Adaptive Parameters within Genetic Algorithm for Machine Layout Design

Suthasinee Singpraya[*1], Surachid Dedklen[2], Srisatja Vitayasak[3] and Pupong Pongcharoen[*4]
Centre of Operations Research and Industrial Applications
Department of Industrial Engineering
Faculty of Engineering, Naresuan University, Phitsanulok, Thailand 65000
E-mail: srisatjav@nu.ac.th and pupongp@nu.ac.th

Abstract

Finding an optimal solution is always a crucial topic in the field of operations research and management science. In the stochastic search process, the performance of metaheuristics usually depends on the setting of its parameters. A majority of the research in this area is often focused on the static parameter settings adopted from previous research or the best guess approach. In this paper, an adaptive Genetic Algorithm (aGA) is proposed for solving the machine layout design (MLD) problem. In the adaptive process embedded in the aGA, the parameter was dynamically adjusted according to the standard deviation of fitness values during the evolution process. The proposed algorithm was aimed at minimising the total handling distance of materials flowing between non-identical rectangular machines located on the manufacturing shop floor. A series of computional experiments was designed and conducted using five datasets, four of which were adpoted from the literature with another larger dataset generated. Three GA adaptive parameters with three adaptive rates were investigated in the adaptive process, by which the adaptive rate of the mutation operator significantly affected the total material handling distances in the large problem. The statistical analysis of the experimental results suggested that the proposed aGA was able to increase the diversity of chromosomes during the searching process, especially for the largest-size problem. The appropriate adaptive parameters for each dataset were different. The average distances obtained from each problem using the proposed adpative GA parameter setting were significantly lower than those obtianed from the GA with the conventional setting. It was also found that the quality of the best-so-far solutions obtained from the GA with both adaptive and optimised parameter settings were statistically insignificant.

Keywords: Machine layout, Multiple rows, Genetic Algorithm, Adaptive mutation rate, Adaptive crossover rate

---

Corresponding author: e-mail: srisatjav@nu.ac.th and pupongp@nu.ac.th
[1,2] Students in Industrial Engineering Programme, Faculty of Engineering, Naresuan University
[3] Lecturer in Industrial Engineering Department, Faculty of Engineering, Naresuan University
[4] Assistant Professor in Industrial Engineering, Faculty of Engineering, Naresuan University [3] Lecturer in Industrial Engineering Department, Faculty of Engineering, Naresuan University
[4] Assistant Professor in Industrial Engineering, Faculty of Engineering, Naresuan University

1. Introduction

Genetic Algorithm (GA) described by Holland and Goldberg [1, 2], is classified as evolutionary algorithm, in which chromosomes are reproduced and mutated [3]. GA is a powerful optimisation tool that enables the fittest candidates among the population to survive, as commonly found in biological organisms [4]. The number of GA applications has increased in the last few decades and can be found in the production and operations management literature [5, 6]. GA has been widely applied to the area of industrial engineering, such as in machine layout design, scheduling, bin packing and other combinatorial optimisation problems. Each step of GA is controlled by parameters such as the number of generations, population size, rates of crossover and mutation, genetic operators and chromosome selection mechanisms. These parameter settings play an important role in GA performance.

The parameters can be constantly determined by either adopting those values from previous research or by using the best guess method, both of which are convenient approaches but do not guarantee the best performance of the searching process. The appropriate parameters setting used in previous research have been studied for particular problems, of which these settings do not guarantee a good performance for other problems. Another easier approach is the trial and error method, according to which the process of the finding best parameter values is very tedious and time consuming depending on the researchers' experience [7]. The results obtained from metaheuristics with poor parameter settings are usually premature and therefore practically not recommended. Nevertheless, a universal optimal parameter setting of GA does not exist [8] and a non-tuned parameter setting usually results in premature convergence [9].

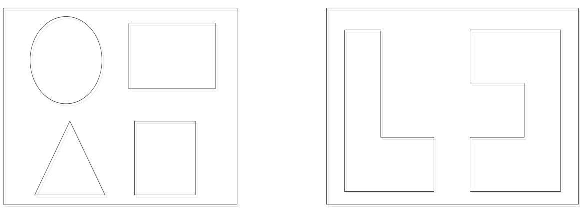
The adaptive GA parameter setting has been studied and reported in the literature. Those parameters were: rate of reproduction [10], population size [11], mutation operator, and the crossover and mutation rate [4, 12-15]. Adaptive GA (aGA) is able to improve GA performance, as revealed in the literature. Therefore, for example, adjusting the rates of GA operators for the next generation in order to obtain faster and better solutions [10], aGA reaches the best result more quickly than the simple GA in bilateral multi-issue negotiation [16].

Machine layout design (MLD) is the process of positioning machines on the shop floor area and has effects on production cost and time [17]. An effective facility layout can help to reduce production costs by 10-30% [18]. The MLD problem has been classified as a Non-deterministic Polynomial-time hard (NP-hard) problem [19], which means that the amount of computation required to find solutions increases exponentially with problem size. Solving this kind of problem using full numerical methods, especially for a large size, can be computationally expensive. The approximation optimisation algorithms, such as GA [20-22], Simulated Annealing [23], Tabu Search [23], Shuffled Frog Leaping [24], Rank-based Ant System [25], Artificial Bee Colony [26], and Bat Algorithm [27], have been applied to solving the MLD problem but these do not guarantee an optimum solution [28].

There are many characteristics of the layout problem depending on the criterion used [21], such as manufacturing systems (fixed layout, process layout, product layout and cellular layout),

layout configurations (single row, multi-rows, loop layout, open field layout and multi-floor layout), machine position constraints (fixed and non-fixed orientation) [29], and facility shapes (irregular and regular shapes). A regular shape means a geometric shape, e.g. square, rectangle; otherwise, an irregular shape refers to a non-geometric shape, e.g. L-shape, U-shape, as shown in Figure 1.

The objectives of this paper were: i) to describe the application of Genetic Algorithm (GA) with a dynamically-adjusted crossover rate, mutaion rate, and both the crossover and mutaion rate for designing a non-identical rectangular machine layout in a multiple-row environment, aiming to minimise the total material handling distance; and ii) to investigate adaptive genetic parameter settings that have an influence on the solution quality.



a) Regular b) Irregular

Figure 1 Shape of machine

The paper is organised as follows: section 2 describes the process of adaptive Genetic Algorithm for solving the MLD problem, followed by machine layout design in section 3. The experiment results are presented in section 4. Finally, a conclusion is drawn in section 5.

2. Adaptive Genetic Algorithm (aGA)

Adaptive Genetic Algorithm is a dynamic parameter setting according to predefined criteria. The self-adaptive rates of three parameters, crossover, mutation, and both crossover and mutation, were investigated in this problem. The pseudo-code of the proposed aGA for the MLD is shown in Figure 2.

Input problem dataset (M, M_W, M_L, N, M_S)
  Parameter setting (Pop_size, Gen, r_c, r_m, F_L, F_W, G)
  Randomly create initial population
  Set i = 1 (first generation)
  While i ≤ Gen do

Figure 2 Pseudo code of aGA for MLD

    For j = 1 to cross do (cross = round ((r_c x Pop_size)/2)))
      Ordering Crossover (OX) operation
    End loop for

    For k = 1 to mute do (mute = round (r_m x Pop_size))
      Two Operations Random Swap (2ORS) operation
    End loop for
    Arrange machines row by row based on F_L, F_W and G
    Calculate material handling distance
    Elitist selection
    If (SD of current generation > SD of previous generation),
      Current rate = current rate - adaptive rate
    If (SD of current generation < SD of previous generation),
      Current rate = current rate + adaptive rate
    else Current rate = current rate
    Chromosome selection using roulette wheel method
    i = i + 1
  End loop while
Output the best solution

Figure 2 Pseudo code of aGA for MLD

The process of aGA can be described in the following form:

Step 1: encode the problem to create a set of chromosomes. Each chromosome represents a set of arranging machines, so the length of chromosome is equal to the total number of machines to be arranged.

Step 2: prepare input data: the number of machines (M), the dimension of machines (width: M_W x length: M_L), the number of parts (N) and the product's machine sequences (M_S); and identify the parameters: population size (Pop_size), number of generation (Gen), rate of crossover (r_c), rate of mutation (r_m), floor length (F_L), floor width (F_W) and gap between machines (G).

Step 3: randomly generate an initial population based on population size.

Step 4: generate a new offspring using the Ordering Crossover [30] adopted from previous work [31] and Two Operations Random Swap [32] adopted from literature [33].

Step 5: arrange the machines row by row based on F_L and F_W.

Step 6: evaluate the fitness function value.

Step 7: select the best chromosome having the shortest material handling distance using the Elitist Selection.

Step 8: compare the standard deviation (SD) of fitness value between the current generation and previous generation, and then adjust the current rate of the adaptive parameter: crossover rate, mutation rate, or both crossover and mutation rate, by the adaptive rate; for example, the adaptive parameter is the mutation rate if (the SD of the current generation > SD of the previous generation), (r_m = r_m – adaptive rate). If (the SD of the current generation < SD of the previous generation), (r_m = r_m + adaptive rate) else (r_m = r_m).

Step 9: choose chromosomes for the next generation by using the Roulette Wheel Selection [34]

Step 10: stop the GA process according to the Gen; otherwise, return to step 4. When the GA process is terminated, the best-so-far solution is concluded.

3. Non-identical machine layout design

Most machines have a rectangular shape and are of different sizes and different models or types, which are called non-identical machines. Arranging machines in a multiple-row environment is when machines are placed row by row within a restricted area such, as that shown in Figure 3. Facility refers to machine, and flow path means movement of material handling equipment, e.g. automated guided vehicles which can move to the left or right side of the row and then move up or down to the destination row.

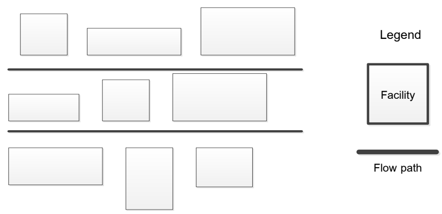


Figure 3 Example of multiple-row MLD [21]

The operated point of each machine is centroid. The objective function for this study is to minimise the material handling distance, as illustrated in equation (1).

$$Minimise\ \ z = \sum_{j=1}^{M}\sum_{i=1}^{M} f_{ij} d_{ij} \qquad (1)$$

M is a number of machines, *i* and *j* are machine indexes (*i* and *j* = *1, 2, 3,..., M*), f_{ij} is the frequency of material flow between machine *i* and *j*, and d_{ij} is the distance from machine *i and j* (*i* ≠ *j*).

4. Experimental results

In this work, a computational experiment was conducted using five datasets consisting of four MLD benchmarking datasets (shown in Table 1) adopted from the literature [35] and one largest-size dataset generated by the research, all of which had different sizes according to the number of machines and products. Dataset M10N3 means that there are three products to be processed on ten non-identical rectangular machines. A machine layout designing program was developed and coded in modular style using the Tool Command Language and Tool Kit (Tcl/Tk) programming language [36]. An experiment was designed and conducted on a personal computer with Intel Core i5 2.8 GHz and 4 GB DDR3 RAM.

Table 1 Testing datasets

Dataset	Number of machines (M)	Number of products (N)
M10N3	10	3
M20N5	20	5
M15N9	15	9
M30N10	30	10
M50N25	50	25

The population size and number of generations investigated in a previous work [8] have been set at 25 and 100 respectively. The initial r_c and r_m were set at 0.5 and 0.1. The three levels of adaptive rate were 0.05, 0.1 and 0.2, and the three adaptive parameters were the crossover rate, the mutation rate, and both the crossover and mutation rate. With five datasets, each of which took thirty replications, the total computational runs of 1,350 were carried out. The results obtained from the computational experiment were analysed using the analysis of variance (ANOVA), as shown in Table 2, in which the degree of freedom (DF), F value, and P value for the adaptive rate are given.

According to the ANOVA table, the adaptive rate of mutation was a significant factor, with a 95% confident interval with the P values less than or equal to 0.05 for only M30N10. The results suggest that the adaptive rate has an effect on solution quality in the large problem, which directly involves the numbers of machines and products. The greater the number of products and machines considered, the more important are the adaptive rates obtained. However, in M50N25, the adaptive rates had no statistically-significant effect on material handling distance. The appropriate adaptive rates for each MLD problem dataset were different. For the other adaptive parameters, the adaptive rates also had no statistically-significant effect on material handling distance. These results may be due to the unsuitable values of the adaptive rate.

Mean and standard deviation (SD) of the total material handling distances obtained from the aGA with different adaptive rates are summarised in Table 3, in which the best solutions for M10N3, M20N5, and M50N25 result from adjusting both the crossover and mutation rate with a 0.1 adaptive rate. For M15N9 and M30N10, changing the mutation rate with the adaptive rate set at 0.2 is best. The performance of the aGA depends on both the GA parameter and adaptive rate. When considering the standard deviation, the problem dataset M50N25 had the highest values of mean and SD because of the number of machines and type of products. When the number of machines was increased, the feasible solutions were increased. A variety of solutions had an effect on the standard variation.

From Table 4, it can be seen that the solutions were obtained from GA with three parameter setting approaches: i) GA with adaptive parameter, called aGA, which is the best adaptive parameter for each dataset referred from Table 3; ii) GA with optimised parameter setting, which can be studied according to the design of experiment (DOE) [37], has been investigated in previous work [8]; and iii) GA with a conventional parameter setting where the parameter value was guessed by the researcher (r_c =0.5, r_m=0.01, pop_size/Gen = 25/100). The results showed that the mean of total handling distance obtained from all datasets was minimised by the GA via DOE. The aG

A was more efficient than the GA by using the guess method. There were a few differences in solutions between the aGA and GA via DOE.

The student's t-test was applied to compare the differences in the mean of the material handling distance within the parameter setting approaches for each dataset. The results (see Table 5) showed that there were no statistically-significant differences between the aGA and GA via DOE with a 95% confident interval in dataset M20N5 and M30N10. The performance of the aGA was similar to the GA with an optimised parameter setting if the adaptive rate and adaptable parameter were appropriate.

A comparison of the solutions obtained from the GA via DOE with aGA showed that the SD's value in all datasets was lower than the aGA, except M30N10. However, the quality of the solution from those two methods was the same in some datasets. An adaptive parameter was useful for increasing the diversity of chromosomes during the searching process. The solutions obtained from the GA by guessing had the highest value of SD but the quality of solutions was worst in all datasets.
The performance of the GA resulted from the parameter setting.

Table 2 Analysis of variance (ANOVA)

Adaptive Parameter	Source	DF	M10N3		M20N5		M15N9		M30N10		M50N25	
			F	P	F	P	F	P	F	P	F	p
Crossover rate	Adaptive rate	2	0.37	0.693	0.02	0.981	0.89	0.415	0.38	0.682	1.13	0.327
	Error	87										
	Total	89										
Mutation rate	Adaptive rate	2	0.30	0.743	0.4	0.671	0.54	0.587	3.49	0.035	0.14	0.868
	Error	87										
	Total	89										
Crossover and Mutation rate	Adaptive rate	2	0.18	0.836	0.22	0.801	0.51	0.600	0.47	0.628	1.13	0.327
	Error	87										
	Total	89										

Table 3 Mean and standard deviation (SD) of aGA with different adaptive rate in each dataset

Adaptive Parameter	Datasets	±0.05		±0.1		±0.2	
		Mean	SD	Mean	SD	Mean	SD
Crossover rate	M10N3	193.162	8.047	194.747	7.493	194.653	8.515
	M20N5	1392.555	62.390	1395.587	67.095	1395.030	64.065
	M15N9	1442.437	55.193	1430.050	42.750	1426.110	47.536
	M30N10	4998.578	145.989	4968.588	144.479	4996.110	150.678
	M50N25	12029.311	536.382	11834.971	520.606	11994.381	542.880
Mutation rate	M10N3	193.970	8.472	193.025	7.270	193.392	8.112
	M20N5	1378.790	68.454	1367.558	46.345	1367.317	53.057
	M15N9	1430.070	50.560	1429.130	51.883	1417.773	51.114
	M30N10	4866.137	149.266	4929.745	130.549	4838.652	130.186
	M50N25	11912.762	486.232	11848.978	526.813	11907.602	529.270
Crossover and Mutation rate	M10N3	193.860	8.522	192.717	8.316	192.798	7.857
	M20N5	1372.562	70.106	1364.293	67.978	1375.425	63.009
	M15N9	1428.273	50.514	1424.373	50.014	1514.713	46.721
	M30N10	4912.932	159.539	4889.373	111.812	4924.157	150.864
	M50N25	11846.365	495.781	11730.949	536.295	11835.573	536.059

Table 4 Comparison of solutions for proposed aGA and GA with/without optimised parameter setting

Handling distance (m)	M10N3			M20N5			M15N9			M30N10			M50N25		
	aGA $r_c r_m \pm 0.1$	GA: DOE	GA: guess	aGA $r_c r_m \pm 0.1$	GA: DOE	GA: guess	aGA $r_m \pm 0.2$	GA: DOE	GA: guess	aGA $r_m \pm 0.2$	GA: DOE	GA: guess	aGA $r_c r_m \pm 0.1$	GA: DOE	GA: guess
Min.	186.9	186.9	187.1	1201.7	1231.7	1359.4	1346.9	1347.8	1444.8	4590.7	4524.4	4869.0	10783.9	10475.8	11378.9
Max.	224.4	187.6	233.3	1507.0	1448.6	1566.5	1543.2	1417.4	1581.4	5075.1	5041.2	5548.7	12729.8	12629.0	14085.4
Mean	192.7	**187.4**	207.9	1364.3	**1361.2**	1474.7	1417.8	**1382.0**	1504.1	4838.6	**4770.6**	5156.9	11730.9	**11373.2**	12723.0
SD	8.316	0.29	13.87	67.98	49.226	54.757	51.114	22.119	37.29	130.19	141.77	165.47	536.29	491.22	673.00

Table 5 P-value of student's t-test

Dataset	DOE and guess	aGA and guess	DOE and aGA
M10N3	0.000	0.000	0.001
M20N5	0.000	0.000	0.839
M15N9	0.000	0.000	0.001
M30N10	0.000	0.000	0.058
M50N25	0.000	0.000	0.009

5. Conclusions

This paper presents the application of adaptive Genetic Algorithm for designing non-identical rectangular machine layouts and also investigated the adaptive genetic parameter setting, which had an influence on the solution quality. The rate of crossover, mutation, and both crossover and mutation was dynamically adjusted according to the standard deviation of fitness values during the evolution process. Three adaptive rates were studied: 0.05, 0.1, and 0.2. The computational experiment was designed using five datasets, in which four MLD benchmarking datasets were adopted from the literature. The experimental results indicated that the an influence on objective function value. The performance of the proposed aGA depended on both the adaptive parameter and adaptive rate. The proper adaptive rate and parameter also helped to yield the material handling distance. The average distances obtained from the proposed aGA were significantly lower than those obtianed from the GA with a conventional setting. However, it had no statistically-significant difference from the GA results with the optimised parameter setting iden -tified via the statisical design of the experiment in some datasets. Further study is being carried out by adopting other adaptive rates to proposed aGA for solving different sizes of the machine layout problem. The adaptive rate for adjusting the current rate of the adaptive parameter to increase or decrease can be unequally set, e.g. r_c+0.2/-0.05

Acknowledgements

This work was a part of the research project funded by the Naresuan University Research Fund under grant number R2555DC035

References

[1] D. E. Goldberg, *Genetic algorithms in search, optimization and machine learning*: Addison-wesley publishing company, inc., 1989.

[2] J. H. Holland, "Outline for a logical theory of adaptive systems," *Journal of ACM,* vol. 3, pp. 297-314, 1962.

[3] M. S. Alam*, et al.*, "Diversity Guided Evolutionary Programming: A novel approach for continuous optimization," *Applied soft computing,* vol. 12, pp. 1693-1707, 2012.

[4] Z. Ye*, et al.*, "Some improvements on adaptive genetic algorithms for reliability-relatedapplications," *Applied soft computing,* vol. 95, pp. 120-126, 2010.

[5] R. G. Aytug*, et al.*, "Use of genetic algorithms to solve production and operations manage -ment problems: A review. ," *International Journal of Production Research,* vol. 41, pp. 3955-4009, 2003.

[6] S. S. Chaudhry and W. Luo, "Application of genetic algorithms in production and operations management: a review," *Inter-national Journal of Production Research,* vol. 43, pp. 4083-4101, 2005.

[7] Z. Michalewicz and D. V. Fogel, *How to solve it: Modern heuristics*: Springer, 2010.

[8] S. Singpraya and S. Dedklen, "Studying crossover operators and adaptive genetic algorithm for designing multiple rows layout," Bachelor, Industiral Engineering, Naresuan University, Pitsanullok, 2011.

[9] Z. Bingul, "Adaptive genetic algorithms applied to dynamic multiobjective problems," *Applied Soft Computing,* vol. 7, pp. 791-799, Jun 2007.

[10] N. Van Hop and M. T. Tabucanon, "Adaptive genetic algorithm for lot-sizing problem with self-adjustment operation rate," *International Journal of Production Economics,* vol. 98, pp. 129-135, Nov 2005.

[11] Y. Liang and K. S. Leung, "Genetic Algorithm with adaptive elitist-population strategies for multimodal function optimization," *Applied Soft Computing,* vol. 11, pp. 2017-2034, Mar 2011.

[12] C. H. Chan and G. J. Liu, "Hysteresis identification and compensation using a genetic algorithm with adaptive search space," *Mechatronics,* vol. 17, pp. 391-402, Sep 2007.

[13] P. C. Chang*, et al.*, "Adaptive multi-objective genetic algorithms for scheduling of drilling operation in printed circuit board industry," *Applied Soft Computing,* vol. 7, pp. 800-806, Jun 2007.

[14] Y. X. Feng and K. L. Pheng, "An exact schema theorem for adaptive genetic algorithm and its application to machine cell formation," *Expert Systems with Applications,* vol. 38, pp. 8538-8552, July 2011.

[15] L. Wang and D. B. Tang, "An improved adaptive genetic algorithm based on hormone modulation mechanism for job-shop scheduling problem," *Expert Systems with Applications,* vol. 38, pp. 7243-7250, Jun 2011.

[16] L. Jian, *et al.*, "An adaptive genetic algorithm and its application in bilateral multi-issue negotiation," *The Journal of China University of Posts and Telecommunications,* vol. 15, pp. 94-97, 2008.

[17] M. Ficko, *et al.*, "Designing the layout of single and multiple rows flexible manufacturing system by genetic algorithms," *Journal of Materials Processing Technology,* vol. 157, pp. 150-158, Dec 2004.

[18] J. A. Tompkins, *et al.*, "Facilities Planning," 2010.

[19] E. M. Loiola, *et al.*, "A survey for the quadratic assignment problem," *European Journal of Operational Research,* vol. 176, pp. 657-690, 2007.

[20] J. Balakrishnan and C. H. Cheng, "A note on "a hybrid genetic algorithm for the dynamic plant layout problem," *International Journal of Production Economics,* vol. 103, pp. 87-89, Sep 2006.

[21] A. Drira, *et al.*, "Facility layout problems: A survey," *Annual Reviews in Control,* vol. 31, pp. 255-267, 2007.

[22] P. Ariyawong, "A Genetic Algorithm for the single and multiple rows layout problem in flexible manufacturing systems," Master, Information Technoology, Naresuan University, Pitsanulok, Thailand, 2007.

[23] P. Wangta and P. Pongcharoen, "Designing Machine Layout Using Tabu Search and Simulated Annealing," *Naresuan University Journal,* vol. 18, pp. 1-8, 2010.

[24] T. Iamtan and P. Pongcharoen, "A comparison of Swap and Adjustment Techniques in Shuffled Frog Leaping Algorithm for Solving Machine Layout Design Problem," in *The Industrial Engineering Network Conference*, Khonkhan, Thailand, 2009.

[25] N. Leechai, *et al.*, "Comparison on Rank-based Ant System and Shuffled Frog Leaping for design multiple row machine layout," *SWU Engineering Journal,* vol. 4, pp. 102-115, 2009.

[26] P. Soimart and P. Pongcharoen, " Multi-row machine layout design using Artificial Bee Colony," in *The International Conference on Economics and Business Information*, Bangkok, Thailand, 2011.

[27] K. Dapa and P. Loreungthup, "A Bat Algorithm for the multiple rows layout problem in flexible manufacturing systems," Bachelor, Industrial Engineering, Naresuan University, Pitsanulaok, Thailand, 2012.

[28] P. Pongcharoen, *et al.*, "Determining optimum Genetic Algorithm parameters for scheduling the manufacturing and assembly of complex products," *International Journal of Production Economics,* vol. 78, pp. 311-322, Aug 2002.

[29] S. Vitayasak and P. Pongcharoen, "Machine selection rules for designing multi-row rotatable machine layout considering rectangular-to-square ratio," *Journal of Applied Operational Research,* vol. 5, pp. 48-55, 2013.

[30] L. Davis, "Job shop scheduling with genetic algorithm," in *Proceedings of the First International Conference on Genetic Algorithms and their Applications*, 1985, pp. 136-140.

[31] S. Singpraya and S. Dedklen, "Studying crossover operators and adaptive genetic algorithm for designing multiple rows

layout," Bachelor Industiral Engineering, Naresuan University, Pitsanullok, 2011.

[32] T. Murata and H. Ishibuchi, "Performance evaluation of genetic algorithms for flow shop scheduling problems," in *Proceedings of the First IEEE Conference on Evolutionary Computation*, 1994, pp. 812-817.

[33] P. Pongcharoen*, et al.*, "Applying designed experiments to optimize the performance of genetic algorithms used for scheduling complex products in the capital goods industry," *Journal of Applied Statistics,* vol. 28, pp. 441-455, Mar-May 2001.

[34] M. Gen and R. Cheng, *Genetic Algorithms and Engineering Design*. Newyork : JOHN WILEY & SONS, INC., 1997.

[35] A. C. Nearchou, "Meta-heuristics from nature for the loop layout design problem," *International Journal of Production Economics,* vol. 101, pp. 312-328, Jun 2006.

[36] J. K. Ousterhout, *Tcl and Tk tookit*, 2nd ed.: Addison Wesley, 2010.

[37] D. C. Montgomery, *Design and analysis of experiments*: Wiley, 2005.