Solution of the Assignment of Schedules Problem with the Gray Wolf Optimizer (GWO) Applying Evolutionary Parameters

Humberto Muñoz Bautista[1], Juana Canul-Reich[1], Alejandro Padilla Díaz[2],
Julio Cesar Ponce Gallegos[2], Jaime Muñoz Arteaga[2], Pedro Daniel García González[3]

[1] Universidad Juárez Autónoma de Tabasco, Villahermosa, Tabasco,
Mexico

[2] Universidad Autónoma de Aguascalientes, Aguascalientes,
Mexico

[3] Triple3 Aguascalientes, Aguascalientes,
Mexico

{hmuntista, jcanulreich, apadillarobot, jmauaa} @gmail.com,
jcponce@correo.uaa.mx danielgarcia@tripletres.com

Abstract. The educational institutions are faced with the problem of assignment their educational resources (students, teachers, classrooms, material) each scholar period, so there is a need to look for new software tools to satisfy this goals, due to the great amount of possible restrictions that has the assignment schedule problem, it is convenient to use metaheuristics algorithms to obtain acceptable solutions in reasonable times, for this reason, the present article shows the proposal for the resolution of the assignment schedule problem in the educational institutions using the metaheuristic GWO, taking as a use case a public university, a private university and a higher education institute. The test was carried out in a private university, obtaining better results in comparison with a genetic algorithm, demonstrating the potential of the metaheuristic GWO with evolutionary parameters.

Keywords: GWO, metaheuristics, assignment of schedules, optimization, artificial intelligence.

1   Introduction

The assignment of schedules problem, is a recurrent situation in all educational institutions [1], where in each scholar period the schedule of the students is planned; assigning signatures, classrooms and teachers, with the objective of avoiding an overlap of hours [2]. Therefore it can be defined as a decision-making problem for the assignment of educational resources (students, teachers, classrooms, etc.), subject to restrictions. As such, it corresponds to optimization problems in computational

complexity theory classified as NP-complete problems or NP-hard problems [3], due to its complexity, resource limitations and number of restrictions.

These problems do not have an algorithm that solves them in a polynomial time [4], that is to say, it is not possible to find its optimal solution with acceptable computational efforts, although you can have high-speed computers working in parallel. A major problem of optimization is the phenomenon called combinatorial explosion that means, when the number of decision variables of the problem grows, the number of feasible decisions and the computational effort grow exponentially, requiring the entry of applications such as intelligent algorithms [5]. However, not all combinatorial problems are so complex to solve; there are some for which there are algorithms that solve these problems with a computational effort that grows polynomial with the size of the problem [6].

There are several techniques for the resolution of the NP-complete problems, one of them are the metaheuristics, which deliver an acceptable solution within a reasonable time or in others words "Meta-heuristic algorithms are black box procedures that, provided a set of candidate solutions, solve a problem or a set of problems instances" [7]. A new metaheuristic algorithm is the Gray Wolf Optimizer, which is based on the behavior of the gray wolf, where the hierarchy of leadership and hunting mechanism of the gray wolves in nature are imitated. This metaheuristic corresponds to the so-called population metaheuristics, specifically considered within the branch of the Swarm Intelligence o SI, this concept was proposed in 1993 [8], and at the same time allows to use evolutionary operators, so when using such operators, can be considered an evolutionary metaheuristic and bioinspired algorithms, which are used in the optimization [9].

Taking the concept of Bonabeau, Dorigo and Theraulaz, SI is "The Intelligent collective of groups of simple agents" [10]. The most popular SI techniques are: Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO) and the Artificial Bee Colony algorithm (ABC) [11]. Such techniques have the following advantages:

− Save the search space information during the iterations of the algorithm.
− Usually use memory to save the best solution obtained so far.
− Usually have few parameters to adjust.
− Have fewer comparison operators than evolutionary methods.
− SI algorithms are easy to implement.

Considering the above is intended to carry out a parameter adjustment to the Gray Wolf Optimizer and implement it in the solution of the assignment schedule problem through the application of evolutionary parameters, to try to obtain a tool that equals or preferably obtains better results than those thrown by the processes currently used (many institutions perform this task manually) or by those systems that apply traditional metaheuristics such as the genetic algorithm.

2   State of the Art

2.1   Metaheuristics

A metaheuristic is a heuristic method to solve one type of general computational problem, using the parameters given by the user about generic and abstract procedures

in a way that is expected to be efficient. The name of metaheuristic comes from putting the prefix meta ("above" or of "higher level") at term "heuristic" (heuriskein, "find") [12]. A heuristic refers to methods based on experience that help us solve problems [13], this means that metaheuristic algorithms perform a process at a higher level to find the solution to a problem. Such algorithms can be conceived as general strategies of design of heuristic procedures for the resolution of problems with a high performance, using the parameters given by the user on generic procedures and abstract in a way that is expected to be efficient, so they can be applied to different optimization problems with minimal changes to be adapted to a specific problem.

The metaheuristics are not the solution to everything and many times are often less efficient tan specific heuristics, in several orders of magnitude, in problems that accept this type of pure heuristics; however, the advantage of metaheuristics is exploited, when are applied in problems that do not have a specific algorithm or heuristic to obtain a satisfactory solution, or when it is not feasible to implement an optimal method. Metaheuristics are widely recognized as one of the best approaches to tackle combinatorial optimization problems [14], so they have become popular in the last couple of decades for its characteristics of simplicity, flexibility, by having free bypass mechanism and that can avoid falling into local optima.

Amazingly some of them as the Genetic Algorithm(GA) [10], Ant Colony Optimization(ACO) [15] and Particle Swarm Optimization (PSO) [16] are well known not only in the area of computational sciences but also in other areas of science as: biology, economy, structural engineering, among many others. In addition a large number of theoretical works, such as optimization techniques, have been applied in various fields of study, because metaheuristics are quite simple, most have been inspired by very simple concepts, typically related to natural phenomena such as: the behavior of animals, evolutionary concepts or physical phenomena.

2.2   Metaheuristic GWO

Is a recent metaheuristic proposed by Mirjalili, Mirjalili y Lewis in the year of 2014 [17], which is based on the behavior of the gray wolf (Canis lupus). The gray wolves are in the top of the food chain, because they are considered one of the best predators. One of the reasons why this happens is by their lifestyle in herds. This lifestyle entails a very strict social dominant hierarchy. The herds are composed by five to twelve wolves on average, with a male and a female leaders (alphas), which make the decision to hunt, place to sleep, time to wake up; that means they make the most important decisions in the herd and their orders are followed by the herd [18]. In the second step of the hierarchy are the beta wolves, this wolves are subordinate that helps the alpha in decision-making or in others activities (alpha counselor, discipline the rest of the herds), for this reason they are the likely candidates to be the following alpha. The lowest ranking in the herds is omega, their plays the role of scapegoat, they always have to summit to all the other wolves in steps above of them and they are the last wolves to eat. In this complex social hierarchy each individual plays an important role, however small it may be, which helps maintain balance and strength of the herd.

If a wolf is not an alpha, beta or omega, is called subordinate or delta, this wolves have submit to alphas and betas, but they dominate the omega, their roles are scouts, sentinels, elders, hunters and caretakers, this wolves watching the boundaries of the

territory, warning the herd in case of any danger, protect and guarantee the safety of the herd, help in the hunting prey to provide food and are responsible for caring for the weak, ill, and wounded wolves in the herd. Elders are the experienced wolves who used to be alpha or beta. However, in some cases the alpha follows the other wolves in the Heard, demonstrating a democratic behavior, another example of such behavior occurs at meetings, where, the entire herd acknowledges the alpha by holding their tails down. The alpha wolves are only allowed to mate in the herd, but interestingly, they are not necessarily the strongest member, but they are the best managing the herd, demonstrating that the discipline and organization in the herd is more important than the individual strength.

Muro [19] describes the main phases of the hunting techniques of the gray wolves as the result of the perfect combination between the social hierarchy and group hunting, the main phases are as follows:

− Tracking, chasing and approaching to the prey.
− Pursuing, encircling, and press the prey until it stops moving.
− Attack towards the prey.

This hunting technique and the social hierarchy of grey wolves are mathematically modeled to design the GWO through of the main use of vectors for the representation of the positions of hunter and prey.

The algorithm presents the following processes, taken from the natural behavior of the gray wolf hunt:

− Encircling prey.
− Hunting.
− Attacking prey (exploitation)
− Search for prey (exploration)

2.3   Assignment of Schedule Problem

Describing in a general way the variables and the resources that the universities have for the assignment of schedule that they realize each scholar period, we have the following points [20]:

− Classrooms,
− Subjects,
− Teachers,
− Days of week,
− Class time,
− Assignments by period,
− Attendees,
− Room capacity,
− Maximum of attendees for subject,
− Department or faculty,
− Teacher availability.

There are many approaches to the assignment of schedule problem but in the present work will be addressed for a general case in the institutions. In that model starts from

the assignment of classes by department, where it is considered the classrooms, teachers, periods and assignments by period, as we can see in the next example.

Semester: 1
Department: A
Groups:
- Group: 1
    o Subjects:
        ▪ Subject: S1
            • Imparted by: T1.
            • Classroom: 1A
            • Periods: 3
                o Monday – 9:00 to 10:00
                o Tuesday – 9:00 to 10:00
                o Friday – 10:00 to 11:00
        ▪ Subject: S2
        ▪ …
        ▪ Subject: SN
- Group: 2
    o Subjects:
        ▪ Subject 1
        ▪ Subject 2
        ▪ Subject N
- Group: 3
    o Subjects:
        ▪ Subject 1
        ▪ Subject 2
        ▪ Subject N
Group: N

As we can see the assignment of schedules entails a big number of variables to be considered, becoming a complex problem. "On a daily basis, when we are faced with a complex problem, we turn to different people with more experience or experts in the area of the problem, to know and analyze their ideas and points of view. What leads us to perform an analysis of this information and group it by a classification to make a decision towards the problem we want to solve" [21]; however by extending the assignment of schedule problem to larger dimensions, as it is the programming of schedule in universities or higher education institute, where there is a large number of students, teachers, subjects and more resources to assign in each school period , taking an appointment to an expert would take a long time, so they have been proposed several techniques to try solve that problem, an example of this techniques are: linear programming [22] and the metaheuristics, like the evolutionary algorithms [23], ACO[24] y GRASP[25]. In such a way that it is proposed to use the algorithm GWO for the resolution of the assignment of schedule problem.

3 Methodology

For the realization of the research we make use of the quantitative methodology, taking into account the fulfillment of the following points:

−   Splices of groups,
−   Teacher splices,
−   Free hours.

The objective is to minimize each point, taking as an optimal result to obtain 0 in each item. For the study case, the available schedules of teachers according to their regulatory academic burden, the subject (s) they can teach according to their academic profile, the subjects per group and the number of students that can be enroll by subject, are taken into consideration. In such a way that the teacher-matter-group relationship can be generated, obtaining the schedules assigned to each group, avoiding the points mentioned at the beginning of the section.

The tests are done using a laptop with a processor Intel Core i7 de 2.6 GHZ, 16 GB in RAM and with a SSD PCI-E of 512 GB and the development of metaheuristic is carried out with Matlab, Reading the information of teachers, subjects and groups from a file in Excel.

The expected results are to minimize the splices of groups, teacher splices and the free hours for the students, using the GWO metaheuristic adapted to solve the problem of assignment schedules as described in the next section.

4 Proposed Model to Resolve the Assignment of Schedule Problem by the GWO

Taking as a start the example of the previous model, the following parameters and restrictions are defined, to obtain a mathematical model that allows us to propose the assignment of schedule problem to be solved by the GWO.

4.1 Parameters

− r = Class rooms,
− s = Subjects or assignment,
− t = Teacher,
− wp = Week periods, (ex.4 times a week)
− sh = Schedule shifts,
− g = Group.

where, all of above integers:

$$\forall r, s, t, wp, sh, g \in Z. \tag{1}$$

Cr= Room capacities. Where each element **r** of set **C** is an Integer that represents the capacity for r' room.

$$\forall r \in [1, \dots, N]. \tag{2}$$

Sa= Max Attendees by Subject. Where each element **a** of set **S** is an Integer that represents the Maximum of attendees that can be signed up to s' subject:

$$\forall a \in [1, \dots, N]. \tag{3}$$

Ph= Hours by Period. Where each element **h** of set **P** is an Integer that represents the Maximum of hours that a period p' is scheduled for:

$$\forall h \in [1, \dots, N]. \tag{4}$$

Th= Hours by Teacher. Where each element h of set T is an Integer that represents the Maximum of hours that a teacher t' can work:

$$\forall h \in [1, \dots, N]. \tag{5}$$

4.2 Variables

To build the viability V of schedules assignment, we analyze that there must be a classroom assigned to the subject with a single teacher in a specific day-hour period. We consider by default that the variable **V** is the viability for a single group **g** of a set of **N** groups. We obtain the next expression.

We can easily imagine this scenario as a Schedule sheet (figure 1) as you have in university; for example: Intelligent computing A that belongs to Department of Computer Science.

$$V r' s' t' h' p' s h' = \begin{cases} 1, & \text{if there a classrom } r' \text{ not assigned for a subject } s' \text{ with a teacher } t' \\ & \text{on a specific hour } h', \text{period } p' \text{ and schedule shift.} \\ 0, & \text{if not avaible.} \end{cases}$$

As another point of view, we can see the same grid of schedule by subjects for groups by semester and by department (figure 2).

Hour\Week	Monday	Tuesday	Wednesday	Thursday	Friday
8:00-9:00	Classroom: C1 Teacher: T1. Subject: S1	…	…	…	…
9:00-10:00	…	…	…	…	…
10:00-11:00	…	…	…	…	Classroom CN Teacher TN Subject: SN

Fig. 1. Simple example of class schedule viewed as a calendar.

Assignments\ Days	Monday	Tuesday	Wednesday	Thursday	Friday
S1	Classroom: C1 Teacher: T1 Period: 9:00-10:00				
S2					
S3					
SN					Classroom: CN Teacher: TN Period: PN

Fig. 2. Example of schedule for group A, in Semester 1 by a Specific Department.

4.3 Restrictions

At this moment we have the main structure of the class schedule problem, but we need to define the hard and soft restrictions.

Hard restrictions

For the general model we consider some mandatory restrictions taking the parameters described above:

1. One Subject must be imparted by only 1 Teacher in 1 Classroom. For example: Biology – Mr. Lee – Classroom A:

$$V(s't'r') \leq 1 \; ; \; \forall t, r. \tag{6}$$

2. One Subject must be imparted in only 1 Classroom in a Single period. For example: Biology – Classroom A – 9:00 to 10:00:

$$V(s'r'h'p') \leq 1 \; ; \; \forall r, h, p. \tag{7}$$

3. Each Subject must complete the all Weekly periods. For example, Biology – 5 periods by week:

$$\sum_{i}^{s} SA[i] = \boldsymbol{wp} \; ; \; where \; SA = \{s_1, s_2, .., s_s\}. \tag{8}$$

There must not be more than 1 Subject assigned to the same period. For example: Monday- 10:00 - Biology, Monday -11:00- Artificial Intelligence:

$$V(s'h'p') \leq 1 \; ; \; \forall h, p. \tag{9}$$

4. Not exceed the maximum capacity of classroom. Example: 50 attendees are impossible to assign in the Classroom A, due it must be less or equals than 40 for Classroom A:

$$X \leq C[r]; where \; C = \{c_1, c_2, \ldots, c_r\} \; capacities. \tag{10}$$

5. Not exceed the maximum teacher period assignment (T). For example, the Teacher T1 is only available for 4 weekly periods, so their assigned classes (X) should not be greater than 4:

$$X \leq T[h]; where \; T = \{h_1, h_2, \ldots, h_t\}. \tag{11}$$

Soft restrictions

For this general model, we do not consider a specific set of soft restrictions due these restrictions are particularly not needed. But inside of this restrictions you can consider any specific situation like: A Subject cannot be imparted in a specific classroom, or a set of teachers prefer a specific turn, and so on.

4.4 Objective Function

For our main fitness function, we have to consider the set of hard restrictions (high cost) are accomplished at the same time that we have a positive result of our Viability result, adding as an optional soft restrictions (low cost).

The equations establish that the calculus of value of goodness an individual i at an instant t is given by:

$$r(i,t) = \sum_{j=1}^{Nc} |s(i,j) - c(i,j)|, \tag{12}$$

where:

− s (i, j) = Desired value for individual i in case j,
− c (i, j) = Obtained value for individual i in case j,
− Nc= Number of cases,
− i= individuals,
− j= Cases.

As we need to minimize "conflicts", these occurrences that would be present for class schedule solution should be calculated:

1. Sum of conflicts for soft restrictions:

$$hrcs = \sum_{i}^{w} confl(Ws[i], HR), \tag{13}$$

where:

- $HR = \{hr_1, hr_2, ..., hr_n\}$ Hard restrictions,
- $Ws = \{w_1, w_2, ..., w_n\}$ Wolf solutions,

```
Initialize the grey wolf population Xi (i = 1, 2, …, n)
Initialize a, A, and C
Calculate the fitness of each search agent
Xα = the best search agent
Xβ = the second-best search agent
Xδ = the third-best search agent
while (t < Max number of iterations)
        for each search agent
                                                    X(t + 1) = (X̄₁ + X̄₂ + X̄₃)/3
                Update the position of the current search agent
        end for
        Update a, A, and C
        Calculate the fitness of all search agents
        Update Xα, Xβ, and Xδ
        t=t+1
end while
return Xα
```

Fig. 3. Algorithm GWO proposed by Mirjalili in 2014.

2. Sum of conflicts for soft restrictions:

$$srcs = \sum_{i}^{w} confl(Ws[i], SR), \tag{14}$$

where:

$SR = \{hr_1, hr_2, ..., hr_n\}$ are soft restrictions.

Confl () is a function that evaluates a generated solution against a Soft or Hard set of restrictions.

Taking as reference the above equations we have the next objective function. Minimize the sum of conflicts for hard restrictions and soft restrictions:

$$f = MIN(hrcs + srcs). \tag{15}$$

In the literature we can see many examples of metaheuristics applied for general problems of scheduling, such as genetic algorithms, ant colony optimizations, swarm particle optimization, and so on.

In this article we are considering a new adaptation for this problem to be solved by an adaptation of general metaheuristic of the Gray Wolf Optimization. (GWO, figure 3)

In that adaptation of the metaheuristic we consider establish the next set of parameters and variables, which we described in the section III of the present article:

```
Initialize the grey wolf population Xi (i = 1, 2, ..., n)
Initialize a, A, and C
Initialize r, s, t, wp, Rc, and Sa
Initialize Hr, Sr restriction sets.
Calculate the fitness of each search agent
Xα = the best search agent
Xβ = the second-best search agent
Xδ = the third-best search agent
while (t < Max number of iterations)
        for each search agent
                Update the position of the current search agent unless V is feasible
                end unless
        end for
        Update a, A, and C
        Calculate the fitness of all search agents
        Update Xα, Xβ, and Xδ
        t=t+1
end while
return Xα
```

Fig. 4. Adapted algorithm GWO.

Table 1. Quantitative comparison of results between GWO, SCAIES (ACO) and manual method.

Method	Splices of groups	Teacher splices	Free hours
GWO	0	10	11
SACAIES	4	0	93
Manual method	0		
The smallest value is better	Avoid splicing groups at the same time and place	The manual method prevents the joining of teachers; however, he does it in a much longer time	Free hours is an important part of the solution to avoid having students with cut-off hours

－ Parameters r, s, t, wp, Rc y Sa,
－ Hard restrictions Hr [],
－ Soft restrictions Sr [],
－ Viability (Search agent positioning) V,
－ Fitness function.

Resulting in the algorithm presented in the figure 4.

Fort the proposed algorithm the update of position of the current search agent occurs when the viability of solution is feasible.

5   Results

This section show, a comparison of the results obtained between GWO metaheuristic with evolutionary parameters, the SACAIES system [26], which uses a genetic

algorithm and a manual method for the assignment of schedules. Table 1 shows the results obtained by each of the methods. Taking into account the main objectives of the assignment of schedules (teacher splices, splices of groups and free hours).

As can be seen, the proposed metaheuristics improves the results obtained by a manual method and that obtained by an ACO metaheuristic, avoiding the splices of groups and reducing both free hours and the teacher splices.

Although it does not avoid 100% of teacher splices and free hours, the result obtained allows have the satisfaction of the students because they do not have cut-off schedules and make the most of their hours per day, while the teacher splices, being a single case, allows the intervention of the administrative responsible for the planning of schedules to solve a single case, instead of assigning all hours from 0. With these solutions, the metaheuristic adapted from the GWO demonstrates good results for the assignment of schedule problem, fulfilling the objective of avoiding splicing of groups and minimizing the free hours and the teacher splices.

6   Conclusions and Future Work

In the present work we established the necessary criteria for proposed the solution to the assignment of schedule problem with the GWO, resulting in an adaptation of the metaheuristic and obtained a general model to the solution of the problem. For future works we are in the approach of the problem to a higher education institute, a private university and a public university of the state of Aguascalientes, because each one of them has a different organization, which causes the problem approach to be modified.

For future works we proposed, the implementation of the GWO adapted to make an allocation of menus per day in a balanced diet, test the metaheuristic in public institutions of higher and middle higher level; as well as attacking the problem of assignment task by processes, recurrent in the industry.

References

1. Baldonado, M., Chang, C. CK., Gravano, L., Paepcke, A.: The Stanford Digital Library Metadata Architecture. Int. J. Digit. Libr. 1, pp.108–121 (1997)
2. Aguilar, L.D.M.O., Valadez, J.M.C., Soberanes, H.J.P., González, C.L.D., Ramírez, C. L., Soria-Alcaraz, J.: Comparativa de algoritmos bioinspirados aplicados al problema de calendarización de horarios. Research in Computing Science, (94), pp. 33–43 (2015)
3. Bruce, K.B., Cardelli, L., Pierce, B.C.: Comparing Object Encodings. In: Abadi, M., Ito, T. (eds.): Theoretical Aspects of Computer Software. Lecture Notes in Computer Science, 1281, Springer-Verlag, pp. 415–438 (1997)
4. Santoyo, A.M., Ortega, J.P., Salgado, G.R., Ortega N.N.A.: Caracterización de instancias difíciles del problema de Bin Packing orientada a la mejora de algoritmos metaheurísticos. Research in Computing Science, (19), pp. 295–308 (2015)
5. van Leeuwen, J. (ed.): Computer Science Today. Recent Trends and Developments. Lecture Notes in Computer Science, 1000. Springer-Verlag (1995)
6. Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolution Programs. 3rd edn. Springer-Verlag (1996)

7. Cosıo-León, M., Martınez-Vargas, A., Gutierrez, E.: An Experimental Study of Parameter Selection in Particle Swarm Optimization Using an Automated Methodology. Research in Computing Science, 82, pp. 9–20 (2014)

8. Ahumada, A., et al.: Generación de horarios académicos en INACAP utilizando algoritmos genéticos (2014)

9. Contreras-Varela, L.E., Mezura-Montes, E.: Comparativo empírico de medidas de diversidad en problemas de optimización evolutiva con restricciones. Research in Computing Science 134, pp. 69–83 (2017)

10. Caballero, J.M.M.: Asignación de horarios de clases universitarias mediante algoritmos. Universidad del norte división de postgrados e investigaciones en ingeniería, maestría en ingeniería industrial, Barranquilla atlántico (2008)

11. Baltazar, R., Vázquez, J.E., Rada, A., Díaz, C.: Desarrollo de un sistema capaz de optimizar rutas de entrega utilizando algoritmos genéticos, SINNCO (2010)

12. Isaza, R.A.H., Porras, C.A.R., Rendón, R.A.G.: Técnicas Heurísticas Aplicadas Al Problema Del Cartero Viajante (TSP), Grupo de Investigación en Planeamiento de Sistemas Eléctricos, Universidad Tecnológica de Pereira (2004)

13. Beni, G., Wang, J.: Swarm intelligence in cellular robotic systems. In: Robots and biological systems: towards a new bionics?, Springer, pp. 703–12 (1993)

14. Prospero, L.A.C., Mejía-Lavalle, M., Ascencio, J.R., Rincón, V.V.V.: Aprendizaje incremental basado en población como buena alternativa al uso de algoritmos genéticos. Research in Computing Science, 116, pp. 51–64 (2016)

15. Bonabeau, E., Dorigo, M., Theraulaz, G.: Swarm intelligence: from natural to artificial systems. OUP USA (1999)

16. Basturk, B, Karaboga, D.: An artificial bee colony (ABC) algorithm for numeric function optimization. In: IEEE swarm intelligence symposium, pp. 12–4 (2006)

17. Rodríguez, A. G.: Algoritmo del Lobo Gris para la solución de un problema de asignación cuadrática, Universidad Autónoma de Aguascalientes, Ingeniería en Computación Inteligente (2016)

18. Dorigo, M., Birattari, M., Stutzle, T.: Ant colony optimization. Comput Intell Magaz, IEEE 1, pp. 28–39 (2006)

19. Cardona, P., Muñoz, Á., Muñoz, J., Álvarez, F., Velázquez, C., Hernández, Y., Collazos, C.: Determinación de Heurísticas en Interfaces de Objetos de Aprendizaje Multiculturales mediante Cadenas de Markov. Conferencias LACLO, 4(1) (2013)

20. Kennedy, J., Eberhart, R.: Particle swarm optimization, in Neural Networks. In: Proceedings, IEEE international conference on, pp. 1942–1948 (1995)

21. Mirjalili, S., Mirjalili, M., Lewis, A.: Grey Wolf Optimizer. Advances in Engineering Software, 69, pp. 46–61 (2014)

22. Escobedo, C., Spector, R., Coppinger, R.: Wolf-pack (Canis lupus) hunting strategies emerge from simple rules in computational simulations. Behav Process, 88, pp.192–197 (2011)

23. Bautista, M. H., Díaz, A. P., Gallegos, J. C. P., Suárez, H. M., Aguilar, J. A. H., Canul-Reich, J.: Accomplishment of generic skills of the middle higher education with the use of system SiCRE, XI Latin American Conference on Learning Objects and Technology (LACLO), San Carlos, pp. 1–8 (2016)

24. Sánchez, D., Cano, P., Reyes, M.: Caso de un Modelo de Optimización para un Problema de Calendarización de Horarios. Universidad Popular Autónoma del Estado de Puebla, Puebla (2009)

25. Hernández, R., Miranda, J., Rey, P. A.: Programación de Horarios de Clases y Asignación de Salas para la Facultad de Ingeniería de la Universidad Diego Portales Mediante un Enfoque de programación Entera. Revista Ingeniería de Sistemas, XXII (2008)

26. Barrientos, J. D. J.: Aplicación de algoritmos genéticos para la asignación de carga académicos en instituciones de educación superior (2000)