Approaches to the General Cell Formation Problem

by L R Foulds and
J M Wilson

Business School

Research Series
Paper 2002: 2
ISBN 1 85901 178 0

Loughborough
University

Approaches to the General Cell Formation Problem

by

L R Foulds and J M Wilson

Business School Research Series

Paper 2002: 2

ISBN 1 85901 178 0

September 2002

THIS PAPER IS CIRCULATED FOR DISCUSSION PURPOSES AND ITS CONTENTS SHOULD BE CONSIDERED PRELIMINARY AND CONFIDENTIAL. NO REFERENCE TO MATERIAL CONTAINED HEREIN MAY BE MADE WITHOUT THE CONSENT OF THE AUTHORS.

APPROACHES TO THE GENERAL CELL FORMATION PROBLEM

L. R. Foulds
Department of Management Systems, University of Waikato, New Zealand
lfoulds@waikato.ac.nz

J M Wilson
Business School, Loughborough University, Great Britain
j.m.wilson@lboro.ac.uk

September 2002

ABSTRACT

It has long been recognized that productivity in manufacturing plants can often be increased by producing similar products in manufacturing cells. This involves: (i) assigning parts to individual machines and (ii) forming machines into manufacturing cells. These two activities have traditionally been carried out separately. However most solution procedures for (i) above, utilize a solution to (ii), and vice versa. Here we present a new model that deals with (i) and (ii) simultaneously. We then extend this model to allow for the reassignment of operations to different machine types by incorporating machine type modification costs. Such modification enables additional machine types to process certain parts, with view to reducing inter-cell travel. The cost of such modifications must be balanced by the consequent reduction in inter-cell travel cost. The extended model specifies which individual machines should be modified to enable them to process additional part types, part-machine assignment, and the grouping of individual machines for cell formation. The objective is to minimize the sum of the machine modification costs and the inter-cell travel. We call this endeavour the General Cell Formation Problem. Computational experience with the models indicates that they are likely to be useful additions to the production engineer's toolkit.

Key Words: manufacturing, general cell formation problem, integer programming.

1. Introduction

Skinner [1974] was the first to propose the concept of a focused factory, in which small manufacturing systems operate independently within large production plants. The idea works best for medium-variety, medium-volume situations, that is, batch production. The focused factory is constructed using the notions of either flexible manufacturing systems (FMS) or group technology (GT), which are based on the precept that certain activities should be dedicated to a family of related parts in a manufacturing cell. Later Burbidge [1975] developed and popularized a systematic approach to this concept, which has subsequently seen widespread adoption in western industry.

Since machines are located in close proximity in a manufacturing cell and a family of related parts are produced, there is usually a reduction in: transport requirements, conveyance times, set up times, and inventory. Moreover, the relatively large autonomy within these manufacturing cells leads to extra motivation of the workers (who are responsible for "their products"), often resulting in higher productivity and product quality. These, and other advantages, have been discussed by Shunk [1985] and Hadley [1996]. However there are also disadvantages to this approach, such as the relatively costly duplication of machines.

FMS is related to GT in so far as both are sub-systems that represent "islands" within the production process, comprising groups of machines (sometimes including a material handling system), which produce a family of items. The main difference is that an FMS represents a fully automated system, whereas in GT conventional technology generally predominates. Most of the recent major results in the GT literature, concerned with the single criterion of minimizing inter- cell materials handling costs, have been discussed by: Billo [1998], Chu [1995], Kusiak and Heragu [1987], Selim, Askin, and Vakharia [1998], Vakharia [1986], Wemmerhov and Hyer [1987], and Wang [1998]. An evolutionary approach to multicriteria manufacturing cell formation has been reported by Pierreval and Plaquin [1998].

Suppose that a number of different products have to be manufactured using certain machine types. It is known from the process plans of the parts, which machine types are required for producing the individual parts, and the routing (machine ordering) for each part is given. We wish to assign the different parts to the individual machines of the types required and to group the machines so that each group forms a manufacturing cell.

This leads to the following activities:

(a) Assign part families to groups of machine types,
(b) Find lot sizes of the parts produced,
(c) Determine the number of machines needed of each machine type,
(d) Assign parts to individual machines,
(e) Group individual machines into manufacturing cells, and
(f) Compute job schedules for the machines.

The so-called machine type-part incidence matrix specifies which parts must be processed by which machine types. It is desirable that the machine type-part matrix should be transformed into a block-diagonal form to solve problem (a) (see, for example, Askin and Standridge [1993], Kumar, Kusiak, and Vanelli [1986], Kusiak and Chow [1988], and Hadley [1996]). Each block then shows which family of parts is to be processed in which group of machine types. This is reviewed in Section 2 of the present paper.

If such a block-diagonal clustering cannot be obtained, activities (b) to (e) have to be carried out. Well-known methods from inventory control (Hillier and Liebermann [1995], Nahmias [1993], Domschke, Scholl, and Voss [1997], and Neumann [1996]) can be used to carry out activity (b). A method that includes specific information relevant to group technology has been proposed by Askin and Chiu [1990]. Given the lot sizes for all parts, we can compute the necessary utilization level of each machine type, which also provides the number of machines needed of each type, ie, the solution to problem (c). Problems (b) and (c) are discussed briefly in Section 3.

Problems (d) and (e) have traditionally been solved separately. (See, for example, Askin and Chiu [1990], Askin and Standridge [1993], Cao and McKnew [1994], Faber and Carter [1986], Garcia-Diaz and Lee [1995], Hadley [1996], Kumar et al. [1986], Kusiak and Chow [1988], Moussa and Kamel [1995], Plaquin and Pierreval [2000], Neumann [1996], Vanelli and Hall [1993], Rajagopalan and Batra [1975], and Zhou and Askin [1998].) However, most solution procedures for problem (d) utilize a solution to problem (e) and vice versa. Problems (d) and (e) are reviewed in Section 4.

In Section 5 of the present paper we present a new approach that we believe represents an improvement on early attempts at solving problems (d) and (e) simultaneously, as reported by Atami, Lashkari, and Caron [1995] and Caux, Bruniaux, and Pierreval [2000].

After the formation of the manufacturing cells, a job-shop scheduling problem has to be solved. A job corresponds to a batch of some part. For each set of manufacturing cells, with some inter-manufacturing cell material flow (briefly called a *manufacturing cell system*), the makespan, that is, the maximum completion time of all jobs, is to be minimized. We seek to determine the job sequence for each machine of the manufacturing cell system and the job schedules, which specify the start and completion times of the jobs. These job-shop problems can be solved by well-known methods (Pinedo [1995]).

In Section 6 we extend the ideas of Section 5 and produce a comprehensive model of the cell formation problem. This model subsequently introduces the possibility of machine modification to reduce inter-cell travel. The model has been tested on a reasonably realistic data set and the results are described. The model is discussed further in Section 7.

Section 8 summarizes our conclusions. We now briefly discuss the abovementioned preliminary problems (a), (b), and (c), and sketch methods for solving them approximately. A glossary of the notation used in this paper is appended.

2. Formation of Families of Parts and Their Assignment to Manufacturing Cells of Machine Types

Assume that n parts, numbered 1, 2,...,*n*, are to be processed on *m* machine types: numbered 1,2,…,*m*, which are to be grouped into up to *p* manufacturing cells, numbered 1.2,…,*p*. The information as to which parts are to be processed on which machine types is given by the so-called machine type-part matrix, with elements:

$$(2.1) \qquad a_{\mathrm{ij}} := 1, \text{ if part j is processed on } M_{\mathrm{i}}, \text{ and}$$
$$= 0, \text{ otherwise, } (i = 1, 2, ..., m; j = 1, 2, ..., n.),$$
$$\text{where } M_{\mathrm{i}} \text{ denotes the machine type } i.$$

We attempt to reorder the machine type rows and part columns of the machine type-part matrix to obtain a block diagonal structure as shown in Figure 1.

Figure 1 about here

The term “block diagonal” implies that we can partition the matrix such that the boxes on the main diagonal contain as many 1’s as possible, but the off-diagonal boxes contain only 0’s. If such a block diagonal structure (as shown) is obtained, the items that correspond to columns of one block (constituting a family of parts) are processed only on those machine types that correspond to the rows of that block (a group of machine types). Each block is a candidate for a manufacturing cell. To order the rows and columns of the machine type-part matrix, we can use, for instance, a binary ordering algorithm (Askin and Standridge [1993], Neumann [1996]). As an example, we consider the machine type-matrix given in Table 1.

Table 1 about here

It can be shown that is impossible to convert Table 1 into a block diagonal structure. When such a conversion is impossible, it is necessary to consider individual machines instead of machine types, and to attempt to reduce the flow of materials between different manufacturing cells by a more detailed investigation. This is now discussed further.

3. Computation of Lot Sizes and Minimum Number of Machines

For each part *j*, let d_j be the demand rate, K_j be the set up cost, and h_j be the inventory holding cost per unit per period. We can often choose the *lot size* or *batch size* q_j , to be the *economic order quantity*:

(3.1)
$$q_j = \sqrt{\frac{2K_j d_j}{h_j}}, j = 1,2,...,n .$$

The rationale for formula (3.1) is well-known (Hillier and Lieberman [1995], Nahmias [1993], Domschke, et al. [1997], and Neumann [1996]). Askin and Chiu [1990] have modified the economic order quantity formula by assuming that the total throughput time for a part is a multiple of the total processing time (including the set up time) for each lot of that part.

Next, for each M_i , 1,2,…,*m*, we determine the minimum number of machines needed. Let C_i be the capacity of each M_i (measured by its running time, including set up) available per period. Let s_{ij} be the required set up time per batch or lot of part *j* for any M_i and let t_{ij} be the processing time (without set up time) for one unit of part *j* on any M_i. We do not consider conveyance times as they can be neglected within each manufacturing cell. The processing time p_{ij} , of a job *j* (of batch size q_j) on any M_i is then

(3.2) $$p_{ij} = s_{ij} + q_j t_{ij}.$$

The *utilization* u_{ij}, of the M_i's by part j is given by

(3.3) $$u_{ij} = \frac{\frac{d_j s_{ij}}{q_j} + d_j t_{ij}}{C_i}.$$

Note that u_{ij} is the number of M_i's, or fraction thereof, that is required for the processing of part j. We set $u_{ij} = 0$ if part j is not processed on any M_i. The numerator in formula (3.3) represents the running time of an M_i per period required for part j. If $u_{ij} > 1$, say, $1 < u_{ij} < 2$, we introduce a dedicated M_i to process part j.

Given the utilization u_{ij} of the M_i's by part j ($j = 1, 2, \quad ,n$) the *minimum number* e_i, of M_i 's required for producing all parts, can be computed as:

(3.4) $$e_i = \left\lceil \sum_{j=1}^{n} u_{ij} \right\rceil$$

where $\lceil c \rceil$ is the smallest integer greater than or equal to c (rounded up). The *average utilization* $\bar{e}_i$ of an M_i is

(3.5) $$\bar{e}_i = \frac{\sum_{j=1}^{n} u_{ij}}{e_i}.$$

We now go on to use the concepts we have just defined to address the two activities (d) and (e) introduced in Section 1. In the next section we survey traditional heuristic methods that address activities (d) and (e) sequentially. In Sections 5 and 6 we present new models that address these two activities simultaneously.

4. Machine-Part Assignment and the Grouping of Machines into Cells

If part j must be processed by an M_i and $e_i > 1$, we must specify to which of the e_i M_i 's that part j is assigned. Thus, for each M_i with $e_i > 1$, we have to solve *a machine-part assignment problem* whose objective function (to be minimized) is a measure of the material flow or material handling cost between different manufacturing cells. In other words, the solving of the machine-part assignment problem requires some preliminary knowledge of the solution to the machine grouping problem.

The machine-part assignment problem can be modelled as a *graph-partitioning problem,* where the nodes of the graph correspond to the parts processed on M_i's. We seek to determine a minimum-cost partition into e_i subgraphs (See, for example, Askin and Chiu [1990], Askin and Standridge [1993], and Neumann [1996].) If $e_i = 2$, and no limits are imposed on the number of nodes of the subgraphs, the graph-partitioning problem can be solved as a multi-terminal network network flow problem, in polynomial time (See, for example, Gomory and Hu [1961], Nagamochi and Ibaraki [1992], and Shahrokhi and Matula [1990].) Otherwise, the graph-partitioning problem is known to be NP-hard in the strong sense (Garey and Johnson [1979]). In the machine-part assignment problem there is a maximum number of parts which can be processed on a single M_i due to the limited capacity of that machine. Thus, the corresponding graph-partitioning problem is NP-hard in the strong sense, whenever $e_i \geq 2$.

To reduce the computational effort required to solve the graph-partitioning problem, it is recommended to use a heuristic method, for example, that of Kandiller [1998], or the Kernighan-Lin heuristic (Kernighan and Lin [1970]), preceded by some partition construction procedure (Askin and Standridge [1993] or Neumann [1996]).

The construction procedure determines an assignment of the parts to the machines so that machine capacities are not exceeded. Consider an M_i with $e_i > 1$. Let $J_i \subseteq \{1,2,...,n\}$ be the set of parts processed on M_i 's. To construct a feasible solution, assign the parts $j \in J_i$ successively to the first M_i , with capacity available. If the capacity of the first M_i is exceeded, we use a second machine, and proceed analogously. Only if none of the M_i 's has sufficient capacity to process all of part *j*, should the operation be divided into partial processing by more than one M_i.

Returning to the example of Section 2, suppose that the utilizations u_{ij} , of the various machine types by parts *j* are given in Table 2, which also shows the minimum numbers e_i and average utilization $\bar{e}_i$ of each M_i .

Table 2 about here

As an example, we apply the construction procedure to the M_4 's, where $J_4 = \{2,4,5,6\}$ and $e_4 = 3$. Let M_i^k represent the k^{th} machine of type *i*. Part 2 has a utilization of M_4 that exceeds one. This creates a dedicated M_4^1, that processes a single part (part 2). The remaining utilization by part 3 is assigned to M_4^2, along with part of the utilization of part 4. The remaining utilization of part 4 is assigned to M_4^3 together with parts 5 and 6. The solution to the assignment problem for all machine types can be summarized in a *machine-part incidence matrix* with elements:

(4.1) b_{rj} = 1, if part j is assigned to machine *r*,
= 0, otherwise, where the number *r*, is the unique numerical label of some individual machine.

Table 3 gives the machine-part incidence matrix for our example.

Table 3 about here

Given a machine-part incidence matrix, the *machine grouping problem* (that is, the formation of manufacturing cells) can be solved in a variety of ways which involve the solution of problems (d) and (e) sequentially. For instance, Kusiak and Chow [1988], and others, have devised decomposition procedures for the machine-part matrix. Methods proposed by Askin and Chiu [1990], Askin and Standridge [1993], Faber and Carter [1986], Neumann [1996], Rajagopalan and Batra [1975], and others, are based upon a machine graph. The nodes of the machine graph correspond to the actual machines. There is an edge, with endnodes *r* and *s*, if some part *j* has to be processed on both machines r and s, and if the machine sequence (or routing) of part *j* contains the subsequence (*r*, *s*) or (*s*, *r*). That is, if part *j* is moved from machine *r* to machine *s*, or vice versa. Edge [*r*, *s*] is weighted according to the sum of the demand rates of the parts moved from machine *r* to machine *s* or vice versa. Again, a graph-partitioning problem can be formulated and solved by the methods mentioned above, where the resulting subgraphs correspond to the manufacturing cells.

Suppose that the minimum number $e_{\min}$, and maximum number $e_{\max}$, of machines that can be grouped in a single manufacturing cell is known. These numbers are usually arrived at by the management of the firm, based upon its plant organization parameters. (In our example we choose $e_{\min} = 4$ and $e_{\max} = 6$.) It is possible to use a binary ordering algorithm (Askin and Standridge [1993] and Neumann [1996]) to transform the machine-part matrix into an initial grouping of machines (as discussed in Section 2). To attempt to improve this initial grouping, it is possible to use the Kernighan-Lin heuristic. However the results may still be decidedly suboptimal. To achieve optimality we develop a new model that solves tasks (d) and (e) simultaneously. It is discussed in the next section.

5. A New Approach to Simultaneous Machine - Part Assignment and the Formation of Manufacturing Cells

We now develop a new integer program that addresses, simultaneously, the two activities of machine – part assignment and the formation of manufacturing cells.

Input data

u_{ij} , $e_{\min}$, and $e_{\max}$.

We assume that all the given machines of the various types are to be formed into a maximum of p manufacturing cells. As *p* is unknown, it is recommended that the following estimate $\hat{p}$, of *p* be used:

(5.1) $$\hat{p} = \left\lceil (1/\bar{e})\sum_{i=1}^{m} e_i \right\rceil,$$

where

(5.2) $$\bar{e} = \left\lceil \frac{(e_{\min} + e_{\max})}{2} \right\rceil.$$

Decision Variables

x_{ijq} = number of M_{i} 's, or fraction thereof, that process part *j* in manufacturing cell q ,

y_{ikq} = 1, if $M_{\mathrm{i}}^{\mathrm{k}}$ is assigned to manufacturing cell *q,*

= 0, otherwise,

w_{jq} = 1, if part *j* is processed in manufacturing cell *q*,

= 0, otherwise,

v_q = 1, if manufacturing cell *q* is formed, and

= 0, otherwise.

MODEL 1: Constraints

(5.3) $$\sum_q x_{ijq} = u_{ij} \ .$$

(The requirements for processing part *j* on the M_{i} ‘s must be satisfied.)

(5.4) $$\sum_j x_{ijq} \leq \sum_k y_{ikq} \ .$$

(The total requirement for processing of parts on the M_{i} 's in cell *q* must be met.)

(5.5) $$\sum_q y_{ikq} = 1 \ .$$

(M_i^k must be assigned to exactly one manufacturing cell*.)*

(5.6) $$\sum_{i,k} y_{ikq} \geq v_q \, e_{\mathrm{min}} \ .$$

(A manufacturing cell *q*, if formed, must contain at least e_{min} machines.)

(5.7) $$\sum_{i,k} y_{ikq} \leq v_q \, e_{\mathrm{max}} \ .$$

(A manufacturing cell *q*, if formed, must contain at most e_{max} machines.)

(5.8) $$x_{ijq} \leq u_{ij} * w_{jq} \ .$$

(A logical constraint.)

(5.9) $$v_{q+1} \leq v_q \ .$$

(Cells are formed in successive numerical order.)

Objective

(5.10) $$\text{Minimize } \sum_{j,q} w_{jq} .$$

(Minimize the occurrence of inter-manufacturing cell travel.)

We now solve a numerical example of Model 1, where:

m = 8, n = 10, p = 4, $e_{\min}$ = 4, and $e_{\max}$ = 6, and $(u_{ij})_{8\text{x}10}$ and the part routings are defined in Tables 2 and 4 respectively.

As part of the optimal solution, four manufacturing cells are formed, as follows:

Manufacturing cell 1 contains: $M_1^1, M_2^1, M_3^1, M_4^1, M_5^1, M_5^2$,

Manufacturing cell 2 contains: $M_2^4, M_4^3, M_5^6, M_5^7$, M_7^1,

Manufacturing cell 3 contains: $M_2^3, M_4^2, M_5^5, M_6^2$, and

Manufacturing cell 4 contains: $M_2^2, M_3^2, M_5^3, M_5^4, M_6^1, M_8^1$.

Manufacturing cell 1 processes Parts 2, 4, 8,

Manufacturing cell 2 processes Parts 4 and 5,

Manufacturing cell 3 processes Parts 2, 6, 10, and

Manufacturing cell 4 processes Parts 1, 3, 7, 9.

The example requires 31265 nodes of the branch and bound tree created by XPRESS-MP to be examined. The optimal solution was found at node 31104, taking 606 CPU seconds on a Sun-Solaris computer. The optimal objective value = 12. (Note that, in general, the objective function value will never be smaller than the value of n). The flow of parts within and between cells is illustrated in Figure 2. It was found that models of this type are not particularly robust. Small changes in the values of u_{ij} lead to instances of the model which solve in less than 10 CPU seconds.

We now go on to the main contribution of this paper, the generalization of the previous model to include costs arising from both machine modification and inter-cell travel.

6. The General Cell Formulation Problem

It has been noted in the literature (Burbidge [1982], Hyer and Wemmerhov [1989], and Selim et al. [1998]) that it is often important in practice to be able to reassign parts to different machine types in order to create better cell configurations. This involves extending the set of parts that certain individual machines can process. Such extensions may be cheaper than simply purchasing additional machines. Also, it may not be feasible to add any additional machines to a particular cell if the number of machines in the cell is already at the upper limit of $e_{\max}$. Machine extension of the nature just proposed is certainly feasible, given the operational flexibility of modern, general-purpose machines currently in existence.

Each given machine type, M_{i} say, is initially capable of processing a known set of parts, as specified by the (a_{ij}) matrix, defined in Section 2. We assume that any individual machine $M_{\mathrm{i}}^{\mathrm{k}}$ say, can be modified so that it can also perform as a machine of another type, at a given cost, thus increasing the operational flexibility of $M_{\mathrm{i}}^{\mathrm{k}}$. We further assume that the cost of extending the flexibility of any individual machine in a particular way is constant for all machines of its type. The decision as to which individual machines to extend, and the particular machine types that they are extended to incorporate, will depend upon their capacity, which cells they reside in, and the corresponding modification costs. Of course, the costs of certain machine type modifications pairs may be prohibitive.

The motivation for incurring the abovementioned modification costs is to reduce inter-cell materials handling costs. To this end, we also assume that the cost of transporting a batch of each given part between a pair of given cells is known. The overall objective is to minimize the total of all modification costs and of all inter-cell materials handling costs. The latter may have to be amortized as they are ongoing costs, whereas the former are one-off costs.

This minimization is achieved by deciding simultaneously: (i) which individual machines are to be modified so that they can behave as which additional machine types (subject to their processing capability), (ii) which individual machines will process which parts, and (iii) to which cell each individual machine will be assigned. Care must be taken that, in the minimum and maximum number of machines allowed in each cell, $e_{\min}$ and $e_{\max}$ respectively, are satisfied for all cells created.

We call the above endeavour the General Cell Formation Problem (GCFP). We now develop a model of GCFP, as an extension of Model 1. To begin this we introduce some additional notation. Let

f_{hi} = the cost of modifying any individual M_{h} so that it can also perform as an M_{i} ,

g_{jqr} = the cost of transporting a batch of part j between cells q and r,

z_{khi} = 1, if $M_{\mathrm{h}}^{\mathrm{k}}$is modified so that it can also perform as an M_{i} , and

= 0, otherwise.

MODEL 2: Constraints

Constraints (5.3) – (5.9), modified to allow for the possibility of machine modification via the introduction of the z_{khi} 's. The sum of the x_{ijq} 's must still not exceed 1 (the capacity of any machine) even with additional processing due to modification.

- **Objective :** Minimize $\sum_{h,k,i} f_{hi} \cdot z_{khi} + \sum_{j,q,r} g_{jqr} \ w_{jq} \ w_{jr}$.

We now continue with the numerical example of Section 5, where additionally:

$f_{hi} = (h)(i)$ [the product of *h* and *i*] $h,i = 1,2,\ldots 8$, and

$g_{jqr} = 4(j)(q)(r)$ $j = 1,2,...,10;$ $q,r = 1,2,3,4.$

Considering the solution to the numerical example given in Section 5 as a solution to Model 2, we have

$$z_{khi} = 0, \text{ for all } h,k,j.$$

Hence the objective function value equals only the cost of inter-cell travel, namely that of part 2 travelling from cell 1 to cell 3, and part 4 travelling from cell 1 to cell 2. That is, the total cost is:

$g_{213} + g_{414}$
= 4(2.1.3) + 4(4.1.2)
= 56.

This cost is necessitated by part 2 travelling from cell 1 to cell 3 to be processed by (0.6) of M_4^2 and (0.7) of M_6^2 , and by part 4 travelling from cell 1 to cell 2 to be processed by (0.5) of M_4^3 and (0.2) of M_7^1 .The utilizations of the individual machines in cell 1 are:

M_1^1 : 0.3 (part 2) + 0.1 (part 4) + 0.1 (part 8)	= 0.5,
M_2^1 : 1.0 (part 4)	= 1.0,
M_3^1 : 0.2 (part 8)	= 0.2,
M_4^1 : 0.6 (part 2) + 0.4 (part 4)	= 1.0,
M_5^1 : 0.2 (part 2) + 0.8 (part 4)	= 1.0,
M_5^2 : 1.0 (part 8)	= 1.0.

Thus only M_1^1 and M_3^1 have spare capacity, while the other four machines are all fully utilized. As cell 1 already has $e_{\max} = 6$ machines assigned to it, it is not possible to add another machine to the cell. Given the available spare capacity in cell 1 it is feasible to modify $M_1^{\,1}$ and $M_3^{\,1}$ to allow either (but not both) of parts 2 or 4 to be processed solely in cell 1. It is cheapest to modify $M_1^{\,1}$ and $M_3^{\,1}$ to become, additionally, machines of type M_7 and M_4 respectively. That is, set $z_{117} = 1$ and $z_{134} = 1$. This allows part 4 to be processed solely in cell 1. The objective function value becomes:

$f_{14} + f_{37} + g_{213}$
$= (1)(4) + (3)(7) + 4(2)(1)(3)$
$= 49$,

which represents an overall cost reduction of 7.

In fact we can do better than this by solving a revised model to optimality. The model also avoids the need for quadratic terms in the objective function. In what follows we assume, without loss of generality, that any machine cannot be modified more than once.

We introduce the new variables:

t_{hiqk} = fraction of machine $M_{\mathrm{i}}^{\mathrm{k}}$ (which is assigned to cell q), which is used as a machine of type i,

s_{jqr} = 1, if part j moves from cell q to cell r for processing,
= 0, otherwise.

The objective function is:

Minimize $\sum_{h,k,i} f_{hi} \,.\, z_{khi} + \sum_{j,q,r} g_{jqr} \,.\, s_{jqr}$.

The model is constraints (5.3), (5.5) - (5.9) together with the constraints (6.1) - (6.5).

(6.1) $$\sum_j x_{ijq} \le \sum_{h,k} t_{hiqk}$$

(The generalisation of constraint 5.4.)

(6.2) $$\sum_i t_{hiqk} \le y_{hqk}$$

(Each assigned machine creates a unit of processing which may or not be subdivided after machine modification.)

(6.3) $$\sum_{i|i\neq h} z_{khi} \le 1$$

(Each machine cannot be modified more than once.)

(6.4) $$t_{hiqk} \le z_{khi} \qquad h \neq i$$

(If a machine is modified then it creates a fraction of a unit of modified processing.)

(6.4) $$s_{jqr} \geq w_{jq} + w_{jr} - 1$$

(if a part uses two machines, the variable indicating inter-cell movement is set to 1.)

This revised model was run to optimality and produced an optimal solution of cost 28. In this solution, no parts travelled between cells. The optimal solution is:

Manufacturing cell 1 contains: M_1^1 ,M_2^1 ,M_4^1 ,M_5^1 ,

Manufacturing cell 2 contains: M_2^4 ,M_4^3 ,M_5^6 ,M_5^7, M_7^1 ,

Manufacturing cell 3 contains: M_2^2 ,M_3^2 ,M_5^3 , M_5^4, M_6^1, M_8^1 , and

Manufacturing cell 4 contains: M_2^3 ,M_3^1 , M_4^2, M_5^2 ,M_5^5 ,M_6^2 .

Manufacturing cell 1 processes Parts 4,

Manufacturing cell 2 processes Parts 5 and 6,

Manufacturing cell 3 processes Parts 1, 3, 7, 9, and

Manufacturing cell 4 processes Parts 2, 8, 10.

M_1^1 is modified to a machine M_7,

M_2^3 is modified to a machine M_4,

M_2^4 is modified to a machine M_5,

M_3^1 is modified to a machine M_1 .

The objective function is calculated as (1)(7)+(2)(4)+(2)(5)+(3)(1) = 28. This solution was obtained after 21702 branch and bound nodes (2632 CPU seconds). The complete search to prove optimality required a total of 67876 nodes (6329 CPU seconds). Although this is a considerable computational effort, it was felt to be appropriate given the generality of the new model.

7. An Extension

Further extensions of the models given in Section 6 are possible. One extension, which has not been investigated but seems a natural consequence of considerations of machine utilisation, would be to make use of further machine modification in order to avoid setting up machines. Considering the solution given at the end of the previous section, we see that cell 1 contains four machines but only processes part 1. Part 1 requires a total of 3.0 units of machine processing, so it could be completely processed in cell 1 but using only three machines. Thus the model could be extended to include costs of allocating machines to cells (setup costs), and if these costs were sufficiently high relative to the cost of modifying machines, then it might be possible to reduce total cost by not using certain machines. Such machines could then be dedicated to new tasks in other new cells.

8. Summary and Conclusions

We have reviewed some issues in the formulation of manufacturing cells, including the assigning of parts to individual machines, the grouping of individual machines into manufacturing cells, and the modification of individual machines to increase their part processing capability. We have presented new integer programs that combine these activities in one model, with the objective of minimizing overall machine modification and materials handling costs. We believe that the resulting models will become useful tools for production planners.

References

1. R.G. Askin and K.S. Chiu, A graph partitioning procedure for machine assignment and manufacturing cell formation in group technology. Int. J. Prod. Res. 28 (1990) 1555-1572.
2. R.G. Askin and C.R. Standridge, Modeling and Analysis of Manufacturing Systems. John Wiley, New York (1993).
3. A. Atamani, R.S. Lashkari and R.J. Caron, A mathematical programming approach to joint manufacturing cell formation and operation allocation in cellular manufacturing. Int. J. Prod. Res. 33 (1995) 1-15.
4. R.E. Billo, A Design Methodology for Configuration of Manufacturing cells. Computers and Industrial Eng. 34 (1998) 63-75.
5. P. Brucker, Scheduling Algorithms. Springer, Berlin (1995).
6. J.L. Burbidge, The Introduction of Group Technology. John Wiley, New York (1975).
7. J.L. Burbidge, The simplification of material flow systems. Int. J. Prod. Res. 20 (1982) 339-359.
8. Q. Cao and M.A. McKnew, On modelling the design of manufacturing cellular manufacturing systems. Int. J. of Flexible Manufacturing Systems. 6 (1994) 155-172.
9. C. Caux, R. Bruniaux and H. Pierreval, Manufacturing cell formation with alternative process plans and machine capacity constraints: A new combined approach. Int J. Production Economics. 64 (2000) 279-284.
10. C.H. Chu, Recent advances in mathematical programming for manufacturing cell formation. In Planning, Design and Analysis of Manufacturing cellular Manufacturing Systems (A.K. Kamrani, H.R. Parei, D.H. Liles eds), Elsevier, Amsterdam (1995) 3-46.
11. W Domschke, A. Scholl, and S. Voss, Produktions Planning, 2nd ed. Springer-Verlag, Berlin (1997).

12. Z. Faber and M.W. Carter, A new graph theory approach for forming machine manufacturing cells in manufacturing cellular production systems. In Flexible Manufacturing Systems: Methods and Studies (A. Kusiak ed), Elsevier, Amsterdam (1986) 301-315.
13. A. Garcia-Diaz and H. Lee, An industrial application of network-flow models in manufacturing cellular manufacturing planning. In Planning, Design and Analysis of Manufacturing cellular Manufacturing Systems (A.K. Kamrani, H.R. Parei, D.H. Liles eds), Elsevier, Amsterdam (1995) 47-61.
14. M.R. Garey and D.S. Johnson, Computers and Intractability. Freeman, San Francisco (1979).
15. R.E. Gomory and T.C. Hu, Multi-terminal network flows. J. SIAM. 9 (1961) 551-570.
16. S.W. Hadley, Finding part-machine families using graph partitioning techniques. Int. J. Prod Res. 34 (1996) 1821-1839.
17. N.L. Hyer and U. Wemmerlov, Group Technology in the US manufacturing industry: a survey of current practices.. Int. J. Prod. Res. 27 (1989) 1287-1304.
18. F.S. Hillier and G.J. Lieberman, Introduction to Operations Research. McGraw-Hill, New York (1995).
19. L. Kandiller, A manufacturing cell formation algorithm: Hypergraph approximation – Cut tree. EJOR. 109 (1998) 686-702.
20. B.W. Kernighan and S. Lin, An efficient heuristic procedure for partitioning graphs. Bell Systems Technical Journal. 49 (1970) 291-307.
21. K.R. Kumar, A. Kusiak and A. Vannelli, Grouping of parts and components in flexible manufacturing systems. *EJOR.* 24 (1986) 387-397.
22. A. Kusiak and W.S. Chow, Decomposition of Manufacturing Systems. J. Robotics and Automation. 4 (1988) 457 – 471.
23. A. Kusiak and S.S. Heregu, Group Technology. Computers and Industry. 9 (1987) 83-91.
24. S.E. Moussa and M.S. Kamel, Partitioning techniques for manufacturing cellular manufacturing. In Planning, Design, and Analysis of Manufacturing cellular Manufacturing Systems (A.K. Kamrani, H.R. Parei, D.H. Liles eds), Elsevier, Amsterdam, (1995) 73-95.
25. H. Nagamochi and T. Ibaraki, Computing edge-connectivity in multigraphs and capacitated graphs. SIAM J. Disc. Math. 5 (1992) 54-66.
26. S. Nahmias, Production and Operations Analysis, Irwin, 2nd ed., Homewood, Illinois (1998).
27. K. Neumann, Produktions und Operations Management. Springer-Verlag, Berlin (1996).
28. H. Pierreval and M-F Plaquin, An Evolutionary Approach to Multicriteria Manufacturing cell Formation. Int. Trans Opl. Res. 5 (1998) 13-25.
29. M. Pinedo, Scheduling. Prentice Hall, Englewood Cliffs, New Jersey (1995).
30. M-F Plaquin and H. Pierreval, Manufacturing cell formation using evolutionary algorithms with certain constraints. Int J. Production Economics. 64 (2000) 267-278.
31. R. Rajagopalan and J.L. Batra, Design of manufacturing cellular production systems: a graph-theoretic approach. Int. J. Prod. Res. 13 (1975) 567-579.

32. H.M. Selim, R.G. Askin, and A.J. Vakharia, Manufacturing cell Formation in Group Technology: Review, Evaluation and Directions for Future Research. Computers and Industrial Eng. 34 (1993) 3-20.

33. D.L. Shunk, Group Technology provides an organized approach to realized benefits of CIM's. Industrial Engineering. 17 (1985) 74-81.

34. W. Skinner, The Focused Factory. Harvard Business Review. May-June (1974) 113-121.

35. A.J. Vakharia, Methods of manufacturing cell formation in group technology: a framework for evaluation. J. Operations Management. 6 (1993) 257-271.

36. A. Vannelli and R.G. Hall, An eigenvector solution methodology for finding part-machine families. Int. J. Prod. Res. 31 (1993) 325-349.

37. J. Wang, A Linear Assignment Algorithm for Formation of Machine Manufacturing cells and Part Families in Manufacturing cellular Manufacturing. Computers and Industrial Eng. 35 (1998) 81-84.

38. U. Wemmerlov and N.L. Hyer, Research issues in manufacturing cellular manufacturing. Int. J. Prod. Res. 25 (1987) 412-422.

39. M. Zhou and R.G. Askin, Formation of General GT Manufacturing cells: An Operation-Based Approach. Computers and Industrial Eng. 34 (1998) 147-157

Glossary of notation

Input data

a_{ij} : = 1, if part j is processed on M_i, and
= 0, otherwise, (i = 1, 2, ..., m; j = 1, 2, ..., n.),

b_{rj} = 1, if part j is assigned to machine r,
= 0, otherwise, where the number r, is the unique numerical label of some individual machine,

C_i = the capacity of each M_i (measured by its running time, including set up) available per period,

e_i = the minimum number of M_i 's required,

e_{min} = the minimum number of machines that can be assigned to a cell,

e_{max} = the maximum number of machines that can be assigned to a cell,

f_{hi} = the cost of modifying any individual M_h so that it can also perform as an M_i ,

g_{jqr} = the cost of transporting a batch of part j between cells q and r,

h_j = the inventory holding cost per period for part j,

J_i = the set of parts to be processed by M_i ,

K_j = the set up cost of part j,

M_i denotes machine type i,

m = the number of machine types,

M_i^k denotes the k^{th} individual machine of type i,

n = the number of different types of parts,

p_{ij} = the processing time of a batch of part j on an M_i ,

p = the maximum possible number of cells,

q_j = the economic order quantity for part j,

u_{ij} = the utilization of M_i by part j,

Indices

h,i ~ machine types,

j ~ parts,

k ~ individual machines,

q,r ~ cells,

k ~ instances of a machine,

Decision Variables

x_{ijq} = number of M_i 's, or fraction thereof, that process part j in manufacturing cell q ,

y_{iqk} = 1, if M_i^k is assigned to manufacturing cell q,

= 0, otherwise,

w_{jq} = 1, if part j is processed in manufacturing cell q,

= 0, otherwise,

v_q = 1, if manufacturing cell q is formed, and

= 0, otherwise,

z_{khi} = 1, if $M_h^k$is modified so that it can also perform as an M_i and

= 0, otherwise,

t_{hiqk} = fraction of machine $M_{\mathrm{i}}^{\mathrm{k}}$ (which is assigned to cell q), which is used as a machine of type i,

s_{jqr} = 1, if part j moves from cell q to cell r for processing,
= 0, otherwise.

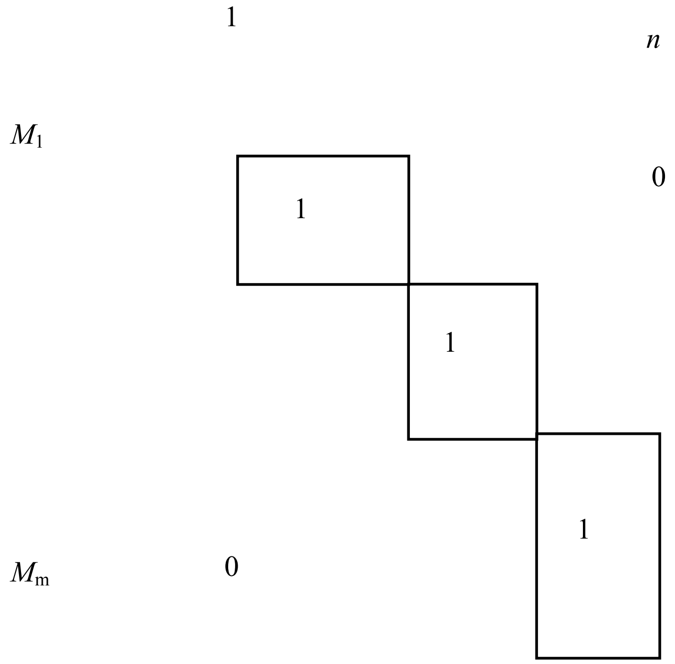

Figure 1

Figure 2. Available as a hard copy.

						Parts					
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
	1	0	1	0	1	0	0	0	1	0	0
	2	0	0	0	1	1	0	1	0	1	1
Machine	**3**	0	0	1	0	0	0	0	1	1	0
Types	**4**	0	1	0	1	1	**1**	0	0	0	0
	5	1	1	0	1	1	1	1	1	1	0
	6	0	1	0	0	0	0	1	0	0	1
	7	0	0	0	1	1	0	0	0	0	0
	8	1	0	0	0	0	0	0	1	1	0

Table 1.

				Parts									
Machine Types	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	$\sum_i u_{ij}$	**e$_i$**	$\bar{e}_i$
1	0.0	0.3	0.0	0.1	0.0	0.0	0.0	0.1	0.0	0.0	0.5	0.5	0.5
2	0.0	0.0	0.0	1.0	0.5	0.0	0.9	0.0	0.1	0.6	3.1	3.1	0.775
3	0.0	0.0	0.9	0.0	0.0	0.0	0.0	0.2	0.1	0.0	1.2	1.2	0.6
4	0.0	1.2	0.0	0.9	0.5	0.4	0.0	0.0	0.0	0.0	3.0	3.0	1.0
5	0.4	0.2	0.0	0.8	1.1	1.0	0.6	1.0	1.0	0.0	6.1	6.1	.871
6	0.0	0.7	0.0	0.0	0.0	0.0	0.8	0.0	0.0	0.3	1.8	1.8	0.9
7	0.0	0.0	0.0	0.2	0.2	0.0	0.0	0.0	0.0	0.0	0.4	0.4	0.4
8	0.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.5	0.0	0.7	0.7	0.7

Table 2

Parts

r		1	2	3	4	5	6	7	8	9	10
1	M_1^1	0	1	0	1	0	0	0	1	0	0
2	M_2^1	0	0	0	1	0	0	0	0	0	0
3	M_2^2	0	0	0	0	1	1	0	0	0	0
4	M_2^3	0	0	0	0	0	1	0	0	0	1
5	M_2^4	0	0	0	0	0	0	0	0	1	0
6	M_3^1	0	0	1	0	0	0	0	0	1	0
7	M_3^2	0	0	0	0	0	0	0	1	0	0
8	M_4^1	0	1	0	0	0	0	0	0	0	0
9	M_4^2	0	1	0	1	0	0	0	0	0	0
10	M_4^3	0	0	0	1	1	1	0	0	0	0
11	M_5^1	1	0	0	0	0	0	1	0	0	
12	M_5^2	0	1	0	1	0	0	0	0	0	0
13	M_5^3	0	0	0	0	1	0	0	0	0	0
14	M_5^4	0	0	0	0	0	1	0	0	0	0
15	M_5^5	0	0	0	0	0	1	0	0	0	0
16	M_5^6	0	0	0	0	0	0	0	1	0	0
17	M_5^7	0	0	0	0	0	0	0	0	1	0
18	M_6^1	0	1	0	0	0	0	0	0	0	1
19	M_6^2	0	0	0	0	0	0	1	0	0	0
20	M_7^1	0	0	0	1	1	0	0	0	0	0
21	M_8^1	1	0	0	0	0	0	0	0	1	0

Table 3

Part	Machine sequences (routings) of parts
1	M_5 , M_8
2	M_1 , M_5 , M_4 , M_6
3	M_3
4	M_1 , M_2 , M_5 , M_4 , M_7
5	M_2 , M_4 , M_5 , M_7
6	M_4 , M_5
7	M_5 , M_2 , M_6
8	M_1 . M_3 , M_5
9	M_3 , M_5 , M_2 , M_8
10	M_2 , M_5

Table 4.

Captions

Figure 1. Block-diagonal structure of a machine type-part matrix
Figure 2. Material flows within and between manufacturing cells

Table 1 Machine type-part matrix
Table 2. Utilizations u_{ij} of machine types M_i by parts j
Table 3. Individual machine-part matrix
Table 4. Machine sequences (routings) of parts