Brazilian Journal of Medical and Biological Research (2003) 36: 463-464
ISSN 0100-879X Short Communication

pp-Blast: a "pseudo-parallel" Blast

E.C. Osório[1]*,
J.E. de Souza[1]*,
A.C. Zaiats[1],
P.S.L. de Oliveira[2,3]
and S.J. de Souza[1]

[1]Instituto Ludwig de Pesquisa sobre o Câncer, São Paulo, SP, Brasil
[2]Departamento da Ciência da Computação, Universidade de Santo Amaro, São Paulo, SP, Brasil
[3]Laboratório de Genética e Cardiologia Molecular, Instituto do Coração, São Paulo, SP, Brasil

Abstract

We have developed a software called pp-Blast that uses the publicly available Blast package and PVM (parallel virtual machine) to partition a multi-sequence query across a set of nodes with replicated or shared databases. Benchmark tests show that pp-Blast running in a cluster of 14 PCs outperformed conventional Blast running in large servers. In addition, using pp-Blast and the cluster we were able to map all human cDNAs onto the draft of the human genome in less than 6 days. We propose here that the cost/benefit ratio of pp-Blast makes it appropriate for large-scale sequence analysis. The source code and configuration files for pp-Blast are available at http://www.ludwig.org.br/biocomp/tools/pp-blast.

Key words
- Blast
- Search engines
- Fasta

Correspondence
S.J. de Souza
Instituto Ludwig de Pesquisa
sobre o Câncer
Rua Prof. Antonio Prudente, 109
4º andar
01509-010 São Paulo, SP
Brasil
Fax: +55-11-3207-7001
E-mail:
sandro@compbio.ludwig.org.br

*These authors contributed equally to this work.

Publication supported by FAPESP.

Received July 29, 2002
Accepted December 12, 2002

The exponential growth of the sequence databases and the importance of database searches in almost all fields of biomedical research have generated a bottleneck in the sense that few laboratories today have the conditions to execute large-scale sequence analyses. Although a reasonable number of WWW sites (see the first issue of 2002 of *Nucleic Acids Research*, 30 (1) for an update) make search engines available to the community, there is always a limit in the number of sequences that can be used as query. Furthermore, budget limitation is also a factor since large servers with enough memory and speed are extremely expensive.

Fasta (1) and Blast (2) are the two most popular programs for database searches. While there is a public parallel version of Fasta (www.virginia.edu/fasta) and attempts have been made to parallelize Blast (http://citeseer.nj.nec.com/376408.html), none of these last tools have been implemented for easy public access and/or for local use.

This motivated us to develop a program that distributes multiple Blast jobs across available processors using PVM (parallel virtual machine) to enhance throughput. We call this program pp-Blast ("pseudo-parallel" Blast). The term "pseudo" comes from the fact that we have not modified the original source code of Blast. The strategy adopted involves the partitioning of multiple query requests. While this does not reduce the time of an individual search, it has a significant effect on queries containing multiple sequences. A clear shortcoming of the present approach is that all nodes need to have a certain amount of memory to allow the entire database to be loaded.

Our major goals in developing pp-Blast were a low cost/benefit ratio, speed, simplicity and applicability. The procedure pp-Blast uses is detailed below and schematically illustrated in Figure 1. All nodes (running Linux or Unix) in the cluster have to be in the same logical network and databases have to

be accessible to all nodes. This is possible through the network although a bandwidth bottleneck may affect efficiency given the large database sizes. Another problem with a centralized database is the status of the database server and its file system that may affect performance. One alternative is to keep the databases in each node for local access. When using pp-Blast, a user can define if all nodes are available for the program and for what period of time. This scheduler is extremely useful, for instance, in academic environments when processors available for teaching can then become search engines during off hours. In the pp-Blast package, there is also a merger program that parses and sorts the output from each node. This step is extremely fast since it is executed at the level of memory and the output is not written to the disk. All Blast programs can be used in the context of pp-Blast.

Table 1. Cluster configuration.

#	Clock	Processor type	RAM (MB)	HD (GB)	HD Interface
2	1.0 GHz	Intel Pentium 3	256	40	IDE
3	1.0 GHz	AMD Athlon	128	40	IDE
5	450 MHz	AMD K6-2	128	10	IDE
4	400 MHz	AMD K6-2	64	5	IDE

Table 2. Comparative performance of pp-Blast and regular Blast both running Blastn. pp-Blast was used in the context of a PC cluster (described in Table 1) while regular Blast was used with ES-40 and ES-45 servers.

Chromosome	Chromosome size (MB)	Query size (MB)	Processing time		
			ES-40	ES-45	Cluster
1	325	40	13 h	5:30 h	3:21 h
22	49	40	2:30 h	1:05 h	31 min

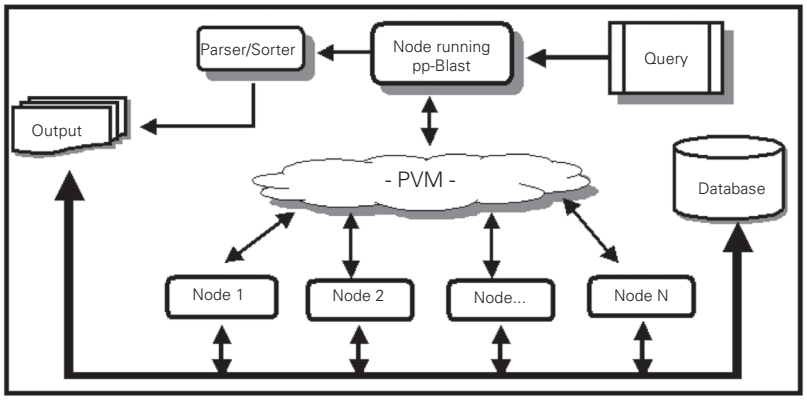

Figure 1. Schematic view of the pp-Blast pipeline. PVM, parallel virtual machine.

We have evaluated the performance of pp-Blast in a cluster containing 14 PCs running Linux. The configuration of the cluster is given in Table 1. We searched 50,000 EST (40 MB of sequence data) derived from the Human Cancer Genome Project (3) against human chromosome 1 (325 MB) and chromosome 22 (49 MB). Table 2 shows the processing time for the cluster using pp-Blast and for regular Blast (both running Blastn) in an ES-40 (4/500 MHz, 6 GB RAM) and ES-45 (4/1 GHz, 16 GB RAM). The Linux cluster running pp-Blast significantly outperformed both servers, although it should be noted that there are more processors in the cluster than in both servers. The smaller gain in performance observed when chromosome 1 is the database is related to both the size of the database and the size of the output file to be parsed. With pp-Blast running megablast we were able to map all human cDNAs (4.6 million sequences including 69,000 known mRNAs) onto the assembled human genome in 6 days.

We hope that the extreme low cost/benefit ratio of pp-Blast may render it a valuable resource for the community.

References

1. Pearson WR & Lipman DJ (1988). Improved tools for biological sequence comparison. *Proceedings of the National Academy of Sciences, USA*, 85: 2444-2448.
2. Altschul S, Madden T, Schaffer A, Zhang J, Zhang Z, Miller W & Lipman D (1997). Gapped BLAST and psi-BLAST: a new generation of protein database search programs. *Nucleic Acids Research*, 25: 3389-3402.
3. Camargo AA, Samaia HP, Dias-Neto E et al. (2001). The contribution of 700,000 ORF sequence tags to the definition of the human transcriptome. *Proceedings of the National Academy of Sciences, USA*, 98: 12103-12108.