Bringing Flexibility to Virtual Screening for Enzymatic Inhibitors on the Grid

Marshall J. Levesque[2], Kohei Ichikawa[3], Susumu Date[1], Jason H. Haga[2]
[1] *Cybermedia Center, Osaka University, Osaka, Japan*
[2] *Department of Bioengineering, University of California, San Diego, La Jolla, CA, USA*
[3] *The Research Center of Socionetwork Strategies, Kansai University, Osaka, Japan*
mlevesqu@ucsd.edu, ichikawa@rcss.kansai-u.ac.jp, date@ais.cmc.osaka-u.ac.jp,
jhaga@bioeng.ucsd.edu

Abstract

Virtual screening has become an important part of the drug discovery process. Grid computing facilitates this process by providing shared computational resources across different international institutions to run computationally intensive, scientific applications without the need for a centralized supercomputer. This study designed and implemented a flexible, scalable platform to perform a large virtual screening experiment on the PRAGMA grid testbed using the molecular docking simulation software DOCK 5.4. Using Opal OP to wrap DOCK as a grid service and PERL for data manipulation purposes, the "druglike" subset of the ZINC database, which contains 2,066,906 compounds, was successfully screened against the catalytic site of a protein tyrosine phosphatase. The screening required 11.56 days laboratory time and utilized 200 processors over 7 clusters. A ranked list of the best binding compounds to the phosphatase was generated and is currently being tested in biological applications for their efficacy and specificity.

1. Introduction

The drug discovery process has accelerated with the growth and availability of databases containing compounds produced by combinatorial chemistry, along with the continual identification of new molecules as potential drug targets. Robotics and other automated experimental procedure machinery have enabled the physical screening of entire libraries, containing thousands to millions of compounds, against a biological target. This process of high-throughput screening (HTS) has become a staple method in the pharmaceutical industry. However, the high cost and low hit-rates associated with HTS experiments have been motives for developing more knowledge-based search methods [1]. One such method is virtual screening, which simulates the interactions between each compound and the target molecule *in-silico*. Binding affinity scores for all compound-target pairs are calculated according to physical and chemical properties. The scores are then ranked to create a list of potential hits.

By initially performing virtual screening experiments and selecting a group of best scoring compounds, the required number of possible inhibitory compounds to be tested in the lab is significantly reduced compared to HTS methods. The insight gained from the *in-silico* experiments greatly improves hit-rate, which is usually defined as the proportion of screened compounds that inhibit enzymatic activity by fifty percent at a concentration of $\leq 100\mu$M. In a comparison between HTS and virtual screening methods to identify an inhibitor for protein tyrosine phosphatase-1B (PTP-1B), virtual screening produced a hit-rate 1,700-fold greater than that from HTS experiments [2]. Reducing the number of compounds to be screened becomes increasingly valuable to academic laboratories, where the requirements of empirical, large-scale HTS experiments are often too great of an investment in equipment, maintenance, and time to perform internally [3].

A major challenge in virtual screening is increasing the accuracy of predicting binding affinities between compound-target pairs. This has been addressed with flexible docking, where a number of possible conformations of the compound are oriented into the binding site of the target molecule [4], and calculating the change in solvation energies for the molecular pairing [3]. Both of these strategies require a significant increase in computational cost. Thus, in order to preserve accuracy while performing a search with an extensive compound database, the availability

of computational resources is an important consideration.

Grid computing has been shown to be a well-suited environment for implementing molecular docking software to perform virtual screening experiments in a number of studies [5-7]. The distributed nature of the Grid allows performing the virtual screening process as a large number of individual computational experiments in parallel, with performance increasing linearly with the number of CPUs in use [8]. This type of environment also displaces any need for dedicated computational resources that are most likely too expensive in both initial costs and maintenance for the majority of biomedical laboratories.

The success of Grid computing as a tool for the life and physical sciences can be attributed to the on-going development of Grid middleware, such as the Globus Toolkit [9]. A push towards integrating Web services into Grid middleware for a better cross platform experience resulted in the introduction of the Open Grid Services Architecture (OGSA) [10]. This led to the creation of Web service-based tools that further aid scientists in bringing their applications to the Grid environment. These technologies were utilized in this project to implement a flexible, scalable Grid service-based virtual screening platform.

2. Objectives

2.1. Biological objective

This virtual screening experiment is to identify novel inhibitory compounds for protein tyrosine phosphatase molecules that can be used to manipulate cellular signaling pathways. Achieving maximums in performance and efficiency is not the highest priority in this virtual screening. The wet-bench experiments used to verify the efficacy of the inhibitory compounds obtained from the virtual screening can typically take several weeks. Therefore, a variance of hours or days that may be incurred due to changes in Grid workload or resource availability during the screening is not a major point of concern. Access to adequate computational resources for performing screenings on a "reasonable" time scale is a more important issue and is the focus of this project.

2.2. IT objective

The aim is to provide a platform that allows biomedical scientists to screen extensive chemical databases against their target molecules using computationally expensive, accurate docking methods within a practical timeframe. Key features include flexibility, scalability, reusability, customization, automated job distribution, progress monitoring, and collection of results.

3. Technologies

3.1. Docking programs

Many studies have been performed comparing the accuracy and required computation time amongst a number of available docking algorithms such as DOCK, AutoDock, Flexx [1, 11, 12]. No single docking program, however, has proven to be the preferred choice from the long list of virtual screening tools. Instead, taking a consensus between results from a number of different docking algorithms has been shown to be a viable method to reduce the number of false-positives encountered when using only a single docking algorithm [13, 14]. A consensus scoring strategy was used in this project for the accurate prediction of compounds that will bind effectively to the active site of a target molecule.

3.2. DOCK

The molecular docking application DOCK [15] was chosen for being a nearly all-in-one virtual screening solution and because of its past successes in identifying effective inhibitory molecules [2, 16]. Other appealing features of the DOCK software package are the inclusion of a number of different docking algorithms, which allows for consensus scoring, and the MPI functionality.

3.3. MPI (message passing interface)

MPI is a library specification for message-passing programming, widely used for parallel computing in a single computing cluster system [17] and has been a feature of DOCK since version 5.0 [18]. The node where DOCK MPI is invoked becomes the master node and takes the role of delivering compound-target pairs to be scored by DOCK MPI processes executed on slave nodes within the cluster. Results are retrieved and ranked by the master node and the next compound-target pair to be scored is sent to a slave node [19]. With the MPI implementation of DOCK, input and output data are all localized within a single cluster. Thus, gains in performance can only be increased with the addition of computational resources to the system. This requirement may be a financial or administrative hurdle that is not possible for most biomedical labs to address, resulting in sacrifices in docking accuracy and/or the size the database to be

screened, or increased laboratory time to finish the screening.

3.4. Databases

One of the most important enablers of drug discovery through virtual screening is the availability of chemical databases [3]. The expansive ZINC database [20] provides a diverse chemical space in which to perform virtual screening. A successful example of ZINC being used in virtual screening was in the identification of novel β-lactamase inhibitors [23]. ZINC has an exhaustive collection of commercially available compounds, free access, and availability in a preprocessed mol2 format, making it an ideal companion for this study. The DOCK program requires the input compounds to have hydrogens added, contain charge information, and be stored in the mol2 format [21]. In addition, property-filtered subsets are also available for download from ZINC. For this project, the 2,066,906 compound "druglike" subset, made up of compounds with molecular weights of 150-500 Daltons and having no more than 10 hydrogen donors [22] was screened.

3.5. Grid technologies

An upper layer of Grid middleware, built on top of the services provided by Globus, usually performs the role of scheduling and distributing the workload. Nimrod/G [24] is an example of this type of software and it was used to implement an earlier version of DOCK in a Grid environment [5], tightly integrated with a number of Globus services to achieve high efficiency. Since maximum efficiency was not the primary objective of this project, a virtual screening platform was designed with a simpler architecture to allow for a rapid, portable setup process and greater flexibility in using a number of different docking methods.

The components and interfaces of the Globus toolkit 4.x (GT4) [9] are structured with Web services for a more secure and consistent Grid middleware architecture. Built on top of GT4, Opal [25] is a method of wrapping a legacy scientific application as a Web service to provide access to an installation on a remote cluster through HTTP. The access to an application on the Grid environment through the use of Web service architecture is referred to as a Grid service. Opal is usually combined with workflow tools such as Kepler [26] to create automated systems that perform experiments on the Grid, but this adds another layer of software for a scientist to learn in creating a custom screening. In the interest of maintaining a simple architecture with platform flexibility, this project employed Opal Operation Provider (Opal OP) [27] coupled with simple Perl scripts to construct the virtual screening platform. Opal OP utilizes Opal's technology, but wraps the legacy application as an operation provider, or a programming module, which can be accessed through a Grid service. This method gives application developers more flexibility by allowing extension of their Grid service. Grid service creation is simplified with the Opal OP toolkit that automates the complex task of writing the web service description language (WSDL) file and building and deploying the service into a GT4 container.

4. Methods

4.1. Two Phase Screening

Utilizing the most rigorous scoring methods available in the DOCK application with a database as expansive as ZINC is still a substantial task, even with grid resources. With the idea that compounds with features unsuitable for binding to the target molecule should not be screened with more accurate, computationally expensive screening methods, the experiment is split into two phases. The first phase performs docking with rapid, less accurate methods to identify and remove compounds with unwanted chemical and geometrical properties. A second phase of more involved molecular dynamics and solvation energy calculations is performed on a smaller portion of the compound search space. This two-phase screening model has been described as a sensible approach to virtual screening [28].

4.2. Screening with DOCK

The first phase screening of the "druglike" subset of ZINC was performed using DOCK's energy scoring method. In this method, electrostatic and Van der Waals forces are calculated to predict the approximate molecular mechanics interaction energies of each compound with the target molecule [29]. Specifically, DOCK uses an "anchor-and-grow" algorithm that first places a rigid portion of the compound, the anchor, into the binding site and calculates scores as it gradually builds the molecule piece-by-piece [4].

The relatively large binding site of the target was represented by 40 "spheres" generated using SPHGEN [30] and settings for the anchor-and-grow portion of the docking process were as follows: Minimum anchor size was defined as six heavy atoms and the max number of allowed orientations was 500. The number of anchor orientations and conformations per growth cycle were set to a maximum of 100 and 20,
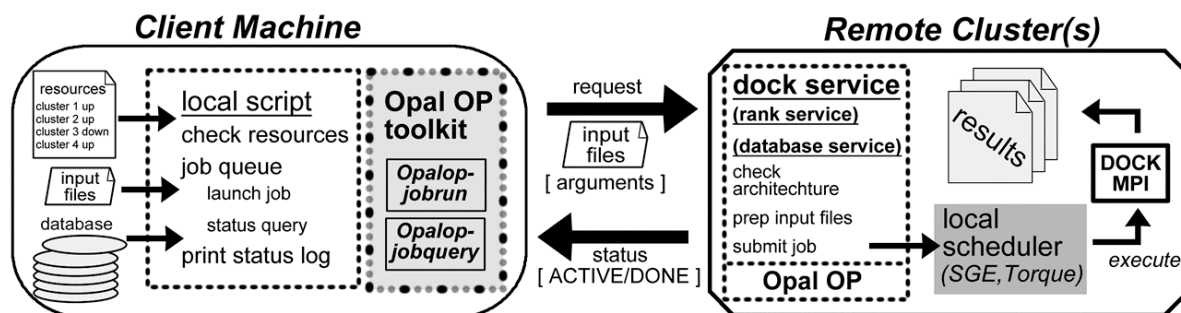
Figure 1. The DOCK MPI job distribution architecture, communicating through Opal OP to the hosted Grid Services.

respectively. This generated a number of conformations that were scored and ranked amongst each other to identify which one is most energetically favorable. Simplex minimizations were also performed on each compound-target docking, reducing the amount of conformation sampling required to find the nearest local energy minimum [29]. Rigid anchor orientation had ten cycles of simplex minimization, while the flexible growth stage was set to twenty.

Energy scoring was used again in the second phase of the screening with more stringent parameters, resulting in more extensive conformation searching and minimization, in order to more accurately predict each of the remaining compound's most favorable binding conformation. After using the energy score, the AMBER score method [29] in DOCK provided a very thorough prediction of binding affinity for compound-target pairs. While still considering the electrostatic and Van der Waals forces present in the pairing, the AMBER score also takes into account solvation energies and has the ability to perform molecular dynamic simulations with both compound and target binding site flexibility.

A consensus was taken between the ranked compound results from both the strict energy score and AMBER score screenings to reduce the chance of false positives in the results. Although more involved consensus scoring methods are available [31, 32], a simple sum of each compound's rankings from both screenings made up the consensus process in this project. After obtaining the final ranked list of compounds, visual inspection of the bound compound-target pair conformation output by DOCK was performed to check for obvious errors in the predicted compound-target interaction. Molecular visualization was done in Chimera [33]. This software was also used to prepare the target molecule for docking by removing any bound ligands or water molecules, adding hydrogens, and calculating charge for the original crystal structure.

4.3. Implementing DOCK on the Grid

Assembling a virtual screening system that took advantage of distributed computational resources offered numerous challenges. The first and most obvious problem with implementing DOCK on the Grid is to distribute an experiment.

The "druglike" database was available from ZINC divided as 88 equal slices and each executed job of DOCK MPI received a slice of the database. In its compressed form, each slice was around 20MB in size and allowed for a number of options in transferring input data to remote clusters. These options included downloading directly from ZINC, storing on one of the clusters to be accessed via HTTP, or having a local copy on each cluster. This project stored one copy of the compressed database on a single cluster, and let each remote cluster retrieve and decompress the database slice it was assigned during the screening.

The workload distribution was achieved by creating Grid services, referred to as the dock service, on each remote cluster to perform any necessary preparation of input files and then submit the DOCK MPI job to scheduling software such as SGE [34] or Torque [35] for execution. With the dock service hosted on remote clusters, the user starts the screening process by running a Perl script on a local machine that keeps a queue of DOCK MPI jobs distributed on a first-come, first-served basis. This local script assigns a slice of the database and submits a job to a dock service through the *opalop-jobrun* command. This system is illustrated in Figure 1. The local script also tracks job status and the overall progress of the experiment by calling the *opalop-jobquery* command. Once a database slice finishes on a remote cluster, the local script receives a "DONE" signal and the next database slice is submitted to that cluster and screened. Other features of the coupled local script and dock services include the ability to add or remove resources being used while the experiment is running and checking the architecture on the remote clusters to ensure proper

submission, attending to the heterogeneous nature of the Grid.

4.4. Ranking service

Once all of the database slices have been docked against the target, the results remain scattered across the Grid resources on remote clusters and need to be retrieved by the user for sorting and analysis. Another pairing of a local Perl script that manages job requests and a Grid service called the ranking service, which is hosted on the remote cluster master node, carries out the task of searching for results stored on the remote clusters. The energy scores output by all compound-target docking experiments are gathered, sorted, and written to a file as a list of compound names with their corresponding scores. When the search and sorting tasks complete on the remote clusters, the script on the local machine gathers all the score lists via HTTP, to be ranked and written to a file in the order of the most energetically favorable (most negative) energy score to the least favorable (most positive). This ranked list serves as the screening results to reference when selecting the compounds to be tested in the wet-bench lab experiments.

4.5. Database service

The two-phase screening method employed in this study calls for the ability to create a smaller subset database made up of the top-ranked portion of the compounds initially screened with the fast docking methods. Output from DOCK includes a score for every compound docked with the target molecule, as well as the optional conformational data of the most energetically favorable compound-target complex. This data can be used for analysis such as visualization, but more importantly, different docking methods can take advantage of these pre-determined, favorable conformations to calculate solvation energies and/or perform molecular dynamics simulations. These second phase scoring methods, such as the AMBER score, often require considerably more computation time, thus requiring the use of a smaller, filtered database.

Grid services, referred to as the database services in this case, are hosted on the master node of remote clusters and are again utilized to orchestrate an automated search for and retrieval of output data scattered across Grid resources. The script on the local machine queries the user about what top percentage of the screened database will be used to construct the new, filtered database. By referencing the ranked list of scored compounds, the names of the top percentage of compounds and the location information for each

compound's output data are sent to the database services on the remote clusters. Only the data for the top percentage of compounds is located, grouped, and retrieved back to the user via HTTP. The resulting grouped data makes up the new, filtered database.

5. Results

The Grid service-based platform developed in this project to distribute the screening workload, gather results, and build a database from those results for subsequent re-screens, served its intended purpose of solving the data management challenges encountered when performing scientific experiments in a Grid environment. The 2,066,906 compounds contained in the "druglike" subset of ZINC library were successfully screened against the catalytic site of the protein tyrosine phosphatase SHP-1.

Up to 200 processors on seven different clusters of the PRAGMA Grid [36] were utilized and an average of 180 processors were used at one time during the first phase of the screening experiment. More detailed resource information is described in Table 1. The experiment ran from July 15-26, 2007, for a total of 11.56 days of laboratory time required to complete this screening, the equivalent of over 5½ years CPU time.

Table 1. Performance of each cluster during the virtual screening experiment

Cluster name	Number of utilized CPUs	Number of finished slices	Average time per slice (hr)	Location
TEA	10-40	16	16.5	Osaka U, JP
CAFÉ	10-20	9	30.5	Osaka U, JP
FSVC	20-60	24	10.7	AIST, JP
XSVC	10-20	15	17.0	AIST, JP
LZU	10-20	12	21.4	LanZhou U, CN
Rocks-52	15-20	8	32.7	SDSC, CA
TGC	10-20	3	19.8	U of Illinois, IL

Table 1 does not reflect a measure of maximum Grid performance since resources were not dedicated for the experiment and there was a variance in workload from other users within each cluster over the course of the screening. The combination of input delivery, input preparation, and execution time determines the time required to finish each slice of the database. Submitted DOCK MPI jobs also could have been idle in a cluster's queue for a period of time as other users utilize the resources. Notably, splitting the database into a greater number of slices would reduce the average time per slice and is a possible method to

Table 2. Top 25 ranked compounds from the second phase screening consensus

| Rank	ZINC ID	RANKINGS			SCORES	
		Total	Energy	AMBER	Energy	AMBER
1	4538484	1	1	-	-132.662	-
2	5553470	2	2	-	-127.161	-
3	3957200	44	4	-	-118.597	-
4	3193067	23	17	6	-92.150	-6.58E+11
5	5518012	33	24	9	-86.708	-4.96E+11
6	4410251	47	33	14	-82.004	-1.31E+11
7	4811922	150	145	5	-68.236	-6.71E+11
8	6645919	162	149	13	-68.067	-1.84E+11
9	5093622	178	157	21	-67.863	-1.96E+10
10	1532056	178	178	-	-67.280	-
11	6645916	183	172	11	-67.505	-2109
12	4892459	185	185	-	-67.120	-
13	4892457	219	219	-	-66.600	-
14	2306159	246	159	87	-67.837	-132.6018
15	2139774	320	68	252	-72.501	-105.0482
16	5093617	322	307	15	-65.064	-1.2E+11
17	2391170	362	350	12	-64.613	-1.94E+11
18	1874963	412	386	26	-64.220	-3.65E+09
19	6664179	563	316	247	-64.928	-105.7376
20	5413467	591	539	52	-63.027	-219.4566
21	5093619	636	607	29	-62.609	-2.41E+09
22	4089903	706	702	4	-62.083	-7.63E+11
23	3434768	714	420	294	-63.919	-102.3496
24	3399076	745	508	237	-63.247	-106.8366
25	4366621	761	392	369	-64.165	-99.03696

increase efficiency to an extent by providing larger portions of the database to the higher performance clusters.

After completing the most expansive phase of the screening experiment, results were gathered from their respective remote clusters and organized into a ranked list. From this list, the top 10% of the scored compounds (~40,000) were selected to make up a new, filtered database for use with the more stringent energy score and AMBER score methods. The automated ranking and database building processes each completed within 10 minutes and required only a single command for execution.

The final list of ranked compounds was obtained from taking a consensus of the results from the second phase screening. The first twenty-five compounds in the list are displayed in Table 2. Compounds without an AMBER score were incorrectly oriented during the energy score and rejected. Before selecting the compounds to be purchased and tested in the lab, the binding conformations and chemical structures of a number of compounds were visualized and reviewed to identify any obvious errors from the screening process. A visual comparison between the number 11 and 25 ranked compounds is shown in Figures 2A and 2B. Interestingly, compound 11 had a more energetically favorable AMBER score than compound 25 (-2109 compared to -99), but compound 11 did not bind to the catalytic site of SHP-1.

The commonly considered causes of error associated with distributed computing, such as power outages, routine resource maintenance, or disk errors [6], are out of the hands of the user in preventing delays in a virtual screening experiment. The scripts described in this paper manage job distribution by communicating with the Grid services hosted on the remote clusters and checking job status. If there is an error, a "DONE" signal will never be received and that remote cluster will not be given more slices of the database to screen. Beyond failures due to resource status, other possible causes of failed jobs are: improperly installed DOCK software, misconfigured Grid services, or errors in the database of compounds. Setup errors in this platform were avoided simply by the use of a test screen on each cluster before starting
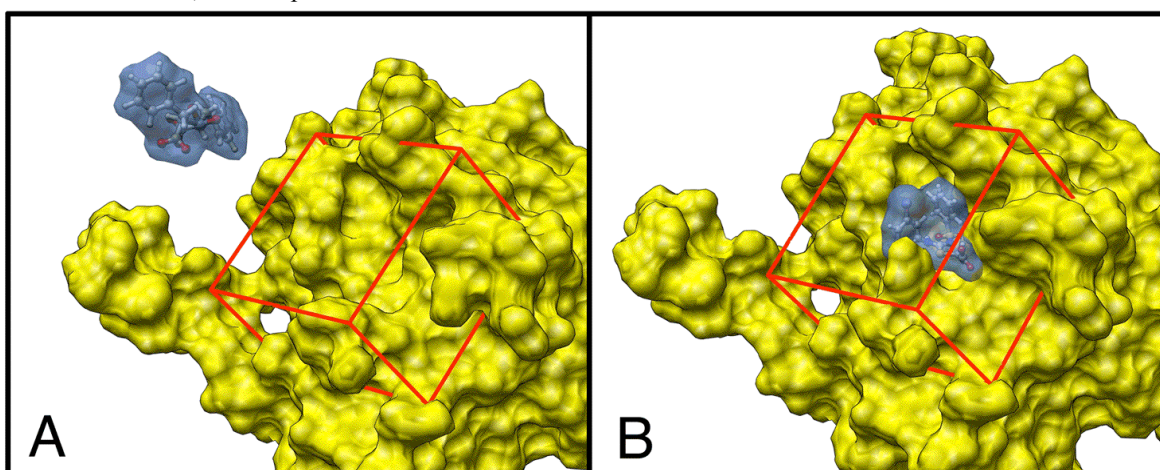
Figure 2. Visual comparison of DOCK output for compounds 11 (A) and 25 (B) in Table 2.

the large experiment. Segmentation faults of DOCK MPI 5.4 were encountered from a select number of compounds in the ZINC database. The fault would not occur until that compound was reached in the progression through its database slice, resulting in wasted CPU. A trend in structure moiety of the 136 identified compounds that caused the faults was found and examples of this structure are presented in Figure 3. The fault inducing compounds were removed and without the occurrence of electrical storms, the virtual screening platform consistently had a 100% success rate.
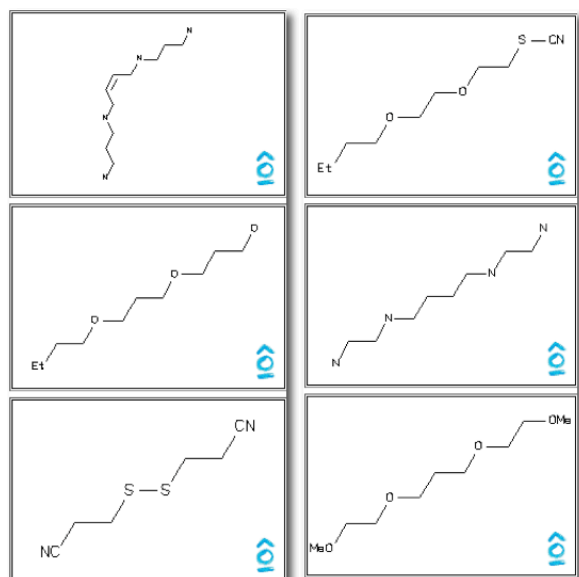
Figure 3. Examples from the 136 compounds that caused DOCK segmentation faults

6. Conclusions and discussion

This study designed a flexible, scalable workflow that successfully screened 2,066,906 compounds against the catalytic site of SHP-1. Our results demonstrate that employing DOCK in the Grid environment to conduct large-scale virtual screenings generates important data that can be used by biomedical scientists.

The speed by which these virtual screenings can be performed utilizing Grid technologies allows smaller academic laboratories to perform screenings that generate potential inhibitors that would be useful in a variety of biological experiments. All of the services and scripts were written specifically for the use with the DOCK application, but the flexibility offered by the Grid service model made it easy to quickly alter existing or create new Grid services in order to implement different docking methods. The relatively simple architecture of combining Opal OP and Perl

scripts developed for this study facilitated the use of the Grid environment with a method that does not require academic laboratories to devote large amounts of resources. There is still a need for technical support at the various clusters located at international sites. As mentioned previously, power outages and routine resource maintenance still remain issues that cannot be resolved by a single user.

The PRAGMA Grid is mainly cluster-based with the aim of scientific computing, so it is a more controlled environment compared to a desktop Grid solution that utilizes the idle CPU time of a heterogeneous network of administrative and workstation PCs. Therefore, investigating the causes of job failure may be a simpler task with the cluster-based solution. Another benefit of the cluster-based system is the use of MPI. The MPI feature of DOCK in a Grid environment is an effective method because the MPI jobs submitted within single clusters result in less inter-cluster management for distributing the screening workload, reducing the possibility of failures due to inter-cluster communication. The overhead of data transfer and job submission is also reduced.

The cause of the segmentation faults in DOCK has not been identified. One possibility is that the issue resides in the DOCK program, which was evident when the fault inducing compounds were removed, the docking completed with no failures. Because of this and due to the issue being outside of the scope of the developed platform, a method to resubmit the failed docking with the same database slice to another cluster was not pursued.

It should be noted that the energy scores alone are not sufficient in determining the best compound binding to the protein target. As our results show, some of the very best energy scores yielded compounds that did not bind to the catalytic site of SHP-1. Only direct visual inspection of the protein-compound can verify the interaction.

In summary, the Grid service model provides a powerful environment to perform virtual screening experiments on a very large scale, without an investment in dedicated computational resources. The distributed nature of the Grid environment creates an opportunity for international collaboration, which will facilitate future virtual screening experiments.

7. Acknowledgements

8. References

[1] T. Hou and X. Xu, "Recent Development and Application of Virtual Screening in Drug Discovery: An Overview," *Curr. Pharm. Des.,* April 2004, pp. 1011-1033.

[2] T. N. Doman, S. L. McGovern, et al, "Molecular Docking and High-Throughput Screening for Novel Inhibitors of Protein Tyrosine Phosphatase-1B," *J. Med. Chem.,* 2002, pp. 2213-2221.

[3] B. K. Shoichet, "Virtual screening of chemical libraries," *Nature,* 2004, pp. 862-865.

[4] S. Makino and I. D. Kuntz, "Automated flexible ligand docking method and its application for database search," *Journal of Computational Chemistry,* 1997, pp. 1812-1825.

[5] R. Buyya, K. Branson, et al, "The Virtual Laboratory: a toolset to enable distributed molecular modeling for drug design on the World-Wide Grid," *Concurrency and Computation: Practice and Experience,* 2003, pp. 1-25.

[6] N. Jacq, J. Salzemann, et al, "Grid-enabled Virtual Screening Against Malaria," *Journal of Grid Computing,* 2008, pp. 29-43.

[7] H. C. Lee, J. Salzemann, et al, "Grid-Enabled High-Throughput In Silico Screening Against Influenza A Neuraminidase," *IEEE Transactions on NanoBioscience,* Dec. 2006, pp. 288-295.

[8] A. Chien, I. Foster and D. Goddette, "Grid technologies empowering drug discovery," *Drug Discov. Today,* 2002, pp. 176-180.

[9] I. Foster, "Globus Toolkit Version 4: Software for Service-Oriented Systems," *Journal of Computer Science and Technology,* 2006, pp. 513-520.

[10] I. Foster, C. Kesselman, et al, *The Physiology of the Grid,* John Wiley & Sons, Ltd, West Sussex, England, 2003.

[11] M. Taufer, M. Crowley, et al, "Study of a highly accurate and fast protein-ligand docking based on molecular dynamics," *Parallel and Distributed Processing Symposium,* 2004, pp. 188.

[12] C. Bissantz, G. Folkers and D. Rognan, "Protein-Based Virtual Screening of Chemical Databases. 1. Evaluation of Different Docking/Scoring Combinations," *J. Med. Chem.,* 2000, pp. 4759-4767.

[13] P. S. Charifson, J. J. Corkery, et al, "Consensus Scoring: A Method for Obtaining Improved Hit Rates from Docking Databases of Three-Dimensional Structures into Proteins," *J. Med. Chem.,* 1999, pp. 5100-5109.

[14] J. M. Yang, Y. F. Chen, et al, "Consensus scoring criteria for improving enrichment in virtual screening," *J. Chem. Inf. Model.,* 2005, pp. 1134-1146.

[15] *UCSF DOCK* http://dock.compbio.ucsf.edu/

[16] S. C. Hopkins, R. D. Vale and I. D. Kuntz, "Inhibitors of Kinesin Activity from Structure-Based Computer Screening," *Biochemistry (N.Y.),* 2000, pp. 2805-2814.

[17] W. Gropp, E. Lusk, et al, *Using MPI: Portable Parallel Programming with the Message-Passing Interface,* MIT Press, Cambridge, Massachusetts, 1999.

[18] D. Moustakas, P. Lang, et al, "Development and validation of a modular, extensible docking program: DOCK 5," *J. Comput. Aided Mol. Des.,* 2006, pp. 601-619.

[19] I. D. Kuntz, D. T. Moustakas and P. T. Lang, "DOCK 5.4 User Manual," 2006, http://dock.compbio.ucsf.edu/

[20] J. J. Irwin and B. K. Shoichet, "ZINC--a free database of commercially available compounds for virtual screening," *J. Chem. Inf. Model.,* 2005, pp. 177-182.

[21] *SYBYL Mol2 Format* http://www.tripos.com/

[22] C. A. Lipinski, "Drug-like properties and the causes of poor solubility and poor permeability," *J. Pharmacol. Toxicol. Methods,* 2000, pp. 235-249.

[23] J. J. Irwin, F. M. Raushel and B. K. Shoichet, "Virtual Screening against Metalloenzymes for Inhibitors and Substrates," *Biochemistry,* 2005, pp. 12316-12328.

[24] D. Abramson, J. Giddy and L. Kotler, "High Performance Parametric Modeling with Nimrod/G: Killer Application for the Global Grid?" *Ipdps,* IEEE Computer Society, 2000, pp. 520.

[25] S. Krishnan, B. Stearn, et al, "Opal: Simple Web Services Wrappers for Scientific Applications," *ICWS,* 2006, pp. 823-832.

[26] W. W. Li, S. Krishnan, et al, "Building Cyberinfrastructure for Bioinformatics Using Service Oriented Architecture," *Ccgrid,* IEEE Computer Society, 2006, pp. 39.

[27] K. Ichikawa, S. Date, et al, "Opal OP: An Extensible Grid-Enabling Wrapping Approach for Legacy Applications," *GCA,* Singapore, 2007, pp. 117-127.

[28] P. A. Kollman, I. Massova, et al, "Calculating structures and free energies of complex molecules: combining molecular mechanics and continuum models," *Acc. Chem. Res.,* 2000, pp. 889-897.

[29] P. T. Lang, D. Moustakas, et al, "DOCK 6.1 Users Manual," 2007, http://dock.compbio.ucsf.edu/

[30] I. D. Kuntz, J. M. Blaney, et al, "A geometric approach to macromolecule-ligand interactions," *J. Mol. Biol.,* 1982, pp. 269-288.

[31] J. M. Yang and D. F. Hsu, "Consensus scoring criteria in structure-based virtual screening," *Emerging Information Technology Conference,* 2005, pp. 3.

[32] A. Oda, K. Tsuchida, et al, "Comparison of consensus scoring strategies for evaluating computational models of protein-ligand complexes," *J. Chem. Inf. Model.,* 2006, pp. 380-391.

[33] E. F. Pettersen, T. D. Goddard, et al, "UCSF Chimera - A visualization system for exploratory research and analysis," *Journal of Computational Chemistry,* 2004, pp. 1605-1612.

[34] W. Gentzsch, "Sun Grid Engine: Towards Creating a Compute Power Grid," *Ccgrid,* IEEE Computer Society, 2001, pp. 35.

[35] *Open PBS* http://www.openpbs.org

[36] C. Zheng, D. Abramson, et al, "The PRAGMA Testbed - Building a Multi-Application International Grid," *Ccgrid,* 2006, pp. 57.