

# Modeling Strategy by Adaptive Genetic Algorithm for Production Reactive Scheduling with Simultaneous use of Machines and AGVs

O. Morandin, Jr, E. R. R. Kato, D. S. Sanches, B. D. Muniz

Department of Computer Science  
Federal University of São Carlos (UFSCar)  
São Carlos, SP, Brazil

orides@dc.ufscar.br; kato@dc.ufscar.br; daniilo\_sanches@dc.ufscar.br; bruno\_muniz@dc.ufscar.br;

**Abstract**—The problem of production scheduling of manufacturing systems is characterized by the large number of possible solutions. Several researches have been using the Genetic Algorithms (GA) as a search method to solve this problem since these algorithms have the capacity of globally exploring the search space and find good solutions quickly. Since the performance of the GA is directly related to the choice of the parameters of genetic operators, and a bad choice can depreciate the performance, this paper proposes the use of Adaptive Genetic Algorithm to solve this kind of scheduling problem considering the machines and the Automated Guided Vehicles (AGVs). The aim of this paper is to get a good production reactive schedule in order to achieve a good makespan value in a low response obtaining time. The results of this paper were validated in large scenarios and compared with the results of two other approaches. These results are presented and discussed in this paper.

**Keywords**—scheduling, manufacturing systems, adaptive genetic algorithm, transportation systems, automated guided vehicle.

## I. INTRODUCTION

Manufacturing systems with shared resources such as machines and Transport Systems have been used for production automation and they allow flexibility consistent with the needs of a good share of manufactures in the market. This flexibility occurs because the machines can be used to several different operations and the products can be processed by several different machines. The problem is that this flexibility requires a considerable effort to define the production scheduling and it uses the available resources over time, in order to reach certain performance criteria.

One of the difficulties in production scheduling of a manufacturing system with shared resources is the product scheduling with simultaneous use of machines and transport systems. Among the existing transportation systems, the Automated Guided Vehicles (AGVs) stands out. AGVs are small and standalone vehicles, which move materials to and from value aggregate operations. Typically, products in a manufacturing system with shared resources visit different machines for different operations, requiring a transport system to transport the products among the machines at its processing route.

Most researches have treated the production planning and transport system problems as independent. Automated guided vehicles (AGVs), are the state-of-the-art, and are often used to facilitate automatic storage and retrieval systems (AS/RS). In this paper, we focus on the dispatching of AGVs in a flexible manufacturing system (FMS). [2].

These scheduling problems are classified as NP-Hard problems, what means there is a large concentration of computational effort that grows exponentially with the increase in the size of the problem [7]. In this context, several approaches for dealing with production scheduling problem indicates a very high efficiency in the use of Genetic Algorithms. However, it is reported that the simple GA often suffers from the troubles of premature convergence, difficulty in constructing fitness functions and parameter dependence [8]. So far, many researches have used an Adaptive Genetic Algorithm to solve the problem of premature convergence [1],[6], [7] and [8], but there is a need to address this problem together with the transport system.

Therefore, this paper proposes an Adaptive Genetic Algorithm based approach for production reactive scheduling of manufacturing systems with shared resources and simultaneous use of machines and AGVs in order to achieve good values of makespan and response time, the latter being an important feature when it comes to a reactive scheduling.

## II. LITERATURE REVIEW

Scheduling of manufacturing systems with shared resources has been extensively investigated over the last three decades, and it continues to attract the interest of both the academic and industrial sectors. Various types of scheduling problems are solved in different Manufacturing environments. A variety of algorithms are employed to obtain optimal or near optimal schedules. For this, some approaches using GA have been proposed to solve the scheduling problem.

Pongcharoen et al. [4] present a modified genetic algorithm for production scheduling for complex products with multiple levels of product structure and multiple resource constraints. Pongcharoen et al. [5] presents an experiment of this GA applied to a manufacturing of industrial goods, to identify

parameters values of the GA that produces the best results in a limited execution time.

In order to solve the premature convergence problem, adaptive genetic algorithm approaches have been used for scheduling problem. Zhang et al. [7] proposed an adaptive genetic algorithm with multiple operators for flowshop scheduling. Simulation results based on benchmarks demonstrate the effectiveness of AGA by contrast with traditional GAs. Yin et al. [8], presents an adaptive genetic algorithm for flowshop problem. The probability of crossover and mutation is dynamically adjusted according to the individual's fitness value. When compared to a basic GA, the effectiveness and efficiency of this adaptive genetic algorithm are better than those of the basic GA. Morandin et al. [1] use an adaptive genetic algorithm for production reactive scheduling of manufacturing systems having as performance criteria the minimum makespan and the response obtaining time. In their paper, the operator probabilities can be dynamically set according to the fitness value of the chromosomes.

There are several approaches making simultaneous use of machines and AGVs for products scheduling using Genetic Algorithms. This approach type has been presenting good makespan results. Jerald et al. [6] use an adaptive genetic algorithm for simultaneous scheduling of parts and AGVs for a particular type of FMS environment. The schedule obtained by adaptive genetic algorithm is compared to genetic algorithm, and experimental results have indicated that the proposed adaptive genetic approach is very effective. Reddy and Rao [2], addresses multiobjective scheduling problems in a flexible manufacturing environment using evolutionary algorithms. The authors made an attempt to consider simultaneously the machine and vehicle scheduling aspects in an FMS and addressed the combined problem for the minimization of makespan, mean flow time and mean tardiness objectives. Ulusoy et al. [3] addresses the problem of simultaneous scheduling of machines and a number of identical automated guided vehicles in a flexible manufacturing system so as to minimize the makespan. Sankar et al. [11] addresses the simultaneous scheduling of incoming jobs, machines, and vehicle dispatching in a flexible manufacturing system having a single device, an automated guided vehicle. The objective is to find an optimal sequence of incoming parts, which will reduce the waiting times due to blocking and starving of resources and deadheading times, resulting in overall minimization of makespan.

### III. PROBLEM DESCRIPTION

The problem of production scheduling on manufacturing systems with shared resources and simultaneous use of machines and AGVs involves decisions on production resource allocation and AGVs over time, as well as the choice of routes for manufacturing each batch of parts of a product, defining the moment of execution of each stage. A common criterion of production performance is the makespan value.

In this way, given a manufacturing system, it is assumed that the goal is to produce a number of products in a particular time. To do that, there are certain production resources whose operations on different materials, at distinct production stages of an specific product, must be determined. In this paper, the

production scheduling problem will be the task of determining which operations should be made and in which moments, so that all products are finished in the shortest time.

This paper addresses on the production scheduling problem of a plant with multiple machines and AGVs and also various types of products to be produced. There are several manufacturing routes to each type of product. A manufacturing route is a sequence of operations that must be followed in a defined order. Each operation is a manufacturing stage of the product performed on a particular machine. At the end of the sequence, the product will be ready. There may be more than a machine capable of perform the same operation, which leads to a product have more than one manufacturing route. In this way, each product's manufacturing routes are represented by the machines which it must move to be produced.

The AGVs scheduling problem addresses on the moment that a particular part needs to be transported to another machine. At this stage, the problem is to decide which of the AGVs shall be selected, since more than one may be available. This paper uses two dispatching rules to this problem: STT/D (Shortest Travel Time/Distance), presented by Benincasa [9], and RV (Random Vehicle), presented by Egbelu and Tanchoco [14].

### IV. THE PROPOSED ADAPTIVE GENETIC ALGORITHM MODELING

In this paper, the adaptive genetic algorithm used is the same as presented in Morandin et al. [1], where the crossover and mutation rates are dynamically adjusted as the performance of this algorithm in the search.

The modeling problem through the use of adaptive genetic algorithm is the same (codification stages of chromosome, crossover and mutation) as proposed by Morandin et al. [12]. In this approach, a chromosome is coded to indicate which products and their production routes should be processed. From there, the genetic algorithm used is responsible for conducting all the production scheduling, that is, define the moment and machine that products will be produced.

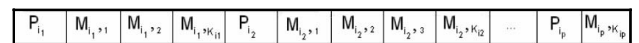


Figure 1. Chromosome codification where  $P_{ip}$  is the product to be processed and  $M_{ip,k_{ip}}$  is a machine of the production route of this product [12].

The crossover is made by changing the sub-vector regarding the same product between C1 and C2, generating C3 and C4. The mutation is made by randomly changing the production route of a product  $P_{ip}$  chosen in a random way by one possible route of that product [12]. These operations can be seen in figures 2 and 3.

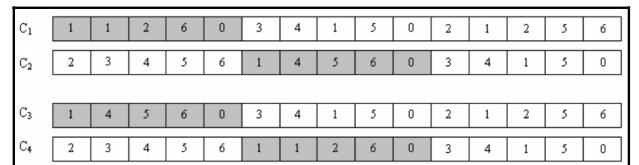


Figure 2. Crossover [12].

$C_1$	1	1	2	6	0	3	4	1	5	0	2	1	2	5	6
$C_1$	1	1	2	6	0	3	4	3	2	0	2	1	2	5	6

Figure 3. Mutation [12].

In this paper, the production scheduling is obtained in the stage of fitness calculation, in which products are allocated to machines following the registered routes. Still, AGVs, whose purpose is to carry a particular product for a particular machine or Load/Unload station, are scheduled with dispatching rules. With that it is possible to find makespan and fitness values of a particular chromosome.

#### A. Fitness Function

Fitness function is based on makespan value of each chromosome and aims to assign a value to individuals, according to their ability. To obtain the makespan value, it is necessary to define how AGVs will be scheduled to transport products among the machines.

#### B. AGVs Scheduling using Dispatching Rules

The AGVs are used to move the products among the machines, according to the selected route. The system adopted dispatching rules to decide which AGV will hold the job. In this paper two rules were used:

RV (Random Vehicle): used when all the vehicles are available at Load/Unload station.

SST/D (Shortest Travel Time/Distance): used when there is more than one available AGV and they are not at Load/Unload station.

After defining which dispatching rule will be used, it is necessary to define how it will be applied during the allocation process to machines. The steps that represent the logical implementation of the AGVs are presented below:

1. Move all the products to the Load/Unload station according to the order given by the chromosome, which is left to right;
2. Verify which product must be transported according to its priority;
  - 2.1. Verify which AGVs are available and apply the appropriate dispatching rule;
  - 2.2. Move the AGV from the current location to where it is requested. The AGV gets and moves the product to the next designed machine as registered in the product route;
  - 2.3. After carry a particular product, the AGV stays parked at machine's parking point or Load/Unload station's parking point in which the product was delivered;
3. Return to step 2. This process is repeated until all products have been produced and delivered to the Load/Unload station.

#### C. Getting the Makespan Value

To calculate the makespan value and obtain a fitness value for each chromosome, it is necessary to find the conclusion time of a determinate product operation, which is given by (1).

$$O_{ij} = T_{ij} + P_{ij} + E_{ij} \quad (1)$$

Where:

$O_{ij}$  = Operation completion time, in that  $j$  is the  $j$ -th operation of the product  $i$ ;

$T$  = Transportation time;  $P$  = Operation processing time and  $E$  = Waiting time

After finding the conclusion time of a particular product operation, it is possible to find the conclusion time of the product  $P_i$ , given in (2).

$$P_i = \sum_{j=1}^n O_{ij} \quad (2)$$

In equation (3) it is possible to find the makespan,  $mkp$ , based on the conclusion time of the products.

$$mkp = Max(P_1, P_2, P_3, \dots, P_n) \quad (3)$$

Lower makespan values chromosomes are considered more fit than chromosome with greater values. To obtain the fitness value,  $f_i$ , where  $i$  is a particular chromosome, an operation is performed on makespan values of current generation, given by (4).

$$f_i = mkp_{max} - mkp_i + mkp_{min} \quad (4)$$

Where,  $mkp_{max}$ , is the greatest makespan value of current generation,  $mkp_{min}$ , is the lowest makespan value of current generation and  $mkp_i$  is the makespan value of the chromosome  $i$  at current generation. This operation gives higher fitness values to individuals with lower makespan values in the generation, as proposed by Chiu and Fu [10]. This expression is used to calculate the fitness value of a chromosome, in a way that lower makespan value chromosome (more adapted), it gets a higher fitness value and has a higher probability of being selected to generate offspring to the next generation.

#### D. Dynamic adjustment of mutation and crossover rates

In order to find good parameters and avoid premature convergence of the GA (Genetic Algorithm), this paper uses a method to dynamically adjust its parameters during the search.

This method was proposed by [1] and used in this paper, where it involves some rules that dynamically adjust the mutation and crossover rates according with to the performance of genetic operators. The steps of Adaptive Genetic Algorithm used and the possible adjustments of genetic operator's rates are presented below:

1. Randomly generate the initial population and calculate fitness value of each individual;
2. Apply the selection method at current population;
3. Selected individuals should go through the crossover proposed process according to the crossover rate. After that, calculate the average fitness value of population;
4. Save the average fitness value of population after the crossover process;
5. Apply the mutation process according to the mutation rate and calculate the average fitness value of population;
6. Save the average fitness value of population after the mutation process;
7. Adjust the mutation and crossover rates by the following rule:
  - 7.1. If the percentage of improvement among offspring chromosomes average fitness and parent chromosomes average fitness is equal or greater than 10%, the occurrence probability of genetic operator should be increased of 0.05 for crossover operations and 0.005 for mutation operations;
  - 7.2. If the percentage of worsening among offspring chromosomes average fitness and parent chromosomes average fitness is equal or greater than 10%, the occurrence probability of genetic operator should be decreased of 0.05 for crossover operations and 0.005 for mutation operations;
  - 7.3. If the percentage of improvement/worsening among offspring chromosomes average fitness and parent chromosomes average fitness is within  $\pm 10\%$ , the occurrence probability of genetic operator should not be changed;
  - 7.4. Rates should be among 0.5 and 1.0 for crossover operator and among 0.00 and 0.10 for mutation operator.

Check the stop condition. If it is met, select the best chromosome of the last generation as the final solution to the problem. Otherwise, generate the next population, that will replace the former, and return to step 2.

#### V. CASE STUDY

Experiments were conducted and the results compared with the approach proposed by Reddy and Rao [2] and Morandin et al. [12], because both use Genetic Algorithm as modeling and search method for the scheduling problem. In this paper, the proposed approach was called "Proposal". The proposed approach by Reddy and Rao [2] and Morandin et al. [12] were called "R&R" and "Morandin", respectively.

A particular scenario was used for the validation of the proposed approach and, to this scenario, 35 tests were set to be subjected to statistical tests. Due to non-deterministic

characteristics of the adaptive genetic algorithm search, the results vary by test.

As the data obtained in the tests not belonged to a normal distribution, could be applied a non-parametrical test to analyze the obtained results. Among the possible tests, the Wilcoxon test [13] was used to check the confidence degree of the results for other tested approaches.

So the Wilcoxon test was used to compare the makespan results obtained in this paper on the proposed approach by Reddy and Rao [2], in order to verify if these values are lower in most compared cases, since Reddy and Rao [2] proposed approach tries to minimize more than one goal. The Wilcoxon test was not applied on the approach proposed by Morandin et al. [12], because this approach does not consider the AGVs scheduling and thus have lower makespan values in most cases. In this case, a comparative analysis was performed in order to verify that the results were not very distant when compared with Morandin et al. [12].

The problem in this paper refers to a scenario with nine machines e nine products, where each product has two possible manufacturing routes with five to seven machines each. The products manufacturing routes were randomly generated, as well as operation times, which ranged between 400 and 500 time units (TU). In this paper, two AGVs were used.

For the tests with the proposed approach, the following values were used for the adaptive genetic algorithm parameters: population size equals to 30, crossover rate starting at 0.6 (60%) e mutation rate starting at 0.005 (0.5%), with crossover rate of between 0.5 and 1.0 and mutation rate of between 0.005 and 0.1. The stop criterion used was 100 generations or when the algorithm converges.

#### VI. RESULTS AND DISCUSSION

The problem tests results can be seen at Table 1. The columns indicate the test number (from 1 to 35), the obtained makespan, that were counted in time units (TU), and the response obtaining time, calculated in minutes for each test, respectively. The next columns represent the makespan values reached in the other tested approaches and their response obtaining time, respectively.

The average makespan found for the proposed approach it was 5572 TU, with standard deviation of 148.48. The minimum value found was 5342 TU and the maximum was 5779 TU. To this problem there was a tendency of improvement in 71% of the results obtained with the results obtained by R&R. Thus, through the makespan values comparison, it appears the proposed approach presented better and statistically different results from the results obtained by R&R, with 95% confidence according to the Wilcoxon test.

TABLE I. THE 35 RESULTS OBTAINED

#	Proposal		R&R		Morandin	
	Makespan	Time	Makespan	Time	Makespan	Time
1	5342	2.14	5677	4.37	4755	0.135
2	5779	2.12	5693	4.37	4975	0.136
3	5342	1.89	5693	4.39	5083	0.133

4	5573	1.87	5677	4.37	4901	0.141
5	5579	1.87	5693	4.39	4972	0.145
6	5593	2.05	5691	4.33	4676	0.148
7	5563	1.81	5677	4.32	5096	0.150
8	5585	2.09	5692	4.31	5187	0.159
9	5755	1.86	5677	4.34	5147	0.141
10	5681	1.86	5689	4.34	5123	0.160
11	5738	2.10	5688	4.30	4901	0.142
12	5596	1.83	5685	4.34	4990	0.146
13	5594	1.86	5677	4.31	4973	0.151
14	5767	1.91	5680	4.32	5175	0.141
15	5402	1.86	5691	4.33	4673	0.138
16	5743	1.91	5677	4.32	4975	0.135
17	5726	2.01	5677	4.31	4754	0.142
18	5349	2.02	5689	4.39	5082	0.160
19	5455	1.90	5677	4.32	5121	0.153
20	5556	1.99	5677	4.31	4676	0.135
21	5595	2.04	5677	4.30	5082	0.132
22	5360	1.86	5692	4.29	5082	0.129
23	5358	2.13	5681	4.32	4942	0.127
24	5563	1.86	5677	4.32	5170	0.136
25	5750	1.85	5693	4.32	5502	0.135
26	5346	1.92	5688	4.30	5181	0.134
27	5663	1.37	5696	4.32	5003	0.130
28	5566	2.08	5689	4.34	5188	0.135
29	5725	1.86	5677	4.33	5082	0.134
30	5420	1.87	5703	4.33	5082	0.130
31	5580	1.34	5696	4.33	5302	0.137
32	5358	1.97	5680	4.32	4983	0.135
33	5524	2.05	5696	4.34	5115	0.134
34	5766	1.38	5693	4.32	5119	0.132
35	5739	1.81	5693	4.32	5518	0.136
<b>Average</b>	<b>5572</b>	<b>1.89</b>	<b>5686</b>	<b>4.33</b>	<b>5054</b>	<b>0.140</b>

When compared with Morandin results, it was observed that the obtained average it was 10.45% higher, while the maximum distance in relation to the minimum makespan value found was not greater than 23.67%, with an average distance of 19.24% of the minimum makespan found in the tests.

Regarding the response obtaining time of the proposed approach, the average was 1.90 minutes, almost 2.5 times smaller than the response obtaining time of R&R, which was 4.33 minutes. The values obtained by Morandin were low due to the non-use of AGVs scheduling.

In order to verify the behavior across the generations in search of a good solution of the adaptive genetic algorithm used in this paper the Figure 4 shows its convergence graph. In this graph were compared the best fitness value and the average fitness of the population for 100 generations. These values are related to makespans obtained for the tested problem. The R&R

convergence graph and Morandin convergence graph are shown in Figure 5 and Figure 6, respectively.

From these convergence graphs, it is possible to see that, when compared the adaptive genetic algorithm with genetic algorithm proposed by other approaches, through the dynamic adjust of crossover and mutation rates, the adaptive genetic algorithm explores other points of the search space converging to a region that contains the best solutions.

## VII. CONCLUSION

From the results, it is possible to conclude that the proposed approach can get a better makespan in most cases tested when compared with Reddy and Rao [2] results, since this approach consider a multiobjective problem. When compared with Morandin et al. [12] results, the results were not very distant, as Morandin does not consider the transport times and as a result of this its makespan values are lower in most cases. Thus, the proposal has achieved the expected goal.

The response obtaining times were low to the tested kind of scenario. This is a very important characteristic when it comes to a reactive scheduling.

From the use of adaptive genetic algorithm could be achieved a good performance on the addressed problem, as well as explore other points of search space through the dynamic adjustment of crossover and mutation rates that led convergence to a region with good solutions.

Based on these results, it is observed that the proposed approach in this paper can be applied to solve manufacturing problems with resource sharing of large scale, in order to achieve a good makespan values and a low response obtaining time.

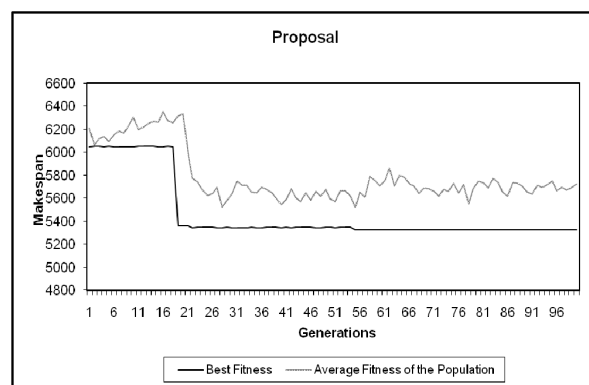


Figure 4. Proposal Converge Graph.

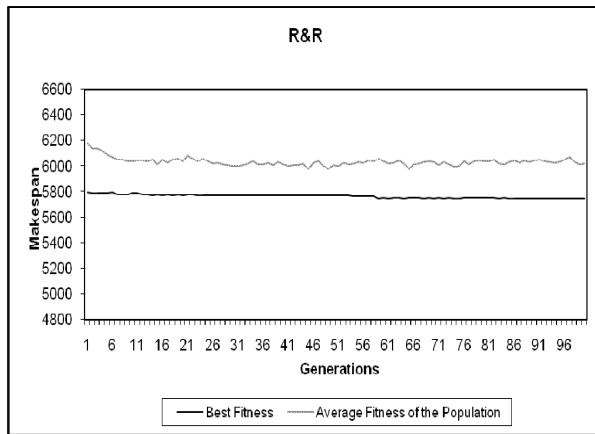


Figure 5. R&R Convergence Graph

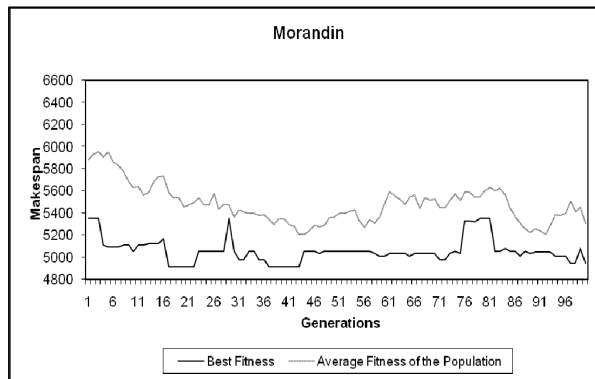


Figure 6. Morandin Convergence Graph

## REFERENCES

- [1] O. Morandin Jr., D.S. Sanches, A.C. Deriz, E.R.R. Kato, and R.H. Tsunaki, "An Adaptive Genetic Algorithm Based Approach for Production Reactive Scheduling of Manufacturing Systems", 34rd Annual Conf. of the IEEE Industrial Electronics Society, Orlando – USA, 2008, in press.
- [2] B.S.P. Reddy and C.S.P. Rao, "A hybrid multi-objective GA for simultaneous scheduling of machines and AGVs in FMS", *International Journal Advanced Manufacturing Technology*, Springer-Verlag, 2006, pp. 601-613.
- [3] G. Ulusoy, F.S. Serifoglu, U. Bilge, "A genetic algorithm approach to the simultaneous scheduling of machines and automated guided vehicles", *Computers and Operations Research*, Elsevier-Science, 1997, pp. 335-351.
- [4] P. Pongcharoen, C. Hicks and P.M. Braiden, "The development of genetic algorithms for the finite capacity scheduling of complex products, with multiple levels of product structure", *European Journal of Operational Research*, Elsevier-Science, 2004, pp. 215-225.
- [5] P. Pongcharoen, C. Hicks, P.M. Braiden, D.J. Stewardson, "Determining optimum genetic algorithm parameters for scheduling the manufacturing and assembly of complex products", *Int. Journal of Production Economics*, Elsevier-Science, 2002, pp. 311-322.
- [6] J. Jerald, P. Asokan, R. Saravanan and A.D.C. Rani "Simultaneous scheduling of parts and automated guided vehicles in a FMS environment using adaptive genetic algorithm", *Int. Journal of Adv Manuf Technol*, Springer-Verlag, 2006, pp. 584-589.
- [7] L. Zhang, L. Wang and D. Zheng "An adaptive genetic algorithm with multiple operators for flowshop scheduling", *Int. Journal of Adv Manuf Technol*, Springer-Verlag, 2006, pp. 580-587.
- [8] Y. Yin, J. Yiu and Z. Cheng "A Genetic Algorithm Based Approach to Flowshop Scheduling", *Proceedings of the 5<sup>th</sup> World Congress on Intelligent Control and Automation*, IEEE, Hangzhou, P.R. China, June, 2004, pp. 3019-3021.
- [9] A.X. Benincasa, O. Morandin Jr., E.R.R. Kato, "Reactive Fuzzy Dispatching Rule for Automated Guided Vehicles", *IEEE International Conference on Systems, Man and Cybernetics*, Washington, IEEE, D.C., USA, , October, 2003, pp. 4375-4380.
- [10] Y.F. Chiu and L.A. Fu, "GA embedded dynamic search algorithm over a petri net model for an FMS scheduling", *Proceedings of the 1997 IEEE Int. Conf. on Robotics and Automation*, Albuquerque, New Mexico, IEEE, April, 1997, pp. 513-518.
- [11] S.S. Sankar, S.G. Ponnambalam, R. Rajkumar and M. Gurumarimuthu, "An Intelligent Integrated Scheduling Model for Flexible Manufacturing System", *Proceedings of the 2004 IEEE Conference on Robotics, Automation and Mechatronics*, IEEE, Singapore, December, 2004, pp. 1095-1100.
- [12] O. Morandin Jr., E.R.R. Kato, A.C. Deriz and D.S. Sanches, "A Search Method using Genetic Algorithm for Production Reactive Scheduling Manufacturing Systems". *IEEE International Symposium on Industrial Electronics*, Cambridge, UK, June, 2008, in press.
- [13] Degroot M.H., Schervish M.J., *Probability and Statistics*, 3rd ed., Addison-Wesley, 2001.
- [14] P.J. Egbelu, J.M.A Tanchoco, "Characterization of Automatic Guided Vehicle Dispatching Rules", *International Journal of Production Research*, 1984, pp. 359-374.