Hindawi Publishing Corporation
BioMed Research International
Volume 2013, Article ID 170356, 7 pages
http://dx.doi.org/10.1155/2013/170356

Research Article

Cloud Computing for Protein-Ligand Binding Site Comparison

Che-Lun Hung[1] and Guan-Jie Hua[2]

[1] *Department of Computer Science and Communication Engineering, Providence University, Taiwan Boulevard, Shalu District, Taichung 43301, Taiwan*
[2] *Department of Computer Science and Information Engineering, Providence University, Taiwan Boulevard, Shalu District, Taichung 43301, Taiwan*

Correspondence should be addressed to Che-Lun Hung; clhung@pu.edu.tw

Received 18 December 2012; Accepted 28 March 2013

Academic Editor: Ching-Hsien Hsu

The proteome-wide analysis of protein-ligand binding sites and their interactions with ligands is important in structure-based drug design and in understanding ligand cross reactivity and toxicity. The well-known and commonly used software, SMAP, has been designed for 3D ligand binding site comparison and similarity searching of a structural proteome. SMAP can also predict drug side effects and reassign existing drugs to new indications. However, the computing scale of SMAP is limited. We have developed a high availability, high performance system that expands the comparison scale of SMAP. This cloud computing service, called Cloud-PLBS, combines the SMAP and Hadoop frameworks and is deployed on a virtual cloud computing platform. To handle the vast amount of experimental data on protein-ligand binding site pairs, Cloud-PLBS exploits the MapReduce paradigm as a management and parallelizing tool. Cloud-PLBS provides a web portal and scalability through which biologists can address a wide range of computer-intensive questions in biology and drug discovery.

1. Introduction

By virtue of its 3D structure, a protein performs thousands of life-critical functions at the molecular level. Detection and characterization of protein structural ligand binding sites and their interactions with binding partners are pivotal to a wide range of structure-function correlation problems—predicting functions for structural genomics targets, identifying and validating drug targets, prioritizing and optimizing drug leads, and correlating molecular functions to physiological processes in drug design [1].

Xie et al. [2–4] proposed an efficient and robust algorithm called SMAP, which quantitatively characterizes the geometric properties of proteins. Ligand binding sites predicted by SMAP have been experimentally validated [4–7]. SMAP has also been applied to drug design problems, such as constructing drug-target interaction networks [4], designing polypharmacology drugs [5], assigning old drugs to new indications [6], and predicting the side effects of drugs [8, 9]. The web service tool SMAP-WS [1] implements SMAP via Opal [10]. Although the parallel implementation of SMAP improves the speed of database searching, it cannot operate at the scale and availability demanded by current Internet technology.

Recently, an Internet service concept known as cloud computing has become popular for providing various services to users. The cloud computing environment is a distributed system with extremely scalable IT-related capabilities, providing multiple external customers with numerous services. Cloud computing also enables the copying of vast datasets to many users with high fault tolerance. Another popular open-source software framework designed for data-intensive distribution is Hadoop [11]. This framework processes petabytes of data intercepting thousands of nodes. Hadoop provides the MapReduce programming model, by which parallel computing of large data sets can be implemented in the cloud computing environment. MapReduce enables distributed computing of the mappers and reducers. Each mapper performs an independent map operation which is parallelized with the tasks of other mappers. Similarly, a set of reducers can perform a set of reduce operations. All

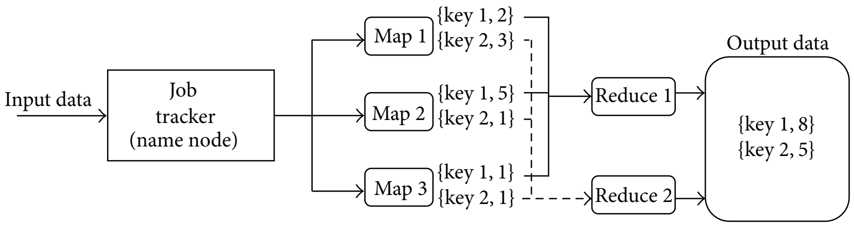


Figure 1: Map/reduce framework of Hadoop.

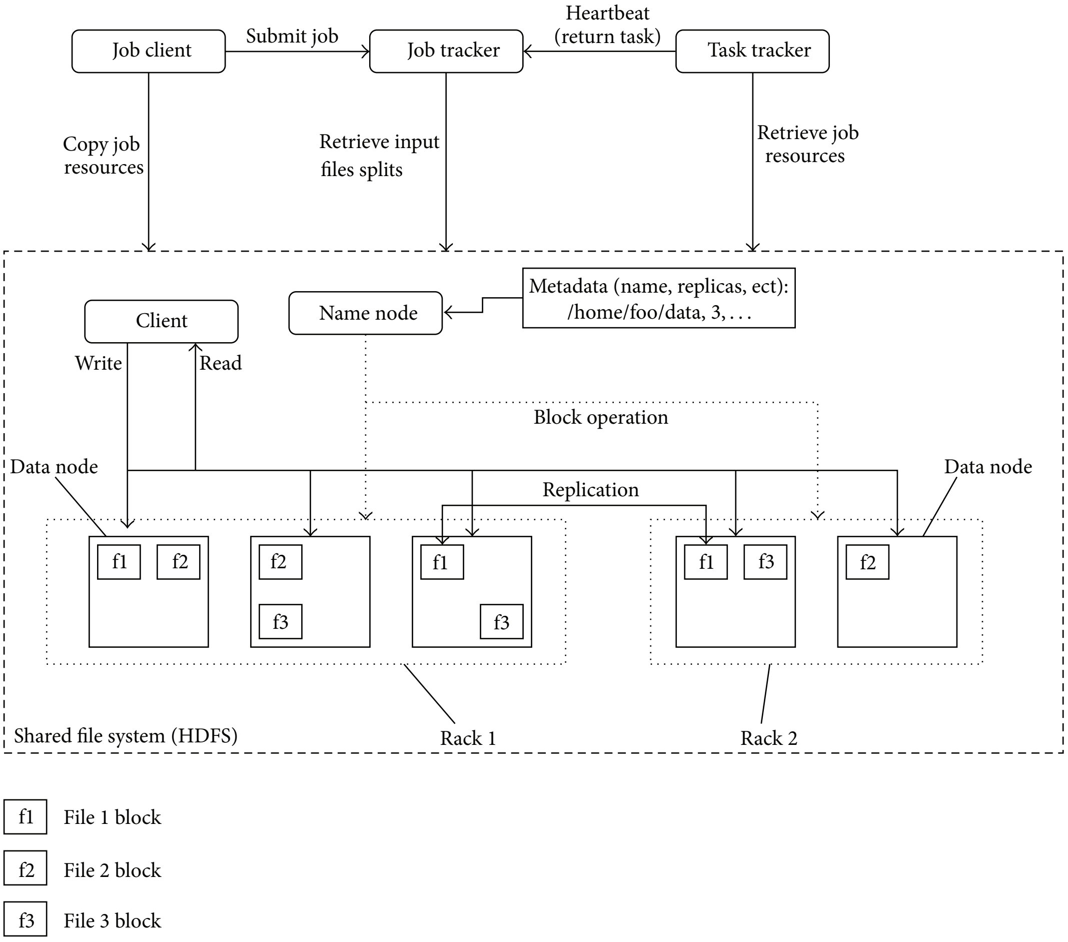


Figure 2: The architecture of Hadoop cluster.

outputs of the map operations possessing the same key are presented to the same reducer at the same time. Two additional important benefits of Hadoop are scalability and fault tolerance. Hadoop can guide jobs toward successful completion even when individual nodes or network components experience high failure rates. Meanwhile, a machine can be readily attached as a mapper and reducer in the Hadoop cluster. The Hadoop platform, therefore, is regarded as a superior solution to real-world data distribution problems. To date, Hadoop has been applied in a range of bioinformatics domains [12–16].

Cloud computing platforms are usually based on virtualization technology. Computing resources are combined or divided into one or more operating environments using methodologies such as hardware and software partitioning or aggregation, partial or complete machine simulation, and emulation and time sharing. A virtual machine (VM) is a machine simulation created by virtualization technology, which resides in a physical machine and shares its physical resources. The web service Amazon Elastic Compute Cloud (Amazon EC2) [17] uses virtualization technology to generate resizable computing capacity in the cloud. The service provides a true virtual computing environment, allowing users to launch VMs with a variety of operating systems. Users can construct their own elastic cluster systems by attaching or removing VMs.

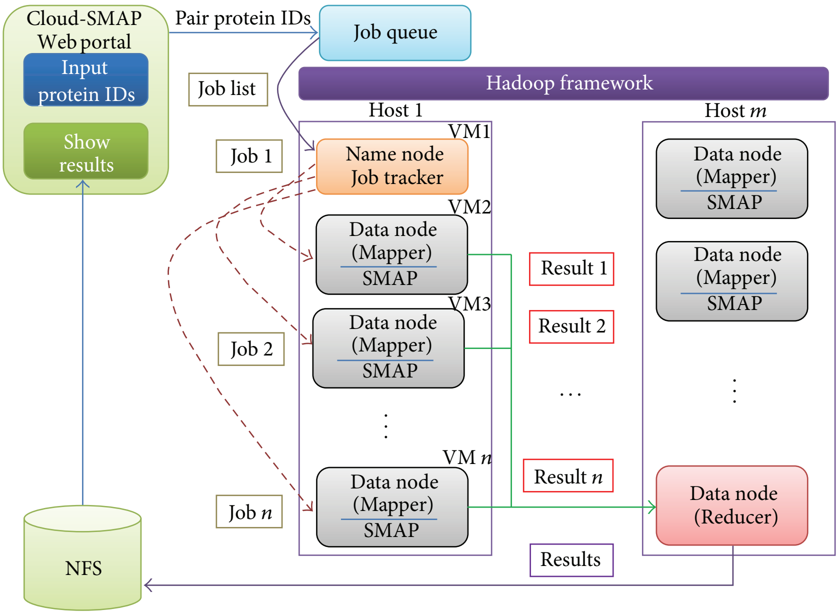


FIGURE 3: The cloud platform of Cloud-PLBS.

In this paper, we combine three technologies, Hadoop framework, virtualization, and SMAP, to develop a cloud computing service for structural ligand binding site comparison. Each mapper or reducer in the cloud platform is a VM. The platform uses MapReduce to simultaneously process numerous comparison jobs. Similarly, the number of VMs can be adjusted to the size of the comparison job (large and small jobs demand more and fewer VMs, resp.). Hadoop enables our cloud platform to recover the comparison job from a crashed VM or physical machine by reassigning the job to a healthy VM or a physical machine. The cloud platform can achieve high performance, scalability, and availability. The experimental results demonstrate that applying the Hadoop framework on a virtualization platform enhances the computational efficiency of the proposed service. The cloud service is available at http://bioinfo.cs.pu.edu.tw/cloud-PLBS/index.html.

2. Method

Cloud-PLBS is a robust, elastic cloud computing service for protein-ligand binding site comparison. It guarantees rapid return of comparison results. Cloud-PLBS embraces three technologies, virtualization, Hadoop, and SMAP, used to build the cloud computing infrastructure, perform parallel computation, and compare ligand binding sites, respectively.

2.1. Structural Proteome-Wide Ligand Binding Site Comparison. SMAP is an efficient and robust algorithm that performs pair-wise comparison of two potential ligand binding sites. The user enters two protein structure IDs, and SMAP downloads the relevant protein structures from the RCSB Protein Data Bank (PDB) [18]. Protein structure binding sites are compared in four stages.

Step 1. The protein structures are represented by C-α atoms for structural variation tolerance.

Step 2. Amino acid residues are characterized by surface orientation and a geometric potential.

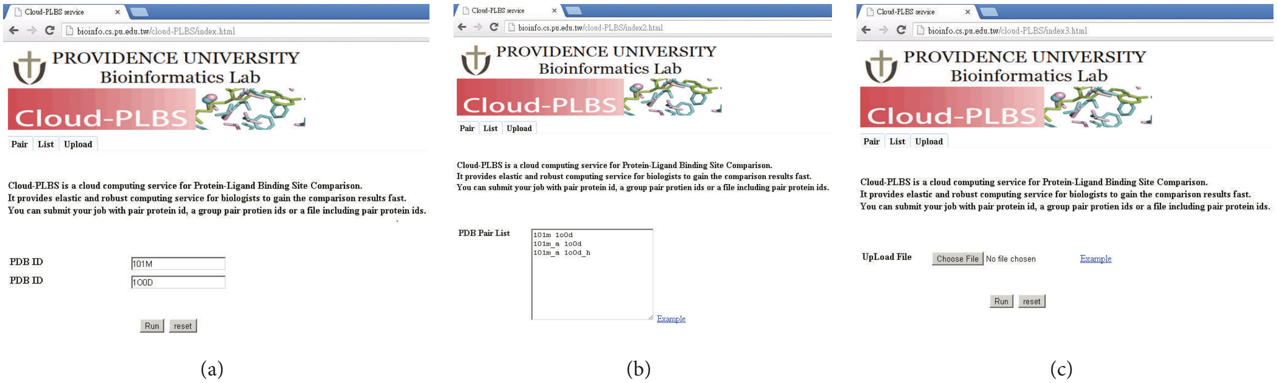
Step 3. Protein structures are compared using a sequence order-independent profile-profile alignment (SOIPPA) algorithm.

Step 4. Similarity between two binding sites is determined through the combination of geometrical fit, residue conservation and physiochemical similarity.

In Cloud-PLBS, each paired protein structure comparison is regarded as an SMAP job. Each SMAP job compares two ligand binding sites by the four stages listed above.

2.2. Cloud-PLBS by Combining Hadoop and Virtualization. As mentioned above, Cloud-PLBS comprises Hadoop, virtualization, and SMAP. Hadoop coordinates computing nodes to parallelize distributed data. Parallel computing applications are developed via the map/reduce parallel programming model. The standard map/reduce mechanism has been applied in many successful cloud computing service providers, such as Yahoo, Amazon EC2, IBM, and Google. The map/reduce framework of Hadoop is illustrated in Figure 1. Input data are divided into smaller chunks corresponding to the number of mappers. The mapper stage output is formatted as ⟨key, value⟩ pairs. Output from all mappers is classified by key before being distributed to the reducer. The reducer then combines the keyed values. Its output is also formatted as ⟨key, value⟩ pairs, where each key is unique.

The Hadoop cluster includes a single master and multiple slave nodes. The master node comprises a job tracker, task tracker, name node and data-node. A slave node, or



bioinfo.cs.pu.edu.tw/cloud-PLBS/index.html

Cloud-PLBS is a cloud computing service for Protein-Ligand Binding Site Comparison.
It provides elastic and robust computing service for biologists to gain the comparison results fast.
You can submit your job with pair protein id, a group pair protien ids or a file including pair protein ids.

PDB ID 101M
PDB ID 100D

bioinfo.cs.pu.edu.tw/cloud-PLBS/index2.html

Cloud-PLBS is a cloud computing service for Protein-Ligand Binding Site Comparison.
It provides elastic and robust computing service for biologists to gain the comparison results fast.
You can submit your job with pair protein id, a group pair protien ids or a file including pair protein ids.

PDB Pair List

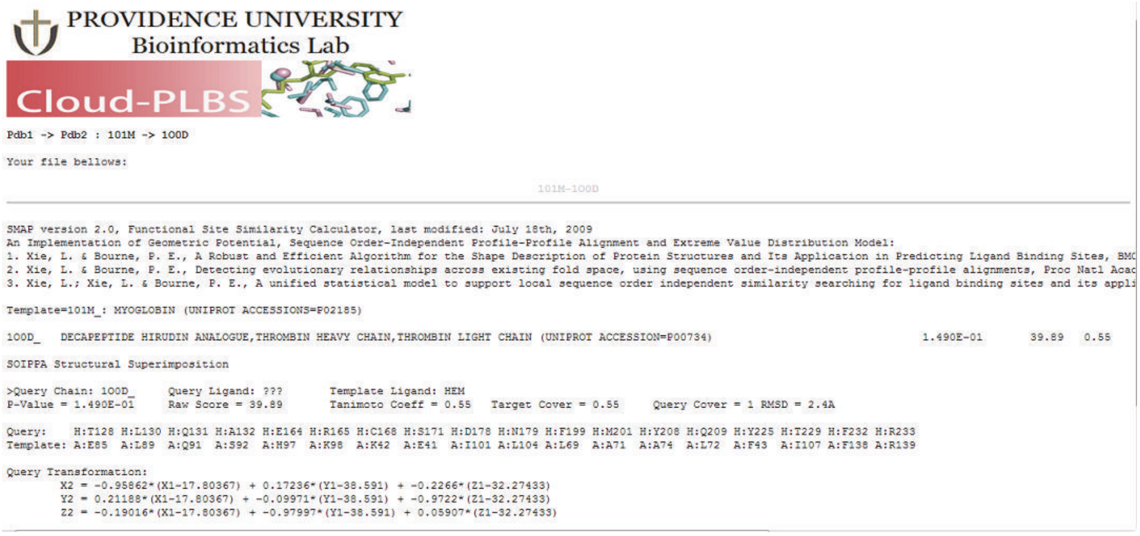
```
101m 1o0d
101m_a 1o0d
101m_a 1o0d_h
```

bioinfo.cs.pu.edu.tw/cloud-PLBS/index3.html

Cloud-PLBS is a cloud computing service for Protein-Ligand Binding Site Comparison.
It provides elastic and robust computing service for biologists to gain the comparison results fast.
You can submit your job with pair protein id, a group pair protien ids or a file including pair protein ids.

UpLoad File No file chosen

Figure 4: Web portal of Cloud-PLBS for entering protein IDs. (a) Two protein IDs. (b) List of paired protein IDs. (c) Upload file.



```
Pdb1 -> Pdb2 : 101M -> 100D

Your file bellows:

                                   101M-100D

SMAP version 2.0, Functional Site Similarity Calculator, last modified: July 18th, 2009
An Implementation of Geometric Potential, Sequence Order-Independent Profile-Profile Alignment and Extreme Value Distribution Model:
1. Xie, L. & Bourne, P. E., A Robust and Efficient Algorithm for the Shape Description of Protein Structures and Its Application in Predicting Ligand Binding Sites, BMC
2. Xie, L. & Bourne, P. E., Detecting evolutionary relationships across existing fold space, using sequence order-independent profile-profile alignments, Proc Natl Acad
3. Xie, L.; Xie, L. & Bourne, P. E., A unified statistical model to support local sequence order independent similarity searching for ligand binding sites and its appli

Template=101M_: MYOGLOBIN (UNIPROT ACCESSIONS=P02185)

100D_   DECAPEPTIDE HIRUDIN ANALOGUE,THROMBIN HEAVY CHAIN,THROMBIN LIGHT CHAIN (UNIPROT ACCESSION=P00734)                1.490E-01      39.89   0.55

SOIPPA Structural Superimposition

>Query Chain: 100D_     Query Ligand: ???       Template Ligand: HEM
P-Value = 1.490E-01     Raw Score = 39.89       Tanimoto Coeff = 0.55   Target Cover = 0.55     Query Cover = 1 RMSD = 2.4A

Query:    H:T128 H:L130 H:Q131 H:A132 H:E164 H:R165 H:C168 H:S171 H:D178 H:N179 H:F199 H:M201 H:Y208 H:Q209 H:Y225 H:T229 H:F232 H:R233
Template: A:E85  A:L89  A:Q91  A:S92  A:H97  A:K98  A:K42  A:E41  A:I101 A:L104 A:L69  A:A71  A:A74  A:L72  A:F43  A:I107 A:F138 A:R139

Query Transformation:
        X2 = -0.95862*(X1-17.80367) + 0.17236*(Y1-38.591) + -0.2266*(Z1-32.27433)
        Y2 = 0.21188*(X1-17.80367) + -0.09971*(Y1-38.591) + -0.9722*(Z1-32.27433)
        Z2 = -0.19016*(X1-17.80367) + -0.97997*(Y1-38.591) + 0.05907*(Z1-32.27433)
```

Figure 5: The result produced by Cloud-PLBS. The protein IDs are 101 M and 100D.

computing node, consists of a data node and task tracker. The job tracker distributes map/reduce tasks to computing nodes within the cluster, ideally those already containing the data, or at least within the same rack. A task tracker node accepts map, reduce and shuffle operations from the job tracker. The architecture of the Hadoop cluster is shown in Figure 2.

Hadoop Distributed File System (HDFS) is the primary file system used by the Hadoop framework. Each input file is split into data blocks that are distributed to data nodes. Hadoop also creates multiple replicas of data blocks and distributes them to data nodes throughout a cluster, ensuring reliable, extremely rapid computations. The name node serves as both a directory namespace manager and a node metadata manager for the HDFS. The HDFS architecture operates on a single name-node.

Resource capacity permitting virtualization technology can host several virtual machines within a physical machine. The proposed cloud service platform combines Hadoop and virtualization technology, such that all nodes of the Hadoop cluster reside in VMs. The cloud computing architecture of Cloud-PLBS is illustrated in Figure 3. As shown in that figure, master node (name node) and slave node (data node) constitute the master VM and slave VM, respectively. Submitted SMAP jobs are recorded in a job queue. The master node periodically obtains SMAP jobs from the job queue and assigns them to slave nodes; a slave node (or mapper) performs the task. Once all of the SMAP jobs are complete, the reducer collects the comparison results from all mappers and stores them in the Network File System (NFS) storage. A single comparison result is stored in a single file in NFS. This architecture imbues Cloud-PLBS with three desirable characteristics: high performance, scalability, and availability.

2.2.1. High Performance. In Cloud-PLBS, the SMAP jobs are performed in parallel by the map/reduce framework. The number of SMAP jobs that can be performed simultaneously is the number of data nodes. If the number of SMAP jobs exceeds the number of data nodes, the number node assigns the remaining jobs as soon as a data node becomes available.

2.2.2. Availability. In the event of system failure, Cloud-PLBS continues performing SMAP jobs via the Hadoop

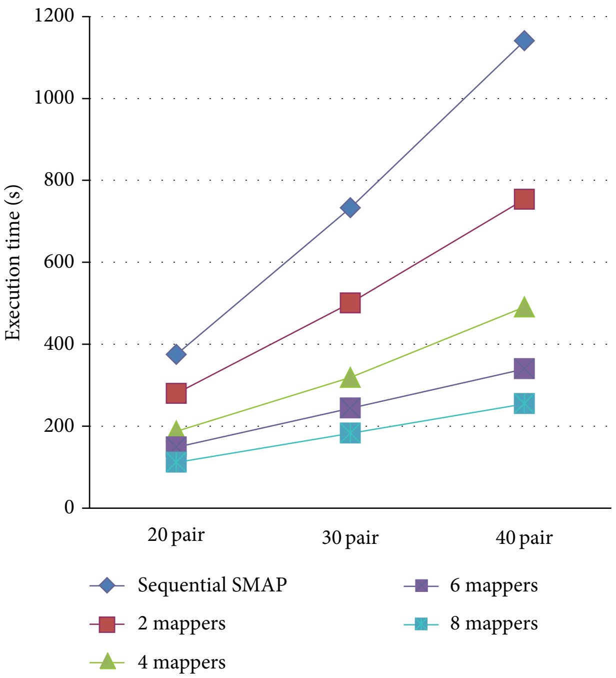


Figure 6: Performance of sequential SMAP program and Cloud-PLBS using 2, 4, 6, and 8 mappers.

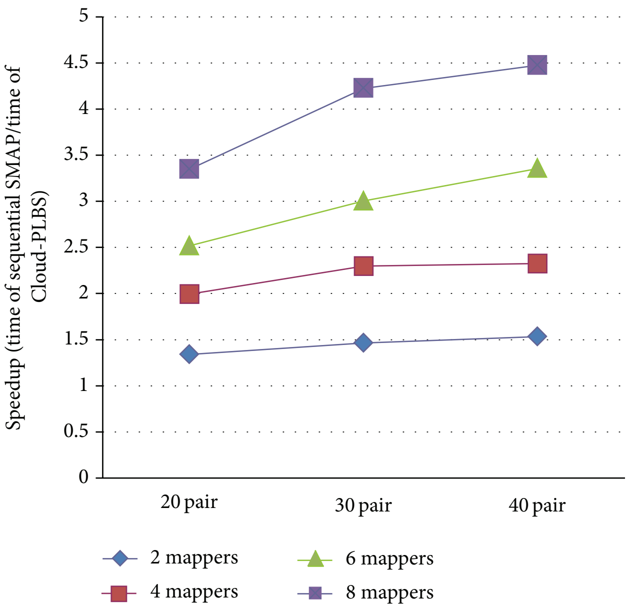


Figure 7: Execution speed of sequential SMAP program and Cloud-PLBS using 2, 4, and 6 mappers.

fault tolerance mechanism. When a data node (mapper) fails during SMAP computation, name node reassigns its job to another slave node (mapper). Therefore, in Cloud-PLBS, all of the submitted SMAP jobs are executed in the event of data node failure. A hardware failure on the physical server will terminate all virtual machines running on it. In this more catastrophic event, SMAP jobs can be reassigned to several new virtual machines created on available hosts. As a result of this operation, Cloud-SMAP has high availability.

2.2.3. Scalability. If excessively many SMAP jobs are submitted, Cloud-PLBS can create new slave VMs as data nodes to accept more jobs, leading to enhanced performance. New VMs are easily created in the Cloud-PLBS architecture. At the same time, redundant VMs can be destroyed to preserve physical resources.

3. Cloud-PLBS Platform

Cloud-PLBS is a software (SaaS) as a service service operating under the Hadoop framework and virtualization technology. The cloud computing platform is composed of an NFS server and four IBM blade servers in the Providence University Cloud Computation Laboratory. Each server is equipped with two Quad-Core Intel Xeon 2.26 GHz CPUs, 24 G RAMs, and 296 G disks. Each server can accommodate 8 virtual machines; each virtual machine is set to one core CPU, 2 G RAM, and 30 G disk running under the Ubuntu operating system version 10.4 with Hadoop version 0.2 MapReduce framework. Each virtual machine is responsible for a map operation and a reduce operation. Therefore, up to eight map/reduce operations may be undertaken.

Figure 4 shows the web portal of Cloud-PLBS. Data may be entered in three ways: by entering two protein IDs (Figure 4(a)), by listing several pairs of protein IDs (Figure 4(b)), or by uploading containing paired protein IDs (Figure 4(c)). All of these pair protein IDs are recorded in a job queue upon submission. The name node (mater node) extracts the paired protein IDs from the queue, and assigns individual SMAP jobs to data nodes (slave nodes). Figure 5 shows the results of comparisons produced by Cloud-PLBS.

4. Performance Evaluation

To assess the performance of the proposed cloud service, we compared the execution time between stand-alone SMAP and Cloud-PLBS. The performance of both programs depends upon the number of SMAP jobs (the number of paired protein IDs) and the number of computing nodes (the number of VMs). Therefore, the performance between the programs is tested with respect to these two factors. The results are shown in Figure 6. As shown in the figure, the execution time of 20 protein pairs (jobs) can be reduced from 375 seconds (consumed by the sequential SMAP program) to 280 seconds, 188 seconds, 149 seconds, and 112 seconds by executing Cloud-PLBS with 2, 4, 6, and 8 mappers, respectively. Given 20, 30 and 40 protein pairs, Cloud-PLBS with 2, 4, and 6 and 8 mappers saves roughly 30%, 54%, 66%, and 74% execution time (relative to sequential SMAP) in average, respectively (see Table 1). Figure 7 demonstrates the enhanced speed achieved by Cloud-PLBS using different numbers of mappers. Clearly, the execution time is effectively reduced when more than two mappers are involved. In general, more mappers (VMs) achieve a faster processing speed.

Table 1: Execution time and proportional reduction (relative to sequential SMAP) of Cloud-PLBS using different numbers of mappers.

Method	20 pairs		30 pairs		40 pairs	
	Execution time (sec)	Reduction rate	Execution time (sec)	Reduction rate	Execution time (sec)	Reduction rate
Sequential SMAP	375		733		1141	
Cloud-PLBS for 2 mappers	280	24.44%	501	31.66%	754	33.92%
Cloud-PLBS for 4 mappers	188	48.87%	319	56.49%	491	56.97%
Cloud-PLBS for 6 mappers	149	60.27%	244	66.72%	340	70.21%
Cloud-PLBS for 8 mappers	112	70.13%	183	75.03%	255	77.65%

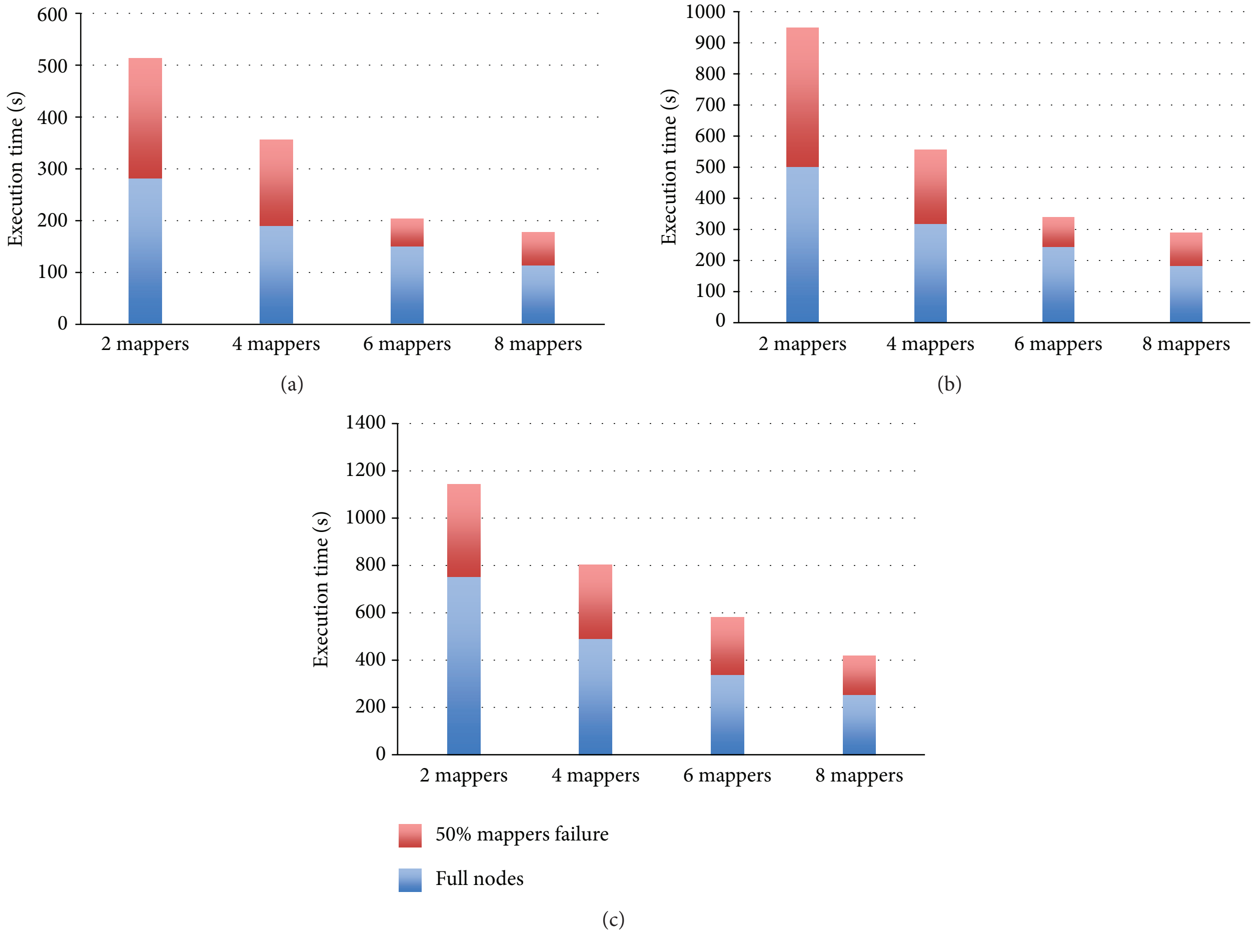


Figure 8: Execution time of a half of node failure of Cloud-PLBS. (a) 20 pair (b) 30 pair (c) 40 pair.

To evaluate the reliability and availability of the proposed cloud service, we performed a simulation to observe the performance when mappers fail. In this simulation, half of the mappers failed in the duration of executing SMAP. According to the features of Hadoop, the computing process at the failed node is able to continue at another node that has the replica of data of the failed node. In this simulation, the heartbeat time is set to one minute, and the number of replica is set to three as default. Therefore, all of jobs can be completed even when some of the nodes fail. Figures 8(a), 8(b), and 8(c) demonstrate the performance between the different number of nodes meeting corresponding half of nodes fail for processing 20 pair, 30 pair, and 40 pair data set, respectively. The execution time with no failure is shown as the blue bar, and the execution time with failure in a half of nodes is the sum of blue bar and red bar which is extra time when failure occurrence. From the experiment results, it shows that the jobs can be completed less than the double successful execution time in the proposed service. Although half of the nodes fail, the execution time of redundancy is related to the number of nodes too. There are extra 165 seconds for 8 mappers, 263 seconds for 6 mappers, 313 seconds for 4 mappers, and 391 seconds for 2 mappers when half of the nodes fail Occurs, respectively. Thereby, our cloud service is node failure-free.

5. Conclusion

The detection and characterization of protein ligand binding sites and their interactions with binding partners are an essential component of modern drug design. The software tool SMAP was designed to achieve these goals. Although SMAP outperforms most existing ligand binding site comparison tools, it cannot achieve the high scalability and availability demanded by huge database searching.

In this paper, we exploit the new internet service concept known as cloud computing. The proposed cloud computing service is called Cloud-PLBS (where PLBS denotes protein-ligand binding site). The platform integrates the Hadoop framework, virtualization technology, and SMAP tool to guarantee high performance, availability, and scalability. Cloud-PLBS ensures that all submitted jobs are properly completed, even on a large cloud platform where individual nodes or network components are prone to failure. We experimentally verified that the platform is computationally more efficient than standard SMAP. Therefore, it presents as a desirable tool for analyzing protein structure and function under reasonable time constraints.

Conflict of Interests

This paper has a potential conflict of interests with the SMAP software developed by J. Ren, L. Xie, W. Li, and P. Bourne and published in Nucleic Acids Research Web Server Issue (2010). Therefore, the authors declare the conflict of interests with SMAP software and Cloud-PLBS here.

Acknowledgment

This research was partially supported by the National Science Council under the Grants NSC-99-2632-E-126-001-MY3 and NSC-100-2221-E-126-007-MY3.

References

[1] J. Ren, L. Xie, W. W. Li, and P. E. Bourne, “SMAP-WS: a parallel web service for structural proteome-wide ligand-binding site comparison,” *Nucleic Acids Research*, vol. 38, supplement 2, pp. W441–W444, 2010.

[2] L. Xie and P. E. Bourne, “A robust and efficient algorithm for the shape description of protein structures and its application in predicting ligand binding sites,” *BMC Bioinformatics*, vol. 8, supplement 4, article S9, 2007.

[3] L. Xie and P. E. Bourne, “Detecting evolutionary relationships across existing fold space, using sequence order-independent profile-profile alignments,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 105, no. 14, pp. 5441–5446, 2008.

[4] L. Xie, L. Xie, and P. E. Bourne, “A unified statistical model to support local sequence order independent similarity searching for ligand-binding sites and its application to genome-based drug discovery,” *Bioinformatics*, vol. 25, no. 12, pp. i305–i312, 2009.

[5] J. D. Durrant, R. E. Amaro, L. Xie et al., “A multidimensional strategy to detect polypharmacological targets in the absence of structural and sequence homology,” *PLOS Computational Biology*, vol. 6, Article ID e1000648, 2010.

[6] S. L. Kinnings, N. Liu, N. Buchmeier, P. J. Tonge, L. Xie, and P. E. Bourne, “Drug discovery using chemical systems biology: repositioning the safe medicine Comtan to treat multi-drug and extensively drug resistant tuberculosis,” *PLoS Computational Biology*, vol. 5, no. 7, Article ID e1000423, 2009.

[7] J. R. Miller, S. Dunham, I. Mochalkin et al., “A class of selective antibacterials derived from a protein kinase inhibitor pharmacophore,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 106, no. 6, pp. 1737–1742, 2009.

[8] L. Xie, J. Li, L. Xie, and P. E. Bourne, “Drug discovery using chemical systems biology: identification of the protein-ligand binding network to explain the side effects of CETP inhibitors,” *PLoS Computational Biology*, vol. 5, no. 5, Article ID e1000387, 2009.

[9] L. Xie, J. Wang, and P. E. Bourne, “In silico elucidation of the molecular mechanism defining the adverse effect of selective estrogen receptor modulators,” *PLoS Computational Biology*, vol. 3, no. 11, article e217, 2007.

[10] S. Krishnan, L. Clementi, J. Ren, P. Papadopoulos, and W. Li, “Design and evaluation of Opal2: a toolkit for scientific software as a service,” in *Proceedings of the 5th 2009 World Congress on Services (SERVICES ’09)*, pp. 709–716, IEEE Conference, Los Angeles, Calif, USA, September 2009.

[11] Hadoop—Apache Software Foundation project home page, http://hadoop.apache.org/.

[12] R. C. Taylor, “An overview of the Hadoop/MapReduce/HBase framework and its current applications in bioinformatics,” *BMC Bioinformatics*, vol. 11, supplement 12, article S1, 2010.

[13] M. C. Schatz, “CloudBurst: highly sensitive read mapping with MapReduce,” *Bioinformatics*, vol. 25, no. 11, pp. 1363–1369, 2009.

[14] B. Langmead, M. C. Schatz, J. Lin, M. Pop, and S. L. Salzberg, “Searching for SNPs with cloud computing,” *Genome Biology*, vol. 10, no. 11, article R134, 2009.

[15] C. L. Hung and Y. L. Lin, “Implementation of a parallel protein structure alignment service on cloud,” *International Journal of Genomics*, vol. 2013, Article ID 439681, 8 pages, 2013.

[16] C. L. Hung and C. Y. Lin, “Open reading frame phylogenetic analysis on the cloud,” *International Journal of Genomics*, vol. 2013, Article ID 614923, 9 pages, 2013.

[17] Amazon Elastic Compute Cloud (Amazon EC2), http://aws.amazon.com/ec2/.

[18] H. M. Berman, J. Westbrook, Z. Feng et al., “The protein data bank,” *Nucleic Acids Research*, vol. 28, no. 1, pp. 235–242, 2000.