Coordinated scheduling of intercell production and intercell transportation in the equipment manufacturing industry

Yunna Tian\textsuperscript{a,b}, Dongni Li\textsuperscript{a*}, Pengyu Zhou\textsuperscript{a} and Leheng Wang\textsuperscript{a}

\textsuperscript{a}Beijing Lab of Intelligent Information Technology, School of Computer Science, Beijing Institute of Technology, Beijing, PR China; \textsuperscript{b}College of Mathematics and Computer Science, Yan’an University, Yan’an, PR China

(Received 18 July 2015; accepted 27 January 2016)

Intercell moves are caused by exceptional parts which need to be processed in multiple cells. Intercell cooperation disrupts the cellular manufacturing philosophy of creating independent cells, but is essential to lower the costs for enterprises. This article addresses an intercell scheduling problem considering limited transportation capability. To solve this problem, a two-stage ant colony optimization approach is proposed, in which pre-scheduling and re-scheduling are performed sequentially. To evaluate and optimize the interaction of production and transportation, a transportation benefit function is presented, according to which the scheduling solutions are adjusted. The computational results show that the transportation benefit function is more effective than other strategies, and the proposed approach has significant advantages over CPLEX in both the production dimension and the transportation dimension.

Keywords: manufacturing; intercell; transportation; coordinated scheduling; ant colony optimization

1. Introduction

In a cellular manufacturing system, parts that have similar processing routes are grouped into one part family. Accordingly, all of the machines which are required by these parts are located in one manufacturing cell. However, such an ideal mode is difficult to realize. The production mode of the equipment manufacturing industry can be characterized by high variety and low volume in mixed production lines, which makes the processing routes of parts more complex. The processing capability of a cell may be limited as some exceptional parts need to be processed in more than one cell. This phenomenon is termed intercell transfers (Garza and Smunt 1991).

A survey of the equipment manufacturing industry indicates that 72.9\% manufacturing firms have adopted manufacturing cells (Johnson and Wemmerlöv 2009) while only 10\% of the surveyed shops process parts completely within cells, and the mean median level of intercell flow is approximately 20\%. Therefore, scheduling in the context of multiple cells with intercell transfers is worth studying, and the impact of intercell transfers on manufacturing systems should be quantitatively analysed.

*Corresponding author. Email: li_dongni@163.com

© 2016 Informa UK Limited, trading as Taylor & Francis Group
Some studies solved the scheduling problem considering intercell transfers (Solimanpur, Vrat, and Shankar 2004; Golmohammadi and Ghodsi 2009; Tang et al. 2010; Tavakkoli-Moghaddam et al. 2010; Li et al. 2013; Solimanpur and Elmi 2013; Halat and Bashirzadeh 2015), and some of them considered transportation costs caused by the intercell transfer (Tang et al. 2010; Li et al. 2013; Halat and Bashirzadeh 2015). However, there is an implicit assumption that the intercell transportation capability is sufficient, so that once the parts complete the operations in the current cell (if any), they can be delivered. Actually, this assumption is difficult to realize.

In real production, each cell usually has one or more transporters for intercell transportation, and the parts have to stay in the cell if the transporters are unavailable, which results in extra waiting time. Owing to the limited transportation capacity, the transporter has to determine which parts to load, which constitutes a batch formation subproblem. On the other hand, parts that are currently in the same cell may have to go to different destination cells for their next operations. Thus, the transporter must determine the sequence of delivering the parts, which is a routing subproblem. Therefore, the intercell scheduling problem becomes more difficult when transporters are involved in decision making.

With respect to the transportation dimension of the intercell scheduling problem, some similar characteristics can be found in the vehicle routing problem (Mousavi, Tavakkoli-Moghaddam, and Jolai 2013; Pillac et al. 2013; Ma and Xu 2015; Wang et al. 2015). However, with respect to the whole intercell scheduling problem including both intercell production and intercell transportation, the complexity is severely increased compared with vehicle routing problems. In the addressed problem, the transporter needs to determine which parts to load and in which sequence these parts should be delivered. These decisions on the transporters directly affect the results of machine scheduling, and vice versa, which results in the interaction of production and transportation. In this regard, the routing cost is not the most important measure of the transporters, but the effects of the transportation dimension on the overall performance, such as the completion time and flow time, should be of more concern.

Based on the above analysis, the key component of the coordination scheduling of intercell production and intercell transportation is to develop an appropriate coordinating mechanism.

In recent research, the model of job-shop scheduling with multi-load automated guided vehicles (Kim and Hwang 2001; Zeng, Tang, and Yan 2014; Saidi-Mehrabad et al. 2015) is similar to that of the addressed problem. In addition, in the transportation process, batch formation is conducted and routes for vehicles are decided. However, there is a common issue in these studies that the capability for intercell transportation is not considered. The coordination of production and transportation is ignored, which exerts a negative effect on the scheduling results.

A two-stage ant colony optimization (ACO) approach which coordinates intercell production and intercell transportation is developed to solve the addressed problem and minimize the makespan. In the first stage, pre-scheduling is performed to generate tentative results for operation assignment and operation sequencing, assuming that the intercell transportation capability is sufficient. In the second stage, re-scheduling is performed considering the intervention of the intercell transportation, which adjusts the tentative scheduling derived from the first stage and finally generates the scheduling solutions for the machines and transporters in an integrated manner. To evaluate and optimize the interaction of the intercell production and intercell transportation, a problem-specific transportation benefit function (TBF) and a sequencing repair strategy are proposed for the re-scheduling stage.

The rest of this article is organized as follows. Section 2 presents the description and formulation of the addressed problem. The two-stage ACO approach is discussed in detail in Section 3. Experimental tests and results are reported in Section 4, and followed by conclusions in Section 5.
2. Problem description

A cellular manufacturing system consists of a number of manufacturing cells. Machines are laid out within each cell.

2.1. Assumptions

The scheduling problem considered in this article is based on the following assumptions.

- All parts are released initially.
- Each machine can only process parts individually.
- The processing route of a part consists of multiple operations that are under sequence constraints and require machines in two or more cells.
- Owing to the overlapping capabilities of machines, parts have alternative processing routes; however, for a given operation, there is at most one machine that can process it within each cell.
- The intercell transfer time is taken into account, while the intracell transfer time is neglected.
- All of the transporters are identical and each cell has only one transporter, which has a capacity constraint.
- A part can only be loaded on to the transporter at its original cell and can only be unloaded at its destination cell.
- A batch of parts on a transporter may consist of the parts with different destination cells. The transporter visits different destination cells in an order and unloads the corresponding parts, and no parts are allowed to be loaded on to the transporter during the transportation.
- Once all of the parts in a batch have been transported to their respective destination cells, the transporter returns to its original cell immediately, and the times for unloading parts are neglected.
- Other conditions, such as set-up time, pre-emption, machine breakdown, limited buffers and the absence of operators, are not considered in this article.

2.2. List of symbols

The notations adopted to describe this problem are presented as follows.

\[
\begin{align*}
  i &= 1, \ldots, N \quad \text{index for parts} \\
  j &= 1, \ldots, O_i \quad \text{index for operations of part } i \\
  m &= 1, \ldots, M \quad \text{index for machines} \\
  c &= 1, \ldots, C \quad \text{index for cells and the associated transporters} \\
  b &= 1, \ldots, B_c \quad \text{index for batches on transporter } c \\
  t &= 1, \ldots, T \quad \text{time} \\
  o_{i,j} &= j\text{th operation of part } i \\
  O_i &= \text{number of operations for part } i \\
  p_{i,j,m} &= \text{processing time of operation } o_{i,j} \text{ on machine } m \\
  S_i &= \text{size of part } i \\
  v &= \text{capacity of a transporter} \\
  d_{c,c'} &= \text{transfer time from cell } c \text{ to cell } c', \text{ where } c \text{ and } c' \text{ are adjacent to each other on the intercell transportation route of a transporter} \\
  t_{i,j} &= \text{transportation time of part } i \text{ on the completion of } o_{i,j} \\
  D_{i,j} &= \text{destination cell of part } i \text{ on the completion of } o_{i,j}
\end{align*}
\]
\( a_{i,j} \) the time part \( i \) arrives at the buffer of the corresponding machine for processing \( o_{i,j} \)

\( l_{c,m} \) \[
\begin{cases}
1, & \text{machine } m \text{ is located in cell } c \\
0, & \text{otherwise}
\end{cases}
\]

\( e_{i,j} \) \[
\begin{cases}
1, & o_{i,j+1} \text{ and } o_{i,j} \text{ are assigned to different cells} \\
0, & \text{otherwise}
\end{cases}
\]

Decision variables:

\[
x_{i,j,m} = \begin{cases}
1, & o_{i,j} \text{ is assigned to machine } m \\
0, & \text{otherwise}
\end{cases}
\]

\[
y_{i,j,t} = \begin{cases}
1, & o_{i,j} \text{ starts at time } t \\
0, & \text{otherwise}
\end{cases}
\]

\[
z_{c,b,t} = \begin{cases}
1, & \text{the transportation of batch } b \text{ on vehicle } c \text{ starts at time } t \\
0, & \text{otherwise}
\end{cases}
\]

\[
u_{i,j,c,b,q} = \begin{cases}
1, & \text{the transportation of part } i \text{ is the } q\text{th to be delivered in batch } b \\
0, & \text{on vehicle } c \text{ on the completion of } o_{i,j}
\end{cases}
\]

\[
s_{i,j} = \sum_{t=1}^{T} y_{i,j,t}, \forall i, j
\]

where \( s_{i,j} \) represents the start time of \( o_{i,j} \).

\[
p_{i,j} = \sum_{m=1}^{M} x_{i,j,m} p_{i,j,m}, \forall i, j
\]

where \( p_{i,j} \) represents the actual processing time of \( o_{i,j} \).

\[
f_{i,j} = s_{i,j} + p_{i,j}, \forall i, j
\]

where \( f_{i,j} \) represents the completion time of \( o_{i,j} \).

### 2.3. Objective function and constraints

Following the assumptions and notations mentioned above, the mathematical model of the addressed problem is presented below.

The objective is to minimize the makespan as follows:

\[
\min \{ C_{\text{max}} \} \tag{1}
\]

where \( C_{\text{max}} \) is the makespan and \( C_{\text{max}} = \max \{ f_{i,j} \} \).
The actual production has many peculiar characteristics and must be subject to

\[ \sum_{m=1}^{M} x_{i,j,m} = 1, \forall i, j \]  \hspace{2cm} (2)
\[ \sum_{m=1}^{M} x_{i,j,m} p_{i,j,m} > 0, \forall i, j \]  \hspace{2cm} (3)
\[ \sum_{t=1}^{T} y_{i,j,t} = 1, \forall i, j \]  \hspace{2cm} (4)
\[ \sum_{\ell=1}^{N} O_{\ell} x_{i,j,m} y_{\ell,j,m} y_{\ell',j,m'} = 0, \forall i, j, t, m \] \hspace{2cm} (5)
\[ \sum_{c=1}^{C} \sum_{b=1}^{B_c} \sum_{t=1}^{T} z_{c,b,t} = 1 \] \hspace{2cm} (6)
\[ \sum_{c=1}^{C} \sum_{b=1}^{B_c} \sum_{q=1}^{Q} u_{i,j,c,b,q} = e_{i,j}, \forall i, j \] \hspace{2cm} (7)
\[ \sum_{b=1}^{B_c} \sum_{q=1}^{Q} u_{i,j,c,b,q} = \sum_{m=1}^{M} x_{i,j,m} l_{m,c}, \forall i, j, c, e_{i,j} = 1 \] \hspace{2cm} (8)
\[ \sum_{i=1}^{N} \sum_{j=1}^{Q} \sum_{q=1}^{Q} u_{i,j,c,b,q} S_{i} \leq v, \forall c, b \] \hspace{2cm} (9)
\[ D_{i,j} = \sum_{m=1}^{M} l_{m,c} x_{i,j,m} c, \forall i, j \] \hspace{2cm} (10)
\[ u_{i,j,c,b,q} u_{i',j',c,b,q} D_{i,j} = u_{i,j,c,b,q} u_{i',j',c,b,q} d_{i,j, c,b,q}, \forall i, j, i', j', c, b, q \] \hspace{2cm} (11)
\[ t_{i,j} \geq u_{i,j,c,b,q} u_{i',j',c,b,q} (t_{i,j} + d_{i,j,c,b,q}), \forall i, j, i', j', c, b, q \] \hspace{2cm} (12)
\[ t_{i,j} \geq u_{i,j,c,b,q} d_{i,j,c,b,q}, \forall i, j, c, b \] \hspace{2cm} (13)
\[ \sum_{t=1}^{T} z_{c,b,t} t \geq u_{i,j,c,b,q} f_{i,j}, \forall i, j, c, b, q \] \hspace{2cm} (14)
\[ a_{i,j+1} = \max \left\{ f_{i,j}, e_{i,j} \left( \sum_{c=1}^{C} \sum_{b=1}^{B_c} \sum_{q=1}^{Q} u_{i,j,c,b,q} \sum_{t=1}^{T} z_{c,b,t} t + t_{i,j} \right) \right\}, \forall i, j \] \hspace{2cm} (15)
\[ s_{i,j} \geq a_{i,j} \] \hspace{2cm} (16)
\[ \sum_{t=1}^{T} z_{c,b+1,t} t \geq \sum_{t=1}^{T} z_{c,b,t} t + t_{i,j} + u_{i,j,c,b,q} d_{c,D_{i,j}}, \forall i, j, c, b, q \] \hspace{2cm} (17)

Equation (2) ensures that each operation is assigned to only one machine. Equation (3) ensures that each operation is assigned to the machine capable of processing it. Equation (4) states that each operation has to and can only be processed once. Equation (5) states that a machine only
processes operations individually. Equation (6) ensures that each batch is transported and only transported once. Equation (7) states that a part is assigned to only one batch on a transporter for a given intercell transportation. Equation (8) indicates that a part can only be assigned to the transporter of the cell where the preceding operation of that part has been processed. Equation (9) states that the sum of all parts’ size in a batch is less than or equal to the capacity of the transporter. Equation (10) gives the definition of the destination cell of a part for a given intercell transportation. Equation (11) ensures that only the parts with the same destinations cell can be unloaded simultaneously. Equation (12) indicates that a part cannot be unloaded until the transporter has delivered all of the preceding parts in the same batch and has arrived at the destination cell of that part. Equation (13) states that the first part in a batch cannot be unloaded until the transporter has arrived at the destination cell of the first part. Equation (14) ensures that the transportation of a batch cannot start until the preceding operations of all parts in that batch have been completed. Equations (15) and (16) give the definitions of the arrival time of the part when it arrives at the buffer of the corresponding machine, which means that an operation cannot start until its preceding operation is completed and the part has been transported to the corresponding cell (if an intercell transportation occurs). Equation (17) states that the transportation of a batch cannot start until the transporter has delivered all of the parts in the preceding batch and returned to the original cell.

3. The two-stage ant colony optimization approach

Metaheuristics can be divided into two categories: constructive metaheuristics and improvement metaheuristics (Gendreau and Potvin 2005). Constructive metaheuristics build a solution from scratch by iteratively introducing new elements. The improvement metaheuristics iteratively modify a solution. Because the number of batches for each transporter is unknown until the entire schedule is built, an improvement metaheuristic is an improper way to construct the initial solution for the transportation subproblem. However, constructive metaheuristics, such as ACO, can solve this problem conveniently. Based on the above analysis, a two-stage ACO approach is proposed.

3.1. Framework of the two-stage ant colony optimization approach

In the first stage, pre-scheduling is conducted, assuming transportation capacity to be sufficient. The ideal start time of each operation, namely the expected time of arrival (ETA), is obtained. In the second stage, re-scheduling is conducted, considering intercell transportation. Owing to the limited capacity, parts waiting in the buffer of a transporter may be delayed if the transporter is unavailable. Therefore, a sequence repair strategy is employed to dynamically adjust the sequence of parts in the buffer. Then, the intercell transportation scheduling is conducted based on the ETA.

The general algorithm of the two-stage ACO approach is presented as follows.

Step 1. Initialization;
Step 2. Each ant builds a feasible solution independently;
   (i) Perform pre-scheduling;
   (ii) Take the ideal start time of each operation as the initial ETA, and perform re-scheduling;
Step 3. The pheromone trails are updated;
Step 4. If no better solution is found for a given number of iterations, end. Otherwise, go to Step 2.
3.2. Pheromone trails

The two-stage ACO approach defines three different pheromone structures for operation assignment, operation sequencing and intercell transportation.

(i) Pheromone trails for operation assignment
For operation assignment, a pheromone matrix with the size of $O \times M$ is defined, where $O$ and $M$ denote the number of operations and machines, respectively. The element $(o_{i,j}, m)$, denoted by $\tau_{i,j,m}$, represents the desirability of having $o_{i,j}$ processed on machine $m$.

(ii) Pheromone trails for operation sequencing
For operation sequencing, $M$ pheromone matrices with the size of $O \times O$ are defined. The element $(o_{i,j}, k)$ in the $m$th matrix, denoted by $\tau_{m,i,j,k}$, represents the desirability of having $o_{i,j}$ as the $k$th operation that machine $m$ processes.

(iii) Pheromone trails for intercell transportation
For intercell transportation, $C$ pheromone matrices with the size of $O \times O$ are defined, where $C$ denotes the number of cells. The element $(o_{i,j}, k)$ in the $c$th matrix, denoted by $\tau_{c,i,j,k}$, represents the desirability of having part $i$ as the $k$th part to be transported on the transporter of cell $c$ when $o_{i,j}$ is completed.

3.3. Pre-scheduling

In pre-scheduling, each operation is first assigned to a machine, then all the operations assigned to the same machine are sequenced. The constraint of transportation capability is neglected in this stage. Therefore, parts can be delivered to the next cell (if any) without delay. Intercell transfer times are only determined by the distance between cells.

3.3.1. Operation assignment

The probability of assigning $o_{i,j}$ to machine $m$ is calculated by

$$
\Pr_{i,j,m} = \frac{\tau_{i,j,m}^{\alpha_1} \rho_{i,j,m}^{\beta_1}}{\sum_{k=1}^{M} \tau_{i,j,k}^{\alpha_1} \rho_{i,j,k}^{\beta_1}} \quad (18)
$$

where $\alpha_1$ and $\beta_1$ are exponent parameters which determine the relative influence of the pheromone and heuristic information, respectively; $\tau_{i,j,m}$ represents the desirability of having $o_{i,j}$ processed on machine $m$; $M$ represents the number of machines; and $\rho_{i,j,m}$ represents the heuristic information of operation $o_{i,j}$ on machine $m$, which is calculated by

$$
\rho_{i,j,m} = \frac{1}{p_{i,j,m} + TT_{m',m}} \quad (19)
$$

In Equation (19), $m'$ is the machine on which the preceding operation of part $i$ is processed ($m = m'$ if $o_{i,j}$ is the first operation of part $i$); $TT_{m',m}$ represents the transfer time from the cell containing machine $m'$ to the one containing $m$.

3.3.2. Operation sequencing

The probability of selecting $o_{i,j}$ as the $k$th operation that machine $m$ processes is calculated by

$$
\Pr_{m,i,j,k} = \frac{\tau_{m,i,j,k}^{\alpha_2} \rho_{m,i,j,k}^{\beta_2}}{\sum_{o_{p,q} \in SO_m} \tau_{m,p,q,k}^{\alpha_2} \rho_{m,p,q,k}^{\beta_2}} \quad (20)
$$
where $\alpha_2$ and $\beta_2$ determine the degree to which the pheromone and the heuristic information impact $P_{r_{m,i,j,k}}$, respectively; $SO_m$ represents the schedulable operation set of machine $m$; $\tau_{m,i,j,k}$ represents the desirability of having $o_{i,j}$ as the $k$th operation that $m$ processes; and $\rho_{m,i,j,k}$ represents the associated heuristic information, which is defined in Equation (21).

$$
\rho_{m,i,j,k} = \begin{cases} 
1/(\max\{f_{i,j-1} + \tau_{m',m}, g_{m,i,j}\} - g_{m,i,j}), & g_{m,i,j} < f_{i,j-1} + \tau_{m',m} \\
1, & g_{m,i,j} \geq f_{i,j-1} + \tau_{m',m}
\end{cases}
$$

where $g_{m,i,j}$ represents the time when machine $m$ is idle; $m'$ is the machine that processes the preceding operation of part $i$ ($m = m'$ if $o_{i,j}$ is the first operation of part $i$); and $\max\{f_{i,j-1} + \tau_{m',m}, g_{m,i,j}\} - g_{m,i,j}$ represents the moment in which the following conditions are satisfied: (1) part $i$ is transported to $m$; and (2) $m$ is idle.

### 3.4. Re-scheduling

Based on pre-scheduling, re-scheduling considers the capacity constraint of transporters and constructs the integrated scheduling solutions for both the machines and the transporters. The scheduling on transporters involves two aspects of decisions, i.e., batch formation, which determines which parts are transported in the same batch, and route decision, which determines the delivery sequence of parts in a batch. In the proposed approach, a single procedure integrates these two decisions: add parts individually to a batch and deliver the parts in the same batch in the sequence of adding them.

#### 3.4.1. The re-scheduling algorithm

The algorithm of re-scheduling is presented as follows.

**Step 1.** The ideal start time of each operation is taken as the initial ETA;

**Step 2.** The simulated time is set to 0;

**Step 3.** If no machine or transporter is idle, lengthen the simulated time until at least one of them becomes idle;

**Step 4.** If a machine is idle, go to Step 5. Otherwise go to Step 6;

**Step 5.** According to the pre-scheduling result, one of the operations assigned to the idle machine is selected to be processed, during which the sequence repair strategy is used (see Section 3.4.2). Go to Step 7;

**Step 6.** According to the intercell transportation algorithm (see Section 3.4.5), form a batch for the idle transporter and update the ETAs of the corresponding operations (see Section 3.4.6);

**Step 7.** If all operations are scheduled, end. Otherwise go to Step 3.

#### 3.4.2. Sequence repair strategy

In re-scheduling, the sequence of the operations is dynamically updated according to the following algorithm.

When machine $m$ is idle, and $o_{i,j}$, $o_{i',j'}$ are the next two operations to process on machine $m$, swap the $o_{i,j}$ and $o_{i',j'}$ if (1) part $i'$ has arrived at the cell of machine $m$, and (2) $o_{i',j'}$ is processed instead of $o_{i,j}$, and $o_{i',j'}$ can be completed before part $i$ arrives.
3.4.3. Transportation benefit

It is hard to decide whether to add being-processed part \( i \) to the current batch. If part \( i \) is considered in the current batch, the transporter has to wait before the part can be added to the batch, and the parts already in the current batch may be delayed. Otherwise, part \( i \) cannot be transported until the transporter returns. Because the arrival time of parts influences the batch decision, it is critical to ascertain whether parts arrive at their corresponding destination cells at the proper time.

Preliminary analysis for the coordination of production and transportation with the simplified two-machine flow shop and one transporter (see the Supplementary material for details) mainly discusses two elements, i.e. the arrival time of a part (entity) and the available time of the corresponding machine, and presents how their interaction influences the makespan. The theorems and lemma in the preliminary analysis indicate that (1) if a part arrives at its corresponding machine earlier than its ETA, the makespan will not be delayed; and (2) if the part arrives later than its ETA, a smaller deviation between the arrival time and the ETA results in a higher transportation benefit value. Therefore, a well-designed decision on transportation is beneficial for shortening the makespan.

It can be concluded that the makespan is totally determined by the start time of each part, which is determined by the above two elements. Therefore, although the addressed problem is much more complex than a two-machine flow shop with one transporter, these properties are useful for the intercell scheduling approach. The transportation benefit is presented to evaluate whether a part is transported to the destination cell at the proper time.

**Definition 1** For a given operation \( o_{i,j} \) processed in cell \( c \), whose preceding operation is processed in a different cell, the transportation benefit of \( o_{i,j} \), denoted by \( TB_{i,j} \), is defined as a function of \( TA_{i,j} \) and \( ETA_{i,j} \),

\[
TB_{i,j} = f (TA_{i,j}, ETA_{i,j})
\]

where \( TA_{i,j} \) represents the time that part \( i \) arrives at cell \( c \), and \( ETA_{i,j} \) represents the ETA of \( o_{i,j} \).

Based on the above analysis, the TBF is designed and detailed below.

\[
TB_{i,j} = \begin{cases} 
TB_{\text{max}}, & \text{cur} \leq TA_{i,j} < ETA_{i,j} \\
TB_{\text{max}} - (TA_{i,j} - d), & ETA_{i,j} \leq TA_{i,j} \leq d \\
ETA_{i,j} - d, & TA_{i,j} < \text{cur} \text{ or } TA_{i,j} > d
\end{cases}
\]

(23)

where \( TB_{\text{max}} \) is a fixed positive number, \( \text{cur} \) is the current time, \( ETA_{i,j} \) is the ETA of \( o_{i,j} \), and \( d \) is the upper limit of \( TA_{i,j} \), which is double the time it takes the transporter to visit all of the other cells and return. TBF is also presented in Figure 1, where \( t_1 \) represents the value of \( TA_{i,j} \) when \( ETA_{i,j} \leq TA_{i,j} \leq d \), and func\((t_1)\) represents the corresponding value of TBF.

3.4.4. Candidate list

In re-scheduling, the algorithm has to search a large-dimensional array for solutions, resulting in low computational efficiency. Therefore, the candidate list is adopted, and an appropriate strategy is designed for it to reach a balance between computational efficiency and optimization performance.

In the addressed problem, parts may have several destination cells and a transporter may visit several destination cells on one trip. To achieve a proper arrival time for each part in a batch simultaneously, the transportation benefit of the batch is calculated.
For batch $b$ on the transporter of cell $c$, the candidate list, denoted $\text{CL}_{c,b}$, is defined as

$$\text{CL}_{c,b} = \left\{ o_{i,j} \mid \text{part } i \in \text{TPS}_c \text{ and } S_i < v - \sum_{l \in U_b} S_l \text{ and } \Delta \text{TB}_{i,j,b} > 0 \right\}$$ (24)

where $\text{TPS}_c$ represents the transported part set of $c$, in which two conditions are satisfied: (1) part $i$ has been, or is being, processed on one of the machines in $c$; and (2) $o_{i,j+1}$ is assigned to another cell; $U_b$ represents the part set of the current batch $b$; and $\Delta \text{TB}_{i,j,b}$ is calculated by

$$\Delta \text{TB}_{i,j,b} = \text{TB}_{i,j,b} - \text{TB}_{a_{i,j,b}},$$ (25)

where $\text{TB}_{i,j,b}$ is the transportation benefit of all of the parts currently in batch $b$, supposing part $i$ has been added to batch $b$ and the transportation starts once $o_{i,j}$ is completed; $\text{TB}_{a_{i,j,b}}$ is the sum of $\text{TB}_{i,j}$ and the transportation benefit of all of the parts in batch $b$, supposing part $i$ has not been added to batch $b$, the transportation of batch $b$ starts immediately and $o_{i,j}$ will be transported once the transporter returns.

3.4.5. Intercell transportation algorithm

When the transporter of cell $c$ is idle, the algorithm of forming the next batch is described as follows.

- **Step1.** Update $\text{TPS}_c$;
- **Step2.** If $\text{TPS}_c = \emptyset$, there are no available parts. End;
- **Step3.** A batch is created with a random element of $\text{TPS}_c$;
- **Step4.** The candidate list is updated by Equation (24);
- **Step5.** If the candidate list is empty, transport the current batch. The unloading of parts follows the sequence of adding them to the batch. End;
- **Step6.** Add a part in the candidate list to the current batch, Go to Step4. The probability of the selected part is calculated by

$$\Pr_{c,i,j,k} = \frac{\tau_{c,i,j,k}^\alpha \rho_{c,i,j,k}^\beta}{\sum \tau_{c,p,q,k}^\alpha \rho_{c,p,q,k}^\beta}$$ (26)

where $\alpha_3$ and $\beta_3$ determine the degree to which the pheromone and the heuristic information impact $\Pr_{c,i,j,k}$ respectively; $\tau_{c,i,j,k}$ represents the desirability of adding $o_{i,j}$ selected as the $k$th operation to the current batch on the transporter of cell $c$; and $\rho_{c,i,j,k}$ represents the associated heuristic information, which is given by

$$\rho_{c,i,j,k} = \Delta \text{TB}_{i,j,b}$$ (27)
3.4.6. Expected time of arrival update

Pre-scheduling obtains the initial ETAs, and during re-scheduling, the start times of parts are usually delayed owing to the limited intercell transportation capability. To evaluate the transportation benefit properly, the ETA of each operation is dynamically updated. If the processing of an operation is delayed, the ETA updating algorithm will be executed when the part is unloaded.

For an operation $o_{i,j}$, the algorithm of updating its ETA is presented below.

**Step1.** If $o_{i,j}$ is the last operation of part $i$, go to Step3;

**Step2.** If $ETA_{i,j} + p_{i,j} > ETA_{i,j+1}$, then $ETA_{i,j+1} = ETA_{i,j} + p_{i,j}$. Update the ETA for $o_{i,j+1}$;

**Step3.** Let $o_{p,q}$ denote the next operation on the machine which has processed $o_{i,j}$. If $ETA_{i,j} + p_{i,j} > ETA_{p,q}$, then $ETA_{p,q} = ETA_{i,j} + p_{i,j}$. Update the ETA for $o_{p,q}$.

3.5. Pheromone update

The pheromone trails are updated after all ants have completed their schedules. Most studies adopted the elitist strategy (Blum 2002), in which the pheromone trails are only updated by the global best solution of the current iteration. However, this strategy performs poorly in convergence, which implies the need to adjust the pheromone evaporation rate; therefore, updating the pheromone trails from a series of good schedules was proposed (Huang and Liao 2008), and this strategy is adopted in the two-stage ACO approach.

The pheromone updating rules are presented as follows.

(i) If $o_{i,j}$ is assigned to machine $m$, then

$$\tau_{i,j,m} = (1 - \rho) \cdot \tau_{i,j,m} + \rho \cdot \Delta \tau$$  \hspace{1cm} (28)

(ii) If $o_{i,j}$ is the $k$th operation that machine $m$ processes, then

$$\tau_{m,i,j,k} = (1 - \rho) \cdot \tau_{m,i,j,k} + \rho \cdot \Delta \tau$$  \hspace{1cm} (29)

(iii) After $o_{i,j}$ is completed, if part $i$ is the $k$th part that the transporter of cell $c$ transports, then

$$\tau_{c,i,j,k} = (1 - \rho) \cdot \tau_{c,i,j,k} + \rho \cdot \Delta \tau, \forall k \in \bigcup_b$$  \hspace{1cm} (30)

In Equations (28), (29) and (30), $\Delta \tau = Q \cdot \text{Score}$, where $Q$ is the pheromone updating factor, which determines the quantity updated for an iteration, and $\rho (0 < \rho < 1)$ is the pheromone evaporation rate.

4. Computational experiments and results

To evaluate the performance of the proposed approach, a series of computational experiments is conducted. All of the algorithms are coded in Java and executed on a personal computer with a 3.40 GHz Core i7-2600 CPU and 4 GB RAM.

4.1. Experimental design

Because there are no benchmarks for the addressed problem, 12 test problems are designed, including small, medium and large sizes, with the number of operations for a part ranging from 5 to 19, the size of parts from 20 to 40, the distance between different cells from 6 to 50 and the
Table 1. Candidate list and heuristic information of the two-stage ant colony optimization approach.

| Alternative strategies | Contents | Descriptions |
|------------------------|----------|--------------|
| TI                     | Candidate list and heuristic information for intercell transportation | In TI, the candidate list is calculated by Equation (24), \( \rho_{c,i,j,k} \) is taken as the heuristic information and \( \Delta TB_{i,j,b} \) is calculated by TBF in Equation (23) |
| AI                     | Heuristic information for operation assignment | In AI, \( \rho_{m,i,j} \) is taken as the heuristic information and is calculated by Equation (22) |
| SI                     | Heuristic information for operation sequencing | In SI, \( \rho_{m,i,j,k} \) is taken as the heuristic information and is calculated by Equation (24) |

Note: TBF = transportation benefit function.

Table 2. Different strategies for intercell transportation.

| Alternative strategies | Contents | Descriptions |
|------------------------|----------|--------------|
| TII                    | Candidate list and heuristic information for intercell transportation | In TII, the candidate list is calculated by Equation (24), \( \rho_{c,i,j,k} \) is taken as the heuristic information and \( \Delta TB_{i,j,b} \) is calculated by TBFC1 in Equation (31) |
| TIII                   | Candidate list and heuristic information for intercell transportation | In TIII, the candidate list is calculated by Equation (24), \( \rho_{c,i,j,k} \) is taken as the heuristic information and \( \Delta TB_{i,j,b} \) is calculated by TBFC2 in Equation (32) |

The addressed problem involves three decisions: operation assignment, operation sequencing and intercell transportation. For the operation assignment and operation sequencing, the processing time and the intercell transfer time, respectively, are selected as the heuristic information. For the intercell transportation, the TBF is used to construct the candidate lists and calculate the heuristic information, as shown in Table 1.

To evaluate the TBF, three different intercell transportation strategies, which use TBF (corresponding to TI in Table 1), TBFC1 and TBFC2, are adopted in this article to construct the candidate lists and calculate the heuristic information, as shown in Table 2.

(i) Transportation benefit function 1 for comparison (TBFC1)

\[
TB_{i,j} = \begin{cases} 
\frac{TB_{\text{max}}}{\text{ETA}_{i,j} - \text{cur}} (\text{TA}_{i,j} - \text{cur}) & , \text{cur} \leq \text{TA}_{i,j} < \text{ETA}_{i,j} \\
\frac{TB_{\text{max}}}{\text{ETA}_{i,j} - d} (\text{TA}_{i,j} - d) & , \text{ETA}_{i,j} \leq \text{TA}_{i,j} \leq \text{d} \\
0 & , \text{TA}_{i,j} < \text{cur} \text{ or } \text{TA}_{i,j} > \text{d}
\end{cases}
\]

(ii) Transportation benefit function 2 for comparison (TBFC2)

\[
TB_{i,j} = \begin{cases} 
\frac{TB_{\text{max}} \cdot \exp(\text{TA}_{i,j} - \text{ETA}_{i,j})}{\text{ETA}_{i,j} - \text{cur}} & , \text{cur} \leq \text{TA}_{i,j} < \text{ETA}_{i,j} \\
\frac{TB_{\text{max}} \cdot \exp(\text{ETA}_{i,j} - \text{TA}_{i,j})}{\text{ETA}_{i,j} - \text{TA}_{i,j}} & , \text{ETA}_{i,j} \leq \text{TA}_{i,j} \leq \text{d} \\
0 & , \text{TA}_{i,j} < \text{cur} \text{ or } \text{TA}_{i,j} > \text{d}
\end{cases}
\]

In TBFC1 and TBFC2, the transportation benefit will decrease if the part arrives earlier or later than the ETA. In TBFC2, the transportation benefit decreases more slowly as the deviation between the arrival time and the ETA becomes larger.
Table 3. Candidate list and heuristic information in comparison with the two-stage ant colony optimization approach.

| Alternative strategies | Contents | Descriptions |
|------------------------|----------|--------------|
| FB                     | Candidate list and heuristic information for intercell transportation | In FB, the full batch strategy is used to generate the candidate list and ETA<sub>i,j</sub> is taken as the heuristic information |
| PT                     | Heuristic information for operation assignment and sequencing | In PT, \( \frac{1}{p_{i,j,m}} \) is taken as the heuristic information and \( p_{i,j,m} \) represents the processing time of operation \( o_{i,j} \) on machine \( m \) |

Furthermore, to evaluate the performance of the proposed approach, more strategies are designed for comparison, as shown in Table 3.

### 4.2. Parameter settings

Because the performance of the proposed approach is very sensitive to the parameters, a full factorial experiment (Youssef, Beauchamp, and Thomas 1994) is conducted. The ratio of \( \alpha_i \) and \( \beta_i \) (\( i = 1, 2, 3 \)) reflects the relative influence of the pheromone and the heuristic information. To reduce the number of parameters, \( \alpha_1, \alpha_2 \) and \( \alpha_3 \) are set to 1; therefore, \( \beta_1, \beta_2 \) and \( \beta_3 \) represent these ratios, respectively.

Based on the above design, 48 combinations of the parameters are tested on the 12 test problems. The average makespan is selected as the performance measure. The effects of each factor and two-factor interaction are estimated using analysis of variance (ANOVA). The proposed approach has the best performance when \( \beta_1 = \beta_2 = 2.0, \beta_3 = 0.01 \) and \( \rho = 0.2 \). These values are set as the default values in the following experiments.

### 4.3. Results and discussions

Three groups of experiments are conducted to evaluate the proposed approach: (1) evaluating the TBF; (2) comparison with different combinations of the strategies shown in Table 3; and (3) comparison with CPLEX.

All of the experiments are conducted using the 12 test problems, each instance of the test problems is tested using five independent replications and the results are averaged.

#### 4.3.1. Evaluation of the transportation benefit function

TI is compared with the other strategies to evaluate its performance, i.e. TII and TIII, shown in Table 2. Moreover, AI and SI are selected for operation assignment and operation sequencing, respectively.

The difference among TI, TII and TIII lies in that: in TI, if a part arrives at the corresponding destination cell before or at the ETA, the transportation benefit will be the maximum; otherwise, it will decrease; in TII and TIII, however, the best transportation benefit is obtained only at the ETA.

As shown in Figure 2, TI performs best in all test problems, because to obtain the best transportation benefit, in TII and TIII, a part should arrive at the destination cell punctually, whereas in TI, parts can arrive earlier than the ETAs. To relieve the transportation pressure, transportation should be conducted once the transporter is available, even when it is earlier than the ETA. Therefore, the TBF proposed in Equation (23) outperforms other functions.
4.3.2. Comparison with different combinations of other strategies

Different combinations of the strategies shown in Table 3 are comparisons to evaluate the performance of the proposed approach. The gaps showing the percentage deviation between other approaches and the proposed approach are calculated by Equation (33).

$$\text{Gap}_{\text{other strategies}} = \frac{C_{\text{max\_other}} - C_{\text{max\_two\_stage\_ACO}}}{C_{\text{max\_two\_stage\_ACO}}}$$

where $C_{\text{max\_other}}$ and $C_{\text{max\_two\_stage\_ACO}}$ represent the makespan obtained by other approaches and the proposed approach, respectively.

As shown in Table 4, for all the problem sizes, the proposed approach obviously outperforms the other approaches. It is observed that when the PT strategy is adopted for the operation assignment, the gap goes up as the problem size increases, because the PT strategy does not consider the transfer time. However, as the intercell transportation capability is limited, the transfer time makes up a significant proportion in the makespan, especially for medium and large problem sizes. Table 4 also shows that the operation sequencing strategy relatively trivially influences the performance, because the sequence repair strategy changes the initial sequencing results. Since re-scheduling considers the limited intercell transportation capability, the intercell transfer times for most of the operations change, which influences the initial sequencing results. Therefore, the sequence repair strategy is introduced.

To evaluate the performance of the intercell transportation strategies, TI and FB are compared. The gaps in Table 5 show the percentage deviation between the approaches with different intercell transportation strategies, which is calculated by

$$\text{Gap}_{\text{A,B}} = \frac{C_{\text{max\_FB}} - C_{\text{max\_TI}}}{C_{\text{max\_TI}}}$$

where $C_{\text{max\_FB}}$ and $C_{\text{max\_TI}}$ represent the makespan obtained using FB and TI, respectively, as the intercell transportation strategy, with A as the operation assignment strategy and B as the sequencing strategy.

Table 5 shows that for small problem sizes, when the FB strategy is used for intercell transportation, the performance is close to that when the TI strategy is adopted. However, as the problem sizes increase, the gaps are significantly enlarged. This is because for small problem sizes, the capability of the transporters is relatively sufficient, but as the problem size increases, intercell transfers happen more frequently and the transporters become rare resources, therefore
Table 4. Comparison between the two-stage ant colony optimization approach and different combinations of other strategies.

| Test problem | TI + AI + SI | FB + AI + SI | TI + AI + PT | FB + AI + PT | TI + PT + SI | FB + PT + SI | TI + PT + PT | FB + PT + PT | Maximum gap (%) | Average Gap (%) |
|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|----------------|----------------|
| 4P17M4C      | 676.32       | 689.40       | 696.77       | 724.90       | 1136.89      | 1098.43      | 1229.40      | 1340.50      | 98.20          | 46.21          |
| 8P17M4C      | 1026.32      | 1099.20      | 1157.95      | 1061.61      | 1902.90      | 2068.56      | 2141.30      | 2029.70      | 108.64         | 146.27         |
| 11P21M5C     | 1178.70      | 1306.43      | 1216.89      | 1256.20      | 2676.39      | 2830.58      | 2709.74      | 2902.80      | 146.27         | 132.65         |
| 20P21M5C     | 1488.60      | 1593.45      | 1500.81      | 1544.44      | 3720.10      | 3655.53      | 3720.80      | 3679.40      | 149.95         | 71.88          |
| 28P21M5C     | 2117.06      | 2406.80      | 2192.49      | 2585.70      | 4359.20      | 4616.64      | 4575.60      | 4925.30      | 132.65         | 132.65         |
| 36P21M5C     | 1881.78      | 2166.49      | 2020.20      | 2276.60      | 4480.90      | 4618.35      | 4684.60      | 4504.18      | 148.95         | 148.95         |
| 40P25M6C     | 2178.22      | 2579.84      | 2219.68      | 2514.88      | 4564.60      | 4637.90      | 4622.16      | 4470.20      | 112.92         | 70.91          |
| 48P25M6C     | 3359.37      | 4076.70      | 3515.30      | 4171.68      | 7548.35      | 7463.90      | 7742.60      | 7493.30      | 130.48         | 130.48         |
| 56P25M6C     | 3363.74      | 4069.99      | 3428.50      | 4391.10      | 7644.40      | 8019.63      | 7773.30      | 7932.77      | 138.41         | 138.41         |
| 64P25M6C     | 3456.29      | 4617.50      | 3703.96      | 4769.80      | 7473.90      | 7838.36      | 7818.17      | 7940.30      | 129.73         | 129.73         |
| 72P25M6C     | 4368.60      | 5161.38      | 4774.52      | 5283.80      | 10185.60     | 9731.48      | 10208.80     | 9643.15      | 133.69         | 133.69         |
| 80P25M6C     | 3917.43      | 5968.70      | 4263.53      | 5507.86      | 9168.90      | 9260.10      | 9454.39      | 9455.52      | 141.37         | 141.37         |
the parts are usually delayed compared with the ETA. For large problem sizes, the FB strategy takes the maximum number of parts in one batch, leading to an increased number of stop-by cells during the intercell transportation, and thus much longer average delays for all parts in the batch. However, TI considers the effects on other parts caused by adding a part into the current batch, and therefore shortens the average delays overall.

### 4.3.3. Comparison with CPLEX

With no benchmarks available, a commercial optimizer (CPLEX 12.4) is used to obtain optimal solutions as a comparison with the proposed approach.

For the feasibility of the experiments, the performance of the operation sequencing strategy and the intercell transportation strategy are evaluated separately, and the time limit of CPLEX is 6 h.

To evaluate the operation sequencing strategy, the results of operation assignment and intercell transportation obtained using the two-stage ACO approach are preset to CPLEX. The parts that constitute each batch on a transporter are preset, but the start times for transporting the batches are determined by the sequencing results obtained by CPLEX. Similarly, to evaluate the intercell transportation strategy and the operation sequencing, the results are preset to CPLEX.

The performance of the operation sequencing and intercell transportation is shown in Tables 6 and 7, respectively. The gaps in Tables 6 and 7 are calculated by

\[
\text{Gap}_{\text{CPLEX}} = \frac{C_{\text{max, CPLEX}} - C_{\text{max, stageACO}}}{C_{\text{max, stageACO}}} \quad (35)
\]

where \( C_{\text{max, CPLEX}} \) represents the makespan obtained by CPLEX.

With respect to the performance of the operation sequencing strategy, as shown in Table 6, the proposed approach outperforms CPLEX with an average gap of 20.95%, and the computational time of CPLEX reaches the upper limit, whereas the average computational time of the proposed approach is only 56 s. The average gaps between the proposed approach and CPLEX of small-, medium- and large-size problems are 0.5%, 13% and 32%, respectively. It is observed that the advantage of the proposed approach becomes more obvious as the problem dimension increases. Therefore, the proposed approach is an effective and reliable approach for providing near-optimal solutions for the addressed complex combinatorial optimization problem.

With respect to the performance of the intercell transportation strategy, as shown in Table 7, for the smallest size problem, it makes no difference which approach is adopted. Moreover, CPLEX

| Test problem | Gap_{AI,SI} (%) | Gap_{AI,PT} (%) |
|--------------|-----------------|-----------------|
| 4P17M4C      | 1.93            | 4.04            |
| 8P17M4C      | 7.1             | 8.32            |
| 11P21M5C     | 10.84           | 3.23            |
| 20P21M5C     | 7.04            | 2.91            |
| 28P21M5C     | 13.69           | 17.93           |
| 36P21M5C     | 15.13           | 12.69           |
| 40P25M6C     | 18.44           | 13.30           |
| 48P25M6C     | 21.35           | 18.67           |
| 56P25M6C     | 21.00           | 28.08           |
| 64P25M6C     | 33.60           | 28.78           |
| 72P25M6C     | 18.15           | 10.67           |
| 80P25M6C     | 52.36           | 29.19           |
Table 6. Performance of operation sequencing comparison between the two-stage ant colony optimization (ACO) and CPLEX.

| Problem     | CPLEX Makespan | CPLEX Time (h:m:s) | Two-stage ACO Makespan | Two-stage ACO Time (h:m:s) | Gap (%) |
|-------------|----------------|--------------------|------------------------|---------------------------|---------|
| 4P17M4C     | 689            | 0:14:05.2          | 676                    | 0:00:06.8                 | 1.92    |
| 8P17M4C     | 1017           | 6:00:0.0           | 1026                   | 0:00:08.6                 | −0.88   |
| 11P21M5C    | 1132           | 6:00:0.0           | 1214                   | 0:00:13.8                 | −6.76   |
| 20P21M5C    | 1832           | 6:00:0.0           | 1488                   | 0:00:18.0                 | 23.12   |
| 28P21M5C    | 2704           | 6:00:0.0           | 2117                   | 0:00:26.8                 | 27.73   |
| 36P21M5C    | 2096           | 6:00:0.0           | 1881                   | 0:00:30.2                 | 11.43   |
| 40P25M6C    | 2556           | 6:00:0.0           | 2178                   | 0:00:46.3                 | 17.36   |
| 48P25M6C    | 4223           | 6:00:0.0           | 3359                   | 0:01:10.6                 | 25.72   |
| 56P25M6C    | 4622           | 6:00:0.0           | 3363                   | 0:01:30.7                 | 37.44   |
| 64P25M6C    | 4760           | 6:00:0.0           | 3456                   | 0:01:30.5                 | 37.73   |
| 72P25M6C    | 5924           | 6:00:0.0           | 4368                   | 0:02:05.5                 | 35.62   |
| 80P25M6C    | 5521           | 6:00:0.0           | 3917                   | 0:02:18.3                 | 40.95   |
| Average     | –              | 5:31:10.0          | –                      | 0:00:56.5                 | 20.95   |

Table 7. Performance of intercell transportation comparison between the two-stage (ACO) and CPLEX.

| Problem     | CPLEX Makespan | CPLEX Time (h:m:s) | Two-stage ACO Makespan | Two-stage ACO Time (h:m:s) | Gap % |
|-------------|----------------|--------------------|------------------------|---------------------------|-------|
| 4P17M4C     | 676            | 6:00:0.0           | 676                    | 0:00:06.8                 | 0.00% |
| 8P17M4C     | 1083           | 6:00:0.0           | 1026                   | 0:00:08.6                 | 5.56% |
| 11P21M5C    | 1212           | 6:00:0.0           | 1214                   | 0:00:13.8                 | 2.89% |
| 20P21M5C    | –              | –                  | 1488                   | 0:00:18.0                 | –     |
| 28P21M5C    | –              | –                  | –                      | –                         | –     |
| 36P21M5C    | –              | –                  | –                      | –                         | –     |
| 40P25M6C    | –              | –                  | 3917                   | 0:02:18.3                 | –     |
| 48P25M6C    | –              | –                  | –                      | 0:00:56.5                 | –     |
| 56P25M6C    | –              | –                  | –                      | –                         | –     |
| 64P25M6C    | –              | –                  | –                      | –                         | –     |
| 72P25M6C    | –              | –                  | –                      | –                         | –     |
| 80P25M6C    | –              | –                  | –                      | –                         | –     |
| Average     | –              | –                  | –                      | 0:00:56.5                 | –     |

fails to obtain any feasible solutions within 6 h for problem sizes larger than 11P21M5C. In the first three test problems, where CPLEX finds feasible solutions, the average gap between the proposed approach and CPLEX is 2.82%. The average computational time of the proposed approach is only 56 s, compared with 6 h for CPLEX.

5. Conclusions

To solve the intercell scheduling problem considering limited transportation capability, a coordinating mechanism for intercell production and intercell transportation is developed, and a two-stage ACO approach has been proposed to minimize the makespan, where integrated scheduling solutions for the machines and transporters are generated via pre-scheduling and re-scheduling stages. In the coordinating mechanism, the TBF has been presented to evaluate the influence of the transportation dimension on the whole scheduling. Computational experiments were conducted to compare the proposed approach with different transportation strategies and CPLEX. The results show that the proposed TBF outperforms the other strategies. Moreover, the two-stage ACO approach is an effective and reliable approach that provides near-optimal solutions.
Engineering Optimization

Funding

This work is supported by National Natural Science Foundation of China [71401014].

Disclosure statement

No potential conflict of interest was reported by the authors.

Supplemental data

Supplemental data for this article can be accessed at http://dx.doi.org/10.1080/0305215X.2016.1150470.

References

Blum, C. 2002. “ACO Applied to Group Shop Scheduling: A Case Study on Intensification and Diversification.” In Ant Algorithms. Lecture Notes in Computer Science. Vol. 2463: 14–27. Berlin: Springer.  
Garza, O., and T. L. Smunt. 1991. “Countering the Negative Impact of Intercell Flow in Cellular Manufacturing.” Journal of Operations Management 10 (1): 92–118.  
Gendreau, M., and J. Y. Potvin. 2005. “Metaheuristics in Combinatorial Optimization.” Annals of Operations Research 140 (1): 189–213.  
Golmohammadi, A., and R. Ghodsi. 2009. “Applying an Integer Electromagnetism-Like Algorithm to Solve the Cellular Manufacturing Scheduling Problem with an Integrated Approach.” Computers & Industrial Engineering 2009. CIE 2009. International Conference on 2009: 34–39.  
Halat, K., and R. Bashirzadeh. 2015. “Concurrent Scheduling of Manufacturing Cells Considering Sequence-Dependent Family Setup Times and Intercellular Transportation Times.” The International Journal of Advanced Manufacturing Technology 77 (9–12): 1907–1915.  
Huang, K. L., and C. J. Liao. 2008. “Ant Colony Optimization Combined with Taboo Search for the Job Shop Scheduling Problem.” Computers & Operations Research 35 (4): 1030–1046.  
Johnson, D. J., and U. Wemmerlöv. 2009. “Why Does Cell Implementation Stop? Factors Influencing Cell Penetration in Manufacturing Plants.” Production and Operations Management 13 (3): 272–289.  
Kim, D. B., and H. Hwang. 2001. “A Dispatching Algorithm for Multiple-Load AGVS Using a Fuzzy Decision-Making Method in a Job Shop Environment.” Engineering Optimization 33 (5): 523–547.  
Li, D., Y. Wang, G. Xiao, and J. Tang. 2013. “Dynamic Parts Scheduling in Multiple Job Shop Cells Considering Intercell Moves and Flexible Routes.” Computers & Operations Research 40 (5): 1207–1223.  
Ma, Y., and J. Xu. 2015. “A Cloud Theory-Based Particle Swarm Optimization for Multiple Decision Maker Vehicle Routing Problems with Fuzzy Random Time Windows.” Engineering Optimization 47 (6): 825–842.  
Mousavi, S. M., R. Tavakkoli-Moghaddam, and F. Jolai. 2013. “A Possibilistic Programming Approach for the Location Problem of Multiple Cross-Docks and Vehicle Routing Scheduling under Uncertainty.” Engineering Optimization 45 (10): 1223–1249.  
Pillac, V., M. Gendreau, C. Guéret, and A. L. Medaglia. 2013. “A Review of Dynamic Vehicle Routing Problems.” European Journal of Operational Research 225 (1): 1–11.  
Saidi-Mehrabad, M., S. Dehnavi-Arani, F. Evazabadian, and V. Mahmoodian. 2015. “An Ant Colony Algorithm (ACA) for Solving the New Integrated Model of Job Shop Scheduling and Conflict-Free Routing of AGVs.” Computers & Industrial Engineering 86: 2–13.  
Solimanpur, M., and A. Elmi. 2013. “A Tabu Search Approach for Cell Scheduling Problem with Makespan Criterion.” International Journal of Production Economics 141 (2): 639–645.  
Solimanpur, M., P. Vrat, and R. Shankar. 2004. “A Heuristic to Minimize Makespan of Cell Scheduling Problem.” International Journal of Production Economics 88 (3): 231–241.  
Tang, J., X. Wang, I. Kaku, and K. L. Yung. 2010. “Optimization of Parts Scheduling in Multiple Cells Considering Intercell Move Using Scatter Search Approach.” Journal of Intelligent Manufacturing 21 (4): 525–537.  
Tavakkoli-Moghaddam, R., N. Javadian, A. Khorrani, and Y. Ghollipour-Kanani. 2010. “Design of a Scatter Search Method for a Novel Multi-criteria Group Scheduling Problem in a Cellular Manufacturing System.” Expert Systems with Applications 37 (3): 2661–2669.  
Wang, C., D. Mu, F. Zhao, and J. W. Sutherland. 2015. “A Parallel Simulated Annealing Method for the Vehicle Routing Problem with Simultaneous Pickup–Delivery and Time Windows.” Computers & Industrial Engineering 83: 111–122.  
Youssef, Y. A., Y. Beauchamp, and M. Thomas. 1994. “Comparison of a Full Factorial Experiment to Fractional and Taguchi Designs in a Lathe Dry Turning Operation.” Computers & Industrial Engineering 27 (1): 59–62.  
Zeng, C., J. Tang, and C. Yan. 2014. “Job-Shop Cell-Scheduling Problem with Inter-cell Moves and Automated Guided Vehicles.” Journal of Intelligent Manufacturing 26 (5): 845–859.