ISSN 1004-9037,CODEN SCYCE4
Journal of Data Acquisition and Processing Vol. 35,No. 4,Jul. 2020,pp. 672－681
DOI:10. 16337/j. 1004-9037. 2020. 04. 008
© 2020 by Journal of Data Acquisition and Processing
http://sjcj. nuaa. edu. cn
E-mail:sjcj@nuaa. edu. cn
Tel/Fax:＋86-025-84892742

基于Hadoop环境下蛋白质与配体分子对接模拟实验研究

钦 淳[1]，许 磊[1]，王 峰[2]，孔 韧[1]，常 珊[1]

(1. 江苏理工学院电气信息工程学院生物信息与医药工程研究所，常州，213001；2. 常州大学信息科学与工程学院，常州，213164)

摘 要：蛋白质种类和小分子的数量过多，对于药物的模拟开发运算量巨大。而分子对接是研究新药的重要手段，因此提高分子对接实验系统的工作效率十分重要。分子对接主要目的是研究蛋白质受体和配体小分子之间的作用与联系。本文通过模拟实验，搭建了一个Hadoop平台，并利用Hadoop强大的并行计算能力对蛋白质(1ppe与1uy6)与多组数量不同的配体小分子进行对接，最后对工作过程进行了相应的比较与优化。实验结果表明，该系统可以有效提高对接效率，并且具有较好的稳定性和便利性。

关键词：蛋白质；分子对接；Hadoop环境；并行计算

中图分类号：TP301　　**文献标志码**：A

Experiment Research of Docking of Protein and Ligand Molecules Based on Hadoop Environment

QIN Chun[1], XU Lei[1], WANG Feng[2], KONG Ren[1], CHANG Shan[1]

(1. Institute of Bioinformatics and Medical Engineering, School of Electrical and Information Engineering, Jiangsu University of Technology, Changzhou, 213001, China; 2. School of Information Science and Engineering, Changzhou University, Changzhou, 213164, China)

Abstract: Because of the excessive protein species and number of small molecules, the computational complexity of medicine simulation development is enormous. Molecular docking is an important method to study new medicine, so it is very important to improve the efficiency of molecular docking experimental system. The main purpose of molecular docking is to research the interaction and relationship between protein receptors and ligand small molecules. We set up a Hadoop platform and use Hadoop's powerful parallel computing ability to dock proteins (1ppe and 1uy6) with a number of small ligands with different numbers through simulation experiments. The corresponding adjustment and optimization of the working process are also carried out. Experimental results show that the system can effectively improve the docking efficiency, and has good stability and convenience.

Key words: protein; molecular docking; Hadoop enviroment; parallel computing

---

基金项目：国家自然科学基金(81603152，81803430)资助项目；江苏省六大人才高峰(2016-XYDXXJS-020)资助项目；江苏省产学研前瞻项目(BY2016030-06)资助项目；常州市应用基础研究计划(CJ20160016)资助项目；江苏省研究生科研与实践创新计划(SJCX18_1041)资助项目。

收稿日期：2019-06-24；**修订日期**：2019-09-04

引　言

随着这几年信息技术的迅速发展,深度学习、大数据和云计算等技术越来越多地被各行各业所使用,现代社会对大量数据的依赖日渐增长,高速数据处理已成为如今十分重要的部分。一般情况下,在生物医药领域中的生物学实验是探究分子之间相互作用的直接手段。但通过生物实验直接研究蛋白质的相互作用并不是一件简单的事情,因为随着生物技术的不断发展,进行蛋白质复合物晶体的实验需要严格且复杂的条件与步骤,并且实验所需的蛋白质获取难度较大,实验成本非常昂贵,于是运用计算机来运行理论模拟得到了越来越多的重视,并且被投入实际的应用中[1-2]。

计算机辅助药物设计的方法有许多,分子对接也是其中之一。随着药物化学、生物信息学和计算机技术的迅速发展,越来越多的小分子化合物和靶标蛋白在不断地被发现,生物信息数据呈现爆炸式增长。因此在新药物研发中,分子对接实验将遭遇海量数据存储与大规模数据计算的多重挑战[3]。虽然在处理单个或者指定的生物分子时,使用传统的方法也就是在Windows系统中利用中央处理器(Central processing unit, CPU)的计算能力通过对接软件进行对接并没有什么问题,但在处理的数据量较大时,仅仅依赖CPU的工作可能会让系统捉襟见肘。这时在没有图形处理器(Graphics processing unit, GPU)等硬件的情况下,通过搭建Hadoop平台,利用Hadoop的并行计算能力来调动多个节点进行批量蛋白质分子的对接不妨是一种可行的方法。

在蛋白质受体与配体小分子对接的过程中,需要考虑两个方面:一是需要搜索能量低的构象,二是在尽量短的时间内在设定的对接范围中找到各种可能的对接情况。在实际应用中,因为随着对药物和疾病的不断研究,很多时候一个蛋白质受体需要连续和上万个小分子进行对接,同时还要对对接完的结果进行筛选和整理,因此在整个系统工作运行过程中,既要保证Hadoop的正常运行,同时也要保证工作的效率。本文首先搭建了一个Hadoop平台,通过Hadoop调用分子对接软件Autodock Vina进行蛋白质与小分子的分子对接,然后进行相应的优化与分析,最后进行了总结与展望。

1　Hadoop计算框架

Hadoop是由Google公司提出的一个软件架构,可以用来处理大规模数据。作为如今十分流行的云计算平台,Hadoop擅长大数据和分布式并行计算,可以在蛋白质与小分子对接中发挥一定的作用。通用并行计算模型MapReduce和Hadoop分布式文件系统(Hadoop distributed file system, HDFS)组成了Hadoop的主要架构,其详细的工作示意图如图1所示。

Main端的主要任务包括定义配体受体文件的存放位置、确定对接结果的存放位置、确定读取文件的文件格式、定义输出key值和value值的类型等。

Map端的任务是将输入路径下的作业也就是每个分子对接的任务分发成更加小的任务来执行,以HDFS存储文件的最小单位为标准来分发,一个map函数处理一个任务,其中map函数会一次性读取多个数据并进行并行处理[4]。

Reduce端将Map端输出的数据进行收集与合并。其中,在进行分子对接处理时,一次对接完成后系统会产生一个对应的文件,并存储在HDFS中。

在分子对接的整个过程中,遵循的基本原理是“锁钥原理”。整个分子对接的过程就是找到蛋白质与一个或多个小分子之间最佳结合模式的过程。它们主要通过能量匹配和几何匹配来进行相互识别[5]。系统会通过计算预测出配体和受体间结合模式的好坏,并以亲和力的数值形式让实验者判断对接的好坏[6-7]。

利用MapReduce的计算框架,通过把作业的调度和分发写入程序中,整个框架就会通过Jobtracker调用各个节点上的tasktracker对数据进行并行对接,对接完成后Reduce函数会自动收集每个节点的对接结果[8-10]。

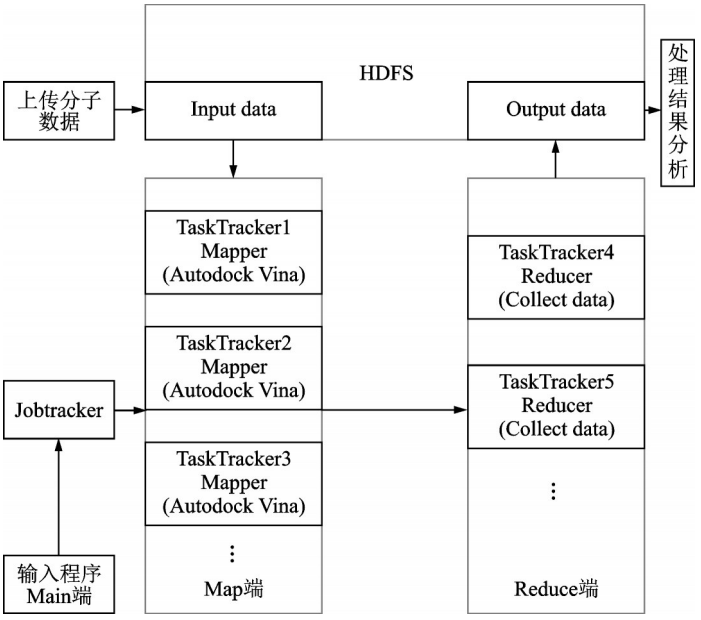


图1 Hadoop工作示意图

Fig.1 Hadoop working diagram

2 工作原理及方法

整个系统的工作流程如图2所示。

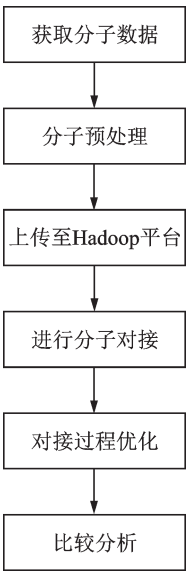


图2 系统流程图

Fig.2 System flowchart

步骤1 获取分子数据

以受体为例，蛋白质受体可以从分子数据库(Protein data bank，PDB)中下载所对应的PDB分子。以配体为例，ZINC数据库是一个较大的生物分子网站，可以从中下载需要的配体小分子。

步骤2 分子数据预处理

在ZINC上下载的配体小分子大多是mol2格式或者sdf格式，不可以用来进行分子对接，因此需要使用OpenBabel软件，将配体小分子批量转换成可以对接的pdbqt格式；蛋白质受体同样需要进行预处理，通过AutoDock Tools软件(分子对接可视插件)，进行删除受体上的水分子、删除蛋白质受体上原有的小分子、添加氢键和选定对接区域等操作，这样做的目的是规范蛋白质受体的格式，从而减轻分子对接时系统的工作负担。这些操作完成后，最后将处理后的受体转化成pdbqt格式。

步骤3 上传至Hadoop平台

在Linux系统中开启Hadoop平台，开启免SSH(Secure shell)登录，开启HBase数据库。待Hadoop启动完成后，将配体分子、受体分子和对接参数文档上传至HDFS，可以通过Eclipse查看上传的各项数据。一切准备就绪后，进行后续操作。

步骤4 批量进行分子对接

Autodock Vina是一款开源软件，由Molecular Graphics实验室开发与管理，它主要是基于拉马克遗

传算法来进行分子对接,具有多核运行、兼容性强和准确性高的特点[11]。使用过程中需要准备好分子对接所需要的配体分子、受体分子和相应的参数,就可以进行对接工作。

要想访问HDFS上的文件,可以使用Xshell或者Eclipse等软件工具,然而HDFS是分布式系统,也就是说分子对接软件Autodock Vina无法直接读取HDFS上的分子数据。因此,如果想操作HDFS文件系统就像操作本地文件系统一样的便捷,实现两者之间的映射是能想到的最好解决方案。

大多数文献大多使用用户空间文件系统挂载协议(File system in user space-distributed file system,FUSE-DFS)和WebDEV(HTTP协议扩展)两种途径来实现HDFS的映射。然而相应的操作配置繁琐,而且随着版本的更新,在Hadoop2.0以后的版本早已取消了FUSE和WebDEV的文件配置。经查阅Hadoop的使用指南,可以使用Hadoop的NFS3插件将HDFS作为一个远程服务器挂载在本地,通过NFS3可以像在Linux本地一样操作HDFS的文件,从而省去许多操作。

步骤5 对接过程优化

对接结束后,一个小分子配体会产生一个相应的文件,其中包含相应的对接数据,分子对接后的数据存储在HDFS中。这时再根据Hadoop自身的工作状况以及任务的分配状况对系统进行优化,主要从节点的分配数量上进行考虑,以提高工作效率。

步骤6 比较分析

将优化后系统的工作状况与原结果进行比较与分析,提出自己的见解。

3　详细测试实例与结果分析

3.1　Hadoop环境配置

Hadoop的环境配置较为繁琐,具体如表1所示。

表1　Hadoop环境配置文件

Table 1　Basic profile of Hadoop

文件名称	格式	功能
hadoop-env.sh	Bash脚本	记录Hadoop所需的环境变量
core-site.xml	Hadoop的xml文件	HDFS和MapReduce常用设置
hdfs-site.xml	Hadoop的xml文件	HDFS工作进程配置项
yarn-site.xml	Hadoop的xml文件	Yarn工作进程配置项
mapred-site.xml	Hadoop的xml文件	MapReduce计算框架配置项
slaves	文本文件	配置各个节点的名称
yarn-env.sh	Bash脚本	配置YARN的管理路径

3.2　系统硬件配置

基于实验室3台高性能电脑安装虚拟机搭建了一个内部局域网的Hadoop平台,详细参数如表2所示。其中电脑1搭配i5-7400型号的4核CPU,分配16 GB运行内存,电脑2和电脑3搭配i5-6500型号的4核CPU,分配8 GB内存。

需要说明的是,Master和Slave 1～3共4台虚拟机安装在电脑1中,Slave 4～6共3台虚拟机安装在电脑2中,Slave 7～9共3台虚拟机安装在电脑3中。其中一台虚拟机代表一个节点,每个工作节点均分配1个CPU核心。

表2　硬件参数

Table 2　Harware parameters

电脑配置	参数
CPU	Intel i5-7400一台,i5-6500两台
内存	一台8 GB×2=16 GB,两台8 GB
操作系统	Centos7.0

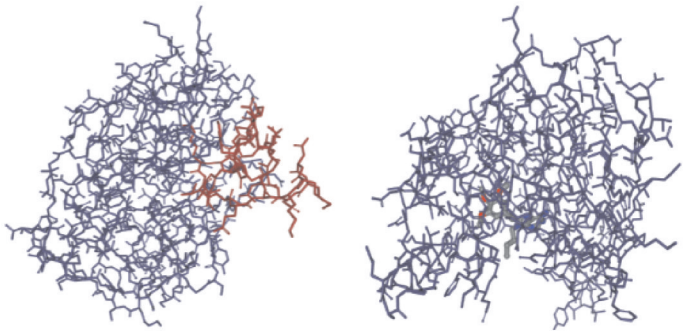
表3 虚拟机节点环境配置

Table 3 Node configuration in virtual system

名称	Ip地址	分配内存/GB	运行进程
Master	192.168.92.128	4	NameNode, TaskTracker
Slave 1	192.168.92.130	4	DataNode, TaskTracker
Slave 2	192.168.92.132	4	DataNode, TaskTracker
Slave 3	192.168.92.134	4	DataNode, TaskTracker
Slave 4～9	192.168.92.136～146	2	DataNode, TaskTracker

3.3 测试过程与结果

整个测试共进行了3种类型的实验,其中第1组实验直接在电脑1的Linux系统中进行运作,第2组实验使用Master和Slave 1～3节点进行工作,也就是第1台电脑的Hadoop系统中运作,其目的是在保证在CPU核心以及分配内存相同的情况下,比较分子对接在Hadoop平台与Linux系统中的工作效率。第3组实验采用Hadoop的全部节点也就是全部10个节点进行对接,目的是为了从硬件的角度对Hadoop的工作效率进行分析测试,在增加Hadoop的工作节点以及硬件的条件下,判断Hadoop的工作效率是否有所提高。本次测试采用代号为1ppe和1uy6的蛋白质作为受体[12]。图3显示了用于对接的蛋白受体的3D结构图。因为每个小分子的数据结构都不同,对接的时间也不尽相同,保险起见,所以每组测试分别使用1,2,5,10,50,100个小分子作为配体进行批量分子对接,对接区域x,y,z轴全部设为80。



(a) 1ppe (b) 1uy6

图3 蛋白质受体结构图

Fig.3 Structure of the protein receptor

通过Hadoop的网页监测可以实时直观地观察整个Hadoop的工作情况,图4为部分Hadoop节点工作时,网页负责监视Hadoop工作状况的界面。如图4所示,通过网页可以直观观察HDFS的存储情况,包括使用状况、各个节点的工作情况、节点分配的任务和是否有任务出现错误等状况。

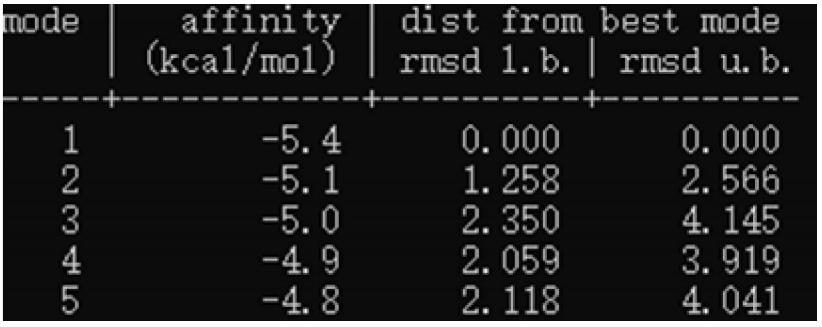
对接结束后,一个小分子配体会产生一个相应的文件,其中包含相应的对接数据,图5显示了一组配体对接完成后形成的数据结果。其中一个蛋白质可以有多个与小分子配体对接的位置,如何判断每个对接的好坏,系统会按照affinity(亲和力)由小到大进行排序,mode代表每个配体在蛋白质中对接的位置,一般情况下,亲和力越低,则表示分子对接的结果越稳定,计算机辅助药物的模拟效果就越好。第2列数据是其他对接部位与第1个的距离,第3列数据是其他对接部位和第1个的差异情况。

Configured Capacity:	140.91 GB
DFS Used:	107.52 MB (0.07%)
Non DFS Used:	35.73 GB
DFS Remaining:	105.08 GB (74.57%)
Block Pool Used:	107.52 MB (0.07%)
DataNodes usages% (Min/Median/Max/stdDev):	0.07% / 0.07% / 0.07% / 0.00%
Live Nodes	3 (Decommissioned: 0)
Dead Nodes	0 (Decommissioned: 0)
Decommissioning Nodes	0
Total Datanode Volume Failures	0 (0 B)
Number of Under-Replicated Blocks	0
Number of Blocks Pending Deletion	0
Block Deletion Start Time	2019/6/2 下午6:45:22

Node	Last contact	Admin State	Capacity	Used	Non DFS Used	Remaining	Blocks	Block pool used	Failed Volumes	Version
slave2:50010 (192.168.92.132:50010)	2	In Service	46.97 GB	35.59 MB	9.28 GB	37.66 GB	85	35.59 MB (0.07%)	0	2.7.1
slave1:50010 (192.168.92.130:50010)	0	In Service	46.97 GB	35.59 MB	9.28 GB	37.65 GB	85	35.59 MB (0.07%)	0	2.7.1
master:50010 (192.168.92.128:50010)	1	In Service	46.97 GB	35.59 MB	9.67 GB	37.27 GB	85	35.59 MB (0.07%)	0	2.7.1

图4　Hadoop网页监测图

Fig.4　Webpage monitoring chart of Hadoop



mode	affinity (kcal/mol)	dist from best mode rmsd l.b.	rmsd u.b.
1	-5.4	0.000	0.000
2	-5.1	1.258	2.566
3	-5.0	2.350	4.145
4	-4.9	2.059	3.919
5	-4.8	2.118	4.041

图5　部分小分子对接结果数据图

Fig.5　Data diagram of partial docking results

表4和表5详细记录了在不同环境下,蛋白质对接的工作时间。由表4,5数据可以说明,Hadoop在进行分子对接时,如果分子数量较少,从工作的时间来看并不具备任何的优势,甚至比在Linux系统中

表4　蛋白质受体1ppe在不同环境下对接工作时间

Table 4　Docking time of protein receptor 1ppe in different systems　s

小分子配体数量	Linux系统	Hadoop部分节点	Hadoop全部节点
1	16	30	27
2	39	41	35
5	166	164	151
10	289	267	249
50	1 420	1 132	1 021
100	2 913	2 541	2 103

表5　蛋白质受体1uy6在不同环境下对接工作时间

Table 5　Docking time of protein receptor 1uy6 in different systems　s

小分子配体数量	Linux系统	Hadoop部分节点	Hadoop全部节点
1	17	28	29
2	38	36	38
5	130	105	103
10	248	214	202
50	1 196	995	897
100	2 604	2 195	1 841

工作还慢;然而随着分子数量的增多,Hadoop的工作效率显著提高,尤其是在进行50个以上小分子对接的时候,从工作时间上可以明显看出Hadoop的工作效率更高;其中在处理一定量的数据时,Hadoop部署的节点越多,系统的运行速度越快,这是因为只有在处理大量数据时才可以真正发挥Hadoop并行计算的特点。

3.4 系统优化

在Linux系统执行分子对接时,整个程序默认是按顺序串行工作的,也就是前一个分子对接结束并且得出结果后再进行下一个分子对接工作。然而在Hadoop系统中,分子对接是多个任务同时运行的,图6和图7分别展示了不同系统中分子对接时的过程图。需要说明的是在Hadoop系统中,有的任务不会到达100%,而有的会超过100%,这时因为Hadoop的工作原理,系统会通过框架将任务分配给每个节点,在每个节点任务完成后,最后系统整合在一起完成整个任务。

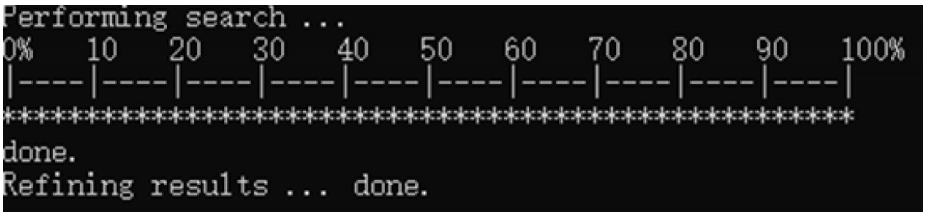


图6 Liunx系统分子对接过程图

Fig.6 Docking process in Liunx system

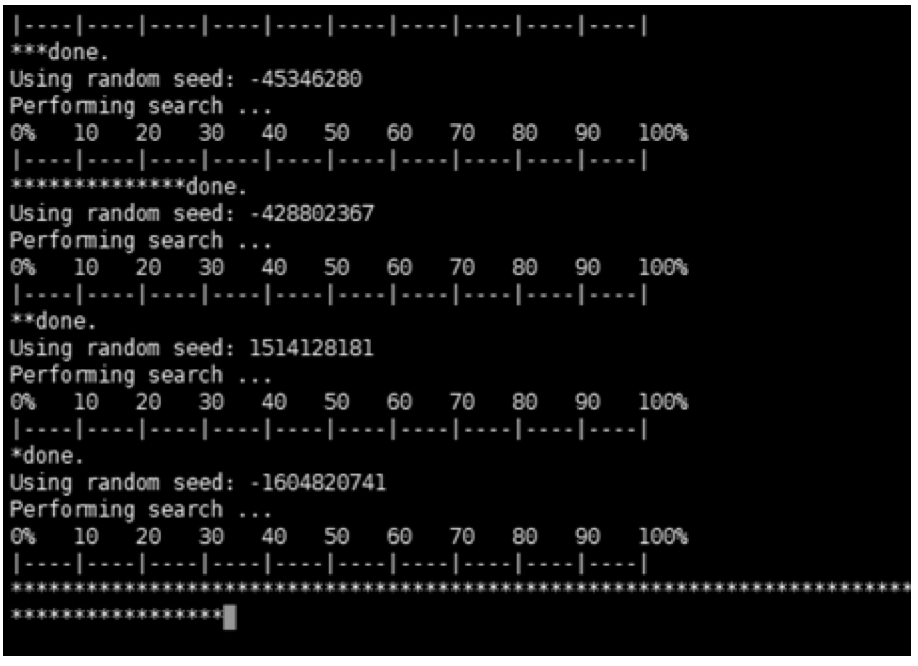


图7 Hadoop系统分子对接过程图

Fig.7 Docking process in Hadoop system

针对上述情况考虑,如何在不提高系统硬件配置的情况下,增强Hadoop的工作性能,成为十分关键的部分。系统工作时,Hadoop框架会执行定义良好的处理阶段序列中的作业[13-14],其中处理时间则

主要依赖于每个阶段流经过的数据以及底层Hadoop集群的性能。因此在保证系统条件不变的情况下,根据硬件和任务的大小合理配置集群会显著提高系统的工作效率[15]。

通过网页观察Hadoop工作状况时,发现在运行少量的分子对接时,Hadoop的部分节点利用率较低,造成了资源的闲置。同时在运行较多的分子对接时,受限于磁盘读写的状况,部分节点不能迅速加载数据。针对上述情况,决定在接下来的几组实验中调整工作节点的数量以及节点的硬件配置。

3.5　后续比较与分析

为评估调整后的效果,又分别进行了4组实验,前两组实验把1,2,5,10个小分子配体分别与1ppe,1uy6两个蛋白质受体进行批量分子对接,实验使用Master和Slave1～2共3个节点进行工作。第3组和第4组实验把5,10,50,100个小分子配体分别与1ppe,1uy6两个蛋白质受体进行批量分子对接,其中实验使用Master和Slave1～6共7个节点进行工作,多节点并行的工作时间如图8,9所示。

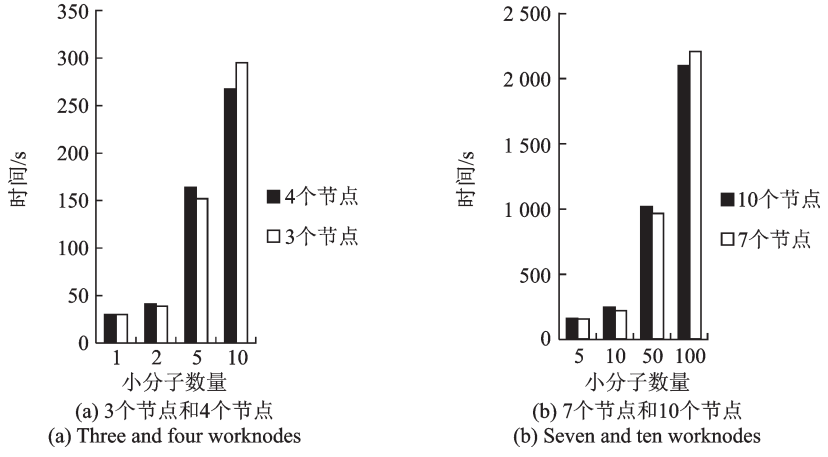


图8　1ppe多节点并行的工作时间比较图

Fig.8　Working time comparison of 1ppe at parallel multi worknodes

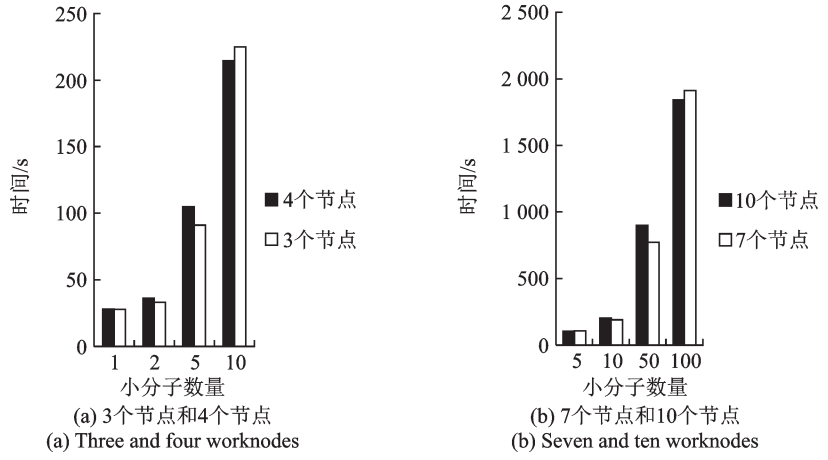


图9　1uy6多节点并行的工作时间比较图

Fig.9　Working time comparison of 1uy6 at parallel multi worknodes

如图8,9所示,在运行单个小分子对接时,不同系统工作时间差距不大,然而在进行2个和5个小分子对接时,配置3个节点的系统工作时间均小于4个节点的系统,在运行10个小分子的对接时,配置3个节点的系统工作时间又大于4个节点的系统。同样的,配置7个节点的系统在运行10个和50个小分子对接时工作时间要小于10个节点的系统,在运行到100个小分子对接时,10个节点才快于7个节点。

从上述测试可以看出,并不是分配越多节点,系统工作效率就越快。根据处理数据的大小合理分配节点可以有效提高系统的工作效率。Hadoop又具有特有的网页检测机制,可以通过网页实时观察节点的工作状况,在工作时要根据任务的大小或者通过网页监测适当调整Hadoop中工作节点的数量,调整各个节点的硬件配置,这样才可提高工作效率。

4 结束语

实验结果表明,基于Hadoop的蛋白质配体分子对接计算系统能够充分发挥并行计算的优势,有效缓解工作耗时过长的问题,从而提高了工作效率。该系统具有容错性低、扩展性高和操作简单等特点,而且通过网页可以直观实时监测Hadoop的工作状况,对于学习相关专业的大学学生或者研究员有一定的参考价值。同时Hadoop相比较一些其他一些大数据处理系统,在设备上对硬件的要求并不大,在各大高校或者实验室有大量性能一般的计算机的情况下,搭建一个Hadoop平台用于批量分子对接或者分子的虚拟筛选,不妨是一个高效且合理的途径。

本实验仍还有研究不足的地方,撇开节点分配的因素,Hadoop的工作效率还受限于系统自身的硬件配置,包括CPU性能、内存大小、磁盘读写状况和网络带宽等因素。如果还需更进一步的优化,可以考虑从监测系统的薄弱环节做起,例如检查CPU和内存是否工作饱和、数据传输是否过大引起网络阻塞和存储的数据是否合理分配等。

参考文献:

[1] 陆旭峰,陆振宇,梅向东,等. 基于CPU和GPU异构的蛋白质分子半柔性对接算法优化[J]. 数据采集与处理,2018,33(4): 603-610.
LU Xufeng, LU Zhenyu, MEI Xiangdong, et al. Optimization of semi flexible docking algorithm for protein molecules based on CPU and GPU heterogeneous[J]. Journal of Data Acquisition and Processing, 2018, 33(4): 603-610.

[2] RITCHIE D W. Recent progress and future directions in protein-protein docking[J]. Current Protein & Peptide Science, 2008, 9(1): 1-15.

[3] 李杰辉,张亮,陈健,等. 基于Hadoop的化合物生物活性分析系统[J]. 计算机工程, 2012, 38(13): 48-50.
LI Jiehui, ZHANG Liang, CHEN Jian, et al. Compounds biological active analysis system based on Hadoop[J]. Computer Engineering, 2012, 38(13): 48-50.

[4] 张丽. 基于云平台的分子对接设计与实现[D]. 成都:电子科技大学,2015.
ZHANG Li. Molecular docking based on optimal search theory research[D]. Chengdu: University of Electronic Science and Technology of China, 2015.

[5] 常珊,陆旭峰,王峰. 蛋白质-配体分子对接中构象搜索方法[J]. 数据采集与处理, 2018, 33(4): 586-594.
CHANG Shan, LU Xufeng, WANG Feng. Review of conformational searching method for protein-ligand molecular docking[J]. Journal of Data Acquisition and Processing, 2018, 33(4): 586-594.

[6] 张影. 大规模虚拟筛选对接结果的分析与研究[D]. 兰州:兰州大学,2012.
ZHANG Ying. Research and analysis of large-scale virtual screening docking results[D]. Lanzhou: Lanzhou University, 2012.

[7] 陈殿伟. 基于Hadoop的虚拟筛选海量数据存储及结果处理的设计和实现[D]. 兰州:兰州大学,2012.
CHEN Dianwei. Design and implementation of mass data storage and result reduction for virtual screening based on Hadoop[D]. Lanzhou: Lanzhou University, 2012.

[8] BONVIN A M. Flexible protein-protein docking[J]. Current Opinion in Structural Biology, 2006, 16 (2): 194-200.

[9] 李竞蔚.基于Hadoop的虚拟筛选数据管理和并行对接研究[D].兰州:兰州大学,2015.
LI Jingwei. Design and implementation of mass data storage and result reduction for virtual screening based on Hadoop[D]. Lanzhou: Lanzhou University,2015.
[10] 刘广才.基于Hadoop的大规模虚拟筛选数据的分析研究[D].兰州:兰州大学,2014.
LIU Guangcai. Analytical studies of large-scale virtual screening data based on Hadoop[D]. Lanzhou: Lanzhou University,2014.
[11] 李杰辉.基于云计算技术的化合物相似性分析系统[D].上海:复旦大学,2012.
LI Jiehui. Compound similarity analysis system based on cloud computing technology[D]. Shanghai: Fudan University,2012.
[12] VAKSER I A. Protein-protein docking: From interaction to interactome[J]. Biophysical Journal, 2014, 107(8): 1785-1793.
[13] SRINATH P, THILINA G. Hadoop MapReduce Cookbook[M]. [S.l.]: Posts&Telecom Press,2015: 36-39.
[14] KHALED T. Optimizing Hadoop for MapReduce[M]. [S.l.]: Posts&Telecom Press, 2015: 4-7.
[15] 顾荣,严金双,杨晓亮,等. Hadoop MapReduce短作业执行性能优化[J].计算机研究与发展, 2014,51(6): 1270-1280.
GU Rong, YAN Jinshuang, YANG Xiaoliang, et al. Performance optimization for short job execution in Hadoop MapReduce [J]. Journal of Computer Research and Development,2014,51(6): 1270-1280.

作者简介:

钦淳(1992-),男,硕士研究生,研究方向:人工智能、云计算、智能家居,E-mail: 309286416@qq.com。

许磊(1983-),男,副教授,研究方向:药物设计理论。

王峰(1983-),男,高级工程师,研究方向:智能算法和生物信息学。

孔韧(1981-),女,教授,研究方向:计算机辅助药物设计方法研究。

常珊(1982-),**通信作者**,男,教授,研究方向:生物信息学和并行计算,E-mail: schang@jsut.edu.cn。

(编辑:王静)