Journal of Industrial and Systems Engineering
Vol. 5, No. 3, pp142-153
Fall 2011

Part-level Sequence Dependent Setup Time Reduction in CMS

Shahram Sharifi[1]*, Stayaveer.S.Chauhan[2], Nadia Bhuiyan[3]

[1,3]Department of Mechanical and Industrial Engineering, Concordia University, Quebec, Canada
[1]ssharifi@encs.concordia.ca, [3]bhuiyan@alcor.concordia.ca

[2]John Molson School of Business, Concordia University, Quebec, Canada
sschauha@alcor.concordia.ca

ABSTRACT

This paper presents the idea of creating cells while reducing part-level sequence-dependent setup time in general cellular manufacturing systems (CMS). Setup time reduction in CMS has gained modest attention in the literature. This could be attributed to the fact that the fundamental problem in cell formation in CMS has been mainly related to material handling and machine utilization while setup time was assumed to implicitly decrease as a result of grouping similar parts in a manufacturing cell. Despite more than three decades of CMS's history, it has been relatively recent that setup time has been included in cell formation problems and found a place in the existing models. However, sequence-dependent setup time in the literature has been dealt with mostly within the context of scheduling "part-families" in a single manufacturing cell or in the allocation of parts to flow line cells. The present model includes the three fundamental elements of a cell formation procedure: machine utilization, intercellular movement and setup time. This therefore provides a basic structure that would serve as a general sub-model for real manufacturing cell formation problems including any type of setup time and manufacturing cell. Due to computation time and complexity of the problem, a solution approach based on theGenetic Algorithm based (GA-based)heuristic has been discussed and the solution of a sample problem has been compared with that of conventional optimization software. The results indicate a reasonably satisfactory performance by the GA-based heuristic in terms of accuracy and computation time.

***Keywords*:**Sequence-dependent, Setup time, Part-level, Cellular manufacturing system(CMS), Genetic Algorithm.

1.INTRODUCTION

Cellular manufacturing, which is an application of group technology (GT), has been recognized as one of the most recent technological innovations in job-shop or batch-type production to gain economic advantages similar to those of mass production (Chang and Lee, 2000; Snead, 1989; Chu and Tsai, 1990; Singh and Rajamani, 1995; Mahdavi and Mahadevan, 2008). The conventional cellular manufacturing system (CMS) is based on optimizing the trade-off between material handling (intercellular movement) and machine utilization as the major criteria for all cell formation

*Corresponding Author

ISSN: 1735-8272,

(CF) procedures. The main tool, if not the sole source of information, used to determine CF procedures in the traditional CMS, is the machine-part incidence matrix or machine-component incidence matrix (MCIM) (Molleman*et al.*, 2002; Ohta and Nakumara, 2002). The CF procedures in conventional CMS assume that setup time tends to decrease due to the grouping of similar component parts of different products (Vilas and Vandael, 2002).

Our research seeks solutions mainly for closed job shops where a fixed number of products are produced on a repetitive basis. Repetitive manufacturing is one solution for situations where market demand warrants large-scale production but demand rate and marketing constraints prohibit continuous production (Rajamani*et. al*., 1992).

In certain cases, with frequent changeovers of the machines in repetitive cycles within the planning horizon, when significant setup times and costs are involved, the overall setup cost incurred could be quite considerable, if it is not incorporated in the cost minimization process. Therefore in such situations, the cell formation problem should consider minimizing the sequence-dependent setup times in order to minimize the production cost.

In this paper, we first argue that the reduction of setup time in CMS is not fully ensured unless it is explicitly addressed: this justifies the necessity of incorporating setup time in CF procedure. Second, we elaborate on the concept of setup time as changeover and conclude that sequence-dependent setup time is the general and most inclusive of all types of setup times in real manufacturing: this ensures the generality of application of the proposed model. Third, we show that the most accurate setup time reduction would be realized when it is considered between pairs of component parts: this supports the increased accuracy of the model.

The paper is organized as follows: Section 2 provides a critical review of setup time in CF problems where the adequacy of conventional CF approaches based on the MCIM in addressing setup time reduction is examined. Section 3 introduces the problem and the formulation. Solution approach has been discussed in Section 4. An illustrative example has been provided in Section 5, while the computational performance of the GA (Genetic Algorithm)based heuristic has been examined in Section 6. Finally, Section 7 concludes the paper and discusses future work.

2. A REVIEW OF SETUP TIME IN CF PROBLEMS

Almost all CF approaches in CMS deal with some manipulation of the machine-component incidence matrix (MCIM) (Molleman*et al.* 2002; Ohta and Nakamura, 2002). In essence, the basic information required to solve a CM problem is based on MCIM (Ohta and Nakamura, 2002). It is a matrix $\boldsymbol{A_{ij}}$, with machines in rows and component parts in columns consisting of elements $a_{ij} = 1$ where part j visits machine i and $a_{ij} = 0$otherwise. Given what machines are required by what parts, intercellular movements and machine utilization can be optimized through manipulating the MCIM matrix with the maximum 1's inside the cells and minimum outside the cells. Thus, the overall material handling costs and machine investment cost can be optimized through a proper cell formation procedure (Moon and Gen, 1999).

The MCIM does not say anything about the setup times, thus is not able to control the setup time reduction. Some mixed results from some surveys can exemplify the uncertainty of setup time reduction in conventional CF approaches based on the MCIM. For example, Wemmerlov and Hyer (1989) stated that a well-organized job shop would result in better flow time and less work-in-process inventories than cellular manufacturing. They claim that surveys of firms adopting cellular manufacturing report better queue related results mainly because they are comparing their new layout to their previous poorly designed and operated job shop. Some literature raises doubts as to whether CMS produces better performance for queue related criteria (Al-Mubarak *et al.*, 2002;Kannan and Ghosh, 1995; Morris and Tersine, 1989)

In this paper, it is presumed that part of the ambiguity around setup time reduction in the CF procedure can be attributed to the definition of setup time in the literature. Some works that deal with introducing parameters or coefficients concerning setup cost or setup time, are implicitly based on consecutive operations of parts on various machines (as an example, see Damodaran*et al.* 1992;Ohta and Nakamura 2002). In some others, e.g. Atmani*et al.* (1995) and Lashkari*et al.* (2004), setup cost has been associated solely with one part and the machine. However Cox *et al.* (1995*),* defines setup as the work required to change a specific machine, work centre, or line from making the last good piece of a unit A to the first good piece of unit B (LaScola Needy *et. al.*, 1998). The above definition is the closest to real life manufacturing. The significance of this definition is that it implies the important concept of *changeover* between two parts. Besides, common sense and real practice both imply that changeover must take place on a *common* machine. The term 'common' should be stressed since it is a key factor in the definition that might lead to confusion if neglected or understated. In fact, setup time depends on three elements without which part of the information for obtaining the changeover time would be missing. These elements are: incoming component part *i*, outgoing component part *i'* and the common machine *k,* as shown in Figure 1.

Figure1 Setup as joint changeover

Therefore setup time, whether or not sequence-dependent, would be the joint changeover time between two interchanging parts on the common machine.

In order to maximize the inclusiveness and generality, a sequence-dependency condition has to be considered, so that:

$$S_{ii'k} = 0 \text{ for } i = i' \quad (1)$$

$$S_{ii'k} \neq S_{i'ik} \text{ for } i \neq i' \quad (2)$$

where $S_{ii'k}$ is the sequence-dependent setup time between parts *i* and *i'* on the common machine *k.*

The definition of setup as sequence-dependent changeover is not only clear and straightforward, but also is the most inclusive one in the real world which includes sequence-independent ones, when the input data for $S_{ii'k}$ and $S_{i'ik}$ are equal; or additive type where the sum of the times of removal of part *i'* and installing of part *i*, would represent the single joint changeover time $S_{ii'k}$.In effect it covers all different cases of changeover, for example, from uninstalling a previous die, say on a press, and installing a new die, to switching from a previous set of adjustments, say on a lathe machine, to a new one, and so on.

Consideration of all pairs of parts on different machines leads to a joint changeover matrix, $S_{ii'k}$ including the sequence-dependent setup times between pairs of parts having common machines. This information in companies is usually obtained from route sheets where the MCIM data is also obtained. It is obvious for a sequence-independent setup time that the corresponding input data for $S_{ii'k}$ and $S_{i'ik}$ will simply be equal. Whenever a pair of component parts have a common machine but will definitely not change over due to routing or technological reasons, the joint setup time in the matrix would be infinity.

Regarding the above discussion and conclusions, in Damodaran *et al.* (1992) and Ohta and Nakumara (2004), setup has been considered between consecutive operations k and k+1 of part j notwithstanding the common machine. In other words in their consideration of setup, the common machine has been tacitly ignored, i.e. the part is common and the machines could be different. In Atmani *et al.* (1995) and Lashkari *et al.* (2004), however, the outgoing part has been ignored.

Rajamani *et al*. (1992) considered a sequence-dependent setup time between pairs of parts notwithstanding the machine in flow shop cells where various parts or products tend to use identical production processes (Rajamani *et al*., 1992). In essence, formation of cells in pure flow shops may not be considered a routine CF problem in the first place, since it is known *a priori* that all machines will appear in all cells. This also implies that MCIM is not required, as all elements of such incidence matrix would consist of 1's only. Consequently intercellular movement does not play a role either. Therefore this paper fills the gaps in the literature regarding setup time in CMS by suggesting a solid and inclusive concept of setup time in general CMS.

3. RESEARCH PROBLEM AND MODELING

3.1. Description of Research Problem

In this problem we assume that a cellular manufacturing system processing component parts. Different component parts are to be processed on different machines in cells. A set of machines on which a part is processed can be input by the user through MCIM. The setup time is measured between every pair of different component parts and is generally sequence-dependent. The setup time between identical parts is considered zero. The setup time between different parts can be collected from the route sheets where other critical manufacturing information is recorded.

The objective is to design a cellular manufacturing system that simultaneously groups the machines and the component parts into cells so as to minimize the overall production cost of setup time and, machine and material handling (intercellular movement).

3.2. Integer programming model

We formulate the research problem as an integer programming model as follows.

Indices:

- i Part number, $i = 1, \ldots, I$
- p Ordinal position number of the part on the machine in the cell, $p = 1, \ldots, P$
- k Machine type number, $k = 1, \ldots, K$
- l Cell number, $l = 1, \ldots, L$

Variables:

$$x_{ipkl} = \begin{cases} 1 & \text{if part } i \text{ visits machine } k \text{ in position } p \text{ in cell } l \\ 0 & \text{otherwise} \end{cases}$$

$$z_{il} = \begin{cases} 1 & \text{if part } i \text{ visits cell } l \\ 0 & \text{otherwise} \end{cases}$$

y_{kl} Number of machine k in cell l

$\lambda_{ipi'(p+1)kl}$ A binary auxiliary variable assuming the value of 1, IFF both x_{ipkl} and $x_{i(p+1)kl}$ *equal 1*

Parameters:

$S_{ii'k}$ joint sequence-dependent setup time for replacing part i with part i' on machine k

C_k unit cost of setup on machine k

E_k cost of having one unit if machine k in the system

D_i cost of moving one unit of part i between cells

MAX_l maximum number of machines allowed in a cell

MIN_l minimum number of machines allowed in a cell

NOM_k available machine type k in the system

Given the above variables and parameters, if the objective is to minimize the overall setup cost, machine depreciation cost and material handling (intercellular movement) cost, then it could be expressed by the following function:

$$\sum_i \sum_j \sum_k \sum_l x_{ijkl} \cdot x_{i'(j+1)kl} \cdot S_{ii'k} \cdot C_k + \sum_{\text{k}} E_k \sum_{\text{l}} y_{kl} + \sum_i D_i \cdot (\sum_l z_{il} - 1) \quad i \neq i' \qquad (3)$$

Since the above function in (3) includes binary integers, it can be transformed to the following linear function by adding binary variable$\lambda_{ipi'(p+1)kl}$ and adding constraint (5) as follows:

$$\sum_i \sum_p \sum_k \sum_l \lambda_{ipi'(p+l)kl} \cdot S_{ii'k} \cdot C_k + \sum_k E_k \sum_{1} y_{kl} + \sum_i D_i \cdot (\sum_l z_{il} - 1) \qquad i \neq i' \qquad (4)$$

$$\lambda_{ipi'(p+1)kl} \geq x_{ipkl} + x_{i'(p+1)kl} - 1 \qquad (5)$$

Since we are dealing with sequence-dependent setup time, the sequence of changing over batches of parts on a machine affects the total setup time of the cellular configuration. Therefore index p specifies the position of parts in a sequence on each machine. Constraint (6) ensures that in each cell, on each machine, two or more parts cannot take over the same position in the processing sequence of their corresponding machine.

On the other hand, it is assumed that each part occupies one and only one position in the processing sequence of its corresponding machine in only one cell. This is restricted by constraint (7). Since the model seeks a minimum sum of setup times, it tends to break the link between pairs of interchanging parts and set them position-wise apart so as to further minimize the joint setup times. Constraint (8) is in place so as to ensure that we will not have such position voids. We should also relate machines and their corresponding visiting parts. Constraint (9) binds these two. Constraints (10) and (11) ensure that when a part does not visit a cell it will not be assigned to any position on any machine in that cell and vice versa. Finally the last constraints (12) and (13) reflect limitations on minimum and maximum number of machine types in cells.

The linear integer programming model is as follows:

$$\text{Minimize} \sum_i \sum_p \sum_k \sum_l \lambda_{ipi'(p+1)kl} S_{ii'k} C_k + \sum_k E_k \sum_l y_{kl} + \sum_i D_i \cdot (\sum_l z_{il} - 1) \qquad i \neq i' \tag{4}$$

Subject to

$$\lambda_{ipi'(p+1)kl} \geq x_{ipkl} + x_{i'(p+1)kl} - 1 \tag{5}$$

$$\sum_{i=1}^{I} x_{ipkl} \leq 1 \tag{6}$$

$$\sum_{p=1}^{P} \sum_{l=1}^{L} x_{ipkl} = 1 \tag{7}$$

$$\sum_{i=1}^{I} x_{ipkl} \geq \sum_{i=1}^{I} x_{i(p+1)kl} \tag{8}$$

$$x_{ipkl} \leq y_{kl} \tag{9}$$

$$\sum_{k=1}^{K} \sum_{p=1}^{P} x_{ipkl} \leq M^{\infty} \cdot z_{il} \tag{10}$$

$$\sum_{k=1}^{K} \sum_{p=1}^{P} x_{ipkl} \geq z_{il} \tag{11}$$

$$MIN_l \leq \sum_{l=1}^{L} y_{kl} \leq MAX_l \tag{12}$$

$$\sum_{l=1}^{L} y_{kl} \leq NOM_k \tag{13}$$

$$x_{ipkl}, y_{kl}, z_{il}, \lambda_{ipi'(p+1)kl} \quad \text{0,1 variables} \tag{14}$$

4. SOLUTION APPROACH

Cell formation problem modeled as a generalized assignment problem, was shown to be NP-hard (Shtubta[a], 1989), therefore exact approaches such as branch and bound algorithm would not be computationally efficient. In a BIP problem as in our model, exponential number of iterations, 2^n would be required in the worst case. Due to the NP-hard nature of the problem, heuristic approaches shall be adopted to solve the problem. A variety of researchers have applied GA to CF in CMS (See for example Wu[a]*et al.,* 2007; De Lit *et al.* 2000;Mahdavi and Mahadevan, 2008;Wu[b]*et al.* 2007;Defersha and Chen, 2008;Goncalves and Resende, 2002;Safaei and Tavakkoli-Moghaddam, 2009;Tavakkoli-Moghaddam*et al*., 2005). Murugan, and Selladurai (2005) applied genetic algorithm to a cell formation problem that would reduce the setup time, however they used group efficacy criteria approach rather than mathematical programming, and the setup time, while not being sequence-dependent, was measured on two different machines as opposed to a common machine.

Gosh *et al.* (2011) conducted a state-of-the- art generic review of the application of various metaheuristics in cellular manufacturing. Ahmed and Tavakkoli-Moghaddam (2004) compared various heuristic methods in solving cellular manufacturing problems in a dynamic environment. However to the best of our knowledge, due to the novelty of the current problem, application of GA to this problem is consequently novel too.

Our experience with commercial software, even for smaller instances, is not very encouraging either and that motivates us towards heuristic approaches. Therefore we have we have proposed a GA-based heuristic in this work. The chromosomal structure of the GA-based heuristic is depicted in Figure 2. Basically in this encoding, machine, encompasses the cell as opposed to the physical reality. This specific structure has been tailored to facilitate handling of the constraints during the recombination process. The 0-1 decision variabley_{kl} indicates whether cell *l* is contained by machine *k*. When cell *l* is contained within machine *k*, then the binary decision variablex_{ipkl} indicates whether part *i* would lie in that contained cell in position *p*. The aforementioned decision variables are read directly from the chromosomes. Other variables such as$z_{il}$are indirectly decoded.

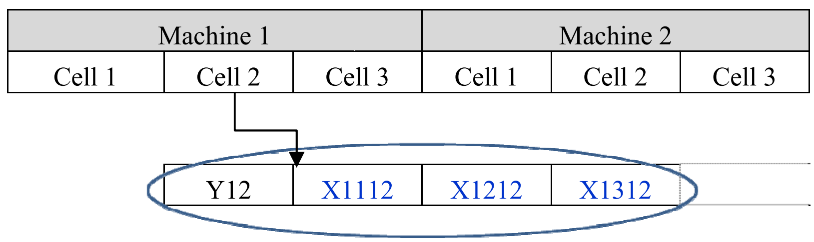

Figure 2 Chromosomal structure of the GA- based heuristic

For illustration, we need to depict the main stages of the GA heuristic in a context level flow chart, as shown in Figure 3, before we proceed we elaborate on each stage.

The various stages of the GA-based heuristic are explained below.

Initial Solution: The initial population is generated as follows: first, it is determined pseudo randomly whether a certain machine will go to a certain cell. If that is the case then it will be determined, pseudo randomly, whether a certain part would be processed in a certain position (sequence) on that machine in that cell. The above process continues until all parts are assigned to their corresponding machines in the cells. Constraint handling control mechanisms ensure that only feasible chromosomes will be generated.

Selection: The selection process is done through a *biased roulette wheel,* where for each chromosome, a frequency is calculated. This frequency is the reciprocal of the percentage of the objective value of the chromosome with respect to the total objective value of all chromosomes. The pseudo random numbers generated by the C++ random function within the range of minimum and maximum of the cumulative frequencies would select the parent in each generation, with replacement, based on which cumulative frequency band-width in which it would lie. Since the frequency bandwidths are proportional to the fitness of different chromosomes, those with higher frequency (fitness) have a higher chance to be hit by the pseudo random numbers thus providing the necessary discriminatory selection of the chromosomes inspired by the principle of natural selection.

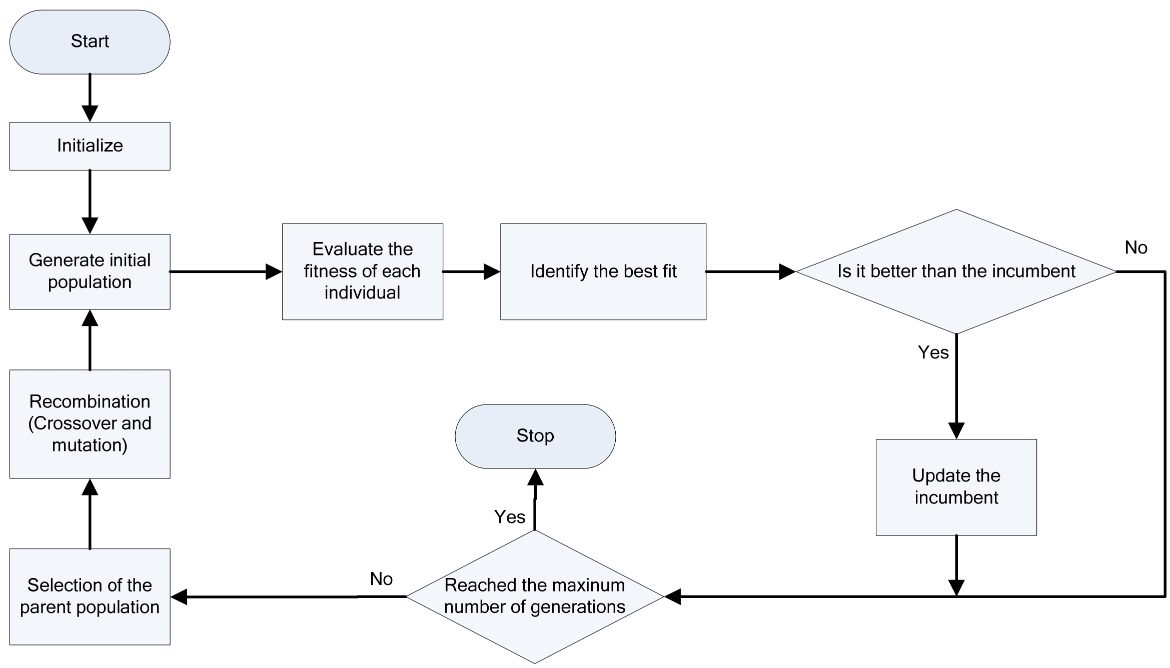

Figure 3 Flowchart of the GA heuristic (context level)

Recombination/Crossover operators: two crossover operators, namely, machine-level and part-level have been designed to recombine the selected parent chromosomes in the mating pool.

Mutation**:** After each mating, a *mutation* process is applied only to a small percentage of the offspring so as to avoid overshooting of the search area while preventing premature local optimization either.

Rejuvenation**:** The population is partially rejuvenated by randomly picking several previous generations and bringing in highest fitness chromosome in each of those generations to mix up with the current population, once the best objective value repeats in a certain consecutive number of generations. Besides, after every 100 generations a full rejuvenation occurs in order to maximize the exploration of the solution space.

5. ILLUSTRATIVE EXAMPLE

For illustration purposes, an example problem is considered as follows: Ten parts are grouped with four different machine types in three different cells in order to minimize the overall cost of setup, machine and intercellular movement. For simplicity, we consider that the cost coefficients of setup time, machine and intercellular movement are all unity. The following example is as an illustration of MCIM, its corresponding joint changeover matrix and the final solution for the following example problem.

Table 1 shows the MCIM. A sample of the corresponding joint changeover matrix of the above mentioned case for machine 1 is shown in Table 2. The setup times are generated pseudo randomly within the range of 10 and 40 minutes.

Table 3 shows the solution result using the GA-based heuristic. In each cell, the parts are shown in the sequence of their processing from left to right.

Table 1Machine-part incidence matrix

	Component Part									
Machine	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
M1	1	0	1	0	1	0	1	1	1	0
M2	1	1	1	1	1	1	1	0	1	0
M3	1	0	0	1	0	0	1	0	0	1
M4	1	1	0	0	1	0	0	0	0	0

Table 2Joint changeover matrix (Machine 1)

Parts	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1			10		19		27	13	39	
2										
3	22				26		10	21	16	
4										
5	15		27				25	14	21	
6										
7	32		30		20			31	14	
8	26		20		37		21		17	
9	37		17		27		13	13		
10										

Table 3 Sequence-dependent solution by GA-based heuristic

C1	C2	C3
P1-P3-P7	P5	P8-P9
P1-P3-P2	P6-P4	P5-P7-P9
P4-P1	P10	P7
P2	P1	P5

6. COMPUTATIONAL PERFOMANCE

In order to test the functionality and evaluate the computational performance of the GA based heuristic, 6 problem sizes each with 6 data sets were considered which results in 36 test problem instances. The 6 problem sizes were chosen to include 6, 7,8,9,10,12 and 15 parts. The sizes were adopted based on experience, such that they could be handled by LINGO in order to make the comparison possible. For each problem size, six different types of input data were used. Input data consist mainly of MCIM and corresponding setup times which were generated by pseudo random function in C++ programming language to maintain a random and unbiased distribution of the input data for the problem instances.

Other parameters are common among all problem sizes: parts are selected to be grouped with four different machine types in three different cells so as to minimize the overall cost of setup, machine and intercellular movement. Besides, while the cell upper bound and lower bound on the number of machines are 4 and 1 respectively, constraint 12 has been relaxed. These problem instances were run on LINGO 9 (LINDO Systems Incorporation) on a 2 MHZ PC. The setup times were generated pseudo randomly within the range of 10 and 40 minutes while the MCIM is formed by randomly picked binary variables. As for the GA based heuristic, the termination criteria were set at 120 generations after which the program would terminate the computation. The best objective values from the GA were picked after several runs of the program and the computation time was measured against the generation in which the best objective was achieved for the first time during the run. Table 4 shows the results from GA based heuristic in comparison with LINGO.

Table 4Results for GA-based heuristic and LINGO for 36 problem instances in 6 sizes

	GA		**Lingo**			**GA**		**Lingo**	
	Best obj	**Time**	**Best obj**	**Time**		**Best obj**	**Time**	**Best obj**	**Time**
7 parts		sec		hrs	**10 parts**		sec		hrs
	50	0.250	50	1		124	0.219	140	1
	22	0.250	22	*		83	0.218	92	1
	11	0.234	10	*		122	0.203	120	1
	22	0.250	21	*		166	0.219	189	1
	60	0.156	63	1		96	0.219	111	1
	32	0.250	32	1		117	0.218	129	1
8 parts					**12 parts**				
	63	0.782	63	1		214	0.203	236	1
	118	0.005	175	1		112	0.297	123	1
	76	0.469	76	1		122	0.297	133	1
	46	0.005	44	1		156	0.261	152	1
	55	0.515	55	1		256	0.281	256	1
	75	0.781	74	1		142	0.328	159	1
9 parts					**15 parts**				
	95	0.127	107	1		347	0.764	**	1
	84	0.172	105	1		237	0.503	216	1
	97	0.172	116	1		366	0.331	**	1
	108	0.187	114	1		344	0.489	379	1
	80	0.172	86	1		298	0.564	336	1
	125	0.141	150	1		307	0.437	393	1

*Optimal solution was found by LINGO

**LINGO did not provide a feasible solution within 1 hour of computation time limit

The results from the GA based heuristic tailored for the current model deem reasonably good delivered in a short period of time as compared with those from LINGO within 1 hour of computation. On average, over 80% of the time, the GA-base heuristic has either outperformed or equaled the best objective values provided by LINGO within 1 hour of computation time. The fact that the heuristic has delivered results in a very short amount of computation time provides a practical tool to solve different problem instances of the presented model.

7. CONCLUSIONS AND FUTURE WORK

In this paper, we presented a cell formation model based on sequence-dependent setup time which can be used as a sub-model with general application in cellular manufacturing models including any type of manufacturing cell and any type of setup time.

We argued that setup time reduction in CMS may not be fully ensured unless it is directly targeted in the model. Furthermore we stressed that setup time, generally being a changeover, requires three elements of incoming part, outgoing part and common machine and compared it with some definitions used in the literature. We concluded that a sequence dependent changeover as defined above would be inclusive of other types as a general case that accommodates adaptation to special cases.

We then improvised a GA based heuristic that could provide reasonably good results in a short amount of time when LINGO would fail to do so in a reasonable amount of computation time. We believe that our GA based heuristic could be even further improved in terms of accuracy; however the computation time would usually increase when higher level of accuracy is sought. The incorporation of this model in advanced CMS models with real life features including various operations for different parts will help achieve a more accurate cost optimization as compared to when our model is not included. Also application of other Meta-heuristics to this model and comparison with GA-based heuristic will be our future work.

REFERENCES

[1] Ahmed P.,Tavakkoli-MoghaddamR.,SafaeiN. (2004), A comparison of heuristic methods for solving a cellular manufacturing model in a dynamic environment;*University of Wolverhampton, Working Paper Series*.

[2] Al-Mubarak F., Canel C., Khumawala B.M. (2002), A simulation study of focused cellular manufacturing as an alternative batch-processing layout;*Internationl Journal of Production Economics*.

[3] AtmaniA., Lashkari R.S., Caron R.J. (1995), A mathematical programming approach to joint cell formation and operation allocation in cellular manufacturing;*International Journal of Production Research* 33;

[4] Atmani A. (1995),A production planning model for flexible manufacturing systems with setup cost consideration;*Computers & Industrial Engineering*29; 1-4.

[5] Chang P-T, Lee E.S. (2000), A multi-solution method for cell formation exploring practical alternatives in group technology manufacturing;*Computers & Mathematics, with application* 40

[6] Chu C-H., Tsai M. (1990), A comparison of three array-based clustering techniques for manufacturing cell formation*; International Journal of Production Research* 28(8); 1417-1433.

[7] Cox III J.F., Blackstone J.H., Spencer M.S. (1995),*APICS dictionary 8t ed*., *Fall Church, VA*.

[8] Damodaran V., Lashkari R.S., Singh N., (1992),A production planning model for cellular manufacturing systems with refixturing consideration*; International Journal of Production Research*30.

[9] De Lit P., Falkenauer E., Delchambre A. (2000), Grouping genetic algorithms: an efficient method to solve the cell formation problem;*Mathematics and Computers in Simulation* 51; 257–271.

[10] Goncalves J.F, Resende G.C. (2002), A hybrid genetic algorithm; *AT&T Labs Research Technical Report, Oct.2002.*

[11] Kannan VR.,Ghosh S. (1995), Using dynamic cellular manufacturing to simplify scheduling in cell based production systems;*Omega, Int.j.Mgmt.Sci,* 23(4); 443-452.

[12] LaScola Needy K.,Billo R. E.,Colosimo Warner R. (1998), A cost model for the evaluation of alternative cellular manufacturing configurations;*Computers industrial. Engineering* 34(1).

[13] Lashkari R.S.,Boparai R., Paulo J. (2004), Towards an integrated model of operation allocation and material handling selection in cellular manufacturing systems;*International Journal of Production Economics* 87.

[14] Mahdavi I.,MahadevanB. (2008), CLASS: An algorithm for cellular manufacturing system and layout design using sequence data;*Robotics and Computer-Integrated Manufacturing* 24; 488–497.

[15] Molleman E.,Slomp J.,Rolefes S. (2002), The evolution of a cellular manufacturing system- a longitudinal case study;*International Journal of Production Economics* 75; 05-322.

[16] Moon C., Gen M. (1999), A genetic algorithm-based approach for design of independent manufacturing cells;*International Journal of Production Economics*; 60-61.

[17] Morris J.S.,Tersine R.J. (1989), A comparison of cell loading practices in group technology;*Journal of Manufacturing and Operations Management* 2; 299-313.

[18] Morris J.S.,Tersine R.J. (1990), A simulation analysis of factors influencing the attractiveness of group technology cellular layouts;*Management Science* 36.

[19] Murugan M.,Selladurai V. (2005), Manufacturing cell design with reduction in setup time through genetic algorithm;*Journal of Theoretical and Applied Information Technology.*

[20] Ohta H., Nakamura M. (2002), Cell formation with reduction in setup times;*Computers & Industrial Engineering* 42.

[21] Rajamani D., Singh N.,Aneja P. (1992), A model for cell formation in manufacturing systems with sequence dependence; International Journal of Production Res. 30; 1227-1235.

[22] Safaei N.,Tavakkoli-Moghaddam R. (2009), Integrated multi-period cell formation and subcontracting production planning in dynamic cellular manufacturing;*International Journal of Production Economics*.

[23] Shtubtaa A. (1989), Modelling group technology cell formation as a generalized assignment problem;*International Journal of Production Research* 27(5); 775 – 782.

[24] Singh N.,Rajamani R. (1995), Cellular Manufacturing Systems;*Chapman & Hall Inc UK*.

[25] Snead C.S. (1989),Group Technology;*Van Norstrand Reinhold New York.*

[26] Tavakkoli-Moghaddam R.,Aryanezhad M.B.,Safaei N Azaron A. (2005), Solving a dynamic cell formation problem using metaheuristics;*Applied Mathematics and Computation* 170; 761–780.

[27] Vilas C.O.Y.,Vandael N. (2002),A cost operations based product heterogeneity index;*International Journal of Production Economics* 79; 45-55.

[28] Wemmerlov U.,Hyer N.L. (1989),Cellular manufacturing in the US industry: A survey of users; *International Journal of Production Research* 27(9).

[29] Wua X., Chu C-H., Wang Y., YanW. (2007), A genetic algorithm for cellular manufacturing design and layout; *European Journal of Operational Research* 181; 156–167.

[30] Wub X., Chu C-H., Wang Y.,Yue D. (2007), Genetic algorithms for integrating cell formation with machine layout and scheduling;*Computers & Industrial Engineering* 53; 277–289.