Multiple genome rearrangement

David Sankoff * Mathieu Blanchette †

1 Introduction

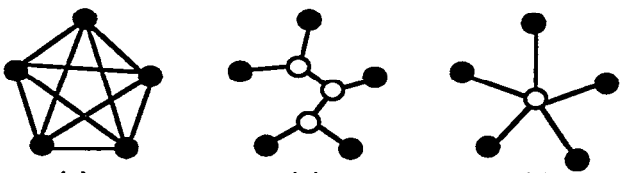
Multiple alignment of macromolecular sequences, an important topic of algorithmic research for at least 25 years [13, 10], generalizes the comparison of just two sequences which have diverged through the local processes of insertion, deletion and substitution. Recently there has been much interest in gene-order sequences which diverge through non-local genome rearrangement processes such as inversion (or reversal) and transposition (reviewed in [15] ch.7, [9], [6] ch. 19 and [3]). What would be the analog of multiple alignment under these models of divergence? In this introduction we first review some formulations of multiple alignment and show which have counterparts in multiple rearrangement. We then discuss the difficulties inherent in edit-distance formulations of multiple rearrangement, referring to relevant work, and argue for a potentially simpler approach based on "breakpoint analysis".

1.1 Multiple sequence alignment

The goal of multiple sequence alignment is to align the terms of N sequences into a number of relatively homogeneous columns through the judicious insertion of one or more null terms, or gaps, between consecutive terms in some or all of the sequences, as in the following figure, so as to optimize an objective cost function.

```
aabbcc                    a|a|b|b|  |c|c|
aebdced   ------------>     |a|e|b|d|c|e|d
aabdd                     a|a|  |b|d|  |  |d
aaebded                   a|a|e|b|d|  |e|d
```

In the simplest case, this objective is just the sum of column costs across all columns of the alignment. Each column cost measures how different the terms in that column are among themselves.For example, in a "complete" comparison the column cost is the number of pairs of sequences which differ in that column, as represented in (a) in the figure below,in which every vertex (sequence) is compared to every other.



Another definition of column cost depends on a given phylogenetic tree, as in (b) in the figure, in which the leaves are the data sequences and the internal nodes (open dots) are hypothetical ancestral sequences reconstructed by some method from the contemporary sequences. Both data and reconstructed sequences must be aligned and the column cost is just the number of tree branches where the sequences at the two ends have different elements in the column. A special case of the tree-based comparison is the "consensus" comparison, represented in (c), where there is just one reconstructed sequence.

There are many other formulations of the problem which we will not discuss, e.g. the column cost may be $N - M$, where M is the number of occurrences of the most frequent term in a column.

1.2 The analogy with genome rearrangement

A key difference between sequence comparison and gene-order comparison is that in the former, algorithms try to identify corresponding terms in the two sequences being compared and the number of divergence steps then falls out directly, whereas in the latter the correspondence (i.e. alignment) is given and it is the number of steps which must be calculated.

Thus version (a) of the multiple alignment problem has no analog in gene-order rearangement, since there is nothing to optimize once the pairwise distances are given. On the other hand, in versions (b) and (c) of the problem, there is something to optimize, namely the ancestral gene orders represented by the white dots. This is the focus of this article.

*Centre de recherches mathématiques, Université de Montréal, CP 6128 Succursale Centre-ville, Montréal, Québec H3C 3J7. E-mail: sankoff@ere.umontreal.ca.

†Centre de recherches mathématiques, Université de Montréal, CP 6128 Succursale Centre-ville, Montréal, Québec H3C 3J7. E-mail: blanchem@iro.umontreal.ca.

RECOMB 98 New York NY USA
Copyright 1998 0-89791-976-9/98/3..$5.00

1.3 Difficulties and a solution

There have been a number of investigations of phylogeny based on the algorithmic comparison of gene order within a number of genomes, using *pairwise* comparisons followed by *distance matrix* methods (e.g. [12]). However, treeing methods which involve the optimal reconstruction of gene order at ancestral nodes [7, 14] have been little used because of the computational difficulty in generalizing measures of genomic distance to more than two genomes. Caprara has recently shown that the most promising case, reversal distance for only three signed permutations, is NP-hard [4]. In [3] and [1] we argued that

(i) this computational difficulty – there are no algorithms guaranteeing exact solutions for even three relatively short genomes – together with
(ii) unwarranted assumptions as to the relative importance of different rearrangement events implicit in genome distances such as minimum reversal distance, minimum transposition distance, minimum translocation distance and even distances combining these (cf. [2]), as well as
(iii) the fallacy that calculation of an edit distance allows the recoverability of the "true" history of genomic divergence – in fact, the severe non-uniqueness of the optimal edit path for moderate or large gene-order distances has much worse (i.e. non-local) consequences than with the classical multiple alignment problem, and
(iv) the bias in simulations, where calculation of genome distance severely underestimates the actual number of events generating moderate or large gene-order differences,

all militate in favour of extending gene-order comparisons to three or more genomes through a much simpler and model-free metric. In this paper we suggest the number of breakpoints as just such a metric. We show how breakpoint analysis addresses all four of these problems.

In the Section 2 we define this measure for unoriented and oriented genomes, and set up the analogy to multiple alignment modes (b) and (c) above. In Section 3 we review how (c), consensus-based multiple rearrangement, can be solved exactly through reduction to a not-unwieldy version of the Travelling Salesman Problem, as proposed in [3]. In Section 4, we use this exact solution applied to simulated data to show how the problem of non-uniqueness is attenuated with increasing numbers of data genomes. In Section 5, we report how (b), tree-based multiple alignment, can be achieved to a great degree of accuracy by decomposing the tree into a number of overlapping 3-stars centred on the non-terminal nodes, and solving the consensus-based problem iteratively for these nodes until convergence. The accuracy depends on very careful initializations at the non-terminal nodes, as assessed through simulation in [1]. In Section 6 we investigate non-uniqueness by means of simulation and accurate heuristics again, this time focusing on the effect of the position of the node in the tree in terms of path length to the terminal vertices.

2 Breakpoint analysis

Consider two genomes $A = a_1 \dots a_n$ and $B = b_1 \dots b_n$ on the same set of genes $\{g_1, \dots, g_n\}$. We say a_i and a_{i+1} are adjacent in A. We also consider that a_1 is adjacent to the genome "start" and a_n is adjacent to the "end". For circular genomes, it suffices to consider that a_n and a_1 are adjacent. If two genes g and h are adjacent in A but not in B, they determine a breakpoint in A. We define $\Phi(A, B)$ to be the number of breakpoints in A. This is clearly equal to the number of breakpoints in B.

The number of breakpoints between two genomes is not only the most general measure of genomic distance, requiring no assumptions about the mechanisms of genomic evolution (inversion versus transposition versus translocation) underlying the data, but it is also the easiest to calculate. In addition, it has proven relations to the edit distances; e.g. half the number of breakpoints is a lower bound on the reversal distance.

2.1 Oriented genomes

Our simulations will involve directed, or oriented, genomes; we assume we know the strandedness, or direction of transcription, of each gene in each genome in the data set. In this case, the notion of breakpoint must be modified to take into account the polarity of the two genes [3]. If gh represents the order of two genes in one genome, then if another genome contains gh or $-h-g$ there is no breakpoint involved. However, between gh and hg there is a breakpoint, similarly between gh and $-g-h, g-h, -gh, h-g$ or $-hg$. Adjacency is no longer commutative.

2.2 Tree-based multiple genome rearrangement

The problem is formulated as follows: Let T=(V,E) be an unrooted tree with $N \geq 3$ leaves and $\Sigma = \{g_1, \cdots, g_n\}$ be a set of genes. Suppose $\{V_1, \cdots, V_N\} \subset V(T)$ are the leaves of the tree and $\{V_{N+1}, \cdots, V_L\}$, where $N < L \leq 2N-2$, are the internal vertices of the tree. The data consist, for each leaf $V_i, i = 1, \cdots, N$, of a circular permutation $G^i = g_1^i \cdots g_n^i$ of the genes in Σ, representing the genome of a contemporary species. The task is to find the permutations $G^{N+1}, \cdots, G^L$ associated with the internal (ancestral) vertices $V_{N+1}, \cdots, V_L$, such that

$$\sum_{V_iV_j \in E(T)} \Phi(G^i, G^j)$$

is minimized.

2.3 Binary tree- versus consensus-based multiple genome rearrangement

We will concentrate on two extreme cases: that of completely resolved, or binary, trees, where $L = 2N-2$ and all non-terminal nodes are of degree 3; and that of completely unresolved trees, or "stars", where $L = N+1$ and the single non-terminal node has degree N.

3 Consensus-based rearrangement

Though all these breakpoint-based multiple rearrangement problems seem NP-hard, as reported for $N = 3$ by Pe'er and Shamir [8], for moderate n they are tractable, since they may be reduced to a number of interconnected instances of the Traveling Salesman Problem (TSP). In the case of consensus-based rearrangement, the solution, involving just one TSP, is globally optimal.

We define Γ to be the complete graph whose vertices are the elements of Σ. For each edge gh in $E(\Gamma)$, let $u(gh)$ be the number of times g and h are adjacent in the N data genomes. Set $w(gh) = N - u(gh)$. Then the solution to TSP on (Γ, w) traces out an optimal genome S on Σ, since

if g and h are adjacent in S, but not in V_1, for example, then they form a breakpoint in S.

For oriented genomes, the reduction of the median problem to TSP must be somewhat different to take into account that the median genome contains g or $-g$ but not both. Let Γ be a complete graph with vertices $V(\Gamma) = \{-g_n, \ldots, -g_1, g_1, \ldots, g_n\}$. For each edge gh in $E(\Gamma)$, let $u(gh)$ be the number of times $-g$ and h are adjacent in the N data genomes and $w(gh) = N - u(gh)$, if $g \neq -h$. If $g = -h$, we simply set $w(gh) = -Z$, where Z is large enough to assure that a minimum weight cycle must contain the edge $-gg$.

Proposition: If $s = s_1, -s_1, s_2, -s_2, \ldots, s_n, -s_n$ is the solution of the TSP on (Γ, w), then the median is given by $S = s_1 s_2 \ldots s_n$.

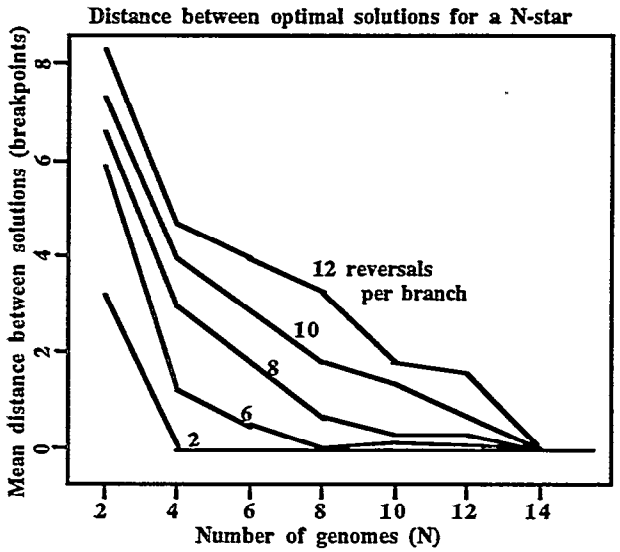
Proof: $\sum_{i=1}^{N} \Phi(S, V_i) = \sum_{gh \in S, g \neq -h} w(gh)$

$$= nZ + \sum_{gh \in s} w(gh).$$

Thus S minimizes $\sum_{i=1}^{N} \Phi(S, V_i)$ iff s is of minimal weight.

4 The uniqueness of the consensus

The reduction to TSP allows us to obtain global solutions for moderate-sized problems in reasonable time – typically 5 seconds for reconstructing the consensus of three or more scrambled genomes with 20 genes, on an Origin 200 computer with a RISC 10000 processor. This enables us to undertake systematic simulation studies. To assess the uniqueness of the solutions to the problem as a function of the number of genomes simultaneously rearranged, we constructed N genomes, each by applying a number R of random reversals to a common ancestor $(1, 2, \cdots, 20)$, and then solved the consensus problem. (Each reversal reverses the order of a number of consecutive terms, and also changes the sign of each of these terms. It adds at most two new breakpoints. We used reversals not because we privilege them as a model of biological evolution but simply as a convenient way of scrambling permutations.)



Though the TSP software we used (C.Hurwitz' tsp-solve) only produces one result, we were able to search for other results by permuting the labels of the 20 genes. We repeated each example with 10 different gene labelling.We compared all 10 solutions obtained by averaging their pairwise distances (i.e. the number of breakpoints between the two solution genomes). For each value of N between 2 and 16, and each value of R between 1 and 12, we repeated the experiment for 10 different examples, and averaged their results, running the programme 100 times – 10 examples times 10 gene labellings per example.

The figure above shows the results of these experiments. We note first that the curves for large R are systematically "worse" than those for low R; the more scrambled are the data genomes, the less likely they are to have a unique consensus.

More interesting perhaps, is the rapid, almost linear, decrease in non-unicity for each R as the number of data genomes increases. For 15 branches or more, there seems to be a unique consensus, no matter how large R is. And in all cases examined, this consensus order was none other than $(1, 2, \cdots, 20)$. Of course, for larger genomes, we could expect a larger cut-off point.

5 Binary tree-based rearrangement

A general method for the inference of ancestral genomes on a fixed binary tree is the iterative improvement method of [11], as adapted for the genomics context in [14, 5]. Each of the $N-2$ internal vertices, together with its three neighbors, defines a 3-star. The solution to the tree-based multiple rearrangement problem will have a reconstructed genome associated with each such vertex, which must be a solution to the consensus-based problem determined by these neighbors.

The strategy is to start with an initial tree where some genome is assigned to each internal genome, then to improve one of these ancestral genomes at a time by solving the consensus problem for the 3-star consisting of its immediate neighbours (the "median problem"), iterating across the tree until convergence. Of course, there is no guarantee that convergence will occur at a global optimum. The following algorithm formalizes this approach.

algorithm optimizetree
  input $G^1, \cdots, G^N$
  initialize each of $G^{N+1}, \cdots, G^{2N-2}$ to some genome.
  $cost \leftarrow \infty$
  routine iteratemedian
  output $G^{N+1}, \cdots, G^{2N-2}$

routine iteratemedian
  while $C = \sum_{V_i V_j \in E(T)} \Phi(G^i, G^j) < cost$,
    $cost \leftarrow C$
    do for $i = N+1, \cdots, 2N-2$,
      $G^* \leftarrow$ median(G^h, G^j, G^k), where V_h, V_j, V_k are the three neighbors of V_i
      if $\sum_{\{h,j,k\}} \Phi(G^*, G^x) < \sum_{\{h,j,k\}} \Phi(G^i, G^x)$
        $G^i \leftarrow G^*$
      endif
    enddo
  endwhile

The median routine is just the solution of the consensus-based rearrangement problem for 3 genomes, based on the reduction to the TSP in Section 3.

The main factor in directing convergence towards a global optimum is the how the initialization is carried out. We

can identify at least six distinct approaches to initialization, which can be grouped into three levels of increasing likelihood that they fall into the domain of attraction of a global optimum. Thus each internal genome can be assigned:

"arbitrarily",

(1) a fixed, arbitrary permutation, e.g.$(1, 2, \cdots, n)$, or
(2) a different random permutation

"reasonably",

(3) the permutation representing a nearest data genome, or
(4) the consensus of three nearest data genomes

"with much effort",

(5) by setting up and solving an initial TSP at each internal node, where the edge-weights are calculated by dynamic programming, minimizing the number of times a given adjacency has to be created or disrupted within the tree to be present or absent, respectively, at that node [1], or
(6) by setting up and solving an initial TSP at each internal node, where the edge-weights are the average of the corresponding edge-weights at the three neighbouring nodes, found by solving a system of linear equations [1].

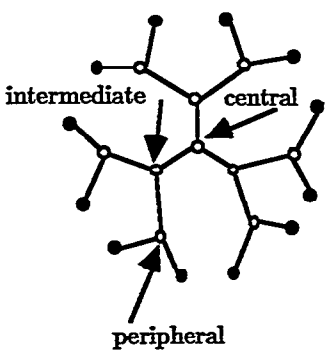
In addition, for the "reasonable" and "much effort" modes, all internal nodes can be initialized "recklessly", i.e. at once, or they can be initialized "cautiously", i.e. one at a time, starting with any internal node with two terminal node neighbours. Once it is initialized, it is treated as a terminal node as the initialization proceeds, and its two neighbours are disregarded.

In [1], we investigated methods (4),(5) and (6) above, the latter two incorporating the "cautious" approach. Simulations on trees generating seven to fifteen 20- and 30-gene data genomes showed that "much effort" paid off with one to two percent better results (i.e. fewer breakpoints over the entire tree) for moderate to highly divergent data. The dynamic programmming approach (5) led to better results than method (6) for moderately divergent data while the latter was superior for highly divergent data, approaching randomness, in each case by about one half of one percent.

In addition, we found that once the number of breakpoints generated per tree branch reached about half the number of genes, underestimation began to be manifested, rapidly worsening so that when the number of breakpoints per branch reached two thirds the number of genes, this number was underestimated by about 30%.

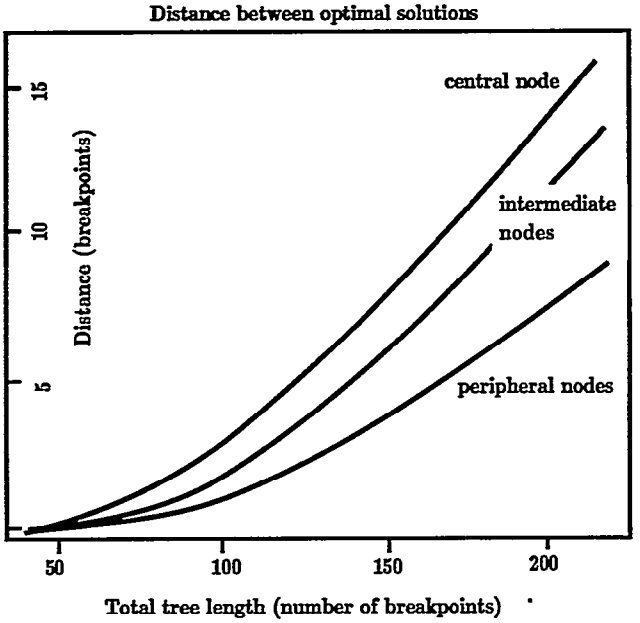
6 Uniqueness in tree-based rearrangement

In our investigation of multiple optima in reconstructed genomes, we simulated genomic rearrangement in the tree:



We again used genomes of size 20 and generated trees containing a total of B breakpoints over all their edges. We applied methods (5), (6) and (7) of Section 5 to calculate the multiple rearrangement for each tree. When at least two heuristics found the same total cost (which happened at least 70% of the time, even for highly divergent data), we calculated the distances between the genomes reconstructed by each method at each internal node.

This experiment was repeated 10 times (and the results averaged) for each value of B between 40 and 230. The results appear in the following figure:



The results of these simulations indicate that multiple solutions for peripheral nodes are relatively close to each other for low and moderate divergence. But with 10 breakpoints per branch, on an average, somewhat more than 200 breakpoints in all, multiple solutions could be non-negligibly far from each other – about 7 breakpoints between them on the average. The situation is progressively worse as we get deeper into the tree, so that there are considerably more breakpoints (around 15) between two solutions for the central genome than between two neighbouring nodes on the tree.

7 Conclusions.

Our previous work has established the feasibility of breakpoint analysis as a method of multiple genome rearrangement [3] compared to the difficulties in edit distance-based approaches, and assessed the relative reliability of various heuristics in achieving a global optimum [1]. To feasibility and reliablity, we add here the study of accuracy of genomic reconstruction, in terms of an analysis if the multiplicity of equivalent local minima and the breakpoint distances amongst them. Non-uniqueness remains a major consideration in genomic reconstruction, but we would conjecture that this is less of a problem in breakpoint analysis than with other approaches.

An important assumption in this work has been the fixed set of genes present in the data genomes. This is unrealistic in many contexts, but relaxing it makes multiple rearrangement and genomic reconstruction, much more difficult [3].

Acknowledgements

Research supported by grants to DS from the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Canadian Genome Analysis and Technology program, and a NSERC fellowship for graduate studies to MB. DS is a Fellow of the Canadian Institute for Advanced Research.

References

[1] M. Blanchette,G. Bourque and D. Sankoff. Breakpoint phylogenies. Genome Informatics 1997 (S.Miyano and T. Takagi, eds.) Tokyo: Universal Academy Press, 25-34, 1997.

[2] M. Blanchette, T. Kunisawa and D. Sankoff. Parametric genome rearrangement. *Gene-Combis* (online) and *Gene* 172, GC11-17, 1996.

[3] M. Blanchette and D. Sankoff. The median problem for breakpoints in comparative genomics. *Computing and Combinatorics, Proceeedings of COCOON '97.* (T.Jiang and D.T. Lee, eds) Lecture Notes in Computer Science 1276, Springer Verlag, 251-263, 1997.

[4] A.Caprara. Formulations and complexity of multiple sorting by reversals. ms., University of Bologna, December 1997.

[5] V. Ferretti, J.H. Nadeau and D.Sankoff. Original synteny. *Combinatorial Pattern Matching. Seventh Annual Symposium* (D.Hirschberg and G.Myers, ed.) Lecture Notes in Computer Science 1075, Springer Verlag, 159-167, 1996.

[6] D. Gusfield. *Algorithms on strings, trees, and sequences: computer science and computational biology.* Cambridge University Press, 1997.

[7] S. Hannenhalli, C. Chappey, E.V. Koonin and P.A. Pevzner. Genome sequence comparison and scenarios for gene rearrangements: a test case. *Genomics* 30: 299-311, 1995.

[8] I. Pe'er and R.Shamir (personal communication, January 1998).

[9] P. Pevzner and M.S. Waterman. Open combinatorial problems in computational molecular biology. In *Proceedings of the Third Israel Symposium on the Theory of Computing and Systems*, 158-173,1995.

[10] D. Sankoff. The early introduction of dynamic programming into computational biology. *Advances in the Mathematical Sciences - CRM's 25 years* (L. Vinet, ed.) CRM Proceedings and Lecture Notes., vol. 11, Providence, RI: American Mathematical Society., 403-413, 1997.

[11] D. Sankoff, R.J. Cedergren, and G. Lapalme. Frequency of insertion-deletion, transversion, and transition in the evolution of 5S ribosomal RNA. *J. Mol. Evol.*, 7:133-149, 1976.

[12] D. Sankoff, G. Leduc, N. Antoine, B. Paquin, B.F. Lang, and R. Cedergren. Gene order comparisons for phylogenetic inference: Evolution of the mitochondrial genome. *Proceedings of the National Academy of Sciences USA* 89: 6575-6579, 1992.

[13] D. Sankoff, C. Morel and R.J. Cedergren. Evolution of 5S RNA and the non-randomness of base replacement.*Nature New Biology* 245:232-234, 1973.

[14] D. Sankoff, G. Sundaram and J. Kececioglu. Steiner points in the space of genome rearrangements. *International Journal of the Foundations of Computer Science*, 7:1-9,1996.

[15] J. Setubal and J. Meidanis. *Introduction to Computational Molecular Biology.* Boston: PWS Publishing. 1997.