Indian Streams Research Journal
Vol - I , ISSUE - V [June 2011] : information Technolog **ISSN:-2230-7850**

Article : Scheduling approaches for protein sequence analysis on the grid

Author : D. Ramyachitra [Bharathiar University, Coimbatore]

Abstract

Most of the scientific applications requires a coordination of resources to solve their problem. In particular, every day, hundreds and hundreds of protein sequences are deposited in the data banks by the researcher. If a new researcher wants to find a similar protein sequence like the one which he has, an extensive database search is required. If the database is distributed in grid environment, the search becomes easy. In practice, there are a number of schedulers applicable to the grid environment. These schedulers can direct the users query to appropriate resources to find the similar queries. Almost all the schedulers including the optimization techniques lead to load imbalance on the resources. A scheduler has to be designed in such a way that all the resources in the grid environment are balanced. This paper discusses about the protein sequence analysis on the grid, the schedulers available and load balancing in the grid environment.

- **Introduction**

Grid computing started as a project to link geographically dispersed supercomputers and now it has grown beyond its original intent. The Grid infrastructure can benefit many applications, including collaborative engineering, data exploration, high throughput computing and distributed supercomputing. The last decade has seen a substantial increase in commodity computer and network performance due to faster hardware and sophisticated software. Due to size and complexity of problems in the field of science, engineering and business, the current generation of supercomputers cannot effectively deal with these problems. A number of teams have conducted experimental studies on the coordinated use of geographically distributed resources to act as a single powerful computer which can be termed as metacomputing, scalable computing, internet computing, global computing and peer to peer or grid computing [1].

Grid resource management provides functionality for discovery and publishing of resources with scheduling, submission and monitoring of jobs. Since, the resources are geographically distributed under different ownerships, each owner have their own access policy, cost and various constraints. The resource owners also may charge different process to different grid users for their resource usage which may vary from time to time. These resource owners also will have an unique way of managing and scheduling resources and the grid schedulers must ensure that they do not conflict with resource owners policies [2].

Computational biology is undergoing a revolution from traditional compute intensive science conducted by individuals to a high-throughput, data driven science conducted by teams from academia and industry. The first computational models for biology and chemistry were developed for the classical von Neumann machine model, that is, for sequential, scalar processors. With the emergence of parallel computing, biological applications were developed to take advantage of distributed or shared memory and locally located disk space to execute a collection of tasks. Applications that compute electronic interactions of a protein fragment are examples of programs developed to take advantage of emerging computational technologies. To support these applications requires analyze or process immense amounts of input/output data. The applications of computational biology are characterized by high-throughput, high technology and data driven. Also, the applications require large scale data analysis and management, wide access through web portals and visualization. To meet the above said requirements, grid computing provides accessible computational and data management environment for bioinformatics applications that requires high end resources above and beyond what is available to them locally. The grid also offers great promise for many applications in bioinformatics that will lead to great improvements in health care and the quality of life [3].

- **Protein Sequence Analysis**

Sequence alignment is the arrangement of sequences of DNA, RNA or protein to identify regions of similarity that may be a consequence of functional, structural or evolutionary relationships between the sequences. The sequences of nucleotide or amino acid residues are typically represented as rows within a matrix. Sequence alignment is used for visualizing the relationships between residues in a collection

of evolutionary or structurally related proteins. Given the amino acid sequences of a set of proteins to be compared, an alignment displays the residues for each protein on a single line, with gaps inserted so that equivalent residues appear in the same column [4]. Aligning the sequences is an interesting problem where, very shot or similar sequences can be aligned by hand whereas lengthy, highly variable or extremely numerous sequences requires human knowledge to be applied in constructing algorithms to produce high quality sequence alignments. Local and global alignments are the computational approaches for sequence alignment. Local alignments identify the regions of similarity within long sequences and this alignment is preferable, but more difficult to calculate because of the additional challenge of identifying the regions of similarity. Calculation of global alignment is a form of optimization which forces the alignment to span the entire length of all query sequences. Computational algorithms applied to the sequence alignment problem include dynamic programming, heuristic algorithms or probabilistic methods.

Evolutionary related proteins have similar sequences. Naturally occurring homologous proteins have similar stable tertiary structures. Artificial protein designed by the Regan group has 50% identical sequence to a specific binding protein of B1 domain [5]. A huge amount of DNA sequence data is available and databases are growing exponentially. Analysis of this enormous amount of data including hundreds and hundreds of genomes from prokaryotes and eukaryotes has given a rise to the field of bioinformatics. Also, challenges of genome sequence analysis include understanding of diseases, gene regulation and metabolic pathway reconstruction. Many bioinformatic tools are developed to identify genes that encode functional properties or RNA. These tools are made available to scientists and the problem is which tool to be used and how best to obtain the answers [6]. As said in the introduction, protein sequence analysis requires large scale data analysis and management, wide access through web portals and visulaization. To meet this challenge, grid computing can provide computational and data management environment as this requires high end resources that are not available locally.

- **Scheduling algorithms**

One of the primary goals of computational grids is access to computing resources over which the user may not have direct control [7] or the main goal of which is to aggregate resources culled from a global resource pool for use by applications and their users. When the user submits the job to the grid, it has to be executed with some constraints like time, deadline and cost. Job scheduling is a NP complete problem which is responsible for management of jobs, such as allocating resources needed for any specific jobs, partitioning of jobs to schedule parallel execution of tasks, data management and service level management capabilities. In the broad sense, scheduling problem can be defined as static or dynamic [8]. In static scheduling, all the information about the task and resources will be known before execution begins. In dynamic scheduling, only little information will be known and the scheduling execution must be made at the time of arrival of the job. Some of the scheduling algorithms available are First Come First Serve (FCFS), Earliest Deadline First (EDF), Opportunistic Load Balancing (OLB), Minimum Execution Time (MET), Minimum Completion Time (MCT), Min Min, Max Min, Tabu Search (TS), Simulated Annealing (SA), Genetic Algorithm (GA), Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), Artificial Bee Colony (ABC) etc. A scheduling algorithm will be used in the grid environment to schedule the jobs to the appropriate resources. An appropriate scheduling algorithm has to be chosen for a particular application. These scheduling algorithms can be used to schedule the queries given by the user to appropriate resources or the databases to find the matching sequence.

- **Protein sequence analysis on the grid**

There are similar protein sequences found in different species. To find out the similar sequences, search has to be conducted in primary databases. If the similar sequence could not be found in the primary database, then secondary database can be searched looking for the matches to the patterns they contain. Basic Local Alignment (BLAST) tool is used to compare a novel sequence with those contained in nucleotide and protein databases by aligning the novel sequence with previously characterized genes. The result of this search gives the regions of sequence similarity, its evolutionary history and homology with other sequences in the databases. Sequences may be in FASTA format, NCBI accession numbers or GIs(GenBank Identifiers). For non published sequences, FASTA is the only format that can be used. The BLAST programs first looks for similar segments between the query sequence and a database sequence, and evaluates the statistical

significance of any matches that were found, and finally reports only those matches that satisfy a user selectable threshold of significance.

Databases are mostly regulated by users. Large degree of redundancy exists within database and between databases and lack of standards for annotation. These factors should be taken into consideration while comparing the sequence. Searching on the large sequence database is carried out by sequentially scanning the entire database to identify the desired sequences. When dealing with large amount of sequences, this approach suffers from prolonged response time. Alternative methods should be used for fast retrieval of sequences.

Considering protein data bank alone, it is updated weekly. In 2010, total structures for protein and nucleic acid were approximately 70000(Wikipedia). If the database is stored in several computers on grid using globus toolkit, it is possible, to retrieve the sequence in less amount of time. K. Somasundaram et al. in [9] has proposed Grid architecture for searching the distributed, heterogeneous genomic databases which contain protein sequences to speed up the large scale sequence data analysis and to perform alignment for residues match.

The protein database can be distributed in a grid environment and a scheduler can be used to schedule the queries from the user to the appropriate machine which can process the query as fast as possible. The scheduler can be constructed based constructive heuristics such as first come first served scheduling , Minimum Execution Time, Minimum Completion Time, Opportunistic Load Balancing etc. There are also optimization techniques such as genetic algorithm, ant colony optimization, particle swarm optimization, artificial bee colony etc that can be used for scheduling which are more suitable for NP complete problems. Any one of these algorithms can be used to construct the scheduler for efficient scheduling of the user queries to the appropriate machines. With the scheduling, load balancing can also be done to achieve fast execution of the user jobs. If it is found that there is a long waiting queue, then the query can be redirected to another machine where that query can be processed faster.

First Come First Serve scheduling can be used for scheduling the queries for protein sequence analysis, but the problem here is, some machine may be overloaded with long queries and some may be lightly loaded with short queries. This may lead to load imbalance on the machines. In the case of Minimum Execution Time, it will consider the queries with shorter execution time first and direct it to the machines which also lead to imbalance in the load on the machines. In the case of Minimum Completion Time also, queries with minimum completion time gets scheduled to the machine first. In the case of Opportunistic Load Balancing, it does not considers the execution or completion time of the queries, but just schedules the queries to the machine that is available. In all these constructive heuristics, load imbalance occurs. In the case of optimization techniques also, the same problem occurs of load imbalance. A scheduling algorithm has to be designed and developed such that all the resources are used efficiently to the maximum so that the users jobs can be executed at a faster rate.

- **Conclusion**

This paper gives a brief discussion on the grid computing and protein sequence analysis. Further, it gives a brief note on the applications which cannot be run using the local resources but at the same time can be executed on grid environment. Protein sequence analysis which involves an extensive search on the database as thousands of sequences are deposited on the databank frequently, if run on a grid, the search can be done in a fraction of millisecond. This paper discusses this issue of how a query can be scheduled to the appropriate resource on the grid environment. It also discusses about balancing the load on the resources as some resource may be lightly loaded and some may be heavily loaded.

Acknowledgement

This work has been funded by University Grants Commission, India under the scheme UGC-MRP.

References

• Mark Baker, Rajkumar Buyya, Domenico Laforenza, Grids and Grid technologies for wide area distributed computing, Software – Practice and Experience, 2002.

• Ajith Abraham, Hongbo Liu, Crina Grosan and Fatos Xhafa, Nature Inspired Meta-heuristics for Grid Scheduling: Single and Multi-objective Optimization Approaches, F.Xhafa, A.Abraham (Eds.): Meta. For Sched. In Distri. Comp. Envi., SCI 146, pp. 247-272, 2008.

• Kim Baldridge and Philip E. Bourne , The new biology and the Grid, Grid Computing – Making the Global Infrastructure a Reality. Edited by F. Berman, A.Hey and G.Fox, John Wiley and Sons, Ltd., 2003, ISBN: 0-470-85319-0.

• Chuong B. Do, Kazutaka Katoh, Protein Multiple Sequence Alignment, From: Methods in Molecular Biology, vol. 484: Functional Proteomics: Methods and Protocols, Edited By: J.D. Thompson et al., DOI: 10.1007/978-1-59745-398-1, Humana Press, Totowa, NJ.

• Szymon Kaczanowski, Piotr Zielenkiewicz, Why similar protein sequences encode similar three-dimensional structures, Theoretical Chemistry Accounts: Theory, Computation, and Modeling (Theoretica Chimica Acta), vol. 125, nos. 3-6, 643-650.

• Bernd H.A. Rehm and Frank Reinecke, Gene/Protein Sequence Analysis, A compilation of bioinformatic tools, From : Molecular Biomethods Handbook, 2nd Edition, Edited By: J.M. Walker and R. Rapley, Humana Press, Totowa, NJ.

• Bryan MacKinnon, Commercial Computational Grids: A Road Map, Ubiquity-The ACM IT Magazine and Forum, 4(14), 2003.

• Yu-Kwong Kwok, Ishfaq Ahmad, Static Scheduling Algorithms for Allocating Directed Task Graphs to Multiprocessors, ACM Computing Surveys, 31(4), 1999.

• K.Somasundaram, S.Radhakrishnan, Nimble Protein Sequence Alignment in a Grid, Journal of Computer Science 4(1): 36 - 41, 2008.