Parallelizing SuperFine

Diogo Telmo Neves
ESTGF - IPP and
Universidade do Minho
Portugal
dtn@ices.utexas.edu

Tandy Warnow
Dept. of Computer Science
The Univ. of Texas at Austin
Austin, TX 78712
tandy@cs.utexas.edu

João Luís Sobral
Departamento de Informática
Universidade do Minho
Portugal
jls@di.uminho.pt

Keshav Pingali
Dept. of Computer Science
The Univ. of Texas at Austin
Austin, TX 78712
pingali@cs.utexas.edu

ABSTRACT

The estimation of the Tree of Life, a rooted binary tree representing how all extant species evolved from a common ancestor, is one of the grand challenges of modern biology. Research groups around the world are attempting to estimate evolutionary trees on particular sets of species (typically clades, or rooted subtrees), in the hope that a final "supertree" can be produced from these smaller estimated trees through the addition of a "scaffold" tree of randomly sampled taxa from the tree of life. However, supertree estimation is itself a computationally challenging problem, because the most accurate trees are produced by running heuristics for NP-hard problems. In this paper we report on a study in which we parallelize SuperFine, the currently most accurate and efficient supertree estimation method. We explore performance of these parallel implementations on simulated data-sets with 1000 taxa and biological data-sets with up to 2,228 taxa. Our study reveals aspects of SuperFine that limit the speed-ups that are possible through the type of outer-loop parallelism we exploit.

Categories and Subject Descriptors

D.1.3 [**Programming Techniques**]: Concurrent Programming—*parallel programming*; G.2.2 [**Discrete Mathematics**]: Graph Theory—*graph algorithms, trees*; J.3 [**Life and Medical Sciences**]: Biology and Genetics—*phylogenetic reconstruction*

General Terms

Algorithms Performance

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
SAC'12 March 25-29, 2012, Riva del Garda, Italy.
Copyright 2011 ACM 978-1-4503-0857-1/12/03 ...$10.00.

Keywords

Supertree, phylogeny estimation, polytomy, irregular applications, parallelization, shared memory

1. INTRODUCTION

Phylogeny (i.e., evolutionary tree) estimation is a fundamental step in many biological analyses, including protein structure and function prediction, comparative genomics, drug design, etc. Most of the phylogenetic estimation methods that have good accuracy take multiple sequence alignments as input and then either attempt to solve an NP-hard optimization problem (such as Maximum Likelihood [21] or Maximum Parsimony, the Hamming distance Steiner Tree problem [10]) or are Bayesian methods (such as MrBayes [22]) that typically utilize MCMC to search through an exponentially large space of phylogenetic trees. All these techniques are computationally very intensive, in some cases taking months of analysis to complete on only moderately large data-sets [14, 15]. Therefore, alternate estimation techniques have been developed that bypass these inherently computationally intensive approaches.

Among these techniques are "supertree" methods, which combine estimated trees on small subsets into a tree on the full set of taxa. The input to a supertree method is a set of unrooted[1] phylogenetic trees (called "source trees"), each on a subset of the full set of taxa, and the output is an unrooted tree on the full set of taxa. These source trees are typically estimated for clade-based subsets (where a clade is a subtree of the full tree obtained by taking all the leaves below an internal node in the tree), or for a set of randomly selected taxa; these two types of source trees are called "clade-based trees" and "scaffold trees", respectively. Supertree methods are studied on simulated data-sets in order to evaluate topological accuracy. Since the most accurate methods are based upon NP-hard optimization problems, supertree methods

[1] The focus on unrooted trees is because locating the root in a phylogenetic tree depends upon having a carefully selected outgroup, something which is not always possible; furthermore, the stochastic models of evolution currently used in phylogenetic estimation are almost all time-reversible, and it is mathematically impossible to identify the root from such models.

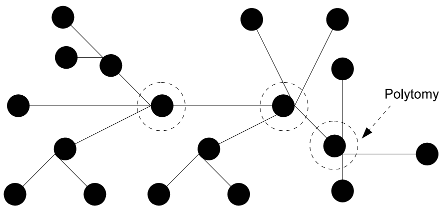

Figure 1: Example of an unrooted tree with three polytomies (each polytomy is highlighted by a dashed circle).

are computationally intensive. This is even true if all the source trees are correct, as constructing the true supertree from unrooted source trees is NP-complete [25]. There is a rich literature in supertree methods, with an overview of early methods provided in [5], and active ongoing research in the area [1, 3, 6, 8, 9, 12, 19, 20, 26, 29, 30].

The most popular supertree method is MRP (Matrix Representation with Parsimony) [2, 18], which replaces the set of source trees by one large "partial binary matrix" (a matrix over $\{0, 1, ?\}$), and runs heuristics for Maximum Parsimony. Studies evaluating heuristics for MRP in comparison to other supertree methods have demonstrated its superior accuracy and feasibility of use on data-sets with more than a few hundred taxa [29, 30].

Recently, Swenson et al. developed a new supertree method called "SuperFine" [31], which is both faster than MRP and also more accurate (as determined on simulated data-sets, described below). SuperFine has three phases. The first phase reads the input source trees, and is very fast, taking much less time than the next phases. The second phase produces a partially resolved unrooted tree called the "Strict Consensus Merger" (SCM) tree [11]. The SCM tree has a mathematical property that results in it typically being only partially resolved (i.e., it has high degree nodes): it only contains edges that are in agreement with every source tree.

SuperFine's third phase refines the SCM tree[2] by replacing each node of degree greater than three (also known as a polytomy, see Figure 1) in the tree by a (hopefully fully) resolved tree; this is called "refining the polytomy". The order in which these refinements of each polytomy are performed does not impact the final tree, and hence this phase is embarrassingly parallel. Refining a single polytomy of degree d is obtained by running an MRP heuristic on a new set of source trees, each with at most d leaves. Since MRP heuristics are computationally intensive, especially on large datasets, refining a polytomy is likely to be computationally intensive when the degree of the polytomy is large, but may be fast when the degree is small.

Swenson et al. studied SuperFine in comparison to other supertree methods (including MRP) on a collection of previously studied biological supertree data-sets and simulated data-sets [31]. SuperFine was generally faster than MRP: it never took more than 2.5 minutes on any simulated supertree data-set, and under 35 minutes on the most difficult biological data-set, while MRP heuristics took up to 2 hours on the simulated data-sets and 14 hours on the most difficult biological supertree data-set. SuperFine is faster than MRP because instead of running MRP on one data-set with a large number of taxa, it runs several MRP analyses on smaller numbers of taxa, where the total number of taxa over all the data-sets is at most the original number of taxa. SuperFine also typically matched or improved upon the topological accuracy of MRP and the other supertree methods. For example, for the biologically most meaningful simulation condition (1000 taxa with scaffold trees containing 20% of the taxa, randomly sampled from the full set of taxa), SuperFine had 16% missing branch rate, MRP had 20% missing branch rate, and the other supertree methods had even higher missing branch rates (where the missing branch rate is the fraction of the internal edges in the model tree that do not appear in the estimated tree). The reason for this improvement is that the SCM tree only includes edges that are in agreement with every source tree, and so these edges tend to be accurate. The restriction imposed by SuperFine prevents MRP solutions from being found that do not refine the SCM tree, and results in improved topological accuracy relative to unconstrained MRP searches.

In this study, we parallelized SuperFine and compared performance of these parallel versions to that of sequential SuperFine, using biological and simulated data-sets obtained from [31]. These biological data-sets consist of:

- CPL (comprehensive papilionoid legumes), 2228 taxa, 39 source trees, studied originally in [16],
- THPL (temperate herbaceous papilionoid legumes), 558 taxa, 19 source trees, studied originally in [33],
- Marsupials, 267 taxa, 158 source trees, studied originally in [7],
- Placental Mammals, 116 taxa, 726 source trees, studied originally in [4], and
- Seabirds, 121 taxa, 7 source trees, studied originally in [13].

The simulated data-sets are based upon mathematical models of taxon sampling that reflect the best practice of systematic biologists, so that each data-set contains several clade-based trees and one scaffold tree. We indicate the percentage of the taxa in the scaffold tree by the "scaffold factor". We present results for scaffold factors of 20% (the most typical case) and 100% (much less likely in practice, but gives highly accurate supertrees).

2. PARALLELIZING REFINEMENT OF THE SCM TREE

2.1 Profiling of SuperFine Baseline Implementation

As noted, the first phase in SuperFine is very fast, and we focus on the last two phases; see Table 1 for average running times (in seconds) for the last two phases on biological and simulated data-sets. Thus, SuperFine spends more time on

[2] Refining a tree is the inverse of contracting edges in the tree; thus, a tree T refines tree T' if T' can be obtained from T by contracting some edges in T.

Table 1: Average Time Spent on the Second (Construct SCM) and Third (Refine SCM) Phases of SuperFine.

Data-set	Average Time (s) Spent to	
	Construct SCM	Refine SCM
Sim. 20%-scaffold	6.446	23.866
Sim. 100%-scaffold	18.682	36.553
CPL	90.953	1003.855
Marsupials	12.733	103.801
Placental Mammals	49.669	131.794
Seabirds	0.307	0.924
THPL	2.745	21.583

the third phase than on the other phases. Therefore, we focus attention on this phase, which refines the SCM tree. Because the polytomy refinements can be done independently, parallelizing the refinement phase represents a natural opportunity for a substantial speed-up. This paper explores this possibility.

We performed an exploratory experiment evaluating the amount of time needed to produce the refinement around each polytomy. Figures 2 and 3 give results for the simulated data-sets with 20% and 100% scaffolds, respectively; results on the biological data-sets are shown in Figures 4-8.

In general we see that running times increased with the degree of the polytomy, and that the refinement of the largest polytomy could take much more time than any other polytomy (see, in particular, results on the CPL and THPL data-sets in Figures 4 and 8). However, these observations did not always hold. For example, on the simulated 1000-taxon 100%-scaffold data-sets, refining the third largest polytomy took more time than any other polytomy (Figure 3), while many data-sets had some small degree polytomies that took more time to refine than many larger degree polytomies (Figures 2, 3, and 7). Thus, predicting the relative amounts of time needed to resolve the different polytomies depends upon factors other than just the degree of the polytomy.

The explanation for this break from the predicted pattern comes from the nature of the problem that MRP heuristics are solving. Recall that we resolve each polytomy of degree d by running an MRP heuristic on a set of source trees whose leaves are drawn from $\{1, 2, \ldots, d\}$. When d is small, this is likely to be computationally fast; however, other factors, such as the number of source trees (which differs between polytomies) and how much the source trees conflict with each other, also contribute to the running time. Also, MRP heuristics employ randomness in order to search effectively for good solutions, and to some extent differences in running time can be due to randomness in the algorithm. Thus, while generally larger degree polytomies will take more time to resolve than smaller degree polytomies, there are good

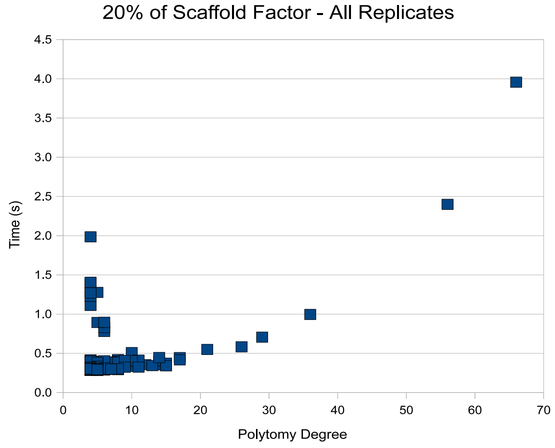

Figure 2: Time spent to process each polytomy of the simulated 1000-taxon data-sets (all replicates), for a scaffold factor of 20%.

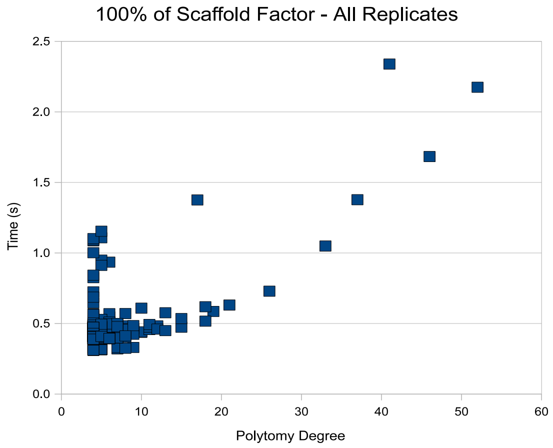

Figure 3: Time spent to process each polytomy of the simulated 1000-taxon data-sets (all replicates), for a scaffold factor of 100%.

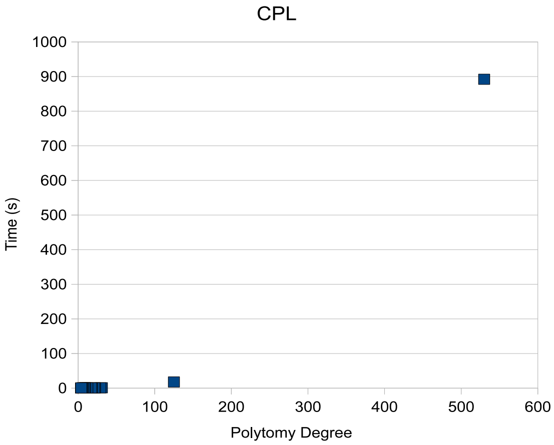

Figure 4: Time spent to process each polytomy of the CPL (biological) data-set, out of 2228 total taxa.

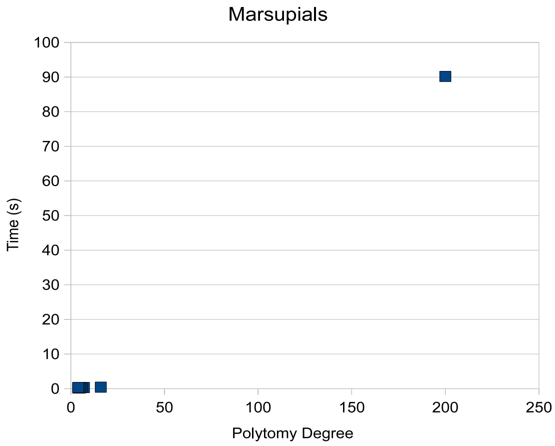

Figure 5: Time spent to process each polytomy of the Marsupials (biological) data-set, out of 267 total taxa.

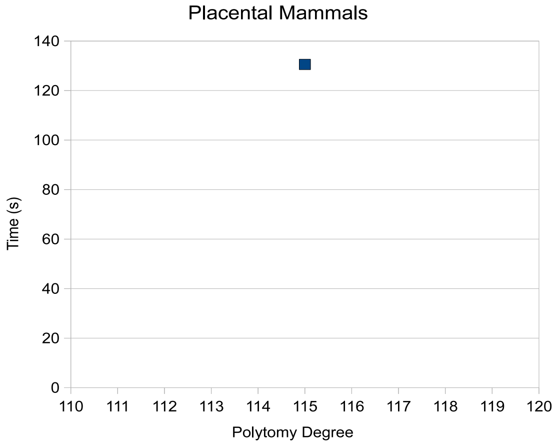

Figure 6: Time spent to process each polytomy of the Placental Mammals (biological) data-set, out of 116 total taxa.

reasons why this is not always the case.

Finally, for three of the biological data-sets (CPL, Placental Mammals, and Marsupials), the refinement of the largest polytomy took from 88% to 99% of the total time in the refinement phase. These results together suggest that largest degree polytomy is likely to dominate the running time of the SCM refinement phase, and in some cases it may limit the speed-up that is possible through this type of approach.

2.2 Polytomy Refinement Parallelization

We considered two simple shared-memory (SM) dynamic parallel algorithms, *SuperFine SM P1* and *SuperFine SM P2*.

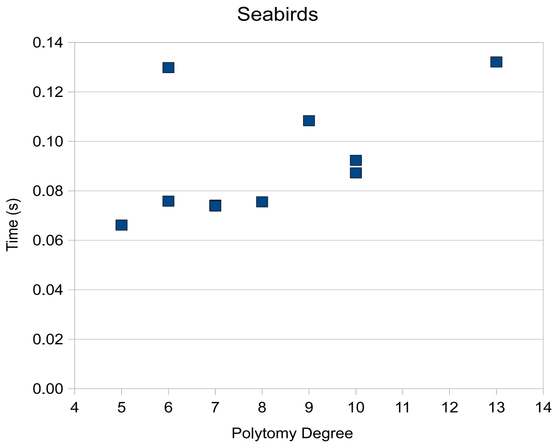

Figure 7: Time spent to process each polytomy of the Seabirds (biological) data-set, out of 121 total taxa.

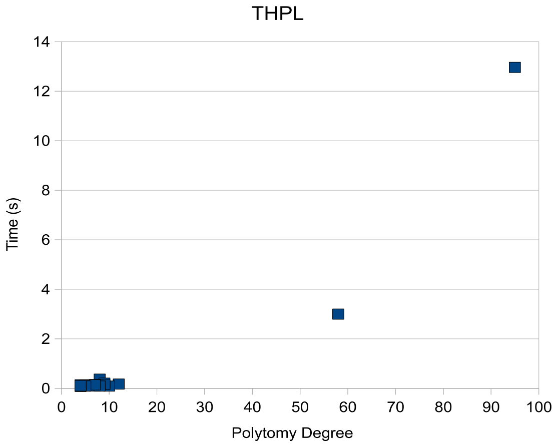

Figure 8: Time spent to process each polytomy of the THPL (biological) data-set, out of 558 total taxa.

The input to each algorithm is an unresolved tree (the SCM tree computed in the second phase of SuperFine) and the set of source trees. We let $p_1, p_2, \ldots, p_k$ denote the polytomies in the SCM tree, and let d_i be the degree of p_i. Both algorithms have the same basic structure: the central shared worklist ($CSWL$) is a list of the unprocessed polytomies in the SCM tree, and each processor picks the first polytomy from the list. The difference between the two algorithms is whether the list of polytomies is sorted or not: in the first version, the list is not sorted, and in the second version the list is sorted according to decreasing polytomy degrees. As observed earlier, this ordering is likely (though not guaranteed) to put the polytomies that will take more time to process before the polytomies that can be processed quickly.

3. EXPERIMENTAL DESIGN

3.1 Data-sets

We used the biological (i.e., real) data-sets and simulated data-sets described earlier; all of which were used in earlier studies to evaluate supertree methods [27, 28]. In all, we had 20 simulated 1000-taxon supertree data-sets and 5 biological supertree data-sets with up to 2228 taxa. Each supertree data-set contains many trees, each on a subset of the taxa.

The trees in the simulated data-sets are estimated by RAxML [24], a very accurate and fast heuristic for maximum likelihood. The sequence data-sets given to RAxML were simulated on subtrees of 1000-taxon model trees under GTR+Gamma. As described earlier, the subtrees for these sequence data-sets reflect realistic patterns of missing data, including both biological processes and taxon sampling strategies used by systematists in phylogenetic studies [27, 28]. These data-sets include clade-based trees and scaffold trees, as described earlier. Clade-based trees by themselves cannot be used to assemble a tree on the entire data-set, but the inclusion of the scaffold tree provides the overlap that makes this possible. As observed in [31], supertrees are typically more accurate when scaffold trees are more densely sampled, but typical practice in systematics produces scaffolds containing only a sparse subset of the taxa. We present results for 1000-taxon model trees with 20% and 100% scaffolds, in order to consider both extremes (the case which is more typical of biological practice, and the case where the best accuracy is obtained, respectively). We produced 10 replicates for each supertree input condition to obtain estimates of average performance.

3.2 Refinement Method

The third phase of SuperFine refines each polytomy in the SCM tree using MRP, which is maximum parsimony on a partial binary character MRP matrix defined by the SCM tree and the source trees. We used an effective maximum parsimony heuristic called the parsimony ratchet [17], as implemented in PAUP* [32]; this is the same MRP heuristic used in the sequential version studied in [31], and enables us to provide a fair comparison. The parsimony ratchet is a popular technique for maximum parsimony analyses of large data-sets.

3.3 Multicore Analyses

We used an 8-core Intel Nehalem with 48 GB of shared memory. For each data-set and for each number of cores, we performed six runs of each of the three methods of analysis (sequential or parallel). Thus, for the 1000-taxon data-sets with 20%-scaffolds, the number of runs involving the sequential and the parallel versions is $10 \times 6 + 10 \times 6 \times 4 \times 2 = 540$, since we made 6 runs per replicate and there are: the baseline implementation (sequential code) plus two parallel versions, 10 replicates per data-set, and 4 configurations of numbers of cores (1, 2, 4, and 8).

4. RESULTS

As explained on Section 2, the only difference between the two parallel versions, *SuperFine SM P1* and *SuperFine SM P2*, is that the central shared worklist ($CSWL$) for *SuperFine SM P2* is sorted by decreasing polytomy degree (d), while it is not for *SuperFine SM P1*.

Figure 9 compares the average running times for the polytomy refinement phase of the sequential code to that of *SuperFine SM P1* and *SuperFine SM P2* on the 1000-taxon simulated data-sets with scaffold factor of 20%. The two parallel versions have almost identical running times, but *SuperFine SM P2* has a small advantage. Results (not shown) for the 1000-taxon data-sets with 100%-scaffolds show a similar pattern, with a small advantage for *SuperFine SM P2* over *SuperFine SM P1*. The only difference between the two conditions is that in absolute terms, all methods take more time on the 100%-scaffold factor data-sets than on the 20%-scaffold factor data-sets.

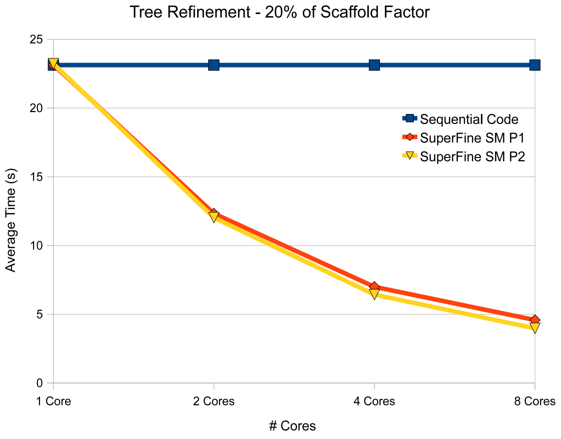

Figure 9: Average running times of sequential code and the first and the second parallel versions, on simulated data-sets with 20% of scaffold factor.

Table 2 gives the average speed-up of the two parallel versions over the sequential code for each biological data-set and simulated model condition, as a function of the number of cores. Here we see substantial differences between the

Table 2: Average Speed-ups of the First and the Second Parallel Versions for Simulated and Biological Data-sets.

Data-set	Parallel Version	Number of Cores			
		1	2	4	8
Simulated 20%-scaf.	SuperFine SM P1	1.01	1.94	3.41	5.21
	SuperFine SM P2	0.98	1.99	3.71	6.00
Simulated 100%-scaf.	SuperFine SM P1	1.00	1.95	3.61	6.02
	SuperFine SM P2	1.01	1.96	3.69	6.34
CPL	SuperFine SM P1	0.98	1.09	1.09	1.13
	SuperFine SM P2	1.05	1.14	1.07	1.13
Marsupials	SuperFine SM P1	0.99	1.02	1.07	1.08
	SuperFine SM P2	0.99	1.04	1.13	1.12
Placental Mammals	SuperFine SM P1	0.99	1.00	1.00	1.01
	SuperFine SM P2	0.95	0.96	0.99	1.02
Seabirds	SuperFine SM P1	1.00	1.88	2.89	3.76
	SuperFine SM P2	1.00	1.88	3.16	4.25
THPL	SuperFine SM P1	1.01	1.53	1.63	1.63
	SuperFine SM P2	0.99	1.55	1.64	1.63

biological and simulated data-sets. On the simulated data-sets we obtain a good but not linear speed-up, but all but one biological data-sets has a speed-up of less than 2 when run with 8 cores. At one extreme, the Placental Mammals data-set shows no speed-up at all (i.e., a speed-up of approximately 1), Marsupials and CPL have speed-ups of approximately 1.1, THPL has a speed-up of 1.63, and the Seabirds has a speed-up of 4.25. An examination of the polytomy sizes and times to resolve these polytomies for each biological data-sets reveals why we see these low speed-ups. For example, the Placental Mammals tree has a polytomy of degree 115 and only 116 taxa in total; this means that the SCM tree has only one internal edge, and the refinement step is essentially starting from scratch. The CPL, THPL, and Marsupials data-sets are problematic for different reasons. In each of their cases, the largest degree of any polytomy in the SCM tree is much less than the full number of taxa (i.e., the SCM tree is fairly well resolved). However, the time needed to resolve the largest polytomy dominates the refinement phase, using at least 88% of the total time in the sequential implementation, thus keeping the speed-up potential close to 1 (see also Figures 4-8 for the scatterplots of polytomy degrees and time to resolve each polytomy). Only the Seabirds data-set has mostly small polytomies, and all take within a narrow range of running times to resolve (from 0.06 to 0.14 seconds, see Figure 7). Thus, it is not surprising that the best speed-up is on the Seabirds data-set, and that the speed-ups on the CPL, Placental Mammals, and Marsupials are extremely small.

Why do we see better results on the simulated data-sets than on the biological data-sets? First, we note that the SCM trees for the simulated data-sets tend to be well resolved, with maximum polytomy degrees that are less than 7% of the total number of taxa. This keeps the total time needed to refine any one polytomy relatively small, and differences in time to resolve different polytomies also relatively small. For the biological data-sets, with the exception of Seabirds, the maximum degree of any polytomy was at least 17% of the total number of taxa. Thus, the SCM trees for the simulated and Seabird data-sets are relatively resolved and do not have very large polytomies that dominate the running time in the refinement step, but the SCM trees for the other biological data-sets have at least one very large polytomy. The question is why we see these differences in resolution between these datsets?

There are several possible answers, but the most likely ones are these. First, the supertree data-sets produced in simulation are based upon maximum likelihood analyses of true alignments, and are therefore likely to be highly accurate. In contrast, the trees in the biological data-sets were based upon many different alignment and phylogenetic reconstruction methods, including ones that are now known to be less accurate (at the time the source trees were constructed, the relative performance of methods was not as well known, and the methods that were then available were not as accurate as the best current methods). It is well known that increases in source tree error results in increases in supertree estimation error; however, here we note an additional consequence of source tree error: because the SCM tree is extremely conservative and only includes edges that are supported by all source trees, the SCM tree will lose resolution when the source trees have high error rates. Therefore, source tree estimation error not only impacts the accuracy of the resultant supertree, it also impacts the running time of SuperFine and its parallel implementations.

Another factor that results in the SCM tree being unresolved is poor taxon sampling in the source trees, especially the use of random sampling in defining the source trees [23]. Unfortunately, many supertree studies are designed using random samples of taxa for the source trees. Optimal taxon sampling strategies are not yet understood, but this is clearly an area for future research with potential for substantial impact on estimations of the Tree of Life.

In summary, our parallelization strategy obtains excellent speed-up for the simulated data-sets, and good speed-ups for one of the biological data-sets. For the biological data-sets, speed-ups are limited by two factors: (i) the number of polytomies in an SCM tree can be small, and (ii) the refinement time for polytomies can vary widely. Further improvements in speed-up require the parallelization of individual polytomy refinements.

5. ACKNOWLEDGMENTS

This research was supported by the US National Science Foundation (grants DEB-0733029 and DBI-1062335), the John P. Simon Guggenheim Foundation, and the Fundação para a Ciência e a Tecnologia (grants SFRH/BD/42634/2007 and UTAustin/CA/0056/2008) in Portugal. This work is, also, supported in part by NSF grants 1111766, 0923907, 0833162, 0719966, and 0702353 and by grants from Qualcomm and NEC. The first author would like to acknowledge the support provided by his current place of employment, Escola Superior de Tecnologia e Gestão de Felgueiras (ESTGF) - Instituto Politécnico do Porto (IPP). We thank Shel Swenson, the author of SuperFine, for reviewing this paper before submission. The SuperFine software, study datasets, and publication [31] are available at `http://www.cs.utexas.edu/users/phylo/software/superfine/submission/`. The source code is also available at Dryad, doi:10.5061/dryad.879st.

6. REFERENCES

[1] M. Bansal, J. G. Burleigh, O. Eulenstein, and D. Fernández-Baca. Robinson-Foulds supertrees. *Algorithms for Molecular Biology*, 5:18, 2009.

[2] B. R. Baum. Combining trees as a way of combining data sets for phylogenetic inference, and the desirability of combining gene trees. *Taxon*, 41:3–10, 1992.

[3] B. R. Baum and M. A. Ragan. The MRP method. In O. R. P. Bininda-Emonds, editor, *Phylogenetic Supertrees: combining information to reveal The Tree Of Life*, pages 17–34. Kluwer Academic, Dordrecht, the Netherlands, 2004.

[4] R. M. D. Beck, O. R. P. Bininda-Emonds, M. Cardillo, F. G. R. Liu, and A. Purvis. A higher-level MRP supertree of placental mammals. *BMC Evol. Biol.*, 6:93, 2006.

[5] O. R. P. Bininda-Emonds. *Phylogenetic Supertrees: combining information to reveal The Tree Of Life.* Computational Biology. Kluwer Academic, Dordrecht, the Netherlands, 2004.

[6] J. G. Burleigh, O. Eulenstein, D. Fernández-Baca, and M. J. Sanderson. MRF supertrees. In O. R. P. Bininda-Emonds, editor, *Phylogenetic Supertrees: combining information to reveal The Tree Of Life*,

pages 65–86. Kluwer Academic, Dordrecht, the Netherlands, 2004.

[7] M. Cardillo, O. R. P. Bininda-Emonds, E. Boakes, and A. Purvis. A species-level phylogenetic supertree of marsupials. *J. Zool.*, 264:11–31, 2004.

[8] D. Chen, O. Eulenstein, D. Fernández-Baca, and M. J. Sanderson. Minimum-flip supertrees: Complexity and algorithms. *IEEE/ACM Trans. Comp. Biol. Bioinform.*, 3:165–173, 2006.

[9] J. A. Cotton and M. Wilkinson. Majority-rule supertrees. *Syst. Biol.*, 56(3):445–452, 2007.

[10] L. R. Foulds and R. L. Graham. The Steiner problem in phylogeny is NP-complete. *Advances in Applied Mathematics*, 3:43–49, 1982.

[11] D. H. Huson, S. M. Nettles, and T. J. Warnow. Disk-covering, a fast-converging method for phylogenetic tree reconstruction. *JOURNAL OF COMPUTATIONAL BIOLOGY*, 6(3):369–386, 1999.

[12] T. Jiang, P. Kearney, and M. Li. A polynomial-time approximation scheme for inferring evolutionary trees from quartet topologies and its applications. *SIAM J. Comput.*, 30(6):1924–1961, 2001.

[13] M. Kennedy and R. Page. Seabird supertrees: combining partial estimates of procellariiform phylogeny. *The Auk*, 119:88–108, 2002.

[14] K. Liu, C. Linder, R. Suri, and T. Warnow. Multiple sequence alignment: a major challenge to large-scale phylogenetics. *PLoS Currents: Tree of Life*, 2010.

[15] K. Liu, T. J. Warnow, M. T. Holder, S. Nelesen, J. Yu, A. Stamatakis, and C. R. Linder. SATé-II: Very fast and accurate simultaneous estimation of multiple sequence alignments and phylogenetic trees. *Syst Biol*, 2011. In press.

[16] M. McMahon and M. Sanderson. Phylogenetic supermatrix analysis of GenBank sequences from 2228 papilionoid legumes. *Syst. Biol.*, 55(5):818–836, 2006.

[17] K. C. Nixon. The parsimony ratchet, a new method for rapid parsimony analysis. *Cladistics*, 15(4):407–414, 1999.

[18] M. A. Ragan. Phylogenetic inference based on matrix representation of trees. *Mol. Phylogenet. Evol.*, 1:53–58, 1992.

[19] V. Ranwez, V. Berry, A. Criscuolo, P. Fabre, S. Guillemot, C. Scornavacca, and E. Douzery. PhySIC: a veto supertree method with desirable properties. *Syst. Biol.*, 56(5):798–817, 2007.

[20] V. Ranwez, A. Criscuolo, and E. J. Douzery. SuperTriplets: a triplet-based supertree approach to phylogenomics. *Bioinformatics*, 26(12):i115–i123, 2010.

[21] S. Roch. A short proof that phylogenetic tree reconstruction by maximum likelihood is hard. *IEEE Trans. Comput. Biol. and Bioinformatics*, 3(1):92–94, 2006.

[22] F. Ronquist and J. Huelsenbeck. MrBayes 3: Bayesian phylogenetic inference under mixed models. *Bioinformatics*, 19:1572–1574, 2003.

[23] U. Roshan, B. M. E. Moret, T. L. Williams, and T. Warnow. Performance of supertree methods on various data-set decompositions. In O. R. P. Bininda-Emonds, editor, *Phylogenetic Supertrees: combining information to reveal The Tree Of Life*, pages 301–328. Kluwer Academic, Dordrecht, the Netherlands, 2004.

[24] A. Stamatakis. RAxML-NI-HPC: Maximum likelihood-based phylogenetic analyses with thousands of taxa and mixed models. *Bioinformatics*, 22:2688–2690, 2006.

[25] M. Steel. The complexity of reconstructing trees from qualitative characters and subtrees. *J Classification*, 9(1):91–116, 1992.

[26] M. Steel and A. Rodrigo. Maximum likelihood supertrees. *Syst. Biol.*, 57(2):243–250, 2008.

[27] M. S. Swenson, F. Barbançon, C. Linder, and T. Warnow. A simulation study comparing supertree and combined analysis methods using SMIDGen. In *Proceedings of the 2009 Workshop on Algorithms in Bioinformatics (WABI)*, pages 333–344, 2009.

[28] M. S. Swenson, F. Barbançon, C. Linder, and T. Warnow. A simulation study comparing supertree and combined analysis methods using SMIDGen. *Algorithms for Molecular Biology*, 5(8), 2010.

[29] M. S. Swenson, R. Suri, C. Linder, and T. Warnow. An experimental study of Quartets MaxCut and other supertree methods. In *Proceedings of the 2010 Workshop on Algorithms in Bioinformatics (WABI)*, 2010.

[30] M. S. Swenson, R. Suri, C. Linder, and T. Warnow. An experimental study of Quartets MaxCut and other supertree methods. *Algorithms for Molecular Biology*, 2010.

[31] M. S. Swenson, R. Suri, C. R. Linder, and T. Warnow. SuperFine: Fast and accurate supertree estimation. *Syst Biol*, 2011. In press.

[32] D. L. Swofford. PAUP*. phylogenetic analysis using parsimony (*and other methods). version 4. Sinauer Associates, 2003.

[33] M. Wojciechowski, M. Sanderson, K. Steele, and A. Liston. Molecular phylogeny of the “temperate herbaceous tribes” of papilionoid legumes: a supertree approach. *Adv. Legume Syst.*, 9:277–298, 2000.