A Services Oriented System for Bioinformatics Applications on the Grid

Giovanni ALOISIO, Massimo CAFARO, Italo EPICOCO, Sandro FIORE and Maria MIRTO [1]

CACT-NNL/CNR-INFM, CMCC, University of Salento, Lecce, Italy

Abstract. This paper describes the evolution of the main services of the ProGenGrid (Proteomics & Genomics Grid) system, a distributed and ubiquitous grid environment ("virtual laboratory"), based on Workflow and supporting the design, execution and monitoring of "in silico" experiments in bioinformatics.

ProGenGrid is a Grid-based Problem Solving Environment that allows the composition of data sources and bioinformatics programs wrapped as Web Services (WS). The use of WS provides ease of use and fosters re-use. The resulting workflow of WS is then scheduled on the Grid, leveraging Grid-middleware services. In particular, ProGenGrid offers a modular bag of services and currently is focused on the biological simulation of two important bioinformatics problems: prediction of the secondary structure of proteins, and sequence alignment of proteins. Both services are based on an enhanced data access service.

Keywords: Grid Problem Solving Environments, Biological Data Banks, Web Services, Grid Computing.

Introduction

Biological experiments are relatively expensive, forcing the scientists to focus on advanced design for experimentation as part of a whole-chain research approach. Furthermore, the data generally contain information outside the scope of the original experiment. Hence, to maximize the scope of experiments, biological data needs to be reusable, shareable and suitable for "in silico" experiments. An in "silico" experiment is a procedure that uses computer-based information repositories and computational analysis to test a hypothesis, derive a summary, search for patterns, or demonstrate a known fact.

In Bioinformatics, an experiment is generally characterized by a search in huge biological databases and by the execution of tools that need to access such data intensively. Moreover, the majority of the experiments are run repeatedly, orchestrating many resources to produce sets of data to be analyzed and validated. Such experiments produce a great deal of fragmented data, each coming from a different resource.

[1] Correspondence to: Maria Mirto, Center for Advanced Computational Technologies of National Nanotechnology Laboratory (NNL/CNR-INFM), University of Salento, Lecce, Italy, Euro-Mediterranean Centre for Climate Change, Lecce, Italy Tel.: +39 0832 298123; Fax: +39 0832 298173; E-mail: maria.mirto@unile.it.

Executing bioinformatics applications involves: i) accessing different biological data banks; ii) accessing experimental data; iii) executing different bioinformatics tools, possibly combining intermediate results, and finally iv) providing results to the users.

There are many open problems arising when implementing this kind of pipeline: i) the biological data banks often use different formats for the same biological entity and are accessible through web interfaces mainly manually; ii) the volume of data grows exponentially: technological advances in biology have made it possible for laboratories to generate un unprecedented amount of data; iii) the data are updated frequently: many data banks are updated daily (e.g. GenBank [1]), whereas others are updated regularly (e.g. PDB [2] is updated on a weekly basis); iv) the data are stored sparsely: each area of molecular biology generates its own databases such as the GenBank DNA database, the protein UniProt database [3], proprietary databases for storing structural information about compounds, databases for storing physical properties and activities of chemical entities, etc.; v) the data are accessed intensively: the public availability of data enables free and easy access for scientists. These data are very often exchanged by researchers as well as among databases.

Therefore, a complete and integrated software environment to execute biological applications is needed, in order to assist scientists and researchers during management and coordination of all of the tasks of an "silico" experiment, while providing large computational power.

Grid Problem Solving Environments (GPSEs) [4] can offer a solution for handling and analyzing so much disparate data connecting many computers within and among institutions through middleware software. Interaction between bioinformatics tools and biological databases should be simplified and each component (i.e. biological data banks, experimental data, bioinformatics tools) should be seen as an atomic service (Web Service) [5] that can be easily integrated in different systems, through standard interfaces and protocols. Moreover, to implement high throughput experiments, the bioinformatics tools need to be Grid-accessible and, to increase performance, they should be parallelized.

This paper describes the design and development of a web based GPSE, named ProGenGrid (Proteomics and Genomics Grid), which allows the design and execution of biological "in silico" experiments on the Grid. Distributed bioinformatics applications are described through a Workflow [6] of Grid services that wrap biological data sources and bioinformatics tools. Main features of ProGenGrid are: i) an integrated environment, where e-scientists can discover available tools and compose them in a graph (Workflow editor). The system guides the user in the selection of these tools, interacting with her, if the graph is not correct; ii) a system for running the experiments on a computational grid and managing the results (Workflow enactment service); iii) enhanced services to access the data, for grid-based sequence alignment and grid-based secondary structure prediction; in the latter case, the predictor is automatically and periodically trained again in order to improve the prediction accuracy. A fundamental feature of this system is that when the basic legacy software packages are updated by third parties, the new versions can simply replace the older ones, since these tools are embedded in high level services and exposed through a WS interface. Moreover, existing alignment services at the NCBI such as BLAST [7], PSI-BLAST [8], etc., although efficient, allow submitting only a sequence at a time, whereas ProGenGrid allows carrying out many submissions in one shot, specifying more sequences as input. Finally, the implemented Grid-based BLAST and PSI-BLAST services exploit parallelism on clusters.

These important results have been obtained thanks to the use of several Grid Middleware services developed at the CACT/NNL of the University of Salento, Lecce and offered by the Grid Resource Broker (GRB) [9] and Grid Relational Catalog (GRelC) [10] projects, which will be briefly described in the following sections.

The first version of ProGenGrid [11] has been developed as a client-server application. Recently, a GRB based Grid Portal has been developed leveraging GRB middleware services. Both data banks and software tools are installed on geographically distributed resources, in a production grid named SPACI (Southern Partnership for Advanced Computational Infrastructures), and are deployed through the Grid Portal, as in a "virtual laboratory".

The remainder of the paper is organized as follows. In Section 1, the ProGenGrid architecture and its grid services are presented. Section 2 describes the implementation of the system and a case study is presented, while Section 3 recalls related works. Finally in Section 4 we draw our conclusions and discuss future work.

1. The Service-based Architecture of ProGenGrid

The ProGenGrid architecture includes the following logical layers, each one offering specific services (Fig. 1): *Core Services* and *Application Services*.

Application Services include biological tools and the Workflow editor. Such services are based on Grid Middleware services, named Core Grid Services. They offer Data Management services to access distributed biological data banks (both flat files and relational databases), Resource Management services to discover existing analysis tools made available as Web Services and to schedule, submit and monitor these jobs. We describe these services in the next sections.

1.1. Core Grid Services

This section describes the core services of ProGenGrid, respectively the data management and resource management services.

1.1.1. Data Management

In order to support high throughput applications such as alignment tools, we have implemented an enhanced data access service, named Split&Merge [12], to retrieve biological data from both flat files and relational databases. The Split&Merge Data Access Service is a data grid system that allows retrieving biological data and splitting them in order to distribute the computation on several grid nodes.

The Split&Merge architecture is based on the following services: GRelC Data Access (GDA) [10], GRelC Data Storage (GDS) [10], and Splitter/Merger. This service, described in [12], has been extended with the following components, for translating biological data banks into a metadata DBMS: Wrapper (WRP) and GRelC Translator (GTrs).

The main feature of the system is a query protocol that manages the query, distributing the entire record set among several computational grid nodes and processing the subsets locally.

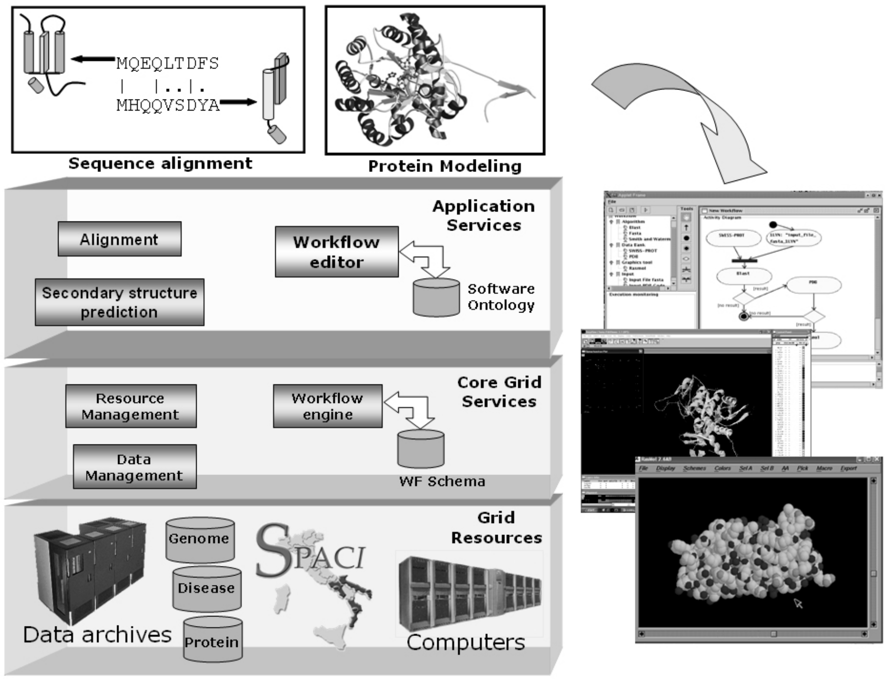

Figure 1. ProGenGrid Architecture

Detailing each service, the GDA acts as a standard front-end for each data bank. This service provides the basic primitives to transparently access and interact with different data sources, concealing the back-end heterogeneity and other low level details (it is built on top of the Globus Toolkit [13], libxml2 and proprietary libraries of several DBMS). The GDA is developed as a Grid Security Infrastructure (GSI) enabled Web Service. To date, the GDA provides (by means of wrappers written using proprietary libraries) dynamic binding to several physical RDBMS such as Oracle or PostgreSQL and ad hoc wrappers for the UniProtKB [3], UTRdb [14], PDBaa [15], Nr [16] and PDB [2] databases.

GDS provides a large set of services to manage group of files on a storage resource (i.e. a Disk Resource Manager, DRM) and to share/make them available among/to a community of users.

Splitter/Merger performs a two phase (split & merge) data management activity (interacting with the GDS). In the former phase the record set is split into several fragments (input data) and uploaded to the GDS, whereas in the latter this component downloads the partial results (output data) from the GDS and merges them composing the final user response.

1.1.2. Resource Management

Resource Management services allow brokering, scheduling and monitoring of jobs on a Computational Grid. ProGenGrid relies on the GRB Scheduler for these functions. It acts as a meta scheduler for the available grid resources and has been designed to be fully compliant with respect to the Job Submission Description Language (JSDL) specification. The JSDL language has been however extended in order to provide better support for:

- batch jobs: these jobs involve staging of the input and output. MPI/OpenMP jobs are also batch jobs;
- interactive jobs: jobs in which the user can follow the job progress watching the job standard output/error from the client browser;
- parameter sweep: the same application executed several times with different inputs. The GRB scheduler balances the workload on the grid resources;
- workflow jobs: the user composes some jobs in a graph. The GRB scheduler submits all of the sub-jobs of the workflow job without violating the precedence constraints.

A simple job for sequence alignment can be a batch job if a single sequence is run, otherwise it can be considered as a parameter sweep job. Moreover, if adjustment of the input requires several operations, the alignment is a workflow job. Other GRB features are: i) it retrieves grid information using the iGrid or MDS2 (MDS4 planned) information systems; ii) it manages data using GridFTP to automatically stage the executable input and output file(s); iii) it supports credential delegation and single sign-on to the Grid; iv) it configures grid resources and services.

1.2. Application Services

The main application services developed in ProGenGrid are a toolkit for parallel alignment of sequence and a system for the prediction of proteins secondary structure. We now describes these services.

1.2.1. A Grid-based Tool for the Alignment of Sequences on the Grid

One of the key features of this service is based on a technological solution to improve the execution of several alignments tools involved in biological "in silico" experiments. There are two approaches to improve the execution of alignment tools: fragmenting the input and/or the databank on which the search is carried out. We have chosen to use both database and query segmentation for BLAST and query segmentation for PSI-BLAST to achieve this goal. We use this optmized version of PSI-Blast in the Secondary Structure Predictor described in the next section. In the former case (BLAST), if more sequences are considered, the performances are better w.r.t. the sequential case [17] whereas in the latter (PSI-BLAST), we have chosen query fragmentation because many queries are involved in the computation and it seems that database fragmentation can not be beneficial. However, in a future work we would like to exploit an hybrid approach, using both database and query segmentation, depending on the problem, the alignment tool, and the available resources.

The proposed system, named BioGAT, exploits a Grid Platform based on the Globus Toolkit 4.0.3 and in which modules are developed in C/C++ and deployed as Web Services (WS), exploiting the gSOAP Toolkit [18] (with GSI support available as a gSOAP plug-in). BioGAT allows splitting the computation on a biological dataset among several computational nodes. It does not matter how the data is partitioned or what subset is computed by a client, because there are not dependencies among data stored in the same or in different subsets (in the BLAST tool case only an adjustment of several statistical parameters must be made at the end of the computation). Our aim is to allocate at runtime these subsets on the idlest machines taking into account some dynamic parameters, such as, for instance, the CPU availability of the grid nodes.

There is a difference related to the involved data banks when BLAST or PSI-BLAST computations are considered. In the former case, computation is made on chunks of databases, so each fragment must be indexed. In the latter case, the fragmentation is on the query so the involved data banks must be located on the same machine on which the job is run and indexing is made just once.

1.2.2. Grid Secondary Structure Predictor

Another important service of ProGenGrid is the prediction of the secondary and tertiary structure of proteins starting by their amino acid sequence. To date, we cover secondary structure prediction but in the future the full deployment of the system will also allow tertiary structure prediction.

The protein structure prediction problem is the problem of predicting the 3D structure of the protein (its native conformation) when the list of amino acids is known. Native conformation determines the biological function of the protein [19]. Native conformations are largely built from Secondary Structure Elements (SSEs), which are local motifs consisting of short, consecutive parts of the amino acid sequence having a very regular conformation. Some of them are: $\alpha - helix$, $\beta - sheet$ and *random coil*.

Computational methods available for secondary structure prediction are based on different approaches. The most recent (defined as third generation) are based on neural networks (NNs) and allow carrying out the prediction with an accuracy between 72% and 80%, on specific proteins classes [20,21]. The aim of these techniques is to assign secondary structure elements to segments of the amino acids sequence of unknown structure, starting with the knowledge of a sufficient large number of proteins, whose sequence and three-dimensional structure are known, used as training set.

Proteins that are similar in sequence are likely to have similar structure and function. Hence similarities can give hints about the evolutionary history of certain sequences. By the analysis of different methodologies, it emerges that the quality of prediction has been improved both by the increasing growth of the protein databases and through the application of evolutionary information related to a single protein. Indeed, secondary structure prediction accuracy has improved to over 70% by incorporating the evolutionary information found in multiple sequence alignment (MSA).

An integrated Grid system for the prediction of secondary structure, based on the frequent regular updating of the training protein dataset, has been implemented. The predictor model is based on a feed forward Multi Layer Perceptron (MLP) neural network which is trained with the back-propagation algorithm, and it has been designed using both existing legacy software and novel components. The predictor is based on the evolutionary information found in MSA and calculated by using the PSI-BLAST tool [8] on a set of proteins of known structure (training set), provided by the University of California's research group. The testing set is R126 [20], which has been used by Rost and Sander in their earlier studies on protein secondary structure prediction. Low correlation among these two protein sets yields a reliable way of evaluating predictor results. A comparison of the system with other predictors has been carried out on the same test set and the proposed system is more accurate of an analyzed predictor, JPred [21], by about 2%.

The system exploits advanced mechanisms for management of biological data and allows the automation of the computational process in order to reduce the network learning time and to guarantee periodic updating of the network. Using Grid technology, the training time has been remarkably reduced and the procedure is completely automatic, whereas the implemented predictor has an accuracy percentage of about 76% on a well known testing set available in literature.

1.2.3. Workflow Editor

In order to support complex bioinformatics experiments, a workflow management system composed of an editor and an enactment service has been deployed.

The first prototype was developed as a client-server architecture; the current implementation provides a grid based portal to discover, compose, execute and monitor jobs. The editor, developed in Java, exploits a graphical notation based on UML. The data flow is described as activity diagrams and an opportune workflow language specification is used. This is translated into JSDL and sent to the GRB scheduler. Graph nodes are classified as:

- **execution node**: these nodes represent a job to be executed remotely on a brokered grid resource. This job can be a single batch job or recursively it can represent either a parameter sweep job or a workflow job;
- **condition node**: this node denotes a condition to be evaluated in order to control at runtime the execution flow of the graph. The condition is based on the execution of a test job and on further evaluation of the condition expressed on the outputs. The expression to be evaluated can be constructed using either a specified input file, a specified output file or standard output/error produced by the test job;
- **storage node**: these nodes represent single stage-in or stage-out resources.

Moreover, a relational schema describing the involved software (logical name, path, environment variables, etc.) has been developed using MySQL.

The enactment service allows executing workflows described by a directed graph and supports graphs with cycles and conditions.

2. ProGenGrid Portal Implementation

The GRB grid portal has been customized to expose the services of ProGenGrid. The resulting Grid portal provides the following services: Grid Configuration, Resource Status, File Transfer, Alignment Jobs, Secondary Structure Prediction.

The Grid Configuration section allows setting up grid user's profiles, retrieving the user's credentials, adding a new Virtual Organization (VO), configuring the grid with the possibility to associate grid machines to the users and finally, the Information Service monitors the grid status. It is possible to configure biological data banks, application tools, and to associate several databases to a software. This option allows (through the Alignment web interface) multiple data banks to be used. Finally, job status can be visualized.

Resource Status allows querying several information services to retrieve a resource status, whereas File Transfer allows using GridFTP to transfer files.

Alignment jobs allows submitting sequences for BLAST and PSI-BLAST, whereas for the secondary structure prediction it is possible to use an interface to submit the sequences whose structures need to be predicted. Finally, a workflow interface allows composing several applications in a graph, using drag-and-drop. The portal is available at the following url: https://sara.unile.it/cgi-bin/bioinfo/enter.

As a case study, an experiment regarding multiple sequence alignment (MSA) among each of the human proteins available in the UniProtKB database (about 70845 sequences, retrieved by the *uniprot_sprot_human.dat* and *uniprot_trembl_human.dat* flat files) and those stored in the UniProt NREF data bank has been carried out. Homologue sequences are hence matched to identify functional domains. PSI-BLAST is used for multiple alignment. Moreover, to reduce storage issues and taking into account that a single run of PSI-BLAST involves a set of iterations in order to converge to a solution (we used two iterations), an adjustment of the result based on results retrieved in the last iteration has been carried out. The experiment has been modelled using the ProGenGrid workflow. After running one experiment we produced about 70 thousands alignments files, carrying out an adjustment of the result (i.e., considering only the result of the last iteration). The experiment took about 4 days, using 30 1.4 GHz Itanium 2 processors.

3. Related Work

The Grid is proposed as the next generation infrastructure necessary to support and enable the collaboration of people and resources through highly capable computation and data management systems (Foster and Kesselman, 1998). A number of BioGrid projects are underway, including myGrid [22], GeneGrid [23] and BioSimGrid (Biomolecular Simulation Grid) [24]. These primarily focus on the sharing of computational resources, large-scale data movement and replication for simulations, remote instrumentation steering or high-throughput sequence analysis.

myGrid [22] is a project targeted at developing open source high-level middleware to support personalised in silico experiments in biology on a Grid. myGrid is building services for integration such as resource discovery, workflow enactment and distributed query processing. Additional services are needed to support the scientific method and best practice found at the bench but often neglected at the workstation, notably provenance management, change notification and personalisation. The target users of myGrid are tool and service providers who build applications for a community of biologists.

GeneGrid [23] is a collaborative industrial grid computing R&D project initiated by the Belfast e-Science Centre (BeSC) under the UK e-Science programme and supported by the Department of Trade &Industry. The project aims at providing a platform for scientists to access their collective resources, skills, experiences and results in a secure, reliable and scalable manner by creating a “Virtual Bioinformatics Laboratory”. GeneGrid provides seamless integration of a myriad of heterogeneous applications and datasets that span multiple administrative domains and locations across the globe, and presents these to scientist through a simple user-friendly interface. GeneGrid meets this challenge through the GeneGrid Portal - the user interface for the project.

Finally, the aim of the BioSimGrid project [24] is to make the results of large-scale computer simulations of biomolecules more accessible to the biological community. Such simulations of the motions of proteins are a key component in understanding how the structure of a protein is related to its dynamic function.

4. Conclusions and Future work

This paper presented a system, ProGenGrid, for bioinformatics. Advantages of ProGenGrid are: i) the possibility to design and execute a bioinformatics application on a Computational Grid; ii) the access to distributed data through a service that hides their heterogeneity, providing a set of useful libraries; iii) the use of common bioinformatics tools, through a unique interface; iv) the submission of many sequences at a glance to alignment tools; v) the possibility to carry out the training of the neural network using grid resources and the prediction of many sequences at a glance.

Future work will regard the possibility to experiment more complex case studies and full deployment of a service for the protein tertiary structure prediction.

Acknowledgments

This work was supported by the MIUR (Italian Ministry for Education, University and Research) in the LIBI (International Laboratory of BioInformatics) project, F.I.R.B. 2003, under grant RBLA039M7M.

References

[1] D. Benson, I. Karsch-Mizrachi, D. Lipman, and D. W. J. Ostell, “GenBank,” *Nucleic Acids Res*, vol. 1, no. 33, pp. 34–38, 2005.

[2] H. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. Bhat, and et al., “The Protein Data Bank,” *Nucleic Acids Res.*, no. 28, pp. 235–242, 2000, www.rcsb.org/pdb/.

[3] A. Bairoch, R. Apweiler, C. Wu, W. Barker, B. Boeckmann, S. Ferro, and et al., “The Universal Protein Resource (UniProt),” *Nucleic Acids Res.*, no. 33, pp. 154–159, 2005, http://www.uniprot.org.

[4] G. V. Laszewski, I. Foster, J. Gawor, P. Lane, N. Rehn, and M. Russell, “Designing Grid-based Problem Solving Environments and Portals,” in *Proceedings of International Conference on System Sciences (HICSS-34)*, 2001.

[5] F. Curbera, M. Duftler, R. Khalaf, N. Mukhi, W. Nagy, and S. Weerawarana, “Unraveling the Web Services Web - An Introduction to SOAP, WSDL, and UDDI,” *IEEE Internet Computing*, vol. 6, no. 2, pp. 86–93, 2002.

[6] W. management coalition reference model, http://www.wfmc.org/.

[7] S. Altschul, W. Gish, W. Miller, E. Myers, and D. Lipman, “Basic local alignment search tool,” *Nucleic Acids Res.*, vol. 3, no. 215, pp. 403–410, 1990, http://www.ncbi.nlm.nih.gov/BLAST/.

[8] S. Altschul, T. Madden, A. Schäffer, J.Zhang, Z. Zhang, W. Miller, and D. Lipman, “Gapped BLAST and PSI-BLAST: a new generation of protein database search programs,” *Nucleic Acids Res.*, vol. 17, no. 25, p. 3389Ű3402, 1997, http://www.ncbi.nlm.nih.gov/cgi-bin/BLAST/nph-psi.

[9] G. Aloisio, M. Cafaro, G. Carteni, I. Epicoco, S. Fiore, D. Lezzi, M. Mirto, and S. Mocavero, “The Grid Resource Broker Portal,” *to appear in Concurrency and Computation: Practice and Experience, Special Issue on Grid Computing Environments*, 2006.

[10] G. Aloisio, M. Cafaro, S. Fiore, and M. Mirto, “The Grid Relational Catalog Project,” in *Advances in Parallel Computing, "Grid Computing: The New Frontiers of High Performance Computing"*, L. Grandinetti, Ed., 2005.

[11] ——, “ProGenGrid: a Grid-enabled platform for Bioinformatics,” in *From Grid to HealthGrid, Proceedings of HealthGrid 2005, Tony Solomonides et al. (Eds), IOS Press*, vol. 112, 2005, pp. 113–126.

[12] ——, “A Split&Merge Data Management Architecture in a Grid Environment,” in *Proceedings of the 19th IEEE International Symposium on Computer-Based Medical Systems (IEEE CBMS 2006), Salt Lake City Utah (USA)*, 2006, pp. 739–744.

[13] I. Foster and C. Kesselman, “Globus Toolkit Version 4: Software for Service-Oriented Systems,” in *IFIP International Conference on Network and Parallel Computing*, S.-V. L. 3779, Ed., 2005, pp. 2–13.

[14] G. Pesole, S. Liuni, and et al., “UTRdb: a specialized database of 5’ and 3’ untranslated regions of eukaryotic mRNAs,” *Nucleic Acids Res.*, no. 27, pp. 188–191, 1999, http://www.ba.itb.cnr.it/UTR/.

[15] G. Wang and R. Dunbrack, “PISCES: improvements to a protein sequence culling server,” *Nucleic Acids Res. (in press)*, http://dunbrack.fccc.edu/Guoli/pisces_download.php.

[16] N. B. databases, ftp://ftp.ncbi.nlm.nih.gov/blast/db/.

[17] G. Aloisio, M. Cafaro, I. Epicoco, S. Fiore, and M. Mirto, “BioGAT: a Grid Toolkit for Bioinformatics Sequence Alignment,” in *Proceedings of the Grid-Enabling Legacy Applications and Supporting End Users Workshop (GELA), France*, 2006, pp. 77–85.

[18] G. Aloisio, M. Cafaro, D. Lezzi, and R. V. Engelen, “Secure Web Services with Globus GSI and gSOAP,” in *Proceedings of Euro-Par 2003, Lecture Notes in Computer Science, Springer-Verlag*, no. 2790, 2003, pp. 421–426.

[19] J. N. Onuchic and P. Wolynes, “Theory of protein folding,” *Current Opinion in Structural Biology*, vol. 14, pp. 70–75, 2004.

[20] B. Rost and C. Sander, “Prediction of protein secondary structure at better than 70% accuracy,” *Journal of Molecular Biology*, vol. 2, no. 232, pp. 584–599, 1993.

[21] J. Cuff, M. Clamp, A. Siddiqui, M. Finlay, and G. Barton, “JPred: a consensus secondary structure prediction server,” *Bioinformatics*, no. 14, pp. 892–893, 1998.

[22] R. Stevens, A. Robinson, and C. Goble, “myGrid: Personalised Bioinformatics on the Information Grid,” *Bioinformatics*, vol. 19, no. 1, pp. 302–302, 2004, http://www.mygrid.org.uk/.

[23] N. Kelly, P. Jithesh, and S. W. et al., “The GeneGrid Portal: A User Interface for a Virtual Bioinformatics Laboratory,” in *UK e-Science All Hands Meeting (AHM 2005)*, 2005, http://193.61.123.75:8080/GeneGrid/Portal.

[24] B. S. G. Project, www.biosimgrid.org/.