Arab J Sci Eng (2017) 42:3609–3620
DOI 10.1007/s13369-017-2627-7

RESEARCH ARTICLE - COMPUTER ENGINEERING AND COMPUTER SCIENCE

A New Discrete Electromagnetism-Like Mechanism Algorithm for Identical Parallel Machine Scheduling Problem with Eligibility Constraints in Metal Nuts Manufacturing

Cheng-Hsiung Lee[1]

Received: 15 August 2016 / Accepted: 7 June 2017 / Published online: 15 June 2017

Abstract This paper presents a real-life scheduling problem of minimizing total weighted tardiness on identical parallel machines with eligibility constraints which is originated from the manufacturing plant of an industrial metal nuts company. Because the problem is NP-hard, a new electromagnetism-like mechanism algorithm is proposed to solve the problem. In the proposed algorithm, the particle is redesigned to represent jobs with valid assignment to machines. A distance measure between particles is proposed by the concept of a number-guessing game. Then, the new attraction and repulsion operators are developed to move a particle to the new particle. The computational results show that the proposed algorithm performs better than the current scheduling method of the metal nut plant and other existing algorithms.

Keywords Scheduling · Parallel machines · Electromagnetism-like mechanism · Eligibility constraints · Metaheuristic

1 Introduction

In this paper we address a real-life scheduling problem originated from the manufacturing plant of an industrial metal nuts company. The metal nut plant receives purchase orders from different clients. The metal nut products may have different specification, specified precision of screw thread and specified due date in each purchase order. Because the demand from clients is very important and changeable, the metal nut production is a typical make-to-order environment and highlights the satisfaction of clients' requirements. For all purchase orders, the metal nut products with the same due date and the same precision are grouped into a single job, which is the basic unit on the scheduling operation. In the metal nut plant, there are three machines which produce the metal nut products based on specified precision of screw threads. One is high-precision machines, and the others are common-precision machines. There are eligibility constraints (hereafter referred to as EC) in such a manufacturing system. Some clients request their purchase order having the quality with high precision. Only the machine with high precision can process this kind of high-precision jobs. The common-precision jobs are allowed to be processed on any machines. That is, high-precision machine is capable of processing all jobs, but common-precision machine cannot process high-precision jobs. The processing time of a job is identical wherever the job is processed on high- or common-precision machine. Therefore, the machine environment can be defined as the identical parallel machines. The processing time of each job normally takes 30–80 h based on the size of the purchase order. Making a setup adjustment is necessary whenever there is a switch from processing a job to another job on a machine; thus, a setup time must be incurred. However, since a setup time only takes about 1–3 min, which is insignificant compared to the processing time, and thus the setup time can be ignored in this research.

The due date is the most important requirement; hence, the clients always pay more attention to the due dates. Tardiness will be incurred when a job is completed later than its due date. If the tardiness is incurred, the metal nut plant has to spend additional time negotiating with the client in order to postpone the due date of the job. Generally, the client might accept a slight change to the due date, but if the completion time of a job exceeds its due date by too long, a penalty cost

✉ Cheng-Hsiung Lee
brucelee@mail.chihlee.edu.tw

1 Department of International Trade, Chihlee University of Technology, No. 313, Section 1, Wunhua Road, Banciao District, New Taipei City 22050, Taiwan

Fig. 1 A variety of industrial metal nut products

associated with the tardiness is usually incurred. The usual situation is that the client accepts the delay of delivery time, but asks to get a discount on the unit price of the tardy job, or cuts the payment directly. The worst situation is that the client cancels the purchase order and turns to other competitors who can meet the requested due date. These situations lead to a significant loss of revenue for the metal nut plant. The metal nut plant must keep the control over tardiness strictly in order to reduce extra penalty costs and opportunity losses. Such penalty costs and opportunity losses can be regarded as the weights associated with tardiness. Therefore, the objective of minimizing total weighted tardiness has been one of the most important tasks for the metal nut plant. The problem is to assign different jobs to eligible machines and the sequence of the jobs allocated to each machine for the objective of minimizing total weighted tardiness under eligibility constraints. Following the well-known three-field notation, the considered problem can be denoted by $P_{\mathrm{m}}|\mathrm{EC}|\sum w_j T_j$, which is strongly NP-hard. Since the concept of eligibility constraints (EC) is quite similar to that of grade of service (hereafter referred to as GoS) which is common in service industry, this research may also use the concept of GoS to explain the EC for the considered scheduling problem in the metal nut plant (Fig. 1).

In this research, a new discrete electromagnetism-like mechanism algorithm is proposed to solve the studied problem. The main contributions of the proposed discrete electromagnetism-like mechanism algorithm are that a novel idea of the discrete distance metric is presented, called "1A2B" distance, by the concept of a number-guessing game. Moreover, a new attraction–repulsion operator, inspired by the crossover operator of genetic algorithm, is introduced to obtain new particle in next iteration. To sum up, the proposed discrete electromagnetism-like mechanism algorithm has a conceptually easy design and is suitable for solving $P_{\mathrm{m}}|\mathrm{EC}|\sum w_j T_j$, especially in large-sized problems.

2 Literature review

Related works about eligibility constraints (EC) and grade of service (GoS) in the literature are briefly reviewed in this section. A GoS provision has received attention in a parallel machine scheduling problem, especially in service industry [1]. For example, the antivirus software company provides an online antivirus scan and divides customers into two GoS levels: free-trial (low) and regular (high) members. In the company, there are a number of servers (machines), each of which be specified by its GoS level. To ensure a regular member better service, the processing capability of the server labeled with a regular level tends to be reserved for regular-level service requests. Thus, the service request from a customer is only allowed to be processed on a particular server when the GoS level of the request is no less than the GoS level of the server. The practice situation is classified as a parallel machine scheduling problem with GoS eligibility [1]. Lu and Liu [2] deal with the semi-online scheduling on two uniform machines under a GoS provision to minimize the makespan. In their problem, one machine is available for all jobs and the other one is only available for partial jobs. Gokhale and Mathirajan [3] address a real-life scheduling problem in automobile gear manufacturing for minimizing total weighted flowtime. They take unequal release times, sequence-dependent setup times, and machine eligibility restrictions into account for the problem and propose several heuristic algorithms to solve the problem. The computational analyses show that the proposed heuristic algorithms can consistently yield near-statistically estimated optimal solutions in a reasonable computational time. Chuang et al. [4] consider a parallel machine scheduling problem in the manufacturing of anodic electro-etching aluminum foil. The problem is to schedule jobs on the high- and medium-voltage equipment, each having several pieces in parallel, with setup times to minimize the total completion time under machine eligibility constraints. They propose a three-stage heuristic for this problem and computationally evaluate the performance of the heuristic relative to the solution obtained using a branch-and-bound algorithm. Other applications for EC and GoS arise in processor scheduling with memory constraints [5] and cargo handling by cranes with weight capacities [6].

Most studies on EC and GoS focus on parallel machine scheduling with makespan objective which is denoted by $P_{\mathrm{m}}|\mathrm{EC\ or\ GoS}|C_{\max}$. Hwang et al. [1] initiate the studies in this area and propose a lowest grade–longest processing times first (LG-LPT) heuristic for the problem. Later, Glass and Kellerer [5] and Ou et al. [6] develop the polynomial-time approximation schemes (PTAS). Ji and Cheng [7] further propose a fully polynomial-time approximation scheme (FPTAS), which greatly improves the bound in [1]. Woeginger [8] derives two FPTASs simplifying the FPTAS of [7].

This research considers the problem of minimizing total weighted tardiness on identical parallel machines with eligibility constraints, denoted by $P_{\mathrm{m}}|\mathrm{EC}|\sum w_j T_j$. To the best of our knowledge, there exists no study directly addressing the total weighted tardiness criterion, which is always regarded as one of the important criteria in real-life applications [9–11]. $P_{\mathrm{m}}|\mathrm{EC}|\sum w_j T_j$ is considered somewhere in between the parallel machine problem $P_{\mathrm{m}} \parallel \sum w_j T_j$ and the unrelated machine problem $R_{\mathrm{m}} \parallel \sum w_j T_j$, which are strongly NP-hard problems. Lin et al. [12] present two robust heuristics and a genetic algorithm (GA) for $R_{\mathrm{m}} \sum w_j T_j$. The results show that the proposed GA outperforms all existing heuristics. Liaw et al. [13] first develop a brand-and-bound algorithm for $R_{\mathrm{m}} \parallel \sum w_j T_j$. The results show that the brand-and-bound algorithm performs well on problems with up to 18 jobs and 4 machines. Raghavan and Venkataramana [9] propose an ant colony optimization (ACO) approach for $P_{\mathrm{m}} \parallel \sum w_j T_j$ and can make improvement over some available heuristics.

Electromagnetism-like mechanism (hereafter referred to as EM), inspired by the attraction and repulsion mechanisms of the electromagnetism theory, is an emerging population-based metaheuristic. EM is proposed by Birbil and Fang [14] for optimization of continuous nonlinear functions. EM has been successfully applied to many continuous and discrete optimization problems [14–20], while the studies of EM on scheduling problems are relatively few. Yan et al. [21] present a two-stage assembly flow shop scheduling problem for minimizing the weighted sum of maximum makespan, earliness, and lateness. Debels et al. [18] first present an EM-based metaheuristic to solve the resource-constrained project scheduling problem. Chang et al. [17] propose a discrete EM algorithm for the single machine scheduling problem with the objective of minimizing total earliness and tardiness. Gao et al. [22] propose a discrete EM (DEM) algorithm to solve the assembly sequence planning problem which is a combinatorial optimization problem with strong constraints aiming to work out a specific sequence to assemble together all components of a product. Abed et al. [23] propose a modified EM with two-direction local search algorithm and genetic algorithm to determine the optimal time of task scheduling for dual-robot manipulators. They use two simulators to verify the modified EM, and the results show the good performance. Chao and Liao [24] present a discrete EM (DEM) algorithm for minimizing the total weighted tardiness in a single machine scheduling problem with sequence-dependent setup times.

Some researches deal with tardiness-related objective and consider eligibility constraints. McLendon et al. [25] deal with a scheduling problem considering job release times, due dates, and machine eligibility constraints to minimize total weighted tardiness. They propose an effective priority dispatching heuristic, called apparent tardiness cost-flexible (ATC-F), and show that the proposed ATC-F heuristic outperforms the well-known apparent tardiness cost (ATC) dispatching rule. Kim and Shin [26] consider a scheduling problem on parallel machines, in which every job has its release time and due date. Sequence-dependent setup times exist between the jobs, independently of the machines. The machines in parallel are either identical or non-identical in terms of the processing times of the jobs. They present a restricted tabu search algorithm that schedules jobs on parallel machines in order to minimize the maximum lateness of the jobs. Shim and Kim [27] consider a scheduling problem on unrelated parallel machines to minimize total tardiness. Processing times of a job on different machines may be different on unrelated parallel machine scheduling problems. They develop several dominance properties and lower bounds and suggest a branch-and-bound algorithm using them for the problem. Li [28] presents uniform parallel machine scheduling problems with unit-length jobs. Each job is only allowed to be processed on a specified subset of machines. He develops some methods to solve the problems with various objectives, including minimizing total tardiness, maximum tardiness, total completion time, number of tardy jobs, and makespan. Differing from [25] and [26], the studied problem $P_{\mathrm{m}}|\mathrm{EC}|\sum w_j T_j$ does not consider job release times and sequence-dependent setup times. Furthermore, the studied problem considers the identical parallel machine environment which is different from [27] (unrelated parallel machine environment) and [28] (uniform parallel machine environment).

3 Problem description

The following notation is used throughout the paper:

- n number of jobs
- m number of machines
- E number of EC levels
- J_j job j ($j = 1, 2, \ldots, n$)
- M_i machine i ($i = 1, 2, \ldots, m$)
- $e(J_j)$ EC level of job j
- $e(M_i)$ EC level of machine i
- p_j processing time of job j
- d_j due date of job j
- w_j weight of job j
- C_j completion time of job j
- T_j tardiness of job j

The considered problem can be formally defined and stated as follows. There are m identical parallel machines. Each machine M_i is specified by its EC level $e(M_i)$. A set of n jobs, all available for processing at time zero, is to be assigned to the eligible machines. Each of the n jobs has an EC level $e(J_j)$, processing time p_j, due date d_j, and weight w_j. A smaller value of $e(J_j)$ or $e(M_i)$ represents higher level. That is, number 1 represents the highest level, number 2 represents the second highest level, etc. Each job is only allowed to be processed on a particular machine when the EC level of the job is no less than the EC level of the machine ($e(M_i) \leq e(J_j)$). The tardiness of job j is defined as $T_j = \max(0, C_j - d_j)$. Each machine can process only one job at any time, and the processing of the job cannot be interrupted.

Due to the essential complexity of the proposed $P_{\mathrm{m}}|\mathrm{EC}| \sum w_j T_j$ problem, a new discrete EM in this research is developed. To the best of our knowledge, we first apply an EM to a parallel machine scheduling problem. In the proposed discrete EM, a new encoding scheme, a new distance measure between solutions and the effective attraction and repulsion operators are designed for the addressed problem. The proposed discrete EM will be compared with the current scheduling method in the metal nut plant, and a recently existing genetic algorithm (GA).

4 Current scheduling method in metal nut plant

For minimizing total weighted tardiness, there are three main considerations in dealing with the considered parallel machine problem: due dates, weights and eligibility constraints. A job with the shorter due date should be sequenced toward the beginning of the schedule. Conversely, since the weights denote extra penalty costs and opportunity losses, a job with the larger weight should be sequenced toward the beginning of the schedule. Therefore, the metal nut plant uses d_j/w_j which represents the priority of jobs. Then, the concept of shortest processing time first (SPT) rule based on d_j/w_j is applied to the identical parallel machine problem, i.e., m jobs with the smallest d_j/w_j are assigned to m machines. After that, whenever a machine is available, the job with largest d_j/w_j among those not yet assigned is put on the machine. All the assignments are not able to violate the eligibility constraints. According to the considerations, here we introduce the scheduling method currently employed in the metal nut plant:

Table 1 Job data for example (in hours)

J_j	1	2	3	4	5	6	7
p_j	71	31	40	35	47	58	42
d_j	79	34	121	52	128	98	126
w_j	4	5	4	3	6	2	3
$e(J_j)$	1	1	2	2	2	2	2

Step 1 Compute d_j/w_j for all jobs. Sequence jobs in no-decreasing order of d_j/w_j.

Step 2 Assign the job with the smallest d_j/w_j to the machine which is capable of processing the job.

Step 3 Whenever a machine is available, choose the unassigned job with the smallest d_j/w_j to allocate to the available machine which is capable of processing the job. Repeat the procedure until all jobs are assigned.

The current scheduling method combines the earliest due date first (EDD) rule, SPT rule and the eligibility constraints. It is a simply constructive method. A real-life example of 7-job, 3-machine, 2-EC-level problem in Table 1 with $e(M_1) = 1$, $e(M_2) = e(M_3) = 2$ is utilized to illustrate the current scheduling method. For convenience, level 1 and level 2 indicate high and common precisions, respectively. Because of $e(J_1) = e(J_2) = 1$, jobs 1 and 2 only can be processed on machine 1. Jobs 3–7 are allowed to be processed on any machine, since the EC levels of the jobs are of the same level and no less than the EC levels of all machines. The final schedule for the example is as shown in Fig. 2. The total weighted tardiness is 162.

5 About EM

EM is a population-based metaheuristic, inspired by the attraction–repulsion mechanism of the electromagnetism theory. EM is first developed by [14] for nonlinear optimization problems. To find the best solution, each solution (X_i) called particle adjusts its searching direction according to

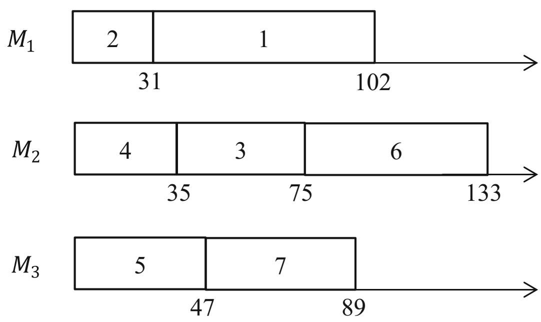


Fig. 2 Gantt chart for example of current scheduling method

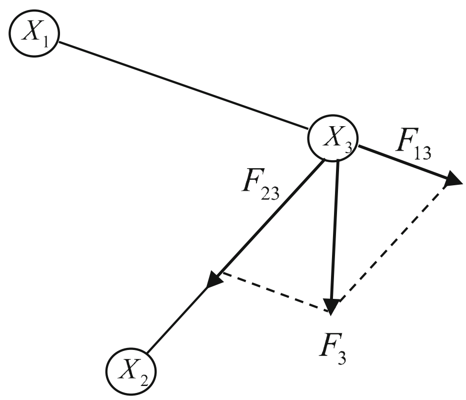


Fig. 3 Attraction–repulsion mechanism

a combination force exerted on it by all other solutions in the population. The better solutions exert attraction forces on X_i, while the inferior solutions exert repulsion forces on X_i. For example, consider three particles (X_1, X_2, X_3) and $(x_1) > f(x_3) > f(x_2)$. The total force exerted on X_3 is shown in Fig. 3. The objective value of X_2 is better than that of X_3, while the objective value of X_1 is worse than that of X_3. So, X_2 and X_1 exert the attraction and repulsion forces (F_{23} and F_{13}) on X_3, respectively. Finally, X_3 moves along with $F_3 = F_{13} + F_{23}$. Therefore, EM constructs a mechanism that encourages the solutions to converge to the highly attractive valleys and avoid moving toward the directions of the inferior solutions [14].

Birbil and Fang [14] further introduce Coulomb's law in the electromagnetism theory to determine the magnitude of attraction or repulsion. The magnitude of each force is directly proportional to the product of their "charges" and inversely proportional to the square of the "distance" between the particles. To obtain the magnitude of force, the "charge" of each particle has to be calculated at the beginning of each iteration. Let $X_i^t = (x_{i1}^t, x_{i2}^t, \ldots, x_{iD}^t)$, $x_{id}^t \in R$, be particle i with D-dimensional space at iteration t. Birbil and Fang [14] define the charge of particle i at iteration t as follows:

$$q_i^t = \exp\left\{-D\frac{f\left(X_i^t\right) - f\left(X_{\text{best}}^t\right)}{\sum_{k=1}^{N_p}\left(f\left(X_k^t\right) - f\left(X_{\text{best}}^t\right)\right)}\right\} \quad (1)$$

where q_i^t denotes the charge of particle i at iteration t. $f(X_i^t)$ and $f(X_{\text{best}}^t)$ represent the objective function value of the particle i and the best objective function value obtained from the population at iteration t, respectively. The parameter N_p represents the population size. Equation (1) specifies that the particles with better objective values possess higher charges.

After the calculation of the charge, Birbil and Fang [14] propose the total force F_i^t exerted on particle i by all other particles as the following equation:

$$F_i^t = \sum_{j \neq i}^{N_p}\begin{cases}\left(X_j^t - X_i^t\right)\dfrac{q_i^t q_j^t}{\left\|X_j^t - X_i^t\right\|^2} & \text{if } f\left(X_j^t\right) < f\left(X_i^t\right)\\ \left(X_i^t - X_j^t\right)\dfrac{q_i^t q_j^t}{\left\|X_j^t - X_i^t\right\|^2} & \text{if } f\left(X_j^t\right) \geq f\left(X_i^t\right)\end{cases} \quad (2)$$

where $q_i^t q_j^t$ is the product of the charges of particle i and particle j. $||X_j^t - X_i^t||^2$ is the square of the distance between particles i and j. Equation (2) indicates that the magnitude of each force exerted on particle i is directly proportional to the product of their charges and inversely proportional to the square of the distance between the particles. If particle j is better than particle i ($f(X_j^t) < f(X_i^t)$), the force is an attraction. Otherwise, the force is a repulsion. We add the forces exerted by all other particles together. Finally, X_i^t moves along with F_i^t and to the next solution X_i^{t+1}.

The original EM consists of four phases: initialization, local search, the calculation of the total force and movement. As shown in Fig. 4, the initialization phase generates a set of initial particles. In the local search phase, a specific local search is selected and employed in each particle. Equations (1) and (2) are used to calculate the total force of each particle in the third phase. In the phase of movement, each particle moves to the next position according to its total force. Phases 2–4 are repeated until a specified stopping criterion is satisfied.

6 Proposed discrete EM

In this section, a new discrete EM is developed to solve $P_{\text{m}}|\text{EC}|\sum w_j T_j$. The proposed discrete EM consists of three phase: initialization, the calculation of the force and movement. Clearly, a discrete particle needs to be designed to represent a valid assignment of jobs to machines. To calculate the force of discrete particle, we propose a discrete distance metric, called "1A2B" distance, by the concept of a number-guessing game (also known as bulls and cows game). Finally, a new attraction–repulsion operator, inspired by the crossover operator of GA, is introduced to obtain new particle in next iteration. Details are elaborated in what follows.

Fig. 4 The steps of the original EM

Initialization. Randomly generate a set of initial particles ($X_1^t, X_2^t, ..., X_{N_p}^t$).

Set $t = 1$.

Repeat the following until the stopping criterion is satisfied.

(1) *Local search*. Apply some local search to each particle (see Birbil and Fang (2003) for details).

(2) *Calculation of the total force*. Calculate the total force of each particle by equations (1) and (2).

(3) *Movement*. For each particle, move to the next position according to its total force (see Birbil and Fang (2003) for more details of the movement equation). Set $t = t + 1$.

6.1 Definition of discrete particles and initialization

A discrete particle is composed of two segments. The first segment is made up of job sequence, and the second segment is the allocation of machines to jobs. We redefine particle i at iteration t as $X_i^t = (x_{i11}^t, x_{i12}^t, \ldots, x_{i1n}^t \,|\, x_{i21}^t, x_{i22}^t, \ldots, x_{i2n}^t)$ where x_{i1k}^t represents the index of job placed at the kth position of the sequence and x_{i2k}^t represents the index of machine for job k. For example, $X_i^t = (1367452 \,|\, 1132312)$ would allocate jobs 1, 6, 2 to machine 1, jobs 7, 4 to machine 2, and jobs 3, 5 to machine 3. Therefore, the second segment is to determine which jobs are allocated to which machines and the first segment is to determine the sequence of the jobs allocated to each machine. In the proposed algorithm, all initial solutions are randomly generated.

6.2 Calculation of force

6.2.1 Calculation of charge

Before calculating the force of a particle, it is necessary to obtain the charge of each particle. In original EM, particles with better objective values possess higher charges. To keep the concept as simple as possible, the charge of particle i is calculated as follows:

$$q_i^t = \exp\left\{-\frac{f\left(X_i^t\right) - f\left(X_{\text{best}}^t\right)}{f\left(X_{\text{worst}}^t\right) - f\left(X_{\text{best}}^t\right)}\right\} \tag{3}$$

where $f(X_{\text{best}}^t)$ and $f(X_{\text{worst}}^t)$ denote the best and worst objective values obtained from particles in the population at iteration t, respectively. $f(X_i^t)$ is the objective function value of particle i. To efficiently measure the difference between $f(X_i^t)$ and $f(X_{\text{best}}^t)$, a range $f(X_{\text{best}}^t) - f(X_{\text{worst}}^t)$ is used as the denominator. If $f(X_{\text{best}}^t) - f(X_{\text{worst}}^t) = 0$, then the denominator is set to 1. Equation (3) states that if the objective value of particle i is closer to the best objective value, the charge of particle i will be higher.

6.2.2 Distance measure between particles

From Eq. (2) in Sect. 3, one of the important keys in successfully applying EM to discrete optimization problems is how to design a distance measure between particles for the specific problem. We propose a "1A2B" distance, inspired by a number-guessing game, for the distance metric. The number-guessing game, also called bulls and crows game, has two players. First, Player 1 writes a 4-digit number. For example, the number is <u>7425</u>. Next, Player 2 tries to guess a 4-digit number and Player 1 drops a hint about the number of matches. If the matching digit is on its right position, it is "A", if on different position, it is "B". For example, Player 2 guesses the number is <u>7462</u> and then Player 1 will answer "2A1B". 2A consists of "7" and "4" and 1B is "2". Finally, Player 2 will continue guessing the number according to Player 1's hints until Player 2 reveals the secret number. The "1A2B" distance is a novel idea, and it is easy to understand. The typical EM algorithm is usually applied to solve continuous optimization problems without general constraints. When the developed idea of "1A2B" distance is used in a typical EM algorithm, the new EM also becomes suitable

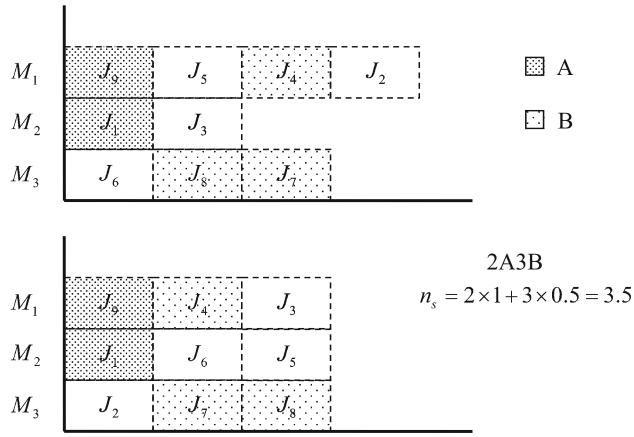


Fig. 5 1A2B distance

to solve the discrete optimization problems. Therefore, the "1A2B" distance is ideally suited to the studied problem for obtaining the discrete metric in order to calculate the distance measure between particles.

The proposed discrete EM applies the concept of "A" and "B" to make a comparison with two particles and obtain a discrete metric for the addressed scheduling problem. "A" denotes the matching job is at the same position on the same machine and gives 1 point. "B" denotes the matching job is at the different position on the same machine and gives a score of 0.5. The total points is defined as n_d. Higher value of n_d implies that two particles are much similar to each other, while lower value indicates a bigger difference. If $n_d = n$, two particles are identical (i.e., the distance between two particles is 0). Therefore, $(n - n_d)$ is used as the discrete distance between two particles. For example, suppose X_1^t and X_2^t are (195683427|212113331) and (192647385|231122331), respectively. By making a comparison between X_1^t and X_2^t, we have $n_d = 3.5$ and the discrete distance $(n - n_d) = (9 - 3.5) = 5.5$ (see Fig. 5). We notice that the different solution may obtain the same value of n_d, i.e., a value of n_d can consist of different combinations of "A" and "B". For example, $n_d = 3.5$ may consist of 3A1B or 7B.

6.2.3 Force of particle

The forces exerted on particle i by all other particles in the population are computed by the following equation:

$$F_{ij}^t = \begin{cases} \frac{q_i^t q_j^t}{n-n_d}, & if\, f\left(X_j^t\right) < f\left(X_i^t\right) \\ -\frac{q_i^t q_j^t}{n-n_d}, & \text{otherwise} \end{cases} \quad j = 1, \ldots, N_p, i \neq j \tag{4}$$

where $q_i^t q_j^t$ is the product of the charges of particles i and j. $n - n_d$ is the discrete distance between particles i and j. The positive value implements an attraction operator, while the negative value carries out a repulsion operator. In order to avoid the denominator from being zero, if $n_d = n$, then set $n_d = n - 1$. It is worth noting that we still keep the concept of Coulomb's law in Eq. (4).

6.3 Movement

6.3.1 Movement method

In the proposed algorithm, we develop a new movement method to generate a new particle as follows:

$$X_i' = \begin{cases} X_i^t \otimes X_{j^*}^t & \text{if } F_{ij^*}^t \geq 0 \\ X_i^t X_{j^*}^t & \text{otherwise} \end{cases} \quad i \neq j^* \tag{5}$$

where $\otimes$ is the attraction operator and $\odot$ is the repulsion operator (see Sect. 6.3.2 for details of the attraction and repulsion operators). X_i' is the new particle. j^* is the index of particle with the maximum (or minimum) force exerted on particle i and is defined as

$$j^* = \begin{cases} \arg\ \max_{j=1,\ldots,N_p} \left\{\left|F_{ij}^t\right|\right\}, & \text{if PA} \leq \text{PA}_0 \\ \arg\ \min_{j=1,\ldots,N_p} \left\{\left|F_{ij}^t\right|\right\}, & \text{otherwise} \end{cases} \tag{6}$$

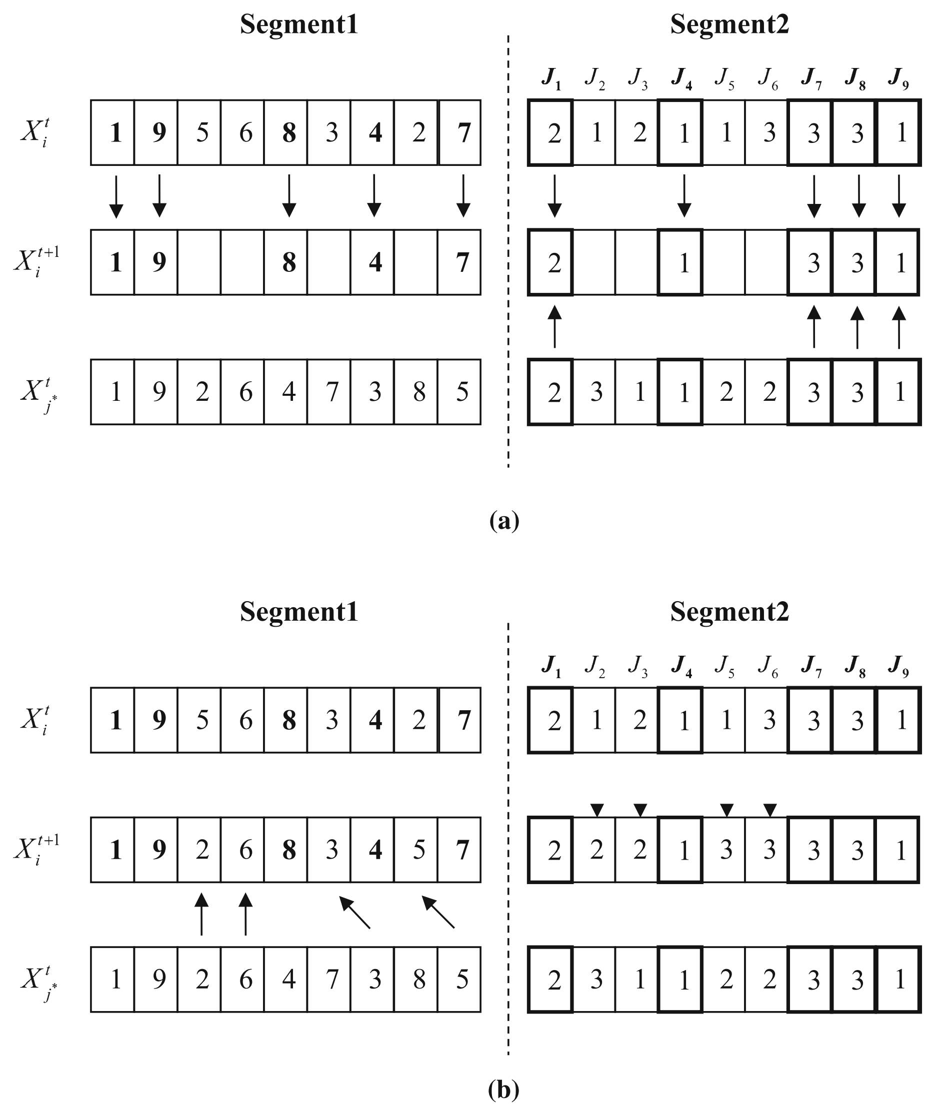
where $|F_{ij}^t|$ is the absolute value of F_{ij}^t, PA is a random number uniformly distributed in [0, 1], and PA_0 is a parameter which determines the relative probability of exploiting the maximum force versus minimum force. In the proposed algorithm, we choose one particle j^*, which is the maximum (or minimum) force exerted on particle i, as the implementation of the operator.

6.3.2 Attraction–repulsion operator

For the addressed problem, we develop a new attraction–repulsion mechanism, motivated by the crossover operator of genetic algorithm (GA). Details are given in what follows.

(1) Attraction operator
Particle j^* exerts an attraction force on particle i. It implies that the elements of particle i should contain some elements of particle j^*. This can be easily achieved by applying the concept of the crossover operator in GA. The procedure can be stated as follows: First, the second segments of particles i and j^* are examined on a position-by-position basis. Identical machines at the same positions are copied to the second segment of new particle, and the corresponding jobs in the first segment of particle i are inherited to the first segment of new particle. As shown in Fig. 6a, there exist five positions with identical machines in the second segments of two particles and they are copied to new particle. The corresponding

Fig. 6 Attraction operator. **a** The first step, **b** the second step



jobs (jobs 1, 4, 7, 8 and 9) in the first segment of particle i are copied to the first segment of new particle. Then, the machine indexes of unfilled positions in the second segment are determined by discrete uniform distributions in $U(1, e(J_j))$. Other jobs in the first segment of new particle are inherited in the relative order of the first segment of particle j^* (see Fig. 6b).

(2) Repulsion operator
Particle j^* exerts a repulsion force on particle i. It implies that the elements of particle i should differ from the elements of particle j^* as soon as possible. Thus, we intend to change the elements in the second segment of particle. The procedure is illustrated in Fig. 7. First, the first segment of particle i is copied to the new particle, and identical machines at the same positions between particles i and j^* are also copied to the second segment of new particle. Then, the machine indexes of unfilled positions in the second segment of new particle are randomly generated by discrete uniform distributions in $U(1, e(J_j))$, but they must differ from the indexes at the same positions of particle j^*.

The steps of the proposed discrete EM can be stated as follows:

Step 1 (Initialization) Determine parameters: N_p and PA_0. Set $t = 0$ and randomly generate initial particles X_i^t, $i = 1, 2, \ldots, N_p$.
Step 2 (Calculation of the total force) Calculate the total force of each particle by Eqs. (3) and (4).
Step 3 (Movement) Move to new particles X_i^{t+1} for $i = 1, 2, \ldots, N_p$ as follows:

a. The new particle X_i' is constructed according to Eq. (5).

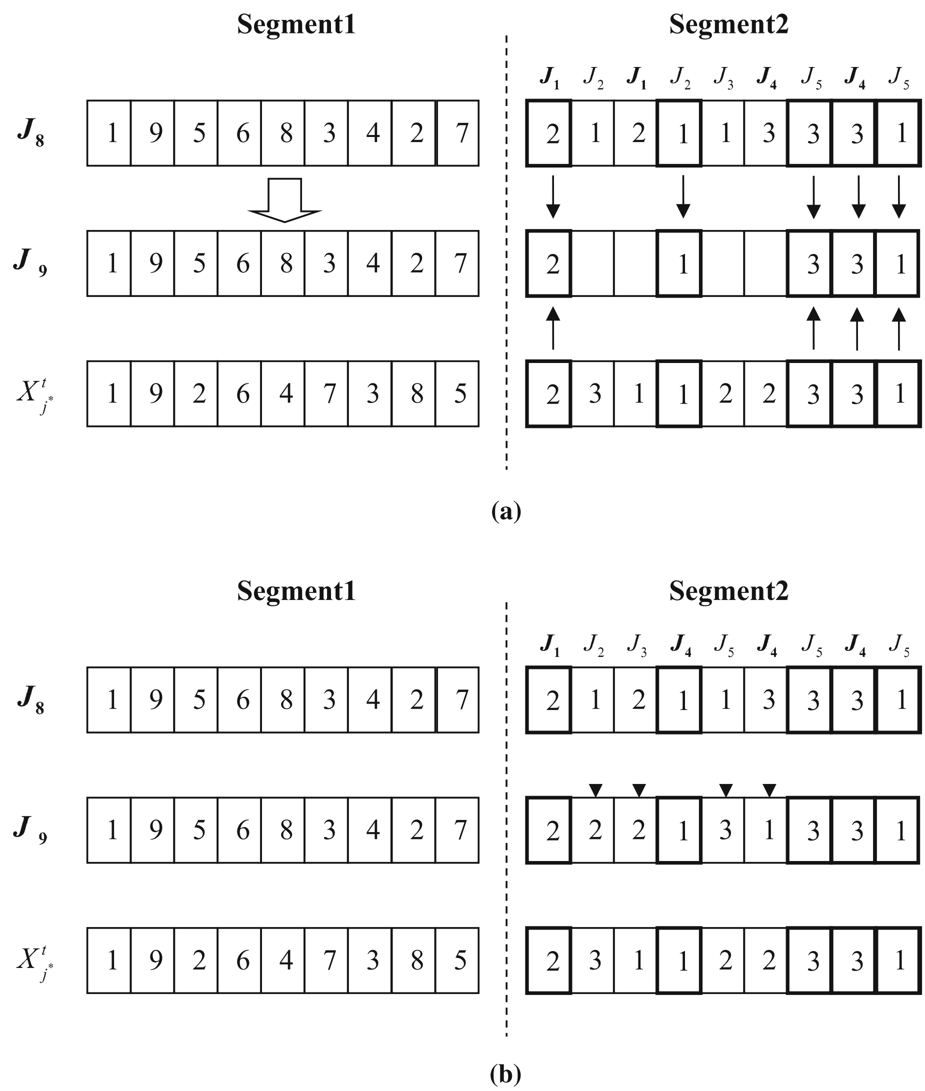


Fig. 7 Repulsion operator. **a** The first step, **b** the second step

b. If $f(X_i') \leq f(X_i^t)$, then $X_i^{t+1} \leftarrow X_i'$, else $X_i^{t+1} \leftarrow X_i^t$.

Step 4 (Termination) Stop the algorithm if the specified stopping criterion is satisfied; return to Step 2 and $t = t+1$ otherwise.

We use the same data in the metal nut plant (see Table 1) to illustrate the proposed discrete EM. The final schedule for the example is as shown in Fig. 8. The total weighted tardiness is 106. The solution ($\sum w_j T_j = 106$) derived by the discrete EM is significantly better than the solution ($\sum w_j T_j = 162$) of the current scheduling method used in the metal nut plant. The percentage improvement is about 34.57%.

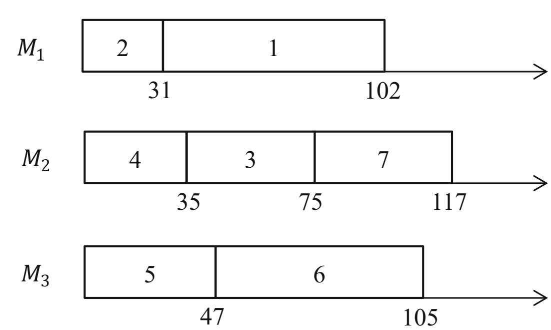


Fig. 8 Gantt chart for example of discrete EM

7 Computational results

Computational experiments were conducted to test the performance of the proposed discrete EM. The proposed discrete EM compares with the current scheduling method (hereafter referred to as CM), a GA and an ant colony optimization (ACO) algorithm. GA and ACO are also population-based metaheuristics. To compare in a fair manner, all the algorithms were coded in Visual C++ and run on an Intel Core 2 CPU (1.86 GHz) with 1.99 GB RAM.

7.1 Instance generation

There exists no benchmark instance for $P_{\mathrm{m}}|\mathrm{EC}|\sum w_j T_j$. The problem instances are generated according to a recent study by Lin et al. [12] for $R_{\mathrm{m}} \parallel \sum w_j T_j$. There are three different numbers of jobs ($n = 20, 50, 100$) with two different numbers of machines ($m = 4, 10$) and two different numbers of EC levels ($E = 2, 3$), where the processing time (p_j) and the weight (w_j) were drawn from the uniform distributions $U[1, 100]$ and $U[1, 10]$, respectively. The due date (d_j) was generated from a discrete uniform distribution ranging between $Y \times (1 - \mathrm{TF} - \mathrm{RDD}/2)$ and $Y \times (1 - \mathrm{TF} + \mathrm{RDD}/2)$, where $Y = (\sum_{j=1}^{n} p_j)/m$, TF was a tardiness factor, and RDD was a range of due date. The EC levels of jobs $g(J_j)$ and machines $g(M_i)$ were generated from the uniform distributions $U[1, E]$.

We utilize two parameters (TF, RDD) to generate different problem instances. (1) TF: {0.4, 0.8}; (2) RDD: {0.4, 0.8}. For each of the due date tightness and problem instance size, 20 instances were randomly generated from the associated distributions. There are 80 instances for each problem size with a total of 960 problem instances, which can be downloaded from http://web.ntust.edu.tw/~ie/index.html.

7.2 Performance of proposed discrete EM

To verify the performance of the proposed discrete EM, we make a comparison with CM, a recently existing GA of [12] for $R_{\mathrm{m}} \parallel \sum w_j T_j$, an ACO of [9] for $P_{\mathrm{m}} \parallel \sum w_j T_j$ and an apparent tardiness cost-flexible (ATC-F) heuristic of [25] for $P_{\mathrm{m}}|\mathrm{EC}|\sum w_j T_j$. The main concept of ATC-F is that the importance on the more flexible jobs and machines should take scheduling priority over less flexible jobs and machines. In the preliminary experiment, the following ranges of parameter values from the EM literature were tested [14,16–18]: $N_p = [30, 300]$ and $\mathrm{PA}_0 = (0.5, 1)$. Computational experiments were conducted on the first instance of each combination for all problem sizes. Based on the experimental results, the proposed discrete EM performs best under the following settings: $N_p = 80$, $\mathrm{PA}_0 = 0.8$. The same as in [12], all algorithms use 50,000 evaluations (population size×iteration) as the stopping criterion.

The relative percentage increase (RPI) is used as the performance measure. Let A_i represent the total weighted tardiness associated with Algorithm i. The RPI of Algorithm i is defined as:

$$\frac{A_i - \min\left(A_j, \forall j\right)}{\min\left(A_j, \forall j\right)} \times 100 \tag{7}$$

The computational results are summarized in Table 2, which gives the mean value of RPI (MRPI) yielded by the CM, the GA of [12], the ACO of [9], the ATC-F of [25], and the proposed discrete EM (DEM). We solved all the 960 instances, each of which was tested for 10 trials. The parameter values of GA which are the same as in [12] are also suitable here. All initial solutions are randomly generated. It is observed from Table 2 that DEM obviously yields better solutions than CM, GA, ACO and ATC-F. In particular, DEM can make significant improvements for the CM. In general, the solution quality of DEM increases as m is increased. Thus, the relative ranking of the algorithms in effectiveness (from best to worst average MRPI% values) is as follows: DEM, ACO, GA, ATC-F and CM.

Table 2 Comparison with other methods for MRPI%

n	m	E	CM%	GA%	ACO%	ATC-F%	DEM%
20	4	2	27.34	18.11	11.58	21.73	3.99
		3	19.81	12.83	7.79	15.19	4.56
	10	2	29.75	24.18	16.36	23.85	0.00
		3	31.66	23.31	14.15	28.44	0.05
50	4	2	44.73	25.62	15.42	32.56	5.27
		3	22.95	15.06	9.61	17.91	3.59
	10	2	72.46	60.27	24.76	64.32	0.00
		3	74.68	67.27	26.28	68.76	0.01
100	4	2	26.27	18.18	10.92	20.75	6.89
		3	15.83	8.02	6.88	8.94	6.40
	10	2	65.87	50.95	21.57	56.57	0.00
		3	70.94	52.80	20.65	60.34	0.13
Mean			41.86	31.38	15.50	34.95	2.57

More detailed comparison among DEM, GA, ACO and ATC-F is presented in Table 3, which gives the average number of times that the associated algorithm produces the best solution. Table 3 shows that DEM is better 787.1 times out of 960 for GA, 791.8 times out of 960 for ACO, and 866.4 times out of 960 for ATC-F. That is, in the comparison of 960 instances, DEM can outperform GA, ACO and ATC-F by 787.1 times, 791.8 times and 866.4 times in average, respectively. Except for the problem size ($n = 100$, $m = 4$ and $E = 3$), DEM consistently gives better results than GA, ACO and ATC-F. Table 4 exhibits the average CPU time required to solve the 960 generated problems using DEM. The CPU time is short enough to make DEM applicable to solve real-world problems.

We further draw a comparison with GA of [12] for $R_{\mathrm{m}} \parallel \sum w_j T_j$, since the GA is originally developed for $R_{\mathrm{m}} \parallel \sum w_j T_j$. The problem instances are generated the same as in [12]. The experiments were conducted on the tighter due date (TF = 0.8 and RDD = 0.4) problems. The results are given in Table 5, which gives the mean value of RPI (MRPI) and average number of times that the associated algorithm produces the best solution (N). We solve the 40 instances by using each of the algorithms, where each instance is tested for

Table 3 Comparison with GA, ACO, ATC-F and DEM

n	m	E	Generated problems	GA	DEM	ACO	DEM	ATC-F	DEM
20	4	2	80	21.0	59.0	23.0	57.0	8.8	71.2
		3	80	29.9	50.1	27.5	52.5	21.0	59.0
	10	2	80	0.0	80.0	0.2	79.8	0.0	80.0
		3	80	1.4	78.6	0.0	80.0	0.0	80.0
50	4	2	80	21.1	58.9	18.6	61.4	12.3	67.7
		3	80	28.9	51.1	33.8	46.2	15.7	64.3
	10	2	80	0.1	79.9	0.0	80.0	0.0	80.0
		3	80	0.4	79.6	0.2	79.8	0.0	80.0
100	4	2	80	26.6	53.4	29.5	50.5	10.8	69.2
		3	80	41.8	38.2	35.4	44.6	25.0	55.0
	10	2	80	0.2	79.8	0.0	80.0	0.0	80.0
		3	80	1.5	78.5	0.0	80.0	0.0	80.0
Total			960	172.9	787.1	168.2	791.8	93.6	866.4

Table 4 Average CPU time in seconds

Job	Machine	
	4	10
20	0.71	0.68
50	1.54	1.49
100	2.97	2.84

Table 5 Comparison with GA and DEM for $R_{\mathrm{m}} \parallel \sum w_j T_j$

$n \times m$	GA		DEM	
	MRPI%	N	MRPI%	N
20 × 4	99.73	8.5	24.82	19.1
100 × 10	488.56	4.6	83.71	15.4

10 trials. The initial populations of GA and DEM are created randomly. It is observed in Table 5 that the proposed DEM is still superior, although the difference is slightly reduced with the increase in the number of jobs.

Finally, an optimal way to solve the $P_{\mathrm{m}}|\mathrm{EC}|\sum w_j T_j$ problem is to use commercial optimization software (such as ILOG CPLEX) for small-sized problems. In this experiment, the test problem sizes are generated with the number of jobs $n = 15$ because almost all the test instances with $n \geq 16$ cannot be solved by ILOG CPLEX solver within 1 h. For each test problem, the proposed DEM runs 10 times to obtain the minimum solutions. The deviations (DVT) are calculated as

$$\mathrm{DVT} = (\text{Minimum solution of DEM}) - (\text{Solution by CPLEX}) \quad (8)$$

The calculated deviations are shown in Table 6. It can be seen from Table 6 that all the values of DVT are zero. The proposed DEM takes very short computation time (less than 0.1 s) for small-sized problems. This result shows that the DEM can obtain the optimal solutions when problem size is $n = 15$. Therefore, the DEM is an effective algorithm for solving the small-sized problems.

Table 6 Comparative results for DEM and CPLEX with $n = 15$

n	m	E	DVT	DEM CPU time (in sec.)
15	4	2	**0**	0.09
		3	**0**	0.08
	10	2	**0**	0.06
		3	**0**	0.06
Average			**0**	0.07

8 Conclusions

The eligibility constraints have been found in many practical situations. Most studies on eligibility constraints focus on parallel machine scheduling with makespan criterion. The main contributions of this paper are to model the real-life scheduling problem as $P_{\mathrm{m}}|\mathrm{EC}|\sum w_j T_j$ and to provide an effective discrete EM for the problem. To the best of our knowledge, we first apply an EM to a parallel machine scheduling problem. Specifically, we have developed a new encoding scheme, which adopts two-segment representation, to deal with the scheduling problem. In EM, it is necessary to obtain a distance measure between particles. A “1A2B” distance is proposed to calculate it. Finally, a new attraction–repulsion mechanism is developed, motivated by the crossover operator of GA. The computational results have demonstrated the superiority of the proposed DEM over the current scheduling method and recently existing GA, ACO and ATC-F. Moreover, the proposed DEM can obtain the optimal solutions for small-sized problems. In addition, the requirement of few parameters is also the advantage of the proposed DEM. Therefore, the proposed DEM has a conceptually easy design and is suitable for solving $P_{\mathrm{m}}|\mathrm{EC}|\sum w_j T_j$. From an application viewpoint, the DEM can make significant improvements for the CM, especially in solving large-sized problems. The schedule manager in the considered metal nut plant is satisfied with the results of the proposed DEM, and it will be arranged to be tested in the scheduling system of the case plant in the near future.

Future research may be conducted to further investigate the applications of discrete EM to other scheduling problems. It is also worthwhile to design other versions of EM to continue pursuing the best performance of discrete EM in solving the parallel machine scheduling problems.

Acknowledgements Funding was provided by Ministry of Science and Technology, Taiwan.

References

1. Hwang, H.C.; Chang, S.Y.; Lee, K.: Parallel machine scheduling under a grade of service provision. Comput. Oper. Res. **31**, 2055–2061 (2004)
2. Lu, X.; Liu, Z.: Semi-online scheduling problems on two uniform machines under a grade of service provision. Theor. Comput. Sci. **489**, 58–66 (2013)
3. Gokhale, R.; Mathirajan, M.: Scheduling identical parallel machines with machine eligibility restrictions to minimize total weighted flowtime in automobile gear manufacturing. Int. J. Adv. Manuf. Technol. **60**, 1099–1110 (2012)
4. Chuang, M.C.; Liao, C.J.; Chao, C.W.: Parallel machine scheduling with preference of machines. Int. J. Prod. Res. **48**, 4139–4152 (2010)
5. Glass, C.A.; Kellerer, H.: Parallel machine scheduling with job assignment restrictions. Nav. Res. Logist. **54**, 250–257 (2007)
6. Ou, J.; Leung, J.Y.T.; Li, C.L.: Scheduling parallel machines with inclusive processing set restrictions. Nav. Res. Logist. **55**, 328–338 (2008)
7. Ji, M.; Cheng, T.C.E.: An FPTAS for parallel-machine scheduling under a grade of service provision to minimize makespan. Inf. Process. Lett. **108**, 171–174 (2008)
8. Woeginger, G.J.: A comment on parallel-machine scheduling under a grade of service provision to minimize makespan. Inf. Process. Lett. **109**, 341–342 (2009)
9. Raghavan, N.S.; Venkataramana, M.: Parallel processor scheduling for minimizing total weighted tardiness using ant colony optimization. Int. J. Adv. Manuf. Technol. **41**, 986–996 (2009)
10. Zhang, R.; Wu, C.: A divide-and-conquer strategy with particle swarm optimization for the job shop scheduling problem. Eng. Optim. **42**, 641–670 (2010)
11. Hamidinia, A.; Khakabimamaghani, S.; Mazdeh, M.M.; Jafari, M.: A genetic algorithm for minimizing total tardiness/earliness of weighted jobs in a batched delivery system. Comput. Ind. Eng. **62**, 29–38 (2012)
12. Lin, Y.K.; Pfund, M.E.; Fowler, J.W.: Heuristics for minimizing regular performance measures in unrelated parallel machine scheduling problems. Comput. Oper. Res. **38**, 901–916 (2011)
13. Liaw, C.F.; Lin, Y.K.; Cheng, C.Y.; Chen, M.C.: Scheduling unrelated parallel machines to minimize total weighted tardiness. Comput. Oper. Res. **30**, 1777–1789 (2003)
14. Birbil, S.I.; Fang, S.C.: An electromagnetism-like mechanism for global optimization. J. Glob. Optim. **25**, 263–282 (2003)
15. Ali, M.M.; Golalikhani, M.: An electromagnetism-like method for nonlinearly constrained global optimization. Comput. Math. Appl. **60**, 2279–2285 (2010)
16. Birbil, S.I.; Fang, S.C.; Sheu, R.L.: On the convergence of a population-based global optimization algorithm. J. Glob. Optim. **30**, 301–318 (2005)
17. Chang, P.C.; Chen, S.H.; Fan, C.Y.: A hybrid electromagnetism-like algorithm for single machine scheduling problem. Expert Syst. Appl. **36**, 1259–1267 (2009)
18. Debels, D.; Reyck, B.D.; Leus, R.; Vanhoucke, M.: A hybrid scatter search/electromagnetism meta-heuristic for project scheduling. Eur. J. Oper. Res. **169**, 638–653 (2006)
19. Garcia-Villoria, A.; Moreno, P.R.: Solving the response time variability problem by means of the electromagnetism-like mechanism. Int. J. Prod. Res. **48**, 6701–6714 (2010)
20. Guan, X.; Dai, X.; Qiu, B.; Li, J.: A revised electromagnetism-like mechanism for layout design of reconfigurable manufacturing system. Comput. Ind. Eng. **63**, 98–108 (2012)
21. Yan, H.S.; Wan, X.Q.; Xiong, F.L.: A hybrid electromagnetism-like algorithm for two-stage assembly flow shop scheduling problem. Int. J. Prod. Res. **52**, 5626–5639 (2014)
22. Gao, L.; Zhang, C.; Li, X.; Wang, L.: Discrete electromagnetism-like mechanism algorithm for assembly sequences planning. Int. J. Prod. Res. **52**, 3485–3503 (2014)
23. Abed, I.A.; Koh, S.P.; Sahari, K.S.M.; Jagadeesh, P.; Tiong, S.K.: Optimization of the time of task scheduling for dual manipulators using a modified electromagnetism-like algorithm and genetic algorithm. Arab. J. Sci. Eng. **39**, 6269–6285 (2014)
24. Chao, C.W.; Liao, C.J.: A discrete electromagnetism-like mechanism for single machine total weighted tardiness problem with sequence-dependent setup times. Appl. Soft Comput. **12**, 3079–3087 (2012)
25. Mclendon, A.M.; Geiger, C.D.; Lodree, Jr, E.J.: Scheduling Parallel Machines with Machine Eligibility Restrictions Minimizing Total Weighted Tardiness. In: IIE Annual Conference. Proceedings. pp. 968. Institute of Industrial Engineers–Publisher (2007)
26. Kim, C.O.; Shin, H.J.: Scheduling jobs on parallel machines: a restricted tabu search approach. Int. J. Adv. Manuf. Technol. **22**, 278–287 (2003)
27. Shim, S.O.; Kim, Y.D.: Minimizing total tardiness in an unrelated parallel-machine scheduling problem. J. Oper. Res. Soc. **58**, 346–354 (2007)
28. Li, C.L.: Scheduling unit-length jobs with machine eligibility restrictions. Eur. J. Oper. Res. **174**, 1325–1328 (2006)