© International Journal of Combinatorial Optimization Problems and Informatics, Vol. 8, No. 1, Jan-April 2017, pp. 39-44. ISSN: 2007-1558

A hybrid algorithm between the ant system and the harmonic search for solving the vehicle routing problem with time windows (VRP-TW)

Edwin Montes-Orozco[1], Roman Anselmo Mora-Gutiérrez[2], Javier Ramírez-Rodríguez[3]

[1] *Posgrado en Optimización, Universidad Autónoma Metropolitana-Azcapotzalco*
[2,3] *Departamento de Sistemas, Universidad Autónoma Metropolitana-Azcapotzalco*
[1]*edwinmorozco@gmail.com*,[2]*mgra@azc.uam.mx*, [3]*jararo@azc.uam.mx*

Abstract. In this paper, we present a hybrid method between the Ant System (*AS*) and the Harmonic Search (*HS*), which was used to solve vehicle routing problem with time-windows (VRP-TW). This method has been called AS-HS. In this sense, both metaheuristics are intertwined. The AS technique guides the behavior through changes in the pheromone matrix, and takes advantage of information from a number of HS executions stored in the harmonic memory (HM). In the proposed procedure, the best solutions are taken to update the pheromone level. This allows the ants to intensify in a promising region and the elements of diversity in the techniques avoid the premature convergence of the algorithm.

On the other hand, for a more efficient construction of the solutions, we took advantage of the structure of the problem that used the time window, the distance between the customers to visit and the load assigned to each vehicle as factors within the update of the pheromone level.

1. Introduction

1.1 Vehicle routing problems

The vehicle routing problem (VRP) belongs to the NP-hard class [5] and it is described as a central depot that has a fleet of vehicles and a set of geographically distributed customers that must be serviced. The VRP has as objective to minimize the cost of delivery of the goods requested by a set of customers, creating routes that originate and end up in the depot. Each customer is served only once by one vehicle, and all customers must be served, so each vehicle is assigned a set of customers whose demand does not exceed their capacity.

One way to extend the traditional VRP is to add the restriction of associate a time window for each customer, thus defining an interval or schedule for each customer to be serviced, resulting in the vehicle routing problem with time windows (VRP- TW) [3].

2. Mathematical Model

The mathematical model for the VRP-TW is presented in this section in which is used the following notation:

$G = (Cl, A)$ non-directed full graph; $Cl = \{cl_0, cl_1, cl_2, \dots, cl_n\}$, customers set, where cl_0 is the depot; $A = \{(i,j): i,j \in Cl, i \neq j\}$ the edge set; $C = \{(C_{ij})\}$ the travel costs between nodes i and j; $d_i \in Z^+$, demand of node i; $Q \in Z^+$, capacity of each vehicle. For this, the decision variables considered are:

$$X_{ij}^{v} = \begin{cases} 1, & \text{if the vehicle } v \text{ travel of client } i \text{ to client } j \\ 0, & \text{in other case} \end{cases}$$

And the auxiliary variables Y_{iv} represent quantity delivered by the vehicle v to the customer i; b_{iv} is arrival time for vehicle v to customer i. In this variant of the problem, in addition to capabilities, each customer $Cl_i \in Cl$ has associated a time window $[e_i, l_i]$ that establishes a service schedule allowed so that a vehicle arrives at this and a time of service s_i. The mathematical model for the *VRP-TW* is:

Received Jun 19, 2016 / Accepted Dec 10, 2016

$$Minimize \sum_{i=0}^{n}\sum_{j=0}^{n}\sum_{v=0}^{k} C_{ij}X_{ij}^{v} \quad (1)$$

Subject to:

$$\sum_{i=0}^{n}\sum_{v=0}^{k} X_{ij}^{v} = 1;\ \forall\, j = 0,1,\dots,n \quad (2)$$

$$\sum_{i=0}^{n} X_{ip}^{v} - \sum_{j=0}^{n} X_{pj}^{v} = 0;\ \forall\, p = 0,\dots,n;\, v = 1,\dots,k \quad (3)$$

$$Y_{iv} = d_i \sum_{j=0}^{n} X_{ij}^{v};\ \forall\, i = 1,\dots,n;\, v = 1,\dots,k \quad (4)$$

$$\sum_{i=0}^{n} Y_{iv} \le Q;\ \ \forall\, v = 1,\dots,k \quad (5)$$

$$e_i \le b_i \le l_i, \forall\, i,j = 0,\dots,n;\, v = 1,\dots,k \quad (6)$$

$$X_{ij}^{v}\left(b_i^{v} + t_{ij} + s_i - b_j^{v}\right) \le 0;\ \forall\, i,j = 0,\dots,n;\, v = 1,\dots,k \quad (7)$$

$$X_{ij}^{v} \in \{0,1\}, Y_{iv} \ge 0;\ \forall\, i = 0,1,\dots,n, v = 1,\dots,k \quad (8)$$

The constraint (2) assures that a vehicle must visit each customer. The constraint (3) indicates that vehicles departing from the depot must return to this. The constraint (4) implies that the customer *i* can be served by the vehicle only if *v* takes the edge that leads to the customer *i*. The constraint (5) implies that the quantity delivered in each route does not exceed the capacity of the vehicle. The constraint (6) assure that the boundaries of time windows are imposed. The constraint (7) ensures that the vehicle *v* cannot start the service, if the sum of the transport time between the *customers i and* j, the duration of service for customer *i* and the time of arrival to *i-th* customer is greater than the time window for the customer *j*. Finally, constraint (8) defines the type of variables used.

3. Description of the used techniques

3.1. Ant System

Ant System (*AS*) is a metaheuristic, which is inspired on the phenomenon of communication in ants through pheromones. Where each ant is able to find paths from its nest to the food sources. Dorigo *et al.*, proposed the *AS* in 1996 [4].

In this paper, a variant of the *AS* is presented. In this *AS* variant the probability election is adapted by means of introducing into its equation an element, in which is considered to the limits of the time window for each customer. Then, a brief description of the *AS* is given.

3.1.1. Pheromones initialization.

For the first cycle, the level of pheromone is initialized as shown in equation (9):

$$\tau_{ij}^{a+1} = \frac{1}{l_j}, with\ a = 0 \quad (9)$$

Where, τ_{ij}^{1} is the level of pheromone that has the path between customer *i* and customer *j* for cycle 1, and l_j the last moment that customer *j* can be served.

3.1.2. Update of Pheromones.

For the other cycles, the level of pheromone matrix is updated with the following equation:

$$\tau_{ij}^{a+1} = \rho\left(\tau_{ij}^{a}\right) + \Delta_{ij} \tag{10}$$

Where ρ is the pheromone evaporation coefficient, τ_{ij}^{a} is the level of pheromone that has the path between customer i and customer *j* for cycle a, τ_{ij}^{a+1} is the level of the pheromone that will guide the next cycle of the *n* ants and Δ_{ij} is the quantity of pheromones deposited by the best ant of the cycle:

$$\Delta_{ij} = sol[a] - sol[a-1] \tag{11}$$

Being sol $[a]$ the value of the solution delivered by the best ant of the cycle a; and sol$[a-1]$ the value of the best solution in the cycle a-1.

3.1.3. Probability of election

Each time that a cycle is completed for *n* ants or a customer is added to the solution, the probability of choosing the path between *i,j* customer is updated by the following equation [4]:

$$P_{ij}^{a+1} = \frac{(\tau_{ij}^{a+1})^{\alpha} + (\eta_{ij}^{a+1})^{\beta} + (\nu_{ij}^{a+1})^{\gamma}}{\sum(\tau_{ij}^{a+1})^{\alpha} + (\eta_{ij}^{a+1})^{\beta} + (\nu_{ij}^{a+1})^{\gamma}} \tag{12}$$

Where, τ_{ij} is the pheromone level, $\eta_{ij} = 1/C_{ij}$ regulates the convenience of the transition state, $\nu_{ij} = 1/l_{ij}$ denotes that the lower the value of the last moment in which it can be visited, the probability to visit it at the beginning will be greater. In addition α is used to control the influence of the pheromone level, β is a parameter to control the influence of closeness between customers and, γ controls the influence of the l_{ij} value.

3.1.4. Construction of the Solution

Every ant of the system constructs a solution departing from the depot, adding to its route a set of customers based on the probability of choosing each one (equation 11), and finally a solution will be represented with the vector (see Figure 1)

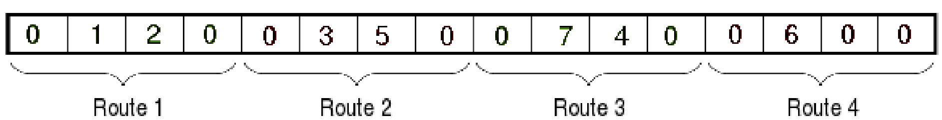

Figure 1. Representation of the solution.

3.2 Harmonic Search

Harmonic Search (*HS*) is a metaheuristic, which is inspired by the way in which a musician searches the best harmony in musical composition into Jazz, this technique was presented by Geem in 2001 [6].

The *HS* has been used to solve complex optimization problems in a successful way. Parameters involved in *HS* are: a) the size of the harmonic memory (*HMS*) which defines the number of elements that are stored, b) the scanning ratio (*HMCR*), c) the pitch adjustment ratio (*PAR*), d) the displacement width (*BW*) and e) the number of improvisations (*IN*) [7]. Then, a description of the *HS* is given.

3.2.1. Initialization of the Harmonic Memory

Initially, a set of *HMS* solutions is randomly built. However, for this work, the list of *HMS* solutions is filled with *n* solutions generated by *AS*, where *n* = *HMS*. This memory is stored in a list, containing the x_i solutions, with $i = 1, \ldots, HMS$.

3.2.2. Improvisation of new harmonies

At this stage, a new solution x_s is improvised, based on memory examination, random re-initialization, and pitch adjustment. For the step of examining memory, it generates a random number r_1 in [0,1]. If r_1 is less than *HMCR* the route *i* for the new improvisation is chosen from any of the existing solutions in *HM*. Otherwise, $x_s(i)$ is obtained from a re-initialization, where the route is built using the probabilities of *AS*. Once the random re-initialization or memory examination is carried out, the solution x_s is modified by the tone adjust depending on the *BW* value, whereby it will be disturbed in two different neighborhoods with one of two disturbances.

The first, it makes the exchange of two customers of the chosen route of random form. The second is considered as one of the best perturbations for the VRP problems because it performs the exchange of customers in a same random position between all routes. Note that the *BW* during the course of the execution will take values that will allow disturbing in the two neighborhoods described.

3.2.3. Update the harmonic memory

After that a new harmony x_s is created, the *HM* memory is updated by comparing the objective function values between x_s and the worst solution x_w contained in the *HM* memory. If the objective function value of x_s is better than x_w, x_s will replace it, otherwise the *HM* content remains unchanged.

3.3 Hybrid Algorithm between Ant System and Harmonic Search (AS-HS)

The metaheuristics described in subsections 3.1 and 3.2 (AS and HS) were used interlaced. Thus, it is possible to guide the construction of each ant solutions, taking advantage of the best melodies generated by HS, taking the solutions of the HM list that are below the average value of HM to update the pheromone for the AS. Thus, it will not intensify over the region of the best melody, but will diversify the search space by taking solutions that are dispersed. The general pseudocode of the AS-HS hybrid is shown in the Algorithm 1.

Algorithm 1. Algorithm for the AS-HS

```
INPUT:        A VRP-TW instance
OUTPUT:       The best solution found
Initialize pheromone matrix
Update the probability of election for the 1st cycle of HMS ants
for a=1 to a=m do
    if a==1 then
        counter=0
        for l=1 to l=HMS do
            Build solution for the ant X_l and evaluate the O.F.
        end for
        Sort ascending the HMS solutions by value of O.F.
    else
        for l=counter to l=HMS do
            Build solution of ant X_l and evaluate the O.F.
        end for
        counter==0
    end if
    while n_imp!= IN do
        Improvise a new harmony X_n
        if X_HMS>X_n then
            X_HMS=X_n
        else
            X_HMS=X_HMS
        end if
    end while
    Sort ascending the HMS solutions by value of O.F.
    Calculate the average of the HMS solutions
    for l=1 to l=HMS do
        if X_l ≤ average then
            Update pheromone based on X_l and update the probability
            Counter increases
        end if
    end for
    Save the X_l solution in HM
end for
Find the best solution among the best of each cycle
```

```
35. return the best solution
```

$1/d_{ij}$, is the inverse of the distance between two customers and 1/TW, is the inverse of l_i

4. Results

4.1 Setting parameters for AS-HS

The setting of the parameters is an important activity, in order to achieve an adequate behavior of whatever metaheuristic. In this work, an adjustment based on Differential Evolution (*DE*) is used. This technique requires two control parameters (cross factor *Cr* and mutation factor *F*), which were 0.5 values.

DE works with populations, where each individual is a vector that contains the parameters for AS-HS. *DE* obtained the following results for the *AS-HS* parameters (Table 1):

Table 1. Setting of parameters**.**

Parameter/Value	
Iterations number	50
Population size	10
α	1.45
β	1
γ	1.55
ρ	0.75
Improvisation number (IN)	6
Scanning ratio (HMCR)	0.65
Pitch adjustment ratio (*PAR*)	0.18

4.2 Comparative results with other authors

For the analysis of the robustness and performance of the technique, used a set of 9 test instances of 100 clients proposed by Solomon in [3] compared the best results obtained with those reported in [1] where applied a Modified Artificial Bee Colony, [2] where applied a variable neighborhood search with compound operators VNS-C and [8] where applied three meta-heuristics designed for the VRP-TW; these techniques are of the latest researches carried out for this problem. The authors of [1] reported around of 20,000 evaluations of the objective function, [2] reported more than 1,000,000 evaluations and [8] reported around of 10,000 evaluations, while AS-HS performs 3,000 evaluations. The comparison of time (in seconds) and results can be seen in Table2.

Table 2. Comparison of results between AS-HS and the state of the art.

Instance	Comparative results											
	AS-HS			**VNS-C[2]**			**Modified ABC[1]**			**Cuckoo search, Central Force Optimization, and Chemical Reaction Optimization[8]**		
	Best	**Average**	**Time**	**Best**	**Average**	**Time**	**Best**	**Average**	**Time**	**Best**	**Average**	**Time**
C101	828.94	828.94	63.4	828.94	828.94	--	828.94	828.94	700	828.94	--	--
C102	828.94	831.25	57.8	828.94	876.79	--	828.94	828.94	700	828.94	--	--
C103	**828.06**	828.06	54.3	828.94	832.65	--	828.94	840.66	700	**828.06**	--	--
C104	**824.78**	825.95	56.7	825.65	831.79	--	858.90	889.10	700	**824.78**	--	--

C105	828.94	828.94	58.6	828.94	852.33	--	828.94	828.94	700	828.94	--	--
C106	828.94	829.43	56.3	828.94	836.25	--	828.94	828.94	700	828.94	--	--
C107	828.94	830.43	67.2	828.94	853.9	--	828.94	828.94	700	828.94	--	--
C108	828.94	828.94	76.2	828.94	840.48	--	828.94	830.85	700	828.94	--	--
C109	828.94	828.94	78.2	828.94	828.94	--	828.94	836.97	700	828.94	--	--
Set average	**828.38**	**828.98**	63.18	828.57	842.45	--	832.26	838.03	700	**828.38**	**--**	**--**

Wilcoxon sum rank test with a 95% confidence was applied to the results obtained with aim compared the behavior of our method with the other methods. The information generated by the Wilcoxon sum rank test on the best results demonstrated that the four methods have similar performing. In contrast, the data produced by the Wilcoxon sum rank test with average results exhibited that the performance our method is distinct that the behavior of VNS-C. Finally, the information obtained by the Wilcoxon sum rank test on time uncovered that the behavior of AS-HS is different that Modified ABC. Also, the proposed method performs less than 50% of calls to the objective function with regards to the others techniques.

5. Conclusions and Future Work

In this paper, a hybrid algorithm of two metaheuristics to solve the VRP-TW is proposed, this technique obtain better results than the state of the art. Furthermore, with regard to the comparison made with the techniques reported in [1], [2] and [8] it can be observed that in AS-HS is performed less than 50% of calls to the objective function that the others. For which it is possible to corroborate that good results are obtained for the problem using less computing resources and less time.

As a future work, a reactive version of the proposed method for efficient search will be performed, to reduce the number of calls to the objective function and computation time.

Acknowledgments

This work has been partially supported by CONACYT Agreement 539200 and PRODEP.

References

1. Alzaqebah M., Abdullah S., and Jawarneh S.: Modified artificial bee colony for the vehicle routing problems with time Windows. Research (Springer), 2016.
2. Binhui, C., Rong, Q., Ruibin, B & Hisao, I.: A Variable Neighbourhood Search Algorithm with Compound Neighbourhoods for VRPTW. The 2016 International Conference on Operations Research and Enterprise Systems (ICORES’16), 23-25, Feb 2016.
3. Cordeau, F., Desaulniers, G., Desrosiers, J., Solomon, M., & Soumis, F.: The VRP with time windows. Technical Report Cahiers du GERAD G-99-13, 1999.
4. Dorigo, M., Maniezzo. V., & Colorni, A.: Ant System:Optimization by a Colony of Cooperating Agents. IEEE Transactions on Systems, Man and Cybernetics, 1996.
5. Lenstra, J. K., & Rinnoy Kan, A. H. G.: Complexity of vehicle routing and scheduling problems. 1981.
6. Geem, Zong Woo, Joong Hoon Kim, and G. V. Loganathan (2001). "A new heuristic optimization algorithm: harmony search." *Simulation* 76.2: pp. 60-68.
7. Mahdavi, M., Fesanghary, M., & Damangir, E.: An improved harmony search algorithm for solving optimization problems. Applied Mathematics Computation, 2007.
8. Saeheaw T. and Charoenchai N.: Comparison of Meta-heuristic Algorithms for Vehicle Routing Problem with Time Windows. Advanced Computer and Communication Engineering Technology, 2015.