BIOINFORMATICS APPLICATIONS NOTE

Vol. 19 no. 3 2003, pages 441–442
DOI: 10.1093/bioinformatics/btg007

Wrapping up BLAST and other applications for use on Unix clusters

Karsten Hokamp[1,*], Denis C. Shields[2], Kenneth H. Wolfe[1] and Daniel R. Caffrey[2,*,†]

[1]Department of Genetics, University of Dublin, Trinity College, Dublin 2, Ireland and [2]Department of Clinical Pharmacology, Royal College of Surgeons in Ireland, 123 Stephen's Green, Dublin 2, Ireland

Received on May 28, 2002; revised on July 18, 2002; accepted on June 6, 2002

ABSTRACT

Summary: We have developed two programs that speed up common bioinformatic applications by spreading them across a UNIX cluster.(1) BLAST.pm, a new module for the 'MOLLUSC' package. (2) WRAPID, a simple tool for parallelizing large numbers of small instances of programs such as BLAST, FASTA and CLUSTALW.

Availability: The packages were developed in Perl on a 20-node Linux cluster and are provided together with a configuration script and documentation. They can be freely downloaded from http://wolfe.gen.tcd.ie/wrapper.

Contact: daniel_r_caffrey@cambridge.pfizer.com; karsten@oscar.gen.tcd.ie

INTRODUCTION

Inexpensive systems such as Beowulf clusters, have become increasingly popular in both the commercial and academic sectors of the bioinformatics community. Clusters typically consist of a master machine/node that distributes the bioinformatic application amongst the other nodes (slaves/clients). These often require installation of special software on each node or modification of the bioinformatics programs. A simpler solution consists of so-called wrappers like MOLLUSC (Jongeneel *et al.*, 1997), that improve search times for SSEARCH (Pearson, 2000), pfscan, and pfsearch (Luthy *et al.*, 1994) This involves the databases being split up into smaller portions that are distributed amongst the clients. On each node, a portion of the database is searched and the master merges the results into a single file. MOLLUSC allows incorporation of modules for other search programs. Here we describe: (1) BLAST.pm, a module that allows MOLLUSC to run BLAST (Altschul *et al.*, 1997) on a UNIX cluster. (2) WRAPID, an independent tool designed for processing large numbers of small applications.

SYSTEMS AND METHODS

BLAST.pm

The success of the BLAST package is partially due to its short search times, relative to programs such as FASTA or SSEARCH (Brenner *et al.*, 1998). However, under heavy load, parallelization would further improve its performance. Also, memory should be big enough to store the entire database or be split into smaller volumes. Thus, dividing a large database into smaller portions that can be individually searched with BLAST on multiple nodes would be advantageous. However, BLAST E-values are dependent on database size, query size, and letter composition. Specifically, the E-values are calculated from an effective search space. This is the product of the effective lengths of the query and the database. Figure 1 shows that the effective search space has a near-linear relationship with the search space. Using these linear regression equations, the BLAST.pm package estimates the effective search space and forces this value using the '-Y' option of BLAST for each client. The use of the estimated effective search space yields similar results. Using a standard set of sequences, we have found that E-values will be almost identical to those produced when searching against the entire, unpartitioned database (see documentation). The BLAST.pm Perl module is called by the MOLLUSC package through a command-line that closely resembles BLAST

```
mollusc blastall -p blastp -a 4 -d nr.aa -i
/fullpath/query -o /fullpath/result -T
```

The majority of command line options are allowed, but certain options are either forced or removed to facilitate formatting or generation of statistics (see documentation).

WRAPID

Where the whole process of a bioinformatics application fits into the memory of each client, a small tool called WRAPID (Wrapper for RApid Parallelized Instruction Dispatching) can be used. It speeds up large numbers of small jobs, such as comparing all protein sequences of

[*]These authors contributed equally to this work.

[†]Current Address: Pfizer Discovery Technology Center, 620 Memorial Drive, Cambridge, MA 02139, USA.

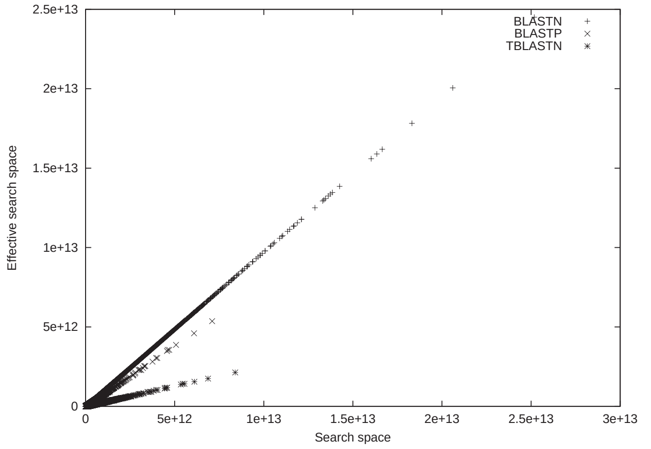

Fig. 1. The linear relationship between the search space (x-axis) and effective search space (y-axis). Using the ENSEMBL human genome peptide and nucleotide databases, an all-against-all BLAST search was done for BLASTP, BLASTN, TBLASTN, BLASTX, and TBLASTX. Each sequence was searched against an entire database that was concatenated to itself (ranging from 5 to 140 copies). For each concatenated database that was searched with a given query, the effective search space and search space were extracted. These were then plotted for each program and the equation of the line was calculated using linear regression the BLASTX and TBLASTX plots are not shown, as they are almost identical to the BLASTP and TBLASTN plots, respectively. All correlation coefficients were between 0.996 and 1. The regression coefficients for this search space are $\beta_0 = -1.53\text{e}{+}10$, $\beta_1 = 0.74$ (BLASTP), $\beta_0 = -1.29\text{e}{+}10$, $\beta_1 = 0.97$ (BLASTN), $\beta_0 = -8.57\text{e}{+}9$, $\beta_1 = 0.25$ (TBLASTN), $\beta_0 = -1.63\text{e}{+}10$, $\beta_1 = 0.74$ (BLASTX), $\beta_0 = -9.48\text{e}{+}9$, $\beta_1 = 0.25$ (TBLASTX).

an organism against one another or aligning a batch of sequences, through parallelization on a cluster. The installation effort was kept at a minimum and the application range as broad as possible. The script was written in Perl and only needs to be run on the initiating node—no installation is necessary on the other computers. The only client requirements comprise remote login, a shared directory holding clients' input files and results, and the availability of Perl. Once this is given, many common bioinformatics programs can be easily parallelized through WRAPID. Its usage involves a simple prepending to a valid execution statement, e.g.

```
wrapid.pl ssearch -Q -b 500 -d 0 -H -m 9 -p -S -E 1
/fullpath/query /fullpath/database
```

```
wrapid.pl clustalw /fullpath/file_of_filenames
```

Additional command line options or configuration files can be used to adjust the process to different set ups, changing work loads and varying requirements of different applications. So far the wrapper has been successfully tested with BLAST and FASTA, as well as with MPSRCH and ClustalW. WRAPID also provides some extra features, e.g. comprehensive checks for prerequisites are carried out on each node, dynamic assignment of nodes, a load balancing mechanism, and job completion is checked.

CONCLUSION

Parallelization of bioinformatic jobs offers improved performance and scalability. Although communication overheads are likely to affect performance as the cluster becomes larger, we observe up to 19-fold improvement in search times on a 20 node cluster (see documentation). When the database is too big for a single node to search, BLAST should be invoked through BLAST.pm. In cases where a single user has many queries and the entire database can be searched efficiently by a single node, WRAPID should be used. WRAPID can also be viewed as a batch queueing system for many applications, but differs from BEOBLAST (Grant *et al.*, 2002) in that it is best suited for the single user with multiple queries. WRAPID differs from other queueing systems such as LSF (Zhou, 1992) or PBS (Henderson, 1995) in having an installation process that is simple enough to be carried out by a normal user. However, it will not have the advanced job management functions of LSF and PBS.

ACKNOWLEDGEMENTS

This work was aided by HRB Ireland, HEA Ireland, a Trinity College Dublin High Performance Computing studentship, and Science Foundation Ireland. Thanks to ISREC Bioinformatics for helpful discussions.

REFERENCES

Altschul,S.F., Madden,T.L., Schaffer,A.A., Zhang,J., Zhang,Z., Miller,W. and Lipman,D.J. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.

Brenner,S.E., Chothia,C. and Hubbard,T.J. (1998) Assessing sequence comparison methods with reliable structurally identified distant evolutionary relationships. *Proc. Natl Acad. Sci. USA*, **95**, 6073–6078.

Grant,J.D., Dunbrack,R.L., Manion,F.J. and Ochs,M.F. (2002) BeoBLAST: distributed BLAST and PSI-BLAST on a Beowulf cluster. *Bioinformatics*, **18**, 765–766.

Henderson,R.L. (1995) Job scheduling under the portable batch system. *Lecture Notes in Computer Science*, **949**, 337–360.

Jongeneel,V., Junier,T., Iseli,C., Hofmann,K. and Bucher,P. (1997) INSECT and MOLLUSCS—supercomputing on the cheap. *EMBNet News*, **4**, 3–5.

Luthy,R., Xenarios,I. and Bucher,P. (1994) Improving the sensitivity of the sequence profile method. *Protein Sci.*, **3**, 139–146.

Pearson,W.R. (2000) Flexible sequence similarity searching with the FASTA3 program package. *Methods Mol. Biol.*, **132**, 185–219.

Zhou,S. (1992) LSF: load sharing in large-scale heterogeneous distributed systems. In *Proceedings of the Workshop on Cluster Computing*.