Mathematics of Evolution and Phylogeny

Edited by Olivier Gascuel

CLARENDON PRESS · OXFORD
2004

CONTENTS

12

RECONSTRUCTING PHYLOGENIES FROM GENE-CONTENT AND GENE-ORDER DATA

Bernard M.E. Moret, Jijun Tang and Tandy Warnow

Abstract

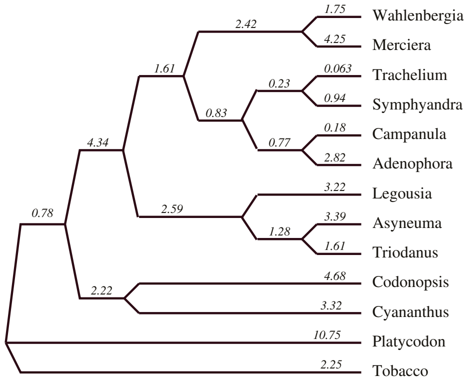
Gene-order data have been used successfully to reconstruct organellar phylogenies; they offer low error rates, the potential to reach farther back in time than through DNA sequences (because genome-level events are rarer than DNA point mutations), and immunity from the so-called gene-tree vs. species-tree problem (caused by the fact that the evolutionary history of specific genes is not isomorphic to that of the organism as a whole). They have also provided deep mathematical and algorithmic results dealing with permutations and shortest sequences of operations on these permutations. Recent developments include generalizations to handle insertions, duplications, and deletions, scaling to large numbers of organisms, and, to a lesser extent, to larger genomes; and the first Bayesian approach to the reconstruction problem. We survey the state-of-the-art in using such data for phylogenetic reconstruction, focusing on recent work by our group that has enabled us to handle arbitrary insertions, duplications, and deletions of genes, as well as inversions of gene subsequences. We conclude with a list of research questions (mathematical, algorithmic, and biological) that will need to be addressed in order to realize the full potential of this type of data.

12.1 Introduction: Phylogenies and Phylogenetic Data

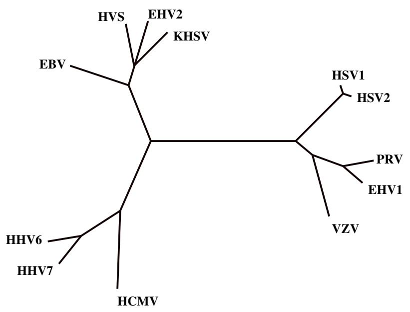
12.1.1 *Phylogenies*

A phylogeny is a reconstruction of the evolutionary history of a collection of organisms. It usually takes the form of a tree, where modern organisms are placed at the leaves and edges denote evolutionary relationships. In that setting, "species" correspond to edge-disjoint paths. Figure 12.1 shows three phylogenetic trees, in different display formats.

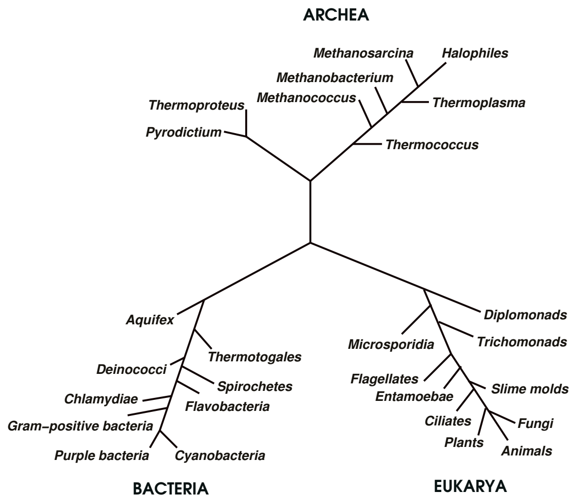
Phylogenies have been and still are inferred from all kinds of data: from geographic and ecological, through behavioral, morphological, and metabolic, to the current data of choice, namely molecular data [74]. Molecular data have the significant advantage of being exact and reproducible, at least within experimental error, not to mention fairly easy to obtain. Each nucleotide in a DNA or RNA sequence (or each codon) is, by itself, a well defined *character*, whereas morphological data (a flower, a dinosaur bone, etc.), for instance, must first be encoded into characters, with all the attending problems of interpretation, discretization, etc.



(a) 12 plants from the Campanulaceae family [14]



(b) Herpes viruses affecting humans [43]



(c) One possible high-level view of the Tree of Life

FIG. 12.1. Various phylogenetic trees, in different formats

The predominant molecular data have been and continue to be sequence data: DNA or RNA nucleotide or codon sequences for a few genes. A promising new kind of data is gene-order data, where the sequence of genes on each chromosome is specified.

Sequence Data In sequence data, characters are individual positions in the string and so can assume one of a few states: 4 states for nucleotides or 20 states for amino-acids. Such data evolve through *point mutations*, i.e., changes in the state of a character, plus *insertions* (including *duplications*) and *deletions*. Figure 12.2 shows a simple evolutionary history, from the ancestral sequence at the root to modern sequences at the leaves, with evolutionary events occurring on each edge. Note that this history is incomplete,

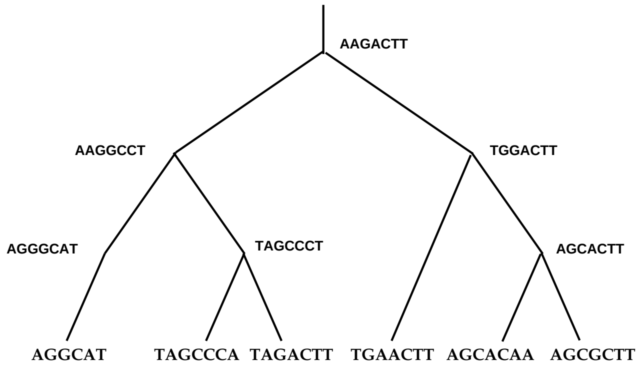


FIG. 12.2. Evolving sequences down a given tree topology

as it does not detail the events that have taken place along each edge of the tree. Thus, while one might reasonably conclude that, in order to reach the leftmost leaf, labeled AGGCAT, from its parent, labeled AGGGCAT, one should infer the deletion of one nucleotide (one of the three G's in the parent), a more complex scenario may in fact have unfolded. If one were to compare the leftmost leaf with the rightmost one, labeled AGCGCTT, one could account for the difference with two changes: starting with AGGCAT, insert a C between the two G's to obtain AGCGCAT, then mutate the penultimate A into a T. Yet the tree itself indicates that the change occurred in a far more complex manner: the path between these two leaves in the tree goes through the series of sequences

AGGCAT↔AGGGCAT↔AAGGCCT↔AAGACTT↔TGGACTT↔AGCACTT↔AGCGCTT

and each arrow in this series indicates at least one evolutionary event.

Preparing sequence data for phylogenetic analysis involves the following steps: (i) finding *homologous* genes (i.e., genes that have evolved from a common ancestral gene—and most likely fulfill the same function in each organism) across all organisms; (ii) retrieving and then aligning the sequences for these genes (typical genes yield sequences of several hundred base pairs) across the entire set of organisms, in order to identify gaps (corresponding to insertions or deletions) and matches or mutations; and finally (iii) deciding whether to use all available data at once for a *combined analysis* or to use each gene separately and then *reconcile* the resulting trees.

Sequence data are by far the most common form of molecular data used in phylogenetic analyses. The main reason is simply availability: large amounts of data are easily available from databases such as GenBank, along with search tools (such as BLAST) and annotations; moreover, the volume of such data grows at an exponential pace—indeed, it is outpacing the growth in computer speed (Moore's law). A second reason is the widespread availability of analysis tools for such data: packages such as `PAUP*` [73], MacClade [37], Mesquite [40], Phylip [18], MEGA [32], MrBayes [28], and TNT [21], all available either freely or for a modest fee, are in widespread use and have provided biologists with satisfactory results on many datasets. Finally, the success of these packages is due in good part to the fact that sequence evolution has long been studied, both in terms of the biochemistry of nucleotides and of the biological mechanisms of change, so that accepted models of sequence evolution provide a reasonable framework within which to define computational optimization problems.

Sequence data do suffer from a number of problems. A fairly minor problem is simple experimental errors: in the process of sequencing, some base pairs are misidentified (miscalled), currently with a probability on the order of 10^{-2}. A more serious limitation is the relatively fast pace of mutation in many regions of the genome; combined with the fact that each position can assume one of only a few values, this fast pace results in *silent changes*—changes that are subsequently reversed in the course of evolution, leaving no trace in modern organisms. (Using amino-acid sequences, with 20 possible states per character, only modestly alleviates this problem.) In consequence, sequence data must be selected to fit the problem at hand: very stable regions to reconstruct very old events, highly variable regions to reconstruct very recent history, etc. This specialized nature may cause difficulties when attempting to reconstruct a phylogeny that includes both recent and ancient events, since such an attempt would require mixing variable and conserved regions in the analysis, triggering the next and most important problem. The evolution of any given gene (or region of the sequence) need not be identical to that of the organism—this is the *gene tree vs. species tree* problem [39, 57]. Thus a combined analysis, based on the use of all available genes, risks running into internal contradictions and the loss of resolution, whereas one based on individual genes will typically yield different trees for the different genes, trees that must then be reconciled through a process known as *lineage sorting*. Sequence data also suffer from computational problems: most prominently, the problem of multiple sequence alignment is currently only poorly solved—indeed, most systematists will align sequence data by hand, or at least edit by hand the alignments proposed by the software. Less importantly, at least in a relative sense, current phylogenetic reconstruction methods used with sequence data do not scale well, whether in terms of accuracy or running time.

Gene-Content and Gene-Order Data The data here are lists of genes in the order in which they are placed along one or more chromosomes. Nucleotide data are not part of this picture: instead, each gene along a chromosome is identified by some name, a name shared with its homologs on other chromosomes (or, for that matter, on the same chromosome, in case of gene duplications). The entire gene order forms a *single* character, but one that can assume a huge number of states—a chromosome with n genes presents a character with $2^n \cdot n!$ states (the first term is for the strandedness of each gene and the second for the possible permutations in the ordering). A typical single circular chromosome for the chloroplast organelle of a *Guillardia* species (taken from the NCBI database) is shown in Fig. 12.3. A gene order evolves through *inversions*, sometimes also called reversals (well documented in chloroplast organelles [31, 58]), and perhaps also *transpositions* and *inverted transpositions* (strongly suspected in mitochondria [7,8]); these three operations are illustrated in Fig. 12.4. (Other, more complex rearrangements may well be possible, particularly in the context of DNA repair of radiation damage.) These operations do not affect the gene content of the chromosome.

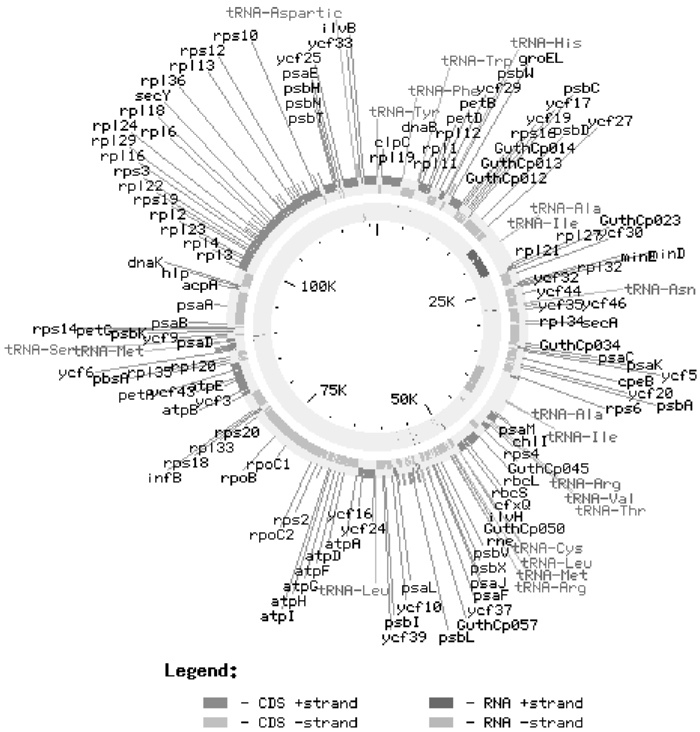


FIG. 12.3. The chloroplast chromosome of Guillardia (from NCBI)

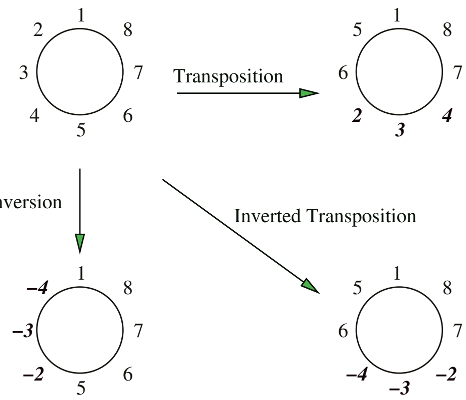


FIG. 12.4. The three rearrangement operations operating on a single circular chromosome, all operating on the gene subsequence $(2,3,4)$

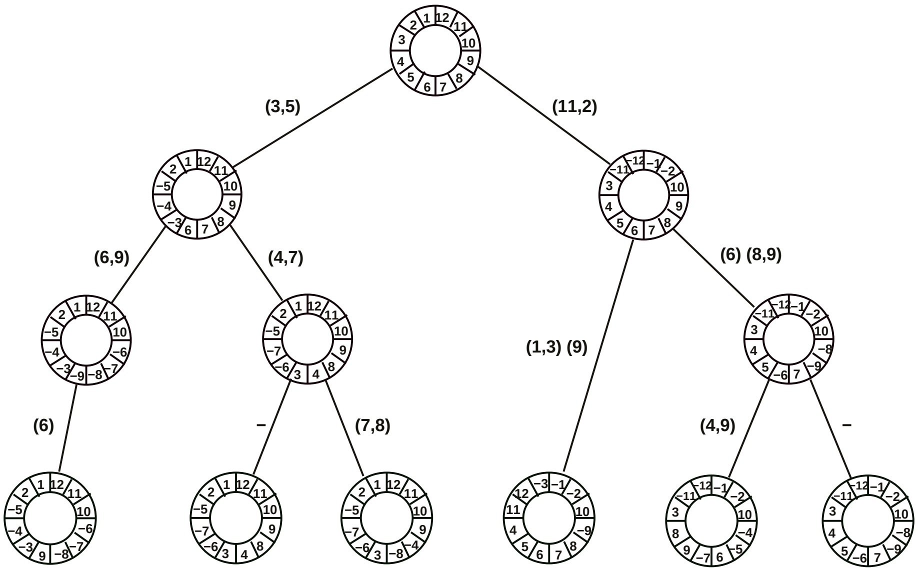


FIG. 12.5. Evolving gene orders down a given tree topology; each edge is labelled by the inversions that took place along it

In the case of multiple chromosomes, other operations come into play. One such operation is *translocation*, which moves a piece of one chromosome into another—in effect, it is a transposition between chromosomes. Other operations that are applicable to multiple chromosome evolution include *fusion*, which merges two chromosomes into one, and *fission*, which divides a single chromosome into two. In multichromosomal organisms, colocation of genes on the same chromosome, or *synteny*, is an important evolutionary attribute and has been used in phylogenetic reconstruction [54, 67, 68]. Finally, two additional evolutionary events affect both the gene content and, indirectly, the gene order: *insertions* (including *duplications*) and *deletions* of single genes or sequences of genes.

In order to conduct a phylogenetic analysis based on gene-order data, we must identify homologous genes (including duplications) within and across the chromosomes. As the system under study is much more complex than sequence data, we may also have to refine the model to fit specific collections of organisms; for instance, bacteria often have conserved *clusters* of genes, or *operons*—genes that stay together throughout evolution, but not in any specific order—, while most chloroplast organelles exhibit a characteristic partition of their chromosome into four regions, two of which are mirror images of each other (the "inverted repeat" structure). Figure 12.5 shows a typical evolutionary scenario based on inversions alone; compare with Fig. 12.2.

The use of gene-order and gene-content data in phylogenetic reconstruction is relatively recent and the subject of much current research. Such data present several advantages: (i) because the entire genome is studied at once, there is no gene tree vs. species

Table 12.1 *Existing whole-genome data ca. 2003 (approximate values)*

Type	Attributes	Numbers
Animal mitochondria	1 chromosome, 40 genes	250
Plant chloroplast	1 chromosome, 140 genes	100
Bacteria	1–2 chromosomes, 500–4,000 genes	50
Eukaryotes	3–30 chromosomes, 2,000–30,000 genes	10

tree problem; (ii) there is no need for alignment; and (iii) gene rearrangements and duplications are much rarer events than nucleotide mutations (they are "rare genomic events" in the sense of Rokas and Holland [61]) and thus enable us to trace evolution farther back than sequence data.

On the other hand, there remain significant challenges. Foremost among them is the lack of data: mapping a full genome, while easier than sequencing the full genome, remains much more demanding than sequencing a few genes. Table 12.1 gives a rough idea of the state of affairs around 2003. The bacteria are not well sampled: for obvious reasons, most of the bacteria sequenced to date are human pathogens. The eukaryotes are the model species chosen in genome projects: human, mouse, fruit fly, worm, mustard plant, yeast, etc.; although their number is quickly growing (with several more mammalian genomes nearing completion), coverage at this level of detail will probably never exceed a small fraction of the total number of described organisms.

This lack of data in turn gives rise to another problem: there is no good model of evolution for the gene-order data—for instance, we still do not have firm evidence for transpositions, much less any notion of relative prevalence of the various rearrangement, duplication, and loss events. This lack of a good model combines with a third problem, the extreme (at least in comparison with sequence data) mathematical complexity of gene orders, to create major computational challenges.

Sequence vs. Gene-Order Data Table 12.2 summarizes the characteristics of sequence data and gene-order data. At present, there is every reason to expect that whole-genome data will remain limited to a small subset of the organisms for which we will have some sequence data: sequencing one gene is fast and inexpensive, whereas sequencing a complete eukaryotic genome is a major enterprise. Yet gene-order data remain worth studying: not only will the advantages discussed earlier enable us to provide valuable

Table 12.2 *Main attributes of sequence and gene-order data*

	Sequence	Gene-Order
evolution	fast	slow
data type	a few genes	whole genome
data quantity	abundant	sparse
# char. states	tiny	huge
models	good	primitive
computation	easy	hard

cross-checking for sequence-derived phylogenies (or even provide a framework around which to build a sequence-derived phylogeny), but the rapid pace of change in genomic technology may yet enable us to sequence entire genomes rapidly and at low cost.

12.1.2 *Phylogenetic Reconstruction*

Methods for phylogenetic reconstruction from sequence data can be roughly classified as (i) *distance-based* methods, such as neighbor-joining; (ii) *parsimony-based* methods, such as implemented in PAUP*, Phylip, MEGA, TNT, etc.; and (iii) *likelihood-based* methods, including Bayesian methods, such as implemented in PAUP*, Phylip, fastDNAml [56], MrBayes, GAML [35], etc. In addition, *metamethods* can be used to scale up any of these three *base* methods: metamethods decompose the data in various ways and rely on one or more base methods to reconstruct trees for the subsets they produce. Metamethods include *quartet-based* methods (see, e.g., [70]) and *disk-covering* methods [29, 30, 55, 62, 76]—about which we will have more to say. We will use the same categories when discussing methods for reconstruction from gene-order data, so we give a brief characterization of each category.

Phylogenetic distances As our discussion of the phylogeny presented in Fig. 12.2 indicates, the distance between two taxa (as represented by sequence or gene-order data) can be defined in several ways. First, we have the *true evolutionary distance*, that is, the actual number of evolutionary events (mutations, deletions, etc.) that separate one datum (gene or genome) from the other. This is the distance measure we would really want to have, but of course it cannot be inferred—as our earlier discussion made clear, we cannot infer such a distance even when we know the correct phylogeny and have correctly inferred ancestral data (at internal nodes of the tree). What we can define precisely and compute (in most cases) is the *edit distance*, the minimum number of permitted evolutionary events that can transform one datum into the other. Since the edit distance will invariably underestimate the true evolutionary distance, we can attempt to *correct* the edit distance according to an assumed model of evolution in order to produce the *expected true evolutionary distance*, or at least an approximation thereof—see Chapter 6 in this volume for a discussion of distance correction.

Distance-based methods Distance-based methods use edit distances or expected true evolutionary distances and typically proceed by grouping (as siblings) taxa (or groups of taxa) whose normalized pairwise distance is smallest. They usually run in *low polynomial time*, a significant advantage over all other methods. Most such methods only reconstruct the tree topology—they do not estimate the character states at internal nodes within the tree. The prototype in this category is the *Neighbor-Joining* (NJ) method [63], later refined to produce `BioNJ` [20] and *Weighbor* [10]. When each entry in the distance matrix equals the true evolutionary distance (i.e., the distance along the unique path between these two taxa in the true tree), NJ is guaranteed to produce the true tree; moreover, NJ is statistically consistent—that is, it produces the true tree with probability 1 as the sequence length goes to infinity [3], under those models for which statistically consistent distance estimators exist. (See also Chapter 1 in this volume for a discussion of statistical consistency.)

Parsimony-based methods These methods aim to minimize the total *number of character changes* (which can be weighted to reflect statistical evidence). Characters are assume to evolve independently—so each character makes an independent contribution to the total. In order to evaluate that contribution, parsimony methods all reconstruct ancestral sequences at internal nodes. In contrast to NJ and likelihood methods, parsimony methods are not always statistically consistent. However, it can be argued that trees reconstructed under parsimony are not substantially less accurate than trees reconstructed using statistically consistent methods, given the restriction on the amount of data and the lack of fit between models and real data. Finding the most parsimonious tree is known to be NP-hard, but scoring a single fixed tree is easily accomplished in linear time; at present, provably optimal solutions are limited to datasets of 20–30 taxa, while good approximate solutions can be obtained for datasets of several hundred taxa; the latest results from our group [62] indicate that we can achieve the same quality of reconstruction on tens of thousands of taxa within reasonable time.

Likelihood-based methods Likelihood-based methods assume some specific model of evolution and attempt to find the tree, and its associated model parameters, which together maximize the probability of the observed data. Thus a likelihood method must both *estimate model parameters* on a given fixed tree and also search through tree space to find the best tree. Chapter 2 in this volume discusses likelihood methods.

Likelihood-based methods are usually (but, perhaps surprisingly, not always) statistically consistent, although, of course, that consistency is meaningless if the chosen model does not match the biological reality. Likelihood methods are the slowest of the three categories and also prone to numerical problems, because the likelihood of typical tree is extremely small—with just 20 taxa, the average likelihood in the order of 10^{-21}, going down to 10^{-75} with 50 taxa. Identifying the tree of maximum likelihood is presumably NP-hard, although no proof has yet been devised; indeed, even computing the likelihood a fixed tree under a fixed model cannot currently be done in polynomial time (see, e.g. [71]). Thus optimal solutions are limited to trees with fewer than 10 taxa, while good approximations are possible for perhaps 100 taxa.

Bayesian methods deserve a special mention among likelihood-based approaches; they compute the posterior probability that the observed data would have been produced by various trees (in contrast to a true maximum likelihood method, which computes the probability that a fixed tree would produce various kinds of data at its leaves). Their implementation with Markov chain Monte-Carlo (MCMC) algorithms often run significantly faster than pure ML methods; moreover, the moves through state space can be designed to enhance convergence rates and speed up the execution. Chapter 3 in this volume discusses Bayesian approaches.

12.2 Computing with Gene-Order Data

As indicated earlier, gene-order data present significant mathematical challenges not encountered when dealing with sequence data. Many evolutionary events may affect the gene order and gene content of a genome; and each of these events creates its own challenges, not least of which is the computation of a pairwise genomic distance. Armed

```
G1=(1  2  3  4  5  6  7  8)

G2=(1  2 -5 -4 -3  6  7  8)
```

FIG. 12.6. Breakpoints

with algorithms for computing distances, we can proceed to phylogenetic reconstruction, starting with scoring a single tree in terms of its total evolutionary distance.

12.2.1 *Genomic distances*

We begin with distances between genomes with equal gene content: in this case, the only operations allowed are rearrangements.

Breakpoint distance A *breakpoint* is an adjacency present in one genome, but not in the other. Figure 12.6 shows two breakpoints between two genomes—note that the gene subsequence 3 4 5 is identical to -5 -4 -3, since the latter is just the former read on the complementary strand. The *breakpoint distance* is then the number of breakpoints present; this measure is easily computed in linear time, but it does not directly reflect rearrangement events—only their final outcome. In particular, it typically underestimates the true evolutionary distance even more than an edit distance does.

Inversion distance Given two signed gene orders of equal content, the *inversion distance* is simply the edit distance when inversion is the only operation allowed. Even though we have to consider only one type of rearrangement, this distance is very difficult to compute. For *unsigned* permutations, in fact, the problem is NP-hard. For signed permutations, it can be computed in linear time [4], using the deep theoretical results of Hannenhalli and Pevzner [23].

The algorithm is based on the *breakpoint graph*. Refer to Fig. 12.7 for an illustration. We assume without loss of generality that one permutation is the identity. We represent gene i by two vertices, $2i-1$ and $2i$, connected by an edge; think of that edge as oriented from $2i-1$ to $2i$ when gene i appears with positive sign, but oriented in the reverse direction when gene i appears with negative sign. Now we connect these edges with two further sets of edges, one for each genome—one represents the identity (i.e., it simply connects vertex j to vertex $j+1$, for all j) and is shown with dashed arcs in Fig. 12.7, and the other represents the other genome and is shown with solid edges in the figure. The crucial concept is that of alternating cycles in this graph, i.e., cycles of even length in which every odd edge is a dashed edge and every even one is a solid edge. Overlapping

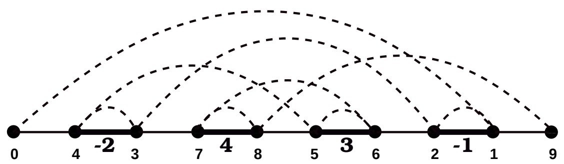


FIG. 12.7. The breakpoint graph for the signed permutations of Fig. 12.6

cycles in certain configurations create structures known as *hurdles* and a very unique configuration of such hurdles is known as a *fortress*. Hannenhalli and Pevzner proved that the inversion distance between two signed permutations of n genes is given by

$$n - \#\text{cycles} + \#\text{hurdles} + (\text{fortress})$$

In Chapter 10 in this volume, Bergeron *et al.* offer an alternate formulation of this result, within a framework based on certain nested intervals.

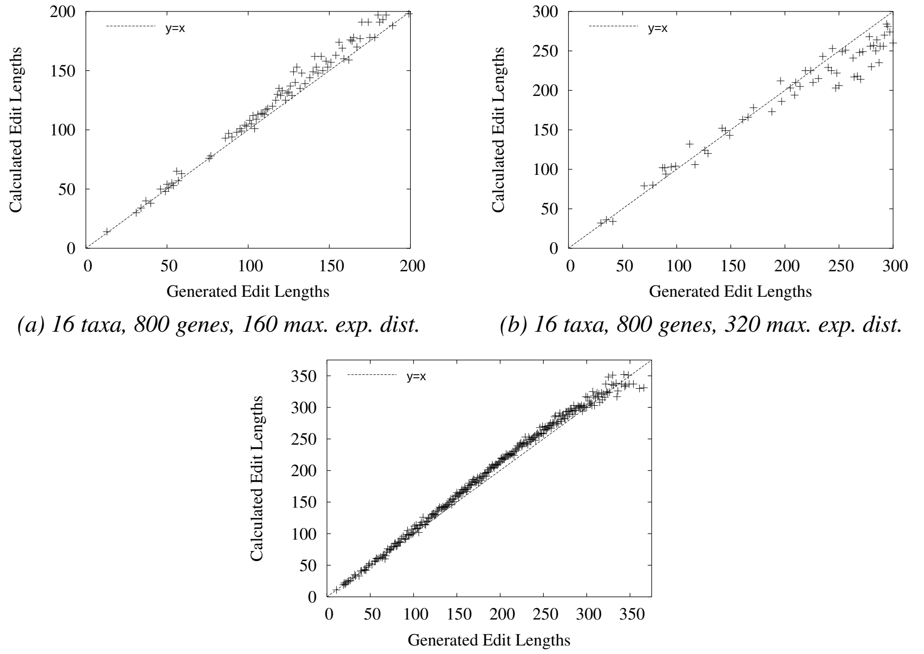
Generalized gene-order distance The restriction that no gene be duplicated and that all genomes contain exactly the same set of genes is clearly unrealistic, even in the case of organellar genomes. However, accounting for additional evolutionary events such as duplications, insertions, and deletions is proving very difficult. One extension has been present since the beginning: in the second of their two seminal papers [24], Hannenhalli and Pevzner showed that their framework (cycles, hurdles, etc.) could account for both insertions and multichromosomal events, namely translocations, fusions, and fissions. Bourque and Pevzner [9] designed a heuristic approach to phylogenetic reconstruction for multichromosomal organisms under inversions, translocations, and fissions and fusions, based upon the work of Tesler [78]; they used the `GRAPPA` core algorithm for inversion and confirmed the findings of Moret *et al.* [48] that inversion-based reconstruction of ancestral genomes outperforms breakpoint-based reconstruction of same.

More recently, El-Mabrouk [17] showed how to compute a minimum edit sequence in polynomial time when both inversions and deletions are allowed; Liu and Moret [36] then showed that the distance itself can be computed in linear time. Because edit sequences are symmetric, these results also apply to combinations of inversions and *nonduplicating* insertions. In the same paper, El-Mabrouk showed that her method could provide a bounded approximation to the edit distance in the presence of both deletions and (nonduplicating) insertions. Sankoff [64] had earlier proposed a heuristic approach to the problem of duplications, suggesting that a single copy—the *exemplar*—be kept, namely that copy whose use minimized the number of other operations. Unfortunately, finding the exemplar, even for a single gene, is an NP-hard problem [11]. Marron *et al.* [41] gave the first bounded approximation algorithm for computing an edit sequence (or distance) in the presence of inversions, duplications, insertions, and deletions; a similar approach was used by Tang *et al.* [77] in the context of phylogenetic reconstruction. Most recently, Swenson *et al.* [72] gave an extension of the algorithm of Marron *et al.*, one that closely approximates the true evolutionary distance between two arbitrary genomes under any combinations of inversions, insertions, duplications, and deletions; they also showed that this distance measure is sufficiently accurate to enable accurate phylogenetic reconstruction by simply using neighbor-joining on the distance matrix.

Work on transposition distances has been limited to equal-content genomes with no duplications and, even then, only to approximations, all with guaranteed ratio 1.5. The first approximation is due to Bafna and Pevzner [5], using much the same framework defined for the study of inversions; the approach was recently simplified, then extended to include inverted transpositions by Hartman [25, 26]. Work on transposition distance is clearly lagging behind work on inversion distance and remains to be integrated with it and extended to genomes with unequal content.

In a different vein, Bergeron and Stoye [6] defined a distance estimate based on the number and lengths of conserved gene clusters; this distance is well suited to prokaryotic genomes (where gene clusters and operons are common), but it still requires that duplicate genes be removed.

Estimating true pairwise evolutionary distances We give a brief overview of the results of Swenson *et al.* [72]. In earlier work [41], the same group had shown that any shortest edit sequence could always be rewritten to that all insertions and duplications take place first, followed by all inversions, followed by all deletions. In order to estimate pairwise evolutionary distances between arbitrary genomes, it remains to handle duplications; this is done gene by gene by computing a mapping from the genome with the smaller number of copies of that gene to that with the larger number of copies, using simple heuristics. Deletions and inversions are computed quite accurately, using extensions to the work of El-Mabrouk [17], while insertions (which now include any "excess" duplicates not matched in the first phase) are computed by retracing the sequence of inversions and deletions. The result is a systematic overestimate of the edit distance, but a very accurate estimate of the true evolutionary distance. Figure 12.8 presents some results from simulations in which evolutionary events were selected through a mix of 70% inversions, 16% deletions, 7% insertions, and 7% duplications, with inversions



(a) 16 taxa, 800 genes, 160 max. exp. dist.

(b) 16 taxa, 800 genes, 320 max. exp. dist.

(c) 57 taxa, 1,200 genes, 240 max exp. dist.

FIG. 12.8. Generated pairwise edit length vs. reconstructed length for three simulated datasets; an exact estimate follows the indicated line $y = x$.

having a mean length of 20 and a standard deviation of 10, and deletions, insertions, and duplications having a mean length of 10 with a standard deviation of 5. The top two examples come from datasets of 16 taxa with 800 genes, with expected pairwise distances of 20 through 160 events (left) and 40 through 320 events (right); the bottom example comes from a dataset of 57 taxa with 1,200 genes and expected pairwise distances from 20 to 280 events. The distance computation, which has a randomized component (to break ties in the assignment of duplicate genes), was run 10 times with different seeds. The figure indicates clearly that the distance estimate is highly accurate up to saturation, which occurs only at very large distances (around 250 events for a genome of 800 genes).

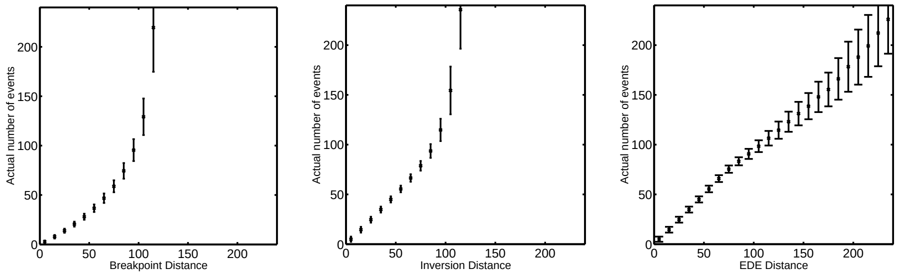
12.2.2 *Evolutionary models and distance corrections*

In order to use gene-order and gene-content data, we need a reasonable model of evolution for the gene order of a chromosome—and here we lack sufficient data for the construction of strong models. To date, biologists have strong evidence for the occurrence of inversions in chloroplasts—and have at least two possible models for the creation of inversions (one through DNA breakage and misrepair, the other through loops traversed in the wrong order during replication). Since DNA breakage is relatively common and particularly pronounced as a result of radiation damage, other rearrangements due to misrepair appear at least possible. Sankoff [65] has given statistical evidence for a distinction between short and long inversions: short inversions tend to preserve clusters (and thus could be common in prokaryotes), whereas long inversions tend to preserve runs of genes (and thus could be more common in eukaryotes); in a subsequent study of prokaryotic data [34], an *ad hoc* computational investigation gave additional evidence that short inversions play a significant role in prokaryotic organisms. However, even if we limit ourselves to (short and long) inversions, the respective probabilities of these two events remain unknown.

While we do not yet have a strong model of genome evolution through rearrangements, we do know that edit distances must underestimate true evolutionary distances, especially as the distances grow large. As is discussed in detail in Chapter 13 in this volume, it is possible to devise effective schemes to convert the edit distance into an estimate, however rough, of the the true evolutionary distance. Figure 12.9 illustrates the most successful of these attempts: working from a scenario of uniformly distributed inversions, Moret *et al.* [49] collected data on the inversion distance vs. the number of inversions actually used in generating the permutations (the middle plot), then produced a formula to correct the underestimate, with the result, the *EDE* distance, shown in the third plot. (The first plot shows that the breakpoint distance is even more subject to underestimation than the inversion distance.) The use of EDE distances in lieu of inversion distances leads to more accurate phylogenetic reconstructions with both distance methods and parsimony methods [49, 50, 79, 80].

12.2.3 *Reconstructing ancestral genomes*

Reconstructing ancestral genomes is an integral part of both parsimony- and Bayesian-based reconstruction methods and may also have independent interest. In a parsimony



breakpoint distance *inversion distance* *EDE corrected distance*

FIG. 12.9. Edit distances vs. true evolutionary distances and the EDE correction

context, we want to reconstruct a signed gene order at each internal node in the tree so as to minimize the sum of genomic distances over all edges of the tree. Unfortunately, this optimization problem is NP-hard even for just three leaves and for the simplest of settings—equal gene content, no duplication, and breakpoint distance [59] or inversion distance [12]. Computing such a gene order for three leaves is the *median problem* for signed genomes: given three genomes, produce a new genome that will minimize the sum of the distances from it to the other three. In the case of breakpoint distances, Sankoff and Blanchette [66] showed how to convert this problem to the *Travelling Salesperson Problem*; Figure 12.10 illustrates the process. Each gene gives rise to a pair of cities connected by an edge that must be included in any solution; the distance between any two cities not forming such pairs is simply the number of genomes in which the corresponding pair of genes is not consecutive (and thus varies from 0 to 3, a limited range that was put to good use in the fast GRAPPA implementation [53]).

No equivalently simple formulation in terms of a standard optimization problem is known for more general genomic distances. Yet even the simple inversion distance gives rise to significantly better results than the breakpoint distance, in terms of computational demands and topological accuracy [48, 49, 51, 76] as well as of the accuracy of reconstructed ancestral genomes [9, 48]. For inversion distances, exact algorithms have been

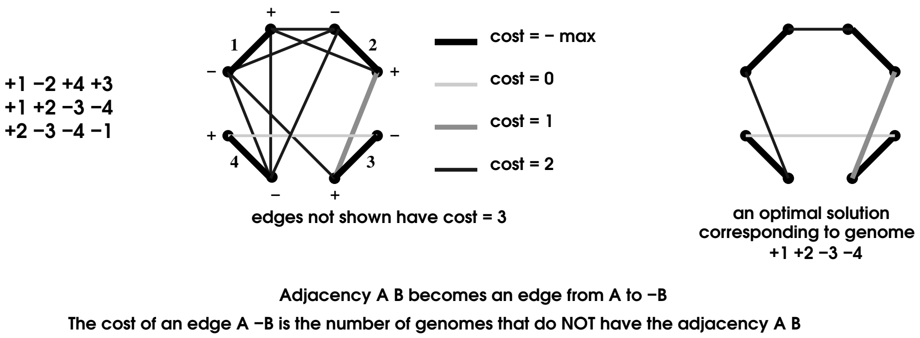


FIG. 12.10. Reducing the breakpoint median to a TSP instance

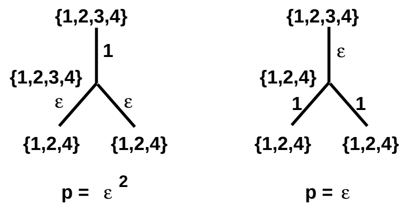
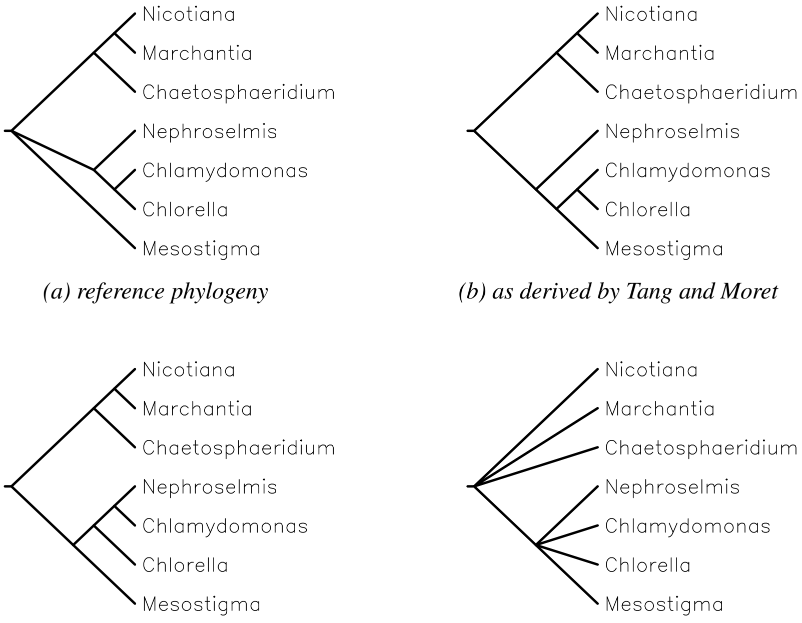


FIG. 12.11. Determining the gene content of the median

proposed [13, 69] that work well for small distances (of fewer than 15 inversions). Tang and Moret [75] showed that the median problem under inversions, deletions, and insertions or duplications could be solved exactly for small numbers of deletions and duplications, using a few simple assumptions; they recently extended that work for somewhat larger changes in gene content [77]. Their approach first determines the gene content of the median, then computes an ordering through those genes via an optimization procedure. The basic assumptions are that (i) no change is reversed and (ii) changes are independent and of low probability. These two assumptions, common in phylogenetic work (see, e.g., [38, 42]), imply that simultaneous identical changes on two edges are vanishingly unlikely compared to the reverse change on the third edge—since the simultaneous changes have a probability on the order of ε^2, for a small ε, compared to a probability of ε for a change on a single edge, as illustrated in Fig. 12.11. The results obtained by Tang and Moret on a small, but difficult dataset of just seven chloroplast genomes from red and green algae and land plants are shown in Fig. 12.12. Part (a) shows the reference phylogeny obtained through combined likelihood and maximum parsimony analyses of the codon sequences of several cpDNA genes; it should be noted that the placement of Mesostigma is unclear from the data. Part (b) shows the phylogeny obtained by Tang and Moret, which is completely consistent with the reference phylogeny. Part (c) shows the phylogeny obtained by using the simple neighbor-joining method on the distance matrix computed from the seven genomes with equalized gene content: the method produced a false positive. Finally, part (d) shows the tree built by using breakpoint distances on equalized gene contents: not that the tree is nearly a star, with just one resolved edge.

In the presence of very large differences in gene content and of many duplicates, the problem is much harder. For one thing, given three genomes with these characteristics, the number of possible optimum medians is very large—indicating that a biologically sound reconstruction will require external constraints to select from these many choices. Knowing the direction of time flow (as is the case after the tree has been rooted) simplifies the problem somewhat—at least it makes the question of gene content much simpler to resolve [16], but it is fair to say that, at present, we simply lack the tools to reconstruct ancestral data for complex nuclear genomes.

In a completely different vein, El-Mabroul (see Chapter 11 in this volume) has shown how to reconstruct ancestral genomes in the presence of a single duplication event, one, however, that duplicated the entire genome just once.



(a) reference phylogeny *(b) as derived by Tang and Moret*

(c) neighbor-joining *(d) breakpoint phylogeny*

FIG. 12.12. Phylogenies on the seven taxon cpDNA dataset [77]

12.3 Reconstruction from Gene-Order Data

Phylogenetic reconstruction methods from gene-order data fall within the same general categories as methods for sequence data, to wit: (i) distance-based methods, (ii) parsimony-based methods, and (iii) likelihood-based methods, all with the possibility of using a metamethod on top of the base method. In Chapter 13 in this volume, Wang and Warnow give a detailed discussion of distance-based methods. Likelihood methods are represented to date by a single effort, from Larget *et al.* [33], in which a Bayesian approach showed evidence of success on a couple of fairly easy datasets; the same approach, however, failed to converge on a harder dataset analyzed by Tang *et al.* [77]. We thus focus here on approaches based on parsimony, which have seen more development. These approaches fall into two subcategories: encoding methods, which reduce the gene-order problems to sequence problems, and direct methods, which run optimization algorithms directly on the gene-order data.

12.3.1 *Encoding gene-order data into sequences*

As we shall see in Section 12.3.2, direct optimization approaches have running times that are exponential in both the number of genomes and the number of genes, so that analyses of even small datasets (containing only ten or twenty genomes) may remain computationally intractable. Therefore an approach that, while remaining exponential in the number of genomes, takes time polynomial in the number of genes, may be of significant interest. Since sequence-based methods have such characteristics, a simple idea is to reduce the gene-order data to sequence data through some type of encoding. Our group developped two such methods.

The first method, *Maximum Parsimony on Binary Encodings (MPBE)* [14, 15], produces one character for each gene adjacency present in the data—that is, if genes i and j occur as the adjacent pair ij (or $-j$-i) in one of the genomes, then we set up a binary character to indicate the presence or absence of this adjacency (coded 1 for presence and 0 for absence). The position of a character within the sequence is arbitrary, as long as it is the same for all genomes. By definition, there are at most $2n^2$ characters, so that the sequences are of lengths polynomial in the number of genes. Thus, analyses using maximum parsimony will run in time polynomial in the number of genes, but may require time exponential in the number of genomes. However, while a parsimony analysis relies on independence among characters, the characters produced by MPBE are emphatically dependent; moreover, translating the evolutionary model of gene orders into a matching model of sequence evolution for the encodings is quite difficult. This method suffers from several problems: (i) the ancestral sequences produced by the reconstruction method may not be valid encodings; (ii) none of the ancestral sequences can describe adjacencies not already present in the input data, thus limiting the possible rearrangements; and (iii) genomes must have equal gene content with no duplication.

The second method is the *MPME* method [79], where the second "M" stands for *Multistate*. In this method, we have exactly one character for each signed gene (thus $2n$ characters in all) and the state of a character is the signed gene that follows it in the gene ordering (in the direction indicated by the sign), so that each character can assume one of $2n$ possible states. Again, the position of each character within the sequence is arbitrary as long as it is consistent across all genomes, although it is most convenient to think of the ith character (with $i \leq n$) as associated with gene i, with the $n+i$th character associated with gene $-i$. For instance, the circular gene order $(1,-4,-3,-2)$ gives rise to the encoding $(-4,3,4,-1,2,1,-2,-3)$. Our results indicate that the MPME method dominates the MPBE method (among other things, the MPME method is able to create ancestral encodings that represent adjacencies not present in the input data). However, it still suffers from some of the same problems, as it also requires equal gene content with no duplication and it too can create invalid encodings. In addition it introduces a new problem of its own: the large number of character states quickly exceeds the computational limits of popular MP software. In any case, both MPBE and MPME methods are easily surpassed by direct optimization approaches.

12.3.2 *Direct optimization*

Blanchette and Sankoff [66] proposed to reconstruct the *breakpoint phylogeny*, i.e., the tree and ancestral gene orders that together minimize the total number of breakpoints along all edges of the tree. Since this problem includes the breakpoint median as a special case, it is NP-hard even for a fixed tree. Thus they proposed a heuristic, based on iterative improvement, for scoring a fixed tree and simply decided to examine all possible trees; the resulting procedure, `BPAnalysis`, is summarized in Fig. 12.13. Sankoff and Blanchette used this method to analyze a small mitochondrial dataset. This method

```
For each possible tree do
    Initially label all internal nodes with gene orders
    Repeat
        For each internal node v, with neighbors labelled A, B, and C, do
            Solve the median problem on A, B, and C to yield label M
            If relabelling v with M improves the score of T, then do it
    until no internal node can be relabelled
```

FIG. 12.13. `BPAnalysis`

is expensive at every level: first, its innermost loop repeatedly solves the breakpoint median problem, an NP-hard problem; second, the labelling procedure runs until no improvement is possible, thus using a potentially large number of iterations; and finally, the labelling procedure is used on every possible tree topology, of which there is an exponential number. The number of unrooted, unordered trees on n labelled leaves is $(2n-5)!!$, where the double factorial denotes the fact that only every other factor is used—that is, we have $(2n-5)!! = (2n-5)\cdot(2n-7)\cdot(2n-9)\cdot\ldots\cdot 5\cdot 3$. For just 13 genomes, we obtain *13.5 billion* trees; for 20 genomes, there are so many trees that merely counting to that value would take thousands of years on the fastest supercomputer.

Realizing this problem (we estimated that running `BPAnalysis` on an easy set of 13 chloroplast genomes would take several centuries), we reimplemented the strategy of Blanchette and Sankoff, but made extensive use of *algorithmic engineering* techniques [46] to speed it up—most notably in the use of lower bounds to avoid scoring most of the trees—and added the use of inversion distances in order to produce *inversion phylogenies*. The various techniques we used are listed in Table 12.3. In the case of the 13-taxon dataset, for instance, our bounding and ordering strategies eliminate all but 10,000 of the 13.5 billion trees. The tree lower bound is based on the triangle inequality that must be obeyed by any metric: in any ordering of the leaves of the tree, half of the sum of the pairwise distances between consecutive leaves must be a lower bound on the total length of the tree edges in the optimal tree. We take advantage of the unordered nature of the trees to compute the largest possible lower bound through swaps of the two children whenever such a swap leads to a larger value. The layering approach precomputes lower bounds for *all* trees and stores the trees in buckets according to increasing values of the lower bound; it then goes through the trees bucket by bucket, starting with those with the smallest lower bound, taking advantage of (i) the high corre-

Table 12.3 *Speedups for various algorithm engineering techniques*

technique used	speedup obtained
improving tree lower bound	500x
reducing memory usage	10x
better median solver	10x
hand-tuning code	5x
“layering” approach	5x
improving median lower bound	2x

lation between lower bound and final score and (ii) the low cost of bounding compared to the high cost of scoring. Reducing memory usage is accomplished by predeclaring all necessary space and re-using much of it on the fly; and hand-tuning code includes hand-unrolling loops, precomputing common expressions, choosing branch order, and, in general, carefully optimizing any inner loop that profiles too high.

The resulting code, `GRAPPA` (Genome Rearrangement Analysis under Parsimony and other Phylogenetic Algorithms) [53], with our best bounding and ordering schemes, can analyze the same 13-taxon dataset in 20 minutes on a laptop [49]—a speedup by a factor of about *two million*. Moreover, this speedup can easily be increased by the use of a large cluster computer, since `GRAPPA` is fully parallel and gets a nearly perfect speedup; in particular, running the code on a 512-processor machine yielded a *one-billion*-fold speedup.

However, a speedup by any constant factor, even a factor as large as a billion, can only add a constant to the size of datasets that can be analyzed with this method: every added taxon multiplies the total number of trees, and thus the running time, by twice the number of taxa. For instance, whereas `GRAPPA` can solve a 13-taxon dataset in 20 minutes, it would need over two million years to solve a 20-taxon dataset! In effect, the direct optimization method is, for now, limited to datasets of about 15 taxa; to put it differently: in order to scale direct optimization to larger datasets, we need to decompose those larger datasets into chunks of at most 14 taxa each.

12.3.3 *Direct optimization with a metamethod:* `DCM−GRAPPA`

Tang and Moret [76] succeeded in scaling up `GRAPPA` from its limit of around 15 taxa to over 1,000 taxa with no loss of accuracy and at a minimal cost in running time (on the order of 1–2 days). They did so by adapting a metamethod, the Disk-Covering Method (DCM), to the problem at hand, producing `DCM−GRAPPA`.

Disk-covering methods (DCMs) are a family of divide-and-conquer methods devised by Warnow and her colleagues. All DCMs are based on the idea of decomposing the set of taxa into overlapping "tight" subsets, using a base reconstruction method on the subsets to obtain trees, then combining the trees thus obtained to produce a tree for the entire dataset. There are three DCM variants to date, differing in their method of decomposition and their measure of tightness for subsets. The first DCM published, DCM-1 [29], is based on a distance matrix. It creates a graph in which each vertex is a taxon and two taxa are connected by an edge if their pairwise distance falls below some predetermined threshold; this graph is then triangulated and its maximum cliques computed (the former is done heuristically, the second exactly, both in polynomial time) to yield the desired subsets. Thus this method produces overlapping subsets in which no pair of taxa is farther apart than the threshold. The second DCM method, DCM-2 [30], also creates a threshold graph, but then computes a graph separator for it and produces subsets, each of which is the union of the separator and one of the isolated subgraphs. Finally, the third DCM method, DCM-3 [62], uses a *guide tree* to determine the decomposition and is best used in an iterative setting, with the tree produced at each iteration serving as guide tree for the next iteration. When used with sequence data, all three DCM variants use tree refinement methods to reduce the number of polytomies in the trees returned for each subset and for the entire dataset. When used for maximum parsimony analysis on sequences with the TNT package as its base method, the recursive and iterative version of DCM3 can easily analyze biological datasets of over 10,000 taxa, producing trees with parsimony scores within 0.01% of optimal in less than a day of computation [62].

Tang and Moret [76] used DCM-1 to produce `DCM-GRAPPA`. Because gene-order data produces very few polytomies, they did not need any tree refinement phase. However, because the size of the subsets cannot be constrained beforehand, they needed to use the DCM recursively in order to keep decomposing subsets until no subset held more than 14 taxa; a recursive decomposition is a natural enough idea, but poses difficult questions, such as the relationship between the size threshold used at one level of the recursion and that used at the level below. On simulated data (there are no biological gene-order datasets of such sizes), they found that `DCM-GRAPPA` scaled gracefully to well over 1,000 taxa (in two days of computation) and retained the high accuracy of the base method, `GRAPPA`—with fewer than 3% of the edges in error.

12.3.4 *Handling unequal gene content in reconstruction*

The method used by Tang and Moret [75] for computing the median of three known genomes in the presence of unequal gene content is not directly applicable to phylogenetic reconstruction in the style of `GRAPPA`, because the latter cannot rely on known gene orders for the three neighbors—certainly not initially, when internal nodes must be assigned gene orders in some rough manner, and not during the process, when every internal gene order is subject to replacement by a new median. To overcome this problem, Tang *et al.* [77] begin by computing the gene *content* of each internal node and then only proceed to assign and iterate over gene orders. Gene contents are assigned starting from the leaves (with known gene contents), using the principle illustrated in Fig. 12.11: if two sibling leaves both contain gene X, then so does their parent, while, if neither leaf contains contains X, then neither does their parent. When one leaf contains gene X and the other does not, gene X is noted as *ambiguous* for the parent; such ambiguities are resolved through propagation of constraints and iterative improvement, much in the style of the basic optimization heuristic of `GRAPPA`. This approach to the handling of unequal gene orders and duplications can be incorporated within `DCM-GRAPPA`, yielding a method for the analysis of large datasets with arbitrary gene content.

12.4 Experimentation in Phylogeny

Before we conclude our survey, we should say a few words about experimentation with phylogenetic reconstruction algorithms. While computer scientists have long evaluated algorithms in terms of their asymptotic running time and performance guarantees, it is only in the last 10 years that more formal approaches to the experimental assessment of algorithms have emerged, under the collective name of *experimental algorithmics*. Experimental algorithmics (see, e.g., [19, 45, 47] and the *Journal of Experimental Algorithmics* at `www.jea.acm.org`) is an emerging discipline that deals with how to test algorithms empirically to obtain reliable characterizations of their performance as well as deepen our understanding of their properties in order to refine them. Because it is based on experimental data, experimental algorithmics can seek inspiration from the physical sciences, but it must adapt to the specific goal—not to understand one phenomenon, but to generalize findings to an infinite range of possible instances.

In phylogenetic reconstruction, an assessment must take into account the accuracy of the reconstruction (in terms of the chosen optimization criterion but also, and more importantly, in terms of the biological significance of the results) as well as the scaling up of resource consumption (time and space). In turn, conducting such an assessment requires the use of a carefully designed set of benchmark datasets [52].

12.4.1 *How to test?*

First, how do we choose test sets? *Biological datasets* test performance where it matters, but they can be used only for ranking, are too few to permit quantitative evaluations, and are often hard to obtain. Moreover, the analysis of any large biological dataset will be hard to evaluate: one cannot just walk up to one's colleague in systematics with a 10,000-taxon tree in hand and ask her whether the tree is biologically plausible! Thus biological datasets are good for anecdotal reports and for "reality checks." In the latter capacity, of course, they are indispensable: no simulation can be accurate enough to replace real data. *Simulated datasets* enable absolute evaluations of solution quality (because the model, and thus the "true" answer, is known) and can be generated in arbitrarily large numbers to ensure statistical significance. Thus a combination of large-scale simulations and reasonable numbers of biological datasets is the only way to obtain valid characterizations of algorithms for phylogenetic reconstruction. The simulations must be based on the best possible models of the application at hand—in our case, we need accurate models of speciation and extinction, of gene duplication, gain, and loss, and of genome rearrangements.

12.4.2 *Phylogenetic considerations*

A typical simulation study runs as follows:

1. generate a rooted binary tree (according to a chosen model of speciation and extinction) with the appropriate number of leaves—this is known as the *model tree*;
2. assign a "length" (i.e., number of evolutionary events) to each edge of the tree according to a chosen model of divergence;
3. place a genome of suitable size and composition at the root;
4. evolve the genomes down the tree, i.e., transform the parent genome along each edge to its children according to the number of evolutionary events on that edge and to the chosen model of genome evolution;
5. collect the genomes thus generated at the leaves and use them as input to the reconstruction algorithm under test; and
6. compare the topology (and, if desired, the internal genomes) of the reconstructed tree with that of the model tree.

This sequence of operations is run many times for the same parameter values (number of taxa, size of genomes, parameters of the model of genome evolution, distribution of edge lengths, etc.) to ensure statistical significance. Naturally, a range of parameters is also explored. Thus the computational requirements are significant—keeping in mind that even a single reconstruction can prove quite expensive in terms of running time.

In the many years of experimental work we have conducted, we have found a number of useful guidelines, summarized below.

- Tree shape plays a surprisingly large role. Thus we need a reasonable model of speciation (and extinction), one that certainly goes beyond the simplistic models of uniform distributions or birth-death processes. Of course, the shape of the true trees is unknown and, in any case, depends on the selection of genomes (tight clades will show very different shapes from that of the entire Tree of Life, for instance), so that good simulations will need to use a selection of parameters.
- The evolutionary models for divergence and genome evolution are important. In particular, most reconstruction methods exhibit poor accuracy when the *diameter* of the dataset (the ratio of the largest to the smallest pairwise distance in the dataset) is large. Methods aimed at minimizing inversion distances may not perform as well on datasets where the predominant events are transpositions. Large numbers of duplications or very large gene losses also confuse most reconstruction methods. Thus the challenge is to devise an evolutionary model with few parameters that is easily manipulated analytically and computationally and produces realistic data.
- Testing a large range of parameters and using many runs for each setting to estimate variance are essential parts of any testing strategy. In the huge parameter space induced by even the simplest of models, it is all too easy to fall within an uncharacteristic region and draw entirely wrong conclusions about the behavior of the algorithm. Of course, the size of the parameter space makes it difficult to sample well.

That tree shape plays such a role was an unexpected finding. Most studies to date have used either a uniform model (popular in computer science) or a birth-death model (so-called Yule trees, popular in biology). Several authors [1, 2, 22, 27, 44] noted that published phylogenies exhibit a shape distribution that deviates from either model: in terms of balance (relative size or height of the two children of a node), published trees tend to be more balanced than uniformly distributed trees, but less balanced than birth-death trees. We subsequently found that simple strategies such as neighbor-joining do very well on datasets generated from birth-death trees and, with all other parameters held unchanged, quite poorly on datasets generated from uniformly distributed trees. Aldous [1, 2] proposed a model with a single balance parameter, the β-*splitting* model, that, according to the value of the parameter β, can generate perfectly balanced trees, birth-death trees, uniformly distributed trees, down to “caterpillar” (or “ladder”) trees (in which each internal node has a leaf as one of its children) and recommended a particular parameter setting to match the balance factors of published phylogenies. Unfortunately, that model lacks a biological foundation—it is a purely combinatorial model; moreover, the single parameter cannot localize tree structure—it acts on the entire tree at once. Heard [27] had earlier published a model with a strong biological foundation, in which the speciation rate is inherited and also subject to variation; again, depending on the setting of the speciation parameters (inheritance and variability), most distributions of tree balance can be produced. Heard's model, because it is founded on the birth-death process, has the added advantage of producing edge lengths (in terms of elapsed times), from which the number of evolutionary events can be inferred in terms of various evolutionary models. We have used both Aldous' and Heard's models in our simulations, with the most convincing results coming from Heard's model.

Many problems of biological verisimilitude appear at every stage, but perhaps most importantly in the process of generating genome rearrangements. Most studies to date, including ours, have used a simple process in which inversions (and, if included, transpositions and inverted transpositions) are generated uniformly at random. However, most chromosomes have internal structure that might prevent the occurrence of certain events (for instance, inversion might not be possible across a centromere) or favor the occurrence of others (for instance, there might be "hotspots" in the chromosome that are frequently involved as the endpoint of inversions or transpositions—for recent evidence of such, see [60]). The length of inversions and transpositions is an important question that has recently been considered in models of genomic evolution [65], in phylogenetics [34], and in comparative genomics—the latter of particular importance in the evolution of cancerous cells, where many short rearrangements are common.

Finally, a thorny issue in all optimization problems is the issue of robustness. NP-hard optimization problems, such as MP and (presumably) ML, often exhibit very brittle characteristics; little is known about the space of trees in the neighborhood of the true tree in phylogenetic reconstruction or about the effect on this space of the choice of parameters in the models.

12.5 Conclusion and Open Problems

Gene-content and gene-order data are being produced at increasing rates for many simple organisms, from organelles to bacteria, and in a few model eukaryotes. In phylogenetic work, such data have been found to carry a very strong and robust phylogenetic signal—reconstructions using such data, both in simulations and with biological datasets, provide information consistent with the best analyses run on sequence data, robust in the face of small changes, and less sensitive to mixes of small and large evolutionary distances than any sequence-based analysis. Moreover, these techniques scale well to large datasets (at least to 1,000 taxa, but most likely many more). That these data do so well in spite of the primitive tools available to date (simplistic models, limited optimization frameworks, enormous computational demands) bodes well and justifies a call for more research, particularly on the following topics.

- Tree models. Heard's model [27] is promising and perhaps even sufficient, but the effect of its various parameters on the accuracy and complexity of phylogenetic reconstruction needs to be better understood.
- Evolutionary models for genomes. As mentioned above, there are many questions and very few answers to date. For the time being, one can run simulations under many different models and verify that certain solutions work better than others; as new data emerge, however, one can expect improvements in the models.

- Extensions of the theory pioneered by Hannenhalli and Pevzner, beyond the work of El-Mabrouk, Marron *et al.*, and Hartman, to handle transpositions alone, transpositions and inversions, length-dependent rearrangements, position-dependent rearrangements, and duplications.
- Good combinatorial formulations of the median problem for inversions and for more general cases and, by extension, of the problem of assigning ancestral gene orders to a fixed tree in order to minimize the total number of evolutionary events (as weighted by the model of evolution). In particular, handling of large multi-chromosomal genomes, by integrating advances such as `MGR` and `DCM-GRAPPA`, would enable the use of gene-order data in the reconstruction of eukaryotic phylogenies.
- Tighter bounds on tree scores under the optimization model, so as to scale up the optimization to the largest possible datasets.
- Integration of the above within a DCM-like framework, in order to scale the computations to (nearly) arbitrarily large datasets.

12.6 Acknowledgements

Research on this topic at the University of New Mexico is supported by the US National Science Foundation under grants ANI 02-03584, EF 03-31654, IIS 01-13095, IIS 01-21377, and DEB 01-20709 (through a subcontract to the U. of Texas), by the US National Institutes of Health under grant 2R01GM056120-05A1 (through a subcontract to the U. of Arizona), and by IBM Corporation, under contract NBCH30390004 from the US Defense Advanced Research Projects Agency (the HPCS initiative). Research on this topic at the University of Texas is supported by the National Science Foundation under grants EF 03-31453, IIS 01-13654, IIS 01-21680, and DEB 01-20709, and by the David and Lucile Packard Foundation.

References

[1] Aldous, D.J. (1996). Probability distributions on cladograms. *Random Discrete Structures*, **76**, 1–18.

[2] Aldous, D.J. (2001). Stochastic models and descriptive statistics for phylogenetic trees, from yule to today. *Statistical Science*, **16**, 23–34.

[3] Atteson, K. (1999). The performance of the neighbor-joining methods of phylogenetic reconstruction. *Algorithmica 25*(2/3), 251–278.

[4] Bader, D.A., Moret, B.M.E., and Yan, M. (2001). A linear-time algorithm for computing inversion distance between signed permutations with an experimental study. *J. Computational Biology 8*(5), 483–491.

[5] Bafna, V. and Pevzner, P.A. (1998). Sorting by transpositions. *SIAM J. Discrete Mathematics*, **11**, 224–240.

[6] Bergeron, A. and Stoye, J. (2003). On the similarity of sets of permutations and its applications to genome comparison. In *Proc. 9th Int'l Conf. on Computing and Combinatorics (COCOON'03)*, Volume 2697 of *Lecture Notes in Computer Science*, pp. 68–79. Springer Verlag.

[7] Boore, J.L. and Brown, W.M. (1998). Big trees from little genomes: Mitochondrial gene order as a phylogenetic tool. *Curr. Opinion Genet. Dev. 8*(6), 668–674.

[8] Boore, J.L., Collins, T., Stanton, D., Daehler, L., and Brown, W.M. (1995). Deducing the pattern of arthropod phylogeny from mitochondrial DNA rearrangements. *Nature*, **376**, 163–165.

[9] Bourque, G. and Pevzner, P. (2002). Genome-scale evolution: reconstructing gene orders in the ancestral species. *Genome Research*, **12**, 26–36.

[10] Bruno, W.J., Socci, N.D., and Halpern, A.L. (2000). Weighted neighbor joining: A likelihood-based approach to distance-based phylogeny reconstruction. *Molecular Biology and Evolution 17*(1), 189–197.

[11] Bryant, D. (2000). The complexity of calculating exemplar distances. In *Comparative Genomics: Empirical and Analytical Approaches to Gene Order Dynamics, Map Alignment, and the Evolution of Gene Families* (ed. D. Sankoff and J. Nadeau), pp. 207–212. Kluwer Academic Pub., Dordrecht, Netherlands.

[12] Caprara, A. (1999). Formulations and hardness of multiple sorting by reversals. In *Proc. 3rd Int'l Conf. on Computational Molecular Biology (RECOMB'99)*, pp. 84–93. ACM Press.

[13] Caprara, A. (2001). On the practical solution of the reversal median problem. In *Proc. 1st Int'l Workshop on Algorithms in Bioinformatics (WABI'01)*, Volume 2149 of *Lecture Notes in Computer Science*, pp. 238–251. Springer Verlag.

[14] Cosner, M.E., Jansen, R.K., Moret, B.M.E., Raubeson, L.A., Wang, L.-S., Warnow, T., and Wyman, S.K. (2000). An empirical comparison of phylogenetic methods on chloroplast gene order data in Campanulaceae. In *Comparative Genomics: Empirical and Analytical Approaches to Gene Order Dynamics, Map Alignment, and the Evolution of Gene Families* (ed. D. Sankoff and J. Nadeau), pp. 99–121. Kluwer Academic Pub., Dordrecht, Netherlands.

[15] Cosner, M.E., Jansen, R.K., Moret, B.M.E., Raubeson, L.A., Wang, L.-S., Warnow, T., and Wyman, S.K. (2000). A new fast heuristic for computing the breakpoint phylogeny and a phylogenetic analysis of a group of highly rearranged chloroplast genomes. In *Proc. 8th Int'l Conf. on Intelligent Systems for Molecular Biology (ISMB'00)*, pp. 104–115.

[16] Earnest-DeYoung, J.V., Lerat, E., and Moret, B.M.E. (2004). Reversing gene erosion: reconstructing ancestral bacterial genomes from gene-content and gene-order data. In *Proc. 4th Int'l Workshop on Algorithms in Bioinformatics (WABI'04)*, Lecture Notes in Computer Science. to appear.

[17] El-Mabrouk, N. (2000). Genome rearrangement by reversals and insertions/deletions of contiguous segments. In *Proc. 11th Symp. on Combinatorial Pattern Matching (CPM'00)*, Volume 1848 of *Lecture Notes in Computer Science*, pp. 222–234. Springer Verlag.

[18] Felsenstein, J. (1993). *Phylogenetic Inference Package (PHYLIP), Version 3.5*. University of Washington, Seattle.

[19] Fleischer, R., Moret, B.M.E., and Schmidt, E.M. (2002). *Experimental Algorithmics*, Volume 2547 of *Lecture Notes in Computer Science*. Springer Verlag.

[20] Gascuel, O. (1997). BIONJ: An improved version of the NJ algorithm based on a

simple model of sequence data. *Molecular Biology and Evolution 14*(7), 685–695.

[21] Goloboff, P. (1999). Analyzing large datasets in reasonable times: solutions for composite optima. *Cladistics*, **15**, 415–428.

[22] Guyer, C. and Slowinski, J.B. (1991). Comparisons between observed phylogenetic topologies with null expectations among three monophyletic lineages. *Evolution*, **45**, 340–350.

[23] Hannenhalli, S. and Pevzner, P.A. (1995*a*). Transforming cabbage into turnip (polynomial algorithm for sorting signed permutations by reversals). In *Proc. 27th ACM Symp. on Theory of Computing (STOC'95)*, pp. 178–189. ACM Press.

[24] Hannenhalli, S. and Pevzner, P.A. (1995*b*). Transforming mice into men (polynomial algorithm for genomic distance problems). In *Proc. 36th IEEE Symp. on Foundations of Comput. Science (FOCS'95)*, pp. 581–592. IEEE Press.

[25] Hartman, T. (2003). A simpler 1.5-approximation algorithm for sorting by transpositions. In *Proc. 14th Symp. on Combinatorial Pattern Matching (CPM'03)*, Volume 2676 of *Lecture Notes in Computer Science*, pp. 156–169.

[26] Hartman, T. and Sharan, R. (2004). A 1.5-approximation algorithm for sorting bytranspositions and transreversals. In *Proc. 4th Int'l Workshop on Algorithms in Bioinformatics (WABI'04)*, Lecture Notes in Computer Science. to appear.

[27] Heard, S.B. (1996). Patterns in phylogenetic tree balance with variable and evolving speciation rates. *Evolution*, **50**, 2141–2148.

[28] Huelsenbeck, J.P. and Ronquist, F. (2001). MrBayes: Bayesian inference of phylogeny. *Bioinformatics*, **17**, 754b. Available at `morphbank.ebc.uu.se/mrbayes/`.

[29] Huson, D., Nettles, S., and Warnow, T. (1999). Disk-covering, a fast converging method for phylogenetic tree reconstruction. *J. Computational Biology 6*(3), 369–386.

[30] Huson, D., Vawter, L., and Warnow, T. (1999). Solving large scale phylogenetic problems using DCM-2. In *Proc. 7th Int'l Conf. on Intelligent Systems for Molecular Biology (ISMB'99)*.

[31] Jansen, R.K. and Palmer, J.D. (1987). A chloroplast DNA inversion marks an ancient evolutionary split in the sunflower family (Asteraceae). *Proc. National Academy of Sciences USA*, **84**, 5818–5822.

[32] Kumar, S., Tamura, K., Jakobsen, I.B., and Nei, M. (2001). MEGA2: Molecular evolutionary genetics analysis software. *Bioinformatics 17*(12), 1244–1245.

[33] Larget, B., Simon, D.L., and Kadane, J.B. (2002). Bayesian phylogenetic inference from animal mitochondrial genome arrangements. *J. Royal Stat. Soc. B 64*(4), 681–694.

[34] Lefebvre, J.-F., El-Mabrouk, N., Tillier, E.R.M., and Sankoff, D. (2003). Detection and validation of single gene inversions. In *Proc. 11th Int'l Conf. on Intelligent Systems for Molecular Biology (ISMB'03)*, Volume 19 of *Bioinformatics*, pp. i190–i196. Oxford University Press.

[35] Lewis, P.O. (1998). A genetic algorithm for maximum likelihood phylogeny inference using nucleotide sequence data. *Molecular Biology and Evolution*, **15**, 277–283.

[36] Liu, T., Moret, B.M.E., and Bader, D.A. (2003). An exact, linear-time algorithm for computing genomic distances under inversions and deletions. Technical Report TR-CS-2003-31, Univ. of New Mexico.

[37] Maddison, D.R. and Maddison, W.P. (2000). *MacClade version 4: Analysis of phylogeny and character evolution*. Sinauer Assoc., Sunderland, MA.

[38] Maddison, W.P. (1990). A method for testing the correlated evolution of two binary characters: are gains or losses concentrated on certain branches of a phylogenetic tree? *Evolution*, **44**, 539–557.

[39] Maddison, W.P. (1997). Gene trees in species trees. *Systematic Biology 46*(3), 523–536.

[40] Maddison, W.P. and Maddison, D.R. (2001). *Mesquite: a modular system for evolutionary analyses, version 0.98*. `mesquiteproject.org`.
[41] Marron, M., Swenson, K.M., and Moret, B.M.E. (2003). Genomic distances under deletions and insertions. In *Proc. 9th Int'l Conf. on Computing and Combinatorics (COCOON'03)*, Volume 2697 of *Lecture Notes in Computer Science*, pp. 537–547. Springer Verlag.
[42] McLysaght, A., Baldi, P.F., and Gaut, B.S. (2003). Extensive gene gain associated with adaptive evolution of poxviruses. *Proc. National Academy of Sciences USA*, **100**, 15655–15660.
[43] Montague, M.G. and Hutchinson III, C.A. (2000). Gene content and phylogeny of herpesviruses. *Proc. National Academy of Sciences USA*, **97**, 5334–5339.
[44] Mooers, A.O. and Heard, S.B. (1997). Inferring evolutionary process from phylogenetic tree shape. *Quarterly Rev. Biol.*, **72**, 31–54.
[45] Moret, B.M.E. (2002). Towards a discipline of experimental algorithmics. In *Data Structures, Near Neighbor Searches, and Methodology: Fifth and Sixth DIMACS Implementation Challenges* (ed. M. Goldwasser, D. Johnson, and C. McGeoch), Volume 59 of *DIMACS Monographs*, pp. 197–213. AMS Press.
[46] Moret, B.M.E., Bader, D.A., and Warnow, T. (2002). High-performance algorithm engineering for computational phylogenetics. *J. Supercomputing*, **22**, 99–111.
[47] Moret, B.M.E. and Shapiro, H.D. (2001). Algorithms and experiments: The new (and the old) methodology. *J. Univ. Comput. Sci. 7*(5), 434–446.
[48] Moret, B.M.E., Siepel, A.C., Tang, J., and Liu, T. (2002). Inversion medians outperform breakpoint medians in phylogeny reconstruction from gene-order data. In *Proc. 2nd Int'l Workshop on Algorithms in Bioinformatics (WABI'02)* (ed. R. Guigo and D. Gusfield), Volume 2452 of *Lecture Notes in Computer Science*, pp. 521–536. Springer Verlag.
[49] Moret, B.M.E., Tang, J., Wang, L.-S., and Warnow, T. (2002). Steps toward accurate reconstructions of phylogenies from gene-order data. *J. Computer and System Sciences 65*(3), 508–525.
[50] Moret, B.M.E., Wang, L.-S., and Warnow, T. (2002). New software for computational phylogenetics. *IEEE Computer 35*(7), 55–64.
[51] Moret, B.M.E., Wang, L.-S., Warnow, T., and Wyman, S.K. (2001). New approaches for reconstructing phylogenies from gene-order data. In *Proc. 9th Int'l Conf. on Intelligent Systems for Molecular Biology (ISMB'01)*, Volume 17 of *Bioinformatics*, pp. S165–S173. Oxford University Press.
[52] Moret, B.M.E. and Warnow, T. (2002). Reconstructing optimal phylogenetic trees: A challenge in experimental algorithmics. In *Experimental Algorithmics* (ed. R. Fleischer, B.M.E. Moret, and E. Schmidt), Volume 2547 of *Lecture Notes in Computer Science*, pp. 163–180. Springer Verlag.
[53] Moret, B.M.E., Wyman, S.K., Bader, D.A., Warnow, T., and Yan, M. (2001). A new implementation and detailed study of breakpoint analysis. In *Proc. 6th Pacific Symp. on Biocomputing (PSB'01)*, pp. 583–594. World Scientific Pub.
[54] Nadeau, J.H. and Taylor, B.A. (1984). Lengths of chromosome segments conserved since divergence of man and mouse. *Proc. National Academy of Sciences USA*, **81**, 814–818.

[55] Nakhleh, L., Roshan, U., St. John, K., Sun, J., and Warnow, T. (2001). Designing fast converging phylogenetic methods. In *Proc. 9th Int'l Conf. on Intelligent Systems for Molecular Biology (ISMB'01)*, Volume 17 of *Bioinformatics*, pp. S190–S198. Oxford University Press.

[56] Olsen, G., Matsuda, H., Hagstrom, R., and Overbeek, R. (1994). FastDNAml: A tool for construction of phylogenetic trees of DNA sequences using maximum likelihood. *Computations in Applied Biosciences 10*(1), 41–48.

[57] Page, R.D.M. and Charleston, M.A. (1997). From gene to organismal phylogeny: Reconciled trees and the gene tree/species tree problem. *Molecular Phylogenetics and Evolution*, **7**, 231–240.

[58] Palmer, J.D. (1992). Chloroplast and mitochondrial genome evolution in land plants. In *Cell Organelles* (ed. R. Herrmann), pp. 99–133. Springer Verlag.

[59] Pe'er, I. and Shamir, R. (1998). The median problems for breakpoints are NP-complete. *Elec. Colloq. on Comput. Complexity*, **71**.

[60] Pevzner, P. and Tesler, G. (2003). Human and mouse genomic sequences reveal extensive breakpoint reuse in mammalian evolution. *Proc. National Academy of Sciences USA 100*(13), 7672–7677.

[61] Rokas, A. and Holland, P.W.H. (2000). Rare genomic changes as a tool for phylogenetics. *Trends in Ecologoy and Evolution*, **15**, 454–459.

[62] Roshan, U., Moret, B.M.E., Williams, T.L., and Warnow, T. (2004). Rec-I-DCM3: A fast algorithmic technique for reconstructing large phylogenetic trees. In *Proc. 3rd IEEE Computational Systems Bioinformatics Conf. CSB'04*, IEEE Press, to appear.

[63] Saitou, N. and Nei, M. (1987). The neighbor-joining method: A new method for reconstructing phylogenetic trees. *Molecular Biology and Evolution*, **4**, 406–425.

[64] Sankoff, D. (1999). Genome rearrangement with gene families. *Bioinformatics 15*(11), 990–917.

[65] Sankoff, D. (2002). Short inversions and conserved gene cluster. *Bioinformatics 18*(10), 1305.

[66] Sankoff, D. and Blanchette, M. (1998). Multiple genome rearrangement and breakpoint phylogeny. *J. Computational Biology*, **5**, 555–570.

[67] Sankoff, D., Ferretti, V., and Nadeau, J.H. (1997). Conserved segment identification. *J. Computational Biology 4*(4), 559–565.

[68] Sankoff, D. and Nadeau, J.H. (1996). Conserved synteny as a measure of genomic distance. *Discrete Applied Mathematics 71*(1–3), 247–257.

[69] Siepel, A.C. and Moret, B.M.E. (2001). Finding an optimal inversion median: Experimental results. In *Proc. 1st Int'l Workshop on Algorithms in Bioinformatics (WABI'01)* (ed. O. Gascuel and B. Moret), Volume 2149 of *Lecture Notes in Computer Science*, pp. 189–203. Springer Verlag.

[70] St. John, K., Warnow, T., Moret, B.M.E., and Vawter, L. (2003). Performance study of phylogenetic methods: (unweighted) quartet methods and neighbor-joining. *J. Algorithms 48*(1), 173–193.

[71] Steel, M.A. (1994). The maximum likelihood point for a phylogenetic tree is not unique. *Systematic Biology 43*(4), 560–564.

[72] Swenson, K.M., Marron, M., Earnest-DeYoung, J.V., and Moret, B.M.E. (2004). Approximating the true evolutionary distance between two genomes. Technical Report TR-CS-2004-15, Univ. of New Mexico.

[73] Swofford, D. (2001). *PAUP*: Phylogenetic analysis using parsimony (*and other methods), version 4.0b8*. Sinauer Assoc., Sunderland, MA.

[74] Swofford, D.L., Olsen, G.J., Waddell, P.J., and Hillis, D.M. (1996). Phylogenetic inference. In *Molecular Systematics* (ed. D.M. Hillis, B.K. Mable, and C. Moritz), pp. 407–514. Sinauer Assoc., Sunderland, MA.

[75] Tang, J. and Moret, B.M.E. (2003*a*). Phylogenetic reconstruction from gene rearrangement data with unequal gene contents. In *Proc. 8th Int'l Workshop on Algorithms and Data Structures (WADS'03)*, Volume 2748 of *Lecture Notes in Computer Science*, pp. 37–46. Springer Verlag.

[76] Tang, J. and Moret, B.M.E. (2003*b*). Scaling up accurate phylogenetic reconstruction from gene-order data. In *Proc. 11th Int'l Conf. on Intelligent Systems for Molecular Biology (ISMB'03)*, Volume 19 of *Bioinformatics*, pp. i305–i312. Oxford University Press.

[77] Tang, J., Moret, B.M.E., Cui, L., and dePamphilis, C.W. (2004). Phylogenetic reconstruction from arbitrary gene-order data. In *Proc. 4th IEEE Symp. on Bioinformatics and Bioengineering BIBE'04*, pp. 592–599. IEEE Press.

[78] Tesler, G. (2002). Efficient algorithms for multichromosomal genome rearrangements. *J. Computer and System Sciences 65*(3), 587–609.

[79] Wang, L.-S., Jansen, R.K., Moret, B.M.E., Raubeson, L.A., and Warnow, T. (2002). Fast phylogenetic methods for genome rearrangement evolution: An empirical study. In *Proc. 7th Pacific Symp. on Biocomputing (PSB'02)*, pp. 524–535. World Scientific Pub.

[80] Wang, L.-S. and Warnow, T. (2001). Estimating true evolutionary distances between genomes. In *Proc. 33rd ACM Symp. on Theory of Computing (STOC'01)*, pp. 637–646. ACM Press.

CONTRIBUTORS

Bernard M.E. Moret
Department of Computer Science,
University of New Mexico,
Albuquerque NM 87131, USA
`moret@cs.unm.edu`

Jijun Tang
Department of Computer Science,
University of New Mexico,
Albuquerque NM 87131, USA
`jtang@cs.unm.edu`

Tandy Warnow
Department of Computer Sciences,
University of Texas,
Austin TX 78712, USA
`warnow@cs.utexas.edu`

INDEX