# Bi-Criteria Single Batch-Processing Machine with Job Release Time and Non-Identical Job Sizes

Hui-Mei Wang and Fuh-Der Chou

***Abstract*—Bi- or multi-objective scheduling problem is important in practice because it offers two and more managerial indicators simultaneously for decision-makers doing some trade-offs. To date the studies of single batch-processing machine problems with bi-criteria were relatively few compared to the single unit-processing machine. In this paper, we consider a bi-criteria scheduling problem of minimizing the makespan and total weighted tardiness on a single batch-processing machine. An exhausted enumeration approach is provided to find all Pareto-optimal solutions for small-scale problems. In addition, an alternative method called SA-based multi-objective (SAMO) algorithm is also developed for comparison with the exhausted enumeration approach. Computational results revealed that the proposed SAMO algorithm could almost find all Pareto-optimal solutions for the small problem. Obviously, the success on small-scale problems suggests the readiness of the proposed SA for large-scale problems.**

***Index Terms*—Makespan, total weighted tardiness, batch-processing machine, simulated annealing.**

## I. Introduction

Batch-processing machine (BPM) scheduling is an increasingly attractive research topic since the pioneer work of Ikura and Gimple [1]. In the batch-processing machine scheduling problems, the machines can process several jobs simultaneously, in the other words, the jobs should be grouped into a batch as long as the total sizes of jobs in the batch do not exceed the capacity of the batch-processing machines, and the processing time of the machine is represented as the longest processing times of jobs among the batch. Once the process begins, it cannot be interrupted, thus the completion time of jobs in the batch are the same. In batch-processing machine scheduling problem, two decisions of forming batches and sequencing batches are made to optimize one or more objectives under some certain constraints, in this way there is more difficult than unit-processing machine scheduling problem.

In the literature, most studies considered an objective function on a single BPM scheduling problem, such as the following: minimum makespan (Lee and Uzsoy [2]; Sung and Choug [3]; Uzsoy [4]; Husseinzadeh Kashan *et al.* [5]; Dupont and Dhaenens-Flipo [6]; Melouk *et al.* [7]; Damodaran *et al.* [8]; Chou *et al.* [9]), minimum total flow time or mean flow time (Chandru *et al.* [10]; Dupont and Ghazvini [11]; Chang and Wang, [12]), minimum related due date criterion (Wang and Uzsoy [13]; Mehta and Uzsoy [14]; Perez *et al.* [15]; Chou and Wang [16]; Wang [17]). There has been little reported research on single BPM problem with more than one objective. To our best knowledge, Kashan *et al.* [18] did the first research on minimizing makespan and maximum tardiness simultaneously for a single BPM problem with non-identical job sizes. They proposed two multi-objective genetic algorithms based on different representation schemes. However, they did not consider the release time of jobs in their study. In the real world, the jobs has different arrival time for being processing, therefore, in this paper, we extend this static single-machine batching problem to the dynamic problem where jobs have different release times. Moreover, we consider simultaneously two objectives of makespan and total weighted tardiness for the problem, because the former represents the utilization of machines, and the latter measures the customers' satisfaction, each of both is very important for increasing a company's competitiveness.

As the first systematic attack to this problem, we construct an exhausted enumeration approach to find all Pareto-optimal solutions for small problems. Moreover, the problem is a well-known NP-hard problem because the problem with one of two objectives is also a NP-hard; therefore, we also developed a SA-based multi-objective (SAMO) algorithm and examine its effectiveness and efficiency by comparing the obtained solutions with those obtained from the exhausted enumeration approach. The experimental results showed that the proposed SAMO algorithm is successful to search Pareto-optimal solutions for small problems.

## II. Problem Definition

According to the three-field notation proposed by [19], the considered problem can be defined as $1|r_j,s_j,w_j,B|C_{\max},\sum w_jT_j$ , and the assumptions and notation for this problem are described as follow:

1) The processing time $p_j$ , release time $r_j$ , size $s_j$ , weights $w_j$ , and due date $d_j$ of each job are known and fixed.
2) The BPM is available at the beginning, i.e., $t=0$ .
3) The capacity ( $B$ ) of BPM is known.
4) All jobs can be grouped as a batch and be processed on a BPM simultaneously as long as the total sizes of jobs in the batch does not exceed the capacity of the BPM, i.e.,

Manuscript received May 1, 2013; revised July 1, 2013. This paper was supported in part by the National Science Council, Taiwan, ROC, under the contract NSC 101-2410-H-238-001.

Hui-Mei Wang is with the Department of Industrial Management, Vanung University, Jung-Li, Tao-Yuan, Taiwan (e-mail: amywang@mail.vnu.edu.tw).

Fuh-Der Chou is with the Department of Industrial Management, Chien Hsin University, Jung-Li, Tao-Yuan, Taiwan (tel.:+886-3-4581196-6103; fax: +886-3-2503908; e-mail: fdchou@uch.edu.tw ).

DOI: 10.7763/IJIET.2013.V3.331

$\sum s_j \le B$.

5) The batch processing time ($bp_k$) for a BPM is represented as the longest processing time among the jobs in a batch, i.e., $bp_k = \max(p_j) \quad job\ j \subseteq batch\ k$.
6) No machine breakdowns and preemptions.
7) Two objectives of makespan ($C_{\max}$) and total weighted tardiness ($\sum w_j T_j$) are considered simultaneously.

## III. Pareto-Optimal Set

This paper attempts to find Pareto-optimal set under the bi-criterion ($C_{\max}$ and $\sum w_j T_j$) for the single BPM problem. According to the definition proposed by Tamaki *et al.* [20], the set $\pi$ of Pareto-optimal solutions to a minimization problem with $n$ objective functions should satisfy the following conditions:

1) Two feasible solutions $\pi_1$, $\pi_2$, $\pi_1$ is dominated by $\pi_2$ if

$$f_i(\pi_1) \ge f_i(\pi_2),\ \forall i = 1,2,...,n$$

and

$$f_i(\pi_1) > f_i(\pi_2),\ \exists i = 1,2,...,n$$

2) If $\pi_0 \in \pi$, there is no other feasible solution $\pi^* \in \pi$ such that $\pi_0$ is dominated by $\pi^*$.

The well-known approach used to search Pareto-optimal solutions in the multi-objective scheduling problems is weight weighting method which search Pareto-optimal solutions in different directions by altering slightly weighted values. Chang *et al.* [21] combined the weight weighting method with a GA to solve multi-objective flowshop scheduling problem. Due to its successful for solving multi-objective flowshop scheduling problem, we also combine the weight weighting method with SA to develop a SAMO algorithm for the bicriteria BPM problem.

## IV. Multi-Objective Simulated Annealing

Meta-heuristic algorithms such as GA, SA, ant colony optimization (ACO), particle swarm optimization (PSO) have been successful to solve combinatorial problem. Recently, some researchers have applied SA, called MOSA, to solve multi-objective scheduling problems because of its simplicity and capability of finding Pareto solutions in a single run with very short computation time compared to other meta-heuristic algorithms.

The proposed SAMO is similar to single-objective SA, both start with initial solutions, and neighbor solutions are generated based on some mechanism. To avoid trapping to local optimum, SA use accepting mechanism to deal with non-improving neighboring solutions. In single-objective SA, the general accepting mechanism is Boltzmann probability as follows:

$$probability(P) = \exp\left(\frac{-\Delta E}{K_B T}\right)$$

where $\Delta E$ is the change in the objective value from one point to the next, and $K_B$ the boltzmann's constant and $T$ the temperature (control parameter). For MOSA, the calculation of $\Delta E$ has to deal with two or more conflicting objective values, and some different accepting mechanism are proposed by different MOSA, such as SMOSA [22], UMOSA [23], WMOSA [24], and PSA [25]. In this paper, we use the concept of UMOSA to integrate two objective values into a fitness value as follows:

$$probability(P) = \exp\left(\frac{-\Delta E}{T_x}\right)$$

where $\Delta E = z^{new} - z_x^c$ and

$$z^{new} = \alpha \times z_1^{new} + \beta \times z_2^{new} = \alpha f_1(\pi^{new}) + \beta f_2(\pi^{new})$$

In addition, we separate the search space into 11 directions shown in Fig. 1. Thus, at the beginning of SA, we have 11 initial solutions. The detail of SA is described in the following.

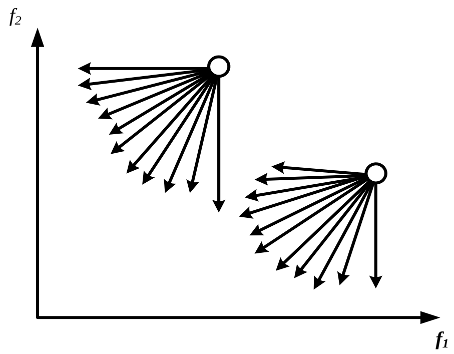


Fig. 1. The search directions.

The pseudo code of SAMO

$Z_x^i = \infty$, $z_{1,x}^i = \infty$, $z_{2,x}^i = \infty$, $\pi_x^i = \{\emptyset\}$, $x$=0, 1, 2,…, 10

$k$=1, and Pareto-optimal set={∅}
Do while ($k \le 10$)
Generate a job sequence, $\pi^{new}$ and maintain the Pareto-optimal set.
Calculate two objective values, $z_1^{new}$ and $z_2^{new}$, for $\pi^{new}$

Let $x$=0
Do while ($x \le 10$)
$\alpha = 0.0 + 0.1x$ , $\beta = 1 - \alpha$ , $Z^{new} = \alpha z_1^{new} + \beta z_2^{new}$
IF ($Z^{new} < Z_x^i$) then
$Z_x^i = Z^{new}$ , $z_{1,x}^i = z_1^{new}$ , $z_{2,x}^i = z_2^{new}$ , $\pi_x^i = \pi^{new}$
End IF
$x$=$x$+1
End Do
$k$=$k$+1
End Do
$Z_x^b = Z_x^c = Z_x^i$ , $z_{1,x}^b = z_{1,x}^c = z_{1,x}^i$ , $z_{2,x}^b = z_{2,x}^c = z_{2,x}^i$ , $\pi_x^b = \pi_x^c = \pi_x^i$
Initial Temperate $T_x = 1.618 \times Z_x^i$, $x$=0,1,2,…,10, cooling $rate\ \delta = 0.98$
Stop_flag=0

Do while (Stop_flag =0)
  $x$=0
  Do while ($x \leq 10$ and Stop_flag $= 0$)
    Generate a neighbor sequence $\pi^{new}$ based on $\pi_x^c$
      Maintain Pareto-optimal set by Comparing $\pi^{new}$ with each of Pareto-optimal set
      $y$=0
      Do while ($y \leq 10$)
        $\alpha = 0.0 + 0.1y$ , $\beta = 1 - \alpha$ , $Z^{new} = \alpha z_1^{new} + \beta z_2^{new}$
        IF ($Z^{new} < Z_y^b$) then
          $Z_y^b = Z^{new}$ , $z_{1,y}^b = z_1^{new}$ , $z_{2,y}^b = z_2^{new}$ , $\pi_y^b = \pi^{new}$
        End IF
        $y$=$y$+1
      End Do
    $\alpha = 0.0 + 0.1x$ , $\beta = 1 - \alpha$ , $Z^{new} = \alpha z_1^{new} + \beta z_2^{new}$
    IF ($Z^{new} < Z_x^b$) then
      $Z_x^c = Z^{new}$ , $z_{1,x}^c = z_1^{new}$ , $z_{2,x}^c = z_2^{new}$ , $\pi_x^c = \pi^{new}$
      $Z_x^b = Z^{new}$ , $z_{1,x}^b = z_1^{new}$ , $z_{2,x}^b = z_2^{new}$ , $\pi_x^b = \pi^{new}$
    Else
      IF ($Z^{new} < Z_x^c$) then
        $Z_x^c = Z^{new}$ , $z_{1,x}^c = z_1^{new}$ , $z_{2,x}^c = z_2^{new}$ , $\pi_x^c = \pi^{new}$
      Else
        Generate a random γ, $p = \exp(\frac{Z_x^c - Z^{new}}{T_x})$
        IF ($\gamma \leq p$)
          $Z_x^c = Z^{new}$ , $z_{1,x}^c = z_1^{new}$ , $z_{2,x}^c = z_2^{new}$ , $\pi_x^c = \pi^{new}$
        End IF
      End IF
    End IF
    $T_x = \delta T_x$, $x$=$x$+1
    IF (execute time ≥ time limit)
      Stop_flag=1
    End IF
  End Do
End Do

## V. Computational Experiments

For testing effectiveness of the proposed SAMO, we compare the approximate Pareto-optimal solutions obtained by the SAMO with those obtained by the exhausted enumeration approach. Because it is difficult for the exhausted enumeration approach to solve large problems, thus in this paper small-scale test problem instances are generated. All the algorithms are implemented in C++, running on a LINUX (Ubuntu 10.10.1) with AMD Phenom 9559 2.2 GHz processor (4 GB RAM). In this following the parameters for producing test instances and the performance measures are described.

### A. Parameters for Producing Test Instances

The test problems are generated randomly; each of test problems has 10 instances with 5, 7, 9, 11, and 13 jobs. Therefore, problem instance is fifty totally. For each job *j*, a release time $r_j$, a processing time $p_j$, job sizes $s_j$ and a weight $w_j$ were generated from the discrete uniform distribution [0, 48], [8, 48], [1, 30], and [1, 11] respectively. The due date $d_j$ is also generated from a discrete uniform distribution in the range $r_j + p_j + \left[\mu\left(1-\frac{R}{2}\right), \mu\left(1+\frac{R}{2}\right)\right]$ where $\mu$ is equal to $(1-T)C_{\max}^*$ , $C_{\max}^*$ is an estimation value for the completion time to finish all jobs. According to the study (Lee and Uzsoy [2]), the estimation value of $C_{\max}^*$ is calculated by the FBLPT algorithm and minimum release times among the jobs, that is, $C_{\max}^* = \min(r_j) + C_{\max}(FBLPT)$ . Additionally, $T$ and $R$ values are dedicated to 0.6 and 0.5 which represents the due date tightness factor, and the spread range of due dates. The machine capacity is 40 for all test problems.

### B. Performance Measures

There are two general types of performance measure are used to compare two algorithms for multi-objective scheduling problems: quantitative and qualitative measures [26].

#### 1) Quantitative index

$|N_A|$ and $|N_B|$ indicate that the number of Pareto-optimal solution obtained by the exhausted enumeration approach and SAMO algorithm, respectively.

#### 2) Qualitative index

$|N_A|/|N_C|$ and $|N_B|/|N_C|$ are used to obtain qualitative measures for the exhausted enumeration approach and SAMO algorithm, respectively, where $|N_c|$ is the number of Pareto-optimal solutions obtained from the two algorithms.

TABLE I: The Experimental Results

| Jobs | Quantitative | | Qualitative | |
|---|---|---|---|---|
| | *A* | *B* | *A* | *B* |
| 5 | 1.5 | 1.5 | 1 | 1 |
| 7 | 1.9 | 1.9 | 1 | 1 |
| 9 | 2.1 | 2.1 | 1 | 1 |
| 11 | 3.1 | 3.1 | 1 | 0.975 |
| 13 | 3.1 | 3.1 | 1 | 0.960 |
| Global ave. | 2.34 | 2.34 | 1 | 0.987 |

TABLE II: The Execution Time Taken by the Exhausted Enumeration Approach

| Jobs | 5 | 7 | 9 | 11 | 13 |
|---|---|---|---|---|---|
| Time (s) | 0.00 | 0.01 | 0.65 | 82.34 | 34361.71 |

In this paper $|N_A| = |N_c|$ because the exhausted enumeration approach can find all true Pareto-optimal solutions. Table I shows that the average quantitative and qualitative indices. From the result of Table I, it is evident that the proposed SAMO algorithm could almost find all the Pareto-optimal solutions except for two problems with 11 and 13 jobs. On the whole, the average solution quality obtained by the proposed SAMO algorithm is from 100% to 96% from 5-job to 13-job problems, although the solution quality of SAMO algorithm seems worse when the problem-scale increases. This reason is that the given time limit in the SAMO algorithm is short (five seconds) such that the SAMO algorithm did not converge. Based on the experimental results, we could conclude that the search strategy by means of ten-directions is effective for the proposed SA. Moreover, the outstanding results lead us to apply the SAMO algorithm to solve the large-scale problems for the practical use.

Table II show the execution time taken by the exhausted enumeration approach, it is evident that the execution time dramatically increases when the problem scale increase. It leads us to apply the SAMO algorithm to solve large problems efficiently.

## VI. Conclusions

In this paper, an exhausted enumeration approach for the bicriteria single BPM problem with nonidentical job sizes and release times is presented, which motivated by semiconductor manufacturing burn-in operations, wherein jobs that belong to the different families can be processed simultaneously. The considered problem is first discussed so far. Experimental results shows that exhausted enumeration approach could find all true Pareto-optimal solution for the sets of 5 to 13 jobs. However, it takes an unreasonable amount of computation time when the jobs is thirteen, which suggest the need for heuristic development to solve more practical bicriteria BPM scheduling problems.

A proposed SAMO algorithm was developed that solve the all test problems in five seconds, a stop condition in the proposed SAMO algorithm. The SAMO algorithm works efficiently to produce an approximate Pareto-optimal set where 98.7% of the solutions in the set were found to be true Pareto-optimal solutions. The outstanding performances on small-scale problems suggest the readiness of the proposed SA for large-scale problems.

The research may be extended in the direction of examining the proposed SAMO algorithm on the general case of the problem, and developing other meta-heuristic algorithm such as GA is also another research direction.

## References


[1] Y. Ikura and M. Gimple, "Efficient scheduling algorithms for a single batch processing machine," *Operations Research Letters*, vol. 5, pp. 61-65, 1986.

[2] C.-Y. Lee and R. Uzsoy, "Minimizing makespan on a single batch processing machine with dynamic job arrivals," *International Journal of Production Research*, vol. 37, pp. 219-236, 1999.

[3] C. Sung and Y. Choung, "Minimizing makespan on a single burn-in oven in semiconductor manufacturing," *European Journal of Operational Research*, vol. 120, pp. 559-574, 2000.

[4] R. Uzsoy, "Scheduling a single batch processing machine with nonidentical job sizes," *International Journal of Production Research*, vol. 32, pp. 1615-1635, 1994.

[5] A. H. Kashan, B. Karimi, and F. Jolai, "Effective hybrid genetic algorithm for minimizing makespan on a single-batch-processing machine with non-identical job size," *International Journal of Production Research*, vol. 44, pp. 2337-2360, 2006.

[6] L. DuPont and C. Dhaenens-Flipo, "Minimizing the makespan on a batch machine with non-identical job sizes: an exact procedure," *Computers & Operations Research*, vol. 29, pp. 807-819, 2002.

[7] S. Melouk, P. Damodaran, and P. Y. Chang, "Minimizing makespan for single machine batch processing with non-identical job sizes using simulated annealing," *International Journal of Production Economics*, vol. 87, pp. 141-147, 2004.

[8] P. Damodaran, P. K. Manjeshwar, and K. Srihari, "Minimizing makespan on a batch-processing machine with non-identical job sizes using genetic algorithms," *International Journal of Production Economics*, vol. 103, pp. 882-891, 2006.

[9] F. D. Chou, P. C. Chang, and H. M. Wang, "A hybrid genetic algorithm to minimize makespan for the single batch machine dynamic scheduling problem," *International Journal of Advanced Manufacturing Technology*, vol. 31, pp. 350-359, 2006.

[10] V. Chandru, C. Y. Lee, and R. Uzsoy, "Minimizing total completion time on batch processing machines," *International Journal of Production Research*, vol. 31, pp. 2097-2121, 1993.

[11] F. J. Ghazvini and L. Dupont, "Minimizing mean flow times criteria on a single batch processing machine with non-identical jobs sizes," *International Journal of Production Economics*, vol. 55, pp. 273-280, 1998.

[12] P. C. Chang and H. M. Wang, "A heuristic for a batch processing machine scheduled to minimise completion time with non-identical job sizes," *International Journal of Advanced Manufacturing Technology*, vol. 24, pp. 615-620, 2004.

[13] C. S. Wang and R. Uzsoy, "A genetic algorithm to minimize maximum lateness on a batch processing machine," *Computers & Operations Research*, vol. 29, pp. 1621-1640, 2002.

[14] S. V. Mehta and R. Uzsoy, "Minimizing total tardiness on a batch processing machine with incompatible job families," *IIE Transactions*, vol. 30, pp. 165-178, 1998.

[15] I. C. Perez, J. W. Fowler, and M. Carlyle, "Minimizing total weighted tardiness on a single batch process machine with incompatible job families," *Computers and Operations Research*, vol. 32, pp. 327-341, 2005.

[16] F. D. Chou and H. M. Wang, "Scheduling for a single semiconductor batch-processing machine to minimize total weighted tardiness," *Journal of the Chinese Institute of Industrial Engineers*, vol. 25, pp. 136-147, 2008.

[17] H. M. Wang, "Solving single batch-processing machine problems using an iterated heuristic," *International Journal of Production Research*, vol. 49, pp. 4245-4261, 2011.

[18] A. H. Kashan, B. Karimi, and F. Jolai, "An effective hybrid multi-objective genetic algorithm for bi-criteria scheduling on a sinlge batch processing machine with non-identical job sizes," *Engineering Applications of Artificial Intelligence*, vol. 23, pp. 911-922, 2010.

[19] J. U. Duncombe, "Infrared navigation—Part I: An assessment of feasibility," *IEEE Trans. Electron Devices*, vol. ED-11, pp. 34-39, Jan. 1959.

[20] H. Tamaki, H. Kita, and S. Kobayashi, "Multi-objective optimization by genetic algorithms: A review," in *Proc. the IEEE International Conference on Evolutionary Computation*, pp. 517-522, 1996.

[21] P.-C. Chang, J.-C. Hsieh, and S.-G. Lin, "The development of gradual-priority weighting approach for the multi-objective flowshop scheduling problem," *International Journal of Production Economics*, vol. 79, pp. 171-183, 2002.

[22] A. Suppapitnarm, K. A. Seffen, G. T. Parks, and P. J. Clarkson, "Simulated annealing: An alternative approach to true multiobjective optimization," *Engineering Optimization*, vol. 33, pp. 59, 2000.

[23] L. E. Ulungu, J. Teghem, and C. Ost, "Interactive simulated annealing in a multiobjective framework: application to an industrial problem," *Journal of Operational Research Society*, vol. 49, pp. 1044, 1998.

[24] B. Suman, "Multiobjective simulated annealing-A metaheuristic technique for multiobjective optimization of a constrained problem," *Foundations of Computing and Decision Sciences*, vol. 27, pp. 171, 2002.

[25] P. Czyzak and A. Jaszkiewicz, "Pareto simulated annealing-A metaheuristic technique for multiple-objective combinatorial optimization," *Journal of Multicriteria Decision Analysis*, vol. 7, pp. 34-, 1998.

[26] J. K. Cochran, S. M. Horng, and J. W. Fowler, "A multi-population genetic algorithm to solve multi-objective scheduling problems for parallel machines," *Computers & Operations Research*, vol. 30, pp. 1087-1102, 2003.


**Fuh-Der Chou** is an associate professor in the Department of Industrial Management at Chien Hsin University, Taiwan. He received his M.S. degree in industrial engineering from Chung Yuan Christian University in 1988, and his Ph.D. degree in industrial engineering and management from National Chiao Tung University in 1997. His research interests include production scheduling, semiconductor manufacturing management, and group technology.

**Hui-Mei Wang** is an associate professor in the Department of Industrial Management at Vanung University. She received her M.S. and Ph. D. degrees in Department of Industrial Engineering and Management from the University of Yuan-Ze. Her present research interests include production scheduling and semiconductor manufacturing management.