

Title	Integration of Multi-objective Genetic Algorithms and Expert Systems for Agroindustrial System Design
Author(s)	Y., Yandra; Tamura, Hiroyuki
Citation	
Issue Date	2005-11
Type	Conference Paper
Text version	publisher
URL	<a href="http://hdl.handle.net/10119/3951">http://hdl.handle.net/10119/3951</a>
Rights	2005 JAIST Press
Description	The original publication is available at JAIST Press <a href="http://www.jaist.ac.jp/library/jaist-press/index.html">http://www.jaist.ac.jp/library/jaist-press/index.html</a> , IFSR 2005 : Proceedings of the First World Congress of the International Federation for Systems Research : The New Roles of Systems Sciences For a Knowledge-based Society : Nov. 14-17, 2161, Kobe, Japan, Symposium 3, Session 7 : Intelligent Information Technology and Applications Computational Intelligence (1)



# Integration of Multi-objective Genetic Algorithms and Expert Systems for Agroindustrial System Design

Yandra and Hiroyuki Tamura

Department of Electrical Engineering and Computer Science  
Faculty of Engineering, Kansai University  
3-3-35 Yamate-cho, Suita, Osaka 564-8680, Japan  
yandra@asdel.ee.kansai-u.ac.jp, h.tamura@kansai-u.ac.jp

## ABSTRACT

This research aims to develop an integrated system for agroindustrial system design based on artificial intelligence approach. In the first instance, a new multi-objective genetic algorithm with Additional Diversity Module (ADM) is developed to optimize combinatorial flow-shop scheduling of an agroindustry. The unique feature of the optimization model used in this research is it uses make-span and Average Product Quality Deterioration (APQD) as optimization objectives. The result of genetic algorithm optimization is a set of compromise solution that called Pareto-optimum solutions. These solutions are then fed to an expert system to finding the most preferred solution by considering higher-level information. A non-trivial case study is then presented to demonstrate the capability of this intelligent system to solve an agroindustrial system design problem. The result indicates that the intelligent system developed in this work is robust and reliable.

**Keywords:** multi-objective, genetic algorithms, expert systems, scheduling, agroindustry

## 1. INTRODUCTION

This paper discusses the application of multi-objective genetic algorithms and expert systems for agroindustry management. Agroindustry is defined as agricultural products processing industry as well as other business-oriented activities to: (1) increase added value of agricultural products, and (2) make more environmental-friendly manufactured goods by using agricultural products as inputs [1,2]. Some examples of agroindustry are biotechnology-based products industry (such as bioplastics, enzymes and bio-lubricants), natural medicine industry and food industry.

Unlike general manufacturing, the raw materials used in agroindustry have unique characteristics, such as perishable, variability, seasonality and bulky. For this reason, management of agroindustry is more complicated than general manufacturing as we have to take into consideration these important factors in the decision making process.

At present, with the rapid development in agroindustry some researchers have focused their research in the optimization of agroindustrial systems design such as Vera (2003) [3]; Garg (1999) [4], Lin (2003) [5] and Matthew (2005) [6], to mention only a few. Despite their advantages, these research works have two major drawbacks. Firstly, most of the models developed are traditional single-objective optimization models. In fact, many problems in real world, including in agroindustry management, are multi-objective in nature, i.e. more than one objective have to be optimized simultaneously. In multi-objective optimization there cannot be a single optimum solution which simultaneously optimizes all objectives. The resulting outcome is a set of non-dominated solutions with varying degree of objectives value that are called Pareto-optimal solutions.

Secondly, the existing research works that dealt with multi-objective models used classical methods such as weighted sum methods,  $\epsilon$ -constraint methods and goal programming methods. Most of these algorithms convert the multi-objective optimization problem into a single-objective optimization problem by using some user-defined procedure. In their practical use each of these algorithms may have to be used many times, hopefully each time finding a different Pareto-optimal solution. However, the field of search and optimization has changed over the last few years by the introduction of a number of non-classical, unorthodox and stochastic searches such as genetic algorithms and evolutionary computation [7,8]. The advantage of genetic algorithms and other

population based search is that they can find multiple optimal solutions in one simulation run. For this reason, a genetic algorithm is used in this research for finding Pareto-optimal solutions.

Furthermore, Deb (2001) [7] stated that the ideal approach for multi-objective optimization consists of two steps: (1) To find multiple trade-off optimal solutions with a range of values for objectives, and (2) To choose one of the obtained solutions using higher-level information.

One best approach to process higher-level information which consists of symbolic data and large knowledge-base is an expert system [9]. For this reason, an expert system is developed in this research to select the most preferred solution based on Pareto-optimal solutions fed by genetic algorithm.

## 2. OBJECTIVES

The objective of this research is to develop an integrated system based on genetic algorithms and expert systems for the design of multi-objective agroindustrial systems design. The focus of this research is optimization of combinatorial flow-shop scheduling problems, as production scheduling is one important area in the design of complex systems such as agroindustry.

## 3. PROBLEM FORMULATION

In principle, flow-shop scheduling problems in agroindustry are similar to manufacturing flow-shop problems, except in the objectives of optimization. The details of general flow-shop scheduling problems are presented in the following paragraphs.

The flow shop problem can be presented as a set of  $N$  jobs  $\{J_1, J_2, \dots, J_N\}$  to schedule on  $M$  machines [10]. The machines are critical resources: one machine cannot be assigned to two jobs simultaneously. Each job is composed of  $M$  consecutive tasks  $J_i = \{t_{i1}, t_{i2}, \dots, t_{iM}\}$ , where  $t_{ij}$  represents the  $j^{\text{th}}$  task of the job  $J_i$  requiring the machine  $m_j$ . It should be noted that jobs have the same processing sequence on the machines. To each task  $t_{ij}$  is associated a processing time  $p_{ij}$ .

Scheduling of tasks on different machines must optimize certain regular criteria such as make-span (total completion time), maximum tardiness,

maximum flow-time and number of jobs delayed with regard to their due date [11,12]. Most of the past research works in scheduling considered these criteria. For example, Talbi *et.al.*[13], took into account two criteria in their research i.e. minimizing make-span and total tardiness. The other researchers [14,15,16,17] used make-span and mean flow time as well as mean tardiness as their optimization objectives. In this research we are interested to use make-span and a new scheduling objective specific for agroindustry, i.e. Average Product Quality Deterioration (APQD) as multiple-objective in the optimization model. The APQD is an important factor for improving customer satisfaction level and product quality in agroindustry. The illustration of make-span and APQD is presented in Figure 1.

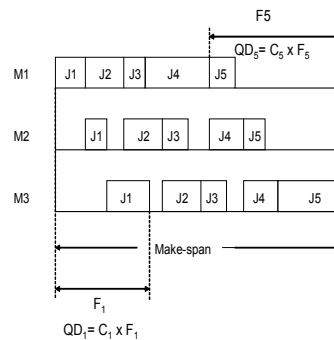


Figure 1. Make-span and APQD in agroindustry flow-shop scheduling

It can be seen in Figure 1 that make-span is an overall completion time for all jobs in a particular schedule or job arrangement. Moreover, as also presented in Figure 1, each job  $i$  has an overall completion time, called flow time  $F_i$  (in minutes). Unlike in general manufacturing, during this completion time the quality of agricultural products will be deteriorated with a certain speed [1,2]. In this research we assume that the deterioration function is linear and the constant value for the deterioration is  $C_i$  (quality unit/mins). Thus, for job  $i$ , the quality deterioration ( $QD_i$ ) of the product  $i$  (in quality unit, abbreviated as q.u) will be :

$$QD_i = C_i \times F_i$$

The average value of  $QD_i$  for all jobs is called Average Product Quality Deterioration (APQD) and can be calculated with the following formula:

$$APQD = \frac{1}{n} \sum_{i=1}^n QD_i$$

where,

$C_i$  = Quality Deterioration Constant for Job  $i$  (q.u/minute)

$F_i$  = Flow time of Job  $i$  (minutes)

$QD_i$  = Quality Deterioration of Job  $i$  (q.u)

$n$  = number of product

It should be noted that if the schedule or the arrangement of the jobs is changed, make-span and APQD will also be changed. The smaller the make-span for all jobs, then the better is the schedule for that flow-shop. However, minimizing make-span will lead to a large APQD value. Thus, both optimization problems of minimizing make-span and minimizing APQD have conflicting optimal solutions.

#### 4. SYSTEM DESIGN AND DEVELOPMENT

In order to solve the schedule optimization with multiple and conflicting objectives as described in Section 3, a genetic algorithm is developed in this work. The unique feature of the new multi-objective genetic algorithm developed in this research is that it has an Additional Diversity Module (ADM) to increase the solutions' diversity by maintaining fully heterogeneous population. The output of this multi-objective genetic algorithm is a Pareto-optimum front that consists of non-dominated solutions. An expert system is then developed in this work to select the most preferred solution from the non-dominated or compromise solutions obtained from genetic search. The details of the development of genetic algorithm and expert system are described in the following sections.

##### 4.1. Multi-objective Genetic Algorithms

In the last few years, there has been a number of research works conducted in the area of multi-objective optimization using genetic algorithms. The example of such research works are the development of Vector Evaluated Genetic Algorithms (VEGA), Multi Objective Genetic Algorithms (MOGA), Non-dominated Sorting Genetic Algorithms (NSGA and NSGA II) [7],

to name only a few. Such genetic algorithms have been implemented to function optimization, combinatorial optimization such as scheduling and other areas in engineering and management [18,19,20].

Despite their success, there are rooms for improving the performance of the existing genetic algorithms. The main weakness of the existing multi-objective genetic algorithms is that they allow identical chromosomes or individuals in the population. This drawback leads to slow convergence to the true Pareto-optimum front and poor diversity of optimum solutions. For this reason, this research aims to develop a new genetic algorithm that has an additional module to preventing unintended identical chromosomes in the population. This additional module is called "Additional Diversity Module" (ADM) and inserted into a widely-used multi-objective genetic algorithm, namely NSGA-II created by Deb (2001) [7]. Thus, the development of multi-objective genetic algorithm in this work consists of several steps as presented below:

**Step 1:** Initialize random population of  $N$  individuals ( $P_0$ ).

**Step 2:** Sort this population into different non-domination levels

**Step 3:** Assign fitness to each solution based on its non-domination level

**Step 4:** Reproduce according to assigned fitness and local crowding distance using Binary Tournament Selection

**Step 5:** Crossover and Mutation according to  $p_c$  and  $p_m$  to create an offspring population  $Q_0$  of size  $N$ .

**Step 6.** Combine parent and offspring populations and create  $R_t = P_t \cup Q_t$ . Perform a non-dominated sorting to  $R_t$  and identify different fronts:  $F_i$ ,  $i=1,2,\dots$ ,etc.

**Step 7.** Set new population  $P_{t+1} = \emptyset$ . Set counter  $i = 1$ . Until  $|P_{t+1}| + |F_i| < N$ , perform  $P_{t+1} = P_{t+1} \cup F_i$  and  $i = i + 1$ .

**Step 8.** Perform the Crowding-sort ( $F_i, <_c$ ) procedure (described in Section 4.2) and include the most widely spread ( $N - |P_{t+1}|$ ) solutions by using the crowding distance values in the sorted  $F_i$  to  $P_{t+1}$ .

**Step 9:** Use Additional Diversity Module (ADM) to detect and then manipulate identical chromosomes in new population

**Step 10:** Create offspring population  $Q_{t+1}$  from fully heterogeneous population  $P_{t+1}$  by using the

crowded tournament selection, crossover and mutation operators.

**Step 11:** If maximum number of generation specified met then iteration is complete, otherwise go to step 6.

It should be noted that in Step 9 above, an ADM is employed to detect and manipulate identical chromosomes in new population. The idea to develop ADM comes from genetic engineering discipline. For many years, genetic engineers have been conducting genetic manipulation to living organisms in order to obtain the intended best offspring. This biological method is adopted and simulated in ADM to change unintended chromosomes or individuals in a population into better chromosomes. Although, some might argue that this operator is pretty similar to mutation, there are some major differences. Firstly, mutation applied randomly to chromosomes in the population and the probability of mutation is very small. Secondly, the purpose of mutation is merely to ensure that the search is not convergent to local optimum, so it could produce individuals that similar to the others existing individuals in the population. In contrast, the procedure in ADM is applied to all identical chromosomes in the population and then changes the genetic structure of those chromosomes to produce new and hopefully better chromosomes in order to obtain maximum diversity of the population.

The other question might arise is what is the difference of the ADM with other techniques for maintaining population diversity? The answer is, most of the techniques developed to maintain diversity manipulate fitness value, not chromosome structures. For example, some techniques use the distance between the solutions as basis for fitness assignment. Solutions in the same front, i.e. in the same non-dominated ranks will share their fitness based on their distance. Solutions in a crowded area will be assigned small fitness values. Solutions in less crowded area will be assigned larger fitness values. So, the chance of solutions from less crowded area to be selected as a parent is larger than the solutions from more crowded area and thus it is expected to create a more diverse population in the next generation.

The other researchers use clustering systems as their methods for maintaining population diversity. In the clustering system, solutions in crowded area are grouped into one solution

based on the “center of gravity” of the area. The solution that is closest to the “center” of the area is chosen as a representative solution in that particular area, while the other solutions are deleted.

As discussed before, the techniques discussed above allow identical individuals in population which could lead to slow convergence the true Pareto-optimum front and poor diversity of final solutions. In contrast, ADM manipulates genes in the chromosomes or individuals instead of fitness value, so it can prevent identical individuals in a population. For this reason, ADM is used in this research to complement the existing techniques. The results of this new innovation of the multi-objective algorithm is very promising compare to the previously developed algorithms. The procedure of ADM is presented below.

**Step 1:** Compare all chromosomes in the new population  $P_t$ .

**Step 2:** If there are two or more identical chromosomes in population  $P_t$  then perform genetic manipulation by changing one or more bits in the duplicate(s) randomly to create new chromosome(s)

**Step 4:** Stop if all chromosomes have been fully heterogeneous

It should be noted that the ADM proposed above is applies to both non-elitist and elitist multi-objective genetic algorithms. In case of elitist multi-objective genetic algorithm, such as NSGA-II used in this research, where new population are created based on the combination population  $R_t$  with the size of  $2N$ , chromosomes manipulation can be carried out in two methods. In the first method, the new chromosomes are created similar to chromosomes in the archive population (population  $R_t - P_t$ ). In the second method, a chromosome manipulation is carried out just after offspring population  $Q_t$  formed, so before this offspring population is combined with its parent population and thus before archive population made. In this case, completely new chromosomes will be created. Although both of these methods can be simulated in computer programs, in this study we have simulated and implemented the first one.

## 4.2. Expert Systems

Expert systems are computer programs that can analyze a stored database of information or a

knowledge-base by using a particular inference mechanism [9]. Expert systems are suitable to solve problems that contain symbolic and higher-level information processing usually solved using heuristics or human expertise. The main components of an expert system are knowledge-base, inference engine and user interface. In this research, the expert system is developed by using an expert system shell that already has pre-programmed inference engine and user interface. An appropriate knowledge-base for this expert system is then developed by using series of steps such as knowledge acquisition, knowledge formalization and refinement as well as knowledge representation using production rules.

The knowledge is acquired from agroindustrial experts as well as from other sources such as text-books, journals and manuscripts and then stored in the knowledge-base of the expert system. In order to select the most preferred solution for the agroindustry flow-shop problems, the expert system developed in this work used four factors: (1) production budget, (2) desired resource utilization, (3) desired customer zone, and (4) the quality of storage and transportation facility. Factor 1 and 2 has three levels: High (H), Medium (M) and Low (L). Factor 3 has three levels: A, B, C, and factor 4 has three levels: Excellent (E), Average (A) and Low (L). The dependence diagram of this expert system is presented in Figure 2.

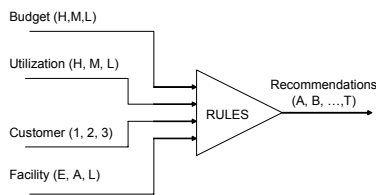


Figure 2. Dependence diagram of the expert system

Basically, there are 81 rules can be created using the combination of the levels of the decision factors. These rules are then formalized, refined and implemented in IF-THEN format. An example of the rules is presented below:

Rule 075:

IF budget is Low AND  
 desired resource utilization is High OR  
 desired resource utilization is Medium AND  
 customer zone is 2 AND  
 storage/transportation is Excellent  
 THEN  
 choose schedule S-08

One of the advantages of the expert system is its knowledge base can be updated in a regular interval basis. In the updating process some new knowledge can be added to the expert systems and the obsolete knowledge can be deleted.

## 5. AN APPLICATION EXAMPLE

To see the performance of this new developed system to solving a complex agroindustry scheduling problem, a literature case study combined with field survey data is used. The details of this case study are as follows. An agroindustry produces 10 types of product using 5 machines by a flow-shop system. The processing times of each product (or each job) at each machine are presented in Table 1.

Table 1. Processing times of each job in each machine (in minutes)

	M1	M2	M3	M4	M5
J1	2.3	4.5	6.8	1.2	4.5
J2	4.2	1.2	3.4	4.5	2.8
J3	1.4	3.8	1.5	8.1	1.9
J4	5.6	2.4	2.5	3.2	2.0
J5	7.8	9.8	3.5	4.2	9.5
J6	3.0	7.6	7.9	2.5	7.8
J7	9.0	4.7	2.0	2.9	3.5
J8	3.4	8.3	8.7	8.9	1.3
J9	5.6	7.0	3.9	7.5	7.4
J10	6.7	6.1	2.1	1.9	3.6

Quality deterioration constant for job 1 to job 10 are 0.7, 0.5, 0.9, 0.1, 0.2, 0.7, 0.6, 0.8, 0.4 and 0.2 q.u/mins respectively.

The objective of the company is to find a set of Pareto optimal solutions that minimizing make-span and APQD and then selecting the most preferred solution based on those optimal solutions. The genetic algorithm was then run with population size = 20, crossover probability ( $P_c$ ) = 0.9, 0.8 and 0.7 respectively, mutation probability ( $P_m$ ) = 0, 0.01 and 0.05 respectively, and 5 replications of each combination of  $P_c$  and  $P_m$ . The plot of make-span and APQD resulted

from the best run of this experiment is presented in Figure 3. The Pareto optimum schedules obtained is presented in Table 2.

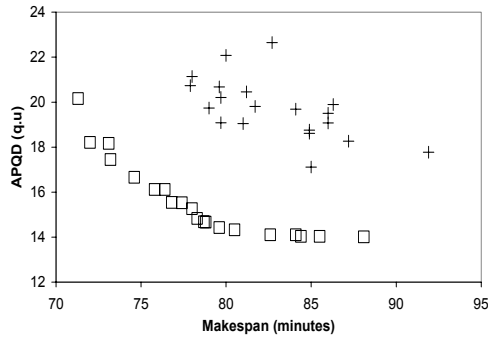


Figure 3. Pareto-optimum solutions after 350 generations ( $\square$ ) compare to initial solutions ( $+$ )

Tabel 2. Pareto optimum solutions obtained from genetic search

Schedule	Make-span	APQD
CFIBHAGEDJ	71.3	20.16
DCJBAGFIHE	88.1	14.02
FIDBHAJGJC	72	18.21
DCJBAFGIEH	82.6	14.11
DCBAHJGFEI	80.5	14.33
DFIBHAGEJC	74.6	16.66
FIDBHJEGCA	73.2	17.45
DCABFIEGHJ	78	15.27
DFIBHJEGCA	75.8	16.12
DAIBHGFEJC	76.8	15.55
DAIBHFGEJC	76.4	16.12
DCBAFGIHJE	79.6	14.43
DCJBAGFEIH	85.5	14.04
DCABFIGHJE	78.3	14.84
FIDBHJAEGC	73.1	18.17
DAIBHGFJEC	77.4	15.53
DCJBAIGFEH	84.1	14.11
DCBAFIGHJE	78.8	14.67
DCJBAFGIHE	84.4	14.04
DCBAHGFIJE	78.7	14.69

It can be seen from Figure 3 that genetic search convergent to Pareto-optimum solutions in relatively small number of generation. Furthermore, final solutions resulted by the genetic algorithm developed in this work has

very good diversity and completely heterogeneous, as presented in Table 2.

The Pareto-optimum solutions found by genetic algorithm were then fed to the expert system for selecting the most preferred solution. As this agroindustry has high budget, medium resource utilization, customer type 1 and excellent level of storage and transportation facility, then rule No:056 was fired. The recommendation was to choose schedule DFIBHJEGCA with make-span 75.8 minutes and APQD = 16.12 q.u.

This numerical example shows that the integration of multi-objective genetic algorithm and expert system developed in this work can be used to solve a difficult flow-shop scheduling problem in agroindustry successfully.

## 6. CONCLUSIONS

This paper has presented the application of multi-objective genetic algorithms and expert systems for the design of agroindustrial scheduling systems. The multi-objective genetic algorithm developed has a unique feature, i.e. ADM (Additional Diversity Module), to increase the Pareto-optimum front diversity. A well-designed expert system has also been developed in order to select a single-preferred solution based on compromise solutions created by the genetic algorithm. The experiment with a non-trivial combinatorial scheduling of an agroindustry shows that this integrated methodology is robust and reliable.

## ACKNOWLEDGEMENTS

This work was supported by MEXT (Ministry of Education, Culture, Sports, Science and Technology) under Grant-in-Aid for JSPS (Japan Society for the Promotion of Science) fellows No.16.04090.

## REFERENCES

- [1] Brown, J.E. 1994, Agroindustrial Investment and Operations, Word Bank Publications
- [2] Austin, J.E. 1992, Agroindustrial Project Analysis, John Hopkins University Press
- [3] Vera, J., de Atauri, P., Cascante, M, Torrens, N., 2003, "Multicriteria Optimization of Biochemical Systems by Linear Programming: Application to Production of Ethanol by *Saccharomyces cerevisiae*", Biotechnology and Bioengineering, Vol. 83, No.3, pp: 335-343

- [4] Garg, S., Gupta, S.K., 1999, "Multiobjective optimization of a free radical bulk polymerization reactor using genetic algorithm", *Macromolecular Theory and Simulation*, Vol. 8, No. 1, pp: 46-53
- [5] Lin, C.-W.R., Chen, H.-Y.S., 2003, "Dynamic allocation of uncertain supply for the perishable commodity supply chain", *International Journal of Production Research*, Vol. 41, No.13, pp:3119-3138
- [6] Matthews, K.B., Buchan, K., Sibbad, A.R., Craw, S. 2005. "Combining deliberative and computer based methods for multi-objective land-use planning", *Agricultural Systems*, (in Press)
- [7] Deb, K. 2001, *Multi-objective Optimization using Evolutionary Algorithms*, John Wiley, NY
- [8] Coello, C.A.C., van Veldhuizen, D.A., Lamont, G.B. 2002, *Evolutionary Algorithms for Solving Multi-Objective Problems*, Kluwer Academic Press
- [9] Turban, E., Aronson, J., Aronson, J.E. 2000. *Decision Support Systems and Intelligent Systems*. 6<sup>th</sup> Edition. Prentice Hall, USA
- [10] Kusiak, A. 1990, *Intelligent Manufacturing System*, Prentice Hall, Englewood Cliffs, New Jersey
- [11] Bagchi, T.P., 1999, *Multiobjective Scheduling by Genetic Algorithms*. Kluwer Academic Publishers
- [12] Silva, J.D.L., Burke, E.K., Petrovic, S. 2004, "An introduction to multiobjective metaheuristics for scheduling and timetabling", in Gandibleux, X., Sevaux, M., Sorensen, K., T'kindt, V., eds.: *Metaheuristics for Multiobjective Optimisation*, Springer, pp 91-129
- [13] Talbi, E., Rahoual, M., Mabed, M.H., Dhaenens, C. 2001, "A hybrid evolutionary approach for multicriteria optimization problems: Application to flow shop", in Zitzler, E., Deb, K., Thiele, L., Coello, C.A.C, Corne, D., eds.: *Evolutionary Multi-Criterion Optimization*, Springer, pp 416-428
- [14] Garen, J. 2004, "A genetic algorithm for tackling multiobjective job-shop scheduling problems", in Gandibleux, X., Sevaux, M., Sorensen, K., T'kindt, V., eds.: *Metaheuristics for Multiobjective Optimisation*, Springer, pp 201-219
- [15] Bagchi, T.P. 2001, "Pareto-optimal solutions for multi-objective production scheduling problems", in Zitzler, E., Deb, K., Thiele, L., Coello, C.A.C, Corne, D., eds.: *Evolutionary Multi-Criterion Optimization*, Springer, pp 458-471
- [16] Brizuela, C., Sannomiya, N., Zhao, Y. 2001, "Multi-objective flow-shop: Preliminary Results", in Zitzler, E., Deb, K., Thiele, L., Coello, C.A.C, Corne, D., eds.: *Evolutionary Multi-Criterion Optimization*, Springer, pp 443-457
- [17] Basseur M., Seynhaeve, F., Talbi, E.G., 2002, "Design of Multiobjective Evolutionary Algorithms to the Flow-shop Scheduling Problem", *Proceedings of the 2002 Congress on Evolutionary Computation*, IEEE Press, pp 1151-1156, 2002
- [18] Chan, F.T.S., Chung, S.H., Wadhawa, S., 2005. "A hybrid genetic algorithm for production and distribution", *Omega*, Vol. 33, pp: 345-355
- [19] Gupta, R.R., Gupta, S.K., 1999, "Multiobjective optimization of an industrial nylon-6 semibatch reactor system using genetic algorithm", *Journal of Applied Polymer Science*, (73), pp: 729-739
- [20] Guillen, G., Mele, F.D., Bagajewicz, M.J., Espuria, A., Puigjaner, L. 2005. "Multi-objective supply chain design under uncertainty", *Chemical Engineering Science*, 60, pp: 1535-1553