Balancing setup workers' load of flexible job shop scheduling using hybrid genetic algorithm with tabu search strategy

Yasuhiko Morinaga and Masahiro Nagao

Graduate School of Environmental Studies,
Nagoya University,
Furo-cho, Chikusa-ku, Nagoya, Aichi 464-4601, Japan
Email: yasu.morinaga@nifty.com
Email: ngo@nagoya-u.jp

Mitsuru Sano*

Chubu University,
1200 Matsumoto-cho, Kasugai, Aichi 487-8501, Japan
Email: sano@isc.chubu.ac.jp

Abstract: This paper describes optimisation of a multi-objective flexible job shop scheduling problem (MO-FJSP) in small and medium-sized enterprises (SMEs) where widely various products are manufactured in make-to-order (MTO) mode. A genetic algorithm using tabu search strategy was applied to solve the MO-FJSP incorporating weighted tardiness, setup worker load balance, and work-in-process. From experiments using data based on real-world SME, the solutions obtained using the proposed method are compared with those obtained by conventional earliest due date (EDD), and GA using multi-island. The results confirmed the effectiveness of the proposed method. Results imply that the proposed approach is applicable not only for production scheduling but also for estimating the investment of resources such as machine and worker capacity.

Keywords: flexible job shop scheduling problem; setup workers' workload balance; work-in-process inventory; tabu search; genetic algorithm.

Reference to this paper should be made as follows: Morinaga, Y., Nagao, M. and Sano, M. (2016) 'Balancing setup workers' load of flexible job shop scheduling using hybrid genetic algorithm with tabu search strategy', *Int. J. Decision Support Systems*, Vol. 2, Nos. 1/2/3, pp.71-90.

Biographical notes: Yasuhiko Morinaga received his Bachelor of Science in Mechanical Engineering from Kyushu University in 1968 (Japan) and his Master's degree in Environment Studies from Nagoya University in 2014. He is currently a PhD candidate in Environmental Studies at Nagoya University. His research particularly addresses sustainability of systems such as small and medium-sized enterprises or local communities. He also has experience in production engineering in MTO. He is interested in high-mix low-volume production.

Masahiro Nagao received his PhD in Engineering from Nagoya University, Japan, in 2011. He is currently an Assistant Professor at Nagoya University. His research interests include application of machine learning; scheduling and optimisation problem related to environmental problems.

Mitsuru Sano received his PhD in Inorganic Chemistry from Nagoya University in 1981. He is currently a Professor Supporting President of Chubu University. His research particularly addresses environmental impact, environmental policy, social systems engineering, energetics, functional materials, and devices.

1 Introduction

Small and medium-sized enterprises (SMEs) play important roles in sustaining local communities and economies in many countries (Ayyagari et al., 2007; Giovanna et al., 2012). However, it is difficult for them to survive competition with rivals during changes in trade structure or shortening of a product life cycle (Jamali et al., 2015). To sustain them, it is necessary for SMEs to establish their own systems for response to the demand of customers quickly and flexibly from a marketing point of view. For the manufacturing sector, an appropriate option is adoption of a flexible manufacturing system that can accommodate various demands of customers (changes in the amount and types of goods and services demanded) (Hendry, 1998; Veen-Dirks, 2005; Simao et al., 2006). The flexible manufacturing system consists of numerous flexible factors that cause difficulty in managing the system. Running the system efficiently requires information systems, especially scheduling systems for flexible job shops to match their various resources to customer demands (Proth, 2007; Wei and Ma, 2014).

Production scheduling in make-to-order (MTO) manufacturing has several important points: meeting the due-date of a customer (Wang et al., 2011; Kaminsky, 2008; Sawik, 2011), balancing the workload of resources that are constrained (Thuerer et al., 2012), and reducing work-in-process for proper cash flow management (Nenes et al., 2014). These points dictate that enterprises with flexible job shop manufacturing systems for MTO production need multi-objective flexible job shop scheduling systems.

Studies in this field examine the multi-objective flexible job shop scheduling problem (MO-FJSP). This is a combination problem for optimising objective functions related to a combination of given jobs or tasks, operations of each job to be processed in the specified order, and machines used to carry out operations.

Given the background presented above, studies in the field of MO-FJSP have been conducted for several decades. At the beginning of this century, several researchers have started using evolutionary algorithms for this field. Kacem et al. (2002a, 2002b) used a method combining fuzzy logic and evolutionary algorithms. They studied the optimisation of three objectives: make span, total load on the machinery, and load of the machine with the largest load on the so-called dataset of Kacem et al. The dataset includes five instances having scale ($n \times m$, where n stands for the number of jobs, and m denotes the number of machines) ranges from 4×5 to 15×10. After these studies, various evolutionary algorithms have been developed for solving MO-FJSP on the Kacem et al. dataset. The performance results of these algorithms have been mutually compared: hybrid approach of particle swarm optimisation (PSO) and simulated annealing (SA) (Xia and Wu, 2005); hybrid approach of PSO and tabu search (TS) (Zhang et al., 2009); improvement of operation sequencing subsystem by ant colony optimisation algorithm (ACO) (Xing et al., 2009); a TS algorithm with neighbourhood search (Li et al., 2010); an approach based on a hybridisation of the PSO and local search algorithm (Moslehi and Mahnam, 2011); hybrid Pareto-based discrete artificial bee colony (ABC) algorithm with several local search approach (Li et al., 2011); hybrid genetic algorithm (GA) and SA (Shahasaveri-Pour and Ghasemishabankareh, 2013); and the discrete ABC algorithm with TS (Li et al., 2014). Nevertheless, few studies have considered due dates for MTO manufacturing set by important customers (Wu and Weng, 2005; Turkyilmaz and Bulkan, 2015). Actually, several researchers have studied FJSP depending on the setup work (Defersha and Chen, 2010; Huang et al., 2013), but studies of systems that balance the setup worker load in MO-FJSP are few (Nagao et al., 2015). Although a few studies of work-in-process in a flow shop can be found in the web of science (Luh et al., 2000), no studies of work-in-process in a flexible job shop have been reported. Additionally, almost all earlier studies of MO-FJSP emphasise a discussion of their algorithm performance on the Kacem et al. dataset or BD dataset (Brandimarte, 1993), but no discussion of the algorithm effectiveness has been reported for data referred from a real-world manufacturing shop.

This report explains three optimisation objectives: weighted tardiness, setup worker load balance, and work-in-process inventory on a shop floor in MTO mode of actual SMEs. The due dates are weighted by trade conditions such as the position of each job in the customer's supply chain (e.g., an assembly plant directly related to production, trading company having their own inventory, and restocking at the company of their own products based on demand forecasts). This study also examines balancing the load of setup workers with different technical levels because, if due dates are overemphasised, workers with low levels of technical skill will feel excessively burdened, which might increase the risk of missing deadlines. Different kinds of products with various numbers are handled in MTO manufacturing. Then the work-in-process inventory on a shop floor is preferably evaluated in terms of production cost based on the unit price and lot size of an individual order.

As the optimisation method, we have developed an algorithm based on GA which advances in global search. However, GA presents shortcomings in local search. We have developed a hybridisation algorithm that combines a TS strategy with GA to compensate for these shortcomings of GA. Its validity was verified in numerical experiments using data related to jobs in shops of actual high-mix, various-volume MTO mode of SMEs.

The remainder of the paper consists of the following. Section 2 defines the MO-FJSP in MTO manufacturing. Section 3 presents a description of the structure of the GA used for this study. Section 4 explains the conditions and data used for the experiments and parameter settings in GA, and Section 5 presents the results of experiments. Section 6 discusses the results. Section 7 presents a summary of the entire paper.

2 MO-FJSP in MTO manufacturing

2.1 Problem description

This paper is targeted at MTO manufacturing, which handles products with high-mix and various-volumes of diverse customers. As shown in Table 1, not only the relations between operations of jobs and machines, which are described in almost all earlier research, but also the relations among due-date, tardiness weight, lot size, and unit price of each order (job) are considered to optimise weighted tardiness, balance of setup workers' workload and work-in-process on a shop floor. Additionally, as Table 2 shows, the relations between setup workers and machines are considered. These are generalised as described below.

1 i^{th} operation O_{ki} of jobs $J_k(J_1, J_2,\ldots,J_k)$ with lot size L_k for each job are processed using m units of machinery $M_m(M_1, M_2,\ldots,M_m)$.

2 The order of operations of each job should be maintained, but the order of passing machines varies among operations.

3 Job J_k is completed, having passed all of O_{ki}.

4 Each operation O_{ki} is linked to machine M_m, which is capable of processing. Each machine M_m is linked to setup-worker W_w who can set up machine M_m.

5 The operation is completed when the processing time $P_{kim}(P_{kim} = p_{kim} \times L_k)$ for each job consisting of the product of processing time p_{kim} per operation and L_k has passed without interruption.

6 If the type of job differs from the previous job when work is changed from one job to the next for machine M_m, then any of w number of setup-workers $W_w(W_1, W_2,\ldots, W_w)$ will set up the machine. Therefore, the setup time S_{kim} arises before processing.

The constraints follow from our earlier paper (Morinaga et al., 2014).

Table 1 Relations among jobs, operations and machines

Job	*Due date*	*Tardiness weight*	*Number of operation*	*Lot size*	*Unit price*	*Operation*	*Machine*			
							M_1	M_2	…	M_m
J_1	d_1	a_1	n_1	L_1	u_1	O_{11}	p_{111}	p_{112}	…	p_{11m}
						O_{12}	p_{121}	p_{122}	…	p_{12m}
						O_{13}	p_{131}	p_{132}	…	p_{13m}
J_2	d_2	a_2	n_2	L_2	u_2	O_{21}	p_{211}	p_{212}	…	p_{21m}
						O_{22}	p_{221}	p_{222}	…	p_{22m}
:	:	:	:	:	:	:	:	:	…	:
J_k	d_k	a_k	n_k	L_k	u_k	O_k	p_{k11}	p_{k12}	…	p_{k1m}
						O_{k2}	p_{k21}	p_{k22}	…	p_{k2m}
						:	:	:	…	:

Table 2 Relations between setup workers and machines

Setup worker	*Machine*			
	M_1	M_2	…	M_m
W_1	S_{k11}	S_{k12}	…	S_{k1m}
W_2	S_{k21}	S_{k22}	…	S_{k2m}
:	:	:	:	:
W_w	S_{kw1}	S_{kw2}	…	S_{kwm}

2.2 Three objectives to minimise

2.2.1 Weighted tardiness

As shown in Table 3, each job has a due date d_k. The difference from the period e_k for the completion of the final operation of job J_k is called the tardiness period $td_k(\max\{0, e_k - d_k\})$. Therefore, weighted tardiness TD is defined as shown below.

$$\min TD = \sum_{k=1}^{k} \alpha_k td_k \tag{1}$$

Therein, α_k is the weighting factor for the tardiness of job J_k.

Table 3 Weighted tardiness of each job

Job	*Due date*	*Shipment date*	*Tardiness*	*Weighting factor*	*Weighted tardiness*
J_1	d_1	e_1	$td_1 = \max\{0, e_1 - d_1\}$	α_1	$\alpha_1 \times td_1$
J_2	d_2	e_2	$td_2 = max\{0, e_2 - d_2\}$	α_2	$\alpha_2 \times td_2$
:	:	:	:	:	:
J_k	d_k	e_k	$td_k = \max\{0, e_k - d_k\}$	α_k	$\alpha_k \times td_k$

2.2.2 Balancing the setup workers load

In MTO, various products are released to the shop floor every day. Then, each job is shipped according to due date d_k, which causes the jobs on the shop floor to be switched every day. That is, jobs kept on the shop floor on the first day, when target jobs are scheduled, are reduced every day and new jobs are added from the next day on. Therefore, to consider balancing the setup worker load, allocation of setting up is expected to be reflected by changes in the job load (Morinaga et al., 2014).

Because the number of jobs (k_0) on the first day, when target jobs are scheduled, will decrease to k_τ on the τ^{th} day, as above described, if the maximum permissible value for providing setups to the jobs on the first day is L_0, then the maximum permissible value for providing setups to the jobs on the τ^{th} day (L_τ) will be $(k_\tau/k_0) \times L_0$. If $k_\tau/k_0 = \beta_\tau(\beta_0 = 1)$, then $L_\tau/\beta_\tau = L_0$. Assuming that this L_0 is the maximum permissible value for providing setups and that SL is the equivalent value for the setup workers load, then SL is set to equal L_0. Balancing the setup load is considered under the condition of $L_\tau/\beta_\tau \leq SL$. Therefore, setup worker load SL is defined as shown below.

$$\min SL = \max\left\{\lceil L_{\tau\_\max} / \beta_t \rceil \tau = 1, 2, \ldots, T\right\} \tag{2}$$

$$L_{\tau\_\max} = \max\left\{L_{\tau w} \,\middle|\, w = 1, 2, \ldots, w\right\} \tag{3}$$

Therein, L_{τ_max} is maximum setup workers load in the evaluation period from $\tau = 1$ to $\tau = T$.

2.2.3 Work-in-process on the shop floor

Table 4 shows the unit price and the number of works-in-process on the shop floor (x_{it}) of each job at the time t. Assuming that the number of released jobs is equal to the number of orders and assuming that there are no losses on the shop floor caused by processing defects, then $x_{it} = 0(t < t_i, t > T_i)$ and $x_{it} = L_i(t_i \leq t \leq T_i)$. In those equations, t_i is the release time of job J_i, and T_i is the shipment time of job J_i. Then, work-in-process on the shop floor of all target jobs in the work-in-process evaluation period from t_i to T_i; noted *WIP* is defined as follows.

$$\min WIP = \sum_{t=t_k}^{T_k} \sum_{i=1}^{k} u_i x_{it} \tag{4}$$

Table 4 Change of work-in-process

Job	*Unit price*	*Date*					
		1	*2*	…	*t*	…	*T*
J_1	u_1	x_{11}	x_{12}	…	x_{1t}	…	x_{1T}
J_2	u_2	x_{21}	x_{22}	…	x_{2t}	…	x_{2T}
:	:	:	:	…	:	…	:
J_i	u_i	x_{i1}	x_{i2}	…	x_{it}	…	x_{iT}
:	:	:	:	…	:	…	:
J_k	u_k	x_{k1}	x_{k2}	…	x_{kt}	…	x_{kT}

3 GA with TS strategy

3.1 Gene representation considering setup worker load

Almost no earlier study has considered the setup worker load. This approach requires the gene representation that involves the relation between machines and setup workers who can make setup of the machines processing operations. A simple example is presented in Figure 1 to illustrate the relation between two jobs (J_1, J_2), in which two operations are processed, with two machines (M_1, M_2) and two setup workers (W_1, W_2). In this figure, three '1 h' show that each setup time is 1 hr, and '2 h' and '3 h' respectively denote processing times of 2 hr or 3 hr. As shown in Figure 1, assuming that O_{11} has an option; processed in M_1 and setup by W_1, each of O_{12}, O_{21} and O_{22} has two options: processed in M_2 and setup by W_1 or processed in M_2 and setup by W_2, the combination of operations consists of eight representation numbers of jobs, as shown in Table 5. Table 6 is derived by sorting the representation numbers of job based on the processing machines in Table 5. Table 6 presents randomised sequences of representation numbers of jobs in each machine. For example, the processing job sequence in M_1 is set to representation number (1) only. That in M_2 is set to (2) → (4) → (1) → (3) → (4) → (6) → (8).

Figure 1 An example of relation among operations, machines and setup workers

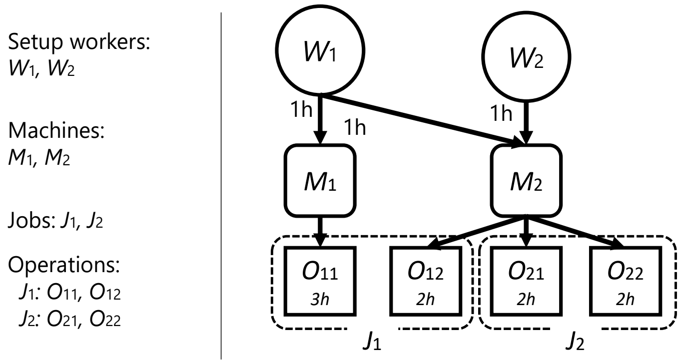


Table 5 An example of representation numbers

Representation number	*Job*	*First operation (O_{11} or O_{21})*	*Second operation (O_{12} or O_{22})*
(1)	J_1	M_1, W_1, 3 h	M_2, W_1, 2 h
(2)	J_2	M_2, W_1, 2 h	M_2, W_1, 2 h
(3)	J_1	-	M_2, W_2, 2 h
(4)	J_2	M_2, W_1, 2 h	M_2, W_2, 2 h
(5)	J_1	-	-
(6)	J_2	M_2, W_2, 2 h	M_2, W_2, 2 h
(7)	J_1	-	-
(8)	J_2	M_2, W_2, 2 h	M_2, W_2, 2 h

Table 6 An example of gene representations refereed in Table 5

Gene position	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*
M_1	(1)	0	0	0	0	0	0	0
M_2	(6)	(4)	(1)	(3)	(4)	(2)	(8)	0

Table 6 is the genotype for GA operation. A phenotype is required to evaluate three objectives. As shown in Table 7, a Gantt chart is derived by reading the representation numbers of a job in Table 6 from left in each machine. In the Gantt chart, each time frame is filled with a number of setup workers, several jobs to be processed, or a blank in each row of the machine. For example, in Table 6, the first locus in the row of machine M_1 is (1). Therefore, in Table 7, 'W_1', representing setup work for the first operation O_{11} of J_1, is set in the first time frame in the row of machine M_1. Furthermore, three 'J_1', standing for the first operation of J_1 are set in from the second time frame to the fourth. Furthermore, no operation is processed expect O_{11}; blanks continue from the fifth time frame. In the row of machine M_2, 'W_2', meaning setup work for the first operation of representation numbers (6) of job J_2, is set in the first time frame. Two 'J_2', representing the first operation of J_2, are set in from the second time frame to the third. The results presented above indicate that both the first operation of J_1 and J_2 are completed. In Table 7, the following vacant time frame is the fourth time frame in the row of machine M_2, in Table 6, the second locus in the row of machine M_2 is (4). Therefore, in Table 7,

'W_2', representing the setup work for the second operation of J_2, which is referred from Table 5, is set in the fourth time frame in the row of machine M_2, and two 'J_2' meaning for the second operation of J_2 are set in from the fifth time frame to the sixth in the row of machine M_2. Next, in Table 6, the third locus in the row of machine M_2 is (1). Therefore in Table 7, 'W_1', representing setup work for the second operation of J_1, which is referred from Table 5, is set in the seventh time frame in the row of machine M_2, and two 'J_1' standing for the second operation of J_1 are set in from the eighth time frame to the ninth in the row of machine M_2. After all operations are completed, from the Gantt chart, each release time to the shop, shipment time from the shop to customers and setup workers' workload of all jobs are derived. Then *TD*, *SL*, and *WIP* are calculated.

Table 7 An example of Gantt chart from Table 6

Time (h)	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*
M_1	W_1	J_1	J_1	J_1	0	0	0	0	
M_2	W_2	J_2	J_2	W_2	J_2	J_2	W_1	J_1	J_1

3.2 GA operation using tabu list

The proposed method is based on GA, which is widely used in the field of combination problem, using a TS (Glover, 1989) strategy in crossover and mutation operation. In this section, the outline of the GA algorithm in the proposed method, the basic idea of TS, and a hybrid GA with TS strategy (TS-HGA) are described.

3.2.1 Outline of GA

GA operations mimic natural selection and biological evolution by creating new offspring from a previous generation to solve problems. The procedure is the following.

1 An initial individual is created.

2 A population that consists of some number of individuals is created in an array randomly by shuffling the initial individual as presented in Figure 2. Each array consists of sub-arrays with fixed length divided by the number of registered machines (Machine 1, 2,…, *n*). Each sub-array indicates the order of representation number to be processed in each machine, as shown in Table 6.

3 A Gantt chart is derived based on the gene representation of each. Then weighted tardiness, setup worker load and work-in-process on the shop floor of each (solution), and Pareto solutions are derived from the population based on NSGA-II (Deb, 2002).

4 To make a valid comparison of the evaluated values, which have different orders of magnitude, the degree of congestion was proposed by normalising the evaluated values for the calculation of Euclidean distance as the congestion degree.

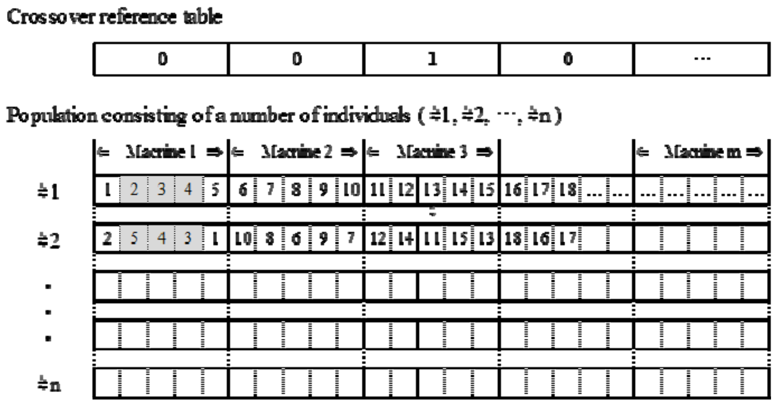
5 Evaluated values from Euclidean distance among points on Pareto surface are defined as the degree of congestion. The larger the evaluated values, the lower the degree of congestion.

6 Solutions are ranked in decreasing order of the value of their degree of congestion. They are selected as parents for creating offspring according to rank-based selection and roulette selection, which increase the diversity of solutions. Next, crossover and mutation operations are performed on the selected parents to generate offspring.

7 Crossover operations have two modes: exchanging whole arrays of the selected same number of machine between individuals, and exchanging the selected loci of the selected same number of machine between individuals. For instance, in the first mode, assumed machine 3 is selected randomly as presented in Figure 2, and gene array (11, 12, 13, 14, 15) in individual #1 and gene array (12, 14, 11, 15, 13) in individual #2 are mutually exchanged. By the second mode, assumed machine 1 and loci 2–4 in machine 1 are selected randomly as presented in Figure 2, gene array (2, 3, 4) in individual #1 and gene array (5, 4, 3) in individual #2 are mutually exchanged. However, in this case, both the array (1, 5, 4, 3, 5) in individual #1 and the array (2, 2, 3, 4, 1) in individual #2 after the crossover conflict with these arrays before the crossover. The conflicts should be amended based on the procedure.

8 Mutation operation is executed by exchanging random selected two genes in randomly selected machine arrays. Details of the procedure will be described later in the sub-section presenting hybrid GA with TS strategy (*TS-HGA*).

Figure 2 An example of population and crossing reference table



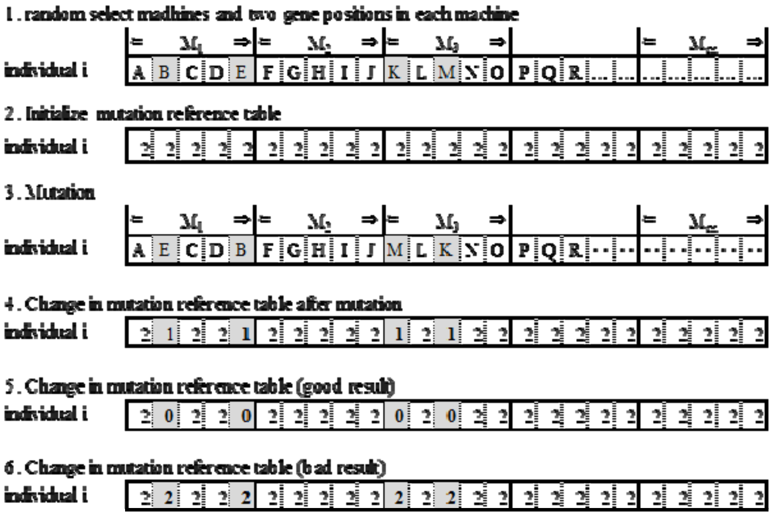
3.2.2 Outline of TS

To search for neighbouring optimal solutions in a short time, local search methods are used for application in this field, but using those methods, a neighbouring optimal solution tends to return to a current solution by repeating local searches. In TS, a tabu list that keeps a set of most recently visited solutions is created to avoid cycling and avoid being trapped in a local optimal solution. A basic procedure of TS is the following: generate an initial solution randomly and initialise a tabu list; create neighbouring solutions from the initial solution; and exchange the initial solution for the best solution involved in neighbourhood solutions which is not involved in solutions included on the tabu list. Then the tabu list, which is vacant by initialisation, is filled with a certain number of the last solutions encountered. These solutions are updated in each time when the solution is exchanged. When the terminating condition is obtained, the incumbent solutions derived to date are provided as the excellent solutions.

3.2.3 TS-HGA based on TS strategy

The crossover operation is executed by replacing the loci in the same number of machines based on crossover reference tables presented in Figure 2. The crossover reference tables are stored in the tabu list by a given condition. The set of crossover reference tables in tabu list is not accepted in the next iteration to avoid useless operation.

Figure 3 An Example of mutation procedure based on TS



Mutation is executed based not only on general GA operation but also on the mutation reference table as depicted in Figure 3. The procedure is the following.

1 As a target of mutation, the number of machines and gene loci on the machines are randomly selected.

2 A mutation reference table is defined in the same format as the format of chromosomes. All loci are set initially to 2.

3 A mutation operation based on general GA is executed. For example in Figure 3, machine 3 and machine 1 are the targets of mutation. Two loci selected randomly in each machine are mutually exchanged: gene B and gene E in machine 1, and gene K and gene M in machine 2.

4 In the mutation reference table, the two loci having the same position as the position in individual are reset to 1.

5 After the GA operation described above, a new Gantt chat is derived and the new individual is re-evaluated to ascertain if the new individual is superior to the prior individual. In the mutation reference table, the two positions with the value of ‘1’ are reset to ‘0’.

6 If the re-evaluated value is not superior to the former individual, in the mutation reference table, then the two positions with values of '1' are reset to '2'.

From the next mutation, the gene loci with the value of '0 'are eliminated from the target of mutation to avoid useless operations.

3.3 Outline of multi-island based GA (MI-GA)

Multi-island based GA (*MI-GA*) (Nagao et al., 2015), which can yield diverse solutions, is provided so that the performance of the proposed method is compared with that of *MI-GA*. The solutions obtained using the proposed methods are confirmed not to fall into local optimal solution. In *MI-GA*, the population in one generation is divided into several sub-populations called 'islands'. The genetic operations are executed independently on each island. For the islands, solutions are evaluated by rule of each island, which differ in weighting factor of crowded distance of each solution in the population; then this independence enables the solution to maintain diversity. Next the exchange of some individuals, named 'migration', is conducted periodically among islands. This mutual migration enables the solutions to avoid falling into a local optimum.

4 Experiment

This study addresses a combination including setup-worker, which is a novel approach in the field of MO-FJSP, as shown in Table 1 and Table 2. Therefore, the effectiveness of the proposed method cannot be compared with those found in prior studies. Two methods are introduced for comparison with the proposed method as follows: a conventional method EDD in which only the due-date of jobs has priority without consideration of the lot size of the jobs, the number of operations in each job and setup workers' workload balance; and *MI-GA* described above.

Sub-Section 4.1 describes the experimental data based on the daily operation of an SME of the MTO system. Sub-Section 4.2 describes parameter settings for the experiment to verify the performance of the proposed method.

4.1 Experimental data based on daily operation of SME of MTO

The SME manufactures high-mix and various-volume timing pulleys for equipment such as sewing machines, optical devices and medical instruments from materials such as metals or resin. The manufacturing process of pulleys consists of several processes such as turning, drilling, milling, gear cutting, assembling, and surface treatment.

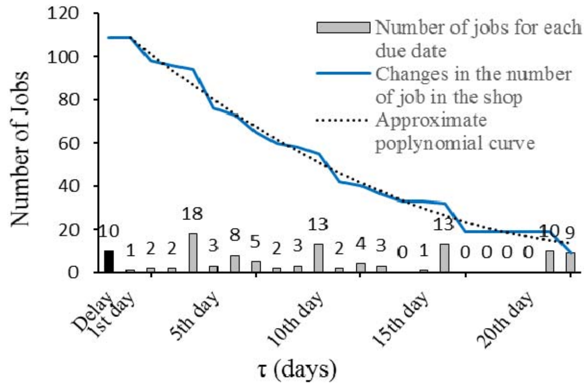
In this experiment, one can assess the following cases based on daily operations.

1 109 jobs which have 1–9 operations each are ordered at scheduling.

2 These jobs are processed by six setup workers and ten machine-processing workers using 32 machines.

3 As portrayed in Figure 4, the bar chart shows a number of jobs with respective numbers of days to the due date. The maximum number of days to the due date is 22. Actually, ten jobs are already overdue.

Figure 4 Changes in the number of jobs on the shop for 109 jobs (see online version for colours)



The number of jobs on the shop floor at the time of scheduling decreases because of daily shipments, as portrayed in Figure 4. Here, $\beta\tau$ is determined by the approximate polynomial derived from the solid line graph, which, in combination with equation (2), is used to derive *SL*.

Table 8 Estimated *TD* value based on EDD

Job	*Lot size*	*Days left for due date*	*Days to complete processing*	td_i	α_i	$\alpha_i\, td_k$
J_1	10	–6	1	7	100	700
J_4	2	6	7	1	100	100
J_5	100	6	8	2	100	200
J_{40}	100	–3	2	5	5	25
J_{41}	100	–2	3	5	5	25
J_{42}	55	–2	4	6	5	30
J_{43}	1	1	2	1	5	5
J_{58}	35	–11	5	16	1	16
J_{93}	2	–3	3	6	5	30
J_{94}	2	–3	3	6	5	30
J_{95}	2	–3	3	6	5	30
J_{96}	2	–3	4	7	5	35
J_{97}	100	–3	3	6	5	30
					TD	1,256

Table 8 presents the *TD* values estimated from the dates of completion of processing obtained from a Gantt chart developed using the earliest due date (EDD) method, a rule to prioritise jobs with fewer days to the due dates, a conventional technique. In this case, J_4, J_5, and J_{43} are added to the delayed ten jobs for which due-date values are negative. In the example of J_1, the tardiness is seven days because the due-date value is '–6' and one day is added to complete processing. That multiplied by α_1 100 is '700'. Applying the same procedure to the other 12 jobs presented in Table 8 gives the *TD* value of 1,256.

The feasibility of this *TD* value is not guaranteed because the setup-workers' load balance is not considered. However, this *TD* value indicates a solution that is close to neighbourhood solutions. If the solutions, which have equivalent values to this *TD* value, are obtained from the proposed method, then the proposed technique can be regarded as valid.

4.2 Parameter setting

Parameter settings of GA, which are based on the results of simple experiment introduced from the operation of real-world SME as described above, are presented in Table 9. The weighting factors in crowdedness are set to diverse values under the condition of 'k_{TD} + k_{SL} + k_{WIP} = 1' in each island. This enables the solution to maintain diversity.

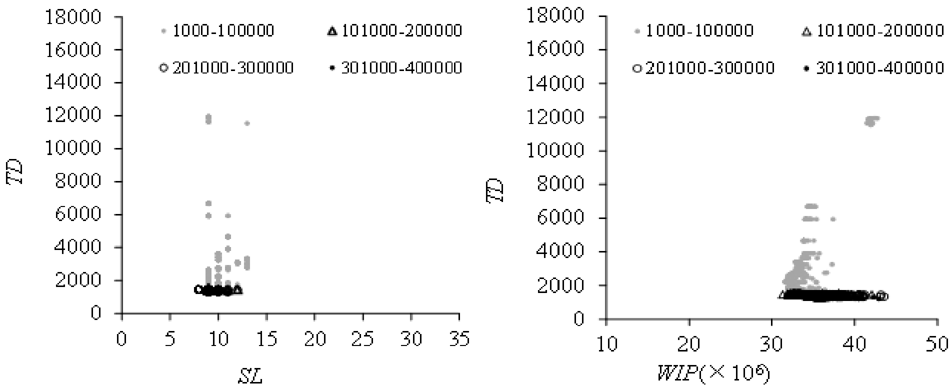
Table 9 Parameters of proposed algorithm

		TS-HGA	*MI-GA*
Population size		20	144
Number of islands		-	12
Population size on island		-	12
Crossover rate between chromosomes in matched machines		70%	70%
Crossover rate between two points of gen in matched machines		70%	70%
Mutation rate		20%	20%
Interval of TS crossing		2,000 generations	-
Number of GA operations in each TS crossing		500	
Interval of TS mutation		1,000 generations	-
Number of GA operations in each TS mutation		5	
Interval of migration		-	5,000 generations
Weight in crowdedness	*TD*	5	k_{TD}
	SL	1	k_{SL}
	WIP	1	k_{WIP}
Number of generations		400,000	400,000

5 Results

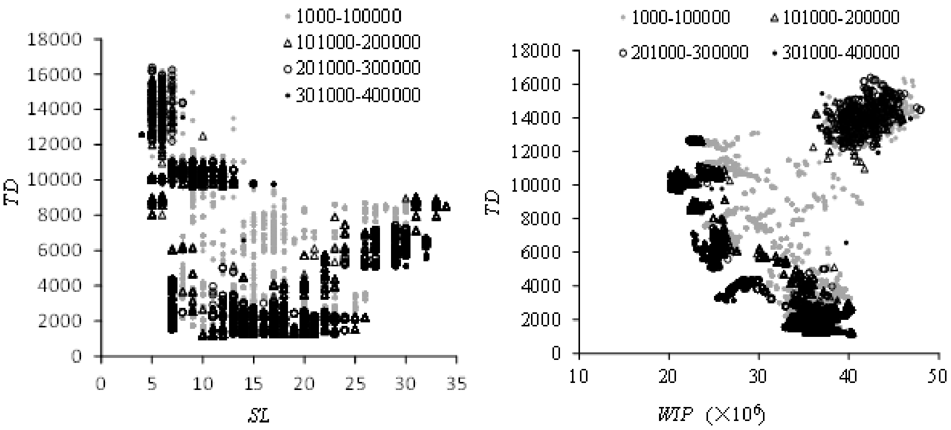
Figure 5 presents the distribution of solutions every 100,000 generations obtained using TS-HGA: Figure 5(a) is an *SL-TD* projection plane; Figure 5(b) is a *WIP-TD* projection plane. In the scatter plot, grey points indicate solutions in the generation of 1,000–100,000, triangles represent solutions in the generation of 101,000–200,000 generations, open circles represent solutions in the generation of 201,000–300,000, and black points indicate solutions in the generation of 301,000–400,000. The greater the increase in the number of generations, the closer the solutions approach to the minimum in each objective.

Figure 5 Derived solutions by TS-HGA at typical generations, (a) *SL-TD* projection plane (b) *WIP-TD* projection plane



(a) (b)

Figure 6 Derived solutions by *MI-GA* at typical generations, (a) *SL-TD* projection plan (b) *WIP-TD* projection plane



(a) (b)

Figure 6 presents the distribution of solutions every 100,000 generations obtained using *MI-GA*: Figure 6(a) is the *SL-TD* projection plane; Figure 6(b) is the *WIP-TD* projection plane. In Figure 6, as in Figure 5, each point denotes a solution in each generation. The scatter plot includes the solutions for all of 12 islands. Even black points scatter over a wide range on both the *SL-TD* projection plane and the *WIP-TD* projection plane, but from two projection planes, the cluster of solutions seems to provide a convex surface.

Figure 7 presents a comparison among solutions near optimal solution in *TS-HGA* based on Figure 5, those in *MI-GA* based on Figure 6, and the solution obtained using EDD.

Figure 7 Neighbourhood solutions based on *TS-HGA* and *MI-GA* and the solution based on EDD (see online version for colours)

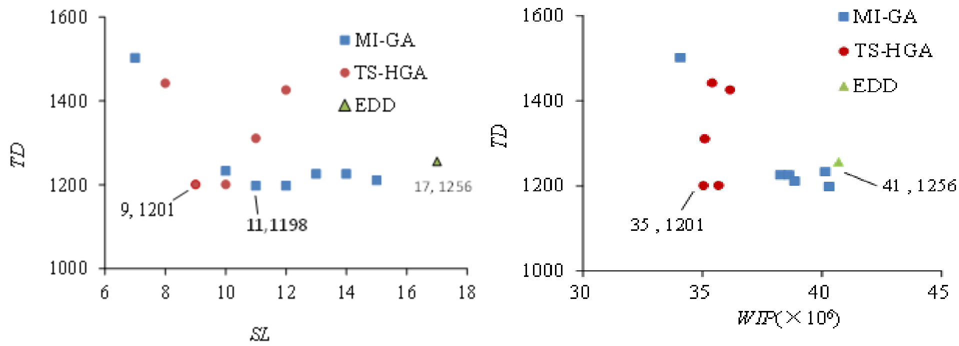


Figure 8 Convergence of *TD* values

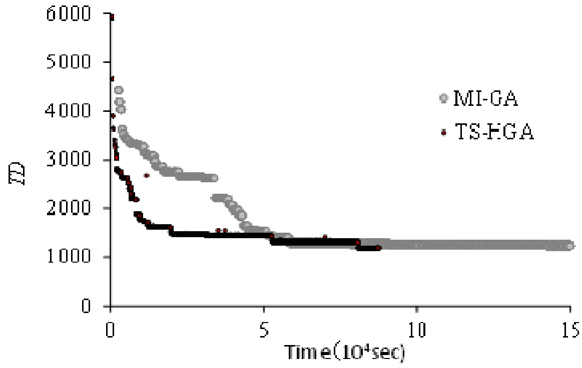


Figure 8 shows how solutions converge over time for two cases that converge to approximately 1200 by *TD* values in Figure 7. In Figure 8, the grey dots represent the curve consisting of solutions with a *TD* value of 1198 obtained using *MI-GA*; the black dots represent that of 1,201 obtained using *TS-HGA*. The solutions for the *TS-HGA* approach converged faster than the solutions for *MI-GA*.

Figure 9 Number of daily setup done by each setup worker (see online version for colours)

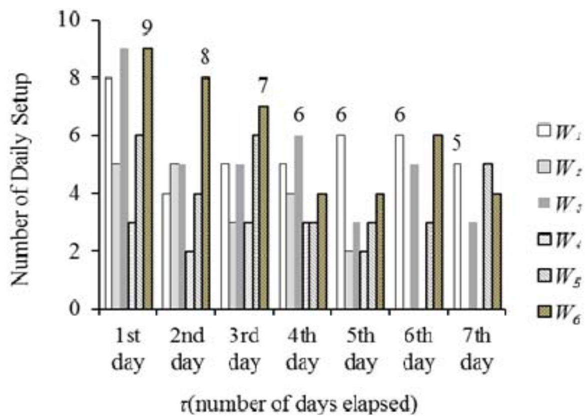


Figure 9 presents changes in the daily load of setup-workers for seven days obtained from the Gantt chart for the solution (*TD* = 1,201, *SL* = 9, *WIP* = 35 × 10^6): the lowest *TD* value in the Pareto-optimal solutions in Figure 7(a). The numbers of daily setup decrease daily as the numbers of jobs decrease daily in Figure 4.

Figure 10 Changes in work-in-process

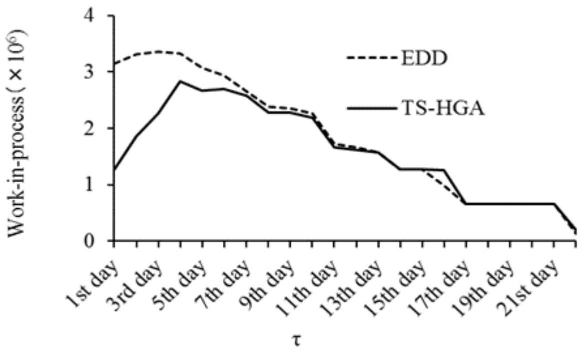


Figure 10 presents a comparison of changes in work-in-process on shop floor referred form Figure 7(b). In Figure 10, the black curve represents the changes in the solution (WIP = 35 × 10^6) obtained using *TS-HGA*. The grey curve represents the changes in the solution (*WIP* = 41 × 10^6) obtained using EDD. The solution obtained using TS-HGA provides not only lower *WIP* but also a lower maximum inventory level on the shop floor.

6 Discussion

Comparing the solutions obtained for *TS-HGA* in Figure 5 and *MI-GA* in Figure 6 reveals that *TS-HGA* obtained fewer solutions than *MI-GA* because of population size, but the solutions obtained by *TS-HGA* converge as the generations advance. This result implies that the TS algorithm functions effectively even with small population size. Actually, *MI-GA* method provides wide-area solutions obtained even at high generation on both *SL-TD* projection plane and *WIP-TD* projection plane because the weighting factors in crowdedness are set to diverse values in each island.

In Figure 7, both the *TS-HGA* method and *MI-GA* method are shown to provide superior performance to EDD from the perspective that the most important object in MTO manufacturing scheduling is the due date. Actually, EDD is inferior to *TS-HGA* method and *MI-GA* method in spite of the due-date priority feature because tardiness results from the mismatched relation between the due date and the production lead time. The production lead-time of each job depends on some factors such as the number of lot size and the number of required operations. Therefore, the jobs with large lot size and or large number of required operations should be released to the shop floor earlier, even if these due-dates are large. However, these factors aside from the due date are not considered in the EDD method. As a result, jobs with a low due date, small lot size, and low number of required operations are released to the shop floor earlier in spite of the small production lead time. Jobs with a long lead time because of the large lot size and large number of necessary operations are released to the shop floor later because of the due date.

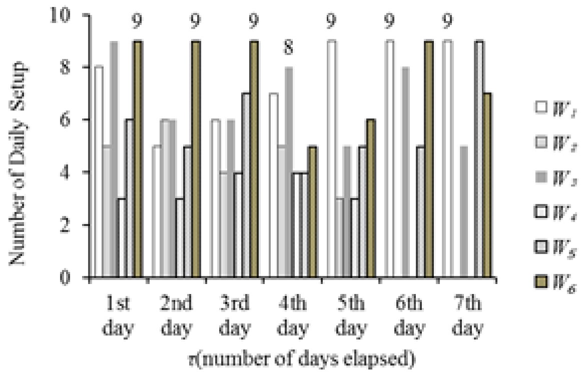
In Figure 7, *TS-HGA* provides more excellent solutions than *MI-GA* for both projection planes, which implies that *TS-HGA* provides equivalently excellent values to *MI-GA* under the condition of small *SL* and small *WIP*. In other words, it does so under the condition of small setup workers' workload and small work-in-process in spite of small population size in GA. TS avoids useless crossover and mutation operations.

Returning to Figure 9, it might be readily apparent that numbers of the daily setup decrease daily as the numbers of jobs decrease daily in Figure 4. However, decreasing ratio in numbers of daily setup should be confirmed based on the decreasing ratio in jobs, as described in Sub-Section 1.2. The maximum setup workloads on each day in Figure 9 are transformed based on β_τ in the middle row of Table 10. The transformed values show that the maximum setup workloads on each day are balanced, as shown in the bottom row of Table 10. These values are also presented in Figure 11.

Table 10 The largest number of setup on each elapsed day

τ (number of elapsed days)	*1*	*2*	*3*	*4*	*5*	*6*	*7*
L_{τ_max}	9	8	7	6	6	6	5
β_τ	1	0.93	0.86	0.8	0.74	0.68	0.62
$\lceil L_{\tau\_max}/\beta_\tau \rceil$	9	9	9	8	9	9	9

Figure 11 Number of daily setup by each workers converted by job decrease (see online version for colours)



7 Conclusions

This study proposed a hybrid GA with TS strategy (*TS-HGA*) for multipurpose optimisation to minimise weighted tardiness, to balance the setup workload, and to minimise work-in-process in a flexible job shop scheduling problem (MO-FJSP). Computational experiments were conducted for the dataset from a real-world MTO of SME to investigate the *TS-HGA* performance. The solutions obtained using the *TS-HGA* are compared with those obtained using conventional EDD, and GA using multi-island methods (*MI-GA*). Results confirmed the effectiveness of the proposed method.

Additionally, the solutions obtained using a *TS-HGA* approach converged faster than when using *MI-GA* because tabu lists help to avoid redundant solutions.

This study investigated the performance of a real-world MTO mode SME from the perspective of due-date, setup workload, and work-in-process inventory views using experimental data based on daily operations. Although these experiments were conducted under the conditions of data reflecting current operations in a MTO of SME, replacing the data related to conditions such as the number of machines, performance of machines, number of setup workers, setup worker skill levels and the upper limit of inventory, the proposed approach is applicable not only to scheduling problems but also to other management decision making related to estimation of the investment of resources such as machines and worker capacity and estimation of the reasonability of orders from customers.

From a practical application perspective, the available computation time may become shorter than several hours because the contractor has to provide the time of delivery to the customer as soon as possible. The diversity of the products also has to be considered in MTO manufacturing. Therefore, to handle both enhancing the calculation speed and adaptive flexibility of the genotype, parameter-free approach will be considered in the future work.

References

Ayyagari, M., Beck, T. and Demirguc-Kunt, A. (2007) 'Small and medium enterprises across the globe', *Small Business Economics*, Vol. 29, No. 4, pp.415–434.

Brandimarte, P. (1993) 'Routing and scheduling in a flexible job shop by tabu search', *Annals of Operation. Res.*, Vol. 41, No. 3, pp.157–183.

Deb, K., Pratap, A., Pratap, Agarwai, S. and Meyarivan, T. (2002) 'A fast and elitist multi objective genetic algorithm: NSGA–II', *IEEE Trans. on Evolutionary Computation*, Vol. 6, No. 2, pp.182–197.

Defersha, F.M. and Chen, M. (2010) 'A parallel genetic algorithm for a flexible job-shop scheduling problem with sequence dependent setups', *Int. J. Adv. Manuf. Technol.*, Vol. 49, Nos. 1–4, pp.263–279.

Giovanna, C., Alfredo, D.M. and Lucio, C. (2012) 'Corporate social responsibility: a survey among SMEs in Bergamo', *Procedia Social and Behavioral Sciences*, Vol. 62, pp.325–341.

Glover, F. (1986) 'Future path for integer programming and links to artificial intelligence', *Comput. & Ops. Res.*, Vol. 13, No. 5, pp.533–549.

Hendry, L.C. (1998) 'Applying world class manufacturing to make-to-order companies: problems and solutions', *Int. J. Operation Operations & Production Management*, Vol. 18, No. 11, pp.1086–1101.

Huang, R.H., Yang, C.L. and Cheng, W.C. (2013) 'Flexible job shop scheduling with due window – a two-pheromone ant colony approach', *Int. J. Production Economics*, Vol. 141, No. 2, pp.685–697.

Jamali, M.A., Voghouei, H. and Mohd Nor, N.G. (2015) 'Information technology and survival of SMEs: an empirical study on Malaysian manufacturing sector', *Inf. Technol. Manag.*, Vol. 16, No. 2, pp.79–95.

Kacem, I., Hammadi, S. and Borne, P. (2002a) 'Approach by localization and multi-objective evolutionary optimization for flexible job-shop scheduling problems', *IEEE Trans. Syst. Man and Cybernetics – Part C: Application and Reviews*, Vol. 32, No. 1, pp.1–13.

Kacem, I., Hammadi, S. and Borne, P. (2002b) 'Pareto-optimality approach for flexible job-shop scheduling problems: hybridization of evolutionary algorithms and fuzzy logic', *Math. and Comp. in Simul.*, Vol. 60, Nos. 3–5 SI, pp.245–276.

Kaminsky, P. and Kaya, O. (2008) 'Scheduling and due-date quotation in a make-to-order supply chain', *Naval Research Logistics*, Vol. 55, No. 5, pp.444–458.

Li, J.Q., Pan, Q.K. and Gao, K.G. (2011) 'Pareto-based discrete artificial bee colony algorithm for multi-objective flexible job shop scheduling problems', *Int. J. Adv. Manuf.*, Vol. 55, Nos. 9–12, pp.1159–1169.

Li, J.Q., Pan, Q.K. and Liang, Y.C. (2010) 'An effective hybrid tabu search algorithm for multi-objective flexible job shop scheduling problems', *Computer Ind. Eng.*, Vol. 59, No. 4, pp.647–662.

Li, J.Q., Pan, Q.K., Fatih, M. and Tasgetiren (2014) 'A discrete artificial bee colony algorithm for the multi-objective flexible job-shop scheduling problem with maintenance activities', *Applied Mathematical Modelling*, Vol. 38, No. 3, pp.1111–1132.

Luh, P.B., Zhou, X. and Tomastik, R.N. (2000) 'An effective method to reduce inventory in job shop', *IEEE Transaction on Robotics and Automation*, Vol. 16, No. 4, pp.420–424.

Morinaga, Y., Nagao, M. and Sano, M. (2014) 'Optimization of flexible job-shop scheduling with weighted tardiness and setup-worker load balance in make-to-order manufacturing', *Proceedings of the Joint Seventh International Conference on Soft Computing and Intelligent Systems(SCIS), and 15th International Symposium on Advanced Intelligent Systems (ISIS)*, pp.87–94, ISBN: 978-1-4799-5954-9.

Moslehi, G. and Mahnam, M. (2011) 'A Pareto approach to multi-objective flexible job-shop scheduling problem using particle swarm optimization and local search', *Int. J. Prod. Econ.*, Vol. 129, No. 1, pp.14–22.

Nagao, M., Sugimoto, T., Morinaga, Y. and Sano, M. (2015) 'Diversity oriented multi-objective island based genetic algorithm for flexible job shop scheduling considering setup operator regulation', *J. Human Environmental Studies*, Vol. 13, No. 1, pp.1–12.

Nenes, G., Castagliola, P., Celano, G. and Panagiotidou, S. (2014) 'The variable sampling interval control chart for finite-horizon processes', *IIE Transactions*, Vol. 46, No. 10, pp.1050–1065.

Proth, J.M. (2007) 'Scheduling: new trends in industrial environment', *Annual Reviews in Control*, Vol. 31, No. 1, pp.157–166.

Sawik, T. (2011) 'Selection of a dynamic supply portfolio in make-to-order environment with risks', *Com. & Oper. Res.*, Vol. 38, No. 4, pp.782–796.

Shahasaveri-Pour, N. and Ghasemishabankareh, B. (2013) 'A novel hybrid meta-heuristic algorithm for solving multi objective flexible job shop scheduling', *J. Manuf. Systems*, Vol. 32, No. 4, pp.771–780.

Simao, J.M., Stadzisz, P.C. and Morel, G. (2006) 'Manufacturing execution systems for customized production', *J. Materials Processing Tech.*, Vol. 179, No. 1, pp.268–275.

Thuerer, M., Stevenson, M., Silva, C. and Hung, G. (2012) 'The application of workload control in assembly job shops: an assessment by simulation', *Int. J. Prod. Res.*, Vol. 50, No. 18, pp.5048–5062.

Turkyilmaz, A. and Bulkan, S. (2015) 'A hybrid algorithm for total tardiness minimisation in flexible job shop: genetic algorithm with parallel VNS execution', *Int. J. Production Research*, Vol. 53, No. 6, pp.1832–1848.

Veen-Dirks, P. (2005) 'Management control and the production environment: a review', *Int. J. Prod. Econ.*, Vol. 93, pp.263–272.

Wang, C., Wang, Z. Ghenniwa, H.H. and Shen, W. (2011) 'Due-date management through iterative bidding', *IEEE Transactions on Systems Man and Cybernetics Part A – Systems and Humans*, Vol. 41, No. 6, pp.1182–1198.

Wei, J. and Ma, Y-S. (2014) 'Design of a feature-based order acceptance and scheduling model in an ERP system', *Computers in Industry*, Vol. 65, No. 1, pp.64–78.

Wu, Z. and Weng, M.X. (2005) 'Multiagent scheduling method with earliness and tardiness objectives in flexible job shops', *IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics*, April, Vol. 35, No. 2, pp.293–301.

Xia, W. and Wu, Z. (2005) 'An effective hybrid optimization approach for multi-objective flexible job-shop scheduling problem', *Computer & Industrial Engineering*, Vol. 48, No. 2, pp.409–425.

Xing, L.N., Chen, Y.W. and Yang, K.W. (2009) 'Multi-objective flexible job shop schedule, design and evaluation by simulation modeling', *Applied Soft Computing*, Vol. 9, No. 1, pp.362–376.

Zhang, G., Shao, X., Li, P. and Gao, L. (2009) 'An effective hybrid particle swarm optimization algorithm for multi-objective flexible job-shop scheduling problem', *Comput. Ind. Eng.*, Vol. 56, No. 4, pp.1309–1318.