ISSN 1870-4069

Implementation of an Algorithm for the Transfer of Citrus Using an Intelligent Model for Trains

Fernando García-Isidro[1], Julissa Martinez-Lugo[1], Alberto Ochoa[2], José Alberto Hernández Aguilar[3]

[1] Instituto Tecnológico de Superior de Naranjos, Mexico

[2] Maestría en Cómputo Aplicado, UACJ, Mexico

[3] FCAeI, Universidad Autónoma del Estado de Morelos, Mexico

Abstract: We consider the implementation of an algorithm for the transfer of citrus using an intelligent model for trains, this research is performed based on the problem of packing in containers (Bin Packing), so that objects or boxes should be accommodated in a finite number of cubes considering to minimize the use of containers. The problem of packing containers will be understood and made according to the space of the container, the calculation will be made by maximizing the number of items - or boxes of citrus- that can be stored safely, the dimensions of the container shall also be addressed, taking into account that all have the same capacity and size, an algorithm based on MATLAB was used for the arrangement of objects in containers with containers crossing name is used.

Keywords: Container, Packaging, Trains, Capacity, Dimension, Objects, Bin Packing, MATLAB.

1 Introduction

For its simplicity and combinatorial nature of the problem of packaging container makes it an NP- hard problem, that is to say, that there is no algorithm that can solve all instances of the problem in a given polynomial time with respect to the number of input objects.

The problem of packing containers has great relevance for transporting, loading trains, trucks and cargo vehicles with weight capacity, distribution of tasks in homogeneous processes, and organization objects using a computer through blocks.

In order to solve the problem for a large number of objects it has been established several very simple algorithms such as:

Next fit (AS): In this algorithm objects are listed, the first is assigned to the container number 1, the rest are assigned to the current container if fit, otherwise is assigned the next container, the new container becomes the current container.

First-fit algorithm (PA): In this algorithm, the objects are listed as containers, therefore the first object is assigned to the container 1, the rest of the objects are assigned to another container where they fit, and if the objects do not fit in any container which are assigned a new one is created.

Best Fit algorithm (MA): This algorithm is a modification of the AS for which the current object to the container having the minimum space available to store the object is assigned. In case of a tie between two or more containers, the one with the lowest risk is chosen.

In order to improve these three types of algorithms as a first step you can rearrange the objects depending on the space they occupy in descending order.

A typical example of the problem and its solution according Lodi A, S Martello, Vigo D are shown in Figures 1 and 2 respectively.

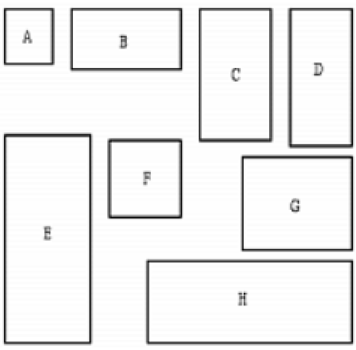

Fig. 1. Input.

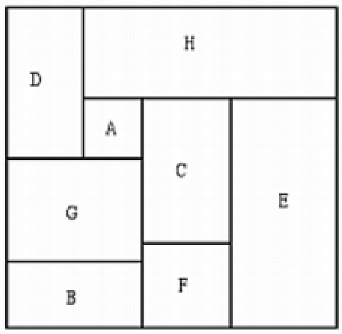

Fig. 2. Output.

Fig. 1. shows the objects (A, B, C, D, E, F, G, H) and observed that there are spaces between objects, on the other hand, Figure 2 shows a solution for this type of objects. [4]

For all types of wheeled vehicles able to run on a railway track are known as trains or rolling stock, which aims to transport different types of cargo.

Man's need to transport large amounts of cargo led him to build machines with great traction. George Stephenson built the first locomotive on 25 July of 1814, which later led to a railroad. At first locomotive it was used in coalmines where the drag of 40 tones was shown at a speed of 40 kilometers per hour.

The railroad was the product of the industrial revolution originated in England in the eighteenth and nineteenth centuries. The evolution of the locomotive has been remarkable, and its use has not only been for freight, but also for the transport of persons, both large and small distances.

The iron transport is the transport used worldwide. It offers a variety of different types of rail freight cars in boxcars, gondolas, hoppers, and trailer, duplex and triplex tank cars.

The vans are used to transport products that require protection from the environment without temperature control.

Otherwise boxcars transport perishable goods with temperature control; gondolier transport materials that do not require protection from the environment; the hoppers are used for the transportation of industrial products such as coal, scrap metal, etc.

Loads and dimensions are stipulated in international standards [6].

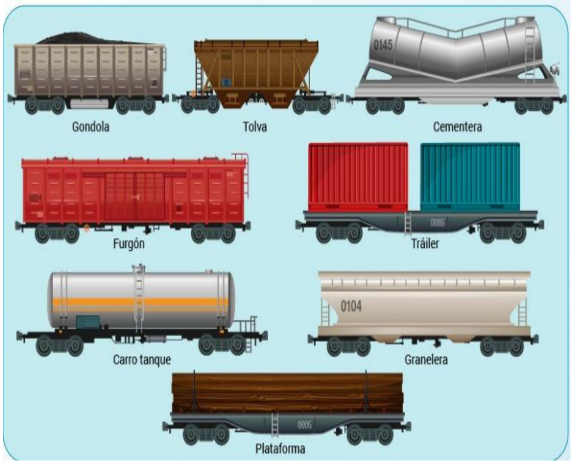

Fig. 3. Railroad transportation [6].

The speed of a freight train is about 70 kilometers per hour, is fixed by the general direction of railways for which takes two conditions [6].

The first: is established for each class of locomotive and is based by building system.

The second: is indicated by the state of the roads and the conditions of protection thereof.

The rate may be increased only with special authorization from the general direction of railroads.

Data on some containers to store items, restrictions of use and maximum total load

Table 1. Specifies the maximum load and dimensions of some transports containers currently on the market [6].

Name	Length	Width	Height
12’ steel dry box	3.431 m.	2.352 m.	2.395 m.
20’ steel dry box	5.900 m.	2.352 m.	2.395 m.
40’ steel dry box	12.022 m.	2.352 m.	2.395 m.
Name		**Maximum load**	
20’ dry		47,782 kg.	
40’ dry		58,058 kg.	
40’ high cube		57,550 kg.	

Fig. 4. Container [6].

2 Kind of Wagons

Boxcars: used to protect the goods from bad weather, theft and vandalism, the most used is the sliding walls.

Wagons edges: they are open boxes above and are used for transporting wood, scrap metal, etc.

Platforms: are used to transport vehicles, are considered high speed reaching 220 km/h, its structure is aluminum and not carbon steel.

Cage: used for transporting livestock.

Containerships: flatcars are having fasteners for containers.

Hoppers: are used to transport solid bulk cargo as aggregates, coal, minerals, etc. [1]

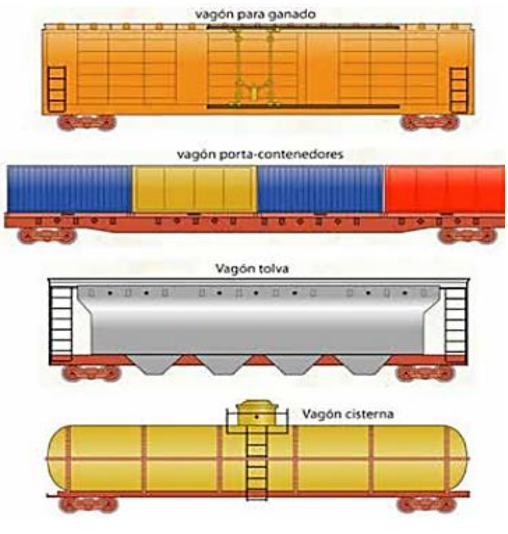

Fig. 5. Type of wagons.

3 Mode and Transport

Transport modes are systems for mobilization, used in the transfer of goods from one source point to a destination.

The means of transport are the different physical elements used in the modes of transfer of goods to distribute in diverse location with far distance.

All modes and means of transport of goods, have advantages and disadvantages which must counter the logistics manager, to select the appropriate transport to the type and amount of cargo to be sent.

Table 2. Mode and meaning of transport train.

Modes	media
Railroad	railway train
	They are special equipment with sealed wagons, chairs, and container type wagons. Usually charge bulk goods such as coal, grain, and platforms are also used for loading containers.

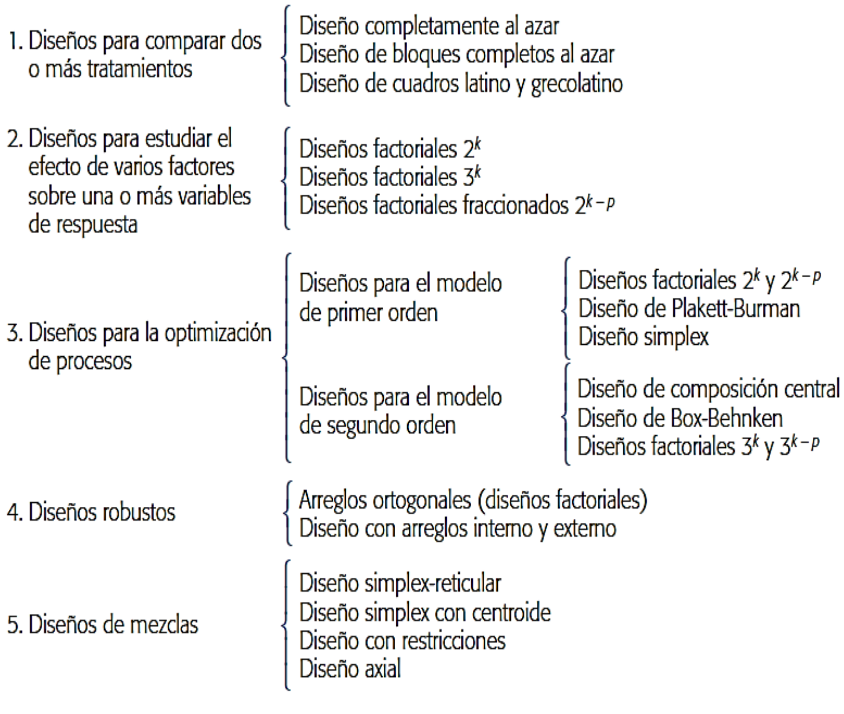

Fig. 6. Classification of experimental designs [8].

An algorithm for the optimization problem is represented by a primary quantity of items to save (multiplied by the size of occupying, being of the same type) , this result is subtracted from the total size of the container, if the space subtracts the container and if there are more items on hold, you choose to fill the container without triple the weight of all items, the weight of the container.

According [3] design of experiments is to plan and perform a set of tests in order to generate data that , when analyzed statistically , provide objective evidence to answer the questions raised by the experimenter situation.

4 Process for Obtaining Citrus Products

Harvesting methods are the same as those used for fruit eaten fresh , fruit is cut from the tree using ladders and is manually selected , the fruit is placed in baskets that can hold from 80 to 100 kilograms , to be carried to the transport vehicle [5].

Table 3. Major producers of citrus municipalities in Veracruz per hectare.

County	Naranja (Ha.)	Lima persa (Ha.)	Mandarina (Ha.)	Tangerina (Ha.)	Toronja (Ha.)	Total (Ha.)
TEMAPACHE, VER.	42,626	53	2,994	2,812	571	48,955
MARTINEZ DE LA TORRE, VER	14,423	16,049	6	644	2,790	33,912
TIHUATLAN, VER.	13,783	270	1,118	813	82	16,066
ATZALAN, VER.	4,840	4,383	55	2,264	210	11,752
TUXPAN, VER.	4,791	69	2,389	2,399	150	9,798
BENITO JUAREZ, VER.	8,000	0	0	0	0	8,000
CASTILLO DE TEAYO, VER	8,675	13	106	178	57	9,029
PAPANTLA, VER.	9,870	664	160	50	203	10,947
GUTIERREZ ZAMORA, VER	8,050	70	200	100	335	8,755
TECOLUTLA, VER.	6,376	441	41	151	265	7,275
MISANTLA, VER.	3,610	1,012	120	1,400	175	6,317
CHICONTEPEC, VER.	3,646	0	0	0	0	3,646
CAZONES DE HERRERA, VER.	3,191	16	58	189	250	3,704
TAMIAHUA, VER.	1,911	20	439	332	4	2,705
TLAPACOYAN, VER.	2,257	0	0	0	0	2,257

Table 4. Main municipalities of citrus growers in Veracruz per tonne.

County	Naranja (Ton.)	Lima persa (Ton.)	Mandarina (Ton.)	Tangerina (Ton.)	Toronja (Ton.)	Total (Ton.)
TEMAPACHE, VER.	701,679	105	59,875	39,339	12,346	813,344
MARTINEZ DE LA TORRE, VER	185,358	173,789	60	8,356	76,093	443,656
TIHUATLAN, VER.	192,962	3,100	22,360	16,260	0	234,682
ATZALAN, VER.	59,865	46,540	750	26,598	4,902	138,655
TUXPAN, VER.	57,492	1,021	38,224	28,788	3,000	128,525
BENITO JUAREZ, VER.	128,000	0	0	0	0	128,000
CASTILLO DE TEAYO, VER	121,450	91	1,900	3,560	0	127,001
PAPANTLA, VER.	98,400	5,500	0	0	4,000	107,900
GUTIERREZ ZAMORA, VER	75,800	750	2,500	1,200	3,800	84,050
TECOLUTLA, VER.	71,520	4,350	550	1,800	4,750	82,970
MISANTLA, VER.	34,397	11,955	864	19,880	3,430	70,526
CHICONTEPEC, VER.	58,336	0	0	0	0	58,336
CAZONES DE HERRERA, VER.	51,056	150	1,200	3,780	0	56,186
TAMIAHUA, VER.	26,754	296	7,016	3,981	80	38,127
TLAPACOYAN, VER.	33,339	0	0	0	0	33,339

Fig. 7. Citrus production in Mexico.

Veracruz leading citrus production in Mexico with a volume of 2.9 million tons. Followed by Tamaulipas with 638 thousand tons, and Mexican lime -producing states, Colima and Michoacan with 590,000 and 462,000 tons respectively [5].

Table 5. States citrus production in Mexico.

No	ESTADO	PRODUCCIÓN (ton.)	No	ESTADO	PRODUCCIÓN (ton.)
1	Veracruz	2,987,973.65	15	Quintana Roo	42,044.45
2	Tamaulipas	638,262.49	16	Jalisco	36,000.58
3	Colima	590,372.49	17	Baja California Sur	32,624.72
4	Michoacán	462,999.49	18	Chiapas	20,438.12
5	San Luís Potosí	436,915.80	19	Nayarit	16,022.84
6	Nuevo león	399,739.22	20	Sinaloa	14,344.80
7	Oaxaca	253,270.70	21	Baja California	11,318.94
8	Yucatán	251,073.34	22	Morelos	6,765.90
9	Puebla	225,595.90	23	Durango	2,420.70
10	Sonora	191,208.30	24	México	1,804.90
11	Tabasco	141,300.00	25	Querétaro	1,415.50
12	Guerrero	85,816.56	26	Zacatecas	260.00
13	Hidalgo	58,731.05	27	Aguascalientes	124.00
14	Campeche	58,267.56	28	Guanajuato	44.60

5 Problem Packaging of Objects in Containers: Bin Packing

The problem of distribution of objects in containers, in English Bin Packing is a classic problem of NP- hard combinatorial optimization . Because Bin Packing is a highly complex problem can not be solved big cases using an exact algorithm . An optimal solution can be found by considering all ways to make a partition of n objects in a subset of size n or smaller.

6 Algorithms for Optimization Problems and Generic Packaging Containers

$$O(g(x)) = \left\{ \begin{array}{l} F(x)\text{: Exists } c,\ x_0 > 0 \text{ such as} \\ \forall x \geq x_0\text{: } 0 \leq |f(x)| \leq c\,|g(x)| \end{array} \right\}$$

A container f (x) belongs to items g (x) , when there is a positive constant c such that from a quantity of the items x0 , f (x) not on passing containers (x) , means that the function f is less aga from a given value by a constant factor.

Generic algorithm (GA) .
Assuming a search space x, and the function:
f: x → R
The general problem is to min f
X ε x

Where x is the vector of decision variables and f is the objective function. This methodology consists of three major phases . The first is the generation population (chromosomes) ; the second is the recombination of these chromosomes or crossover criterion and the last mutation . Finding the best solution it is guided by the results and evaluation of the objective function f for each of the individuals generated . Finally , based on this evaluation, the chromosome with higher value of f represents the best solution [4].

```
Elegir una población inicial de cromosomas;
 mientras (la condición no es satisfecha) hacer
    Repetir
        si (condición de cruce es satisfecha) entonces
        {Seleccionar los cromosomas padres;
         Elegir parámetros de cruce;
         mostrar cruce};
        si (condición de mutación es satisfecha) entonces
        {elegir puntos de mutación;
        mostrar mutación}
        evaluar funcion objetivo
        hasta que se crean hijos suficientes;
      Seleccionar nueva población;
    fin mientras.
```

Fig. 8. Description of the operation of AG [4].

Program in Matlab of the generic algorithm for the problem of packaging in containers. By using a container crossing.

```
function [pob bins]=cruce(numind,nuevabins,nuevapob,pesos,capacidad)
Cruce de contenedores
bjacm=[];
 for i3=1:size(listap,2)
 [listac indc]=sort((hijo(1:numbinsaux,numobj+2))',2,'ascend');
 for i4=1:size(listac,2)
 if (listac(i4)+listap(i3)) < capacidad
 hijo(indc(i4),hijo(indc(i4),numobj+1)+1)=vecobj(indp(i3));
 hijo(indc(i4),numobj+1)=hijo(indc(i4),numobj+1)+1;
 hijo(indc(i4),numobj+2)=hijo(indc(i4),numobj+2)+listap(i3);
 objacm=[objacm vecobj(indp(i3))];
 listac(i5)=listac(i5)+listap(i3);
```

```
break
end
end
end
vecobj=setdiff(vecobj,objacm);
if size(vecobj,2)˜=0
indbin=1;
numbinsaux=numbinsaux+1;
for i3=1:size(vecobj,2)
if (hijo(numbinsaux,numobj+2)+pesos(vecobj(i3)))>capacidad
numbinsaux=numbinsaux+1;
indbin=1;
end
hijo(numbinsaux,indbin)=vecobj(i3);
hijo(numbinsaux,numobj+1)=indbin;
hijo(numbinsaux,numobj+2)=hijo(numbinsaux,numobj+2)+pesos(vecobj(i3));
indbin=indbin+1;
end
end
nuevapob(:,:,conthijo)=hijo;
nuevabins(conthijo)=numbinsaux;
conthijo=conthijo+1;
end
end
pob=nuevapob;
bins=nuevabins;
```

In this code is explain each iteration related with the correct location of objects in a transport model by train.

Table 6. Iteration 1.Testing generic algorithm: Crossing container.

Number of executions	Data	Result	% Total	Probability Overload
1	0,1,1,1,1	39.8	37.1	0.374
2	1,0,0,0,1	24.4	22.8	0.226
3	1,1,0,0,0	22.4	20.9	0.208
4	0,1,0,0,0	20.4	19.0	0.189
Sum		107	99.8	0.998

Table 7. Executions in Matlab program.

>> [pob bins]=cruce(0,1,1,1,1) **pob =** **1** **bins =** **1**	**>> [pob bins]=cruce(1,0,0,0,1)** **pob =** **0** **bins =** **0**	**>> [pob bins]=cruce(0,1,0,0,0)** **pob =** **0** **bins =** **1**	**>> [pob bins]=cruce(1,1,0,0,0)** **pob =** **0** **bins =** **1**

Table 8. Iteration 2.

Number of executions	Data	Result	% Total	Load problems
1	1,1,0,0,0	22.1	31.9	0.319
2	0,1,0,0,0	20.4	28.9	0.289
3	1,0,0,1,1	18.4	26.1	0.261
4	0,1,1,0,1	9.8	13.9	0.139
Suma		70.4	100.8	1.008

Table 9. Executions in Matlab program.

>> [pob bins]=cruce(1,1,0,0,0) **pob =** **0** **bins =** **1**	**[pob bins]=cruce(0,1,0,0,0)** **pob =** **0** **bins =** **1**	**[pob bins]=cruce(1,0,0,1,1)** **pob =** **0** **bins =** **0**	**[pob bins]=cruce(0,1,1,0,1)** **pob =** **1** **bins =** **1**

Table 10. Iteration 3.

Number of executions	Data	Result	% Total	Load problems
1	1,0,0,0,1	24.4	22.8	0.228
2	0,1,1,1,1	39.8	37.1	0.371
3	1,1,0,0,0	22.4	20.9	0.209
4	0,1,0,0,0	20.4	19.0	0.190
Suma		107	99.8	0.998

Table 11. Executions in Matlab program.

>> [pob bins]=cruce(1,0,0,0,1) pob = 0 bins = 0	**[pob bins]=cruce(0,1,1,1,1) pob = 1 bins = 1**	**[pob bins]=cruce(1,1,0,0,0) pob = 0 bins = 1**	**[pob bins]=cruce(0,1,0,0,0) pob = 0 bins = 1**

Table 12. Iteration 4.

Number of executions	Data	Result	% Total	Load problems
1	1,1,0,0,0	22.4	31.5	0.315
2	0,1,0,0,0	20.4	28.7	0.287
3	1,0,0,1,1	18.4	25.9	0.259
4	0,1,1,0,1	9.8	13.8	0.138
Suma		71	99.9	0.999

Table 13. Executions in Matlab program.

[pob bins]=cruce(1,1,0,0,0) pob = 0 bins = 1	**[pob bins]=cruce(0,1,0,0,0) pob = 0 bins = 1**	**[pob bins]=cruce(1,0,0,1,1) pob = 0 bins = 0**	**[pob bins]=cruce(0,1,1,0,1) pob = 1 bins = 1**

Table 14. Iteration 5.

Number of executions	Data	Result	% Total	Load problems
1	1,0,0,0,1	24.4	22.8	0.374
2	0,1,1,1,1	39.8	37.1	0.226
3	1,1,0,0,0	22.4	20.9	0.208
4	0,1,0,0,0	20.4	19.0	0.189
Suma		107	99.8	0.998

Table 15. Executions in Matlab program.

>> [pob bins]=cruce(1,0,0,0,1) **pob =** **0** **bins =** **0**	**>> [pob bins]=cruce(0,1,1,1,1)** **pob =** **1** **bins =** **1**	**>> [pob bins]=cruce(1,1,0,0,0)** **pob =** **0** **bins =** **1**	**>> [pob bins]=cruce(0,1,0,0,0)** **pob =** **0** **bins =** **1**

Table 16. Capacity and containers are listed in the following table.

	Container 1	Container 2	Container 3	Container 4	Container 5
Capacity (tons)	2	2.4	3	4	4.4

The capacity of all containers is determined as follows :

(1, 1, 1, 1, 1) = 41.8 (1, 1, 1, 1, 0) = 29.4 (1, 1, 1, 0, 1) = 29.8
(1, 1, 1, 0, 0) = 7.4 (1, 1, 0, 1, 1) = 38.8 (1, 1, 0, 1, 0) = 8.4
(1, 1, 0, 0, 1) = 8.8 (1, 1, 0, 0, 0) = 22.4 (1, 0, 1, 1, 1) = 39.4
(1, 0, 1, 1, 0) = 9 (1, 0, 1, 0, 1) = 9.4 (1, 0, 1, 0, 0) = 23
(1, 0, 0, 1, 1) = 18.4 (1, 0, 0, 1, 0) = 24 (1, 0, 0, 0, 1) = 24.4
(1, 0, 0, 0, 0) = 20 (0, 0, 0, 0, 0) = 18 (0, 1, 1, 1, 1) = 39.8
(0, 1, 1, 1, 0) = 9.4 (0, 1, 1, 0, 1) = 9.8 (0, 1, 1, 0, 0) = 23.4
(0, 1, 0, 1, 1) = 18.8 (0, 1, 0, 1, 0) = 24.4 (0, 1, 0, 0, 1) = 24.8
(0, 1, 0, 0, 0) = 20.4 (0, 0, 1, 1, 1) = 19.4 (0, 0, 1, 1, 0) = 25
(0, 0, 1, 0, 1) = 25.4 (0, 0, 1, 0, 0) = 21 (0, 0, 0, 1, 1) = 34.4
(0, 0, 0, 1, 0) = 22 (0, 0, 0, 0, 1) = 22.4

Results are calculated as follows:

The values 18 and 8 only considered the containers 4 and 5 have a storage capacity of relatively equal [8].

Examples:
(1, 1, 1, 1, 1)= 2+2.2+3+4+4.4+18+8=41.8,
(0, 1, 0, 1, 0)=2,4+4+18=24.8,
(0, 0, 1, 1, 0)=3+4+18=25.

7 Conclusions and Future Work

The results obtained on the problems of packaging containers experimentally demonstrate that combinatorial problems that are untreatable or difficult to solve with an exhaustive search or classical techniques, can be solved satisfactorily with evolutionary algorithms that are modified appropriately.

An advantage of these processes is that it is not necessary to know the description of the problem, a way to solve the problem is with the implementation of the generic algorithm, but is complicated adapting techniques to find an optimal solution, and thats where the generic algorithms have found an important area of solution.

In future work we should be studied more chromosomes of the cell or humans to solve problems with the help of artificial intelligence.

References

1. Berbey, A., et al.: Trenes: Material rodante del transporte ferroviario.
2. Caliz, E.: Validación de la calidad de las soluciones obtenidas en la tesis de maestría optimización integral del problema de carga de contenedores tridimensionales (2012)
3. Gutiérrez, A., et al.: Análisis y diseño de experimentos. Tercera edición, Editorial Mc Graw Hill (2013)
4. Hernández, N., et al.: Introducción a Matlab para Resolver Problemas de Ingeniería Aplicando Algoritmos Genéticos (2013)
5. Licona, J.: Estudio de Mercado para identificación de necesidades de infraestructura logística para la comercialización de jugo de cítricos en Veracruz (2009)
6. Mata, G., et al.: Un problema de tipo Bin Packing (2011)
7. Reglamento General de Ferrocarriles. Aprobado por Decreto Nº 90.325 del 12 de Setiembre de 1936 y actualizado al 31 de Diciembre de 1995
8. Vidal, A.: Algoritmos heurísticos de optimización. Universidad de Santiago de Compostela, Facultad de Matemáticas Master en Técnicas Estadísticas (2013)