Scheduling Homology Modeling in Grid Environment

D Ramyachitra
Department of Computer Science,
Bharathiar University,
Coimbatore - 641046
Tamil Nadu, India.

P Pradeep Kumar
Department of Computer Science,
Nehru College of Arts and Science,
Coimbatore
Tamil Nadu, India

***Abstract-* Grid computing discipline involves the actual networking services and connections of a potentially unlimited number of ubiquitous computing devices within a grid. This research shows how homology modeling works for given protein sequences in grid environment. The quality of homology modeling is dependent on the quality of the sequence alignment and template structure. In First Come First Served (FCFS) strategy, the protein sequence is scheduled to the resource on first come first serve order and processed until the particular process comes to a completion. On average it takes more time to search for number of sequences and users have to wait for a long time to submit their queries and get the results. To overcome this time delay, FCFS is implemented in Grid. Here the time taken to process the protein sequences gets minimized. The scheduling is done based on the size of the protein sequence such that the system that takes the minimum time to process the particular protein sequence is found out initially in the Grid environment and it is allotted for further processing.**

Keywords- Grid, Scheduling, Homology Modeling, FCFS.

I. INTRODUCTION

A grid is a type of parallel and distributed system that enables the sharing, selection and aggregation of geographically distributed autonomous resources dynamically at runtime depending on their availability, capability performance, cost, and user’s quality of service requirements [1]. Grid is a form of distributed system with non-interactive workloads that involves large number of files [2]. Grid computing enables the sharing of hardware and data resources to create a cohesive resource environment for executing distributed application [3]. Although it has been used within the academic and scientific community for some time, standards, enabling technologies, toolkits, and products are becoming available that allow businesses to use and reap the advantages of grid computing.

Grid computing provides high performance mechanism for discovering access to remote computing resources in a seamless manner [4]. Grid computing is a high performance computing environment to solve large scale computational demands.

Grid scheduling is a process of mapping grid tasks to grid resources over multiple administrative domains. The grid scheduler has four phases, which consists of resource discovery, resource selection, job selection and job execution [5]. The goal of scheduling is to achieve highest possible system throughput and to match the application need with the available computing resources.

Grid Scheduling is a sophisticated decision making that operates at different levels of grids. Local scheduling is used at the level of clusters, usually to balance load. Global schedulers (grid schedulers) are used to map user jobs to resources according to their requirements and properties. Higher level schedulers can be used to select brokers or grids to a specific job [6].

Grid applications often involve large amounts of data and/or computing resources that require secure resource sharing across organizational boundaries [7]. Grid computing offers a way to solve Grand Challenge problems such as protein folding, financial modeling, earth quake simulation and climate/weather modeling. Bioinformatics is essential for achieving so many complex tasks such as use of genomic information in understanding human diseases, identification of new molecular targets for drug discovery and in unraveling human evolution mysteries [8]. Larger bodies of scientific and engineering applications stands to benefit from grid computing, including molecular biology, weather forecasting, aircraft design, fluid mechanics, biophysics, biochemistry, biology, drug design, data mining, neuroscience/brain activity analysis, and astrophysics [8].

Grid environment is designed to facilitate certain bioinformatics research problems, such as sequence alignment, alternative splicing, protein function/structure prediction, gene identify and bio-chip data analysis [9].

Homology modeling is based on the reasonable assumption that two homologous proteins will share very similar structures. Because a protein's fold is more evolutionarily conserved than its amino acid sequence, a target sequence can be modeled with reasonable accuracy on a very distantly related template, provided that the relationship between target and template can be discerned through sequence alignment [10].

II. Related Work

Abdulal, W proposed an algorithm, which minimizes Make span, Flow time, and Time to release as well as it maximizes Reliability of Grid Resources. It takes Transmission time and waiting time in Resource Queue into account. It uses Stochastic Universal Sampling or Rank Roulette Wheel Selection and single Change Mutation to outperform other Genetic Algorithms, speeds up convergence, and provides better solutions than other Genetic Algorithm solutions. Moreover Genetic Algorithm based on Stochastic Universal Sampling has superior solutions over all remaining Genetic Algorithms. The simulation results demonstrates that proposed algorithm reduces total execution time of tasks, increases the Reliability of whole Grid System, and boosts user satisfaction [11].

Punhani, A presents an approach for CPU scheduling when considering the multiple criteria with the help on Multi objective optimization. Multi objective genetic algorithm is implemented to provide better solution and have considered two factors as over objectives first is optimal waiting time and second is the execution of jobs based on their priorities and evaluates the performance and efficiency of the proposed algorithm using simulation results [12]. Reddy, K.H.K proposed a dynamic load balancing technique over a tree based grid model and demonstrate the efficacy of Hierarchical Job Scheduling (HJS) approach over Flat Structure Job Scheduling (FJS). Experiments have been carried out using a grid test bed with gridgain 2.0 as middleware and results show that HJS performs better than FJS [13].

Amudha T et al stated that a grid scheduler first allocates the important(high priority) jobs to the resources and then it allocates the low prioritize job so as to achieve the maximum resource utilization rate, minimize the makespan and avoid the load balancing level problem. In this paper, the author propose a new framework and QoS(Quality of Service) Priority Based Scheduling Algorithm for effective task scheduling to the resources in the grid environment. The algorithm is simulated using Java. The results show that our proposed QoS priority based scheduling algorithm gives better results in makespan and resource utilization rate than other algorithms such as QoS guided weighted mean time min (QWMTM), Min-Min and Max-Min heuristic algorithms [14].

Ravi, V.T proposed four novel scheduling schemes that can automatically and dynamically map jobs onto heterogeneous resources. Additionally, to improve the utilization of massively parallel resources, the author also proposed heuristics to automatically decide when and which jobs can share a single resource [15].

K. Vivekanandan et. al., have proposed the use of Bacteria Foraging Optimization (BFO) for finding similar protein sequences in the existing databases. The authors state that the proposed BFO performs well compared to the existing algorithms in terms of makespan, resource utilization and minimization in the case of non-execution of client requests [16].

Guoshi Xu et. al., have proposed certain bioinformatics research problems, such as sequence alignment, alternative splicing, protein function/structure prediction, gene identify, bio-chip data analysis, and so on, that requires massive computing power, which is hardly available in a single computing node. In order to facilitate bioinformatics research, it is designed and implemented a distributed and parallel computing environment with grid technology, in which, biologists can solve bioinformatics problems using distributed computing resources in parallel and reduce execution time [9].

Homologous sequence is viewed as one evolutionary instance of the target sequence and all the homologous sequences constitute one homology bag. The similarity definition between two homology bags, called Homology-based Multi-instance Kernel (HoMIKernel). The top-level kernel called HoMIKernel+ achieves better predictive performance than the baseline models and the incorporation of homologous sequences does increase the predictive performance [17].

III. Homology Modeling

Homology modeling can produce high-quality structural models when the target and template are closely related, which has inspired the formation of a structural genomics consortium dedicated to the production of representative experimental structures for all classes of protein folds [18]. Homology is a computational approach for three-dimensional protein structure modeling and prediction [19].

When two proteins share similar sequence, they will have similar three-dimensional structures. If one of the protein sequences has a known structure, then that structure can be superimposed onto the unknown protein with a high degree of confidence. The method of homology modeling is based on the observation that protein tertiary structure is better conserved than amino acid sequence [20]. The quality of the homology model is dependent on the quality of the sequence alignment and template structure.

In homology modeling, once the sequence is input, the input sequence is analyzed and then the alignment is done for the sequence. The input sequence is compared with the sequence available in the Protein Data Bank and the comparative result is obtained.

IV. HOMOLOGY MODELING IN GRID

Before The main criteria in homology modeling are template selection and sequence alignment between the target and the template. Homology models are unreliable in predicting the conformations of insertions or deletions. The homology modeling routine will proceed to arrange the backbone of the target sequence according to that of the template, using the sequence alignment to decide where to position each residue. Therefore, the quality of the sequence alignment is of crucial importance.

Among all current computational approaches, homology modeling is the only method that can reliably generate a three-dimensional model for a protein [22]. If a target protein shares significant amino acid sequence similarity to at least one experimentally solved three dimensional structures (template), homology or comparative modeling can be applied to construct a three-dimensional model for the new protein.

The grid environment was set up with ten systems. The protein database was stored in all the systems and the scheduler runs in one machine which schedules the query given by the user to the appropriate machines and the machines were utilized to the maximum extent when the proposed scheduling algorithm was used.

The processing time for the protein sequences through homology modeling in single system through FCFS is too long and the sequences are queued for an extended time. To overcome this, the research work is proposed to minimize the processing time for the protein sequences effectively through homology modeling FCFS in the Grid environment. When the protein sequence is input to the system in the Grid environment, the search of the input in PDB (Protein Data Bank) is done and the search result is produced as the percentage of sequence matches. When more than one sequence arrives at a time, the sequences will be moved to the scheduling queue and from there the server retrieves sequences and assigns it to the grid systems.

If a single sequence is input to all the systems in the grid and the execution time will be noted, the system which produces minimum execution time that is the relevant system to execute the particular protein sequence. Protein sequences are allocated to the grid system, based on their size and the minimum execution time is noted based on the percentage match found from the PDB in the Grid and the timing is tabulated. The tabulated time is cross checked with FCFS in single and in Grid, compared to those two, FCFS in Grid gives the minimum execution time.

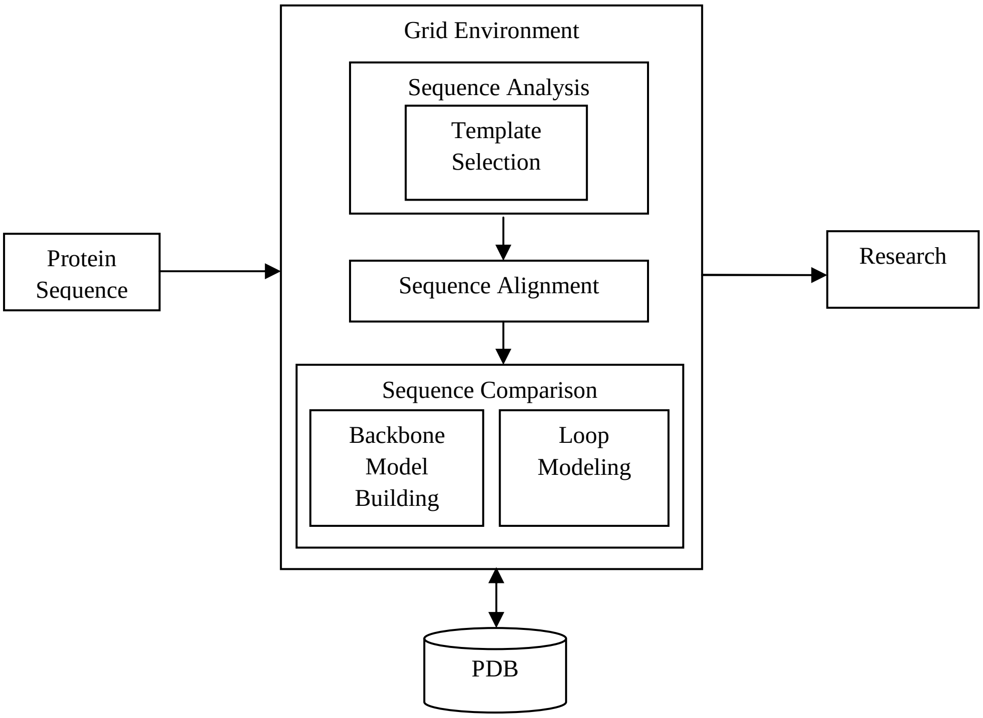

Figure 1. Architectural Framework for Homology Modeling in Grid [21]

V. RESULTS AND DISCUSSION

The FCFS algorithm using single system and Grid environment are evaluated for homology modeling. The time taken for performing homology modeling using FCFS in Grid is minimum when compared to FCFS in single system.

The execution time of 1 to 20 protein sequences are taken in two different groups, one is from 1 to 10 and other is from 11 to 20. The sequence size is considered as minimum of 150 and a maximum of 1000.

Figures 1-5 and the tables 1-5 shows the execution time of 1 to 10 protein sequences for the size 150 to 1000. Figures 6-10 and the tables 6-10 shows the execution time of 11 to 20 protein sequences for the size 150 to 1000.

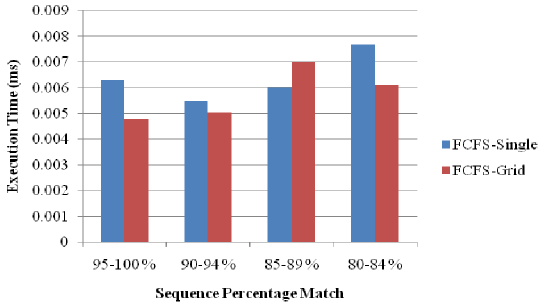

Figure 2. Execution Time of the Protein Sequences for the range 150 to 200

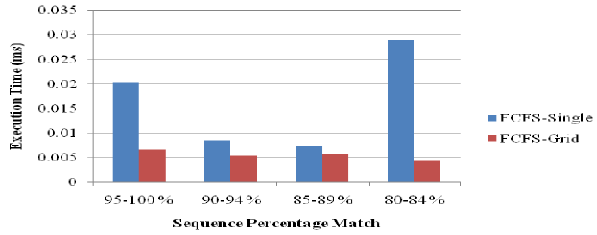

Figure 3. Execution Time of the Protein Sequences for the range 201 to 300

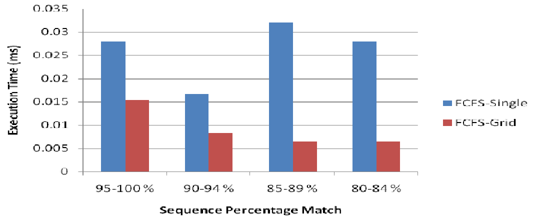

Figure 4. Execution Time of the Protein Sequences for the range 301 to 450

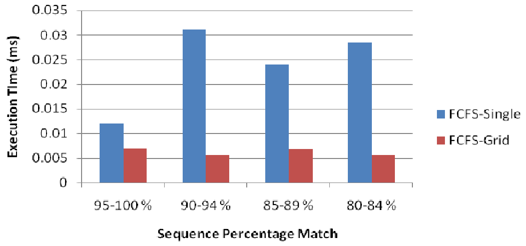

Figure 5. Execution Time of the Protein Sequences for the range 451 to 700

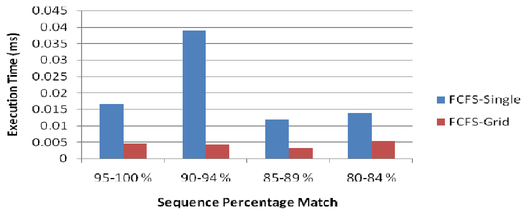

Figure 6. Execution Time of the Protein Sequences for the range 701 to 1000

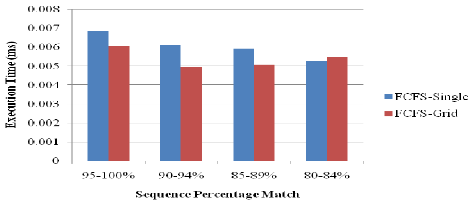

Figure 7. Execution Time of the Protein Sequences for the range 150 to 200

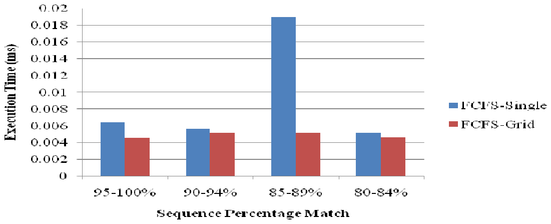

Figure 8. Execution Time of the Protein Sequences for the range 201 to 300

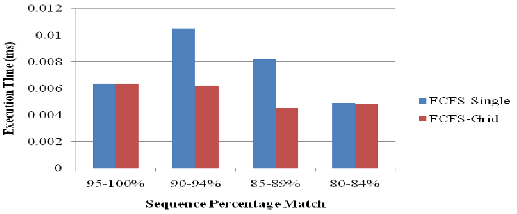

Figure 9. Execution Time of the Protein Sequences for the range 301 to 450

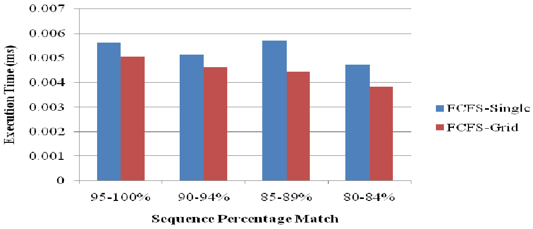

Figure 10. Execution Time of the Protein Sequences for the range 451 to 700

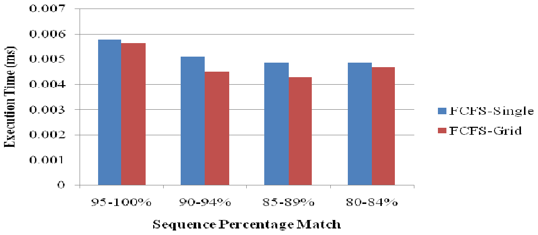

Figure 11. Execution Time of the Protein Sequences for the range 701 to 1000

TABLE I. EXECUTION TIME (MILLISECONDS) OF THE PROTEIN SEQUENCES FOR THE RANGE 150 TO 200

Percentage Match	Execution Time(milliseconds)	
	FCFS(Single)	FCFS(Grid)
95-100%	0.0063337	0.0047963
90-94%	0.0055003	0.0050620
85-89%	0.0060003	0.0070004
80-84%	0.00766711	0.0061225

TABLE II. EXECUTION TIME (MILLISECONDS) OF THE PROTEIN SEQUENCES FOR THE RANGE 201 TO 300

Percentage Match	Execution Time(milliseconds)	
	FCFS(Single)	FCFS(Grid)
95-100%	0.0202508	0.0066336
90-94%	0.0085342	0.0054002
85-89%	0.0075004	0.0056670
80-84%	0.0290017	0.0045002

TABLE III. EXECUTION TIME (MILLISECONDS) OF THE PROTEIN SEQUENCES FOR THE RANGE 301 TO 450

Percentage Match	Execution Time(milliseconds)	
	FCFS(Single)	FCFS(Grid)
95-100%	0.0280001	0.0155000
90-94%	0.0166730	0.0082824
85-89%	0.0320019	0.0065003
80-84%	0.0280016	0.0065878

TABLE IV. EXECUTION TIME (MILLISECONDS) OF THE PROTEIN SEQUENCES FOR THE RANGE 451 TO 700

Percentage Match	Execution Time(milliseconds)	
	FCFS(Single)	FCFS(Grid)
95-100%	0.0120000	0.0068892
90-94%	0.0310012	0.0056058
85-89%	0.0240010	0.0068545
80-84%	0.0285017	0.0057288

TABLE V. EXECUTION TIME (MILLISECONDS) OF THE PROTEIN SEQUENCES FOR THE RANGE 701 TO 1000

Percentage Match	Execution Time(milliseconds)	
	FCFS(Single)	FCFS(Grid)
95-100%	0.0166675	0.0044447
90-94%	0.0390012	0.0043335
85-89%	0.0120016	0.0032002
80-84%	0.0140012	0.0051669

TABLE VI. EXECUTION TIME (MILLISECONDS) OF THE PROTEIN SEQUENCES FOR THE RANGE 150 TO 200

Percentage Match	Execution Time(milliseconds)	
	FCFS(Single)	FCFS(Grid)
95-100%	0.0068725	0.0060747
90-94%	0.0061035	0.0049407
85-89%	0.0059169	0.0050835
80-84%	0.0052576	0.0054819

TABLE VII. EXECUTION TIME (MILLISECONDS) OF THE PROTEIN SEQUENCES FOR THE RANGE 201 TO 300

Percentage Match	Execution Time(milliseconds)	
	FCFS(Single)	FCFS(Grid)
95-100%	0.0064177	0.0045574
90-94%	0.0056256	0.0052046
85-89%	0.0189905	0.0052108
80-84%	0.0052145	0.0047145

TABLE VIII. EXECUTION TIME (MILLISECONDS) OF THE PROTEIN SEQUENCES FOR THE RANGE 301 TO 450

Percentage Match	Execution Time(milliseconds)	
	FCFS(Single)	FCFS(Grid)
95-100%	0.0063573	0.0063730
90-94%	0.0105033	0.0062318
85-89%	0.0081879	0.0045627
80-84%	0.0048974	0.0048502

TABLE IX. EXECUTION TIME (MILLISECONDS) OF THE PROTEIN SEQUENCES FOR THE RANGE 451 TO 700

Percentage Match	Execution Time(milliseconds)	
	FCFS(Single)	FCFS(Grid)
95-100%	0.0056398	0.0050689
90-94%	0.0051576	0.0046328
85-89%	0.0057160	0.0044282
80-84%	0.0047218	0.0038467

TABLE X. EXECUTION TIME (MILLISECONDS) OF THE PROTEIN SEQUENCES FOR THE RANGE 701 TO 1000

Percentage Match	Execution Time(milliseconds)	
	FCFS(Single)	FCFS(Grid)
95-100%	0.0057625	0.0056636
90-94%	0.0051208	0.0045022
85-89%	0.0048600	0.0043002
80-84%	0.0048512	0.0046815

This experiment is performed with 1000 protein sequences. The protein sequence given by the user is compared with that of in the database. As the number of sequence is increased, the time taken to find the similar sequence also increases sharply. If this is performed in Grid environment, where the protein sequences are distributed, the time for searching will be less.

From the experimental results, it is inferred that the execution time of FCFS in Grid environment gives minimum time when compared with FCFS in single system. In some cases the execution time of FCFS in single system is minimum when compared with FCFS in Grid environment. That is the execution time of FCFS in Grid environment increases by 5%. This is due to the reason that the sequences are arranged randomly.

The execution time of sequence using FCFS in single system and in Grid environment are 5% same for few cases. Especially FCFS in grid environment performs better when compared to other scheduling algorithms.

VI. Conclusion

This research work has been implemented in Grid environment, to reduce the time taken to find similar protein sequence for Homology Modeling. The implemented algorithm works in manner that the grid systems process more than one sequence in a single process, so the search result is produced for more than one input sequence at a time.

The protein sequence search in a single system will result in time delay and the user has to wait for long time to find each sequence result. To overcome this time delay and waiting time of a user, the homology modeling is implemented in grid environment, where more than one sequence can be input to the system in the grid environment at a time and search in PDB is done. This results in minimum sequence search time using FCFS in grid when compared with FCFS in single system.

VII. Future Enhancement

This research work has been implemented in Grid environment, to reduce the time taken to find similar protein sequence for Homology Modeling. The implemented algorithm works in manner that the grid systems process more than one sequence in a single process, so the search result is produced for more than one input sequence at a time. The protein sequence search in a single system will result in time delay and the user has to wait for long time to find each sequence result. To overcome this time delay and waiting time of a user, the homology modeling is implemented in grid environment, where more than one sequence can be input to the system in the grid environment at a time and search in PDB is done. This results in minimum sequence search time using FCFS in grid when compared with FCFS in single system.

References

[1] “A Gentle Introduction to Grid Computing and Technologies (PDF)”, May 2005.

[2] “The Grid Cafe – The place for everybody to learn about grid computing”, CERN, December 2008.

[3] Ian Foster, “What is the Grid? A Third Point Checklist”, Argonne National Laboratory & University of Chicago, July 20, 2002.

[4] Ian Foster, Carl Kesselman, “The Grid: Blueprint for a New Computing Infrastructure”, Morgan Kauffmann Publishers, USA, 1999.

[5] Maozhen Li, Mark Baker, “The Grid Core Technologies”, A John Wiley & Sons, Inc., 2005.

[6] F.Davoli, N.Meyer, R.Pugliese, S.Zappatore, et. al, “Grid enabled remote instrumentation”, 2008.

[7] L.J.Zhang, J.Y.Chung and Q.Zhou, “Developing grid computing applications, part 1: Introduction of a grid architecture and toolkit for building grid solutions”, IBM Corporation New York, October 2002.

[8] Manjula.K.A, Dr.G.Raju, “A Study on Applications of Grid Computing in Bioinformatics”, IJCA Special Issue on Computer Aided Soft Computing Techniques for Imaging and Biomedical Applications, 2010.

[9] Guoshi Xu, Fakai Lu, Huashan Yu, Zhuoqun Xu, “A Distributed Parallel Computing Environment for Bioinformatics Problems”, 6th International Conference on Grid and Cooperative Computing, 2007, 593-599.

[10] Zhang Y and Skolnick J, “The protein structure prediction problem could be solved using the current PDB library”. Proc Natl Acad Sci USA doi:10.1073/pnas.0407152101. PMC 545829. PMID 15653774, 102(4) 2005 1029-34.

[11] Abdulal W and Ramachandram, S, “Reliability-Aware Genetic Scheduling Algorithm in Grid Environment”, International conference on Communication Systems and Network Technologies (CSNT), Print ISBN 978-1-4577-0543-4 2011 673-677.

[12] Punhani A, Kumar S, Chaudhary R, Sharma A.K, “A Cpu scheduling based on multi criteria with the help of evolutionary algorithm”, 2nd IEEE International Conference on Parallel Distributed and Grid Computing (PDGC), 2012, pp. 730 – 734.

[13] Reddy K.H.K, Roy D.S, “A hierarchical load balancing algorithm for efficient job scheduling in a computational grid testbed”, 1st International Conference on Recent Advances in Information Technology (RAIT), 2012 1st, pp. 363–368.

[14] Amudha T and Dhivyapraphа T, “Qos priority based scheduling algorithm and proposed framework for task scheduling in a grid environment”, International Conference on Recent Trends in Information Technology, 2011, pp. 650-655.

[15] Ravi V.T, Becchi M, Agrawal G, Chakradhar S, “ValuePack: Value-based scheduling framework for CPU-GPU clusters, International Conference on High Performance Computing”, Networking, Storage and Analysis (SC), 2012, pp. 1–12.

[16] K.Vivekanandan, D.Ramyachitra, “Bacteria foraging optimization for protein sequence analysis on the grid”, Future Generation computer systems”, Elsevier Journal, 28 (2012) 647-656.

[17] Suyu Mei, Wang Fei, “Homology based Multi-instance Kernel combination for Gram-negative protein subcelluar localization”, International Conference on Bioinformatics and Biomedical Technology (ICBBT), 2010, pp. 5-9.

[18] Williamson AR, “Creating a structural genomics consortium”, Nat Struct Biol 7 S1(11s):953, 2000.

[19] http://iitb.vlab.co.in/?sub=41&brch=118&sim=657&cnt=1.

[20] Marti-Renom MA, Stuart AC, Fiser A, Sanchez R, Melo F, Sali A, “Comparative protein structure modeling of genes and genomes”, Annu Rev Biophys Biomol Struct 29, 2009 291–325.

[21] Pradeep Kumar P, Ramyachitra D, “Architectural Frame Work for Homology Modeling in Grid Environment”, International Conference on Research Trends in Computer Technologies, January 2013.

[22] Tramontano,A., Leplae,R. and Morea,V., “Analysis and assessment of comparative modeling predictions in CASP4”, Proteins, 45(5), 2001 22-38.