Manufacturing and Fractional Cell Formation Using the Modified Binary Digit Grouping Algorithm

Mroue Hassan and Dao Thiên-My
Department of Mechanical Engineering
École de Technologie Supérieure (University of Quebec)
Montreal, Quebec H3C 1K3, Canada

Abstract

A modified version of the new binary digit grouping algorithm is presented in order to search for the optimal solution to the manufacturing and fractional cell formation problem in the context of the design of cellular manufacturing systems. This algorithm leads to a configuration which groups the machines having common production characteristics as well as the parts having common processing requirements into part families according to the concept of group technology. As a final step, manufacturing cells can be formed each of which involves a set of machines and parts. The nonzero entries which remain outside the manufacturing cells are called exceptional elements. When a lot of exceptional elements are obtained, an additional cell called fractional (or remainder) cell can be formed. The aim of the remainder cell is to reduce the number of exceptional elements since it must contain all of the machines as well as the greatest possible number of these elements. The nonzero entries which are included within the manufacturing or fractional cells will be no longer considered as exceptional elements. This algorithm was tested through a MATLAB code by using illustrative examples from the literature and succeeded to give optimal final solutions.

Keywords
Binary digit grouping algorithm, Cell formation, Manufacturing cell, Fractional cell, Exceptional elements

1. Introduction
A new algorithm entitled "Binary Digit Grouping Algorithm" used for the formation of manufacturing cells according to the concept of group technology was presented in a previous conference (Mroue and Dao 2012). In this paper, a modified version of the same algorithm is provided in order to demonstrate its capability to form not only manufacturing cells, but also an additional fractional cell when applicable. The cellular manufacturing system (CMS) results from applying the concept of group technology (GT) (Asokan et al. 2001). This concept is an industrial philosophy which aims to group the machines having common production capabilities into manufacturing cells, as well as the parts having common geometric shapes or processing requirements into part families in order to benefit from these similarities (Xiaodan et al. 2007). The manufacturing cell formation problematic is considered as a non-polynomial hard (NP-hard) problem and it was classified for a long time as being the most challenging one because the processing time required to solve it increases exponentially with the size of the problem (Ben Mosbah and Dao 2010). On the other hand, the inter-cell movement occurs when a part is treated by the relevant machine outside the manufacturing cells . In such cases, the allocated elements will be called exceptional elements. In the problems where we find that a lot of exceptional elements a fractional cell (also called remainder cell) may be formed. The remainder cell must contain all of the machines of the system as well as the greatest possible number of exceptional elements. By this way, the movements of parts between the manufacturing cells on one hand, and the remainder cell on the other hand are not considered inter-cell movements (Chandrasekharan and Rajagopalan 1989). Numerous researchers worked on this problem and provided methodologies which succeeded to give optimal or near-optimal solutions. Mark et al. (2000) proposed an adaptive genetic algorithm in order to solve the cell formation problem. Liang and Zolfaghari (1999) provided a new neural network approach to solve the comprehensive grouping problem. Solimanpur et al. (2010) worked on this problem through an ant colony optimization (ACO) method. Lei and Wu (2006) approached the problem by presenting a tabu search method etc. Concerning the formation problematic of the additional fractional cell, the authors who addressed it are very few, Venkumar and Noorul Haq (2006) applied a modified ART1 neural networks algorithm in order to treat it whereas Chandrasekharan and Rajagopalan (1989) used a simulated annealing approach for the same purpose.

2. The Modified Binary Digit Grouping Algorithm

The first step of forming manufacturing cells consists of using a matrix which is called incidence matrix. The size of an incidence matrix is $M \times N$ where M represents the machines and N the parts. The matrix can be presented in the following form: A = [a_{mn}] where a_{mn} is the workload (production volume multiplied by the unit processing time) of the part number n when being processed on the machine number m (Mak et al. 2000). Let us take the following 5×5 incidence matrix as an example:

Table 1: A 5×5 incidence matrix

m \ n	1	2	3	4	5
1	1	0	1	0	0
2	0	0	0	1	0
3	0	0	1	0	0
4	0	0	0	1	1
5	0	1	0	0	0

A nonzero entry (i.e. a 1 digit) means that the relevant part will be processed by the concerned machine: if we take the nonzero entry in the upper left corner as an example, it means that the part number 1 will be processed by the machine number 1; whereas, a zero entry means the inverse. We can divide the elements of the incidence matrix into 3 categories:

- Elements in the corner of the matrix (the 4 elements highlighted in pink in the following matrix):

Table 2: The corner elements (highlighted in pink) of the incidence matrix

m \ n	1	2	3	4	5
1	1	0	1	0	0
2	0	0	0	1	0
3	0	0	1	0	0
4	0	0	0	1	1
5	0	1	0	0	0

- Elements in the borders (but not the corners) of the matrix (highlighted in bright green):

Table 3: The border elements (highlighted in bright green) of the incidence matrix

m \ n	1	2	3	4	5
1	1	0	1	0	0

2	0	0	0	1	0
3	0	0	1	0	0
4	0	0	0	1	1
5	0	1	0	0	0

- Finally, elements in the heart (i.e. not in the borders) of the matrix (highlighted in turquoise):

Table 4: The elements in the heart (highlighted in turquoise) of the incidence matrix

m \ n	1	2	3	4	5
1	1	0	1	0	0
2	0	0	0	1	0
3	0	0	1	0	0
4	0	0	0	1	1
5	0	1	0	0	0

If we begin with the corner elements and consider only (isolate) the nonzero entries as well as their surrounding elements (in this example, we have a single nonzero entry in the 4 corners which is located in the upper left one is shown in the following table):

Table 5: A nonzero entry in the upper left corner of the incidence matrix with its surrounding elements

m \ n	1	2	3	4	5
1	1	0	1	0	0
2	0	0	0	1	0
3	0	0	1	0	0
4	0	0	0	1	1
5	0	1	0	0	0

1	0
0	0

For each non zero entry, we calculate the nonzero entry neighboring factor (N_f) as being the sum of all the surrounding elements:

$$N_f = 0 + 0 + 0 = 0$$

Now let us consider the second type of the nonzero entries which are those located in the borders (but not corners) of the matrix and we take the nonzero entry located in the first row and third column as an example by isolating it together with its surrounding elements:

Table 6: A nonzero entry in the upper border of the incidence matrix with its surrounding elements

m \ n	1	2	3	4	5
1	1	0	1	0	0
2	0	0	0	1	0
3	0	0	1	0	0
4	0	0	0	1	1
5	0	1	0	0	0

0	1	0
0	0	1

We proceed as previously:

$$N_f = 0+0+0+1+0 = 1$$

Finally, let us consider the third and last type of nonzero entries which are those located in the heart of the matrix by taking one of them (the element located in the fourth row and fourth column in the next figure) as an example:

Table 7: A nonzero entry in the heart of the incidence matrix with its surrounding elements

m \ n	1	2	3	4	5
1	1	0	1	0	0
2	0	0	0	1	0
3	0	0	1	0	0
4	0	0	0	1	1
5	0	1	0	0	0

1	0	0
0	1	1
0	0	0

$$N_f = 1+0+0+0+0+1+0+0 = 2$$

After calculating the individual nonzero entry neighboring factor (N_f) of all the nonzero entries of the incidence matrix, we multiply them by 2 and sum up the resulting values all together in order to get the "nonzero entries neighborhood factor of the whole matrix" (N_M). As a next step, we swap randomly either 2 rows and/or 2 columns (swapping 2 rows as well as 2 columns once every few iterations helps to avoid getting trapped in local optima) in order to get a new configuration of the incidence matrix. The maximum number of different combinations (i.e. of permutations) which may result from applying this process on a certain matrix will be equal to the multiplication of the factorial of the number of rows by the factorial of that of the columns (in this example, we have $5! \times 5! = 14400$ different possible configurations of the incidence matrix). After each swapping process, we re-calculate everything from the beginning for the new configuration in order to get the new value of (N_M) called (N_M*) and compare it with the previous one: if the new one is greater than the previous one, the new matrix will be considered as the new solution for the problem; otherwise, the previous configuration will be kept. The computational process continues in this manner until testing all (or at least a great number) of the possible configurations and finally, the one with the greatest value of (N_M) will be selected as the final solution. In the case where we have big size problems (i.e. when we have numerous rows and/or columns in the incidence matrix), the computational time may become very long; that is why, it will be limited to a reasonable duration (such as 30 minutes) and the best found solution will be adopted as the final one (even in such cases, the best found solution will be mostly the best possible one because a lot of apparently different configurations will lead to the formation of the same cells). The binary digit grouping algorithm can be illustrated as follows:

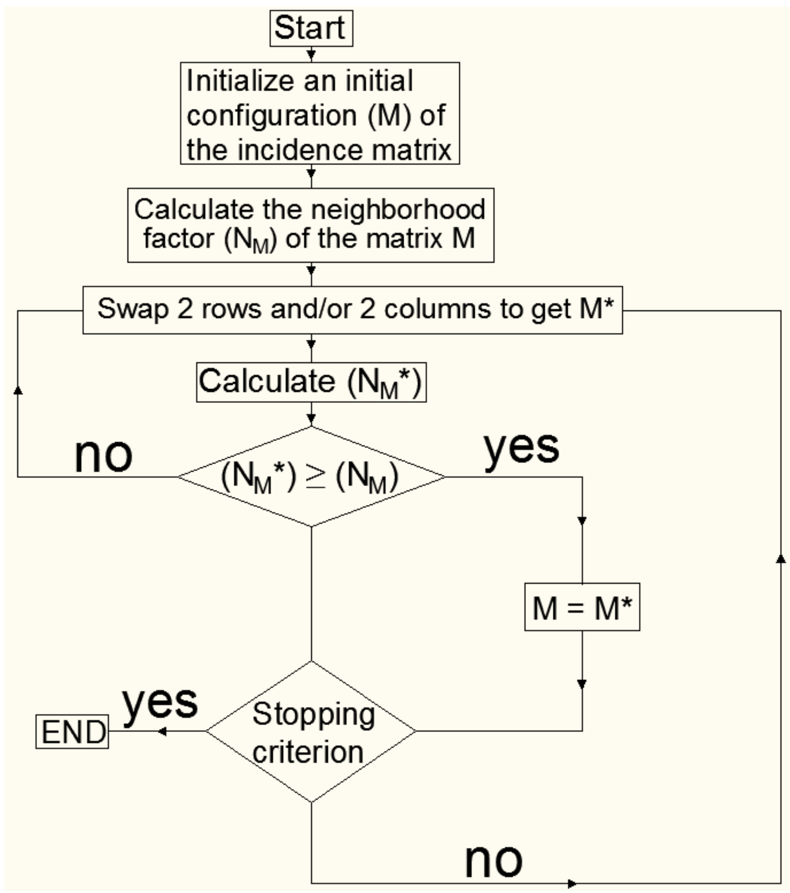

Figure 1: The binary digit grouping algorithm

As can be easily deduced, the binary digit grouping algorithm aims to group the nonzero entries as much as possible together within the incidence matrix in order to come up with manufacturing cells which have to include the greatest possible numbers of nonzero entries. The effectiveness of the resulting solution can be tested and compared to those resulting from the application of other algorithms using some formulas which were used for the evaluation of other algorithms such as the genetic algorithm (Mark et al. 2000). If we let:

- K be the number of manufacturing cells which will be formed within the incidence matrix
- n_1 represents the number of nonzero entries existing in the manufacturing cells
- M_k and N_k where $k = (1,2,...,K)$ be consecutively the numbers of machines and parts which are assigned to the manufacturing cell number k; we can get:

$$e_1 = \frac{n_1}{\sum_{k=1}^{K} M_k N_k} \qquad (1)$$

Where (e_1) is a measure of the cell density which means that the more the machines and parts in a manufacturing cell are similar, the greatest is the value of (e_1) and vice-versa. On the other hand, if we let (n_2) be the number of the exceptional elements (which are the nonzero entries that are not located within the manufacturing cells), we can get:

$$e_2 = 1 - \frac{n_1}{n_1 + n_2} \qquad (2)$$

Where (e_2) is a measure of the intercellular material flow and it increases with the increase in the number of exceptional elements and vice-versa. Finally, the grouping efficiency (e) can be calculated as follows:

$$e = e_1 - e_2 \qquad (3)$$

The numerical value of (e) always belongs to the interval [-1, 1].

3. Cell formation

After applying the algorithm, a final configuration of the incidence matrix will be gotten. The nonzero entries are expected to be grouped as close to each other as possible. The next step consists of creating only manufacturing cells by keeping in mind three rules:

- The greatest possible number of nonzero entries must be contained in the formed cells.
- Every machine and/or part must be involved in a cell.
- A machine and/or a part cannot be assigned to more than one cell.

According to the distribution of the nonzero entries in the matrix configuration obtained and in cases where we obtain a lot of exceptional elements, a re-distribution of the cells may take place according to what follows:

- Creating manufacturing cells where each contains a certain number of machines and parts.
- Not all of the machines but all of the parts must be assigned to the manufacturing cells.
- A machine and/or a part cannot be assigned to more than one manufacturing cell.
- Creating one additional fractional cell which contains all of the parts in addition to only the machines which are not assigned to manufacturing cells.

The nonzero entries which are included in the manufacturing cells or the fractional one are not considered as exceptional elements; that is why, the addition of the fractional cell may play a major role in reducing the number of these elements. The following section gives further illustrations through two illustrative examples.

4. Illustrative Examples

We are going in what follows to test this algorithm through two illustrative examples taken from the literature. The first one results only in the formation of manufacturing cells and the second one shows the formation of manufacturing cells as well as an additional fractional cell. Consider the following incidence matrix is taken from SrinlVasan et al. (1990).

Table 8: The incidence matrix for the first illustrative example

m \ n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	1	1	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0
3	0	1	1	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0
4	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	1	0	1
6	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	1	0	0	1	0	1	1	0	0	0	1	0	0	1	0
8	0	0	0	0	0	1	0	0	1	0	1	1	0	0	0	1	0	0	1	0
9	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	1	0	1
10	0	0	0	0	0	1	0	0	1	0	1	1	0	0	0	1	0	0	1	0

After running the MATLAB code of the binary digit grouping algorithm, the following solution was gotten:

Table 9: The cell design for the first illustrative example

m \ n	19	9	11	16	12	6	7	4	1	18	14	17	15	13	20	10	2	5	8	3
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
8	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0
6	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0

In this example, 4 manufacturing cells were created; the number of nonzero entries existing in the manufacturing cells (n_1) is equal to 49, the number of exceptional elements (n_2) is equal to 0 which means that there is no intercellular material flow. The calculations led to a value of the cell density measure (e_1) equal to 1 and to a value of 0 for the intercellular material flow (e_2). The value of the grouping efficiency: $e = e_1 - e_2 = 1$. There is no need for a fractional cell in this example since there are no resulting exceptional elements.

Let us consider now the second illustrative example which is represented by the following matrix which is taken from King and Nakornchai (1982).

Table 10: The incidence matrix for the second illustrative example

m \ n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43
1																																					1						1
2		1								1																		1				1					1	1		1		1	
3							1										1																	1	1	1							
4					1				1					1					1		1		1						1														
5					1			1	1					1	1	1			1		1		1						1				1								1		1
6	1	1				1	1	1				1	1	1			1		1				1									1	1	1			1		1	1		1	1
7	1												1												1																		
8	1	1	1					1	1		1	1			1				1	1	1		1	1			1	1			1						1	1			1		1
9		1		1						1								1										1				1					1	1		1		1	
10	1											1	1												1	1					1								1				
11			1						1											1				1			1			1													
12											1											1		1			1			1													
13			1																					1																			
14		1				1											1																		1								
15					1									1					1		1												1								1		1
16		1					1			1								1														1					1	1				1	

After applying the modified binary digit grouping algorithm, the following solution which includes only manufacturing cells was obtained, with a value of (N_m) equal to 650:

- Manufacturing cell 1:
  - Machines 10, 7
  - Parts 26, 25, 13, 1, 12, 39, 31
- Manufacturing cell 2:
  - Machines 15, 2, 9, 16, 14, 3, 12, 11, 13
  - Parts 3, 24, 27, 30, 22, 36, 35, 17, 7, 40, 42, 2, 37, 32, 38, 10, 18, 28, 4, 16, 20, 11, 34, 6
- Manufacturing cell 3:
  - Machines 1,4,5,8,6
  - Parts 15, 8, 23, 43, 19, 21, 14, 5, 29, 41, 33, 9,
- Number of exceptional elements:
  - 34

Table 11: The design of the manufacturing cells for the second illustrative example

m\n	26	25	13	1	12	39	31	15	8	23	43	19	21	14	5	29	41	33	9	3	24	27	30	22	36	35	17	7	40	42	2	37	32	38	10	18	28	4	16	20	11	34	6
10	1	1	1	1	1	1	1																																				
7		1	1	1																																							
15											1	1	1	1	1		1	1																									
2																													1	1	1	1	1	1	1		1						
9																													1	1	1	1	1	1	1	1	1	1					
16																												1		1	1	1	1	1	1	1							
14																										1	1				1												1
3																									1	1	1	1														1	
12																					1	1	1	1																	1		
11																			1	1	1	1	1																	1			
13																				1	1																						
1											1																					1											
4										1		1	1	1	1	1			1																								
5								1	1	1	1	1	1	1	1	1	1	1	1																				1				
8				1	1		1	1	1	1	1	1	1				1		1	1	1	1									1	1		1			1			1	1		
6			1	1	1	1			1	1	1	1		1				1									1	1	1	1	1	1	1									1	1

As a result, a great number of exceptional elements were obtained (which is not good). In such cases, the problem may be solved by constructing manufacturing cells as well as one additional fractional (remainder) cell. The nonzero entries will be distributed between the manufacturing cells and a remainder cell in order to get a number of exceptional elements as low as possible. The reformation of the cells is illustrated as follows:

- Manufacturing cell 1:
  - Machines 10, 7
  - Parts 26,, 25, 13, 1, 12, 39, 31
- Manufacturing cell 2:
  - Machines 15
  - Parts 15, 8, 23, 43, 19, 21, 14, 5, 29, 41, 33
- Manufacturing cell 3:
  - Machines 2, 9, 16, 14, 3, 12, 11, 13
  - Parts 9, 3, 24, 27, 30, 22, 36, 35, 17, 7, 40, 42, 2, 37, 32, 38, 10, 18, 28, 4, 16, 20, 11, 34, 6
- Remainder (fractional) cell:
  - Machines 1, 4, 5, 8, 6
- Number of exceptional elements:
  - 0

Table 12: The design of the manufacturing cells in addition to one fractional cell for the second illustrative example

m\n	26	25	13	1	12	39	31	15	8	23	43	19	21	14	5	29	41	33	9	3	24	27	30	22	36	35	17	7	40	42	2	37	32	38	10	18	28	4	16	20	11	34	6
10	1	1	1	1	1	1	1																																				
7		1	1	1																																							
15											1	1	1	1	1		1	1																									
2																													1	1	1	1	1	1	1		1						
9																													1	1	1	1	1	1	1	1	1	1					
16																												1		1	1	1	1	1	1	1							
14																										1	1				1												1
3																									1	1	1	1														1	
12																					1	1	1	1																	1		
11																			1	1	1	1	1																	1			
13																				1	1																						
1											1																					1											
4										1		1	1	1	1	1			1																								
5								1	1	1	1	1	1	1	1	1	1	1	1																				1				
8				1	1		1	1	1	1	1	1	1				1		1	1	1	1									1	1		1			1			1	1		
6			1	1	1	1			1	1	1	1		1				1									1	1	1	1	1	1	1									1	1

The solution demonstrates that the problem of exceptional elements problem may be completely solved by implementing a remainder cell. In addition, it demonstrates that the binary digit grouping algorithm may be able not only to solve the manufacturing cell formation problems, but also the fractional (remainder) cell formation problematic as well.

4. Conclusion

A modified version of the binary digit grouping algorithm is presented and explained in order to solve the machine-part grouping problem for manufacturing and fractional cell-formation within the flexible manufacturing systems. The simple steps and computational procedures make it particularly powerful in quickly creating and conserving the matrix configurations which sequentially get closer and closer to the optimal solutions for both small and big size problems. The advantages of using such an algorithm have been demonstrated by using it to solve two illustrative examples.

References

Mark, K. L., Y. S. Wong, et al. (2000). "An Adaptive Genetic Algorithm for Manufacturing Cell Formation." *The International Journal of Advanced Manufacturing Technology* 16(7): 491-497.

SrinlVasan, G., T. T. Narendran, et al. (1990). "An assignment model for the part-families problem in group technology." *International Journal of Production Research* 28(1): 145-152.

King, J. R. and V. Nakornchai (1982). "Machine-component group formation in group technology: review and extension." *International Journal of Production Research* 20(2): 117-133.

Mroue, H. and T.-M. Dao (2012). "Optimization of manufacturing cell formation with a new binary digit grouping algorithm." *2012 ICMSE International Conference on Manufacturing Science and Engineering.*

Venkumar, P., & Noorul Haq, A. (2006). Fractional cell formation in group technology using modified ART1 neural networks. *The International Journal of Advanced Manufacturing Technology, 28*(7-8), 761-765. doi: 10.1007/s00170-004-2421-z

Asokan, P., G. Prabhakaran, et al. (2001). "Machine-Cell Grouping in Cellular Manufacturing Systems Using Non traditional Optimization Techniques - A Comparative Study." *The International Journal of Advanced Manufacturing Technology,* **18**(2): 140-147.

Xiaodan, W., Chao-Hsien, C., Yunfeng, W., Weili, Y. (2007). "A genetic algorithm for cellular manufacturing design and layout." *European Journal of Operational Research*, Vol. 181, pp.156-167.

Ben Mosbah, A. and T.-M. Dao (2010). "Optimization of group scheduling using simulation with the meta-heuristic Extended Great Deluge (EGD) approach." *Industrial Engineering and Engineering Management (IEEM), 2010 IEEE International Conference*.

Liang, M. and S. Zolfaghari (1999). "Machine cell formation considering processing times and machine capacities: An ortho-synapse Hopfield neural network approach." *Journal of Intelligent Manufacturing* **10**(5): 437-447.

Solimanpur, M., S. Saeedi, et al. (2010). "Solving cell formation problem in cellular manufacturing using ant-colony-based optimization." *The International Journal of Advanced Manufacturing Technology* **50**(9): 1135-1144.

Lei, D. and Z. Wu (2006). "Tabu search for multiple-criteria manufacturing cell design." *The International Journal of Advanced Manufacturing Technology* **28**(9): 950-956.

Chandrasekharan, M. P. and R. Rajagopalan (1989). "GROUPABIL1TY: an analysis of the properties of binary data matrices for group technology." *International Journal of Production Research* **27**(6): 1035-1052.

Biography

Mroue Hassan is a PhD in engineering student in the Department of Mechanical Engineering at the École de Technologie Supérieure (University of Quebec), Montreal, Canada. He earned B.S. in Mechanical Engineering from Lebanese American University, Lebanon, Masters in International Automotive Engineering from University of Applied Sciences-Ingolstdat, Germany. He has taught courses in engineering drawing and CAD in addition to practical works in mechanical engineering. His research interests include mechanical design, manufacturing, simulation, optimization, and lean.

DAO Thiên-My is a Professor in the Department of Mechanical Engineering. He received his Bachelor's, Master's and PhD degrees in Mechanical Engineering (Design option) from Sherbrooke University, Sherbrooke, Canada. After spending four years in the industry, he joined the Department of Mechanical Engineering of École de Technologie Supérieure (University of Quebec), Montreal, Canada, where he is teaching particularly production and operations management, quality management and design of manufacturing systems courses. His research interests include optimization of the manufacturing systems design, reliability and management of manufacturing systems.