A game-theory approach based on genetic algorithm for flexible job shop scheduling problem

Li Nie¹*, Xiaogang Wang¹ and Fangyu Pan¹
¹Shanghai Polytechnic University, Shanghai, 201209, P. R. of China
*Corresponding author’s e-mail: nieli@sspu.edu.cn

Abstract. In the paper, flexible job shop scheduling problem (FJSP) which joints the objective of maximizing the manufacturer’s efficiency and the objective of maximizing the customer’s delivery satisfaction is considered. An optimization model based on game theory is put forward for the FJSP. Therefore, the problem of FJSP is transferred into a game, in which all jobs and the manufacturer are regarded as players in the game. The players behave with the objective of maximizing their own profits. The manufacturer wants to minimize the makespan of all the jobs, whereas each job wants to minimize the own tardiness. Eventually they gain the equilibrium. In order to solve the game, Nash equilibrium (NE) searching approach based on genetic algorithm (GA) is designed and developed. The efficiency of the proposed approach is validated on several benchmark instances.

1. Introduction

In modern manufacturing enterprises, many flexible manufacturing systems and numerical control machines are introduced in order to improve the production efficiency [1], which makes the scheduling problem in the shop floor of these manufacturing enterprises more complex because these flexible manufacturing systems and numerical control machines usually can process several types of the operations. It means the scheduling decision maker not only needs to choose the routing for each job according with its processing constraints but also needs to sequence the jobs on each machine. The scheduling problem is usually called flexible job shop scheduling problem (FJSP). Since FJSP is more complicate than job shop scheduling problem (JSP) which it is a NP-hard problem, it is necessary to develop effective optimization technology to eliminate the conflicts between the manufacturing facilities and to take advantages of the flexibility of the manufacturing facilities in order to reduce flow-time and work-in-process and to improve production resources utilization.

The scheduling optimizing approaches for FJSP is increasingly attractive to researchers and practitioners [2]. Many researches focused on the single objective FJSP with different kinds of meta-heuristics such as genetic algorithm [3], tabu search [4], swarm optimization algorithm [5] etc., and makespan was usually used as their scheduling objective in most situations while other objectives such as tardiness may also be taken as the scheduling objective in certain situations [6]. Some researchers focused on the multi-objective flexible job shop problem. The difficulty with the multi-objective problem partially lies in the conflicts between different objectives, and a group of Pareto optimal solutions are often adopted to cope with them. For example, Rahmati et al. (2018) [7] developed Pareto envelope-based selection algorithm (PESA) to solve the multi-objective FJSP which joint maintenance and production planning problem. Another method for the multi-objective problem is to find the equilibrium between the different objectives by game theory based approach. Zhang et al.
(2017) [8] considered the FJSP for the manufacturing shop floor to improve energy efficiency and production efficiency. To solve this problem, a dynamic game theory based two-layer scheduling method was developed to reduce makespan, the total workload of machines and energy consumption to achieve multi-objective optimization. Krenczyk and Olender (2015) [9] designed game theory models to solve the FJSP in order to minimize the cycle length and the production cost. Sun et al. (2013) [10] applied non-cooperative game theory with complete information to build new scheduling model for flexible job shop scheduling problem subject to machine breakdown in order to optimize the conflicting objective of robustness and stability simultaneously. Zhan et al. (2012) [11] applied non-cooperative game theory to multi-objective scheduling problem in the automated manufacturing system. The methods mentioned above usually considered the objectives of the efficiency and cost for the manufacturer, such as makespan, machine utilization and so on. There were few researchers to consider the optimization problem for both the manufacturer and customers. In other words, a schedule should consider not only the manufacturer's efficiency and cost objectives, but also the customer’s delivery satisfaction.

In this paper, the scheduling problem considering both the manufacturer's efficiency and customer’s delivery satisfaction is modeled as a FJSP with the objective of minimizing the makespan and maximum tardiness of all jobs simultaneously. An effective game theory approach which hybridizes the genetic algorithm (GA) has been proposed for the FJSP. This paper is organized as follows: Section 2 gives the game model for the FJSP. NE solution approach based on GA is given in Section 3. Section 4 presents experiments and the results. Lastly, the conclusion is drawn in Section 5.

2. Formulating of FJSP

2.1. Description of FJSP

The FJSP can be stated as follows. A number of production jobs come from different orders of different customers are to be processed on a number of machines. The sequence of operations for each job is fixed. Each operation can be operated on any of its alternative machines but visit only one of them exactly once. The processing time of each operation on each of its alternative machine is known and deterministic. It is assumed that:

- Each machine can process at most one operation at any time.
- Each operation can be processed only at one machine at a time.
- Operations of all the jobs must be processed in a given order.
- The setup time of any operation is independent of the schedule, fixed, and included in the corresponding processing time.
- Transport time of each job to transfer from one machine to another is neglected.
- All jobs are ready to start at time zero. And there is no new jobs arrive.
- Machines are available all the time.

In order to discuss the problem in detail, the notations listed below are employed:

- There are \( m \) machines indexed by \( M_j \) \((j = 0, \ldots, m-1)\) and all machines construct the machine set of \( M \).
- There are \( n \) jobs to be processed indexed by \( J_i \) \((i=0, \ldots, n-1)\) and all jobs construct the job set of \( J \).
- Each job \( J_i \) consists of a fixed sequence of operations \( O_{i,l} \) \((i=0, \ldots, n-1; l=0,\ldots,h_i-1)\) and \( h_i \) is the number of operations in job \( J_i \).
- A set of alternative machines for operation \( O_{i,l} \) is presented by \( M_{i,l} \subset M \).
- The processing time for operation \( l \) of job \( i \) on machine \( j \) is denoted as \( p_{i,l,j} \) \((i=0, \ldots, n-1; l=0,\ldots, h_i-1; j=0,\ldots,m-1)\).
- \( f_i \) denotes the finish time of job \( J_i \).
- \( d_i \) denotes the due date of take \( J_i \).
2.2. Game modeling for FJSP
The FJSP addressed in this paper is modelling into an $N+1$-person non-cooperative game with complete information where each job and the manufacturer act as players in the game. They make the decision of the processing strategies (i.e., selection of suitable machines and the sequence entering the system) to achieve their goal individually. The model is expressed as a tuple: $G = (N+1, S, U)$, where $N+1$ denotes number of the players in the game, $S$ denotes the strategy profile of all players, $S = \{s_0, s_1, ..., s_{n-1}, s_n\}$, where $s_i$ denotes the strategy of player $i$, and $U$ denotes the payoff, $U = \{u_0, u_1, ..., u_n\}$, where $u_i$ denotes the payoff of player $i$.

2.2.1. Players. In the job scheduling game, $n$ jobs come from different customers’ orders is regarded as $n$ players, and the manufacturer is regarded as the ($n+1$)th player.

2.2.2. Strategies. Each job has several operations. Each operation should be assigned to a suitable machine in order to optimize the scheduling objectives. $s_i$ represents the strategy set of player $i$. ($i=0,...,n-1$), where $j=0,...,h_i-1$. Assumed $J_i$ has 3 operations and there are 3 alternative machines for each operation, the number of the possible strategies for the job is 27. $s_n = \{l_1, l_2, ..., l_n\}$ means a sequencing of integer of 1 to $n$.

2.2.3. Payoffs. In the scheduling game, each job tries to schedule their operations on correspondent machines to be processed to maximize its own profit. On one hand, in the customers’ position, each customer’s order should be delivered as quickly as possible. On the other hand, in the manufacturer’s position, the manufacturing efficiency should be as high as possible. Therefore, in the game, the payoff of player $i$ ($i = 0,...,n-1$) is defined to be the decreasing function of tardiness of job $i$, and the payoff of play $n$ is defined to be the decreasing function of makespan of all the jobs. It is noticeable that profit of each player is affected not only by its own scheduling strategies but also by the strategies of other players.

$$u_i = u_i(S) = 1/L_i \quad (i = 0,...,n-1)$$

Where $L_i = \max\{0, f_i - d_i\}$

$$u_n = u_n(S) = 1/\max\{f_i \mid i = 0,...,n-1\}$$

2.2.4. Nash equilibrium. Nash equilibrium (NE) is applied as the solution for $N+1$-person non-cooperative game. A NE point is an $N+1$-tuple of strategies, one for each player, such that anyone who deviates from it unilaterally can impossible improve its expected payoff. Therefore, a NE point is repressed as $s^* = \{s^*_0, s^*_1, ..., s^*_n\}$. It is satisfied the equation $u_i(s^*_i, s^*_{-i}) \leq u_i(s_i, s^*_{-i})$, $i = 0,...,n$, where $s^*_{-i} = \{s^*_0, s^*_1, ..., s^*_i, ..., s^*_n\}$.

According to the definition above, scheduling jobs are translated into searching the NE of the job scheduling game. In this paper, GA is incorporated into the approach to search the NE point of this game. The GA mechanisms will be described in detail in the next section.

3. Proposed approach based on GA
Since the NP complexity of the game model, GA is used to search for the NE point of this game. In the paper, the traditional process of GA is employed. Firstly, set the parameters for the algorithm. Then, generate an initial population randomly and evaluate every individual’s fitness. Thirdly, generate a new generation of the population through applying the genetic operators to the current generation of the population. Finally, judge if the termination criteria are satisfied. If the criteria are satisfied the algorithm is terminated, otherwise, iterations continues by transferring to the third step. Various basic genetic operators, such as roulette selection method, two-point crossover, precedence operation crossover, neighbourhood mutation and swapping mutation are used in the paper. The
subsection only focuses on the discussion of encoding and decoding scheme and fitness function because of the limitation of length.

Table 1. Data of a simple FJSP

| Job | Ope | Num | Alternative machines | Processing time |
|-----|-----|-----|----------------------|-----------------|
| J0  | O0,0 | 2   | M0, M2              | 5, 4, -         |
|     | O0,1 | 3   | M0, M1, M2          | 2, 4, 5         |
| J1  | O1,0 | 2   | M0, M1              | 4, 4, -         |
|     | O1,1 | 3   | M0, M1, M2          | 5, 2, 1         |
|     | O1,2 | 2   | M1, M2              | 2, 3, -         |
| J2  | J2,0 | 3   | M0, M1, M2          | 3, 2, 2         |
|     | J2,1 | 2   | M0, M1              | 1, 2, -         |

3.1. Encoding and Decoding Scheme

Encoding means to describe a feasible solution of the scheduling problem into a “chromosome” of the GA population. Decoding is the inverse process of encoding. Each chromosome is designed to compose of $N+1$ parts if the FJSP problem includes $n$ jobs. As for each job, its correspondent part of the chromosome depicts its strategy to assign its operations to alternative machines. As for the manufacturer, its correspondent part of the chromosome determines the permutation of all the jobs. According to the instance in Table 1, the integer string of $\{1,0;1,2,0;2,0;0,1,2\}$ is a chromosome example. Since the problem includes 3 jobs, the chromosome is composed of 4 parts. The first part 1,0 is the strategy of the first job $J_0$. It means that the first operation of the job, i.e. $O_{0,0}$ is assigned to its second alternative machine $M_2$ and second operation $O_{0,1}$ is assigned to its first alternative machine $M_0$. The second part 1,2,0 is the strategy of the second job $J_1$. It means that the first operation of the job, i.e. $O_{1,0}$ is assigned to its second alternative machine $M_1$, the second operation $O_{1,1}$ is assigned to its third alternative machine $M_2$, and the third operation $O_{1,2}$ is assigned to its first alternative machine $M_1$. The third part 2,0 is the strategy of the third job $J_2$. It means that the first operation of the job, i.e. $O_{2,0}$ is assigned to its third alternative machine $M_2$ and second operation $O_{2,1}$ is assigned to its the first alternative machine $M_0$. The last part 0,1,2 is the strategy of the manufacturer. It means the 3 jobs are released to the system in the sequence of 0,1,2.

![Gantt chart](image)

Figure 1. Gantt chart corresponding the chromosome

The Gantt chart corresponding the chromosome is shown in Fig 1. Obviously, according to this encoding and decoding method, it is easy to convert a chromosome into a schedule. And it is also convenient to carry out various genetic operators on the chromosome without generating infeasible offspring.
3.2. Fitness function
In order to evaluate the individual in the population, the fitness function is designed as below:

\[ F = 1 / \max(f_i | i = 0, ..., n - 1) + 1 / \max(L_i | i = 0, ..., n - 1) \] (3)

Where, \( F \) denotes the fitness of an individual. It is found that an individual with a better makespan and better tardiness for each job will be assigned a higher fitness and the individual will survive with a higher probability in the iteration.

In the paper, the parameters for GA are listed below: The size of the population is 100, the time of iterations is 100. The probability of crossover and mutation operation is 0.8 and 0.2, respectively.

4. Experiment and results
Some instances in [18] are used as benchmarks in the paper. The due date of each job is generated randomly according to the equation.

\[ d_i = r_i + kp_i (i = 0, ..., n - 1) \] (4)

Where \( k \) is 0.5 or 1, and \( p_i \) is the minimum processing time of Job \( i \).

Table 2 summaries the results. The column of “Makespan” and “Tardiness” are optimal result obtained by the approaches. From Table 2, it is found that the pressure of due date increases with the value of \( k \) decrease. The proposed approach could make a good balance between makespan and tardiness, which means the interests of customers and the manufacturer gain the equilibrium. For example, the optimal makespans and tardiness of the instance of TFJSP 15*10 are 30 and 23 when \( k = 1 \). Whereas, the optimal makespan and tardiness are 29 and 27 when \( k = 0.5 \). Although the latter is worse than the former in the aspect of tardiness, the latter’s improve the makespan. The Gantt chart of TFJSP 10*10 and TFJSP 15*10 when \( k = 1 \) are shown in Fig 2 and 3.

| Instance     | Makespan | Tardiness | Makespan | Tardiness |
|--------------|----------|-----------|----------|-----------|
| TFJSP 10*7   | 19       | 12        | 19       | 15        |
| TFJSP 10*10  | 14       | 11        | 14       | 12        |
| TFJSP 15*10  | 30       | 23        | 29       | 27        |

5. Conclusion
It is important to assign jobs to the manufacturing facilities of complex manufacturing systems to optimize the objective of maximizing the manufacturer’s efficiency and the objective of maximizing the customer’s delivery satisfaction simultaneously. However, the objectives of two aspects are usually conflicting each other. In the paper, the FJSP with the objective of minimizing the makespan
and maximum tardiness of all jobs is regarded as a game, and concept and method of game theory is employed to establish a mathematical model and a GA-based algorithm is designed to find the NE solution. Jobs from several costumers’ order are players in the game and try to minimize its tardiness. On the other hand, the manufacturer is also regarded as a player in the game and tries to minimize the global makespan. In order to find the NE solution of the scheduling problem, the encoding and decoding scheme is developed and the fitness function is designed base on the principle of GA. The proposed GA-based approach is tested on several instances. It is found that the approach has the ability to schedule jobs with good strategies in both the position of costumers and the manufacturer, which validate its effectiveness. However, in practice there are many other conflict objectives in different position of customers and/or manufacturers. Nash Equilibrium is only one commonly used strategy to solve this kind of problem. Stackelberg strategy is a leader-follower type solution that works well in a situation where one player dominates over the other in the process of decision-making. To use the Stackelberg strategy of GT to develop optimizing approaches for FJSP is one of the future research directions.

Acknowledgments
The authors would like to thank the editor and anonymous referees for their valuable comments. This research is supported by the National Natural Science Foundation of China under Grant No. U1537110 and 51605273 and the National Key R&D Program of China under Grant No. YS2017YFGH000967.

References
[1] Seebacher, G., Winkler, H. (2014) Evaluating flexibility in discrete manufacturing based on performance and efficiency. Int.J.Prod.Econ., 153: 340–351.
[2] Chaudhry, I.A., Khan,A.A. (2016) A research survey: review of flexible job shop scheduling techniques. Int.Trans.Oper.Res., 23: 551–591.
[3] Gu, X., Huang, M., Liang, X. (2017) The improved simulated annealing genetic algorithm for flexible job-shop scheduling problem. In: 6th International Conference on Computer Science and Network Technology, Dalian, pp. 22-27.
[4] Li, X., Gao, L. (2016) An effective hybrid genetic algorithm and tabu search for flexible job shop scheduling problem. Int. J. Prod. Econ., 174: 93-110.
[5] Xu, S., Wu, D., Kong, F., Ji, Z. (2017) Solving Flexible Job-Shop Scheduling Problem by Improved Chicken Swarm Optimization Algorithm. Journal of System Simulation, 29: 1497-1505.
[6] Ning, T., Guo, C., Chen, R., Jin, H. (2016) A novel hybrid method for solving flexible job-shop scheduling problem. Open Cybernetics and Systemics Journal, 10: 13-19.
[7] Rahmati, S.H.A., Ahmadi, A., Karimi, B. (2018) Multi-objective evolutionary simulation based optimization mechanism for a novel stochastic reliability centered maintenance problem. Swarm Evol. Comput., 40: 255-271.
[8] Zhang, Y.F., Wang, J., Liu, Y. (2017) Game theory based real-time multi-objective flexible job shop scheduling considering environmental impact. J. Clean Prod., 167: 665-679.
[9] Krenyczk, D., Olender, M. (2015) Simulation Aided Production Planning and Scheduling Using Game Theory Approach. Applied Mechanics and Materials, 809-810: 1450-1455.
[10] Sun, D.H., He, W., Zheng, L.J., Liao, X.Y. (2013) Scheduling Flexible Job Shop Problem Subject to Machine Breakdown with Game Theory. Int. J. Prod. Res., 52: 3858-3876.
[11] Zheng, X., Zhang, J., Gao, Q. (2012) Application of non-cooperative game theory to multi-objective scheduling problem in the automated manufacturing system. In: International Conference on Automatic Control & Artificial Intelligence IET, Xiamen, pp. 554-557.
[12] Kacem, I., Hammadi, S., Borne, P. (2002) Pareto-optimality approach for flexible job-shop scheduling problems: hybridization of evolutionary algorithms and fuzzy logic. Math. Comput. Simul., 60: 245-276.