Hybrid Cloud and Cluster Computing Paradigms for Life Science Applications

Judy Qiu[1,2], Thilina Gunarathne[1,2], Jaliya Ekanayake[1,2], Jong Youl Choi[1,2], Seung-Hee Bae[1,2],
Hui Li[1,2], Bingjing Zhang[1,2], Yang Ryan[1,2], Saliya Ekanayake[1,2], Tak-Lon Wu[1,2],
Scott Beason[2], Adam Hughes[2], Geoffrey Fox[1,2]

[1]School of Informatics and Computing, [2]Pervasive Technology Institute
Indiana University, Bloomington.
{xqiu, tgunarat, jekanaya, jychoi, sebae, lihui, zhangbj, yangruan, sekanaya, taklwu, smbeason, adalhugh, gcf}@indiana.edu

SALSA project http://salsahpc.indiana.edu/
Twister software and license http://www.iterativemapreduce.org/ and http://www.iterativemapreduce.org/license.html

Cloud computing [1] offers new approaches for scientific computing that leverage the major commercial hardware and software investment in this area. Closely coupled applications are still unclear in clouds as synchronization costs are still higher than on optimized MPI machines. However loosely coupled problems are very important in many fields and can achieve good cloud performance even when pleasingly parallel steps are followed by reduction operations as supported by MapReduce. However we can use clouds in several ways and we have compared five different approaches using two biomedical applications. We look at the cloud infrastructure service based virtual machine utility computing models of Amazon AWS and Microsoft Windows Azure; MapReduce based computing frameworks Apache Hadoop (deployed on raw hardware as well as on virtual machines) and Microsoft DryadLINQ. We compare performance showing strong variations in cost between different EC2 machine choices and comparable performance between the utility computing (spawn off a set of jobs) and managed parallelism (MapReduce). Our main emphasis is Cloud techniques as provider comparison is very subject to changes. The MapReduce approach offered the most user friendly approach. Typical results [2] are shown in Fig. 1.

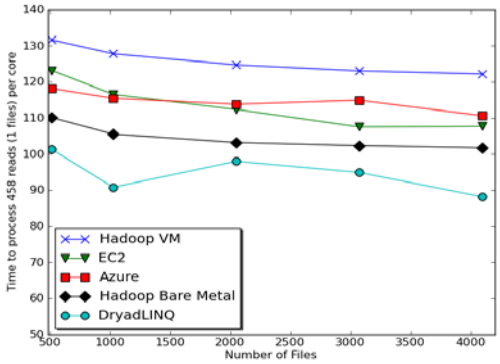

Figure 1 Time to process a single biology sequence file (458 reads) per core with different frameworks

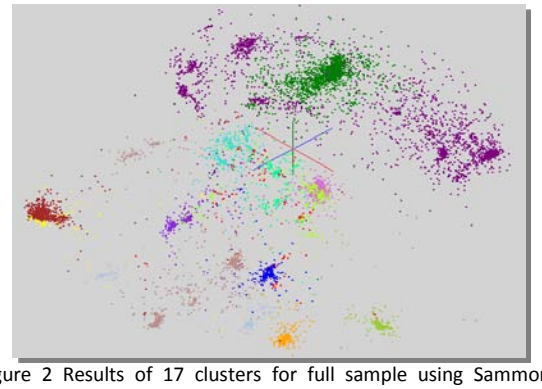

Figure 2 Results of 17 clusters for full sample using Sammon's version of MDS for visualization.

A typical bioinformatics pipeline of Smith-Waterman distance Computation, Deterministic Annealing Clustering and MDS visualization is shown below in Fig. 3, which can give results such as Fig. 2 where the results of 30,000 Metagenomics sequences in 3D are shown. The visualization uses dimension reduction where we have implemented two powerful methods GTM (Generative Topographic Mapping) and MDS (Multidimensional Scaling) [3] [4].

Only MDS can be used for DNA sequence visualization as GTM requires a vector representation of original high dimensional data whereas MDS only requires the N by N matrix of dissimilarity scores between sequences. Multiple Sequence Alignment needed to obtain a uniform vector representation of sequences is typically infeasible. The distance matrix calculation needed by MDS is very suitable for cloud implementation as the computations are independent. However both clustering and MDS require parallel implementation as they are expensive $O(N^2)$ computations; the run time of these on a 768 core cluster is about 3 hours for 30,000 sequences

with a speed up of 500. These parallel implementation run poorly on clouds or MapReduce as their iterative algorithms require the long running processes and low latency of MPI. Thus we see hybrid cluster-cloud architectures as needed for this class of problem where a complete workflow is gotten by linking separate services in clouds and closely coupled clusters.

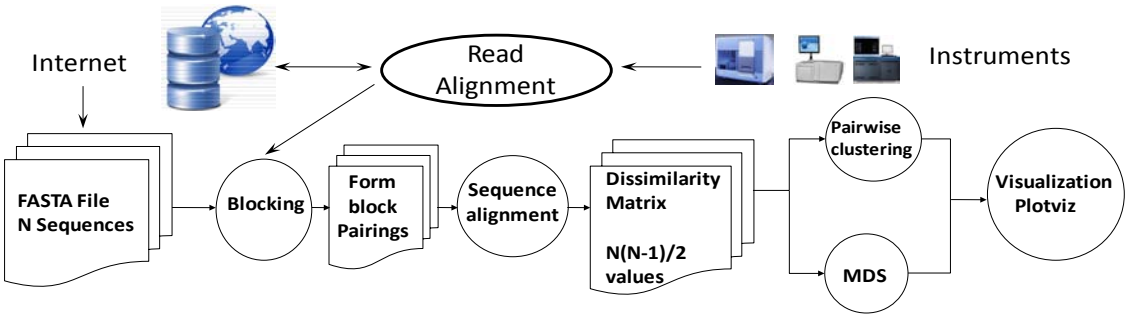

Figure 3 Pipeline for analysis of metagenomics Data

We have developed new interpolation algorithms for both MDS and GTM which can exploit clouds and MapReduce for the dominant part of the computation for large problems. These perform a basic dimension reduction for a sample of the data (20,000-100,000 points) which runs using MPI on a cluster; the remaining points are interpolated which is a pleasingly parallel cloud application. We will present performance results for run time and quality of dimension reduction.

Alternatively we have extended MapReduce in an open source system, Twister [5] [6], that supports iterative computations of the type needed in clustering, MDS and GTM. This programming paradigm is attractive as it supports all phases of the pipeline in Fig. 1. We present performance comparisons between MPI, MapReduce and Twister on kernel applications such as matrix multiplication as well as the core services of Fig. 1.

References

[1] Jaliya Ekanayake, Xiaohong Qiu, Thilina Gunarathne, Scott Beason, Geoffrey Fox High Performance Parallel Computing with Clouds and Cloud Technologies to appear as a book chapter to Cloud Computing and Software Services: Theory and Techniques, CRC Press (Taylor and Francis), ISBN-10: 1439803153.

[2] Thilina Gunarathne, Tak-Lon Wu, Judy Qiu, and Geoffrey Fox, Cloud Computing Paradigms for Pleasingly Parallel Biomedical Applications, Proceedings of Emerging Computational Methods for the Life Sciences Workshop of ACM HPDC 2010 conference, Chicago, Illinois, June 20-25, 2010.

[3] Seung-Hee Bae, Jong Youl Choi, Judy Qiu, Geoffrey Fox Dimension Reduction and Visualization of Large High-dimensional Data via Interpolation, Proceedings of ACM HPDC 2010 conference, Chicago, Illinois, June 20-25, 2010.

[4] Jong Youl Choi, Seung-Hee Bae, Judy Qiu, Geoffrey Fox, Bin Chen, and David Wild, Browsing Large Scale Cheminformatics Data with Dimension Reduction, Proceedings of Emerging Computational Methods for the Life Sciences Workshop of ACM HPDC 2010 conference, Chicago, Illinois, June 20-25, 2010.

[5] Jaliya Ekanayake, Hui Li, Bingjing Zhang, Thilina Gunarathne, Seung-Hee Bae, Judy Qiu, Geoffrey Fox Twister: A Runtime for Iterative MapReduce, Proceedings of the First International Workshop on MapReduce and its Applications of ACM HPDC 2010 conference, Chicago, Illinois, June 20-25, 2010.

[6] Twister software at http://www.iterativemapreduce.org/