BIOINFORMATICS APPLICATIONS NOTE Vol. 18 no. 11 2002
Pages 1546–1547

QuickTree: building huge Neighbour-Joining trees of protein sequences

Kevin Howe, Alex Bateman and Richard Durbin*

The Wellcome Trust Sanger Institute, The Wellcome Trust Genome Campus, Hinxton CB10 1SA, UK

Received on April 12, 2002; revised on May 20, 2002; accepted on May 24, 2002

ABSTRACT

Summary: We have written a fast implementation of the popular Neighbor-Joining tree building algorithm. `QuickTree` allows the reconstruction of phylogenies for very large protein families (including the largest Pfam alignment containing 27 000 HIV GP120 glycoprotein sequences) that would be infeasible using other popular methods.

Availability: The source-code for `QuickTree`, written in ANSI C, is freely available via the world wide web at http://www.sanger.ac.uk/Software/analysis/quicktree.

Contact: klh@sanger.ac.uk

Phylogenetic analysis is an important step in understanding the evolution of species and of gene and protein families. Molecular phylogenies have been particularly useful in resolving many taxonomic debates. Phylogenetic trees can also allow the definition of functional subfamilies within protein or gene families with multiple functions. As the deluge of genomic data continues unabated the size of data sets that need to be routinely analysed increases. Many protein families have hundreds or even thousands of members. An extreme example is that of HIV GP120 glycoprotein where over 27 000 sequences have been deposited in the SWISS-PROT/TrEMBL database (Bairoch and Apweiler, 1999). The 20 largest protein families in the Pfam database all contain over 4000 sequences (Bateman *et al.*, 2002).

Neighbor-Joining (Saitou and Nei, 1987; amended by Studier and Keppler, 1988) is a method for reconstructing phylogenies from a set of distances between each pair of sequences by successive clustering. Unlike simpler algorithms such as UPGMA (Sokal and Michener, 1958), Neighbor-Joining can reconstruct trees with additive edge lengths without making the assumption that the divergence of the sequences occurs at the same constant rate at all points in the tree.

Depending on the specific implementation, the run-time complexity of the Neighbor-Joining algorithm is between $O(n^3)$ and $O(n^4)$ with respect to the number of sequences, making the computation prohibitively expensive for large datasets. This is particularly apparent in some popular implementations, where the time taken to reconstruct a tree from a few hundred sequences or more becomes non-interactive (see Figure 1a).

We have produced an efficient $O(n^3)$ implementation of Neighbor-Joining in the program `QuickTree`. As a starting point, we implemented a re-factored version of the algorithm described in Durbin *et al.* (1998), which although having the same run-time complexity as earlier presentations, eliminates many unnecessary computations. We then used standard code optimization techniques (see, for example, Dowd and Severance, 1998) to further improve the efficiency of each iteration.

`QuickTree` accepts either a multiple sequence alignment or a distance matrix as input. In the case of the former, a distance matrix is internally constructed by the method used in `CLUSTAL W`, with one difference: the alignment is pre-processed to treat identical sequences as one. Although this may have no impact for the majority of uses, it can increase the speed by a significant factor for a small number of large, highly redundant Pfam families; the aforementioned GP120 for example merges 4611 sequences (17% of total) into existing identical sequences, reducing the time taken by a factor of two.

To assess the speed of `QuickTree`, we have compared it with three popular programs for tree construction. `CLUSTAL W` (Thompson *et al.*, 1994) , although primarily a multiple sequence alignment tool, gives the option of producing a Neighbor-Joining tree from a fixed multiple alignment. The `PHYLIP` package (by J. Felsenstein, http://evolution.genetics.washington.edu/phylip.html) is a set of tools for phylogenetic analysis. We have used the `NEIGHBOR` program as interfaced from the `EMBOSS` package (Rice *et al.*, 2000), allowing us to perform analyses in batch mode. Finally, `BIONJ` (Gascuel, 1997) is an efficient $O(n^3)$ implementation of a variant of the Neighbor-Joining algorithm that produces more accurate trees in many cases. Figure 1 shows how the run-time of `QuickTree` compares with these programs for a range of problem sizes.

*To whom correspondence should be addressed.

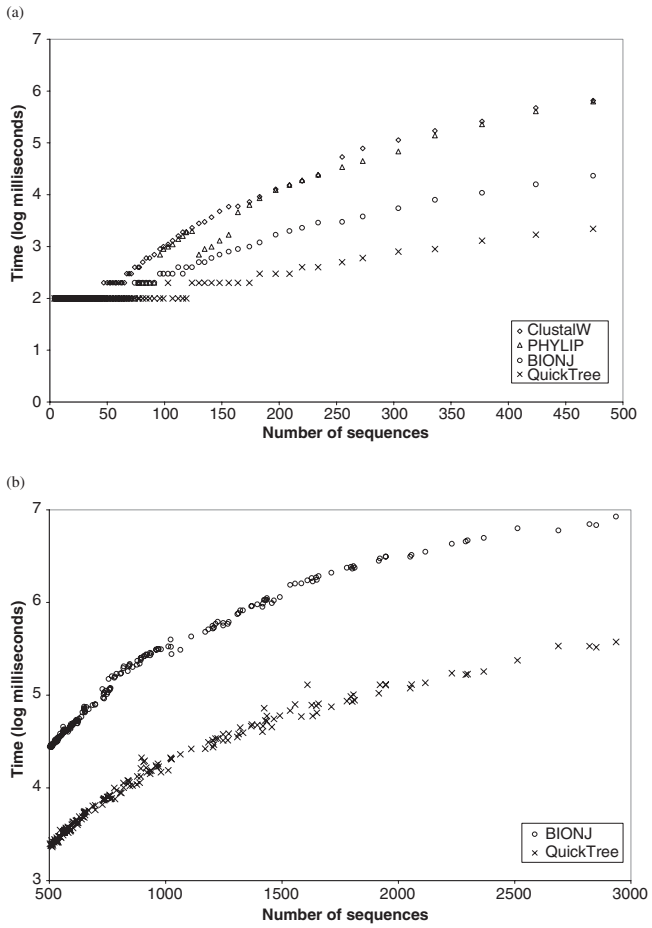

Fig. 1. Time taken to build a tree (on a Compaq DS10 workstation) against number of sequences for (a) a random selection of Pfam families with fewer than 500 members and (b) all Pfam families with between 500 and 3000 members. The times do not include the calculation of a distance-matrix for each alignment (with the exception of `CLUSTAL W`, for which it is a compulsory step). The absence of plots for `CLUSTAL W` and `PHYLIP` from (b) reflects the infeasibility of using these programs for large numbers of sequences.

Although `QuickTree` promises no advances in the accuracy of phylogeny reconstructions, we hope that the program will be useful to the phylogenetics research community at large, as its rapid turnaround time facilitates activities such as bootstrapping and the investigation of more sophisticated distance measures. For convenience, `QuickTree` includes an option to bootstrap when the input is a sequence alignment. In addition, `QuickTree` makes it feasible to construct trees for large databases of sequence alignments such as Pfam (Bateman *et al.*, 2002) when only limited resources are available. Building all 3621 trees for the Pfam (release 7.1) full alignments took 73 h on a desktop Compaq DS10 workstation (40 h of this time was taken up building the tree for the GP120 glycoprotein family). These trees are available in New Hampshire/Newick format (http://evolution.genetics.washington.edu/phylip/newicktree.html) from the Pfam website (http://www.sanger.ac.uk/Software/Pfam/).

REFERENCES

Bairoch,A. and Apweiler,R. (1999) The SWISS-PROT protein sequence data bank and its supplement TrEMBL in 1999. *Nucleic Acids Res.*, **27**, 49–54.

Bateman,A., Birney,E., Cerruti,L., Durbin,R., Etwiller,L., Eddy,S.R., Griffiths-Jones,S., Howe,K.L., Marshall,M. and Sonnhammer,E.L.L. (2002) The Pfam protein families database. *Nucleic Acids Res.*, **30**, 276–280.

Dowd,K. and Severance,C. (1998) *High Performance Computing*. O'Reilly, California.

Durbin,R., Eddy,S.R., Krogh,A. and Mitchison,G. (1998) *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, Cambridge.

Gascuel,O. (1997) BIONJ: An improved version of the NJ algorithm based on a simple model of sequence data. *Mol. Biol. Evol.*, **14**, 685–695.

Rice,P., Longden,I. and Bleasby,A. (2000) EMBOSS: the European Molecular Biology Open Software Suite. *Trends Genet.*, **16**, 276–277.

Saitou,N. and Nei,M. (1987) The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Mol. Biol. Evol.*, **4**, 406–425.

Sokal,R.R. and Michener,C.D. (1958) A statistical method of evaluating systematic relationships. *University of Kansas Scientific Bulletin*, **28**, 1409–1438.

Studier,J.A. and Keppler,K.J. (1988) A note on the neighbor-joining algorithm of Saitou and Nei. *Mol. Biol. Evol.*, **5**, 729–731.

Thompson,J.D., Higgins,D.G. and Gibson,T.J. (1994) CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res.*, **22**, 4673–4680.