Journal of Zhejiang University-SCIENCE A (Applied Physics & Engineering)
ISSN 1673-565X (Print); ISSN 1862-1775 (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

Solving composite scheduling problems using the hybrid genetic algorithm*

Azuma OKAMOTO, Mitsumasa SUGAWARA

(*Faculty of Software and Information Science, Iwate Prefectural University, 152-52 Sugo, Takizawa, Iwate, Japan*)

E-mail: lfo@iwate-pu.ac.jp; sugawara@iwate-pu.ac.jp

Received Oct. 28, 2010; Revision accepted Oct. 29, 2010; Crosschecked Oct. 29, 2010

Abstract: This paper dealt with composite scheduling problems which combine manufacturing scheduling problems and/or transportation routing problems. Two scheduling models were formulated as the elements of the composite scheduling model, and the composite model was formulated composing these models with indispensable additional constraints. A hybrid genetic algorithm was developed to solve the composite scheduling problems. An improved representation based on random keys was developed to search permutation space. A genetic algorithm based dynamic programming approach was applied to select resource. The proposed technique and a previous technique are compared by three types of problems. All results indicate that the proposed technique is superior to the previous one.

Key words: Composite scheduling, Manufacturing scheduling, Transportation routing, Hybrid genetic algorithm
doi:10.1631/jzus.A1001136 **Document code:** A **CLC number:** TP301.6; U11; F406.2

1 Introduction

The composite scheduling problems are composed of manufacturing scheduling problems and/or transportation routing problems. In contemporary manufacturing, meeting promised delivery dates requires production schedules that take into account elements such as transportation. Previously, scheduling focused on improvements in facility workloads and reduction to lower costs through mass production. Traditional optimization techniques for scheduling typically consider only the manufacturing constraints, and optimization techniques for transportation routing consider only the transportation constraints. These traditional techniques cannot provide schedules that accommodate customer demands for just-in-time delivery.

It is difficult to develop optimization techniques for composite scheduling problems. Most of the studies concerning scheduling, which consider both manufacturing and transportation, have established a number of resource and transportation restrictions to systematize the problem from the viewpoint of complexity. These approaches offer only specialized solutions (Lee and Chen, 2001; Soukhal *et al.*, 2005).

The meta-heuristics approaches are applicable to solve such complex problems. Moon *et al.* (2004) proposed an advanced process planning and scheduling model. In this model, the transportation time between machine resources is considered, but the constraint of a number of transportation resources such as vehicle is not considered. Okamoto *et al.* (2006a) expanded the model by integrating the manufacturing process and transportation between plants and proposed a solution based on the genetic algorithm. Okamoto *et al.* (2006b) proposed a new integrated scheduling problem that considers the transportation resource. A genetic algorithm is also proposed for the problem. To address these concerns, an improved random number generation method was developed by Okamoto *et al.* (2009).

* Project supported by the Grant-in-Aid for Young Scientists (B) from the Ministry of Education, Culture, Sports, Science and Technology, Japan
© Zhejiang University and Springer-Verlag Berlin Heidelberg 2010

2 Mathematical models

First, two scheduling models were formulated as the elements of the composite scheduling model. Then, the composite model was formulated by composing them with indispensable additional constraints.

2.1 Modeling for manufacturing scheduling problem

The manufacturing scheduling problem is an extension of the job-shop scheduling problem. In this problem, the precedence constraints (r_{kij}) can be given for any set of operation (o_{ki}, o_{kj}) in each job (O_k), and each operation (o_{ki}) can be processed in a given processing time (p_{kim}) by any machine (M_m) from a given set. The difficulty occurs when assigning each operation to a machine and when ordering the operations on the machines, such that the maximal completion time ($c_{\max}$) of all operations is minimized.

The objectives and constraints are as follows:

$$\min c_{\max} = \max_{i,k}\{c_{ki}\}, \tag{1}$$

$$\text{s. t. } (c_{kj} - p_{kjm} - c_{ki}) r_{kij} x_{kjm} \geq 0, \forall i, j, k, \tag{2}$$

$$\left(\max\{c_{ki} - p_{ki}, c_{lj} - p_{lj}\} - \min\{c_{ki}, c_{lj}\}\right) x_{kim} x_{ljm} \geq 0, \quad \forall (k,i), (l,j), m, (k,i) \neq (l,j), \tag{3}$$

$$(c_{ki} - p_{kim}) x_{kim} \geq 0, \ \forall i, k, m, \tag{4}$$

$$\sum_{m=1}^{N} x_{kim} = 1, \ \forall (k,i), \tag{5}$$

$$x_{kim} \in \{0,1\}, \ \forall i, k, m, \tag{6}$$

where k, j are the indices of jobs, k, j=1, 2, …, L, L is the total number of jobs; i, j are the indices of operations, i, j=1, 2, …, J_k, J_k is the total number of operations for O_k, O_k is the order (job) k; m, n are the indices of machines, m, n=1, 2, …, N, N is the total number of machines; $c_{\max}$ is the maximal completion time (makespan); c_{ki} is the completion time of o_{ki}, o_{ki} is the operation i of O_k; x_{kim} is the machine (M_m) assignment to o_{ki}, M_m is the machine m, and

$$x_{kim} = \begin{cases} 1, & \text{if } o_{ki} \text{ is assigned to } M_m, \\ 0, & \text{otherwise;} \end{cases}$$

p_{kim} is the processing time of o_{ki} on M_m, r_{kij} is the precedence constraint between two operations, and

$$r_{kij} = \begin{cases} 1, & \text{if } o_{ki} \text{ precedes } o_{kj}, \\ 0, & \text{otherwise.} \end{cases}$$

Eq. (1) is the objective function that minimizes the makespan. Eq. (2) indicates precedence constraints. Eq. (3) means that multiple operations cannot be processed by the same machine simultaneously. Eq. (4) shows that all machines cannot start at negative time. Eqs. (5) and (6) ensure that only one machine is assigned to a process.

2.2 Modeling for transportation scheduling problem

The transportation scheduling problem is an extension of the vehicle routing problem. In this problem, each task (T_e) consists two activities: pickup (T_e^{P}) and delivery (T_e^{D}). Each task can be assigned by any vehicle V_v from a given set. Generally, vehicles spend time for transportation (d_{mn}), since different locations (M_m, M_n) have been assigned in activities ($a_{em}^{\mathrm{P}} = 1$, $a_{en}^{\mathrm{D}} = 1$, $m \neq n$). Not only does a vehicle move directly between locations for activities of a task, but it may also pass a location for other task. All vehicles (V_1, V_2, …, V_W) start from the depot, execute transportation tasks, and then return to the depot. The problem is to assign each task to a vehicle and to order the activities on the vehicles, such that the maximal completion time ($t_{\max}$) of all tasks is minimized. Still, the number of active vehicles (w) is also minimized.

The objectives and constraints are as follows:

$$\min t_{\max} = \max_{e,m}\left\{\left(t_e^{\mathrm{D}} + d_m\right) a_{em}\right\}, \tag{7}$$

$$\min w = \sum_{v=1}^{W}\left(\max\left\{\sum_{e=1}^{F} y_{ev}, 1\right\}\right), \tag{8}$$

$$\text{s. t. } t_e^{\mathrm{D}} - t_e^{\mathrm{P}} \geq 0, \forall e, \tag{9}$$

$$\left(| t_f^h - t_e^g | - d_{mn}\right) a_{em}^g a_{en}^h y_{ev} y_{fv} \geq 0, \quad \forall (e,g), (f,h), m, n, v, \tag{10}$$

$$\left(t_e^{\mathrm{P}} - d_m\right) a_{em}^{\mathrm{P}} \geq 0, \forall e, m, \tag{11}$$

$$\sum_{v=1}^{W} y_{ev} = 1, \forall e, \tag{12}$$

$$y_{ev} \in \{0,1\}, \forall e, v, \quad (13)$$

where v is the index of vehicles, v=1, 2, …, W, W is the total number of vehicles; e, f are the indices of pickup and delivery tasks, respectively, e, f=1, 2, …, F, F is the total number of tasks; g, h are the indices of pickup and delivery activities, respectively, g, $h \in \{P, D\}$; T_e^g is the pickup/delivery activity (g) of task e, d_m is the transportation time from/to depot to/from M_m, a_{em}^g is the location assignment of T_e^g, and

$$a_{em}^g = \begin{cases} 1, & \text{if } T_e^g \text{ located at } M_m, \\ 0, & \text{otherwise.} \end{cases}$$

t_e^g is the pickup/delivery time for T_e^g, y_{ev} is the vehicle (V_v) assignment to T_e, and

$$y_{ev} = \begin{cases} 1, & \text{if } T_e \text{ is assigned to } V_v, \\ 0, & \text{otherwise.} \end{cases}$$

Eq. (7) is the objective function that minimizes the makespan. Eq. (8) is the objective function that minimizes the number of active vehicles. Eq. (9) means that a vehicle must execute pickup activity in advance to delivery activity. Eq. (10) indicates that vehicles spend time for transportation between different locations. Eq. (11) shows that all vehicles cannot start from the depot at a negative time. Eqs. (12) and (13) ensure that only one vehicle is assigned to a task.

2.3 Mathematical model for composite scheduling

To compose multiple homogeneous scheduling problems, the model may not be changed, and the job/task sets of the problems may be simply merged.

In this section, to compose heterogeneous scheduling models, additional constraints which express how to compose are necessary. The following model is composition of production scheduling model and transportation scheduling model. The vehicle picks up a product of an operation from the assigned machine/location, and transports it to machine/location of the next operation to consume as material.

Each transportation task (T_e) is mapped to a set of operations (o_{ki}, o_{kj}) which have precedence constraints (r_{kij}=1). The pickup activity (T_{kij}^{P}) of the task can execute when the predecessor operation (o_{ki}) of the set is completed; the delivery activity (T_{kij}^{D}) must complete before starting the successor operation (o_{kj}).

The objectives and constraints are as follows:

$$\min t_{\max} = \max_{i,j,k,m} \left\{ \left(t_{kij}^{\mathrm{D}} + d_m \right) r_{kij} x_{kjm} \right\}, \quad (14)$$

$$\min w = \sum_{v=1}^{W} \left(\max \left\{ \sum_{k=1}^{L} \sum_{i=1}^{J_k} \sum_{j=1}^{J_k} y_{kijv}^{\mathrm{P}}, 1 \right\} \right), \quad (15)$$

$$\text{s. t. } \left(t_{kij}^{\mathrm{D}} - t_{kij}^{\mathrm{P}} \right) r_{kij} \geq 0, \forall i, j, k, \quad (16)$$

$$\left(| t_{li'j'}^{\mathrm{P}} - t_{kij}^{\mathrm{P}} | - d_{mn} \right) r_{kij} r_{li'j'} x_{kim} x_{li'n} y_{kijv} y_{li'j'v} \geq 0, \quad \forall i, j, k, i', j', l, m, n, v, \quad (17)$$

$$\left(\left| t_{li'j'}^{\mathrm{D}} - t_{kij}^{\mathrm{P}} \right| - d_{mn} \right) r_{kij} r_{li'j'} x_{kim} x_{lj'n} y_{kijv} y_{li'j'v} \geq 0, \quad \forall i, j, k, i', j', l, m, n, v, \quad (18)$$

$$\left(| t_{li'j'}^{\mathrm{D}} - t_{kij}^{\mathrm{D}} | - d_{mn} \right) r_{kij} r_{li'j'} x_{kjm} x_{lj'n} y_{kijv} y_{li'j'v} \geq 0, \quad \forall i, j, k, i', j', l, m, n, v, \quad (19)$$

$$\left(t_{kij}^{\mathrm{P}} - d_m \right) r_{kij} x_{kim} \geq 0, \forall i, j, k, m, \quad (20)$$

$$\sum_{v=1}^{W} y_{kijv} = r_{kij}, \forall i, j, k, \quad (21)$$

$$y_{kijv} \in \{0,1\}, \forall i, j, k, g, v, \quad (22)$$

$$\left(t_{kij}^{\mathrm{P}} - c_{ki} \right) r_{kij} \geq 0, \forall i, j, k, \quad (23)$$

$$\left(c_{kj} - p_{kjm} - t_{kij}^{\mathrm{D}} \right) r_{kij} x_{kjm} \geq 0, \forall i, j, k, m, \quad (24)$$

where T_{kij} is the transportation task for product of o_{ki} as material of o_{kj}, T_{kij}^g is the pickup/delivery activity for product of o_{ki} and material of o_{kj}, t_{kij}^g is the pickup/delivery time for T_{kij}^g, y_{kijv} is the vehicle (V_v) assignment to T_{kij}, and

$$y_{kijv} = \begin{cases} 1, & \text{if } T_{kij} \text{ is assigned to } V_v, \\ 0, & \text{otherwise.} \end{cases}$$

Eqs. (23) and (24) indicate additional constraints to compose production scheduling and transportation scheduling. The rests of all equations are the same as the original model or replacements from original model. Objective functions Eq. (14) and Eq. (15)

are replacements of Eqs. (7) and (8), respectively. Eq. (16) is a replacement of Eq. (9). Eqs. (17), (18), and (19) are replacements of Eq. (10). Eqs. (20), (21), and (22) are replacements of Eqs. (11), (12), and (13), respectively.

3 Hybrid genetic algorithm

A hybrid genetic algorithm is developed to solve the composed scheduling problems. Since decision variables of two problems (c_{ki}, t^g_{kij}) constrain to each other in Eqs. (23) and (24), it is difficult to be solved by decomposing to original problem.

The hybrid genetic algorithm has two parts employing a new approach. Firstly, a sequence of operations must be made. An encoding using improved random keys is developed for this part. Secondly, the starting/completion time of operations is obtained by the sequence generated by assigning resources for each operation.

3.1 Sequencing operations

An improved representation based on random key (Bean, 1994) is used for the operation sequence. A random keys vector represents the sequence (or a priority list) of all operations, but the strength of the characteristics of each sequence is different. Therefore, even if a random keys vector that represents a good sequence is generated, the characteristics might not be succeeded to the next generation.

For example, a sequence vector $[o_1, o_2, o_3]$= [0,2,1] represents a sequence. All of the sequence vectors are "permutations". If the vector length is n, the permutation can be mapped $(n-1)$-dimensional polytope (known as Permutohedron) in an n-dimensional space. The random keys vector (for example, $[x_1, x_2, x_3]$) in a random keys space (Fig. 1a) can be mapped to a sequence space (Fig. 1b).

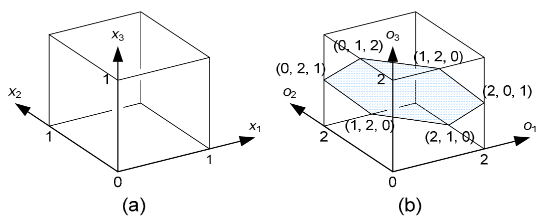

Fig. 1 Random key space (a) and sequence space (b)

In the improved representation, a random keys vector $[x_1, x_2, \ldots, x_n]$ is normalized before crossover operation by the following procedures:

1. The initial random key is generated in the range of $-1 \le x_i \le 1$.
2. To map $(n-1)$-dimensional polytope, in other words, to keep $\sum_{i=1}^{n} x_i = 0$, update $x_k \leftarrow x_k - \frac{1}{n}\sum_{i=1}^{n} x_i$.
3. To normalize vector length, in other words, to keep $\sum_{i=1}^{n} x_i^2 = X$ (X is a constant), update $x_k \leftarrow x_k \sqrt{X \big/ \sum_{i=1}^{n} x_i^2}$.

3.2 Selecting resources

To generate a better combination of selected resources, our approach combines resource assignment for each operation from two parents. If the result of tentative scheduling is a scalar value, the dynamic programming can be applied simply. However, the result of scheduling contains a set of completion times for each resource; it is not one scalar value.

So a genetic algorithm based dynamic programming approach is applied for this part. This approach is very simple. The representation is a simply set of resources for each operation. The uniform crossover is used to generate candidates of resource assignment.

3.3 Genetic algorithm

Overall algorithm is based on Bean (1994)'s algorithm (Fig. 2). Immigration, meaning the generation of new members of the population at each new generation, is used instead of the traditional mutation operator.

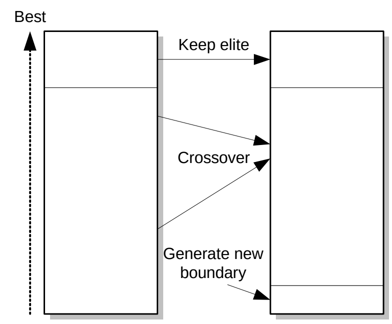

Fig. 2 Generation model based on Bean (1994)'s algorithm

The boundary of search space is actually shown by the precedence constraint, though the initial search space has no boundary (Fig. 3). By repeating an arithmetic-type crossover for real number representation of parents (P in Fig. 3), the boundary search space becomes difficult to search by generated new child (C in Fig. 3). To avoid this, an operation sequence at search space boundary in the immigration should be generated (Fig. 4).

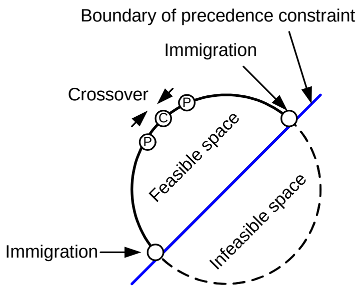

Fig. 3 Characteristics of search space

```
function edgeSeq(O: set of operations): Sequence
begin
  S: sequence←randomOrder{o|pred(o)=∅; o∈O};
  R: set←O;
  while R≠∅, do
  begin
    repeat
      z: operation←shift(S);
    until z∈R
    while pred(z)∩R≠∅, do
    begin
      z←randomSelect(pred(z)∩R);
    end
      edgeSeq←[edgeSeq, z];
      R←R−{z};
      S←[randomOrder(succ(z)), S];
  end
end
```

Fig. 4 Pseudo code to generate operation sequence at search space boundary

4 Numerical experiments

The proposed technique and previous one (Okamoto *et al.* 2006b), without a local search procedure, are compared by three types of problems: manufacturing scheduling problem (M), transportation routing problem (T), and composite problem (C). Three different sizes (the number of operations) of problem datasets are prepared for each type of problem.

Each experiment runs 100 times. Table 1 shows the rate that each algorithm found the best (known) solution. All results indicate that the proposed technique is better than the previous one. Originally, the previous technique was improved by adding a local search procedure, and it is possible to add it even by the proposed technique.

Table 1 Results of numerical experiments

Type	Size	Rate	
		Proposed technique (%)	Okamoto *et al.* (2006b) (%)
M	10	100	100
M	20	100	100
M	50	100	94
T	20	100	100
T	40	100	95
T	100	82	73
C	–30*	100	96
C	–60*	92	67
C	–150*	74	51

* The number of transportation operations could be changed

5 Conclusions

The composite scheduling problem was formulated. To solve this problem, the solution space to search consists of operation sequences and resource selections as well as manufacturing, scheduling, and transportation routing problems. Two new approaches were developed to search each solution space. The proposed technique produces better results than the previous one based on the numerical experiment performed.

Both previous and proposed techniques are separated to search operation sequence space and resource selection space. It contributes to the simplification of the technique; however, these solution spaces are actually interdependent. In a future study, a reasonable solution space that combined both solution spaces will be developed for a more efficient search.

References

Bean, J.C., 1994. Genetics and random keys for sequencing and optimization. *ORSA Journal on Computing*, **6**:154-160.

Lee, C.Y., Chen, Z.L., 2001. Machine scheduling with transportation considerations. *Journal of Scheduling*,

4(1):3-24. [doi:10.1002/1099-1425(200101/02)4:1<3::AID-JOS57>3.0.CO;2-D]

Moon, C., Kim, J.S., Gen, M., 2004. Advanced planning and scheduling based on precedence and resource constraints for e-plant chains. *International Journal of Production Research*, **42**(15):2941-2955. [doi:10.1080/002075404 10001691956]

Okamoto, A., Gen, M., Sugawara, M., 2006a. Integrated data structure and scheduling approach for manufacturing and transportation using hybrid multistage operation-based genetic algorithm. *Journal of Intelligent Manufacturing*, **17**(4):411-421. [doi:10.1007/s10845-005-0014-9]

Okamoto, A., Gen, M., Sugawara, M., 2006b. Integrated scheduling problem of manufacturing and transportation with pickup and delivery. *International Journal of Logistics and SCM Systems*, **1**(1):19-27.

Okamoto, A., Gen, M., Sugawara, M., 2009. Integrated scheduling using genetic algorithm with quasi-random sequences. *International Journal of Manufacturing Technology and Management*, **16**(1/2):147-165. [doi:10.1504/IJMTM.2009.021507]

Soukhal, A., Oulamara, A., Martineau, P., 2005. Complexity of flow shop scheduling problems with transportation constraints. *European Journal of Operational Research*, **161**(1):32-41. [doi:10.1016/j.ejor.2003.03.002]

Journals of Zhejiang University-SCIENCE (A/B/C)

Latest trends and developments

These journals are among the best of China's University Journals. Here's why:

- *JZUS (A/B/C)* have developed rapidly in specialized scientific and technological areas.
  JZUS-A (Applied Physics & Engineering) split from JZUS and launched in 2005
  JZUS-B (Biomedicine & Biotechnology) split from JZUS and launched in 2005
  JZUS-C (Computers & Electronics) split from JZUS-A and launched in 2010

- We are the first in China to completely put into practice the international peer review system in order to ensure the journals' high quality (more than 7600 referees from over 60 countries, http://www.zju.edu.cn/jzus/reviewer.php)

- We are the first in China to pay increased attention to Research Ethics Approval of submitted papers, and the first to join **CrossCheck** to fight against plagiarism

- Comprehensive geographical representation (the international authorship pool enlarging every day, contributions from outside of China accounting for more than 46% of papers)

- Since the start of an international cooperation with Springer in 2006, through SpringerLink, *JZUS*'s usage rate (download) is among the tops of all of Springer's 82 co-published Chinese journals

- *JZUS*'s citation frequency has increased rapidly since 2004, on account of DOI and Online First implementation (average of more than 60 citations a month for each of *JZUS-A* & *JZUS-B* in 2009)

- *JZUS-B* is the first university journal to receive a grant from the National Natural Science Foundation of China (2009–2010)