Available online www.jocpr.com

Journal of Chemical and Pharmaceutical Research, 2015, 7(4):303-306

Research Article

ISSN : 0975-7384
CODEN(USA) : JCPRC5

The grid resource assignment research using genetic algorithm

Guo-Ping Zou

School of Mathematics and Computer Sciences, Xinyu University, Xinyu, P. R. China

---

ABSTRACT

A novel genetic algorithm is proposed to the grid resource assignment problem. The improved genetic algorithm extracts the heuristic feedback from these obtained solutions, and applies the heuristic feedback to guide the subsequent optimization process. Experimental results suggest that this method is efficient to the resource optimization problem.

Key words: Genetic Algorithm; Grid Resource; Assignment

INTRODUCTION

With people's endeavor in this aspect, techniques like parallel technique, cluster technique and distributed technology are generated [1]. However, these techniques can only help people use computing resources within a certain scope which is the range of management domain. Nevertheless, resources that use these techniques and can be shared are limited and overall strength of aggregation is not powerful enough [2]. As Internet and Web technique become mature as well as are popularized and applied, people have the idea that internet resources can be integrated for use and they want to use existing internet facilities to establish a kind of new infrastructure, integrate all kinds of computing resources and provide good interfaces when these resources are used for users in the whole world. This new infrastructure is Grid [3].

Grid utilizes existing internet framework, integrates all kinds of resources which are widely distributed geographically, including computing resources, storage resources, bandwidth resources, software resources, data resources, information resources and knowledge resources, into a logic whole (or called a 'virtual super computer'), provides users with application service like integrative information, computing, storage and visit, realizes resource sharing and cooperative work ultimately and eliminates resource 'islet' completely [4-5]. The virtual super computer organized in this way has two advantages, i.e., super-strong data handling capacity and ability to make full use of idle resources on internet [6]. Currently, grid technology has become a hot spot and frontier domain of domestic and foreign researches so that it is praised as the third information technology wave after internet and Web.

PROBLEM FORMULATION

Activity is the fundamental element in the practical engineering, suppose there are totally N activates and M persons. Suppose that T_{di} , T_{di}^{c} and T_{di}^{n} denote the period of activities, the time remainder of activities and the actual completion time of activities Respectively. $Role_{i,j}$ means the i^{th} activity can be done by the j^{th} person. $RoleNum_i$ is the completion person number of activities. T_D and T_N are the planned completion time and the actual completion time of projects. C_D and C_N denote the planned cost and the actual cost of projects. $f_T(EM)$ and $f_C(EM)$ denote the completion date function and the cost function of projects.

The cost of software process contains: management fee and development fee. The former is daily cost C_1 which maintaining software development and cost of administrators C_2 . The latter points at cost of device resources C_3 and development cost C_4 of different developers with various abilities and labor-hours.

$$\begin{cases} CN = C_1 + C_2 + C_3 + C_4 \\ C_4 = \sum_{i=1}^{M} c_i T_i \end{cases} \quad (1)$$

Here, c_i and T_i denote the development cost and labor-hour of developers respectively. Cost and construction period is the main attributes of software process. Assume the sum weight of cost and period as optimized objective function, the definition of which is as shown in the following equation. In which, C means cost, T signifies the weight value of construction period x_1, x_2 determined by demand of decision makers, EM is the executive matrix.

$$\max Fitness(C,T) = x_1 * C + x_2 * T$$
$$s.t. \begin{cases} C = f_C(EM) \\ T = f_T(EM) \\ 0 \le x_1 \le 1 \\ 0 \le x_2 \le 1 \\ x_1 + x_2 = 1 \end{cases}$$

IMPROVED GENETIC ALGORITHM

The improved genetic algorithm is characterized by the extraction and application of heuristic feedback in the evolution process. In this paper, the near-optimal solutions obtained throughout the search are analyzed to extract the heuristic feedback, and then the obtained heuristic feedback is used to guide the subsequent search. The computational flow of IGA is shown in Fig. 1.

(1) Heuristic Feedback. The first kind of heuristic feedback is called the activity assignment position which is applied to establish a beneficial order for the given activity. A matrix HF_1 with size $N \times N$ is defined for the activity assignment position, $HF_1(i, j)$ denotes the total number of times of assigning the activity i to the j^{th} position among the near-optimal solutions obtained throughout the search. The second kind of heuristic feedback is called the activity assignment person which is applied to establish the beneficial person for one given activity. A matrix HF_2 with size $N \times M$ is defined for the activity assignment person, $HF_2(i, j)$ denotes the total number of times of assigning the activity i to the j^{th} person among the near-optimal solutions obtained throughout the search.

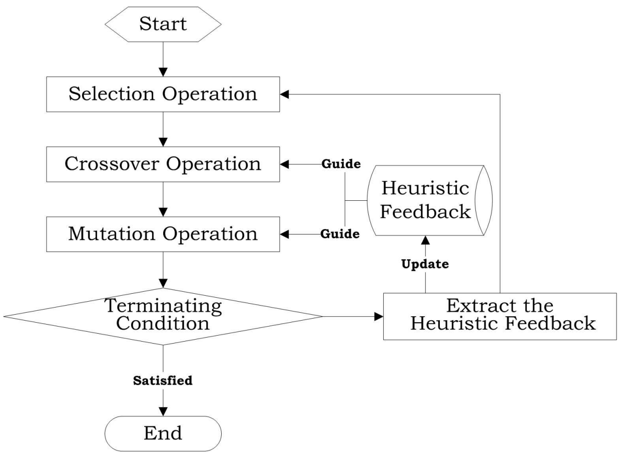

Fig. 1. The computational flow of IGA

(2) Application of Heuristic Feedback. In IGA, the activity assignment position is applied to guide the crossover operation. The activity assignment position is employed to determine one beneficial position for the given activity. To the activity assignment position matrix displayed in Table 1, if we want to determine the beneficial position for activity 3, then we can obtain the following probabilities, and the beneficial position to activity 3 is decided by a random way with the following probability distribution.

Table 1 An Example of Activity Assignment Position

3	5	4	2	1	0
1	5	6	2	1	0
0	1	2	8	2	2
0	2	2	6	5	1
0	0	4	5	6	0
0	0	2	3	4	6

$$\text{Position } 1: \frac{0}{1+2+2+8+2} = 0$$
$$\text{Position } 2: \frac{1}{1+2+2+8+2} = 0.07$$
$$\text{Position } 3: \frac{2}{1+2+2+8+2} = 0.13$$
$$\text{Position } 4: \frac{8}{1+2+2+8+2} = 0.53$$
$$\text{Position } 5: \frac{2}{1+2+2+8+2} = 0.13$$
$$\text{Position } 6: \frac{2}{1+2+2+8+2} = 0.13$$

In IGA, the activity assignment person is applied to guide the mutation operation. The activity assignment person is employed to determine one beneficial person for the given activity. To the activity assignment person matrix displayed in Table 2, if we want to determine the beneficial person for activity 6, then we can obtain the following probabilities, and the beneficial person to activity 6 is decided by a random way with the following probability distribution.

Table 2 An Example of Activity Assignment Person

4	2	5	2	4	6
5	7	5	11	6	6
6	6	5	2	5	3

$$\text{Person } 1: \frac{6}{6+6+3} = 0.40$$
$$\text{Person } 2: \frac{6}{6+6+3} = 0.40$$
$$\text{Person } 3: \frac{3}{6+6+3} = 0.20$$

(3) Updating of Heuristic Feedback. After each generation, if the global optimal solution (the best solution from the start) was obtained at this iterative, then the knowledge level will be updated by the following rule, which is based on the optimal solution to accomplish knowledge updating. If the activity i to the j^{th} position among the best solution, then

$$HF_1(i,j) = HF_1(i,j) + 1 \quad (2)$$

If the activity i to the j^{th} person among the best solution, then

$$HF_2(i,j) = HF_2(i,j) + 1 \quad (3)$$

RESULTS

The IGA was implemented using Visual C++ language, and executed on a personal computer with the 2 GHz processor and 2GB memory. In this paper, the final experimental results were averaged over 30 trials, and 10 testing instances were randomly produced to validate the performance of our approach. The optimal objectives obtained by the IGA are summarized in Table 3. From the experimental results of Table 3, we can see that, there exists the small gap among different trials. Experimental results suggest that it is efficient to the given problem.

Table 3. The Final Experimental Results

SN	*N*	*M*	*Cost*		*Time*	
			Avg.	*Std.*	*Avg.*	*Std.*
1	20	5	55.6	0.28	22.9	0.15
2	20	5	86.9	0.15	25.6	0.19
3	20	5	75.3	0.31	26.7	0.21
4	50	10	128.6	1.26	42.5	0.59
5	50	10	135.6	2.14	43.8	0.61
6	50	10	129.4	2.09	46.9	0.63
7	200	30	658.1	12.58	59.8	1.25
8	200	30	682.7	15.49	61.2	1.38
9	200	30	648.2	10.87	62.7	1.09
10	200	30	684.1	11.68	59.4	1.53

CONCLUSION

The contribution of this paper can be summarized as follows: A novel genetic algorithm is proposed to the resource optimization problem. Experimental results suggest that it is efficient to the given problem.

REFERENCES

[1] Lin C; Hu J; Kong XZ, *Chinese Journa*., **2012**, 35(1), 1-15
[2] Hu CP; Deng SL, *J. Intell*., **2006**, 24(3), 321-325
[3] L Dudek, *ACM Trans. Comput. Hum. Interact*., **2006**, 18(1), 1-30
[4] Li XQ, *J. Intell*., **2010**, 28(5), 763-767
[5] Pang QH, *Computer Engineering*, **2007**, 33(18), 59-61
[6] Zhou RG, *Computer Engineering and Applications*, **2007**, 43(31), 102-105