

# ARTIFICIAL INTELLIGENCE EFFECTIVENESS IN JOB SHOP ENVIRONMENTS

## EFFECTIVIDAD DE LA INTELIGENCIA ARTIFICIAL EN AMBIENTES JOB SHOP

OMAR CASTRILLON

*Departamento de Ingeniería Industrial, Universidad Nacional de Colombia, Bogotá, Colombia. odcastrillong@unal.edu.co*

WILLIAM SARACHE

*Departamento de Ingeniería Industrial, Universidad Nacional de Colombia, Bogotá, Colombia. wasarachec@unal.edu.co*

JAIME GIRALDO

*Departamento de Ingeniería Industrial, Universidad Nacional de Colombia, Bogotá, Colombia. jaiagiraldog@unal.edu.co*

Received for review December 3<sup>rd</sup>, 2009; accepted March 11<sup>th</sup>, 2011; final version March 29<sup>th</sup>, 2011

**ABSTRACT:** The aim of this paper is to define a new methodology that allows the comparison of the effectiveness among some of the major artificial intelligence techniques (random technique, taboo search, data mining, evolutionary algorithms). This methodology is applied in the sequencing production process in job shop environments, in a problem with  $N$  orders, and  $M$  machines, where each of the orders must pass through every machine regardless of its turn. These techniques are measured by the variables of total makespan time, total idle time, and machine utilization percentage. Initially, a theoretical review was conducted and showed the usefulness and effectiveness of artificial intelligence in the sequencing production processes. Subsequently and based on the experiments presented, the obtained results showed that these techniques have an effectiveness higher than 95%, with a confidence interval of 99.5% measured by the variables under study.

**KEY WORDS:** Makespan time, idle time, evolutionary algorithms, taboo search, data mining, random techniques

**RESUMEN:** El objetivo del presente trabajo, es definir una nueva metodología la cual permita comparar la efectividad de algunas de las principales técnicas de inteligencia artificial (aleatorias, búsqueda tabú, minería de datos, algoritmos evolutivos). Esta metodología es aplicada en los procesos de secuenciación de la producción en ambientes job shop, en un problema con  $N$  pedidos y  $M$  máquinas, donde cada uno de los pedidos debe pasar por todas las máquinas sin importar el orden. Estas técnicas son medidas en las variables tiempo total de proceso, tiempo total muerto y porcentaje de utilización de las máquinas. Inicialmente, una revisión teórica fue realizada, esta muestra la utilidad y efectividad de la inteligencia artificial en los procesos de secuenciación de la producción. Posteriormente y con base en la experimentación planteada, los resultados obtenidos, muestran que estas técnicas presentan una efectividad superior al 95%, con un intervalo de confiabilidad del 99.5% medido en las variables objeto de estudio.

**PALABRAS CLAVE:** Tiempo de proceso, tiempo muerto, algoritmos evolutivos, búsqueda tabú, minería de datos, técnicas aleatorias

### 1. INTRODUCTION

The sequencing production in job shop environments is perhaps one of the most difficult combinatorial optimization problems to solve due to its great complexity, its dynamism, the variety of its products, its processes, its levels of production, and its difficulties [1]. These problems, of real programming, are quite complex and difficult to solve by conventional optimization techniques, especially when there is a wide array of products characterized by a combinatorial optimization subject to very complex constraints [2].

In these systems, the analytical methodologies, like priority rules and heuristics, are not enough. The effectiveness of these techniques depends on: the makespan time, system initial load, system balance, and delivery times, with no existing rule to guarantee the maximization of the proposed objective. In many cases, it is necessary to use more efficient methodologies, like those based on artificial intelligence (AI) techniques [3] to achieve the effectiveness that industry expects.

Among these techniques, there are expert systems defined as intelligent applications capable of imitating

the complex processes of a human mind. These systems achieve feasible quality solutions in little time, solving programming or reprogramming job problems in an effective way [4]. Basically, these expert systems are made up of the following parts: a) knowledge base (KB), b) inference mechanisms (IMs), c) consistency control (CC), and d) user interface [5].

These systems, as genetic algorithms, build new solutions from existing solutions. Genetic algorithms, understood as adaptive search optimization methods, based on crossover and mutation genetic operators, can also be called niche antibodies [6]. These AI techniques have shown great efficiency in solving different flexible programming—job shop (FJSP) problems [7], such as the distributed flexible job shop programming (DFJSP) [8] that can solve both FJSP and DFJS problems with excellent results [9].

Hybrid algorithms may be mentioned among the different AI techniques that allow the combination of deterministic and stochastic parameters, complementing the different methodologies which improve the quality and efficiency of the solutions found [10] with greater effectiveness [11]. Hybrid genetic algorithms (HGA) and genetic algorithms (GA), have been useful in solving problems in no-wait job shops (NWJS); an NP-Hard problem with complex constraints [12].

Similarly, the HGA can be used in complete local search methods with limited memory (CLLM). This heuristic allows for integration and schedule changes in sequencing, solving the JSSP problem in an effective way, which almost always allows the finding of an optimal solution [13]. Hybrid genetic algorithms (HGA), GA, global scheduling characteristics prediction model (GSCPM) [14], genetic bottleneck [15] heuristics, and meta heuristics [16], among others [17], have also been useful in troubleshooting job shop scheduling (JSP), using fuzzy logic controllers (FLC), with very good results, especially in large problems [18].

These hybrid algorithms can be adapted to the search area (GSA) [2] and can also be used to find the lower limit of the optimal solution [19]. The previous algorithms have been linked to heuristics techniques with good performance and flexibility in solving problems [20], demonstrating genetic programming effectiveness [21] and its various techniques, such as

gene expression programming (GEP), that has shown better performance than conventional models and even dynamic models. [22].

Optimization through ant colonies (ACO), can be cited among other evolutionary techniques based on AI [23] which tries to emulate the behavior of the ants to solve combinatorial optimization problems [24].

Ant colony optimization is a good algorithm for solving job shop problems, as it produces quick results, and it is easy to implement because it is not as robust as a GA [25, 26].

Similarly, fuzzy logic has become a support system for the decisions of programming and /or rescheduling in parallel machines in presence of uncertain disturbances [27]. This technique has proven to be highly effective in manufacturing systems with flexible routing, and the technique has demonstrated very good results in the subsequent development of adaptive approaches [28]. Some authors [29] have developed fuzzy logic techniques combined with neural techniques, achieving the stimulation of machine production more efficiently than a human expert— improving performance and controlling workloads, punctuality, and waiting times.

On the other hand, techniques based on particle swarm optimization (PSO) and its modified methods introducing genetic operators into genetic particle swarm optimization (GPSO) represent a study based on the optimization method inspired by the behavior of birds, proving to be efficient in continuous multidimensional problems, and in research problems of operations such as identification processes and controllers' projects [30].

Other authors [31] have proposed the combination of the above mentioned techniques with nested simulation techniques in FJSP problems. Although the results have not been the best, it is important to emphasize that effective solutions have been found to this kind of problems, similar to those found with hybrid particle swarm optimization (HPSO) [32] using algorithms of binary particle swarms optimization (BPSO) [33].

Among other intelligent techniques, there is the taboo search, an effective technique, capable of producing good solutions with a short calculation time [34]. From this technique, other techniques such as simulated

annealing (SA), variable neighborhood search (VNS) [35] and relaxation solutions [36] have been derived. These techniques might potentiate strong features in both methods of heuristic search in favor of the speed in which solutions are found in [37], displaying the effectiveness of this technique in terms of quality and calculating times [38]. The above techniques have been successfully applied in solving transmission problems in working environments, even allowing the finding of optimal solutions in reasonable computing time [39], minimizing the delay in a workshop with various routing alternatives [40].

Similarly, artificial neural networks (ANN) can also be considered, which have made it possible to automatically detect the complex nonlinear relationships between work characteristics and the makespan time, providing more accurate estimates than traditional methods [41]. Although they have had certain limitations, these networks have been applied in solving job shop planning problems in the solution of problems with a deterministic time-varying range over a demand pattern within a fixed planning horizon [42].

Artificial neural network models, properly utilized, can contribute greatly to simplify the sequencing and programming of decision-making processes. Furthermore, these models are very useful as a tool for optimization packet programming in the simulation of partial solutions, such as the GA-based programs. The meta models, built from these networks, offer important advantages regarding time, consumption, and simplicity. Their performance is remarkably fast compared to the time required to operate conventional simulation packages [43], when evaluated using multiple criteria in different situations.

A better performance in the solution of JSSP problems has been achieved using the hybrid approach between neural networks and genetic algorithms, which is proposed by some authors, and includes adaptive neural network with heuristics, an approach in which neurons may represent treatment restrictions and also may solve restriction conflicts.

Clone selection is equally important among the theories of AI. This theory states that when an antigen enters the system, a (B) cell is activated and cloned antibodies begin to reproduce. The number of clones

is determined by the affinity between the antigen and antibody. Specifically in job shop systems, the scheduling problem would be the antigen, and the candidates for solution would be the B cells, which are chosen to obtain the final solutions or antibodies. These final solutions can be evolved until the desired solution is reached [44].

In the same manner, other techniques can be cited, like the multi-cycle techniques, MC, where the primary objective is to minimize the sum of installation costs, the work in process, the finished products, and the inventory costs, while demand is achieved without backorders, based on the assumption that the cycle time of each product is a whole multiple of a base period.

The MC method decomposes the problem into three sub-problems: an allocation sub-problem, a sequence and lot size sub-problem, and lot size and the programming sub-problem. The first two sub-problems are solved by a heuristic while the third is solved by optimization in mathematical programming. In this method, the quality of the results depends on the structure of the problem [45].

It is important to highlight that in the search for excellent solutions, besides emulating reality using AI techniques, it is necessary to translate these results into computer simulations, using some heuristic algorithms to evolve the given solution [46]. It is also important to note that it is not enough to find a solution through sequencing problems in job shop environments.

The effectiveness of this solution is closely related to how this solution is integrated into a Gantt chart [47]. In the following jobs [48-60], there is a detailed explanation of these artificial intelligence techniques and their use in other areas.

Finally, the fundamental objective of this paper is to define a new methodology for comparing and showing the effectiveness of the optimal solution of some of the main evolutionary AI techniques (random techniques, taboo search, data mining, and evolutionary algorithms). This methodology is applied in the production sequencing process in job shop environments, in a problem with N orders and M machines, where each order must go through all the machines regardless of its turn.

## 2. METHODOLOGY

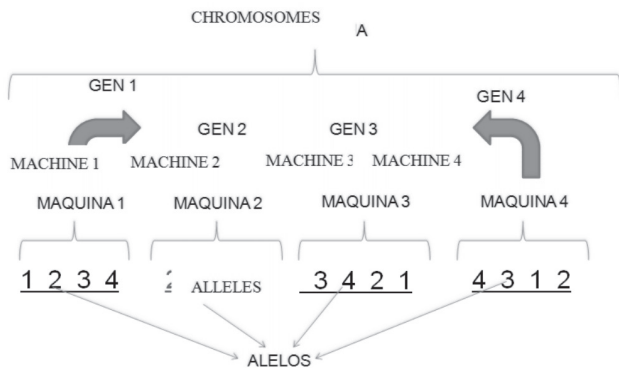
**Step 1: Representation.** A JSSP<sub>NXM</sub> problem [23] can be represented by a NXM two dimensional matrix, see Table 1:

**Table 1.** Structure of the JSSP<sub>NXM</sub> problem.

	<i>ord<sub>1</sub></i>	<i>ord<sub>2</sub></i>	...	<i>ord<sub>N</sub></i>
<i>C<sub>1</sub></i>	M			
..				
<i>C<sub>M</sub></i>				

*C<sub>1</sub>* represents machine I; *Ord<sub>J</sub>*, the order; and M is the process time of machine I for the J order.

**Step 2: Codification.** The initial solutions of the problem under study are coded using the structure shown in Figure 1. In each coded solution of chromosome shape, the genes represent the position in which an order is served by each of the different machines. The genes' alleles represent the order number.



**Figure 1.** Solutions codification, adapted from [48]

**Step 3: Generation of solutions.** An evolution of the initial solution is begun by using AI techniques (random, taboo, data mining, and evolutionary algorithms) until an optimal solution is reached or until a certain number of iterations have been done, with no improvement in the last solution found. The methodologies used to evolve an initial solution (valid random solution, according to the structure of Step 2, called codification) and to generate new solutions using each of the AI techniques mentioned here, are described in the bibliographic references[48, 49, 50].

**Step 4: Gantt chart.** In this step, the newly proposed solution can be changed, and is carried out as follows: a) the first allele of each gene is sequenced in the Gantt chart, in the machine that represents the gene, subject to the following restrictions: 1) An order cannot be processed simultaneously by two or more machines, so if the same order can be sequenced simultaneously in two or more different machines, the order is shifted to the right in the Gantt chart, in one or several of these machines. This last step is achieved by introducing an idle time at the beginning of the makespan time of each machine, where the order is to be processed. 2) Each request must be handled on one machine until its process is completed. 3) Any other restrictions set by some of the authors must be taken into account [60]. b) The sequencing of each gene allele is continued in its respective machines, keeping the constrains established in step a. c) In the case of having an idle time large enough to store an order in the Gantt chart between the starting processing time and the final processing time in each machine, the order is sequenced in this time interval provided that the constrains in step a are kept.

The previous Gantt chart construction technique, involves a mutation in the solution set by the AI technique used. This mutation contributes to optimize the solution under analysis.

**Step 5: Quality.** The effectiveness of each solution found is set based on equations 1, 2, and 3. The equations mentioned in this step allow the calculation of the value of the variables: the makespan (Eq. 1), the idle time (Eq. 2), and the percentage of machine utilization (Eq. 3).

$$Makespan = \text{Min} \sum_{i=1}^N \sum_{j=1}^M P_j \tag{1}$$

$$Idle = \text{Min} \sum_{j=1}^m f_j \tag{2}$$

$$Utilization = \frac{(Makespan * M - Idle)}{Makespan * M} * 100 \tag{3}$$

Since equations 2 and 3 are a direct result of the first equation, the fundamental objective is to minimize the first equation: fitness. In the previous equations, N represents the number of orders, M represents the number

of machines,  $P_{ij}$  is the time of processing of orders  $i$  in machine  $j$ , and  $f_j$  is the total idle time in machine  $j$ .

**Step 6:** Response. An estimate regarding the optimum for each finding and for each problem is shown on a response surface. A response surface must be built for each of the techniques. It is not always possible to calculate an optimal solution for each problem due to the great number of possibilities. Therefore, it is necessary to make an optimal estimate, which assumes that the optimal makespan time is never less than the longest process time in machine  $M_j$ , taking into account an idle time of zero in the same machine. In other words, the machine with the longest processing time should be completely busy.

**Step 7:** Consistency. The greatest (JSSP<sub>20x16</sub>) problem proposed is solved by using each of the techniques, a certain number of times (treatments), in order to ensure the consistency of the proposed methodology, and also to determine whether or not there is a variation among the techniques: random, data mining, taboo search, or evolutionary algorithms. The results obtained in regard to the fitness function are taken

as a reference in each of the treatments. Later on, a variance analysis under the model  $y_i = \mu + T_i + \varepsilon_i$  is done to determine if the treatments are statistically similar or different. In this variance analysis,  $y_i$  represents the response variables;  $T_i$  the effects caused by treatment  $i$ ; and  $\varepsilon_i$ , the  $i$ th experimental error. The information collected must meet the requirements of independence and normality in order to allow the use of the required tests.

### 3. EXPERIMENTATION

The techniques under study are analyzed in 160 general JSSP<sub>NXM</sub> problems proposed by the authors, for  $10 \leq N \leq 20$ , and  $1 \leq M \leq 16$  to determine the effectiveness of the proposed methodology. Table 2 shows the problems analyzed.

One hundred and sixty matrices, formed by the ranges:  $P(1: I, 1: J)$  for  $1 \leq I \leq 16$ , and  $10 \leq J \leq 20$ , can be derived from Table 2, beginning at position (1,1) for  $N \geq 10$ . Each of the above ranges represents a JSSP<sub>NXM</sub> sequencing problem, which will be analyzed using the different AI techniques in this paper.

**Table 2.** JSSP problems analysis.

M / M		ORDERS = N																			
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
MACHINES = M	1	5	9	2	2	2	1	2	5	6	7	8	8	7	7	6	5	4	3	2	1
	2	9	3	4	4	3	2	3	6	9	9	9	4	4	5	6	3	2	1	3	7
	3	7	4	6	5	4	3	4	9	8	2	3	4	4	3	2	5	6	7	8	9
	4	3	5	8	7	5	4	5	8	7	3	5	1	2	6	7	8	2	7	9	2
	5	2	1	9	8	6	5	6	3	2	4	6	5	4	4	3	2	7	6	5	4
	6	7	4	5	9	7	6	2	5	2	4	7	1	8	9	5	4	3	2	6	7
	7	8	6	2	2	7	7	9	6	2	2	2	9	9	8	2	7	6	5	4	2
	8	9	8	5	5	8	4	8	6	3	3	5	1	4	2	3	4	5	6	1	5
	9	9	2	7	8	9	9	6	4	2	6	6	9	9	9	8	1	2	7	7	6
	10	3	4	2	9	9	9	7	3	3	2	7	8	8	4	3	2	8	9	1	2
	11	5	6	8	9	8	2	2	6	6	7	5	1	9	8	7	2	5	4	3	2
	12	4	3	9	1	7	3	6	6	5	8	8	6	5	4	3	2	7	8	9	3
	13	9	9	9	8	4	4	9	9	9	9	9	9	8	7	6	5	4	3	2	1
	14	8	9	9	8	6	5	9	9	8	3	6	7	6	5	2	5	4	3	2	2
	15	9	9	9	8	6	9	8	8	8	9	8	9	9	8	9	9	8	9	9	9
	16	8	9	9	9	4	2	9	9	9	3	5	1	9	5	7	4	6	2	2	8

### 4. RESULTS AND DISCUSSIONS

Steps 1 to 5. The response surfaces are set for each of the IA techniques (shown in Figures 2-5) as a result of Steps 1-5, which show the effectiveness of each

of the IA techniques. On average, these response surfaces show that in the worst case scenario, the AI techniques employed are 95% near to the optimal solution.

On the other hand, as expressed in the previous section, it is not possible in all cases to calculate the optimal solution within a reasonable computation time, given the large number of possibilities for the problems  $N!^M$  ( $1.5066 * 10294$  possibilities for  $N = 20$  and  $M = 16$ ).

Therefore, an optimal estimate must be considered in order to calculate the response surfaces, assuming that, in the best case, the optimal makespan time is never less than the longest processing time on an  $M_i$  machine having an idle time of zero. In other words, the machine with the longest processing time is fully busy.

Figure 6. shows the optimal solution for a JSSP 20X16. problem. Note that on machine 15 there is no idle time.

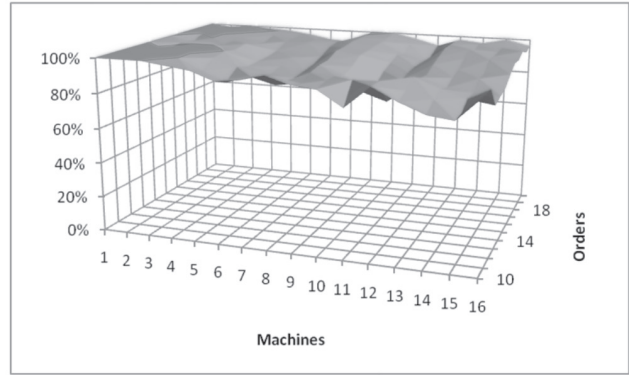


Figure 4. Data mining.

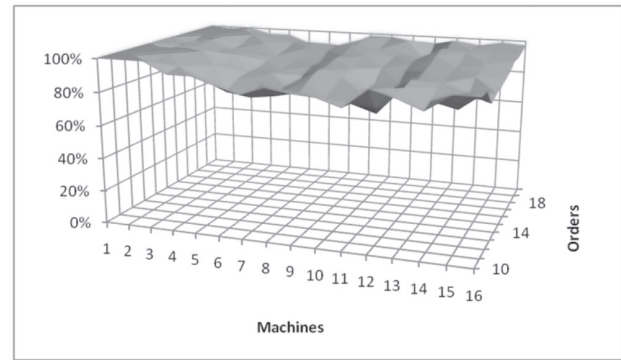


Figure 5. Genetic alg.

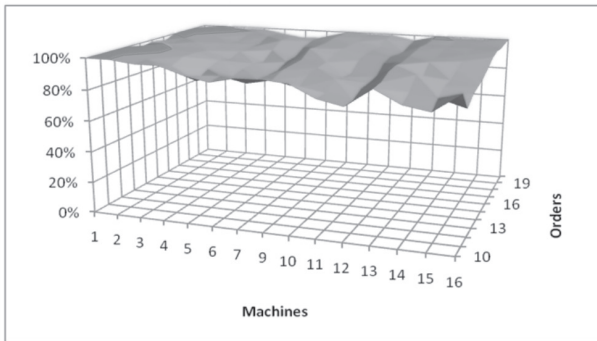


Figure 2. Random alg.

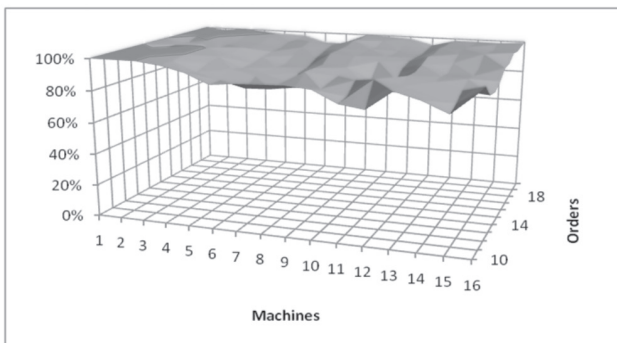


Figure 3. Taboo search

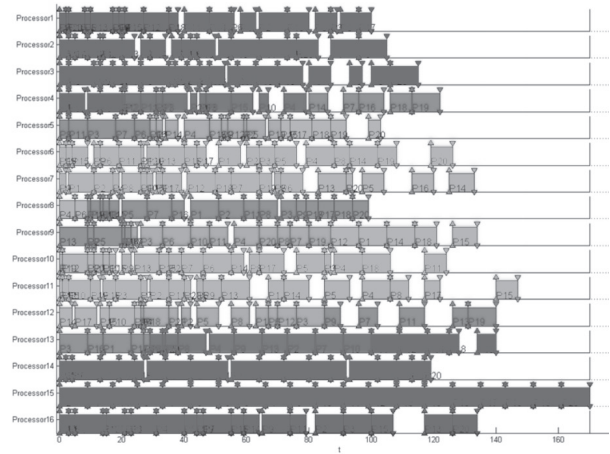


Figure 6. Optimal Solution JSSP<sub>20X16</sub>, makespan = 170, T<sub>I</sub> = 950

**Step 6.** The greatest JSSP20X16 problem is solved using the different proposed techniques (treatments) 10 consecutive times (replications), and taking each solution found as a reference. The respective analysis of variance is built based on the results obtained in the variable total processing time (makespan) and total down time (idle, Tables 3-6).

Table 3. Replications JSSP<sub>20X16</sub> processing time.

Processing ime	Replications										Sum
JSSP <sub>20X16</sub>	1	2	3	4	5	6	7	8	9	10	
Random	173	170	171	171	173	172	172	171	174	173	1720
Taboo	174	174	172	175	170	171	172	175	173	170	1726
Data mining	175	175	178	176	171	170	170	178	181	172	1746
Genetics	179	173	170	180	174	176	175	176	177	170	1750

**Table 4.** Anova processing times JSSP<sub>20X16</sub>.

Variation source	GL	SC	CM	Fcal	F Tab
<b>Sum C Total</b>		341.90			
<b>Treatment</b>	3.00	65.10	21.70	2.82	2.87
<b>Exper Error</b>	36.00	276.80	7.69		
<b>Total</b>	39.00	341.90	29.39		

THE MODEL IS NOT SIGNIFICANT

**Table 5.** Replications JSSP20X16. Idle time.

Idle time JSSP <sub>20X16</sub>	Replications										Sum
	1	2	3	4	5	6	7	8	9	10	
<b>Random</b>	998	950	966	966	998	982	982	966	1014	998	<b>9820</b>
<b>Taboo</b>	1014	1014	982	1030	950	966	982	1030	998	950	<b>9916</b>
<b>Data mining</b>	1030	1030	1078	1046	950	950	950	1078	1126	982	<b>10220</b>
<b>Genetics</b>	1094	998	950	1110	1014	1046	1030	1046	1062	950	<b>10300</b>

**Table 6.** ANOVA idle time JSSP<sub>20X16</sub>.

Variation source	GL	SC	CM	Fcal	F Tab
<b>Sum C Total</b>		89081.60			
<b>Treatment</b>	3.00	16147.20	5382.40	2.66	2.87
<b>Exper Error</b>	36.00	72934.40	2025.96		
<b>Total</b>	39.00	89081.60	7408.36		

THE MODEL IS NOT SIGNIFICANT

The variance analysis in Tables 4 and 6 show that the value of the two F distributions, calculated at (0.05, 3, and 36), is less than the value of the same real F distributions. Consequently, it can be said that the results are statistically equal, with a reliability of 99.5%.

Finally, it is stated that at first instance, an analysis of the response surfaces and processing replicas (Figures 2 to 5, and Tables 3 and 5) shows a greater effectiveness in random techniques (95.4%), followed by taboo search techniques (94.93%), data mining (94.85%), and genetic algorithms (93.1%). But the above variance analysis (Tables 4 and 6) show that, in fact, there is no significant difference in the results found using the different AI techniques.

## 5. CONCLUSIONS

At the time of constructing the Gantt chart, the mutation generated by the different AI technique solutions reduces the makespan time when attempting to sequence the orders at idle time on each machine.

The results presented by the different techniques are of equal value, according to the different variance analysis, under a 99.5% significance level. In this sense, and taking the average results obtained in the different response surfaces as the reference, it is possible to conclude that the different techniques used allow for solutions with an effectiveness superior to 95% in regard to the optimal solution in the variables under analysis.

## 6. ACKNOWLEDGMENTS

A special thanks to Natalia Ruiz Mazuera, an industrial engineering student, for her keen collaboration.

## 7. REFERENCES

- [1] XIANG, W. AND LEE, H. Ant colony intelligence in multi-agent dynamic manufacturing scheduling. *Engineering Applications of Artificial Intelligence.*, 21 - 1, 73 – 85, 2008.
- [2] WATANABE, M., IDA, K. AND GEN, M. A genetic algorithm with modified crossover operator and search area adaptation for the job-shop scheduling problem. *Computers & Industrial Engineering.*, 48 - 4, 743 – 752, 2005.
- [3] PRIORE, P., PINO, R. AND GÓMEZ, A. Secuenciación de sistemas de tipo job shop mediante aprendizaje automático. XII congreso, la gestión de la diversidad., Logroño España, 1, 1281 – 1288, Junio 1999.
- [4] Li, H., Li, Z., Li, L., Et al. A production rescheduling expert simulation system. *European Journal of Operational Research.*, 124 - 2, 283 - 293, 2000.
- [5] CASTRILLÓN, O., GIRALDO, J. AND SARACHE, W. Aplicación de un algoritmo evolutivo en la solución de problemas Job Shop - Open Shop. *Información tecnológica.* 22 -1, 83 – 92, 2011.
- [6] LUH, G. AND CHUEH, C. A multi-modal immune algorithm for the job-shop scheduling problem. *Information Sciences.*, 179 -10, 1516 – 1532, 2009.
- [7] PEZZELLA, F., MORGANTI, G. AND CIASCETTI, G. A genetic algorithm for the flexible Job-shop scheduling problem. *Computers & Operations Research.*, 35 -10, 3202 – 3212, 2008.
- [8] ESSAFI, I., MATI, Y. AND DAUZÈRE, S. A genetic local search algorithm for minimizing total weighted tardiness in the job-shop scheduling problem. *Computers & Operations Research.*, 35 -8, 2599 – 2616, 2008.

- [9] DE GIOVANNI, L. AND PEZELLA, F. An Improved Genetic Algorithm for the Distributed and Flexible Job-shop Scheduling problem. *European Journal of Operational Research.*, 200 -1, 395 -408, 2010.
- [10] TAVAKKOLI, R., JOLAI, F., VAZIRI, F., Et al. A hybrid method for solving stochastic job shop scheduling problems. *Applied Mathematics and Computation.*, 170 -1, 185 – 206, 2005.
- [11] HUANG, K. And LIAO, C. Ant colony optimization combined with taboo search for the job shop scheduling problem. *Computers & Operations Research.*, 35 -4, 1030 – 1046 , 2008.
- [12] CHAO, J. AND CHIANG, H. A hybrid genetic algorithm for no-wait job shop scheduling problems. *Expert Systems with Applications.*, 36 -3-2, 5800 - 5806, 2009.
- [13] ZHU, J., LI, X. AND WANG, Q. Complete local search with limited memory algorithm for no-wait job shops to minimize makespan. *European Journal of Operational Research.*, 198 -2, 378 – 386, 2009.
- [14] LIU, M., HAO, J. AND WU, C. A prediction based iterative decomposition algorithm for scheduling large-scale job shops. *Mathematical and Computer Modelling.*, 47 -3-4, 411 – 421, 2008.
- [15] ZHANG, R. AND WU, C. A hybrid immune simulated annealing algorithm for the job shop scheduling problem. *Applied Soft Computing.*, 10 -1, 79-89, 2010.
- [16] FIGLALI, N., OZKALE, C. AND ENGIN, O. Investigation Ant of System parameter interactions by using design of experiments for job-shop scheduling problems. *Computers & Industrial Engineering.*, 56 -2, 538 – 559, 2009.
- [17] CHAN, F., WONG, T. AND CHAN, L. An evolutionary algorithm for assembly job shop with part sharing. *Computers & Industrial Engineering.*, 57 -3, 641-651, 2009.
- [18] YUN, Y. Genetic algorithm with fuzzy logic controller for preemptive and non-preemptive job-shop scheduling problems. *Computers & industrial engineering.*, 43 -3, 623-644, 2002.
- [19] MANIKAS, A. AND CHANG, Y. Multi-criteria sequence-dependent job shop scheduling using genetic algorithms. *Computers & Industrial Engineering.*, 56 -1, 179 – 185, 2009.
- [20] ZHOU, H., CHEUNG, W. AND LEUNG, L. Minimizing weighted tardiness of job-shop scheduling using a hybrid genetic algorithm. *European Journal of Operational Research.*, 194 -3, 637– 649, 2009.
- [21] TAY, J. AND HO, N. Evolving dispatching rules using genetic programming for solving multi-objective flexible job-shop problems. *Computers & Industrial Engineering.*, 54 -3, 453 – 473, 2008.
- [22] BAYKASOĞ˘LU, A. AND GOÇKEN, M. Gene expression programming based due date assignment in a simulated job shop. *Expert Systems with Applications.*, 36 -10, 12143 – 12150, 2009.
- [23] CASTRILLÓN, O., GIRALDO, J. AND SARACHE, W. Job shop methodology based on an ant colony. *Revista Dyna*, 76 -159, 177-184, 2009.
- [24] ROSSI, A. AND BOSCHI, E. A hybrid heuristic to solve the parallel machines job-shop scheduling problem. *Advances in Engineering Software.*, 40 -2, 118 – 127, 2009.
- [25] HEINOEN, J. AND PETTERSSON, F. Hybrid ant colony optimization and visibility studies applied to a job-shop scheduling problem. *Applied Mathematics and Computation.*, 187 -2, 989 – 998, 2007.
- [26] BAOJIANG, Z. AND SHIYONG, L. Ant colony optimization algorithm and its application to Neuro-Fuzzy controller design. *Journal of Systems Engineering and Electronics.*, 18 -3, 603 – 610, 2007.
- [27] PETROVIC, D. AND DUENAS, A. A fuzzy logic based production scheduling/rescheduling in the presence of uncertain disruptions. *Fuzzy Sets and Systems.*, 157 -16, 2273 – 2285, 2006.
- [28] BILGE, U., FIRAT, M. AND ALBEY, E. A parametric fuzzy logic approach to dynamic part routing under full routing flexibility. *Computers & Industrial Engineering.*, 55 -1, 15-33 ,2008.
- [29] BONFATTI, M., CARIDI, M. AND SHIAVINA, L. A fuzzy model for load-oriented manufacturing control. *International Journal of Production Economics.*, 104 -2, 502 – 513, 2006.
- [30] NIU, Q., JIAO, B. AND GU, X. Particle swarm optimization combined with genetic operators for job shop scheduling problem with fuzzy processing time. *Applied Mathematics and Computation.*, 205 -1, 148 – 158, 2008.
- [31] XIA, W. AND WU, Z. An effective hybrid optimization approach for multi-objective flexible job-shop scheduling problems. *Computers & Industrial Engineering.*, 48 -2, 409 – 425, 2005.
- [32] FAN, K., ZHANG, R. AND XIA, G. Solving a Class of Job-Shop Scheduling Problem based on Improved BPSO Algorithm. *Systems Engineering - Theory & Practice.*, 27 -11, 111 – 117, 2007.
- [33] SHA, D. AND HSU, C. A hybrid particle swarm optimization for job shop scheduling problem. *Computers & Industrial Engineering.*, 51 -4, 791 – 808, 2006.



- [34] HURINK, J. AND KNUST, S. Tabu search algorithms for job-shop problems with a single transport robot. *European Journal of Operational Research.*, 162 -1, 99 – 111, 2005.
- [35] ADIBI, M. AND ZANDIEH, M. Multi-objective scheduling of dynamic job shop using variable neighborhood search. *Expert Systems with Applications.*, 37 -1, 282-287, 2010.
- [36] BAPTISTE, P., FLAMINI, M. AND SOURD, F. Lagrangian bounds for just-in-time job-shop scheduling. *Computers & Operations Research.*, 35 -3, 906 – 915, 2008.
- [37] ZHANG, C., LI, P., RAO, Y., Et al. A very fast TS/SA algorithm for the job shop scheduling problem. *Computers & Operations Research*, 35 -1, 282 – 294, 2008.
- [38] KIS, T. Job-shop scheduling with processing alternatives. *European Journal of Operational Research.*, 151 -2, 307 – 332, 2003.
- [39] BUSCHER, U. AND SHEN, L. An integrated tabu search algorithm for the lot streaming problem in job shops. *European Journal of Operational Research.*, 199 -2, 385 – 399, 2009.
- [40] DRÓTOS, M., ERDOS, G. AND KIS, T. Computing lower and upper bounds for a large-scale industrial job shop scheduling problem. *European Journal of Operational Research.*, 197 -1, 296 – 306, 2009.
- [41] LI, S., LIU, Y. AND XU, Y. A GA-based NN approach for makespan estimation. *Applied Mathematics and Computation.*, 185 -2, 1003 -1014, 2007.
- [42] FENG, S., LI, L., CEN, L., Et al. Using MLP networks to design a production scheduling system. *Computers & Operations Research.*, 30 -6, 821 – 832, 2003.
- [43] FONSECA, D., NAVARESE, D. AND MOYNIHAB, G. Simulation metamodeling through artificial neural networks. *Engineering Applications of Artificial Intelligence.*, 16 -3, 177 – 183, 2003.
- [44] YANG, J., SUN, L., LEE, H., Et al. Clonal Selection Based Memetic Algorithm for Job Shop Scheduling Problems. *Journal of Bionic Engineering.*, 5 -2, 111 – 119, 2008.
- [45] AKCORA, E., GRASMAN, E. AND SAYGIN, C. A job shop scheduling heuristic for varying reward structures. *Mathematical and Computer Modelling.*, 42 -11-12, 1163 – 1174, 2005.
- [46] XING, L., CHEN, Y. AND YANG, K. Multi-objective flexible job shop schedule: Design and evaluation by simulation modeling. *Applied Soft Computing.*, 9 -1, 362 -376, 2009.
- [47] JIA, H., FUH, J., NEE, A., Et al. Integration of genetic algorithm and Gantt chart for job shop scheduling in distributed manufacturing systems. *Computers & Industrial Engineering.*, 53 -2, 313 – 320, 2007.
- [48] CASTRILLÓN, O., GIRALDO J. AND SARACHE, W. Técnicas inteligentes y estocásticas en scheduling. Un enfoque en la producción y las operaciones. Unibiblios, Bogotá, 2010.
- [49] CASTRILLÓN, O., GIRALDO, J. AND SARACHE, W. Técnicas de programación de la producción. Aplicación en ambientes Job shop. Unibiblios, Bogotá, 2010.
- [50] BECERRA, F., CÁRDENAS, D., CASTRILLÓN, O., Et al. Gestión de la producción: Una aproximación conceptual. Unibiblios, Bogotá, 2008.
- [51] TAMAYO, J. AND CASTRILLON, O. Aplicación del algoritmo de Dijkstra en la interacción del comercio colaborativo en las empresas logísticas. *Revista Espacios.* 29 -1, 19, 2008.
- [52] CASTRILLÓN, O., SARACHE, W. AND GIRALDO, J. Solución de un problema Job shop por medio de un agente Inteligente. *Ingeniería y Ciencia*, 5 -10, 75-92, 2009.
- [53] CASTRILLON, O., SARACHE, W. AND GIRALDO, J. Design of a hyperheuristic for production scheduling in job shop environments. *Revista Chilena de Ingeniería.* 18 -10, 203-214, 2010.
- [54] GIRALDO, J., SARACHE, W. AND CASTRILLON, O. Metodología integral soportada en simulación para el mejoramiento de sistemas de producción Job Shop. *Aplicaciones en pymes metalmeccánica. Ingeniería e investigación*, 30 -1, 97-106, 2010.
- [55] SARACHE, W., CASTRILLON, O. GONZALES, G. AND FOLLECO, A. Una aplicación multicriterio para la decisión de reemplazar un equipo. *Ingeniería y Desarrollo* 25, 80-98, 2009.
- [56] WILLIAM, S., CASTRILLON, O. AND ORTIZ, L. Selección de proveedores una aproximación al estado del arte, *Cuadernos de administración.* 22 -38, 145-167, 2009.
- [57] TAMAYO, J., HIGUITA, J. AND CASTRILLON, O. Funcionalidades del comercio colaborativo en las empresas logísticas y su decisión de tercerización, *Cuadernos de Administración.* 23 -41, 81-105, 2010.
- [58] DORIGO, M. AND STUTZLE T. *Ant Colony Optimization.* Massachusetts Institute Technologic. 2004. USA.
- [59] DORIGO, M., DI CARO, G. AND SAMPELS M. *Ant Algorithms.* Springer Verlag. New York. 2002. USA.
- [60] ROHER, M. Simulation of manufacturing and material Handling Systems. En: *Handbook of simulation.* Jhon Wiley & Son, Atlanta, 2000.