대한산업공학회/한국경영과학회 2000 춘계공동학술대회 논문집. 2000년 4월 21-22일. 경남대학교
Session OC7.5

통합적인 공정순서와 가공기계 선정을 위한 유전 알고리즘
Genetic Algorithm for Integrated Process Sequence and Machine Selection

문치웅*, 서윤호**, 이영해*, 최경현*

* 한양대학교 산업공학과
** 울산대학교 산업공학부

Abstract

The objective of this paper is to develop a model to integrate process planning and resource planning through analysis of the machine tool selection and operations sequencing problem. The model is formulated as a travelling salesman problem with precedence relations. To solve our model, we also propose an efficient genetic algorithm based on topological sort concept.

1. INTRODUCTION

Process planning is an off-line planning task to specify the technological requirements necessary to transform a workpiece from its raw material state to the final product state as specified in the product design. Process plans play a key role to realize the workpieces in a specific manufacturing system by specifying the required processes, machine tools selection, operation sequence, machining conditions (*i.e.*, speed, feed, depth of cut, *etc.*), requirements for tools and auxiliary devices, etc. Since process plans play a key role in succeeding stages of production, it is impossible to attain total efficiency of production with the process plans designed without simultaneously or interactively considering the other stages of production, *i.e.*, resource planning, production planning and control, *etc*. For example, a process plan designed without considering alternative resources for a process cannot offer the most efficient means for dynamic batch production in which product mix and their lot sizes change .

Also, in a dynamic batch production environment in which part mix and their production quantities changes from one production session and session, if the same process plan is used over and over for the manufacturing of a particular part, the efficiency and flexibility of the manufacturing system can be devastated of [3].

The general objective of this research is to develop a model to integrate process planning and resource planning through analysis of the machine tool selection and operations sequencing problem. The integration is intended to provide a flexible structure in which process plan can be concurrently designed with the other functions of production. To solve our model, we also propose an efficient genetic algorithm (GA) based on topological sort (TS) concept.

2. Problem Description

In a dynamic batch production environment in which several part types are simultaneously produced in small lots, decisions on machine tool selection and machine visiting sequence should be made whenever part mix configuration in a shop floor changes. Also, for a given production order - part mix and their lot sizes, the process plan for each part type should be designed for the maximum efficiency of production of all the parts, rather than choosing the best process plan for each part type without regard to shop workload.

From an unordered set of machining processes with precedence relations, operations sequencing problem in domain of this problem is to determine a sequence by exploiting a sequence space derived from the combination of parallel processes. Let the transition cost between two operations be expressed as the sum of total machining time of the parts on a machine for the first operation and transportation time of the parts from the first machine to a machine for the second operation. Since the transition cost is sequence dependent, with given transitional costs between all the operations, the operation sequencing problem can be formulated as a well-known Traveling Salesman Problem (TSP) [4].

The transition time between operations can be calculated only with a machine tool assigned for each operation. On the other hand, selecting machine tools for each operation is not trivial since multiple machine tools for each operation can be considered as candidate and each of them resuls in the various performance measures including total machining and transportation times. Therefore, the operation sequencing problem can be formulated as a TSP which solve a subproblem - the machine tool selection problem - to calculate the cost of the trips. Moreover, since each TSP determines the operation sequence for each part type, multiple TSP's corresponding to the number of part types in part mix should be considered simultaneously.

The integrated machine tool selection and process sequencing problem in this paper is defined as follows: given a production order - part mix to be simultaneously manufactured in a shop and their production lot sizes, an unordered set of machining functions with precedence constraints, multiple machine tools for each machining process, machining times on all alternate machining tools for each machining process, and transportation times for all pairs of machine tools, determine a set of machine tool visiting sequences for all part types, such that the total production time (*i.e.*, cost) for the production order is minimized and workloads among machine tools are balanced.

3. Model

To formulate a TSP with precedence constraints, the two-commodity network flow model can be used [1]. Suppose that there are two distinct commodities p and q given in the network with J nodes or cities. While commodity p is supplied by $(J-1)$ units at a selected starting node and used by one unit at each node that is not the starting node, q is a commodity to consume $(J-1)$ units at the starting node and to be supplied by one unit at the other nodes. Such network flow of the commodities are characterized by two properties: First, the sum of commodities p and q in any feasible tour should be equal to $J-1$. And, the quantity of commodity p (or q) outbounded from a node is decreasing as the tour proceeds. These properties are used to model the precedence relations for a constraint case TSP. Any more details on this model can be referred to in the papers .

The following notations are used to describe the problem through whole paper:

- $\boldsymbol{K}$ product mix, set of K different part types, *i.e.*, $\boldsymbol{K} = \{1, 2,\cdots, k,\cdots, K\}$.
- $\boldsymbol{M}$ set of machine tools, $\boldsymbol{M} = \{1, 2,\cdots, m,\cdots, M\}$.
- $\boldsymbol{q}$ set of lot sizes for part types in product mix, *i.e.*, $\boldsymbol{q} = \{q_1, q_2,\cdots, q_k,\cdots, q_K\}$.
- $\boldsymbol{G_k}$ set of operations for part type k, *i.e.*, $$\boldsymbol{G_k} = \left\{ g_{ki} \,\middle|\, \forall i = 1, 2,\cdots, J_k \right\},$$ where g_{ki} is operation name of the ith element and J_k is the number of operations, $J_k = |\boldsymbol{G_k}|$.
- p_{kim} processing time of ith operation of part type k on machine m.
- τ_{mn} transportation time of AGV from machine m to n.
- u_{kij} unit load size of part k from operation g_{ki} to g_{kj}, the number of parts in a unit load.
- s_k first selected operation for part type k.

Operations sequencing problem is defined as: given part mix $\boldsymbol{K}$ and their lot sizes $\boldsymbol{q}$, sets of unordered machining operations $\boldsymbol{G_k}$ with precedence constraints for all part types $k \in \boldsymbol{K}$, determine the operation sequences for all part types such that the total transition cost for all part types is minimized. Since a single machine is fixed for each operation in this problem, a notation $\mu(k, i)$ is used to denote the machine assigned to ith operation of part type k.

Let c_{kij} be transition time from operation g_{ki} to g_{kj} for q_k parts of type k. The transition time c_{kij} is defined as the sum of total processing time on a assigned machine $\mu(k, i)$ for operation g_{ki} for q_k parts and total transportation time from the machine $\mu(k, i)$ to machine $\mu(k, j)$ for q_k parts. With known unit load size u_{kij}, the number of transportation v_{kij} from operation g_{ki} to g_{kj} for q_k parts can be calculated from the following Eq. (1).

The expression $\lceil x \rceil$ means the smallest integer greater than the x.

$$v_{kij} = \left\lceil \frac{q_k}{u_{kij}} \right\rceil \tag{1}$$

Then the transition time c_{kij} can be expressed as follows:

$$c_{kij} = q_k p_{ki\mu(k,i)} + v_{kij} \tau_{\mu(k,i)\mu(k,j)}, \tag{2}$$

Three variables are introduced to adapt the two-commodity network flow model as follows:

y_{kij}^{p} quantity of commodity p from operation g_{ki} to g_{kj} for part type k.

y_{kij}^{q} quantity of commodity q from operation g_{ki} to g_{kj} for part type k.

$$y_{kij} = \begin{cases} 1, \text{ if operation } i \text{ is performed immediately after operation } j \text{ for part type } k, \text{ and} \\ 0, \text{ otherwise,} \end{cases}$$

$$x_{kim} = \begin{cases} 1, \text{ if machine tool } m \text{ is selected for operation } i \text{ of part type } k, \\ 0, \text{otherwise.} \end{cases}$$

With the introduction of variable, the transition time from operation i to j for q_k parts, c_{kij} can be redefined as follows:

$$c_{kij} = \sum_{m=1}^{M} \sum_{n=1}^{M} \{q_k p_{kim} x_{kim} + v_{kij} \tau_{mn} x_{kim} x_{kjn}\} \quad (3)$$

If the decisions on operations sequencing and machine selection are made in only way to minimize the objective function, balanced workloads among all machine tools cannot be attained. To deal with workload balancing constraint, let W_m be the maximum allowable machining time for machine m. Then the workload balancing constraint can be expressed as total machining time assigned for each machine m must not excess the maximum allowable time W_m. In addition, the constraints can be used to control the workload for each machine tool by changing W_m. Then, the complete zero-one integer programming model for the integrated operations sequencing and machine selection problem is as summarized below:

Minimize

$$\sum_{k=1}^{K} \sum_{i=1}^{J_k} \sum_{\substack{j=1 \\ i \neq j}}^{J_k} \sum_{m=1}^{M} \sum_{n=1}^{M} \frac{1}{J_k - 1} \{q_k p_{kim} x_{kim} + v_{kij} \tau_{mn} x_{kim} x_{kjn}\} (y_{kij}^{p} + y_{kij}^{q}) \quad (4)$$

$$\sum_{j=1}^{J_k} y_{kij}^{p} - \sum_{j=1}^{J_k} y_{kji}^{p} = \begin{cases} J_k - 1, \text{for } i = s_k, \\ -1, \text{elsewhere,} \end{cases} \quad \forall k, \quad (5)$$

$$\sum_{j=1}^{J_k} y_{kij}^{q} - \sum_{j=1}^{J_k} y_{kji}^{q} = \begin{cases} -(J_k - 1), \text{for } i = s_k, \\ +1, \quad \text{elsewhere,} \end{cases} \quad \forall k, \quad (6)$$

$$\sum_{j=1}^{J_i} (y_{kij}^{p} + y_{kij}^{q}) = J_k - 1 \qquad \forall k \text{ and } i, \quad (7)$$

$$y_{kij}^{p} + y_{kij}^{q} = (J_k - 1) y_{kij} \qquad \forall k, i \text{ and } j, \quad (8)$$

$$\sum^{J_k} y^{p}_{kuj} - \sum^{J_k} y^{p}_{kvj} \geq 1, \qquad \forall k \text{ and } (g_{ku} \rightarrow g_{kv})(g_{kv} \neq s_k) \quad (9)$$

$$\sum_{k=1}^{K} \sum_{i=1}^{J_k} q_k p_{kim} x_{kim} \leq \overline{W}_m, \qquad \forall m, \quad (10)$$

$$\sum_{m=1}^{M} x_{kim} = 1 \qquad \forall k \text{ and } i, \quad (11)$$

$$y_{kij}^{p} \geq 0, \qquad \forall k, i \text{ and } j, \quad (12)$$

$$y_{kij}^{q} \geq 0, \qquad \forall k, i \text{ and } j, \quad (13)$$

$$x_{kim} \in \{0,1\}, y_{kij} \in \{0,1\} \qquad \forall k, i, j \text{ and } m, \quad (14)$$

The objective function (4) expresses total transition times for all part types, since the sum of commodities p and q between operation g_{ki} to g_{kj} on any feasible operations sequence (*i.e.*, y_{kij} = 1) is equal to J_k - 1 (*i.e.*, y_{kij}^{p} + y_{kij}^{q} = J_k - 1). The constraints (5) and (12) are used to ensure the feasibility of network flow of commodity p. Similarily, constraints (6) and (13) are expressed for commodity q. Constraint (7) ensures the feasible tour, *i.e.*, feasible operations sequence. Constraint (8) explains that if y_{kij} =1 the sum of commodities p and q between g_{ki} and g_{kj} be J_k - 1. Constraint (9) explains any precedence relationship between machining operations. Constraint (10) controls the workloads among machine tools. Constraint (11) ensures that only one machine tool for each operation should be selected. Constraint (14) ensures the integrity of variables.

4. SOLUTION METHOD

4.1. Representation and Solution Generation

A procedure to generate a feasible path from a directed graph by using TS and random priority assignment technique is described below:

procedure: a feasible path generation
 input: directed graph.
 while (any vertex remains) **do**
 if every vertex has a predecessor,
 then the network is infeasible: **stop.**
 else pick a vertex v with the highest priority among vertices with no predecessors;
 que ← v;
 delete v and all edges leading out of v from the directed graph;
 end_while.
end_procedure.

4.2 Generation of offspring population

For genetic algorithm, we propose a new efficient crossover operator for solving the TSP. The proposed crossover is called the *moon crossover* because the process is very similar moon's movement such as waxing moon → half-moon → gibbous → full moon. A subtour is compared to the waxing moon or the half-moon. The procedure of the moon crossover operator is described in Figure 5.

```
procedure: moon crossover
  begin
    osp ← null;
    k ← 0;
    Select two random chromosomes p_a and p_b,
    where p_a = g_1 g_2 g_3 g_4 ... g_J, and p_b = q_1 q_2 q_3 q_4 ... q_J;
    Select two genes from the p_a at random. The
    substrings defined by the two genes;
    osp ← the substring between g_i and g_j
    selected from the p_a;
    if the length of osp = J then end;
    else sub_p_b ← the remaining substring results
    from the deleting genes which are already
    selected from the p_a, i.e., sub_p_b = p_a - osp;
    end_if
    while (length of osp ≠ J) do
      if i = 1 then i = J+1;
        i ← i - 1;
        k ← k + 1, k = 1, 2, ... , length of sub_p_b;
      if g_i ≠ q_k, then osp ← <osp, g_i, q_k>;
      else g_i = q_k, then osp ← <osp, g_i>;
      else if j = J then
        i ← i - 1;
        k ← k + 1, k = 1, 2, ... , length of sub_p_b;
      if g_i ≠ q_k, then osp ← <q_k, g_i, osp>;
      else g_i = q_k, then osp ← <g_i, osp>;
      else
        i ← i - 1;
        k ← k + 1, k = 1, 2, ... , length of sub_p_b;
      if g_i ≠ q_k, then osp ← <g_i, osp, q_k>;
      else g_i = q_k, then osp ← <g_i, osp>;
      end_if
    end_while.
end_procedure.
```

The swap mutation operator is also introduced here.

4.3 Evaluation and selection

To improve the solutions, each chromosome in the population is evaluated using some measures of fitness. The fitness is simply equal to the objective function value in Eq. (4).

Selection strategy is concerned with the problem of how to select chromosomes from population space. It may create a new population for the next generation based on either parent and offspring or part of them. A mixed strategy based on the roulette wheel and elitist selection is adopted as the selection procedure.

4.4 Genetic Algorithm Application

The PPS problem considered deals with multiple TSPPR, each of which corresponds to a process plan selection for each part type. Let $P(t)$ and $C(t)$ be respectively populations for parent and offspring in generation t. Overall procedure of the proposed genetic algorithm is described as shown in Figure 4. Since the algorithm is to solve a TSPPR, multiple executions are required for the multiple TSPPRs. Details of the algorithm are explained below.

```
procedure: Applied Genetic Algorithm
  Initialization
    t ← 0 ;
    initialize parent population P(t);
    evaluate P(t) and select the best solution σ* with
    the minimum objective function among P(t);
  while (no termination criteria) do
      obtain C(t) from P(t) by applying the genetic
      operators;
      evaluate C(t) and select the current best
      solution σ with the minimum objective
      function among C(t);
      update the best solution σ*, i.e., if σ < σ*,
      then σ* = σ;
        select P(t+1) from P(t) and C(t);
        t ← t + 1;
      end_while.
  end_procedure.
```

5. Conclusion

In this paper, we developed a model to integrate process planning and resource planning through analysis of the machine tool selection and operations sequencing problem. The model is formulated as a travelling salesman problem with precedence relations. To solve our model, we also proposed an efficient Genetic algorithm based on topological sort concept.

Reference

[1] Finke, G., A. Claus, and E. Gunn, 1984, A two-commodity netweork flow approach to the traveling salesman problem, *Congressus Numerantium*, Vol. 41, pp.167-178.

[2] Gen, M. and R. Cheng, *Genetic Algorithms and Engineering Design,* John Wiley & Sons, New york, 1997.

[3] Hancock, T. M., 1988, "Effects of adaptive process planning on job cost and lateness measures," *International Journal of Operations and Production Management*, Vol. 8, No. 4, pp 34-49.

[4] Malek, M., Guruswamy, M., Pandya, M. and Owens, H., 1989, "Serial and parallel simulated annealing and tabu search algorithms for the traveling salesman problem," *Annals of Operations Research,* Vol 21, pp 59-84.