Solving the Scheduling Problem in Computational Grid using Artificial Bee Colony Algorithm

Seyyed Mohsen Hashemi[1] and Ali Hanani[2]

[1] **Assistant Professor, Computer Engineering Department, Science and Research Branch, Islamic Azad University, Tehran, Iran.**
hashemi@srbiau.ac.ir

[2] **Computer Engineering Department, Songhor and Koliaei Branch, Islamic Azad University, Songhor, Iran.**
ali_hanani@yahoo.com

Abstract

Scheduling tasks on computational grids is known as NP-complete problem. Scheduling tasks in Grid computing, means assigning tasks to resources such that the time termination and average waiting time criteria and the number of required machines are optimized. Based on heuristic or meta-heuristic search have been proposed to obtain optimal solutions. The presented method tries to optimize all of the mentioned criteria with artificial bee colony system with consideration to precedence of tasks. Bee colony optimization is one of algorithms which categorized in swarm intelligence that can be used in optimization problems. This algorithm is based on the intelligent behavior of honey bees in foraging process. The result shows using bees for solving scheduling problem in computational grid makes better finish time and average waiting time.

Keywords: *artificial bee colony; Grid scheduling; communication cost; precedence right.*

1. Introduction

Grid computing features important role in accelerating computational operations. Through these systems one or more tasks are run on a couple of machines in parallel. Computational Grids enable the sharing, selection, and aggregation of geographically distributed resources for solving large-scale problems in science, engineering, and commerce. As the resources in the Grid are heterogeneous and geographically distributed with varying availability and a variety of usage and cost policies for diverse users at different times and, priorities as well as goals that vary with time. The management of resources and application scheduling in such a large and distributed environment is a complex task [1]. Therefore, using heuristic methods to solve the scheduling problem is a very common and also acceptable approach in these systems [2]. Resource management and task scheduling are very important and complex problems in grid computing environment. It is necessary to do resource state prediction to get proper task scheduling Relation among the components of a parallel process is demonstrated by a directional acyclic graph named the task graph. In this graph each node is dedicated to a specific task. The operation time of task i is shown by the node weight Wi. The communication cost between the two tasks i and j is shown by Cij. This cost is computed while two tasks get operated on two different processors otherwise is not considered. The aim of solving multi-processor time scheduling problem is the reduction of tasks operation time on a limited and determined number of processors [3].

Different method and algorithms are presented for solving job scheduling up to now [4-6]. In all of the proposed methods the grid resources are heterogeneous. Some algorithms have considered communication cost and some others have accentuated the precedence of work operations. Of course considering communication cost in scheduling can make the solution applicable for using in media like Grid and Cloud. Time scheduling problems contains various types where a lot of papers have presented about. Alahverdi with some others [7] have collected different methods. The main diversity of the problems presented are the approach of task dedication, type of processors, communication cost existence and precedence among tasks. The cited paper is about scheduling algorithms not exactly in Grid system. From the most famous heuristic algorithms in solving scheduling problem are the genetic algorithm PGA [8] and optimization algorithm of ant colony named Antls [9] and also bee colony optimization[3] that presented by some authors of this paper for multiprocessor systems. The PGA algorithm makes chromosomes by the use of coding based on precedence. The PGA algorithm tries to improve task termination time using the two actions as combination and mutation. The Antls algorithm which is one of the ant colony optimization solves the problems using scheduling list. This paper tries to present an appropriate solution according to artificial bee colony algorithm. In advance through the second part task scheduling problem in distributed systems is considered. In the third part, the bee behavior in nature and artificial bee colony algorithm are explained. In advance and in the forth part the offered solution is presented. At the end and in the fifth part the obtained results are presented and discussed.

2. Task Scheduling Problem in Grid computing

Scheduling jobs on computational grids is considered as NP-complete problem. In Grid environment, the resource scheduler is one of the most critical components of the grid middleware. There is three main tasks that broker have to done:

1) *Resource discovery and selection*
2) *Job scheduling*
3) *Job migration [10].*

In this article we focus on the second step that explain how to map pending jobs to specific physical resources with consideration to optimize total finish time and waiting time criteria.

Task scheduling problem in a system containing m resources consists of dedicating task to machines or other resources like this so that the precedence relation among tasks is retained and also all of the tasks complete in the minimum possible time according to the following mathematical formulation (1):

$$\begin{aligned} \min \quad & f = \max_{j} \{t_j\} \\ \text{s.t.} \quad & t_k - p_k - d_{jk} \geqslant t_j, \quad T_j \succ T_k \ \ \forall j,k, \\ & t_j \geqslant 0 \quad \forall j. \end{aligned} \tag{1}$$

In this formula, the index i, is for depicting number of machines. j and k are used for the number of tasks where f shows the task termination time. Ti shows the jth task termination time where djk denotes the communication cost among tasks i and j and pk shows the process time of task k. also Ti>Tj, shows that the task Ti should be operated before task Tj [3] . The problem of task scheduling is indicated by a directed acyclic graph (DAG). This graph is shown by G (V, E, w, c) which has four characters that are:

V is the set of vi nodes, and each node vi represent a task W is a V computation costs array in which each wi gives the estimated time of task execution, E is the set of communication edges. The directed edge eij joins node vi and vj and C is the set of communication costs. In figure 1 we represent a sample of a DAG.

3. The behavior of bees in the nature

The social colony of insects like ants and honey bees have instinctive intelligence is famous as collective understanding. This organized behavior enables the colony to solve the problems with the assistance of behavior of the group. A bee colony prospers by deploying its foragers to good fields. In principle, flower patches with plentiful amounts of nectar or pollen that can be collected with less effort should be visited by more bees, whereas patches with less nectar or pollen should receive fewer bees [11].

Each bee hive has a place which is called dance floor. Every Bee starts to dance after when it came back to its hive from a foraging. The main purpose of this kind of dancing is to convince the other bees to be accompanied by them [12].

In a society of honey bees, the forager bees search for finding the flower paths and if they find a suitable food source, they share that place as common with other bees. While the forager bee comeback to the cave, they share the information of the food sources with other bees by a movement named waggle dance. The studies on the dance of the bees show that while operating the waggle dance some information like direction, distance, quantity and quality of the food sources are shared with respect to other bees.

In the midst of the information collection and some other work like extracting from the food source, each bee do a part of the work because of its specific behavior. Generally in the midst of food foraging, the bee colony includes two types of bees. Unemployed foragers and employed foragers.

3.1 Unemployed foragers

When a bee starts to search for food resources it has no information about the food source in the look-up environment, this bee initiates its search as a free or jobless bee. Generally there are two types of unemployed forager.

A) Scout bee: If a bee starts to search singularity without any information then this bee is named as a scout bee. The number of scout bees varies between 5 to 30 percent according to the cave information. The average number of these bees in various conditions is about 10 percent.

B) Recruit bee: If a bee watches another insect dance then it starts to browse according to the information of the waggle dance. These bees are named recruit bees.

3.2 Employed foragers

While a recruit bee finds nectar and extracts it, this bee is a forager browser bee. This bee memorizes the place of food source. After that comes back to the hive and empties the collected nectar in the place of the hive. At this stage according to the amount of remained nectar in the food source, three states are possible:

- If the existed nectar in the food source has become very small or finished, the forager browser bee puts the food source aside and is changed to a jobless or free bee.

- If there still is enough nectar in the food source then the bee can continue to browse with its hive mates without sharing food information.
- If there is enough nectar in the food source, the bee can go to the dance area and operate waggle dance so as to share the information of food source with its hive mates.

4. Artificial Bee Colony Algorithm

The probability of each of these states depends perfectly on the food source quality [11]. The colony algorithm of figurative bees has tried to solve complicated problems by simulating the behavior of these bees. Pseudo code stages of the main algorithm are as beneath (2).

```
Send the scouts onto the initial food sources
    REPEAT
  1. Send the employed bees onto the food sources and
     determine their nectar amounts
  2. Calculate the probability value of the sources with
     which they are preferred by the onlooker bees
  3. Stop the exploitation process of the sources
     abandoned by the bees
  4. Send the scouts into the search area for
     discovering new food sources, randomly
  5. Memorize the best food source found so far
    UNTIL (requirements are met)
                                                    (2)
```

5. The Suggested Solution for Time Scheduling Problem

For using artificial bee algorithm to schedule the grid jobs at the first stage all of the used values are determined which contains the problem size, DAG matrix, and other used parameters in the algorithm which will be explained. Then first scout bees begin to build the initial solutions and after that the browser worker and forager bees start to evaluate these solutions. This action is repeated until the termination condition is desired. These stages are explained in advance.

5.1 Constructing solution (scout bees)

At first, each of the pre-active bees get one of the root tasks in DAG randomly. Root tasks have no pre-needed task initially. The way tasks locate in DAG and the approach of exerting precedence are depicted in Fig.1.

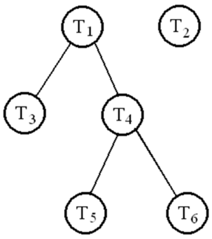

Figure 1. Tasks in a DAG with their priorities

Then one task is chosen among the eligible tasks randomly and using minimum termination time algorithm, dedicate the task to one of the resources. This stage is repeated until all of the tasks are assigned to resources. After that, the quality of each solution is evaluated by (3) in which *Fit* denotes the relevance of the solution and *finisheTime* shows the end of scheduling termination. In advance after evaluating all of the constructed solutions, best scout bees are selected and change to forager bees.

$$Fit = \frac{1}{FinishTime} \tag{3}$$

5.2 Attraction (forager bees)

In the attraction phase, each forager bee shows its solution with waggle dance and with this method tends and employs some of the forager bees for the way that its found. The probability of absorption each employed bee is determined by forager bees and calculated by the formula(4).

$$R_K = \frac{Fit_K}{\sum_{e=1}^{n} Fit_e} \tag{4}$$

In this formula R_k is the probability of each employee bee by kth forager bee and Fit_k denotes the solution relevance of number k forager bee where n is the number of all forager bees.

5.3 Foraging (employee and forager bees)

A solution that constructed by a bee schedules all of the tasks that represented in DAG and actually is a full solution that involves all tasks. A sequence of task operation is determined in each solution. The solution in witch is obtained by a task replacement operation on the bee main solution is named near neighborhood of the bee main solution. The decision of each bee to change its near neighborhood tasks results in construction of new solutions. These new paths are neighborhood solutions with bee main solution. Each bee tries to browse neighborhood solutions with its solution by operating a determined number of displacement operations on the tasks which are possible to be displaced. Each of the tasks is dedicated to processors using minimum termination time algorithm after the determination of tasks sequence in the new solution of each bee.

5.4 Information updating

At the updating stage, the relevance of all constructed neighborhood solutions by the bees are calculated by bees respectively based on termination time, average waiting time and the required number of machines. If the relevance of the constructed neighborhood solution by the bee is better than the relevance of the main solution of the same bee then this bee changes its place with the bee which has presented the main solution and from this time on continues its job as a forager bee [3].

6. Simulation and Evaluation of Offered Algorithm

For evaluating the algorithms we use the problems with different sizes. Also we make the DAGs with Gaussian estimation method. All of the simulations are written in c# language and operated on the serial processor Intel 3.02 GHz. Through all of the presented algorithm operations, the number of forager bees was 50. The number of displacement operations is considered to be 5 while the population of worker and pre-active bees is zero. Best obtained results after 20 algorithm iteration are presented in tables 1 and chart 1 and 2. In these tables the best results obtained from the recommended method along with the obtained results out of genetic algorithm (PGA) [7] and an algorithm based on ant colony (ANTLS) [8] are presented.

Table 1: Comparing proposed algorithm with ANTLS and PGA with considering 18 tasks

Total processors	Number of tasks	Proposed algorithm			ANTLS			PGA		
		Makespan	*AWT*	*Number of used processors*	*Makespan*	*AWT*	*Number of used processors*	*Makespan*	*AWT*	*Number of used processors*
2	18	440	268.88	2	470	263.88	2	440	269.44	2
4	18	470	271.66	4	510	263.33	4	440	269.44	4
6	18	470	271.66	4	530	282.22	5	440	269.44	6
3	33	890	516.66	3	950	516.33	3	1030	512.89	3
6	33	890	509.66	6	950	506.96	5	980	509.74	6
9	33	890	507.87	7	950	489.09	7	950	506.96	9
5	52	1440	794.80	5	1490	823.07	5	2010	1065.38	5
8	52	1450	784.42	8	1620	953.65	8	1830	986.56	8
11	52	1450	782.88	9	1660	926.15	11	1790	1056.43	11
5	75	2080	1131.86	5	2340	1453.86	5	2950	1684.69	5
8	75	2050	1098	8	2350	1546.93	8	2950	1679.75	8
11	75	2050	1086.26	11	2390	1504.53	11	2950	1680.44	11
6	102	2820	1492.10	6	3440	2063.62	6	3840	2646.89	6
10	102	2740	1443.3	10	3250	2014.90	10	3780	2523.47	10
14	102	2740	1435.4	13	3340	2070.19	14	3720	2546.88	14

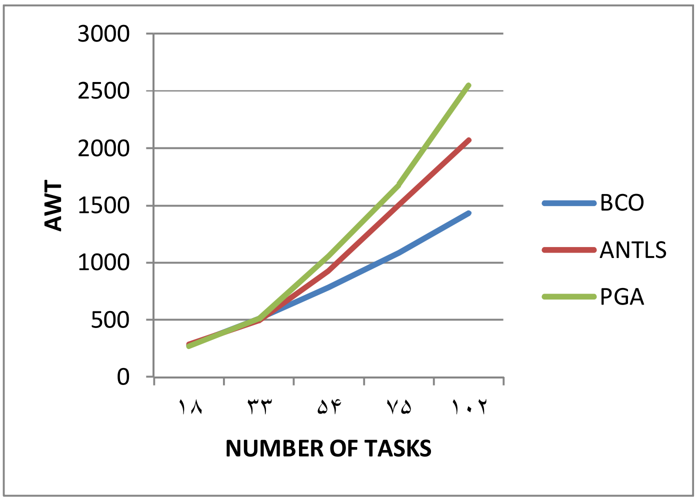

CHART1. COMPARING AWT PARAMETER FOR proposed algorithm(bcom) with ANTLS and PGA

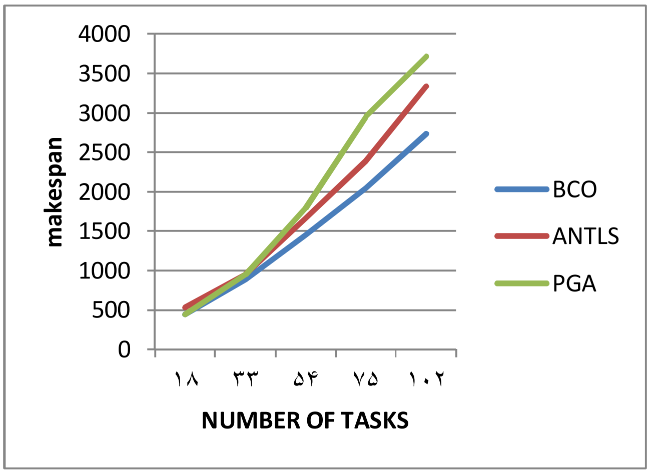

CHART2. COMPARING FINISH TIME PARAMETER FOR PROPOSED ALGORITHM(BCOM) WITH ANTLS AND PGA

7. Consequences and related work

At this stage a new solution is mooted for solving scheduling problem in Grid systems. As is clear from the tables 1 and charts, the offered solution through this paper gives better results in all of the tested items. This algorithm reduces the average waiting time and number of required processors in addition with optimizing termination time. This algorithm has no memory overflow due to using no extra memory in comparison other algorithm that purposed in [3]. Therefore the presented algorithm operates better than the compared algorithm from the aspects of algorithm operating speed, consumed memory and the results it obtains. This idea can uses for same problem in Cloud Computing issue as a new research interest with other parameters.

References

[1] S. Hashemi and A. Khatibi, “Cloud Compitung Vs. Grid Computing”, ARPN Journal of Systems and Software, VOL. 2, NO.5, MAY 2012.

[2] Z. Mousavinasab, R. entezari-malaki and A. Movaghar,”A Bee Colony Task Scheduling Algorithm in Computational Grids”, Communications in Computer and Information Science Volume 188, 2011, pp 200-210.

[3] A. Hanani, S. Nourossana, H. Haj Seyed Javadi and A.M. Rahmani, “Solving the Scheduling Problem in multi-processors Systems with Communication Cost and Precedence using Bee Colony Systems”, ICACTE 2010, in press.

[4] T. Thanalapati and S. Dandamudi, “An efficient adaptive scheduling scheme for distributed memory multicomputer”, IEEE Transactions on Parallel and Distributed Systems;12(7):758–68, 2001.

[5] M. Guntsch and M. Middendorf, “A population based approach for ACO”, In S. C. et al. editor, Application of Evolutionary Computing – Evo Workshop: Evo COP, EvoIASP, EvoSTIM/ EvoPLAN, number 2279 in Lecture Notes in Computer Science, pp. 72-81, Springer Verlag, 2002.

[6] N. Nissanke, A. Leulseged, and S. Chillara, “Probabilistic performance analysis in multiprocessor scheduling”, Journal of Computing and Control Engineering Journal vol. 13, Aug. 2002, pp. 171-179.

[7] A. Allahverdi, C.T. Ng, T.C.E Cheng, and M. Kovalyov, “A survey of scheduling problems with setup times or costs”, European Journal of Operational Research, vol. 187, June. 2008, pp. 985–1032, doi: 10.1016/j.ejor.2006.06.060.

[8] R. Hwang, M. Gen and H. Katayama, “A comparison of multiprocessor task scheduling algorithms with communication costs”, Computers and Operations Research, vol.35 n.3, page. 976-993, March. 2008.

[9] M. Bank, U. Honig, and Schiffmann, W “An ACO-based approach for scheduling task graphs with communication costs”, Proceedings of the 2005 International Conference of Parallel Processing (ICPP’ 05), Oslo, 2005.

[10] R. Moreno and A. Alonso-Conde, “Job Scheduling and Resource Management Techniques in Economic Grid Environments”, Across Grids 2003, LNCS 2970, pp. 25–32, 2004.

[11] C.S. Chong, Y.M Hean Low and etc, “A bee colony optimization algorithm to job shop scheduling”, Proceedings of the 2006 Winter Simulation Conference, pp. 1954-1961, 2006.

[12] V. Arabnejad, A. Moeini and N. Moghadam,” Using Bee Colony Optimization to Solve the Task Scheduling Problem in Homogenous Systems”, IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 5, No 3, pp. 348-353,September 2011.