International Journal of Computer Applications (0975 – 8887)
Volume 63– No.11, February 2013

Hybridized TABU-BFO Algorithm in Grid Scheduling

Devi Priya
Post Graduvate Student
CSE department
Karunya university, india

Joshua Samuel Raj
Assistant Professor
CSE department
Karunya university, india

V. Vasudeven
Registrar,
KalasalingamUniveristy,
Srivilliputhur,India

ABSTRACT
Task Scheduling is an important issue in Grid Environment of multiprocessors system. The problem of scheduling a set of dependent and independent Task in a distributed system is considered. In this paper we are going to study a detailed study on tabu search, BFO and its hybridization. Tabu search is a heuristic procedure which uses its adaptive memory structures in order to find optimal solution in grid scheduling. Bacterial Foraging Optimization Algorithm is a well-known optimization algorithm for Task scheduling. The Hybridization of tabu-BFO approach is considered for reliability factor which increase the performance by means of efficient Task Scheduling.

KEYWORDS
chemotaxis, swim, tumble, neighborhood set, tabu set, adaptive memory.

1. INTRODUCTION
In a non-centralized system, set of dependent and Independent task has been scheduled using Task scheduling algorithms. Tabu search is a “higher level” heuristic procedure for solving optimization problems, designed to guide other methods to escape the trap of local optimality [7]. By using local search procedure iteratively tabu search obtains the potential solution by moving one place to another. The search crieteria will be stopped until the optimal solution obtained. One of the main issue in task Scheduling is the NP-Hard problem. Thus to avoid such a problem in the grid environment Tabu search concept is introduced.

Tabu search is a Meta heuristic local search algorithm that can be used for solving optimization problems in the Grid environment. By using local search procedures, problem may occur, because of a stuck in poor scoring area. Using memory structures in Tabu search finds the solution by iteratively moving from the current solution x to an another solution X1 in the memory space. Thus by the use of adaptive memory in tabu search, the flexible search behavior will be created.

Bacterial foraging optimization algorithm (BFOA) has been widely accepted as a global optimization algorithm of current interest for distributed optimization and control [4]. In General the social bacteria tends to eliminate animals with poor foraging strategies (methods for locating, handling, and ingesting food) and favor the propagation of genes of those animals that have successful foraging strategies, since they are more likely to enjoy reproductive success . Since many generations are passed by, poor foraging strategies are shaped into good ones (redesigned) or eliminated. In accordance with, such evolutionary principles have led scientists into the field of foraging theory to hypothesize that it is appropriate to model the activity of foraging as an optimization process [3].

To solve the optimization problem in the grid computing environment of multi processors, the tabu search method is introduced along with the combination of Bacterial Foraging Optimization Algorithms (BFOA). In the Real world scheduling approach the application can be used to the Heterogeneous processors Scheduling, Machine scheduling applications, job scheduling, sequence and batching.

1.1 DEPENDENT TASK.
The task which works on a specific work entity, may relates the entity in some assured interconnections with a common goal to be achieved is called Dependent Task. Tasks are said to be dependent on each other. A single task will be splited into many smaller sub tasks, where subtask are dependent to each other (the output of one task will be the input of other Task). Each and every sub task will be maintained with a well planned progress.

1.2 INDEPENDENT TASK.
Each task are independent to each other. Every task will have its own goal and progress towards it. The failure of one task will not affect other task.

1.3 FUNCTIONS OF RELIABLITY IN GRID

• Reliability in computational (h/w, s/w),resources that contain relate data which comprise of user application unit to be executed.

• Applicable capabilities initiated by users till they they submit to the grid for execution;

• Essential functions in grid systems which are necessary for grid systems to operate(resources).

2. TABU SEARCH
Pure and hybrid Tabu Search approaches have set new records in finding better solutions to problems in scheduling [1]. The memory structures are a set of rules and banned solution used to filter in which solutions will beadmitted to the neighborhood $N^{*}(x)$ to be explored by the search called Tabu list. A tabu list is a set of the solutions that have been visited in the recent past (with some condition list). Mainly the memory structure can be classified into three classes.

2.1 Short-term:
Recently considered solution.(specific changes of recent moves within the search space and preventing future moves from undoing those changes)

2.2 Intermediate-term:
A list of Tabu rules constructed to bias the search towards promising areas of the search space.(to focus the search on promising areas of the search space (intensification), called aspiration criteria.[8])

2.3 Long-term:

set of Rules that help to improve general diversity in the search process across the search space. (to encourage useful exploration of the broader search space, called diversification.[8])

Short-term memory alone may be enough to achieve solution superior to those found by conventional local search methods, but intermediate and long-term structures are oftennecessary for solving harder problems[10]. Intermediate and long-term structures primarily serve to intensify and diversify the search (the short-term memory also intensifies the search by temporarily locking in certain locally attractive attributes, i.e., those belonging to moves recently evaluated to be"good"[12]).One major issue with Tabu Search is that it is only effective in discrete spaces[11].

Tabu search (TS) is based on the premise that problem solving, in order to qualify as intelligent, must incorporate adaptive memory and responsive exploration [1]. The adaptive memory contain the implementation of procedures that are capable of searching the solution space effectively. Most important key factor of tabu search is the adaptive memory and the responsive exploration. Here the adaptive memory feature allows the implementation of set of procedures that are capable of searching the solution search space efficiently. Responsive exploration integrates the basic principle of intelligent search (explores the promising region).

Tabu search points to finding new and more effective ways of taking advantage of mechanisms associated with both the adaptive memory and responsive exploration. It uses flexible structures memory (to permit search information to be exploited more thoroughly than by rigid memory systems or memory less systems), conditions for strategically constraining and freeing the search process (embodied in tabu restrictions and aspiration criteria), and memory functions of varying time spans for intensifying and diversifying the search (reinforcing attributes historically found good and driving the search into new regions)[7].

The memory structure allotted in Tabu search has mainly 4 basic principal dimension namely, recency, frequency, quality, and influence. In recency and frequency based memory structures are complement to each other. The way/ability to differciate the possible solution visited during the search is called the quality. The increase in flexibility of memory structure will allow the used to find more optimal solution towards the given problem, in a multi objective environment.

The metrics used to select the Impact of choice made during the search both quality and structure is called influence. The memory structure in Tabu has both explicit and attributive. The complete solution visited during the search will be contained in Explicit in record format, in which they used to expand the local search.

3. BFO

Bacterial foraging optimization is an algorithm that has its thrust on the group foraging behavior of Escherichia coli (E-Coli) present in the human intestine and has been used to solve optimization problems and is applied to task scheduling[13]. BFO has been hybridized with different algorithms in order.Bacterial Foraging Optimization (BFO)is a novel optimization algorithm based on the social foraging behavior of E. coli bacteria [5]. Researchers have illustrated how groups and individual of bacteria which forage indeed of nutrients, to model into a distributed optimization process in a heterogeneous environment, which is called the Bacterial Foraging Optimization. The Bacterial Foraging Optimization system consists of four steps mainly chemotaxis (direction of organisms' movement), swarming, reproduction, and elimination-dispersal [3].

The locomotion movement of the bacterium is attained by flagella. It helps the bacterium to swim or tumble. For a bacterium (E. coli), have several flagella per cell. The rotation can be done in two ways, namely counter-clockwise this breaks the flagella bundle apart, which points to the new direction such a process is called "tumble".

In other words tumble refers to unit walk of bacteria toward a different direction. Swim refers to unit walk of bacteria in the same direction. A motile *E. coli* propels itself from place to place by rotating its flagella. To move forward direction, the flagella takes counter clockwise direction by rotating itself, then organism obtain "swims"[9]. But when flagella rotation abruptly changes to clockwise direction, then the bacterium "tumbles" in same place and it incapable of going around anywhere. Then again bacterium starts swimming in some new, random direction.

3.1 Chemotaxis

The simulation movement of an *E.coli* cell by continuous swimming and tumbling via flagella. As we discussed an *E.coli* bacterium can move in two different ways. It can swim for a period of time in the same direction or it may tumble, and alternate between these two modes of operation for the entire lifetime. Consider for a bacterium θ^i ,j be the chemotactic loop , k be the reproduction loop, and l be the elimination dispersal loop. Where $C(i)$ represents the size of the step taken in the direction obtained from the unit walk. This movement of bacterium occurs by directing their cell movement for searching their food in the respective environment.

$$\theta^i(j_1, k, l)) = \ (j_0, k, l) + c(i).\frac{\sqrt{\Delta(i)}}{\Delta(i)} \qquad (1)$$

3.2 Reproduction

Correspondingly through the chemotactic movement the health of the bacterial will be calculated. Thus bacteria which have the least health eventually die. In other hand every healthier bacteria (lower value of yielding the objective function) asexually split into two bacteria. Then they placed in the same location. This keeps the swarm size constant.

3.3 Elimination and Dispersal

The dispersion event happens after a certain number of reproduction processes. A lowest healthier bacterium is chosen, according to a preset probability, to be dispersed and moved to another position within the environment. These events may prevent the local optima trapping effectively [6]. According to a preset probability, some bacteria are chosen to be killed and moved to another position within the environment [2]. By continuously alternating swim and tumble the overall total movement will be calculated. Repellents. The flowchart of BFO Algorithm is shown in Fig 1.1.

Step1: Initialization.

Step2: Evaluation. (Evaluate using fitness function, Save the obtained valued in order to avoid best value not to be missed)

Step3: Movement.

-Tumble(moving in different direction)

-Move(movement of bacterium)

- Swimming (moving in same direction)

Step4: Reproduction,(if the evaluation loop is incomplete move to Step2 until reaches the reproduction time).

-the healthier bacterium on first half of saved index will divide into two child by obtaining copy of parent,as this makes population constant.

Step5: Elimination,(executed only when the set of evaluvation and reproduction step are met, else go back to Step 2 unless reaches the elimination time).

-the bacterium having least health will die

Fig 1.1 Pseudo code of Bacterial Foraging Optimization Algorithm

4. COMPARITIVE STUDY

In 2010, SuphaphornPanikhom, NuapettSarasiri and SarawutSujitjorn proposed a paper on Hybrid Bacterial Foraging and Tabu Search Optimization (BTSO) Algorithms for Lyapunov's Stability Analysis of Nonlinear Systems.Where BTS Algorithm focus its point on rapidly increasing searches rapidly which tends to high quality solution on the basis of operation performed by adaptive tabu search (ATS).The BTSO algorithm is applied to stability analysis of linear and nonlinear systems based on the Lyapunov's methods[14].

In 2009 NuapettSarasiri, and SarawutSujitjorn introduced the idea of Bacterial Foraging Optimization and Tabu Search: Performance Issues and Cooperative Algorithms. It presents comparison studies on various algorithm .the performances between the adaptive bacterial foraging optimization (ABFO), the adaptive tabu search (ATS) and the cooperative algorithms is considered. Assessment scenario adopts some 3D surface optimization problems, one of which contains many global Solutions [15]. The computing result indicates the superiority of the cooperative BF-TS algorithms over other algorithm. Application of the proposed algorithms also includes the control of a hard disk head-stack.

In 2011 NuapettSarasiri and SarawutSujitjorn come up with a concept on Control Design Optimization of Truck Braking System using Bacterial-Foraging Tabu-Search Metaheuristics.the presentation of the paper includes controlling the design of truck braking system.It describes design comparisons among the simple internal model control (SIMC), Ziegler-Nichol (ZN), Cohen-Coon (CC) methods, and the proposed bacterial-foraging-tabu-search (BFTS) metaheuristics [16].

5. Task Scheduling Using Tabu-BFO

Count k: =1
The step by step process in a TABU-BFO algorithm is us follows
Step 1. Initialize parameters n, *S, Nc, Ns, Nre, Ned, Ped, C(i) (i = 1, 2, . . . , S),* θ^i, where:
n: dimension of the search space,
S: the number of bacteria in the colony,
Nc: chemotactic steps,
Ns: swim steps,
Nre: reproductive steps,
Ned: elimination and dispersal steps,
Ped: probability of elimination,
C(i): unit walk (i.e., the size of the step taken in each run or tumble).
*Step 2.*Elimination-dispersal loop: $l = l + 1$.
*Step 3.*Reproduction loop: $k = k + 1$.
While (stopping Criteria is not met)
Identify Neighbourhood set IN_s
*Step 4.*Chemotaxis loop: $j = j + 1$.
a) For i = 1, 2, . . .,*S*, take a chemotactic step for each bacterium.
b) Compute fitness function, $F(i, j, k, l)$.
c) Save the obtained value,Fsave= F(i, j, k, l) d) in order to avoid missing best value(Identify aspiration set)
d) Tumble: Generate a random vector $\Delta(i)$where $\Delta m(i), m = 1,2, \dots \dots . . n$, with a random walk ranges [−1, 1].
e) Move. Let $\theta^i(j_1, k, l) = \theta^i(j_0, k, l)$ + C(i)$\sqrt{\Delta(i)}/\Delta^T(i)$
f) Compute $F(i, j_1, k, l)$ *with* $\theta^i(j_1, k, l)$
g) Swim
(i) Let $m = 0$ (initialize counter)
(ii) While $m < Ns$

- Let m =m + 1.
- If $F(i, j_1, k, l) < Fsave$ *(add to tabu list)*
- Else let m =Ns.

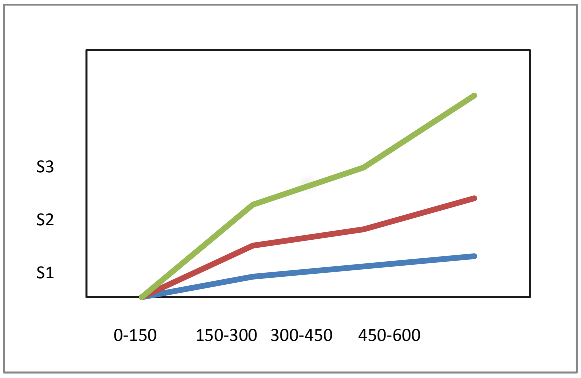
*Step 5.*If $j < Nc$, go to Step 4. (till the criteria is met)
Step6. Reproduction.
a). For the given *k* and *l*, and for each *i = 1, 2, . . . , S,*
let$R_{Health} = \sum_{j=1}^{N_c} F(i, j, k, l)$
b) R_{Health} will be sorted in the acsending order as the bacterium having least health will die,and the bacterium with highest value will be split into two,in such case the child will have the exact copy of the parent.
c) choose the best solution using B_{sol}.where
B_{sol}=$Neighbourhood_{set}$ -$Tabu_{set}$
$(solution\ in\ neighbourhood\ set\ not\ in\ tabu\ set\ will\ be\ taken)$

Memorises if it improves previous best known solution
S: $=s^1$(moves from s to s^1)
K: =k+1;
*Step 7.*If *k <Nre,* go to Step 3 loop.
*Step 8.*Elimination-dispersal: for *i = 1, 2, . . . S,* with probability *Ped,* eliminates and disperses each bacterium. To do this, if a bacterium is eliminated, simply disperse one to a random location on the optimization domain. If *l <Ned,* then go to Step 2; otherwise end.

For a number of Tabu iteration K, the Basic BFO parameter such as dimension of the search space, number of bacteria in the colony, chemotactic steps, swim steps, reproductive steps, elimination and dispersal steps, probability of elimination are initialized. The neighborhood set IN_s is initially identified. From the identified neighborhood set the chemotactic movement is observed. Meanwhile the unit walk of bacteria is executed using tumble and swim from which fitness function is calculated. Optimal solution will be added to the tabu list. In order to avoid the local optima, aspiration criteria is tagged. Thus when next iteration occurs the new optimal solution is find out, same solution will not be repeatable. To find the recently used optimal solution adaptive memory techniques is used.

The neighborhood set which contain a set of solution. Tabu set has the recently used solution. When difference operation is done in neighborhood set from tabu set. (i. e) solution available in neighborhood set which is not in tabu set(newly obtained optimal solution). It is considered a best solution.B_{sol}

6. EXPERIMENTATIONAL RESULT



In reproduction step the available optimal solution's fitness value will be assigned in ascending order, the saved array will be divided into two parts. The bacterium which has highest health(i. e) lowest fitness will be split into two, which has the exact copy of parent. Thus the population of the bacterium size is constant in tabu-bfo. The bacterium which has the lowest health will die in the elimination step. The loop will be executed until the optimal solution is obtained.

Y Axis represents 3 set namely s1,s2,s3. Set s1 is the tabu execution set. S2 is the BFO execution set. S3 is the hybridization of tabu and Bfo namelyTBFO algorithm. Where in the X Axis the no of Task execution is mentioned. The performance metric is measured.in Tabu algorithm id good in convergence where increase in no of task will decrease the performance metric, due to the lack of adaptive memory structures. In other hand BFO is a good optimization algorithm but failed to provide a good convergence due to struck in the local optima. Thirdly we have the Tabu-BFO algorithm, by combining both tabu's adaptive memory structure concept and the optimization algorithm in BFO has been provided good performance when compared to tabu and BFO separately.

In the tabu-BFO the chemotactic movement will be observed in the selective neighborhood set. Thus tabu set already has the list of available solution, where through tumble and swim concept the best movement obtained will be calculated using fitness function. From the obtained optimal solution will be saved according to the health of the bacteria from which reproduction and elimination step is done.

7. CONCLUSION

Task scheduling is an important issue in grid computing. Tabu search is a meta heuristic algorithm used to solve optimization problem using adaptive memory structure concepts.Task Assignment using Bacteria Foraging Optimization (BFO) Natural selection tends to eliminate animals with poor foraging strategies and favor the propagation of genes of those animals that have successful foraging strategies, since they are more likely to enjoy reproductive success. After many generations, poor foraging strategies are either eliminated or shaped into good ones. which has the ability to find the solution for the host of problems such as applied science, engineering and computing.

Tabu-BFO has the ability to resolve the problem in local optima using adaptive memory concepts. The algorithm can be extended to global optimization where BFO is well known optimization algorithm. From the analysis made and the Result obtained, it is identified that tabu-BFO has obtained with the promising result in measuring performance in grid scheduling.

8. REFERENCES

[1] Hernández-Ocaña, E.Mezura-Montes and E.A.Portilla "*Mechatronic Multi-Objective Optimization using a Modified Bacterial Foraging Optimizer*"CongresoInternational de Electronics, Instrumentacion Computation 22 vol 5 3-17

[2] SamanehZareh*"Grid Scheduling using Cooperative BFO Algorithm*" American Journal of Scientific Research.ISSN 1450-223 Issue 62(2012) pg 78-87

[3] Y. Liu,K. M. Passino, M. A. Simaan"*Biomimicry of Social Foraging Bacteria for Distributed Optimization" Models, Principles, and Emergent Behaviors*" JOURNAL OF OPTIMIZATION THEORY AND APPLICATIONS: Vol. 115, No. 3, pp. 603–628.

[4] Swagatam Das, ArijitBiswas, SambartaDasgupta, and Ajith Abraham "*Bacterial Foraging Optimization Algorithm: Theoretical Foundations, Analysis, and Applications*" IEEE Transaction on Evolutionary

[5] Jason Brownlee “*Clever Algorithms: Nature-Inspired Programming Recipes,*” Jason Brownlee pp.257-264, 2011.Soft Computing 10(2010)

[6] Sambarta Dasgupta, ArijitBiswas, Ajith Abraham “*Adaptive Computational Chemotaxis in Bacterial Foraging Algorithm”*IEEE Transaction on Evolutionary VOL 13, NO 4 Aug 2001

[7] webhttp://interfaces.journal.informs.org/content/20/4/74. abstract

[8] webhttp://www.cleveralgorithms.com/natureinspired/stochastic/tabu_search.htm

[9] Fred Glover, Manuel Laguna *“Principles of Tabu Search”* Faculty de Matemáticas, University Valencia, Dr. Moliner 50, 46100 Burjassot (Valencia) Spain. Rafael.Marti@uv.es

[10] M. Malek, M. Huruswamy, H. Owens, M. Pandya (1989). *Serial and parallel search techniques for the traveling salesman problem*.

[11] Sean Luke (2009). *Essentials of Metaheuristics*.

[12] Fred Glover (1990). "Tabu Search: A Tutorial". *Interfaces*.

[13] Joshua Samuel Raj (2012) “Contribution of BFO in Grid Scheduling”

[14] Suphaphorn Panikhom, Nuapett Sarasiri and Sarawut Sujitjorn, “Hybrid Bacterial Foraging and Tabu Search Optimization (BTSO) Algorithms for Lyapunov’s Stability Analysis of Nonlinear Systems ” International Journal Of Mathematics and Computer In Simulation,2010.

[15] Nuapett Sarasiri, and Sarawut Sujitjorn” Bacterial Foraging Optimization and Tabu Search: Performance Issues and Cooperative Algorithms” International JournalOf Mathematics and Computer In Simulation, 2009.

[16] Nuapett Sarasiri and Sarawut Sujitjorn ” Control Design Optimization of Truck Braking System using Bacterial-Foraging-Tabu-Search Metaheuristics ” World Academy of Science, Engineering and Technology 56 2011

AUTHOR’S PROFILE

Devipriya.S. pursuing her M.Tech in Software Engineering from the Department of Computer Science in Karunya University received her B.Tech in 2011from Karunya University.

Joshua Samuel Raj, working towards the Ph.D degree in grid scheduling received his B.E degree from PETEC under Anna University and M.E degree from Jeya College of Engineering under Anna University. His areas of interest include Grid computing, Mobile Adhoc Networking, Multicasting and so forth

Dr. V. Vasudevan, Registrar of Kalasalingam univeristy received his Ph.D degree from Madurai Kamaraj University in the year 1992. He is the Project Director for the Software Technologies Group of TIFAC Core in Network Engineering. His areas of interest include Grid computing, Mobile Adhoc Networking, Multicasting and so forth