JOURNAL OF COMPUTATIONAL BIOLOGY
Volume 15, Number 8, 2008

Pp. 947–964
DOI: 10.1089/cmb.2008.0118

Descendants of Whole Genome Duplication within Gene Order Phylogeny

CHUNFANG ZHENG, QIAN ZHU, and DAVID SANKOFF

ABSTRACT

Genome doubling simultaneously doubles all genetic markers. Genome rearrangement phylogenetics requires that all genomes analyzed have the same set of orthologs, so that it is not possible to include doubled and unduplicated genomes in the same phylogeny. A framework for solving this difficulty requires separating out various possible local configurations of doubled and unduplicated genomes in a given phylogeny, each of which requires a different strategy for integrating genomic distance, halving and rearrangement median algorithms. In this paper we focus on the two cases where doubling precedes a speciation event and where it occurs independently in both lineages initiated by a speciation event. We apply these to a new data set containing markers that are ancient duplicates in two yeast genomes.

Key words: molecular evolution, sequence analysis, statistics.

1. INTRODUCTION

BASIC REARRANGEMENT PHYLOGENY methods require that the genomic content be the same in all the organisms being compared, so that every marker (whether gene, anchor, probe binding site or chromosomal segment) in one genome be identified with a single orthologous counterpart in each of the others, though adjustments can be made for a limited amount of marker deletion, insertion and duplication.

Many genomes have been shown to result from an ancestral doubling of the genome, so that every chromosome, and hence every marker, in the entire genome is duplicated simultaneously. Evidence for the effects of genome duplication has shown up across the eukaryote spectrum. Aside from the well-known controversy about doubling in vertebrates more than two hundred million years ago (Hughes, 1999; McLysaght et al., 2002), more recent genome duplications are known to have occurred in some vertebrate lines, such as the the zebrafish (Postlethwait et al., 1998), the salmoniform fish (Ohno et al., 1968), frogs (Xu et al., 1997) and even mammals, as evidenced by the rats *Tympanoctomys barrerae* (Gallardo et al., 1999) and *Pipanacoctomys aureus* (Gallardo et al., 2004). Genome duplication is widespread in insects and particularly prevalent in plants (for a survey of the flowering plants, see Cui et al. [2006]).

Departments of Biology, Biochemistry, and Mathematics and Statistics, University of Ottawa, Ottawa, Canada.

For some generations after a true doubling event (called autotetraploidization in classical genetics), the meiotic process, characterized by the normal bivalent pairings of one maternal and one paternal chromosome, may be disrupted by abnormal quadrivalent or trivalent alignments involving homeologous chromosomes, singleton chromosomes and other aberrant structures, resulting in reduced fertility. Doubling is followed by a period of "re-diploidization," where distinct pairings again emerge, though in approximately twice the original number, a process mediated by sequence divergence and by chromosomal rearrangement, through intra- and interchromosomal movement of genetic material. These rearrangement processes continue after diplosomy is attained, so that eventually the chromosomal neighborhood of a marker need bear no resemblance to that of its duplicate.

Doubling may also occur as a fusion of two distinct but related genomes (allotetraploidy) instead of autotetraploidy. With allotetraploidy, the period of unpredictable meiotic patterns may be absent or attenuated, to the extent that meiotic pairing remains true to the parental types, and for this reason it is thought that this process has a better chance of leading to viable offspring. Intra- and interchromosomal movement of genetic material will also eventually result in the karyotypic mixture of the two founding genomes after this type of doubling.

The present-day genome, which we refer to here as a doubling descendant, can be decomposed into a set of duplicate or near-duplicate markers dispersed among the chromosomes.

For descendants of autotetraploids, there is no direct way of partitioning the markers into two sets according to which ones were together in the same half of the original doubled genome. In the case of allotetraploids, if there are data on unduplicated species more closely related to one of the contributing genomes than the two contributors are to each other, this may allow systematically partitioning the pairs into two sets accordingly. The mathematics of this case are different (El-Mabrouk and Sankoff, 1999), and will not be discussed here. When there are no intervening unduplicated genomes, it may not even be possible to detect that this was an allotetraploid rather than an autotetraploid and there will be little distinction to be made on the practical level for the analysis of the two cases.

Genomic distance or rearrangement phylogeny algorithms are not applicable to doubling descendants, since there is a two-to-one relationship between markers in the doubling descendant and related species whose divergence predates the doubling event, or unresolved two-to-two relationships between two doubling descendants, whereas these algorithms require a one-to-one correspondence.

We have undertaken a program (Zheng et al., 2006; Sankoff et al., 2007) of studying rearrangement phylogeny where doubling descendants are considered along with related unduplicated genomes. We believe there is no other computationally-oriented literature on this particular problem. To focus on the problem of marker ambiguity in doubling descendants, and to disentangle it from the difficulties of constructing phylogenies, we pose our computational problems only within the framework of the "small" phylogenetic problem, i.e., identifying the ancestral genomes for a given phylogeny that jointly minimize the sum of the rearrangement distances along its branches.

In Section 2, we outline a model for generating an arbitrary pattern of doubling descendants observed at the tips of a given phylogeny. Based on this model, we then present an simple algorithm for inferring the doubling status of the ancestral genomes in terms of an economical set of doubling events along the branches of the phylogeny. Once we have the ancestral doubling statuses, we can approach the actual rearrangement problem.

First, in Section 3, we identify the three basic algorithms underlying the study of genome rearrangements and gene order phylogeny, one a calculation of the genomic distance between two given genomes with clearly identified orthologs, i.e., the minimum number of rearrangements necessary to transform one genome into another; the second a "halving" algorithm for inferring the ancestor of a doubled genome based on internal evidence from its modern descendant only, and the third a "medianizing" process for inferring an ancestral genome from its three neighboring genomes in a binary branching tree.

In Section 4, we review the methodology for the small phylogeny problem using gene order data, based on the iterative application of the median algorithm successively to all the ancestral vertices of a phylogenetic tree. When doubling descendants are considered together with unduplicated genomes, there are four kinds of median problem; these are reviewed in Section 5.1. In Section 5.3, we discuss our recent papers (Zheng et al., 2006; Sankoff et al., 2007) on incorporating the three basic algorithms: distance,

halving and median, into an overall procedure for inferring ancestral genomes in the case of one doubling descendant and two related unduplicated genomes. The contribution of the present paper starts in Section 6 where we analyze two ways of relating genomes from two doubling descendants, one where they result from a single genome doubling event followed by a speciation, and the other where speciation precedes two genome doublings, one in each lineage. To systematically assess the biases in these approaches, in Section 7, we carry out a series of simulations. In Section 8, we apply these two methods to a large data set on yeast.

1.1. Terminology and scope

In biology, the concept of genome doubling is usually expressed as tetraploidization or autotetraploidization, and the both the doubled genome and its doubling descendant are called tetraploid, even though, generally, the descendants soon undergo a process called (re-)diploidization and function as normal diploids, still carrying many duplicate markers that evolve independently of each other. Though unambiguous in biological context, implicit in this terminology are many assumptions that are not pertinent to our study. In the yeast data we study here, for example, *Saccharomyces cerevisiae* exists during most of its life cycle as a haploid, only sometimes as a diploid, while *Candida glabrata* exists uniquely as a haploid.

In our considerations, the key aspect of genome doubling is the global duplication of chromosomes and markers at the moment of doubling. Whether these are haploid, diploid or some other ploidy is not relevant in that in any organism that reproduces by meiosis, the order of the markers on any of the aligned components (e.g., maternal versus paternal chromosomes) is essentially identical. There may be different alleles, or other local differences, but the order is basically invariant. Ongoing variation and evolution at the level of chromosomal structure in an individual or species are considered negligible in comparison with the major rearrangements that exist between genomes separated on an evolutionary time scale.

Although this paper is about polyploidy, then, we will rely largely on terminology independent of ploidy: genome doubling, doubling descendant, unduplicated genomes, genome halving.

2. INFERENCE OF DOUBLING EVENTS

Our algorithms require genomic sequence data or other high resolution marker data spanning the entire genome. This, of course, is only available in a limited number of phylogenetic domains within the eukaryotes, and then only from selected organisms. Our analysis may also benefit from information on doubling status not only about the sequenced or mapped genomes, but also from closely related organisms. Fortunately mapping information is much easier to obtain experimentally and to come by in the literature than complete sequence information, though the nature and timing of ancestral events often require inferential leaps based on the number of chromosomes or the distribution of the number of copies of each marker.

Our first task, given some mixture of doubling descendants and unduplicated genomes related by a phylogenetic tree, is to infer the doubling status of the all the ancestral genomes. Under the simplifying assumptions that all ploidies are powers of two and can only remain unchanged or change by a factor of two at each step, and the parsimony criterion that the number of doubling steps is to be minimized, the task is achieved by the recurrence

$$\Pi(v) = \min_{\text{daughter species } u \text{ of v}} \Pi(u)$$

at each ancestral vertex v of a phylogenetic tree, as depicted in Figure 1.

Once Π is inferred, the doubling events may be inferred to occur on those branches of the tree where the Π differs at the two ends. This is also depicted in Figure 1.

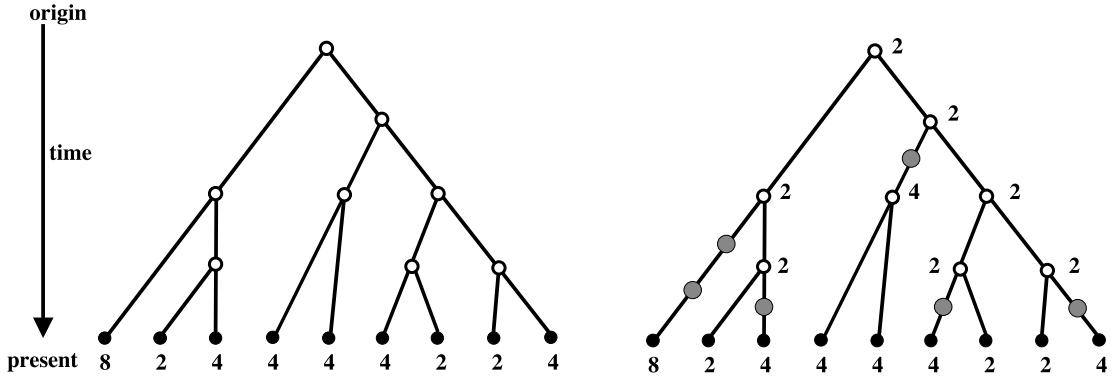

FIG. 1. Example of doubling inference problem. Genomes observed only for leaves (filled dots) of phylogeny. 2 = (diploid) unduplicated genome. Inferred doubling events indicated by gray dots.

3. BASIC ALGORITHMS

Once we have inferred the doubling status of the ancestral genomes, how are we to approach our original problem: to reconstruct the marker order of the ancestral genomes and thus infer the cost of the phylogeny in terms of rearrangement events? Here we discuss some basic elements of the solution.

Genomic distance. Distance based on genomic structure $d(X, Y)$ is calculated by linear-time rearrangement algorithms for finding the minimum number of operations necessary to convert one genome X into another Y. The biologically-motivated rearrangement operations we consider include inversions (implying as well change of orientation) of chromosomal segments containing one or more markers, reciprocal translocations (of telomere-containing segments—suffixes or prefixes—of two chromosomes) and chromosome fission or fusion. Genomic distance is defined only between undoubled genomes.

The essence of the various algorithms[1] for genomic distance resides in the "breakpoint graph" of the two genomes being compared. In this graph separate vertices are defined for the $5'$ and $3'$ end of each gene, and the adjacencies between two genes are represented by a black edge between the adjacent vertices in one genome and a gray edge between the adjacent vertices in the other genome. Though there are competing, and equivalent, formalisms for this, the graph thus defined decomposes into c cycles and paths of alternating gray and black edges, and it can be shown that the genomic distance is of the form $d(X, Y) = n + \chi - c$, where χ is the number of chromosomes in each genome.

Genome halving. Given a genome T containing a set of markers, each of which appears twice on the genome, on the same or on different chromosomes, how can we construct a genome A containing only one copy of each marker, and such that the genome $A \oplus A$ consisting of two copies of each chromosome in A minimizes $d(T, A \oplus A)$? Here we use a linear-time algorithm for solving this problem (El-Mabrouk and Sankoff, 2003).

Rearrangement median. Given three genomes X, Y, and Z, how can we find the *median* genome M such that $d(X, M) + d(Y, M) + d(Z, M)$ is minimized. For this NP-hard problem, we have a new implementation of a previous heuristic (Adam and Sankoff, 2008), which uses the principles of Bourque's MGR (Bourque and Pevzner, 2002; Siepel and Moret, 2001), but is based on the constrained version of the Bergeron et al. (2006) algorithm mentioned in the next to last paragraph. For the instances involving a few hundred genes we studied, this algorithm proposes a solution after a few minutes to a few hours computation on a MacBook computer.

[1]For our calculations in the present paper, we actually use the versatile rearrangement algorithm of Bergeron et al. (2006), which we constrain to allow only the operations we have listed. This avoids difficulties of graph-theoretical programming associated with the breakpoint graph, while arriving at essentially the same result.

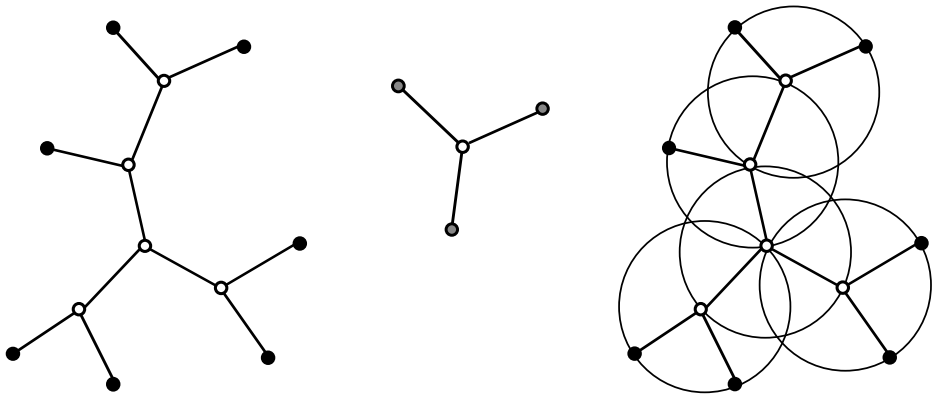

FIG. 2. **(Left)** Unrooted binary phylogenetic tree; filled dots represent given genomes, open dots represent ancestral genomes to be inferred. **(Center)** Smallest phylogeny requiring solution to the median problem. Gray dots can represent either given or ancestral genomes. **(Right)** Decomposition of phylogenetic inference problem on left into overlapping median problems.

4. GENE ORDER PHYLOGENY BASED ON ITERATIONS OF A MEDIAN ALGORITHM

One of the main approaches to gene-order phylogeny is based on iterating the median algorithm to overlapping parts of a given binary branching tree as illustrated in Figure 2. Though we can reconstruct the ploidy at the root using the methods of Section 2, and while knowing the gene order of the ancestor would, of course, be of great interest, the rearrangement approach produces inferences only about tree nodes of degree three or higher, and the reversibility of the rearrangement operations preclude any notion of the time direction of the tree edges. So whether the given tree is rooted or not is irrelevant to our analysis. In fact, we can assign the status of root to any node in the tree without changing the gene order results. Thus we work with unrooted trees, with the possibility that the position of the root is known from previous biological work or phylogenetic analysis.

An unrooted tree with all non-terminal vertices of degree three as on the left of the figure can be decomposed into overlapping median problems as on the right, where the three "given" genomes in the centre of the figure may be terminal vertices or ancestral vertices with previously calculated or assigned genomes.

The median algorithm is applied to each ancestral vertex in turn, perhaps involving several passes over the set of ancestors, until no further improvement is obtained. In the first pass, the ancestral vertices must be initialized, either using random genomes, copies of the closest given genome, or some other strategy.

5. MEDIAN-BASED PHYLOGENY CONTAINING DOUBLING DESCENDANTS AND UNDOUBLED GENOMES

In the case where some of the given genomes are descended from whole genome duplication events, as evidenced by widespread pairs of paralogs, we can infer through the methods of Section 2 which ancestors are likely to have been doubling descendants. Then when we decompose the tree into median problems, we may encounter any of the four configurations in Figure 3.

5.1. The four cases

We introduce some notation: Let T be a doubling descendant, i.e., with n different chromosomes, and $2m$ markers, $g_{1,1}, \ldots, g_{1,m}; g_{2,1}, \ldots, g_{2,m}$, dispersed in any order on these chromosomes. Each $g_{i,j}$ is signed positive or negative to indicate the reading direction of the DNA strand it is on. For each i, we call $g_{1,i}$ and $g_{2,i}$ "duplicates," and the subscript "1" or "2" is assigned arbitrarily. A potential ancestral doubled genome of T is written $A \oplus A$, and consists of $2n'$ chromosomes, where some half (n') of the chromosomes

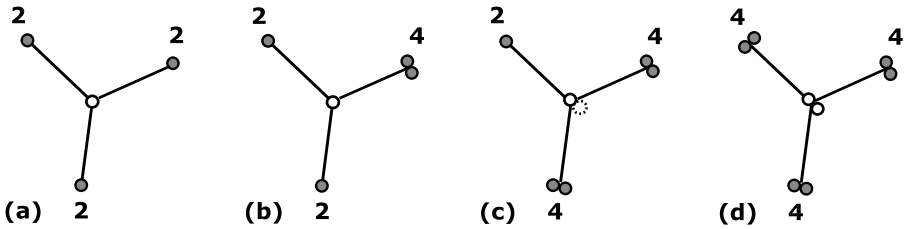

FIG. 3. Four possible median configurations of undoubled (denoted "2") and doubling descendants (denoted "4"). **(a)** Classical median problem with three undoubled genomes. **(b)** Genome halving with two outgroups. **(c)** Two doubling descendants from either one (include dotted circle) or two (without dotted circle) doubling events, as studied in this paper. **(d)** Median problem with three doubling descendants. Although the inference of the median takes no account of the position of the root, the considerations of Section 2 imply that in all cases (a)–(d), we may assume either that the median is the root and the three edges lead to its offspring or that the upper left genome is the root or, especially in (d), on a path from the median to the root. In the latter case the median genome, which is unknown, is one of the two offspring of the upper left genome, the other two genomes shown are the offspring of the median, either or both of which can be given or unknown (but initialized).

contains exactly one of each of $g_{1,i}$ or $g_{2,i}$ for each $i = 1, \ldots, m$. The remaining n' chromosomes are each identical to one in the first half, in that where $g_{1,i}$ appears on a chromosome in the first half, $g_{2,i}$ appears on the corresponding chromosome in the second half, and *vice versa*. We define A to be either of the two halves of $A \oplus A$, where the subscript 1 or 2 is suppressed from each $g_{1,i}$ or $g_{2,i}$. These n' chromosomes, and the m markers they contain, $g_1, \ldots, g_m$, constitute a potential ancestor of T that incurred the doubling event.

5.2. Case (a)

This case is just the classical median problem, and may be solved using the previously cited methods.

5.3. Case (b)

We first consider a simpler problem called Genome halving with an outgroup. Consider T and a related unduplicated genome R with markers orthologous to $g_1, \ldots, g_m$. The problem is to find an unduplicated genome A that minimizes

$$D(T, R) = d(R, A) + d(A \oplus A, T). \tag{1}$$

The solution in Zheng et al. (2006), as represented on the left of Figure 4, is to generate the set $\mathbf{S}$ of genome halving solutions, then to focus of the subset $X \in \mathbf{S}' \subset \mathbf{S}$ where $d(R, X)$ is minimized. We then

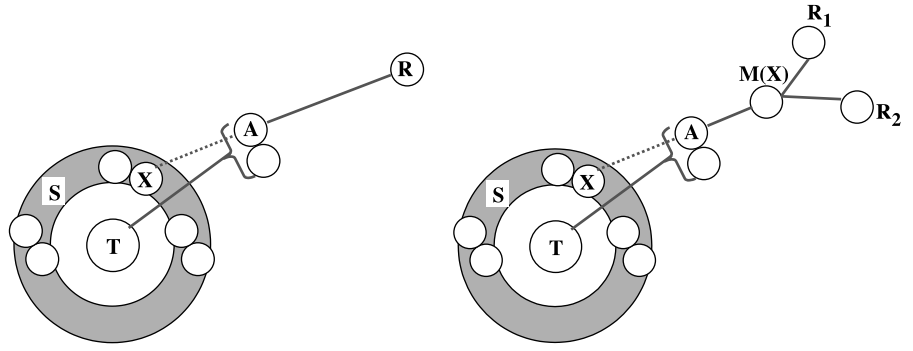

FIG. 4. Halving a doubling descendent T, with one (R) or two (R_1, R_2) unduplicated outgroups. The double circles represent two copies of potential ancestral genomes, including solutions to the genome halving in the shaded ring $\mathbf{S}$. Ancestors labeled A, which minimize D, are found on best trajectories between solutions $X \in \mathbf{S}$ and the outgroup (or between X and M, the median of X, R_1, and R_2).

minimize $D(T, R)$ by seeking heuristically for A along any trajectory between elements of $\mathbf{S}'$ and the outgroups.

We can use these same ideas for Case (b) in Figure 3. Consider T and two unduplicated genomes R_1 and R_2 with markers orthologous to $g_1, \ldots, g_m$. Our problem here is to find a diploid genome A and a median genome M of A, R_1 and R_2 that minimize

$$D(T, R_1, R_2) = d(R_1, M) + d(R_2, M) + d(A, M) + d(A \oplus A, T). \tag{2}$$

Our solution in Sankoff et al. (2007), as on the right of Figure 4, is to generate the set $\mathbf{S}$ of solutions of the genome halving problem, then to focus of the subset $X \in \mathbf{S}' \subset \mathbf{S}$ where $d(R_1, M) + d(R_2, M) + d(X, M)$ is minimized. Then the A minimizing $D(T, R_1, R_2)$ is sought, heuristically, along all trajectories between all elements $X \in \mathbf{S}'$ and $M(X)$.

5.4. Cases (c) and (d)

The present paper is most pertinent to Case (c). In the ensuing sections, although we will not directly address the case of two doubling descendants and one unduplicated genome, we will deal with the difficulties inherent in comparing two doubling descendants. The procedures we develop can then be extended to encompass the third genome, the unduplicated one, much as in Case (b).

Case (d), where the orthology relations among the two copies of a marker in one genome and the two copies in the other genomes is not known, remains to be investigated, although many of the considerations of Case (c) will be pertinent. If the various orthology relations are known, then Case (d) reduces to Case (a).

6. THE CASE OF TWO DOUBLING DESCENDANTS

Two related doubling descendants may arise in two ways, depending on the timing of the speciation event in relation to the doubling. Either speciation at V follows a single doubling event, as at A on the left of Figure 5, or the speciation precedes two independent doubling events in the two lineages, as at A and B on the right of the figure. Knowing which of the two scenarios is correct depends on knowing whether their common ancestor is doubled or not, information obtained from the algorithm in Section 2 or other data.

We will introduce new methods based on tweaking the distance and halving algorithms, conserving the optimality of the solutions, but allowing one of them to affect the arbitrary choices required to construct the solution for the other. First we sketch the original halving algorithm.

6.1. Halving

Without entering into all its details, we can present enough of the essentials of the halving algorithm to understand the techniques we use in our heuristics. There are two parts to algorithm, the first, **con-**

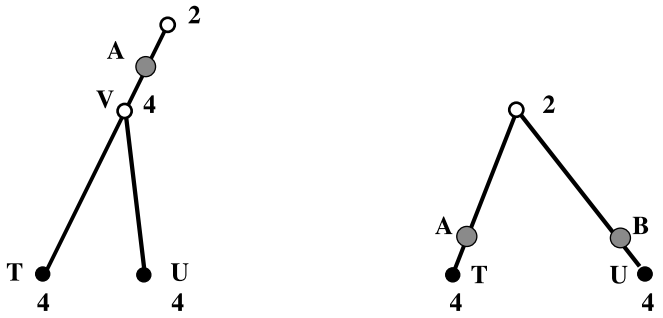

FIG. 5. **(Left)** Doubling, then speciation. **(Right)** Speciation, then two independent doublings. Numbers indicate ploidy as in Figure 1.

struct_SNGs, being the partitioning of the adjacencies between markers in a doubling descendant T into a set of *supernatural graphs* (SNGs) such that all edges in the breakpoint graph (defined in Section 3) induced by T and $A \oplus A$ connect vertices within the same SNG. The second part of the algorithm, **construct**_$A \oplus A$ is actual construction of the adjacencies in $A \oplus A$ within the SNGs.

As a first step in **construct_SNGs**, whose pseudocode is presented below, each marker x in a doubling descendant is replaced by an oriented pair of vertices (x_t, x_h) or (x_h, x_t) depending if the DNA is read from left to right or right to left. The duplicate of marker $x = (x_t, x_h)$ is written $\bar{x} = (\bar{x_t}, \bar{x_h})$. Of course $\bar{\bar{a}} = a$.

Following this, for each pair of neighboring markers, say (x_t, x_h) and (y_h, y_t), the two adjacent vertices x_h and y_h are linked by a black edge, denoted $\{x_h, y_h\}$ in the notation of Bergeron et al. (2006). For a vertex at the end of a chromosome, say y_t, it generates a edge of form $\{y_t, O\}$, where O is a dummy symbol.

The edges thus constructed are then partitioned into *natural graphs* according to the following principle: If an edge $\{a, b\}$ belongs to a natural graph, then so does some edge of form $\{\bar{a}, c\}$ and some edge of form $\{\bar{b}, d\}$. This key step is underlined in the pseudocode for the algorithm below.

If a natural graph has an even number of edges, as illustrated on the left of Figure 6, it can be shown that in all optimal ancestral doubled genomes, the edges colored gray, representing adjacent vertices in the ancestor (recall the definitions in Section 3), and incident to one of the vertices in this natural graph, necessarily have as their other endpoint another vertex within the same natural graph.

Gray edges are added in pairs, so that each edge in the reconstructed doubled ancestor is duplicated, in accordance with the required output of the algorithm to produce two copies of the same ancestral genome. (Whether a is connected to b and $\bar{a}$ to $\bar{b}$ or a to $\bar{b}$ and $\bar{a}$ to b is immaterial, since the two versions of the same gene were originally identical.) These edges are added in such a way as to maximize the number of (alternating colored) cycles and paths that make up this subgraph, in accordance with the definition of genomic distance in Section 3.

For natural graphs with an odd number of edges, it is impossible to complete them by adding pairs of gray edges. Nevertheless, as on the right of Figure 6, they may be grouped pairwise into *supernatural graphs* so that they may be completed with pairs of gray edges. Then, as with even natural graphs, an optimal doubled ancestor exists such that the edges colored gray incident to any of the vertices in a supernatural graph have as their other endpoint another vertex within the same supernatural graph.

The rules for drawing the gray edges, contained in **construct**_$A \oplus A$ (not repeated here), can be found in El-Mabrouk and Sankoff (2003).

Along with the multiplicity of solutions caused by different possible constructions of supernatural graphs (choices of u and t to join), within these SNGs there may be many ways of drawing the gray edges during the procedure **construct**_$A \oplus A$. Without repeating here the lengthy details of **construct**_$A \oplus A$, it suffices to note that these alternate ways can be generated by choosing a different one of the vertices within the SNG as a starting point.

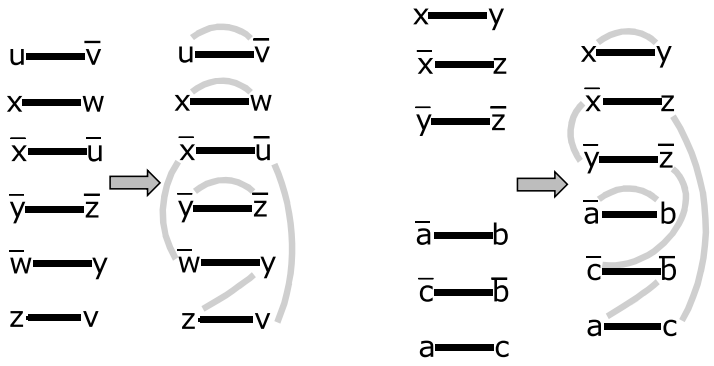

FIG. 6. **(Left)** Even-size natural graph completed by adding three pairs of gray edges in a way that maximizes the number of cycles. **(Right)** Two odd-size natural graphs, containing x, y, z vertices and a, b, c vertices, respectively, combined into one supernatural graph so that three pairs of gray edges may be added.

Algorithm 1. Construct_SNGs

Input: doubling descendant T
Output: set of SNGs for T
for *all* x *in* T **do**
define oriented pair of vertices (x_t, x_h) or (x_h, x_t), depending on whether X is signed positive or negative;
// The duplicate of a marker, e.g., $x = (x_t, x_h)$, is denoted $\bar{x} = (\bar{x_t}, \bar{x_h})$.
end
for *all pairs of neighboring markers, e.g.,* (x_t, x_h) *and* (y_h, y_t), **do**
create unoriented black edge $\{x_h, y_h\}$ between adjacent vertices
end
for *each vertex at the end of a chromosome, e.g.,* y_t, **do**
create unoriented black edge $\{y_t, O\}$
end
All black edges and all vertices are initially unassigned to natural graphs;
while *there are still unassigned edges* **do**
choose one and assign it (and its two incident vertices) to a new natural graph s;
while *there is a vertex* a*, other than* O*, in* s *such that vertex* $\bar{a}$ *is not in* s **do**
assign $\bar{a}$ to s, as well as the previously constructed edge $\{\bar{a}, b\}$ and vertex b
end
end
Natural graphs with even numbers of edges are automatically supernatural graphs;
while *there are natural graphs* u *and* t *both containing an odd number of edges* **do**
two such are joined to form a supernatural graph $s = u \cup t$;
// priority is given to constructing supernatural graphs with four O's instead of two. Other details on which u and t to join, and how to handle odd numbers of natural graphs, in El-Mabrouk and Sankoff (2003)
end

6.2. *Ortholog identification*

When comparing a doubling descendant to an undoubled related genome, there is no question of which of the two paralogs of a gene in the former is the true ortholog of the corresponding single gene in the latter: the two paralogs have precisely the same historical relationship with the single gene because they were both identical at the instant of doubling.

When comparing two descendants of the same doubling event that have subsequently diverged though a speciation event, however, we cannot avoid the problem of identifying two orthologous pairs of genes, each involving one gene in each genome. Evolution will have affected the two copies of the original gene differently in the time interval between the doubling and speciation events, for example with respect to the genomic contexts they are found, so that at the time of speciation and thereafter, there are two distinct pairs of orthologous genes deriving from a single gene pre-duplication, as in Figure 7. Depending on the time elapsed between doubling and speciation, sequence divergence analysis may help identify the correct orthologies via gene tree/species tree methodology, and further phylogenetic information may also be brought to bear, but this type of information may not always be available.

Another way of deducing the orthologies is through parsimony. Incorrectly posited orthologies will tend to increase the genomic distance between the two doubling descendants, so it is reasonable to look for the ensemble of orthology assignments that minimizes this distance.

In Sections 6.3 and 8, we will deal with both the case where the correct orthologies are given, and the case where they must be inferred. Note that if the two doubling descendants are descended from separate, post-speciation, doubling events, as to be described in Section 6.4 and also in Section 8, the question of correct orthology identification does not arise, much as in the case of one doubling descendant and one unduplicated genome, because the two paralogs in one doubling descendant are both related in exactly

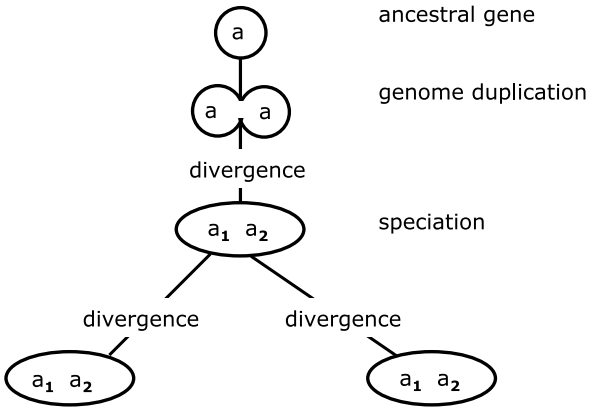

FIG. 7. Why ortholog identification is pertinent to the doubling first scenario. The genes labeled a_1 in the two genomes are orthologous, as are the two labeled a_2.

the same way to each of the two paralogs in the other; each pair is derived independently from the same single-copy gene.

Note that when we carry out the halving algorithm, this effectively partitions the genes among the chromosomes, and tells us which copy of each gene were together on a chromosome in the original doubled genome (original synteny). Of course, the other copy of each of these genes must also all have been together on a single chromosome. The algorithm, however, does not partition the ancestral chromosomes thus defined into two classes corresponding to the two original copies of the doubling genome, as this distinction has no biological meaning past the moment of doubling, and so has no trace in the doubling descendant. In any case, given the considerable degree of non-uniqueness in the halving algorithm, inferences of original synteny based on halving alone may be unreliable.

6.3. Doubling first

Given two doubling descendants T and U as on the left of Figure 5, we would ideally like to find the doubling descendant V that minimizes $d(T, V) + d(V, U) + d(V, A \oplus A)$, where A is any solution of the halving problem on V. Though d is calculated in linear time, multiple genome rearrangement problems based on d (e.g., the median problem in Section 3) are hard, so here we propose a somewhat constrained version of our problem, where V is assumed to be on a shortest trajectory between T and U. Because $d(T, V) + d(V, U) = d(T, U)$ is then constant, the problem becomes that of finding V to minimize $d(V, A \oplus A)$.

Because it is an edit distance, a genomic distance measurement $d(T, U)$ is associated with at least one trajectory containing $d(T, U) - 1$ genomes as well as T and U themselves, where each successive pair of genomes along the trajectory differ by exactly one rearrangement operation.

Before explaining a greedy (with look-ahead) heuristic for a solution to the constrained version of the problem, we recall the edge notation we use to represent the adjacencies in a genome (Bergeron et al., 2006). If two vertices a and b from different markers are adjacent in a genome, we represent this by an edge $\{a, b\} = \{b, a\}$; for a vertex c is at the end of a chromosome and hence adjacent to no other vertex, we construct a virtual edge $\{c, O\}$. Then any rearrangement operation can be represented by an operation on one or two terms in the representation, such as $\{a, b\}, \{c, d\} \rightarrow \{b, d\}, \{a, c\}$ or $\{a, b\} \rightarrow \{b, O\}, \{a, O\}$ or $\{a, b\}, \{c, O\} \rightarrow \{b, O\}, \{a, c\}$.

We initially define $T^* = T, U^* = U$ and associate this pair with the root node of a search tree. Then our heuristic consists of a search, at each step, for the "most promising" operation that moves T^* towards U^* or U^* towards T^*. For each operation, we define a score $W = x + 6y$ as follows. The y component will measure whether the operation actually diminishes $d(V, A \oplus A)$, while the x will measure whether the operation increases the potential of diminishing $d(V, A \oplus A)$ in a subsequent operation.

In a greedy approach, y should be more heavily weighted than x. But how do we justify the coefficient 6 of y? Consider the possible operations that remain on a trajectory from T to U, i.e., if V_1 is transformed

into V_2 by the operation, then $d(T, V_2) = d(T, V_1) + 1$ and $d(V_2, U) = d(V_1, U) - 1$. Let A_1 and A_2 be solutions of the halving problem for V_1 and V_2, respectively. Even if $A_2 \neq A_1$, we have $|d(V_1, A_1 \oplus A_1) - d(V_2, A_2 \oplus A_2)| \leq 1$, because only one operation is involved. We set $y = d(V_1, A_1 \oplus A_1) - d(V_2, A_2 \oplus A_2) + 1$, so that y is in the range $[0, 2]$.

In evaluating an operation changing T^*, such as $\{a, b\}, \{c, d\} \rightarrow \{b, d\}, \{a, c\}$, we consider the following eight pairs: $\{a, b\}, \{c, d\}, \{b, d\}, \{a, c\}, \{\bar{a}, \bar{b}\}, \{\bar{c}, \bar{d}\}, \{\bar{b}, \bar{d}\}, \{\bar{a}, \bar{c}\}$.

The operation would clearly seem advantageous for subsequent operations if $\{\bar{b}, \bar{d}\}$ and/or $\{\bar{a}, \bar{c}\}$ were in T^* and/or U^*. There are from zero to four advantageous possibilities. In addition, although one of $\{b, d\}, \{a, c\}$ must be in U^* for the operation not to veer from an optimal trajectory, it is not necessary that both of them be. There are zero or one advantageous possibilities. We count how many h of the total of five advantageous possibilities occur and set $x = h + 1$. Then x is in the range of $[1, 6]$.

The score W is in the range of $[1, 18]$. Note that the effect of y always outweighs that of x in in the formula $W = x + 6y$. Even if $x = 6$, this can only happen if $y = 2$, since having all five advantageous possibilities can only happen in the context of V_2 having a smaller halving cost than V_1. Thus, the greedy approach mandates a coefficient of y of at least 6; we chose a coefficient of exactly 6 in order to maximize the effect of the look-ahead.

We calculate W_{T*} in this way and W_{U*} by considering operations changing U^* in the direction of T^*. Let $W_X = \max_{\text{all operations}} W_{X*}$.

If $W_T \geq W_U$ and $W_T \geq 6$, we apply the highest score operation to T^*. Otherwise apply the highest score operation to U^*, as long as this $W_U > 1$. The results of this operation and any other having the same score are added as nodes to a search tree. (The search tree was initialized when $T^* = T$ and $U^* = U$.)

When there are no more operations that can be applied, we continue to build the search tree at a higher node. Finally, the leaves of the search tree are examined to find the highest scoring genome to be V, the last common ancestor of T and U. Pseudocode for this algorithm, **search_trajectory**, follows. Because of the asymmetric way it treats T and U, we repeat it reversing the roles of the two genomes, and choose the better solution of the two.

Algorithm 2. Search_trajectory

Input: two doubling descendants T and U with given orthologies between each pair of paralogs in T and corresponding pair of paralogs in U
Output: doubling descendant V on trajectory between T and U with minimum halving cost
define the pair $T^* = T, U^* = U$ to constitute the root node of a search tree;
while $T^* \neq U^*$ **do**
  for *each possible operation decreasing* $d(T^*, U^*)$ **do**
    calculate y, x and hence W_T and W_U according to the actual and potential changes to the halving distance
  end
  if *the highest scoring operation has* $W_T \geq W_U$ *and* $W_T \geq 6$ **then**
    apply this operation to update T^* and create a new node (T^*, U^*) on the search tree
  end
  if *the highest scoring operation does not satisfy the above conditions on* W_T*, but* $W_U > 1$ **then**
    apply this operation to update U^* and create a new node (T^*, U^*) on the search tree
  end
  if *no operation satisfies the conditions on* W_T *or* W_U **then**
    resume search at the previous node on the search tree
  end
end
Find the optimal $V = T^*$ or $V = U^*$ on the search tree

This search requires $O(m^3)$ time in the worst case for a depth-first traversal of the search tree, where m is the number of markers. This follows since the traversal, which provides an initial estimate of the solution and an initial upper bound on the solution cost, verifies the **while** conditions for $O(m)$ pairs (T^*, U^*) and

examines $O(m^2)$ possible operations each time. The total time to find the locally optimal solution depends on how large a search tree we are willing to maintain.

Using a range of $W \in [1, 18]$ proves clearly better than simply choosing an operation according to whether it $y = 1$ or $y \neq 1$. For example, in simulations generated with $d(T, V) = 60, d(V, U) = 55, d(V, A \oplus A) = 24$, the average estimate $d(V, A \oplus A)$ using an 18-value scale was 29.8, an overestimate of 24%, compared to 31.7 with a two-value scale, an overestimate of 32%.

6.4. *Speciation first*

In Section 6.3, $d(T, V) + d(V, U)$ was fixed and the problem was to find the common ancestor V with the shortest history from the doubling event. We now consider the halving distances of T and U both to be fixed, and look for the particular unduplicated genomes, ancestral to T and U, that are closest together. Our Algorithm **halve_two** simultaneously halves T and U, with a subroutine **add_gray_edges** choosing the initial vertex within each of the supernatural graphs (henceforward SNGs) so as to maximize the number of gray edges in common in the two ancestral genomes being constructed. The SNGs have previously been ordered by a subroutine **sort_SNGs** to favor this maximization.

This search also requires $O(m^3)$ time in the worst case for a depth-first traversal of the search tree, where m is the number of markers, since the number of SNGs to construct is $O(m)$ and both **sort_SNGs** and **add_gray_edges** require $O(m^2)$ to compare SNGs from T with those from U. This traversal provides an initial estimate of the solution and an initial upper bound on the solution cost. The total time to find the locally optimal solution depends on how large a search tree we are willing to maintain.

Algorithm 3. Sort_SNGs

Input: σ_T and σ_U, the set of supernatural graphs for T and U, respectively
Output: ordered sets of SNGs $\sigma_T^{(1)}$ and $\sigma_U^{(1)}$
initialize $\sigma_T^{(1)}$ = the subset of SNGs with 2 black edges and $\sigma_T^{(0)} = \sigma_T \setminus \sigma_T^{(1)}$;
initialize $\sigma_U^{(1)}$ = the subset of SNGs with 2 black edges and $\sigma_U^{(0)} = \sigma_U \setminus \sigma_U^{(1)}$;
// if there are no SNGs with only two black edges, some other SNG is chosen to initialize either $\sigma_T^{(1)}$ or $\sigma_U^{(1)}$
while *there remain SNGs in* $\sigma_T^{(0)}$ *or SNGs in* $\sigma_U^{(0)}$ **do**
  while *there remain SNGs in* $\sigma_T^{(0)}$ *and either* $\sigma_U^{(0)}$ *is empty or the number of black edges in* $\sigma_T^{(1)} \leq$ *the number in* $\sigma_U^{(1)}$ **do**
    // find a SNG in $\sigma_T^{(0)}$ to move to $\sigma_T^{(1)}$
    for *each SNG* s *in* $\sigma_T^{(0)}$ **do**
      // calculate the largest number of gray edges it could have in common with SNGs in $\sigma_U^{(1)}$
      for $i = 1, \ldots, |\sigma_U^{(1)}|$ **do**
        Let k_i be the number of vertices SNG s has in common with u_i, the $i-$th SNG in $\sigma_U^{(1)}$; the number of gray edges they could have in common is $\leq [\frac{k_i}{2}]$
      end
      then the score of s is $\sum_{i,\ldots,|\sigma_U^{(1)}|} [\frac{k_i}{2}]$
    end
    add the highest scoring s to $\sigma_T^{(1)}$
  end
  while *there remain SNGs in* $\sigma_U^{(0)}$, *and either* $\sigma_T^{(0)}$ *is empty or the number of black edges in* $\sigma_U^{(1)} <$ *the number in* $\sigma_T^{(1)}$ **do**
    find a SNG in $\sigma_U^{(0)}$ to move to $\sigma_U^{(1)}$ in the analogous way as for T
  end
end

Algorithm 4. Add_gray_edges

Input: ordered sets of SNGs $\sigma_T^{(1)}$ and $\sigma_U^{(1)}$
Output: search tree where each node is the set of SNGs of T and U with partial assignments of gray edges
At the root of the search tree, add gray edges to all 2-edge SNGs in σ_T and σ_U;
// if there are no SNGs with only two black edges, some other SNG is chosen
while *there remain SNGs in* σ_T *or* σ_U *without gray edges* **do**
  while *there remain SNGs in* σ_T *without gray edges and either all SNGs in* σ_U *have gray edges or the number of gray edges in* $\sigma_T \leq$ *the number of gray edges in* σ_U **do**
    let s be the first SNG in σ_T (using the order in $\sigma_T^{(1)}$) that has no gray edges, where s has v_s vertices **for** $i = 1, \ldots, v_s$ **do**
      starting with the i-th vertex, add gray edges to s according to the deterministic procedure in the halving algorithm, and retain a new node in the search tree if starting with this i-th vertex maximizes the number of gray edges in common with σ_U
    end
    // At this point we can disregard orthology assignments in counting potential gray edges, since these may be reassigned with no other consequence
  end
  while *there remain SNGs in* σ_U *without gray edges and either all SNGs in* σ_T *have gray edges or the number of gray edges in* $\sigma_U <$ *the number of gray edges in* σ_T **do**
    add gray edges to the first SNG in σ_U that has no gray edges using the same procedure as with T, thus creating one or more new nodes in the search tree
  end
end

Algorithm 5. Halve_two

Input: T and U
Output: (optimal) genome halvings A and B, of T and U respectively, which minimize $d(A, B)$
construct_SNG for T and for U;
sort_SNGs;
add_gray_edges;
find closest halvings A and B on search tree

7. SIMULATIONS

For simulations of the doubling first model, we chose the following parameters: five chromosomes, number of markers $m = 200$, inversions to translocations proportion 5:3 or 10:1, random choice of chromosomes to be rearranged, random breakpoints on chromosomes, true orthology relations available for reconstruction. Our algorithm accurately reconstructs the number ν of rearrangements (ten replications for each value of ν) between the doubling event and the speciation event, as long as this is not too large (Fig. 8, left), about $\nu = 50$ for a reversals to translocations ratio of 5:3 and $\nu = 30$ for a ratio of 10:1. With a longer interval between doubling and speciation, the halving algorithm reconstructs the unduplicated ancestor too economically. The shortfall is a function not only of the number ν of rearrangements in the simulation, but also of the proportion of reversals versus translocations ($r : t$) and of the number of markers m. If the number of markers is doubled from 200 to 400, the shortfall in the inferred number of rearrangements is largely corrected, as indicated by the square dots in the figure.

Thus in any inference based on the doubling first model, we must account for the fact that small m, large ν and high inversion proportion may bias the results towards an underestimate of the cost of fitting the model.

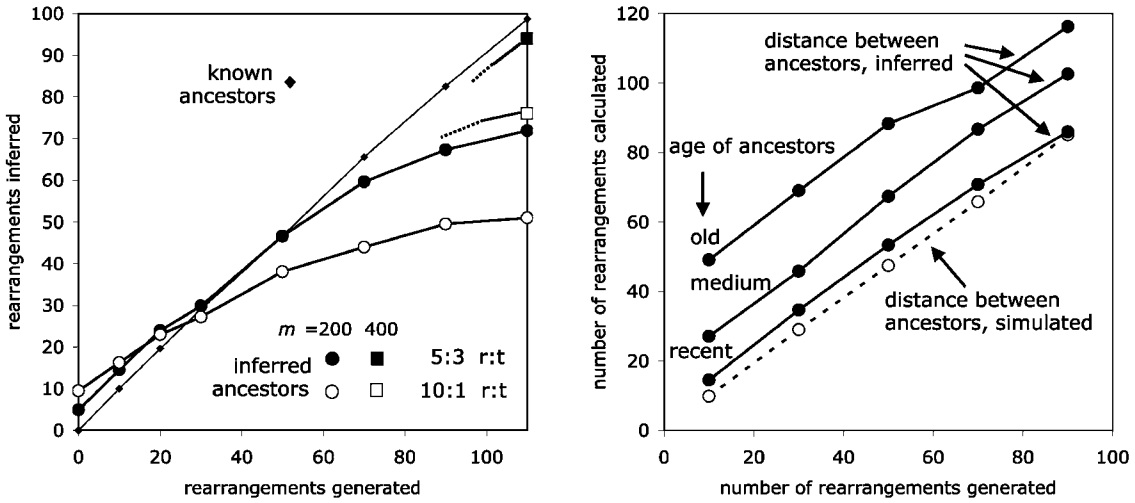

FIG. 8. Estimated distance. **(Left)** Between doubling and speciation (age of ancestor = 50) in doubling first model. **(Right)** Between unduplicated ancestors (ages: old = 80, medium = 50, young = 20) in speciation first model.

Simulations of the speciation first model ($m = 400$) show that while the genome halving distances accurately estimate the number of rearrangements between doubled ancestor and doubling descendant in the simulation (data not shown here), the estimated unduplicated ancestors are further apart than the genomes actually generated in the simulation (Fig. 8, right). This bias increases dramatically as a function, not of the distance itself, but of the amount of rearrangement these ancestors incur to produce the observed doubling descendant. When this "age" is 20, 50, and 80 rearrangements, the bias in the distance between the ancestors increases from 4 to 18 to 37, respectively. This reflects the severely non-unique result of the halving algorithm, which our algorithm attenuates by forcing the reconstructed doubled genomes to resemble each other as much as possible, but cannot eliminate, especially as the age of the doubling events recedes into the past.

In any use of the speciation-first model, we must account for the fact that a large number of rearrangements in the doubled genomes may bias inter-ancestor distance towards an overestimate of the cost of fitting the model.

Nonetheless, the superior accuracy and efficiency of our algorithm in constraining the two simultaneous halving processes to create ancestor genomes as close as possible, in comparison with a search over all pairs in $\mathbf{S_T} \times \mathbf{S_U}$, the Cartesian product of the two complete sets of solutions of the halving algorithm, is clear in another experiment. We set the initial number of markers to be 150, randomly assigned to 8 chromosomes. Then we carried out 45 random rearrangements to create one doubling ancestor and 38 independent rearrangements to create another. After tetraploidization formed two 300-marker genomes, we applied another 42 and 50 rearrangements, respectively, to create the modern doubling descendants. Then, using our knowledge of the ancestral genomes, we found that the distance between the two simulated ancestors was 75 and that the halving distances were 38 and 50, respectively. Running our speciation first algorithms to completion on the two doubling descendants, we reached an inter-ancestor distance $d(A, B) = 84$ (instead of the simulated distance of 75) after three hours of calculation while the search of the Cartesian product only dropped to 87 (from 102) after 24 hours of calculation, involving almost 1,000,000 pairs of optimal ancestors. We stopped the latter calculation when several hours elapsed without any improvement. These results appear as experiment number 1 in Table 1. This table also contains the results from three similar experiments with different values for divergence d between the ancestors and subsequent evolution of the doubled genomes t. While these experiments confirm that there is still a long way to go to eliminate the overestimate bias in inferring d, they indicate that with a tiny fraction of the computing time, **halve_two** achieves substantial reductions in this bias.

TABLE 1. COMPARISON OF PERFORMANCE OF THE **halve_two** ALGORITHMS WITH THE SAMPLING METHOD, VARYING DIVERGENCE d BETWEEN THE ANCESTORS AND SUBSEQUENT REARRANGEMENT DISTANCE OF THE DOUBLED GENOMES t

Experiment number	*Simulated*			*Sample*			***halve_two***	
	d	t	$\hat{d}$	*Size*	$\hat{d}$	*Time*	$\hat{d}$	*Time*
1	$45+38$	$42+50$	75	1.0×10^6	87	24 h	84	3.0 h
2	$20+20$	$60+60$	38	1.47×10^6	70	35 h	57	0.8 h
3	$40+40$	$40+40$	71	0.22×10^6	83	12 h	76	1.5 h
4	$60+60$	$20+20$	100	0.26×10^6	101	17 h	100	0.6 h

8. GENOME DOUBLING IN YEAST

Wolfe and Shields (1997) discovered an ancient genome doubling in the ancestry of *Saccharomyces cerevisiae* in 1997 after this organism became the first to have its genome sequenced (Goffeau et al., 1996). According to Kurtzman and Robnett (2003), the recently sequenced *Candida glabrata* (Dujon et al., 2004) shares this doubled ancestor. We extracted data from YGOB (Yeast Genome Browser) (Byrne and Wolfe, 2005), on the orders and orientation of the exactly 600 genes identified as duplicates in both genomes, i.e., 300 duplicated genes.

Orthology considerations. The algorithm in Section 6.3 requires input about orthology relations among the genes in the two doubling descendants. This question is not pertinent to the algorithm in Section 6.4. We were able to obtain this information, i.e., about which of the two duplicates in one genome is orthologous to which duplicate in the other genome, from YGOB.

We also calculated a separate estimate of the orthology relations, motivated by the parsimony argument in Section 6.2, using the following heuristic:

Algorithm 6. Find_orthologies

Input: two doubling descendants T and U, genome size $2m$, block size parameter k
Output: two resolved orthologies (inferred) between each pair of paralogs in T and corresponding pair of paralogs in U
for *each paralogous marker* **do**
  randomly construct two orthology relations each connecting one of the paralogs in T to one of the paralogs in U
end
calculate $d = d(T, U)$;
construct a list of all $2m$ markers by concatenating all chromosomes of genome T. This list can be grouped into $2m - k + 1$ blocks of k consecutive markers;
for *all blocks of k consecutive markers* **do**
  try changing all the orthology assignments in the block of k genes;
  calculate $d' = d(T, U)$ based on this trial reassignment;
  if $d' < d$ **then**
    retain the changed assignments and set $d' = d$
  end
end
Continue cycling through the list until no further decrease is possible

With the yeast data, increasing k decreased $d(T, U)$, though the improvement after $k = 5$ was minimal. For random orthology assignments, $d(T, U) = 463 \pm 7.45$, while for $kmax = 5$, the algorithm produces $d(T, U) = 327.3 \pm 4.16$. For comparison, the YGOB assignment, in which we have the most confidence,

TABLE 2. DOUBLING FIRST (D.F) AND SPECIATION FIRST (S.F.) ANALYSES EACH PRODUCE A MORE PARSIMONIOUS ANALYSIS (ASTERISKED) OF SIMULATIONS PRODUCED BY THE CORRESPONDING MODEL (D.F. OR S.F., RESPECTIVELY)

	Analysis							
	Doubling first (d.f.)				*Speciation first (s.f.)*			
Data source	$d(T, V)$	$d(V, U)$	$d(V, A \oplus A)$	*Total*	$d(T, A \oplus A)$	$d(A, B)$	$d(U, B \oplus B)$	*Total*
Control								
Sim by d.f.:	102	213	166	481				
Inferred	119	181	157	**457***	214	163	255	**632**
Yeast	92	245	168	**505***	193	182	250	**625**
	122	215	184	**521***				
Control								
Sim by s.f.:					177	164	225	566
Inferred	146	354	104	**604**	164	228	197	**589***

Averages of at least five simulations shown, but the effect holds for each simulation individually. The d.f. analysis gives a far better fit to the yeast data than s.f. Second yeast row reverses the roles of U and T in the algorithm.

gives $d(T, U) = 337$. If we initialize the algorithm with this assignment, it is improved to $d(T, U) = 318$. We use the assignment giving the latter value as our estimate of the most parsimonious assignment.

Competing hypotheses. Though the results of the algorithm in Section 2 suggests that the theory in Kurtzman and Robnett (2003) is the most parsimonious, there is still enough uncertainty in yeast phylogenetics and enough independent occurrences of genome doubling, that it is worth comparing the results of our two methods to dispute or confirm the common doubled ancestor hypothesis. We will compare the analysis of the yeast data, using the method in Section 6.3, assuming the doubling first model depicted on the left of Figure 5, with the analysis in Section 6.4, assuming the speciation first model depicted on the right in Figure 5.

The key point to keep in mind in comparing analyses based on these two different models is that they are both liable to considerable and variable bias, and we must avoid choosing one model simply because its results tend to be biased towards smaller distances while the other is biased towards larger values. The way we do this is first to analyze the yeast data by both of the two methods. Then for each model, we use the inferred rearrangement and halving distances to simulate new data sets with the same number of markers. Finally, we apply both methods to both simulated data sets. Only if each method is superior at inferring the data generated by its own model can we have some confidence that the method that most economically analyzes the yeast data reflects the appropriate model. Otherwise, if one method systematically produced better analyses, even for data generated by the other model, our inferences about the yeast data would be spurious.

Thus, we analyzed the yeast data using the doubling first and speciation first algorithms. The results appear in the centre row of Table 2. (Because of the asymmetry of the doubling first algorithm with respect to T and U, there are two sets of inferences for this case.) We then used the numbers of rearrangements inferred for yeast, using the same number of markers and chromosomes, to simulate the same number of rearrangements in a random model, both with doubling first and speciation first.

We then applied both algorithms, doubling first (**search_trajectory**) and speciation first (**halve_two**), to both sets of data. Note first in Table 2 that the number of rearrangements inferred for the doubling first model using the doubling first algorithm is not exactly the same as that used to generate the data, and likewise for the speciation first case. This is normal, because the inference of rearrangements often is more economical than the rearrangements actually used. The rows in Table 2 show that the doubling first analysis is better (indicated by an asterisk) than the speciation first analysis (457 rearrangements versus 632) when the data are generated by doubling first, whereas the speciation first analysis is better (589 versus 604) when the data is generated with speciation first. The doubling first analysis clearly accounts better for the yeast data (505-521 versus 622), while the simulated controls assure that the biases in the two methods cannot be invoked, so our analysis confirms the hypothesis in Kurtzman and Robnett (2003).

Note that the way we have controlled our analysis of the yeast data by using simulations also eliminates the effects of any general bias of rearrangement methods on the results (though no such bias shows up in comparing the number of events generated in the simulations to the number inferred, across the six sets of simulations in Table 2). The key point is that the speciation first analysis gives better results for the data generated by speciation first, and that the doubling first analysis gives far better results for the data generated by doubling first. This validates our use of the methods as diagnostic of evolutionary history. Thus the fact that the doubling first analysis gives far better results than the speciation first analysis when applied to the yeast data makes it very unlikely that these data were generated by speciation first.

The doubling first analysis of yeast in Table 2 used the YGOB orthology assignment, and for comparability's sake the doubling first analysis of the simulated data used an orthology assignment traced through the actual generation of the data. When we use instead **find_orthologies**, the parsimonious orthology assignment, the totals of 505 and 521 change to 491 and 501, respectively. This reflects the tendencies of parsimony methods to underestimate rearrangement distances (by 3–4% in this case), but does not affect our conclusions.

9. CONCLUSION

We return to the question of a median between an unduplicated genome and two doubling descendants raised in Section 5.4. Though we have focused in this paper on the problems of comparing only the two doubling descendants, both of our algorithms can be directly integrated with existing techniques to take care of the remaining genomes. The doubling first algorithm identifies a doubling descendant whose halving distance is minimal. This genome can be used directly as input to the genome halving with an outgroup algorithm (Zheng et al., 2006) discussed at the beginning of Section 5.3, though with the reservations mentioned below. This solves the version of Case (c) in Figure 3, where the median is a doubling descendant.

The version of Case (c) where the median is an unduplicated genome cannot be so directly addressed using the methods developed here. Nevertheless, the basic principle of the speciation first algorithm, namely the search for common gray edges, can easily accommodate another criterion, namely the prioritization of gray edges representing adjacencies in the third (unduplicated) genome of the median problem. This is the subject of promising ongoing research.

Our previous work on integrating genome halving and other algorithms as a way of incorporating polyploids into rearrangement phylogeny used this software "off the shelf," searching all the many alternate outputs from one as inputs to the other. In the present paper we have avoided an exhaustive search strategy by intervening at the choice points in the genomic distance algorithm in the case of the doubling first problem and in the genome halving algorithm in the case of the speciation first problems. We have shown that these heuristics increase the efficiency of the search and to provide better upper bounds.

The main difficulty in this problem area remains the great multiplicity of solutions to the halving problem. Though this was only encountered here in the speciation first problem, leading to a overestimation of the inter-ancestor distance, it will also have to be dealt with in the doubling first scenario, when the inferred ancestor has to be integrated into a larger phylogenetic tree, via the median problem as mentioned above.

Our focus in this work has been on estimating the chromosomal structure of the doubled and undoubled ancestors of given genomes, in a given phylogeny. Although we are primarily interested in working out the mathematical aspects of these questions on the basis of gene order data, it is clear that information at the level of DNA or amino acid sequence could also be helpful. For example, under the speciation first model, the two copies of a gene within each species should be more similar than the between-gene comparisons, while under the doubling first model, each copy of a gene should have its own unique ortholog in the other species, more similar than its own paralog. We should also note that although we work with phylogenies, these are given and not to be inferred. We are interested here in the small phylogeny problem and not the large problem of actually inferring phylogenies. And though questions of cross-validation of sequence-based and order-based phylogenetics, or of their preferred domains of application, are of great interest, they are outside the scope of the present work, which aims only at inferring gene order, and through that, the occurrence of evolutionary events that affect gene order.

DISCLOSURE STATEMENT

No conflicting financial interests exist.

REFERENCES

Adam, Z., and Sankoff, D. 2008. The ABCs of MGR with DCJ. *Evol. Bioinform.* 4, 69–74.

Bergeron, A., Mixtacki, J., and Stoye, J. 2006. A unifying view of genome rearrangements. *Lect. Notes Comput. Sci.* 4175, 163–173.

Bourque, G., and Pevzner, P. 2002. Genome-scale evolution: reconstructing gene orders in the ancestral species. *Genome Res.* 12, 26–36.

Byrne, K.P., and Wolfe, K.H. 2005. The Yeast Gene Order Browser: combining curated homology and syntenic context reveals gene fate in polyploid species. *Genome Res.* 15, 1456–1461.

Cui, L., Wall, P.K., Leebens-Mack, J.H., et al. 2006. Widespread genome duplications throughout the history of flowering plants. *Genome Res.* 16, 738–749.

Dujon, B., Sherman, D., Fischer, G., et al. 2004. Genome evolution in yeasts. *Nature* 430, 35–44.

El-Mabrouk, N., and Sankoff, D. 1999. Hybridization and genome rearrangement. *Lect. Notes Comput. Sci.* 1645, 78–87.

El-Mabrouk, N., and Sankoff, D. 2003. The reconstruction of doubled genomes. *SIAM J. Comput.* 32, 754–792.

Gallardo, M.H., Bickham, J.W., Honeycutt, R.L., et al. 1999. Discovery of tetraploidy in a mammal. *Nature* 401, 341.

Gallardo, M.H., Kausel, G., Jiminez, A. et al. 2004. Whole-genome duplications in South American desert rodents (Octodontidae). *Biol. J. Linnean Soc.* 82, 443–451.

Goffeau, A., Barrell, B.G., Bussey, H., et al. 1996. Life with 6000 genes. *Science* 275, 1051–1052.

Hughes, A.L. 1999. Phylogenies of developmentally important proteins do not support the hypothesis of two rounds of genome duplication early in vertebrate history. *J. Mol. Evol.* 48, 565–576.

Kurtzman, C.P., and Robnett, C.J. 2003. Phylogenetic relationships among yeasts of the *Saccharomyces complex* determined from multigene sequence analyses. *FEMS Yeast Res.* 3, 417–432.

McLysaght, A., Hokamp, K., and Wolfe, K.H. 2002. Extensive genomic duplication during early chordate evolution. *Nat. Genet.* 31, 200–204.

Ohno, S., Wolf, U., and Atkin, N.B. 1968. Evolution from fish to mammals by gene duplication. *Hereditas* 59, 169–187.

Postlethwait, J.H., Yan, Y.L., Gates, M.A., et al. 1998. Vertebrate genome evolution and the zebrafish gene map. *Nat. Genet.* 18, 345–349.

Sankoff, D., Zheng, C., and Zhu, Q. 2007. Polyploids, genome halving and phylogeny. *Bioinformatics* 23, i433–i439.

Siepel, A., and Moret, B.M.E. 2001. Finding an optimal inversion median: experimental results. *Lect. Notes Comput. Sci.* 2149, 189–203.

Wolfe, K.H., and Shields, D.C. 1997. Molecular evidence for an ancient duplication of the entire yeast genome. *Nature* 387, 708–713.

Xu, R.H., Kim, J., Taira, M., et al. 1997. Differential regulation of neurogenesis by the two Xenopus GATA-1 genes. *Mol. Cell. Biol.* 17, 436–443.

Zheng, C., Zhu, Q., and Sankoff, D. 2006. Genome halving with an outgroup. *Evol. Bioinform.* 2, 319–326.

Address reprint requests to:
Dr. David Sankoff
Department of Mathematics and Statistics
University of Ottawa
Ottawa, K1N 6N5 Canada

E-mail: sankoff@uottawa.ca