

# wolfPAC

## Building a High-Performance Distributed Computing Network for Phylogenetic Analysis Using 'Obsolete' Computational Resources

Patrick A. Reeves,<sup>1</sup> Philip H. Friedman<sup>2</sup> and Christopher M. Richards<sup>1</sup>

1 United States Department of Agriculture, Agricultural Research Service (USDA-ARS), National Center for Genetic Resources Preservation, Fort Collins, Colorado, USA

2 College of Natural Sciences, Colorado State University, Fort Collins, Colorado, USA

### Abstract

wolfPAC is an AppleScript®-based software package that facilitates the use of numerous, remotely located Macintosh® computers to perform computationally-intensive phylogenetic analyses using the popular application PAUP\* (Phylogenetic Analysis Using Parsimony). It has been designed to utilise readily available, inexpensive processors and to encourage sharing of computational resources within the worldwide phylogenetics community.

**Availability:** wolfPAC for Mac OS® 9.x is available for free from <http://lamar.colostate.edu/~reevesp/wolfPAC.shtml>

**Contact:** Christopher M. Richards ([chris.richards@colostate.edu](mailto:chris.richards@colostate.edu))

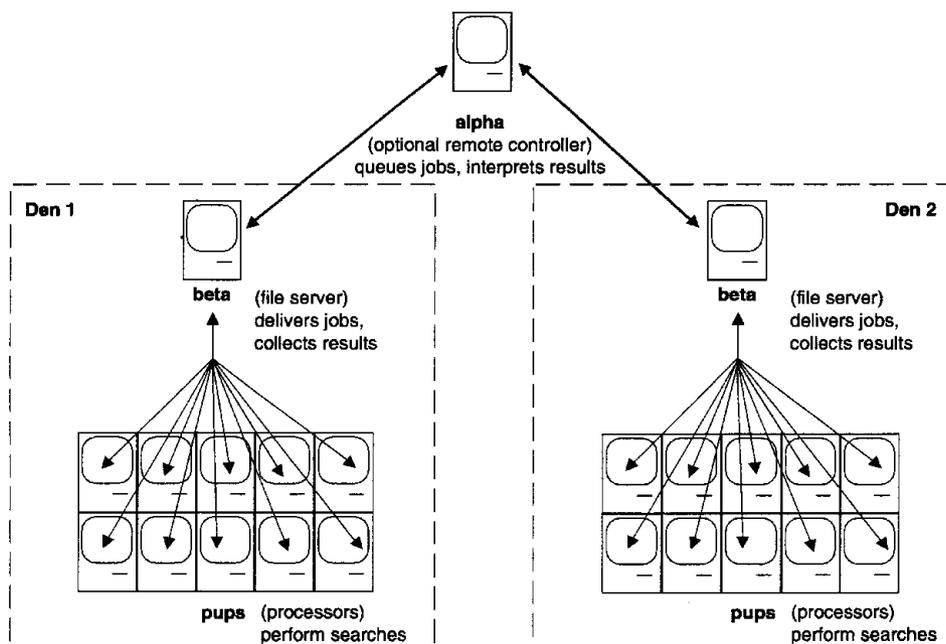
The reconstruction of phylogenetic trees under the maximum parsimony criterion is a non-deterministic polynomial time (NP)-complete problem.<sup>[1]</sup> For rooted bifurcating trees, the number of possible solutions increases as  $(2n-3)!/[2^{n-2}(n-2)!]$  where  $n$  is the number of labelled terminals;<sup>[2]</sup> thus, a significant computational effort is required to discover optimal trees. Another commonly used optimality criterion, maximum likelihood, shares a similarly large tree space, but owing to a computationally complex objective function, it requires substantially greater search time than parsimony.<sup>[3]</sup> Although heuristic methods have been developed to expedite searches, local optima and islands of equivalent optima are common occurrences on both the parsimony and likelihood surfaces.<sup>[4,5]</sup> Therefore, the use of simple hill-climbing algorithms alone cannot guarantee that global optima will be discovered. To increase the probability of finding all globally optimal trees, multiple independent searches, with random starting points for each search, are routinely performed.

For large datasets (e.g. >500 terminals), the computer processor time necessary to complete multiple independent searches can be prohibitive. Although refinements in search strategy,<sup>[6-8]</sup> new search algorithms<sup>[9,10]</sup> and parallel-computing approaches<sup>[11,12]</sup> have shown promise for decreasing overall search times, the continually increasing size of phylogenetic datasets has resulted in a situation where the computational resources available to a typical researcher may not be adequate to complete rigorous analyses

in a timely manner. To facilitate rapid and thorough searches of phylogenetic tree space, we have developed a distributed computing system, wolfPAC, that utilises the batch-processing capability of the phylogenetic analysis software PAUP\* (Phylogenetic Analysis Using Parsimony)<sup>[13]</sup> to perform multiple independent searches on numerous networked Macintosh® computers.

The wolfPAC analysis environment is a hierarchically organised network of processors that communicate via AppleScript® using the Apple® file protocol over an IP connection (figure 1). Written for Mac OS® 9.x, wolfPAC was designed to take advantage of the flood of surplus Macintosh® computers that have been made obsolete by the introduction of the UNIX®-based Mac OS® X operating system and, more recently, the 64-bit G5 processor. Despite their age, these machines contain powerful processors, some of which can take advantage of the AltiVec (Apple® developer connection, <http://developer.apple.com/hardware/ve>) accelerated instruction set available for PAUP\*.

A single wolfPAC phylogenetic analysis cluster includes a Mac OS® 9 server, which mediates job acquisition and acts as a repository for result files, and 1-10 client processors, which perform analyses using PAUP\*. Two AppleScript® programs are used in conjunction with script scheduling software to trigger a job query from the client to the server, commence a search and establish search duration. Jobs are created by the user as PAUP\* batch files that are written in accordance with guidelines described



**Fig. 1.** Organisation of processors in a wolfPAC distributed computing cluster. Although practical limitations may be encountered, there is no theoretical limit to the number of 'beta' servers that an 'alpha' machine can control. Access to 'beta' servers can be shared with the worldwide phylogenetics community through the wolfPAC website. Reproduced from Reeves et al.,<sup>[14]</sup> with permission.

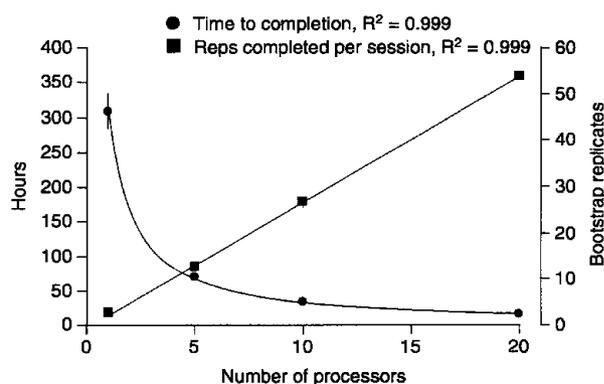
in the wolfPAC user manual (available from the wolfPAC website). Results for the analyses defined in the batch files are written to the server by the client processors. Job submission and routine maintenance can occur from the server, or from any remote machine with an IP connection, using the AppleScript® programs provided. Searches may be scheduled to utilise recurring idle periods (e.g. overnight) in large Macintosh® computer labs or may be conducted on dedicated clusters. Any number of wolfPAC clusters may be utilised simultaneously, so there is no theoretical limit on the number of processors that can be used.

In addition to streamlining the initiation and termination of PAUP\* runs on numerous computers, wolfPAC offers features that facilitate sharing of computational resources among researchers. Jobs may be submitted at various priority levels, so a researcher with a local wolfPAC may permit colleagues to access secondary priority processing time with the knowledge that, should the need arise, the privilege can be usurped by submission of a primary priority job. An optional third priority level is available to provide anonymous user access when no higher priority jobs have been submitted. Sharing of resources can be arranged through the wolfPAC website.

## Performance

Existing parallel-processing strategies for phylogenetic analysis have shown decreasing marginal performance as processors are added.<sup>[11,12]</sup> Systems that use a multiprocessor, serial approach are expected to exhibit linear scaling properties for replicated proce-

dures (figure 2) until a fixed minimum time, which is equal to the mean time necessary to complete a single replicate, is reached. Because all instructions are resident in client random access memory (RAM), and no data are exchanged between processors during a search, limitations based on network speed and the timing of data exchange operations (input/output) are obviated.



**Fig. 2.** Scaling curve for a replicated task using wolfPAC. Three 100-replicate parsimony bootstrap analyses were performed on the 'Zilla' dataset<sup>[15]</sup> using different numbers of processors. Search progress per unit time, as measured by the number of bootstrap replicates completed per processing session, increased linearly with the number of processors utilised. The amount of search time necessary to complete the analysis decreased as an inverse function of the form  $t = Cn^{-1}$ , where  $t$  is time,  $n$  is the number of processors and  $C$  is a constant. Therefore, when the number of processors was doubled, the search time was approximately halved.  $R^2$  = coefficient of determination.

Replicate searches, statistical analyses, simulation studies, and other such 'embarrassingly parallel'<sup>1</sup> phylogenetic analysis tasks can be efficiently accomplished using the multiprocessor, serial approach of wolfPAC. Decreasing the time necessary to complete such tasks is a simple matter of adding more processors. However, more sophisticated parallel-processing approaches (e.g. DOGMA [Distributed Object Group Metacomputing Architecture]<sup>2</sup> and GAML [Genetic Algorithm for Maximum Likelihood phylogeny inference]<sup>3</sup>) will become essential for many maximum likelihood analyses and for parsimony analysis of some extremely large datasets, such as those currently under investigation for the US National Science Foundation's 'Assembling the Tree of Life' project (<http://www.nsf.gov/od/lpa/news/02/pr0294.htm>). In these instances, the time to complete a single replicate search may be so large as to make a serial approach intractable.

Serial approaches have a logistical drawback, in that large quantities of output generated from the independent client processors must be fused into a single coherent result and interpreted subsequent to a search. Fortunately, for most phylogenetic analyses, the time required for post-processing procedures will be negligible compared with that of the original search effort. Several AppleScript® scripts are provided with wolfPAC to facilitate post-processing of results for three common phylogenetic procedures:

1. generation of the maximum parsimony tree from replicated random-taxon-addition-sequence searches conducted on numerous processors;
2. production of a bootstrap consensus tree from bootstrap replicates obtained on multiple independent processors.
3. generation of likelihood scores and an input file to test for an appropriate model of nucleotide evolution using MODELTEST.<sup>[16]</sup>

While likely to be of greatest utility for parsimony analyses because of their relatively short replicate search times and simple-to-parallelise nature, any search procedure available at the PAUP\* command line, including maximum likelihood analysis, can be performed using wolfPAC. Because of the large variety of available procedures, and because many phylogenetic studies require a novel combination of approaches, an efficient means of post-processing results for procedures other than the three listed above is left to the user.

To verify the utility of wolfPAC, we compared the rate at which shortest trees were recovered from the 500-taxon 'Zilla' dataset<sup>[15]</sup> with a variety of published searches (table I). A 20-processor wolfPAC using overnight idle-time and a simple search strategy

yielded trees of the shortest known length (length = 16 218) for Zilla approximately every 55 hours. The estimated minimum time necessary to recover trees of 16 218 steps using this strategy was 1.4 hours. This time could be achieved (with no modification of search strategy or algorithm) by adding more processors, which increases the probability that, at any given time, one or more searches is climbing the 'correct hill' (i.e. the one that leads to trees of 16 218 steps). Performance data for the wolfPAC implementation of MODELTEST<sup>[16]</sup> are shown in figure 3. Although addition of more processors would improve the 5-fold decrease in overall search time observed, there would be no performance advantage to using >56 processors simultaneously to calculate the required 56 likelihood scores. Given the 'embarrassingly parallel' implementation of MODELTEST used in wolfPAC, the theoretical minimum time to complete all tests is equal to the time required for the longest individual test, regardless of the number of processors used.

When using recurring idle periods, searches are triggered to commence at a specific clock time. We examined whether the values obtained from the system clock and used by PAUP\* to seed replicated searches were nonrandom, resulting in a redundant search effort (i.e. multiple processors search the same tree space).

**Table I.** Estimates of elapsed time necessary to find maximum parsimony trees of a given length for the 'Zilla' dataset<sup>[15]</sup> using a variety of search strategies

Search strategy <sup>a</sup>	Elapsed time (h)		Reference
	tree length	tree length	
	16 218	16 220	
TBR	NA	8280	Rice et al. 1997 <sup>[15]</sup>
wolfPAC/PAUP* <sup>b</sup>	55	3.6	
'NONA'	150	78	Nixon 1999 <sup>[6]</sup>
wolfPAC/'NONA'	38	1	
Ratchet	1.75	0.75	Nixon 1999 <sup>[6]</sup>
wolfPAC/ratchet	0.3	0.08	
Parallel ratchet	0.08	NA	Snell et al. 2000 <sup>[11]</sup>

a WolfPAC analyses conducted using 20 processors (450 MHz Macintosh® G3) and PAUP\* 4.0b10 software. Published analyses used different hardware and software, so performance estimates shown here are relative, rather than absolute, indicators of search efficiency.

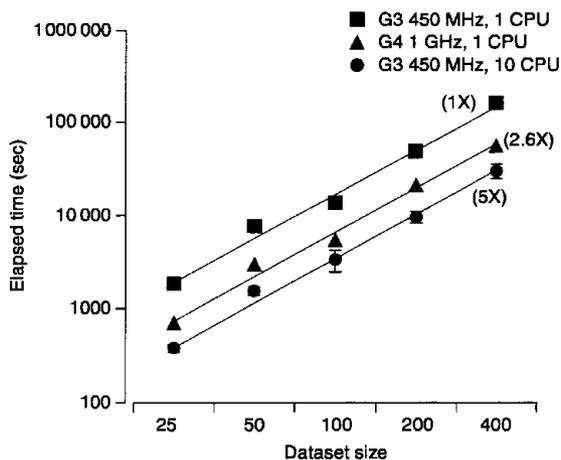
b 'PAUP\*' search strategy used TBR branch swapping, random taxon addition sequence, and saved ≤100 trees per replicate.

NA = not available; PAUP\* = Phylogenetic Analysis Using Parsimony; TBR = tree bisection reconnection.

**1** In high-performance computing jargon, 'embarrassingly parallel' refers to any problem that can be obviously divided into a number of completely independent parts such that no communication between processors is necessary to find the solution. Many phylogenetic analysis procedures belong to this category.

**2** Computational Phylogenomics Research Group, Brigham Young University. Available from <http://genome.byu.edu/masses.html>

**3** National Partnership for Advanced Computational Infrastructure, enVision Magazine. Available from <http://www.npaci.edu/envision/v16.3/hillis.html>



**Fig. 3.** Elapsed time necessary to calculate the 56 maximum likelihood scores required by MODELTEST<sup>[6]</sup> using single processors and using a 10-processor wolfPAC cluster. Five datasets with increasing numbers of taxa (derived from the 'Zilla' dataset)<sup>[15]</sup> were used for comparison. Improvement over baseline (1X) in time required to complete the calculations is shown in parentheses. CPU = central processing unit.

In approximately 5000 independent searches conducted over a 2-month period, seed values were randomly distributed from 0 to the maximum value of 2 147 483 646 used by PAUP\*. Redundant searches should therefore be extremely uncommon in the wolfPAC environment.

## Conclusion

Although progress continues to be made, it is unclear whether any hardware or software improvements will, in the end, permit timely recovery of globally optimal trees for very large datasets when using computationally slow maximum-likelihood algorithms.<sup>[3]</sup> The use of a serially-organised, distributed data-processing environment such as wolfPAC should encourage thorough searches of the parsimony tree space, allow rapid evaluation of statistical support for phylogenies when using resampling methods such as the bootstrap,<sup>[17]</sup> and expedite simulation studies. Furthermore, wolfPAC should facilitate parsimony analysis of large, genetic marker-based, intraspecific datasets, where no generally accepted model of evolution exists.

In summary, wolfPAC will help to mitigate the ever-increasing computational demands on phylogenetic research by:

- harnessing numerous, inexpensive, readily-available processors;

- developing an internet site (<http://lamar.colostate.edu/~reeves/wolfPAC.shtml>) to coordinate computational resource sharing;
- reducing overall search times, thereby encouraging more rigorous phylogenetic analyses.

## Acknowledgements

Thanks to Drs Mark Simmonds and Alan Yen for editorial improvements and discussions about phylogenetic algorithms. Funding for this project came from the USDA Agricultural Research Service base funding for the National Center for Genetic Resources Preservation.

The authors have no conflict of interest in creating wolfPAC as a purely research resource for phylogenetic studies.

## References

1. Foulds LR, Graham RL. The Steiner problem in phylogeny is NP-complete. *Adv Appl Math* 1982; 3: 43-9
2. Cavalli-Sforza LL, Edwards AWF. Phylogenetic analysis: models and estimation procedures. *Evolution* 1967; 21: 550-70
3. Sanderson MJ, Kim J. Parametric phylogenetics? *Syst Biol* 2000; 49: 817-29
4. Maddison D. The discovery and importance of multiple islands of most-parsimonious trees. *Syst Zool* 1991; 40: 315-28
5. Salter LA. Complexity of the likelihood surface for a large DNA dataset. *Syst Biol* 2001; 50: 970-8
6. Salter L, Pearl D. Stochastic search strategy for estimation of maximum likelihood phylogenetic trees. *Syst Biol* 2001; 50: 7-17
7. Goloboff PA. Analyzing large data sets in reasonable times: solutions for composite optima. *Cladistics* 1999; 15: 415-28
8. Nixon KC. The parsimony ratchet, a new method for rapid parsimony analysis. *Cladistics* 1999; 15: 407-14
9. Dopazo J, Carazo JM. Phylogenetic reconstruction using an unsupervised growing neural network that adopts the topology of a phylogenetic tree. *J Mol Evol* 1997; 44: 226-33
10. Lewis P. A genetic algorithm for maximum-likelihood phylogeny inference using nucleotide sequence data. *Mol Biol Evol* 1998; 15: 277-83
11. Snell Q, Whiting M, Clement M, et al. Parallel phylogenetic inference. In: SC2000 Proceedings. SC2000; 2000 Nov 4-10; Dallas. Available from URL: <http://www.sc2000.org/techpaper/papers/pap.pap283.pdf> [Accessed 2004 May 6]
12. Brauer MJ, Holder MT, Dries LA, et al. Genetic algorithms and parallel processing in maximum-likelihood phylogeny inference. *Mol Biol Evol* 2002; 19: 1717-26
13. Swofford DL. PAUP\* Phylogenetic Analysis Using Parsimony (and other methods) [computer program]. Version 4.0b10. Sunderland (MA): Sinauer Associates, 1999
14. Reeves PA, Friedman PH, Richards CM. wolfPAC [online]. Available from URL: <http://lamar.colostate.edu/~reeves/wolfPAC.shtml> [Accessed 2005 May 20]
15. Rice KA, Donoghue MJ, Olmstead RG. Analyzing large data sets: rbcL 500 revisited. *Syst Biol* 1997; 46: 554-63
16. Posada D, Crandall KA. MODELTEST: testing the model of DNA substitution. *Bioinformatics* 1998; 14: 817-8
17. Felsenstein J. Confidence limits on phylogenies: an approach using the bootstrap. *Evolution* 1985; 39: 783-91

Correspondence and offprints: Dr Christopher M. Richards, USDA-ARS, National Center for Genetic Resources Preservation, 1111 South Mason St, Fort Collins, CO 80521, USA.

E-mail: [chris.richards@colostate.edu](mailto:chris.richards@colostate.edu)