The General Cell Formation Problem: Manufacturing Cell Creation with Machine Modification Costs

*by L R Foulds and
J M Wilson*

Business School

Loughborough University

The General Cell Formation Problem: Manufacturing Cell Creation with Machine Modification Costs

by

L R Foulds and J M Wilson

THE GENERAL CELL FORMATION PROBLEM: MANUFACTURING CELL CREATION WITH MACHINE MODIFICATION COSTS

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

J M Wilson[*]
Business School, Loughborough University, UK
j.m.wilson@lboro.ac.uk

* Corresponding author. John M Wilson, Business School, Loughborough University, Loughborough LE113TU, UK. j.m.wilson@lboro.ac.uk 'phone +44(0)1509-223108, fax +44(0)1509-223960

SCOPE AND PURPOSE

In manufacturing systems there is frequently the need to process parts on a series of machines. To create suitable working environments cells are usually created comprising a subset of machines. Then when the processing is undertaken each part may be handled more efficiently because the processing operations may be confined to only a small number (ideally one) of cells thus avoiding the need for transportation of in-process parts across substantial distances on the factory floor.

A number of papers have considered previously the problem of optimal cell formation, but in this paper we introduce the possibility that machines within cells may be modified, at a cost, to allow them to undertake the role of a different machine in addition to their usual role. Such machine modification is technically feasible given the current operational flexibility of machines. With the additional possibility of machine modification the production system is able to move further to the ideal of conducting all processing of a part within the confines of one cell. In the paper a new model is developed for the cell formation problem with machine modification. Heuristics are developed to solve the problem and these are analyzed and discussed.

ABSTRACT

An approach for manufacturing cell formation with machine modification is presented. In cell formation it is often important in practice to be able to reassign parts to additional 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. Thus, there is the possibility of machine modification to reduce inter-cell travel. The cost of such modifications must be balanced by the consequent reduction in inter-cell travel cost. The extended machine cell formation problem to be described involves the specification of 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 the General Cell Formation Problem (GCFP). As far as the authors are aware, there have not been any solution procedures for this important problem reported in the open literature. It is our purpose to fill this gap by presenting a mixed integer programming model of the GCFP. We also proposed and analyze greedy and tabu search heuristics for the design of large-scale systems related to the GCFP. Computational experience with the solution procedures indicates that they are likely to be useful additions to the production engineer's toolkit.

Key Words: manufacturing cell formation, machine modification costs, models, integer programming, heuristics.

Introduction

As is well known, the activities required for the formation of a system of manufacturing cells, include:
(a)   Assigning part families to groups of machine types,
(b)   Finding lot sizes of the parts produced,
(c)   Determining the number of machines needed of each machine type,
(d)   Assigning parts to individual machines, and
(e)   Grouping the individual machines into cells.

Most of the major results in the literature on manufacturing cell system formation, where the single criterion of minimizing inter- cell materials handling costs is adopted, have been discussed by: Billo [7], Chu [11], Kusiak and Heragu [21], Selim et al. [30], Vakharia [33], Wemmerlov and Hyer [37], and Wang [36]. An evolutionary approach to multicriteria cell system formation has been reported by Pierreval and Plaquin [27]. Many solution procedures for cell system formation involve solving the problems represented by activities (d) and (e) above separately and iteratively. (See, for example, Askin and Chiu [4], Askin and Standridge [5], Faber and Carter [12], Garcia-Diaz and Lee [14], Hadley [16], Kumar et al. [19], Kusiak and Chow [20], Moussa and Kamel [25], Plaquin and Pierreval [28], Neumann [26], Vanelli and Hall [34], Rajagopalan and Batra [29], and Zhou and Askin [38].)

The most effective solution procedures that solve the problems represented by activities (d) and (e) simultaneously involve integer programming models (that have the special structure of being multi-divisional) or meta heuristic approaches. Such models have been reported by: Sundaram and Fu [1987], Ahmed et al. [1991], Jain et al. [1991] (also involving tool provisioning), Logendran [1993], Wang and Roze [1995], Zhu et al. [1995], Atamani et al. [1995] (involving cost trade-offs), Cao and McKnew [1998], (involving Lagrangian relaxation), Moon and Gen [1999] (involving a genetic heuristic), Abdelmola et al. [1998] (also involving unutilized machine time usage), Taboun et al. [1999] (involving cost trade-offs), Caux et al. [2000] (also involving alternative process plans and machine capacity constraints), Arzi et al. [2001] (also involving lumpy demand), Lozano et al. [2001] (involving quadratic programming, neural network solution procedures), and Foulds and Neumann [2003] (involving a network flow approach). In the next section we present a new cell system formation model, which also solves the problems represented by activities (d) and (e) simultaneously.

It has been noted by Burbidge [8], Hyer and Wemmerlov [17], and Selim et al. [30], that it is often important in practice to be able to reassign parts to different machine types in order to create better cell system configurations. This involves extending the set of parts that certain individual machines can process. Such extensions may be cheaper than simply purchasing additional machines. Thus, there is the possibility in cell system formation, of machine modification to reduce inter-cell travel. When such machine modification is feasible, we call the resulting cell system formation problem the General Cell Formation Problem (GCFP). However, as far we are aware, there have not been any solution procedures for this important problem reported in the open

literature. It is our purpose to fill this gap. Section 3 is the main contribution of the paper in which we modify the model we developed in Section 2 to present an effective mixed integer programming model. We also develop greedy and tabu search heuristics for the design of large-scale systems related to the GCFP. Possible extensions of the model are discussed in Section 4. Section 5 summarizes our conclusions. A glossary of the notation used in this paper is appended.

2.   A New Model of Manufacturing Cell System Formation

2.1   The development of the model

We now develop a new model that addresses, simultaneously, the two activities of machine-part assignment and the formation of cells. 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 cells, numbered 1.2,…,p. Let:

M_i denote machine type i ($i = 1,2,...,m$),
M_i^k denote the kth individual machine of type i,
e_{min} = the minimum number of machines that can be assigned to form a cell, and

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

For each M_i, ($i = 1,2,...,m$), we can determine e_i, the minimum number of machines needed. To this end, let:

C_i = the capacity of each M_i (that is, its running and set up time) available per period,
d_j = the demand per period for part j,
s_{ij} = the required set up time per batch or lot of part j for any M_i,
t_{ij} = the processing time (without set up time) for one unit of part j on any M_i.

We neglect travel times within each cell. The processing time p_{ij}, of a job j (of batch size q_j) on any M_i is then

$$p_{ij} = s_{ij} + q_j t_{ij}.$$

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

$$u_{ij} = \frac{\dfrac{d_j s_{ij}}{q_j} + d_j t_{ij}}{C_i}.$$

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 to be processed on any M_i. The numerator in the above formula for u_{ij} 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 ($j = 1, 2,$,n) the *minimum number e_i, of*

M_i 's required for producing all parts, can be computed as:

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

where $\lceil c \rceil$ is the smallest integer greater than or equal to c (rounded up).

As was stated at the beginning of this section, we assume that all of the individual machines of the various types are to be formed into a maximum of p cells. As p is unknown, it is recommended that the following estimate $\hat{p}$, of p be used:

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

where

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

We now construct a new model of cell system formation. Let:

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

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

    = 0, otherwise,

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

    = 0, otherwise,

v_q = 1, if cell q is formed, and

    = 0, otherwise.

MODEL 1

Constraints

(2.3) $$\sum_{q} x_{ijq} = u_{ij}.$$

(The requirements for processing part j on the M_i's must be satisfied.)

(2.4) $$\sum_{j} x_{ijq} \leq \sum_{k} y_{ikq}.$$

(The total requirement for processing of parts on the M_i's in cell q must be met.)

(2.5) $$\sum_{q} y_{ikq} = 1.$$

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

(2.6)
$$\sum_{i,k} y_{ikq} \geq v_q \, e_{\min}.$$

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

(2.7)
$$\sum_{i,k} y_{ikq} \leq v_q \, e_{\max}.$$

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

(2.8)
$$x_{ijq} \leq u_{ij} * w_{jq}.$$

(A constraint to determine w_{jq}.)

(2.9)
$$v_{q+1} \leq v_q.$$

(Cells are formed in successive numerical order.)

Objective

(2.10)
$$\text{Minimize} \sum_{j,q} w_{jq}.$$

(Minimize the occurrence of inter-cell travel.)

We now solve a numerical instance of Model 1.

2.2 Example 1

$m = 8$, $n = 10$, $p = 4$, $e_{\min} = 4$, and $e_{\max} = 6$, and $(u_{ij})_{8 \times 10}$ and the part-machine utilizations and the part-machine routings are defined in Tables 1 and 2 respectively.

Table 1 about here.
Table 2 about here.

The optimal solution was identified using the XPRESS integer programming code on a Sun-Solaris computer. The optimal objective value = 12. (Note that, in general, the objective function value can never be smaller than n.) As part of the optimal solution, four cells are formed, as follows:

- Cell 1 contains: $M_1^1, M_2^1, M_3^1, M_4^1, M_5^1, M_5^2$,

- Cell 2 contains: $M_2^4, M_4^3, M_5^6, M_5^7, M_7^1$,

- Cell 3 contains: $M_2^3, M_4^2, M_5^5, M_6^2$, and

- Cell 4 contains: $M_2^2, M_3^2, M_5^3, M_5^4, M_6^1, M_8^1$.

Also,

- Cell 1 processes Parts 2, 4, 8,

- Cell 2 processes Parts 4 and 5,

- Cell 3 processes Parts 2, 6, 10, and

- Cell 4 processes Parts 1, 3, 7, 9.

The part-machine assignment is given Table 3. The cells formed and the flow of parts (with the machine utilization requirements given in parentheses) are illustrated in Figure 1. Note that in cell 1, M_2^1 and M_5^2 are dedicated to parts 4 and 8 respectively; in cell 2, M_5^7 is dedicated to part 5; in cell 3, M_5^5 is dedicated to part 6; and in cell 4, M_5^3 is dedicated to part 9.

Table 3 about here.

Figure 1 about here.

We now go on to generalize the previous model to include costs arising from machine modification.

3. The General Cell Formulation Problem

3.1 A description of the GCFP
We noted earlier that it is often important in practice to be able to reassign parts to be processed on individual machines of additional types in order to create better cell system 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 e_{max}, the maximum number of machines allowed in any cell. Machine modification of the nature just proposed is certainly feasible given the operational flexibility of modern, general-purpose machines currently in existence. Suppose that each given machine type, M_i say, ($i = 1, 2 ,..., m$) is initially capable of processing a given set of parts, as specified by the $(a_{ij})_{mxn}$ matrix, defined as follows:

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

We assume that any individual machine M_i^k say, can be modified so that it can also perform as a machine of at most one other type, at a given cost, thus increasing its operational flexibility. 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 type that each is extended to, will depend upon their capacity, which cells they are assigned to, and the corresponding modification costs. Of course, certain machine type modifications may be technically infeasible or prohibitively costly. In the former case,

as is usual in these circumstances, the costs of such modifications should be set at a relatively high level. In this paper we assume that a machine can be modified at most once.

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 all batches of each given part between a pair of given cells is known for the time horizon of the analysis. The overall objective is to minimize the total of all modification costs and of all inter-cell materials handling costs for the given time horizon. In order to make a useful comparison of costs for a given time horizon, the handling costs may have to be amortized as they are ongoing, whereas the modification costs are one-off. This point will be discussed further in the analysis of the numerical examples given later. The minimization of the total costs is achieved by deciding simultaneously:

For each individual machine:
- (i) Whether or not it is to be modified so that it can behave as at most one additional machine type,
- (ii) If it is to be modified, which additional machine type capability it will assume,
- (iii) To which cell it is assigned, and

For each individual part:
- (i) Its individual machine routing sequence, and
- (ii) Its utilization of each individual machine by which it is processed.

Care must be taken that the minimum and maximum number of machines allowed in each cell e_{min} and e_{max}, respectively, are satisfied for all cells created. We now develop a model of the GCFP, as an extension of Model 1.

3.2 The development of a model of the GCFP

To begin the development of a GCFP model, we introduce some additional notation. Let

Input parameters

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

g_{jqr} = the cost of transporting all batches of part j between cells q and r, for the given time horizon,

Decision Variables

z_{khi} = 1, if M_h^k is modified so that it also has the capabilities of machines of type M_i,

= 0, otherwise.

t_{hiqk} = the fraction of machine M_i^k (if the machine is assigned to cell q), that is used as a machine of type i,

s_{jqr} = 1, if part j is transported from cell q to cell r for processing, and

= 0, otherwise.

MODEL 2
Constraints

Constraints (2.3), (2.5) - (2.9) together with the following constraints (3.1) - (3.5):

(3.1) $$\sum_j x_{ijq} \leq \sum_{h,k} t_{hiqk}$$

(The generalization of constraint 2.4.)

(3.2) $$\sum_i t_{hiqk} \leq y_{hqk}$$

(Each assigned machine creates a unit of processing that may possibly be subdivided after machine modification.)

(3.3) $$\sum_{i|i \neq h} z_{khi} \leq 1$$

(Each machine cannot be modified to encompass more than one additional machine type.)

(3.4) $$t_{hiqk} \leq z_{khi} \qquad h \neq i$$

(Machine modification creates a fraction of a unit of modified processing.)

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

(If a part is processed on two machines, the variable indicating inter-cell movement is set to 1.)

Objective

(3.6) $$\text{Minimize} \quad \sum_{h,k,i} f_{hi} \cdot z_{khi} + \sum_{j,q,r} g_{jqr} \cdot s_{jqr}.$$

(The sum of all inter-cell travel costs and all machine modification costs.)

We now develop heuristics for the GCFP based on the greedy strategy.

3.3   The development of heuristic solution procedures for the GCFP
We first develop a heuristic for the GCFP based on the greedy strategy. Considering the solution to Example 1 as a solution to Model 2, we have, as there is no machine modification,

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

Referring to Figure 1, in calculating the objective function value, as defined by (3.6), we need focus only on the cost of inter-cell travel. This cost arises from: (i) part 2 travelling from cell 1 to cell 3, and (ii) part 4 travelling from cell 1 to cell 2. Thus, objective function value is

(3.7) $$g_{213} + g_{412} = 4(2.1.3) + 4(4.1.2) = 56.$$

There is the possibility of reducing the objective function value by reducing inter-cell travel by machine modification. Such reductions can be investigated in a systematic, greedy manner. This consideration suggests the following greedy heuristic for the GCFP. In the statement below, a "cell system" implies a feasible solution to the GCFP, that is, a solution that satisfies all the constraints of Model 2. An illustration of such a solution is given in Figure 1. Methods for the identification of such systems, without regard to machine modification costs, have been surveyed by Foulds and Neumann [13], and include those reported by Askin and Chiu [4], Askin and Standridge [5], and Neumann [26].

The FW1 (Greedy) Heuristic
Input: A cell system
Step
1. For the current cell system, identify for each part j that is subject to inter-cell travel:
   i. The sequence of cells that part j visits, say: $r^j_1, r^j_2, ..., r^j_{b-1}, r^j_b$.
   ii. The cheapest feasible combination of machine modifications (that maintain a cell system) in:
      a) Cell r^j_2, that enables the sequence to become: $r^j_2, ..., r^j_{b-1}, r^j_b$. (Eliminate travel the initial travel of part j, from cell r^j_1 to cell r^j_2.)
      b) Cell r^j_b, that enables the sequence to become: $r^j_1, r^j_2, ..., r^j_{b-1}$. (Eliminate the final travel of part j, from cell r^j_{b-1} to cell r^j_b.)
2. Implement the combination of machine modifications identified in Step (1) that reduces the objective function by the greatest amount to create a new current cell system. (Ties are settled by choosing the combination with the least number of modifications and, beyond that, arbitrarily.)
3. Return to Step (1) until no further objective function can be identified, in which case terminate.

We now illustrate the FW1 heuristic by applying it to the following numerical instance of Model 2.

Example 2

The input data is as for Example 1 with, additionally:
$f_{hi} = (h)(i)$ [the product of h and i] $h, i = 1, 2, ... 8$, (h not equal to i) and

$g_{jqr} = 4(j)(q)(r)$ $j = 1, 2, ..., 10$; $q, r = 1, 2, 3, 4$ (q not equal to r).

The cell system inputted is the solution to Example 1 given in Section 2.

Step 1(i)
The objective function value arises due to:
   (i)   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. Thus for $j = 2, b = 2$ and $(r^j_1, r^j_b) = (1, 3)$.
   (ii)   Part 4 travelling from cell 1 to cell 2 to be processed by (0.5) of M_4^3 and, subsequently, (0.2) of M_7^1. Thus for $j = 4, b = 2$ and $(r^j_1, r^j_b) = (1, 2)$.

Step 1(ii)
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. As there is insufficient machine capacity, it is infeasible, to process either part 2 in cell 3 or part 4 in cell2.

Step 2

It is cheapest to modify M_1^1 and M_3^1 so that they also have the capabilities of machines of types 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 alternative of processing part 2 solely in cell 1 entails modifying both M_1^1 and M_3^1 so that they have the capabilities M_4 and M_7 respectively. The objective function value of this alternative is

$$f_{14} + f_{37} + g_{412} = (1)(4) + (3)(7) + 4(4)(1)(2) = 57,$$

which is more expensive than the original solution.)

Step 3

As there are no further machine modifications that bring about a reduction in the objective function, the heuristic is terminated. 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 over the solution to Example 1. The part-machine assignment is given Table 4. Machine-cell assignment is shown in the table in square parentheses and the additional machine capability, due to modification, is shown in round parentheses. The cells formed and the flow of parts are illustrated in Figure 2. It can be seen in Figure 2 that part 4 is processed twice on $M_1^1(7)$, initially as (.1) of an M_1 and finally as (.2) of an M_7. Thus the corresponding entry in Table 4 is (.3).

Table 4 about here.
Figure 2 about here.

We now go on to develop a more effective heuristic, that is based on a tabu search learning metaheuristic. (The reader is referred to the book by Glover and Laguna [15] for an introduction to tabu search.) We provide a simple overview of the heuristic first, follwed by a more rigorous description.

Once again, as with the FW1 heuristic, the procedure requires, as input, a cell system. The method proceeds, at each iteration, by altering the cell system into another (feasible) cell system by employing one of the following transformations:

1   Reassign an individual machine from its cell to another cell,
2   Interchange the cell locations of two individual machines that are assigned to two different cells,

3 Modify the capability of an individual machine by:
  a. Adding the capability of a second machine type,
  b. Removing the capability of one of its two machine types, or
  c. Replacing one of its machine type capabilities by a new machine type capability.

The cell system of least objective function value that can be identified by implementing one of the above transformations is recorded and updated to prevent cycling. Methods for identifying, the processing sequence of individual machines for each part have been surveyed by Foulds and Neumann [13].

The Choice Rule
Implement the transformation that produces the new cell system with the least objective function value.

Tabu Tenure
Make tabu, for a given number N, iterations, the reverse transformation just implemented. (A reverse transformation is one that would return the cell system to that from which it has just been transformed.)

Aspiration Criterion
Override the tabu tenure restriction if a different transformation produces a new cell system that is a cell system with the lowest objective function value yet identified.

The FW2 (Tabu Search) Heuristic
Input:
A cell system and,
T_{min}, T_{max} the minimum and maximum tabu tenures, respectively.
Step
1 Set the tabu tenure T, to be T_{max}. Empty the tabu list and the diversification list. Set the current cell system to be the inputted cell system.
2 Evaluate all the cell systems that can be created by a single transformation.
3 If a cell system with a lower objective function value is found, transform the cell system to the one with the lowest objective function value and go to Step 1, otherwise go to Step 4.
4 If none of the cell systems found in Step 2 is better than the best encountered current cell system, select the cell system found in Step 2 with the least objective function value that is created by a transformation not in the tabu list. (This defines a new tabu transformation.) Update the tabu list by adding the transformation just identified, and remove all transformations that have remained in the list longer than N.
5 If this iteration is one of the first two iterations following Step 1 or Step 8, add the new tabu transformation to the diversification list.
6 If the transformation to the current cell system just performed in Step 4 increases the value of the objective function, decrease the increase the tabu tenure by 1.
7 If $T > T_{min}$, go to Step 2.
8 If $T = T_{min}$, diversify by:
  a. Set the current cell system to be the best encountered cell system.

    b. Set $T = T_{max}$.

    c. Create a tabu list containing all entries in the diversification list, with the present iteration as their recorded entry to the tabu list (that is, after T additional iterations, all of them will be removed from the tabu list).

    d. Go to Step 2.

The FW2 heuristic was applied to Example 2. The inputted cell system was the final solution produced by the FW1 heuristic when it was applied to Example 2, as illustrated in Table 4 and Figure 2. T_{min}, T_{max} were set at 2 and 10 respectively. In the final solution produced by FW2, four cells are formed, as follows:

- Cell 1 contains: $M_1^1, M_2^1, M_4^1, M_5^1,$

- Cell 2 contains: $M_2^4, M_4^3, M_5^6, M_5^7, M_7^1,$

- Cell 3 contains: $M_2^3, M_3^1, M_4^2, M_5^2, M_5^5, M_6^2,$

- Cell 4 contains: $M_2^2, M_3^2, M_5^3, M_5^4, M_6^1, M_8^1,$

- Cell 1 processes Parts 4,

- Cell 2 processes Parts 5 and 6,

- Cell 3 processes Parts 2, 8, 10, and

- Cell 4 processes Parts 1, 3, 7, 9.

Further:

- M_1^1 is modified so that it also has the capabilities of machines of type M_7,

- M_2^3 is modified so that it also has the capabilities of machines of type M_4,

- M_2^4 is modified so that it also has the capabilities of machines of type M_5, and

- M_3^1 is modified so that it also has the capabilities of machines of type M_1.

The part-machine assignment is given Table 5. As there is no inter-cell travel, the matrix is in block-diagonal form, as indicated by the shaded entries. The cells formed, and the flow of parts, are illustrated in Figure 3.

Table 5 about here.

Figure 3 about here.

In Figure 3, it can be seen that part 8 is processed twice on $M_3^1(1)$, first as (.1) of an M_1 and then as (.2) of an M_3. Thus the corresponding entry in Table 5 is (.3). The objective value is

(3.8) $f_{17} + f_{24} + f_{25} + f_{31} = (1)(7)+(2)(4)+(2)(5)+(3)(1) = 28.$

The preceding solution was identified as an optimal solution to the corresponding numerical instance of Model 2, by using the XPRESS integer programming code on a Sun-Solaris computer. In practice, the cost of this solution, which arises solely from

machine modification costs, should be compared with solutions such as that to Example 1, given in Section 2. Although the cost in (3.8) is half that of (3.7), care must taken in making comparisons in order to decide if the machine modifications specified in "optimal" solutions to Model 2 are worthwhile. This is because ongoing costs of transport are being added to, and being compared with, the one-off costs of modification. Obviously, the definitions of the f_{hi}'s and the g_{jqr}'s, the time horizon chosen, and the calculation of the net present value of the total transport costs over the time horizon must be made carefully if a rational decision on machine modification selection is to be made.

4 Extensions to Model 2

Extensions of the model given in Section 3 are possible. One extension, which seems a natural consequence of considerations of machine utilisation, is 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 processes only part 1. This part requires a total of 3.0 units of machine processing, so it could be completely processed in cell 1, using only three machines. Thus the model could be extended to include the set up costs of allocating machines to cells, and if these costs were sufficiently high relative to the cost of modifying machines, then it might be possible to reduce the total cost by not using certain machines. Such machines could then be dedicated to new tasks in other cells. We plan to publish our investigation into this proposed extension, and others, elsewhere. We now summarize the paper and present our conclusions.

5. Summary and Conclusions

We have reviewed some issues in the formulation of manufacturing cell systems, including the assigning of parts to individual machines, the grouping of individual machines into cells, and the modification of individual machines to increase their part processing capability. We have presented a mixed integer programming model that combines these activities, with the objective of minimizing overall machine modification and inter-cell travel costs. We term this model the General Cell Formation Problem. The GCFP heuristics that have been proposed and analyzed are suitable for the design of large-scale systems. We believe that the resulting solution procedures that have been developed from the models will become useful tools for production planners.

Glossary of notation

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

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

d_j = the demand per period for part j,

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 all batches of part j between cells q and r, per period,

M_i denotes machine type i,

m = the number of machine types,

M_i^k denotes the kth 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,

$\hat{p}$ = an estimate of the maximum possible number of cells

q_j = the batch size of part j,

r_q^j = the qth cell visited in the processing sequence part j,

s_{ij} = the required set up time per batch or lot of part j for any M_i,

t_{ij} = the processing time (without set up time) for one unit of part j on any M_i,

T_{min}, T_{max} the minimum and maximum tabu tenures, respectively for the FW2 heuristic,

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

Decision Variables

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

= 0, otherwise.

t_{hiqk} = fraction of machine M_i^k (which is assigned to cell q), which is used as a machine

of type i,

v_q = 1, if cell q is formed, and

= 0, otherwise,

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

= 0, otherwise,

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

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

= 0, otherwise,

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

= 0, otherwise.

References

1. A. I. Abdelmola, S. M. Taboun and S. Merchawi, Productivity optimization of cellular manufacturing systems. Computers & Industrial Engineering, 35 (1998) 403-406.

2. M. U. Ahmed, N. U. Ahmed, and U. Nandkeolyar, A volume and material handling cost based heuristic for designing cellular manufacturing cells. Journal of Operations Management. 10 (1991) 488-511.

3. Y. Arzi, J. Bukchin and M. Masin, An efficiency frontier approach for the design of cellular manufacturing systems in a lumpy demand environment, European Journal of Operational Research. 134 (2001) 346-364 .

4. R.G. Askin and K.S. Chiu, A graph partitioning procedure for machine assignment and manufacturing cell formation in group technology. International Journal of Production Research. 28 (1990) 1555-1572.

5. R.G. Askin and C.R. Standridge, Modeling and Analysis of Manufacturing Systems. John Wiley, New York (1993).

6. A. Atamani, R.S. Lashkari and R.J. Caron, A mathematical programming approach to joint manufacturing cell formation and operation allocation in cellular manufacturing. International Journal of Production Research. 33 (1995) 1-15.

7. R.E. Billo, A Design Methodology for Configuration of Manufacturing cells. Computers & Industrial Engineering. 34 (1998) 63-75.

8. J.L. Burbidge, The simplification of material flow systems. International Journal of Production Research. 20 (1982) 339-359.

9. Q. Cao and M.A. McKnew, Partial termination rule of Lagrangian relaxation for manufacturing cell formation problems. Computers and Operations Research. 25 (1998) 159–168.

10. C. Caux, R. Bruniaux and H. Pierreval, Manufacturing cell formation with alternative process plans and machine capacity constraints: A new combined approach. International Journal of Production Economics. 64 (2000) 279-284.

11. 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.

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. L. R. Foulds and K. Neumann, Techniques for Machine Group Formation in Manufacturing Cells. Mathematical and Computer Modelling. Mathematical and Computer Modelling (2003) (to appear).

14. 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.

15. F. Glover and M. Laguna, Tabu Search, Kluwer Academic, Boston, 1997.

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. International Journal of Production Research 27 (1989) 1287-1304.

18. A. K. Jain, R. G. Kasilingam, and S. D. Bhole, Joint consideration of cell formation and tool provisioning problems in flexible manufacturing systems. Computers & Industrial Engineering. 20, (1991) 271-277.

19. K.R. Kumar, A. Kusiak, and A. Vannelli, Grouping of parts and components in flexible manufacturing systems. European Journal of Operational Research. 24 (1986) 387-397.

20. A. Kusiak and W.S. Chow, Decomposition of Manufacturing Systems. Journal of Robotics and Automation. 4 (1988) 457 – 471.

21. A. Kusiak and S.S. Heragu, Group Technology. Computers and Industry. 9 (1987) 83-91.

22. R. Logendran, A binary integer programming approach for simultaneous machine-part grouping in cellular manufacturing systems. Computers & Industrial Engineering. 24 (1993) 329-336.

23. S. Lozano, D. Canca, F. Guerrero and J. M. García, Machine grouping using sequence-based similarity coefficients and neural networks. Robotics and Computer-Integrated Manufacturing. 17 (2001) 399-404.

24. C. Moon and M. Gen, A genetic algorithm-based approach for design of independent manufacturing cells. International Journal of Production Economics. 60-61 (1999) 421-426.

25. 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.

26. K. Neumann, Produktions und Operations Management. Springer-Verlag, Berlin (1996).

27. H. Pierreval and M-F Plaquin, An Evolutionary Approach to Multicriteria Manufacturing Cell Formation. Int. Trans Opl. Res. 5 (1998) 13-25.

28. M-F Plaquin and H. Pierreval, Manufacturing cell formation using evolutionary algorithms with certain constraints. Int J. Production Economics. 64 (2000) 267-278.

29. R. Rajagopalan and J.L. Batra, Design of manufacturing cellular production systems: a graph-theoretic approach. International Journal of Production Research. 13 (1975) 567-579.

30. 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 Engineering. 35 (1998) 3-20.

31. R. Sundaram and S-S. Fu, Group cell formation – Some new insights, Computers & Industrial Engineering. 13 (1987) 267-276.

32. S. M. Taboun, N. S. Merchawi and T. Ulger, A two-stage model for cost effective part family and machine cell formation. Computers & Industrial Engineering. 34 (1998) 759-776.

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

34.   A. Vannelli and R.G. Hall, An eigenvector solution methodology for finding part-machine families. International Journal of Production Research. 31 (1993) 325-349.

35.   J. Wang and C. Roze, Formation of machine cells and part families in cellular manufacturing: an experimental study. Computers & Industrial Engineering. 29 (1995) 567-571.

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

37.   U. Wemmerlov and N.L. Hyer, Research issues in manufacturing cellular manufacturing. International Journal of Production Research. 25 (1987) 412-422.

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

39.   Z. Zhu, R. B. Heady and S. Reiners, An efficient zero-one formulation of the cell formation problem. Computers & Industrial Engineering. 28 (1995) 911-916.

Biographies

Leslie Foulds is Professor in the Department of Management Systems at Waikato University. His principal research areas are facility location and layout design, decision support systems, and applications of graph theory. He has published in journals such as Management Science Computers and Operations Research, Annals of Operations Research and Journal of the Operational Research Society and is the author of an established book on Graph Theory.

John Wilson is Professor of Operational Research in the Business School at Loughborough University. His principal research areas are integer programming, use of logic in mathematical programming and model validation. He has published in such journals as Naval Research Logistics, Annals of Operations Research, Computers and Operations Research and Journal of the Operational Research Society and is the joint author of a book on Business Optimisation.

Captions
Table 1.   The part-machine utilization for Example 1.
Table 2.   The machine sequences (routings) of parts for Example 1.
Table 3.   The part-machine utilization matrix of the optimal solution to Example 1.
Table 4.   The part-machine utilization matrix for the solution to Example 2 produced by the FW1 heuristic.
Table 5.   The part-machine utilization matrix for the solution to Example 2 produced by the FW2 heuristic.
Figure 1   The cells formed and the part flows, within and between cells, for Example 1.
Figure 2   The cells formed and the part flows, within and between cells, for the solution to Example 2 produced by the FW1 heuristic.
Figure 3   The cells formed and the part flows, within cells, for the solution to Example 2 produced by the FW2 heuristic.

Parts /M_i	1	2	3	4	5	6	7	8	9	10	$\sum_j u_{ij}$	e_i	$\overline{e}_i$
M_1	0.0	0.3	0.0	0.1	0.0	0.0	0.0	0.1	0.0	0.0	0.5	1	0.5
M_2	0.0	0.0	0.0	1.0	0.5	0.0	0.9	0.0	0.1	0.6	3.1	4	0.775
M_3	0.0	0.0	0.9	0.0	0.0	0.0	0.0	0.2	0.1	0.0	1.2	2	0.6
M_4	0.0	1.2	0.0	0.9	0.5	0.4	0.0	0.0	0.0	0.0	3.0	3	1.0
M_5	0.4	0.2	0.0	0.8	1.1	1.0	0.6	1.0	1.0	0.0	6.1	7	.871
M_6	0.0	0.7	0.0	0.0	0.0	0.0	0.8	0.0	0.0	0.3	1.8	2	0.9
M_7	0.0	0.0	0.0	0.2	0.2	0.0	0.0	0.0	0.0	0.0	0.4	1	0.4
M_8	0.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.5	0.0	0.7	1	0.7

Table 1

Parts	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_6

Table 2

Parts /M_i^k	1	2	3	4	5	6	7	8	9	10	$\sum_j u_{ij}$
M_1^1	0	.3	0	.1	0	0	0	.1	0	0	.5
M_2^1	0	0	0	1.0	0	0	0	0	0	0	1.0
M_2^2	0	0	0	0	0	0	.9	0	.1	0	1.0
M_2^3	0	0	0	0	0	0	0	0	0	.6	.6
M_2^4	0	0	0	0	.5	0	0	0	0	0	.5
M_3^1	0	0	0	0	0	0	0	.2	0	0	.2
M_3^2	0	0	.9	0	0	0	0	0	.1	0	1.0
M_4^1	0	.6	0	.4	0	0	0	0	0	0	1.0
M_4^2	0	.6	0	0	0	.4	0	0	0	0	1.0
M_4^3	0	0	0	.5	.5	0	0	0	0	0	1.0
M_5^1	0	.2	0	.8	0	0	0	0	0	0	1.0
M_5^2	0	0	0	0	0	0	0	1.0	0	0	1.0
M_5^3	0	0	0	0	0	0	0	0	1.0	0	1.0
M_5^4	.4	0	0	0	0	0	.6	0	0	0	1.0
M_5^5	0	0	0	0	0	1.0	0	0	0	0	1.0
M_5^6	0	0	0	0	.1	0	0	0	0	0	.1
M_5^7	0	0	0	0	1.0	0	0	0	0	0	1.0
M_6^1	0	0	0	0	0	0	.8	0	0	0	.8
M_6^2	0	.7	0	0	0	0	0	0	0	.3	1.0
M_7^1	0	0	0	.2	.2	0	0	0	0	0	.4
M_8^1	.2	0	0	0	0	0	0	0	.5	0	.7

Table 3

Parts /$\mathbf{M_i^k}$	1	2	3	4	5	6	7	8	9	10	$\sum_j u_{ij}$
[1] M_1^1 (7)	0	.3	0	.3	0	0	0	.1	0	0	.7
[1] M_2^1	0	0	0	1.0	0	0	0	0	0	0	1.0
[4] M_2^2	0	0	0	0	0	0	.9	0	.1	0	1.0
[3] M_2^3	0	0	0	0	0	0	0	0	0	.6	.6
[2] M_2^4	0	0	0	0	.5	0	0	0	0	0	.5
[1] M_3^1 (4)	0	0	0	.5	0	0	0	.2	0	0	.7
[4] M_3^2	0	0	.9	0	0	0	0	0	.1	0	1.0
[1] M_4^1	0	.6	0	.4	0	0	0	0	0	0	1.0
[3] M_4^2	0	.6	0	0	0	.4	0	0	0	0	1.0
[2] M_4^3	0	0	0	0	.5	0	0	0	0	0	.5
[1] M_5^1	0	.2	0	.8	0	0	0	0	0	0	1.0
[1] M_5^2	0	0	0	0	0	0	0	1.0	0	0	1.0
[4] M_5^3	0	0	0	0	0	0	0	0	1.0	0	1.0
[4] M_5^4	.4	0	0	0	0	0	.6	0	0	0	1.0
[3] M_5^5	0	0	0	0	0	1.0	0	0	0	0	1.0
[2] M_5^6	0	0	0	0	.1	0	0	0	0	0	.1
[2] M_5^7	0	0	0	0	1.0	0	0	0	0	0	1.0
[4] M_6^1	0	0	0	0	0	0	.8	0	0	0	.8
[3] M_6^2	0	.7	0	0	0	0	0	0	0	.3	1.0
[2] M_7^1	0	0	0	0	.2	0	0	0	0	0	.2
[4] M_8^1	.2	0	0	0	0	0	0	0	.5	0	.7

Table 4

Parts /[Cells]M_i^k	4	5	6	2	8	10	1	3	7	9	$\sum_j u_{ij}$
[1] M_1^1 (7)	.1	0	0	0	0	0	0	0	0	0	0.1
[1] M_2^1	**1.0**	0	0	0	0	0	0	0	0	0	1.0
[1] M_4^1	**.9**	0	0	0	0	0	0	0	0	0	0.9
[1] M_5^1	**.8**	0	0	0	0	0	0	0	0	0	0.8
[2] M_2^4 (5)	0	**.5**	**0**	0	0	0	0	0	0	0	0.5
[2] M_4^3	0	**.5**	**0.4**	0	0	0	0	0	0	0	0.9
[2] M_5^6	0	**0**	**1.0**	0	0	0	0	0	0	0	1.0
[2] M_5^7	0	**1.0**	**0**	0	0	0	0	0	0	0	1.0
[2] M_7^1	0	**.2**	**0**	0	0	0	0	0	0	0	0.2
[3] M_2^3 (4)	0	0	0	**0.2**	**0**	**0.6**	0	0	0	0	0.8
[3] M_3^1 (1)	0	0	0	**0.3**	**0.3**	**0**	0	0	0	0	0.6
[3] M_4^2	0	0	0	**1.0**	**0**	**0**	0	0	0	0	1.0
[3] M_5^2	0	0	0	**0.2**	**0**	**0**	0	0	0	0	0.2
[3] M_5^5	0	0	0	**0**	**1.0**	**0**	0	0	0	0	1.0
[3] M_6^2	0	0	0	**0.7**	**0**	**0.3**	0	0	0	0	0.3
[4] M_2^2	0	0	0	0	0	0	**0**	**0**	**0.9**	**0.1**	1.0
[4] M_3^2	0	0	0	0	0	0	**0**	**0.9**	**0**	**0.1**	1.0
[4] M_5^3	0	0	0	0	0	0	**0**	**0**	**0**	**1.0**	1.0
[4] M_5^4	0	0	0	0	0	0	**0.4**	**0**	**0.6**	**0**	1.0
[4] M_6^1	0	0	0	0	0	0	**0**	**0**	**0.8**	**0**	0.8
[4] M_8^1	0	0	0	0	0	0	**0.2**	**0**	**0**	**0.5**	0.7

Table 5

8 M_5^2 ← 8 (1.0) ← M_3^1

M_4^3 ← 5(.5) ← M_2^4 ← 5(.5)

8 (.2)

M_2^1

4 (1.0)

4 (.8)

4(.5)

5(1.0)

5(.1)

M_5^6 M_5^7

2 (.3)
4 (.1) M_1^1 → 2 (.2) → M_5^1
8 (.1)

4

5

5

5(.2)

2 (.6) 4 (.4)

2 (.6)

M_4^1 → 4

4

4(.2)

M_7^1 → 4
→ 5

Cell 1

2

Cell 2

6 (.4) → M_4^2 → 2

2

2 (.7)

M_5^3 ← 9(1.0) ← M_3^2
← 3(.9)
→ 3
← 9(.1)

6 (1.0)

9(.1)

6 ← M_5^5

M_2^2 ← 7(.9) ← M_5^4 ← 1(.4)
← 7(.6)

9(.5)

1(.2)

2
10 M_6^2

7(.8)

M_8^1 → 1
→ 9

10 (.3)

10(.6) → M_2^3

M_6^1 → 7
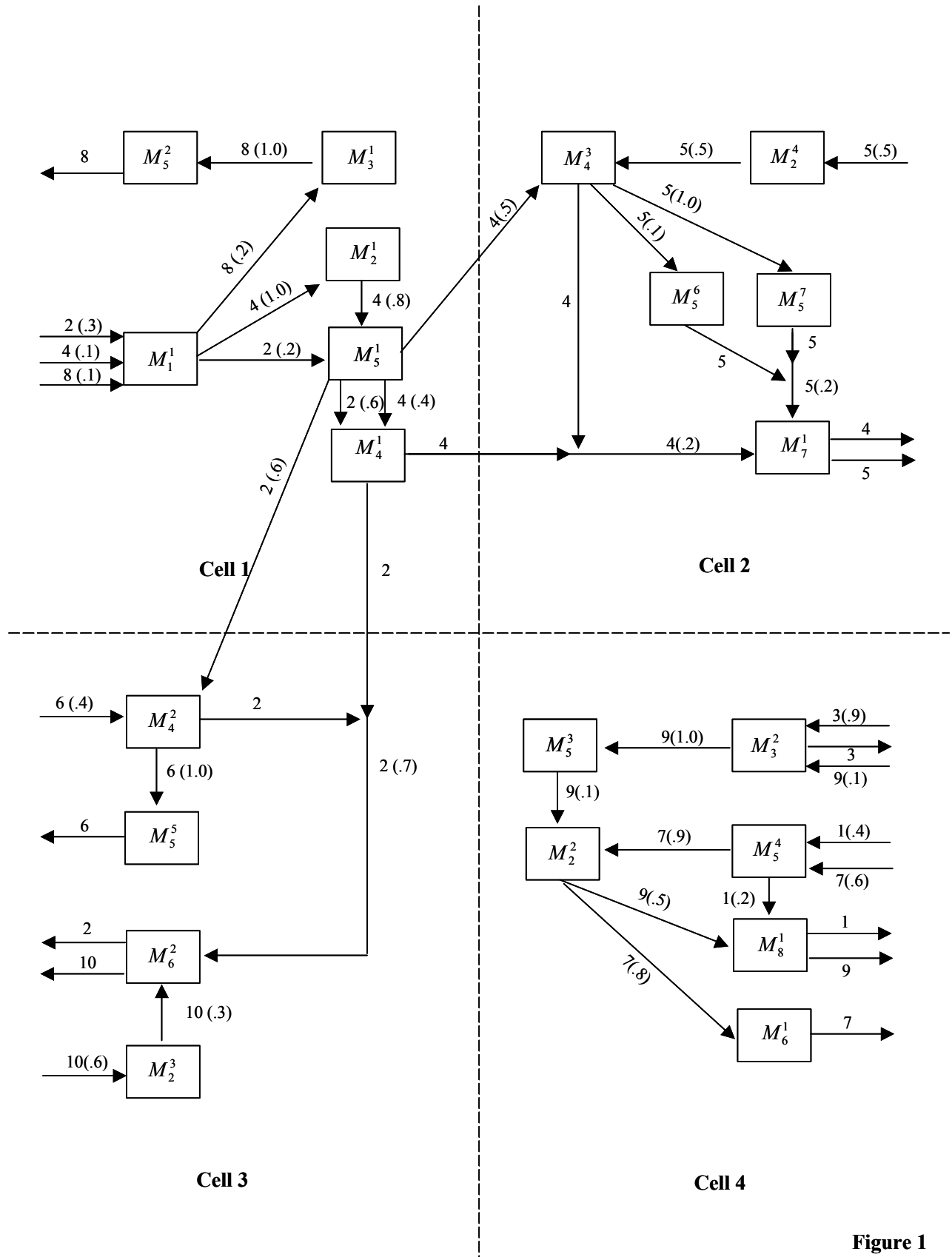
Cell 3
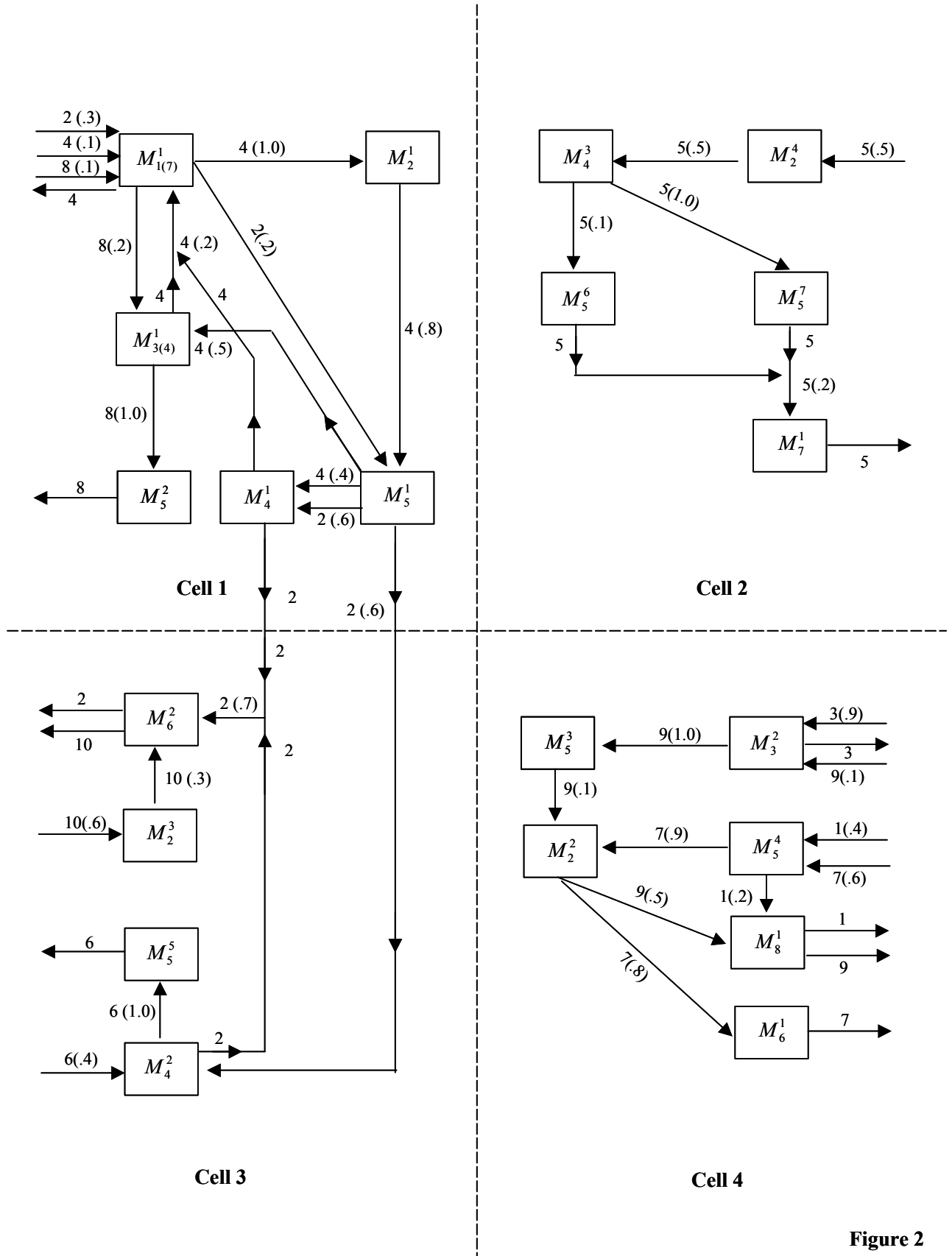
Cell 4
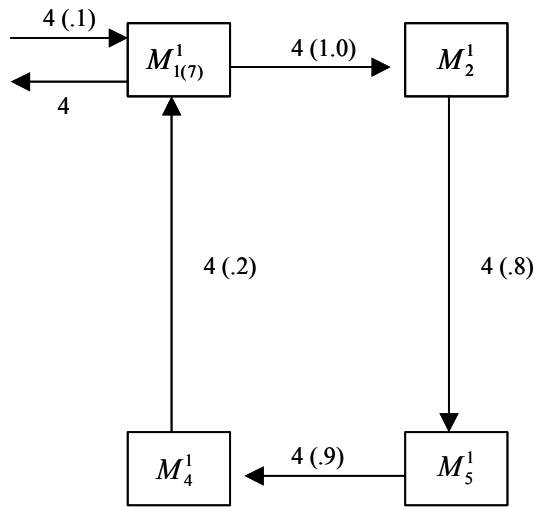
Figure 1

Figure 2

Cell 1
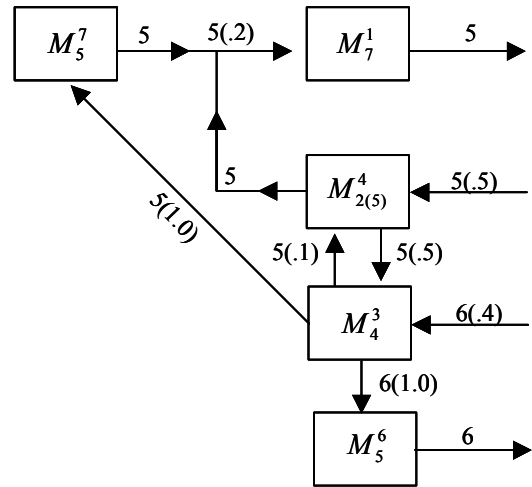
Cell 2

Cell 3

Cell 4

Figure 3