濟州大學校 産業技術硏究所 論文集, 第8卷, 1號, pp.105~111, 1997. 6
Cheju National Univ. R.I.I.T. Jour., Vol.8, No.1, pp.105~111, 1997. 6

Evolutionary Programming for Designing Independent Cells

Chi-Ung Moon*, Jang-Hyung Kim** and Yu-Kyung Lee***

독립셀 설계를 위한 진화프로그래밍

문 치 웅* · 김 장 형** · 이 유 경***

ABSTRACT

In this paper, an evolutionary programming approach is proposed for the designing independent cells in cellular manufacturing with alternative process plans and machine duplication consideration. Several manufacturing parameters such as production volume, machine capacity, processing time, number of cells and cell size are considered in the process. The model is formulated as a 0-1 integer programming and solved using genetic algorithm. It determines the machine cells, part families and process plan for each part simultaneously.

Key words : Evolutionary Programming, Genetic Algorithm, 0-1 Integer Programming, Independent cells

Ⅰ. INTRODUCTION

In recent years, the application of group technology(GT) has become a starting point for the design of manufact.uring systems. Cellular manufacturing(CM) is an application of GT concept to organize cells which contain a set of machines to process a part family. The advantages derived from CM include reduced number of setups and material handling costs, decreased work-in-process inventories, improved space utilization, and simplified planning and scheduling(1)(2) .

The procedure of grouping the machines and parts to form cells in CM is called manufacturing cell design. The manufacturing cell design is an

* 아시카가 공과대학 방문연구원
Ashikaga Institute of Technology Visiting Researcher
** 제주대학교 정보공학과
Dept. of Information Eng., Cheju Nat'l Univ.
*** 제주대학교 대학원
Graduate School, Cheju Nat'l Univ.

important step in the development and implementation of CM systems. For the successful implementation of CM[2][3], identification of independent manufacturing cells, i.e., cells where parts are completely processed in the cell and no intercell movements, is necessary in the design phase of CM .

The formation of independent cells is a common goal for cell design.

Generally, the design of independent manufacturing cells may not be possible without duplication of bottleneck machines, but at the same time, the duplication requires additional capital investment. Although the design of independent manufacturing cells requires additional investment, the more simplified of production planning and scheduling functions might provide enough savings to justify them. And if we allow that a part has two or more process plans and each operation associated with a part can be processed on alternative machines, it may be possible to design of independent cells without much additional investment. Also, Consideration of alternative process plans may greatly enhance the efficiency of grouping[4].

Over the past decade, many models have been developed to solve the manufacturing cell design problem, but only a few models have been reported to design the manufacturing cells considering above mentioned issues. Kusiak[4], Gunasingh and Lashkari[5], Rajamani[2], and Sankaran and Kasilingam[6] have discussed the necessity for considering the alternative process plans in designing manufacturing cells. But, these models do not simultaneously form the machine cells and part families, or have limitations in the number of machines and parts they can handle. And, no model is found to form the independent manufacturing cells.

In this research, we develop a model to solve the problem of independent cells design considering the alternative process plans and machines duplication. Several manufacturing parameters such as production volume, machine capacity, processing time, number of cells and cell size are considered in the process. The model is formulated as a 0-1 integer programming model and an approach using evolutionary programming is developed to solve the model.

II. PROBLEM STATEMENT

In this section, a 0-1 integer programming model is developed to decide the process plan for each part and to form the manufacturing cells. It incorporates the process plans and other manufacturing factors mentioned in the previous section. The model determines the process plan for parts, part families and machine cells simultaneously. in order to present a 0-1 integer programming model of cell design problem, the assumption as given as follows:

(1) alternate process plans for each part are known.

(2) number of cells for configuration and upper limits on number of parts in each cell are known.

(3) only one same machine type within each cell exist.

and, in developing the model, the following notation is defined:

i : part index ($i=1,2,\cdots,np$)

j : process plan index ($j = 1,2,\cdots,npp_i$)

k : machine index ($k = 1,2,\cdots,nm$)

c : cell index ($c = 1,2,\cdots,nc$)

p_{ij} : process plan j in part i

ms_{ij} : a set of machines for part i when the process plan j is selected

pm_{ij} : cardinality of ms_{ij}, $pm_{ij}=n(ms_{ij})$

pc_{ijk} : processing cost of machine k for part i under process plan j

mc_k : cost of machine type k

U_c : upper limit on the number of parts in cell c

um_k : number of copies of machine type k

cam_k : capacity of machine type k

pv_i : production volumes for part i

$$x_{ijc}=\begin{cases}1 & \textit{if part i produced under process plan j belongs to cell c}\\ 0 & \textit{otherwise}\end{cases}$$

$$y_{kc}=\begin{cases}1 & \textit{if machine type k belongs to cell c}\\ 0 & \textit{otherwise}\end{cases}$$

The objective of the model is to minimize the sum of the processing costs and the duplication costs. The overall model is formulated as follows:

$$\textit{minimize } TC=\sum_i\sum_j\sum_k\sum_c pv_i pc_{ijk} x_{ijc} + \sum_c\sum_k mc_k y_{kc}$$

subject to

$$\sum_j\sum_c x_{ijc} = 1, \quad \forall i \tag{1}$$

$$\sum_i\sum_j x_{ijc} \le U_c, \quad \forall c \tag{2}$$

$$\sum_i\sum_j pv_i pc_{ijk} x_{ijc} \le cam_k, \quad \forall (k,c) \tag{3}$$

$$\sum_{k\in ms_{ij}} y_{kc} \ge pm_{ij} x_{ijc}, \quad \forall (i,j,c) \tag{4}$$

$$x_{ijc}, y_{kc} \in \{0, 1\}, \quad \forall (i,j,k,c) \tag{5}$$

The constraint (1) ensures that only one process plan is selected for each part, and that a part belongs to only one cell. The constraint (2) imposes the restriction on the maximum number of parts in a cell in order to have easy planning and control. The constraint (3) implies that total processing time for all parts using a machine type less than or equal to one's capacity. The constraint (4) represents that all machines needed for the produce of a part under a selected process plan ought to be assigned to the same cell. The Constraint (5) indicates the integer variables.

III. GENETIC ALGORITHM

The above described model has a few critical limitations. Because the variables are constrained to integer values, the model is difficult to solve for large number of parts and machines due to the computational complexity. Also, this model does not offer cell designers the flexibility to change objective functions and constraints.

In this section, an evolutionary programming approach using genetic algorithms is developed to overcome these limitations, and to solve the simultaneous process plans selection and independent manufacturing cell design problem.

3.1 Representation

Representation is the first step in applying GA to cell design problem. It is a key issue in GA because the representation links the real-world problem to GA and because GA directly manipulates the represented chromosomes as the problem. The chromosomes might be bit strings, real number, permutation of elements or many others[7]. In the machine cell design problem considered in this research, each gene represents a process plan number and cell number for each part. The length of the chromosome represents the number of parts considered in the problem. Let the number of parts is np, a chromosome can be represented as follows:

$$S_i = [\, g_1 \;\; g_2 \;\; g_3 \;\; \cdots \;\; g_k \;\; \cdots \;\; g_{np} \,]$$

where, S_i is the i-th chromosome and g_k is the k-th gene.

In this chromosome, let the number of process plans for each part is npp_i and upper limit on number of cells is uc, the range of g_k value is $[1,\ npp_i \cdot uc]$. If a is a g_k from S_i, the process plan number for the part i and the cell number can be calculated by the following equations:

$$j \leftarrow (d-1) \setminus cn + 1$$
$$c \leftarrow (d-1) \bmod cn + 1 \qquad (6)$$

By the equation (6), the S_i can be represented as follows:

$$SS_i = [(j_1, c_1), (j_2, c_2), \cdots, (j_k, c_k), \cdots, (j_{np}, c_{np})]$$

For instance, let $np = 5$, $npp_1 = 2$, $npp_2 = 3$, $npp_3 = 2$, $npp_4 = 1$, $npp_5 = 2$ and $uc = 3$, we have the following chromosome:

$$S_1 = [\ 5 \quad 2 \quad 7 \quad 1 \quad 6\ ]$$

By the equation (6), it is calculated that the process plan number and cell number for each part. The result values of above chromosome are shown as follows:

$$SS_1 = [(2,1)\ (1,2)\ (2,3)\ (1,1)\ (2,2)]$$

3.2 Initialization

The second step in GA is to initialize the population of chromosomes. This process can be executed with either a randomly generated population or a well adapted population. In this model, the chromosome is generated randomly.

3.3 Evaluation Function

In many optimization problems, the objective function is more naturally stated as the minimization of some cost function. In this research, the objective function for the cell design model was formulated in order to minimize the sum of the processing costs and duplication costs. The fitness value is calculated based on the original objective function defined as equation TC in the section 2. However, solution space for the selection of the process plan number and cell number in the cell design problem contain two parts: feasible area and infeasible area. For handling infeasible chromosome, we use the regenerating technique[8].

3.4 Crossover

Crossover is aimed at exchanging bit strings between two parent chromosomes. The crossover used this model is one cut-point method. For example, the parent chromosomes as follows and the cut point is randomly selected at position 2. The crossover operation is shown as follows:

cut point
↓

parent 1 = [5 2 7 1 6]
parent 2 = [3 7 2 2 5]
⇓
offspring 1 = [5 2 2 2 5]
offspring 2 = [3 7 7 1 6]

3.5 Mutation

Mutation is performed as random perturbation. Each value with the chromosome which is randomly selected to be mutated is changed with probability equal to the mutation rate. In this cell design model, mutation is designed to perform random exchange, i.e. selects each gene randomly in a pair of chromosomes and their genes are exchanged to create two offspring. An example is given as follows:

selected gene
↓

parent 1 = [5 2 2 2 5]
parent 2 = [3 7 7 1 6]
⇓
offspring 1 = [5 2 7 2 5]
offspring 2 = [3 7 2 1 6]

3.6 Selection

Deterministic selction strategy is adopted as the selection strategy, i.e. sort of parents and offspring on ascending order and select the first *pop_size* chromosomes as the new population.

IV. NUMERICAL EXAMPLE

In order to demonstrate the efficiency of the proposed approach, we considered in a manufacturing system with 10 machines, 7 parts, 3 number of cells, and 200 duplication costs for each machine. And the maximum number of parts in each cell is restricted to five. The set of process plans, production volumes for each part, process cost per unit and machine capacity are given in Table 1.

The genetic parameters for example problem were set as follows: crossover rate, p_c = 0.6; mutation rate, p_m = 0.2; maximum generation, *max_gen* = 1000; population size, *pop_size* = 100.

For this setting, we run genetic algorithm 5 times and the results are listed in Table 2.

Also, we solved the 0-1 integer programming model by branch and bound technique. The optimal solution over 5 runs is 8120 and the average processing time is 340 seconds.

For the evaluate the effectiveness of the genetic algorithm based approach, 2

Table 1 Data of machines-parts pocessing

part	M_k / p_{ij}	1	2	3	4	5	6	7	8	9	10	pv_i
1	1	2*		3							2	80
2	1					2				3	1	80
	2	4		5		4				7		
	3	6		5		7				5		
3	1	6				5		6				80
	2					3		3				
4	1			4		5					7	80
	2		4			7					6	
	3					3				4	3	
5	1			4	2					5	2	80
6	1	5	6				4		8			80
	2	2	4				2			4		
7	1			5	7					3		80
cam_k		2500	2300	2000	2200	2000	2500	2500	2000	2000	2000	

Table 2 Optimal solution

Solution	Cell no.	Part(no. of process plan)	Machine types	Fitness value	Average processing time (s)
Alternative 1	1	1(1) 5(1) 7(1)	1 3 4 9 10	8120	247
	2	2(1) 3(2) 4(3)	5 7 9 10		
	3	6(2)	1 2 6 9		
Alternative 2	1	3(2)	5 7		
	2	2(1) 4(3) 6(2)	1 2 5 6 9 10		
	3	1(1) 5(1) 7(1)	1 3 4 9 10		

Table 3 Comparision of results

size (m × p)	AST of GA (s)	AST of branch and bound (s)	Optimal value
10 × 10	293	440	10720
10 × 15	362	549	13780
AST: average searching time			

different size problems are adopted. The test results are shown in Table 3. Here, the genetic algorithm based approach is able to find a optimal solution within a satisfactory execution time.

V. CONCLUSION

The problem of independent cells design considering alternative process plans and machines duplications is studied. The problem is formulated as a 0-1 integer programming model, and a genetic algorithm-based approach is developed to solve the problem. The model identifies part families and machine cells simultaneously, and it also specifies the process plan for each part.

From the result of examples the proposed approach using genetic algorithm is more effective than existing method in terms of searching efficiency and problem size.

VI. 요 약

본 본문에서는 독립셀 설계를 하나의 0-1 Integer Programming 모델이 개발되었다. 모델은 대안적인 공정계획과 중복설계, 공정시간, 그리고 가용능력등과 같은 현실적인 관련 파라메터들을 고려하여 설계되었다. 그리고 이 모델의 해결을 위해 유전알고리즘에 기초한 진화적 해결방법을 제시하였다. 이 방법은 기존의 방법에 비하여 빠른 시간내에 최적해를 구하였으며, 대규모의 문제에 대하여 효율적으로 적용 가능하였다.

REFERENCE

1) Wemmerlov, U. and Johnson, D. J., 1997, "Cellular Manufacturing at 46 User Plants: Implementation Experiences and Performance Improvements," IJPR, Vol. 35, No 1. pp. 29-49.
2) Rajamani, D., Singh, N. and Anefa, Y. P., 1996, "Design of Cellular Manufacturing Systems," IJPR, Vol. 34, No. 7, pp. 1917-1923.
3) Sarker, B. R. and Balan, C. V., 1996, "Cell Formation with Operation Times of Jobs for Even Distribution of Workloads, IJPR, Vol. 34, No. 5, pp. 1447-1468.
4) Kusiak, A., 1987, The Generalized Group Technology Concept," IJPR, Vol. 25, No. 4, pp. 561- 569.
5) Gunasingh, K. and Lashkari, R. S., 1989, "Part Routing and Machine Grouping in FMS-An Integrated Approach," Computer Applications in Production and Engineering, North-Holland, pp. 651- 658.
6) Sankaran, S. and Kasilingam, R. G., 1990, "An Integrated Approach to Cell Formation and Part Routing in Group Technology Manufacturing Systems," Eng. Opt., Vol. 16, pp. 235-245.
7) Gen, M. and Cheng, R., 1997 Genetic Algorithm and Engineering Design, John Wiley & Sons.
8) Moon, C. U. and Kim, J. H., 1997, "Machining Process Sequencing and Load Balancing by Genetic Algorithm," *Proceedings of International Conference on Engineering Design and Automation, Bangkok.*