Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Scheduling of Uncertain Multi-product Batch Processes Under Finite Intermediate Storage Policy

XU Zhen-hao * GU Xing-sheng JIAO Bin *****

** Research Institute of Automation, East China University of Science and Technology, Shanghai 200237 China (e-mail: xuzhenhao@ecust.edu.cn)*
*** Research Institute of Automation, East China University of Science and Technology, Shanghai 200237 China (e-mail: xsgu@ecust.edu.cn)*
**** Electrical Engineering Department, Shanghai Dianji University, Shanghai 200237 China (e-mail: binjiaocn@163.com)*

Abstract: There are various uncertainties in the production scheduling of the real-world applications. And in multi-product batch processes, the intermediate storage often can be used to increase plant productivity and operational efficiency. In this paper, the scheduling mathematical model for multi-product batch processes under Finite Intermediate Storage (FIS) policy with uncertain processing time has been established based on fuzzy programming theory. And the Maximum Membership Functions of Mean Value method is applied to convert the fuzzy scheduling model to the general optimization model. Furthermore, a fuzzy immune scheduling algorithm combined with the feature of the Immune Algorithm is proposed, which can prevents the possibility of stagnation in the iteration process and achieves fast convergence for global optimization. The effectiveness and efficiency of the fuzzy scheduling model and the proposed algorithm are demonstrated by simulation results.

1. INTRODUCTION

Even though continuous operation is prevalent and desirable in many chemical processes, batch processing is still very important and has a permanent place in the chemical process industry. It is especially widely used for producing pharmaceuticals, polymers, food and specialty chemicals, which involve high-value-added, low-volume products and complex processing requirements. Batch processes are economically desirable when a large number of products are made using similar processing units, but productivity of their processes is heavily dependent on production scheduling (Ku *et al*., 1987).

During the latest decade, scheduling problems for multi-product processes have received considerable attention and some research were made for completion time algorithm and mathematical models for optimal scheduling considering various intermediate storage policies, such as unlimited intermediate storage, no intermediate storage, finite intermediate storage, zero wait, mixed intermediate storage (Jung *et al*., 1994). Intermediate storage has an important role in improving operating efficiency by decoupling the periodic operation of adjacent batch or semi continuous units. In addition, batch operators are susceptible to high levels of processing variability and operator vagaries and error. These kinds of process parameter variations can also be mitigated by intermediate storage if an adequate size of storage facility and an appropriate level of initial holdup are chosen (Kim *et al*., 1996).

For most scheduling problems, some data are treated as determinative values. But it is not suitable for all actual situations, because uncertainty is a natural feature of real-world applications. Recently, fuzzy set theory is more and more frequently used to describe uncertainties. And McCahon and Lee (McCahon *et al*., 1992) were the first to illustrate the application of fuzzy set theory as a means of analyzing performance characteristics for a flow shop system. In the paper, the uncertain processing time of each job will be considered when solving the scheduling problems.

In this study, the scheduling problems under finite intermediate storage policy for multi-product batch processes will be researched. It is organized as follows. In Section 2, the definition and mathematical model of the problems that we intend to treat are given. The proposed fuzzy scheduling method is introduced and described at length in Section 3. Section 4 presents the effectiveness of the scheduling model and the proposed algorithm through results of computational simulation. Conclusions from this work are drawn in Section 5.

2. PROBLEM DESCRIPTION

In many real world applications, job processing time may vary dynamically with the situation. If the time required to process each job is uncertain, the completion time of a scheduling schema is apparently also uncertain.

2.1 Problem Definition

Mathematically, Multi-product batch processes scheduling problems can be described as: there are N products which need to be processed, and the number of processing units is M . The processing time of products i in unit j is $\widetilde{T}_{i_k j}$, where i_k is the order of product i in processing procedure, which includes the transfer time, the set-up time and the clean-up time, etc. Because it is mutative and uncertain, it is represented by the triangular fuzzy number.

Every product has the same processing sequence in all units, $\widetilde{S}_{i_k j}$ and $\widetilde{C}_{i_k j}$ respectively represents the starting time and the finishing time of product i in unit j . For uncertainty of the processing time of each product, the starting time and the finishing time are also uncertain. Accordingly, $\widetilde{S}_{i_k e}$ and $\widetilde{T}_{i_k e}$ mean the starting time and the finishing time of the last operation of product i . And $BTW_{i_k j}$ is the maximal storage time between unit j and unit $j+1$ of product i . The formulation of the objective function is makespan, that is, the last product should be completed as soon as possible.

2.2 Mathematical Model

According to those assumptions of the multi-product batch processes under FIS policy, the scheduling problem can be modeled as follows:

$$min \quad \{ \widetilde{Z} = max \quad \left(\widetilde{S}_{i_k e} + \widetilde{T}_{i_k e}\right) \}$$

$$s.t. \qquad \widetilde{S}_{i_k j} \geq \widetilde{S}_{(i_k-1)j} + \widetilde{T}_{(i_k-1)j} \tag{1}$$

$$\widetilde{S}_{i_k j} \geq \widetilde{S}_{(i_k-1)(j+1)} + \widetilde{T}_{(i_k-1)(j+1)} - BTW_{i_k j} - \widetilde{T}_{i_k j} \quad i \in N,\ j \in M \tag{2}$$

$$\widetilde{S}_{i_k j} \geq \widetilde{S}_{(i_k-1)(j+1)} - \widetilde{T}_{i_k j} \tag{3}$$

Equation (1) is the resource constraint of products, which represents that the production of product i_k cannot start in unit j until completing previous product $i_k - 1$ by unit j, which is a processing unit can not process the different products at the same time.

Equation (2) means that when the intermediate products are transferred into the storage unit, the residence time in intermediate storage unit for storing the intermediate product can not more than the maximal storage time of the storage unit.

Equation (3) represents that there is only one product can store in the intermediate storage unit between the downstream unit and upstream unit.

And the objective function intends to find out the best production sequence to minimize the makespan.

2.3 Description of the Solution

A triangular fuzzy number is given to describe the uncertain processing time of products. The membership function of a triangular fuzzy number is interpreted as a possibility distribution, determined by the following three points:

$$\widetilde{A} = \left(A^L,\ A^M,\ A^U\right)$$

Where, A^L is the lower limit, A^M means the center value, and A^U is the upper limit. In the scheduling problem, the three points can be represented the uncertain processing time as: the optimistic time, the most likely time and the pessimistic time.

Arithmetic operations on fuzzy numbers that will be used for the objective function and the constraints, such as addition, subtraction, taking the maximum of two fuzzy numbers are defined based on the extension principle, which allows the extension of operations on real numbers to fuzzy numbers. In order to compute the completion times of products with fuzzy durations, the addition and maximum operations are needed.

Let $\widetilde{x} = \left(x^L, x^M, x^U\right)$, $\widetilde{y} = \left(y^L, y^M, y^U\right)$ be triangular fuzzy numbers, the addition and maximum of fuzzy numbers are defined on the basis of the extension principle as follows:

Addition: $\widetilde{x} + \widetilde{y} = \left(x^L + y^L, x^M + y^M, x^U + y^U\right)$

Maximum: $\widetilde{x} \vee \widetilde{y} = \left(x^L \vee y^L, x^M \vee y^M, x^U \vee y^U\right)$

Because the resolvability of the fuzzy addition and maximum operations, the details of solution can be described:

If $i = 1, \quad j = 1$:

$$\widetilde{S}_{i_k j} = 0, \qquad \widetilde{C}_{i_k j} = \widetilde{S}_{i_k j} + \widetilde{T}_{i_k j} = \widetilde{T}_{i_k j} \tag{4}$$

If $i = 1, \quad j > 1$:

$$\widetilde{S}_{i_k j} = \widetilde{C}_{i_k (j-1)}, \quad \widetilde{C}_{i_k j} = \widetilde{S}_{i_k j} + \widetilde{T}_{i_k j} = \widetilde{C}_{i_k (j-1)} + \widetilde{T}_{i_k j} \tag{5}$$

If $i > 1, \quad j = 1$:

$$\widetilde{S}_{i_k j} = max\begin{pmatrix} \widetilde{C}_{(i_k-1)j}\ ,\ \widetilde{C}_{(i_k-1)(j+1)} - BTW_{i_k j} - \widetilde{T}_{i_k j}\ , \\ \widetilde{S}_{(i_k-1)(j+1)} - \widetilde{T}_{i_k j} \end{pmatrix}, \quad \widetilde{C}_{i_k j} = \widetilde{S}_{i_k j} + \widetilde{T}_{i_k j} \tag{6}$$

If $i > 1, \quad j > 1$:

$$\widetilde{S}_{i_k j} = max\begin{pmatrix} \widetilde{C}_{(i_k-1)j}\ ,\ \widetilde{C}_{i_k (j-1)}\ ,\ \widetilde{C}_{(i_k-1)(j+1)} - BTW_{i_k j} - \widetilde{T}_{i_k j}\ , \\ \widetilde{S}_{(i_k-1)(j+1)} - \widetilde{T}_{i_k j} \end{pmatrix}, \quad \widetilde{C}_{i_k j} = \widetilde{S}_{i_k j} + \widetilde{T}_{i_k j} \tag{7}$$

The objective function is:

$$\begin{aligned} min(makespan) &= min\left(max\ \widetilde{S}_{i_e j} + \widetilde{T}_{i_e j}\right) \\ &= min\left(\widetilde{C}_{i_e j}\right) \\ &= min\left(C^L_{i_e j}, C^M_{i_e j}, C^U_{i_e j}\right) \end{aligned} \quad (8)$$

Based on the operations of different cases, the Maximum Membership Function of Mean Value (MMFMV) is applied to transform the fuzzy models into the singular-objective accurate model.

Firstly, let us assume that $\widetilde{Z}_r = \widetilde{C}^r_{i_e j}(r = L, M, U)$, and the Zimmermann method (Badell *et al*., 1998) is used to define the Positive Ideal Solution (PIS) and the Negative Ideal Solution (NIS) of the three objective functions, that is $Z^{PIS}_r(r = L, M, U)$ and $Z^{NIS}_r(r = L, M, U)$. And every $C^{PIS}_{i_e j}(r = L, M, U)$ and $C^{NIS}_{i_e j}(r = L, M, U)$ mean the optimistic solution and the pessimistic solution, the membership function of the Agreement Index of $\widetilde{Z}_r(r = L, M, U)$ is:

$$\mu_{Z_r}(x) = \begin{cases} 0, & x > Z^{NIS}_r \\ \dfrac{x - Z^{PIS}_r}{Z^{NIS}_r - Z^{PIS}_r}, & Z^{PIS}_r \le x \le Z^{NIS}_r \quad r = L, M, U \\ 1, & x < Z^{PIS}_r \end{cases} \quad (9)$$

Then, the fuzzy scheduling model can be transformed as the singular objective nonlinear programming:

$$max\left\{\Gamma\alpha^L + (1-\Gamma)\alpha^U\right\} \quad (10)$$

$$\text{s.t.} \quad \alpha^L \le \mu_{Z_r} \le \alpha^U, \qquad i = L, U \quad (11)$$

$$\alpha^U \le \mu_{Z_M} \quad (12)$$

$$\alpha^L, \alpha^U \in [0, 1] \quad (13)$$

Where α^L is decided by the minimum value of $U_{Z_r}(x)(r = L, M, U)$ and α^U is determined by the maximal value of $U_{Z_r}(x)(r = L, M, U)$. The actual decision-making process, often in the hope that the target will be obtained with the highest level of satisfaction in the most possible situation, not gain the greatest satisfaction in the worst or optimal cases.

So, in the above model, let $U_{Z_M}(x)$ be the maximum of the membership value. However, the minimum value will be produced in the worst or the best circumstances. And the operator Γ is used to reflect the decision-makers in a positive and the negative decision-making tendencies, the smaller the value of Γ, the more active decision-making, on the contrary, the more negative decisions.

3. THE FUZZY SCHEDULING ALGORITHM

The Immune Algorithm is inspired by the characteristic of the natural immune system. It promotes diversification; instead, it evolves antibodies which can handle different antigens. Furthermore, the algorithm operation on the memory cell will achieve very fast convergence during the search process. So, these features make it widely used in many fields, such as Intelligent Control, Pattern Recognition and Optimization Design (John *et al*., 1996, Shao *et al*., 2000). Based on the simple IA, the improved IA for uncertain scheduling problems of multi-product batch processes is proposed, which can be used effectively to solve the scheduling problems of multi-product batch processes under FIS policy. The computation steps of the fuzzy scheduling algorithm are discussed below.

Step 1: Encoding method.

The scheduling problem of multi-product batch processes is Sequencing problem or Ordering Problem, and the objective function is affected not only by the value of the solution, but also its position in the coding string. For the reason, the character coding can be applied to reduce the time and complexity of encoding and decoding. Taking into account the production characteristics of the multi-product batch processes, every product is represented by a character, and it can appear only once in the coding. Then, the sequence of the characters in the coding is the job sequence.

Step 2: Initial antibody population formulation.

In the algorithm, the antigen and antibody are treated as the objective and the feasible solution of a conventional optimization method. In the initial step, the antibodies are generated randomly in the feasible space. A population pool comprises these antibodies. And a group of genes form an antibody. Each antibody represents a possible solution to a schedule of products.

Step 3: Affinity calculation.

The concept of information entropy is introduced as a measure of diversity for the population to avoid falling into a local optimal solution. Two affinity expressions are considered in the approach. One is to elucidate the relationships between the antibody and the antigen, where the combination intensity between the objective and the solution are investigated. The other accounts for the degree of association between antibodies, where the mutual diversity of antibodies is evaluated.

The affinity between the antigen and antibody:

$$ax_v = \frac{1}{1 + opt_v} \qquad v \in N \quad (14)$$

Where, ax_V is the affinity between the antigen and antibody v, and opt_v is the combination intensity between the antigen and the antibody v. In this paper, opt_v is expressed by the total production period, which is makespan.

The affinity between antibodies:

$$ay_{vw} = \frac{1}{1+H(2)} \qquad v, w \in N \tag{15}$$

Where ay_{vw} is the affinity between the v th antibody and the w th antibody, and $H(2)$ is the information entropy of the antibody i and j only. However, the affinity value is computed between zero and one. And $H(2)$ can be computed as below.

If there are N antibodies in the antibody pool, each has M genes. From the information theory, the entropy $H_j(N)$ of the j th gene($j = 1, 2, \cdots M$) can be computed as given below:

$$H_j(N) = \sum_{i=1}^{s} \left(-P_{ij} \log P_{ij} \right) \tag{16}$$

Where P_{ij} is the probability that the i th allele comes out of the j th gene. Note that, if all alleles at the j th genes are the same, the entropy of the j th is zero. From Eq.(16), the diversity of genes can be estimated and the average information entropy can be also defined as below:

$$H(N) = \frac{1}{M} \sum_{j=1}^{M} H_j(N) \tag{17}$$

Where, M is the size of genes in an antibody. Then, the entropy can illustrate the diversity of the antibody population.

Step 4: Renewal of the memory pool.

Based on the results of the computation procedure mentioned earlier, the antibody which has high affinities with the antigen is added to the new memory pool. As most selected antibodies exhibit higher affinities with the antigen, the averaged affinity of the new population pool will be higher than that of the original pool. Therefore, a new antibody chosen from the pool comes with a higher affinity with the antigen. Owing to the size constraint of the memory pool, the new antibody can replace the antibody that has the highest affinity with the new one. The average affinity of a new antibody pool chosen from this new memory pool is higher than that of the old antibody pool.

Step 5: Boost or restriction of antibody generations

The density and the expected propagation proportion of antibodies are the criterion, when the individual of population is evaluated. The density is defined as the ratio of the single individual to the population, and the expected propagation proportion of the antibody represents the selection ratio in the populations.

The density can be defined as:

$$R_v = \frac{1}{N} \sum_{w=1}^{N} K_{vw} \qquad v, w \in N \tag{18}$$

Where, $K_{vw} = \begin{cases} 1 & ay_{vw} \geq T \\ 0 & ay_{vw} < T \end{cases}$ and T is the threshold.

In contrast, the expected propagation proportion is expressed as follows:

$$E_v = \frac{ax_v}{R_v} \qquad v \in N \tag{19}$$

The promotion and suppression of the proliferation of antibodies have much to do with the density and affinity.

Step 6: updating of antibody population.

Based on the new antibodies selected from the memory pool, the crossover and mutation of the new antibodies are performed. Crossover is a random process of recombination of strings. With the probability of crossover, a partial exchange of characters between two strings is performed. With the crossover operation, the proposed algorithm is able to acquire more information with the generated individuals. The search space is thus extended and more complete. Mutation is the occasional random alteration of the bits in the string. Even though the selection and crossover process can effectively search and recombine, they do not introduce any new information into the bit level. The mutation operator helps reproduce some individuals that may be vital to the performance. Through the operations of crossover and mutation, the antibody population can be updated.

Step 7: Decisions

All the antibodies in each generation must be evaluated. The antibodies with higher affinity are tracked to the memory pool for each generation. If the termination criterion is satisfied or no further improvement in relative affinity can be obtained, the optimal search will end. And genes of the antibody can be decoded to be solutions of the scheduling problem. Otherwise, the procedure must turns to step 3.

4. COMPUTATIONAL EXPERIENCE

To illustrate the utility of scheduling model and algorithms, consider the following example where ten products have to be processed in five units. The fuzzy processing time data for the example can be found in Table 1. The fuzzy processing times are specified by three parameters, which represent the lower bound, the most likely value and the upper bound on the processing time. And Table 2 is the maximal storage time between the intermediate storage units.

Table 1 the fuzzy processing times of products

JOB	UNIT1	UNIT2	UNIT3	UNIT4	UNIT5
1	(37 40 43)	(13 15 18)	(11 12 13)	(22 25 28)	(8 10 11)
2	(6 7 9)	(39 41 44)	(21 22 23)	(32 36 38)	(7 8 9)
3	(37 41 43)	(136 155 166)	(30 33 35)	(106 121 131)	(145 160 174)
4	(10 12 14)	(69 74 82)	(20 24 27)	(44 48 53)	(72 78 82)
5	(5 7 8)	(82 95 101)	(67 72 78)	(48 52 54)	(140 153 166)
6	(11 12 14)	(11 14 15)	(60 62 68)	(29 32 35)	(131 162 176)
7	(9 11 14)	(6 7 8)	(27 31 33)	(23 26 28)	(29 32 35)
8	(28 31 35)	(37 39 41)	(123 141 158)	(5 6 8)	(17 19 21)
9	(28 32 33)	(88 92 95)	(11 12 13)	(11 14 16)	(93 102 112)
10	(23 27 29)	(105 114 121)	(19 21 22)	(84 90 96)	(48 52 59)

Table 2 the maximal storage time of intermediate storages

JOB	Between U1,U2	Between U2,U3	Between U3,U4	Between U4,U5
1	36	31	71	72
2	65	71	31	59
3	34	97	26	54
4	39	82	34	16
5	27	48	69	35
6	47	63	65	33
7	37	69	68	91
8	61	52	68	31
9	55	26	50	30
10	44	83	42	35

In the algorithm, the population size is 40, the size of memory pool is 20 and the number of iteration is 300. The scheduling algorithm has been executed many times. Figure.1, 2, 3 and 4 are the evolution curves of the algorithm at different Γ .

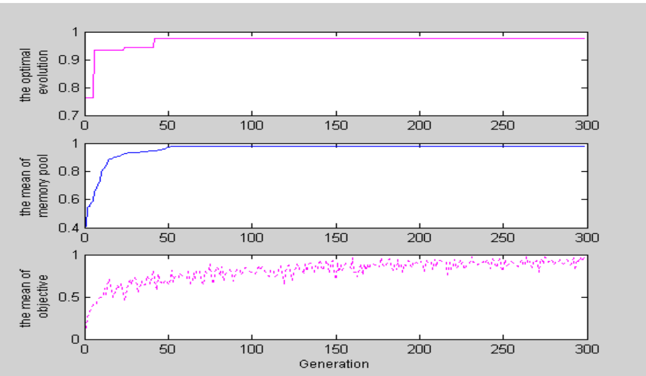

Fig. 1 the evolution curve ($\Gamma = 0.3$)

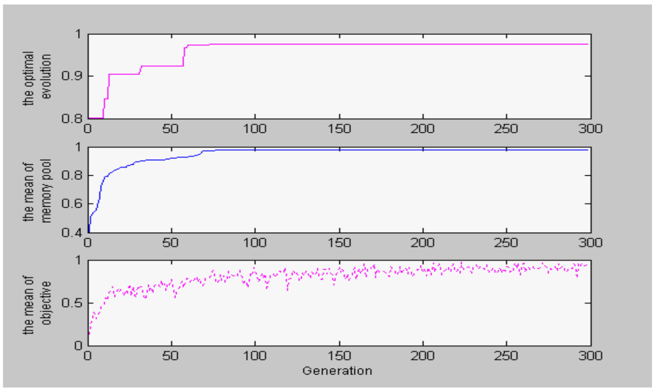

Fig. 2 the evolution curve ($\Gamma = 0.5$)

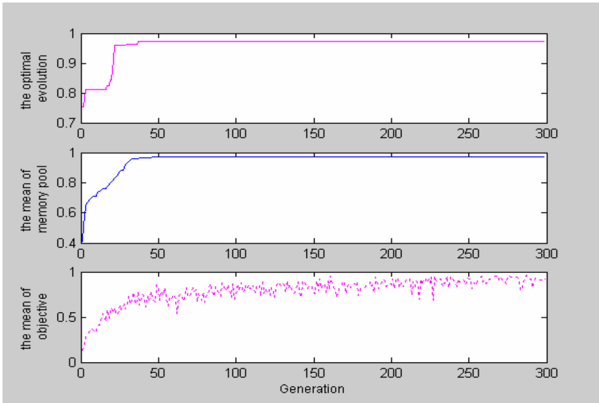

Fig. 3 the evolution curve ($\Gamma = 0.7$)

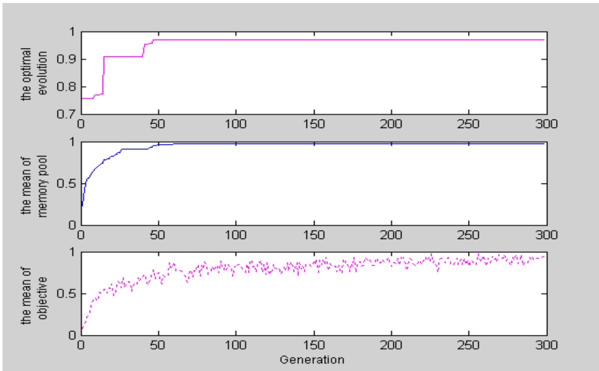

Fig. 4 the evolution curve ($\Gamma = 0.9$)

In each figure, the above curve is the optimal curve, which means the optimal value of each generation in iteration; the middle curve is the mean value of the objective of the memory pool, which represents the average objective of all individuals in antibody population. Along with the evolution of the computing process, the optimal curve is getting to near-one-point, and the mean values of the memory pool and the antibody population tend to be steady. Those indicate the astringency of the method.

Then, the starting time and the finishing time of products can be decided. The Gantt chat of the job sequence of the optimal objective is shown in Fig. 5.

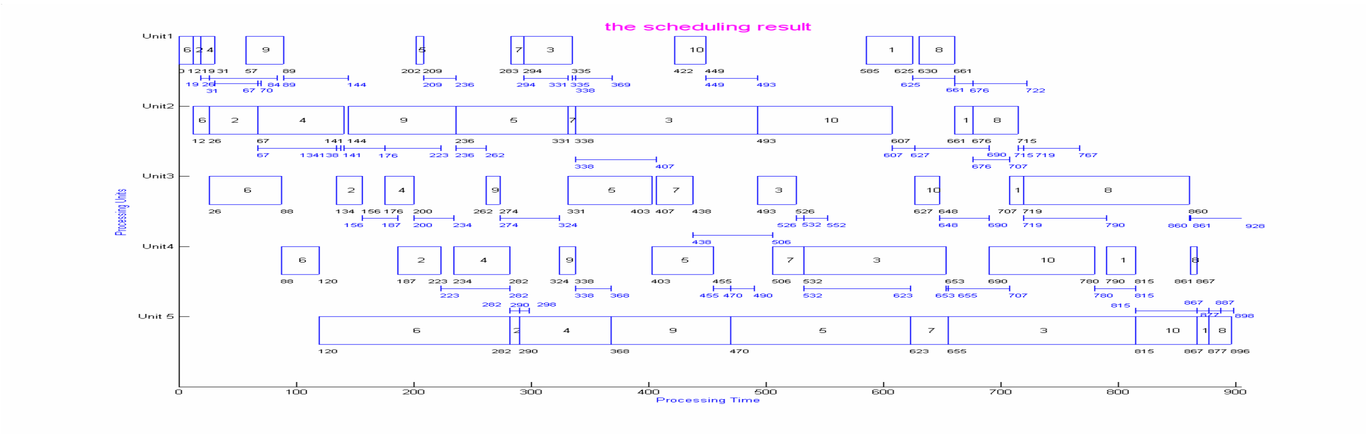

Fig. 5 the Gantt chart of optimal schedule ($\Gamma = 0.3$)

As we can see from the figure, all the jobs are completed in accordance with the optimal job processing sequence. And the residence time of the product which needs to be stored in the intermediate storage unit in the production process is not more than the maximal storage time of the intermediate storage unit.

The scheduling results of the example under different value of Γ can be shown in Table 3.

Table 3 the schedule results with different Γ

Γ	Obj	Sequence	$makespan^L$	$makespan^M$	$makespan^U$
0.1	0.9804	6 2 4 9 5 7 3 10 8 1	802	896	985
0.3	0.9773	6 2 4 9 5 7 3 10 1 8	802	896	985
0.5	0.9748	6 2 4 9 5 7 3 10 8 1	802	896	985
0.7	0.9720	6 1 4 9 5 7 3 10 8 2	801	896	985
0.9	0.9693	6 1 4 9 5 7 3 10 8 2	801	896	985

From the table we can see that the smaller the value of Γ , the larger the value of the objective function, the better the results of scheduling, and decision-making more active; Instead, the objective function value of the smaller, poorer results scheduling, decision-making more conservative. When Γ is continuously from small to large, the objective function value was falling. But in all cases, the algorithm has shown a good convergence, and the schedules results are being with greater satisfaction.

5. CONCLUSIONS

Appropriate scheduling not only reduces manufacturing costs but also reduces the possibility of violating due date. Since the processing times of products in real applications are usually uncertain, how to deal with the scheduling problems under uncertainty in practical reality has important significance. In this paper, the scheduling problem of multi-product batch processes under Finite Intermediate Storage (FIS) policy with uncertain processing time is discussed. The triangular fuzzy numbers is applied to describe the uncertain processing time, and the fuzzy mathematical model is proposed based on the fuzzy set theory. And integrated with the characteristic of the Immune Algorithm, the fuzzy scheduling algorithm is presented to solve the model. By the results obtained from experimental data, it is proven that the feasibility and effectiveness of the scheduling model of multi-product batch processes and the fuzzy scheduling algorithm.

Acknowledgement

This work was supported by National Natural Science Foundation of China (Grant No.60674075 and No. 60774078) , the Key Technologies Program of Shanghai Educational Committee (Grant No. 05ZZ73) and Shanghai Leading Academic Discipline Project (Project No. 13504).

REFERENCES

Badell, M., J.M. Nougues and L. Puigjaner (1998). Integrated on line production and financial scheduling with intelligent autonomous agent based information system. *Computers Chem.Engng*, **22**(Suppl.), 271-278.

John, E.H. and E.C. Denise (1996). Learning using an artificial immune system. *Journal of Network and Computer Applications*, **19**, 189-212.

Jung, J. H., H.K. Lee & I.B. Lee (1994). Completion times and optimal scheduling for serial multi-product processes with transfer and set-up times in zero-wait policy. *Computers & Chemical Engineering,* **18,** 537-544.

Kim, M. S., J.H. Jung & I.B. Lee (1996). Optimal scheduling of multiproduct batch processes for various intermediate storage policies. *Industrial Engineering & Chemical Research*, **35**, 4058-4066.

Ku, H.M., D. Rajagopalan & I.A. Karimi (1987). Scheduling in batch processes. *Chemical Engineering Progress,* **22**, 35-42.

McCahon, C.S. and E.S. Lee (1992). Fuzzy job sequencing for a flow shop. *European Journal of Operational Research,* **62**, 294-305.

Shao, X., Z. Chen and X. Lin (2000). Resolution of multicomponent overlapping chromatogram using an immune algorithm and genetic algorithm. *Chemometrics and Intelligent Laboratory Systems*, **50**, 91-99.

J.Balasubramanian, I.E. Grossman (2003). Scheduling optimization under uncertainty-an alternative approach. *Computers and Chemical Engineering*, 27, 469-490.