Intelligent Tool for Decision Making Associated With Hospitalization and Sandstorms for the Optimization of Ambulances

Chapter · August 2019

DOI: 10.4018/978-1-5225-8131-4.ch003

Luis Ernesto Cervera-Gómez
El Colegio de Chihuahua

Estrella Herrera
University of Texas at El Paso

Handbook of Research on Metaheuristics for Order Picking Optimization in Warehouses to Smart Cities

Alberto Ochoa Ortiz-Zezzatti
Autonomous University of Juarez City, Mexico

Gilberto Rivera
Autonomous University of Juarez City, Mexico

Claudia Gómez-Santillán
National Institute of Technology of Mexico, Mexico & Technological Institute of Ciudad Madero, Mexico

Benito Sánchez-Lara
National Autonomous University of Mexico, Mexico

A volume in the Advances in Human Resources Management and Organizational Development (AHRMOD) Book Series

Published in the United States of America by
IGI Global
Business Science Reference (an imprint of IGI Global)
701 E. Chocolate Avenue
Hershey PA, USA 17033
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@igi-global.com
Web site: http://www.igi-global.com

Library of Congress Cataloging-in-Publication Data

Names: Ochoa Ortiz Zezzatti, Carlos Alberto, 1974- editor.
Title: Handbook of research on metaheuristics for order picking optimization in warehouses to smart cities / Alberto Ochoa Ortiz-Zezzatti [and three others], editors.
Description: Hershey, PA : Business Science Reference, [2019]
Identifiers: LCCN 2018047356| ISBN 9781522581314 (hardcover) | ISBN 9781522581321 (ebook)
Subjects: LCSH: Warehouses--Management. | Order picking systems. | Inventory control--Mathematical models. | Inventory control--Data processing. | Metaheuristics.
Classification: LCC HF5485 .H24 2019 | DDC 658.7/8--dc23 LC record available at https://lccn.loc.gov/2018047356

This book is published in the IGI Global book series Advances in Human Resources Management and Organizational Development (AHRMOD) (ISSN: 2327-3372; eISSN: 2327-3380)

British Cataloguing in Publication Data
A Cataloguing in Publication record for this book is available from the British Library.

All work contributed to this book is new, previously-unpublished material. The views expressed in this book are those of the authors, but not necessarily of the publisher.

For electronic access to this publication, please contact: eresources@igi-global.com.

Chapter 3
Intelligent Tool for Decision Making Associated With Hospitalization and Sandstorms for the Optimization of Ambulances

Estrella D. Molina-Herrera
University of Texas at El Paso, USA

Luis Ernesto Cervera-Gómez
El Colegio de Chihuahua, Mexico

Carlos Herrera
University of Texas at El Paso, USA & El Paso VA Health Care System, USA

ABSTRACT

The shortest path problem is a typical problem of optimization. This chapter presents an innovative model associated with the use of case-based reasoning to solve a problem of routing vehicles in a Hospital of El Paso, United States. In this chapter, diverse components are described to characterize this problem through the use of a knowledge system. The algorithm was developed in Java, thus obtaining a tool which determines the best tracks to the vehicles associated with ambulances. An experiment was realized to probe the validations; the results were used to compare it with the Dijkstra algorithm and determine the quality of the results. The future research of this intelligent tool is to determine an innovative perspective related to episodic knowledge applied to resolution of diverse ambulances, and as this topic is determinative to find and remember the best solutions quickly, additionally the authors compare it with a code from other postgraduate students trying to implement an algorithm similar to logistics but using a shuffled frog leap algorithm.

DOI: 10.4018/978-1-5225-8131-4.ch003

INTRODUCTION

Dust storms are recurring weather effects in a BW climate in the Köppen climate classification, where the city of El Paso, in Texas, is located and affects about one million people, in a database obtained by DSHS and associated with 387500 records compiled in different events of seven years. Each patient hospitalized in the main hospital of the city of El Paso is linked to at least 127 diseases associated with this type of weather events; it is crucial to know this for the current research project. Each of the scenarios that occur each time a dust storm happens, in order to adequately link a requirement of a set of ambulances for the population at risk during this climatological event, is recurrent during the minus seven months of the year.

Nowadays, digital maps are increasingly common to greatly improve the optimization of evacuations performed by emergency vehicles such as ambulances or fire trucks. With the progress that has been made in technology, these maps are becoming more sophisticated, in the way that they are able to find specific locations, draw routes and so forth (Rahaman, Mei, Hamilton & Salim, 2017). Another thing that is noteworthy is that they show how the information has improved dramatically, as they changed from traditional maps to maps with 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 nowadays, in order to achieve an improvement in the logistics after a huge dust storm which affects many people and determine what is the best way to organize the ambulances to move patients to diverse hospitals (van Barneveld, Bhulai & van der Mei, 2016). The objective of this work is to develop a system to help create routes on the basis of the emergencies given in El Paso, Texas, through the use of a system of neighborhoods of ants that allows them to create routes to take care of patients affected by a dust storm or other types of emergencies in a quick way. This is important because the life of the people is at risk. In the United States, to minimize the arrival time to the place of the accident, they do a comparison according to three possible emergencies at the same time and require other vehicles to respond to them. To provide assistance to citizens, paramedics in an ambulance need a route to arrive as quickly as possible to the place of the incident (Talarico, Meisel & Sörensen, 2015). If there are many emergencies, they are classified in order of importance: Hospitalization related to a dust storm, Rescue, and Prevention Action on public hazard. In all these activities, the time is vital because with a timely arrival the effect of the damage in a dust storm can be decreased to prevent an explosion in the leak case and find alive persons, among others. The bio-inspired algorithms are a technique of artificial intelligence focused on the solution of different problems, especially optimization problems. One of these algorithms is the swarm intelligence algorithm, where we can find the algorithm of the ant colony (ACO), particle swarm optimization (PSO), bees and so on (Şimşek and Kara, 2018). The proposed algorithm to solve the routing problem in El Paso city is an Ant Colony System.

DESCRIPTIONS OF THE MODEL COMPONENTS

In this section, we offer details of each component related to the application domain that is involved in the problem, in our case we solve a Logistics problem related to the El Paso Health System's Hospitals using a bioinspired algorithm to create routes of vehicles to attend emergencies.

The Shortest Route

The problem known as the shortest path or the shortest route, as its name suggests, tries 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 $V = \{V0, \mathrm{V}1, \ldots, \mathrm{Vn}\}$, and arcs are represented by $A = \left\{\left(vi,\ vj\right) \mid i = j\right\}$. The distances associated with each arc are represented by the variable Cij measured by the Euclidean distance. The objective functions of the problem are:

$$min\, Z = All\, the\, defined\, arcs\, C_{ij} X_{ij} \tag{1}$$

Decision variables are as follows:

- Xij : Action to move from node i to node j 0 indicates that there is no displacement and 1 indicates that yes there is movement.
- Cij : Cost or time to get from node i to node j.
- Restrictions for the Total input flow = total output flow (external input into node j) + i All the defined arcs $\left(i, j\right)$ Xij $=$ (external output from node j) + k All the defined arcs

$$\left(j, k\right) Xjk \tag{2}$$

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

The Shortest Path Algorithm

The shortest path algorithm, also called the Dijkstra algorithm, is an algorithm for determining the shortest path given in a source vertex to other vertices in a directed graph with weights on each edge. The shortest path algorithm belonging to the greedy algorithm (Johnsonbaugh, 2004) is an efficient algorithm of complexity $O\left(n^2\right)$, 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 the Dutchman Edsger Wybe Dijkstra in 1959 (Taha, 2008). The foundation in which this algorithm sits is the principle of optimality; the solution is built with the election of local optima in the hope of obtaining a global optimum

PROPOSAL METHODOLOGY

It is important to define that among a plethora of artificial intelligence techniques, the most appropriate one and the one that was selected for the present research was combined Case-based Reasoning and Ant Colony Optimization algorithm to improve a correct solution, which was validated with other similar investigations.

The internal structure of the RBC systems is composed of two parts: the obtaining of the case and the reasoning of the case. The first is responsible for finding the appropriate cases in the base of cases and the second is responsible for finding a solution to the problem given its description (see Figure 1).

The RBC fulfills a cycle from the beginning of the problem until the obtained solution; the cycle covers four parts: recover, reuse, review and retain (see Figure 2). This is also known as the 4 R's (Retrieve, Reuse, Revise and Retain). The parts that comprise the RBC cycle depend on the base case or case library since in this case the previous cases that contain valuable information are stored for the RBC to be successful. We must remember that cases are problems that have a solution, for that it is necessary to obtain a representation of the cases so that they are stored in the base of cases since not all the information that is available about the problem is important to solve the problem. For this reason, we add one more part to the RBC that is the representation of cases.

Case Representation: It is one of the most important parts of the RBC because the four parts of the RBC life cycle depend on it. Its importance lies in the fact that a case is a piece of knowledge that represents an experience and includes a problem, which is the description of the task to be solved and a solution, which corresponds to how the task was solved. In turn, a set of cases is called base case or case library. Usually, a case is represented as an attribute-value pair; this represents the problem and the solution of the case. For Behbahani, Saghaee and Noorossana (2012), in some cases the case contains a third element that is the result, that is, the state of the problem once the solution was applied. The experience can be represented in a different way; the classic one includes vector form, frame-based, object-oriented and textual, although there are already more sophisticated representations that are hierarchical cases, generalized cases, cases based on the based design, cases based on planning.

Recover Cases: The quality in the result of the RBC systems depends on the similarity measures used for the recovery of similar cases. The soft computing techniques used in this part of the RBC are:

Figure 1. Basic structure of an RBC system

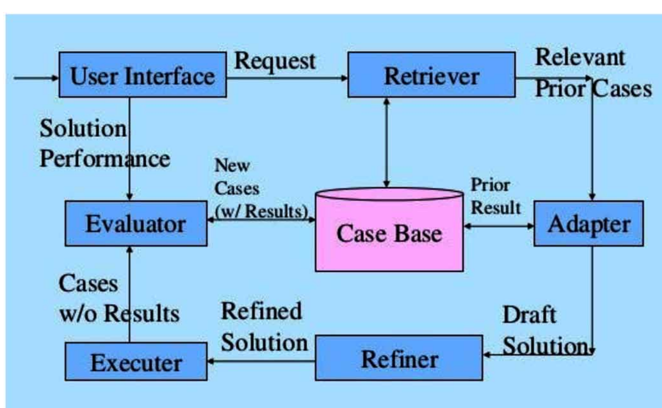
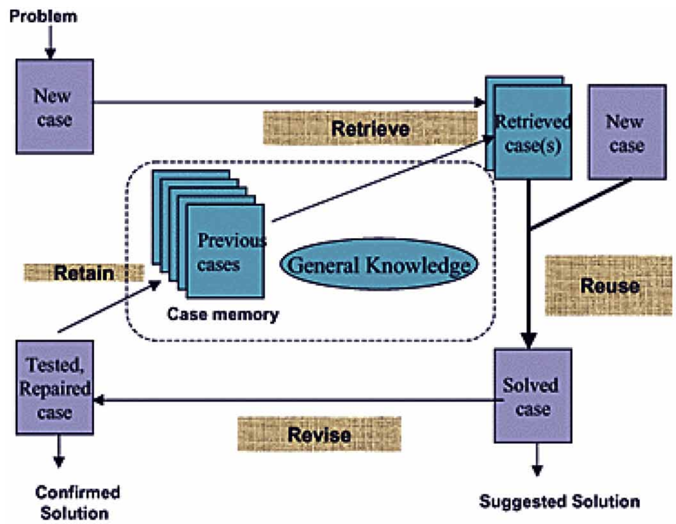


Figure 2. Life cycle of the RBC



diffuse indexing, diffuse grouping, case classification, probability, Bayesian models for the selection of cases, the nearest neighbor. In this phase, the current problem is checked against the problems stored in the base of cases. For Behbahani et al. (2012), the comparison is a process of comparing two cases among themselves and determining the Degree of Similarity (DOS). Besides the measures of similarity, knowing the domain helps determine the similarity of the new case with a previous case and having the degree of similarity lead us to a degree of adequacy of the solution of the problem or current case.

Reuse Cases: The reuse can be given by means of copying or integrating the solution of the cases that were recovered in the previous part. In reuse, interactive and conversational diffuse reasoning can be used, learning to reuse case knowledge and diffusional approaches. This part is also known as adapting the solution since the solution that was obtained in some occasions is necessary to adapt it to be given a solution to the case.

Case-Based Reasoning to Solve Route Problems

There are three forms of adaptation that are the most used: substitution, transformation, and generative adaptation.

Review Cases: The evaluation of the solution originated in the reuse of the case is carried out, this is usually carried out by domain experts. In case the solution needs some modification, this is done in this phase, and it is called repair. Must remember that the success or failure of the solutions originated is useful information to improve the RBC. The techniques used here are neural networks and evolutionary approach, rules of adaptation using set theories.

Retain Cases: Then the new case or problem and its solution are retained or stored in the base of cases for future use, that is to say that the solution was already confirmed or validated by domain experts. The decision whether the new case is stored in the case base also depends on how useful the knowledge of that case will be in the future. The techniques that can be used in this part are fuzzy rules, neural networks, set theory. This learning is incremental, thus we must bear in mind that the more cases stored in the database, the RBC will increase, as it will reach the time needed to maintain the database and continues fulfilling its function. Many authors described different applications to resolve this kind of problem. Lianxi Hong (2012) proposed an algorithm which permits to determine the time to reach a specific point in a city according to different possible scenarios. On the other hand, Lei, Laporte & Guo (2011) proposed a concept similar to ours, with relation to “emergencies”, which may occur in any time and place, and organize the demand to the vehicles, in our case is to the rescue units, try to minimize the effort to attend the demands in a day with the estimation of arrive, and solve another emergence, for example bee swarm which can be damage to children.

THE PROPOSED MODEL OF OUR HYBRID ALGORITHM

An intelligent tool was developed using Case-based Algorithm based on Ant Colony Algorithm and the programming language Java (J2SE) and as the first step we begin with the creation of the graph for the central area covering the Main Hospital in El Paso, a total of 2451 streets, avenues and boulevards (edges) and 1710 nodes. Subsequently, an entity was designed 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 this structure does not cause the overflow of memory related with the cells which compound the grid and generate a square incidence matrix, it stores only the necessary track which is visualized in their analysis.

The Ant Colony algorithm has been proved effective to solve NP-Hard problems when they use multidimensional arrays (Bell & McMullen, 2004). The structure of the generic algorithm is as follows (Dongo & Stützle, 2003):

The optimization quantity is the distance of the route. Thus, the truck movement cost between loading spots i and j is a function of all separate costs for each factor which affects the track route:

$$dij = \alpha daij + \beta dbij + \gamma dcij + \cdots \tag{3}$$

Let $tij\left(t\right)$ be the intensity of trail on edge (i, j) at time t. Each ant at time t chooses the next node, where it will be at time t + 1. Therefore, if we call an iteration of the ACO algorithm the n moves carried out by the n ants in the interval $(t,\ t+1)$, then for every n iterations of the algorithm, each ant has completed a tour. At this point the trail intensity is updated according to the following formula:

Algorithm 1. Optimization based on Ant Colony

```
Initialize parameters
while not stop condition
for ant = 1 to n construct solution
evaluate solution
update pheromones
end
end while
```

$$\tau ij\left(t+n\right)\rho\cdot\tau ij\left(t\right)+\Delta\tau ij \tag{4}$$

where ρ is a coefficient that represents the evaporation of trail between $\Delta\tau ij = m$.

$$k = 1\Delta\tau k\,ij \tag{5}$$

$$dij = adaij + \beta dbij + \gamma dcij + \ldots \tag{3}$$

The coefficient ρ must be set to a value < 1 to avoid unlimited accumulation of trail (see note 1).

In our experiments, we set the intensity of trail at time 0, tij (0), to a small positive constant c. In order to satisfy the constraint that an ant visits all the n different loading spots, we associate with each ant a data structure called the hlist, that saves loading spots already visited up to time t and forbids the ant to visit them again before n iterations (a tour) have been completed. When a tour is completed, the hlist is used to compute the ant's current solution (i.e., the movement cost of the path followed by the ant). The hlist is then emptied, and the ant is free to choose again,

$$\tau ij\left(t+n\right)\rho\tau ij\left(t\right)+\Delta\tau ij \tag{4}$$

$$k = 1\Delta\tau kij \tag{5}$$

$$\eta ij = \frac{1}{dij} \tag{6}$$

We call visibility hij the quantity $1 / dij$. This quantity is not modified during the run of the AS, as opposed to the trail, which instead changes according to the previous formula (4). We define the transition probability from loading spot i to loading spot j for the kth ant as

$$pk\,ij= \tau ij\left(t\right)\alpha\cdot\eta ij\beta\,k \in allowed\,k\tau ik\left(t\right)\alpha\cdot\eta ik\beta \tag{7}$$

The software implements the ability to block and alter the meaning of the streets, a fact that occurs in the central city of El Paso because of events, accidents, public works and so on. The method Initialize parameters enters the source node, the destination node, blocked streets and the number of ants involved in the search for the solution similar to the proposal in (Xiong, Wang & Yan, 2007). Construct solution takes place when ants move randomly with both probabilities using the Monte Carlo method if there is already a trail of pheromone. Once an ant has found the evaluate solution, 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 the pheromone if you have found a shorter route.

The user interface displays the found routes to the destination, with the option of display all of them or one in particular in a map (see Figure 3), which has the options of adding landmarks (churches, schools, hospitals, parks, rivers), zoom, viewing the different layers, storing in the route file, exporting the map as an image and sending it via Bluetooth to a mobile device.

EXPERIMENTAL RESULTS

The proposed algorithm was compared to the algorithm of Operations Research: The shortest path (Dijkstra).

The comparison was carried out with the generation of 20 runs starting from the central zone associated with El Paso Hospital (node 759) to different nodes (see Table 1).

Figure 3. Drawing of a single route (left) and drawing of five routes (right)

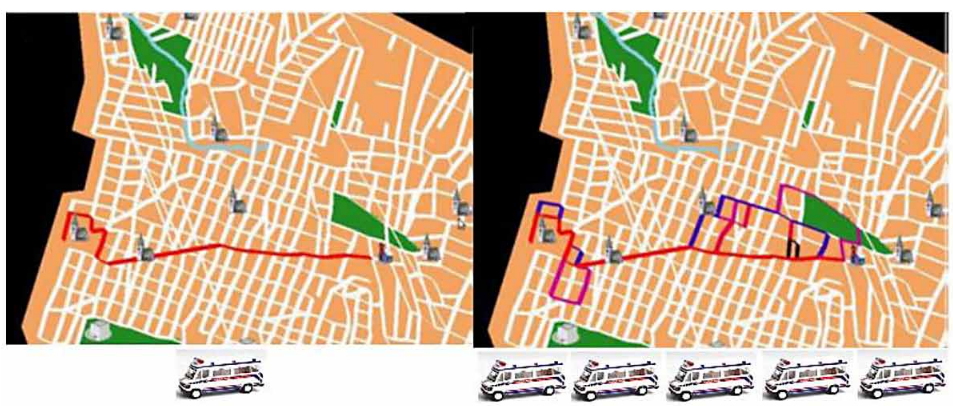


Figure 4. ACO and Case-based Reasoning and Dijkstra comparison

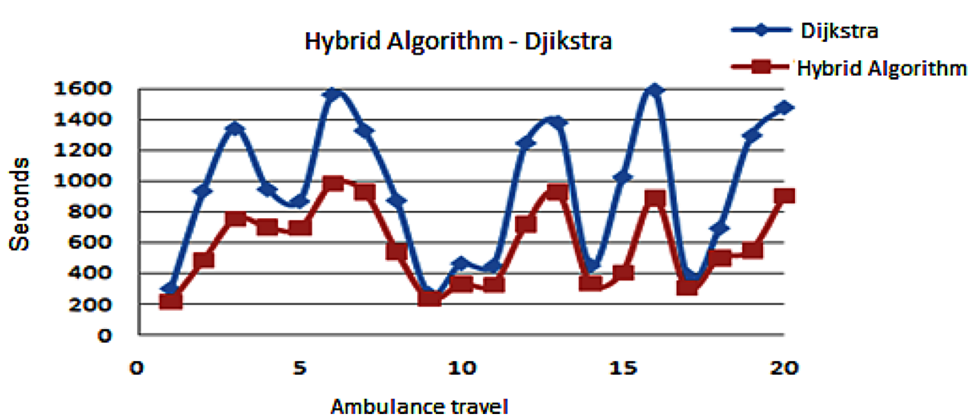


Table 1. Result for each ambulance travel from the houses of the patients to a Hospital in El Paso

#	Origin	Destination	Dijkstra	Hybrid Algorithm
1	945	759	222	212
2	614	759	464	507
3	903	759	755	841
4	941	759	732	698
5	1044	759	693	709
6	1202	759	984	1093
7	1094	759	953	927
8	1057	759	538	538
9	1418	759	231	231
10	170	759	338	328
11	526	759	347	324
12	462	759	718	718
13	846	759	859	1030
14	524	759	359	333
15	809	759	365	406
16	1107	759	886	1011
17	698	759	302	302
18	1062	759	517	499
19	1342	759	519	564
20	1199	759	885	984

The results were obtained with $\mu = 25.15$ seg and $\sigma = 15.65$ seg, in 35% of cases. While the Ant Colony gives 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 the shortest paths as in Figure 4. We select from a list of 1487 sickness related with the effects of a Dust Storm, two incidences and with a Kriging Model used in ArcGis, determine the expected value of patients that required an ambulance to travel to Hospital, as it is shown in Figure 5.

Another comparison was using the same instances and information obtained from Table 1 of twenty different ambulances from El Paso Health System's Hospitals, with the intention of building a robust design of experiments to try to understand the accumulative number of optimal solutions to reach the best track using the search space to three different codes of Hybrid Algorithm, ACO and Cultural Algorithm. The results will be observed in Figure 6.

CONCLUSION AND FUTURE RESEARCH

The algorithm currently implemented gives good quality solutions to an NP-hard problem, improving by 35% of cases the routes provided by the shortest path algorithm. The 45% where the shortest path algorithm exceeds the ACO which is attributed to not yet implementing the evaporation of the phero-

Figure 5. Incidence of a specific sickness in Neighborhoods of El Paso, and a Kriging Model of incidence of this sickness in the future with the necessity of more ambulances to transport more patients during a Dust storm

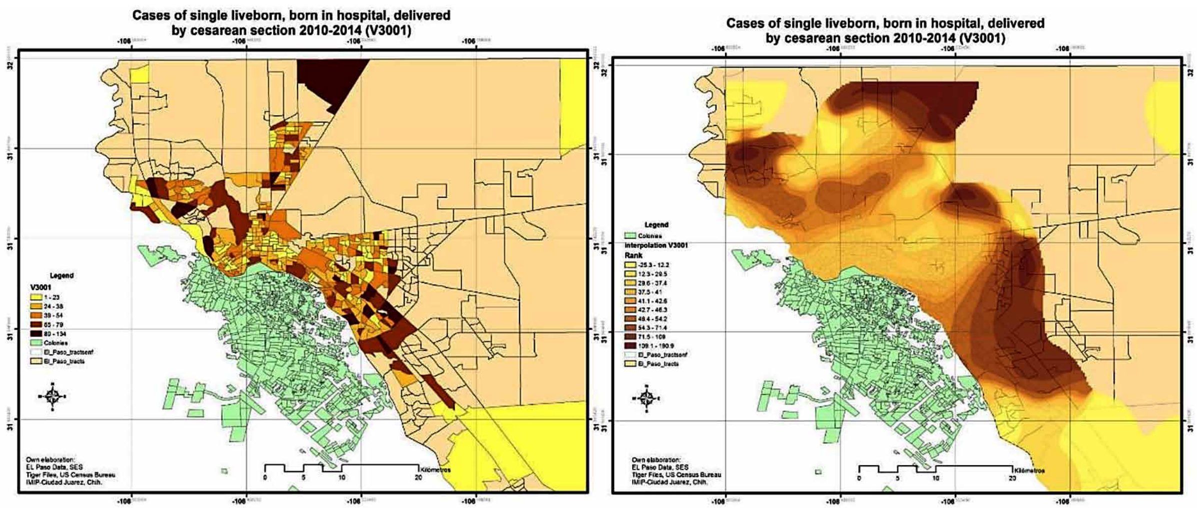
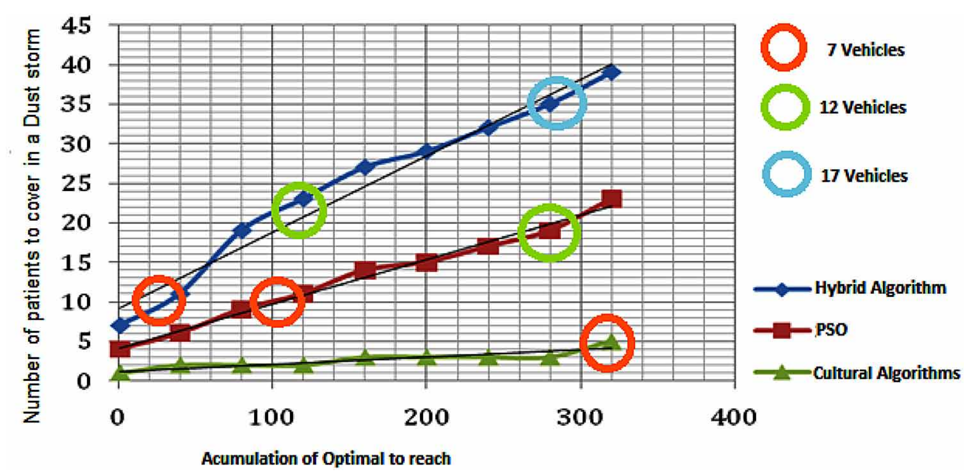


Figure 6. Comparative analysis of a Hybrid Algorithm, a PSO and a Cultural Algorithm for an instance of three emergencies associated with a Dust Storm at the same time, when we consider its performance based in Table 1



mone, the pheromone amount in nature may remain a few hours to several months depending on different aspects, such as ant species or soil type (Grimaldi, 1997), causing a minor influence on the effect of evaporation in the process of finding the shortest path. Due to the long persistence of the pheromone, it is difficult for the ants to "forget" a path that has a high level of pheromone but have found a path even shorter. Keep in mind that if this behavior is transferred to the computer to design a search algorithm sometimes, it can converge quickly to the local optimum. In this section, the results of the trial are presented. First of all, the data collection and the measurement of variables are described. Based on the

Figure 7. Using Bluetooth, it is possible to use our proposed model of ambulances in a huge contingency related to a Dust storm

results obtained, we recommend the implementation of heuristic algorithms such as ant colony, which have demonstrated to do well on a variety of problems (Birattari, Pellegrini & Dorigo, 2006; Dongo, Birattari & Stützle, 2006). As future work, it would be important 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 the system in units of El Paso Health System' Hospitals. Making an Intelligent Tool requires Access from any device including cellphones with different screen size and reorganizing the correct decisions in a mobile device as is shown in Figure 7.

We decide to make a comparison of our algorithm with relation to a PSO Algorithm and a Cultural Algorithm. We discovered the proposal of combining Case-based Reasoning and ACO Algorithm to obtain three different paths to a successful number of emergencies occurring at the same time, and that its performance improves by 22% the performance of PSO Algorithm and by 37% the performance of Cultural Algorithms. In the future research, it is important to describe the different times in other quadrants (the city is divided into four regions named Quadrants) of the city in the border zone which covers Hospitals and covers only the 48.7% of territorial space of the city.

REFERENCES

Behbahani, M., Saghaee, A., & Noorossana, R. (2012). A case-based reasoning system development for statistical process control: Case representation and retrieval. *Computers & Industrial Engineering*, *63*(4), 1107–1117. doi:10.1016/j.cie.2012.07.007

Bell, J. E., & McMullen, P. R. (2004). Ant colony optimization techniques for the vehicle routing problem. *Advanced Engineering Informatics*, *18*(1), 41–48. doi:10.1016/j.aei.2004.07.001

Birattari, M., Pellegrini, P., & Dorigo, M. (2006). On the Invariance of Ant System. In M. Dorigo, L. M. Gambardella, M. Birattari, A. Martinoli, R. Poli, & T. Stützle (Eds.), *Ant Colony Optimization and Swarm Intelligence*. Berlin, Germany: Springer. doi:10.1007/11839088_19

Dorigo, M., Birattari, M., & Stützle, T. (2006). Ant Colony Optimization: Artificial Ants as a Computational Intelligence Technique. *IEEE Computational Intelligence Magazine*, *1*(4), 28–39. doi:10.1109/MCI.2006.329691

Dorigo, M., & Stützle, T. (2003). The Ant Colony Optimization Metaheuristic: Algorithms, Applications, and Advances. In F. Glover & G. A. Kochenberger (Eds.), *Handbook of Metaheuristics, International Series in Operations Research & Management Science*. Boston, MA: Springer. doi:10.1007/0-306-48056-5_9

Grimaldi, R. P. (1997). *Matemáticas Discretas y Combinatoria*. Editorial Addison-Wesley.

Hong, L. (2012). An improved LNS algorithm for real-time vehicle routing problem with time windows. *Computers & Operations Research*, *39*(2), 151–163. doi:10.1016/j.cor.2011.03.006

Johnsonbaugh, R. (2004). *Matemáticas Discretas*. Pearson.

Lei, H., Laporte, G., & Guo, B. (2011). The capacitated vehicle routing problem with stochastic demands and time windows. *Computers & Operations Research*, *38*(12), 1775–1783. doi:10.1016/j.cor.2011.02.007

Rahaman, M. S., Mei, Y., Hamilton, M., & Salim, F. D. (2017). CAPRA: A contour-based accessible path routing algorithm. *Information Sciences*, *385-386*(April), 157–173. doi:10.1016/j.ins.2016.12.041

Şimşek, A., & Kara, R. (2018). Using swarm intelligence algorithms to detect influential individuals for influence maximization in social networks. *Expert Systems with Applications*, *114*(December), 224–236. doi:10.1016/j.eswa.2018.07.038

Taha, H. A. (2008). *Investigación de Operaciones*. Editorial Alfaomega.

Talarico, L., Meisel, F., & Sörensen, K. (2015). Ambulance routing for disaster response with patient groups. *Computers & Operations Research*, *56*(April), 120–133. doi:10.1016/j.cor.2014.11.006

van Barneveld, T. C., Bhulai, S., & van der Mei, R. D. (2016). The effect of ambulance relocations on performance of ambulance service providers. *European Journal of Operational Research*, *252*(1), 257–269. doi:10.1016/j.ejor.2015.12.022

Xiong, W.-Q., Wang, L., & Yan, C. (2007). Binary Ant Colony Evolutionary Algorithm. *ACTA Automatica Sinica, 33*(3).