Application of a simple method of cell design accounting for product demand and operation sequence

Janet Efstathiou
Department of Engineering Science, University of Oxford, Oxford, UK

Peter Golby
Deloitte & Touche, London, UK

Keywords
Cells, Spreadsheets, Case studies, Demand, Sequencing, Manufacturing

Abstract
Manufacturing facilities may simplify their operations by converting from a process-based layout to manufacturing cells. Mathematically, many possible configurations of cells exist, so it may prove computationally infeasible to analyse them all. Also, some current methods of cell design do not take account of the pattern of demand of the existing products or the sequence of the operations that are performed on the products. Presents a simple method of designing manufacturing cells, which uses product demand and operations sequence to design feasible cells, while remaining computationally simple. The method uses a standard spreadsheet tool, so it is accessible to a wide range of manufacturing facilities. The method is illustrated with an actual application to a press shop manufacturing over 200 products on 20 presses.

Introduction
Manufacturers are aware of the importance of delivering reliably and predictably to their customers. This important aspect of business may be facilitated by organising the shop-floor as a series of manufacturing cells rather than as a process-based layout. However, the conversion from a process-based layout to a cell-based layout requires the generation and assessment of many possible cell configurations. Techniques are offered in the literature, but these are computationally demanding and not all methods take account of the pattern of demand or the sequence of operations. Furthermore, they may require specialist analytical skills and facilities in order to be implemented. While the methods offered in the literature may be robust for very large facilities with many machines and products, the methods may be too sophisticated for facilities with a much smaller number of products. Furthermore, a cell-based layout that involves a large amount of inter-cell transfer is likely to be unstable to fluctuations in demand and operational performance. Other computational results suggest that cells involving long sequences of processes with little buffering between machines can reduce throughput considerably under dynamic manufacturing conditions (Calinescu et al., 1998).

We offer a cell design method that takes the pattern of demand of the products and the sequence of operations into account to produce a design of a cell-based layout. This method is quick and simple to apply, since it uses a widely available spreadsheet computer program. The method was developed and tested on an actual manufacturing facility, which wanted to make over 200 different products on 20 machines. The results of the application to this domain are reported.

The next section reviews briefly the literature on the design of cell-based layouts, and the use of simple programming tools in manufacturing industry. The following section explains the new method for cell design and its data requirements. This is followed by the results of the application of this method to the manufacturing facility. The paper concludes with a summary and short discussion.

Previous methods for the design of cell-based layouts
There is an extensive literature on methods for the design of cell-based layouts. These methods are often based on an incidence matrix, which indicates, with a 0 or 1, which machines each product requires during manufacture. By rearranging the rows and columns of the matrix, the elements of the matrix may fall into clusters, with the possible exception of a few outliers. These clusters of products and machines can be used, therefore, as the combination of machines that may be used to create each cell, and the groupings of products that should be processed on each cell.

There are several weaknesses to this approach, which arise from the inadequacy of the 0-1 representation of a manufacturing facility. Features that are omitted include the sequence of operations, the capacities of machines of each type, product demand etc. As well as the inadequacy of the 0-1 representation, the computational complexity of the solution methods makes such approaches difficult to apply in practice.

For a matrix of $n$ rows and $m$ columns, there are $n!m!$ possible ways of arranging the matrix. For a large facility, this quickly leads to a computationally infeasible number of combinations. Re-arranging the rows and columns of the matrix to obtain clusters of...
machines is an NP-complete problem, and many methods have been used to tackle it. In order to avoid the computational difficulties, heuristic algorithms have been used to search more effectively the space of possible matrix arrangements. However, the heuristic algorithms are likely to become complex too, placing themselves beyond the reach of many facilities, so that it may not be attempted or may not be carried out effectively. However, these methods still fail to address the inadequacy of the underlying 0-1 representation.

One of the problems of the 0-1 representation, as mentioned above, is the inability to represent demand for each product. Since the incidence matrix contains only 0 and 1, a low demand product receives the same weight during the cell design as a high demand, high profit product. Cell designs which unduly weight the low value products can only be weeded out by a process of testing and comparison of each design after the designs have been generated.

Methods which take account of demand on each machine include Wu (1998), which permits product to flow in both directions through a cell and to use a machine in a cell more than once. Tang and Abdel-Malek (1996) use a phased procedure for developing the clusters of machines and then building a master flow network, in order to take account of the practicalities of real life factories. Dahel (1995) models both demand and machine sequence and uses a constraint relaxation method to find the allocation of machines to cells and the bottleneck resources, while minimising intercell traffic. Other methods that take account of machine sequence include Nair and Narendran (1998), who use a non-hierarchical method to identify machine clusters. Wei and Gaither (1990) present a heuristic algorithm that takes account of the capacity of the processes. Suresh et al. (1995) present a hierarchical method for cell formation based on a three-stage process of identifying part families, forming cells and minimisation of inter-cell traffic. Kang and Wemmerlöv (1993) propose a method of cell design that explicitly allows products to move between cells.

An important consideration in practice, and often overlooked in the literature, is that the cell design problems, which are encountered in reality, have a different level of complexity from those used to test the algorithms developed in the literature. In some cases, the search space of solutions that would be practical and feasible in the factory may be quite small. Actual constraints of physical space, number of machines etc. may limit the size of the cells that are designed. It may not be feasible or desirable for products to flow in more than one direction through the cell. Similarly, it may not be desirable for products to skip or repeat processes within the cells, since this may complicate scheduling and tie up the cell for an unduly long time. For processes that have an inherent variability in processing, it is unwise to set up cells with more than four or five machines, since the natural variability and unpredictability will need to be smoothed out with some buffering, or else low-through-put will result. Rather than searching through a very large search space, the real problems of industry are likely to need to find the best of a small number of solutions, or to find which constraints to relax successively in order to generate any solutions.

We may summarise the problems of existing cell design methods as follows:
- computationally demanding;
- may require specialist programming skills and tools;
- do not always take account adequately of the existing or planned demand on products;
- do not take account of sequence of operations;
- may be appropriate for designing infeasibly large cells;
- produce cell designs that make scheduling difficult;
- do not take account of practical constraints.

Despite the use of MRP, MRPII and other highly automated, computer integration techniques, many manufacturing facilities make use of informal, spreadsheet-based tools to solve both day-to-day and more strategic design problems. For example, Beversluis and Jordan (1995) use a spreadsheet for capacity planning and scheduling. We shall follow this example and make use of spreadsheet tools wherever possible.

A new method for the design of cell-based layouts

This section will describe the method for generating manufacturing cells using a computer-based spreadsheet. The data requirements of the method will be explained, followed by the use of the pivot table tool in the spreadsheet, here Excel 7. A simple method of generating sequences of machines for cells will be presented, followed by a discussion on the aspects of the cell design which are important for implementation in the factory.
Data requirements
For each product that is made in the facility, the following data are required:
- the sequence of machines through which the product passes during manufacture;
- the current or anticipated level of demand.

The data may be arranged as shown in Table I. In this example, each product may undergo up to three different operations (denoted as O1, O2 and O3), which may be performed on machines of type M1, M2 and M3. Each product is processed by no more than three machines. The sequence of operations undergone by each product is listed in the row of the Table, together with the demand for that product in the final column. Thus, product P0 has a total demand of 40 and is subject to three operations on machines of type M1, M2 and M3. Note that some products require only up to one operation (i.e., Product P0 requires processing on a machine of type M3 only), and some require up to three (i.e., Products P0, P1, P2 and P4). Note also that a product may require processing on a machine of a particular type more than once (e.g., Product P1 is processed on machine type M1 again and then on machine type M2).

Use of pivot table
A pivot table in Excel is a table generated by grouping together data from individual records. Data can be gathered together and summarised according to user-selected data items. In this way, overall trends or patterns may be identified and displayed. In this application, we use the pivot table data tool to identify which sequences of presses have the largest demand.

This may best be understood through an example. With the data in the format of Table I, Excel’s pivot table tool may be invoked. To set up the pivot table in Excel, it should be generated with the first operation in the processing sequence as the row heading, and the remaining operations as the contents of the columns. The cells of the table should contain the sum of demand. The pivot table generated from Table I is shown in Table II.

This shows the total demand for each of the occurring sequences of machines, exactly as generated by Excel. In this example, O1 is used as the row heading, and O2 and O3 as the column headings. To enable explanation of the contents of the table, each row and cell are labelled with a capital letter to indicate the row and a number to indicate the column. The pivot tables can become difficult to interpret, because of all the sequences that have to be displayed.

The contents of Table II may be interpreted as follows. The first operation in the sequence is shown under O1, i.e., in the first column of the table. This means that M1, M2 or M3 may occur as the first operation in the production sequence for the products under consideration. The second and third operations are found in rows B and C respectively of Table II. (This is denoted by O2 and O3 in cells A2 and A3.) Thus, the total demand for the sequence M1, M1, M2 may be found by reading across row D (i.e., the row headed M1), finding M1 in row B, and M2 in row C, under column 2. The total demand for the sequence M1, M2, M3 is therefore in cell D2, i.e., 5, the demand for product P1. A more complex example is the sequence M1, M2, M3. The total demand for this sequence is found in cell D4, reading M1 in row D and M2, M3 in column 4. The demand for this sequence is made up of 40 from product P0 and 50 from Product P2. Likewise, cell E6 (containing the value 45) contains the demand for the sequence M2, M3, M3 from Product P4. The one-machine sequence M3 is found at cell F8. The first operation, M3, is placed at row F and column 8 is headed “(blank) (blank)” to show that there are no operations at the second and third positions in the sequence.

An important feature of the pivot table is that it only contains those sequences of operations that actually occur in practice. With four machine types and up to three operations in the sequence, there are 84 possible ways of choosing up to three operations (84 = 4 + 4*4 + 4*4*4). The pivot table only includes 12 possible combinations, with non-zero demand for only four of those.

Designing the cells
Once the pivot table has been generated, it is possible to select the operation sequences with non-zero demand and sort them by decreasing demand. The sequences generated from this example are shown in Table III, sorted according to ascending sequence. Given the table of demand for each sequence, it is now possible to generate cell layouts to satisfy demand and meet other constraints on cell length, machine sequence, inter-cell movements etc.

| Table I |
| Example product data |
| Part number | Operation O1 | Operation O2 | Operation O3 | Demand |
| P0 | M1 | M2 | M3 | 40 |
| P1 | M1 | M1 | M2 | 5 |
| P2 | M1 | M2 | M3 | 50 |
| P3 | M3 | M2 | M3 | 7 |
| P4 | M2 | M3 | M3 | 45 |
The proposed method is based on construction of cell layouts, based on a core of machines that satisfy the highest demand short sequences. The method is suitable for the design of small cells, of up to five or six machines, and may be applied by hand.

It may be seen from Table III that the sequence with the highest associated demand is M1, M2, M3, with a demand of 97 (i.e. 90 + 7). This sequence may be extended by comparing with the next highest demand, which requires the sequence M2, M3, M3. This suggests the sequence M1, M2, M3, M3, which would be capable of satisfying demand of 142 (90 + 45 + 7).

At this point, the designer would need to consider the design criteria, before deciding whether to proceed with the design and add another M1 at the start of the cell. Alternatively, the extra machine could be used to establish a separate cell.

The algorithm is outlined in Figure 1. A flowchart summarising the logic of the method is presented in Figure 2. The algorithm is based on a sorted list of the demand for each product, with the associated machine sequence. This list can be generated and sorted using standard spreadsheet tools. The first sequence in the list is selected and used to initialise the first cell. The next sequence on the list is selected and compared with the current cell for common elements. If they exist, the union of the sequences is formed and a check is made that it satisfies constraints on cell size, manufacturing capacity and number of movements. If the constraints are satisfied, this sequence becomes the sequence for the current cell, and the algorithm moves on to the next product on the list. If there are no common elements or the constraints are not satisfied, then this product’s sequence forms the basis for a new cell and the process continues with this cell as the focus of attention. The algorithm halts when all the products are considered or the number of allowed cells is exceeded.

The algorithm is simple enough to be carried out by hand. In practical cases, there are so many special considerations that it would be extremely difficult to capture them all in a generic algorithm. A few of these considerations are listed below:

- When New_sequence yields an unsatisfactory design, shown in steps 7 and 8, the algorithm currently creates a new cell. Another possible way to proceed is to omit the sequence, S, which has caused the problem, and move on to the next product sequence. This may identify a sequence with a higher degree of overlap with the current cell, so that the maximum cell length criterion will not be violated. This may be a valid action if the product has low demand. Where cell design proves to be difficult, this could be implemented by only including products with demand above a variable threshold.

- Because of the particular constraints within a factory, there might not be a single limiting value on the maximum number of machines per cell. Another possibility is that cells involving particular machines may be limited to a specific length. These constraints could be modelled by amending the decision criterion at step 8.

- At step 4, initialise the design of the first cell to be the first product to have more than one machine. This may be appropriate if there are two products with near equal demand, but one requires one machine only and so causes cells to be created based around that machine type. If the other high demand product does not use that machine type, a cell that meets its demand may not be created. The combination of cells with different starting values of sequence_length may be combined and compared.

---

**Table II**

<table>
<thead>
<tr>
<th></th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>A</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>B</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>C</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>D</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>E</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>F</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>G</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table III**

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Total demand</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1 M2 M3</td>
<td>90</td>
</tr>
<tr>
<td>M2 M3 M3</td>
<td>45</td>
</tr>
<tr>
<td>M3</td>
<td>7</td>
</tr>
<tr>
<td>M1 M1 M2</td>
<td>5</td>
</tr>
</tbody>
</table>
The version of the algorithm presented here only considers amending the cell at the current value of Cell_count. This is feasible if the sorting of the product sequences at the start of the execution of the algorithm includes some sorting on sequence after demand, since this would bring similar sequences together. An alternative method would be that, when a new sequence is examined, it should be compared with all existing cell designs and the best of those designs selected. This would mean making steps 7 and 8 part of an inner loop that considers all cells up to and including that at Cell_count.

A more complex way of dealing with the problem of when to create a new cell is to allow the algorithm to branch at that point and consider alternative designs where a new cell is added to the current list of cells, or an existing cell is extended to include another machine. Table IV shows how this method would increase the number of possible designs for the example of Tables I to III.

- The version of the algorithm presented here only considers amending the cell at the current value of Cell_count. This is feasible if the sorting of the product sequences at the start of the execution of the algorithm includes some sorting on sequence after demand, since this would bring similar sequences together. An alternative method would be that, when a new sequence is examined, it should be compared with all existing cell designs and the best of those designs selected. This would mean making steps 7 and 8 part of an inner loop that considers all cells up to and including that at Cell_count.

- The current algorithm sorts the products by descending demand. Another method would be to sort the products by machine sequence, followed by descending demand. This would be likely to bias the cell designs to meet those sequences that occur first lexicographically, but would be likely to work well with the current algorithm’s method of considering just one cell at a time.

Many other practical considerations may need to be kept in mind. For example, inter-cell transfers may be feasible in some parts of the factory, but not in others. Similarly,
skipping machines may be allowed in some cells, but there may not be enough space in others. It would be very difficult to capture this large amount of flexibility in a comprehensible algorithm, so a reasonable compromise is to present an algorithm in enough detail, so that designers can adapt it to their own use.

**Design assessment criteria**

The cell design criteria that we use in this paper are:
- number of machines in the cell;
- demand met;
- demand which skips machines;
- demand which makes one inter-cell transfer;

---

**Figure 2**

Flow chart showing logic of cell design algorithm

1. Use Pivot Table to generate list of sequences and demand for every product.
   Sort the list by ascending sequence, and descending demand where more than one product has the same sequence.
2. Initialise first cell with top item on the list.
3. Find next sequence on list.
4. Do common elements exist between New_sequence and the current cell? (No)
5. Form the union of the new sequence and the current cell. Call this New_sequence.
6. Check that New_sequence obeys constraints on:
   - capacity
   - cell length
   - number of movements
7. Does New_sequence obey all constraints? (No)
   - Update current cell with New_sequence
   - Create a new cell and make it the current cell.
8. Repeat until allowed number of cells exceeded, or all products included.
• demand which makes more than one inter-cell transfer;
• expense of purchasing and installing machines;
• robustness to demand changes;
• scheduling feasibility.

It is assumed in these designs that a product cannot use a machine in the cell twice, i.e. use one machine for two operations. So, for the M1, M1, M2 sequence, two machines would be needed to provide operation 1.

The process of completing the cell design would require some assessment of the problems of scheduling a facility with this arrangement of cells. Where inter-cell transfers are allowed, the cells cannot be scheduled independently. However, designs 3 and 5 in Table IV would provide a greater degree of independence between the cells, because the transfer to the one-machine cell occurs at the end of the processing sequence, freeing up the four-machine cell for the next product. One of the advantages of the cell-based layout is the reduction in complexity of the scheduling problem. By permitting inter-cell transfers, this advantage is lost to a considerable extent, since any disruption to schedule in one cell must necessarily affect the other linked cell or cells.

Another important consideration is that all the demand for products P0, P2, P3 and P4 can be met using only four machines. A fifth machine, of type 1, is needed for product P1. The planner would need to take into account future demand for products in making the decision about the pattern of investment for the future. This could be done by changing the demand levels in Table I and regenerating the pivot table.

This section has presented a spreadsheet based method of tabulating and sorting the data required for processing cell design, taking account of the demand for the products produced. The next section of the paper will describe an application of this method to an actual pressing facility.

### Application to a press shop

The method described in section 3 was developed to assist a press shop that had to relocate its premises. In the previous location, the facility had been arranged as a process-based layout, with all the machines of a particular type located close together on the shopfloor. The facility had been manufacturing over 1,000 different products on over 60 machines. This required a substantial amount of transport of the work around the shopfloor from press to press. A diagram of this original layout may be seen in Figure 3. At the new location, there was space for only 20 machines, broken up into three or four small areas. Over 200 different products had to be made on these machines.

Under the process-based layout, the facility suffered from poor schedule adherence and unpredictable performance. The need to move to a new location provided an opportunity to simplify and improve the operation of the facility. The requirement of non-interrupted processing of each product, the space constraints and the need to simplify the operation of the facility created the possibility of changing to a cell-based layout.

In order to ensure good quality of the product, the facility required that all the presses for a particular job should be set up, so that a job could run through all the presses without interruption. This meant that jobs could not be left partly completed, awaiting a press to become available. Thus, a product could not use a machine in one cell for two different operations.

Seven different kinds of machine were included in the analysis. The products need operations by between one and six machines. These operations may be on more than one machine of the same type.

The process of designing the cells consisted of three stages:
1. static analysis to select the pool of 20 machines which would make up the cells;
2. design of manufacturing cells from the 20 machines;

### Table IV

<table>
<thead>
<tr>
<th>Design number</th>
<th>Machine sequence</th>
<th>Number of machines in each cell</th>
<th>Demand met</th>
<th>Unmet demand</th>
<th>Inter-cell transfers</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1</td>
<td>M1 M2 M3</td>
<td>3</td>
<td>97</td>
<td>50</td>
<td>0</td>
</tr>
<tr>
<td>D2</td>
<td>M1; M1 M2 M3</td>
<td>1; 3</td>
<td>102</td>
<td>45</td>
<td>5</td>
</tr>
<tr>
<td>D3</td>
<td>M1 M2 M3; M3</td>
<td>3; 1</td>
<td>142</td>
<td>5</td>
<td>45</td>
</tr>
<tr>
<td>D4</td>
<td>M1 M2 M3 M3</td>
<td>4</td>
<td>142</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>D5</td>
<td>M1 M1 M2 M3; M3</td>
<td>4; 1</td>
<td>147</td>
<td>0</td>
<td>45</td>
</tr>
<tr>
<td>D6</td>
<td>M1; M1 M1 M2 M3 M3</td>
<td>1; 4</td>
<td>145</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>D7</td>
<td>M1 M1 M2 M3 M3</td>
<td>5</td>
<td>147</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
3 dynamic simulation of the proposed new layout to identify any outstanding problems and test robustness to system change.

The first two stages were carried out using computer spreadsheets. The third stage used a specialised manufacturing simulation package, and will not be discussed in detail in this paper.

**Static analysis**

The first stage of the design process for the new facility had been to perform a static analysis of the products and their demand in order to find a combination of 20 machines from the original pool of over 60 machines that would meet the demand for the 200 products. This analysis used data of the form shown in Table I. For each product, the demand per unit time period was specified, together with the sequence of machines which the product needed. The static analysis consisted of counting how many presses of each type were used by each product, multiplying by the demand for each product and summing to calculate the total demand on each type of press. This calculation indicated which were the top 20 most busy machines and these were chosen as the pool from which the cells would be designed.

The proposed set of 20 machines was compared with the products that were to be made in the new facility. A few low demand products had to be rejected, because they required a press that was not needed by other products in the 200. These products could only have been made by including a press that would have had a comparatively low utilisation, at the expense of a higher utilisation press needed by higher demand products.

Another output of this stage of the analysis was the determination of the lot sizes and cycle length. The facility assumed an efficiency of 80 per cent, which meant that 80 per cent of working hours would be available for set-up and production.

A number of different batch sizing strategies were compared. These were based on:
- all products on the same cycle of one or two months;
- all products on the same cycle of five or more weeks;
- batch sizes calculated according to economic batch quantities, creating different cycle lengths for each product;
- batch sizes based on the MRP settings in use at that time in the organisation.
It was decided to simplify scheduling of the facility by fixing all the products to run at the same cycle length, i.e. with the same time interval between subsequent runs of the same job. This would enable each cell to run to a fixed sequence of jobs, making scheduling simpler and more predictable. Two cycle lengths were chosen for comparison, of about six and seven weeks.

Given the choice of cycle length, and data on demand and production rate for every product on each machine, it was then possible to calculate the time required on each machine type to produce all the required products during each cycle. To this could be added the set-up times for each product. This could be compared with the amount of time available on each machine type (assuming 80 per cent efficiency). This analysis identified which machines would be under the highest demand.

The next stage is to check which products can be completed on this selection of machines. A spreadsheet can be used here to sort the products according to demand. This can be combined with the data on which machines are used by each product to check that the machine types are available to complete the products, and to calculate the time required on each machine type for each product. The spreadsheet was also used during this stage to compare different fixed cycle lengths.

The data required at this stage of the analysis were:

- the level of operating efficiency;
- production rates on each machine and set-up times;
- machine types required by each product.

The outcome of the first stage of the analysis is:

- selection of 20 machines from which the cells will be chosen;
- rejection of a small number of products which would require under-utilised machines;
- choice of cycle length and batch sizes.

The next stage of the analysis is the design of the cells themselves.

**Design of the cells**

With the set of machines chosen, the next step was to decide how to group them into cells. The data consisted of the sequence of presses and total demand for each of the 200 products. Using a pivot table, the data were grouped into sequences. It was found that 58 different sequences of presses occurred. This is a considerable reduction on the number, \( N \), of possible ways of selecting between one and seven machines from a set of 20:

\[
N = \sum_{k=1}^{7} \frac{20!}{(20-k)!}
\]

The value of \( N \) in this case would be 420 million, which would be computationally infeasible.

The data were processed by Excel’s Pivot Table tool to identify all the active sequences. A total of 58 sequences were identified, from over 200 products. Part of the pivot table referring to products, which required one or two operations only, is shown in Table V.

The 20 machines had to be arranged as short cells with no more than five machines in each cell. The initial stage of each design was to generate cells or part cells to satisfy the very high demand sequences. Once these core presses had been arranged, the remaining presses were then accommodated around them. This was done so as to maximise the demand that could be processed without inter-cell transfer or skipping presses in a line. Two cell designs (Design A and Design B) were chosen for final comparison. These are shown in Figure 4.

These designs were compared according to the extent to which presses had to be skipped, or the products had to move from one cell to another. Table VI shows part of the assessment of the two designs. The column headings are explained below:

- 0 movements: sequence is satisfied exactly by plan;
- 1 movement: demand will have to skip presses in the line or swap cells once;
- 2 movements: demand will have to skip presses or swap cells twice;
- Not enough presses: not enough presses to meet demand without passing through a machine more than once.

It can be seen from Table VI that Design A and Design B are unable to satisfy the demand for the sequence 6-6-6-5-5. However, demand for this product at 5,540 is only 0.33 per cent of the total demand. Design A requires the sequences 2-6-7, 1-4-4-4 and 3-7-7-7-7 to transfer across two cells, but requires no inter-cell transfer for the sequences 3-6,
3-6-6 and 3-6-6-6. Furthermore, it includes the sequence 4-4-4-4, which was not possible under Design B. However, Design B meets more of the total demand than Design A, with more demand covered without inter-cell transfers, as may be seen on the bottom row of Table VI.

**Robustness to demand changes and dynamic analysis**

To complete the exercise, two more investigations were performed:
1. robustness to planned changes in demand, and
2. simulation for scheduling feasibility.

It was known that the demand for the products would change over the next few years, and it would be desirable not to have to make too many alterations to the cell layouts in response to these changes. Since these would be declining products, they would not justify the expense of further re-configuration of the factory layout.

The cell designs were compared for robustness to planned changes in demand over the next three years, in response to planned decline of products. This was done by changing the demand patterns and obtaining a new pivot table. The much reduced number of sequences was then compared with the planned demand patterns to calculate how much inter-cell transfer would be needed for the two designs in the future. It turned out that, given the future pattern of demand, there would be no need for drastic overhaul of the cells, with a reduced configuration based on the existing cells matching the needs.

A static analysis, as carried out so far, does not indicate how well the demand can be scheduled on the arrangement of presses. It was anticipated that the small size of the presses, with the small amount of press skipping and inter-cell transfer, would make scheduling straightforward. However, practical complicating factors include:
- the range of number of presses that the products use;
- the differing batch sizes;
- the presses do not all work at the same rate.

Using the Witness simulation package, simulations were set up of cell designs to test the demand levels for feasibility. The simulations had to cope with the possibility of products commencing processing part-way through a cell, rather than at the beginning. Inter-cell transfers also had to be accommodated, permitting products to leave a cell and possibly re-enter it. The complexity of programming these possibilities into the simulation reflects the complexity of scheduling these possibilities in real press shops.

The simulations were extended to cope with the following variabilities in processing:
- set-up times: ±10 per cent, ±20 per cent;
- press stamping rates: ±5 per cent, ±10 per cent, ±15 per cent;
- batch size changes: five-week period and 6.5-week period;
- change job order slightly.

Changing the press stamping rate by +5 per cent is the same as having a demand fluctuation of −5 per cent and vice versa.
The simulations provided a convincing demonstration to the collaborating organisation of the feasibility of the proposed designs and confirmed the importance of maintaining the expected press stamping rates.

**Summary and concluding remarks**

The paper reviews brief methods of cell-based design for manufacturing systems, with the observation that these methods may not be well suited to designing cells which meet the practical needs of industry. The considerations which are often overlooked in traditional methods include the fact that cells should be compact, that skipping machines in cells is not desirable (both from the logistical and machine utilisation points of view), and that inter-cell transfers make scheduling difficult and complex. If these aspects are not taken into considerations, they will counteract many of the supposed advantages of cell-based designs.

We propose a simple method of cell-based design, based on the demand of the products. A spreadsheet is used to identify all the active sequences of machines, with associated demand. The core sequences that satisfy significant demand are used as the basis around which to construct cells. Each time a machine is added to a cell, the new cell is assessed thoroughly for the demand that it satisfies, as well as the amount of inter-cell transfers and machine skipping that it involves. A thorough discussion of the algorithm is included to indicate the ways in which a designer may simplify or extend the algorithm. In some situations, it may be best to carry out the method by hand, so as to permit the high degree of customisation that may be required to meet a particular factory’s design problems.

The paper concludes with a case study example based on an actual design problem in a large UK manufacturer. A total of 200 products were manufactured, on 20 machines. A static analysis of demand was used to identify which 20 machines should be used for the cell design. A total of 58 active sequences of machines were identified using the Excel pivot table tool. These were used to generate two cell designs, which were compared for demand met and movements (skips and transfers). The comparison of the designs was completed by investigating future planned demand, and scheduling feasibility using simulation techniques.
This investigation shows the effectiveness and adaptability of simple spreadsheet-based methods to solve realistic problems in industry, without requiring high levels of programming skills, computational resources, or mathematical training. Rather, the multiple objectives and constraints of realistic problems can be addressed and adapted as necessary.

References