Intelligent system to emergences based on ant colony

Omar Cano Cendejas, S. Jöns and José Sandoval.

[1] Centro de Innovación Aplicada en Tecnología Competitivas,
omar_kano7@hotmail.com

(Paper received on August 27, 2010, accepted on October 20, 2010)

Abstract. The problem of the shortest path is a typical problem of optimization. This paper presents the technique of Ant Colony Optimization heuristic used in the solution of this problem applied to the improve of routing vehicles in the H. Fire Department of the city of Leon at Mexico. We describe the diverse components to characterize this problem, and the use of a Bioinspired Algorithm in this case, we propose the use of Ant Colony, thus implementing a Tool in Java which determine the best tracks to the vehicles and realize experiments to probe the validations of this software, focusing in the comparative with Dijikstra algorithm and determine the quality of our results, in the future of this research, we try to determine an innovative perspective related with pheromone evaporation and as this topic is determinative to found and remember best solutions quickly.

Keywords: Ant Colony, Shortest path, vehicle routing.

1 Introduction

Today's digital maps are an increasingly common, with the progress that has been made in technology, these maps are becoming more sophisticated, able to come to find specific locations, draw routes, among others. Also noteworthy that show how the information has improved dramatically, as they can get to show from traditional maps to maps with totally real images taken from the air, satellite, or even a hybrid version of these two.

The motivation of this project is specifically focused on the use of this increased interaction today, to achieve an improvement in logistics H. Fire Department of the Central Apollo in the city of Leon, Mexico to get faster to the scene. To provide assistance to citizens, firefighters need the path of a route to arrive as quickly as possible to an incident, and these in order of importance: Firefighting, rescue and Prevention Action on public hazard.

In all these activities the time is vital, because with a timely arrival can decrease the effect of damage in a fire, to prevent an explosion in the leak case, finding a person alive, among others.

©G. Arroyo-Figueroa (Ed)
Special Issue in Advances in Artificial Intelligence and Applications
Research in Computing Science 51, 2010, pp. 31-37

2 Description of the Model Components

In this section we offer details of each component related with the application domain involve in the problem, in our case Logistics to emergence although the improve of the route of vehicles of Fire Departments by the use of a Bioinspired Algorithm.

2.1 Shortest Route

The problem known as the shortest path or shortest route, as its name suggests trying to find the minimum or shortest route between two points. This minimum may be the distance between origin and destination points or the time to travel from one point to another. Mathematically, this system is described as a weighted graph G= (V, A, d) where vertices are represented by V= $\{V_0, V_1, \ldots, V_n\}$, and arcs are represented by A= $\{(v_i, v_j) \mid i \neq j\}$. The distances associated with each arc are represented by the variable C_{ij} measured by the Euclidean distance.

The objective function of the problem [6]:

$$\min z = \sum_{\substack{\text{Todos los arcos} \\ \text{definidos}}} C_{ij} X_{ij} \tag{1}$$

Decision variables:

X_{ij}: Action to move from node i to node j
0 indicates that there is no displacement and 1 that indicates that yes there is movement.
C_{ij}: Cost or time to get from node i to node j.

Restrictions

Total input flow = Total output flow

$$(\text{External input into node } j) + \sum_{\substack{i \\ \text{Todos los arcos} \\ \text{Definidos (i,j)}}} X_{ij} = (\text{External output from node } j) + \sum_{\substack{k \\ \text{Todos los arcos} \\ \text{Definidos (j,k)}}} X_{jk} \tag{2}$$

This type of optimization problems can't be solved using exact methods, we can't find its optimal solution with acceptable computational efforts. Since the early 50's many algorithms have been developed to find the solution to this problem by finding good solutions but not necessarily optimal solutions. In the 80 's, the solution techniques focused on the implementation of general-purpose metaheuristics including between the ant colony, genetic algorithms, tabu search, among others.

2.2 Shortest path algorithm

The shortest path algorithm, also called the Dijkstra algorithm is an algorithm for determining the shortest path given a source vertex to other vertices in a directed graph with weights on each edge. The shortest path algorithm belonging to the greedy algorithm [7] is an efficient algorithm of complexity $O(n^2)$ where n is the number of vertices, used to find the least cost path from a source node to all other nodes in the graph. It was designed by Dutchman Wybe Edsger Dijkstra in 1959 [10]. The foundation on which sits this algorithm is the principle of optimality, the solution is built with the election of local optima in the hope of obtaining a global optimum.

2.3 Ant Colony

Ant colony optimization (ACO) algorithm is a new simulative evolutionary algorithm named ant colony system and was proposed in the 1990s by Italian scholar M. Dorigo. It has been applied to TSP, allocation problem, JSP, and got excellent results. Hence, more attention has arisen to the ant colony system, and the model has been applied to many practical problems [5, 9].

The Ants are social insects that live in colonies and that, because of their mutual cooperation, are capable of displaying complex behaviors and difficult tasks from the point of view of an individual ant. An interesting aspect of the behavior of many species of ants is their ability to find the shortest path between their nest and food sources. This is especially interesting when you consider that many species of ants are almost blind, which avoids the use of visual cues [8]. While make their way between the nest and food source, some species of ants deposit a chemical called pheromone. If there is no trace of pheromone, the ants move essentially random manner, but when there is pheromone deposited, are more likely to be traced [8]. The choice between different ways takes place when several paths cross. Then, the ants choose the way forward with a probabilistic decision biased by the amount of pheromone: the stronger the pheromone trail, the more likely selection. Because ants deposit pheromone on the path to follow, this behavior leads to a self-reinforcing process that concludes with the formation of traces marked by a high concentration of pheromone. This behavior also allows the ants find the shortest path between their nest and food source [2].

As time passed and while the ants are most promising on the roads, they will receive a higher amount of pheromone. This happens with that being the shortest path, the ants that are able to find food more quickly, they begin their return journey before. Then, in the shortest path a trail of pheromone be slightly higher and, therefore, decisions of the following ants will be directed more to the way [8]. In addition, this road will receive a greater proportion of pheromone by ants returning him by returning to the road longer. This process ends by making the probability that an ant chooses the shortest path increases progressively and ultimately the path of the colony converge to the shortest of all possible ways.

3 Tool Developed

Intelligent tool was developed using the Ant Colony algorithm and programming language Java (J2SE), began with the creation of the graph for the central area covering the Apollo fire a total of 2451 streets, avenues and boulevards (edges) and 1710 nodes, subsequently designed an entity called "object" to store information about each node, as the impact to neighboring nodes and their respective distance. These objects were related to a data structure called a multidimensional array which saves computer resources, because it does not cause the waste of memory cells generates a square incidence matrix, it stores only the necessary space.

Once the multidimensional array is implemented the Ant Colony algorithm, which has proven effective in solving NP-Hard problems [2]. The structure of the generic algorithm is as follows [5]:

Algorithm: Optimization based on Ant Colony

```
Inicializar_parametros()
while not condicion_parada()
    for ant=1 to n
          construir_solucion()
          evaluar_solucion()
          actualizar_feromonas()
    end for
end while
```

The software implements the ability to block and alter the meaning of the streets, a fact that occurs in the central city of Leon because of events, accidents, public works, among others. The method Inicializar_parámetros enters the source node, the destination node, blocked streets and the number of ants involved in the search for the solution. Construir_solución takes place ants moving randomly with both probability using the Monte Carlo method if there is already a trail of pheromone. Once an ant has found Evaluar_solución the destination node determines if the journey is of good quality, discarding those paths that do not decrease the distance obtained by other ants, and Updating pheromone if you have found a shorter route.

The system in the user interface displays the found routes to the destination, with the option of displaying all or a particularly within a map (Figure 1), which has the options of adding landmarks (churches, schools , hospitals, parks, rivers, etc.), zoom, view the different layers, stored in the route file, export the map as an image and sent via Bluetooth to a mobile device.

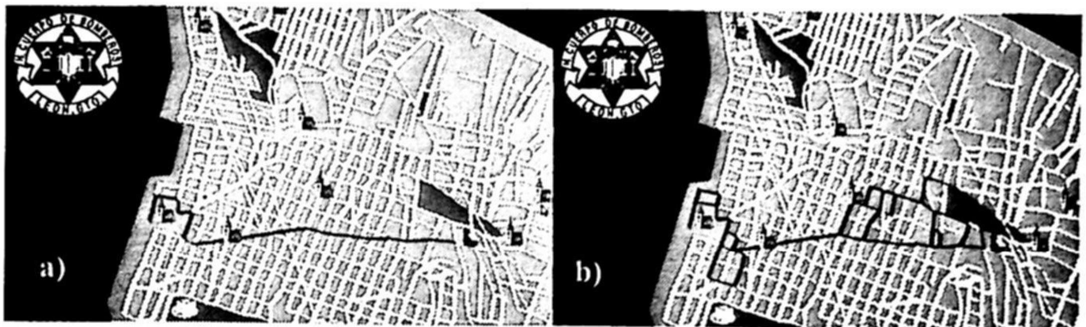

Figure 1. *a) Drawing of a single route b) Drawing of five routes.*

4 Experimental Results

The proposed algorithm was compared with the algorithm of Operations Research: Shortest path (Dijkstra), The comparison was carried out with the generation of 20 runs starting from the central fire (node 759) to different nodes (Table 1).

Table 1. *Results.*

No.	Node		Obtained Distance	
	Origin	*Destination*	*Dijkstra*	ACO
1	759	945	222	212
2		614	464	507
3		903	755	841
4		941	732	698
5		1044	693	709
6		1202	984	1093
7		1094	953	927
8		1057	538	538
9		1418	231	231
10		170	338	328
11		526	347	324
12		462	718	718
13		846	859	1030
14		524	359	333
15		809	365	406
16		1107	886	1011
17		698	302	302
18		1062	517	499
19		1342	519	564
20		1199	885	934

The results were ontained with μ=25.15seg and σ=15.65seg, in 35% of cases the Ant Colony give better results than the shortest path algorithm (1, 4, 7, 10, 11, 14 and 18), in 20% the results were similar (8, 9, 12 and 17) and 45% was surpassed by shortest paths.

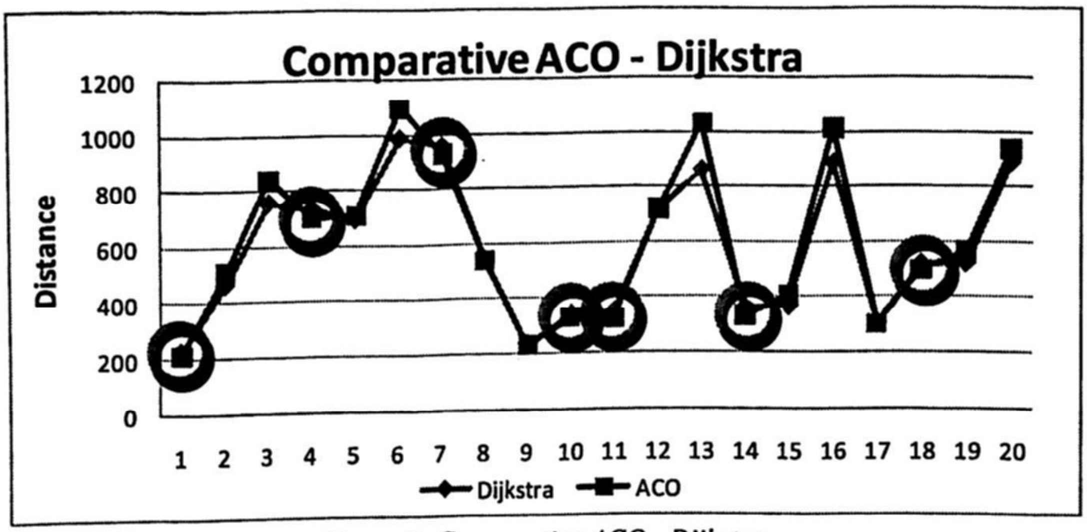

Figura 2. Comparative ACO - Dijkstra

5 Conclusions and Future Work

The algorithm is currently implemented gives good quality solutions to an NP-hard problem, improving by 35% of cases the routes provided by the shortest path algorithm.

In the 45% when the shortest path algorithm exceeds the ACO is attributed to not yet been implemented evaporation of the pheromone, the pheromone in nature may remain from a few hours to several months depending on several different aspects, such as ant species, soil type [8], causing a minor influence on the effect of evaporation in the process of finding the shortest path. Due to the long persistence of pheromone, it is difficult for the ants "forget" a path that has a high level of pheromone but have found a way even shorter. Keep in mind that if this behavior is transferred to the computer to design a search algorithm can find that quickly become stuck in a local optimum.

Based on the results obtained we recommend the implementation of heuristic algorithms such as ant colony, which have proven to do well on a variety of problems [4]. As future work remains to implement the evaporation of the pheromone, find benchmarks that are being used at international level and prove to those instances of the problem, replicate the project using Java (J2ME) for the system to operate on mobile devices which provide advantages to having with the system in units of H. Fire Department.

6 References

1. J. E. Bell. Ant colony optimization techniques for the vehicle routing problem. Science Direct. 2004

2. M. Dorigo, M. Birattari. Ant Colony Optimization; Artificial Ants as a Computational Intelligence Technique. IEEE Computational Intelligence Magazine. 2006

3. M. Dorigo, M. Birattari. On the Invariance of Ant System. Springer-Verlag Berlin Heidelberg. 2006

4. J.C. Ponce. Diseño de un Modelo de Optimización de Colonias de Hormigas Aplicado al Problema del Clique Máximo. Tesis Doctoral. UAA. 2010.

5. M. Dorigo y G. Di Caro. The Ant Colony Optimization meta-heuristic. En D. Corne, M.Dorigo, y F. Glover, editores, New Ideas in Optimization, páginas 11-32. McGraw Hill, London, UK, 1999.

6. H.A. Taha. Investigación de Operaciones. Prentice Hall. 2004.

7. Grimaldi, R. P., Matemáticas Discretas y Combinatoria, Editorial Addison Wesley. 1997.

8. Sergio Alonso, Oscar Cordón, Iñaki Fernández de Viana y Francisco Herrera. La Metaheurística de Optimización Basada en Colonias de Hormigas: Modelos y Nuevos Enfoques. 2003.

9. Weiqing Xiong, Liuyi Wang & Chenyang Yan. Binary Ant Colony Evolutionary Algorithm. International Journal of Information Technology, Vol.12, No.3, 2006.

10. Johnsonbaugh, R. Matemáticas Discretas. Grupo Editorial Iberoamericana. 1988.