discretos.

En la actualidad existen controladores de tráfico urbano; entre ellos destaca la plataforma SICE, la cual se especializa en la gestión inteligente de la movilidad, la gestión de la prioridad al transporte público mediante sus equipos RBG, la gestión del *Enforcement* aplicando tecnología tal como sistemas de foto rojo, control de velocidad en travesías, radares por tramo y giros prohibidos [14]. CROSS es una empresa que desarrolla sistemas inteligentes de control de tránsito. Los sistemas incluyen herramientas de despacho para controlar intersecciones, también herramientas para control en coordinación dependiente del tránsito y para control adaptable. Permite la preferencia activa de vehículos de transporte público y crear rutas para el paso de vehículos de servicios de emergencia [15].

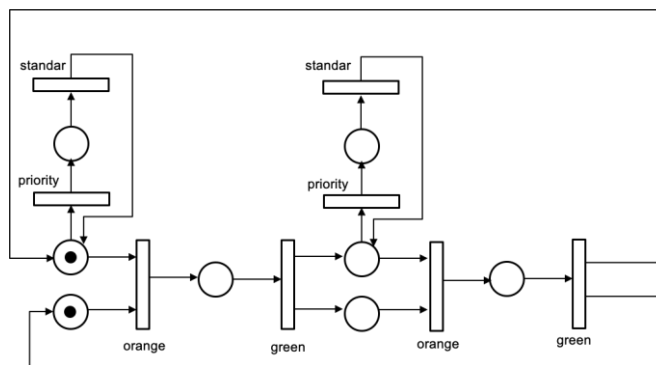
SciELO es un controlador de tráfico urbano basado en agentes inteligentes. En su trabajo se presenta el desarrollo de una metodología novedosa que permite incluir un modelo formal basado en agentes autónomos e inteligentes capaces de manipular las fases de los ciclos en una infraestructura de semáforos de acuerdo con las exigencias y limitaciones de la carretera. Este proceso mejora efectiva e inmediata de la calidad del servicio en una intersección, aumentando el rendimiento de la movilidad de los vehículos y mejorando la generación de emisiones, cuando los vehículos se paran en un semáforo rojo [16]. SUMO (Simulation of Urban Mobility) [17] es un simulador de código abierto que permite diseñar una red de tráfico, así como sus diferentes componentes. Dicha herramienta es utilizada por muchos investigadores para poder incluir sus propios algoritmos para el control de semáforos. SUMO ha demostrado permitir la integración de redes de Petri para el control de tráfico urbano [18]. Al ser una herramienta demasiado compleja, se decidió en este trabajo utilizar una herramienta propia que era suficiente para la arquitectura que se propone.

### **3. Arquitectura basada en agentes**

La composición de un sistema es muy importante, ya que permite identificar el propósito de su desarrollo y a su vez la estructura de cada uno de sus componentes con



**Fig. 1.** Arquitectura macroscópica del simulador con las relaciones que existen entre módulos y el orden de las ejecuciones.



**Fig. 2.** Red de Petri temporizada.

el fin de justificar por qué son indispensables en el desarrollo del proyecto. La Figura 1 presenta la arquitectura propuesta.

Antes de detallar cada uno de los módulos que componen la arquitectura se presentará la red de Petri temporizada de tres fases empleada para el sistema de eventos discretos. En la figura 2 se describe la lógica del flujo de un vehículo en el simulador; en cada fase existe la posibilidad de que ingrese un vehículo de prioridad influyendo en el flujo de la calle y la coordinación de los semáforos.

A continuación, se describen cada uno de los módulos que componen la arquitectura de agentes.

### 3.1. Agente de control de tráfico

El agente de control de tráfico coordina las señales individuales de tráfico, por ejemplo, los semáforos. A su vez, un sistema de control tiene como componentes básicos: intersecciones reguladas por semáforos, una red de comunicaciones que permite el envío o intercambio de información, y un computador central que gestiona el funcionamiento de todo el sistema.

Como parte principal del simulador y dado que este módulo lleva por nombre “agente”, éste debe conocer las características que lo rodean.

El agente de control necesita saber el estado en el que se encuentra cada semáforo en cada iteración de tiempo, así como el sentido de la calle, si va de norte a sur, este a oeste, entre otras cosas. La facultad del agente de control de tráfico no solamente repercute en los vehículos que transitan durante la simulación, también determina el permiso del flujo en toda una fila de autos cuando los semáforos cambian y cuando un vehículo de prioridad aparece en la calle.

El agente opera con dos políticas: la política fija toma como referencia valores de tiempo definidos para el cambio de los semáforos en distintas horas del día; en cambio la política adaptativa permite que el agente de control determine una acción, teniendo como valores de entrada el ciclo de cambio de los semáforos, la densidad de vehículos en la calle, el flujo de vehículos en una intersección y la hora del día. Sin embargo, se diseñó para que sea posible integrar nuevas políticas en el futuro.

### **3.2. Generador de autos**

Este módulo conlleva la responsabilidad de la creación e identificación de los vehículos generados; es de suma importancia mencionar que el módulo encargado de generar los autos no puede tener ningún fallo, de lo contrario afectaría a todo el simulador generando un error que termine con la simulación.

También se encarga de generar tanto vehículos de prioridad, como vehículos estándar; éste crea un objeto denominado “*car*” el cual es identificado por un número irrepetible y añadido a la lista de eventos tomando en cuenta el segundo de creación; los vehículos de prioridad se generarán con un identificador para diferenciar de un vehículo estándar. Aparte de solamente generar los autos, este módulo crea de forma aleatoria un vehículo cada segundo, controlando así que la simulación sea de lo más natural a una intersección de vía pública. El movimiento de los vehículos ocurre de forma autónoma ya que el agente generador aplica variables de estado y tiempo para que el vehículo elija si va a moverse o no en cada iteración de simulación.

### **3.3. Base de datos (objetos)**

Es el módulo donde el conjunto de toda la información perteneciente a los objetos, estados y vectores estarán almacenados en la “base de datos de objetos”. Aparte de las operaciones básicas que realizará la base de datos como almacenar información, también actualizará los procesos reorganizando los datos generados por el simulador, convirtiendo la base de datos en un proceso dinámico.

### **3.4. Agente calendarizador**

El agente calendarizador es el principal intermediario en el simulador; éste se encarga de obtener información de los vehículos en movimiento como pueden ser su posición que por el momento es asignada por espacios dentro de un carril, su orientación, el estado en el que se encuentra el vehículo, sea en movimiento o detenido. Por otra parte, el agente obtiene la información de la base de datos (FEL) que trabaja en conjunto para poder lograr una actualización real y constante de la simulación; de igual manera, envía la información generada al agente de visualización para su graficación.

La conexión que existe entre los agentes con los módulos de la arquitectura se da por medio de la programación en hilos de Python donde un hilo llamado H se está ejecutando dentro de un método sincronizado y necesita acceso a un recurso llamado R; si éste no se encuentra disponible, H renuncia temporalmente al control del objeto, permitiendo la ejecución de otro hilo hasta que R se encuentre disponible.

### **3.5. Lista de eventos futuros**

Un evento significa un cambio del estado del simulador, por lo tanto, el simulador debe actualizar su contenido. Cada evento tiene asignado un tiempo de ocurrencia como son los semáforos y los vehículos, de tal manera que cuando suceda esto, los eventos generados son inscritos en la lista de eventos futuros (FEL) [9]. El simulador se complementa de la FEL cuando se integra un sistema de control de tráfico urbano.

La FEL almacena la información perteneciente a los objetos generados por el simulador procesándolos e interpretándolos para su lectura por el usuario. Los datos son escorados por el tiempo de creación del evento seguido del tipo de evento y los atributos de cada uno de ellos. El usuario podrá visualizar la FEL en la interfaz de código, pero el simulador también exportará todos los datos almacenados en una tabla ordenada.

### **3.6. Agente de visualización**

Las ventajas de la simulación permiten al usuario detectar el comportamiento de los objetos puestos a prueba; en el simulador se implementó una interfaz gráfica para la proyección de los cambios y movimientos de los vehículos permitiendo una experiencia agradable para el usuario.

El agente de visualización se compone de las calles, los vehículos y los semáforos; éstos son pintados mediante la herramienta Tkinter de Python. Tkinter tiene como ventaja la actualización de la ventana permitiendo ver el movimiento a tiempo real de los vehículos.

### **3.7. Implementación del sistema multi-agente**

Mediante un complejo diseño del simulador microscópico se determinó que es necesario de diversos módulos que sean capaces de tomar decisiones por sí mismo e interactuar entre ellos mismos.

El simulador tiene un componente de tiempo que funciona como un reloj para medir en qué punto se origina cada evento; esto permite a los agentes sincronizarse con base al ciclo de simulación. Todos los vehículos son detectados por el agente de control una vez son sincronizados con la base de datos; esta lectura de datos se da de manera cíclica por cada segundo de iteración de la simulación. En la arquitectura presentada se muestra un módulo el cual se encarga de detectar los vehículos de prioridad; dado que los vehículos son generados por el simulador, la detección de un vehículo de prioridad es mediante un identificador para determinar si el vehículo es de prioridad o es estándar.

El agente de control tiene como prioridad la sincronización de los datos de la lista de eventos futuros y la lectura de los vehículos de prioridad para la elección del nuevo ciclo. Se utilizaron objetos para implementar los agentes.

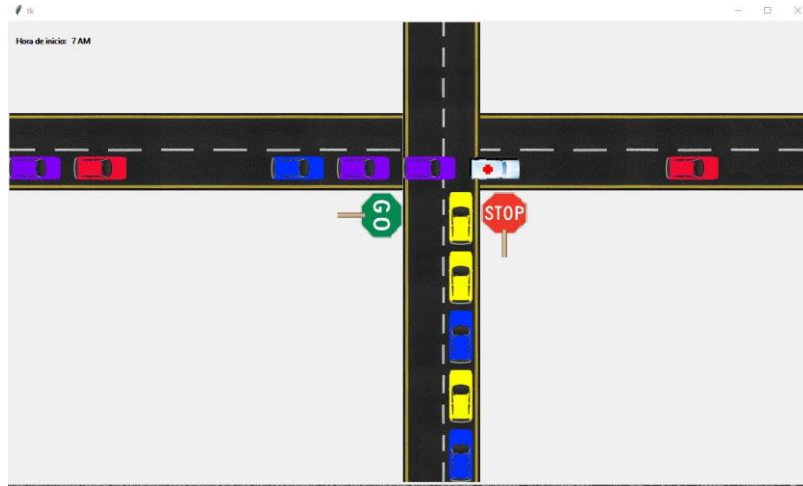


Fig. 3. Interfaz gráfica del simulador.

El agente de control contiene métricas de flujo, densidad y tiempo; también se definieron límites de la calle y conoce el momento de entrada de la calle, así como su salida. Los vehículos tienen variables de tiempo, posición, orientación y el identificador de vehículo de prioridad o estándar.

Los agentes se comunican a través de llamadas directas entre dichos objetos. Actualmente, el sistema que se presenta es centralizado, pero puede extenderse a un ambiente distribuido a través de llamadas a objetos remotos. La comunicación entre estos objetos en la simulación es por medio del “*threading*” que está incluido en una interfaz de alto nivel orientada a objetos para trabajar con concurrencia desde Python, donde los objetos “*Thread*” se ejecutan al mismo tiempo dentro del mismo proceso compartiendo memoria.

En la Figura 3 se presenta la interfaz gráfica del simulador donde se representan los dos tipos de vehículos, la orientación y la hora de la simulación actual. Es importante mencionar que antes de la simulación es posible la modificar la hora inicial y la duración de la simulación; si se requiere el cambio de políticas o los parámetros de generación del vehículo, es necesario modificar el código.

### 3.8. Experimentación

Para el análisis comparativo de tiempos se emplearon políticas de control que se basan en la hora del día, el número de autos, la duración del semáforo, entre otras cosas.

Con el fin de probar la arquitectura, se realizan varios experimentos utilizando una política adaptativa, y una política fija, sin embargo, por cuestiones de espacio solamente se mencionan los resultados experimentales más relevantes.

La política de control adaptativa da prioridad a los vehículos prioritarios. En esta política, cuando aparecía un vehículo de prioridad, se ajustaban los semáforos y se modificaban los eventos futuros. Para más detalle de esta política referirse a [10].

La política fija funciona con un horario preestablecido que define las fases durante todo el tiempo de la simulación.

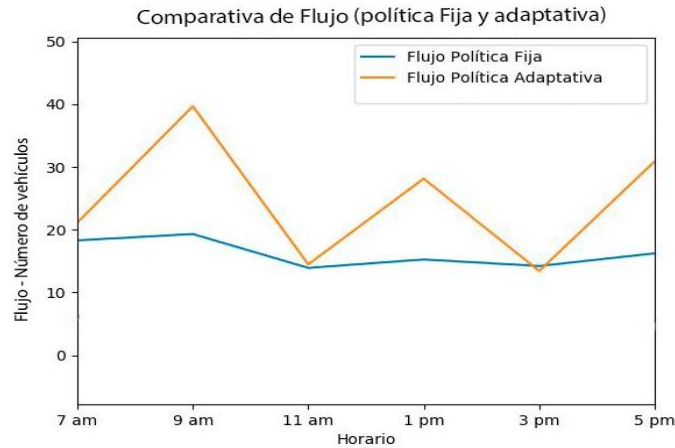


Fig. 4. Comparación del flujo - Política fija y adaptativa.

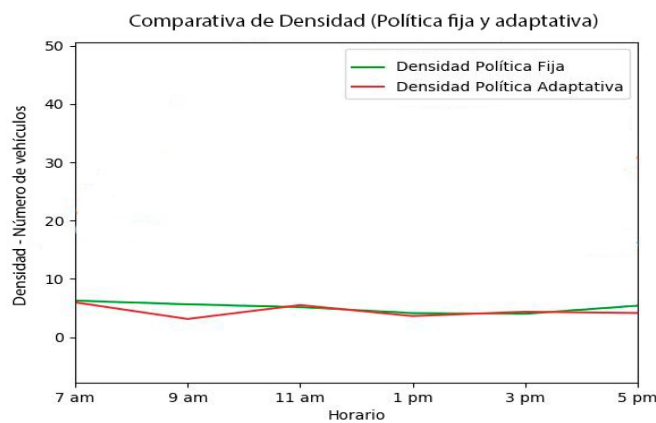


Fig. 5. Comparación de densidad – Política fija y adaptativa.

**Comparación de flujo y densidad.** Debido a que la política fija tiene definido los parámetros de estado de un semáforo, se vuelve impredecible el comportamiento de variables como el flujo; si en un momento de la simulación existe una congestión vehicular, el flujo mostraría el alto paso de vehículos, sin embargo, no lo haría en un momento óptimo para mantener una estabilidad del flujo en la calle. De igual forma, la densidad se estancaría hasta que los semáforos permitan el paso de los vehículos.

En la Figura 4 se presentan los resultados de la comparación del flujo de la política fija y adaptativa. Al realizar la evaluación del sistema se pudo notar que en ciertas situaciones el sistema de control de tráfico no reduce los tiempos de espera cuando se utilizan políticas fijas tradicionales, por lo tanto, se puede decir que no son convenientes para minimizar la congestión vehicular.

Además, este sistema permite la implementación de políticas de control con el fin de estudiar nuevos métodos para la reducción del tráfico vehicular demostrando las ventajas de este sistema sobre los tradicionales.



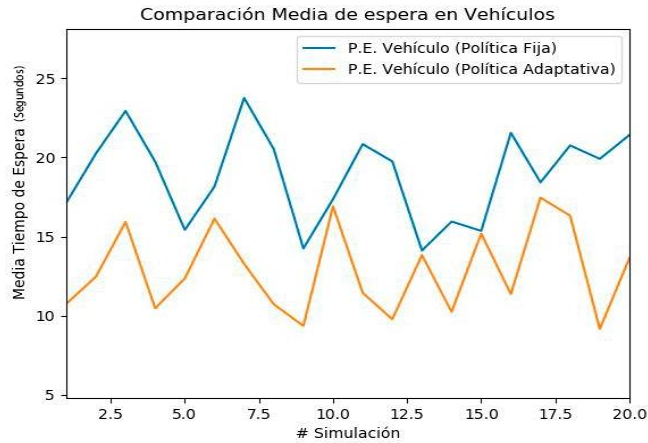


Fig. 6. Comparación media de espera en vehículos.

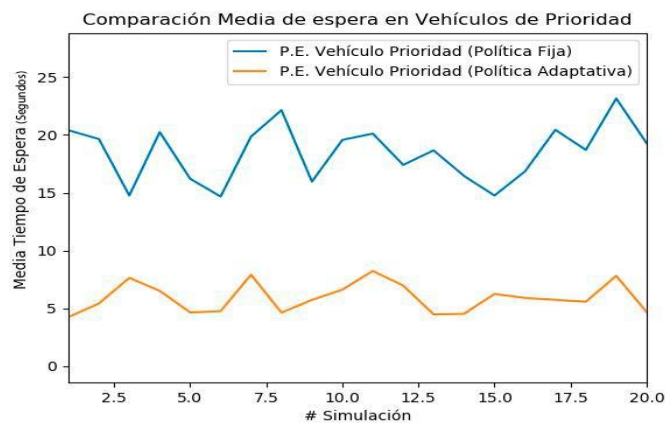


Fig. 7. Comparación media de espera en vehículos de prioridad.

En la Figura 5 se presentan los resultados comparativos de la densidad entre ambas políticas.

**Comparación en tiempos de espera.** En la gráfica media de espera en vehículos de la política fija se puede observar que el tiempo de espera de los vehículos puede ser muy variante llegando a los 24 segundos, lo cual implica que los vehículos están esperando los ciclos del semáforo completo, y, por lo tanto, puede generarse una congestión vehicular si aumenta la densidad en un sentido de la calle, a comparación de la política adaptativa que el rango de tiempo de espera va desde los 10 segundos, hasta los 16 en promedio (ver Figura 6).

Por otra parte, la media de espera en vehículos de prioridad en la política adaptativa presenta una gran diferencia de tiempo con la política fija, teniendo un rango de espera de entre 4 y 8 segundos en promedio, disminuyendo de 10 a 12 segundos de tiempo en

promedio, solamente con la implementación de la política adaptativa; por lo tanto, se puede decir que la existe una reducción de tiempo de espera notable en los vehículos de prioridad (ver Figura 7).

#### 4. Conclusiones y trabajo a futuro

Con el diseño del sistema de semáforo inteligente se buscó desarrollar un sistema que requiera de poca interacción humana para tomar decisiones y que permita controlar el flujo de tráfico vehicular de una intersección. El sistema permite la detección de vehículos y conteo de éstos; dicha información es almacenada en una memoria, para que posteriormente sea analizada por el agente de control, ya que esto le permitirá tomar decisiones sobre el control del tráfico, y a su vez interactuar con el resto de los módulos para cumplir el ciclo de la simulación.

La implementación de la política de control permite al agente de control elegir la opción óptima que permita el flujo vehicular, dando prioridad a los vehículos designados como importantes en la simulación.

Uno de los elementos primordiales dentro de la gestión del tráfico urbano es el funcionamiento de los semáforos que regulan una intersección vial. El congestionamiento del tráfico representa en la actualidad un gran reto a resolver debido al número de usuarios cada vez mayor que necesitan transportarse para realizar sus actividades económicas, sociales, culturales y de cualquier índole.

El resultado de esta investigación fue la creación de un sistema de control de tráfico urbano, permitiendo analizar en comportamiento del tráfico vehicular durante la simulación gracias a la capacidad de generación, detección y dibujo de vehículos, así como la interacción de los semáforos con el entorno cumpliendo el objetivo principal del presente trabajo, y demostrando en los resultados experimentales la problemática que existe con respecto al tráfico vehicular, y aportando un simulador de eventos discretos con interfaz gráfica que demuestra la reducción de tiempos de espera en vehículos de prioridad mediante políticas de control, con el fin de la experimentación para la contribución a la investigación y mejora del flujo vial.

Se espera que este proyecto de paso a próximas investigaciones relacionadas al tráfico vehicular, dejando como trabajo a futuro en el área la implementación de diferentes políticas de control, el anexo de nuevos módulos al simulador, el incremento en el número de calles simuladas simultáneamente, y la posibilidad de integrar la detección de un vehículo de prioridad a este simulador.

**Agradecimientos.** Este trabajo fue realizado gracias al apoyo brindado a través de los proyectos FORDECYT 296737 "Consortio en Inteligencia Artificial" y TecNM 6176.19-P "Análisis de sistemas de flujo urbano para el diseño de ciudades inteligentes".

#### Referencias

1. Evans, G.W., Carrère, S.: Traffic congestion, perceived control, and psychophysiological stress among urban bus drivers. *Journal of Applied Psychology*, 76(5), pp. 658 (1991)

2. Rao, A.M., Rao, K.R.: Measuring urban traffic congestion-a review. *International Journal for Traffic & Transport Engineering*, 2(4) (2012)
3. Roess, R.P., Prassas, E.S., McShane, W.R.: *Traffic engineering*. Pearson/Prentice Hall (2004)
4. Covenin: Equipos de control de semáforos. Comité de electricidad de Venezuela (1999)
5. Lema, C.S., Pedreira, L.P., Bouza, G., Allende, S.: Estudio de la optimización del tráfico en un cruce a través del ajuste de los ciclos de los semáforos mediante recocido simulado. En: XIX Jornadas ASEPUMA – VII Encuentro Internacional Anales de ASEPUMA (2019)
6. Balaji, P.G., Srinivasan, D.: Multi-agent system in urban traffic signal control. *IEEE Computational Intelligence Magazine*, 5(4), pp. 43–51 (2010)
7. Tlig, M., Bhour, N.: A multi-agent system for urban traffic and buses regularity control. *Procedia-Social and Behavioral Sciences*, 20, pp. 896–905 (2011)
8. Massobrio, R., Nesmachnow, S.: Urban data analysis for the public transportation system of Montevideo. In: *Ibero-American Congress on Information Management and Big Data Springer*, pp. 199–214 (2019)
9. Trejo-Sánchez, J.: Control de tráfico urbano basado en sistemas multiagentes (2006)
10. Fernández-Estrella, G.: Control de tráfico urbano para vehículos de prioridad (2020)
11. Loborg, P.: Error recovery in automation – an overview. In: *AAAI-94 Symposium on Detecting and Resolving Errors in Manufacturing Systems* (1994)
12. Di Febbraro, A., Giglio, D., Sacco, N.: Urban traffic control structure based on hybrid Petri nets. *IEEE Transactions on intelligent transportation systems*, 5(4), pp. 224–237 (2004)
13. Ng, K.M., Reaz, M.B.I., Ali, M.A.M.: A review on the applications of Petri nets in modeling, analysis, and control of urban traffic. *IEEE Transactions on Intelligent Transportation Systems*, 14(2), pp. 858–870 (2013)
14. SICE: <https://www.sice.com/lineas-de-negocio/sistemas-inteligentes-de-trafico/trafico-urbano> (2019)
15. CROSS: <http://www.cross.cz/es/productos-conteo-y-clasificacion-de-transito> (2018)
16. Robles, D., Ñañez, P.: Control y simulación de tráfico urbano en Colombia (2009)
17. Behrisch, M., Bieker, L., Erdmann, J., Krajzewicz, D.: SUMO—simulation of urban mobility: an overview. In: *Proceedings of SIMUL 2011, The Third International Conference on Advances in System Simulation* (2011)
18. Elidrissi, H.L., Tajer, A., Nait-Sidi-Moh, A., Dakkak, B.: A SUMO-based Simulation for Adaptive Control of Urban Signalized Intersection Using Petri Nets. In: *4th World Conference on Complex Systems (WCCS)*, pp. 1–6 (2019)