

The Application Research of Improved Genetic Algorithm Based on Chaos for job shop scheduling

Juping Peng^{1,a}

¹Gansu Normal University for Nationalities, Hezuo, Gansu, 747000, China

^ajupingpeng@163.com

Keywords: Genetic Algorithm; Job Shop Scheduling Problem; job splitting; Chaos

Abstract: An improved genetic algorithm is proposed for solving the job shop scheduling problem. Starting with the characteristic of job shop scheduling problem, Therefore, we use the advantage of chaos method and combine the simulated annealing algorithm to improve the genetic algorithm, and then We do a lot of experiments and evaluate the performance of the improved genetic algorithm. and comparisons with standard genetic algorithm demonstrate the effectiveness of the improved genetic algorithm.

1. Introduction

The job flow shop scheduling problem has been widely studied in the recent decades. An flow shop system consists of m machines in a serial layout where a set of n jobs has to go through first machine one, then machine two, and so on until machine m . That is, all the jobs have identical routes. Each job requires an operation with uninterrupted processing time on each of the machines. Each job can be preceded at most one machine at the same time. The processing of a job on a machine cannot be interrupted. All jobs are independent and are available for processing at time 0; and etc. The aim is to find a sequence for processing all jobs on all machines so that the given criterion is minimized. Although the process constraint of the model is relatively simple, but it has proven more than 3 machine of permutation flow shop scheduling problem is NP problem^[1].

More attention was paid on applying intelligent bionic algorithm to solve the problem in recently decades. Evolutionary algorithms such as simulated annealing algorithm^[2], ant colony algorithm^[3], PSO algorithm^[4] etc. Evolutionary algorithm can improve the population size and the number of iterations to improve accuracy, the solving time is proportional to iterative times, time efficiency decreased obviously. Some scholars, which based on the simulation the behavior of swarm intelligence, presented for solving combinatorial optimization problems, Such as genetic algorithms, particle swarm optimization, ant colony algorithm, swarm algorithm and this paper will discuss the simulation of plant growth algorithm. Those intelligent algorithms have been widely used in many areas, also include job shop production scheduling problem^[5-7].

2. Job shop scheduling problem

Job shop scheduling problem studies the flow process of n work pieces on m machines. If the scheduling objective is the maximum completion time, the case of Permutation flow shop scheduling problem mathematics is described as follows: n represents the number of jobs, m represents the machine number, $prmu$ shows that all the workpiece after each machine processing consistency, C_{max} represents the maximum completion time of the work pieces, t_{ij} denotes the processing time of workpiece i on machine j , $C(j_i, k)$ represents the completion time of workpiece j_i on machine k , π represents a sort of all jobs, T is collection of all sort. Assume that the workpieces are processed according to machine 1 to m , then the completion time of n work pieces on m machines can be obtained by formula(1) to formula(5).

$$c(j_1, 1) = t_{j_1, 1} \quad (1)$$

$$c(j_i, 1) = c(j_{i-1}, 1) + t_{j_i, i} \quad i = 2 \dots n \quad (2)$$

$$c(j_1, k) = c(j_1, k-1) + t_{j_1, k} \quad k = 2 \dots m \quad (3)$$

$$c(j_i, k) = \max\{c(j_i, k-1), c(j_{i-1}, k)\} + t_{j_i, k} \quad i = 2 \dots n, k = 2 \dots m \quad (4)$$

$$c_{\max}(\pi) = c(j_n, m) \quad (5)$$

$$\pi' = \arg\{c_{\max}(\pi) = c(j_n, m)\} \rightarrow \min, \forall \pi \in T \quad (6)$$

Which formula (5) is the maximum completion time, formula (6) represents the corresponding scheduling scheme of minimize the maximum completion time.

3. The improved GA

3.1 Genetic Algorithm

The GA is one kind of iterative adaptive heuristic probabilistic searching algorithm. It has global optimization strategy which can avoid falling into local optimum. GA generate the code under necessary conditions of arranging, which is called the initial individual; Then select the appropriate fitness function, calculate its value to measure individual quality degree of fit and unfit, the larger the value, the more superior the individual. The greater probability to be selected to inherit, it is conforms to the natural evolution rule. Then, judgment individual satisfy the constraints, thus carry on heredity operations such as choice, exchange or variation to the individual. To get the better of the progeny population, which arrange the better solution? Then compute new generation community's compatible value again, if we can output the arranging result, otherwise continues to make the heredity operation, until found a more superior filial generation.

The GA is one kind of quick, simple, the fault-tolerant strong algorithm, the search process does not directly act on the variable, but on the individual of parameter sets encoded, this causes the genetic algorithm to be possible to operate directly to the structure object.

3.2 Genetic Searching Algorithm

Traditional GA has the shortcomings of early convergence and variation problem, makes the improvement based on the genetic algorithm merit. We can use chaos theory to improve early convergence and mutation question. And combine the simulated annealing algorithm to find the minimum value in the optimization problem.

The GA selects next generation population in the evolution by basing on the fitness value of each individual. Fitness function set directly influence on speed of the GA convergence and to find optimal solutions. In this article, the fitness function's design concept is weight sum conflict type which exists in each chromosome, If one of chromosomes violate rule i , its value will be set to $1/P_i$, the conflict which exists to the chromosome carries on the weighting to sum and to add on 1, then make reciprocal, the chromosome sufficiency function value is bigger, its next-generation evolution of the probability of survival is larger.

Selecting operation is used to simulate the phenomenon of natural selection in biosphere. It selects a high fitness chromosome from the old population, and put into the matching set, and prepare to chromosome crossover and mutation operation to generate a new population. Higher fitness chromosomes are selected more possibly. We choose a method of local selection method, its truncation selection method.

Basic thought: with delicate inherent law of chaotic sequence to control the crossover and mutation genetic operations, to replace completely random probability crossover and mutation operations, including two aspects: determine whether to cross or variation operation; and the operation of the specific gene position. From short-term, the genetic operation seemingly random, and in the long run, there are some delicate internal relations, which avoid to pure random operation of the "blindness". Operations are the same with the standard genetic algorithm, except for chaos cross instead of cross and chaos variation instead of variation. We introduce four chaotic sequences which independent of each other in the algorithms. Records as: $T_{k1}, T_{k2}, T_{k3}, T_{k4}$. In theory, we use the chaotic sequence, present in random distribution characteristics in short-term (adjacent several value), and the whole sequence is in the (0, 1) with no interval repeat ergodic sequence. This kind

of short term random characteristics helps population present individual diversity in the short term, avoid the local optimum, and the chaos ergodicity may overcome "repeatability" and "blindness" possible in the simple random operation, thus, can further ensure the multiplicity of the chromosome, avoid premature convergence and unnecessary repeat search and improve the search efficiency.

The concrete steps of the algorithm are as follows:

Step1: Parameters initialization. Parameters, which need to be initialized, include: the population size N , the maximum algebra M , the selection rate P_s , and the sequence $T_{k1}, T_{k2}, T_{k3}, T_{k4}$.

Step2: Generate an initial population of population size N randomly according to the encoding scheme.

Step3: Begin to iterate.

Step4: Do the chaotic crossover operation to the chromosomes in population according to the current value of T_{k1} and T_{k2} . All the chromosomes, generated by mixed crossover operation, constitute a sub-population.

Step5: Do the mutation operation to the chromosomes in the sub-populations according to the current value of T_{k3} and T_{k4} .

Step6: Calculate the fitness of all chromosomes in populations and sub-populations.

Step7: Do the selection operation to all of the chromosomes in populations and sub-populations according to the selection probability P_s , eliminate low adaptation chromosomes, the else chromosomes retained constitute new populations.

Step8: When the standard of iteration stopping is m , end the iteration, output current optimal chromosome. It is the chromosome of highest adaptation degree, the optimal chromosome is the original problem better solution; otherwise returns step 4.

4. The experiment results and discussion

In order to explain the performance of the proposed algorithm, the numerical examples used in this paper are generated by random method. Examples are 10 Liao benchmark problems^[9]. Each example contains 5 stages. The number of parallel machines in each phase is distributed between 3 and 5. The range of the workpiece processing time is [1, 1000]. On this basis, the assumption that all the machines are not always available, they may have a failure, the failure of the machine will be immediately repaired, where the time of machine failure and repair spent are subject to normal distribution. In the experiment, there are 5 failures in the setting of the machine, and the mean value of fault time is 200, 400, 600, 800 and 1000 respectively. The mean value of the time used in the repair is 20 and the variance is 5. Run environment of algorithm is MATLAB (R2010b) under Win7 operating system. Under two situations of the algorithm in preempt-resume (case 1) and preempt-repeat (case 2) solving results respectively as shown in Table 1 and 2.

Table 1 Comparison results on benchmark problems of Casel

Problem	PSO			RKGA			IA			IGA		
	AVE	MIN	SD									
j30c5e1	511.7	507	2.4	513.0	505	3.9	509.0	506	1.5	502.9	500	1.4
j30c5e2	670.3	670	0.4	670.5	670	0.8	670.0	670	0.0	669.5	669	0.0
j30c5e3	663.6	657	3.7	667.5	655	7.5	659.6	655	1.7	653.9	650	1.4
j30c5e4	631.3	628	2.1	630.7	626	3.1	629.2	627	1.2	622.1	620	1.0
j30c5e5	668.5	666	1.7	671.7	666	4.5	662.8	662	0.4	662.4	661	0.3
j30c5e6	675.4	670	3.3	679.0	670	5.6	669.1	665	2.5	663.3	660	2.1
j30c5e7	686.1	684	1.8	686.0	683	2.5	683.9	682	0.9	681.0	680	0.3
j30c5e8	737.9	732	3.2	738.3	731	5.1	734.2	731	1.8	729.2	728	1.0
j30c5e9	708.8	705	2.0	709.8	703	5.1	705.4	702	1.6	699.1	697	1.1
j30c5e10	650.5	641	5.3	648.8	636	7.2	642.5	637	2.2	634.2	630	1.4
Average	660.4	656	2.6	661.5	655	4.5	656.6	654	1.4	652.0	650	1.0

Table 2 Comparison results on benchmark problems of Case2

Problem	PSO			RKGA			IA			IGA		
	AVE	MIN	SD	AVE	MIN	SD	AVE	MIN	SD	AVE	MIN	SD
j30c5e1	561.6	557	5.1	562.1	555	8.7	559.6	554	2.4	552.4	548	2.5
j30c5e2	720.0	701	4.4	723.8	701	7.8	716.3	701	3.2	700.1	696	2.3
j30c5e3	747.5	739	6.0	748.6	735	6.2	743.0	736	4.5	723.0	711	4.1
j30c5e4	702.2	696	7.5	701.3	694	8.1	697.5	688	5.9	685.1	673	5.6
j30c5e5	733.7	726	8.7	734.1	726	13.5	731.5	726	5.9	724.9	708	3.4
j30c5e6	737.1	730	3.6	737.0	720	6.0	730.7	718	3.1	717.8	707	8.5
j30c5e7	752.2	745	4.3	751.8	743	6.9	747.9	741	3.8	731.7	722	4.3
j30c5e8	836.7	802	10.4	839.2	792	14.9	826.4	790	9.5	786.4	770	6.5
j30c5e9	799.4	791	6.8	800.2	789	15.4	793.6	788	5.9	766.4	759	4.6
j30c5e10	748.9	740	3.2	746.2	715	11.6	738.1	712	9.3	718.1	709	8.4
Average	733.9	723	6.0	734.4	717	9.9	728.5	715	5.4	710.5	700	5.0

From the test data can be seen, for the selected Liao problems, the plant growth simulation algorithm has good optimization performance in the field of combinatorial optimization, and it is an effective tool for solving the permutation flow shop scheduling problem.

5. Conclusions

This paper proposes an improve GA, the improve GA is applied to solve the job shop scheduling problem, the performance has been improved obviously, and as the problem size increases, the improvement is more obvious. Experiments show that: The algorithm is effective and robust, it is feasible and effective for solving the job shop scheduling problem. It shows a good application prospect in production scheduling field.

References

- [1] Li T, Wang C F ,et al. A global optimization bionics algorithm for solving integer Programming plant growth simulation algorithm[J]. Systems Engineering Theory & Practice ,2005 ,25(1) :76 - 85.
- [2] Li T, Wang Z t. Application of plant growth simulation algorithm on solving facility location problem [J]. Systems Engineering Theory & Practice, 2008, (12):107-115.
- [3] Wang C ,Cheng H Z. Reactive power optimization base on plant growth simulation algorithm[J]. Power System Technology,2006 ,30(21):37-41.
- [4] Ding Xuefeng, Ma Liang, Ding Xuesong. The location allocation of logistics center of perishable products based on plant growth simulation algorithm[J]. Systems Engineering, 2009,27(2):96-101.
- [5] Pinedo M. Scheduling: theory, algorithms, and systems, 2nd Prentice-Hall, Englewood Cliffs, New Jersey. 2002, pp.126-144 [6] Marcelo Bertalmio, Luminita Vese. Image Filling-in in a Decomposition Space [J]. IEEE, 2003, (1):851 -853.
- [6] Rao Y Q, Yan Z X, et al. Application of a Hybrid Genetic Algorithm for Job Shop Scheduling[J]. Mechanical Science and Technology ,2006 , 25 (5) :584-587,607.
- [7] Liu Y, Ye C M, Shen Y H. Solving the Job shop scheduling problem based on improved particle swarm algorithm [J]. Systems Engineering and Electronics, 2006 , 28 (4) :602-606.