A Heuristic for Phylogenetic Reconstruction Using Transposition

Feng Yue*, Meng Zhang† and Jijun Tang*

*Department of Computer Science and Engineering
University of South Carolina, Columbia, SC 29208, USA
Email: {yue2, jtang}@cse.sc.edu

†College of Computer Science and Technology
Jilin University, Changchun 130012, China
Email: zm@mail.edu.cn

***Abstract*—Because of the advent of high-throughput sequencing and the consequent reduction in cost of sequencing, many organisms have been completely sequenced and most of their genes identified; homologies among these genes are also getting established. It thus has become possible to represent whole genomes as ordered lists of gene identifiers and to study the evolution of these entities through computational means, in systematics as well as in comparative genomics. As a result, gene order data (also known as genome rearrangement data) has attracted increasing attention from both biologists and computer scientists as a new type of data for phylogenetic analysis. Methods for reconstructing phylogeny from genome rearrangements include distance-based methods, MCMC methods and direct optimization methods. The latter, pioneered by Sankoff and extended in the software packages of GRAPPA and MGR, is the most accurate approach for inversion phylogeny. However, due to the difficulty of computing the transposition distance, this type of methods has not been applied to datasets where transposition is the only or dominant event.**

In this paper, we present a heuristic transposition median solver and extend GRAPPA to handle transpositions. Our extensive testing using simulated datasets shows that this method (GRAPPA-TP) is very accurate in terms of ancestor genome inference and phylogenetic reconstruction. It also suggests that model match is critical in phylogenetic analysis, and a fast and accurate method for transposition distance computation is still very important. The new GRAPPA-TP is available from phylo.cse.sc.edu.

I. Introduction

A phylogeny is a representation of the evolutionary history of a collection of organisms or genes (known as taxa). The basic assumption of process necessary to phylogenetic reconstruction is repeated divergence within species or genes. A phylogenetic reconstruction is usually depicted as a tree, in which modern taxa reside at the leaves and ancestral taxa occupy internal nodes, with the edges of the tree denoting evolutionary relationships among the taxa. Reconstructing phylogenies is a major component of modern research programs in biology and medicine.

While phylogenetic studies in the pre-genome era primarily focused on DNA or protein sequence differences among organisms, informative comparisons can in fact be made at various organizational levels. Higher-level evolutionary events of relevance to phylogenetics include genome duplication, lateral gene transfer, inversion, transposition, deletion and insertion. Phylogenetic analysis of whole genomes that model these types of events are proving to be extremely useful in elucidating the evolutionary relationships among organisms [9]. Since the pioneering papers of Sankoff [3], genome rearrangement data has attracted increasing attention from both biologists and computer scientists as a new type of data for phylogenetic analysis.

During the past several years, computer scientists have been able to make substantial progress in genome rearrangement research. With the solution for inversion distance [12] and inversion median [7], we were able to estimate phylogenies and ancestral genomes based on inversions. There are several widely used methods for genome rearrangement analysis, including neighbor-joining [22], GRAPPA [16], MGR [6] and Badger [15]. The main software packages for reconstructing the inversion (or breakpoint) phylogeny are GRAPPA [16] and MGR [6]. Their basic optimization tool is an algorithm for computing the inversion (or breakpoint) median of three genomes. Extensive testing has shown that the trees returned by these methods are superior to those returned by other methods, such as distance-based methods and parsimony based on encodings [19], [28].

Much of the research on genome rearrangement has focused on organellar genomes, such as mitochondrial [5] and chloroplast genomes [8]. GRAPPA and MGR have been applied successfully to chloroplast genomes in which inversion is the most important event. In other datasets (e.g., mitochondrial genomes), transpositions are viewed as more likely, although their relative preponderance with respect to inversions is unknown. However, due to the lack of efficient method to compute transposition distances, GRAPPA and other methods have to rely on using inversion or breakpoint to estimate transpositions, hence their accuracy for transposition-dominant datasets is not very good. In this paper, we will introduce a heuristic for transposition median solver and use it to infer phylogenies and ancestral genomes when transposition is the only event.

This paper is structured as follows: we will first give a

brief background of genome rearrangements and methods for phylogenetic analysis; we will then introduce our new method GRAPPA-TP and give details about its two major components; at the end of this paper, we will provide experimental results from our simulation studies to assess the accuracy of GRAPPA-TP and discuss the importance of model match in genome rearrangement analysis.

II. BACKGROUNDS

A. Genome Rearrangements

We assume a reference set of n genes $\{g_1, g_2, \cdots, g_n\}$, thus a genome can be represented as a signed ordering of these genes, and each gene is given an orientation that is either positive, written g_i, or negative, written $-g_i$. Genomes can evolve through events such as inversions, transpositions and transversion, as well as other events. When transposition is the only event, the sign of each gene is irrelevant and can be ignored.

Let G be the genome with signed ordering of $g_1, g_2, \cdots, g_n$. An *inversion* (reversal) between indices i and j $(i \leq j)$, transforms G to a new genome with linear ordering

$$g_1, g_2, \cdots, g_{i-1}, -g_j, -g_{j-1}, \cdots, -g_i, g_{j+1}, \cdots, g_n$$

A *transposition* on genome G acts on three indices i, j, k, with $i \leq j$ and $k \notin [i, j]$, picking up the interval $g_i, g_{i+1}, \cdots, g_j$ and inserting it immediately after g_k. Thus genome G is replaced by (assume $k > j$):

$$g_1, \cdots, g_{i-1}, g_{j+1}, \cdots, g_k, g_i, g_{i+1}, \cdots, g_j, g_{k+1}, \cdots, g_n$$

An *transversion* is a transposition followed by an inversion of the transposed subsequence; it is also called an *inverted transposition*.

B. Distance Computation

Given two genomes G_1 and G_2, we define the *edit distance* $d(G_1, G_2)$ as the minimum number of events required to transform one genome into the other.

The *breakpoint distance* [3] is not a direct evolutionary distance measurement. A breakpoint in G_1 is defined as an ordered pair of genes (g_i, g_j) such that g_i and g_j are adjacent in G_1 but not in G_2. The breakpoint distance is simply the number of breakpoints in G_1 relative to G_2.

When only inversions are allowed, the edit distance is the *inversion distance*. Hannenhalli and Pevzner [12] developed a mathematical and computational framework for signed gene-orders and provided a polynomial-time algorithm to compute the edit distance between two signed gene-orders under inversions; Bader et al. [1] later showed that this edit distance can be computed in linear time. However, computing the inversion distance is NP-hard in the unsigned case [7].

The *transposition distance* is the minimum number of transpositions needed. Computing the transposition distance is of unknown complexity and after 10 years of research, the best available method is only a 1.375-approximation [10]. Yancopoulos et al. [29] proposed a "universal" double-cut-and-join (DCJ) operation that accounts for inversions, translocations, fissions and fusions, resulting in a new genomic distance that can be computed in linear time.

C. Median Problem of Three

The median problem on three genomes is to find a single genome that minimizes the sum of the pairwise distances between itself and each of the three given genomes. This problem is NP-hard [20] even for the simplest breakpoint distance. Seeking a median that minimizes the breakpoint distance can be transformed into a special instance of the well-studied Traveling Salesperson Problem [3], hence can be solved relatively efficient. But in practice, the breakpoint median is not effective—it is easy to obtain trivial solutions (where the median gene-order coincides with one of the leaves) and many equally optimal trees are returned.

The *inversion median* problem is to find a median genome that minimizes the summation of inversion distances on the three edges. Two exact median solvers have been proposed, all using a branch-and-bound strategy. Caprara's solver [7] is based on an extension of the breakpoint graph, while that developed by Siepel and Moret [23] runs a direct search. Using the inversion median has dramatically improved the accuracy of GRAPPA [18]. Two heuristics, MGR [6] and rEvoluzer [2], are also proposed to improve the speed of inversion median, for a sacrifice of accuracy. However, inversion median problem is NP-hard on both the edge length and number of genes, thus when the genomes are large, all these solvers may require centuries to find a solution.

D. Phylogenetic Reconstruction from Genome Rearrangements

Reconstructing phylogenies from genome rearrangement data is computationally much harder than from sequence data. For example, finding the minimum number of evolutionary events given a fixed tree can be done in linear time if the leaves are labeled with DNA or protein sequences, whereas such task for genome rearrangement data is NP hard even when the tree has only three leaves.

Methods for reconstructing trees based on genome rearrangement data include distance-based methods (for example, neighbor-joining [22]), maximum parsimony methods based on encodings [27], [28], and direct optimization methods. The latter, pioneered by Sankoff and Blanchette [3] in their package BPAnalysis and improved by GRAPPA [16] and MGR, is the most accurate method. Besides returning a phylogeny, these three methods can also give an estimate of ancestral gene orders, which will have great utility for biologists interested in the process of genome rearrangement.

GRAPPA is an exhaustive search method, moving systematically through the space of all $(2N-5)(2N-7)\cdots 3$ possible trees on N genomes [11]. For each tree, the program tests a lower bound to determine whether the tree is worth scoring; if so, then the program will iteratively solve the median problems at internal nodes until convergence, as outlined in Figure 1.

Existing methods can still be used when transposition is the dominant event. For example, we can estimate the transposition distance and use neighbor-joining method to reconstruct the phylogeny. Since one transposition can be simulated by using three inversions, we can also apply GRAPPA or MGR to

```
Initially label all internal nodes with gene orders
Repeat
    For each internal node v, with neighbors A, B and C, do
    Solve median problem on A, B, C to yield m
    If relabeling v with m improves the tree score, then do it
Until no change occurs
```

Fig. 1. The GRAPPA scoring procedure

obtain the phylogeny, using either breakpoint median solver or inversion median solver. Because the evolutionary model is mismatched, their performance on transposition datasets is questionable, as indicated by our experimental results shown in section IV.

III. Algorithm Details

We extended GRAPPA to handle transpositions. The new method is named GRAPPA-TP, with two major extensions: a heuristic method to estimate transposition distance, and new median solver for transpositions.

A. Transposition Distance Estimation

Although no polynomial algorithms for transposition distance has been reported, researchers are able to estimate the distance using the 1.375-approximation by Hartman [10] or the DCJ distance by Yancopoulos et al. [29].

The only existing software that can compute transposition distance is derange2 developed by Blanchette [4], which exhaustively searches for a minimum number of transpositions to transform one genome to another. Our tests have shown that when the distance is less than 10% of the number of genes, this method is very fast and the results are very close to the true distances. However, any test above this threshold cannot be finished after several days of computation. For phylogenetic analysis, even when the genomes are close, the distance between some leaves can easily exceed this threshold, thus derange2 may not be applicable. In this paper, we propose a heuristic method which gives satisfactory results.

Given two permutations π_1 and π_2, a transposition applied on π_2 can reduce the number of breakpoints by 3, 2, 1 or 0, as indicated in Figure 2.

Based on this observation, computing the transposition distance can be transferred to find the fewest number of steps that will bring the number of breakpoint to zero. Since no polynomial algorithm exists to find the transposition distance, in this paper, we develop a heuristic that uses a brute-force approach to quickly reduce the number of breakpoints.

The algorithm works as follows: it starts from π_2 and move towards π_1, and at each step, it will enumerate all transpositions and apply the one on π_2 that can reduce the most number of breakpoints. It will continue the process until the number of breakpoints becomes 0 (i.e. π_2 is transformed to π_1). The transposition distance is then the number of steps it uses to bring π_2 to π_1. When there are multiple choices, it will randomly choose one.

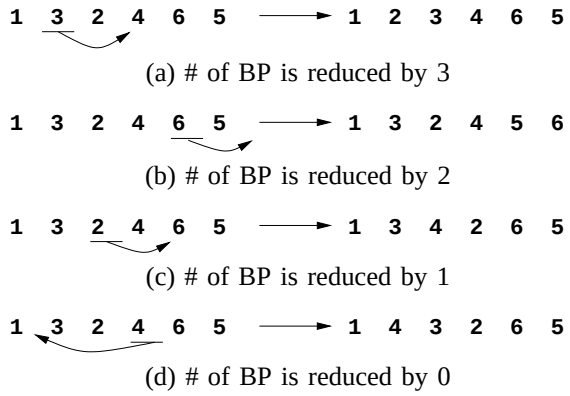


Fig. 2. Number of breakpoint changes by applying different transpositions, compared to the identity permutation (1 2 3 4 5 6).

To get a better result, we can repeat the above process several times and report the smallest value as the distance. In our experiments, we found that no more than 10 repeats are needed. Obviously the heuristic will return a distance which is greater or equal to the edit distance.

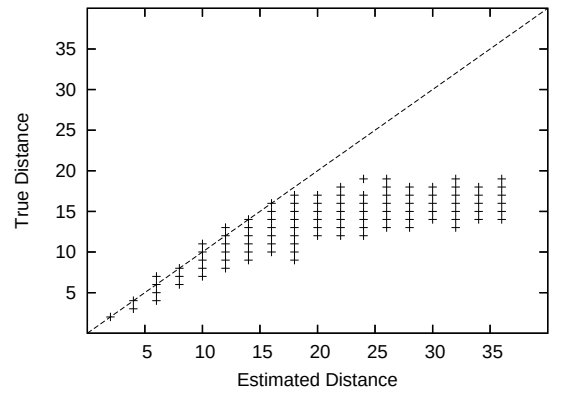
Figure 3 shows the performance of our brute-force distance estimation on simulated datasets with 37 and 100 genes. This figure suggests that the estimated distance closely follows the true distance when $\frac{r}{n} < 20\%$, where r is the number of transpositions and n is the number of genes. Above this ratio, even our heuristic will severely under-estimate the true distance. The estimated distance appears to converge onto $n/2$, which is close to the conjectured diameter of transposition distances [10].

One should note that this estimator will fail badly for the reverse identity. For example, it only needs four steps to transform (7 6 5 4 3 2 1) into (1 2 3 4 5 6 7), while our estimator needs seven steps. However, such cases are very rare, as indicated by Figure 3.

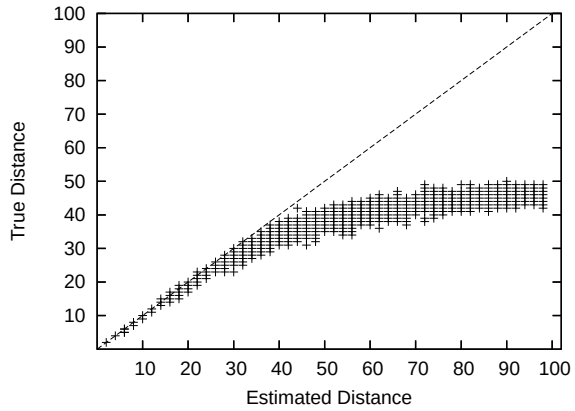
B. Transposition Median Solver

We extended Siepel's algorithm [23] to handle transpositions using a branch-and-bound approach:

- Given the three permutations π_1, π_2 and π_3, compute the lower bound on the median score, $D(M) = \frac{d_{1,2}+d_{2,3}+d_{3,1}}{2}$.
- Pick one permutation from π_1, π_2, π_3 as start (the so-called trivial median) and push it into a queue; its median score is the initial best-so-far.



(a) 37 genes



(b) 100 genes

Fig. 3. Distance estimation results for 37 genes (top) and 100 genes (bottom).

- Iteratively remove a permutation π from the queue until the queue is empty:
  - If the median score of π meets the lower bound, $d_{\pi,1} + d_{\pi,2} + d_{\pi,3} = D(M)$, then stop.
  - If the median score of π is less than the current best-so-far, update the latter.
  - create all $\binom{n}{3}$ neighboring permutations (one transposition away from π), discard those with lower bounds that exceed the best-so-far, and add the surviving ones to the queue.

Clearly, since there are $\binom{n}{3}$ neighbors for each step, the success of this algorithm relies on good lower bounds to eliminate as many neighbors as possible. Several lower bounds have been proved. Among them, the following two bounds are the most effective [23], [26]:

(Bound 1) If ϕ is a permutation on the shortest path from π_1 to the median, then it obeys:

$$d_{1,\phi} + \frac{d_{2,\phi} + d_{3,\phi} + d_{2,3}}{2} \leq D(M).$$

(Bound 2) If ϕ is a permutation on the shortest path from π_1 to the median and γ is derived from ϕ by applying one inversion, then, if γ is also on the shortest path from π_1 to M, it obeys (Figure 4): $d_{1,\gamma} + d_{2,\gamma} + d_{3,\gamma} \leq d_{1,\phi} + d_{2,\phi} + d_{3,\phi} + 1$. In the other words, we will ignore those neighbors which bring the search back more than one step.

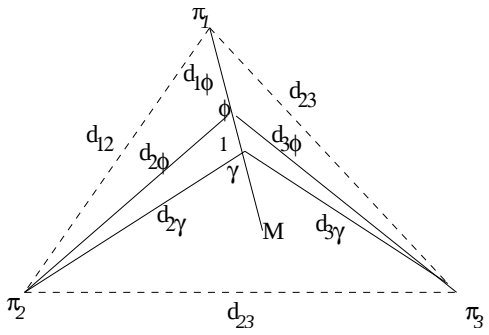


Fig. 4. Illustration of the two bounds

When the genomes are relatively close, our distance estimation is near optimal, hence the above bounds will still be effective.

The speed of our median solver is regulated by two factors: the distance from the median to its closest leave, and the number of genes present. To make the genome length relatively unimportant, We can condense the genomes using the concept of conserved adjacency. A gene pair (x, y) is conserved adjacent if (x, y) or its inverse (-y, -x) is present in all genomes as consecutive elements [13]. A block of k adjacent genes can be replaced by a pseudo-gene and the total number of genes reduces by $k - 1$ [6]. This condensation procedure is very effective when the genomes are close.

C. Phylogenetic Analysis

When a tree is given, we will use the iterative approach outlined in Figure 1 to find the best tree score. This procedure of scoring depends on the initial assignment of gene orders to internal nodes, which has no gene-orders assigned when the scoring starts.

Internal genomes can be initialized trivially, by giving each internal node a random gene order. However, other complex procedures yield better results, such as the *Nearest Neighbor Method*, which assigns each internal node the median solution from its three nearest leaves, using a median solver of choice. In GRAPPA-TP, we choose to use transposition median solver in the initialization procedure as well. Although using breakpoint median solver may be faster, it may introduce gene signs that is hard to deal with, due to the fact that transposition does not deal with signs at all.

To search through the large tree space, we will enumerate all trees and use the tightened circular-ordering lower bounds to discard bad trees before scoring them [17].

The lower bound used by GRAPPA is derived from the following theorem:

Theorem 1: Let d be a $n \times n$ matrix of pairwise distances between the taxa in a set S; let T be a tree leaf-labeled by the taxa in S and w be an edge-weighting (tree score) on T, so that we have $w_{ij} = \sum_{e \in P_{ij}} w(e) \geq d_{ij}$, where P_{ij} is a path from i to j on tree T. Set $w(T) = \sum_{e \in E(T)} w(e)$. If

$1, 2, \ldots, n$ is a circular ordering of the leaves of T, then we have $2w(T) \geq d_{1,2} + d_{2,3} + \cdots + d_{n,1}$.

This immediately gives us a (circular ordering) lower bound for the tree score, i.e. the tree score $w(T)$ should at least be $\frac{d_{1,2}+d_{2,3}+\cdots+d_{n,1}}{2}$. Since our transposition distance computation is not exact, using the lower bound to prune trees become become heuristic, however it is still very effective and more than 99.9% are pruned away in our experiements. Because the lower bound can be computed very efficiently and is much cheaper than the scoring procedure, such high pruning rate generally indicates more than 100 times speed-up. Other lower bounds have been developed recently, all based on pairwise distances, hence the speed of GRAPPA-TP can be further improved by using those bounds.

IV. Experimental Results

We set out to examine the performance of the new GRAPPA-TP, through two simulation studies: the first study is to measure the accuracy of the inferred median genome (estimated ancestor) compared to the true ancestor, using datasets of three input genomes; the second is to measure the accuracy of the inferred phylogeny compared to the true tree, using datasets of 10 genomes. All the experiments are conducted on a Linux cluster with 152 Intel Xeon CPUs, but each CPU works independently on a test task.

A. Accuracy of Ancestor Inference

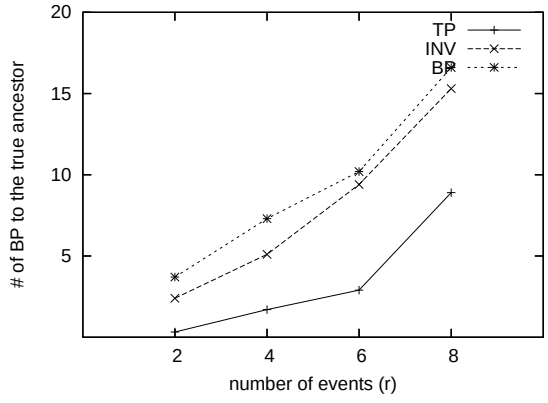
We first examine the quality of GRAPPA-TP for finding ancestor genomes. In our simulation study, each genome has 37 and 100 genes, spanning the range from mitochondria to chloroplast.

We create each dataset by first randomly generating a tree topology with three leaves and assigning its three edges with different lengths. The length (number of events) of each edge is sampled from a uniform distribution on the set $\{0.5r, \ldots, 1.5r\}$, where r is the expected number of evolutionary events. In this experiment, we use $r = 2 \sim 8$, where $r = 2$ is considered easy and $r = 8$ is very difficult especially for datasets with 37 genes.

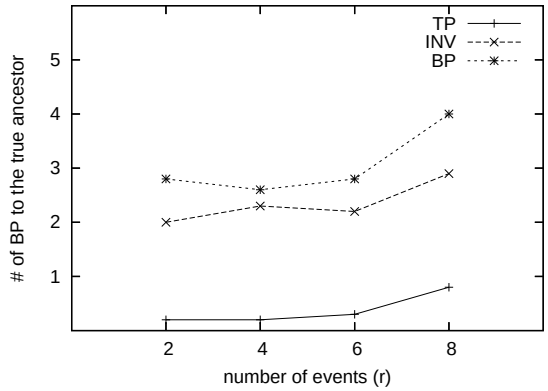
The gene orders on the leaves are generated by first assigning the identity permutation G_0 to the root, then evolving the permutation down the tree, applying along each edge a number of transpositions equal to the assigned edge length.

Given an estimated ancestor gene order G_M, we can use the breakpoint distance between G_M and G_0 as a measurement of how close the inferred ancestor is to the true ancestor. For each dataset, we obtain the estimated ancestors by using the following three methods: GRAPPA-TP (TP), breakpoint median solver (BP) and inversion median solver (INV). We repeat 100 times for each setting and the average of the results are reported.

Figure 5 shows the result. From this figure, we find that the median genomes returned by GRAPPA-TP are always the closest to the true ancestors. The medians returned by both breakpoint and inversion median solvers are further away from the true ancestors, a result mainly due to the usage of mismatched evolutionary models. Although breakpoint distance and median are generally viewed as not so sensitive to model mismatch, our testing results directly contradict this conjecture.



(a) 37 genes



(b) 100 genes

Fig. 5. Breakpoint distance from the inferred median to the true ancestor. TP indicates the result obtained from GRAPPA-TP, INV indicates the result obtained by using the Caprara's inversion median solver, and BP indicates the result obtained by using the breakpoint median solver.

As indicated above, our simple distance estimator uses a randomized approach, thus the number of repeats may have impact on its performance. To assess the impact, we compare GRAPPA-TP using two numbers of repeats: 1 and 10, and report the results in Figure 6. Surprisingly this figure shows that the impact of number of repeats is very small, even when the genomes are getting distant ($r = 6 \sim 8$).

B. Accuracy of Phylogeny Inference

We also test the performance of GRAPPA-TP on phylogeny analysis.

We first define our measure for the accuracy of reconstructed trees. Given an inferred tree, we compare its "topological accuracy" by computing "false negatives" and "false positives" with respect to the "true tree" [14]. For every tree there is a natural association between every edge and the bipartition on the leaf set induced by deleting the edge from the tree.

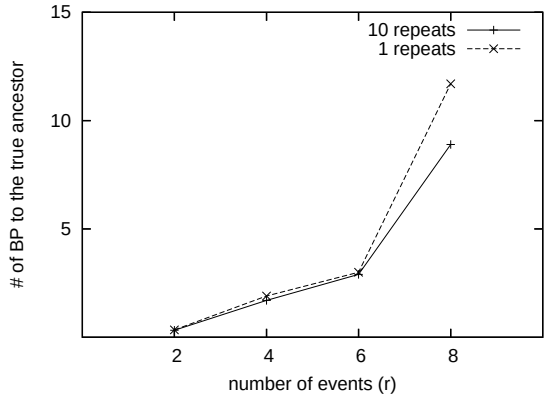


Fig. 6. Breakpoint distance from the inferred median to the true ancestor, using two number of repetitions for the distance computation.

Let T be the true tree and let T' be the inferred tree. An edge e in T is "missing" in T' if T' does not contain an edge defining the same bipartition; such an edge is called a *false negative* (FN). Note that the external edges (i.e. edges incident to a leaf) are trivial in the sense that they are present in every tree with the same set of leaves. The *false negative rate* is the number of false negative edges in T' with respect to T divided by the number of internal edges in T. The *false positive (FP) rate* is defined similarly, by swapping T and T'. The *Robinson-Foulds* (RF) rate is defined as the average of the FN and FP rates.

In this study, we generate uniformly random tree by randomly picking a tree from all possible trees—there are $(2N-5)\times(2N-7)\times\cdots\times 3$ trees for N taxa. We use trees with $N = 10$ and 37 genes, which is the number of genes in mitochondrial genomes. We choose $r = 2$, 3 and 4 to vary the level of difficulty, where $r = 4$ is considered very hard for for these datasets. For each combination of parameters, we generate 10 datasets and report the average results.

In our experiments, each dataset is tested using six methods: GRAPPA-TP (TP), GRAPPA using inversion median (INV), GRAPPA using breakpoint median (BP), NJ using transposition distances (TP-NJ), NJ using inversion distances (INV-NJ) and NJ using breakpoint distances (BP-NJ). Figure 7 shows the results; we placed a line at the 5% error level, the typical threshold of acceptability for accuracy in phylogenetic reconstruction [24].

We make the following two observations.

First, NJ has remarkably good performance when the genomes are close ($r = 2$), but its accuracy quickly dropped when the genomes are getting distant. Since NJ is guaranteed to be accurate when the distance between any pair of genomes is very close to the true distance, thus the result suggests that our distance estimator is valid when the genomes are close.

Second, GRAPPA-TP always returns highly accurate trees, although its performance is slightly worse than NJ for $r = 2$. The accuracy of GRAPPA-TP is also very stable and does not suffer when the genomes are relatively distant. Using breakpoint and inversion median solvers again give very bad results, even for easy datasets of $r = 2$. The results clearly show the importance of model match in genome rearrangement analysis. One should also note that unlike the results in median accuracy, using breakpoint medians in phylogenetic analysis has better performance than using inversion medians. More research in the future is needed to determine the factors contribute to this discrepancy.

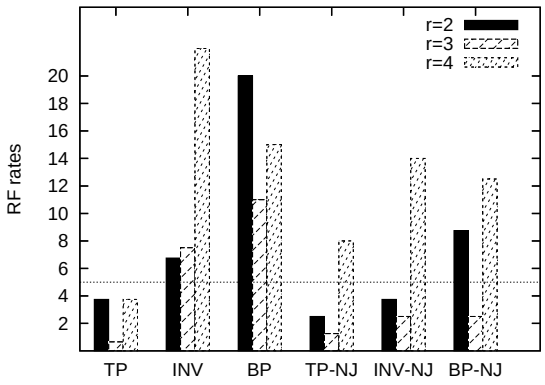


Fig. 7. RF errors for six methods under different expected number of events. The horizontal line indicates the acceptance threshold.

Although the number of genomes is relatively small in this test, the high accuracy of GRAPPA-TP makes it ideal as a base method for the DCM-GRAPPA developed by Tang et al. [25], hence can be easily extended to handle several hundred genomes.

V. Conclusions

In this paper, we present our new method to handle transpositions and report experimental results on simulated datasets. Although GRAPPA-TP uses a brute-force distance estimator, it remains very accurate for transposition phylogeny. Our studies suggest that model match is very important in both ancestor inference and phylogenetic reconstruction. The main problem of this method is of course its distance estimator, thus a fast and exact method to compute transposition distance is always desirable.

VI. Acknowledgments

FY and JT are supported by US National Institutes of Health (NIH grant number R01 GM078991-01) and by the University of South Carolina. MZ is supported by NSF of China No.60473099.

References

[1] Bader, D.A., B.M.E. Moret, and M. Yan (2001). A fast linear-time algorithm for inversion distance with an experimental comparison. *J. Comput. Biol.* 8(5), 483–491.

[2] Bernt M., D. Merkle, and M. Middendorf (2006). Genome rearrangement based on reversals that preserve conserved intervals. *IEEE/ACM Trans. on Comput. Biol. and Bioinformatics* 3 (3), 275–288.

[3] Blanchette, M. and D. Sankoff (1997). The median problem for breakpoints in comparative genomics. In T, Jiang and D.T. Lee (Eds.), *Proc. 3rd Comput. and Combin. Conf. (COCOON'97)*. Volume 1276 of *Lecture Notes in Computer Science*, 251–263.

[4] Blanchette, M. (1997) `derange2`, the software package is available at ftp://ftp.ebi.ac.uk/pub/software/unix/derange2.tar.Z.

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

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

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

[8] Cosner, M.E., R.K. Jansen, J.D. Palmer and S.R. Downie SR (1997) The highly rearranged chloroplast genome of Trachelium caeruleum (Campanulaceae): multiple inversions, inverted repeat expansion and contraction, transposition, insertions/deletions, and several repeat families. *Current Genetic* 31, 419–429.

[9] Downie, S. and J. Palmer (1992). Use of chloroplast DNA rearrangements in reconstructing plant phylogeny. In P. Soltis, D. Soltis, and J. Doyle (Eds.), *Plant Molecular Systematics*, 14–35.

[10] Elias, I. and T. Hartman (2005). A 1.375-Approximation Algorithm for Sorting by Transpositions. *Proc. 5th Workshop on Algorithms in Bioinformatics (WABI'05)*, 204-215.

[11] Felsenstein, J. (1978). The number of evolutionary trees. *Systematic Zoology* 27, 27–33.

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

[13] Hannenhalli, S. and P.A. Pevzner (1996). To cut… or not to cut (applications of comparative physical maps in molecular evolution). *Proc. 7th ACM-SIAM Symp. on Discrete Algorithms (SODA'96)*, 304–313, SIAM Press.

[14] Kumar, S. (1996). Minimum Evolution Trees. *Mol. Biol. and Evol.*, 15, 584–593.

[15] Larget, B., D.L. Simon, J.B. Kadane, and D. Sweet (2005). A Bayesian analysis of metazoan mitochondrial genome arrangements. *Mol. Biol. and Evol.*, 22 (3), 486–95.

[16] Moret, B.M.E., S. Wyman, D.A. Bader, T., Warnow, and M. Yan (2001). A new implementation and detailed study of breakpoint analysis. *Proc. 6th Pacific Symp. on Biocomputing (PSB'01)*, World Scientific Pub., 583–594.

[17] Moret, B.M.E., J. Tang, L.-S. Wang, and T. Warnow (2002). Steps toward accurate reconstructions of phylogenies from gene-order data. *Journal of Computer and System Sciences* 65(3), 508–525.

[18] Moret, B.M.E., A. Siepel, J. Tang, and T. Liu (2002). Inversion medians outperform breakpoint medians in phylogeny reconstruction from gene-order data. *Proc. 2nd Workshop on Algorithms in Bioinformatics (WABI'02)*. Volume 2452 of *Lecture Notes in Computer Science*, 521–536.

[19] Moret, B.M.E., J. Tang, and T. Warnow (2005). Reconstructing phylogenies from gene-content and gene-order data. In *Mathematics of Evolution and Phylogeny*, edited by O. Gascuel, Oxford University Press, 321–352.

[20] Pe'er, I. and R. Shamir (1998). The median problems for breakpoints are NP-complete. *The Electronic Colloquium on Computational Complexity* 71.

[21] Raubeson, L. and R. Jansen (1992). Chloroplast DNA evidence on the ancient evolutionary split in vascular land plants. *Science* 255, 1697–1699.

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

[23] Siepel, A. and B.M.E. Moret (2001). Finding an optimal inversion median: experimental results. *Proc. 1st Workshop on Algorithms in Bioinformatics (WABI'01)*. Volume 2149 of *Lecture Notes in Computer Science*, 189–203.

[24] Swofford, D.L., G. Olson, P. Waddell, and D.M. Hillis. Phylogenetic inference. In D.M. Hillis, C. Moritz, and B. Mable, editors, *Molecular Systematics, 2nd ed.*, chapter 11. Sinauer Associates, 1996.

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

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

[27] Wang, L.-S. (2001). Exact-IEBP: a new technique for estimating evolutionary distances between whole genomes. *Proc. 1st Workshop on Algorithms in Bioinformatics (WABI'01)*. Volume 2149 of *Lecture Notes in Computer Science*, 176-190.

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

[29] Yancopoulos, S., O. Attie and R. Friedberg (2005). Efficient sorting of genomic permutations by translocation, inversion and block interchange. *Bioinformatics*, Aug 2005, 21, 3340-3346.