Journal of Industrial Engineering International (2020) 16:135–146
https://doi.org/10.1007/s40092-018-0278-2

TECHNICAL ARTICLE

Role of batch size in scheduling optimization of flexible manufacturing system using genetic algorithm

Muhammad Umair Akhtar[1] · Muhammad Huzaifa Raza[1] · Muhammad Shafiq[1]

Received: 1 November 2017 / Accepted: 12 June 2018 / Published online: 16 June 2018

Abstract

Flexible manufacturing system (FMS) readily addresses the dynamic needs of the customers in terms of variety and quality. At present, there is a need to produce a wide range of quality products in limited time span. On-time delivery of customers' orders is critical in make-to-order (MTO) manufacturing systems. The completion time of the orders depends on several factors including arrival rate, variability, and batch size, to name a few. Among those, batch size is a significant construct for effective scheduling of an FMS, as it directly affects completion time. On the other hand, constant batch size makes MTO less responsive to customers' demands. In this paper, an FMS scheduling problem with n jobs and m machines is studied to minimize lateness in meeting due dates, with focus on the impact of batch size. The effect of batch size on completion time of the orders is investigated under following strategies: (1) constant batch size, (2) minimum part set, and (3) optimal batch size. A mathematical model is developed to optimize batch size considering completion time, lateness penalties and setup times. Scheduling of an FMS is not only a combinatorial optimization problem but also NP-hard problem. Suitable solutions of such problems through exact methods are difficult. Hence, a meta-heuristic Genetic algorithm is used to optimize scheduling of the FMS.

Keywords Flexible manufacturing system (FMS) · Scheduling optimization · Batch size, due dates · Completion time · Genetic algorithm (GA)

Introduction

An FMS is a highly automated production system consisting of a group of computer numerical control machine tools, linked by material handling system and controlled by a distributed control system. An FMS can simultaneously process medium variety and medium size volumes of products (Browne et al. 1984). An FMS is a sophisticated production system to respond dynamic variations of contemporary market including lead time reduction, flexibility to respond market variations and higher productivity (Atmani and Lashkari 1998). The major problems that an FMS has to face include designing, planning, scheduling and controlling. Among these, scheduling problem is a major challenge (Stecke 1985). An FMS's scheduling differs from a conventional job shop scheduling due to routing flexibility of parts (Jain and Elmaraghy 1997).

Scheduling is the assignment of resources over time to perform tasks. In conventional scheduling system, only one resource is considered, which processes the parts. An FMS scheduling differs from conventional job shop scheduling. The complexity of FMS scheduling is due to the flexibility of an FMS in terms of machine, product, operation and routing (Browne et al. 1984). Scheduling of an FMS is effected by several factors including orders arrivals, due dates and batch size (Liu and MacCarthy 1996). The production management has to set policies whether to handle periodic or continuous orders of the customers. Meeting customers' due dates play key role in make to order system and the FMSs are mostly suited for MTO system. Due dates are effected by various factors including arrival rate,

✉ Muhammad Shafiq
dr.shafiq@uettaxila.edu.pk; shafiqaatir1@gmail.com

Muhammad Umair Akhtar
rana.umair977@gmail.com

Muhammad Huzaifa Raza
huzaifaraza4@gmail.com

[1] Department of Industrial Engineering, University of Engineering and Technology, Taxila, Pakistan

variability and batch size. Among these factors, batch size is significant factor for effective scheduling of an FMS as it directly affects due dates. In addition, constant batch size makes MTO less responsive to customers' demands (Beemsterboer et al. 2017).

Extensive literature has been studied to address scheduling of an FMS. The scheduling of an FMS is NP-hard problems, and is solved by heuristics to find near optimal solution (Baker 1974). Close conformance to due dates is accomplished by minimizing mean and maximum tardiness. In MTO system, minimization of tardiness is significant. Nidhiry and Saravanan (2014) addressed FMS scheduling problem to minimize idle time of the machines and total penalty cost. Sixteen computer numeric-control machine tools were used to process eighty different parts in an FMS. The results of the proposed heuristic were compared with cuckoo search and particle swarm optimization (PSO). The results depicted that the proposed approach outperformed cuckoo search and PSO.

Keung et al. (2003) proposed intelligent hierarchical control technique for an FMS. The paper aimed to optimize machine utilization as well as to balance capacity of tool magazine. The proposed model was assessed based on two benchmarks: earliness and tardiness penalty costs. Jerald et al. (2005) determined optimal Scheduling of three FMS with eight, fifteen and sixteen machines. The objective was to optimize idleness of the machine and penalty costs. Saravanan and Haq (2008) applied scatter-search approach to optimize FMS scheduling, considering multiple objectives, including minimization of machine idle time and total penalty costs in case of exceeding the due dates.

An important performance measure to determine utilization of an FMS is in optimizing makespan and completion time. Chan et al. (2005) addressed an FMS problem in two different situations. In first case, only three machines were under consideration, whereas in second scenario thirty-three machines were accommodated. The objective was to optimize makespan. Reddy and Rao (2006) addressed scheduling of machines as well as Automated Guided Vehicles (AGVs) simultaneously in an FMS to optimize makespan and mean flow time. The proposed approach was practiced for an FMS having six machines with two guided vehicles. The results depicted that the presented algorithm has diverse solutions. Kim et al. (2007) solved scheduling problems in an FMS with the objective to optimize makespan in an FMS having ten machines. The results obtained were compared with other techniques and it was found that the proposed algorithm outperformed in terms of quality of solution and convergence speed. Multi-mode Resource Constrained Project Scheduling Problem was solved using hybrid genetic algorithm. Feasible schedules were based on the minimum project completion time (Lova et al. 2009).

Udhayakumar and Kumanan (2010) performed multi-objective scheduling of tasks of AGVs using ant colony optimization (ACO). The objective was to maximize AGVs utilization which has also been validated on several problems. Machines and AGVs scheduling in flexible manufacturing system was also addressed by Chaudhry et al. (2011) using genetic algorithm spreadsheets to minimize total completion time. Zhang et al. (2012) proposed a model for flexible job shop scheduling considering transportation constraints and bounded processing time to minimize makespan and storage. Different types of problems were tested including sequencing as well as bounded processing times. The proposed model outperformed said types of scheduling problems. Raj et al. (2014) presented combined machines and tools scheduling for multi-machine in an FMS to achieve best optimal sequences with the aim to minimize makespan. A meta-heuristic GA was developed to minimize makespan of the scheduling problem. The parameters of the GA were determined by Taguchi orthogonal array. Analysis of variance was performed to examine significant factors affecting makespan (Candan and Yazgan 2015).

In MTO system, batch size plays a significant role for effective scheduling of an FMS due to its direct effect on completion time. Furthermore, constant batch size makes MTO less responsive to customers' demands. A few researchers have addressed batch size problem in an FMS scheduling. Machine utilization of a flexible manufacturing cell was improved using hierarchic approach. In first stage, batches of parts were determined. In second stage, batches were sequenced and scheduled. A fuzzy-based approach was proposed to solve machine loading problem in an FMS. The sequence of jobs was determined by evaluating the contribution of each job in terms of batch size and operation processing time (Vidyarthi and Tiwari 2001). Cheng et al. (2012) developed a mixed integer linear programming model to address scheduling problem of parallel batch processing of jobs to minimize makespan and completion time using polynomial time algorithm. Geng and Yuan (2018) investigated the unbounded parallel batching machine to minimize makespan and completion time using pareto optimization. Matin et al. (2017) proposed PSO to optimize batch size in flow shop to minimize makespan and completion time. Different batch size compositions were used on machines and an optimal batch size of all jobs on each machine was obtained. Ying and Lin (2018) addressed hybrid flow shop scheduling problem to minimize makespan using self-tuning iterated greedy algorithm.

Scheduling of an FMS is NP-hard problem and it is more complex as compared to classical job shop scheduling problems. Different heuristic-based approaches including cuckoo search, PSO, GA and artificial neural network have been used in an FMS scheduling (Sankar et al. 2003; Jerald et al. 2005; Burnwal and Deb 2013; Sadeghian and Sadeghian 2016). A hybrid multi-objective GA was presented to optimize makespan, AGV travel time and penalty cost due to jobs lateness. The proposed algorithm proved feasibility and effectiveness for all the objectives (Umar et al. 2015). A non-dominated sorting biogeography-based approach to solve multi-loading–unloading problem in an FMS was addressed to minimize makespan and total earliness (Rifai et al. 2018).

The main research gap identified in the literature is the lack of considering the effect of batch size on due dates and makespan in MTO system in an FMS scheduling. In this paper, an FMS scheduling problem is addressed to minimize lateness in meeting due dates. A mathematical model is developed for optimizing the batch size. GA is used to optimize lateness of each order and penalty cost. Then, multi-objective Pareto optimization is performed for three batch size strategies to evaluate its effect on completion time. These strategies are then compared with each other to investigate completion time.

The rest of the paper is organized as follows. Section 2 discusses mathematical model description. Section 3 gives insight of technique used and Sect. 4 presents results and discussion. In Sect. 5, conclusions and future work recommendations are presented.

Description of model

In this study, an FMS scheduling problem is studied to minimize penalty costs incurred due to lateness, which helps in minimizing completion time of jobs. The scheduling problem of five parts and six machines in an FMS having four flexible machining cells (FMC) is discussed. Each part has predefined sequence of operations and corresponding processing time. The structure of our FMS is depicted in Fig. 1.

The FMS contains one to two CNC machines, each supported with tool magazines. ATC is capable of changing tool according to the part. Buffer storage is used to store parts temporarily to mitigate the effect of starving and blockage. Part-carrying conveyors and shuttle are used to transfer parts between machine and AS/RS. Automated storage and retrieval systems are used to store parts and to deliver/retrieve materials to/from the storage racks, respectively. All FMCs are linked by AGVs which are responsible for: intercellular movements, finish jobs movements to unloading station and work in process to AS/RS. Four flexible manufacturing cells and six machines are considered. Each cell comprises one to two machines and is capable to process five parts A, B, C, D and E. The sequence as well as processing time of each part is different. The distribution of the machines at each FMC is shown in Table 1.

Assumptions

The following assumptions have been set for our proposed model:

- There are five different parts to be processed ensuring the presence of multi-tools in tool magazine.
- Every part has a predefined processing sequence incorporating batch size, due dates and penalty costs in case of not meeting due dates. The processing time of every part is known and deterministic.
- Every machine can perform only one operation at a time.
- The sequence of operations for all parts is predetermined.
- The sequence-dependent setup times are considered.

Notations

In this paper, FMS scheduling is addressed to minimize penalty costs incurred due to lateness and an optimal batch size is achieved to minimize completion time. Genetic algorithm is used to solve the complex scheduling problem. The notations used in the development of mathematical model are listed in Table 2.

Mathematical model

This objective function minimizes total lateness and total penalty cost incurred in case of exceeding due dates of each order received from customers. Total penalty cost for all orders is calculated using Eq. (1). Second objective minimizes total completion time of all jobs in an order o and it is given in Eq. (2). Third objective function minimizes sequence-dependent setup times and it is given in Eq. (3).

$$Z_1 = (\mathrm{Min}) \sum_{o=1}^{o} \mathrm{LT}_o \times \mathrm{PC}_o, \quad \forall o \tag{1}$$

$$Z_2 = \sum_{o=1}^{O} \quad \sum_{i=1}^{I} X_{ik} \times \mathrm{CT}_i \Bigg), \quad \text{for } k = 1, 2, 3, \ldots, K, \quad \forall o \tag{2}$$

$$Z_3 = (\mathrm{Min}) \sum_{m=1}^{M} \sum_{i=1}^{I} \sum_{j=1}^{I} X_{mij} \times \mathrm{ST}_{mij}, \quad \forall m \tag{3}$$

Subject to

$$\sum_{m=1}^{M} X_{im} = 1, \quad \forall_i \tag{4}$$

$$\mathrm{LT}_o = \mathrm{Max}(0, \mathrm{CT}_o - \mathrm{DD}_o), \quad \forall o \tag{5}$$

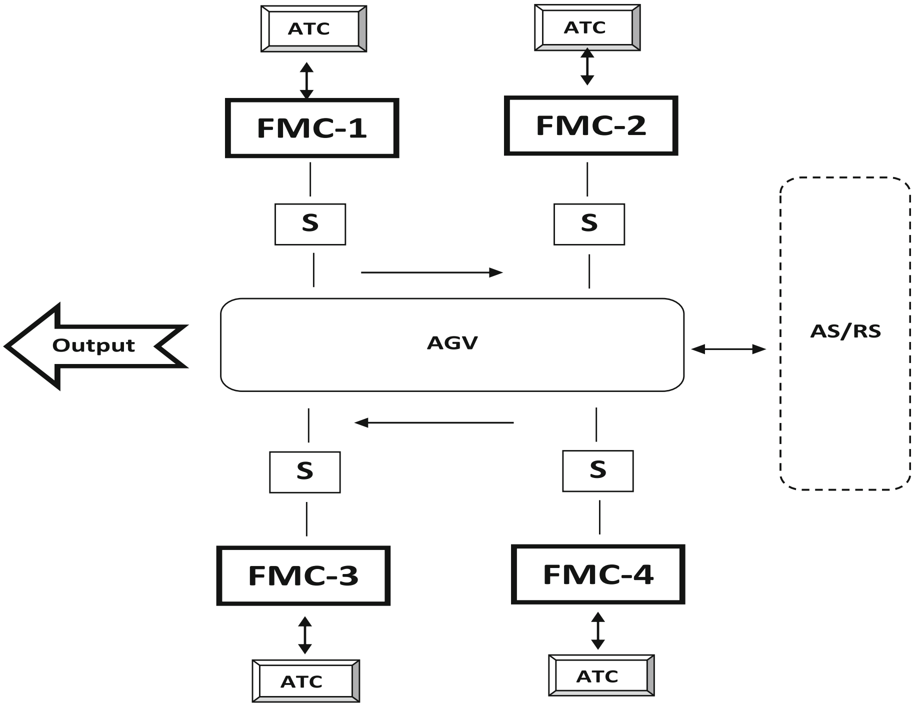


Fig. 1 Structure of an FMS. *FMC* Flexible manufacturing cell, *ATC* Automatic tool changer, *AGV* Automated guided vehicles, *AS/RS* Automated storage and retrieval system, *S* Shuttle

Table 1 Machine distribution in each FMC

Flexible manufacturing cell (FMC)	Machines
FMC-1	1
FMC-2	3.6
FMC-3	5
FMC-4	2.4

Table 2 Model's notations

Notations	
k	Position of batch in sequence k = 1, 2, 3, …, K
i, j	Parts $i, j \in \{1, 2, 3, \ldots, I\}$
m	Machine number m = 1, 2, 3, …, M
o	Order number o = 1, 2, …O
q_i	Batch size of part i
n_i	Number of batches of part i
d_i	Demand of part i
N	Total number of batches
PC_o	Penalty cost of an order o
LT_o	Lateness of an order o
CT_i	Completion time of part i
ST_i	Starting assembly time of an order in a sequence of part i
RL_o	Release of an order o
CT_o	Completion time of an order o
DR_o	Estimated duration of an order o
DDo	Due date of an order o
MPS	Minimum part set

$$q_i \times n_i = d_i \quad \text{for } i = 1, 2, 3, \ldots, I \tag{6}$$

$$N = \sum_{i=1}^{I} n_i \tag{7}$$

$$\text{ST}_i \geq \sum_{o=1}^{O} \text{RL}_o \times X_{io}, \quad \forall o \tag{8}$$

$$\text{ST}_{i+1} \geq \text{ST}_i + \sum_{o=1}^{O} \text{RL}_o \times X_{io}, \quad i = 1, 2, \ldots, I-1 \tag{9}$$

$$\text{CT}_i = \text{ST}_i + \sum_{o=1}^{O} \text{DR}_o \times X_{io}, \quad \forall i \tag{10}$$

Equation (4) shows that one part i cannot be assigned to more than one machine. In Eq. (5), CT_o shows completion time of order o and DD_o is the corresponding due dates, LT_o guarantees that lateness is a positive value. Equation (6) shows that demand of each part i must be satisfied. Total number of batches can be computed by Eq. (7). Equation (8) shows that starting time of an order o in sequence of part i must be synchronized with order release date. Equation (9) ensures that order production in sequence $i + 1$ can only be started once order in production sequence i is completed. Equation (10) represents completion time of order of part i.

Genetic algorithm

Genetic algorithm introduced by Holland was inspired by the concept of natural and biological evolution (Sivasankaran and Shahabudeen 2014). GA has proven a powerful tool for finding the near optimal solution of combinatorial problems that cannot be handled by exact methods due to their complexity (Zandieh et al. 2010). To apply genetic algorithm to solve real life optimization problems, two basic issues must be addressed: (1) Encoding the solution in the form of chromosomes and (2) fitness function evaluation (Hsu et al. 2005). In GA, the chromosomes encode candidate to find optimum results. Each result is represented by binary numbers as a string of 0 and 1 but other types of encodings are also possible in the GA (Chan et al. 2006). Brief description of the GA implementation procedure is presented in the following section.

GA implementation

Chromosome representation

A chromosome consists of a set of locations known as genes that assume discrete values pertaining to the problem's solution. Each gene represents a job number. Initial job sequence is generated randomly. Each chromosome represents a feasible solution. Chromosome representation is shown in Fig. 2.

Job Sequence	A	E	C	B	D

Fig. 2 Chromosome Representation

Initial population

In this stage, an initial population is created consisting of N chromosome as shown in Fig. 3. Each chromosome represents a potential solution. The greater the size of population, the more chances are to get better solution.

Fitness function evaluation

Each chromosome is evaluated against fitness function. Fitness function minimizes maximum lateness and completion time. Fitness function of each chromosome is calculated to choose best parents among all other chromosomes.

Crossover and mutation

Chromosomes from the population are then selected for reproduction. For crossover, the population is paired into even number by dividing the population by two. Chromosomes having the best fitness function value have greater probability to be selected for reproduction. Chromosomes of worst fitness value are discarded. Crossover method adopted in this paper is elaborated with an example. Demand of each part in an order is A = 15, B = 10, C = 10, D = 10, E = 15 and batch size is 5. The number of batches of each part is given as A = 3, B = 2, C = 2, D = 2, E = 3. The base sequence is AAABBCCDDEEE. For crossover, each pair has two sequences of parts and a random position is selected for partition as shown in Fig. 4.

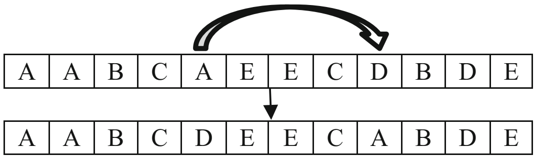
Offspring 1 is produced by taking left portion of parent 1 as it and for right portion, parts from parent 2 are selected in such a way that feasibility of each part batches does not violate. Similarly, offspring 2 is produced. Mutation process is used to increase diversity of solution and it helps to

Fig. 3 Initial Population of N chromosomes

Chromosome 1	A	E	C	B	D
Chromosome 2	C	E	A	D	B
Chromosome 3	E	A	D	B	C
⋮	⋮	⋮	⋮	⋮	⋮
Chromosome N	B	D	A	C	E

Parent 1	AABCA	EEEBDDC
Parent 2	BBAED	EECACDA
Offspring 1	AABCA	EECDBDE
Offspring 1	BBAED	EEDCAAC

Fig. 4 Representation of crossover operator



A	A	B	C	A	E	E	C	D	B	D	E
A	A	B	C	D	E	E	C	A	B	D	E

Fig. 5 Representation of mutation operator

explore search space for global optimization. Mutation rate used is 20%, which means that 0.2 of total genes in the population will be mutated. Mutation of offspring 1 is shown in Fig. 5. The procedure of genetic algorithm is shown in Fig. 6.

Results and discussion

To apply GA, a random schedule is generated with different processing time of parts on machines as shown in Table 3. Each part has a predefined sequence of operations to be followed. A sequence-dependent setup is required each time a part changes in an FMS. The sequence-dependent times are shown in Table 4.

In MTO system, several customers' orders with order date, order quantity and due dates are received. There are total five orders with different order requirements. For example, order 1 is placed on 4rth of the month and its expected duration and due date is 3 and 12 days, respectively. Quantity of each part is also given. Complete details of these orders are shown in Table 5.

Optimization of lateness

At the first stage, scheduling is addressed to optimize total lateness and penalty costs of all orders. The objective function for lateness minimization is coded in MATLAB using and optimized using GA. The coded program is run ten times and best results are achieved. The parameters values used for the GA are listed in Table 6.

The optimum sequence achieved using the GA is 2–1–3–4–5. Total lateness is 1 day and the corresponding penalty cost is $300/day. Only order 4 is exceeding its due date by 1 day. This due date can be adjusted by negotiating with customers or improving the time study data to minimize its expected duration. The scheduling results are summarized in Table 7.

Table 3 Processing time of parts on machines

	M1	M2	M3	M4	M5	M6
A	0	2	5	7	2	0
B	3	2	0	4	0	3
C	1	3	0	2	5	1
D	4	0	4	0	1	5
E	2	5	4	6	1	3

Table 4 Sequence-dependent setup times

	A	B	C	D	E
A	0	1	3	5	8
B	2	0	3	4	6
C	2	5	0	2	3
D	4	7	2	0	8
E	3	8	2	5	0

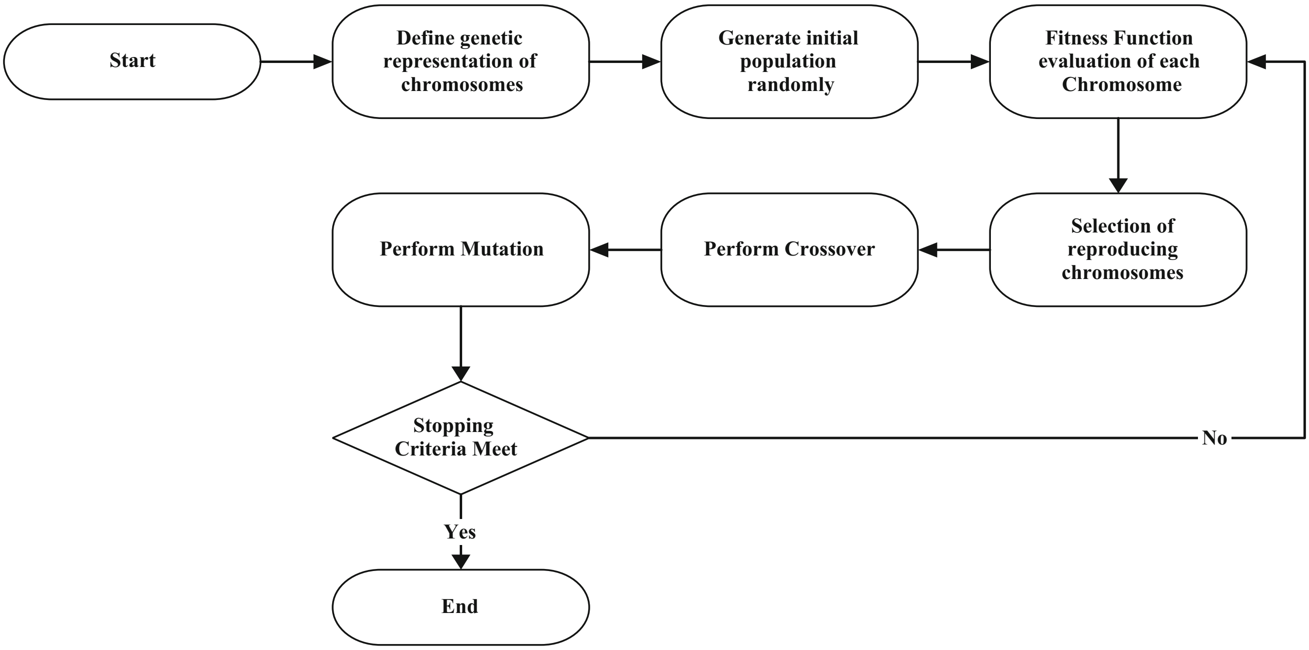

Start
Define genetic representation of chromosomes
Generate initial population randomly
Fitness Function evaluation of each Chromosome
Selection of reproducing chromosomes
Perform Crossover
Perform Mutation
Stopping Criteria Meet
No
Yes
End

Fig. 6 Procedure of Genetic Algorithm

In the second stage, the effect of batch size is investigated on total completion time of jobs. Three strategies are opted and then compared to analyze its effect on completion time. The strategies considered are: (1) Constant batch size, (2) Minimum Part Set, and (3) Variable batch size.

Effect of constant batch size on total completion time

To investigate the effect of batch size on completion time using GA, order 2 is chosen. The rest of the orders can be evaluated in the same way. Constant batch size of 5 and 10 is considered for each part. The number of batches of each part is represented as {A = 2}. For batch size of 5, the number of batches for order 2 is represented as {A = 4, B = 2, C = 6, D = 2 E = 4}. After completion of each batch, a setup is required to change the workstation settings and tools for the next batch. Sequence-dependent setup times, which are more realistic to the actual production environment, are considered. The batch size of each part, processing times and sequence-dependent setup times are imputed in the MATLAB. Completion time and setup time are conflicting objectives. The results of the first scenario in which batch size is taken as 5 are represented in Table 8. Optimum batch sequences, corresponding completion time, and setup time are also given. It can be seen from the Table 8 that as the completion time increases, setup time decreases and vice versa. To get one optimal solution for such conflicting objectives is not possible. Therefore, a set of solutions known as Pareto optimal solutions exist for such kind of problems and solutions need to tradeoff between these two conflicting objectives. The tradeoff solution for batch size of 5 is the first sequence as completion time is more significant to meet due dates.

For second scenario, experimentation is performed to observe larger batch size effect on completion time and setup time. For this case, batch size is taken as 10 for each part and the results are summarized in Table 9. Larger batch size gives better results of completion time and setup time as compared to smaller batch size of 5. Now, the minimum completion time is 362, which is lower than scenario 1. Sequence-dependent setup times are also significantly reduced. The tradeoff solution by comparing the scenario 1 results is the first sequence having completion time of 362 and setup time as 78 min.

Table 6 Parameters values used in the GA

Sr. no.	Parameters	Parameters values
1	Population size	300
2	Generations	500
3	Crossover probability	0.8
4	Mutation probability	0.2

Table 7 Optimal scheduling of orders using the GA

	2	1	3	4	5
Start date	1	6	9	13	16
Completion time	6	9	13	16	21
Due date	8	12	13	15	22
Lateness	0	0	0	1	0
Penalty cost	0	0	0	$300	0
Total penalty costs	$300				

Effect of minimum part set strategy on total completion time

The Minimum Part Set strategy is used in flexible production systems to deal with variety of different parts simultaneously. Mixed model production systems possess soft flexibility of different part styles and mid volume production. In mixed model systems, different parts can be processed on the same machine with small changeovers. For the FMS, MPS strategy is investigated to analyze its effect on completion time by considering sequence-

Table 5 The details of the orders of the customers

	Order 1	Order 2	Order 3	Order 4	Order 5
Order release date	4	1	6	10	14
Duration	3	5	4	3	5
Due date	12	8	13	15	22
Penalty cost ($/day)	100	150	200	300	100
Quantity A	10	20	30	10	30
Quantity B	20	10	20	30	20
Quantity C	0	30	10	10	10
Quantity D	30	10	10	20	0
Quantity E	20	20	40	0	10

Table 8 Scenario#1: Effect of batch size on total completion time and setup time

Batch size	Number of batches	Optimum sequence	Completion time (Min)	Setup time (Min)
Batch size = 5	A = 4, B = 2, C = 6, D = 2, E = 4	AAAAACCCCCCCCCCCCCCC AAAAAAAAAAAAAAABBBBBBBBBB CCCCCDDDDDDDDDDCCCCCCCCCC EEEEEEEEEEEEEEEEEEEE	364	96
		BBBBBBBBBBCCCCCCCCCC AAAAAAAAAAAAAAAAAAAA CCCCCCCCCCDDDDDDDDDD CCCCCCCCCCEEEEEEEEEEEEEEEEEEEE	382	90
		CCCCCCCCCCCCCCCCCCCC AAAAAAAAAAAAAAAAAAAA BBBBBBBBBBCCCCCDDDDDDDDDD CCCCCEEEEEEEEEEEEEEEEEEEE	398	78
		BBBBBBBBBBAAAAAAAAAA AAAAAAAAAACCCCCCCCCC CCCCCCCCCCDDDDDDDDDD CCCCCCCCCCEEEEEEEEEEEEEEEEEEEE	407	72
		AAAAAAAAAAAAAAAAAAAA BBBBBBBBBBCCCCCCCCCC CCCCCCCCCCCCCCCDDDDDDDDDD CCCCCEEEEEEEEEEEEEEEEEEEE	423	66

dependent setup times. MPS demand is obtained by dividing each part original demand by highest common divisor (*h*). For example, demand of order 2 is {A = 20, B = 10, C = 30, D = 10, E = 20} and *h* is 10. The MPS demand for order 2 is {A = 2, B = 1, C = 3, D = 1, E = 2} and MPS sequence is AABCCCDEE. To determine the completion time and setup time for complete order 2, results are multiplied with *h*. The completion time and sequence-dependent setup times are solved as multi objective as both are conflicting objectives. The program is run ten times and best sequence results are shown in Table 10.

The results show that completion time and setup times are conflicting objectives. Optimization of one objective is giving significant results but at the same time, other objective is not giving better results, as it can be seen for sequences 1–5. As completion time decreases, setup time increases and vice versa. To determine the tradeoff point, each obtained sequence is repeated *h* times to get total objective functions values for order 2. The results are listed in Table 11.

The results show that as completion time decreases, setup times drastically increase. The reason is that, in MPS strategy, a great number of setups are required as compared to batch production. For the sequence 3, completion time is 378 while setup time is 540 min. Even for the tradeoff solution, setup times are much higher than constant batch size results.

Effect of variable batch size on total completion time

Effect of constant batch size and MPS strategy has been investigated in previous sections. In this section, batch size is considered as control variable to analyze its effect on completion time and setup time. For each part in order 2, lower bound and upper bound values of batch size are defined as {lb = 1, ub = 10}. The objective is to optimize batch size by minimizing conflicting objectives. The obtained optimum batch size and the number of batches are represented as {A = (1,20), where 1 means that batch size is 1 and 20 means total number of batches of part A. Similarly, other part's batch size and total batches are represented in Table 12.

The optimum sequences of variables batch sizes listed in Table 12 are the best solution obtained in each run while all Pareto solution of variable batch sizes are shown graphically in the Fig. 7. Each sequence is represented by a different color and the best solution of each sequence is represented in graph by a relatively larger size solid circles. The results show that the optimum solution among all sequences is of sequence 1 which is composed of batch size A = (1,20) B = (2,5) C = (5,6) D = (10,1) E = (2,10).

Table 9 Scenario #2: Effect of constant batch size on total completion time and setup time

Batch size	Number of batches	Optimum sequence	Completion time (Min)	Setup time (Min)
Batch size = 10	A = 2, B = 1, C = 3, D = 1, E = 2	AAAAAAAAAABBBBBBBBBB DDDDDDDDDDCCCCCCCCCC CCCCCCCCCCCCCCCCCCCC EEEEEEEEEEEEEEEEEEEEAAAAAAAAAA	362	78
		AAAAAAAAAAAAAAAAAAAA BBBBBBBBBBDDDDDDDDDD CCCCCCCCCCCCCCCCCCCC EEEEEEEEEEEEEEEEEEEECCCCCCCCCC	393	72
		AAAAAAAAAAAAAAAAAAAA BBBBBBBBBBDDDDDDDDDD CCCCCCCCCCCCCCCCCCCC CCCCCCCCCCEEEEEEEEEEEEEEEEEEEE	401	60
		EEEEEEEEEEEEEEEEEEEE CCCCCCCCCCCCCCCCCCCC CCCCCCCCCCAAAAAAAAAAAAAAAAAAAA BBBBBBBBBBDDDDDDDDDD	429	60

Table 10 Effect of MPS strategy on total completion time

Sr. No.	MPS sequence	Optimum sequence	Completion time (Min)	Setup time (Min)
1	AABCCCDEE	EECCCAABD	51	54
2		EECDCCAAB	46	54
3		AABDCCCEE	42	60
4		AABDCCECE	41	90
5		AACBDCCEE	40	102

Table 11 Total completion time and setup time for complete order

Sr. No.	Optimum sequence	Completion time (Min)	Setup time (Min)
1	EECCCAABD	459	486
2	EECDCCAAB	414	486
3	AABDCCCEE	378	540
4	AABDCCECE	369	810
5	AACBDCCEE	360	918

Comparison of three strategies of batch size

The effect of batch size on completion time and setup time has been investigated and elaborated in previous sections. In this section, each of the strategy results is compared to conclude which batch strategy should be opted in an FMS. All three strategies are represented graphically in the Fig. 8. Constant and variable batch size strategy shows a similar behavior for setup time based on the best value. However, completion time is minimum for variable batch size scenario. The MPS strategy shows abnormal results as compared to other strategies and values of completion time and setup time are too high. Comparing all three scenarios, variable batch size shows the minimum completion time of 353 min and setup time of 78 min. Therefore, it is recommended to use variable batch size strategy to have the flexibility to opt different batch size of each part to meet completion time as well as setup time reductions.

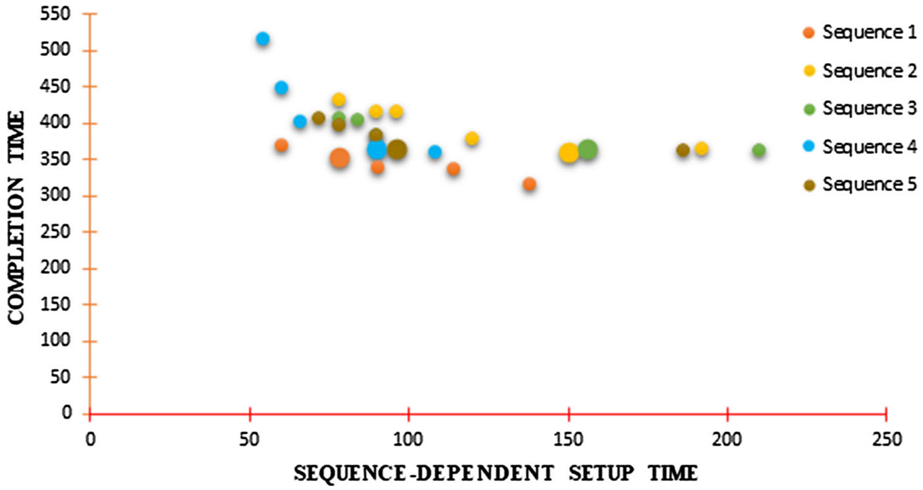
Conclusion

This research has presented an effort to optimize scheduling with objective of minimizing lateness and penalty costs in an FMS. To fulfill on time delivery of orders, the suitable batch size strategy selection and execution is

Table 12 Results of variable batch sizes

Batch	Optimum batch size	Optimum sequence	Completion time (Min)	Setup time (Min)
1	A = (1,20) B = (2,5) C = (5,6) D = (10,1) E = (2,10)	AAAAAAAAAAAAAAAAAAAA BBBBBBBBBBCCCCCCCCCCCCCCC DDDDDDDDDDCCCCCCCCCC EEEEEEEEEECCCCC	353	78
2	A = (2,10) B = (10,1) C = (1,30) D = (5,2) E = (2,10)	AACCCCCCCCCCAAAAAAAAAAAAAA CCCAAAABBBBBBBBBBCCCCCCC DDDDDCCCCCDDDDDCCCCC EEEEEEEEEEEEEEEEEEEE	359	150
3	A = (5,4) B = (1,10) C = (10,3) D = (2,5) E = (5,4)	AAAAABBBBBCCCCCCCCCC AAAAAAAAAAAAAAABBBBB DDDDDDDDCCCCCCCCCCEEEEE DDCCCCCCCCCCEEEEEEEEEEEEEEE	364	156
4	A = (10,2) B = (5,2) C = (10,3) D = (5,2) E = (10,2)	AAAAAAAAAACCCCCCCCCC CCCCCCCCCCAAAAAAAAAA BBBBBBBBBBDDDDDDDDDD CCCCCCCCCCEEEEEEEEEE EEEEEEEEEE	364	90
5	A = (5,4) B = (2,5) C = (5,6) D = (10,1) E = (5,4)	AAAAABBBBBBCCCCCCCCCC CCCCCAAAAAAAAAAAAAAA BBBBDDDDDDDDDDCCCCC CCCCCCCCCCEEEEEEEEEE EEEEEEEEEE	364	96

Fig. 7 Pareto Solutions of all variable batch sizes



significant as batch size directly affects completion time. Different batch sizes yield different completion time and sequence-dependent setup times. Multi-objective Pareto optimization using GA is performed to investigate and compare the results of three batch size strategies: (1) constant batch size of 5 and 10. (2) MPS strategy (3) variable batch size. In first scenario, the batch size of each part is kept 5 and then 10. The larger batch size shows better results. Then, MPS strategy is addressed and results show that setup times are too high for the minimum value of completion time. In the last scenario, variable batch size is used to optimize conflicting objectives. All possible batch sizes of each part are evaluated using the GA and five best results are discussed. The results show that variable

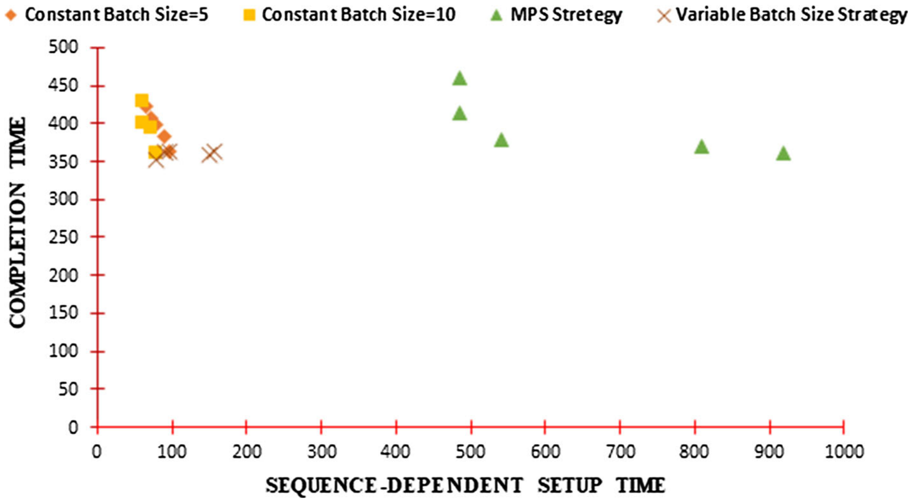


Fig. 8 Comparison of three batch size strategies

batch size strategy exhibits superior results as compared to constant and MPS batch size strategies. This work significantly contributes to an FMS to opt suitable batch size strategy in various demand patterns.

References

Atmani A, Lashkari R (1998) A model of machine-tool selection and operation allocation in FMS. Int J Prod Res 36(5):1339–1349

Baker KR (1974) Introduction to sequencing and scheduling. Wiley, Hoboken

Beemsterboer B, Land M, Teunter R (2017) Flexible lot sizing in hybrid make-to-order/make-to-stock production planning. Eur J Oper Res 260(3):1014–1023

Browne J, Dubois D, Rathmill K, Sethi SP, Stecke KE (1984) Classification of flexible manufacturing systems. FMS Mag 2(2):114–117

Burnwal S, Deb S (2013) Scheduling optimization of flexible manufacturing system using cuckoo search-based approach. Int J Adv Manuf Technol 64(5–8):951–959

Candan G, Yazgan HR (2015) Genetic algorithm parameter optimisation using Taguchi method for a flexible manufacturing system scheduling problem. Int J Prod Res 53(3):897–915

Chan FT, Chung S, Chan P (2005) An introduction of dominant genes in genetic algorithm for scheduling of FMS. In: Intelligent control, 2005. Proceedings of the 2005 IEEE international symposium on, mediterrean conference on control and automation, IEEE

Chan F, Chung S, Chan P (2006) Application of genetic algorithms with dominant genes in a distributed scheduling problem in flexible manufacturing systems. Int J Prod Res 44(3):523–543

Chaudhry I, Mahmood S, Shami M (2011) Simultaneous scheduling of machines and automated guided vehicles in flexible manufacturing systems using genetic algorithms. J Cent South Univ Technol 18(5):1473–1486

Cheng B, Yang S, Hu X, Chen B (2012) Minimizing makespan and total completion time for parallel batch processing machines with non-identical job sizes. Appl Math Model 36(7):3161–3167

Geng Z, Yuan J (2018) Scheduling family jobs on an unbounded parallel-batch machine to minimize makespan and maximum flow time. J Ind Manag Optim 14:985–1032

Hsu C-M, Chen K-Y, Chen M-C (2005) Batching orders in warehouses by minimizing travel distance with genetic algorithms. Comput Ind 56(2):169–178

Jain A, Elmaraghy H (1997) Production scheduling/rescheduling in flexible manufacturing. Int J Prod Res 35(1):281–309

Jerald J, Asokan P, Prabaharan G, Saravanan R (2005) Scheduling optimisation of flexible manufacturing systems using particle swarm optimisation algorithm. Int J Adv Manuf Technol 25(9):964–971

Keung K, Ip W, Yuen D (2003) An intelligent hierarchical workstation control model for FMS. J Mater Process Technol 139(1):134–139

Kim YK, Kim JY, Shin KS (2007) An asymmetric multileveled symbiotic evolutionary algorithm for integrated FMS scheduling. J Intell Manuf 18(6):631–645

Liu J, MacCarthy B (1996) The classification of FMS scheduling problems. Int J Prod Res 34(3):647–656

Lova A, Tormos P, Cervantes M, Barber F (2009) An efficient hybrid genetic algorithm for scheduling projects with resource constraints and multiple execution modes. Int J Prod Econ 117(2):302–316

Matin HNZ, Salmasi N, Shahvari O (2017) Makespan minimization in flowshop batch processing problem with different batch compositions on machines. Int J Prod Econ 193:832–844

Nidhiry NM, Saravanan R (2014) Scheduling optimization of FMS using NSGA-II. Adv Ind Eng Manag 3(1):63–72

Raj JA, Ravindran D, Saravanan M, Prabaharan T (2014) Simultaneous scheduling of machines and tools in multimachine flexible manufacturing systems using artificial immune system algorithm. Int J Comput Integr Manuf 27(5):401–414

Reddy BS, Rao CS (2006) A hybrid multi-objective GA for simultaneous scheduling of machines and AGVs in FMS. Int J Adv Manuf Technol 31(5):602–613

Rifai AP, Nguyen H-T, Aoyama H, Dawal SZM, Masruroh NA (2018) Non-dominated sorting biogeography-based optimization for bi-objective reentrant flexible manufacturing system scheduling. Appl Soft Comput 62:187–202

Sadeghian R, Sadeghian MR (2016) A decision support system based on artificial neural network and fuzzy analytic network process for selection of machine tools in a flexible manufacturing system. Int J Adv Manuf Technol 82(9–12):1795–1803

Sankar SS, Ponnanbalam S, Rajendran C (2003) A multiobjective genetic algorithm for scheduling a flexible manufacturing system. Int J Adv Manuf Technol 22(3–4):229–236

Saravanan M, Haq AN (2008) Evaluation of scatter-search approach for scheduling optimization of flexible manufacturing systems. Int J Adv Manuf Technol 38(9–10):978–986

Sivasankaran P, Shahabudeen P (2014) Literature review of assembly line balancing problems. Int J Adv Manuf Technol 73(9–12):1665–1694

Stecke KE (1985) Design, planning, scheduling, and control problems of flexible manufacturing systems. Ann Oper Res 3(1):1–12

Tohidi, H., Jabbari, M.M., (2012). "Decision role in management to increase effectiveness of an organization". Procedia-social and behavioral sciences, 32: 825-828. https://doi.org/10.1016 j.sbspro.2011.12.149

Tohidi, H., & KhedriLiraviasl, K. (2012). Six Sigma Methodology and its Relationship with Lean Manufacturing System. Advances in Environmental Biology, 6(2): 895-906.

Udhayakumar P, Kumanan S (2010) Task scheduling of AGV in FMS using non-traditional optimization techniques. Int J Simul Model 9(1):28–39

Umar UA, Ariffin M, Ismail N, Tang S (2015) Hybrid multiobjective genetic algorithms for integrated dynamic scheduling and routing of jobs and automated-guided vehicle (AGV) in flexible manufacturing systems (FMS) environment. Int J Adv Manuf Technol 81(9–12):2123–2141

Vidyarthi N, Tiwari M (2001) Machine loading problem of FMS: a fuzzy-based heuristic approach. Int J Prod Res 39(5):953–979

Ying K-C, Lin S-W (2018) Minimizing makespan for the distributed hybrid flowshop scheduling problem with multiprocessor tasks. Expert Syst Appl 92:132–141

Zandieh M, Mozaffari E, Gholami M (2010) A robust genetic algorithm for scheduling realistic hybrid flexible flow line problems. J Intell Manuf 21(6):731–743

Zhang Q, Manier H, Manier M-A (2012) A genetic algorithm with tabu search procedure for flexible job shop scheduling with transportation constraints and bounded processing times. Comput Oper Res 39(7):1713–1723