Bat Algorithm to modeling distribution of Nomads Kitchen Vehicles in a largest city

Algoritmo bioinspirado basado en Murciélagos para el Modelado de Vehículos de venta de comida rápida en una ciudad de tamaño grande

Alberto Ochoa1, José Alberto Hernández 2, Miguel Basurto 3 & Julio Ponce 4
1 alberto.ochoa@uacj.mx, corresponding author, Universidad Autónoma de Ciudad Juárez, México
2 jose_hernandez@uaem.mx, FCAeI, Universidad Autónoma del Estado de Morelos, México
3 mbasurto@uaem.mx, CIICAp, Universidad Autónoma del Estado de Morelos, México
4 julk_cpg@hotmail.com, Universidad Autónoma de Aguascalientes, México

PALABRAS CLAVE:

Problema de enrutamiento de vehículos, Algoritmo del murciélago, dispositivos móviles.

RESUMEN

El objetivo de esta investigación es entender desde una perspectiva de optimización multivariable, la optimización relacionada con un grupo de camiones de alimentos en una ciudad grande, tratamos de analizar una serie de cuestiones específicas y determinar la ruta óptima involucrando velocidad, almacenamiento de alimentos, bienes perecederos y recursos para viajes como el petróleo y el gas para cocinar y determinar la relación costo-beneficio que se ha asociado con un plan de viajes diarios para vender alimentos especializados, que tiene como base principal la orografía y las restricciones de las distancias, aunque este problema ha sido estudiado en varias ocasiones por la literatura, no se ha podido establecer una solución mediante el apoyo de la computación ubicua para interactuar con los diferentes valores asociados con el logro del grupo de vehículos y su relación costo-beneficio de cada cuestión relacionada con la misma empresa y comparar sus viajes individuales asociados con la mejora financiera de la empresa. Hay varios factores que pueden influir en el logro de una compañía de camiones de alimentos. Para nuestra investigación nos proponemos utilizar el algoritmo del murciélago (Bat Algorithm), el cual ha demostrado ser eficaz para la convergencia de varios sujetos (murciélagos artificiales), cuando estos tienen tales restricciones y obstáculos deberían usar esta energía para evitar una perdida en recursos como alimentos, que en nuestro caso es representada como la utilización de los alimentos de manera óptima para la duración de un largo viaje, con la incertidumbre de no saber cuándo se tiene un reabastecimiento o es limitado al combustible por día.

KEYWORDS:

Vehicle Routing Problem, Bat Algorithm, Mobile Devices.

ABSTRACT

The purpose of this research is to understand from a Multivariable optimization perspective, the optimization related with a group of Food truck vehicles in a large city, we try to analyze a specific number of issues and determine the optimal route involve speed, food storage, perish goods and travel resources as oil and gas to cooking for determining the cost benefit have partnered with a daily travel plan to sell specialized food, which has as principal basis the orography and distances restrictions, although this problem has been studied on several occasions by the literature, failed to establish a solution by supporting ubiquitous computing for interacting with the various values associated with the achievement of the group of vehicles and their cost-benefit of each issue related with the same company and comparing their individual trips associated with the financial improve of the company. There are several factors that can influence in the achievement of a Food truck Company. For our research we propose use Algorithm Bat on, which has proven to be efficient for the convergence of several issues (artificial bats), when they have such restrictions and obstacles should use this energy to avoid a lose resource as food, which in our case is represented as the use of food optimally for the duration of a long travels, with the uncertainty of not knowing when having a resupply or is limited to fuel per day.

Recibido: 10 de junio de 2015 • Aceptado: 14 de octubre de 2015 • Publicado en línea: 29 de febrero de 2016

INTRODUCCIÓN

Celebrated for their contributions to gastronomy and champion of pet food porter, you have been selected to Barcelona to model various types of food and make a simulation as this business model and how long it should move to bring food quality as many diners in the shortest possible time and adapting to the needs of space and transfer between various points of the city analyzed, our Food truck is shown as in Figure 1.

Figure 1. A standard Food truck with its respective food preparation equipment

The most obvious use of the Food truck would be Perish Food and storage space to make appetizers. This is useful when moving around the city in our case Barcelona or even for shorter distances as they are as a storage model and can optimize the space associated with each element of the charge. These can be equipped with storage racks or cabinet space that will help you organize all the equipment and supplies needed for a day of rest and associated with a long distance journey. When buying a food truck in particular should have a list of what perish items to fit inside it, if this will be the main purpose of its use associated with the food preparation in short time. Furthermore, the space is small and the supplies have limits to keep. In addition it can support adapted furniture to clients, with the addition of some decoration materials. It is very useful for long-use when is on road, and saves the cost of reuse of supplies along the travel, making it one of the most popular uses. Routing problems vehicle (Vehicle Routing Problem - VRP) are actually a broad range of variants and customizations problems. From those that is simplest to some that remain today research as in [1].

They generally were trying to figure out the routes of a transportation fleet to service a customer. This type of problem belongs to the combinatorial optimization problems. In the scientific literature, Dantzig and Ramser were the first authors in 1959, when they studied the actual application in the distribution of gasoline for fuel stations. The objective function depends on the type and characteristics of the problem. The most common is to try: minimize the total cost of ownership, minimize total transportation time, minimize the total distance traveled, minimize waiting time, maximize profit, maximize customer service, and minimize the use of vehicles, balance of the resource utilization.

I PROJECT DEVELOPMENT

This research project was developed by dividing it into three sections which are modules of application development, implementation of the server and the intelligent module associated with Bat Algorithm and Data Mining. Android is the operating system that is growing in to Notes 4 from Samsung, for this reason we select this mobile dispositive along with other manufacturers are propelling the Latin American landing on Android with inexpensive equipment, and on the other hand, some complain about the fragmentation of the platform due to the different versions. Android is free software, so any developer can download the SDK (development kit) that contains your API [2]. This research tries to improve group travel related with Recreational Vehicles in Chihuahua where 7500 people conforms the Caravan Range Community.

COMPONENTS OF THE APPLICATION

BAT ALGORITHM

Bats are fascinating animals and their advanced capabilities of echolocation have attracted attention of researchers from different fields, we propose Artificial bat algorithms as in Figure 2. Echolocation works as a type of sonar: bats, mainly micro-bats, emit a loud and short pulse of sound, wait it hits into an object and, after a fraction of time, the echo returns back to their ears [4]. Thus, bats can compute how far they are from an object [5]. In addition, this amazing orientation mechanism makes bats being able to distinguish the difference between an obstacle and a prey, allowing them to hunt even in complete darkness [6].

Based on the behavior of the bats, Yang [3] has developed a new and interesting meta-heuristic optimization technique called Bat Algorithm. Such technique has been developed to behave as a band of bats tracking prey/foods using their capability of echolocation. In order to model this algorithm, Yang [3] has idealized some rules, as follows:

1) All bats use echolocation to sense distance, and they also "know" the difference between food/prey and background barriers in some magical way;

2) A bat bi fly randomly with velocity vi at position xi with a fixed frequency f/min, varying wavelength and loudness A0 to search for prey. They can automatically adjust the wavelength (or frequency) of their emitted pulses and adjust the rate of pulse emission r E [0, 1], depending on the proximity of their target;

3) Although the loudness can vary in many ways, Yang [3] assumes that the loudness varies from a large (positive) A0 to a minimum constant value *Amin.*

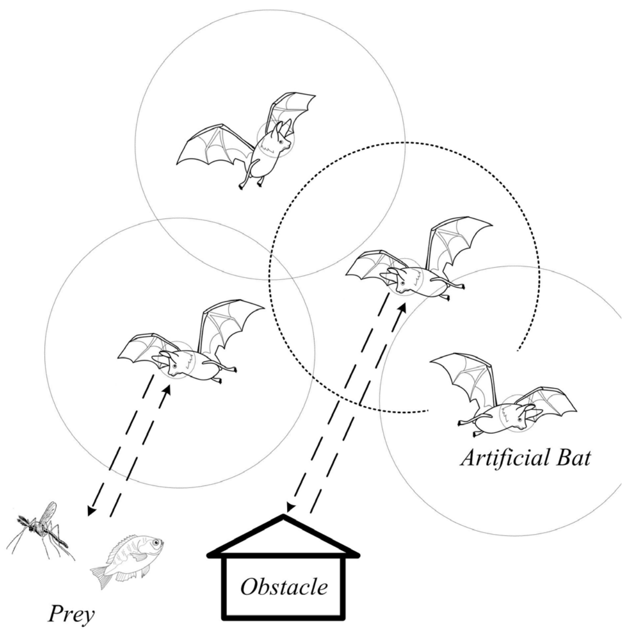

Figure 2. Representation of Artificial bats proposed in Bath Algorithm featured obstacles and food (prey) [11]

Algorithm 1 presents the Bat Algorithm (adapted from [3]):

Algorithm 1. – BAT ALGORITHM
Objective function f(x), x = (x1, ..., xn).
Initialize the bat population xi and vi, = 1, 2, ...,m.
Define pulse frequency fi at xi, □i = 1, 2, . . . , m.
Initialize pulse rates ri and the loudness Ai, i = 1, 2, . . , □.
1. While Tt< T
2. For each bat bi, do
3. Generate new solutions through Equations (1), (2) and (3).
4. If rand > ri, then
5. Select a solution among the best solutions.
6. Generate a local solution around the best solution.
7. If rand < Ai and f(xi) < f(ˆx), then
8. Accept the new solutions.
9. Increase ri and reduce Ai.
10. Rank the bats and find the current best ˆX.

Firstly, the initial position xi, velocity vi and frequency fi are initialized for each bat . For each time step t, being T the maximum number of iterations, the movement of the virtual bats is given by updating their velocity and position using Equations 1, 2 and 3, as follows:

$$fi = f\min + (f\min - f\max)\beta, (1)$$
$$vji\,(t) = vji\,(t-1) + [\hat{}xj - xji\,(t-1)]fi, (2)$$
$$xji\,(t) = xji\,(t-1) + vji\,(t), (3)$$

Where β denotes a randomly generated number within the interval [0, 1]. Recall that $xji\,(t)$ denotes the value of decision variable for bat at time step t. The result of fi (Equation 1) is used to control the pace and range of the movement of the bats. The variable $\hat{}xj$ represents the current global best location (solution) for decision variable j, which is achieved comparing all the solutions provided by the bats. In order to improve the variability of the possible solutions, Yang [3] has proposed to employ random walks. Primarily, one solution is selected among the current best solutions, and then the random walk is applied in order to generate a new solution for each bat that accepts the condition in Line 5 of Algorithm 1:

$$xnew = xold + \epsilon A(t), (4)$$

In which $A(t)$ stands for the average loudness of all the bats at time t, and $\epsilon \in [-1, 1]$ attempts to the direction and strength of the random walk. For each iteration of the algorithm, the loudness Ai and the emission pulse rate ri are updated, as follows:

$$Ai(t+1) = \alpha Ai(t)\ (5)\ \text{and}$$
$$ri(t+1) = ri(0)[1 - exp(-\gamma t)], (6)$$

Where α and γ are ad-hoc constants. At the first step of the algorithm, the emission rate $ri(0)$ and the loudness (0) are often randomly chosen. Generally, $Ai(0) \in [1, 2]$ and $ri(0) \in [0, 1]$ [3]. Bat Algorithm is very different from PSO Algorithm because specify better casual minor variations when is affected by exogenous events [3]. In addition we compare the results of another novel research as Wolf Search Algorithm [8] and ideas form a proposal of Spotted Salamanders Algorithm which try to understand the mimetically concept in addition of jump distances and support to the rest of its herd.

Bin Packing Problem

Bin packing problem is a complex problem NP (non-deterministic Polynomial), the problem is trying to fit in a container in our case the issue of vehicle fleet with the maximum number of objects of different value and volume, occupying much space as possible.

In this problem what is sought is to try to take the maximum amount of space that is available in the package or container and arrange everything in a better way so you can fit more and can optimize space as shown in the following Figure 3. As we can see in the above figure there is no space. All that space was well spent because the objects within the container were properly placed and located by the ranking of importance.

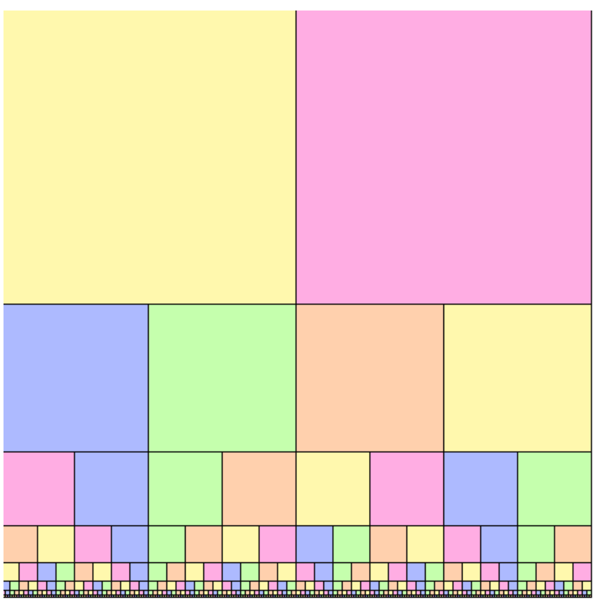

Figure 3. Representation of Bin Packing Problem to our research

This problem can be stated in different ways, depending on the algorithm that we have to solve. Could be viewed as a problem of definition in the following case:

Imagine you have several objects of different volumes and values. In order to obtain the maximum total value of items that could fit in the container need to ask if there are N objects in value (Preferably we must use a big one).

For example, we can ask if within the container could be obtained with a total value of 500. If the answer to this decision is yes, then we will decrease the number until it tells us that the value provided is not the greatest. After telling us that we should not increase the values related with its distribution on the space until the correct maximum value.

As for the optimization algorithm, following discarding objects alone is not essential or does not have much value and everything will settle down in an automated manner.

Justification of the use of Bin Packing Problem is associated with that this issue really struck me because it has a huge variety of applications and uses. It can be applied for example in dumpsters in the trailers they have to maintain a stable weight, to preserve the ecology and do less waste of resources and space that are normally used.

It is essential to know to manage space and do pay well. That is why this is considered as a very common problem in today's world that can be applied in many ways.

Complexity related with this problem is associated by the values to resolve and as we explained previously, this problem is NP Combinatorial (non-deterministic Polynomial) Complex.

We propose a novel algorithm to resolve this, in the next pseudocode; we show the description of an algorithm we found on different literature about this problem. We recognize that this is an interesting proposal because the algorithm has detailed information.

Algorithm 1. Proposed Algorithm

```
float[] used = new float[n + 1];
// used[j] is the amount of space in bin j already used up
int i, j;
Initialize all used entries to 0.0
Sort S into descending (nonincreasing) order, giving the sequence S1 >= S2 >= … >= Sn.
for(i = 1; i <= n; i++)
    //Look for a bin in which s[i] fits
    for(j = 1; j <=n ; j++)
        if (used[j]+si < +1.0)
            bin[i] = j;
            used[j] += si;
break; // exit for(j)
    //continue for(i)
```

It should be noted that to solve this problem, means of linear relaxation is required to solve about 420 linear programming problems for a solution could be optimal; without global optimal conditions but considering bigger problems the number of scheduling linear problems to solve is higher and the computational time required could be prohibitive. For this reason alternatives are sought, one alternative is the random search. This technique is relatively simple: generate a random number of reasonably feasible solutions to a given optimization problem and among these choose the best. These feasible solutions are typically generated by means of a probability distribution, usually uniform. Thus, no is necessary to explore all the possible space, but a sample. Of course, the higher the explored portion, the more likely find the optimal solution.

In this case, the idea is to generate so random matrices and solve distribution linear programming through the associated problem. To do this it was developed the algorithm which propose a solution as random search type together with the following functions:

- mdist (m, n, c) this function generates, through a biased roulette constructed from the vector d a matrix distributions. The objective of this is first store objects whose amount is larger. Here, c is the vector capabilities packages (c1 = 4 and c2 = 6).

- pl (W, A, d): This feature solves means the linear programming linear relaxation of the problem:

Close: w · x T

Subject to: $A \cdot w \geq xT\ dT$ to $1\ j \in \{0,0, ..., c,\}\ x\ T \in N$

If the solution obtained through linear relaxation is whole, then it is the optimal solution of the whole problem.

If the solution obtained through linear relaxation is not full, it is rounded up (to ensure that it satisfies the problem), although it is possible that none of rounding provide the optimal solution of the whole problem.

The algorithm we propose to solve the enhanced arrangement of objects, and proposed under the scheme of random search should find a solution to the problem, not necessarily optimal. Has as tickets: the number No object types, m the number of types of containers, the vector d of demands, the vector c capabilities, the cost vector w packaging and the number k of solutions. They will be generated randomly. The exit the algorithm is the tuple (x + A -) where x + is the local optimal solution found and To distribution matrix objects.

When executing the algorithm with data example and k = 50 It was obtained as follows solution:

A k March 2 = 1 2 0 2 x + = (10,34)

That is, it should be a total of 44 packages distributed as follows form: 10 packs of accommodating 7.

The algorithm that we propose and based on random search was presented in the previous section, is not deterministic in the sense that the run on the same instance of issue can produce different solutions. The algorithm presented has the characteristic that if it is for the same instance of the problem, it always produces the same solution, so it is a deterministic.

This algorithm is heuristic type and like the above solution found is not necessarily optimal.

The algorithm starts from the assumption (not always true) that it is necessary that they satisfy the demands to go on time to start building the distribution for each of the containers.

At example we presented, what you want is that some of the demands runs in the first container, and then seek all possible distributions for the other claims; It determined how it needed to meet for each product and then filled from the following container, applying the same idea recursively.

Eventually, it will get to the case in which you have to fill the last container, would be the basis of the recursion, which It can be solved optimally (as It was shown in [2]), using the algorithm shown below.

Using a single type of container associated with the vehicle fleet of Mail postal service of our research.

Suppose you have a single type container, the demands of different types of n objects d = (d1 D2 ... Dc) and the c capacity of the container such that n ≤ c. They will determine the matrix distributions:

A = (a11 = a21, ..., ac1) T

Where i1 represents the number of copies of the ith type objects stored in the only type of container and the number of repetitions x such that: is minimized: x(3) subject to:

$i1 = a11 + a21 + ... + c\ n\ 1 = i = 1$

$x\ \Sigma\ (4)\ i = 1,2, ...\ n$

$x \in N$

$ai1 \in \{0,1, ..., n\}\ i \in \{1,2, ...\ n\}$

In this case, the cost is no container as it is unique. Minimize x number of containers is to use same as minimize w · x for values w positives.
Suppose juices d = (97, 76, 68) T it would only put on sale in packages of 6, all the same. Post you need to have at least one juice each flavor in the container, a first possible solution is: i Copies Reps Wanted
Required
January 97 January 97
February 76 January 76
March 68 January 68
Thus, the minimum numbers of packages that allow exhaust all stocks is 97, since less the existence of the first package no juice it would be exhausted. As you can still put more juice in the package, it could add more juice of the first type: i Copies Reps Wanted required
January 97 February 49
February 76 January 76
March 68 January 68

Thus, 97 are no longer needed packages (Since having two juices type 1, up stocks required just 49 packets), but 76 needed up stocks of the second type of juice. Thus, repeated the procedure until the container it is full. These values are represented as:

Matrix Mnx (C – n + 1) = mij [ ] Containing the complete example.

(1.97 +) (2.49) (2.49) (2.49 +) (1.76) (1.76 +) (2.38) (2.38) (1.68) (1.68) (1.68 +) (2. 3. 4)

Optimal distribution of a single type container seating is 6.

The solution for this example,
d = (97, 76, 68)
T and c = 6 is x + = 49 and
A = (2, 2, 2)
T, as shown in the last column in our values.

In each column a solution shown is feasible, each constructed from the above solution except the first column, which corresponds to place a juice each type. The first term of tuple corresponds to juice and the second term the number of all packages required to that the system use the juices of the respective type. The demand which requires larger number of packages marked with an asterisk, and in the following Step juice that type is added.

The idea is to solve the problem by means the minimum required number of repetitions, corresponds container the ceiling of the greater proportion of the demand required ith object d1 and the number of copies the vessel ith object to I 1 , namely, max {[dj / aj 1] / i = 1,2, ..., n }.

Therefore, the first step is to place a copy of all objects in the container (this is possible thanks to that c ≥ n, then the object is selected that defines the minimum number of repetitions of the container to meet all claims and a copy is added of that object in our container, repeating the procedure until it is full.

The algorithm that we evaluated only those associated with a container for objects and make better accommodations to each object related with this problem optimally for the case one type of container. Taken as entry number n of different types of objects, container capacity c and demands objects d = (d1 D2 ... Dc) T, such that c ≥ n. The output of the algorithm is tuple (x + , TO -), where x + is the optimum solution to the problem and A- determines the number copies of each object in the container.

II IMPLEMENTATION OF AN INTELIGENT APPLICATION

When designing an interface for mobile devices has to take into account that the space is very small screen, plus there are many resolutions and screen sizes so it is necessary to design an interface that suits most devices. This module explains how to work with different layout provided by the Android API.

The programming interface is through XML. Obtaining the geographical position of a device can be made by different suppliers; the most commonly used in this project through GPS and using access points (Wi-Fi) nearby, and perform the same action but differ in some as accuracy, speed and resource consumption. Data Server Communication is the module most important because it allows communication with the server, allowing you to send the GPS position obtained by receiving the processed image and map of our location, thus showing the outcome of your application that is the indicator of insecurity.

To communicate to a server requires a HTTP client which can send parameters and to establish a connection using TCP / IP, client for HTTP , can access to any server or service as this is able to get response from the server and interpreted by a stream of data.

The Android SDK has two classes with which we can achieve this, HttpClient and HttpPost. With the class HttpClient is done to connect to a remote server, it needs HttpPost class will have the URI or URL of the remote server. This method receives an URL as a parameter and using classes HttpPost HttpClient and the result is obtained and received from the server, in this specific case is only text, which can be JSON or XML format. Here, the server responds with a JSON object which would give the indicator is then used to create the map, as is shown in the Figure 4.

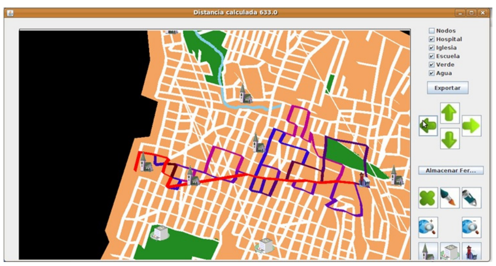

Figure 4. Intelligent Tool recommends a food truck associated with limited resources and optimize energy (oil), time and food sell.

In Figure 4 is shown the dish associated with a specific time and the price in this case is 5 Euros, this try to improve the real possibility of selling all perish food.

For the construction of the polygon that indicates the rate of incidents in a certain radius of the current position is not possible to create it using the GPS coordinates that yields, as these are specified in "degrees" and requires the unit to convert to meters. For this purpose is necessary to know how an arc equals the terrestrial sphere, which depends on the place on earth where it is located and the address where you are, the simplest case is to measure an arc in Equator, considering that the earth is 3670 km radius, the perimeter of serious Equator radio 2, which would be equal to 40.024 miles. With this you can get a relationship that would be as follows. If 360 degrees is 40.024 miles then a degree is 111,000.18 miles, this relationship can add and subtract yards to the position, as shown in the figure 4, this process related a specific point of interest in our case organize a group of locations to sell specialized food during a specific time (8:00 am to 7:00 pm in different parts of the city, the information is displayed on the mobile device that recommends inclusively number of people waiting for food, including variations according weather and waiting time to eat.

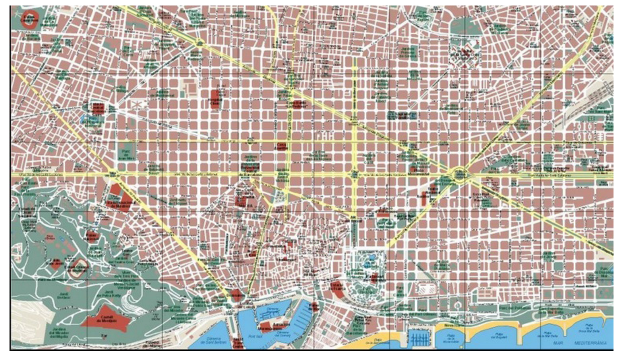

Figure 5. Acquisition of a graph map with the position of the Android application

For the preparation of graphics, we propose using a class supported with Bat Algorithm supported which facilitates to manipulation of data to express visually using different types of graphs, as in the figure 5 where for proposed each iteration is related with a generation. To the 6th Iteration a specific behavior is described to the most possible places where is possible advance the most of vehicles associated with our model of Food Truck Company.

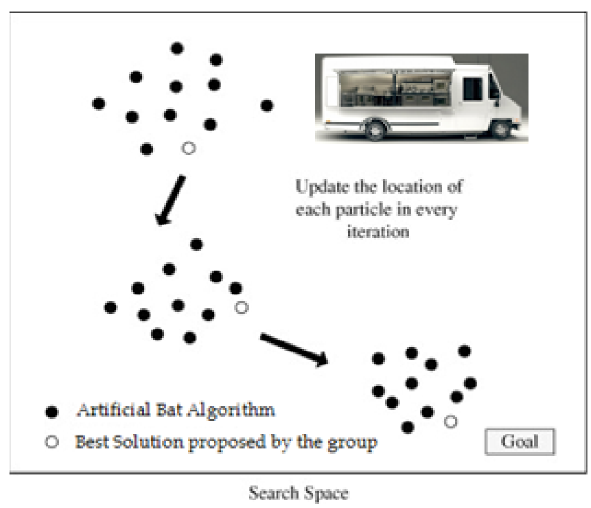

Figure 6. Graphical Model associated with the solution proposed by Bat Algorithm.

To implement the application is necessary an operating system devices with Android 2.2 or higher, which tests the system in different areas of four different quadrants on the city based on the previously research related with Cultural Algorithms on Urban Transport [7], by answering a questionnaire of seven questions to all users related with the Food truck company have elapsed since installing the application, the questions are to raise awareness of the performance, functionality and usability of the system, the demonstrate use of this application is shown in figure 6. To understand in an adequate way the functionality of this Intelligent Tool, we proposed evaluate our hybrid approach and compare with only data mining analysis and random selected activities to protect in the city, we analyze this information based on the unit named "époques" used in Bat Algorithm, which is a variable time to determine if exist a change in the proposed solution according at different situation of different routes with better use of restricted resources.

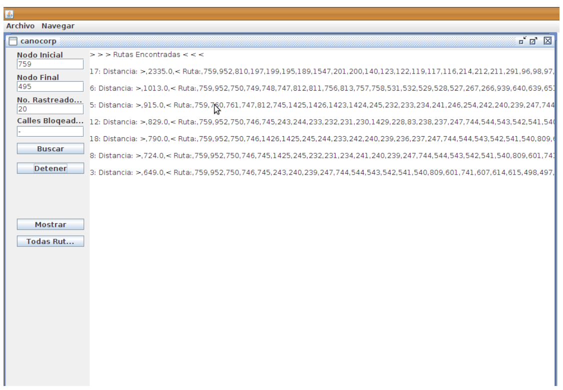

Figure 7. Hybrid Intelligent Application based on Bat Algorithm and Data Mining.

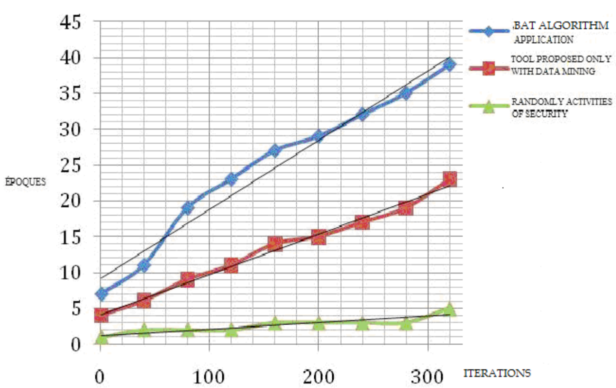

Figure 8. Proposed solutions to group travel problem: (blue) hybrid approach; (red) only using data mining analysis and (green) using randomly actions to improve the safety of the users.

We consider different scenarios to analyze during various times, as is possible see in the Figure 7, and apply a questionnaire to a sample of users to decide to search a specific number of places for waiting the issue of food truck with the food in a short time, when the users receive information of another past food routes (Data Mining Analysis) try to improve their space of solution but when we send a solutions amalgam, our proposal with a Bat Algorithm and Data Mining was possible determine solutions to improve the resources of the group (see Figure 8), the use of our proposal solution improves in 73.47% against a randomly action and 26.82% against use Data Mining analysis only , the possibilities of recommending to arrive in a specific time by using less energy, these messages permit in the future decrease the possibility of deplete supply of perish food and spend time rerouted in an orography terrain with more traffic jam, and uncertainty of the weather conditions to organize the food furniture and the use of limited resources.

III CONCLUSIONS

With the use of this innovative application that combines Bat Algorithm and Data Mining based on a mobile dispositive is possible determine correctly locations where is possible sell specialized food in a less time and with the optimization of resources, this research including the use with a mobile device with GPS, providing statistical information through as a Web server that returns the level of insecurity in the consulted area [8], a specific factor decisive in this part of Mexico. The future research will be to improve the visual representation of group travel to a social networking, to do this we proposed an Intelligent Dyoram –An Intelligent display in 3D used on Social Networking- with real on time information for each one of their integrants and establishment a travel, including saving money and time and enjoy with the rest of group of consumers of similar food. The most important contribution is to prevent spending time and money in this kind of food routes, because waiting in an unsafe place when arrive the food truck, on a wrong time is possible to suffer an assault. Our future research is adequate the information to actualize from the central server of security of the City (Mossos da squadra), to the users, considering that the number of people try to obtain specialized food is higher at least 78000 people at day with a short time to eat, because the specialized food truck restaurants is very high as economic activity, this innovative application is possible to use in another cities in Latin America specific in Sao Paulo –This society has 975,000 people trying to eat in this kind of Food truck Restaurants and realize an accumulate of 27.975,000 food sessions during one year- with similar conditions of time and cost by their respective Model of Food truck, this Intelligent Tool will be used by different kind of people whom requires eating together.

In addition, this application will be used as Recommender System when traveling to another cities or places in different societies to eat in this kind of services [9] and explain different scenarios according time, limited resources and location. Another field topic will be benefited with more adequate organization, is Logistics of products or services as in [10] which describes the use of Cultural Algorithms to improve a Logistics networking associated with the deliveries of a bottle product.

ACKNOWLEDGMENT

The authors were supported with dataset information from Barcelona Supercomputing Center and used information from a specific company of Food truck in Barcelona, which permits comparing the simulation with real travels realized by them. The Dell mobile device was bought with funds from a Promep project supported by SEP.

REFERENCES

1. Barbucha, D. Experimental Study of the Population Parameters Settings in Cooperative Multi-agent System Solving Instances of the VRP. In: T. Computational Collective intelligence. 2013, 9, 1-28.
2. Andreu R. Estudio del desarrollo de aplicaciones RA para Android, Trabajo de fin de Carrera. Catalunya, España, 2011.
3. Yang, X.S. Bat algorithm for multi-objective optimisation, International Journal of Bio-Inspired Computation. 2011, 3(5), 267–274,
4. Griffin, D. R., Webster, F. A., Michael, C. R. The echolocation of flying insects by bats, Animal Behaviour. 1960, 8(34), 141 – 154.
5. Metzner, W. Echolocation behaviour in bats. Science Progress Edinburgh. 1991, 75(298), 453–465.
6. Schnitzler, H.U., Kalko, E. K.V. Echolocation by insect-eating bats, BioScience. 2001, 51(7) , 557–569.
7. Cruz, L., Ochoa C.A., Gómez, C., Hernández, P., Villa Fuerte, M. A Cultural Algorithm for the Urban Public Transportation. In: 5th International Conference, HAIS 2010. San Sebastián, Spain, 2010, 135-142, doi: 10.1007/978-3-642-13803-4_17
8. Glass, S., Muthukkumarasamy V., Portmann, M. The Insecurity of Time-of-Arrival Distance-Ranging. In: IEEE 802.11 Wireless Networks. ICDS Workshops 2010, 227-233.
9. Souffriau, W., Maervoet, J., Vansteenwegen, P.; Vanden Berghe, G., Van Oudheusden, D. A Mobile Tourist Decision Support System for Small Footprint Devices. In: 10th International Work-Conference on Artificial Neural Networks, IWANN 2009. Salamanca, Spain, Part I IWANN (1) , 2009, 1248-1255.
10. Ochoa, A. Garcí, Y. Yañez, J. Logistics Optimization Service Improved with Artificial Intelligence. In: Soft Computing for Intelligent Control and Mobile Robotics, Volume 318 of the series Studies in Computational Intelligence, 2011, 57-65, doi: 10.1007/978-3-642-15534-5_4
11. Ochoa, A., Margain, L., Arreola, J., De Luna, A. Improved solution based on Bat Algorithm to Vehicle Routing Problem in a Caravan Range Community. In: Hybrid Intelligent Systems (HIS), 2013 13th International Conference on, 2013, pp. 18-22, doi: 10.1109/HIS.2013.6920479.

Carlos Alberto Ochoa Ortiz (Bs 1994 – Eng. Master 2000, PhD 2004 at CICATA at Instituto Politécnico Nacional, México, Postdoctoral Researcher 2006 at University of Campinas (UNICAMP), Brazil and Industrial Postdoctoral Research 2008 at Centro de Innovación Aplicada en Tecnologías Competitivas (CIATEC - CONACYT). He has participated in the organization of different international congress like HAIS, HIS, ENC, and MICAI. His research interests include Evolutionary Computation, Natural Processing Language and Social Data Mining; he is part time professor at the Social Science department at Juarez City University. He is member in the National System of Researches Level 1 in Mexico (SNI).

José Alberto Hernández Aguilar. He finished his Doctorate thesis in 2007 at Universidad Autónoma del Estado de Morelos (UAEM) and received his PHD degree in 2008. He obtained a Master of Business Administration degree, Cuma Sum Laude in 2003 at Universidad de las Americas (UDLA), A.C. He has a B.S. in Computers Engineering at Universidad Nacional Autónoma de México. Since 2010, he is full time professor at the accounting, management and computer science School at Universidad Autónoma del Estado de Morelos. His areas of interest are: Databases, Artificial Intelligence, Online Assessment Systems, Data Mining and Marketing Research. He has participated in the organization of different international congress like HIS and MICAI.

Miguel Basurto. He was born in Mexico in 1973. He receivd the B.S. degree in Electronics and Industrial Engineering from InstitutoTecnológico de Veracruz in 1995, and a Master on Sciences from Instituto Nacional de Astrofísica, Óptica y Electrónica in 1997 and finally a PhD in 2001 in the same institution. He is a full professor on CIICAp-UAEM Center, his research includes Lasers Fibers, Optical Fibers Sensors and Artificial Intelligence applied to Optics.

Julio César Ponce Gallegos. He received the B.S. degree in computer system engineering from the Universidad Autónoma de Aguascalientes in 2003. He received the M.S. degree in computer sciences from the Universidad Autónoma de Aguascalientes in 2007, and the PhD. Degree in computer sciences from the Universidad Autónoma de Aguascalientes in 2010. He is currently full time professor in the Universidad Autónoma de Aguascalientes. He is author and coauthor of several international papers and book chapters in Computer Sciences related fields. His research interests include Evolutionary Computation, Data Mining, Software Engineering and Learning Objects. He has participated in the organization of different international congress like HIS and MICAI.