Grid Multi-Resolution Docking*

J.I. Garzón
Centro de Investigaciones Biológicas-CSIC
28040 Madrid (Spain)
garzon@cib.csic.es

E. Huedo
Facultad de Informática, Universidad Complutense
28040 Madrid (Spain)
ehuedo@fdi.ucm.es

R.S. Montero
Facultad de Informática, Universidad Complutense
28040 Madrid (Spain)
rubensm@dacya.ucm.es

I.M. Llorente
Facultad de Informática, Universidad Complutense
28040 Madrid (Spain)
llorente@dacya.ucm.es

P. Chacón
Centro de Investigaciones Biológicas-CSIC
28040 Madrid (Spain)
pablo@cib.csic.es

Abstract

Detailed knowledge of macromolecular structure is essential for the understanding of how the cellular machines work. Rigid body fitting is the common way to interpret the information contained in a 3D electron microscope (3DEM) medium-low resolution map in terms of its available atomic structural components. This fitting process, termed multi-resolution docking, consists in localizing atomic resolution structures into the 3DEM map by means of an exhaustive search of all possible relative rotations and translations. This exhaustive search is a highly computing demanding process and several search queries are also typically needed to select good fitting structures.

Here, we present a novel and efficient Grid approach for performing these docking searches. This approach has been designed over the Gridway meta-scheduler. Results showing the high efficiency achieved are discussed together with the corresponding analysis of the performance obtained. The experiments were conducted on a Grid testbed built up from resources inside EGEE (LCG version of the pre-WS Globus components), the European production-level Grid infrastructure, and resources from a research testbed based on the Globus Toolkit 4 (Web Services components).

1. Introduction

Despite of the explosive growth of research in structural biology in last decades, the atomic resolution access to large macromolecular complexes implicated in the main cellular functions is still rather limited. Electron microscopy (EM) techniques are able to capture such large macromolecules in diverse near-physiological conditions [4]. However, the resolution that can be obtained with EM is constrained and we can only obtain 3D density maps of such large complexes at low-medium resolutions (10-20Å). By localizing available atomic resolution components into 3D EM low resolution maps is possible to obtain a detailed description of the structure of the entire macromolecular cellular machine. This localization, termed multi-resolution docking, can be reduced to geometrically register two 3D electron density maps: the experimental EM map with a simulated map obtained by lowering the resolution of the atomic structure of the component (for reviews see [18], [2]).

In practical terms, the multi-resolution docking process consists in estimating the 3D rotation matrix and the translational vector that maximizes the density overlap, i.e. maximizes a simple density correlation function (scalar product of the densities). To this end, a full 6D rigid-body search to explore all possible docking solutions must be performed. The exhaustive exploration is needed to avoid any missing valid registration. Note that we are confronting a non trivial problem and several docking alternative poses can be obtained due to the resolution differences, the EM low signal to noise ratio or small change between atomic and EM

*This research was supported by Ministerio de Ciencia y Tecnología through the research grants TIC2003-01321 and BPU2004-01282, and also by the Fundación BBVA. Authors from Universidad Complutense de Madrid participate in the EGEE project, founded by the European Union.

15th EUROMICRO International Conference on Parallel, Distributed and Network-Based Processing (PDP'07)
0-7695-2784-1/07 $20.00 © 2007 IEEE

structures (eg. missing regions, disorder or conformational changes).

Unfortunately, the required exhaustive exploration is highly computational demanding. Moreover, it can be even more demanding in practical situations where density maps to be aligned are of the order of few thousands. Therefore, use of both efficient algorithms and suitable computing platforms is essential to obtain a correct and fast solution.

Grids enable efficient and secure sharing of a large variety of computational resources scattered across several administrative domains [3]. Thus, offering a dramatic increase in the number of available processing and storing resources that can be delivered to applications. This new computational infrastructure provides a promising platform to execute loosely coupled, high-throughput computing applications, like the one described above. In general, these applications comprise the execution of a high number of tasks, each of which performs a given calculation over a subset of input values. However, in spite of the relatively simple structure of these applications, their efficient execution on computational Grids involves challenging issues [5], mainly due to the nature of the Grid itself, namely: dynamic resource availability and load, heterogeneity and a high fault rate.

In this work, we combine a novel rigid-body registration tool based on spherical harmonics, termed FRM (Fast Rotational Matching), with the computing potential provided by Grid infrastructures. We analyze the execution and the adaptation to the Grid of a multi-resolution docking application. In particular, we consider a highly heterogeneous Grid infrastructure, which comprises resources from the EGEE [1] (Enabling Grids for E-sciencE) production testbed and a research testbed (based on the Globus[2] Web Services components), and the GridWay meta-scheduler [6]. In this way, we will asses the suitability of this Grid environment to execute these large-scale bioinformatic applications.

The rest of this paper is organized as follows. In Section 2, we briefly describe the multi-resolution docking problem considered. The Grid environment used in this research is then introduced in Section 3, along with the GridWay meta-scheduler used to execute the application. In Section 4, we discuss the modifications introduced in the application to adapt its execution to the Grid. The experimental results obtained are then analyzed in Section 5. Finally, Section 6 presents a discussion of our results and hints of our future work.

2. Problem Description

As a benchmark test case, here we center our study in a concrete multi-resolution docking case. It is very frequent that the original atomic structures to be docked into the EM map are unknown. In this case, one can appeal to homology modelling bioinformatics tools which can give us an extensive set of possible atomic models. Homology modelling is based on the reasonable assumption that two proteins that have a good similarity in their sequence of amino acids will share very similar structures. Predictions of the structure of a target protein can be done finding one or more related proteins whose structure are known, aligning the target sequence to the sequences of the related proteins and building structure models based on the previous sequence alignments. The amount of related proteins and possible sequence aligning can be very wide, so many different models can be constructed. Also different homology model algorithms can be used increasing the number of possible docking candidates, but for simplicity here we only use those obtained by MODELLER [9].

In summary our computational challenging experiment will consist in performing an exhaustive docking search over a big set of homology models and then select those with higher density correlation. Combining the structures of the highest correlation models for all the subunits of the complex will lead to the atomic structure of the whole macromolecule imaged by electron microscopy (see Figure 1).

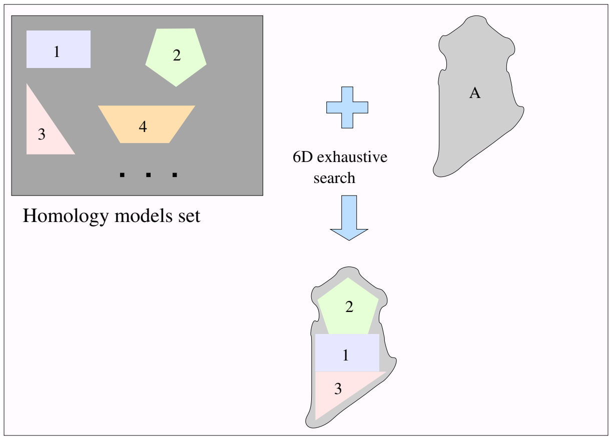

Figure 1. Schematic representation of a general multi-resolution docking problem. The localization of the atomic structures (numbered polygons) into a given low density map (curved shape) is reduced to independent 6D exhaustive searches of all possible relative rotations and translations. The fitting criterion is based on the density correlation, thus structures with high correlation value will correspond to the correct poses (e.g. shapes 1-3) where low correlation values will be rejected (e.g. shape 4).

[1] http://www.eu-egee.org
[2] www.globus.org

15th EUROMICRO International Conference on Parallel, Distributed and Network-Based Processing (PDP'07)
0-7695-2784-1/07 $20.00 © 2007 IEEE

3. Grid Infrastructure

In this section, we describe the Grid infrastructure used in this work. Table 1 shows the characteristics of the machines in the research testbed, build up from resources in the EGEE production testbed (using the LCG version of the pre-WS Globus components), and resources using the Web Services (WS) components of the Globus Toolkit. This organization results in a heterogeneous testbed, since it presents several processor speeds, Distributed Resource Management Systems (DRMS) and network links.

The whole infrastructure is composed by seven sites and 317 CPUs. In the experiments below, the number of jobs simultaneously submitted to the same resource was limited to four, to not saturate the testbed, so only 29 CPUs were used at the same time.

The execution of the application has been done through the GridWay[3] meta-scheduling system [6]. GridWay allows unattended, reliable, and efficient execution of jobs on heterogeneous and dynamic Grids; and performs all the job scheduling and submission steps transparently to the end user. Job execution is performed in three steps, namely: *prolog*, for creating the remote experiment directory and transferring the executable and input files; *wrapper* for executing the actual job and obtaining its exit code; and *epilog* for transferring back output files and cleaning tasks. The *prolog* and *epilog* phases are done by interacting with the Grid file transfer services (GridFTP), while the *wrapper* step interfaces the Grid execution services (GRAM).

The simultaneous use of two different testbeds based on different middlewares and components (execution, file transfer and information services), has been possible thanks to the decentralized, end-to-end and modular architecture of the meta-scheduling system [17]. GridWay uses different middleware access drivers to interface different Grid services. The architecture of GridWay and the interaction with the Grid services of the target infrastructures is shown in Figure 2. The experiments have been performed using the EGEE security infrastructure, and with a low priority certificate within a development virtual organization.

GridWay adapts job scheduling and job execution to changing Grid conditions by combining the following:

- Adaptive scheduling: to periodically adapt the schedule to the available resources and their dynamic characteristics. The GridWay scheduler considers the applications demands, in terms of requirements and preferences, and the dynamic characteristics of Grid resources, in terms of load, and availability [11].

- Adaptive execution: to migrate running applications to more suitable resources, improving application performance by adapting it to the dynamic availability, capacity and cost of Grid resources. Moreover, an application can migrate to a new resource to satisfy its new requirements or preferences [7].

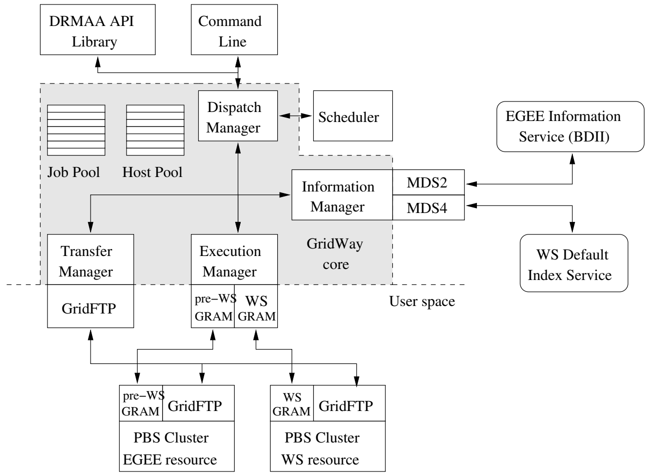

Figure 2. GridWay architecture, and interaction with Grid Services

GridWay also provides the application with fault tolerance capabilities by capturing GRAM callbacks, by periodically probing the GRAM job manager, and by inspecting the output of each job. GridWay implements the Distributed Resource Management Application API (DRMAA) [13] specification, which constitutes a homogeneous interface to different DRMS to handle job submission, monitoring and control, and retrieval of finished job status. In this sense the DRMAA standard represents a suitable and portable framework to express the kind of distributed computations, under study in this work.

4. Implementation of the Multi-docking Algorithm

The computational cost of the problem relays in two facts, namely:

- The exhaustive docking of an atomic model into a density map by a 6D (3 translational + 3 rotational) search is by itself a high computing demanding process.

- This docking operation must be applied over a large collection of different models or docking candidates (that could be extremely large if the subunits of the complex are numerous).

The combination of these two aspects can increase significantly the computing demand, making the docking pro-

[3] www.gridway.org

Table 1. Summary of the Grid resource characteristics († hosts with Globus 4, WS components).

Resource Name	Site	Processor	Speed (MHz)	Nodes	DRMS
ifaece01	PIC	Intel P4	2800	11	PBS
ce-egee	BIFI	Intel P4	3200	5	PBS
ce2	CESGA	Pentium III	512	46	PBS
lcg2ce	IFIC	AMD Athlon	1200	126	PBS
ramses	UPV	Intel PIII	866	26	PBS
lcg-ce	USC	Intel P4	2500	98	PBS
hydrus†	UCM	Intel P4	2500	4	PBS
ursa†	UCM	Intel P4	2500	1	fork

cess even unapproachable. Therefore, both aspects must be tackled in an efficient way. In the case of the docking algorithm, several methods have been developed to speed up the exhaustive search of compute correlations [18], [2]. If we use the standard multi-resolution docking tool COLORES [1] which accelerates the translational search by the use of the convolution theorem and fast Fourier transform, a single docking can take from many minutes to several hours. Here we employ a novel tool based on spherical harmonics, termed FRM (Fast Rotational Matching), detailed elsewhere [8]. Briefly, FRM accelerates the rotational search by expressing the density objects as spherical harmonics representations. This harmonic representation together with a convenient representation of the rotational group permits a fast computation of the rotational correlation function by the Fourier Transform. An optimized version of this method is able to reduce the docking time from minutes to few seconds.

With respect to the necessity of multiple executions, the Grid platform offers extensive computing resources for performing heavy computations. A multi-task adaptation of the problem over a Grid platform can speed up the application of the problem when a large amount of models to be docked in a map are provided. In the following, we discuss the way the multi-resolution docking problem is ported to a Grid environment.

4.1 Grid Application

We focused our work to the challenging docking case where multiple atomic resolution structures or models must be localized into a given target EM density map. In this case, each model can be independently docked and the searches can be performed in independent tasks. Thus, by using the Grid framework, all of the dockings (tasks) can run concurrently making use of different computing resources. After all the tasks have been fulfilled the outputs must be merged and sorted to bring out the best fitting models. Following this scheme the Grid version of our multi-docking tool was divided in the next three phases (see also Figure 3).

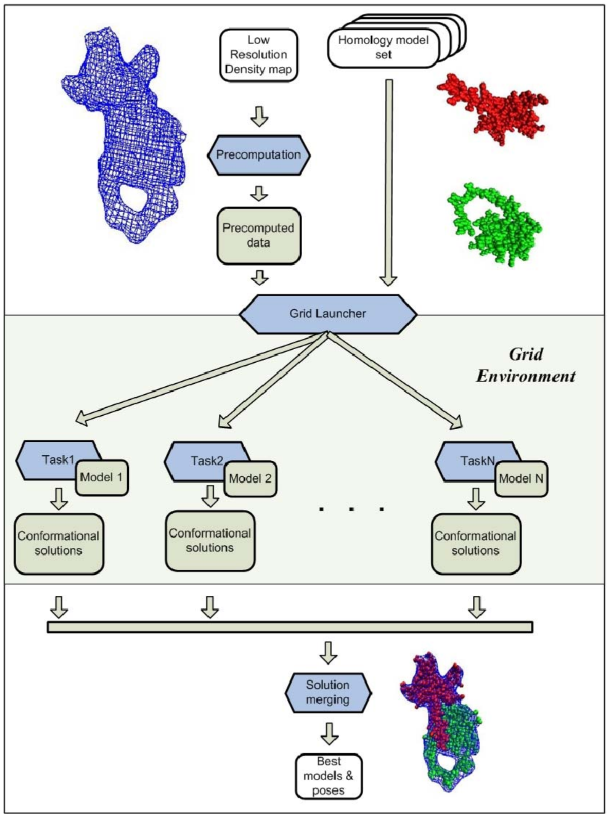

Figure 3. Scheme of the Grid implementation of the Fast Rotational Matching.

4.1.1 Pre-computation Phase

All the tasks perform several calculations that only depend on the same target EM density map. To save computing time these common calculations can be pre-computed. The pre-computations are related to:

- The translational search space limits. The shape and dimensions of the target density map constrain the possible positions that any atomic structure could occupy. Based on these geometric properties a mask of valid translational positions can be pre-established for all the 6D searches.

- FRM pre-computations. Since the EM map is always fixed, several calculations of the FRM algorithm can be also pre-computed.

These operations are locally performed, and generate all the pre-computed data from the density map.

4.1.2 Correlation Phase

Independent tasks are launched through the Grid environment. Each task performs the docking between the density map and a different assigned atomic model. To this end, a specific script is called which takes the binary docking tool (FRM) and two input files. These input files correspond to the pre-computation files and the atomic coordinates of the model to be docked. The final output of this phase will be a list of possible poses (position and rotations) sorted by higher correlation values.

4.1.3 Combination Phase

The best fittings of each atomic model are merged in a single file which is subsequently sorted. This task is performed by other script subroutine. In the file created solutions are sorted by the correlation value, so the first solutions will correspond to the best fitting models that can be used to correctly model the atomic structure located inside the EM map.

5 Result Analysis

The aim of the test case defined in Section 2 is to find an atomic structural model into macromolecule complex EM density map by fitting alternative comparative models of the map underlying structure. Here we show the results in a docking of 300 atomic homology models into a single simulated map of the protein rodent urinary (PDB entry 1mup). The resolution of this map was of 12Å and also Gaussian noise was added to proper simulation of experimental conditions. MODELLER was used to generate the alternative comparative models from distant homologs (<30% of sequence identity) of 1mup. These homology models were fitted by our FRM docking tool through the Grid environment. For validation purposes, we included in the data set the original protein structure used for generate the map. Consequently,this real structure is expected to be the model with the highest correlation value.

Table 2. Correlation values of the best fitting results.

Model	Normalized correlation
Model 0	0.9947
Model 6	0.9504
Model 3	0.9493
Model 291	0.9492
Model 288	0.9492
Model 298	0.9490
Model 21	0.9485
Model 241	0.9478
Model 263	0.9476
Model 260	0.9474

5.1 Validation and Efficiency of the Solution

In Table 2, the scoring correlation of the ten best fitting models is shown. As expected, the Model 0 which corresponds to the original structure of the target EM map is on the top of the list. The next model (model 6) corresponds to the best homology model obtained. As it can be seen in figure 4B, the structure of this model (light grey) fits very well into the EM map. This correspondence can be also observed by comparing the best model obtained with the original atomic structure (dark grey) used for generate the map (Figure 4C). The great resemblance of both structures validates the developed Grid base multi-resolution docking approach. In real world, the similarity of the best fitting comparative model found ensures a proper atomic resolution interpretation of the EM even if the original underlying atomic structure is not available.

5.2 Performance Analysis of the Grid Application

The overall execution time of the multi-resolution docking application is 8243 seconds (2h 17') with a peak dynamic throughput of 0.05 jobs per second. The dynamic throughput has been defined as the number of jobs com-

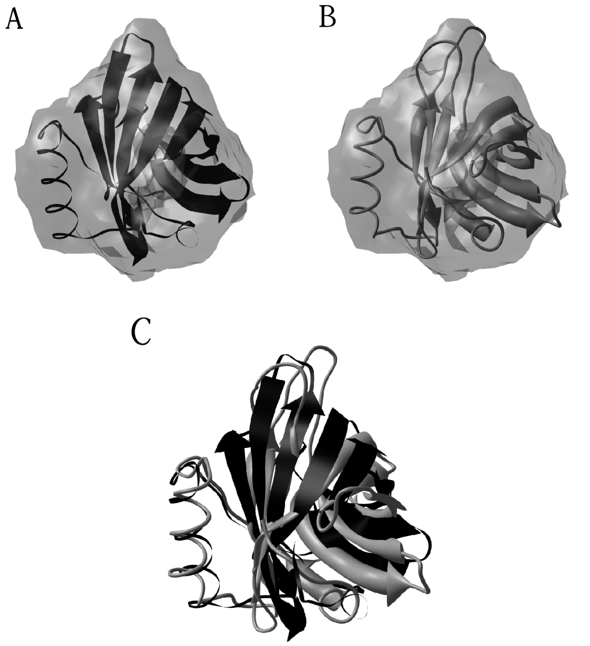

Figure 4. Docking results. Panel A) Original atomic structure and its corresponding target EM density map. Panel B) The best docking structure obtained is superposed into the target EM density map. This structure corresponds to model 6 of table 2. Panel C) Structural comparison between the best fitting model (light ribbons) and the original structure underlying the target EM map (dark ribbons). Note the high structural similarity.

pleted per second:

$$r(t) = \frac{N(t)}{t} \quad (1)$$

Compared to the single host execution on the fastest machine in the testbed (hydrus, 0.01 jobs per second, without considering file transfer, middleware overhead and queue wait times), these results represent 51% reduction in the overall execution time. This is, if all the docking process had been performed in the fastest cluster the full exploration of all the models should have been completed approximately in 16950 seconds (4h 42').

The dynamic throughput (Eq. 1) obtained during the execution of the application is shown in Figure 5. The maximum rate of performance in jobs executed per second is obtained from 150 jobs. Also to obtain half of this performance 12 jobs have to be executed. This parameters are useful to evaluate the performance gain that can be expected when executing the multi-resolution docking application in the Grid [10].

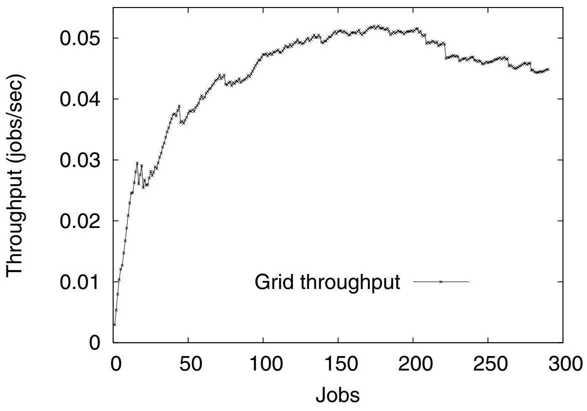

Figure 5. Throughput (jobs per second), in the execution of the multi-resolution docking application.

Figure 6 presents the average job turnaround, execution, file transfer and queue wait times on each host of the testbed; error bars represent the standard deviation of these measurements. These times include the overhead induced by the Globus middleware. The standard deviation of the total time is greater for the measurements performed on the EGEE clusters. This fact is mainly because of the variability in the queue wait time in the PBS system, as these resources are at production level and being used by other experiments. File transfer times between the client (at the UCM site) is very uniform with a low variability, being lower with those resources in the UCM site (hydrus and ursa).

Note also the high heterogeneity in the execution time because of the differences in the computational power of the Grid resources. The execution time presents a moderate variability as the computational cost of each multi-resolution docking problem is not uniform. These results justify the moderate performance gain mentioned above.

Let us now consider the overhead to execution ratio on each resource of the Grid:

$$s = \frac{T_f + Tq}{T_x} \quad (2)$$

where T_x is the execution time, T_f is the file transfer time, and Tq is the queue wait time in the local DRMS. This ratio ranges from $s = 0.09$ in ursa to $s = 0.52$ in ramses, which reflects a right distribution of the application in these hosts. However, lcg2ce ($s = 0.90$) and lcg-ce ($s = 1.41$) exhibit a unfavourable ratio (i.e. the overhead is equal or greater than the execution time). In these situations the application could benefit from using a coarser grain distribution (i.e.

0-7695-2784-1/07 $20.00 © 2007 IEEE

performing more than one multi-resolution docking problem per job).

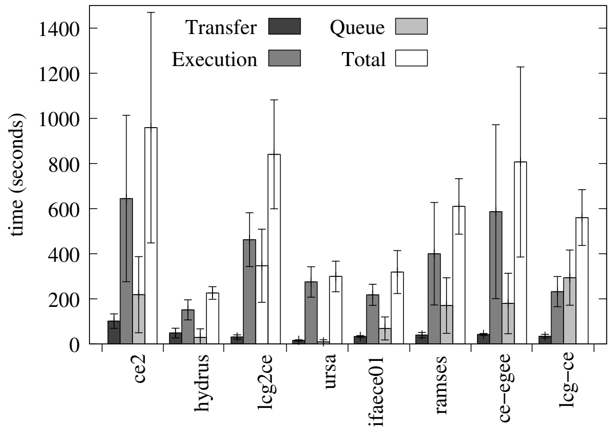

Figure 6. Average and standard deviation in execution, file transfer and queue wait times on each resource of the testbed.

We will next evaluate the schedule performed in the above experiments. Figure 7 shows the number of jobs successfully executed, those that were migrated, and failed executions on each resource in the testbed. When an execution fails, it is re-scheduled and executed on other host. In our case, a job migration only occurs when the job has been waiting in the queue system of the remote host more than a given threshold (15 minutes). A highly fault rate is observed in several resources in the Grid, as outlined in Section 3. These failures are mainly due to a known Globus problem in the LCG middleware. So, these results suppose an overall failure rate of 29%, and a migration rate of 6%, making the fault tolerance and adaptive execution capabilities of the meta-scheduling system necessary to successfully complete the experiment. Finally, as expected those resources with more and faster nodes contribute in higher degree to the problem resolution.

6 Conclusions and Future Work

The proposed Grid based solution for multiple docking of atomic models in a density map has been proved to be efficient. Using a Grid environment permitted one-half reduction in the searching time spent in the determination of the best docking models respect to a single-task implementation. Considering the small size of the Grid and the low priority used in this preliminary benchmark test, this is a very promising result. In fact, the behavior of the Grid environment employed is not optimal. Remarkable long file transfer and queue wait times on some resources of the testbed slow down the global performance together with the existence of hosts with high fault rates.

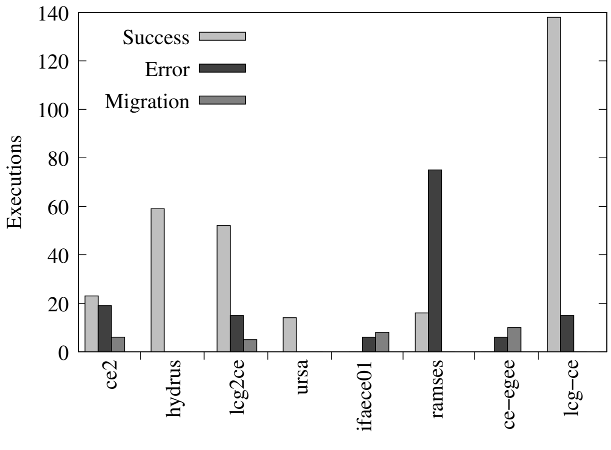

Figure 7. Number of jobs scheduled on each resource.

The present approach will be improved by using a dynamic grain scheduler to minimize the overhead to execution ratio on a given resource. Moreover, in the benchmark studied, models represent the full complex, so the input files are large and the docking search is quite limited because of the short translational possible positions of the model inside the density map. A higher productivity is also expected when models of small sub-units of the complex will be used.

In summary, Grid computing has been proved to be a helpful platform for confronting multi-resolution docking. Besides, this Grid application can be easily adapted to the wide set of existing problems where a 3D matching is needed. Note that this problem is equivalent to a general rigid body 3D registration problem, and it can be found in a diverse range of fields such as structural Biology [14], [15] or image processing [12], [16]. We are particularly interested in extending the application range of this implementation to protein-protein or protein-ligand problems. In these cases, the problem consists in predicting how two proteins (or a protein and a ligand) can interact by using a similar Grid docking search which tests all 3D rearrangements between them.

References

[1] P. Chacon and W. Wriggers. Multi-resolution contour-based fitting of macromolecular structures. *J Mol Biol*, 317:375–384, 2002.

[2] F. Fabiola and M. Chapman. Fitting of high-resolution structures into electron microscopy reconstruction images. *Structure (Camb)*, 13:389–400, 2005.

[3] I. Foster. What Is the Grid? A Three Point Checklist. *GRIDtoday*, 1(6), 2002.

[4] J. Frank. *Three-Dimensional Electron Microscopy of Macromolecular Assemblies*. Academic Press, San Diego, EEUU, 1996.

[5] E. Huedo, R. S. Montero, and I. M. Llorente. Experiences on Adaptive Grid Scheduling of Parameter Sweep Applications. In *Proc. 12th Euromicro Conf. Parallel, Distributed and Network-based Processing (PDP2004)*, pages 28–33. IEEE CS, 2004.

[6] E. Huedo, R. S. Montero, and I. M. Llorente. A Framework for Adaptive Execution on Grids. *Software – Practice and Experience (SPE)*, 34(7):631–651, 2004.

[7] E. Huedo, R. S. Montero, and I. M. Llorente. The Gridway Framework for Adaptive Scheduling and Execution on Grids. *Scalable Computing: Practice and Experience Journal*, 6(3):1–8, 2005.

[8] J. Kovacs, P. Chacon, Y. Cong, E. Metwally, and W. Wriggers. Fast rotational matching of rigid bodies by fast fourier transform acceleration of five degrees of freedom. *Acta Crystallogr D Biol Crystallogr*, 59:1371–1376, 2003.

[9] M. Marti-Renom, A. Stuart, A. Fiser, R. Sanchez, F. Melo, and A. Sali. Comparative protein structure modeling of genes and genomes. *Annu. Rev. Biophys. Biomol. Structl*, 29:291–325, 2000.

[10] R. S. Montero, E. Huedo, and I. M. Llorente. A Performance Model for High Throughput Computing on Grids. *Parallel Computing (to appear)*.

[11] R. S. Montero, E. Huedo, and I. M. Llorente. Grid Resource Selection for Opportunistic Job Migration. In *Proc. 9th Int'l Conf. Parallel and Distributed Computing (Euro-Par 2003)*, volume 2790 of *LNCS*, pages 366–373. Springer-Verlag, August 2003.

[12] G. Papaioannou, E. Karabassi, and T. Theoharis. Reconstruction of three-dimensional objects through matching of their parts. *IEEE Transactions on Pattern Analysis and Machine In-telligence*, 24:114–124, 2002.

[13] H. Rajic, R. Brobst, W. Chan, F. Ferstl, J. Gardiner, J. P. Robarts, A. Haas, B. Nitzberg, and J. Tollefsrud. Distributed Resource Management Application API Specification 1.0. Technical report, DRMAA Working Group – The Global Grid Forum, 2003.

[14] M. Rossmann and E. Arnold. *Crystallography of Bio-logical Macromolecules.* Kluwer Academic Publishers, Dordrecht, The Netherlands, 2001.

[15] G. Smith and M. Sternberg. Prediction of protein-protein interactions by docking methods. *Curr Opin Struct Biol*, 12:28–35, 2002.

[16] E. Trucco and A. Verri. *Introductory Techniques for 3-D Computer Vision*. Prentice-Hall, New Jersey, 1998.

[17] J. Vazquez, E. Huedo, R. S. Montero, and I. Llorente. Coordinated Harnessing of the IRISGrid and EGEE Testbeds with GridWay. *Journal of Parallel and Distributed Computing*, 66(5):763–771, 2006.

[18] W. Wriggers and P. Chacon. Modeling tricks and fitting techniques for multiresolution structures. *Structure (Camb)*, 9:779–788, 2001.