Contents lists available at GrowingScience

International Journal of Industrial Engineering Computations

homepage: www.GrowingScience.com/ijiec

Multi-fidelity simulation optimization for production releasing in re-entrant mixed-flow shops

Zhengmin Zhang[b], Zailin Guan[b] and Lei Yue[a*]

[a]*School of Mechanical and Electrical Engineering, Guangzhou University, Guangzhou 510000, China*
[b]*School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan 430000, China*

CHRONICLE

Article history:
Received August 3 2022
Received in Revised Format August 31 2022
Accepted September 15 2022
Available online September, 15 2022

Keywords:
Queueing theory
Multi-fidelity simulation-based optimization
Re-entrant mixed-flow shops
Production release planning

ABSTRACT

This research focuses on production releasing and routing allocation problems in re-entrant mixed-flow shops. Since re-entrant mixed flow shops are complex and dynamic, many studies evaluate release plans by developing discrete event simulation models and selecting the optimal solution according to the estimation results. However, a high-accurate discrete event simulation model requires a lot of computation time. In this research, we develop an effective multi-fidelity optimization method to address product release planning problems for re-entrant mixed-flow shops. The proposed method combines the advantages of rapid evaluation of analytical models and accurate evaluation of simulation models. It conducts iterative optimization using a low-fidelity mathematical estimation model to find good solutions and searches for the optimal solution via a high-fidelity simulation estimation model. Computational results of large-scale production releasing and routing allocation problems illustrate that the proposed approach is good at addressing large-scale problems in re-entrant mixed-flow shops.

1. Introduction

The production release planning determines manufacturing efficiency and plays an essential role in re-entrant mixed-flow shops (Chen et al., (2015). Re-entrant mixed-flow shops have the characteristics of mass manufacturing, re-entrant flows, long processing routes, and simultaneous processing for multiple types of products. In some production lines, multiple processing routes may be available for one type of product. Thus, it is important to provide an effective releasing and routing allocation plan for production lines. Generally, the production releasing and routing allocation problem seeks the best solution by determining the release rate for each type of product and choosing a specific processing route for products with multiple processing routes. Due to the particularity of re-entrant mixed-flow shops, production planning and scheduling methods for traditional workshops and single-product manufacturing systems with re-entrant flows are difficult to apply to the studied systems. Some basic approaches, such as mathematical programming (Asmundsson et al., 2006; Kacar et al., 2016; Kacar et al., 2013; Leachman, 2001), simulation optimization (Zhang et al., 2020; Hong & Chien, 2020; Thuerer et al., 2019), and analytical modeling (Chung & Lai, 2006; Schneckenreither et al., 2021), have been proposed for production optimization problems. Simulation optimization methods search for the best solution by evaluating candidate solutions using simulation models and selecting the best one according to objective values (Chen & Lee, 2010; Lee et al., 2010; Xu et al., 2010). While these methods provide accurate estimates of solutions, they require high computational costs because of their high-precision modeling capability (Asmundsson et al., 2009; Fowler & Mönch, 2017). Therefore, many references only use simulation models to evaluate the solution obtained by mathematical models (Bang & Kim, 2010; Kopp et al., 2019; Rashmi & Mathirajan, 2018; Ziarnetzky, (2015). Some approaches (Missbauer, 2020; Wolosewicz et al., 2015; Albey & Bilge, 2011; Puergstaller & Missbauer, 2011; Kim & Lee, 2016; Kim & Kim, 2001) use the iterative simulation and linear programming method to solve release planning problems. However, this method is hard to converge if the convergence process is not stable or the initial solution is ineffective. Therefore, we propose our research question: *How to address production releasing and*

* Corresponding author
E-mail: leileiyok@gzhu.edu.cn (L. Yue)
ISSN 1923-2934 (Online) - ISSN 1923-2926 (Print)
2023 Growing Science Ltd.
doi: 10.5267/j.ijiec.2022.9.004

routing allocation problems in re-entrant mixed-flow shops by simulation optimization?

To address this question, we design a multi-fidelity simulation optimization framework for production releasing and routing allocation problems. We develop the proposed framework based on the multi-fidelity optimization with ordinal transformation and optimal sampling (MO2TOS) method (Xu et al., 2016). MO2TOS constructs a low-precision model (We can also call it the 'low-fidelity model', which means a model with less accuracy) and a high-precision model (the 'high-fidelity model'). We develop the high-fidelity model using the discrete-event simulation technology based on practical production scenarios. Thus, it has high accuracy but results in a high computational burden. We develop the low-fidelity model as a mathematical expression, which is established based on the open queueing network. The low-fidelity model ignores unnecessary details of manufacturing scenarios to guarantee less computing time. In the proposed MO2TOS, we use a low-fidelity model with a multiple population genetic algorithm (MPGA) to evaluate available solutions and select plenty of good solutions from the whole solution space. Then, we select the top solutions from these good solutions, using the ordinal transformation and optimal sampling strategies, and develop a high-fidelity simulation model to evaluate these solutions and choose the best solution.

The main contributions of our research are as follows: 1. We establish an effective simulation optimization framework to solve production releasing and routing allocation problems in re-entrant mixed-flow shops. Compared with MO2TOS (Xu et al., 2016), the proposed method saves 90% of the computing time when solving large-scale problems. 2. Based on the open queueing network, we present a mathematical expression that can effectively estimate cycle times for release plans.

2. Related works

2.1 Multi-fidelity simulation optimization

To perform efficient simulation optimization, some scholars present multi-fidelity simulation optimization when models with several fidelity levels are available (Xu et al., 2014a; Chen et al., 2015; Chiu & Lin, 2020). A standardized multi-fidelity simulation optimization method (Xu et al., (2016) first establishes a low-fidelity model and a high-fidelity simulation model based on real manufacturing scenarios. It screens all the candidate solutions through a low-fidelity model and selects a fixed number of high-quality solutions to build a solution set. This method evaluates this solution set by a high-fidelity model and selects the best decision. Recently, researchers have shown an increased interest in optimizing the efficiency of multi-fidelity simulation optimization. For example, Chiu et al. (2016) combine multi-fidelity models with genetic algorithms to develop an improved multi-fidelity optimization framework and use it to improve the efficiency of large-scale optimization problems. Chen et al. (2015) improve the accuracy of estimation by developing an effective learning algorithm. Xu et al. (2016) propose a multi-fidelity optimization method named MO2TOS. The promising performance of MO2TOS has been testified and confirmed recently (Zhang et al., 2020; Song et al., 2019; Zhang et al., 2020). MO2TOS shortens the computing time of finding the optimal solution since it combines the advantages of the low-fidelity models and the high-fidelity models (Zhang et al., 2020). Li et al. (2015) design a multi-objective MO2TOS method for deterministic optimization problems. Qiu et al. (2016) present the MO2TOS-based multi-fidelity simulation optimization approach to optimize patient flow in health care systems. The application of MO2TOS and other multi-fidelity optimization methods in discrete production systems is still in its infancy (Shao et al., 2019). Zhang et al. (2020) develop an improved multi-fidelity simulation optimization method to address production planning problems on shaft parts shop floors. Similarly, Zhang et al. (2020) present a multi-fidelity simulation optimization method in wafer manufacturing systems. Experiment results show that this method improves the computational efficiency of simulation-based production planning.

2.2 Lead time estimates

Generally, MO2TOS comprises a low-fidelity model and a high-fidelity model. High-fidelity models are usually designed by discrete-event simulation technology, while low-fidelity models are established by mathematical expression, simulation, analytical modeling, or other approaches. A low-fidelity model with less accuracy may produce a large estimation deviation in solution space searching, which affects the selection of high-quality solutions. Thus, it is essential to provide a proper modeling method for low-fidelity estimation. In re-entrant mixed-flow production systems, manufacturing cycle time estimation is directly related to the rationality of the release planning model. The average cycle time of the overall process is usually long and variable because of the difference in product processing routes and release plans. Moreover, the flow of material may face temporary queues or substantial blockages during manufacturing, which makes cycle time estimation more difficult. There has been an increasing interest in estimating cycle time correctly (see e.g., Mather & Plossl, 1978; Billington et al., 1983; Selcuk et al., 2006; Milne et al., 2015). Generally, we consider the estimated cycle time as the lead time for releasing (Kacar et al., 2016). We summarise some basic estimation methods. In most literature, several basic approaches, for instance, mathematical programming and simulation models, have been developed to obtain lead times. The most common approach is to provide a mathematical representation for lead times. In the last few decades, exogenous integer and exogenous non-integer lead times have been widely used (Yanıkoğlu et al., 2017). For instance, some production planning models (e.g., Material Requirements Planning) consider lead times as static parameters (Baker, 1993; Orlicky, 1975; Vollmann et al., 1988; Hackman & Leachman, 1989). This simplification of the problem may lead to underestimating or overestimating lead times. Non-integer lead time-based models also have been extensively studied and implemented by Leachman et al. (1994, 1996). Some articles describe the use of non-integer lead times in linear programming models (Leachman, 2001; Leachman, 1993;

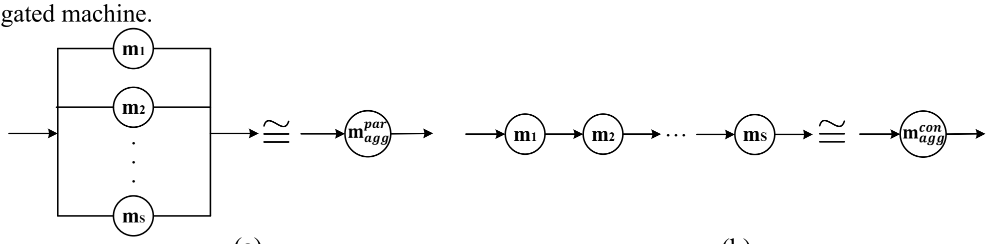
Leachman, 1996). Kacar et al. (2016) find that the performance of the models with non-integer lead times is substantially better than those with integer lead times.

On the other hand, an effective mathematical modeling approach for production planning named clearing functions is suggested (Yanıkoğlu et al., 2017). Clearing functions describe the relationship between the expected WIP and output (Asmundsson et al., 2006; Albey et al., 2014; Kacar et al., 2012; Kacar et al., 2013; Kacar et al., 2016). Kacar et al. (2013) evaluate the performance of the production planning model with clearing functions by simulation. Results indicate that the proposed model yields substantial improvements in profit over conventional linear programming models even in large-scale problems. Unfortunately, the practical use of clearing functions has been hampered by the lack of effective methods for estimating them (Yanıkoğlu et al., 2017). Currently, simulation estimation, least-squares regression, and percentile fit (Asmundsson et al., 2009) are usually employed to estimate lead times. Asmundsson et al. (2006) use a clearing function model to capture the nonlinear relationship between workload and throughput. They develop a simulation study of a production planning model to reflect the nonlinear relationship between resource utilization and lead time. Missbauer (2011) proposes an alternative transient clearing function and derives a procedure for its parameterization. Chen et al. (2015) reveal that multi-dimensional clearing functions can better predict system performance in the presence of mix-dependent capacity losses. Some approaches (Missbauer, 2020; Wolosewicz et al., 2015; Albey & Bilge, 2011; Puergstaller & Missbauer, 2011; Kim & Lee, 2016; Kim & Kim, 2001) use iterative simulation and linear programming to calculate lead times. Through this iterative procedure of simulation and optimization, some important variables (such as manufacturing lead times and the WIP level) can be updated for the synchronization of planning and scheduling decisions. Kim and Kim (2001) develop this iteration procedure and simultaneously update lead times and available capacities that are assumed in the mathematical model. Experiment results show that this iteration procedure has a great performance in terms of manufacturing lead time, demand satisfaction, and feasibility. However, these methods may be hard to converge if the convergence process is not stable or the initial solution is ineffective. With the development of machine learning, several papers present release planning models that use artificial neural networks (Philipoom et al., 1994; Philipoom et al., 1997; Hsu & Sha, 2004; Patil, 2008; Wang & Ting, 2008; Schneckenreither et al., 2021). For instance, Patil (2008) uses a hybrid method that combines machine learning with genetic algorithms to predict cycle times to set due dates for job shops. Similarly, Chang et al. (2008) combine a neural network and fuzzy logic to forecast cycle times for semiconductor factories. Schneckenreither et al. (2021) propose a flow time estimation procedure to set lead times dynamically using an artificial neural network. Considering the requirements of computational speed for low-fidelity models, mathematical approximation models are more suitable for low-fidelity estimation. Thus, we develop a mathematical model and use the queueing network to estimate cycle times. We further verify the estimation accuracy of the mathematical model in different production releasing and routing allocation problems.

3. Problem formulation based on structural modeling

3.1 Problem introduction and simplification

We consider the following scenario: Different types of products are produced in a re-entrant mixed-flow production line. Some products may have several alternative processing routes. Processing routes may have re-entrant flows and the demand for products is known. We need to calculate the optimal release rate and the routing allocation plan for each type of product to meet their demand requirements. Since the processing routes are complicated in this research, we consider some aggregation operations to simplify the non-critical operations of the processing routings. According to Li and Meerkov (2009), some production systems can be reduced to standard serial production lines. They believe this simplification is beneficial to the analysis of the production line and refer to this process as structural modeling. Although the studied mix-flow shop can hardly be transferred to a standard serial production line, the simplification method can also be used in our research. For instance, the parallel machines (as shown in Fig 1. (a)) or the consecutive dependent machines (as shown in Fig 1. (b)) can be considered as an aggregated machine.



(a) (b)

Fig. 1. Aggregating parallel machines and consecutive machines

For parallel machines, we assume that the processing time of each machine m_i (i=1,2,…,S) is $\tau_i(i = 1,2,\ldots,S)$. Machines operate normally all the time. We can calculate τ_{agg}^{par} for m_{agg}^{par} according to the following equation:

$$\tau_{agg}^{par} = 1/(\textstyle\sum_{i=1}^{S}\frac{1}{\tau_i}). \tag{1}$$

For consecutive machines, we can calculate τ_{agg}^{con} for m_{agg}^{con} according to the following equation:

$$\tau_{agg}^{con} = \max_{i} \tau_i. \tag{2}$$

Inspired by their research, we aggregate parallel machines of the studied production line into one machine before modeling. We use this operation to simplify the modeling of the production line.

3.2 *Problem formulation*

In this problem, we want to calculate the optimal release rate and choose the optimal processing route for each type of product. We consider a scenario where demand exceeds supply. Under this scenario, we need to meet demands for different types of products while maintaining a stable mix of products to increase throughput. The objectives of the problem are to minimize the makespan and the maximum WIP of the production line while satisfying the demand requirements of customers. We develop a mathematical model for this problem. The relevant parameters are as follows:

Indices:

k: The product type index

r: The processing route index

l: The operation index

m: The machine index

p: The input period index, one period equals one-time unit

Parameters:

K: Total number of production types

R_k: Number of processing routes of product *k*

$L_{k,r}$: Number of operations for product *k*

M: Number of machines

d_k: Demand for product *k*

$M_{k,r,l}$: Machine for the l-th operation of $L_{k,r}$

Variables:

$x_{k,r}^{p}$: The external release material quantity for product *k* of route *r* in period *p*

$y_{k,r}^{p}$: The output for product *k* of route *r* in period *p*

$F(k,r,p,l)$: The complete time for the l-th operation of product *k* of route *r* released in period *p*

$WIP(k,r,p,m)$: Cumulative WIP for product *k* of route *r* at machine *m* in period *p*. $WIP(k,p,m) = \sum_{r=1}^{R_k} WIP(k,r,p,m)$

The objective function is as follows:

$$min(w1 \cdot makespan + w2 \cdot work\_in\_process).$$

where

$$makespan = max\, F\big(k,r,p,L_{k,r}\big), \forall k,r,p,m, \tag{3}$$

and

$$work\_in\_process = max\Big(\frac{\sum_{p=1}^{\lceil F(k,r,p,l)\rceil} \sum_{k=1}^{K} \sum_{r=1}^{R_k} WIP(k,r,p,m)}{\lceil F(k,r,p,l)\rceil}\Big), \forall m. \tag{4}$$

subject to

$$\sum_{p=1}^{[F(k,r,p,l)]} \sum_{r=1}^{R_k} y_{k,r}^{p} \geq d_k, \forall k, m. \tag{5}$$

$$x_{k,r}^{p}, y_{k,r}^{p}, F(k,r,p,l), WIP(k,r,p,m) \geq 0, \forall k,r,p,m. \tag{6}$$

In this mathematical model, the decision variable $x_{k,r}^{p}$ means the release rate for product k of route r in period p. In the proposed mathematical model, $y_{k,r}^{p}$, $F(k,r,p,l)$, and $WIP(k,p,m)$ are linked with $x_{k,r}^{p}$. Thus, we can obtain $y_{k,r}^{p}$, $F(k,r,p,l)$, and $WIP(k,p,m)$ once we obtain the relationship between $x_{k,r}^{p}$ and other variables.

4. The low-fidelity model based on queueing theory

In this section, we present a low-fidelity approximate method based on queueing theory, to obtain the relationship between $x_{k,r}^{p}$ and other variables. Once we know $x_{k,r}^{p}$, we can estimate the distribution of the product arrival. According to queueing theory, we can further calculate the waiting time and the queue length. $F(k,r,p,l)$, $WIP(k,p,m)$, and cycle time could be calculated accordingly. Moreover, we can obtain $y_{k,r}^{p}$ according to $x_{k,r}^{p}$ and cycle time. Here is a list of the notations:

τ_m:	The unit processing time for m
ρ_m^p:	The utilization of machine m
$P_{k,r,l}$:	The processing time of the l-th operation for product k of route r
$W(k,r,p,l)$:	The waiting time for the l-th operation for product k of route r released in period p
λ_m^p:	The expected arrival rate for all products at machine m in period p
$\lambda_m^p(k,r)$:	The expected arrival rate for product k at machine m in period p
$\lambda_{0,m}^p(k,r)$:	The external arrival rate for product k at machine m in period p
$\lambda_{n,m}^p(k,r)$:	The arrival rate for product k from machine n to machine m in period p

$\lambda_m^p(k,r)$ comprises the external arrival flows and the re-entrant flows. Thus, $\lambda_m^p(k,r)$ can be calculated as:

$$\lambda_{0,m}^{p}(k,r) = x_{k,r}^{p} \cdot [M_{k,r,1} = m], \tag{7}$$

and

$$\lambda_{n,m}^{p}(k,r) = \sum_{n=1}^{M} (\sum_{l=1}^{L_{k,r}-1} (x_{k,r}^{p} \cdot [M_{k,r,l} = n, M_{k,r,l+1} = m])), n \neq m. \tag{8}$$

Therefore, $\lambda_m^p(k,r)$ is

$$\lambda_m^p(k,r) = \lambda_{0,m}^p(k,r) + \lambda_{n,m}^p(k,r), \forall m,k,r,p. \tag{9}$$

λ_m^p is then given by

$$\lambda_m^p = \sum_{k=1}^{K} \sum_{r=1}^{R_k} \lambda_m^p(k,r), \forall m,p. \tag{10}$$

Therefore, the utilization of machine m is:

$$\rho_m^p = \tau_m \cdot \lambda_m^p, \forall m,p. \tag{11}$$

We refer to the approximation method (Shown in Eq. (12) and Eq. (13)) of the GI/G/1 server (Whitt, (1983) to calculate the expected waiting time for each product in a machine.

$$EW = \frac{\tau\rho(C_a^2 + C_s^2)g}{2(1-\rho)}, \tag{12}$$

where g is defined as:

$$g(\rho, C_a^2, C_s^2) = \begin{cases} \exp\left[-\frac{2(1-\rho)}{3\rho}\frac{(1-C_a^2)^2}{C_a^2+C_s^2}\right], C_a^2 < 1 \\ 1, \quad C_a^2 \geq 1 \end{cases}. \tag{13}$$

In Eq. (12) and Eq. (13), τ is the service time and ρ is the utilization of this server. C_a^2 and C_s^2 represent the squared coefficient of variation of the arrival interval distribution and the service-time distribution. Therefore, we $W(k,r,p,l)$ can be represented by the expected waiting time EW. Accordingly, $F(k,r,p,l)$ is

$$F(k,r,p,l) = P_{k,r,l} + W(k,r,p,l). \tag{14}$$

According to Little's law,

$$WIP(k,p,j) = W(k,r,p,l) \cdot \lambda_m^p, \tag{15}$$

where $F\left(k,r,p,L_{k,r}\right)$ represents the makespan of product k released in period p. $y_{k,r}^p$ can be estimated as

$$y_{k,r}^p = \sum_{z=1}^{p} \alpha_{k,r}^z \eta_{k,r}^z, \tag{16}$$

where

$$\alpha_{k,r}^z = \begin{cases} x_{k,r}^z, z + F\left(k,r,z,L_{k,r}\right) - 1 < p \le z + F\left(k,r,z,L_{k,r}\right) + 1 \\ \quad 0 \ , otherwise \end{cases}, \tag{17}$$

and

$$\eta_{k,r}^z = \begin{cases} \left\lceil F\left(k,r,z,L_{k,r}\right)\right\rceil - F\left(k,r,z,L_{k,r}\right), \left\lceil F\left(k,r,z,L_{k,r}\right) - 1\right\rceil = p \\ F\left(k,r,z,L_{k,r}\right) - \left\lfloor F\left(k,r,z,L_{k,r}\right)\right\rfloor, \left\lceil F\left(k,r,z,L_{k,r}\right) - 1\right\rceil = p - 1' \end{cases} \tag{18}$$

5. The multi-fidelity optimization approach

5.1 The framework of the basic MO2TOS

We can use the approximation method proposed in Section 4 as a low-fidelity model to estimate the objective value for each solution and select good solutions in the solution space. However, this approximation model is hard to find the real optimal solution because it is a low-precision model. Therefore, we consider combining a high-fidelity model to find the optimal solution to the problem. Discrete-event simulation is usually used to establish high-fidelity simulation models for combinatorial optimization problems. Recently, researchers have studied the combination of models with different precision levels, which is called multi-fidelity modeling (Xu et al., 2014; Lester et al., 2014; Sébastien & Mathieu, 2011). Multi-fidelity optimization methods combine the advantages of low- and high-fidelity models, which can reduce computing costs and shorten the computing time of finding the optimal solution. This research refers to a multi-fidelity simulation optimization framework named MO2TOS, to solve production releasing and routing allocation problems. MO2TOS has been applied to some planning problems and proved to be effective (Zhang et al., 2020; Zhang et al., 2021). Fig. 2 depicts the basic optimization framework of MO2TOS. The optimization framework of MO2TOS has a low-fidelity mathematical model and a high-fidelity simulation model. In this optimization framework, we first evaluate all feasible solutions by the low-fidelity model and obtain the objective values for solutions. Then, we use the ordinal transformation strategy (Xu et al., (2016) of MO2TOS to transform the original solution space into an ordinal space according to the estimation results. We further use the optimal sampling strategy (Xu et al., 2016) of MO2TOS to sample the transformed space and select good solutions to form a solution set. Finally, we estimate the selected solution set through the high-fidelity model and choose the optimal solution.

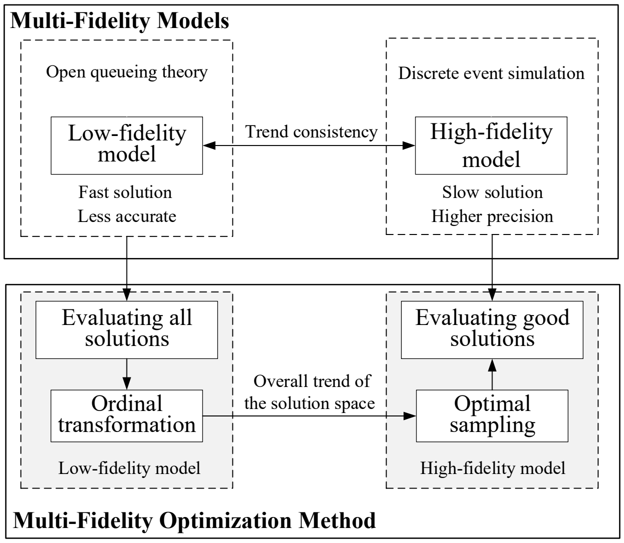


Fig. 2. The basic optimization framework of MO2TOS

5.2 The framework of MO2TOS with MPGA

The solution space for large-scale release planning problems in realistic re-entrant mixed-flow production lines may be more than several million. The computational costs and runtime are not negligible even if we use low-fidelity models to evaluate the entire solution space. Therefore, we combine MO2TOS with multiple population evolutionary algorithms (MPGA) to accelerate the evaluation efficiency. The steps of MO2TOS with MPGA are exhibited in Fig. 3. The framework of MO2TOS with MPGA conducts three phases (*Solution space searching using multiple population genetic algorithms*, *Ordinal transformation, and Optimal sampling)* sequentially. We list the definition of parameters that we used in the mathematical model.

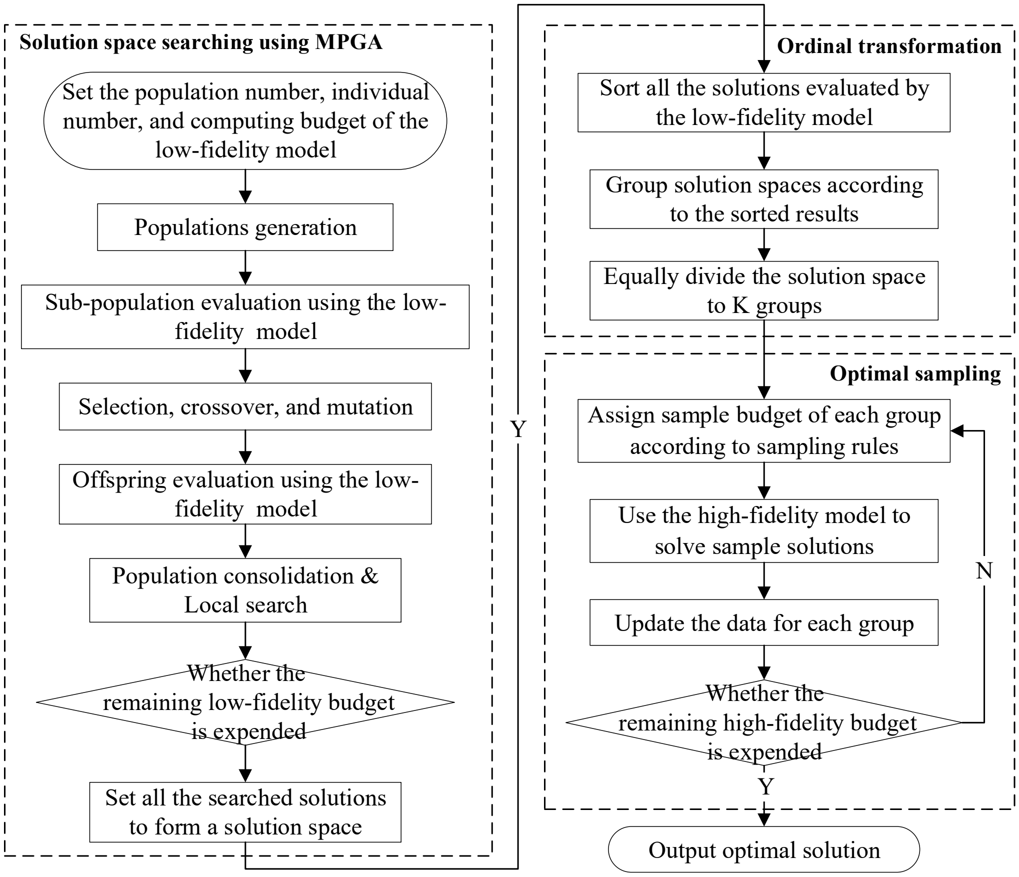

Solution space searching using MPGA
Set the population number, individual number, and computing budget of the low-fidelity model
Populations generation
Sub-population evaluation using the low-fidelity model
Selection, crossover, and mutation
Offspring evaluation using the low-fidelity model
Population consolidation & Local search
Whether the remaining low-fidelity budget is expended
Set all the searched solutions to form a solution space
Y
Ordinal transformation
Sort all the solutions evaluated by the low-fidelity model
Group solution spaces according to the sorted results
Equally divide the solution space to K groups
Optimal sampling
Assign sample budget of each group according to sampling rules
Use the high-fidelity model to solve sample solutions
Update the data for each group
Whether the remaining high-fidelity budget is expended
N
Y
Output optimal solution

Fig. 3. MO2TOS with MPGA

L_{max}:	Low-fidelity evaluation budget
H_{max}:	High-fidelity evaluation budget
P:	Population number in MPGA, $p = 1,2, \dots, P$
Y:	The individual quantity of a population, $y = 1,2, \dots, Y$
p_c:	Probability of the crossover operation
p_m:	Probability of the mutation operation
L:	The local search times for each individual
r:	Index of iteration
x_{py}^{r}:	The y-th solution in population p in r-th iteration process
$l(x_{py}^{r})$:	The result of the low-fidelity model evaluated at solution x_{py}^{r}
I:	Number of solutions in the solution set, $\mathrm{i} = 1,2, \dots, I$
x_i:	The i-th solution in the solution space
x_{OTi}:	The i-th solution after ordinal transformation
K:	Number of groups, $k = 1,2, \dots, K$
N_k^r:	The new allocated budget to the group k in r-th iteration process
N_0:	Sample budget for initial allocation, $KN_0 << N_{max}$
μ_k:	Sample mean of group k

σ_k^2: Sample variance of group k

N_k: The allocated budget to the k-th group

- Solution space searching using MPGA

We design MPGA to accelerate the searching efficiency of the solution space for large-scale optimization problems. Generally, the solution space of a production releasing problem is multi-peaks. Therefore, we develop the framework of the MPGA for global searching. Based on this framework, we present some improved operations for the problem. We use Matlab R2017a to code the algorithms based on the standard toolbox gatbx of MATLAB. The evolution operations and the population consolidation operation adopt the standard operation algorithm in gatbx. The steps to search solution space using the MPGA are as follows:

1) Inputting population number P, the individual quantity of population Y, and the low-fidelity evaluation budget L_{max}.
2) Let index of iteration $r = 0$. Generating initial populations $\{x_{p1}^r, \dots, x_{py}^r, \dots, x_{pY}^r\}$ randomly.
3) Calculating the low-fidelity estimates $\{l(x_{p1}^r), \dots, l(x_{py}^r), \dots, l(x_{pY}^r)\}$ for populations using the proposed queueing theory-based low-fidelity model.
4) Adopting selection, crossover (probability of crossover p_c), and mutation (probability of crossover p_m) operations in the toolbox *gatbx* according to the low-fidelity estimates $\{l(x_{p1}^r), \dots, l(x_{py}^r), \dots, l(x_{pY}^r)\}$. generating subpopulations $\{x_{p1}^r{}', \dots, x_{py}^r{}', \dots, x_{pY}^r{}'\}$.
5) Calculating the low-fidelity estimates $\{l(x_{p1}^r{}'), \dots, l(x_{py}^r{}'), \dots, l(x_{pY}^r{}')\}$ for subpopulations using the proposed queueing theory-based low-fidelity model.
6) For each population, merging the original population $\{x_{p1}^r, \dots, x_{py}^r, \dots, x_{pY}^r\}$ with the subpopulation $\{x_{p1}^r{}', \dots, x_{py}^r{}', \dots, x_{pY}^r{}'\}$ to form a new population $\{x_{p1}^{r+1}, \dots, x_{py}^{r+1}, \dots, x_{pY}^{r+1}\}$ using the population consolidation operation of the toolbox *gatbx*. The population number P and the individual quantity of population Y are fixed. Let $r = r + 1$.
7) Each individual in the new population performs a local search for L times.
8) Recording and adding the searched individuals above in the solution space. These individuals form the solution space of feasible solutions.
9) One individual represents a low-fidelity evaluation budget. If the solution space budget is used up, outputting the solution space $\{x_1, \dots, x_i, \dots, x_I\}$; otherwise, back to step 4).

- Ordinal transformation strategy

Sorting all the solutions in solution space $\{x_1, \dots, x_i, \dots, x_I\}$ from the best to the worst according to their estimation results. The sorted solution space is recorded as $\{x_{OT1}, \dots, x_{OTi}, \dots, x_{OTI}\}$. Dividing the sorted solution space into K equal groups, that is, the scheme number of each group is the same.

- Optimal sampling strategy

We employ an optimal sampling strategy based on optimal computational budget allocation (OCBA) (Xu et al., (2014; Qiu et al., (2016). We provide the steps for optimal sampling strategy:

1) Inputting the high-fidelity simulation budget H_{max}.
2) Let the index of iteration $r = 0$. Allocating $N_k^r = N0$ samples for group *k*.
3) Let $r = r + 1$. Defining a total incremental sample size Δ, and calculating the budget value N_k^r for group *k* according to Lemma 4.2.
4) If the high-fidelity simulation budget is used up ($\sum_{r=0} \sum_{k=1}^{K} N_k^r \geq H_{max}$), outputting the current allocation values and continue step 5); otherwise, back to step 3).
5) Sorting the solutions in each group and selecting top solutions based on the high-fidelity budget N_k allocated to group *k*.

Lemma 4.2 (Xu et al., 2014) (page 7): *If* N_k^{r+1} *is the allocated high-fidelity evaluation budget for group k,* $k = 1, \dots, K$. $\delta_{b,k}$ *is the average difference (for example, the group distance) between group b and group k. Then, we have*

$$\frac{N_l^{r+1}}{N_k^{r+1}} = \left(\frac{\delta_{b,k}/\sigma_k}{\delta_{b,l}/\sigma_l}\right)^2, \textit{ when } \mathrm{k} \neq \mathrm{l} \neq \mathrm{b}, \tag{17}$$

and

$$N_b^{r+1} = \sigma_b \sqrt{\sum_{l=1, l \neq b}^{k} \frac{N_l^2}{\sigma_l^2}}. \tag{18}$$

From formula (17), the value of high-fidelity evaluation budgets for group k decreases if the average distance between group k and the best group b increases. In addition, more high-fidelity budgets are allocated for group k if ${\sigma_k}^2$ is larger.

6. Case study and analysis

We develop some typical re-entrant mixed-flow production lines to evaluate MO2TOS with MPGA. First, we randomly generate two small-scale production releasing and routing allocation cases. We evaluate all the available solutions for the cases using the low-fidelity model and the high-fidelity model, respectively. We compare the estimates obtained by the low-fidelity model with those obtained by the high-fidelity model. We find the output trends of the two fidelity models are of obvious consistency. Therefore, we conclude that the proposed low-fidelity model is feasible. Then, we generate a large-scale case and use MO2TOS with MPGA to solve the large-scale optimization problem. Computational results show that MO2TOS with MPGA provides the optimal solution for large-scale problems and saves about 90% of computing time than MO2TOS.

6.1 Low-fidelity model formulation and efficiency analysis

We develop two small-scale cases (each case has about 7000 alternative solutions) to examine the efficiency of the proposed low-fidelity estimation model. Case 1 and Case 2 have short processing flows and long processing flows, respectively. We first present Case 1 to evaluate the efficiency of the proposed low-fidelity estimation model. There are 10 machines on the production line, and we ignore the transfer time between machines. Fig. 4 describes the processing flows for products in Case 1. Table 1 gives the distributional parameters of processing times for all machines. All processing times follow a lognormal distribution. The demands for Product 1, Product 2, and Product 3 in Case 1 are 800, 1600, and 1400, respectively. We set a lower bound and an upper bound for the release rate for each type of product. In this scenario, we set the lower and upper bound as 4 lots/hour and 10 lots/hour, respectively.

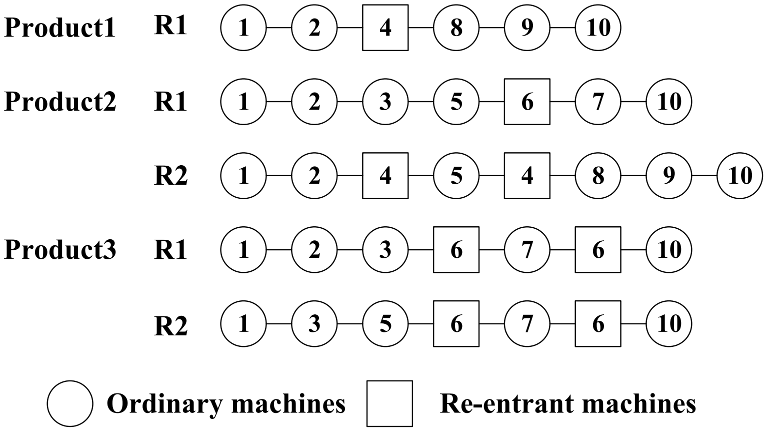


Fig. 4. Processing flow for Case 1

Table 1
The distributional parameters of processing times for Case 1.

Machine	Mean	Std.Dev	Machine	Mean	Std.Dev
1	90	7	6	100	2
2	115	16	7	140	8
3	155	4	8	165	4
4	130	4	9	170	5
5	130	2	10	85	2

We establish a low-fidelity estimation model for these cases according to the low-fidelity approximation method in Section 4. C_s^2 is known because the mean and variance of processing time are given in advance. Since release rates of products are stable in all cases, the mean and variance for the renewal interval are fixed for the first machine. However, because of the randomness of processing times and the re-entrant characteristic, the actual mean value for the renewal interval may change and the actual variance may not be equal to 0 in the following machines. Thus, C_a^2 is not a fixed value for each machine. To calculate C_a^2, we conduct a large number of sample experiments and intend to estimate $C_{a_{sample}}^2$ using massive simulation experiment results. We randomly generate 70 groups of experiments and run these sample experiments using the high-fidelity simulation model. We calculate the mean and variance of the renewal interval in each machine under each plan and take the average values as the final sample results to calculate $C_{a_{sample}}^2$.

We evaluate all the feasible solutions by the high-fidelity simulation model, the low-fidelity model with $C_a^2 = 0$, and the low-fidelity model with $C_a^2 = C_{a\,sample}^2$, respectively, to estimate the efficiency of the proposed low-fidelity model and select better values of C_a^2. Fig. 5 plots the solution space of the three models for Case 1. For a clear display, we add 100 to all the results of the low-fidelity model with $C_a^2 = 0$ and add 200 to all the results of the low-fidelity model with $C_a^2 = C_{a\,sample}^2$. Fig. 6 shows the output of the solution space after ordinal transformation. As illustrated in Fig. 6, we find the consistency of the output evaluated by the three models is obvious, which means the proposed low-fidelity mathematical method is feasible. Overall, the output of the solution space approximated by the low-fidelity model with C_a^2 equals 0 has better consistency than that of the high-fidelity simulation model. Thus, we approximate C_a^2 equals 0, that is, we approximate the mean value of the renewal interval equals the release rate and the variance value of the renewal interval equals 0.

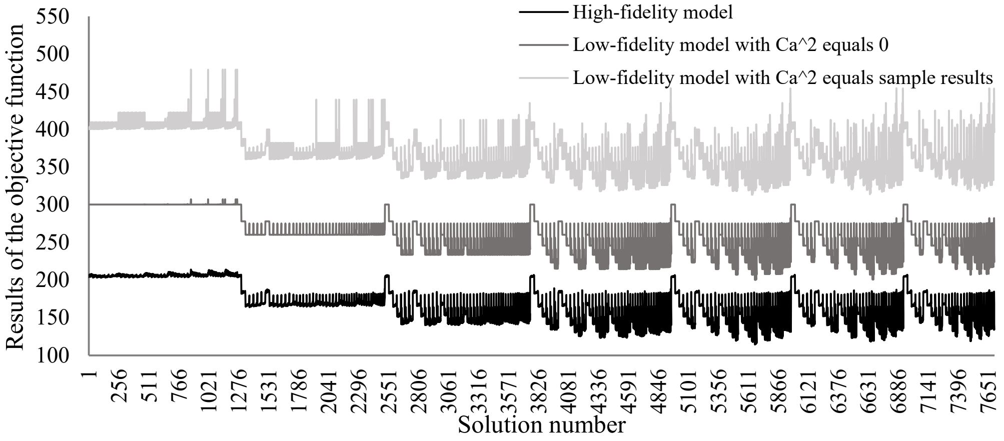


Fig. 5. The output of the solution space for Case 1

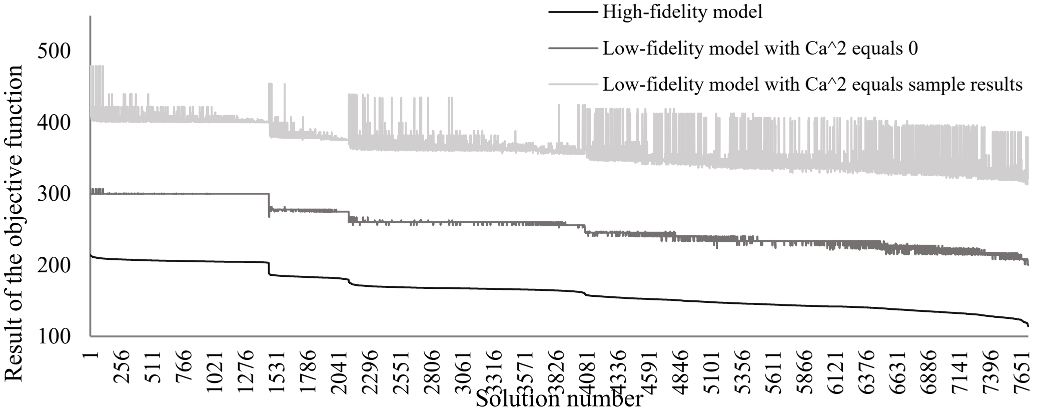


Fig. 6. The output of the solution space for Case 1 after ordinal transformation

From Case 1, the consistency of the trends in Fig. 6 proves that the low-fidelity model is feasible. To confirm that the model is also effective in more complex problems, we extend the processing routes and increase the number of re-entrant machines based on Case 1 and present Case 2. There are 28 machines on the production line. Fig. 7 describes the processing flows for products in Case 2. Table 2 gives the distributional parameters of processing times for all the machines. All processing times follow a lognormal distribution. The demands for Product 1, Product 2, and Product 3 are 800, 1600, and 1400, respectively.

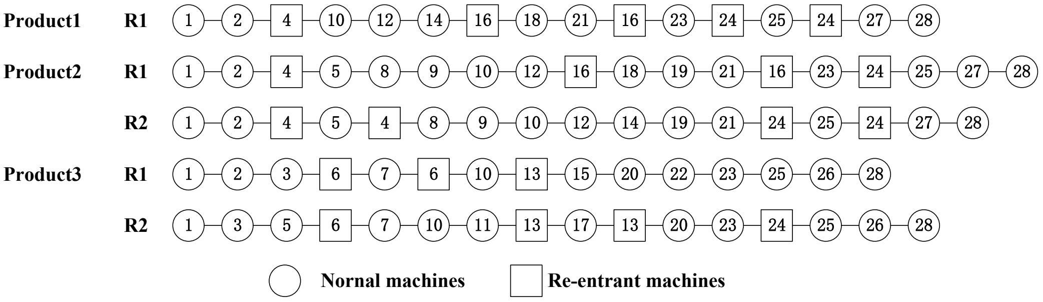


Fig. 7. Processing flow for Case 2

Table 2
The distributional parameters of processing times for Case 2.

Machine	Mean	Std.Dev	Machine	Mean	Std.Dev
1	85	7	15	175	25
2	100	16	16	105	8
3	175	14	17	175	18
4	105	10	18	165	22
5	130	10	19	165	18
6	160	10	20	170	10
7	160	12	21	125	8
8	175	14	22	175	15
9	175	15	23	100	8
10	85	4	24	72	2
11	180	20	25	85	8
12	115	18	26	175	22
13	155	10	27	135	18
14	170	10	28	85	15

We use the low-fidelity approximation model and the high-fidelity simulation model to solve this problem, respectively, to obtain the results of the solution space. Fig. 8 plots the output of the solution space for different models and Fig. 9 shows the output of the solution space after ordinal transformation. For better display, we add 100 to all the results of the low-fidelity model. From Fig. 8 and Fig. 9, the trends of the output solved by the two fidelity models are similar before (Fig. 8) and after (Fig. 9) ordinal transformation.

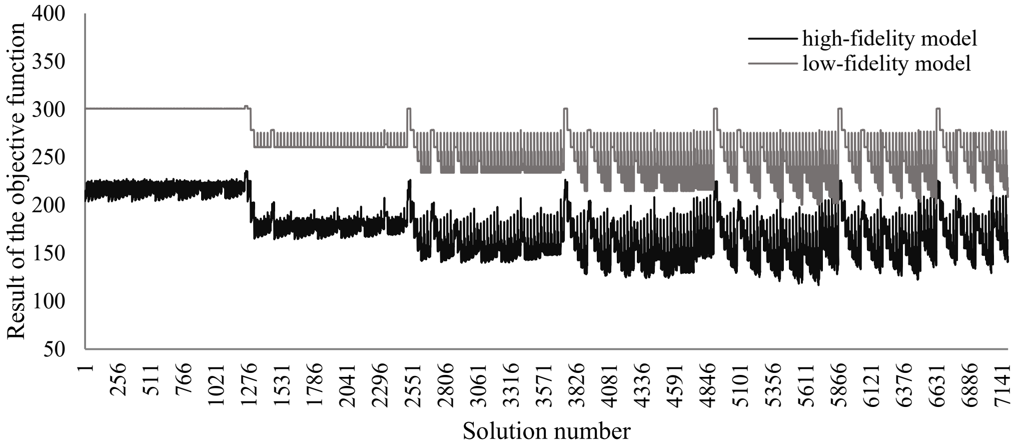


Fig. 8. The output of the solution space for Case 2

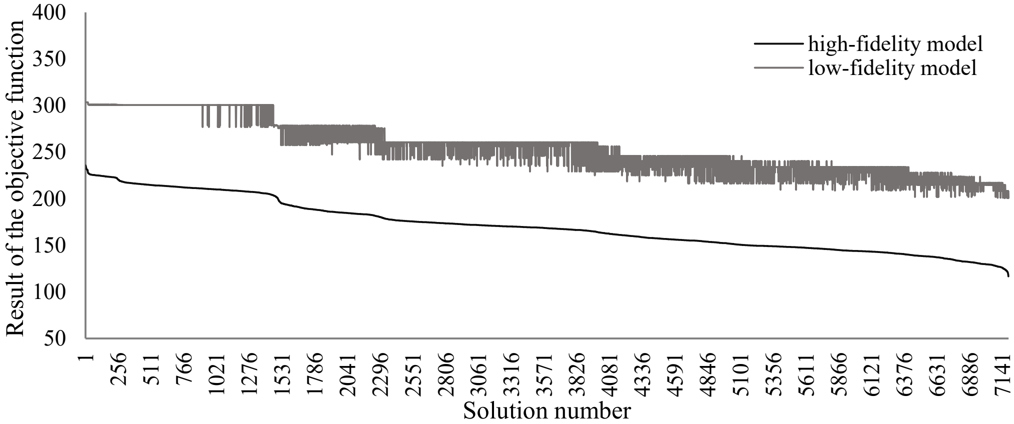


Fig. 9. The output of the solution space for Case 2 after ordinal transformation

6.2 Multi-fidelity optimization with the evolutionary algorithm for large-scale problems

We demonstrate the effectiveness of the proposed low-fidelity model in the previous section. This section compares the proposed MO2TOS with MPGA with MO2TOS and the basic simulation evaluation (BSE) method. The effectiveness of MO2TOS in combinatorial optimization problems has been fully proved in some typical studies (Zhang et al., 2020; Zhang et al., 2021).

We generate a large-scale release planning and routing allocation case (Case 3) and employ the proposed method to solve Case 3. Fig. 10 describes the processing flows for products in Case 3. Table 3 gives the distributional parameters of processing times for all the machines. All processing times follow a lognormal distribution. The demands for Product 1, Product 2, and Product 3 are 800, 1600, and 1400, respectively.

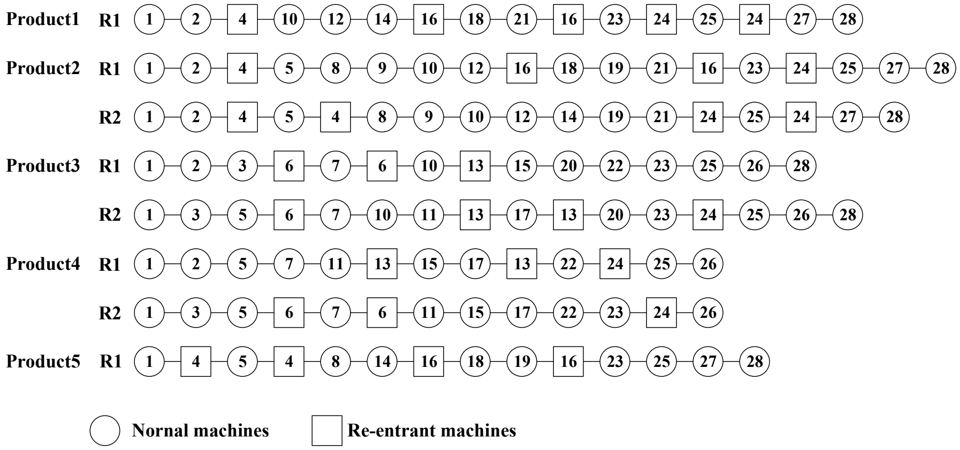

Product1 R1 1 2 4 10 12 14 16 18 21 16 23 24 25 24 27 28
Product2 R1 1 2 4 5 8 9 10 12 16 18 19 21 16 23 24 25 27 28
R2 1 2 4 5 4 8 9 10 12 14 19 21 24 25 24 27 28
Product3 R1 1 2 3 6 7 6 10 13 15 20 22 23 25 26 28
R2 1 3 5 6 7 10 11 13 17 13 20 23 24 25 26 28
Product4 R1 1 2 5 7 11 13 15 17 13 22 24 25 26
R2 1 3 5 6 7 6 11 15 17 22 23 24 26
Product5 R1 1 4 5 4 8 14 16 18 19 16 23 25 27 28
Nornal machines
Re-entrant machines

Fig. 10. Processing flow for Case 3

Table 3
The distributional parameters of processing times for Case 3

Machine	Mean	Std.Dev	Machine	Mean	Std.Dev
1	55	7	15	130	18
2	90	8	16	70	8
3	130	15	17	110	12
4	75	6	18	130	15
5	80	8	19	130	13
6	82	8	20	150	20
7	100	12	21	125	12
8	135	15	22	130	15
9	150	18	23	68	6
10	78	4	24	60	4
11	128	14	25	57	8
12	132	18	26	100	10
13	82	6	27	100	18
14	140	12	28	68	15

We first examine the optimization performance of MPGA. Fig. 11 shows the convergence curve of the objective value of the optimal solution for Case 3 when the same low-fidelity budget (10000) is allocated. As shown in Fig. 11, the proposed algorithm has stable optimization capability. The proposed MO2TOS with MPGA method is compared with the basic simulation evaluation (BSE) method and MO2TOS (Zhang et al., 2020). BSE randomly selects solutions within the limit of the maximum high-fidelity simulation budget and uses a discrete event simulation model to evaluate selected solutions. Then, it chooses the best solution based on the evaluated results. MO2TOS evaluates all the feasible solutions by a low-fidelity model, selects good solutions by ordinal transformation and optimal sampling strategies, and evaluates them by the high-fidelity model to find the best solution. MO2TOS with MPGA differs from MO2TOS in the first phase, which uses MPGA combined with a low-fidelity model to accelerate the search for the solution space.

Table 4 shows the objective values of the best solution obtained by different methods for different cases. A longer computing time is required when more fidelity budgets are allocated. BSE performs the worst compared to other methods. For small-scale problems (Case 1 and Case 2), both MO2TOS and MO2TOS with MPGA can accurately obtain the global optimal solution, and MO2TOS runs faster. In the large-scale problem (Case3), **MO2TOS with MPGA finds the optimal solution when the number of the high-fidelity budget is more than 200 and saves about 90% of the computing time.** We conclude that if the release planning problem is small-scale (The number of feasible solutions is less than 100,000), we suggest choosing the MO2TOS. Otherwise, the MO2TOS with MPGA is recommended.

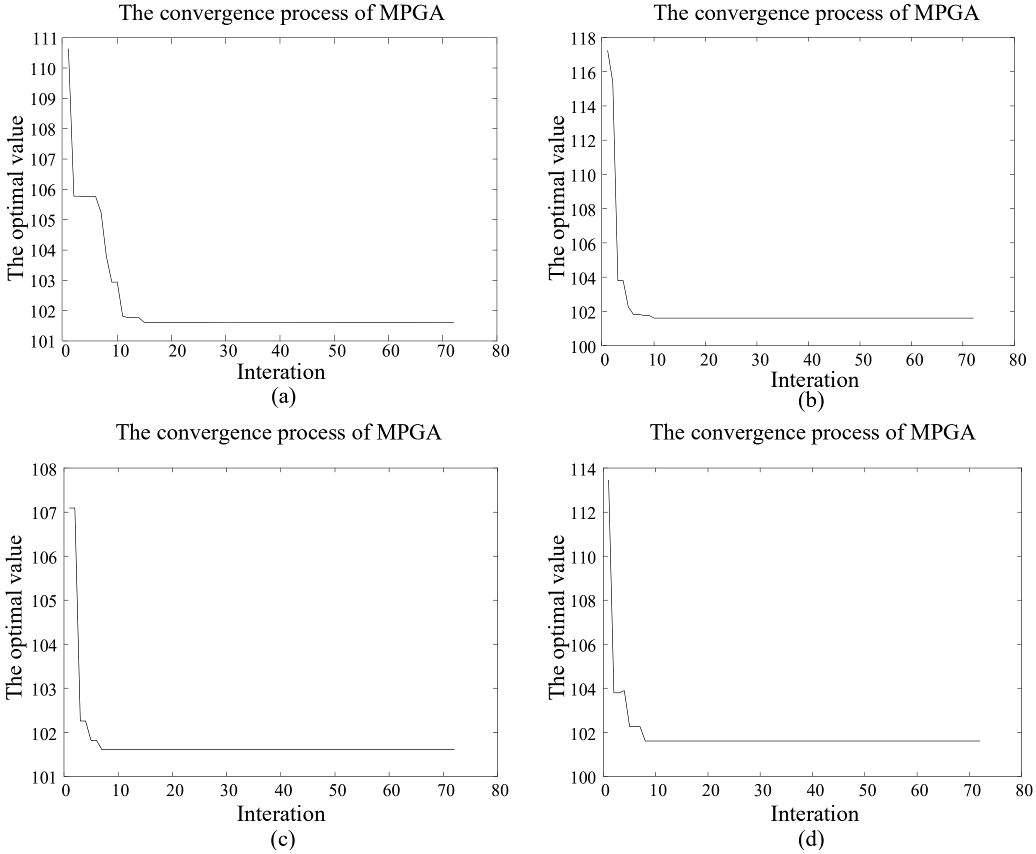


Fig. 11. The convergence curve of the objective values for Case 3

Table 4
The optimal results for different cases.

a) Case 1

High-fidelity budget	The optimal result			Runtime (s)		
	BSE	MO^{2}TOS	MO2TOS with MPGA	BSE	MO^{2}TOS	MO^{2}TOS with MPGA
200	123.647	**114.445**	114.445	275	**187**	201
400	124.098	**114.445**	114.445	818	**669**	684
600	122.671	**114.445**	114.445	1918	**1425**	1490
800	120.084	**114.445**	114.445	3565	**2499**	2654
1000	117.575	**114.445**	114.445	5149	**4365**	4693
1200	118.295	**114.445**	114.445	7906	**6475**	6807

b) Case 2

High-fidelity budget	The optimal result			Runtime (s)		
	BSE	MO^{2}TOS	MO^{2}TOS with MPGA	BSE	MO^{2}TOS	MO^{2}TOS with MPGA
200	127.637	**116.647**	116.647	385	**265**	287
400	124.951	**116.647**	116.647	1233	**777**	803
600	123.797	**116.647**	116.647	2420	**1568**	1632
800	120.927	**116.647**	116.647	3120	**2794**	3004
1000	121.552	**116.647**	116.647	5409	**4504**	4733
1200	124.458	**116.647**	116.647	8460	**6610**	6998

c) Case 3

High-fidelity budget	The optimal result			Runtime (s)		
	BSE	MO^{2}TOS	MO^{2}TOS with MPGA	BSE	MO^{2}TOS	MO^{2}TOS with MPGA
200	128.048	113.475	***113.292***	425	2221	***358***
400	120.651	113.447	**113.292**	1351	2874	**824**
600	121.166	113.292	**113.292**	3028	3944	**1672**
800	124.113	113.292	**113.292**	4908	5503	**2785**
1000	120.061	113.292	**113.292**	7948	7569	**4906**
1200	117.943	113.292	**113.292**	10244	10348	**7149**

7. Concluding remarks

In this research, we present an effective multi-fidelity simulation optimization method named MO2TOS with MPGA, to address release planning and routing allocation problems in re-entrant mixed-flow shops. The method combines the advantages of rapid evaluation of analytical models and accurate evaluation of simulation models. In our research, low-fidelity models are mathematical expressions and high-fidelity models are developed by discrete-event simulation. We examine the feasibility of the mathematical expression and the multi-fidelity optimization method through small-scale release planning problems and conclude that MO2TOS with MPGA can quickly obtain the global optimal solution for small- and medium-scale problems. We then use a large-scale case to test the method by comparing it with MO2TOS (Zhang et al., (2020) and BSE. Results show that the proposed method can obtain the same optimal solution as Zhang et al. (2020), and save about 90% of the computing time in this problem. Therefore, we conclude that the proposed method can achieve good effects in solving large-scale problems.

In the current research, we ignore some realistic features such as setups, dynamic events, and transport time. We plan to consider the typical features in the future and abstract release planning models with actual production scenarios.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant No. 51905196), International and Hong Kong, Macao and Taiwan high end talent exchange funding of Guangdong Province (2022A0505020007), and Guangzhou basic and applied basic research project (202201010261).

References

Albey, E., Bilge, Ü., & Uzsoy, R. (2014). An exploratory study of disaggregated clearing functions for production systems with multiple products. *International Journal of Production Research*, *52*(18), 5301–5322.

Albey, E., & Bilge, U. (2011). A hierarchical approach to FMS planning and control with simulation-based capacity anticipation. *International Journal of Production Research*, *49*(10-11), 3319-3342.

Asmundsson, J., Rardin, R. L., & Uzsoy, R. (2006). Tractable nonlinear production planning models for semiconductor wafer fabrication facilities. *IEEE Transactions on Semiconductor Manufacturing*, *19*(1), 95–111.

Asmundsson, J., Rardin, R. L., Turkseven, C. H., & Uzsoy. (2009). Production planning with resources subject to congestion. *Naval Research Logistics*, *56*(2), 142–157.

Baker, K. R. (1993). *Requirements planning*, in Handbooks in Operations Research and Management Science, 3, 571–627.

Bang, J., & Kim, Y. (2010). Hierarchical Production Planning for Semiconductor Wafer Fabrication Based on Linear Programming and Discrete-Event Simulation. *IEEE Transactions on Automation Science and Engineering*, *7(*2), 326-336.

Billington, P., McClain, J., & Thomas, L. J. (1983). Mathematical programming approaches to capacity-constrained MRP systems: review, formulation, and problem reduction. *Management Science*, *29*, 1126–1141.

Chang, P. C., Wang, Y. W., & Ting, C. J. (2008). A fuzzy neural network for the flow time estimation in a semiconductor manufacturing factory. *International Journal of Production Research*, *46*(4), 1017–1029.

Chen, C., & Lee, L. (2010). Stochastic Simulation Optimization (An Optimal Computing Budget Allocation). *Back Matter*, 175-227.

Chen, R, Xu, J., Zhang, S., & Chen, C. (2015). An effective learning procedure for multi-fidelity simulation optimization with ordinal transformation. *IEEE International Conference on Automation Science & Engineering.* IEEE.

Chen, W., Wang, Z., & Chan, F. (2015). Robust Production Capacity Planning of a Wafer Fabrication System with Uncertain Wafer Lots Transfer Probabilities. *IFAC PapersOnLine*, *48*(3), 1586-1591.

Chiu, C. C., & Lin, J. T. (2021). Hybrid Evolutionary Algorithm with an Optimal Sample Allocation Strategy for Multifidelity Simulation Optimization Problems. *Asia-Pacific Journal of Operational Research*, *38*(02), 2050043.

Chiu, C., Zhang, S., Lin, J. T., Zhen, L., & Huang, E. (2016). Improving the efficiency of evolutionary algorithms for large-scale optimisation with multi-fidelity models. In *2016 Winter Simulation Conference*, Washington, DC, USA, 815-826.

Chung, S. H., & Lai, C. (2006). Job releasing and throughput planning for wafer fabrication under demand fluctuating make-to-stock environment. *The International Journal of Advanced Manufacturing Technology*, *31*(3), 316-327. doi: http://dx.doi.org/10.1007/s00170-005-0185-8

Fowler, J., & Mönch. L. (2017). Modeling and Analysis of Semiconductor Manufacturing. *Advances in Modeling and Simulation.*

Li, J., & Meerkov, S. (2009). *Production Systems Engineering*. Springer US.

Hackman, S. T., & R. C. Leachman. 1989. A general framework for modeling production. *Management Science, 35*(4), 478–495.

Hong, T., & Chien, C. (2020). A simulation-based dynamic scheduling and dispatching system with multi-criteria performance evaluation for Industry 3.5 and an empirical study for sustainable TFT-LCD array manufacturing. *International Journal of Production Research*, *58*(24), 7531-7547.

Hsu, S. Y., & Sha, D. Y. (2004). Due date assignment using artificial neural networks under different shop floor control strategies. *International Journal of Production Research*, *42*(9), 1727–1745.

Kacar, N. B., Irdem, D., & Uzsoy, R. (2012). An Experimental Comparison of Production Planning Using Clearing Functions and Iterative Linear Programming-Simulation Algorithms. *IEEE Transactions on Semiconductor Manufacturing*, *25*(1), 104-117.

Kacar, N. B., Monch, L., & Uzsoy, R. (2013). Planning Wafer Starts using Nonlinear Clearing Functions: A Large-Scale Experiment. *IEEE Transactions on Semiconductor Manufacturing*, *26*(4), 602-612.

Kacar, N. B., Monch, L., & Uzsoy, R. (2016). Modeling Cycle Times in Production Planning Models for Wafer Fabrication. *IEEE Transactions on Semiconductor Manufacturing*, *29*(2), 153-167.

Kim, B., & Kim, S. (2001). Extended model for a hybrid production planning approach. *International Journal of Production Economics*, *73*(1), 165-173.

Kim, S. H., & Lee, Y. H. (2016). Synchronized production planning and scheduling in semiconductor fabrication. *Computers & Industrial Engineering*, *96*(6), 72-85.

Kopp, D., Monch, L., Pabst, D., & Stehli, M. (2019). Qualification Management in Wafer Fabs: Optimization Approach and Simulation-Based Performance Assessment. *IEEE Transactions on Automation Science and Engineering*, *17*(1), 475-489.

Leachman, R. C. (1993). *Modeling techniques for automated production planning in the semiconductor industry. in Optimization in Industry*, T. A. Ciriani and R. C. Leachman, Eds. Chichester, U.K.: Wiley 1–30

Leachman, R. C. (2001). *Semiconductor production planning. in Handbook of Applied Optimization*, P. M. Pardalos and M. G. C. Resende, Eds. New York, NY, USA: Oxford Univ. Press 746–762.

Leachman, R. C., Benson, R., Liu, C., & Raar, D. J. (1996). IMPReSS: An automated production planning and delivery quotation system at Harris corporation—Semiconductor sector. *Interfaces*, *26*(1), 6–37.

Leachman, R. C., & Raar, D. J. (1994). *Optimized production planning and delivery quotation for the semiconductor industry*. in Optimization in Industry 2, T. A. Ciriani and R. C. Leachman, Eds. Chichester, U.K.: Wiley 63–72.

Lee, L., Chen, C., Chew, E., Li, J., Nugroho, A., & Zhang, S. (2010). A Review of Optimal Computing Budget Allocation Algorithms for Simulation Optimization Problem. *International Journal of Operations Research*, *7*(2), 19-31.

Lester, C., Yates, C., Giles, M., & Baker, R. (2014). An adaptive multi-level simulation algorithm for stochastic biological systems. *Journal of Chemical Physics*, *142*(2), 024113.

Li, H., Li, Y., Lee, L., Chew, E., Pedrielli, G., & Chen, C. (2015). Multi-objective multi-fidelity optimisation with ordinal transformation and optimal sampling. In *2015 Winter Simulation Conference*, California, USA, 3737-3748.

Lim, S., Kim, J., & Kim, H. (2014). Simultaneous order-lot pegging and wafer release planning for semiconductor wafer fabrication facilities. *International Journal of Production Research*, *52*(11-12), 3710-3724.

Mather, H., & Plossl, G. W. (1978). Priority fixation versus throughput planning. *Production and Inventory Management, 19*, 27–51.

Milne, R. J., Mahapatra, S., & Wang, C. T. (2015). Optimizing planned lead times for enhancing performance of MRP systems. *International Journal of Production Economics*, *167*(9), 220–231.

Missbauer, H. (2011). Order release planning with clearing functions: A queueing-theoretical analysis of the clearing function concept. *International Journal of Production Economics*, *131*(1), 399-406.

Missbauer, H. (2020). Order release planning by iterative simulation and linear programming: theoretical foundation and analysis of its shortcomings. *European Journal of Operational Research*, *280*(2), 495-507.

Orlicky, J. (1975). *Material Requirements Planning: The New Way of Life in Production and Inventory Management*. New York, NY, USA: McGraw-Hill.

Patil, R.J. (2008). Using ensemble and metaheuristics learning principles with artificial neural networks to improve due date prediction performance. *International Journal of Production Research*, *46*(21), 6009–6027.

Pürgstaller, P., & Missbauer, H. (2011). Rule-based vs. optimisation-based order release in workload control: A simulation study of a MTO manufacturer. *International Journal of Production Economics, 140*(2).

Philipoom, P. R., Rees, L. P., & Wiegmann, L. (1994). Using Neural Networks to Determine Internally-Set Due-Date Assignments for Shop Scheduling. *Decision Sciences*, *25*(5-6), 825–851.

Philipoom, P. R., Wiegmann, L., & Rees, L.P. (1997). Cost-based due-date assignment with the use of classical and neural-network approaches. *Naval Research Logistics*, *44*(1), 21–46.

Qiu, Y., Song, J., & Liu, Z. (2016). A Simulation Optimisation on the Hierarchical Health Care Delivery System Patient Flow Based on Multi-Fidelity Models. *International Journal of Production Research, 54*(21-22), 1-16.

Singh, R., & Mathirajan, M. (2018). Experimental investigation for performance assessment of scheduling policies in semiconductor wafer fabrication—a simulation approach. *The International Journal of Advanced Manufacturing Technology*, *99*(5), 1503-1520.

Schneckenreither, M., Haeussler, S., & Gerhold, C. (2021). Order release planning with predictive lead times: a machine learning approach. *International Journal of Production Research*, *59*(11), 3285-3303.

Sébastien, P., & Mathieu, P. (2011). An interaction-oriented model for multi-scale simulation. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, Catalonia, Spain, 332-337.

Selcuk, B., Fransoo, J. C., & De Kok, A. (2006). The effect of updating lead times on the performance of hierarchical planning systems. *International Journal of Production Economics*, *104*(2), 427–440.

Shao, G., Jain, S., Laroque, C., Lee, L. H., Lendermann, P., & Rose, O. (2019). Digital Twin for Smart Manufacturing: The Simulation Aspect. In *2019 Winter Simulation Conference (WSC*), National Harbor, MD, USA, 2085-2098.

Song, J., Qiu, Y., Xu, J., & Yang, F. (2019). Multi-fidelity sampling for efficient simulation-based decision making in manufacturing management. *IISE Transactions*, *51*(7), 792-805.

Thuerer, M., Stevenson, M., Land, M., & Fredendall, L. (2019). On the combined effect of due date setting, order release, and output control: an assessment by simulation. *International Journal of Production Research*, *57*(6), 1741-1755.

Vollmann, T. E., Berry, W. L., & Whybark, D. C. (1988). *Manufacturing planning and control systems*. Dow Jones-Irwin.

Whitt, W. (1983). The Queueing Network Analyser. *Bell Labs Technical Journal*, *62*(9).

Wolosewicz, C., Dauzère-Pérès, S., & Aggoune, R. (2015). A lagrangian heuristic for an integrated lot-sizing and fixed scheduling problem. *European Journal of Operational Research*, *244*(1), 3-12.

Xu J., Nelson, B., & Hong, L. (2010). Industrial strength COMPASS: A comprehensive algorithm and software for optimization via simulation. *Acm Transactions on Modeling & Computer Simulation*, *20*(1), 1-29.

Xu, J., Zhang, S., Huang, E., Chen, C., Lee, L., & Celik, N. (2014a). Efficient multi-fidelity simulation optimization. *Proceedings of the Winter Simulation Conference*, Savanah, GA.

Xu, J., Zhang, S., Huang, E., Chen, C., Lee, L., & Celik, N. (2014b). An Ordinal Transformation Framework for Multi-fidelity simulation-based optimisation. In *2014 IEEE International Conference on Automation Science and Engineering*, Taiwan, China, 385-369.

Xu, J., Zhang, S., Huang, E., Chen, C., Lee, L., & Celik, N. (2016). MO2TOS: Multi-Fidelity Optimisation with Ordinal Transformation and Optimal Sampling. *Asia-Pacific Journal of Operational Research*, *33*(3), 1650017.

Yanıkoğlu, I., Albey, E., & Uzsoy, R. (2017). Load Dependent Lead Time Modeling: A Robust Optimization Approach. *Winter Simulation Conference*.

Zhang, F., Song, J., Dai, Y., & Xu, J. (2020). Semiconductor wafer fabrication production planning using multi-fidelity simulation-based optimisation. *International Journal of Production Research*, *58*(21), 6585-6600.

Zhang, Z., Guan, Z., Gong, Y., Luo, D., & Yue, L. (2022). Improved multi-fidelity simulation-based optimisation: application in a digital twin shop floor. *International Journal of Production Research*, *60*(3), 1016-1035.

Ziarnetzky, T., Kacar, N., Monch, L., & Uzsoy, R. (2015). Simulation-based performance assessment of production planning formulations for semiconductor wafer fabrication. *Winter Simulation Conference.* IEEE.