Grid enabled high throughput virtual screening against four different targets implicated in malaria

J. Salzemann, V. Kasam, N. Jacq, A. Maass, H. Schwichtenberg, Vincent Breton

Grid Enabled High Throughput Virtual Screening Against Four Different Targets Implicated in Malaria

Jean SALZEMANN [a], Vinod KASAM [a,b,1], Nicolas JACQ [a], Astrid MAASS [b]
Horst SCHWICHTENBERG [b], Vincent BRETON [a]
[a] *Laboratoire de Physique Corpusculaire, 63177 Aubière cedex, France.*
[b] *Fraunhofer Institut for Algorithms and Scientific Computing, Germany.*

Abstract. After having deployed a first data challenge on malaria and a second one on avian flu, respectively in summer 2005 and spring 2006, we are demonstrating here again how efficiently the computational grids can be used to produce massive docking data at a high-throughput. During more than 2 months and a half, we have achieved at least 140 million dockings, representing an average throughput of almost 80,000 dockings per hour. This was made possible by the availability of thousands of CPUs through different infrastructures worldwide. Through the acquired experience, the WISDOM production environment is evolving to enable an easy and fault-tolerant deployment of biological tools; in this case it is the FlexX commercial docking software which is used to dock the whole ZINC database against 4 different targets.

Keywords: large scale deployment, Computational grids, Malaria, In silico docking, Virtual Screening, WISDOM.

Introduction

WISDOM stands for World-wide In Silico Docking On Malaria. Malaria together with many other tropical and protozoan diseases is one of the most neglected diseases by the developed countries as well as by the pharmaceutical industries. Plasmodium is the protozoan genus causing malaria. Due to very high costs associated to the drug discovery process as well as due to late stage attrition rates, novel and cost effective strategies are absolutely needed for combating the neglected diseases, especially malaria [1].

In silico screening of chemical compounds against a particular target is termed as Virtual Screening. The costs associated to the virtual screening of chemical compounds are significantly reduced when compared to screening of compounds in experimental laboratory. Beside the costs, virtual screening is fast and reliable [2, 3]. However, it is computationally intensive: docking a single compound within the active site of a given receptor requires about 1 minute CPU. With the development of combinatorial

[1] Corresponding author: Vinod Kasam: Laboratoire de Physique Corpusculaire, Campus des Cézeaux - 24 av des Landais, 63177 Aubière cedex, France; E-mail: kasam@clermont.in2p3.fr.

chemistry technology, millions of different chemical compounds are now available in digital databases [4]. To screen all these compounds and store the results is a real data challenge. To address this problem computational grid infrastructures are used.

WISDOM-I [5] is the first large scale deployment of molecular docking application on EGEE grid infrastructure. It took place from August 2005 to September 2005 and achieved 41 million dockings which is equivalent to 80 CPU years. The docking was performed on Plasmepsins, a aspartic protease involved in haemoglobin degradation. On the biological front three scaffolds were identified, of them one is guanidino scaffold which is likely to be novel as it was not known as a plasmepsin inhibitor before [6].

With the success achieved by the WISDOM-I project both on the computation and biological sides, several scientific groups around the world proposed targets implicated in malaria, which led to the second assault on malaria, WISDOM-II.

1. Materials and methods

Virtual Screening by molecular docking requires a target structure, a chemical compound database and docking software. The targets used in the current project are and Glutatione–S-trasferase (GST, pdbid: 1Q4J) [7], Plasmodium falciparum Dihydrofolate reductase (DHFR) wild type (pdbid: 1J3I), quadrupule mutant (pdbid: 1J3K) [8], Plasmodium vivax Dihydrofolate reductase wild type (pdbid: 2BL9), double mutant (pdbid: 2BLC) [9]. In another experiment the same structures of Plasmodium vivax Dihydrofolate reductase wild type (pdbid: 2BL9), double mutant (pdbid: 2BLC) but after minimization are used. The chemical compound database used is ZINC database [10, 11] and the docking software used is FlexX. To store the results of docking, MySQL databases are used, but it is still in process. FlexX [12, 13] is an extremely fast, robust and highly configurable computer program for predicting protein-ligand interactions. During our experiment, after several control tests, standard parameter settings are used except for two cases: "Place particles" and "Maximum overlap volume".

2. Procedure

The goal of WISDOM II is two fold, the biological goal is to find the best hits against the targets implicated in malaria and the computational goal is to keep improving the relevance of computational grids in drug discovery applications. Here in this paper we are going to discuss in details the grid architecture and deployment.

2.1. Virtual screening experimental setup

The complete virtual screening experiment is segmented into five different phases.

i. Target preparation
ii. Compound database
iii. Validation of the docking experiment
iv. Screening
v. Result analysis

2.1.1. Target preparation

A standard protocol is used while preparing the target structures. The initial coordinates for all the target structures are obtained from Brookhaven protein database (www.pdb.org). Depending upon the inclusion of the significant residues, cofactors and the binding pocket, active site is defined as 8.0 Å - 10.0 Å around the co-crystallized ligand.

2.1.2. Compound database

The Compound library used for WISDOM was obtained from the ZINC database [14, 15]. The ZINC database is a collection of 4.3 million chemical compounds ready for virtual screening from different vendors. We have chosen to use the ZINC library because ZINC is an open source database and the structures have already been filtered according to the Lipinski rules. Moreover the data are available in different file formats (Sybyl mol2 format, sdf and smiles). A total of 4.3 million compounds were downloaded from the ZINC database and screened against four targets.

2.1.3. Validation of the docking experiment

Re-docking against the co-crystallized compound is performed to check and tune the docking experiment requirements. Re-docking serves as a control for finally selecting the parameters for target structure, before subjecting it to large scale docking. Docking pose is validated at two levels; RMSD value (the lower, the better) and binding pose of ligand (the more similar the docking pose to the co-crystallized ligand, the better).

2.2. Grid infrastructure and Deployment

2.2.1. Grid Infrastructures

The deployments were achieved on several grid infrastructures: Auvergrid [14], EELA [15], EGEE [16], EUChinaGrid [17] and EUMedGrid [18]. All these infrastructures are actually using the same middleware, gLite. EGEE is the main infrastructure offering the largest resources; they are all interconnected with EGEE, in the sense that all of these Grids share some resources with EGEE. In the case of Auvergrid, it is even more evident as all the resources available through the Auvergrid Virtual Organization (VO) are also shared with several EGEE VOs. The EUChinaGrid project for instance made available all the grid sites belonging to its infrastructure; seven Computing Elements in total and two Storage Elements were used to store the databases and result files on the EuChinaGrid.

2.2.2. WISDOM production Environment

WISDOM environment has been used two times in previous large-scale experiments, WISDOM-I in the summer 2005 [19] and a second deployment against avian flu in the spring 2006 [20]. WISDOM environment keeps evolving in order to make it more user friendly and easier to use by non grid expert. The main objective was also to improve the fault-tolerance of the system, in implementing, for instance, a persistent environment, that can be stopped and restarted at any time without risk of loosing significant information, which proved to be also very useful as it enables the whole maintenance of the scripts and code and improve the interactivity with the user, as the user can also manage jobs finely, for instance force the cancellation and resubmission of a scheduled job. Along with this, we tried to minimize the cost of the environment in terms of disk space and CPU consumption for the user interface. Most of the job files are now generated dynamically: this allows the user as well to modify on the fly the configuration of the resource brokers and the jobs requirements. This way, the user is sure that the next submissions will take these modifications into account. The Figure 1, shows the overall architecture of the environment.

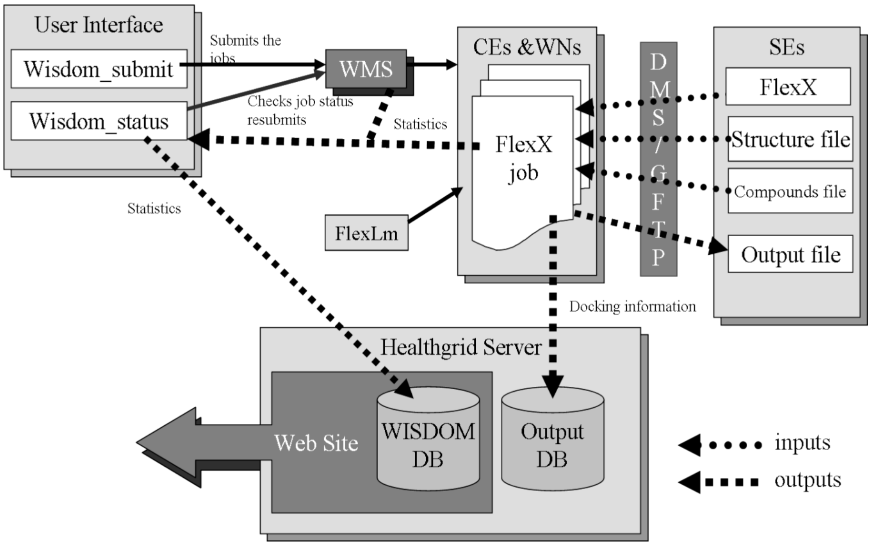

Figure 1. Schema of the WISDOM production environment.

The user is interacting with the system through the two main scripts (widom_submit and wisdom_status) deployed on the user interface. These scripts will take care of job files generation, submission, status follow-up and eventually resubmission automatically. The jobs are submitted directly to the grid Workload Management System (WMS), and are executed on the grid computing elements and worker nodes (CEs and WNs). As soon as it is running, a job transfers all the files stored on the Storage Elements (SEs) via the Data Management System of the grid (DMS) with the gridFTP protocol. Once the job is finished, the outputs are stored back on the grid Storage Elements via the Data Management System and the useful docking results are inserted directly from the grid to a relational database where they can later be more easily queried and analyzed.

2.2.3. Data Challenge Deployment

The deployment was performed on the infrastructures listed under section grid infrastructures, and involved at least one manager to oversee the process on each of them. The three groups of targets (GST, Plasmodium vivax and falciparum DHFR) were docked against the whole ZINC database (4.3 millions of compounds). The database was actually cut into 2,422 chunks of 1,800 compounds each. This splitting was chosen because we wanted to have an approximated processing time ranging from 20 to 30 hours for each job (one docking process takes from 40s to 1min depending on the CPU power). The subsets were stored on the grid and replicated on several locations whenever possible to improve fault-tolerance. We define a WISDOM instance as one target structure docked against the whole ZINC database, with a given parameter set.

A total number of 32 instances were deployed, corresponding to an overall workload of 77,504 jobs, and up to 140,000,000 docking operations. On these total 32 instances, 29 instances ran on EGEE, and 3 were on Auvergrid, EELA and EuChinaGrid.

As shown in Figure 1, the environment included a FlexLm server that was providing the floating licenses for the FlexX commercial software. The FlexX software was already used during the first WISDOM deployment in 2005, and the license server was identified as a potential bottleneck and point of failure because we had just one server available at this time. For WISDOM-II, up to 3 servers were made available at the SCAI Fraunhofer institute (http://www.scai.fraunhofer.de), with 3,000 licenses available on each server.

As the average duration of a job was around 30 hours, we submitted 1 instance per day, with a delay of 30 seconds between each submission. As one instance was submitted in about 20 hours, the submission process was quite continuous during the first month of deployment. The jobs were submitted to 15 Resource Brokers (the components of the Workload Management System) in a round-robin order. At the end of a job, the results were stored on the grid Storage Elements and directly into a relational database. The job repartition was quite similar to the previous deployments, but here the UKI federation played an even bigger part. For instance, one of the British sites offered for quite a long period of time more than 1,000 free CPUs, which is half of the average used CPUs. Auvergrid, EELA, Euchinagrid and Eumedgrid contributed by running each 3% of the jobs.

3. Results and Statistics

Table 1, shows the overall statistics of the deployment. The number of jobs here are the number of awaited results, but far more jobs were actually submitted on the grid. When a job was done on the grid, the environment checked a status file specifying the final result of the job: a job can be done in the point of view of the worker node, without having produced the result files, in this specific case, the status of the job, which was stored on the grid as well, was labeled as failed, and the environment had to resubmit the job. In some cases, the environment failed at retrieving the status from the grid, and thus considered implicitly the job has failed, even if the job has succeeded. It explains why some jobs ran several times, and why the final completed docking number is bigger than the useful awaited dockings. Anyhow the average docking throughput is coherent with the crunching factor. The crunching factor is the ratio of the total CPU time over the duration of the experiment. It represents the average number of CPUs used simultaneously all along the data challenge and is a metric of the parallelization gain. If we consider 80,000 dockings per hour for 2,000, it means 40 dockings for one CPU per hour, which is coherent with the empiric observation of one docking process lasting approximately 1 minute on a 3.06 Intel Xeon processor.

Table 1. Overall statistics concerning the deployment.

Number of Jobs	77,504
Total Number of completed dockings	156,407,400
Estimated duration on 1 CPU	413 years
Duration of the experiment	76 days
Average throughput	78,400 dockings/hour
Maximum number of loaded licences (concurrent running jobs)	5,000
Number of used computing elements	98
Average duration of a job	41 hours
Average crunching factor	1,986
Volume of output results	1,738 TB
Estimated distribution efficiency	39%
Estimated grid success rate	49%
Estimated success rate after output checking	37%

In the Table 1, the estimated grid success rate is the ratio of successful grid jobs on the total of submitted jobs. The success rate after output checking will consider just the jobs that succeeded in producing the result files, that’s why this score is lower. One can notice that these values are very small, but there are several explanations for this. At the beginning of the data challenge, the observed grid success rate was about 80 to 90%, but it decreased constantly because of sites overload. Sometimes the available disk space was decreasing on some Resources Brokers, up to a point where some of the job data could not reach the Computing Element. In other cases, the sites were simply producing a lot of aborted job for an undetermined reason. The Resource Brokers failed again to balance reasonably the jobs on the Computing Elements, and some of them ended up with more than 500 jobs in queue, the site administrator had no other choice than kill all these jobs, producing in a single row more than 500 aborted jobs. Actually, because of the automatic resubmission, this information should not be taken as an overall significant way to evaluate the efficiency of the grid, because the automatic resubmission guaranteed a successful job, and the aborted jobs are not staying a long time on the grid consuming useful resources. The grid is a very dynamic system, and errors can occur at the last minute.

4. Observations and Issues

The scheduling efficiency of the grid is still a major issue. The resource broker is still the main bottleneck, and even if used in high number (>15), is always a source of trouble. Moreover things get worse as load is increasing on the grid. The « sink-hole » effects can result in sites overloading in a very short amount of time, and if not taken care quickly can lead to an impressive overhead caused by the long lasting waiting state of the jobs. Added to that the sometimes unreliable and incomplete information provided by the information system, which does not publish the available slots and VO limitations that would be mandatory to perform an efficient scheduling.

Another issue was that to be able to store and treat the data in a relational database, the machine hosting the database must have good performances or the number of queries coming from the grid may also overload the machine significantly. In this deployment we used a MySQL database and planned to put all the produced result in the same table, but finally we had to spit this database in several ones (one per target), because MySQL would not have been able to withstand the total number of records, It was generating CPU overloads on the machine, which lead to serious slowdowns.

All these elements demonstrate clearly that even if the grid can show very good result in comparison to very simple architecture it is still missing robustness and reliability, and can indeed be improved performance-wise.

5. Conclusions

We have demonstrated the role and significance of computational grids in the life science applications like structure based drug design. Large scale virtual screening on four different targets of malaria was performed in search for potential hits on several grid infrastructures: Auvergrid, EELA, EGEE, EUChinaGrid and EUMedGrid. One of our goals was to further demonstrate the impact of computational grids in life science applications like virtual screening where large amounts of computing power is required; we have achieved it by successfully screening the whole ZINC database for three malaria targets in 76 days instead of 413 CPU years. We have reached during this ten-week period an average docking throughput of 78,400 dockings. MySQL databases are used for the analysis of the docking results, which will ease the final analysis of the virtual screening data. On the biological front 1,738 TB of valuable data has been produced. Analysis of the results (identification of the best hits) is under way: the best hits will be post processed by molecular dynamic simulations and tested in the experimental laboratories.

Acknowledgements

Information on WISDOM collaboration can be found at http://wisdom.healthgrid.org.

The authors acknowledge the contribution of Emmanuel Medernach (AuverGrid), Ignacio Blanquer (EELA), Gabri Aparico (EELA), Enrico Fattibene (EUChinaGrid), Kostas Koumantaros (EUMedGrid) and Domenico Vicinanza (EUMedGrid).

The Enabling Grids for E-sciencE (EGEE) project is co-funded by the European Commission under contract INFSO-RI-031688. Auvergrid is a project funded by the

Conseil Regional d'Auvergne. The BioinfoGRID project is co-funded by the European Commission under contract INFSO-RI-026808. The EMBRACE project is funded by the European Commission within its FP6 Programme, under the thematic area "Life sciences, genomics and biotechnology for health", contract number LHSG-CT-2004-512092.

We gratefully acknowledge the support of BioSolveIT who provided more than 6,000 free licenses for their commercial docking program FlexX.

References

[1] J. G. Breman, M. S. Alilio, A. Mills. Conquering the intolerable burden of malaria: what's new, what's needed: a summary. Am. J. Trop. Med. Hyg. 71S (2004), 1-15.

[2] K. H. Bleicher, H. J. Boehm, K. Mueller, A. I. Alanine. Hit And Lead Generation: Beyond High-Throughput Screening. Nat. Rev. Drug. Discov. , 2 (2003), 369-378.

[3] H. J. Boehm, G. Schneider. Virtual Screening For Bioactive Molecules. Chapter 1, High Throughput Screening and Virtual Screening Entry Points To Drug Discovery. Methods and Principles in Medicinal Chemistry, Volume 10.

[4] R.W. Spencer, High throughput virtual screening of historic collections on the file size, biological targets, and file diversity, Biotechnol. Bioeng 61 (1998), 61-67.

[5] N. Jacq, J. Salzemann, Y. Legré, M. Reichstadt, F. Jacq, E. Medernach, M. Zimmermann, A. Maaß, M. Sridhar, K. Vinod-Kusam, J. Montagnat, H. Schwichtenberg, M. Hofmann, V. Breton, Grid enabled virtual screening against malaria, accepted for publication in Journal of Grid Computing (2007)

[6] V. Kasam, M. Zimmermann, A. Maaß, H. Schwichtenberg, A. Wolf, N. Jacq,V. Breton, M. Hofmann. Design of Plasmepsin Inhibitors: A Virtual High Throughput Screening Approach on the EGEE Grid, submitted to Journal of Chemical Information and Modeling (2006).

[7] M. Perbandt, C. Burmeister, R.D. Walter, C. Betzel, E. Liebau, Native and inhibited structure of a Mu class-related glutathione S-transferase from Plasmodium falciparum, J. Biol. Chem. 279 (2004), 1336-1342

[8] P. Kongsaeree, P. Khongsuk, U. Leartsakulpanich, P. Chitnumsub, B. Tarnchompoo, M.D. Walkinshaw, Y. Yuthavong. Crystal Structure of Dihydrofolate Reductase from Plasmodium Vivax: Pyrimethamine Displacement Linked with Mutation-Induced Resistance, Proc. Natl. Acad. Sci. USA. 2 (2005), 13046-13051.

[9] J. Yuvaniyama, P. Chitnumsub, S. Kamchonwongpaisan, J. Vanichtanankul, W. Sirawaraporn, P. Taylor, M.D. Walkinshaw, Y. Yuthavong. Insights into antifolate resistance from malarial DHFR-TS structures, Nat. Struct. Biol. 10 (2003), 357-365.

[10] ZINC database: J. J. Irwin, B. K. Shoichet. ZINC - A Free Database of Commercially Available Compounds for Virtual Screening. J. Chem. Inf. Model. 45 (2005), 177-182.

[11] A Free database for virtual screening. ZINC is not commercial. http://blaster.docking.org/zinc/ UCSF, University of California, San Francisco.

[12] M. Rarey, B. Kramer, T. Lengauer, G. Klebe, Predicting Receptor-Ligand interactions by an incremental construction algorithm, J. Mol. Biol. 261 (1996), 470-489.

[13] BioSolveIT Homepage: http://www.biosolveit.de

[14] Auvergrid, available at www.auvergrid.fr

[15] EELA, available at www.eu-eela.org

[16] F. Gagliardi, B. Jones, F. Grey, M.E. Begin, M. Heikkurinenn. Building an infrastructure for scientific Grid computing: status and goals of the EGEE project, Philosophical Transactions: Mathematical, Physical and Engineering Sciences, 363 1729-1742 (2005) and http://www.eu-egee.org/

[17] EuChinaGrid, available at www.euchinagrid.org

[18] EuMedGrid, available at www.eumedgrid.org

[19] N. Jacq, V. Breton, H-Y. Chen,L-Y. Ho, M. Hofmann, H-C. Lee, Y. Legré, S. C. Lin, A. Maaß, E. Medernach, I. Merelli, L. Milanesi, G. Rastelli, M. Reichstadt, J. Salzemann, H. Schwichtenberg, M. Sridhar, V. Kasam, Y-T. Wu, M. Zimmermann. Virtual Screening on Large Scale Grids, accepted for publication in Parallel Computing, (2007).

[20] H.-C. Lee, J. Salzemann, N. Jacq, H.-Y. Chen, L.-Y. Ho, I. Merelli, L. Milanesi, V. Breton, S. C. Lin, Y.-T. W. Grid-Enabled High-Throughput In Silico Screening Against Influenza A Neuraminidase, IEEE transactions on nanobioscience, (2006) 288-295.