MULTI OBJECTIVE BEE COLONY OPTIMIZATION FRAMEWORK FOR GRID JOB SCHEDULING

Sana Alyaseri[1] and Ku Ruhana Ku-Mahamud[2]

[1]Nazwa College of Technology, Oman, sana.alyaseri@nct.edu.om
[2]Universiti Utara Malaysia, Malaysia, ruhana@uum.edu.my

ABSTRACT. Grid computing is the infrastructure that involves a large number of resources like computers, networks and databases which are owned by many organizations. Job scheduling problem is one of the key issues because of high heterogeneous and dynamic nature of resources and applications in the grid computing environment. Bee colony approach has been used to solve this problem because it can be easily adapted to the grid scheduling environment. The bee algorithms have shown encouraging results in terms of time and cost. In this paper a framework for multi objective bee colony optimization is proposed to schedule batch jobs to available resources where the number of jobs is greater than the number of resources. Pareto analysis and k-means analysis are integrated in the bee colony optimization algorithm to facilitate the scheduling of jobs to resources.

Keywords: Multi objective bee colony optimization, Grid scheduling, Pareto analysis, k-means algorithm

INTRODUCTION

One of the key challenges for the grid computing researchers is to find the efficient allocation of resources to the jobs submitted by users. Job scheduling is the process of mapping jobs into particular available resources. Therefore, the coordination and allocation of resources for efficient execution of user jobs is the main objective of grid job scheduling problem. Sometimes there is a data dependency between submitted jobs and in all cases the submitted jobs require diverse resources like computation resources, storage resources and specific instruments (Christodoulopoulos, Sourlas, Mpakolas & Varigos, 2009).

Grid scheduling is a very complicated problem and is considered as multiple objective functions problem. The resource providers commonly have different motivations when connected to the grid. These motivations are represented by objective functions in scheduling (Zhu & Lionel, 2013). The objective functions can be divided into two types, namely, application centric and resource centric (Pop & Cristea, 2009). The scheduling algorithms adopting the application centric objective plan to optimize the performance of each individual job which leads to reduce makespan. Makespan is the time spent from the beginning of the first task of the job to the end of the last task of the job (Abdullah, Othman, Ibrahim & Subramaniam, 2007). Most scheduling strategies in grid try to reduce the makespan, which is commonly used as an objective function in many scheduling algorithms.

Honey bees are one of the most well studied social insects. In the last few years a lot of research based on different bee behaviors has been carried out to solve complex combinatorial and numerical optimization problems (Bitam, Batouche & Talib, 2010). Many features of

bees behavior like their memories, navigation systems, group decision making processes, the bee dance (communication) and bee foraging have inspired the researchers to mimic them in algorithms (Karaboga & Akay, 2009).

In this paper bee colony optimization (BCO) will be enhanced to be more efficient to optimize job scheduling in batch mode. A multi objective frame work will be proposed to tackle job scheduling problem in the grid. The paper is organized as follows. Related work is described in Section 2 while the bee colony optimization is shown briefly in Section 3. Section 4 explains the job scheduling problem. The proposed frame work is presented in Section 5. Finally, the conclusion and future work are showed in Section 6.

RELATED WORK

A large variety of approaches have been presented to tackle the grid scheduling problem with the single objective. Besides several heuristics and meta heuristics algorithms have been proposed to address grid scheduling problems based on users specified QoS constraints such as budget and deadline. The population based heuristics approaches are employed in the grid job scheduling problem and have given significant solutions. Ant Colony Optimization (ACO), Particles Swarm Optimization (PSO) and Genetic Algorithm (GA) are examples of population based heuristics and have been applied in job scheduling with different parameters and given feasible optimization solutions (Mathiyalagan, Suriya & Sivanandam, 2010).

The main weakness of these techniques is the optimization of one objective for the grid scheduling heedless the multi objective grid scheduling nature. Very few papers are proposed for the level of grid scheduling problem with multi objective optimization. For example, multi objective artificial bee colony (MOABC) is an extension of artificial bee colony (ABC) presented as the multi objectives workflow scheduling. The time and the cost are the objectives of MOABC. This approach is compared with schedulers like Dead Line Budget Constraint algorithm (DBC) from Nimrod-G) and (Workload Management System (WMS) offered by middleware gLite) (Arsuaga-Ros, Vega-Rodrguez & Prieto-Castrillo, 2012). The goal of the ABC algorithm is to optimize makespan and the execution cost of the application within a constraint budget in the scheduling process (Udomkasemsub, Xiaorong & Achalakul, 2012).

Another multi objective algorithm based on GA is proposed by Prabhu & Kumar (2011) to solve resource scheduling problem in grid. The objectives of the proposed algorithm are optimization of makespan and flowtime functions. The results are compared with Min-min algorithm. The MOGSA and MOSWO algorithms enhanced gravitational search algorithm and small world algorithm respectively to solve the job scheduling problem in grid environment (Arsuaga-Ros, Prieto-Castrillo & Vega-Rodrguez, 2012).

BEE COLONY OPTIMIZATION

The BCO includes a colony of artificial bees searching collaboratively for the optimal solution of a given problem. Each artificial bee generates one solution to the problem. Forward pass and backward pass are two alternating phases constituting a single step in the BCO algorithm. In the forward pass the artificial bee flies to create one partial solution. The artificial bees have to explore the search space and choose new partial solutions by the roulette wheel (Teodorović, Davidović & Šelmić, 2011).

When the backward pass starts, the bee returns to the hive to dance and this is used as a communication technique to exchange information about the quality of the obtained bee solutions. Depending on the information obtained from the dancing activities, some bees become recruits and the others are uncommitted bees. The recruits fly to the next stage as the

problem already has been divided into stages to extend their partial solutions. The uncommitted bees decline their partial solutions and fly to the same partial path of the recruits and expand this path by selecting new partial solutions in the next stage (Teodorovic et al., 2011). All bees will come back to the hive (dancing area) and start exchanging their information to other bees after the forward pass is completed. In the dancing area, the probability of each bee's loyalty (*P*) (loyalty decision to be recruiting) could be calculated by using equation (1):

$$P_b^{u+1} = e^{-\frac{O_{max}-O_b}{u}} \qquad (1)$$

where $b = 1, 2\ldots\ldots B$, B is the number of bees, u is the forward pass counter, $u = 1, 2\ldots\ldots NC$, NC is the number of forward (backward) in each iteration. $O_b \in [0, 1]$ is calculated by equation (2):

$$O_b = \frac{Y_{max}-Y_b}{Y_{max}-Y_{min}} \qquad (2)$$

where Y_b is the partial solution result of *b*, Y_{max} and Y_{min} are the largest and smallest partial solution results. The probability recruitment for partial solution by any uncommitted bee is calculated using equation (3).

$$P_b = \frac{O_b}{\sum_{k=1}^{R} O_k} \qquad (3)$$

where $b = 1, 2\ldots\ldots R$, and R is number of uncommitted bees. The main steps of BCO algorithm are given in Figure 1.

```
B: the number of bees involved in the search.
NC: the number of forward (backward) passes in a single iteration.
Do
1- Initializing: every bee is set to empty solution
2- Repeat
     //forward pass
       Set b=1;
   a)   Repeat
            (1) Evaluate all possible moves;
            (2) Choose one move using the roulette wheel.
             (3) b=b+1;
          Until b >B
        //backward pass start, all bees are back to the hive.
     b) For each bee
          Evaluate (partial/complete) solution for bee b;
     c) Sort the bees according their objective function value
     d) For each bee
          The Loyalty decision is taking by using the roulette wheel;
     e) For each bee
        If (bee is follower), choose a recruiter by the roulette wheel.
3. Evaluate all solutions and find the best one.
While stopping criteria is not satisfied
```

Figure 1. Steps of Bee Colony Optimization (Teodorvić, 2009)

JOB SCHEDULING PROBLEM

Grid job scheduling includes efficient assignment of jobs to the resources in global heterogeneous and dynamic environment. The efficiency can be measured in terms of makespan and execution cost. In this problem the jobs have the following characters. First, they are created from different users/applications. Next, each job has to be completed by a

unique resource. Lastly, they are independent. There are also a number of heterogeneous available nodes that process one job at a time and can dynamically be added or dropped from the grid.

The following has to be considered in formulating the job scheduling problem:

    i.   Number of independent jobs to be scheduled in batch mode
    ii.   Number of resources to participate in the planning
    iii.   Workload of each job (millions of instructions)
    iv.   Computing capacity of each resource (millions of instructions per second)
    v.   Ready machine
    vi.   The expected time to complete any job for each resource. A matrix (MT) will be used to represent this.
    vii. The execution cost of any job by each resource. A matrix (EC) will be used for this purpose.
    viii. A matrix *X* which is represented in equation (4).

$$X_{i,j} = \begin{cases} 1 \; if \; the \; job \; i \; schedule \; to \; resource \; j \\ 0 \qquad\qquad\qquad\qquad\qquad otherwise \end{cases} \qquad (4)$$

The objectives of this study are to find the global optimization of the makespan that satisfies the cost constraint which means developing a schedule that minimizes the makespan value as well as minimizes the execution cost. One of the objectives is minimization of the total execution cost (*Cost*) of the schedule schema. The total cost (*Cost*) can be calculated by combining the total execution cost (*TEC*) and the total communication cost (*TCC*) as in equation (5). The total execution cost is the combination of the execution cost for all the jobs which are scheduled to be run on the resources as in equation (6). The execution cost for any job on any resource can calculated by multiply the cost unit of using a resource (*UR*$_j$) by the execution time of the job for that resource (*J*$_{ij}$). This is calculated by using equation (7). The calculation of the *TCC* depends on the amount of transferred data during the execution. In equation (9) *D*^{in} and *D*^{out} represent the transferred data (in and out) respectively for the resource (*j*) of job (*i*). *UC*$_j$ represents the communication cost unit.

$$Cost = TEC + TCC \qquad (5)$$

$$TEC = \sum_{i=1}^{n} \sum_{j=1}^{m} EC_{ij} \times X_{ij} \qquad (6)$$

$$EC_{ij} = UR_j \times J_{ij} \qquad (7)$$

$$TCC = \sum_{i=1}^{n} \sum_{j=1}^{m} CC_{ij} \times X_{ij} \qquad (8)$$

$$CC_{ij} = UC_{ij} \times \left(D_{ij}^{in} + D_{ij}^{out} \right) \qquad (9)$$

Another objective of the scheduler is to find a better resource for a particular job which will lead to minimize the makespan. Equations (10) and (11) can be used to calculate the expected time for executing a particular job on a selected resource and makespan respectively.

$$MT_{ij} = \partial_j + J_{ij} \qquad (10)$$

$$M = Max_{i=1}^{n} \sum_{j=1}^{m} MT_{i,j} \times X_{i,j} \qquad (11)$$

where *MT* a matrix of the expected time for each job on each resources, ∂_i represents the ready time of resource (*j*). Comparison between any two solutions quality cannot be made for both objectives at once. The best solution in term of makespan may be of low quality when execution cost is considered. For that reason, in this study, Pareto analysis concept is employed to balance the solution quality. All solutions evaluated to be non-dominated and bring a total advantage relatively close to the optimal one in all dimensions will be stored in a

Pareto-optimal set. The best solution can then be selected from this set. The two objectives can then be combined to be considered as multiple criteria decision making within proposed framework.

Multi objectives problem can be defined as the problem of concurrently maximization of multiple conflicting objectives. In this paper the two objectives are considered as minimization objectives. In the state of multi objective problem many solutions are generated which are available in a set of solutions called Pareto optimal set. All solutions in Pareto set are classified as non-dominate. Non-dominate solution means is better at least in one objective with respect to other solutions in the set. Therefore finding Pareto optimal set of a problem is the main concern of multi objectives optimization. The Pareto dominance concept can be formulated as:

Let function $f_{(X)} = (f_{1(X)}, f_{2(X)}, f_{3(X)}, \ldots\ldots, f_{m(X)})$ contain *m* objectives.

Consider x_1, x_2 decision vector (solutions),

$x_1 \prec x_2$ *if and only if*

i. $f_i(X_1) \leq f_i(X_2) \; ; \; \forall i = 1,2 \ldots\ldots\ldots m$
ii. $f_i(X_1) < f_i(X_2) \; ; \; \exists_j = 1,2 \ldots\ldots\ldots m$

Assume that *P* is a Pareto optimal set containing non-dominate solutions, therefore all solutions in *P* have to meet the following conditions:

i.   Any solution in *P* non-dominate to any other solutions in *P* with respect to all objectives.
ii.   Any solution in *P* dominate at least one solution does not belong to *P*.

PROPOSED MULTI OBJECTIVE BEE COLONY FRAMEWORK

A framework of multi objective bee colony MO-BCO is proposed for grid job scheduling. MO-BCO is an expansion of the original bee colony optimization to find a set of optimal solutions for grid job scheduling. In this research the solution is assigned by using a vector of integers where the index of this vector represents the job and the value represent resource. The modules of the proposed MO-BCO framework for grid job scheduling can be divided into three as shown in Figure 2. The modules are for the initialization, forward movement and waggle dancing.
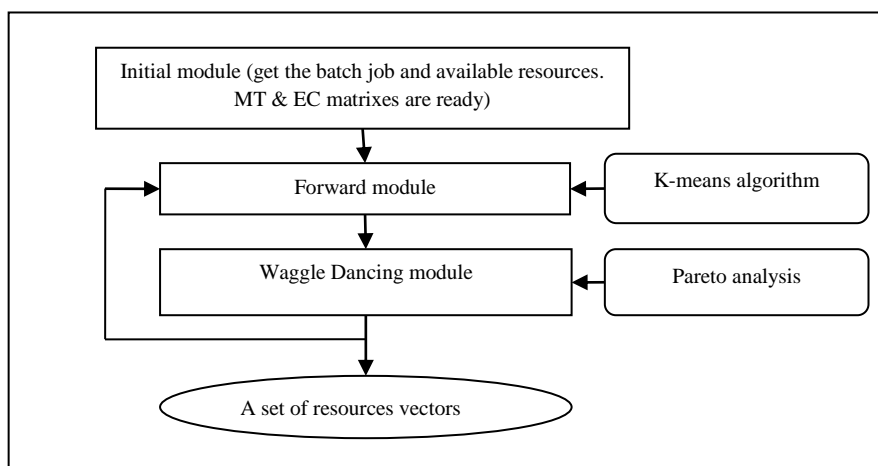
Figure 2. Multi objective bee colony optimization framework for grid job scheduling

In the initialization module, the number of jobs and the available resources will be collected from the available grid. In this work the *n* jobs assigned to *m* resources with an

objective to minimize the makespan and execution cost. If the number of jobs is less than number of resources, the jobs can be assigned to the resources according to the first come first serve rule. Otherwise if the number of jobs is more than number of resources, the allocation of jobs is to ensure an efficient scheduling schema. This scenario will be the focus of this work. For this purpose, an empty integer vector is allocated to each bee. All bees in the hive, initially, will have an empty vector. The matrix (*MT*) which is used to record the expected completion time for any job on any resource is initialized in this model. The matrix (*EC*) is also initialized to include the execution cost for any job on any resource.

Once all the initializations have been completed, the pruning strategy will be added to the forward module by using data clustering to enhance bee colony optimization. K-means is used to find clusters of the resources of the similar execution cost. However, k-means is a method of classifying grouping items where k is the number of pre chosen groups. The bees have to fly looking for available clusters of resources. The number of clusters should be equal to the number of bees. Each bee should select exactly one cluster and no other bee can select that cluster. Later each bee selects one resource from the identified cluster. Choosing resources and clusters are done by using roulette wheel. After selecting the resources, the bees have the partial solutions and return to the hive to evaluate the partial solutions.

In the waggle dancing module, the bees return to the hive with partial solutions. The bees are divided into three sets according to their dancing which will be evaluated by using the Pareto analysis. The division of bees is considered the minimum of makespan and execution time for the selecting resources of each bee. The best solutions (non-dominate) will be with recruiters and worst solutions (dominate) will be with followers. The followers will cancel their solutions and follow the recruiters to extend the partial solutions. The third group of bees having the best solution will continue in their paths. After completion of this module, the bees will again go forward to extend their partial solutions. The second and third modules will be repeated until the feasible solutions are created.

CONCLUSION AND FUTURE WORK

The successful applications of bee colony optimization for solving different job scheduling in different systems has been adopted and adapted in this study to solve the multi objectives nature of grid scheduling. A multi objectives bee colony framework for batch mode scheduling is proposed. The enhancement is made possible with the application of k-means technique and Pareto analysis concept to solve grid job scheduling. This framework allows multiple objectives to be set in order to optimize the performance of job scheduling in grid environments and fulfill the requirements of users. It is expected that the implementation of MO-BCO will produce a high quality solutions. A simulator will be developed to test the proposed framework.

REFERENCES

Abdullah, M., Othman, M., Ibrahim, H. & Subramaniam, S. (2007). Improved genetic algorithm for scheduling divisible data grid application. *Proceedings of The IEEE International Conference on Telecommunications and Malaysia International Conference on Communications*, 461-465.

Arsuaga-Ros, M., Prieto-Castrillo, F. and Vega-Rodrguez, M. (2012). Multi-objective optimization comparison – MOSWO vs. MOGSA - for solving the job scheduling problem in Grid environments. *Proceedings of the 10th IEEE International Symposium on Parallel and Distributed Processing with Applications*, 570-576.

Arsuaga-Ros, M., Vega-Rodrguez, M. and Prieto-Castrillo, F. (2012). Multi-objective artificial bee colony for scheduling in grid environments. *IEEE Symposium on Swarm Intelligence,* 1-7.

Bitam, S., Batouche, M. and Talib, E. (2010). A survey on bee colony algorithms. *Proceedings of The 24th IEEE International Parallel & Distributed Symposium*, 1-8.

Christodoulopoulos, K., Sourlas, V., Mpakolas, I & Varigos, E. (2009). A comparison of centralized and distributed meta-scheduling architectures for computation and communication tasks in grid network. *Computer Communication*, 32, 1172-1184.

Karaboga, D. and Akay, B. (2009). A survey: algorithms simulating bee swarm intelligence. *Artificial Intelligent Review*, 31(1-4), 61-85.

Mathiyalagan, P., Suriya, S., and Sivanandam, S. (2010). Modified ant colony algorithm for grid scheduling. *International Journal of Computer Science and Engineering*, 2(2), 132-139.

Prabhu, S. and Kumar, V.N. (2011). Multi-objective optimization based on genetic algorithm in grid scheduling, *International Journal of Advanced Research in Technology*, 1(1), 54-58.

Pop, F. and Cristea, V. (2009). Optimization of scheduling process in grid environments. UPB Scientific Bulletin Series C, 71(4), 35-48.

Teodorović, D , Davidović, T. and Šelmić, M. (2011). Bee Colony Optimization: the application survey. *ACM transactions on Computational Logic*, 12(4), 1-20.