Journal of Applied Sciences

ISSN 1812-5654

ANSInet
an open access publisher
http://ansinet.com

Journal of Applied Sciences 13 (15): 2947-2952, 2013
ISSN 1812-5654 / DOI: 10.3923/jas.2013.2947.2952

Research on Time Table Problem Based on Improved Genetic Algorithm Combined Chaos and Simulated Annealing Algorithm

Dong Yunfeng
Computer Center, Qilu University of Technology, Jinan, China

Abstract: The scheduling problem is a typical time table problem in educational administration. For such a NP complete problems, when the genetic algorithm solves this problem, it has precociousness phenomenon and quickly converges not to the global optimal solution but to the local optimal solution. Therefore, we use the advantage of simulated annealing algorithm to transform the fitness function and chaotic sequence to control the crossover and mutation genetic operations and then overcome the weakness of genetic algorithm in the Time Table Problem. We do a lot of experiments and evaluate the performance of the improved genetic algorithm. The experiment results show that improved genetic algorithm is a more superior algorithm to apply to the TTP problem.

Key words: TTP, chaos algorithm, simulated annealing algorithm, genetic algorithm

INTRODUCTION

Scheduling is the most important and most complex part of educational administration. The essence of course scheduling question is the curriculum, the teacher and the student should be assigned in the appropriate time section to the appropriate classroom (Dimopoulou and Miliotis, 2004). It is a multi-objective scheduling problem which involve more factors and known as the Time table problem in operations research (Time table Problem, called TTP). Automated course scheduling algorithm, TTP, was proved to be NP-complete problem as early as in the 70s. The computing time is exponential increase and integration time and space as dual constraints. From classroom (space) point of view it is similar to the bin-packing problem (Charlga and Chung, 2005). But increased many restraints in the time. Solving the NP complete problem only to be able to depend on the approximate method, many scholars have conducted the research regarding this. We combine the advantage of the chaos and simulated annealing algorithm, present a improved genetic algorithm and use it to seek better solution of TTP.

ARRANGEMENT OF CONDITIONS TO BE FOLLOWED

Definition of course scheduling: In the Course Scheduling question, it is Involves 5 restricted mutually factors, the class and grade, the teacher, the time, the curriculum, the classroom. The schedule question's solution process is to any class and grade, the teacher, the curriculum, the classroom arranges same idle time. When arrangement cannot have the conflict, simultaneously satisfies the experience and general knowledge as far as possible.

Constraints: We will restrain divides into two levels constraints; first-level restraint for the basic restraint which must satisfy the second-level restraint on the trimming in the first-level restraint satisfied foundation to better meet the requirements.

First-level restraint: Each curriculum serial number must assign in one classroom, attends class or the instruction by one teacher; several curriculum per week must have a certain time intervals; The identical teacher cannot in the same time in the different classroom on different subjects; Within the same classroom at the same time can not be arranged two or more courses; Classroom seat quantity must be greater than or equal to the students; Some classes must use the special classroom or multimedia and successive periods, such as art design; The identical class's students cannot have two or two more courses in the same time. These are the conditions that must be met when arranging, otherwise is unable to carry on the normal teaching (Van Heiningen *et al*., 2006).

Second-level restraint: The teacher's special request, some teachers want the lecture time to be concentrated; Some teachers do not want to class in the evening.

IMPROVED GENETIC ALGORITHM FOR SCHEDULING SYSTEM

Traditional GA has the shortcomings of early convergence and variation problem, makes the improvement based on the genetic algorithm merit (De Almeida *et al*., 2007). We can use chaotic sequence to control the crossover and mutation genetic operations, to replace completely random probability crossover and mutation operations. And we will use the simulated annealing algorithm to transform the fitness function to the objective function value f(x), which meets the simulated annealing algorithm. The methods partially matched crossover operation to improve early convergence question which caused by Cross-Stage and improve the mutation phase variation problems.

Coding: Composite each course and teacher, to solve "multi-class", “one more class for one teacher " conflict, fixed time interval problems. We use the mixed style teacher to code, set the teachers' hybrid coding as the GA genes. The genes are composed by the teacher number, the class and grade serial number, the curriculum number, the curriculum characteristic. As shown in Fig. 1.

Initial population's production: There are two steps for initial population's production: The first step, for each class Ci (i = 1... n), teacher numbers will be arranged randomly in 20 time slices. If there are 5 classes (i = 5), then we can get chromosome after five circulations, according to the population scale's size which assigned produces the certain amount individual to form the initial population. Of course, there are possible initial generation of individuals, but also feasible. The second step, use GA to find the near-optimum solution from the feasible solution. This step can be described as: through the loop, the fitness of a feasible solution to a given fitness value, that solution is the optimal solution obtained; After circulates massively, if the feasible solution sufficiency value cannot achieve the sufficiency value which assigns, then the conflict will take the penalty function and define the penalty function in the sufficiency function.

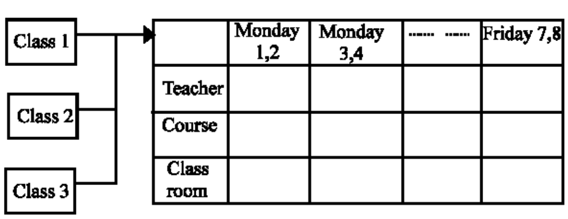

Fig. 1: Gene encoding

Fitness function: 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 Pi, 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, then indicate that has the good teaching time interval and the classroom, its next-generation evolution of the probability of survival is larger.

In this step, we should find the minimum value in the optimization problem to improve early convergence question. Simulated annealing algorithm will get the minimum equilibrium energy point through annealing step by step, is applied to find the minimum value in the optimization problem. So, when the genetic algorithm combines the simulated annealing algorithm to be applied to course scheduling problem, we will transform the fitness function to the objective function value f(x), which meets the simulated annealing algorithm.

The transformation formula is as follows:

$$f(x) = \exp(-\frac{\text{fit}(x)}{1000})*1500 \quad (1)$$

Then, the individual simulated annealing procedure is:

- Step 1: Take the individual after variation; calculate the individual simulated annealing objective function value f(x) in the current state
- Step 2: Set the initial temperature: T_0, make the $T_k \leftarrow T_0$
- Step 3: Generate field solutions, on the current state of the individual genes: Lesson element-time-classroom coping within the same time slice random changes to produce a new individual to determine whether an individual after their changes reasonable to calculate the new value of the objective function and calculate the value of the objective function increment Δf

If Δf<-0. Accepts the new individuals generated for the current most advantages If Δf>0, with probability p = exp ($-\Delta f/T_k$) to accept the new individuals generated for the current most advantages:

- Step 4: The temperature T_k decline continue the cycle
- Step 5: Determine the value of T_k, if T_k< = 0, the end of the annealing operation

Selecting operation: 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.

In truncation selection method, chromosome is sorted by fitness function value from high to low. Only the best individual can be selected as parent individual. The percentage parameters, which is used to determine the parent individual, is called truncation threshold trunc, its range is 10-50%, the individual, which out of threshold, can not produce sub-individuals.

Crossover operation: Restructuring to achieve individual crossover design should consider the following questions:

- First, any crossover operator will need to meet the assessment criteria of crossover; Crossover operator needs to ensure that the parents generation's best traits of individuals can be inherited as much as possible in the new generation of individuals
- Second, the designed crossover operator coordinates with the code. We use the part match interlace operation in this algorithm, the request chromosome gene are the teacher serial numbers, the teacher's code should not repeat in any slice of a day (Sorin, 2009)

Firstly, in parent generation of individual even randomly selects two points; Next exchanges the strings between parent generation of individual two spots, determines the mapping relations between a parent generation of string once more; Finally according to mapping relations replaces corresponding filial generation gene.

For example, two parent strings and the matching region:

- A = 984|567|1320 B = 871|230|9546

First exchanges A and B two match regions, we obtain:

- A' = 984|230|1320 B' = 871|567|9546

Repeat traversal outside of A', B' two sub-string matching area, according to the position mapping relations of match region, exchange it one by one. For A', there is symbol map location of 2to5, 3 to 6, 0 to 7 and use 5,6,7 replace 2,3,0, out of match area of A', we will get:

- A' = 984|230|1657 B' = 801|567|9243

The best advantage of this method is that it can satisfy the chromosome constraints of no duplicate genes encoding. The primary role of crossover operator is to adjust the conflict between number of students and the number of classroom seats.

Mutation operation combined with tabu search: Mutation of genetic algorithm is the traditional method to solve schedule problem, it find two schedule unit in a line random by a certain mutation probability and swap their position and determine whether to satisfy the conflict of the classroom and course, if there is the conflict we will select repeat, until there is no conflict, it does not violate the hard constraints of classroom. This method has "global" search capability, but if the individual is too high fitness in the evolution, then the probability, which its being to select copy will be large, too many generations and inbreeding will lead the algorithm to "local convergence" rather than get the global best optimal solution. Therefore, the individual must be mutated by mutation probability, break the balance and its mutation become the key, which use GA to solve course timetabling problem.

Tabu search algorithm can accept inferior solutions in the search process; the new solution is not randomly generated in the current field, but selected the best solution, which the best solution of probability is more than other solutions. Then the search can escape from local optimal solution and turn to other areas, increasing the probability of global optimum solution for local search. We use tabu search to replace the mutation operator. The tabu search can get better individual solutions and be out of local optimum

The tabu search is applied to the local search, it can greatly avoid the phenomenon of over-mature for genetic algorithms, but it is also unwise to use tabu search frequently, it waste running time. In fact, we can call the tabu search to replace the mutation operation in appropriate time.

Chaos operation

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. Record as: t_{kl}, 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 (aperiodic) 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. Chaos cross and chaos variable specific operation as follows:

- **Step 1:** Chaos crosses

A cross interval for L_c, $L_c \in (0, 1)$, According to the current value of chaotic sequence T_{KL} whether to belong to cross interval L_c, determine whether the selected two matching chromosome for cross operation: If T_{KL} do not belong to the L_c, It is no need for cross operation; conversely to cross operation. Before cross operation, according to the number of chromosomes gene fragment N_g, interval (0, 1) is divided into several equation sub-interval and Number sub-interval; According to the current value of T_{K2} subsidiary interval number, to determine the gene location which will be carried out by crossover operation. After determining the cross position, corresponding matching chromosome gene section to carry on cross operation, produce two new chromosomes:

- Step 2: Chaotic mutation

Set variation interval is L_m, $L_m \in (0, 1)$, usually the length of interval L_m is much smaller than the length of interval L_c. According to whether the current value of the chaotic sequence t_{k3} is in the variation interval L_m, determine to do the mutation operation to the selected chromosome or not: if $t_{k3} \notin Lm$, then don't do the mutation operation, else do it. Before the mutation operation, according to the number of gene fragment in chromosome, N_g, divide the interval (0, 1) into several sub-intervals and number the sub-intervals; According to subinterval number, which the current value of T_{k4} belongs to, determine the location of the gene fragment of the mutation operation. When the location of mutation is determined, do the mutation operations to the corresponding gene segment of chromosome, then generate new chromosomes.

Step 3: According to the current value of the chaotic sequence t_{k1} and t_{k2}, do the chaos crossover operation to the chromosome of populations. All of the chromosomes generated by the chaos crossover operation form the sub-populations.

Step 4: According to the current value of chaos sequence t_{k3} and t_{k4}, do the chaos mutation operation to the chromosome of population of sub–populations.

Process of improved genetic algorithm: In this algorithm, we use genetic algorithm for global search firstly, search all of the possible search conditions, individuals of groups is distributed in most regions of the solution space, then use tabu search to search the individual schedule from of group and improve the population quality. Algorithm flow:

- Step 1: Given initial parameters (including population size, maximum number of iterations, the exchange probability, the most mature recognition parameters)
- Step 2: Determine the encoding and t = 0, c = 0, k = 0 (t is the current number of iterations, c is used to identify premature, k is cycles number)
- Step 3: Randomly generated initial population, generate N of individuals, x_B= the first individual
- Step 4: Calculate the fitness of each individual group, set the first individual to be the largest fitness individual in the generation
- Step 5: If x_B= the first individual, set c = c+1; Otherwise set the first individual to x_B, set c = 0
- Step 6: If c> the most mature recognition parameters, the individual will be ordered by descending according to fitness, part of the individual will be duplicated to the next generation, all of the others will mutate and a new generation of individuals will be appear, then go to the step8; Otherwise go to step 7
- Step 7: Selecting operation
- Step 8: Swap operation
- Step 9: Traditional GA mutation operation
- Step 10: If k = 8, tabu search Mutation operation is called to improve the quality of population points, set k = 0; otherwise go to step 11

- Step 11: If t<T or The current time is not overtime, set t = t+1, k = k+1, go to Step 4; otherwise go to step 12
- Step 12: Chaos operation
- Step 13: Determine the termination criterion

Determine the termination criterion is first: Find acceptable excellent individuals; Second: Reach the intended evolution generation; Third: The maximum fitness individual will not improve in before and after generation. Fourth: Ratios of best adapted individual in the population achieve desired data. Fifth: the running time reach the maximum time. In these criterions, No.2 and No.5 is Priority principle.

PERFORMANS EVALUATION EXPERIMENTS

Time complexity analysis: In test, we choose to two periods each morning and a time period each afternoon, fifteen time period in a week; we choose two school buildings classrooms.

We test Simulated Annealing algorithm, genetic algorithm and our own Genetic Search Algorithm. Set population size is 50, genetic generation is 50, meta course is 113, Results are as follows in Table 1.

Table 1 shows that genetic algorithms time-consuming is least under the same conditions, genetic search algorithm is slightly more, because the chaos use part of time, but the characteristic value is greatly improved.

Once again, we will compare the three algorithms, prerequisite is that adaptive parameters are same, test their number of iterations, the results are as follows in Table 2.

Thus, the number of iterations of the genetic search algorithm is at least under the same conditions, which means that its time complexity reduce.

In summary, the time complexity of genetic search algorithm is the most optimal.

Test results of genetic serch lgorithm: Set experiment l popultion to be 50, Test group 1 nd test group 2 set genetic genertion to be 100 nd 50.

- **Test one:** Estblishing popultions time is 3.775s. Estblishing 50 of chromosome, filure is 2, time is 11.466 s (Table 3).
- **Test two:** Estblishing popultions time is 4.75s. Estblishing 50 of chromosome, Filure is 6, time is 7.751 s (Table 4).

The experiment shows tht the more genetic genertion, consuming more time, but the finl fetures vlue of the best chromosome is higher. Therefore, ccording to results of the test when the genetic genertion is 50, we conclude tht hs reched the purpose of rrnging nd get better results.

Table 1: Time consumption of algorithm

Algorithm name	Population size	Genetic generation	Time (sec)	Feature value
Simulated annealing	50	50	10.86	209
Genetic	50	50	7.75	206
Genetic search	50	50	8.001	243

Table 2: No. of test iterations

Adaptive parameter	SAA	GA	GAS
0.5	334	328	323
0.9	382	356	325

Table 3: Genetic generation is 100

Genetic Operation time	Optimal chromosome Characteristics value	Time
1	185	0.70
2	188	0.70
3	190	0.70
4	191	0.76
5	194	0.72
6	195	0.70
8	196	0.70
14	191	0.79
55	204	0.71
89	206	0.70

Table 4: Genetic generation is 50

Genetic Operation time	Optimal chromosome Characteristics value	Time
1	188	0.81
2	191	0.70
3	195	0.70
4	196	0.70
5	200	0.70
8	202	0.70
9	204	0.80
12	202	0.90
21	204	0.80

CONCLUSION

Genetic serch lgorithm hs been vlidted in the test nd chieved good results. Genetic serch lgorithm is more prcticl method, convergence is fst, nd time distributes uniform. But in prctice conditions my not be terminted, the purpose is to provide different possible solutions in order for users to choose. However, if the constrint conditions re too hrsh, there my be no fesible solution, in such cse, humn intervention is necessry. Course scheduling problem is multidisciplinry hrd problems, this rticle ttempt genetic lgorithm only on the field of course scheduling, the encoding of genetic lgorithms nd prmeter settings needed to be improved nd explored, the effect on the lgorithm need be improved in further.

REFERENCES

Charlga, S.C. and Y.C. Chung, 2005. From timetable to train regulation-a new train operation model. Inform. Software Technol., 47: 575-585.

De Almeida, E.S., A. Alvaro, V.C. Garcia, J.C.C.P. Mascena and V.A. de Arruda Buregio, 2007. C.R.U.I.S.E.-Component reuse in software engineering. http://www.academia.edu/179616/C.R.U.I.S.E_-_Component_Reuse_in_Software_Engineering.

Dimopoulou, M. and P. Miliotis, 2004. An automated university course timetabling system developed in a distributed environment: A case study. Eur. J. Operat. Res., 153: 136-147.

Sorin, D.J., 2009. Fault Tolerant Computer Architecture. Morgan and Claypool Publishers, USA., Pages: 103.

Van Heiningen, W., T. Brecht and S. MacDonald, 2006. Babylon v2.0: Middleware for distributed, parallel and mobile Java applications. Proceedings of the 11th International Workshop on High-Level Parallel Programming Models and Supportive Environments, April, 2006, Rhodes Island, Greece.