Available online at www.elixirpublishers.com (Elixir International Journal)

Computer Science and Engineering

Elixir Comp. Sci. & Engg. 38 (2011) 4353-4357

ISSN: 2229-712X

A particle swarm optimization algorithm for job shop scheduling in grid environment

N.Jayapandian[1] and A.M.J.Md. Zubair Rahman[2]

[1]Department of Computer Science & Engineering, Knowledge Institute of Tech., Salem

[2]Department of Computer Science & Engineering, Al-Ameen Engineering College, Erode.

ARTICLE INFO

Article history:

Received: 22 August 2011;

Received in revised form:

26 August 2011;

Accepted: 31 August 2011;

Keywords

Grid Scheduling,
PSO,
Tabu Search,
Local Search,
Meta Heuristic.

ABSTRACT

Grid computing is a high performance computing environment to solve larger scale computational demands. Grid computing contains resource management, task scheduling, security problems, information management and so on. Task scheduling is a fundamental issue in achieving high performance in grid computing systems. A computational GRID is typically heterogeneous in the sense that it combines clusters of varying sizes, and different clusters typically contains processing elements with different level of performance. The scheduling problem is computationally hard even when there are no dependencies among jobs. Thus, the new local search (LS) and particle Swarm Optimization (PSO) algorithm seems to be efficient for the problem of batch job scheduling on computational grids. We consider the grid scheduling as a bi-objective optimization problem consisting of the minimization of the makespan and flowtime. The bi-objectivity is tackled through a hierarchic approach in which makespan is considered a primary objective and flowtime a secondary one. In this, a heuristic approach based on particle swarm optimization algorithm is adopted for solving task scheduling problem in grid environment. Particle Swarm Optimization (PSO) is one of the latest evolutionary optimization techniques by nature. It has the better ability of global searching and has been successfully applied to many areas such as, neural network training etc. Due to the linear decreasing of inertia weight in PSO the convergence rate becomes faster, which leads to the minimal makespan time when used for scheduling. To make the convergence rate faster, the PSO algorithm is improved by modifying the inertia parameter, such that it produces better performance and gives an optimized result. The computational results show that our PSO & TS implementation clearly outperforms the compared algorithms. This work proposes optimization technique called Tabu search that is combined with the ant colony optimization and PSO technique to solve the grid scheduling problems.

Introduction

Grid computing is a form of distributed computing that involves coordinating and sharing computing, application, data and storage or network resources across dynamic and geographically dispersed organization [1].Grid technologies promise to change the way organizations tackle complex computational problems. Grid computing is an evolving area of computing where standards and technology are still being developed to enable this new paradigm

Heuristic optimization algorithm is widely used to solve a variety of NP-complete problems. Abraham et al and Braun et al [2] presented three basic heuristics implied by Nature for Grid scheduling, namely Genetic Algorithm (GA) [4], Simulated Annealing (SA) [5] and Tabu Search (TS) [10], and heuristics derived by a combination of there three algorithms. GA and SA are powerful stochastic optimization methods, which are inspired form the nature.

Tabu Search

Tabu search is a mathematical optimization method, belonging to the class of local search techniques. Tabu search enhances the performance of a local search method by using memory structures: once a potential solution has been determined, it is marked as "taboo" so that the algorithm does not visit that possibility repeatedly. Tabu [1] search is a metaheuristic algorithm that can be used for solving combinatorial optimization problems. Grid is a type of parallel and distributed system that enables the sharing, selection and aggregation of geographically distributed autonomous and heterogeneous resources dynamically at runtime depending on their availability, capability, and performance, cost and users quality of service requirements

The demand for scheduling is to achieve high performance computing. It is very difficult to find an optimal resource allocation for specific job that minimize the schedule length of jobs. The scheduling problem is a NP-hard problem [5] and it is not trivial The main goal is to schedule all the incoming applications to the available computational power. Meta heuristic approaches have shown their effectiveness for a wide variety of hard problems. These approaches produce best results in practice.

Tabu Search provides solutions very close to optimality and is used to tackle difficult problems. These successes have made TS extremely popular among those interested in finding viable solutions to the large combinatorial problems encountered in

Tele:
E-mail addresses: njayapandian@gmail.com

many practical settings. TS is based on local search (LS) improvement techniques.

PSO Algorithm

Particle Swarm Optimization (PSO) [6] is one of the latest evolutionary optimization techniques inspired by nature. It is simulated the process of a swarm of birds preying. It has the better ability of global searching and has been successfully applied to many areas such as neural network training, control system analysis and design, structural optimization and so on. It also has fewer algorithm parameters than both genetic algorithm and simulated algorithm. Furthermore, PSO algorithm works well on most global optimal problems. In this paper, Tabu & PSO algorithm is employed to solve the scheduling problem in a grid environment.

Literature Review

A computational grid is “a hardware and software infrastructure that provides dependable, consistent, pervasive and inexpensive access to high-end computational capability. These algorithms assign tasks to the best machines which produced better quality of service. Scheduling on a grid has three main phases. Phase one is resource discovery, which generates a list of potential resources. Phase two involves gathering information about those resources and choosing the best set to match the application requirements. In phase three the job is executed.

This section reviews a set of heuristic algorithms which has been designed to schedule the meta-tasks in the computational grids. The collection of independent tasks with no data dependencies is called as meta-task. Metatasks are mapped on to the available machines statically; each machine in the computational grid executes a single task at a time. For this mapping, it is assumed that the number of machines, ‘m’ and the number of tasks, ‘t’, are known a priori. A large number of heuristic algorithms have been designed to schedule tasks to machines on grid computing systems

One of the best and simple heuristics methods is called Min-min. In this method, compute the minimum completion time of each task with respect to all machines. The task with the overall minimum completion time is selected and assigned to the corresponding node. The currently assigned job is removed from the unscheduled task list and the above process is repeated until all the tasks are scheduled. Here, all jobs have a good chance to select a suitable resource. So this method automatically minimizes the makespan and balances the load to an extent.

Particle swarm optimization (PSO) technique is employed in many optimization and search problems due to its simplicity and ability to tackle these problems successfully. The PSO optimizes an objective function by iteratively improving a swarm of solution vectors, called particles, based on special management of memory. Each particle is modified by referring to the memory of individual swarm’s best information. Due to the collective intelligence of these particles, the swarm is able to repeatedly improve its best observed solution and converges to an optimum. A heuristic approach proposed by Lei Zhang, Yuehui Chen, Bo Yang [1] based on particle swarm optimization is adapted to solving scheduling problem in the grid environment. Each particle is represented a possible solution. The approach aims to generate an optimal schedule so as to get the minimum makespan and maximum resource utilization while completing the tasks.

The study of Brian Ivers ,Gary G.Yen [4] examines the optimization of the job shop scheduling (JSP) by a search space division scheme and use of the meta-heuristic method of particle swarm optimization (PSO) to solve it. The job shop scheduling problem (JSP) is a well known huge combinatorial problem from the field of deterministic scheduling. It is considered the one of the hardest in the class of NP hard problems. Particles are initialized in the search space of a particular problem by assigning them a position, which represents a solution to the objective function, and a velocity. The PSO algorithm is considered a very fast algorithm and is emerging as a widely studied used algorithm for optimization problems. Due to the memory characteristics of the PSO algorithm, it works better and there is no need to have the knowledge of other particles as, PSO takes care of it.

Improvement of Particle Swarm Optimization Based on Neighborhood Cognizance and Swarm Decision was proposed by ZHU Meijie, LIU Hanxing, SUN Weiwei, ZHU TongLin [5]. The original PSO usually converges prematurely, and falls into the local optimal solution. Aimed at the shortcoming of PSO, here they have put forward an Improved PSO based on Neighborhood Cognizance (NCPSO) and Improved NCPSO based on Swarm Decision (SDNCPSO). These two improved PSO can reduce the possibility of converging prematurely. The results of experiment prove that these two improved PSO can improve the performance of global convergence in PSO and make PSO converge to global optimal solution faster. The parameters used here is not optimized, the effect of these parameters and optimized parameter setting should be analyzed and taken care.

This investigation [Shih-Tang Lo, Ruey- Maw Chen, Der-Fang Shiau and Chung-Lun Wu] introduced a particle swarm optimization (PSO) approach to solve the multi-processor resource constrained scheduling problems [6]. There are two new rules are proposed and evaluated, named antiinertia solution generation rule and bidirectional searching rule of PSO. The anti-inertia solution generation rule enables some jobs with anti-inertia velocity used to decide the start processing time, and escaping from local minimum. The bidirectional searching rule combines forward and backward scheduling to extend the search solution space. These two suggested rules applied in PSO scheme are capable of finding global minimum. The simulation results reveal that the proposed approach gives us the better and optimizes results. In a dynamic situation, there may be some emergency jobs arriving at a certain time or changing the resources available and requirement which also have to be considered. An Improved PSO Algorithm was proposed (BU Yan-ping, ZHOU Wei, YU Jin-shou) against the optimal objective of to minimize the total completing time [7]. This presents an improved particle swarm optimization (PSO) algorithm with discrete coding rule for grid scheduling problem. The improved PSO algorithm can keep all the advantages of the standard PSO, such as implementation simplicity, low computational burden, and few control parameters, etc. A set of experiments show that the algorithm is stable and presents low variability. It has been tested the improved PSO algorithm against the MaxMin heuristic and found that improved PSO outperforms MaxMin by the total makespan and other performance. Fixed values for parameters are used and the algorithms are scripted in the Mat lab. But these optimization algorithms should be implemented for simulation part in the grid environment in java which can yield a better result and performance.

Another heuristics approach is Max-min. Like Min-min, Max-min also calculates the minimum completion time of each job. It selects a job with the overall maximum of minimum completion time. On comparison with MCT, Max-Min considers all unmapped tasks during their mapping decision. The Max-Min may produce a balanced load across the machine. When compare to Max-Min Min-Min proves to be the best one.

The ant colony optimization (ACO) algorithm is a distributed algorithm used to solve NP-hard combinatorial optimization problems.[1] NP stands for non-deterministic polynomial. NP-hard problems are hard in the sense that the answer cannot be found in polynomial time and the answer can be guessed and verified in polynomial time. The ACO uses a population of co-operating ants also known as agents. The cooperation phenomenon among the ants is called foraging and recruiting behavior.

This describes how the ants explore the world in search of food sources, then find their way back to the colony and indicate the food source to the other ants of the colony. To do so, the ants use an indirect way of communicating through tracks of pheromone, a chemical substance that ants deposit on their paths. Each ant deposits a fraction of pheromone on the way back to the colony so as to indicate the source to the others. Hence the pheromone information plays a major role in finding the shortest path from the food to the colony.

Existing System

Local search

The local search is meta heuristic for solving computationally hard optimization problem. Local search can be used on problem that can be formulated as finding a solution maximizing a criterion among a number of candidate solutions.[1] Local search algorithm move form solution to solution in the space of candidate (the search space) until a solution deemed optimal is found or a time bound is elapsed. In general a solution will have one or more ‘problem’ resources (those with schedule lengths equal to the makespan of the whole solution). Try to reduce the ‘problem’ resource makespan as this will immediately reduce the overall makespan of the solution. The neighborhood is a solution of single transfer of a job from the problem resource to any other resources. The local search technique analysis, the neighborhood and the transfer which reduces the maximum schedule length of the two resources is involved the most. The above process is repeated until no further improvement is possible. In local search there are three parts namely swap, move and Tabu search.

Algorithmic frame for a local search algorithm

```
 S = current solution
s = NULL
Repeat until s <> S
Find out the problem resource’s and problem
resource’s problem job
Create neighbor of S ( s ) to transfer the problem
job to some other resource
If s is better quality than S then
S = s
End repeat
The final output is in s
```

Tabu Search

Tabu search (TS) is a meta heuristic approach and global iterative optimization method designed to find the optimal solution for combinatorial problems. The basic principle of TS is to pursue local search whenever it encounters a local optimum by allowing non-improving moves of the neighbors and neighborhood structure. The search space is simply the space of all possible solutions that can be considered during the search. Let k be the total number of neighborhood solution that can be produced from the current solution. In the search space, a neighbor s_i ($0 \leq i \leq k$) is defined by a pair (x, y) provided x and y are successive operations on a machine and on some critical path in the graph G. The critical path is the longest path from the node 0 to the node ñ+1 corresponding to the makespan of the constructed solution. Let N(S) be the set of neighbors that can be applied to the current solution. The neighborhood is defined by the processing orders of the operations at the time of application of a neighbor si to the current solution. In each iteration, the application of each neighbor si∈N(S) defines a set of neighboring solutions in the search space.

$Atransf\ (s) = \{s\_ \mid s\_ = s \oplus mtransf\ (ti,mj\ ), TL[i][j] + asp\ value) \leq k\}$

$Aswap(s) = \{s\_ \mid s\_ = s \oplus mswap(ti,\ tj\ ), \max(TL[i][s(j)], TL[j][s(i)]) + asp\ value) \leq k\},$

Tabu is one of the important elements of TS and is used to prevent cycling. One can avoid cycling by declaring the tabu moves that prevent the application of recent neighbors, which are called forbidden neighbors. Tabus are stored in a short-term memory of the search space (the tabu list) and usually a fixed and fairly limited quantity of information is recorded. The most commonly used tabus involve recording the last few neighbors that are not applied for transformations performed on the current solution and prohibiting the reverse transformations. Variable tabu list size The basic role of the tabu list is to prevent cycling.[1] The fixed length tabus cannot prevent cycling . We can observe that if the length of the list is too short, cycling cannot be prevented, and long-size tabu creates many restrictions so as to increase the mean value of the visited solutions. An effective way of removing this difficulty is to use a tabu list with variable size according to the current iteration number. The length of the tabu list is initially assigned according to the size of the problem and it will be decreased and increased during the construction of the solution so as to achieve better exploration of the search space.

PSO Algorithm

Particle swarm optimization (PSO) is an algorithm modeled on swarm intelligence, that finds a solution to an optimization problem in a search space, or model and predict social behavior in the presence of objectives. The particle swarm simulates this kind of social optimization. A problem is given, and some way to evaluate a proposed solution to it exists in the form of a fitness function. A communication structure or social network is also defined, assigning neighbors for each individual to interact with. Then a population of individuals defined as random guesses at the problem solutions is initialized. These individuals are candidate solutions. They are also known as the particles, hence the name particle swarm. An iterative process to improve these candidate solutions is set in motion. The particles iteratively evaluate the fitness of the candidate solutions and remember the location where they had their best success. The individual's best solution is called the particle best or the local best. Each particle makes this information available to their neighbors.

Each particle represents a candidate solution to the optimization problem. The position of a particle is influenced by the best position visited by itself i.e. its own experience and the position of the best particle in its neighborhood i.e. the experience of neighboring particles. When the neighborhood of a particle is the entire swarm, the best position in the neighborhood is referred to as the global best particle, and the resulting algorithm is referred to as the gbest PSO. When smaller neighborhoods are used, the algorithm is generally referred to as the ibest PSO. The performance of each particle is measured using a fitness function that varies depending on the optimization problem. Each Particle in the swarm is represented by the following characteristics:

- ➢ The current position of the particle
- ➢ The current velocity of the particle

The particle swarm optimization which is one of the latest evolutionary optimization techniques conducts searches uses a population of particles. Each particle corresponds to individual in evolutionary algorithm. Each particle has an updating position vector and updating velocity vector by moving through the problem space.

$V_i^{k+1} = wV_i^k + c_1 \text{ rand}_1(\ ) \text{ x } (\text{pbest}_i - s_i^k) + c_2\text{rand}_2(\ ) \text{ x } (\text{gbest} - s_i^k)$ -------- Eq (1)

$S_i^{k+1} = S_i^k + V_i^{k+1}$ -------- Eq (2)

Where, v_i^k is the velocity of i at iteration k, s_i^k is the current position of i at iteration k. c1 and c2 are positive constants and rand1 and rand2 are uniformly distributed random number in [0,1].The velocity vector is range of [-Vmax,Vmax]. In Velocity updating eq (1), eq (3) terms that creates new velocity are,

- ➢ Inertia term, forces the particle to move in the same direction as before by adjusting the old velocity.
- ➢ Cognitive term (Personal best), forces the particle to go back to the previous best position.
- ➢ Social Learning term, forces the particle to move to the best previous position of its neighbors.

Proposed Work

The inertia weight *w* weighting function in Eq (1), controls the momentum of the particle. The inertia weight can be dynamically varied by applying a scheme for the setting of the PSO, where *w* decreases over the whole run. The decrease depends on the start and end value of the weight given. A significant performance improvement is seen by varying the inertia.

Meta- Heuristic methods have turned out to be a standard approach in combinatorial optimization. Dealing in practice with real-size problems makes the use of such methods the de facto choice. One such method is the Tabu Search (TS), which has shown its effectiveness in a broad range of combinatorial optimization problems. Another one is PSO algorithm, In this algorithm is an adaptive method that can be used to solve optimization problem. Conducting search uses a population of particles. The ant algorithm uses a colony of artificial ants that behave as cooperative agents in a mathematical space where they are allowed to search and reinforce pathways (solutions) in order to find the optimal ones. This approach which is population based has been successfully applied to many NP-hard optimization problems

PSO and Tabu search Method with ant algorithm

The combination of this PSO and Tabu search we propose a new PSO with TS algorithm for the grid scheduling combined with the ant algorithm on computational grids for the bi-objective case, namely, the minimization of makespan and flowtime.

In General grid environment, one of the most important issues is how to represent a solution for task scheduling. The solution representation ties up with the PSO algorithm performance. It's defined one particle as a possible solution in the population. The dimension n corresponding to n tasks and each dimension represents a task. The position vector of each particle makes transformation about the continuous position. The smallest position value, namely, (SPV) rule is used first to find a permutation corresponding to the continuous position x_i^k. For the n tasks and m resource problem, each particle represents a reasonable scheduling scheme. The position vector x_i^k has a continuous set of values. Where x_i^k is the position value of i particle with respect to the n dimension and s_i^k is the sequence of task of *i* particle in the processing order with respect to the n dimension. Then the operation vector r_i^k is defined by following Equation, R= s_i^k mod m.

This Proposed algorithm distinguishes for its flexibility in exploiting domain/problem knowledge in the selection of parameters and other inner ingredients. This proposed approach is able to outperform other known heuristic approaches. It reduces the 'problem' resource makespan as this will immediately reduce the overall makespan of the solution.

The Tabu search technique analysis, the neighborhood and the transfer which reduces the maximum schedule length of the two resources is involved the most. The combination of ant algorithm and Tabu search with PSO algorithm will reduce the makespan and there by increases the performance. This makespan is defined as the finishing time of latest job. This algorithm can get better effect for a large scale optimization problem. Task scheduling algorithm based on PSO algorithm can be applied in the computational grid environment.

Conclusion

Here we Combine PSO with Tabu Search algorithm. This technique will produce better result compared with previous aspects. The scheduling algorithm based on PSO is proposed for task scheduling problem on computational grids. Task scheduling algorithms based on PSO algorithm can be applied in computational grid environment. Each particle represents a feasible solution. This project aims at generating an optimal schedule so as to complete the tasks in a minimum time as well as utilizing the resources in an efficient way. The main objective of the proposed approach is to reduce the makespan. The future work may include other hybridization techniques to further minimize the execution time.

References

[1] Lei Zhang, Yuehui Chen, Bo Yang "Task Scheduling Based on PSO Algorithm inComputational Grid", 2006 Proceedings of the Sixth International Conference on Intelligent Systems Design and Applications.

[2] M. Fikret Ercan "A hybrid particle swarm optimization approach for scheduling flow-shops with multiprocessor tasks", 2008 International Conference on Information Science and Security.

[3] Xiaohong Kong, Jun Sun and Wenbo Xu " Particle Swarm Algorithm for Tasks Scheduling in Distributed Heterogeneous System", 2006 Proceedings of the Sixth International Conference on Intelligent Systems Design and Applications.

[4] Foster and C. Kesselman (editors), The Grid: Blueprint for a Future Computing Infrastructure, Morgan Kaufman Publishers, USA, 1999.

[5] Abraham, R. Buyya and B. Nath, Nature's Heuristics for Scheduling Jobs on Computational Grids, The 8th IEEE

International Conference on Advanced Computing and Communications (ADCOM 2000), pp. 45-52,Cochin, India, December 2000.

[6] Brian Ivers ,Gary G.Yen "Job Shop Optimization Through Multiple Independent Particle Swarms" 2007 IEEE Congress on Evolutionary Computation.

[7] Fang Dong, Junzhou Luo, Lisha Gao and Liang Ge(2006)' A Grid Task Scheduling Algorithm Based on QoS Priority Grouping',IEEE

[8] Fatos Xhafa and Javier Carretero(2008)' Design and Evaluation of Tabu Search Method for Job Schedulingin Distributed Environments',IEEE

[9] Ho.S.L,Shiyou Yang,Guangzheng Ni, and Jose Marcio Machado(2006),' A Modified Ant Colony Optimization AlgorithmModeled on Tabu-Search Methods ' , IEEE,Vol 42,No.

[10] Kobra Etminani(2007)' A Min-Min Max-Min Selective Algorihtm for Grid Task Scheduling',IEEE

[11] ZHU Meijie, LIU Hanxing, SUN Weiwei, ZHU TongLin "Improvement of Particle Swarm Optimization Based on Neighborhood Cognizance and Swarm Decision" 2007 IEEE

Prof N.Jayapandian has completed his B.Tech (Information Technology) in IRTT, Erode, India in 2006. Then he completed M.E., (Computer Science Engineering) in Kongu Engineering College, India in 2009. Now he is doing research in the field of A Meta-Heuristics Task Scheduling in Grid Computing. Currently, he is working as Assistant professor in the Department of Computer Science & Engineering at Knowledge Institute of Technology, Salem, India. Previously he has worked as a Software Engineer at Web Designing Field. He has completed Two years of Teaching Service. He has published papers in 6 National Conferences and 1 international Conference.

Dr.A.M.J.Md.Zubair Rahman has completed his Ph.D., in Association Rule Mining algorithms in 2010 in Anna university and has completed his M.S.,(Software Systems) in BITS Pilani , India in 1995. Then he completed M.E., (Computer Science & Engineering) in Bharathiar Univeristy , Tamilnadu , India in 2002 . Currently, he is working as Principal at Al-Ameen Engineering College, Erode. Previously he has worked as Assistant Professor in the Department of Computer Science & Engineering , Kongu Engineering College , Tamil Nadu, India. He has completed 20 years of teaching service. He has published 15 articles in International / National journals.