International Journal of Computers, Communications & Control
Vol. I (2006), No. 3, pp. 71-83

Multiobjective Optimization Scheduling Problems by Pareto-optimality in Agro-alimentary Workshop

Fatma Tangour, Ihsen Saad

Abstract: This paper deals with the multiobjective optimization problem of an agro-alimentary production workshop. Three criteria are considered in addition to this initial cost of production: the cost of the out-of-date products, the cost of the distribution discount and the makespan, and a new coding is proposed for this type of workshop. The adopted approach consists in generating optimal solutions diversified in the search space of solutions, and to help the decision maker when it cannot give a particular preference to one of the objective functions to make the good decision with respect to the quoted criteria.

Keywords: Agro-alimentary workshop, scheduling problems, genetic algorithms, Pareto-optimality, multiobjective optimization, production cost, makespan.

1 Introduction

Multi-objective optimization aims to optimize several components of an objective functions vector. Contrary to mono-objective, the multi-objective problem usually does not have a solution optimizing the whole concerned criteria, but a set of solutions, known as the set of the Pareto-optimal solutions. Any solution of this unit is optimal in the sense that no improvement can be made on a component without degradation of at least another component of the vector [15]. Given that a solution chosen by a decision maker can not be acceptable by another, it proves to be useful to envisage several alternatives to the choice of a Pareto optimal solution [18]. In this article, the scheduling problems in the agro-alimentary production workshops are dealt [6]. The principal objective is to search a realizable scheduling minimizing the makespan, the cost of the out-of-date products and the cost of the distribution discount. The transformation methods of the multi-objective problems into mono-objective problems are applied [2]. This article is organized as follows. The one machine scheduling problem is formulated in section 2; the resolution approach suggested with this problem is described in section 3. The effectiveness of this approach is tested for some examples in section 4.

2 Problem formulation

The problem is to build a multi-objective one machine scheduling problem adapted to agro-alimentary industries. Among the constraints and the criteria specific to agro-food industry, the out-of-date of the products and the discount of distribution can be distinguished. The objective is then to select among the cases of realisable scheduling the one which presents the best reducing compromise between the various criteria [7].

The goal of this study is, then, to minimise these criteria such as:

- C_1 : the cost of the out-of-date products,
- C_2 : the cost of the distribution discount,
- C_3 : the makespan.

The basic production cost on the one machine problem is supposed independent from the scheduling.

The data of the considered case are as follows:

We have a set n of operations, each operation is characterised by its earliest starting time, its effective starting time, its processing time and its effective completion time.

Notations :

t_i : effective starting time of operation O_i,
r_i : earliest starting time of the operation O_i,
γ_i : effective completion time of the operation O_i,
p_i : processing time of the operation O_i,
P_i : finished product of the operation O_i,
c_{ik} : k^{th} component of the components set of the operation O_i,
v_{ik} : validity limit date of the component c_{ik},
C_{P_i} : completion time of product P_i,
$d_{P_i}^{liv}$: delivery date of the product P_i,
Dv_{P_i} : lifespan of the product P_i,
Dr_{P_i} : return delay of the product P_i,
P_{ik}^{rev} : cost price of the component of the c_{ik} product P_i,
$P_{P_i}^{ven}$: unit selling price of the product P_i
$C_{P_i}^{stk}$: cost of storage per unit of time of a unit of the product P_i.

2.1 Criteria formulations

Three criteria are considered. The two first constitute criteria specific to the agro-alimentary production workshops [16]. The last criterion is traditional and used for the optimization of the scheduling problems of a traditional production workshop.

The considered objectives relate to minimization:

- C_1: the cost of the out-of-date products

$$C_1 = \sum_i \sum_k P_{ik}^{rev} \left(\frac{\max(0, t_i - v_{ik})}{(t_i - v_{ik})} \right) \tag{1}$$

- C_2: the cost of the distribution discount

$$C_2 = \sum_i \max\left(0, d_{P_i}^{liv} - C_{P_i}\right) \times \left(\frac{P_{P_i}^{ven}}{Dv_{P_i} - Dr_{P_i}} + C_{P_i}^{stk} \right) \tag{2}$$

- C_3: the makespan

$$C_3 = \sum_i (t_i + p_i) \tag{3}$$

2.2 Lower bound Formulations

Proposition 1. *$C_i \geq 0, \forall i \in \{1,2\}$ and $C_i^b = 0$; where C_i^b represents the lower bound of the criteria C_i.*

Proposition 2. *The lower bound of the makespan, C_3^b is defined as follows:*

$$C_3^b = \sum_i \min(r_i + p_i) \tag{4}$$

Proof. When: $C_3 = \max_{1\le i\le n} \gamma_i$, $t_i \ge r_i$ and then

$$C_3^b = \min\left(\sum_i (t_i + p_i)\right) \tag{5}$$

$$C_3^b = \sum_i \min(r_i + p_i) \tag{6}$$

□

3 Genetic Algorithms application for the scheduling problems

3.1 principle

Various approaches have been proposed to solve scheduling optimization problems, among them the Genetic Algorithms (GAs) approach can be distinguish. This approach was largely adopted these last years [11], [14]. The use of GAs in many fields proved reliable in particular in combinatorial problems such as the scheduling problems [3], [4], [12]. Other hybrid algorithms have also been proposed [1] [8].The main difficulty in the resolution of these problems types results in their algorithmic representation form, which constitutes the most significant point in genetic search. Several representation approaches and various standard AGs operators were proposed, to solve these problems. Among them, the representation based on the priority rules [5]. The principle of a simple Genetic Algorithm is as follows, figure 1.

3.2 Proposed Genetic Algorithm Coding

Proposed coding in the application case is: Ordered Operations Coding Lists “OOCL”, table 1. Inspired from the CLO (Operation List Coding) coding [9] and the CPM (Parallel Machines coding) coding [10], it consists in proposing ordered lists for the products line. The proposed coding defined the ordered, the starting time and the completion time of the operations. These dates are calculated and updated by the “dates calculation algorithm”, table 2.

Table 1: OOCL Coding

1	2	3	4
O_2, t_2, γ_2	O_1, t_1, γ_1	O_4, t_4, γ_4	....

Table 2: dates calculation algorithm

t_i	: effective starting time of operation O_i,
	$t_i = \max(r_i, \gamma_j)$
γ_i	: completion time of the operation O_i,
	$\gamma_i = \max(r_i, \gamma_j) + p_i$
where γ_j represents the completion time of the operation O_j that preceded O_i	

The operators used for this coding are: mutation, crossover at a point and crossover at two points. The mutation operator chooses two points of the same individual (list), to generate another individual, table

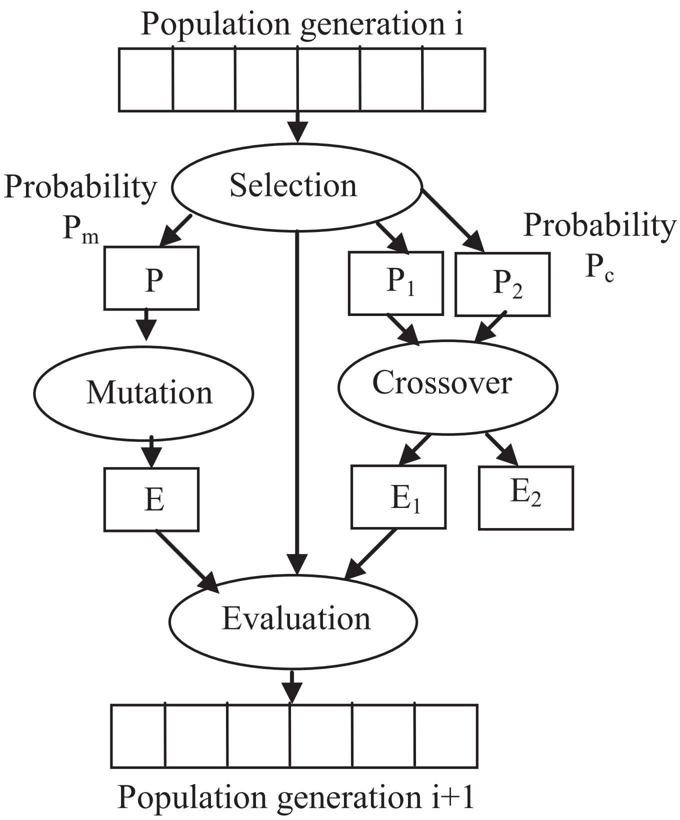

Figure 1: Genetic Algorithm principle

3. The crossover at a point operator chooses, two individuals' parents to generate two other individuals' children starting from only one point, table 4. And the two points crossover operator chooses, two individuals to generate two other individuals starting from two points, table 5.

Table 3: Mutation algorithm

Beginning

1. Choose two positions i and j of the same individual, for each position correspond an operation O_i and O_j,
2. Permute between the operations O_i and O_j to obtain the child,
3. Update the child,
4. Calculate C_1 ,C_2 ,C_3 of the new individual according to the "dates calculation algorithm",

End

3.3 Multi-objective evaluation approach

Generally, the considered criteria present nonlinear and complexes relations and do not have the same importance from the point of view of decision maker. Thus, much of considerations can be retained to take account of all these difficulties. With this intention, a fuzzy method evaluation is proposed. This method is based on the steps which follow [13]:

For each objective function a lower bound is calculated as follow:

$$C_i(x) \geq C_i^b \ \ \forall x \in S, \quad 1 \leq i \leq n_c \tag{7}$$

where S represents the space of realisable solutions and n_c the number of objective functions.

The fuzzification is applied by the functions described, figure 2.

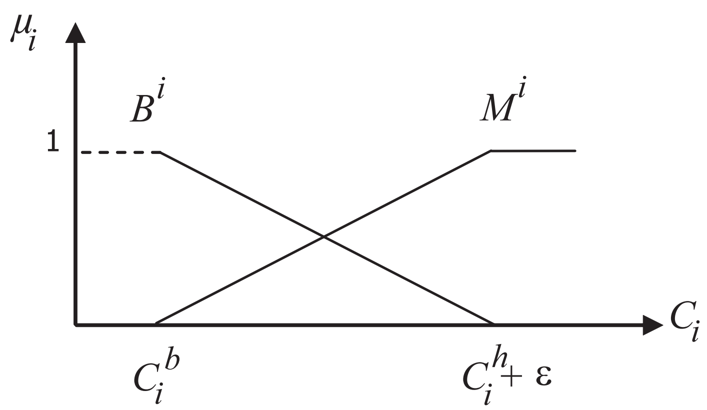

Figure 2: Fuzzy application in the resolution of the scale problem

For each realizable solution x, a vector $C(x)$ is associated, $C(x) \in \left[C_1^b, +\infty\right] \times ... \times \left[C_{n_c}^b, +\infty\right]$, then $C(x) = (C_1(x), ..., C_{n_c}(x))^T$; for each vector $C(x)$, a fuzzification of their components is proposed and

Table 4: Crossover I algorithm

Beginning

1. Choose two individuals P_1 and P_2, and a crossover point
2. Go to all the operations

   While $i < n$ *do*

   - *If* $j < i$ *then*
     - copy the operations of the P_1 in the child1
     - copy the operations of the P_2 in the child2
   - *Else* copy from P_2 (respectively P_1) with the same position, the missing operations in the child1(respectively child2)
   - *End If*

   End while

3. Finish the construction of the child1 (respectively of the child2) with the missing operations (by respecting the order)
4. Update the child1 and the child2
5. Calculate C_1, C_2, C_3 of the two new individuals according to the "dates calculation algorithm"

End

Table 5: Crossover II algorithm

Beginning

1. Choose two individuals P_1 and P_2 and two crossover points i and k
2. Go to all the operations

   While i < n do

   - Copy the operations of the P_1, which precede the first crossover point and which follows the second crossover point, in the child1
   - Copy the operations of the P_2, which precede the first crossover point and which follows the second crossover point , in the child2
   - Copy, with the same position, the missing operations of the P_2 in the child1
   - Copy, with the same position, the missing operations of the P_1 in the child2

   End while
3. Finish the construction of the child1 (respectively of the child2) with the missing operations (by respecting the order)
4. Update the child1 and the child2
5. Calculate C_1, C_2, C_3 of the two new individuals according to the “dates calculation algorithm”

End

considered as two sub-sets B^i and M^i, figure 2.

$$\text{if } C_i(x) \in \left[C_i^b,\, C_i^h + \varepsilon\right] \text{ then } \mu_i^B(C_i(x)) = \frac{\mathrm{C}_i^h - \mathrm{C}_i(\mathrm{x}) + \varepsilon}{C_i^h - C_i^b + \varepsilon},\ \text{else } \mu_i^B(C_i(x)) = 0 \tag{8}$$

when C_i^hrepresents the maximum value of the solution given by a considered heuristics according to the i^{th} objective function.

$\mu_i^B(C_i(x))$ is considered as the fuzzy measurement of $C_i(x)$ in the sub-set B^i.

Then, the quality of each solution is characterized by the vector $C_B(x)$ where all the components are homogeneous since they belong to the same interval and are all without dimension.

$$\begin{array}{l} C_B(x) = (a_1, ..., a_{n_c})^T \\ a_i = \mu_i^B(C_i(x)), \quad \forall i = 1, 2, ..., n_c \end{array} \tag{9}$$

For the multi-objective evaluation, the objective function $C_g(x)$ is reduced to the minimization of the balanced sum of the criteria relating to the use of the aggregation operator OWA [17].

$$C_g(x) = \sum_{i=1}^{n_c} w_i a_i \tag{10}$$

A set of Pareto-optimal solutions is built without according privilege to a particular search direction, to help the decision maker when it cannot clearly give a particular preference to an objective function. This approach is based on an algorithm in which, the objective function $C_g(.)$, defined at the relation (10), is used for the evaluation of solutions. Weightings w_i $(1 \leq i \leq n_c)$ are calculated by using a fuzzy rule. The idea is to measure the average quality of the solutions according to each criterion for each iteration and to calculate the various weights according to the degree of this quality.

The goal is to study the profits and the possible improvements of the solutions by giving the priority to the optimization of the objective functions whose average values is far from the lower bound. This approach is called aggregative approach with dynamic search direction. Let $\overline{C}_i^k$ be the solutions average of the i^{th} objective function found at k^{th} iteration.

$$\overline{C}_i^k = \frac{\sum\limits_{x \in P_k} C_i^k(x)}{card(P_k)} \tag{11}$$

where P_k represents the solutions population at this iteration.

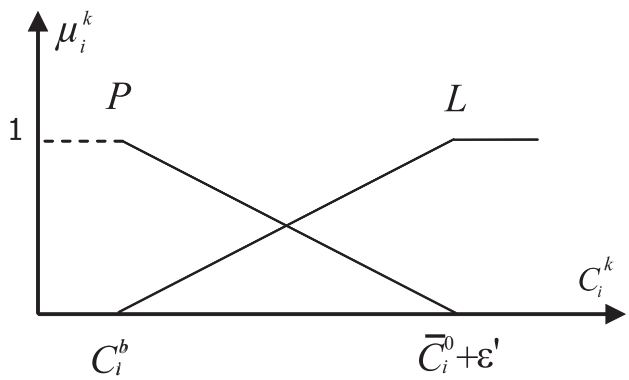

Figure 3: Criteria membership function

For each vector $C(x)$, a fuzzification is applied to its components $C_i(x)$ according to their positions in the interval $\left[C_i^b,\, \overline{C}_i^0 + \varepsilon'\right]$; where ε' is a little positive value introduced to avoid the division by zero, if $\overline{C}_i^0 = C_i^b$ then $\varepsilon' = 0.1C_i^b$, else $\varepsilon' = 0$.

The evaluation of the solutions quality is done by using the membership functions defined in figure 3, relating to the two fuzzy subsets, “P" and “L" of the lower bound.

The membership functions can thus be formulated as follows:

$$\text{if } \overline{C}_i^k \in \left[C_i^b,\, \overline{C}_i^0 + \varepsilon'\right] \text{ then } \mu_{ik}^L\left(\overline{C}_i^k\right) = \frac{\overline{C}_i^k - C_i^b}{\overline{C}_i^0 - C_i^b + \varepsilon'} \text{ else } \mu_{ik}^L\left(\overline{C}_i^k\right) = 1 \tag{12}$$

The calculation of various weightings is carried out by using the two following fuzzy rules:

- If (C_i^k is “P” fromC_i^b) then (w_i^{k+1} decrease)
- If (C_i^k is “L” fromC_i^b) then (w_i^{k+1} increase)

Which lead to the following expression:

$$w_i^k = \frac{\mu_{ik}^L\left(\overline{C}_i^k\right)}{\sum_{j=1}^{n_c} \mu_{jk}^L\left(\overline{C}_j^k\right)},\ \forall i\, \forall k \tag{13}$$

where $1 \leq i \leq n_c$ and $2 \leq k \leq Q$, with Q the total number of iterations and “L” the index relating to the fuzzy subset.

w_i^1 corresponds at the first iteration defined as follow:

$$w_i^1 = \frac{1}{n_c},\ \forall i = 1, \ldots, n_c \tag{14}$$

The various weighting vectors $(W^1, W^2, \ldots, W^Q)$ are gradually calculated from the k^{th} generation P_k at the generation P_{k+1}, according to the distance between the lower bounds and the average of the k^{th} generation individuals, represented by a black circle in the figure 4.

The objective is to improve of the solutions by giving the priority to the objective functions optimization whose average of the values is far from the lower bound. Indeed, by using a fuzzy rule, it is possible to control the search direction in order to build a final set with solutions approaching as much as possible the optimal values.

This method, can be used when the decision maker cannot give a particular preference to an objective function, it also makes it possible to generate weights of the different criteria from an iteration to another in a dynamic way according to the average of the solutions.

4 Simulation

To illustrate the effectiveness and performance of the proposed approach, six representative examples based on practical data have been selected to compute. These examples deal with 5 to 10 operations. The proposed approach is applied to them to optimize three criteria, represented in eqs. (1-3).

For example, the data relating to the example which treats 10 operations and which treats 5 operations is represented respectively in table 6 and table 7.

By application of the proposed approach, the following experimental results are obtained, table 8.

The different results show that the solutions obtained are generally acceptable and satisfactory. The values of the different objective functions show the efficiency of the suggested approach, table 8.

Moreover, the proposed method enables us to obtain good results in a polynomial computation time. In fact, the various values of the criteria given by the multiobjective optimization method by Pareto-optimality show its effectiveness, table 8. The values of the criteria for the Pareto border are in the neighbourhood of to the lower bounds. Indeed, such an approach makes it possible to generate Pareto-optimal solutions of good quality.

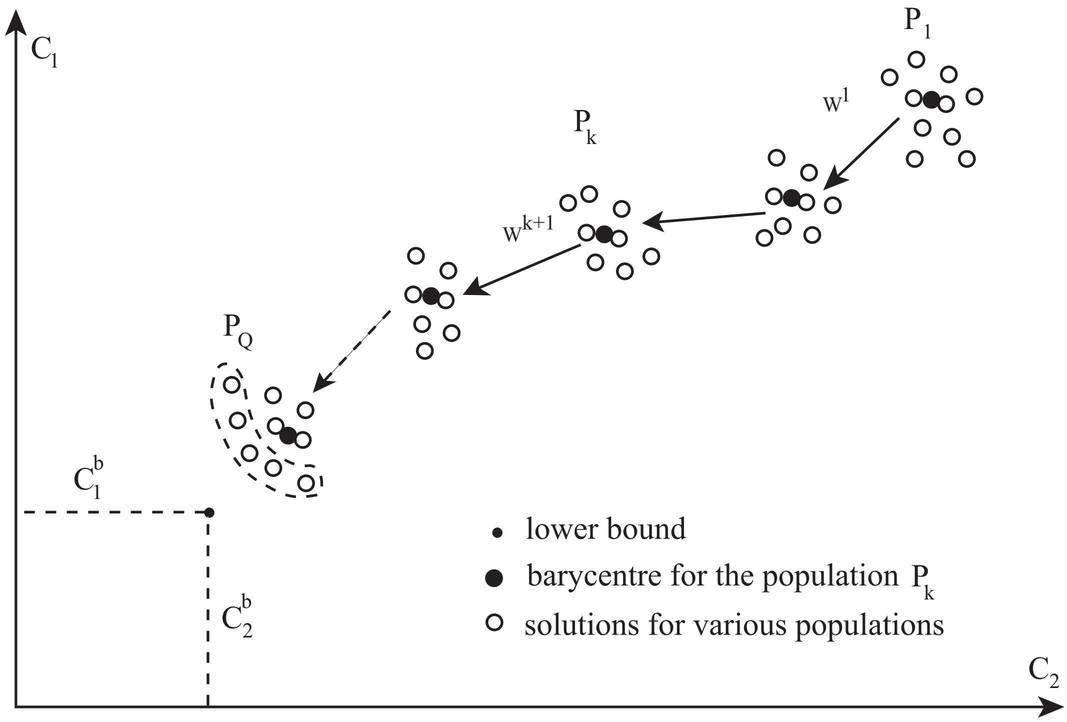

Figure 4: Search direction

Table 6: data relating to 10 operations

	O_1	O_2	O_3	O_4	O_5	O_6	O_7	O_8	O_9	O_{10}
r_k	0	1	2	3	4	1	3	2	1	3
p_k	1	2	4	2	1	2	1	3	2	4
v_{i1}	13	14	13	13	12	7	7	13	9	9
v_{i2}	15	14	5	14	13	15	15	16	15	15
v_{i3}	-	12	13	12	11	14	14	12	14	14
P_{i1}^{rev}	2	3	4	3	2	1	1	2	2	2
P_{i2}^{rev}	1	2	2	4	3	2	2	1	4	1
P_{i3}^{rev}	-	4	3	2	2	1	3	2	3	3
Dv_{P_k}	35	32	35	33	35	36	31	34	36	31
Dr_{P_k}	14	10	9	11	8	12	7	9	11	10
$d_{P_k}^{liv}$	21	22	25	22	21	20	21	24	26	22
$C_{P_k}^{stk}$	3	2	2	5	2	3	4	3	1	2
$P_{P_k}^{ven}$	4	6	6	8	7	5	6	8	3	5

Table 7: data relating to 5 operations

	O_1	O_2	O_3	O_4	O_5
r_k	2	3	1	4	3
p_k	1	2	4	2	3
v_{i1}	3	3	4	6	5
v_{i2}	4	1	2	4	3
v_{i3}	2	-	3	-	4
P_{i1}^{rev}	1	2	1	2	4
P_{i2}^{rev}	3	1	2	4	2
P_{i3}^{rev}	4	-	3	-	1
Dv_{P_k}	14	16	10	11	14
Dr_{P_k}	5	6	4	7	5
$d_{P_k}^{liv}$	10	13	14	16	12
$C_{P_k}^{stk}$	1	2	3	4	5
$P_{P_k}^{ven}$	8	3	8	5	4

Table 8: experimental results

n	Scheduling	C_1	C_2	C_3	$C_g(.)$
10	$O_1O_3O_5O_9O_4O_2O_7O_6O_8O_{10}$	14	4	24	0,915
9	$O_1O_2O_3O_4O_5O_7O_6O_8O_9$	9	4	20	0,95
8	$O_1O_7O_5O_2O_3O_8O_4O_6$	14	1	18	0,963
7	$O_1O_3O_6O_7O_4O_2O_5$	12	9	17	0,752
6	$O_1O_4O_3O_5O_2O_6$	4	10	14	0,977
5	$O_1O_4O_5O_3O_2$	12	10	14	0,53

5 Conclusion

A new approach based on the hybridization with the Pareto-optimality for solving multiobjective problems in agro-alimentary workshop, is presented. The approach developed in this work provides the possibility to determine an optimal scheduling among several realizable ones; this optimal solution generates the minimization of objective function (10). Besides, the proposed approach uses Pareto to estimate and to classify obtained decisions. Indeed, we can avoid the preemption of certain components, the cost of the out-of-date products, the cost of the distribution discount and the completion time(makespan).

The proposed hybrid approach presented in this paper can be considered as effective mechanisms from the computation complexity.

References

[1] S. Cavalieri, P. Gaiardelli, Hybrid genetic algorithms for a multiple-objective scheduling problem, *Journal of Intelligent Manufacturing*, Vol.9, pp.361-367, 1998.

[2] Y. Collette, P. Siarry, Optimisation multiobjectif, *Editions Eyrolles*, Paris, 2002.

[3] L. Davis, Job shop scheduling with genetic algorithm, *Proceedings of the first International Conference on Genetic Algorithms*, Lawrence Erlbaum Associates, pp.136-140, 1985.

[4] F. Della Croce, R. Tadei, G. Volta, A genetic algorithm for the job shop problem, *Computer and Operations Research*, Vol. 22, No 1, pp. 15-24, 1995.

[5] U. Dorndorf, E. Pesch, Evolution based learning in a job shop environment, *Computers and Operations Research*, Vol. 22, pp.25-40, 1995.

[6] E. Gargouri, S. Hammadi, A distributed scheduling for agro-food manufacturing problems, *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, Vol. 33, No 2, 2003.

[7] E. Gargouri, S. Hammadi, P.Borne, New constraints of agro-food industry scheduling problem, *Acte de IFDICON'2001, Europeen Workshop on Intelligent Forecasting, Diagnostic and Control*, 24-28 juin, pp. 73-80, Santorin, 2001.

[8] A. K. Jain, H. A. El Maraghy, Single process plan scheduling with genetic algorithm, *Production Planning and Control*, Vol.8, No 4, pp.363-376, 1997.

[9] I. Kacem, S. Hammadi, P. Borne, Pareto-optimality approach for flexible job-shop scheduling problems: Hybridization of evolutionary algorithms and fuzzy logic, *Math. and Computers in Sim.*, 60, pp. 245-276, 2002.

[10] K. Mesghouni, Application des algorithmes évolutionnistes dans les problèmes d'optimisation en ordonnancement de la production , PhD Thesis, *Université des Sciences et Technologiques de Lille*, Lille, 1999.

[11] M. Mori, C. Tseng, Genetic algorithms for multimode resource constrained project scheduling problem, *European Journal of Operational Research*, Vol.100, No 1, pp.134-141, 1997.

[12] R. Nakano, T. Yamada, Conventional genetic algorithm for job shop problems, *Proceedings of the 4th International Conference on Genetic Algorithms*, University of California, pp. 474 -479, 1991.

[13] I. Saad, M. Benrejeb, Optimisation multicritère par Pareto-optimale des problèmes d'ordonnancement en tenant compte du coût de production, *Revue Sciences et Technologies de l'Automatique, e-STA*, Vol. 3, No. 1, 2006.

[14] M. Sakawa, Genetic algorithms and fuzzy multiobjective optimization, *Dordrecht: Kluwer Academic*.

[15] E. G. Talbi, Métaheuristiques pour l'optimisation combinatoire multiobjectif, *Tutorial, Journées Evolutionnaires Trimestrielles*, Paris, 1999.

[16] F. Tangour, S. Hammadi, P. Borne, M. Benrejeb, Ordonnancement dynamique dans un atelier de production agroalimentaire, *Séminaire d'Automatique-Industrie, SAI'06*, Matmata, 2006.

[17] R. R. Yager, On weighted median aggregation operators in multicriteria decision making, *IEEE Trans. on Systems, Man and Cybernetics*, 18, pp. 183-190, 1988.

[18] E. Zitzler, L. Thiele, Multi-objective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. *IEEE Transactions on Evolutionary Computation*, Vol.3, No 4, pp. 257-271, November, 1999.

Fatma TANGOUR[1,2], Ihsen SAAD[1,2]

[1]Ecole Nationale d'Ingénieurs de Tunis
Unité de recherche LARA-Automatique
BP 37, Le Belvédère, 1002 Tunis, Tunisie

[2]Ecole Centrale de Lille, Cité scientifique
Laboratoire d'Automatique, Genie Informatique et Signal
BP 48, 59651 Villeneuve d'Ascq Cedex, France

E-mail: {fatma.tangour, ihsen.saad}@enit.rnu.tn

Editor's note about the authors:

Fatma TANGOUR was born in 1969 in Nabeul, Tunisia. She graduated from "Ecole Normale Superieure de l'Enseignement Technique" and obtain the Master of automatic and signal treatment in 2004 at the "Ecole Nationale d'Ingénieur de Tunis". She is currently preparing the Ph.D. degree in automatic and computer science within the framework of LAGIS-EC-Lille and LARA-ENIT cooperation. Her research interests are in the area of optimization methods for discrete events systems, computer science and operational research.

Ihsen SAAD was born in Monastir, Tunisia, in 1977. He received the Engineer Diploma degree in electrical automatic control engineering from the "Ecole Nationale d'Ingénieur de Gabès", Tunisia, in 2002. He obtain the Master of automatic and signal treatment in 2004 at the "Ecole Nationale d'Ingénieur de Tunis". He is currently preparing the Ph.D. degree in automatic and computer science within the framework of LAGIS-EC-Lille and LARA-ENIT cooperation. His research is related to the evolutionary optimization methods for discrete events systems, computer science and operational research.