A Hybrid GABFO Scheduling for Optimal Makespan in Computational Grid

Shiv Prakash, School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi, India
Deo Prakash Vidyarthi, School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi, India

ABSTRACT

Scheduling in Computational Grid (CG) is an important but complex task. It is done to schedule the submitted jobs onto the nodes of the grid so that some characteristic parameter is optimized. Makespan of the job is an important parameter and most often scheduling is done to optimize makespan. Genetic Algorithm (GA) is a search procedure based on the evolutionary technique that is able to solve a class of complex optimization problem. However, GA takes longer to converge towards its near optimal solution. Bacteria Foraging Optimization (BFO), also derived from nature, is a technique to optimize a given function in a distributed manner. Due to limited availability of bacteria, BFO is not suitable to optimize the solution for the problem involving a large search space. Characteristics of both GA and BFO are combined so that their benefits can be reaped. The hybrid approach is referred to as Genetic Algorithms Bacteria Foraging Optimization (GABFO) algorithm. The proposed GABFO has been applied to optimize makespan of a given schedule in a computational grid. Results of the simulation, conducted to evaluate the performance of the proposed model, reveal the effectiveness of the proposed model.

Keywords: Bacteria Foraging Optimization (BFO), Computational Grid (CG), Genetic Algorithm (GA), Load, Makespan, Scheduling

1. INTRODUCTION

Computational Grid (CG) primarily processes the compute intensive jobs (Prakash & Vidyarthi, 2012). Next generation scientific researches are being carried out by the large collaboration of researchers across the globe. CG is defined as a global computing infrastructure that provides dependable, consistent, pervasive and inexpensive access to high end computational capabilities despite the grid resources and the grid users being geographical distributed. Grid facilitates resource coordination and sharing towards problem solving in dynamic, multi-institutional virtual organizations (Foster & Kesselman, 2004). In the grid, resource coordination are not subject to centralized control and using standard, open, general-purpose protocols and interfaces nontrivial qualities of service is delivered (Foster & Kesselman, 2004). CG facilitates the sharing of computational power and storage capacity over the Internet (Foster &
Kesselman, 2004). CG allows the execution of a complex job through an intelligent interface. A grid middleware searches the appropriate resources, from the pool of grid resources, to execute the job depending upon the execution policies and the job requirements (Prakash & Vidyarthi, 2012).

GA is often used to solve the complex combinatorial optimization problem whose solution involves exploring and identifying the solution from a big search space. GA works on the principle of natural selection and evolution. GA operators, such as crossover, mutation etc. (Goldberg, 2005; Mitchell 2006), are applied on a randomly generated population comprised of a set of chromosomes (solutions). These chromosomes are evaluated against a fitness function, the optimization objective, and evolve over the generations.

Bacterial Foraging Optimization (BFO), a concept from biology, has drawn the attention of researchers in recent because of its efficiency towards solving real-world optimization problems (Biswas, et al. 2007; Kim, et al. 2007; Niu et al. 2011). BFO algorithm is an optimal foraging of bacteria that evolves using behavior of E.coli bacteria (Passino, 2002). Searching strategy, in BFO, is random as searching food in E.coli bacteria. BFO algorithm has three types of operators; chemotaxis, reproduction and elimination-dispersal. In BFO, searching is performed using bacterium tumble and run. Mapping BFO to grid scheduling involves locating the heavily and lightly loaded node using the bacterium behavior (Niu et al. 2011). In tumbling (Biswas, et al. 2007), a random vector R between -0 to 1 is generated. Shift direction of bacteria with random vector R is multiplied by a constant C and is divided by the square root of vector R and its transpose. Constant C differs for each bacterium. In the proposed problem, tumbling (Biswas, et al. 2007) is mapped as the random load transfer from one node to another in the grid. A task is randomly selected, from a heavily loaded node, and transferred to a lightly loaded node. Chemotaxis step of the bacterium is performed by tumble and run.

The search in BFO can be favorable or unfavorable, swimming up a nutrient gradient or swimming down a nutrient gradient (Kim, et al. 2007). Therefore climbing up in the nutrient hills (Biswas, et al. 2007) depend on the bacterium and nutrient by avoiding noxious substances. The sensors, it needs for optimal resolution (Kim, et al. 2007), are receptor proteins which are very sensitive and posses high gain. A small change in the concentration of nutrients can cause a significant change in the behavior. In chemotaxis step (Biswas, et al. 2007; Kim, et al. 2007; Niu et al. 2011), bacteria try to maintain current position and direction by adding multiplication of step length and direction. In duplicate step, bacteria try to maintain constant population. Also, in duplicate step, bacteria are ranked according to their position and half of the bad ranked bacteria are replaced by the good ranked bacteria. Elimination-dispersal occurs with a given probability. In elimination dispersal step, certain bacteria are eliminated due to unexpected change in temperature. A simple BFO algorithm (Chitra, et al. 2011; Nayak, et al. 2012; Vivekanandan & Ramyachitra 2012) is given below in Box 1.

The condition for loop termination may be set depending on the problem. Reproductive efficiency of bacteria is affected by temperature variation. Bacteria (Kim, et al. 2007) secret from cell to cell under necessary conditions attracting singles, so that they make group to protect each other.

GA has the problem of the premature convergence whereas BFO is not suited for the problem with a large search space. This work proposes a hybrid GABFO technique for CG scheduling that aims to improve the job makespan. A comparative study has also been performed with a GA based CG scheduling model.

The outline of the paper is as follows. After the introduction in section 1, some related work has been briefed in section 2. Hybrid GABFO technique has been elaborated in section 3. The problem is stated in section 4 and the proposed model is discussed in section 5. Performance evaluation of the model is done through its simulation and the observations of the experiments conducted are depicted in section 6. Concluding remarks of the work appears in section 7.
25 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the product's webpage: www.igi-global.com/article/a-hybrid-gabfo-scheduling-for-optimal-makespan-in-computational-grid/120691?camid=4v1

This title is available in InfoSci-Journals, InfoSci-Journal Disciplines Computer Science, Security, and Information Technology. Recommend this product to your librarian: www.igi-global.com/e-resources/library-recommendation/?id=2

Related Content

Reserve Constrained Multi-Area Economic Dispatch Employing Evolutionary Approach

A Remote and Sensorless Stator Winding Temperature Estimation Method for Thermal Protection for Induction Motor

In Search of an Integrating Construct
Dynamic Genetic Algorithms with Hyper-Mutation Schemes for Dynamic Shortest Path Routing Problem in Mobile Ad Hoc Networks
www.igi-global.com/article/dynamic-genetic-algorithms-hyper-mutation/62836?camid=4v1a