BIOINFORMATICS APPLICATIONS NOTE
Vol. 18 no. 3 2002
Pages 502–504

TREE-PUZZLE: maximum likelihood phylogenetic analysis using quartets and parallel computing

Heiko A. Schmidt[1], Korbinian Strimmer[2], Martin Vingron[1] and Arndt von Haeseler[3,*]

[1]Max-Planck-Institut für Molekulare Genetik, Ihnestr. 73, D-14195 Berlin, Germany, [2]University of Oxford, Department of Zoology, South Parks Road, Oxford OX1 3PS, UK and [3]Max-Planck-Institut für Evolutionäre Anthropologie, Inselstr. 22, D-04103 Leipzig, Germany

Received on August 5, 2001; revised on September 20, 2001; accepted on October 9, 2001

ABSTRACT

Summary: TREE-PUZZLE is a program package for quartet-based maximum-likelihood phylogenetic analysis (formerly PUZZLE, Strimmer and von Haeseler, *Mol. Biol. Evol.*, **13**, 964–969, 1996) that provides methods for reconstruction, comparison, and testing of trees and models on DNA as well as protein sequences. To reduce waiting time for larger datasets the tree reconstruction part of the software has been parallelized using message passing that runs on clusters of workstations as well as parallel computers.

Availability: http://www.tree-puzzle.de. The program is written in ANSI C. TREE-PUZZLE can be run on UNIX, Windows and Mac systems, including Mac OS X. To run the parallel version of PUZZLE, a Message Passing Interface (MPI) library has to be installed on the system. Free MPI implementations are available on the Web (cf. http://www.lam-mpi.org/mpi/implementations/).

Contact: hschmidt@molgen.mpg.de; haeseler@eva.mpg.de

INTRODUCTION

As more and more sequence data become available in public databases, the runtime of sequence analysis software accumulates a serious bottleneck. To cope with this problem parallel computing increasingly enters the different areas of molecular sequence analysis (Trelles, 2001). Parallel software based on threads and message passing can for example be found in database searching (e.g. FASTA, BLAST, or HMMER) and in subsequent analysis like phylogenetic tree reconstruction, e.g. several parallelizations of the DNAml algorithm (Felsenstein, 1981; Trelles, 2001).

Here we present a parallelization of quartet puzzling (Strimmer and von Haeseler, 1996), a Maximum-Likelihood (ML) based tree reconstruction method.

PARALLELIZATION

The quartet puzzling algorithm is a three-step procedure (Strimmer and von Haeseler, 1996). In the *ML step* all $\binom{N}{4}$ quartet ML trees are reconstructed to find the most likely relationship for each set of four out of N sequences. In the *puzzling step* these quartet trees are composed into an overall so-called intermediate tree adding sequences one by one. Since the result of this step is highly dependent on the order of sequences, many intermediate trees from different input orders are constructed. From these intermediate trees a majority rule consensus tree is built in the *consensus step*.

The first two steps get very time-consuming when the number of sequences in the alignment increases. Fortunately both steps consist of many independent tasks (quartet-evaluations and puzzlings) and therefore are well suited for parallelization, to reduce the wall-clock time needed for the analyses.

For our parallelization we used master/worker concepts (cf. Figure 1a). This means one master process coordinates the tasks and sends them to the worker processes, which then do the computation. We did not only consider pure parallel platforms, but also took into account the problems of communication overhead that arise in a Cluster of Workstations (COW). This is important because COWs are likely to be more common and available for practicing systematics and evolutionary biology than are supercomputer-like systems.

We used the *Guided Self-Scheduling* algorithm (GSS, Polychronopoulos and Kuck, 1987; Hagerup, 1997) for load balancing. GSS is a well performing scheduling algorithm that reduces communication by clustering tasks, i.e. quartets or intermediate trees, into groups of decreasing size to keep all the workers equally busy. In comparison to other algorithms of this kind it does not need additional overhead to measure the speed of the processors. We modified the algorithm by introducing a lower threshold for the

*To whom correspondence should be addressed.

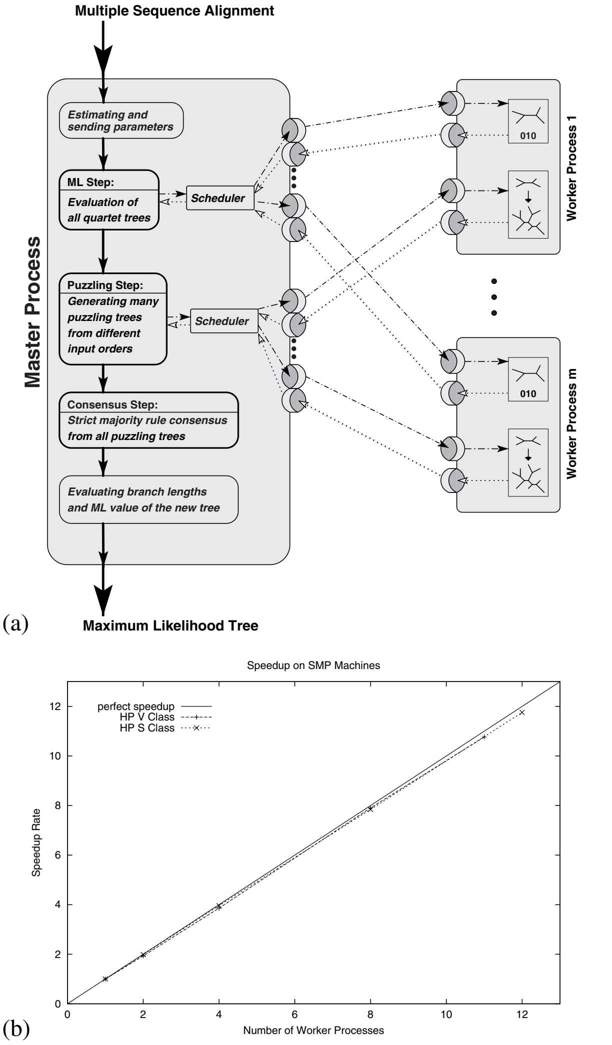
Multiple Sequence Alignment
Master Process
Estimating and sending parameters
ML Step:
Evaluation of all quartet trees
Scheduler
Puzzling Step:
Generating many puzzling trees from different input orders
Scheduler
Consensus Step:
Strict majority rule consensus from all puzzling trees
Evaluating branch lengths and ML value of the new tree
Maximum Likelihood Tree
010
Worker Process 1
010
Worker Process m
(a)
Speedup on SMP Machines
perfect speedup
HP V Class
HP S Class
Speedup Rate
0
2
4
6
8
10
12
Number of Worker Processes
(b)

Fig. 1. Parallelization of the quartet puzzling algorithm. (a) Parallelized workflow. The dashed arrows stand for communication between the processes. The communication to transmit the parameters to the workers was omitted to increase clarity. (b) The speedup of the program on an alignment of 60 gag sequences from HIV and SIV on HP shared memory parallel computers constructing 10 000 intermediate trees.

group size to keep the communication overhead low which is particularly important at the end of the analysis steps.

For the parallelization we used the Message Passing Interface (MPI, Snir *et al.*, 1998; Gropp *et al.*, 1998). Since MPI has become the *de facto* standard in parallel computing using message passing, implementations of the MPI libraries are available on almost all parallel platforms from massively parallel computers to COWs.

Speedup tests with different datasets on different machine types showed that the parallelization works very efficiently on parallel machines (e.g. Figure 1b for 60 gag sequences from HIV/SIV on HP shared memory V-Class and S-Class computers). The speedup of the parallel TREE-PUZZLE version reaches almost the perfect speedup, i.e. increasing the number of worker processes by a factor k reduces the runtime by the same factor. Because of the communication overhead a perfect speedup cannot be achieved.

On a very heterogeneous COW comprising 20 SUN workstations (Sparc 20, Sparc 4/5, Ultra 1, Ultra 5 with CPU speed from 65 to 300 MHz) the implementation of the scheduling proves to be very efficient, because it keeps all worker processes equally busy. Measuring the speedup on heterogeneous COWs is not possible because of the different CPU rates.

We also applied the parallel version to some large datasets, e.g. we computed the gene tree for 215 red algae small subunit rRNA sequences as present in the European ssu rRNA database (Van de Peer *et al.*, 2000). Applying the default settings, HKY85 model and 50 000 intermediate trees, on an 12-processor HP V-Class, this analysis took us 2 weeks with 12 worker processes instead of over 5.5 months with the non-parallelized program.

FURTHER TREE-PUZZLE FEATURES

Types of analysis

TREE-PUZZLE allows different types of phylogenetic analysis.

- *Reconstruction of phylogenetic trees* using the quartet puzzling algorithm (Strimmer and von Haeseler, 1996; Strimmer *et al.*, 1997).
- *Likelihood mapping* to examine the clustering of user defined subgroups of the aligned sequences or to visualize the phylogenetic content of the alignment (Strimmer and von Haeseler, 1997).
- *Evaluation of the maximum-likelihood value* of a given tree topology under a given evolutionary model (Felsenstein, 1981).
- *Kishino–Hasegawa test* to compare different tree topologies (Kishino and Hasegawa, 1989).
- The branch lengths can also be calculated under the assumption of a *molecular clock* (Felsenstein, 1988).

Evolutionary models

TREE-PUZZLE also includes a broad variety of evolutionary models.

- *Models for DNA sequences*: TN (Tamura and Nei, 1993), HKY (Hasegawa *et al.*, 1985), F84 (Felsenstein, 1984).

- *Models for protein sequences*: Dayhoff (Dayhoff *et al.*, 1978), JTT (Jones *et al.*, 1992), mtREV24 (Adachi and Hasegawa, 1996), BLOSUM 62 (Henikoff and Henikoff, 1992), VT (Müller and Vingron, 2000), WAG (Whelan and Goldman, 2001).
- *Model for doublets* (pairs of dependent nucleotides): SH (Schöniger and von Haeseler, 1994).
- *Model for binary state data*: F81 (Felsenstein, 1981).
- *Rate heterogeneity*: modeled by a discrete Gamma distribution and by allowing invariable sites (Yang, 1994).

With its features TREE-PUZZLE is providing an interesting resource for the phylogenetic analysis of large datasets.

Finally we would like to encourage everybody to test our program and make suggestions as to what other features to include in future releases.

ACKNOWLEDGEMENTS

This work is supported by Deutsche Forschungs–gemeinschaft and the Max Planck Society. We thank Antje Krause and Lutz Voigt for helpful discussions.

REFERENCES

Adachi,J. and Hasegawa,M. (1996) Model of amino acid substitution in proteins encoded by mitochondrial DNA. *J. Mol. Evol.*, **42**, 459–468.

Dayhoff,M.O., Schwartz,R.M. and Orcutt,B.C. (1978) A model of evolutionary change in proteins. In Dayhoff,M.O. (ed.), *Atlas of Protein Sequence Structure*, vol. 5, National Biomedical Research Foundation, Washington, DC.

Felsenstein,J. (1981) Evolutionary trees from DNA sequences: a maximum likelihood approach. *J. Mol. Evol.*, **17**, 368–376.

Felsenstein,J. (1984) Distance methods for inferring phylogenies: a justification. *Evolution*, **38**, 16–24.

Felsenstein,J. (1988) Phylogenies from molecular sequences: inference and reliability. *Annu. Rev. Genet.*, **22**, 521–565.

Gropp,W., Huss-Lederman,S., Lumsdaine,A., Lusk,E., Nitzberg,B., Saphir,W. and Snir,M. (1998) *MPI: The Complete Reference—The MPI Extensions,* vol. 2, 2nd edn, MIT Press, Cambridge, MA.

Hagerup,T. (1997) Allocating independent tasks to parallel processors: an experimental study. *J. Parallel Distrib. Comput.*, **47**, 185–197.

Hasegawa,M., Kishino,H. and Yano,T.-A. (1985) Dating of the human–ape splitting by a molecular clock of mitochondrial DNA. *J. Mol. Evol.*, **22**, 160–174.

Henikoff,S. and Henikoff,J.G. (1992) Amino acid substitution matrices from protein blocks. *Proc. Natl Acad. Sci. USA*, **89**, 10 915–10 919.

Jones,D.T., Taylor,W.R. and Thornton,J.M. (1992) The rapid generation of mutation data matrices from protein sequences. *Comput. Appl. Biosci.*, **8**, 275–282.

Kishino,H. and Hasegawa,M. (1989) Evaluation of the maximum likelihood estimate of the evolutionary tree topologies from DNA sequence data, and the branching order in Hominoidea. *J. Mol. Evol.*, **29**, 170–179.

Müller,T. and Vingron,M. (2000) Modeling amino acid replacement. *J. Comput. Biol.*, **7**, 761–776.

Polychronopoulos,C.D. and Kuck,D.J. (1987) Guided self-scheduling: a practical scheduling scheme for parallel supercomputers. *IEEE Trans. Comput.*, **36**, 1425–1439.

Schöniger,M. and von Haeseler,A. (1994) A stochastic model for the evolution of autocorrelated DNA sequences. *Mol. Phylogenet. Evol.*, **3**, 240–247.

Snir,M., Otto,S.W., Huss-Lederman,S., Walker,D.W. and Dongarra,J. (1998) *MPI: The Complete Reference—The MPI Core,* vol. 1, 2nd edn, MIT Press, Cambridge, MA.

Strimmer,K. and von Haeseler,A. (1996) Quartet puzzling: a quartet maximum-likelihood method for reconstructing tree topologies. *Mol. Biol. Evol.*, **13**, 964–969.

Strimmer,K. and von Haeseler,A. (1997) Likelihood-mapping: a simple method to visualize phylogenetic content of a sequence alignment. *Proc. Natl Acad. Sci. USA*, **94**, 6815–6819.

Strimmer,K., Goldman,N. and von Haeseler,A. (1997) Bayesian probabilities and quartet puzzling. *Mol. Biol. Evol.*, **14**, 210–213.

Tamura,K. and Nei,M. (1993) Estimation of the number of nucleotide substitutions in the control region of mitochondrial DNA in humans and chimpanzees. *Mol. Biol. Evol.*, **10**, 512–526.

Trelles,O. (2001) On the parallelisation of bioinformatics applications. *Brief. Bioinform.*, **2**, 181–194.

Van de Peer,Y., De Rijk,P., Wuyts,J., Winkelmans,T. and De Wachter,R. (2000) The European small subunit ribosomal RNA database. *Nucleic Acids Res.*, **28**, 175–176.

Whelan,S. and Goldman,N. (2001) A general empirical model of protein evolution derived from multiple protein families using a maximum likelihood approach. *Mol. Biol. Evol.*, **18**, 691–699.

Yang,Z. (1994) Maximum likelihood phylogenetic estimation from DNA sequences with variable rates over sites: approximative methods. *J. Mol. Evol.*, **39**, 306–314.