Available online at www.sciencedirect.com

ScienceDirect

Expert Systems with Applications 36 (2009) 1259–1267

Expert Systems with Applications

www.elsevier.com/locate/eswa

A hybrid electromagnetism-like algorithm for single machine scheduling problem

Pei-Chann Chang [a,b,*], Shih-Hsin Chen [b], Chin-Yuan Fan [b]

[a] *Department of Information Management, Yuan Ze University, Taoyuan 32026, Taiwan, ROC*
[b] *Department of Industrial Engineering and Management, Yuan Ze University, Taoyuan 32026, Taiwan, ROC*

Abstract

Electromagnetism-like algorithm (EM) is a population-based meta-heuristic which has been proposed to solve continuous problems effectively. In this paper, we present a new meta-heuristic that applies the EM methodology to the single machine scheduling problem. To the best of our knowledge, there are only few researches in solving the combinatorial optimization problem (COP) by EM. This research attempts to employ the random-key concept combining with genetic operators in the hybrid algorithm to obtain the best/optimal schedule for the single machine problems. This new approach attempts to achieve the convergence and diversity effects when it is iteratively applied to solve the problem. This hybrid algorithm is tested on a set of standard test problems available in the literature. The computational results show that this hybrid algorithm performs better than the standard genetic algorithm.

Keywords: Single machine scheduling; Electromagnetism-like algorithm; Total deviations; Dominance property

1. Introduction

Single machine scheduling problem with the objective to minimize the total sum of earliness/tardiness is shown to be NP-hard in the literature. The results derived from the literature are very significant since they not only provide the insights into the single machine problem but also for more complicated environment (Pinedo, 2002).

In this study we apply the random-key approach to represent a schedule and incorporate the EM methodology to solve the single machine scheduling problem. In our algorithm, the EM procedures are modified to obtain better quality of solutions effectively. For example, the local search operator perturbs the best solution and generates a new solution. As long as the new one with a better solution than the worst one, we will replace the worst one with the new one. In addition, Debels, Reyck, Leus, and Vanhoucke (2006) proposed a new method in calculating the particle charge and exertion force. Both of them are adopted in the research. According to the experimental results, EM algorithm can provide good solution diversity because there are only few overlapped or redundant solutions. Consequently, a hybrid framework that integrates EM algorithm with GA is proposed to quickly converge the searching procedure by the selection and crossover operators.

The rest of the paper is organized as follows: Section 2 is the review of single machine problem; Section 3 is the definition of single machine problem; the methodology is described in Section 4. The experimental result is presented in Section 5, which compared the EM approach with genetic algorithms (GAs). Section 6 draws the discussion and conclusions.

2. Literature review

Recently, EM type algorithm has been used for optimization problems and the approach starts with a randomly

* Corresponding author. Address: Department of Information Management, Yuan Ze University, Taoyuan 32026, Taiwan, ROC. Tel.: +886 3 4638909; fax: +886 3 4638884.
E-mail address: iepchang@saturn.yzu.edu.tw (P.-C. Chang).

0957-4174/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2007.11.050

selected point from the feasible region for a given optimization problem. EM employs an attraction–repulsion mechanism to move points (particles) towards the optimal solution. Each point (particle) is treated as a solution and has a charge. A better solution contains a stronger charge. The charge of each point relates to the objective function to be optimized. EM method has been tested on available test problems in Birbil and Fang (2003). In this study, it is shown that EM is able to converge to the optimal solution in less number of function evaluations without any first or second order derivative information. A theoretical study of this EM analysis and a modification for convergence to the optimal solution are presented in Birbil et al. (2004). However, these above two studies only deal with continuous optimization problems.

EM type algorithms are used to solve fuzzy relation equations (Birbil & Feyzioglu, 2003), and to train artificial neural network for textile retail operations (Wu, Yang, & Wei, 2004), and also to obtain fuzzy if–then rules (Wu, Yang, & Hung, 2005). Debels et al. (2006) integrated a scatter search with EM for the solution of resource constraint project scheduling problems. This is the first paper that includes an EM type methodology for the combinatorial optimization problem. Their experimental results show that the hybrid method of incorporating EM type analysis outperforms the current best solution available in the literature.

Though EM algorithm is designed for solving optimization problems with bounded variables, the algorithm can be extended to solve combinatorial problem (COP). When we extend the EM algorithm to combinatorial optimization problems, the first important step is the representation of a solution. Bean (1994) introduced a random-key (RK) approach for real-coded GA for solving sequencing problem. Subsequently, numerous researchers show that this concept is robust and can be applied for the solution of different kinds of COPs (Mendes, Gonçalves, & Resende, 2005; Norman & Bean, 1999, 2000; Snyder & Daskin, 2006). Other applications of the random-key approach are in solving single machine scheduling problems and permutation flowshop problems using particle swarm optimization (PSO) algorithm by (Tasgetiren, Sevkli, Liang, & Gencyilmaz, 2004, 2007).

In this paper, the random-key approach to represent a schedule incorporated with the EM methodology are applied to solve a single machine scheduling problem and the objective is to minimize the total sum of earliness and tardiness penalties. A detailed formulation of the problem is described as follows: a set of n independent jobs $\{J_1, J_2, \ldots, J_n\}$ has to be scheduled without preemptions on a single machine that can handle at most one job at a time. The machine is assumed to be continuously available from time zero onwards and unforced machine idle time is not allowed. Job J_j, $j = 1, 2, \ldots, n$ becomes available for processing at the beginning, requires a processing time p_j and should be completed on its due date d_j. For any given schedule, the earliness and tardiness of J_j can be, respectively, defined as $E_j = \max\{0, d_j - C_j\}$ and $T_j = \max\{0, C_j - d_j\}$, where C_j is the completion time of J_j. The objective is then to find a schedule that minimizes the sum of the earliness and tardiness penalties of all jobs $\sum_{j=1}^{n}(\alpha_j E_j + \beta_j T_j)$, where α_j and β_j are the earliness and tardiness penalties of job J_j. The inclusion of both earliness and tardiness costs in the objective function is compatible with the philosophy of just-in-time production, which emphasizes producing goods only when they are needed. The early cost may represent the cost of completing a product early, the deterioration cost for a perishable goods or a holding (stock) cost for finished goods. The tardy cost can represent rush shipping costs, lost sales and loss of goodwill. It is assumed that no unforced machine idle time is allowed, so the machine is only idle if no job is currently available for processing.

Some specific examples of production settings with these characteristics are provided by Ow and Morton (1988), Azizoglu, Kondakci, and Krica (1991), Wu, Storer, and Chang (1993), Su and Chang (1998, 2001). The set of jobs is assumed to be ready for processing at the beginning which is a characteristic of the deterministic problem. As a generalization of weighted tardiness scheduling, the problem is strongly NP-hard in Lenstra, Rinnooy Kan, and Brucker (1977). To the best of our knowledge, the earlier work in this problem is due to Chang and Lee (1992a, 1992b), Wu et al. (1993), Chang (1999). Belouadah et al. (1992) dealt with the similar problem, however, with a different objective in minimizing the total weighted completion time and the problem is the same as discussed in Hariri and Potts (1983). Kim and Yano (1994) discussed some properties of the optimal solution, and these properties are used to develop both optimal and heuristic algorithms. Valente and Alves (2003a, 2003b) presented a branch-and-bound algorithm based on a decomposition of the problem into weighted earliness and weighted tardiness subproblems. Two lower bound procedures were presented for each subproblem, and the lower bound for the original problem is then simply the sum of the lower bounds for the two subproblems. In Valente and Alves (2003b), they analyse the performance of various heuristic procedures, including dispatch rules, a greedy procedure and a decision theory search heuristic.

The early/tardy problem with equal release dates and no idle time, however, has been considered by several authors, and both exact and heuristic approaches have been proposed. Among the exact approaches, branch-and-bound algorithms were presented by Abdul-Razaq and Potts (1988), Li (1997) and Liaw (1999). The lower bounding procedure of Abdul-Razaq and Potts (1988) was based on the subgradient optimization approach and the dynamic programming state-space relaxation technique, while Li and Liaw used Lagrangean relaxation and the multiplier adjustment method. Among the heuristics, Ow and Morton (1988) developed several dispatch rules and a filtered beam search procedure. Valente and Alves (2003b) presented an additional dispatch rule and a greedy procedure, and also considered the use of dominance rules to further improve the schedule obtained by the heuristics. A neighborhood search algorithm was also presented by Li (1997).

3. Introduction of an electromagnetism-like algorithm

EM simulates the attraction–repulsion mechanism of electromagnetism theory which is based on Coulomb's law. Each particle represents a solution and the charge of each particle relates to its solution quality. The better solution quality of the particle, the higher charge the particle has. Moreover, the electrostatic force between two point charges is directly proportional to the magnitudes of each charge and inversely proportional to the square of the distance between the charges.[1] The fixed charge of particle i is shown as follows:

$$q^i = \exp\left(-n\frac{f(x^i)-f(x^{\text{best}})}{\sum_{k=1}^{m}(f(x^k)-f(x^{\text{best}}))}\right), \quad \forall i. \tag{1}$$

where q^i is the charge of particle i, $f(x^i)$, $f(x^{\text{best}})$, and $f(x^k)$ denote the objective value of particle i, the best solution, and particle k. Finally, m is the population size.

The solution quality or charge of each particle determines the magnitude of an attraction and repulsion effect in the population. A better solution encourages other particles to converge to attractive valleys while a bad solution discourages particles to move toward this region. These particles move along with the total force and so diversified solutions are generated. The following formulation is the force of particle i.

$$F^i = \sum_{j\neq i}^{m}\left\{\begin{array}{lll}(x^j-x^i)\frac{q^iq^j}{\|x^j-x^i\|^2} & \text{if} & f(x^j)<f(x^i)\\ (x^i-x^j)\frac{q^iq^j}{\|x^j-x^i\|^2} & \text{else} & f(x^j)\geqslant f(x^i)\end{array}\right\}, \quad \forall i. \tag{2}$$

Take the following figure for example. There are three particles and their own objective values are 20, 15, and 10, respectively. Because particle 1 is worse than particle 3 while particle 2 is better than particle 3, particle 1 represents a repulsion force which is the F_{13} and particle 2 encourages particle 3 that moves to the neighborhood region of particle 2. Consequently, particle 3 moves along with the total force F (see Fig. 1).

The fundamental procedures of EM include initialize, local search, calculating total force, and moving particles. The generic pseudo-code for the EM is as follows:

[1] http://en.wikipedia.org/wiki/Coulomb's_law.

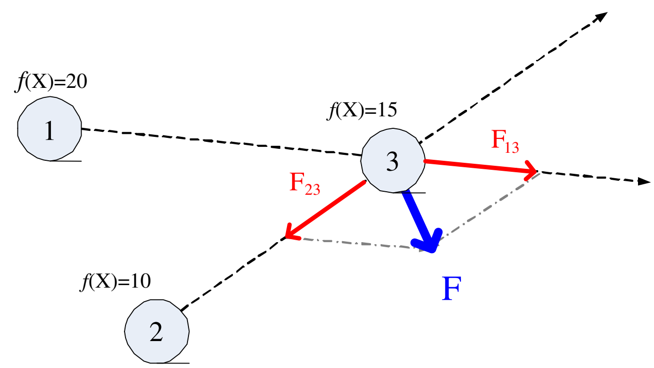

Fig. 1. An example of attract–repulse effect on particle number 3.

Algorithm 1. EM()

1. initialize()
2. **while** (hasn't met stop criterion) **do**
3. localSearch()
4. calculate total force $F()$
5. move particle by $F()$
6. evaluate particles()

End while

4. Methodology

This paper proposes a hybrid framework that combines EM-like algorithm and genetic operator for solving scheduling problems. The fundamental method is the random-key technique that enables EM to solve this kind of problems. Because the time-complexity is high and to obtain better solution quality for EM-like meta-heuristic with RK approach, some procedures like local search, particle charge, and electrostatic force are modified. The purpose of this hybrid framework is to take the advantage of EM, which yields a high diversity population, and GA operator let the algorithm converge faster. Since the random-key technique is a fundamental method in this paper, it is introduced in the beginning and the later sections describe the detail approaches of the hybrid framework and modified EM procedures.

4.1. A random-key method

In order to enable EM to solve scheduling problems, the random-key technique is introduced. The concept of RK technique is simple and can be applied easily. When we obtain a k-dimension solution, we sort the value corresponding to each dimension. Any sorting algorithm can be used in the method and the paper uses quick sort because its time-complexity is O ($nlogn$). After having a sequence, we can use it to compute the objective function value of this sequence.

Fig. 2 demonstrates a 10-dimension solution. Values of dimension 1, 2 and 3 are 0.5, 9.6 and 3.0. The rest

Activities

	1	2	3	4	5	6	7	8	9	10
Before	0.5	9.6	3.0	2.9	2.2	8.0	4.2	0.1	7.1	5.6

(a) Value of Activities

After	8	1	5	4	3	7	10	9	6	2

(b) Schedule List

Fig. 2. An example of attract–repulse effect on particle number 3.

are shown in Fig. 2. Then, we apply the random-key method to sort these values in ascending order. Thus sequence at position 1 is 8 that mean we schedule job 8 in the beginning and job 2 is scheduled at the last position. By the random-key method, the continuous EM algorithm is able to be applied to solve different kind of sequencing problems.

4.2. A hybrid framework combines the modified EM and genetic operators

The hybrid framework includes modified EM procedures and genetic operators, which adopts selection and mating. The selection operator is binary tournament and uniform crossover operator is applied in the framework. Generic EM provides an excellent diversity while GA is able to converge to a better solution quickly. Thus the hybrid method takes the advantage of both sides.

The hybrid system starts from determining which particle is moved by EM or mated by GA crossover operator. In the paper by Debels et al. (2006), they suggested that a new solution can be obtained from crossing by a better solution selected from a binary tournament method. And EM is used to move the inferior solution to a new position. This hybrid approach may encourage solutions converging toward better region quickly and to prevent from trapping into the local optimal and still maintaining the population diversity. The Algorithm 2 is the pseudo-code of the main procedures of the hybrid framework.

Algorithm 2. A Hybrid Algorithm

1. initialize()
2. **while** (hasn't met stop criterion) **do**
3. localSearch()
4. $avg \leftarrow$ calcAvgObjectiveValues()
5. **for** $i = 1$ to m **do**
6. **if** $i \neq$ best and $f(x^i) < avg$ **then**
7. $j \leftarrow$ a selected particle to mate particle i by binary tournament()
8. uniformCrossover(x^i,x^j)
9. **else if** $f(x^i) > avg$ **then**
10. CalcF and Move(x^i)
11. **end if**
12. **end for**
13. find sequence by random-key method()
14. evaluate particles()
15. **end while**

According to the Algorithm 2 (Algorithm 2, line 1), we initiate the particles in the population. Then, the local search procedure is implemented before the EM procedures and genetic operators. To determine which solution is good or inferior one, an average objective value *avg* is calculated. Then, if the solution is better than *avg*, this solution is mated by the other better solution obtained by binary tournament (Algorithm 2, line 7 and 8). Otherwise, this solution is moved by modified EM algorithm (Algorithm 2, line 10). After these particles are mated or move along with their own total force, the next step is to generate corresponding sequences by random-key technique. As soon as the sequence is obtained, we can obtain objective value of the solution. Finally, because the initialization, local search, particle charge, calculated total force, and move are modified, we discuss them in the following sections.

4.3. Initialization

The Algorithm 3 initiates particles in the population. The initial value is between lower bound and upper bound. The lower bound and upper bound are set between [−1 1]. After all particles are generated, the RK method is used to generate sequence of the corresponding values of each particle. As soon as we obtain the sequence, the objective values of these particles are evaluated and we can obtain current best solution from these solutions (Algorithm 3, line 8 and 9).

Algorithm 3. Initialize()

1: **for** $i = 1$ to m **do**
2: **for** $k = 1$ to n **do**
3: $\lambda \leftarrow U(0, 1)$
4: $x_k^i \leftarrow l_k + \lambda(u_k - l_k)$
5: **end for**
6: **end for**
7: find sequence by random-key method
8: evaluate particles ()
9: $x^{best} \leftarrow \arg\min\{f(x^i), \forall i\}$

4.4. Local procedure

The algorithm that perturbs each dimension of the best solution (Algorithm 4, line 5–12) then finds its corresponding sequence and their objective value. This new solution will replace the worst solution when its objective value is better than the worst solution (Algorithm 4, line 12–15).

Therefore, it attempts to improve average solution quality iteratively for *LSITER* times. This procedure may find a better solution to substitute the current best solution (Algorithm 4, line 16–20).

Algorithm 4. Local (*LSITER*)

1: Length $\leftarrow$ argmax$\{u_k - l_k, \forall_k\}$
2: $i \leftarrow$ argmin$\{f(x^i), \forall i\}$
3: **for** $j = 1$ to *LSITER* **do**
4: $y \leftarrow x^i$
5: **for** $k = 1$ to n **do**
6: $\lambda \leftarrow U(0, 1)$
7: **if** $U(0, 1) > 0.5$ **then**
8: $y_k \leftarrow y_k + \lambda$ (Length)
9: **else**
10: $y_k \leftarrow y_k - \lambda$ (Length)
11: **end if**
12: **end for**
12: $l =$ argmax$\{f(x^i), \forall i\}$
13: **if** $f(y) < f(x^l)$ **then**
14: $x^l \leftarrow y$
15: **end if**
16: **if** $f(y) < f(x^i)$ **then**
17: $x^{\text{best}} \leftarrow y$
18: $x^i \leftarrow y$
19: $i \leftarrow l$
20: **end if**
21: **end for**

4.5. Particle charges, electrostatic force and move

The study uses the total force algorithm proposed by Debels et al. (2006), which determines the force exerted on particle i by point j that does not use the fixed charge of q_{i} and q_{j}. Instead, q_{ij} depends on the relative deviation of $f(x^i)$ and $f(x^j)$. Thus this particle charge is calculated as follows:

$$q_{\text{ij}} = \frac{f(x^i) - f(x^j)}{f(x^{\text{worst}}) - f(x^{\text{best}})} \tag{3}$$

If the objective value $f(x^i)$ is larger than $f(x^j)$, particle j will attract particle i. On the other hand, when $f(x^i) < f(x^j)$, a repulsion effect is occurred. There is no action when $f(x^i) = f(x^j)$ because q_{ij} is equal to zero. After the q_{ij} is obtained, the force on particle i by particle j is

$$F^{ij} = (x^j - x^i) \cdot q_{\text{ij}} \tag{4}$$

Thus the particle x^i moves to $x^i + F^{ij}$ in the direction of particle x^j. This method is similar to the path relinking method by Glover, Laguna, and Marti (2000) which gradually moves from one point to another by Debels et al. (2006).

Table 1
The parameter settings of the EM algorithm

Factor	Treatments
Population size (popSize)	50 and 100
Number of local search (LS)	10 and 25
Methods	1. Modified EM algorithm 2. Hybrid model (modified EM algorithm and genetic operators)
Job instance (size)	20, 30, 40, 50
Number of examined solutions	100,000
Number of replications	30

Algorithm 5. CalcF and Move(x^i)

$F^i \leftarrow 0$
for $j = 1$ to m **do**
if $x^i \neq x^j$ **then**
$q_{\text{ij}} \leftarrow \frac{f(x^i) - f(x^j)}{f(x^{\text{worst}}) - f(x^{\text{best}})}$
$F^{ij} \leftarrow (x^j - x^i)q_{ij}$
$x^i \leftarrow x^i + F^{ij}$
end if
end for

Finally, in order to maintain the feasibility of each solution, we check the boundary feasibility by the Algorithm 6.

Algorithm 6. checkBoundary()

for $i = 1$ to m **do**
for $j = 1$ to n **do**
if $x^{ij} > u_j$ **then**
$x^{ij} \leftarrow u_j$
else if $x^{ij} < l_j$ **then**
$x^{ij} \leftarrow l_j$
end if
end for
end for

5. Experimental results

The study proposed a hybrid framework that combines modified EM meta-heuristic and genetic operator in solving the single machine problem in minimizing the earliness and tardiness penalty. In order to evaluate the performance of this hybrid framework, it is compared with GA which is a well known meta-heuristic. Across these experiments, we adopt the scheduling instances of Sourd and Sidhoum (2005) whose job size are 20, 30, 40, and 50.[2] Each experiment is replicated 30 times and the stopping criterion

[2] The name of each instance for 20, 30, 40, and 50 jobs are sks222a, sks322a, sks422a, and sks522a, respectively.

is to fix the number of examined solutions that is set to 100,000.

Before we validate these methods and to compare the performance between the proposed algorithm and GA, a design of experiment (DOE) is carried out to examine the parameter settings of the hybrid framework. The DOE result of it is shown in Section 5.1. Then, we compare the performance of the hybrid framework with GA under the job-dependent due date. It is presented in Section 5.2.

5.1. Design of experiment for EM in single machine scheduling problems

There are two parameters that should be tuned in EM algorithm. In continuous EM, Birbil and Fang (2003) suggested a population size that is four times the dimensions. However, since there is no result for this problem, this experiment fills up the gap which identifies the appropriate population size. Secondly, the local search method is mod-

Table 2
The ANOVA result of parameter configuration

Source	DF	Seq SS	Adj SS	Adj MS	*F*	*P*
Size	3	1.11E+11	1.11E+11	3.7E+10	203188.7	0.000
popSize	1	9,147,142	9,298,221	9,298,221	51.09	0.000
LS	1	6,962,894	7,063,195	7,063,195	38.81	0.000
Methods	1	1.34E+08	1.36E+08	1.36E+08	745.04	0.000
Size * popSize	3	8,514,003	8,514,003	2,838,001	15.59	0.000
Size * LS	3	3,596,976	3,596,976	1,198,992	6.59	0.000
Size * Methods	3	1.02E+08	1.02E+08	34,062,150	187.16	0.000
popSize * LS	1	730,291	741,170	741,170	4.07	0.044
popSize * Methods	1	17,670,972	17,953,144	17,953,144	98.65	0.000
LS * Methods	1	7,113,306	7,225,338	7,225,338	39.7	0.000
Size * popSize * LS	3	592,779	592,779	197,593	1.09	0.354
Size * popSize * Methods	3	15,263,422	15,263,422	5,087,807	27.96	0.000
Size * LS * Methods	3	9,831,152	9,831,152	3,277,051	18.01	0.000
popSize * LS * Methods	1	777,051	783,908	783,908	4.31	0.038
Size * popSize * LS * Methods	3	1,454,635	1,454,635	484,878	2.66	0.047
Error	936	1.7E+08	1.7E+08	181,995		
Total	967	1.11E+11				

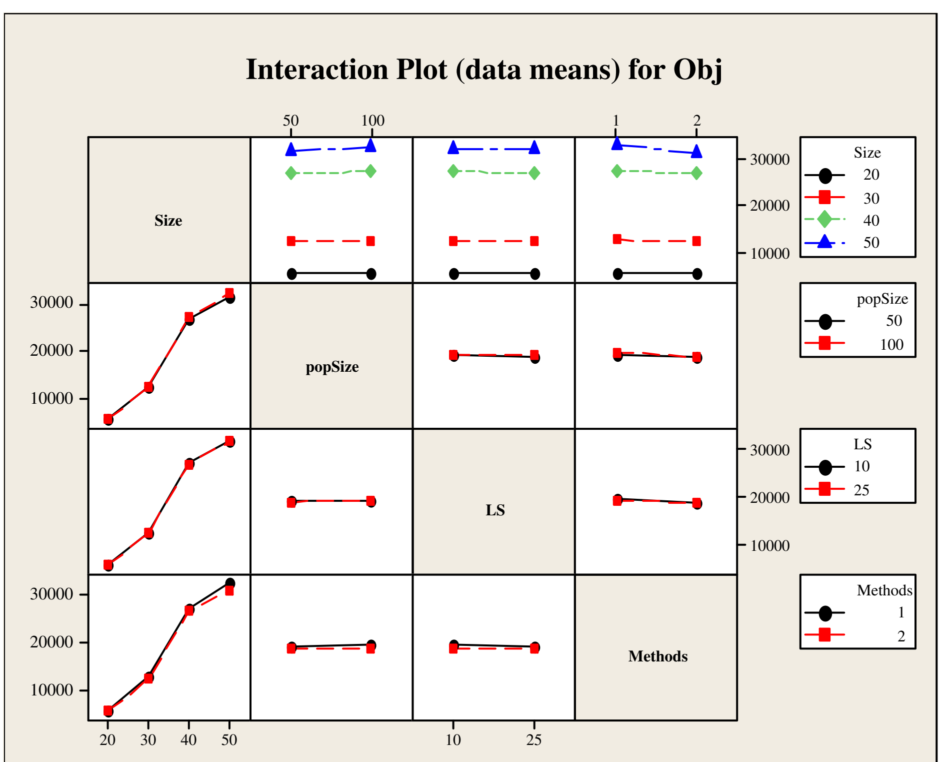

Fig. 3. The interaction plot of the parameter configuration experiment.

Table 3
The parameter settings of the hybrid algorithm

Factor	Treatments
Population size (popSize)	50
Number of local search (LS)	25
Methods	Hybrid model (modified EM algorithm and genetic operators)

ified and the number of local search is unknown. Thus the number of local search is considered in the DOE experiment.

Except for the parameter setting of EM algorithm, the study includes the comparison of the performance of hybrid model and the modified EM algorithm that works alone. The parameter setting is shown in Table 1 and the AVNOVA result is in Table 2.

If the confidence level α is set to 0.05, the main effect factors Size, popSize, LS, and method are significant. Furthermore, their two factors and three factors interaction effects are also significant. There is only one exception that is the combination of Size, popSize, and LS. Consequently, we analyze their interaction effect first by Fig. 3.

The interaction plot shows when the population size is set to 50 and the number of local search is assigned 25, it gets better solution quality. Most important of all, the hybrid model is better than the modified EM that works alone. Thus it encourages us to adopt the hybrid framework compared with GA.

Based on the initial parameter settings, a second time DOE is carried out. Because these factors don't cause any statistics significance, the paper adopts the configuration from our first DOE experiment. The final parameter setting of this hybrid framework is shown in Table 3.

5.2. The comparison between hybrid framework and GAs

We consider the scheduling problem under the job-dependent due date without learning consideration first. The proposed hybrid framework is compared with genetic algorithm. The parameter of GA includes crossover rate, mutation rate, and population size, which are set to 0.8, 0.3, and 100, respectively. The above GA parameter settings and experimental result of GA are adopted from our previous research in Chang, Chen, and Fan (2008). The comparison results are presented in Table 4 and the hybrid framework outperforms GA in average across all instances, except for 60 job problem.

In order to compare the performance of the two algorithms, ANOVA is applied to test if there is any difference between these two algorithms. The result of ANOVA is shown in Table 5 and it shows there is significant difference between these two algorithms. As shown in Table 6, the Duncan test also presents the SGA and Hybrid Algorithm that are in different groups. Therefore, the proposed algorithm outperforms the SGA. Finally, the the box-plot of these two algorithms is shown in Fig. 4. It indicates the

Table 6
The comparison between the hybrid algorithm and GA

Duncan grouping	Mean	N	Algorithm
A	18575.3	120	SGA
B	18462.07	120	Hybrid algorithm

Table 4
The comparison between the hybrid algorithm and GA

Job	GA				Hybrid algorithm			
	Min	Mean	Max	Secs	Min	Mean	Max	Secs
20	5286	5401.7	5643	1.0573	5287	5331.8	5464	1.9542
30	11,623	12,066	12,916	1.6838	11,584	11,794	12,223	2.8208
40	25,656	26,211	27,462	2.4548	25,706	25,933	26,294	3.3386
50	29,485	30,623	32,340	3.5406	29,490	29,902	30,447	4.1182
60	43,930	45,018	46,017	3.6405	43,873	45,138	47,820	12.07
90	91,516	93,966	96,966	7.3611	91,133	93,905	99,297	17.535

Table 5
The comparison between the hybrid algorithm and GA

Source	DF	Seq SS	Adj SS	Adj MS	F	P
Algorithm	1	769,307	769,307	769,307	4.96	0.027
Size	3	2.48E+10	2.48E+10	8.28E+09	53332.74	0
Algorithm * Size	3	2,238,570	2,238,570	746,190	4.81	0.003
Error	232	36,000,870	36,000,870	155,176		
Total	239	2.49E+10				

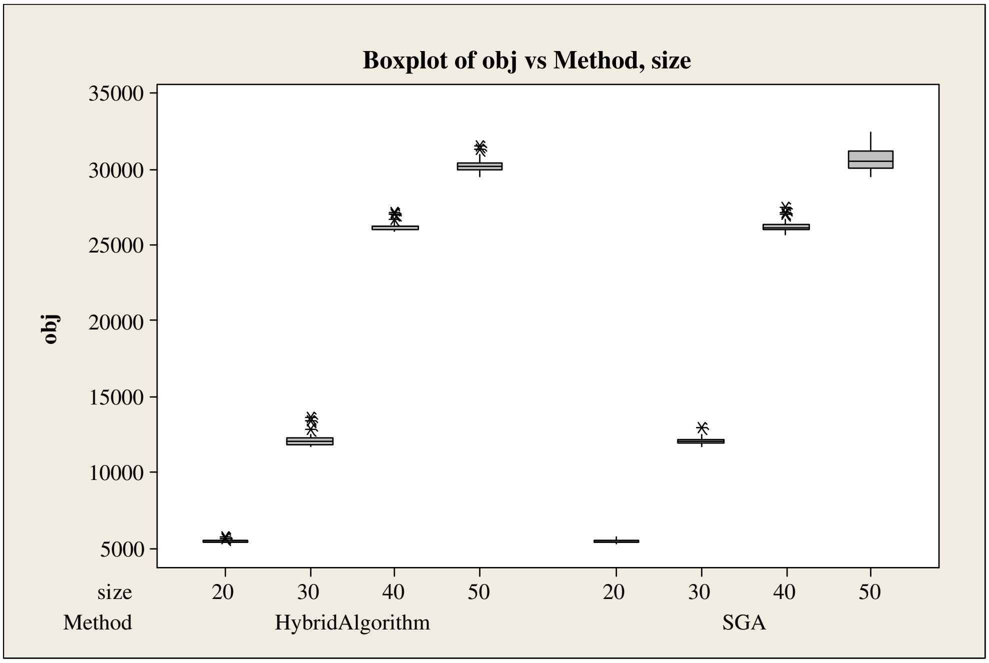

Fig. 4. The box-plot of the two algorithms.

range of the Hybrid algorithm is less than SGA. However, both algorithms have outlier values.

6. Discussion and conclusions

Owing to the development of a random-key method, the EM algorithm is able to be applied in solving the sequencing problem. To improve the performance of the EM algorithm, a hybrid method is developed in this research which combines EM algorithm and genetic operators together. The purpose of this hybrid method is to take advantage of the EM algorithm and genetic operators applied in genetic algorithms. The hybrid method can provide better solution diversity and good convergence ability during the evolutionary procedure, respectively. A DOE experiment shows the performance of hybrid method is better than that of the EM algorithm alone.

According to experimental results, the hybrid method outperforms SGA in most of the instances. However, since random-key technique has to sort out each final solution in order to generate a feasible sequence, it takes O (*nlogn*) time-complexity in the computational times while GA is able to provide a sequence representation through the chromosome directly. As a result, the computational times of the hybrid method is higher than that of GA. For future research, a better local search such as variable neighborhood search (VNS) can be applied in EM which may further improve the solution quality. Furthermore, since EM can be extended to multi-objective algorithm, it is an entirely new research area to be explored.

Acknowledgement

Thanks Dr. Yaochu Jin and Prof. Shaw S. Wang gave us some suggestions for this paper.

References

Abdul-Razaq, T., & Potts, C. N. (1988). Dynamic programming state-space relaxation for single machine scheduling. *Journal of the Operational Research Society, 39*, 141–152.

Azizoglu, M., Kondakci, S., & Krica, Omer (1991). Bicriteria scheduling problem involving total tardiness and total earliness penalties. *International Journal of Production Economics, 23*, 17–24.

Bean, J. C. (1994). Genetic algorithms and random keys for sequencing and optimization. *ORSA Journal on Computing, 6*(2), 154–160.

Belouadah, H., Posner, M. E., & Potts, C. N. (1992). Scheduling with release dates on a single machine to minimize total weighted completion time. *Discrete Applied Mathematics, 36*, 213–231.

Birbil, S. I., & Fang, S. C. (2003). An Electromagnetism-like Mechanism for Global Optimization. *Journal of Global Optimization, 25*, 263–282.

Birbil, S. I., & Feyzioglu, O. (2003). A global optimization method for solving fuzzy relation equations. *Lecture Notes in Artificial Intelligence, 2715*, 718–724.

Birbil, S.I., Gükan, G., & Listes, O. (2004) *Solving Stochastic Mathematical Programs with Complementarity Constraints Using Simulation*. Winter Simulation Conference 2004, pp. 550-558.

Chang, P. C. (1999). A Branch and bound approach for single machine scheduling with earliness and tardiness penalties. *Computers and Mathematics with Applications, 37*, 133–144.

Chang, P. C., & Lee, H. C. (1992a). A greedy heuristic for Bi-criterion single machine scheduling problems. *Computers and Industrial Engineering, 22*(2), 121–131.

Chang, P. C., & Lee, H. C. (1992b). A two phase approach for single machine scheduling: Minimizing the total absolute deviation. *Journal of the Chinese Institute of Engineers, 15*(6), 735–742.

Chang, P. C., Chen, S.-H., & Fan, C.-Y. (2008). Mining gene structures to inject artificial chromosomes for genetic algorithm in single machine scheduling problems. *Applied Soft Computing Journal, 8*(1), 767–777.

Debels, D., Reyck, B. D., Leus, R., & Vanhoucke, M. (2006). A hybrid scatter search/electromagnetism meta-heuristic for project scheduling. *European Journal of Operational Research, 169*, 638–653.

Glover, F., Laguna, M., & Marti, R. (2000). Fundamentals of scatter search and path relinking. *Control and Cybernetics, 39*, 653–684.

Hariri, A. M. A., & Potts, C. N. (1983). Scheduling with release dates on a single machine to minimize total weighted completion time. *Discrete Applied Mathematics, 36*, 99–109.

Kim, Y. D., & Yano, C. A. (1994). Minimizing mean tardiness and earliness in single-machine scheduling problems with unequal due dates. *Naval Research logistics, 41*, 913–933.

Lenstra, J. K., Rinnooy Kan, A. H. G., & Brucker, P. (1977). Complexity of machine scheduling problems. *Annals of Discrete Mathematics, 1*, 343–362.

Li, G. (1997). Single machine earliness and tardiness scheduling. *European Journal of Operational Research, 96*, 546–558.

Liaw, C. F. (1999). A branch and bound algorithm for the single machine earliness and tardiness scheduling problem. *Computers and Operations Research, 26*, 679–693.

Mendes, J. J. M., Gonçalves, J. F., & Resende, M. G. C. (2005). A random key based genetic algorithm for the resource constrained project scheduling problem. AT&T Labs Research Technical Report TD-6DUK2C.

Norman, B. A., & Bean, J. C. (1999). A genetic algorithm methodology for complex scheduling problems. *Naval Research Logistics, 46*(2), 199–211.

Norman, B. A., & Bean, J. C. (2000). Scheduling operations on parallel machine tools. *IIE Transactions, 32*(5), 449–459.

Ow, P. S., & Morton, T. E. (1988). Filtered beam searches in scheduling. *International Journal of Production Research, 26*, 35–62.

Pinedo, M. (2002). *Scheduling: Theory, algorithms, and systems*. Upper Saddle River, NJ: Prentice Hall.

Snyder, L. V., & Daskin, M. S. (2006). A random-key genetic algorithm for the generalized traveling salesman problem. *European Journal of Operational Research, 174*(1), 38–53.

Sourd, F., & Sidhoum, S. K. (2005). An efficient algorithm for the earliness/tardiness scheduling problem. Working Paper – LIP6 (http://www-poleia.lip6.fr/sourd/).

Su, L. H., & Chang, P. C. (1998). A heuristic to minimize a quadratic function of job lateness on a single machine. *International Journal of Production Economics, 55*(2), 169–175.

Su, L. H., & Chang, P. C. (2001). Scheduling *n* jobs on one machine to minimize the maximum lateness with a minimum number of tardy jobs. *Computers and Industrial Engineering, 40*(4), 349–360.

Tasgetiren, M. F., Sevkli, M., Liang, Y. C., & Gencyilmaz, G. (2004). Particle swarm optimization algorithm for single machine total weighted tardiness problem. In *Proceedings of the 2004 Congress on Evolutionary Computation (CEC'04)* (pp. 1412–1419), Portland, Oregon.

Tasgetiren, M. F., Sevkli, M., Liang, Y. C., & Gencyilmaz, G. (2007). Particle swarm optimization algorithm for makespan and total flowtime minimization in permutation flowshop sequencing problem. *European Journal of Operational Research, 177*(3), 1930–1947.

Valente, J. M. S., & Alves, R. A. F. S. (2003a). Heuristics for the early/tardy scheduling problem with release dates. Working Paper 129. Faculdade de Economia do Porto, Portugal.

Valente, J. M. S., & Alves, R. A. F. S. (2003b). Improved heuristics for the early/tardy scheduling problem with no idle time. Working Paper 126. Faculdade de Economia do Porto, Portugal.

Wu, P., Yang, W. H., & Wei, N. C. (2004). An electromagnetism algorithm of neural network analysis – An application to textile retail operation. *Journal of the Chinese Institute of Industrial Engineers, 21*(1), 59–67.

Wu, P., Yang, K. J., & Hung, Y. Y. (2005). The study of electromagnetism-like mechanism based fuzzy neural network for learning fuzzy if–then rules. *Lecture Notes in Computer Science, 3684*, 382–388.

Wu, S. D., Storer, R. H., & Chang, P. C. (1993). One machine rescheduling heuristic with efficiency and stability as criteria. *Computers and Operations Research, 20*(1), 1–14.