Applied Mathematics, Modeling and Computer Simulation
C. Chen (Ed.)

doi:10.3233/ATDE220106

Hybrid Gs-Ga Algorithm for Parallel Machine Scheduling Problem with Different Transport Vehicles

Hongxiang ZHANG[1]
School of Management, Hefei University of Technology, China

Abstract. This paper studies the scheduling problem of the same kind of parallel machines with different transportation vehicle capacity. The scheduling problem model is composed of a single warehouse and multiple manufacturers located in different geographical locations. Each manufacturer has the same batch processor, and several vehicles assemble the jobs in the warehouse in batches, transport them to different manufacturers and return them. The manufacturer of vehicle transportation depends on the destination of the jobs when the code is generated. At the manufacturer's place, because the capacity of the vehicle may exceed the maximum capacity of the batch processor, the job will be re batch for processing after it arrives. Through modeling, a $2n$ dimensional code is designed, and the gravitational search algorithm is introduced. The mutation and crossover process of genetic algorithm are added. A hybrid genetic algorithm based on gravitational search is proposed. The algorithm is used to optimize the sum of manufacturing time span and transportation cost. The effectiveness of the algorithm is verified by different scale simulation experiments.

Keywords. Batch processing, Transportation coordination scheduling, Gravity search algorithm, transportation cost.

1. Introduction

In recent years, with the development of economic globalization, network information technology has also reached a higher level. The degree of information interaction and information sharing among members of the production supply chain is getting higher and higher. More and more manufacturers are aware of developing information sharing to improve competitiveness and coordinate transportation capacity. At present, due to the continuous development and change of transportation mode and the different road conditions among the members of the production supply chain, how to choose the corresponding means of transportation has become a consideration in the scheduling model. Considering a variety of different modes of transportation batch scheduling has a strong industrial background, which is important to traditional manufacturing industries such as steel, home appliances and other enterprises. For example, Gree in Zhuhai will choose sea transportation, railway, highway and other transportation modes to transport raw materials to the factory for processing. Most of the traditional scheduling

[1] Corresponding Author, Hongxiang ZHANG, School of Management, Hefei University of Technology, China. Email: 963383855@qq. com.

models focus on machine, production, resource optimization and the impact of transportation without considering transportation cost. This paper discusses the selection of transportation vehicles for the production process and the scheduling problem of the influence of transportation cost on production.

Ikura Yoshiro and Gimple Mark[1] first proposed the concept of batch scheduling on 1986, and the batch scheduling problem has been widely discussed and studied. According to the number of machines, the problems are divided into single machine and multi machine scheduling. Then, according to the different processing process, the problems are divided into flow shop, parallel batch processing and operation workshop. Fariborz Jolai Ghazvini and Lionel Dupont[2] proposed BFLPT heuristics, and proved that the results were better than FFLPT. Kwanwoo Kim[3] considering the batch preparation time, when all batches arrive at the factory at zero time, the adaptive genetic algorithm is used to solve the production scheduling problem. Wong[4] proposed a hybrid particle swarm optimization algorithm to optimize the production cycle. Marimuthu[5] uses ant colony optimization algorithm to optimize the scheduling of jobs, and studies the batch optimization scheduling problem of flow shop. For parallel batch scheduling problem, Purushothaman Damodaran et al. [6] proposed a variety of heuristic algorithms to solve parallel batch scheduling problems of different size jobs, and proved the advantages of the heuristic algorithm. Ali Husseinzadeh Kashan et al. [7] proposed a discrete particle swarm optimization algorithm to minimize the manufacturing time span, and mixed with local search heuristic to solve parallel batch scheduling problem with inconsistent size of job. Kashan[8] also proposes hybrid genetic heuristic to solve similar problems and minimize the manufacturing time span.

Aiming at the production scheduling model of transportation cooperation, zegordi et al. [9] integrated production and transportation scheduling into two-stage production scheduling problem considering order allocation, and proposed dynamic genetic algorithm to solve the problem. Lixin Tang et al. [10] The transportation cooperative scheduling problem with single batch processor and multiple transportation vehicles is studied. The objective is to minimize the manufacturing time span and the total processing cost. A pseudo polynomial time algorithm is proposed to solve the problem. Mazdeh et al. [11] A branch and bound algorithm is designed to minimize the manufacturing time span and the delivery cost. Condotta and other scholars[12] had studied the cooperative scheduling problem of multiple vehicles for the same capacity, in which the initial dynamic arrival of the job and the delivery before the deadline is specified, and the goal is to find and minimize the maximum delay of the job. A tabu search algorithm is proposed to solve the problem. Wang and Cheng[13] proposed that the multi batch transportation cooperative scheduling problem is NP hard, and the goal is to minimize the sum of manufacturing time span and delivery cost, and proposed a dynamic programming algorithm to solve the problem.

In this paper, we study the cooperative scheduling problem of multi-location transportation for manufacturers with different capacity of transportation vehicles. By designing job scheduling, job loading and job batching schemes, we can minimize the sum of transportation cost and maximum completion time. According to the coding method proposed to this paper, a hybrid genetic algorithm based on gravitational search is proposed.

2. Notation and Problem Definitions

2.1. Problem Definitions

There is a production scheduling model consisting of a supplier and m manufacturers distributed in different locations considered in this paper. Each manufacturer has one identical batch processor. The supplier has enough non differential semi-finished jobs, and the assembly of the parts is recorded as $J = \{J_1, J_2, \ldots, J_n\}$, the job needs to be transported to the machine for further processing. This paper focuses on the choice of transportation mode in the transportation process, such as which road sections choose air transportation and which road sections choose land transportation, each transportation mode has its own speed and cost, so as to minimize the transportation cost and time.

The jobs with different processing time and size are located at the supplier at zero time. They will be divided into different batches and transported to the manufacturer's parallel batch processor by various transportation modes for processing. There is only one parallel batch processor in each manufacturer, and each batch processor has the same processing speed. The size of all jobs in each batch cannot exceed the capacity of the machine. After the batch starts processing, the whole processing process is not allowed to be interrupted. The goal of scheduling is to minimize the time span and transportation cost.

(1) In the stage of semi-finished products transportation from supplier to manufacturer, l kinds of transportation vehicle can be selected, and each kind of transportation vehicle has and only one. Each transport vehicle has a different speed and cost, which is recorded as $V_T = \{V_{T1}, V_{T2}, \ldots, V_{TL}\}$ and $C_T = \{C_{T1}, C_{T2}, \ldots, C_{TL}\}$, And the maximum capacity of the transport vehicle is $N_T = \{N_{T1}, N_{T2}, \ldots, N_{TL}\}$. Suppose that the unit round-trip distance from each supplier to the manufacturer is fixed, denoted as $t = \{t_1, t_2, \ldots, t_m\}$, Therefore, the actual one-way transportation time is $V_{Tk}/2t_k$

(2) Suppose that if the manufacturer's machine is idle, when the transport vehicle transports the job batch, because the capacity of the transport vehicle is not consistent with the capacity of the machine, it will be processed immediately after rescheduling. And the processing time is the maximum processing time of the batch.

2.2. Notation and Mathematical Model

It is assumed that supplier can transport semi-finished products to m manufacturers through l transportation modes, and the relevant symbols are defined as follows:

Notation	Definition
j	The index of job
i	The index of machine
k	The index of transportation mode
b, f	The index of batch
n	The number of jobs
m	The number of the machine
l	The number of transport vehicle
B	The capacity of machine
P_j	The processing time of job

r	The total number of batches, $\left\lceil \frac{n}{B} \right\rceil \le r \le n$
s_i	The dimension of job i
V_T	The speed vehicle
C_T	The cost of vehicle
N_T	The capacity of vehicle
P^b	The processing time of batch b
S^b	The start time of batch b
x_{jb}	1, if job j belongs to batch b in the stage in which the job is loaded onto the carrier. 0, else
y_{ib}	1, if batch b belongs to machine i 0, else
z_{ibf}	1, if batch b is processed before batch f in machine i
C_{max}	The manufacturing time span
d_k	The used times of vehicle k

$$Minimize \qquad C\sum_{k=1}^{l} d_k \times C_{Tk_{max}} \tag{1}$$

$$\sum_{b=1}^{r} x_{jb} = 1, j = 1,2,\dots,n \tag{2}$$

$$\sum_{b=1}^{r} y_{ib} = 1, i = 1,2,\dots,m \tag{3}$$

$$\sum_{j=1}^{n} y_{ib} \times s_j \le B, j = 1,2,\dots,n \tag{4}$$

$$\sum_{j=1}^{n} x_{jb} * s_j \le N_{Tk}, j = 1,2,\dots,n; k = 1,2,\dots,l \tag{5}$$

$$\sum_{b=1}^{r}\sum_{j=1}^{n} x_{jb} = n \tag{6}$$

$$P^b = max\{p_j \times y_{ib}\}, j = 1,2,\dots,n, i = 1,2,\dots,m \tag{7}$$

$$((y_{ib} + y_{if})z_{ibf} - 1)(S^b + P^b)y_{ib} \le S^f y_{if}, i = 1,2,\dots,m; b \ne f; \tag{8}$$

$$(S^b + P^b) \times z_{ibf} \le S^f, b \ne f; i = 1,2,\dots,m \tag{9}$$

$$C_{max} \ge (S^b + P^b) \times y_{ib}, i = 1,2,\dots,m \tag{10}$$

$$x_{jb} \in \{0,1\}, j = 1,2,\dots,n \tag{11}$$

$$y_{ib} \in \{0,1\}, i = 1,2,\dots,m \tag{12}$$

$$z_{ibk} \in \{0,1\}, i = 1,2,\dots,m; k = 1,2,\dots,l \tag{13}$$

Constraint(1) represents the objective function of the model to minimize the manufacturing span time and the sum of transportation costs, in which k parameter is adjusted to make the proportion of the two approximate. Constraint (2) means that the job can only be assigned to one batch in the loading phase. Constraint (3) means that the job can only be assigned to one batch in the processing phase. Constraint (4) means that the size and processing capacity of any batch in the second stage can’t exceed that of the batch processor. Constraint (5) means that the size of any batch of jobs in the first stage cannot exceed the maximum capacity of the transport vehicle. Constraint (6) means that there are n jobs. Constraint (7) means that the processing time of a batch depends on the one with the largest processing time in the batch. Constraint (8) means that each machine can only process one batch at the same time. Constraint (9) means that the next batch can be processed only after one batch is processed. Constraint (10) indicates that the maximum time span is the latest batch of jobs. Constraint (11), (12), (13) represents the value of the decision variable, and the value can only be 0 or 1.

In this paper, the machine coding method is used to determine the processing machine, and because each job needs to be transported to a specific factory by a transport vehicle for processing, the coding length is designed to be equal to twice the length of the number of processed jobs, and the gene string is composed of the serial number of the processing machine including the number and length of the front job and the serial number of the transport vehicle including the number and length of the back job. Referring to the paper of Jiang[14], all the jobs are arranged in descending order of processing time.

Table 1. Coding example of six jobs

J1	J2	J3	J4	J5	J6	J1	J2	J3	J4	J5	J6
1	1	2	2	1	1	1	2	3	1	2	1

The above table shows how to code in a scheduling model composed of two machines, six jobs and three different transport vehicles. The scheduling scheme is as follows: J_1、J_2、J_5、J_6 are assigned to machine 1, J_3、J_4 are assigned to machine 2.And J_1、J_4、J_6 are transported by vehicle 1, J_2、J_5 are transported by vehicle 2, J_3 are transported by vehicle 3. If the transport vehicle capacity is large enough at this time, vehicle 1 transports J_1、J_6 to manufacturer 1 and then transports J_4 to manufacturer 2. vehicle 2 transports J_2、J_5 to manufacturer 1. vehicle 3 transports J_3 to manufacturer 2.

(1) In the first stage, BFLPT heuristic rule is used in the packing process: when the coding situation is known, the transportation vehicles used by each job and which factory is assigned have been determined. Therefore, it is necessary to loop all the jobs to determine which kind of transport vehicle the job is loaded to which factory, and loop the existing batch of jobs. If the existing batch of jobs cannot hold the job or the batch attribute does not belong to the transport vehicle or factory to which the job belongs, create a new batch of jobs, put the job into it, and write the batch attribute of the job as the batch attribute of the job. Loop until all jobs belong to batches, and all batches belong to transport vehicles and factories.

(2) In the second stage, the first come first packing rule is adopted: the first come first batch processing. The arrival time of each job can be obtained by coding. At this time, each job in the factory will be sorted according to the arrival time, and added from the first job until it just does not exceed the capacity of the machine to form a batch.

3. Hybrid GS-GA Algorithm

Gravitational search algorithm is a group intelligence algorithm proposed by Esmat et al. [15] in 2009. Similar to particle swarm algorithm, it selects several protons which are determined by the fitness function value. Each proton has mutual attraction. After calculating the gravitation, each proton moves to its corresponding position and iterates the coding under the influence of gravitation. At the same time, some studies show that the convergence of this algorithm is better than proton swarm optimization and genetic algorithm.

First, N protons with multiple dimensional location information are generated, and the location information of the i proton can be expressed as $X_i = (x_i^1, x_i^2, \ldots, x_i^n), i = 1, \ldots n$.

Calculate the force $F_{ij}(t)$ between proton i and j using the following formula

$$F_{ij}(t) = G(t)\frac{M_{pi}(t)M_{aj}(t)}{R_{ij}(t) + \varepsilon}[x_j(t) - x_i(t)]$$

$M_{aj}(t)$ is the active gravitational mass of proton $j. M_{pi}(t)$ is the active gravitational mass of proton $i. G(t)$is the gravitational constant of the t generation. ε is a very small constant number, to prevent denominator from being 0； $R_{ij}(t)$ is the distance of proton i and proton j.

Next, the resultant force $F_i(t)$ received on the proton i is calculated

$$F_i(t) = \sum(rand * F_{ij}(t))$$

$rand$ ∈[0, 1].

Then calculating acceleration $a_i(t)$ of matter by resultant force and mass of matter

$$a_i(t) = \frac{F_i(t)}{M_{ii}(t)}$$

$M_{ii}(t)$ is the inertia mass of proton i,it is used to update the speed and position of protons

$$v_i(t+1) = rand * v_i(t) + a_i(t)$$
$$x_i(t+1) = x_i(t) + v_i(t+1)$$

The constant of gravitation is related to the number of iterations t

$$G(t) = G_0 \times e^{-\alpha\frac{t}{T}}$$

G_0 is initial value, T is the largest iterate, t is the current number of iterations, α is the time constant number.

The inertia mass and attractive mass of each material are related to the current fitness function of the proton. The specific update can be obtained by the following formula

$$M_{ai} = M_{pi} = M_{ii} = M_i, i = 1, 2, \ldots, n$$
$$m_i(t) = \frac{fit_i(t) - worst(t)}{best(t) - worst(t)}$$
$$M_i(t) = \frac{m_i(t)}{\sum_{j=1}^{n} m_j(t)}$$

$fit_i(t)$ means the fitness value of proton i in the t generation, $best(t), worst(t)$ represents the best and worst fitness values in the proton group of the t generation.

The new protons are generated by the above-mentioned method, and the mutation and cross operation of genetic algorithm are carried out.

Cross: The new proton will cross with the local optimal solution P_{best}^t of the current proton and the local worst solution P_{worst}^t of current protons, and the crossover point is selected randomly. Among the four new protons, the one with the best fitness function is selected to enter the next generation.

Mutation: Every proton and every dimension element have a 0.02 probability of mutation, and the new protons generated by the mutation directly enter the next generation.

In this paper, the discrete proton swarm optimization algorithm proposed by Kashan and Karimi [7] is used in the comparative experiment.

In 2008, Kashan and Karimi proposed a discrete PSO method (DPSO) to minimize the manufacturing span. Kashan and Karimi proposed the following formula for updating the position and velocity.

$$V_k^{t+1} = V_k^t \overset{+}{\circ} ((R_1 \overset{\times}{\circ} (P_k^t \overset{-}{\circ} X_k^t)) \overset{+}{\circ} (R_2 \overset{\times}{\circ} (P_g^t \overset{-}{\circ} X_k^t)))$$
$$A \overset{\times}{\circ} B$$

$$X_k^{t+1} = X_k^t \overset{+}{\circ} V_k^{t+1}$$

V_k^t is the velocity of the k particle in the population of the t generation, X_k^t is the position of the k particle in the population of the t generation, R_1 and R_2 is an array with only 0 or 1, which is obtained by Bernoulli distribution. In this paper, we use 0. 5 probability distribution 0 or 1, P_k^t is the optimal position of particle k in the previous t generation. P_g^t is the global optimal position of particle k in the previous t generation.

In the above expression, the definition of three operators is described as follows: Subtraction operator（$\overset{-}{\circ}$） define the distance between the current position and the expected position of particle k. $A \overset{-}{\circ} B$ is specifically expressed as follow: When the elements A and B are the same, the result is 0； When the element in A is different from the corresponding element in B, the calculation result is the element in A. However, in the original text, the heuristic method is used to allocate the jobs that have been grouped into batches, and the batches with element 0 are filled with machines. In this paper, the corresponding elements are filled randomly.

Multiplication operator（$\overset{\times}{\circ}$）is also used to enhance the search ability of the algorithm. Firstly, a series of one-dimensional arrays A with only 0 and 1 are generated by Bernoulli distribution, $A \overset{\times}{\circ} B$ is specifically expressed as follow: When the element in A is 1, the calculation result is the element in B; When the element in A is 0, the calculation result is 0. After calculation, all batches with 0 elements are filled with random values.

Finally, the Multiplication operator（$\overset{+}{\circ}$） is used two-point crossover in Kashan and Karimi's paper. In this paper, the two particles of the addition operator are randomly obtained with a probability of 0.5, that is, each element of the offspring is filled by one of the two corresponding position elements of the parents.

4. Computational Experiments and Comparison

In this paper, the initial coding is obtained by random generation. In the next method comparison, hybrid GS-GA algorithm is compared with the discrete particle swarm optimization proposed by Kashan and Karimi, genetic algorithm and gravitational search algorithm. The codes will be loaded by BFLPT heuristic rules, and then assembled to the machine for processing in the way of first come first batch. The crossover probability of genetic algorithm is 0.6, the mutation probability is 0.02, and the number of chromosomes is 10. The particles number of discrete particle swarm optimization algorithm is 10. The particles number of hybrid GS-GA algorithm and GS algorithm is 10, the crossover probability is 0.6 and the mutation probability is 0.02. The four algorithms all iterate 100 times, and each algorithm runs 300 times, taking the average value. The algorithm scale is depended on the following three aspects: the number of job J, the number of transport vehicles C, and the number of manufacturers M. Transportation cost, speed and distance between supplier and manufacturer are shown in the table below. The GS and GS-GA algorithm G_0 set to 100, ε set to 0.01.

Table 2. Parameters setting

Parameters	Number
Number of job J	J1=50; J2=100; J3=200
Number of transport vehicles C	C1=3; C2=5
Number of manufacturers M	M1=3; M2=5
Job size	$U[1,8]$
Job processing time	$U[1,8]$
Single cost of each transport vehicle	$U[3,7]$
Maximum capacity of transport vehicle	$U[8,12]$
Speed of transport vehicle	$U[8,12]$
Distance between supplier and manufacturer	$U[80,120]$

Table 3. Comparison of problem J1

Problems	GA	DPSO	GS	GS-GA
J1M1C1	233.7580	225.6563	168.2179	166.9860
J1M1C2	215.7747	207.4620	174.8335	162.7900
J1M2C1	251.0683	241.0889	208.1045	188.8487
J1M2C2	228.2326	217.5203	196.3771	176.3346

Table 4. Comparison of problem J2

Problems	GA	DPSO	GS	GS-GA
J2M1C1	451.3199	437.6565	367.4581	346.2339
J2M1C2	402.9401	389.8427	366.8365	333.8347
J2M2C1	472.8391	456.9781	431.7683	390.4560
J2M2C2	416.9418	400.9440	392.4856	348.6038

Table 5. Comparison of problem J3

Problems	GA	DPSO	GS	GS-GA
J3M1C1	847.2641	824.9090	779.1333	724.5511
J3M1C2	747.2094	728.2647	734.4077	663.4919
J3M2C1	887.3423	863.7265	854.1208	777.6340
J3M2C2	774.5580	750.9127	769.1104	682.3232

When the number of jobs is 50, 100, 200, the operation results are shown in the table above. After 100 iterations and 300 times of operation, the average value is taken. The solution of the GS-GA algorithm is obviously better than the former three, especially compared with GA algorithm. Moreover, GS-GA has more obvious advantages when the amount of computation increases and the scale of the problem becomes larger. When the scale is larger, GS algorithm is not easy to jump out of local optimal solution because the more optimal fitness has the smaller the search range, and some of the optimal values of the functions are worse than GA and DPSO. The GA-GA designed in this paper can effectively solve the problem that the original algorithm cannot jump out of the local optimal solution.

5. Conclusions

In this paper, the transportation and production collaborative scheduling problem is considered. Considering the transportation cost and production time span factors in the transportation process, the mathematical model of the problem is established under the condition of limited production capacity and transportation vehicles, and the hybrid GS-GA Algorithm is proposed. For factories, it is necessary to reduce the large amount of social wealth in the division of raw material logistics. In this problem, according to the known job sequence, the information of each job assembly transportation vehicle and distribution factory is generated, and then heuristic rules are used to group batch loading and group batch processing, in order to reduce the manufacturing time span and minimize the transportation cost, the transportation of resources is reasonably arranged, and the unnecessary waste of logistics cost in transportation is reduced. Compared with discrete particle swarm optimization, genetic algorithm and gravitational search algorithm, the effectiveness of the proposed algorithm is verified.

In the follow-up study, the production deterioration rate is considered, the production cost and transportation cost are solved separately, and the multi-objective optimization problem is adopted to find the Pareto optimal solution.

References

[1] Ikura Yoshiro and Gimple Mark. Efficient scheduling algorithms for a single batch processing machine[J]. 1986, 5(2) : 61-65.

[2] Fariborz Jolai Ghazvini and Lionel Dupont. Minimizing mean flow times criteria on a single batch processing machine with non-identical jobs sizes[J]. International Journal of Production Economics, 1998, 55(3) : 273-280.

[3] Kwanwoo Kim,In-Jae Jeong. Flow shop scheduling with no-wait flexible lot streaming using an adaptive genetic algorithm[J]. The International Journal of Advanced Manufacturing Technology,2009,44(11-12).

[4] T.C. Wong and S.C. Ngan. A comparison of hybrid genetic algorithm and hybrid particle swarm optimization to minimize makespan for assembly job shop[J]. Applied Soft Computing Journal, 2013, 13(3) : 1391-1399.

[5] S. Marimuthu,S.G. Ponnambalam,N. Jawahar. Threshold accepting and Ant-colony optimization algorithms for scheduling m -machine flow shops with lot streaming[J]. Journal of Materials Processing Tech.,2008,209(2).

[6] Purushothaman Damodaran and Ping-Yu Chang. Heuristics to minimize makespan of parallel batch processing machines[J]. The International Journal of Advanced Manufacturing Technology, 2008, 37(9-10) : 1005-1013.

[7] Ali Husseinzadeh Kashan and Behrooz Karimi. A discrete particle swarm optimization algorithm for scheduling parallel machines[J]. Computers & Industrial Engineering, 2008, 56(1) : 216-223.

[8] Ali Husseinzadeh Kashan and Behrooz Karimi and Masoud Jenabi. A hybrid genetic heuristic for scheduling parallel batch processing machines with arbitrary job sizes[J]. Computers and Operations Research, 2006, 35(4) : 1084-1098.

[9] Seyed Hessameddin Zegordi and Mohammad Ali Beheshti Nia. Integrating production and transportation scheduling in a two-stage supply chain considering order assignment[J]. The International Journal of Advanced Manufacturing Technology, 2009, 44(9-10) : 928-939.

[10] Lixin Tang and Hua Gong. The coordination of transportation and batching scheduling[J]. Applied Mathematical Modelling, 2009, 33(10) : 3854-3862.

[11] Mohammad Mahdavi Mazdeh and Mansoor Sarhadi and Khalil S. Hindi. A branch-and-bound algorithm for single-machine scheduling with batch delivery and job release times[J]. Computers and Operations Research, 2006, 35(4) : 1099-1111.

[12] Alessandro Condotta et al. Tabu search and lower bounds for a combined production–transportation problem[J]. Computers and Operations Research, 2013, 40(3) : 886-900.

[13] Guoqing Wang and T.C.Edwin Cheng. Parallel machine scheduling with batch delivery costs[J]. International Journal of Production Economics, 2000, 68(2) : 177-183.

[14] Lu Jiang et al. Uniform parallel batch machines scheduling considering transportation using a hybrid DPSO-GA algorithm[J]. The International Journal of Advanced Manufacturing Technology, 2017, 89(5-8) : 1887-1900.

[15] Esmat Rashedi and Hossein Nezamabadi-pour and Saeid Saryazdi. GSA: A Gravitational Search Algorithm[J]. Information Sciences, 2009, 179(13) : 2232-2248.