http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (4): 728-734, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.728.734

A Dual-system Method with Differential Evolution and Genetic Algorithm for Loop-based Station Sequencing Problem

Shuai Ma, Zhuang-Cheng Liu and Yan-Jun Shi
School of Mechanical Engineering, Dalian University of Technology,
Dalian 116024, People's Republic of China

Abstract: The facility layout problem is one of the most complex problems in many industries and the Loop-based Station Sequencing Problem (LSSP) is a classical sub-problem. In this study, a dual-system method based on Differential Evolution (DE) and Genetic Algorithm (GA) (DDEGA) was presented to solve the LSSP. The DDEGA duplicates the system P, which represents the original problem, as systems A and B. The systems A and B are solved by DE and GA, respectively. Since the elite migration between two systems can contribute to increasing the diversity and decreasing the premature convergence, the DDEGA can obtain better solutions and robustness. Numerical studies on four different scales showed that the proposed method can obtain a challenging solution.

Key words: Dual-system, differential evolution, genetic algorithm, loop-based station sequencing problem

INTRODUCTION

Flexible Manufacturing Systems (FMS) play a crucial role in modern advanced manufacturing. Facility layout problem in FMS is a manufacturing setting designed to satisfy the contemporary market demands, such as productivity, flexibility and on-time delivery (Drira *et al*., 2007). About 20-50% of the total production cost is allocated to facility layout and materials handling and an appropriate placement of facilities may reduce at least 10-30% of the total operating expenses (Tompkins, 2010). Therefore, facility layout in early design period needs to be schemed with much greater effort.

The facility layout in a FMS is typically determined by the type of the material handling devices used, such as Automated Guided Vehicles (AGVs), conveyors and gantry robots. The most commonly used types of facility layouts are the following, (1) Linear single row layout, (2) Linear double row layout, (3) Cluster layout based on gantry robot, (4) Semi-circular layout and (5) Closed loop layout (Kusiak and Heragu, 1987). Compared with other layout types, the loop layout has been generally studied due to relatively low initial costs and high flexibility (Afentakis, 1989). The original Loop Layout Design Problem (LLDP) is shown in Fig. 1.

This study addresses the unidirectional Loop-based Station Sequencing Problem (LSSP) (Afentakis, 1989), i.e., how to determine an order of facilities to minimize the material-handling costs measured by the total number of times the material cross the loading/unloading station, regardless of the size, shape and orientation. Since LSSP had been proven non-deterministic polynomial-time hard (NP-hard) (Leung, 1992), heuristic algorithms are naturally the most promising methods.

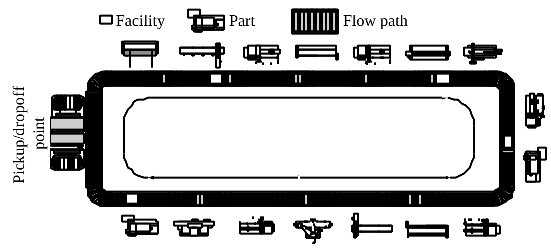

Fig. 1: Loop layout in flexible manufacturing systems

Since, Afentakis (1989) proposed a graph theoretic model for designing a loop network to minimize the average number of machines that all workpieces cross per unit time, many researchers have developed heuristics techniques for LLDP. Leung (1992) solved the MIN-SUM LLDP by using a heuristic which constructed a layout from a solution to the linear-programming relaxation based on a graph-theoretic framework. Kaku and Rachamadugu (1992) addressed the problem of minimizing the material handling costs for linear-track conveyor systems as a Quadratic Assignment Problem (QAP) using a heuristic method. Cheng and Gen (1998) introduced Genetic Algorithm (GA) for both MIN-SUM and

Corresponding Author: Yan-Jun Shi, School of Mechanical Engineering, Dalian University of Technology, Dalian 116024, People's Republic of China Tel: +8604118409130 Fax: +86041184708412

MIN-MAX LSSP. Tansel and Bilen (1998) discussed a heuristic algorithm based on moves and 2-way interchanges applied on permutations of facilities. Plaquin and Pierreval (2000) applied an Evolutionary Algorithm (EA) to solve cellular design problems taking into account specific constraints, such as the bounded size of cells, facilities that should stay together or be separated. Adel El-Baz (2004) described a GA for LLDP considering various material flow patterns with multi-products. Nearchou (2006) used subrange coded DE to solve LSSP. Pour and Nosraty (2006) coped LLDP as a QAP with an ant-colony algorithm. Kumar *et al.* (2008, 2009) solved LSSP using Particle Swarm Optimization (PSO) and Artificial Immune Systems (AIS) and the latter introduced crossover turntables into flow path of the loop layout to enhance the performance of the system. More information about the research status should be referred to the following surveys: classification of the facility layout problems (Drira *et al.*, 2007). State-of-the-art reviews of techniques applied to Facility Layout Design Problem (FLDP) (Singh and Sharma, 2006; Arikaran *et al.*, 2010).

In this study, a dual-system framework integrating Differential Evolution (DE) and GA is proposed to solve LSSP. The numerical experiment shows that the hybrid approach performed well on a problem set consisting of four characteristic test instances ranging from small to large size of LSSPs.

MAPPING APPROACH REVIEW

A candidate solution to LSSP is naturally described as a permutation vector concerning the arrangement of the facilities. In the literature, four mapping approaches prevail in practice.

Random-keys encoding (Bean, 1994) is widely applied to floating point chromosome with permutation property. The components of the chromosome are sorted and the order determines the sequencing of the facilities. For a five sequencing problem, the chromosome (.46, .91, .33, .75, .51) would represent the sequence 2→5→1→4→3, which means that the first facility will be installed at the second position.

Relative position encoding (Zheng and Teng, 2010) is proposed specifically to deal with LSSP or QAP, which is inspired by the procedure of insertion sort algorithm in computer science. The basic idea is explained as follows: the sequence facilities to be sorted are divided into sorted group ($f_0, f_1, ..., f_{i-1}$) and unsorted group ($f_i, f_{i+1}, ..., f_n$). Since a_0 is initialized at the first position, it does not participate in sorting. For the next facility to be sorted, f_i, there are (i+1) vacancies in the sorted group as the relative positions, which are respectively coded as (0, 1,..., I). According to the coding strategy, the chromosome ($p_1, p_2, ..., p_n$) determines the relative position of the facility sequencing ($f_1, f_2, ..., f_n$) correspondingly and $p_i \in \{0, 1, ..., i\}$. In Fig. 2, for instance, the fifth facility f_4 could be placed in five relative positions and these five vacancies are coded as 0, 1, 2, 3 and 4, respectively.

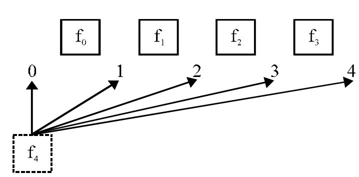

Fig. 2: An example of relative position encoding

Beside the former two strategies, sequence encoding and subrange encoding are also popular in combinatorial problems with permutation property. Applications of these two coding strategies in LLDP should be referred to Adel El-Baz (2004) and Nearchou (2006), respectively. The pseudo-code of the decoding procedure in MATLAB style is showed in Algorithm 1.

Algorithm 1:

```
% input: a chromosome RelativePos = (a1, a2, …, an-1)
%output: a permutation of the facilities FacilityPos = (b1, b2, …, bn)
FacilityPos = Decode_RelativePos (RelativePos)
FacilityPos = [1,zeros(size(RelativePos))];
for i = 1:size (RelativePos)
    Pos = RelativePos (i);
    index = find(FacilityPos > Pos);
    FacilityPos (index) = FacilityPos (index) + 1;
    FacilityPos (i+1) = Pos + 1;
end
```

A DUAL-SYSTEM METHOD USING DE AND GA

The term “dual-system” in algorithm field is firstly mentioned by Teng *et al.* (2010). The idea of dual-system is inspired by previous researches in numerous fields such as computer security, router in communication and even in biology and it is similar to the multi-island algorithm (Whitley *et al.*, 1998), which often has several populations to preserve genetic diversity and each population can potentially follow a different search trajectory through the search space. Dual-system and multi-island algorithm share the same intrinsic characteristics of parallelism and distributedness since different systems/islands interact with each other in order to achieve a common goal.

Assume that only two populations with individuals' migration between them in the multi-island algorithm, it turns out to be a dual-system which has system A and system B. In this study, system A executes the DE and system B uses a GA to follow a different search trajectory. The dual-system framework is shown in Fig. 3.

With elite migrations between system A and system B, genetic diversity is hoped to achieve. The interval generation of migrating the elite individuals from A to B

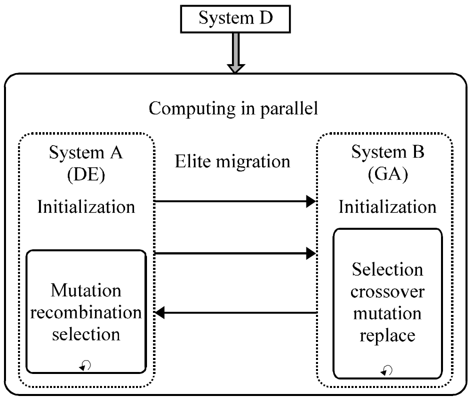

Fig. 3: The dual-system framework

is defined as K_{ab}, whereas K_{ba} is the migration interval generation from B to A. When the elite migration happens, the relatively poor individuals in the objective system are replaced. The migration ratio from A to B and from B to A are denoted as P_{ab} and P_{ba}, respectively. In the experiments, we set K_{ab} = 150, K_{ba} = 200, P_{ab} = 30%, P_{ba} = 10%.

FORMULATION OF THE LSSP

The LSSP concerns a set of machines {1, 2,..., N} arranged in a loop network and materials flow in single direction. The parts enter and leave the system only at the loading/unloading station and each part is characterized by its part-route, the sequence of machines it should visit to complete its processing. For a given part, suppose processing on machines j immediately follows processing on machine i and if the position of machine j is lower than that of machine i, the part must cross the loading/unloading station, which is called a reload. The total number of reloads necessary to complete the processing for all parts constitutes a measure of traffic congestion in the production system (Afentakis, 1989).

A solution to the LSSP corresponds to a permutation of machines by some measures. Usually two measures used for the evaluation of a loop layout design: (a) MIN-SUM, in which the objective is to minimize the total congestion of all the parts, while (b) MIN-MAX is to minimize the maximum reload time among parts of the same system. (a) is preferred here.

The average cost of the generated best loop layouts given by the function:

$$\text{Cost}_{\text{min-sum}}(S) = \sum_{i=1}^{M} \text{reload}_i$$

for the MIN-SUM LSSP, where S is a loop layout solution, i.e., a sequence of facilities and reload_i (I = 1…M) is the summation of the reloads for the ith part in the S layout.

The average percentage Solution Effort (SE) spent by an algorithm to achieve a near optimal solution:

$$SE(\%) = \frac{ne_{opt}}{ne_{Total}} \times 100$$

where, ne_{opt} is the number of evaluations performed by the algorithm to achieve the best solution and ne_{Total} the total number of evaluations performed by the algorithm.

EXPERIMENT

In order to evaluate the performance of the proposed dual-system framework, an experimental study consisting of four characteristic test instances ranging from small to large size of LSSPs given in Nearchou (2006) is illustrated. Types and routes are set beforehand. The facility number and the part number are (10, 3), (20, 5), (15, 9) and (30, 10) and the corresponding routes are shown in Table 1.

Two heuristics algorithms, DE and GA, are compared with the dual-system whose subsystems are DE and GA (DDEGA) under the same conditions, with two different mapping mechanisms. Specifically, six versions are examined: (1) Random-keys GA (GA_1), (2) Random-keys DE (DE_1), (3) Relative position-coded GA (GA_2), (4) Relative position-coded DE (DE_2), (5) Random-keys DDEGA (DDEGA _1) and (6) Relative position-coded DDEGA (DDEGA _2). The performance of the algorithms is quantified by five performance criteria: (1) The optimal value of the objective function cost(S), (2) The average value of the objective function cost(S), (3) The standard deviation of the objective function Cost(S), (4) The average percentage solution effort (SE) and (5) The processing time of the algorithm measured in second.

SETTING THE CONTROL PARAMETERS

Setting for the control parameters is hard to be perfect since the countless possible choices. In order to determine suitable settings for GA and DE to solve the LSSP, the forth test problem, the 30-machines-10-parts MIN-SUM LSSP, is employed as the test problem for parameters with the population size Np = 2-D = 60. The performance of GA is only depended on the crossover probability (Cr) since Gaussian mutation adopted; the performance of DE is depended on the values of two control parameters: the crossover probability (Pc) and the mutation scale factor (F). In order to determine the suitable settings, various crossover probability Pc∈{0.01, 0.05, 0.10,..., 0.95} is tested in GA; F∈{0.4, 0.5,... , 1.0} and Cr∈{0.01, 0.1, 0.2,..., 1.0} are experimented in DE, since the effective range of F is usually between 0.4 and 1.0 (Das and Suganthan, 2011). The algorithms are tested 10 times on each scheme

Table 1: The required machine sequence for each part

Part	Required machine sequence
LSSP with 10-machines, 3-parts	
1	2-1-6-5-8-9-3-4
2	10-8-7-5-9-6-1
3	9-2-7-4
LSSP with 20-machines, 5-parts	
1	4-2-3-12-1-9-16-18-5-8-20-15-14-6-11
2	10-9-1-3-18-17-5-6-2-11-4
3	17-11-6-8-7-15-16-9-1-20
4	14-17-11-3-16-5-13-18-20-19-12-10-6-8-15
5	6-18-8-4-2-7-5-9-14-19-1-20-10-16-11-15-13-12
LSSP with 15-machines, 9-parts	
1	4-2-5-1-6-8-14-9-11-3-15-12
2	3-2-15-14-11-1-7-10-4-5-13-6-9
3	5-6-11-15-2-12-3-4
4	10-9-4-14-2-3-15-8
5	11-2-4-14-5-3-15
6	8-10-12-11-15-13-1-14-4-5-3
7	5-11-10-3-7-13-8
8	7-3-2-8-4-10-6-15-13-9-1
9	11-13-3-1-12-14-4-8-9-2
LSSP with 30-machines, 10-parts	
1	6-3-4-18-5-1-14-24-26-7-11-30-23-21-13-27-9-16-17-2-25-8-15
2	17-9-11-8-10-22-24-13-2-29-23-21-25-16-4-20-26-18-15-12-27-6-3-7-28
3	13-2-6-29-21-3-14-24-12-15-17-8-1-22-28-10-7-30-20-19
4	7-2-6-11-21-8-16-30-1
5	3-17-1-2-20-22-8-6-26-19-14-11-15-12-7-16-21-10-28-23-18-4-27-24-25-13-30-9-5
6	30-9-2
7	15-9-30-19-12-3-6-5-8-14-7-28-23-1-29-24-27-2-13-4-26-16-11-10-25-21-22-20-18
8	7-19-5-4-9-16-3-14-28-13-11-2-21-10-17-22-26-23-29-30
9	21-4-1-6-11-22
10	12-6-17-15-13-30-26-18-14-9-7-11-23-2-4-25-24

LSSP: The Loop-based station sequencing problem

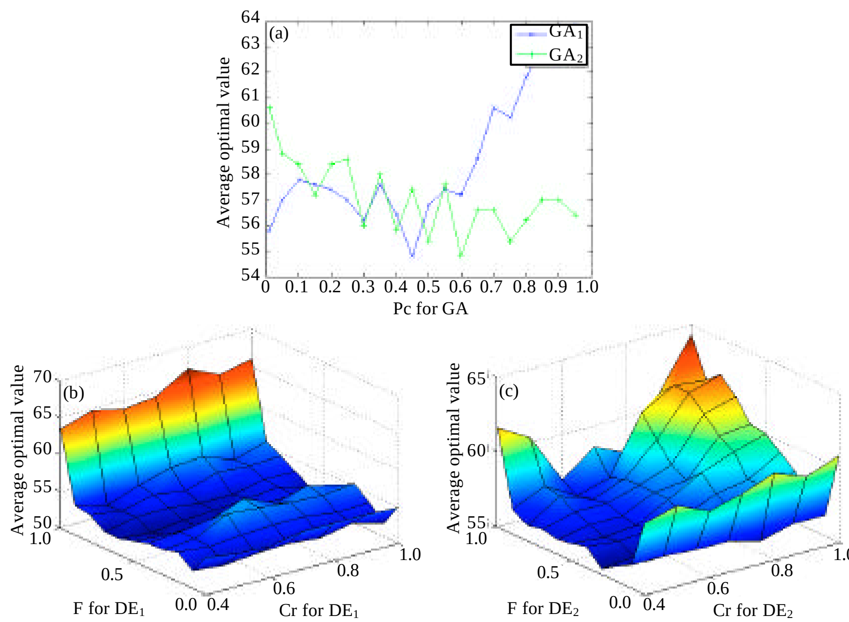

Fig. 4(a-c): (a) The effect of various Pc on the performance of GA_1 and GA_2, (b) The effect of F and Cr on the performance of DE_1 and (c) The effect of various F and Cr on the performance of DE_2

with running for a maximum of 1000 generations. The convergence charts are shown in Fig. 4.

The optimal control parameter values are obtained through the experiments above: Pc = 0.45 for random-keys

GA (GA_1), Pc = 0.60 for relative position-coded GA (GA_2), (F, Cr) = (0.5, 0.9) for random-keys DE (DE_1) and (F, Cr) = (0.4, 0.3) for relative position-coded DE (DE_2).

In the dual-system framework, system A is employed as the main system, which conducts the better one between GA and DE with the same encoding mechanism. The dual-system (A and B) are evolved cooperatively, which means elite individual migration is implemented at intervals of some generations. In this study, the number of elite individual from system A to system B is 15, the number reversely is 5 and the internal number of generation is 3 and 4, respectively.

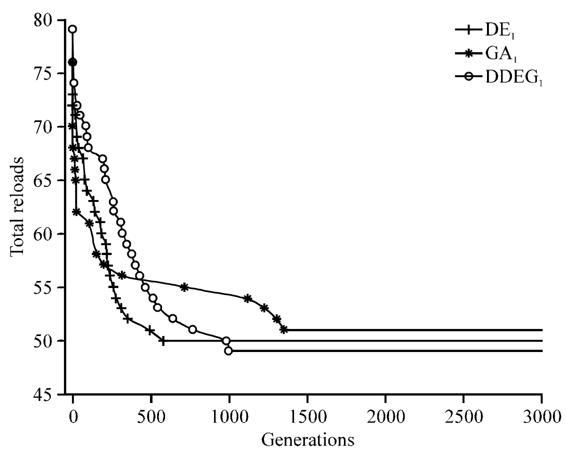

Fig. 5: Convergence curves of different algorithms with random-keys encoding mechanism

COMPARISON RESULTS

A maximum of 3000 generations is set to algorithms with their respective optimal parameter settings and all data are obtained by 50 independent runs. The results obtained by GA, DE and DDEGA with different encoding mechanisms are shown in Table 2, which including the optimal and average reload time with standard deviation, the processing time, the average percentage Solution Effort (SE) and the best order of machines for each family of parts. DDEGA achieves brilliant solutions in some critical indicators such as optimal, average and standard deviation (STD), albeit GA wins on the CPU-time. The convergence curves with different encoding mechanism are showed in Fig. 5 and 6, which show that DDEGA assimilates the merits of DE and GA and gets a better solution.

Figure 4 shows that (1) in GA, the relative position-coded mechanism could get similar results with a relatively wider range of crossover probability (Pc) and the quality of solution deteriorated rapidly when (i) Pc for the random-keys GA level off to 1, (ii) Pc for the relative position-coded GA approaches to 0, (2) in the random-keys DE, the mutation scale factor (F) played a more critical role than the crossover

Table 2: Comparative results for benchmark problems

Algorithm	Optimal	Average	STD	CPU-time (sec)	SE (%)	Optimal layout
LSSP with 10-machines, 3-parts						
GA_1	3	3.22	0.42	12.40	0.01	10-8-9-6-2-1-7-3-4-5
DE_1	3	3.02	0.14	18.61	2.69	6-5-10-8-9-2-1-3-7-4
DDEGA_1	3	3.02	0.14	13.84	0.99	10-8-9-2-6-3-7-4-1-5
GA_2	3	3.00	0.00	10.77	0.73	5-10-8-9-6-2-7-3-1-4
DE_2	3	3.40	0.49	20.13	0.56	10-8-9-6-3-2-7-5-4-1
DDEGA_2	3	3.00	0.00	14.81	1.37	10-6-5-8-9-3-2-7-4-1
LSSP with 15-machines, 9-parts						
GA_1	24	25.20	0.79	55.70	2.40	4-7-5-11-10-3-15-13-1-12-6-8-14-9-2
DE_1	24	24.68	0.55	126.02	17.83	7-4-5-11-10-13-3-2-1-6-15-12-8-14-9
DDEGA_1	24	24.72	0.54	91.44	12.56	5-11-7-10-3-15-13-1-6-8-14-9-2-12-4
GA_2	25	27.04	1.11	51.05	34.44	7-10-12-5-6-11-3-15-13-8-14-9-1-4-2
DE_2	25	25.00	0.00	122.17	7.03	7-10-12-14-6-9-4-5-11-3-2-15-13-8-1
DDEGA_2	24	24.92	0.27	90.56	47.83	7-4-5-11-13-10-3-1-6-15-8-14-9-2-12
LSSP with 20-machines, 5-parts						
GA_1	16	18.10	1.24	51.27	80.55	14-17-19-10-4-16-5-9-13-12-6-2-1-11-3-18-8-7-20-15
DE_1	16	17.90	0.99	125.83	45.50	14-10-6-16-19-18-17-5-9-8-11-4-2-7-1-20-15-3-13-12
DDEGA_1	16	17.02	0.68	92.00	8.47	14-10-16-18-17-5-6-9-19-8-4-2-11-7-1-3-20-15-13-12
GA_2	17	19.58	1.11	63.48	35.10	14-19-10-16-13-18-12-17-5-9-1-6-8-4-2-20-7-11-3-15
DE_2	17	17.44	0.54	129.88	49.89	10-16-14-18-17-5-6-11-8-9-20-4-2-7-15-19-3-13-12-1
DDEGA_2	16	16.38	0.57	101.72	16.94	14-19-13-12-10-6-16-18-17-5-8-4-2-11-9-7-1-20-3-15
LSSP with 30-machines, 10-parts						
GA_1	51	55.78	2.23	184.39	45.61	7-6-29-16-11-21-30-3-10-4-17-20-22-26-19-25-28-18-5-23-8-15-1-14-24-12-13-27-9-2
DE_1	50	53.08	1.38	521.67	75.66	21-8-15-12-13-10-3-4-17-30-1-27-14-22-9-7-2-6-25-20-28-29-24-26-16-11-19-23-18-5
DDEGA_1	49	53.10	1.56	357.23	67.46	21-15-12-10-3-13-4-17-1-14-27-9-7-2-25-6-29-16-11-22-30-24-20-28-26-23-18-19-5-8
GA_2	51	55.58	2.18	210.84	82.63	29-23-31-21-8-10-15-12-16-3-4-17-1-27-14-9-22-7-20-28-2-24-6-13-11-26-18-19-25-5
DE_2	54	55.20	0.67	493.05	94.50	29-18-15-21-14-24-13-12-10-27-9-7-16-6-5-11-3-17-2-4-22-25-30-28-20-26-19-8-23-1
DDEGA_2	51	55.20	0.49	345.94	73.87	26-25-23-21-19-18-15-13-12-27-9-8-10-6-5-16-3-17-1-14-7-11-2-4-29-30-22-20-28-24

LSSP: The Loop-based station sequencing problem

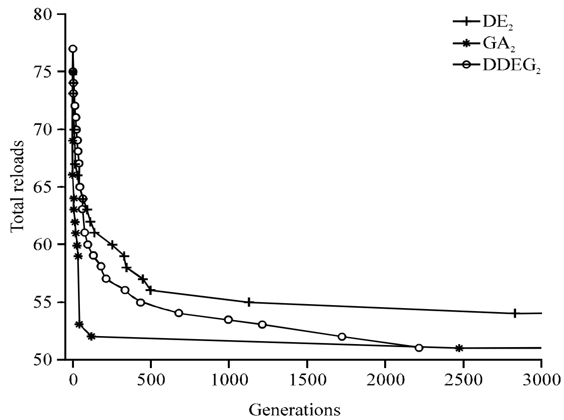

Fig. 6: Convergence curves of different algorithms with relative position encoding mechanism

probability (Cr), whereas F and Cr functioned together in the relative position-coded DE.

It can be easily seen from Table 2 that, though all six algorithms obtained the same objective values for small-scale problem sets, the dual-system framework achieved the highest performance outperforming both DE and GA alone for the largest-scale LSSP, that is because it combined two algorithms and employed the better one as the main system and strengthened the diversity by elite migration. It is worth mentioning that, DDEGA_1 achieved the optimal reload time 49, which was better than any other solutions in other literatures, more specifically, 53 acquired by the sub-ranges keys DE (Nearchou, 2006) and PSO (Kumar *et al*., 2008); 51 acquired by the relative position-coded DE (Zheng and Teng, 2010) and 50 acquired by the AIS (Kumar *et al*., 2009).

However, it can be easily noticed that the worst object value (54) found in the relative position-coded DE, which might implicate that the relative position encoding mechanism was not suitable for DE or needed modifying. Generally, algorithms with relative position encoding result to be weak as DE and GA are original.

Since all of the algorithms run for 3000 generations, the processing time in Table 2 represents the speed of algorithm. Obviously GA is the fastest and the dual-system is faster than DE.

CONCLUSION

This study focused on the Loop-based Station Sequencing Problem (LSSP) for flexible manufacturing systems. A dual-system framework which employs DE and GA was proposed. With two different encoding mechanisms, six kinds of algorithms were examined through experimental comparisons over different scales of test problems and the best layout for the largest scale LSSP was achieved by DDEGA with random keys encoding mechanism, which illustrated that the dual-system framework is superior to DE or GA alone.

Further studies could be conducted as follows: (1) The proposed dual-system framework should work better with adjusted opportune moment and the number of elite migration, (2) Since the inherent parallelism, the proposed framework can be easily implemented in parallel in order to be faster, (3) With improved DE and GA the relative position encoding mechanism may achieve better solutions since the dimension of the chromosome is one less than random keys encoding mechanism, (4) The proposed framework should be applied to many other problems even with constraints or multi-objectives.

ACKNOWLEDGMENTS

This study was supported by National Natural Science Foundation (Grant No. 50975039) and National Ministries Project (Grant No. A0920110001) of People's Republic of China.

REFERENCES

Adel El-Baz, M., 2004. A genetic algorithm for facility layout problems of different manufacturing environments. Comput. Ind. Eng., 47: 233-246.

Afentakis, P., 1989. A loop layout design problem for flexible manufacturing systems. Int. J. Flexible Manuf. Syst., 1: 175-196.

Arikaran, P., V. Jayabalan and R. Senthilkumar, 2010. Analysis of unequal areas facility layout problems. Int. J. Eng., 4: 44-51.

Bean, J.C., 1994. Genetics and random keys for sequencing and optimization. ORSA J. Comput., 6: 154-160.

Cheng, R. and M. Gen, 1998. Loop layout design problem in flexible manufacturing systems using genetic algorithms. Comput. Ind. Eng., 34: 53-61.

Das, S. and P.N. Suganthan, 2011. Differential evolution: A survey of the State-of-the-art. IEEE Trans. Evol. Comput., 15: 4-31.

Drira, A., H. Pierreval and S. Hajri-Gabouj, 2007. Facility layout problems: A survey. Ann. Rev. Control, 31: 255-267.

Kaku, B.K. and R. Rachamadugu, 1992. Layout design for flexible manufacturing systems. Eur. J. Operational Res., 57: 224-230.

Kumar, R.M.S., P. Asokan and S. Kumanan, 2008. Design of loop layout in flexible manufacturing system using non-traditional optimization technique. Int. J. Adv. Manuf. Technol., 38: 594-599.

Kumar, R.M.S., P. Asokan and S. Kumanan, 2009. Artificial immune system-based algorithm for the unidirectional loop layout problem in a flexible manufacturing system. Int. J. Adv. Manuf. Technol., 40: 553-565.

Kusiak, A. and S.S. Heragu, 1987. The facility layout problem. Eur. J. Operational Res., 29: 229-251.

Leung, J., 1992. A graph-theoretic heuristic for designing loop-layout manufacturing systems. Eur. J. Operational Res., 57: 243-252.

Nearchou, A.C., 2006. Meta-heuristics from nature for the loop layout design problem. Int. J. Prod. Econ., 101: 312-328.

Plaquin, M.F. and H. Pierreval, 2000. Cell formation using evolutionary algorithms with certain constraints. Int. J. Production Econ., 64: 267-278.

Pour, H.D. and M. Nosraty, 2006. Solving the facility and layout and location problem by ant-colony optimization-meta heuristic. Int. J. Prod. Res., 44: 5187-5196.

Singh, S.P. and R.R.K. Sharma, 2006. A review of different approaches to the facility layout problems. Int. J. Adv. Manuf. Technol., 30: 425-433.

Tansel, B.C. and C. Bilen, 1998. Move based heuristics for the unidirectional loop network layout problem. Eur. J. Operational Res., 108: 36-48.

Teng, H.F, Y. Chen, W. Zeng, S. Yan-jun and H. Qing-Hua, 2010. A dual-system variable-grain cooperative coevolutionary algorithm: Satellite-module layout design. IEEE Trans. Evol. Comput., 14: 438-455.

Tompkins, J.A., 2010. Facilities Planning. John Wiley and Sons, New York, ISBN: 0470444045, Pages: 854.

Whitley, D., S. Rana and R.B. Heckendorn, 1998. The island model genetic algorithm: On separability, population size and convergence. J. Comput. Inf. Technol., 7: 33-47.

Zheng, X.J. and H.F. Teng, 2010. A relative position-coded differential evolution for loop-based station sequencing problem. Int. J. Prod. Res., 49: 1235-1236.