Adaptation of Particle Swarm Optimization technique to optimize the choice of routes between two points in the transport system of the city of Leon, Guanajuato.

Oscar Alberto García Ortiz[1] Alberto Ochoa Zezzatti[2] Sodel Vázquez Reyes[3]

[1] ITESM/CIATEC (Academia Mexicana de Ciencias), Zacatecas, Zacatecas, México 2009
Cooreal08@gmail.com
[2] Universidad Autónoma de Ciudad Juárez megamax8@hotmail.com
[3]ITESM, Zacatecas, Zacatecas, México 2009 vazquezs@itesm.mx

Abstract. The optimization problems have always existed and the pair of them have developed various methods to propose solutions to them. This paper addresses the optimization of the transshipment problem in public transport routes in the city of Leon, Guanajuato. It was used the metaheuristic known as Particle Swarm Optimization (PSO), through which it was determined what was best route to take to travel from a specific point in the city to another, which is made taking into account the timing, frequency of buses and number of buses in circulation at any given time.

So that is presented using PSO to optimize especially the time to travel from one point to another as long as I have to change.

Keywords: Particle Swarm Optimization, particle velocity, acceleration coefficients, weight of inertia.

INTRODUCTION

The efficiency of public transport in the city of Leon has been an important issue, which is why improved transport system by Optibus system, where there are different types of routes planned to try to provide a quality transport service. However, if you want to travel from one point to another in the city there are several options when choosing the route to take, in order to arrive as quickly as possible and have to walk as little as possible. People choose the paths based on their experience; however we can not say that their choices are always the best.

There are several methods that could attack this optimization problem, this paper addressed the problem by the technique of PSO, one of the techniques of collective intelligence which is inspired by the movement which describes the flocks of birds or schools of fish.

METHODOLOGY

PSO simulates the social behavior of birds, they individually share information about their position, speed and adaptation, and behavior of the flock is influenced to increase the probability of migration to regions of greater adaptability [1].

In PSO a swarm of n individuals directly or indirectly communicates with others through search directions [2]. As shown in Fig 1.

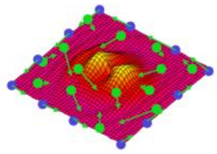

Fig 1. Movement and communication between the particles of PSO.

In PSO the particles never die, in this model the particles travel through an n-dimensional space of decision variablesX_{id}, each particle keeps track of its position X_{id}, V_{id} speed and recalls the best position it has found pBest. The particle with the best pBest is called the leader and global best position is called, GBest org_{id}. The next position of the particle is calculated by adding the term rate at its current position as follows:

$$X_{id} = X_{id} + V_{id}\ . \qquad (1)$$

On the other hand the term combines the speed of the particle local information with global information in the flock, as shown below:

$$V_{id} = \omega \cdot V_{id} + \underbrace{\varphi_1 \cdot rnd(\ ) \cdot (pBest_{id} - X_{id})}_{\text{Cognitive}} + \underbrace{\varphi_2 \cdot rnd(\ ) \cdot (g_{id} - X_{id})}_{\text{Social}} \qquad (2)$$

As can be seen from the above formula, the rate is influenced by a social exchange, the second term is known as cognitive, while the latter as the social component. [3]

Regarding the parameters that were used Φ1 = Φ2 = 2, are called acceleration coefficients, w means the weight of inertia and rnd () refers to a uniform random number between 0 and 1.

The motion of particles is guided by the best particle at the present time [4]. Below shows the pseudo code of a basic local PSO.

```
t=0;
for i=1 until number_particles
     initialize X_i and V_i
while (no stop condition is met) do
     t ← t + 1
     For i=1until number_particles
     Evaluate X
     If F (X_i) is better than F(pBest_i) then
     pBest_i ← X_i; F(pBest_i) ← F(X_i);
     For i=1 until number_particles
          Choose lBest, the particle with the best fitness of
          environment X_i
          Calculate V_i the velocity of X_i according to
          pBest_i and lBest_i
          Calculate the new position of X_i according to X_i and V_i
Return the best solution found
```

TOOLS DEVELOPED

The prototype is a hybrid system, developed in Java, using PSO technique.

It began with the analysis of the structure and functioning of transport routes in the city, and the factors that should be taken into account to solve the transshipment problem. In order to generate the modeling of the problem from its restrictions.

So, we took into account schedules, frequency of passing buses and distances, and the stops of each route.
From the structure of routes, there was a small prototype to test the algorithm, keeping the current organization and the seasons, ie simulating the feeder routes, conventional, core, suburban and auxiliaries.

The software calculates the route in 3 stages or phases, as it is designed for trips that include mandatory transfers, the first step involves taking a route anyone other than trunk and take you to the transfer station where you can transfer to trunk road.

The second stage is the trunk road route to reach the nearest station to final destination. And the last stage is the transfer which uses the PSO technique to determine the best route for transshipment, based on time it will take to reach certain season, the time it takes to get the next bus at that stop, it will take time reach the nearest station to the final destination and the time it takes to move if necessary walk from the station where low to its destination.

Which was used for the following function:

$$\min z = t1 + t2 + tcp + +terf + tr + tc \,. \qquad (3)$$

Where:

t1 = time it takes from the point of origin to the transshipment point with a trunk road (transfer station)

t2 = travel time for the trunk road to reach the nearest station to final destination

tcp = time it takes to walk to the final stop on the route to take

terf = timeout of the final route

tr = travel time of the final route

tc = time walking if necessary to final destination

The algorithm searches for routes that have stations closer to where they fell from the trunk road, and from there runs a local PSO each way, to finally get the best option in terms of time is concerned. The best local PSO particle is obtained from 50 iterations, and set a maximum output condition.

RESULTS

The prototype uses a bird population is depending on the route, since the particles represent the buses running at some point, and during peak hours there are more buses running, also depending on the number of users of a particular route.

The program initializes the position of birds, representing the bus, at one point at a certain time, and assesses their suitability based on how long it will take to get to a certain station where the user will take to reach their final destination.

To evaluate the ability of birds was used

$$\left.\begin{array}{l} \min w = tcer + trr + tcdf \\ \text{restricted to: } hauet + tcer \ \leq hpart \end{array}\right\} \quad (4)$$

Where:

hauet = time of arrival to the last station trunk

tcer = time walking, if necessary, to stop the next route to take

trr = travel time of the last route taken

tcdf = time walking to the final destination

hpart = hour of the particle

Furthermore the position of the particle must be prior to the station where the user will take the bus. If a particle violates any of the restrictions is given a high value of fitness to be a particle can not lead the flock.

The software determines the best 3 choices for the user chooses. The first two stages, ie the stage where you take the route no backbone, transshipment and trunk road tour, are equal in the 3 options, what changes is the choice of the latter route, where you applied the PSO . Each option shows the time it takes to reach its final destination.

Below are the outputs of the program.

Fig 3. Figure showing the program interface.

Fig 3 shows the output interface of the program, it loads a text file with important data on the routes, and also takes additional information from a database on timing and correlation.

Then there are fields to put your origin and destination, which are names of streets whose location is in a database also must include at what time you want to travel.

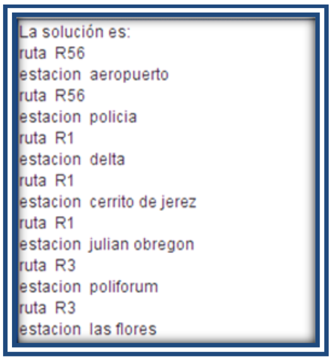

Fig 4 Solution to the 2 stage (the last station of the trunk road)

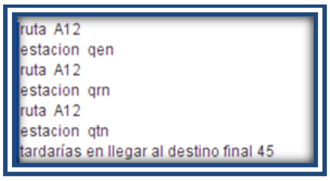

Fig 5. Solution 1 granted by PSO

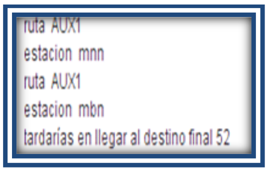

Fig 6. Solution 2 granted by PSO

Fig 7. Solution 3 granted by PSO.

As shown in Figure 4 shows the part of the solution is the same for the 3, ie the first two stages, in Figures 5, 6 and 7 show the 3 options granted by PSO which describes the route to take, what stops, stops that will go up to the nearest your final destination and the approximate time it will take to realize the full itinerary.

CONCLUSIONS

The work succeeded in creating an intelligent system, using a technique of collective intelligence as the PSO. This technique can optimize the choice of routes when transshipment at public transport system of the city of León, Guanajuato. Providing users a new option to move more efficiently within the city, not just based on experience.

The results show that the PSO is a technique with many applications, as this study shows, among them is the transfer within a transportation system satisfactorily.

REFERENCES

1] Huang, H. , Zhifeng,H. Particle Swarm Optimization Algorithm for Transportation Problems. Particle Swarm Optimization (January 2009).

[2] Kennedy, J., Eberhart, R. Particle Swarm Optimization. In Proceedings of IEEE International Conference On Neural Networks, pp 1942-1948, IEEE, (1995).

[3] Ochoa A., Hernández A., Artificial Societies and Social Simulation using Ant Colony, Particle Swarm Optimization and Cultural Algorithms.

[4] Herrera, F., Alcalá R., Bioinformática 2008-2009, (2009). http://sci2s.ugr.es/docencia/bioinformatica/bio/ficheros/Temario/Tema%2010%20-%20Evolucion%20Diferencial.pdf