Journal of Theoretical and Applied Information Technology
31st December 2013. Vol. 58 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

HYBRID OPTIMIZATION FOR GRID SCHEDULING USING GENETIC ALGORITHM WITH LOCAL SEARCH

[1] B.RADHA, [2] Dr. V.SUMATHY
[1] Sri Ramakrishna Engineering College, Department of MCA, Coimbatore, INDIA
[2] Government College of Technology, Department of Electronics and Communication Engineering, Coimbatore, INDIA
E-mail: [1] radhab_216@rediffmail.com

ABSTRACT

With network technology development, Grid Computing satisfies the demand of scientific community's computing power. Grid facilitates global computing infrastructure for user to utilize the services over the network. A robust multi-objective scheduling algorithm is required to optimize workflow grid execution. Scheduling is a research area in Grid computing. Grid scheduling's objective is to deliver Quality of Service (QoS) to grid users and raise resource use. Three conflicting objectives like execution time (makespan), total cost and reliability are considered in this investigation. A Genetic Algorithm (GA) including an Ant Colony Optimization (ACO) to schedule grids is proposed in this investigation. The proposed GA with local search proves its effectiveness in Grid Scheduling.
Keywords: *Grid Computing, Grid scheduling, Genetic Algorithm (GA), Ant Colony Optimization (ACO)*

1. INTRODUCTION

Grid computing is a heterogeneous/dynamic environment. Scheduled jobs rarely coincide with execution times and expected ones in a computing environment. So, job scheduling is a main challenge in grid systems, as no one can control jobs [1]. Another challenge is availability of dynamic resources, and difference between the algorithm's expected execution time and actual time. In a Scheduling Module, all grid system information like CPU speed, its load, CPUs number of resources is collected from Grid Information Service (GIS) to calculate optimal resources to process jobs.

With network technology development, grid computing solves large complex problems. Task scheduling is challenging in grid computing [2]. As many tasks are computed on geographically distributed resources, a scheduling algorithm must be adopted to get minimum completion time. Hence, task scheduling; a NP-complete problem is focused on, by grid computing scholars. Heuristic optimization algorithm solves varied NP-complete problems. GA and Simulated Annealing (SA) are stochastic optimization methods, inspired by nature. GA simulates evolutionary natural selection. A better generation solution is evaluated according to fitness value and candidates with improved fitness values create further solutions through crossover/mutation processes. Simulated annealing is based on annealing about solid matter in physics. Both methods are valid, and are applied to various fields due to their convergence properties.

Evaluating resource sharing effect on application performance is challenging in shared environments task scheduling where a remote task competes with local jobs, or other remote tasks, for resources. A task's completion time is determined by its workload and resource availability. Deploying a distributed task in a Grid computing environment after scheduling decisions still requires integrated system solutions [3]. Deployment involves job submission, data movement/resource monitoring. It is hard in a shared environment, where computing crosses multiple administration domains that use varied resource management middleware/policies. Task scheduling needs integrated solutions of scheduling algorithms, performance prediction and system development. Hence, developing suitable/applicable scheduling system is elusive. This is true in Grid computing, which shares computing resources and where local jobs are autonomic. Grid tasks are distributed/deployed under different middleware/local management systems.

Grid scheduling makes efficient scheduling decisions. Scheduling process involves resources

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

over many administrative domains. Multiple administrative domains can be searched to use a single machine/scheduling a single job to use multiple resources at one/multiple sites. A Grid scheduler called *resource broker*, is an interface between user/distributed resources, hides Grid computing [4] complexities. It performs resource discovery, negotiates access costs using trading services, maps jobs with resources (*scheduling*), stages application/data for processing (*deployment*), starts executing jobs, and gets results. It monitors/tracks application execution progress with changes in Grid run time environment, resource share availability variation and failures.

To compute, a scheduling algorithm requires:

- Resources list with the grid. Let a set R represent m number of resources in grid where each resource $r_j \in R$, $(1 \leq j \leq m)$. Resources have varied processing capabilities delivered at different prices. Thus, a Cost vector requires where $cost_j$ specifies cost of using resource r_j per unit of time.

- Expected time to compute (ETC) matrix, where entry ETC_{ij} gives estimated execution time to complete task t_i on resource r_j. Task execution time information is found from specifications provided by user/literature [5]

- Bandwidth linkage between 2 resources. A m × m Data Transfer Time matrix is taken representing data transfer time (for a data unit) between 2 resources i.e. each entry $B_{s,t}$ stores time to transfer data unit from resource r_s to r_t.

As grid environments are dynamic and computing resources heterogeneous, different criteria are used to evaluate scheduling algorithms efficacy, makespan and flowtime being the most important. Makespan is time when grid finishes latest job and flowtime is sum of finalization time for all jobs [6]. An optimal schedule optimizes flowtime/makespan. This investigation proposes an ACO version for grid job scheduling with the scheduler's goal being to minimize flowtime and makespan. The motivation to use hybrid algorithms is to locate a schedule where completion time of all tasks is minimal. The aim was to improve results by using ACO algorithm. Traditional optimizations methods are deterministic, fast providing exact answers but are often stuck on local optima. So, another approach was needed as conventional methods cannot be applied to modern heuristic general purpose optimization algorithms. Heuristic based algorithms specifically, population based heuristics suit task scheduling in grid environments, but there are complex population based heuristics which take long execution time. The most popular/efficient meta-heuristics in grid scheduling are ad-hoc, local search and population-based methods.

Grid Schedulers design space is usually rich as it is based on objective function users wanting to minimize/maximize - examples being lowering overall job completion time, reducing communication time/volume, and maximizing resource use. It is also based on the proper use of job requirements/job performance models/Grid resource models. The scheduler must choose between varied user authentication, allocation and reservation implementations. Other choices include scheduling application components for single/multiple users and whether rescheduling/ re-planning are required.

The aim in this study is minimizing overall job completion time/application makespan. Makespan represents lapsed time from start of first task to end of last scheduled task. Makespan reduction arranges tasks to level differences between work phase completion time. This investigation proposes a hybrid GA with local search incorporating ACO for grid scheduling to optimize makespan.

2. LITERATURE REVIEW

Resource management/scheduling plays an important role in achieving efficient resources use in grid computing environments. Due to resources heterogeneity, application scheduling is complicated/challenging task in grids. Most researches in this area are focus on improving grid system performance. Some allocation models were proposed based on divisible load theory with varied workloads and a single originating processor. A new resource allocation model having multiple load originating processors as an economic model was introduced by Murugesan and Chellappan (2010). Solutions for optimal allocation of load fraction to nodes were got to reduce grid users cost through linear programming approach. The Resource allocation model efficiently/effectively allocated workloads to proper resources. Experiments revealed that the suggested model obtained better solution regarding cost/time.

Scheduling is an important Grid computing

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

research topic as Grids aim to offer high quality services to Virtual Organizations members based on efficient resource uses. Achievement is through good scheduling strategy applied to local level (clusters)/global level (entire system) of Grids. As scheduling problem is NP complete, sub-optimal solutions alone to this problem can be afforded. Additionally, grid components (users/processes/ resources) are highly dynamic which makes it hard to find a scheduling solution. Pop, et al., (2006) suggested a new approach to solving scheduling issues through simulation. Using simulator shortens distance between real Grid system and its analysis model. To transform a simulator into a tool to cope with Grid dynamicity, solution proposed couples simulator with a Grid monitoring / optimization tool, to ensure that scheduling decisions are taken/used in real time for next short period.

Network technology's rapid development resulted in Grid Computing satisfying demands of computing power of computing community. Grid ensures a global computing infrastructure for users to consume network services. A robust multi-objective scheduling algorithm is needed to optimize workflow grid execution. Navaz and Ansari (2012) considered 3 conflicting objectives like execution time (makespan), reliability and total cost. A multi-objective scheduling algorithm, based on evolutionary computing paradigm and using R-NSGA-II approach was proposed. Simulation showed the proposed algorithm generating multiple scheduling solutions near Pareto optimal front with reduced computation overhead. Use of epsilon dominance based MOEA approach was suggested to solve the Grid's workflow scheduling problems. In a scheduling issue 2 major conflicting objectives, make span/cost were addressed.

A Hybrid algorithm to solve combinatorial optimization problem using Ant Colony and Genetic programming algorithms was proposed by Salami (2009). Evolutionary process of Ant Colony Optimization algorithm adapts genetic operations to enhance ant movement towards solutions. The algorithm converges to optimal final solution, through accumulation of most effective sub-solutions.

An improved ACO cluster algorithm based on a classics algorithm- LF algorithm was proposed by Dai, et al., (2009). By introducing a new formula/probability of similarity metric conversion function and a new formula of distance, this algorithm deals with category data. It suggested a new adjustment process that adjusts clusters generated by carry process iteratively. The algorithm improves cluster efficiency/convergence theoretically. Data experiments showed that improved ACO forms more accurate/stable clusters than K-Modes algorithm, Information Entropy- Based Cluster Algorithm, and LF Algorithm. Scalability experiments revealed that running time had a linear relationship with dataset size. Also, algorithm's process and idea of usage, described by mobile customer classification case, analyse cluster results. This algorithm handles large category dataset rapidly/accurately/effectively ensuring good scalability simultaneously.

Dorigo, et al., introduced ACO. Swarm intelligence is a new approach to problem solving that was inspired by the social behaviours of insects/animals. Specifically, ants inspired many methods/techniques of which the most studied/successful is the general optimization technique called ACO which was inspired by ants foraging behavior. Ants deposit pheromone on ground to mark a favorable path to be followed by other ants. ACO exploits a similar mechanism to solve optimization issues. From the early 90's, when the ACO algorithm was proposed, it attracted researchers attention resulting in the present availability of many successful applications. Also, a substantial corpus of theoretical results now provides guidelines to researchers/practitioners in ACO applications.

Scheduling dependent tasks (DAG) problem is a scheduling version, to efficiently exploit grid systems computational capabilities. Scheduling tasks of a graph onto different machines is an NP Complete problem. Many heuristic/meta-heuristic approaches were used over years due to their providing high quality solutions in a reasonable computation time. Discrete Particle Swarm Optimization (DSPO) is such a meta-heuristic to solve grid scheduling's discrete problem, but it converges to sub optimal solutions because of premature convergence. To deal with this, Garg and Singh (2013) proposed design/implementation of hierarchical discrete particle swarm optimization (H-DPSO) for task scheduling in grid environment. Particles are arranged in a dynamic hierarchy with the good particles lying above exerting a bigger influence on the swarm in H-DPSO. The bi- objective problem version to reduce makespan and total cost simultaneously are optimization criteria. The H-DPSO based

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

scheduler was evaluated under different application task graphs. Simulation proved that H-DPSO based scheduling is a very viable/effective approach for grid computing.

Pop and Cristea (2009) optimized Grid scheduling problems. The Grid scheduling algorithms models were presented. The authors proposed scheduling algorithms taxonomy, based on functional levels, offering a reference to design a complete Grid scheduling solution. The analyses for multi-criteria optimization were presented. A scheduling algorithms critical analysis describes issues in this field/represents support of optimization methods design. The Grid scheduling tools comparison highlights the state of the art for Grids scheduling. Some scheduling mechanism evaluation was presented. The experiments demonstrated good improvement in load-balancing/execution time for optimization scheduling algorithm.

Evolutionary Algorithms (EA) are robust/powerful global optimization techniques to solve large scale problems with local optima. EA needs high CPU times, and are poor regarding convergence. In contrast, local search algorithms converge in a few iterations without a global perspective. Combining global/local search procedures offers advantages of both optimization methods and removes disadvantages. Kelnera, et al., proposed a new hybrid optimization technique merging a GA with local search strategy based on Interior Point method. The efficiency of proposed hybrid approach was proved through solving a constrained multi-objective mathematical test-case.

3. METHODOLOGY

3.1 Genetic Algorithms (GA)

GAs are mechanics of natural selection/natural genetics based search algorithms and are a direct, parallel, stochastic method for global search/optimization imitating evolution of living beings, described by Charles Darwin. GA is part of Evolutionary Algorithms (EA) which uses 3 natural evolution principles [15]: reproduction, natural selection and species diversity maintained by differences of each generation with the earlier one. GA works with an individual set representing possible task solutions. Application of selection principle is through use of a criterion, evaluating the individual regarding a desired solution. Best- suited individuals create next generation.

Chromosomes represent set of genes for GA, which code independent variables. Every chromosome represents a solution for a problem. Individual and vector of variables are used as other chromosomes words. A different chromosomes (individuals) set forms a generation. Through evolutionary operators (like selection, recombination, mutation), an offspring population is created, this being repeated through much iteration until other criteria are met, based on population size, and best solution sought from final individual's population. The suggested solution has advantages like [16],

Chromosome representation

- Number of genes for a chromosome required in hybrid genetic algorithm is less when compared to GA chromosomes.
- The population size (SOP) of GCE is independent from size of environment as SOP is reduced due to local search.

Initial population

In GA, every chromosome represents a potential solution with more than one solution initially. Paths obtained from route discovery phase are considered initial chromosomes.

Fitness function

An obtained solution quality should be accurately evaluated with help of fitness function. The aim of using GA is to find shortest path, lowest throughput between source/destination and larger buffer size than that of the path. It is important to obtain shortest path/lowest delay time as then choosing is based on buffer size. Each chromosome's fitness can be calculated as [17],

$$f\left(Ch_i\right)=\left[\sum_{l\in P(s,r)} c_l + c_d\right]$$

Ch represents chromosome fitness value and Cd delay time of each chromosome where Cl represents path cost. The above fitness function is maximized involving only shortest path/delay constraint; buffer size for each path is checked in evolutionary process.

Crossover operator

Crossover operator is applied to a pair of routes, say X and Y, having a common node, apart

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

from source/destination nodes. As every pair of routes established dynamically cannot be crossed, this operator is not applicable to randomly chosen pairs as in conventional GA's

Mutation Operator

Route mutation is as follows: a route node is chosen randomly and a new route generated

randomly from selected node (mutation node) to destination node, using the scheduler's information.

3.2 Ant Colony Optimization (ACO)

As swarm intelligence and artificial life techniques are more and more being used for explaining optimization problems, they have demonstrated themselves as a good candidate in this area. A computational grid is a hardware/software infrastructure providing pervasive, consistent, dependable and inexpensive access to high-end computational abilities. Simply put, Grid computing connects geographically distributed computers allowing sharing of their computational power/storage capabilities. With the rapid data and computational needs growth, distributed systems/Grids solve issues of large scale computing [18]. There are many options to establish distributed systems and Grid Systems are common for distributed applications. ACO algorithm is a member of swarm intelligence. Here some meta-heuristic optimizations calculate shortest path of each colony. This algorithm is based on behaviour of ants seeking a path between colony and food source.

An efficient algorithm is necessary to solve combinatorial optimization problem. In this section, a hybrid algorithm is proposed for job scheduling combining advantages of ACO and local search. The problem is defined as follows. There are N jobs and M machines. Each job has its own execution order to be performed on M machines. Each job has its own starting time [19]. This algorithm's aim is reducing makespan. It can also be used to schedule jobs in scientific/high power computing.

Ant colony Algorithm

Step 1: Initialization

Initialize pheromone trails, Ant solution construction.

Step 2: Construction

For every ant in each step, choose task number requiring scheduling.

Step 3: Local search

Perform local search using cuckoo search and update path j.

Step 4: Pheromone update

For each path j compute fitness value/update pheromone.

Step 5: Termination condition

If total_ iteration <max_iteration go to step 2 otherwise terminate.

3.3 Local Search

Combining local search techniques to take a solution to its local optimum in search space often helps improve heuristic algorithms. Local search technique defines solution neighbourhood [20]. A solution will have one/more problem resources.
'Problem' resource makespan should be reduced to achieve immediate reduction in the overall solution makespan. The neighbourhood is a solution of single job transfer from problem resource to other resources. Local search technique analysis, neighbourhood and transfer lowering maximum schedule length of 2 resources are involved most. The above is repeated till no other improvement is possible.

4. RESULTS AND DISCUSSION

The proposed scheduling method's effectiveness is assessed/evaluated using makespan - which is time taken by grid system to complete latest task. Experiments were conducted with 20 resource clusters/100 jobs. Experiments were undertaken using proposed hybrid grid scheduling algorithm. Figure 1 shows Makespan time vs. number of iterations.

Journal of Theoretical and Applied Information Technology
31st December 2013. Vol. 58 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

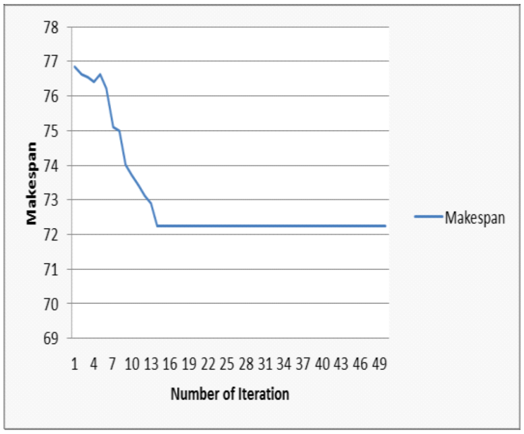

Figure 1: Makespan Vs. Number Of Iterations.

Figure 1 shows that incorporation of elitism using ACO and local search in GA achieves optimal makespan. Also, convergence is achieved in 15th generation.

5. CONCLUSION

With network technology development, grid computing solves larger scale complex problems becoming a focus technology. Task scheduling is challenging in grid computing which is a NP-

Complete problem. Conventional methods for optimizations are deterministic, fast providing exact answers but often being stuck in local optima. So, another approach is required when conventional procedures are not applicable to modern heuristic as they are general purpose optimization algorithms. This investigation proposes a GA incorporating local search and ACO for grid scheduling. Simulation results prove the proposed GA's effectiveness when combined with local search to schedule Grids.

REFRENCES

[1]. Lorpunmanee, S., Sap, M. N., Abdullah, A. H., &Chompoo-inwai, C. (2007). An ant colony optimization for dynamic job scheduling in grid environment. *International Journal of Computer and Information Science and Engineering*,*1*(4), 207-214.

[2]. Zhang, L., Chen, Y., Sun, R., Jing, S., & Yang, B. (2008). A task scheduling algorithm based on pso for grid computing. *International Journal of Computational Intelligence Research*, *4*(1), 37-43.

[3]. Wu, M., & Sun, X. H. (2006, April). The GHS grid scheduling system: implementation and performance comparison. In *Parallel and Distributed Processing Symposium, 2006. IPDPS 2006. 20th International* (pp. 8-pp). IEEE.

[4]. Buyya, R., & Murshed, M. (2002). A deadline and budget constrained cost-time optimisation algorithm for scheduling task farming applications on global grids.*arXiv preprint cs/0203020*.

[5]. Garg, R., & Singh, A. K. (2013). Enhancing the Discrete Particle Swarm Optimization based Workflow Grid Scheduling using Hierarchical Structure. *International Journal of Computer Network and Information Security (IJCNIS)*,*5*(6), 18.

[6]. Branch, S., & Sari, I. Tasks Scheduling in Computational Grid using a Hybrid Discrete Particle Swarm Optimization.

[7]. Murugesan, G., & Chellappan, C. 2010). An economic-based resource management and scheduling for grid computing applications. *arXiv preprint arXiv:1004.3566*.

[8]. Pop, F., Dobre, C., Godza, G., &Cristea, V. (2006, September). A simulation model for grid scheduling analysis and optimization. In *Parallel Computing in Electrical Engineering, 2006. PAR ELEC 2006.International Symposium on*(pp. 133-138). IEEE.

[9]. Navaz, S., & Ansari, U. (2012) An Evolutionary Algorithm in Grid Scheduling by multi- objective Optimization using variants of NSGA. International Journal of Scientific and Research Publications, 2(9).

[10]. Al Salami, N. M. (2009). Ant colony optimization algorithm. *UbiCC Journal*,*4*(3), 823-826.

[11]. Dai, W., Liu, S., & Liang, S. (2009). An improved ant colony optimization cluster algorithm based on swarm intelligence. *Journal of Software*, *4*(4), 299-306.

[12]. M Dorigo, M Birattari, and T Stutzle. Ant Colony Optimization. Artificial Ants as a Computational Intelligence Technique.

[13]. Pop, F., & Cristea, V. (2009). Optimization of Schedulig Process in Grid Environments. *UPB Sci. Bull., Series C*.

[14]. V Kelnera., F Capitanescub., O Leonarda., and L Wehenkel. An Hybrid Optimization Technique Coupling Evolutionary and Local Search Algorithms.

[15]. Popov, A. (2005). Genetic algorithms for optimization. *User Manual, Hamburg*.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[16]. Gurav, U. B., & Netak, L. D. Hybrid Genetic Algorithm Based Task Scheduling in Heterogeneous Grid. In *Proceedings of SPIT- IEEE Colloquium and International Conference, Mumbai, India*.

[17]. D.Sureshkumar., K.Manikandan., & M.A.SaleemDurai. (2011). Secure On-Demand Routing Protocol for MANET using Genetic Algorithm. *International Journal of Computer Applications, 19(8).*

[18]. Kumar, C. K. U. K., & Gogulan, R. (2012). Max-Min Particle Swarm Optimization Algorithm with Load Balancing for Distributed Task Scheduling on the Grid Environment.

[19]. Babukartik, R. G., & Dhavachelvan, P. (2012). Hybrid Algorithm using the advantage of ACO and Cuckoo Search for Job Scheduling. *International Journal of Information Technology Convergence and Services (IJITCS) Vol, 2*.

[20]. Kousalya, K., & Balasubramanie, P. (2009). To improve ant algorithm's Grid scheduling using local search. *International Journal Of Computational Cognition (Http://Www. Ijcc. Us)*, *7*(4), 47.