

Selecting an Optimization Algorithm for the Administration of Human Resources Process in the Production Line of a Textile Enterprise

Alejandro Ernesto González Alegrant¹, Yenny Villuendas-Rey²,
Jarvin Alberto Antón Vargas³, Cornelio Yáñez-Márquez⁴

¹ Centro Provincial de Información de Ciencias Médicas, Ciego de Ávila, Cuba
alegrant.gonzalez@gmail.com

² Instituto Politécnico Nacional, Centro de Innovación y Desarrollo Tecnológico en Cómputo,
Mexico City, Mexico
yenny.villuendas@gmail.com

³ Universidad “Máximo Gómez Báez” de Ciego de Ávila, Ciego de Ávila, Cuba
jarvinalberto@gmail.com

⁴ Instituto Politécnico Nacional, Centro de Investigación en Computación, Mexico City,
Mexico
coryanez@gmail.com

Abstract. In this paper, we use Particle Swarm Optimization, with the aim of facilitating and optimizing the Human Resources administration process along the lines of production for the “Confecciones Trébol” textile enterprise of Ciego de Ávila city, Cuba, making use of historical data and metaheuristic algorithms. The evident need to optimize the process of allocation of Human Resources in the Production Line of “Confecciones Trébol”, led us to analyze the different aspects of the problem, and to use a bio-inspired algorithm to solve it. The validity of the mathematical models of optimization in the solution of a practical problem was determined; contributing to the efficient use of material re-sources. The implementation of a bio-inspired algorithm (PSO) is effectively achieved by complying with the constraints of the defined mathematical model, with which it is possible to obtain feasible solutions to reduce the production time of a textile product.

Keywords: optimization, bio-inspired algorithms, textile enterprises.

1 Introduction

Within Artificial Intelligence, the branch of bio-inspired algorithms are characterized by emulating the behavior of natural systems and, from them, designing non-deterministic heuristic methods of search, optimization, learning, recognition, simulation and characterization [1].

Hence, Artificial Intelligence is the future of computing and artificial systems in general; bio-inspired algorithms give naturalness to systems that, little by little, will be refined more, in order to achieve greater similarity with natural methods.

The textile industry is an important sector today in society and is constantly developing, which cannot be alien to new technologies [2]. In Cuba, there are several companies related to this area of the economy that have decided to participate in this process. Among them is the "Confecciones Trébol" enterprise of Ciego de Ávila.

This textile enterprise is dedicated to the manufacture of garments and other products and is dependent on work orders that are managed within it. The production of each of the garments is guided by different operations (cutting, sewing, manual work), due to the non-use of information that is managed administratively (average time of completion of the operation by operators); sometimes they don't take the best strategic decisions to optimize the time of making a garment. There are several insufficiencies in the planning of Human Resources in the production line of the textile "Confecciones Trébol" enterprise. To address such issues, we intended to apply a bio-inspired algorithm, in order to optimize the human resources allocation process.

The rest of the paper is as follows: Section 2 gives a panoramic view of the "Confecciones Trébol" enterprise. Section 3 addresses the materials and methods, while Section 4 presents the results. The article ends with conclusions and future works.

2 The "Confecciones Trébol" Enterprise

The "Confecciones Trébol" enterprise is an entity that is dedicated to the manufacture of garments and other products and is dependent on work orders that are managed within it. This enterprise is composed of three fundamental workshops: cutting, sewing and manual work; which all depend on the service order.

Once the service order is made, it is taken to the cutting shop, where it is analyzed thoroughly, and depending on the products to be elaborated, the sizes are distributed and the cloths are stretched and then cut. Once the cloths are cut, they are passed to the sewing workshop, where the same process is carried out; but in this case, the cloths cut by tailors, and by the operation that is going to be carried out, are distributed. After finishing in the sewing workshop, the next process is the manual workshop. The manual work workshop includes the operations of thread cutting, final iron, packing, opening seam, pocket plate; it is important to understand that the operations of manual work are of the intercalated type, because they are in the order of the product to be elaborated, and the moment when it is needed.

This entity has a client-server system that covers the activity of Human Resources, Production and pre-payroll management, but the system does not have access, nor does it operate with any information after biweekly or brief closing, and the address cannot make decisions, or analyze past events. The information is stored incrementally in cycles of 15 days in a database, with which important decisions could be made.

Due to the non-use of this information, the enterprise is not able to determine if there has been an indiscriminate use of standards in consumer operations for the same product in different 15 day cycles. Also, it is not possible to determine which seamstresses best perform a certain function or access sequences of operations that have been made in each cycle. There is no access to statistical information of the payments made for each product manufactured. In addition, it is not possible to determine the

seamstresses taking more time to execute the same operation, nor to determine the optimum value of the number of seamstresses operating the same product in relation to those that supply the packages and are responsible for supplying all the positions in the workshop. The analysis of productivity and production levels is made on physical documents instead of accessing this digital information that remains unused.

3 Materials and Methods

3.1 Particle Based Optimization Algorithm (PSO)

The "Particle Cumulus Based Algorithm" or Swarm Optimization Particle is a population-based metaheuristic technique inspired by the social behavior of flocks of birds in flight as well as the movement of schools of fish. The original PSO algorithm was developed by the psychologist and sociologist James Kennedy and the electronic engineer Russell Eberhart in 1995, from an approach known as the "social metaphor" [3], which describes this algorithm and summarizes the following form: individuals who live in a society have opinions that are part of a "set of beliefs" (the search space) shared by all possible individuals.

The particle cluster (swarm) is a multi-agent system, the particles are simple agents that move through the search space and save (possibly also communicate), the best solution they have found. Each particle has its own "fitness", position and vector velocity that directs its "movement". The movement of the particles in space is guided according to the optimal particles at the present time.

The PSO algorithm is an iterative and stochastic process, which operates on a cluster of particles; the position of each particle is the representation of a potential solution of the problem to be solved, see Table 1.

Table 1. Terms of the PSO Algorithm.

Particle	A person from the cloud.
Position	Coordinates of an agent in an N-Dimensional space that represents a solution to the problem.
Cloud	A whole collection of agents. A population of individuals.
Fitness	A number that indicates the quality of a given solution (represented by a location in the solution space).
Pbest	The best location obtained by a certain particle throughout the process.
Gbest	The best location of a particle in the whole cloud.
Vmax	The maximum speed allowed in a given direction.

The cluster is initialized from the generation of the positions and the initial velocities of the particles. It is possible to generate the positions randomly in the search space (perhaps with the help of a construction heuristic), on a regular basis, or by combining both forms; Once the positions are generated, the fitness of each is calculated and the values of *fitness_{xi}* and *fitness_{pBesti}* are updated, [4].

The speeds are generated randomly, with each component in the interval $[-vmax; vmax]$, where *vmax* will be the maximum speed that a particle can take in each movement); it is not convenient to fix them to zero, because good results are not obtained [5].

According to Sancho Caparini in [6], each particle (individual) has a position in the search space that is determined by a vector, and a speed with which it moves through the search space, equally determined by a vector. Like all particles of the real physical world, they have a capacity of inertia that keeps them in the same direction of movement, and an acceleration (change of speed) dependent on two characteristics mainly:

- Each particle is attracted to the best location that they, personally, has found in their story (best staff).
- Each particle is attracted to the best location that has been found by the set of particles in the search space (better global).

The global version, is where the distance of the social component is given by the difference in the position of the current particle and the position of the best particle found in the complete swarm *gBest_i*, in Figure 1, the pseudocode of the global version of the algorithm is given.

```
1- E= Initialize Population
2- While no stop condition do
3-   For i = 1 to size(E) do
4-     Calculate fitness of each individual
5-     If fitness(Xi) > fitness (pBest) then
6-       pBest = Xi
7-       fitness (pBest) = fitness (X)
8-     End If
9-     If fitness(pBest) = fitness (gBest) then
10-      gBest = pBest,
11-      fitness (gBest) = fitness (pBest)
12-     End If
13-   End For
14-   For j = 1 to size(E) do
15-     Calculate speed (Xj)
16-     Calculate Position (Xj)
17-   End For
18- End While
```

Fig. 1. Pseudocode of the PSO algorithm (Taken from [1]).

The global version of the PSO algorithm tends to converge faster because the visibility of each particle is better and closer to the best of the swarm, [1, 14]. S. Hillier and J. Lieberman states in [7] that the mathematical model of the problem is the system

of equations and related mathematical expressions that describe the essence of the problem.

In “Confecciones Trébol” textile enterprise of Ciego de Ávila city, it is necessary to optimize the production operations, and minimize the time. It must be specified that what is sought to be optimized is the production time of a textile product, taking into account that each product is made using a set of operations, carried out by the textile enterprise operators group. The decisions that should be made are subject to the number of operators per job in the production line, adjusted to the average time per unit of product and the types of products to be developed. According to authors in [8], an optimization model is constructed following three fundamental steps:

1. Define the essential or decision variables.
2. Construction of the system of restrictions.
3. Construction of the objective function.

Each decision variable is identified with each of the activities in which the problem that is studied is broken down, due to this the conceptual description for the variables in this problem are given in Table 2.

Table 2. Conceptual description of the decision variables for a product.

Variables	Description
O_n	Number of operators per job (Unknown)
T_n	Average time per product unit
$n \geq 1$	Initial conditions for each job

The optimization problem is subject to several restrictions, described as follows.

Regarding the condition of work, it is based on the fact that there must be at least one operator per job who intervenes in the production:

$$O_n \geq 1. \tag{1}$$

The workshop currently has 64 operators that work in the textile enterprise, therefore the following restriction is obtained:

$$\sum_{i=1}^n O_i \leq cant_{oper}. \tag{2}$$

In the workshop different operations are carried out that are adjusted to a time constraint determined by the work to be done in 60 minutes (1 hour); all this, in an 8-hour day (expressing it in minutes):

$$\frac{O_1}{Jt} + \frac{O_2}{Jt} + \frac{O_3}{Jt} + \dots + \frac{O_n}{Jt} \leq 60. \tag{3}$$

The objective function directly expresses the objective or purpose pursued depending on the problem to be investigated. The objective function for this model is constituted in the following way:

$$\min \rightarrow f(O) = \frac{O_1}{T_1} + \frac{O_2}{T_2} + \dots + \frac{O_n}{T_n}. \quad (4)$$

Table 3 summarizes the description of the optimization problem to be solved.

Table 3. Optimization problem.

Variables	Description
O_n	Number of operators per job (Unknown)
T_n	Average time per product unit
$n \geq 1$	Available jobs
Restrictions	Description
$O_n \geq 1$	Working condition: at least one operator per job, which is involved in the production
$\sum_{i=1}^n O_i \leq cant_oper$	Availability of workforce
$\frac{O_1}{Jt} + \frac{O_2}{Jt} + \frac{O_3}{Jt} + \dots + \frac{O_n}{Jt} \leq 60$	Work to be done in 1h
Objective Function	
$\min \rightarrow f(O) = \frac{O_1}{T_1} + \frac{O_2}{T_2} + \dots + \frac{O_n}{T_n}$	

4 Using PSO to Solve the Human Resources Administration for “Confecciones Trébol” Textile Enterprise

The authors state in [9] which PSO potential solutions, called particles, are within the search space, following the optimal particle, which are updated with an internal speed. They also have a memory, which is an important aspect for this algorithm. In addition, while PSO is easy to implement and has few parameters to adjust, it has successful applications in various areas: optimization functions, training of artificial neural networks, and diffuse control system [9, 10].

Before coding and implementation, the authors state in [11, 12], to verify that the algorithm will use few resources, the most important being the time it takes to run and the amount of memory space required. The analysis of the algorithmic complexity is shown, being this of $O(n * m) \equiv O(n^2)$.

As a first step of the algorithm consistent with the generation of the population, it was generated randomly. The number of particles to be generated is given in a typical range of [20, 40]. This value is subject to previous experiments carried out during the implementation and adjustment of the algorithm, looking for the results to be consistent with expected results.

The PSO model can be purely cognitive if for the particle trend it depends on the best positions found in your personal past; or, it can be purely social if this tendency is proportional to the past of the cluster. If the model has both components (cognitive and social) the velocity of each particle can determine how they converge to the optimal value. That is why Shi and Eberhart [13] proposed an improvement of the basic algorithm, modifying the formula for updating the speed and introducing a new variable called the inertial factor w .

Both the process of exploration and exploitation of the search space, as well as the control that allows the speed does not exceed its established limits, are balanced by the inertia factor. This factor regulates the speed, multiplying its weight by the previous speed. A large weight of inertia facilitates the global search, while a small weight of inertia facilitates local search. Generally the best value for w is dependent on the problem, by incorporating the inertial factor w the update of the speed is determined in the formula shown in Figure 2, where the variable V_{id} is the speed of the previous iteration of the particle i in dimension d (where d is the number of problem variables), r_1 and r_2 are random values in the range [0,1], p_1 and p_2 are the personal and social learning coefficients respectively, pb_{ia} (personal best) is the best personal position found by the particle i in the dimension d , pos_{ia} is the current position of the particle i in the dimension d y gbd (global best) is the global position vector, best found by all the particles, obtained in the objective function and it is updated after each iteration of the algorithm. The value of the inertial factor w can remain fixed or it can be linearly decremented in each flight cycle, in the case of our investigation the inertial factor is fixed.

$$v_i \leftarrow w \cdot v_i + \varphi_1 \cdot rand_1 \cdot (pBest_i - x_i) + \varphi_2 \cdot rand_2 \cdot (lBest_i - x_i)$$

$$x_i \leftarrow x_i + v_i$$

Fig. 2. Formula proposed by Cárdoma y Nieto in [1, 4].

In order to control the possible solutions generated in such a way that they were adjusted to the feasible solutions, the strategy of keeping the minimum value on the components that were below the allowed value was adopted; in the same way a respective strategy was adopted for the maximum values. Table 4 shows the values for each of the parameters of the algorithm.

Throughout the development of the system and adjustment of parameters of the algorithm, it was necessary to check the efficiency of the same; for this purpose, predefined and real data were used. Within the first runs of examples, a text-type file was created where the times were initialized by operation, parameters such as the number of operators, number of particles, the coefficient of cognitive or personal learning, the overall learning coefficient were adjusted and the values of the inertial factors.

Table 4. Parameters for the PSO algorithm.

Parameters	Values
Number of particles	40
Number of iterations	100
Inertia factor value	0,1
Value of the personal learning coefficients	0,1
Value of social learning coefficients	0,9

For the adjustment of the appropriate values for the inertial weights, it was taken into account that they depended mainly on the problem to be solved. In the process of trial and error, different values of weights were tested, each time closer to the final solution, where the set of values was analyzed, leading to the best results in the problem.

Based on the other part of what is addressed by Kowalski and Lukasik in [14] where *"the choice of the parameter is based on the results of the numerical simulation that indicate the quality of the solution: the average value of error with its standard deviation, the best result obtained by the swarm and, additionally, the speed of its convergence"*, it is decided to decrease the value of the weight inertia parameter. Finally, a sequence of the best global particles in each of the iterations generated by the algorithm, taking into account the lower fitness value of the particles is obtained.

After adjusting parameters of the model and the algorithm, it was necessary to check the efficiency of the same, for this purpose, predefined and real data were used. Within the first run of examples, a text file was created where the times were initialized by operation, parameters such as the number of operators, number of particles, the values of inertial factors, the coefficient of cognitive learning were adjusted or personal and the overall learning coefficient. In the test program, several iterations were obtained and each obtained results closer and closer to the final solution, which was only obtained when the stop condition was met (maximum number of iterations). Finally a sequence of the best global particles in each of the iterations generated by the algorithm taking into account the lower fitness value of the particles. An example of this process can be seen in Figure 3.



Fig. 3. Examples Algorithm PSO in several iterations.

Now with real data, as shown in Figure 4, once the operations to be carried out by a garment have been loaded, the possibility of choosing how to carry them out in three priority groups is offered.

When the priorities of these operations have been delimited, the algorithm will executed it, and will be able to offer an optimal solution of how to organize the human resources of the workshop in order to optimize the time of making a garment.

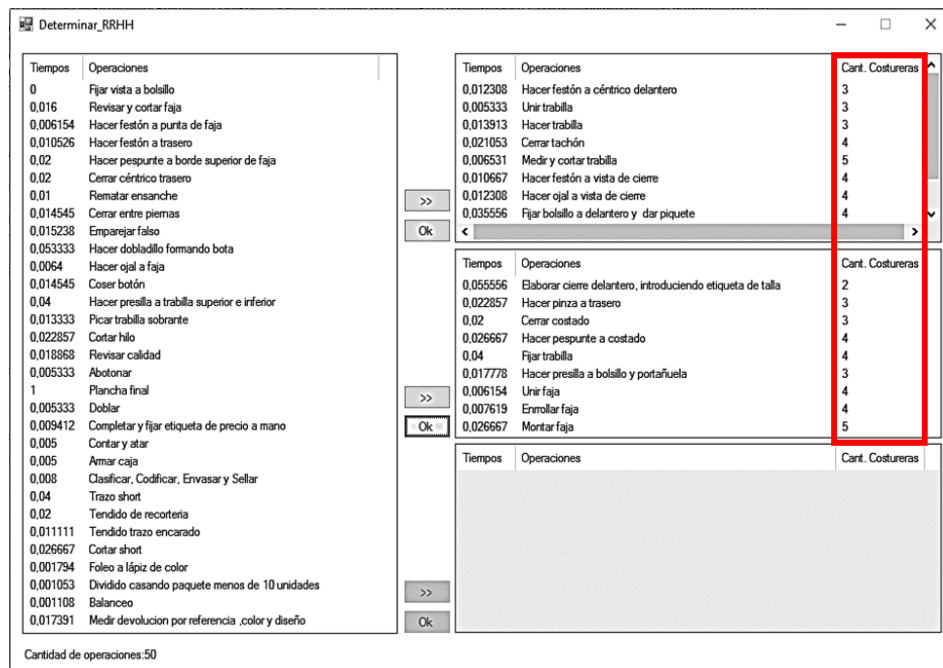


Fig. 4. Final result of the allocation of Human Resources.

Therefore, the algorithm correctly satisfies the constraints defined in the model, returning values within the range of acceptable values. Once again, fulfilling the condition of stoppage and the objective of determining the allocation of number of operators per operation of a product to be elaborated it in the textile enterprises. An analysis of the process of allocation of Human Resources of the “Confecciones Trébol” textile enterprises was achieved, which allowed defining a mathematical model of optimization adjusted to the characteristics of the process of making a textile product, taking into account the inherent restrictions of the process and the real capabilities of the entity.

5 Conclusions

The evident need to optimize the process of allocation of Human Resources in the Production Line of the “Confecciones Trébol” textile enterprise, led to analyzing the

different aspects of each of the methods and heuristic techniques to select the appropriate one according to the speed and certainty of the answers.

The validity of the mathematical models of optimization in the solution of a practical problem was determined, contributing to the efficient use of material resources. The implementation of a bio-inspired algorithm (PSO) was effectively achieved by complying with the constraints of the defined mathematical model, with which it is possible to obtain feasible solutions to reduce the production time of a textile product.

Finally, future work will continue with the study of the historical data of the textile enterprise, for a future adjustment of the model, achieving a greater accuracy in the response of the PSO something-rhythm.

References

1. Cárdenas Cardona, A.: *Inteligencia artificial, métodos bio-inspirados: un enfoque funcional para las ciencias de la computación*. Universidad Tecnológica de Pereira (2012)
2. Becerra Días, L.: *Metodologías para fortalecer los mecanismos evaluadores de la efectividad económica a través de la reducción de los costos de producción en la empresa de confecciones Trébol*. Ciego de Ávila, Cuba (2012)
3. Eberhart, R. C., Shi, Y., Kennedy, J.: *Swarm Intelligence*. The Morgan Kaufmann Series in Evolutionary Computation (2001)
4. Nieto, J. G., Polo, G. J. L.: *Algoritmos basados en cúmulos de partículas para la resolución de problemas complejos*. (2006)
5. Kennedy, J. F., Kennedy, J., Eberhart, R. C., Shi, Y.: *Swarm intelligence*. Morgan Kaufmann (2001)
6. Sancho Caparini, F.: *Swarm Inteligencia. PSO: Optimización por enjambres de Partículas*, Sevilla, Dpto Cienc. Comput. E Intel. Artif. Univ. Sevilla (2015)
7. Hillier, F. S. L., Hillier, G. J. F. S., Lieberman, G. J.: *Introducción a la Investigación de Operaciones*. McGraw-Hill (1989)
8. López Gutiérrez, N., Albelo Martínez, M., del Valle Cruz, A., Ruíz de Zárate del Cueto, J.: *Elementos de Álgebra lineal y Programación lineal*. 2da ed. La Habana: Felix Varela (2008)
9. Eberhart, R. C., Shi, Y.: Comparison between genetic algorithms and particle swarm optimization. In: *International Conference on Evolutionary Programming*, pp. 611–616 (1998)
10. Gudise, V. G., Venayagamoorthy, G. K.: Comparison of particle swarm optimization and backpropagation as training algorithms for neural networks. In: *Swarm Intelligence Symposium, SIS'03, Proceedings of the 2003 IEEE*, pp. 110–117 (2003)
11. Mañas, J. A.: *Análisis de Algoritmos: Complejidad*. Disponible en: <http://www.lab.dit.upm.es/~lprg/material/apuntes/o/index.html> [Accedido: 04-jun-2017]
12. Ruz Valenzuela, V.: *Manual Análisis de Algoritmos* (2003)
13. Shi, Y., Eberhart, R. C.: Parameter selection in particle swarm optimization. In: *International Conference on Evolutionary Programming*, pp. 591–600 (1998)
14. Kowalski, P. A., Lukasik, S.: Experimental study of selected parameters of the krill herd algorithm. In: *Intelligent Systems' 2014*, Springer, pp. 473–485 (2015)