The Research of Method Based on Complex Multi-task Parallel Scheduling Problem

Xiang Li, Peng Chen, and Li Zhu

China University of Geosciences
No. 388 Lumo Road Wuhan, P.R. China

Abstract. The key point of project management is scheduling problem. While scheduling optimization,which determines the profit of projects, has been one of the hot research spots for domestic and foreign experts or scholars. This paper mainly proposes an optimization method of network planning project by using an improved genetic algorithm for solving complex parallel multi-task scheduling problems. It used a method that gradually increases the number of parallel tasks for increasing the complexity of scheduling algorithm. This paper mainly focuses on the feasibility of the large-scale and complex multi-task parallel scheduling problem in the use of improved genetic algorithm. The experiment results show that using improved genetic algorithm for solving large-scale and complex multi-task parallel scheduling problem is feasible, and meanwhile,produces better results.

Keywords: genetic algorithm, multi-task parallel scheduling problem, scheduling optimization, complexity.

1 Introduction

The key point of scheduling optimization problems is network planning optimization. In the long-term practice of engineering applications, network planning optimization almost uses the network planning skills and mathematical programming skills in the field of operational research, but these methods have many deficiencies in solving complex parallel multi-task scheduling problems in large-scale network planning, they can neither bear the compute complexity nor the optimization effects are very limited. Genetic algorithm has more advantages than others, especially in the field of solving large-scale complex optimization problems; it can gain the better results. In literature from one to four, the author named Xiang Li and others proposed an improved genetic algorithm and hybrid genetical gorithm for solving the multi-task parallel scheduling problem based on resource-constrained project scheduling problems and gained better optimization results. But none of them considered the feasibility of multi-task parallel scheduling problem when the number of tasks and algorithmic complexity are increasingly getting more and more. Thus, this paper proposes an optimization method based on improved genetic algorithm to resolve the problem of large-scale complex multi-task parallel scheduling.

Z. Cai et al. (Eds.): ISICA 2009, CCIS 51, pp. 25–34, 2009.

2 Resource—Constrained Project Scheduling Model

Assuming there are several parallel tasks, and a shared resource pool containing a number of resources that can update (renewable resources), and all our resources which are supplied limitedly. In addition to sharing resources, tasks are independent with each other. To facilitate the description of the problem, the establishment of the mathematical model is as follows: RCPSP has P independent Multi-task scheduling tasks; the task k contains nk+ 1 works. The nk+ 1 task is for terminating work for virtual tasks, and it self occupies no resources and time. P missions' works are to share this kind of M renewable resources; the total for the mth resource is Rm. With Wi expressed the ith misson's work set, Wij expressed in the ith task jth work, its duration is dij, the requirement of the mth resources is rijm, the starting time is Sij, its all tight front duties formed the collection is Pij. The set composed of all tasks on time t is named It. Considering the importance of different tasks, ak is used for the weight k task. Taking these assumptions and symbols, a multi-task scheduling with resource constraints can be described as formula 1 to 6).

$$\min \sum_{k=1}^{P} \partial_k * (S_{k,n_k+1}) \tag{1}$$

$$s.\ t. \qquad S_{i,j} \geq S_{i,h} + d_{i,h} \ , \quad \forall h \in P_{ij} \ , \forall i,j \tag{2}$$

$$\sum_{w_{i,j} \in I_t} r_{ijm} \leq R_m \ , \quad \forall m,t. \tag{3}$$

$$I(t) = \{ Wkj \in \bigcup_{k=1}^{P} Wk \mid S_{kj} \leq t \leq S_{kj} + d_{kj} \ \} \tag{4}$$

$$r_{ijm} \geq 0 \tag{5}$$

$$\sum_{k=1}^{P} \partial_k = 1 \tag{6}$$

3 Algorithm Design

3.1 RCPSP Problems

As a number of tasks share common resources, so the solution is different form single tasks'. In the way to solve this problem, some documents have proposed a bilevel decision method of multi-resource constraint and multi-task scheduling problem [1]. However, this method has two major flaws. First, the prerequisite is that the duration of shared resource in each task is determined by the distributed resource. The use of shared resources and the allocation of resources were inversely related to working time. Working time = Workload + Resources. In real life, the workload is not an accumulation work. Working time = Workload /Resources. So the tasks do not accumulate in the work of the workload. The model can not be solved. The second defect is that the model adopts the specific resource allocation scheme between tasks. When resources are allocated to a particular task, it will always belong to the task; other tasks can no longer use the resources. This approach would make a larger allocation of resources wasted.

Because such a distribution strategy, There is not exist mutual adjustment of flow of resources between tasks. So when a task in which there is a sort of free resources, other tasks will not use, and it is wasted.

The literature [7] also discussed the multi-task scheduling problem and its solution. However, the solution is to add more tasks network plan to merge into a virtual work of the network plan for the Solution. This method lowers the efficiency and flexibility of the solution.

This paper presents a method based on genetic algorithm to solve tasks of the type of work for more than an accumulation of the task. Between the various tasks and resources are dynamic to the circulation of each other. It will not waste the resources. Various tasks to multi-task and do not plan to conduct any merger of the network. Only needs to input all the tasks, resources and other information, we will be able to carry out all tasks scheduling.

3.2 Data Structure for the Algorithm

Before the generation of initial population, it is first necessary to establish a data structure for storing information of the various tasks.

First, defining a task structure.

```
struct project

{int n;int TP; int TC; int TS;work *w[MAX]; entity
*E[MAX];}*pro;
```

Among them, n is the number of tasks, TP is the task completion time. TC is the calculated completion time of the task, TS is the actual completion time of the task, and work is the definition of the structure. w[MAX] task is working at the point-array. Entity is the definition of the network structure. E[MAX] is the point-array related to the each task. Dynamic information to the importation of various tasks, Cross linked-list node is used store the relationship between the various tasks.

Cross linked-list node of the structure as follows:

```
typedef struct cnode

{int hang;int lie;int value;struct cnode *down;struct
cnode *right;}crossnode;

typedef struct

{int n; crossnode *hc[MAX];}clinknode; clinknode
*Head=new clinknode;
```

‘hang’ and ‘lie’ in the structure crossnode mark the position in the linked-list. ‘value’ is the index of the work, ‘down’ is the eligible point of it and ‘right’ is the immediate point. ‘In the ‘clinknode’ hc[MAX]’ and ‘Head’ is the head note of cross linked-list.

In this paper, a key of the multi-task scheduling is for all of tasks using a cross linked-list. And the identification of various tasks is proposed in the linked-list. After the establishment of schedule, the activation of various tasks is based on the linked-list. This is an innovation of this paper.

3.3 The Framework of the Algorithm and the Code of Chromosome

Resource-constrained multi-task scheduling problems can be also considered that the scheduling problem with certain constraints. The key to solving the problem is how to use genetic algorithm to find an appropriate sequencing of multitask. According to a certain sequence of chromosome, we code all of issues on the table. If there are M tasks, each task has n works, then there m*n works. The first expression Vk is Kth chromosome in current population. Suppose Pkij is a gene of chromosome. It expressed a work of the Pkth task at location j of chromosome.

Chromosome can be expressed as follows:

Vk=[P1i1,…, P1ii,…,P1in,…,Pki1,…, Pkii,…Pkin,…,Pmi1,…, Pmii,…,Pmin]
Pkij bound by the random order with the constraint.

3.4 Initialization Population

For the Chromosomes initialization, this paper uses the method that randomly generates various initial chromosomes by the topological scheduling.

Similar with the optimal scheduling method with the shortest Resource Constrained time of the single task, the only difference is that for every work. Firstly judge the task to which it belongs, and then determine the constraint relationship. It is determined the first consider work by a Left to Right approach with a fixed sequence. At each stage of the work order has been maintained and assembled Scheduling can work and completely random from a pool selected to work in the pool. This process is repeated indefinitely until all the work was arranged.

In each iteration of the process all the work at one of the following three conditions:

1. The work has index: Part of the chromosome structure in the work.
2. The work can schedule: All immediate works are ordered.
3. Free work: All of work

v_k^t is a part of chromosome, including t activities. Q_t is in phase t with a corresponding arrangements can be made to the activities set, Pi is all the activities set ,with a corresponding set of arrangements can be made activity set Q_t is defined as formula 7.

$$Q_t = \{ j \mid P_i \subset v_k^t \} \tag{7}$$

It includes specific arrangements in the end nodes, and it is all competition activity set.

Scheduling changes which may work in the following manner: a. deleted Qt from the work j which has been selected. b. judging if there is work k in all immediate works or not, and its all immediate works are ordered . c. if there is k, it can be added to the scheduling work set Qt.

3.5 Genetic Operators Design

On the base of established cross linked-list, re-number various tasks in the work. Then apply the algorithm described in this paper. All tasks will be mixed with the work. After encoding into a chromosome, the problems become simple. Genetic chromosome operation with the single task is similar to the chromosomal genetic operations.

Participate in the two crossover operator for mothers M and fathers F. We choose a random integer q,1<=q<=J,J is chromosome length. M and F through radio spots overlapping operations produced two generations of daughters D and sons S. In the list D work, the former q tasks come from M.

$$j_i^D = j_i^M, \mathrm{i} = 1, 2 \ldots \mathrm{q}$$

i = q + 1,…J, come from the location of the F, And the relative position of the F resurveyed.

$$j_i^D = j_k^F, \mathrm{i} = \mathrm{q}+1, \mathrm{q}+2, \ldots, \mathrm{J}$$

Which $k = \min\{k \mid j_k^F \notin \{j_1^D, j_2^D, \cdots, j_{i-1}^D\}, k = 1, 2, \cdots, J\}$

S and D are similar to the formation of the linked-list, not going to repeat here.

Mutation operator adopts centralized search strategy combined with the neighborhood technology to improve the offspring. D genes will not exceed the change in the scheduling of the neighborhood gathered as scheduling x.If x neighborhood than any other solution, then x is called scheduling optimization. Mutation process is shown as follows:

Begin
If (rand () <pmutation)
Take n continuous in a row as the chromosome genes;
Permutations and combinations of the n genes;
Check each gene sequence, the genome sequence will be its discarded if it unreasonable;
Assessing all the neighborhood scheduling;
Neighborhood as the best choice for future generations;
End

In the process of mutation, two consecutive complete gene permutations and combinations. Of course, there has a violation of constrain sequence. Here can directly be discarded, but will not have an impact on the choice of the best neighborhood. This method avoids repairing the chromosome after mutation and improves the efficiency of the procedure.

3.6 Fitness Function

Calculate the shortest duration of tasks, we must first decode chromosome. Calculate the earliest starting time, and then find the earliest completion time for each task. Finally, according to the weight ratio of the various tasks calculate the shortest weighted average duration. Reached the objective function formula 8.

$$\min \sum_{k=1}^{P} \partial_k * (S_{k,n_k} + 1) \tag{8}$$

Since the task is to minimize the total built-constrained Project Scheduling Problem issues. We change the original objective function to ensure that the individual is suited to meet the greatest value. The current population vk is located k chromosome; g (vk) is a fitness function. f (vk) is the original target value. fmax and fmin are the biggest target value and the minimum target value. Conversion methods such as formula 9.

$$g(v_k) = \frac{f_{\max} - f(v_k) + \varepsilon}{f_{\max} - f_{\min} + \varepsilon} \quad (9)$$

Where ε is a positive number, often limited the open interval (0,1).

3.7 Selection Operator

The operation that winning individuals are chosen from the group and the bad individuals are eliminated is named selection.

Selection is based on the currently popular "breeding pool (Breeding Pool) choice", "To adapt to-value ratio of options", "ranking selection (Ranking Selection)", "Mechanism based on local competitive choices", and so on. Roulette choice (Roulette Wheel Selection) is based on fitness than A choice of the most widely used methods. It is the first calculation of the relative fitness of individual fi/∑fi, named Pi, and then chooses probability {pi. = 1,2,…, N} a disk cut into N copies, 2лPi fan angle of the center of Pi. In making the selection, rotating disk can be assumed that if a reference point to fall into the itch fan, then chose individual i. Generation a random number r in [0, 1], if Po+Pi+… +Pi-1<r<Pl+P2… +Pi, i individual choice, the assumption here Po=0. It is easy to see that this was very similar to roulette choice of trouble. For sector bigger area, it has the greater the probability fall in and it was an opportunity to be chosen.

Thus, the structure of the gene was likely to pass it on to the next generation greater. This paper discusses the use of multi-robin scheduling optimization algorithm choice. Every generation of new groups in the use of optimal preservation strategy that will preserve the best, so far in the contemporary individual and overcome random sampling error.

4 Experiments and Analysis

4.1 Experimental Environment

This paper is to implement the design of optimization algorithm, establishment of optimization algorithm model and the realization of optimization algorithm. The specific experimental environment is shown as table 1.

Table 1. Development environment

Operating System	Windows 2003 server
Programming Tools	Visual C++ 6.0
Processor	Quad-Core AMD Opteron(tm) Processor 2384 2.75GHz
Memory	4G

4.2 Experimental Data

The examples of experimental cases in this paper are based on multi-task parallel scheduling problem. Assuming that every task has a same network structure, the structure is shown as figure 1.Every task has 1202 works and 800 items, the whole 1202 works are to share the whole 20 kinds of resources., the resource have mentioned is single model, every work uses one of the 20 kinds of resources, and has different construction period and requirements, which are shown as table 2.The type and quantity of resource are shown as table 3.

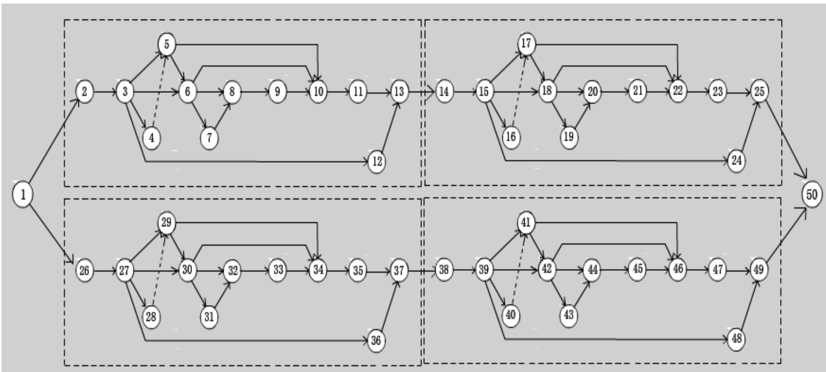

Fig. 1. The map of network structure of project

Table 2. Type and quantity of resource and operation time required by 50 items or 75 works

Serial number	Starting item	Finshing item	Time(s)	Resource type	Quantity
1	1	2	2	1	7
2	1	26	2	1	7
3	2	3	3	2	9
……	……	……	……	……	……
73	47	49	3	11	20
74	48	49	5	8	11
75	49	50	3	16	5

Because each 50 items and 75 works in each task used in this paper based on the examples of experimental cases have same type and quantity of resource, table 1 only make a list of type and quantity of resource required by each 50 items and 75 works in each task. This paper simplify each 50 items and 75 works as one work to operate, simplified network structure is shown as figure 2.

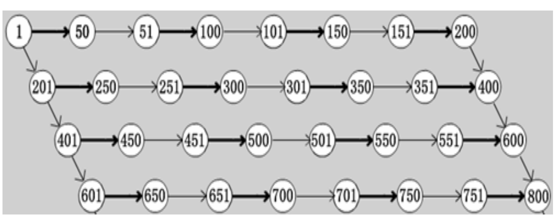

Fig. 2. The brief map of net structure

Table 3. Type and quantity of resource

Serial number	Quantity
0	40
1	50
2	30
……	……
17	40
18	30
19	50

4.3 The Experimental Results and Analysis

Using the improved genetic algorithm to solve large-scale complex multi-task parallel scheduling problem mentioned in this paper, it assumes that population size as 150, the maximum evolution population as 100, crossover rate as 0.8, mutation rate as 0.01, and neighborhood length of mutation as 4.The experiment is starting from one task (800 items and 1202 works) to 100 tasks with gradually increasing the number of tasks. In the first 25 task, increasing the number of tasks with 1 task and increasing the number of tasks with 10 tasks from the 30th to 100th work. In other word, starting from one task including 800 items and 1202 works to 100 tasks including 80000 items and 120200 works. The relationship between operation time and number of tasks are shown as figure 3 and figure 4.

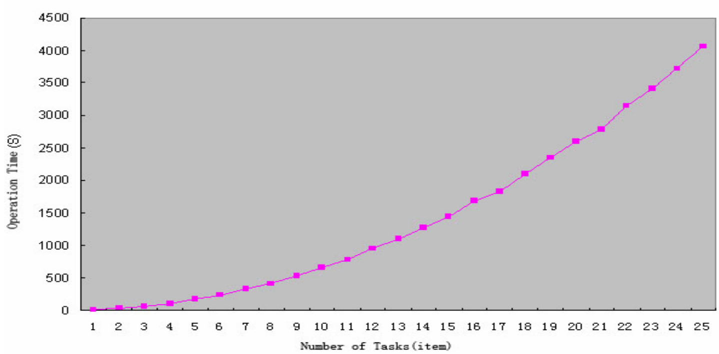

Fig. 3. Coordinate graph between operation time and task number with a total of 25 tasks

Coordinate graphs between operation time and task number are shown as figure 3. The number of tasks is from 1 to 25 with one increasing step. From the figure 3 we can infer even though the task number reaches 25 including 20000 items and 30050 works, the operation time is only 4065.80 seconds or 1.13 hours.

Coordinate graphs between operation time and task number are shown as figure 4. The number of tasks is from 1 to 100 with ten increasing step. From the figure 4 we can infer even though the task number reaches 100 including 80000 items and 120200 works, the operation time is only 65078.28 seconds or 18.07 hours.

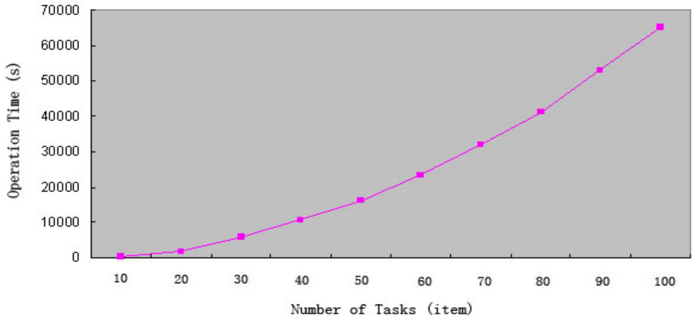

Fig. 4. Coordinate graph between operation time and task number with a total of 100 tasks

From figure 3 and figure 4, we can infer that with the increasement of task number and algorithmic complexity, the operation time obtained by using improved genetic algorithm to solve large-scale complex multi-task parallel scheduling problem mentioned in this paper basically present the trend with increasing linearly. If increasing the number of tasks and algorithmic complexity, it is feasible to solve large-scale complex multi-task parallel scheduling problem using the algorithm mentioned in this paper.

5 Conclusion

This paper proposed a new genetic algorithm to solve large-scale complex multi-task parallel scheduling problem of resource constraints. It mainly studied the feasibility of large-scale complex multi-task parallel scheduling problem. The experimental results show that using improved genetic algorithm to solve large-scale complex multi-task parallel scheduling problem mentioned in this paper linear increased in the basic, and gained better results. It overcame the deficiencies that other algorithms could only solve small-scale and low algorithmic complexity multi-task parallel scheduling problem. Further, it also show that using improved genetic algorithm had more great advantages than others in the field of solving large-scale complex multi-task parallel scheduling problem.

References

1. Li, X., Tan, W., Kan, L.: Research of Resource Equilibrium Optimization Based on Genetic Algorithm. Computer Engineering and Design, 4447–4449 (2008)
2. Li, X., Tong, H., Tan, W.: Network Planning Multi-objective Optimization Based on Genetic Algorithm. In: International Symposium on Intelligence Computation and Applications Progress, pp. 143–147 (2007)
3. Li, X., Tan, W., Tong, H.: A Resource Equilibrium Optimization Method Base on Improved Genetic Algorithm. China Artificial Intelligence Progress 2, 737–743 (2007)
4. Lova, A., Tormos, P., Cervantes, M., Barber, F.: An efcient hybrid genetical gorithm for scheduling projects with resource constraints and mulitiple execution modes. Int. J. Production Economics 117, 302–316 (2009)

5. Li, X., Chen, Q., Li, Y.: Impact on Genetic Algorithm of Different Parameters. In: The 3rd International Symposium on Intelligence Computation and Applications, pp. 479–488 (2008)
6. Xiang, L., Yanli, L., Li, Z.: The Comparative Research of Solving Problems of Equilibrium and Optimizing Multi-resources with GA and PSO. In: 2008 International Conference on Computational Intelligence and Security (2008)
7. Liao, R., Chen, Q., Mao, N.: Genetic algorithm for resource - constrained project scheduling. Computer Integrated Manufacturing Systems 10(7) (July 2004)