A Flexible Ancestral Genome Reconstruction Method based on Gapped Adjacencies

Yves Gagnon*[1], Mathieu Blanchette*[2] and Nadia El-Mabrouk*[1]

[1]Département d'Informatique (DIRO), Université de Montréal, H3C 3J7, Canada
[2]McGill Centre for Bioinformatics, McGill University, H3C 2B4, Canada

Email: Yves Gagnon*- y.gagnon@umontreal.ca; Mathieu Blanchette*- blanchem@mcb.mcgill.ca; Nadia El-Mabrouk*- mabrouk@iro.umontreal.ca;

*Corresponding author

Abstract

Background: The "small phylogeny" problem consists in inferring ancestral genomes associated with each internal node of a phylogenetic tree of a set of extant species. The existing methods can be grouped into two main categories: the distance based methods aiming at minimizing a total branch length, and the synteny-based (or mapping) methods that first predict a collection of relations between ancestral markers in term of "synteny", and then assemble this collection into a set of Contiguous Ancestral Regions (CARs). The predicted CARs are likely to be more reliable as they are more directly deduced from observed conservations in extant species. However the challenge is to end up with a completely assembled genome.

Results: We develop a new synteny-based method that is flexible enough to handle a model of evolution involving whole genome duplication events, in addition to rearrangements and gene insertions and losses. Ancestral relationships between markers are defined in term of *Gapped Adjacencies*, i.e. pairs of markers separated by up to a given number of markers. It improves on a previous, more conservative method, restricted to direct adjacencies, that revealed a high accuracy for adjacency prediction, but with the drawback being of generating a high number of CARs. Applying our algorithm on various simulated data sets reveal good performance as we usually end up with a completely assembled genome, while keeping a low error rate.

Availability: Text for this section of the abstract ...

1 Background

One of the aims of comparative genomics is to reveal the evolutionary scenario that has led to an observed set of present-day genomes from hypothetical common ancestors. When a speciation history, represented as a phylogenetic tree, is already known, then the problem reduces to that of finding ancestral genomes, in terms of content and organization, for non-terminal nodes of the tree. The reconstruction of ancestral karyotypes and gene (or any markers) content and order has been largely considered by the computational biology community [1–8]. For most formulations in terms of different kinds of genomes (circular, multichromosomal, single or multiple gene copies, signed or unsigned genes) and different distance metrics, even the simplest restriction in term of the median of three genomes, has been shown NP-hard [9]. As reviewed in [10, 11], the considered methods can be grouped into two main classes. The distance-based methods aim at labeling ancestral nodes in a way minimizing total branch length over the phylogeny [3, 6–8, 10]. On the other hand, the synteny-based (or mapping) methods [2, 4, 5, 12] rely on three steps: (1) Infer a collection of ancestral genes; (2) Infer a collection of relations between ancestral genes in terms of "synteny"; (3) Assemble this collection into an ancestral genome. In contrast to a distance-based approach, the output of a synteny-based approach is a set of Contiguous Ancestral regions (CARs) that is not guaranteed to be completely assembled into a genome. However, the predicted CARs are likely to be more reliable as they are more directly deduced from observed conservations in extant species. The first formal method based on this approach was developed by Ma *et al.* [5]. In this algorithm, syntenies are adjacencies, sets of ancestral relations are computed by the Fitch parsimony algorithm and a greedy heuristic is used for the assembly. Another class of synteny-based methods [4, 13] define ancestral relations in term of common intervals, represent them in a 0-1 matrix, and then use an approach known as the *Consecutive Ones problem (C1P)* [14] to translate the matrix into sets of ancestral CARs. The translation is direct in case of a collection of ancestral relations being all compatible, but in general the problem of transforming the matrix into a C1P matrix in an "optimal" way is hard, and appropriate simplifications are considered. The result of such methods is not a unique ancestral gene order but rather a PQ-tree representing a collection of possible orders.

Most computational methods for comparative genomics account only for markers with exactly one copy in

every considered extant genome. A few extensions to genomes with unequal gene content have also been considered [2,13,15]. The case of multiple gene copies is more challenging as the one-to-one correspondence between orthologs is missing. Recently, a number of ancestral genome inference studies have accounted for multiple gene copies in the very special case of an evolution by Whole Genome Duplication (WGD). WGD is a spectacular evolutionary event that has the effect of simultaneously doubling all the chromosomes of a genome. Evidence of WGD has shown up across the whole eukaryote spectrum. A distance-based approach for inferring a pre-duplicated genome has been developed in 2003 [16], and extended to the median problem [17–19]. However, the synteny-based approach is more naturally extendable to WGD events. Indeed, as the pre-duplicated genome has single gene copies, as long as an appropriate way for inferring "Double Conserved Synteny" (DCS) relations between ancestral markers is found, the assembly part can be taken without any modification. In [20], Gordon *et al.* used a "manual" approach to reconstruct the ancestral yeast genome. Formal extensions of the synteny-based approach to handle WGD have also been developed [2,10,21].

In this paper, we present a new synteny-based method for ancestral genome inference, allowing for evolutionary scenarios involving WGDs and gene losses, where relations between ancestral genes are defined as *Gapped Adjacencies*, i.e. pairs of genes separated by up to a fixed number of genes. It is an extension of a previous method [2] where relations between genes were defined in term of "direct" adjacencies. The assembling step is based on the computation of a rigorous score for each potential ancestral gapped adjacency (g, h), reflecting the maximum number of times g and h can be adjacent in the whole phylogeny, for any setting of ancestral genomes. To make the link with the "consecutive one" framework [4,13], the syntenies that we consider in this paper can be related to gapped gene teams, while those considered in [4] are related to various types of common intervals [22]. However the assembling methods and the output of the algorithms (a set of CARs versus a PQ-tree) are very different. In the absence of WGD events and gene losses, the approach most comparable to ours is the one developed by *Ma et al.* [5]. In case of direct adjacencies, the algorithm in [2] revealed a higher accuracy for adjacency prediction than *Ma*'s algorithm, but with the counterpart being a higher number of CARs, preventing from recovering a completely assembled genome. In this paper, relaxing the constraint of adjacency to gapped adjacency allows to improve on these results. Indeed, applying our algorithm on simulated data sets reveals that we usually end up with a completely assembled genome, while keeping a low error rate.

2 Methods

2.1 Problem statement and preliminary concepts

Problem statement:

Input: A set Γ of n modern genomes, a species tree S for Γ, and an internal node ν of S representing a speciation event of interest;

Output: An ancestral genome at node ν.

Formally, a species tree (or phylogeny) for Γ is a tree S with n leaves, where each genome of Γ is the label of exactly one leaf, and each internal node (called *speciation node*) has exactly two children and represents a speciation event. We say that S is *labeled* if each internal node u of S has a label $G(u)$ corresponding to a hypothetical ancestral genome just preceding the considered speciation event.
Considering a set Σ of genes, a genome is a set $\{C_1, C_2, \cdots C_N\}$ of chromosomes, where each chromosome is a sequence of signed elements from Σ. Chromosomes can be circular or linear, but we always use a circular representation by adding an artificial gene O at the end of a linear choromosome and considering the augmented chromosomes as circular. Given a genome G, we call the *gene set* of G and denote by $\Sigma_G \subseteq \Sigma$ the set of genes present in G (including O). For example, the gene set of the genome labeling the leftmost leaf of the tree in Figure 1 is $\{O, a, b, c\}$. We further denote by $\pm\Sigma_G$ the set obtained from Σ_G by considering each gene in its positive and negative directions. By convention, the gene O is always considered positive. A *multiset* of $\pm\Sigma_G$ is a subset of $\pm\Sigma_G$ with possibly repeated genes. Given a gene $g \in \Sigma_G$, we denote by $mult(g, G)$ the *multiplicity*, i.e. number of copies, of g in G. In particular, the multiplicity of O is the number of chromosomes of G. For example, the multiplicity of gene a in the genome labeling the leftmost leaf of the tree in Figure 1 is 4. We extend our notation to define, for node u of the tree, Σ_u and $mult(g, u)$ as the set of genes present in the genome at node u and the multiplicity of g in that genome.

2.1.1 Evolutionary model

Our model involves rearrangements and content-modifying operations. As we adopt a synteny-based approach, rearrangements are only implicitly considered, as only traces of these rearrangements in term of disrupted gene adjacencies are considered. In other words, all kinds of rearrangement events can be present in the history. Our approach also allows for unequal gene content, resulting from gene losses or insertions. As for the multiplicity of genes, the only operation leading to multiple gene copies (genes with multiplicity

≥ 2) considered is the *Whole Genome Duplication* (WGD). Formally, a WGD is an event transforming a genome $G = \{C_1, C_2 \cdots C_N\}$ of N chromosomes into a genome G^D containing $2N$ chromosomes, i.e. $G^D = \{C_1, C'_1, C_2, C'_2 \cdots C_N, C'_N\}$, where, for each $1 \leq i \leq N$, $C_i = C'_i$.

In addition to the assumption that WGDs are the only events responsible for gene multiplicity (in particular, single-gene duplications are not considered), we suppose that, in each genome, at least one gene reflects the doubling status of the genome, i.e. there exists a gene that has not lost any copy. As noticed by Zheng *et al.* [19], under these assumptions, the number and position of WGD events can be easily deduced from the multiplicity of the most frequent gene found in each genome. To account for such events, new internal nodes, called *WGD nodes*, are added appropriately on the edges of S (see Figure 1). Contrary to speciation nodes, each WGD node has only a single child. Moreover, if all extant genomes have a gene with multiplicity greater than 1, then a WGD node is inserted above the root of S.

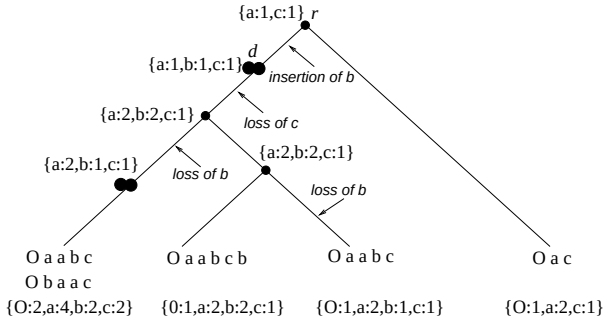


Figure 1: A species tree with each leaf labeled with its corresponding genome. For simplicity, we consider all the genes to be positively signed. The last line below each leaf is the gene set and multiplicity of each gene. Single circles indicate speciation nodes, while double-circles indicate WGD nodes. Applying the procedure described in the text leads to the gene set assignment and multiplicity given as labels of internal nodes. This assignment leads to the indicated insertion and losses.

2.1.2 Adjacencies

Given a genome G, let $g \in \Sigma_G$ and $h \in \pm\Sigma_G$. We say that h is a *1-adjacency*, a *direct adjacency* or simply an *adjacency* of g in G iff it is a left or right adjacency of g where: h is a left-adjacency of g in G iff "$h\ +g$" or "$-g\ -h''$ is a substring of G. Symmetrically h is a right-adjacency of g in G iff "$+g\ h''$ or "$-h\ -g$" is a substring of G.

We now extend 1-adjacencies to *gapped adjacencies*, i.e. to *α-adjacencies* for an arbitrary value of α, by allowing for interleaving genes. Let $G = g_1 g_2 ... g_n$. As already defined, the set of 1-adjacencies of g_i is $\{g_{i-1}, g_{i+1}\}$. We can as well define the set of 2-adjacencies of g_i as $\{-g_{i-1}, g_{i-1}, -g_{i+1}, g_{i+1}\}$, etc. In general, for $\alpha \geq 1$, g_i is α-adjacent to $\{g_{i\pm k} \mid 1 \leq k \leq \lfloor(\alpha+1)/2\rfloor\} \cup \{-g_{i\pm k} \mid 1 \leq k \leq \lfloor\alpha/2\rfloor\}$.

We denote by $LA(g, \alpha, G)$ and $RA(g, \alpha, G)$, or just $LA(g, G)$ and $RA(g, G)$ if $\alpha = 1$, the *multisets* of left

and right α-adjacencies of the one or more copies of g in G. For example, for the genome G labeling the leftmost leaf in the tree of Figure 1, we have $LA(a, 1, G) = \{O, a, b, a\}$ and $RA(a, 1, G) = \{a, b, a, c\}$.

2.1.3 Conserved Adjacencies

For genomes with single gene copies, it is easy to define the number of α-adjacencies preserved along a branch (u, v) of a labeled tree S as the number of substrings of size $\alpha + 1$ between $G(u)$ and $G(v)$ bounded by the same genes. This definition is not directly transposable for genomes with multiple gene copies, as the one to one orthology between genes is not set. Instead, for each gene g, we compare its left and right α-adjacency multisets in $G(u)$ and $G(v)$. More precisely, we define
$adjCons(g, \alpha, G(u), G(v)) = |LA(g, \alpha, G(u)) \cap LA(g, \alpha, G(v))| + |RA(g, \alpha, G(u)) \cap RA(g, \alpha, G(v))|$, as the number of left and right conserved α-adjacencies of g on (u, v), and

$$adjCons(\alpha, G(u), G(v)) = \sum_{g \in \Sigma_u \cap \Sigma_v} adjCons(g, \alpha, G(u), G(v))$$

as the number of conserved α-adjacencies on the branch (u, v). Finally, the number of conserved α-adjacencies over the whole tree S, denoted as $adjCons(\alpha, S)$ (or just $adjCons(S)$ for $\alpha = 1$), is the sum of $adjCons(\alpha, G(u), G(v))$ for all branches (u, v) of S.
Remark: In $adjCons(\alpha, G(u), G(v))$ we account for each adjacency conservation twice. It may appear that right adjacencies alone (or, symmetrically, left adjacencies) are sufficient to reflect adjacency conservation between two genomes. But consider, for example, the sequence "$+1\ -2\ +3\ -4$". If we just consider right 1-adjacencies, then the subsequence "$+1\ -2$" will be considered twice (as -2 is the right adjacency of 1 and -1 is the right adjacency of 2) but the subsequence "$-2\ +3$" will not be considered (as -3 is the left adjacency of 2 and -2 is the left adjacency of 3).

2.2 Ancestral gene content

The first step of any ancestral inference method is to assign ancestral gene content and multiplicity at each ancestral node. We consider a natural procedure, inspired from [20], assuming a model with no convergent evolution and minimum losses. We say that a node v is a *direct descendant* of a WGD node u if and only if v is a WGD node or a leaf and there is no other WGD node on the branch from u to v. To assign gene content Σ_u and gene multiplicity at each internal node u of S, we apply the two following operations in two bottom-up traversals of S: **(1)** For each WGD node u and each gene g, let v be the direct descendant of u with maximum multiplicity for g. If $mult(g, v) \geq 2$ then assign g to u and define $mult(g, u) = \lfloor \frac{mult(g,v)}{2} \rfloor$.

For example after a traversal of the species tree S of Figure 1, the gene set of the WGD node d only contains a and b, as the maximum multiplicity of c in the direct descendants of d is 1; **(2)** Assign a gene g to any internal node u of S on a path from the node of S representing the least common ancestor (LCA) of all the nodes containing g (leaves or WGD nodes), to any leaf containing g. Moreover, if not already defined, define $mult(g, u)$ as the maximum multiplicity of g in u's children.

In the rest of this paper, we will assume that gene content and multiplicity is set for all nodes of S. A *correct labeling*, or simply a *labeling* $G(u)$ of a node u of S will refer to a genome respecting the content and multiplicity constraints given by Σ_u. Notice that, by construction (taking the maximum multiplicity of each gene at each internal node), there is no increase of multiplicity (except in case of an insertion) from a node u to a child v, unless u is a WGD node, in which case the multiplicity of a gene is at most doubled. Such a construction guarantees that any labeling of S can be explained by an evolutionary scenario in agreement with the hypothesis of WGDs being the only events responsible for gene multiplicity.

2.3 A synteny-based method accounting for direct adjacencies

In [2], we have presented a synteny-based method that infers a pre-duplicated ancestral genome at a node ν corresponding to a highest WGD node of S, or any node preceding a first WGD node. More precisely, the method infers a genome $G(\nu)$ such that $adjCons(S|G(\nu))$ is maximized, where $adjCons(S|G(\nu))$ is the maximum number of conserved adjacencies over the whole tree S, for any ancestral genome assignment, with the constraint that genome $G(\nu)$ is assigned at node ν (see details in [2]).

For any node u of S, define $LeftAdj(g, S|_{LA(g,G(u))=X})$ (resp. $RightAdj(g, S|_{RA(g,G(u))=X})$) as the maximum number of left (respec. right) adjacencies that can be preserved over the whole tree, for any ancestral genome assignment with the constraint that the genome $G(u)$ satisfies $LA(g, G(u)) = X$, where X is a multiset of $mult(g, u)$ potential adjacencies selected from $\pm\Sigma_u \setminus \{g\}$. The following upper bound on the objective function allows to treat each gene independently.

$$adjCons(S|G(u)) \leq \sum_g LeftAdj(g, S|_{LA(g,G(u))=X}) + RightAdj(g, S|_{RA(g,G(u))=X})$$

The method, that we call DirectAdj, proceeds in two steps summarized below.

STEP 1: For each internal node u of the tree (speciation or WGD node), each gene $g \in \Sigma_u$, and each multiset X of possible adjacencies of g at node u, we compute $LeftAdj(g, S|_{LA(g,G(u))=X})$ and

$RightAdj(g, S|_{LA(g,G(u))=X})$ using a Dynamic Programming Algorithm. The values at a node u are computed from the values at the two children and also at the parent of u. An illustration is given in Figure 2.

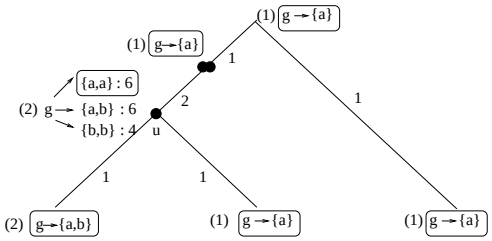


Figure 2: An illustration of STEP 1 for a gene g and an internal node u. Numbers in brackets indicate the multiplicity of gene g at each node of the tree. Multisets at leaves represent (say left) adjacencies of gene g in the corresponding genome. All multisets X of possible adjacencies of g at node u are shown, followed by the value of $LeftAdj(g, S|_{LA(g,G(u))=X})$. The rest of notation illustrates how the value 6 is obtained at u for the multiset $\{a, a\}$: the root and WGD node labels are the adjacencies that have to be set for g, and the label of an edge (v, w) is the number of conserved adjacencies for g on that branch.

STEP 2: For the node ν for which an ancestral genome is sought, we obtain the desired pre-duplicated genome by chaining adjacencies. As ν is a node in a tree with no WGD, or a first WGD node in a history, the multiplicity of genes can be ignored at ν, as in the first case each gene g of Σ_ν is present exactly once at ν, and in the second case all copies of g have the same adjacency. At this node we use the notations $L(g, h) = LeftAdj(g, S|_{LA(g,G(\nu))=\{h\}})$ and $R(g, h) = RightAdj(g, S|_{RA(g,G(\nu))=\{h\}})$. We proceed by a reduction to the Traveling Salesman Problem (TSP) on a complete undirected graph Q where vertices correspond to genes, and an edge (g, h) is weighted according to a ratio $(L(g, h) + R(h, g))/MaxAdj(g, S)$, where $MaxAdj(g, S)$ is the number of nodes of S containing g. The division by $MaxAdj(g, S)$ allows to correct for genes that are lost in some parts of the tree, which avoids favoring genes with high multiplicity. Moreover, as noticed in [2], the result of the TSP is usually one long chromosome concatenating long CARs. To avoid this drawback, we define TSP-τ by augmenting the initial TSP heuristic with the procedure of cutting, from the inferred ancestor, all adjacencies with weight less than a certain threshold τ (see Section 2.4.2). All details on costs, the heuristic used to solve the TSP and how to handle chromosomal endpoints and gene signs, are given in [2]. In the following section, we generalize the approach described above to allow for a more flexible notion of synteny in term of gapped-adjacencies.

2.4 Generalization to gapped adjacencies

Before describing our new algorithm called GapAdj, which is a generalization of DirectAdj accounting for α-adjacencies for increasing values of α, we motivate our new approach in the following section.

Many adjacencies in an ancestral genome are likely to be no longer present in some present-day genomes due to rearrangements and content-modifying operations, preventing from reconstructing large CARs. However, assuming that small and local evolutionary events are more frequent than large and far-reaching operations, which has been largely supported in the literature [23], we can expect to reconnect neighboring CARs by considering gapped adjacencies of increasing gap-size.

Consider for example the species tree (A) of Figure 3. As a and b are neighboring genes in all three genomes, we expect the inferred ancestral genome at the root of the tree to have a CAR with neighboring genes a and b. However, as all (right) direct adjacencies of a are different (b in 1, $-b$ in 2 and x in 3), none of these adjacencies would have a score attaining a reasonable minimum cost τ for the TSP, and a and b will end up in two different CARs with algorithm DirectAdj. However, as b (and also $-b$) is a $2-adjacency$ of a in two extent genomes, and a $3-adjacency$ of a in all three genomes, they will be in the same CAR after the second or third iteration of GapAdj algorithm.

As another example, consider a "true" evolutionary scenario depicted in Figure 3.(B). Consider a threshold τ for TSP-τ corresponding to an adjacency being present in two of the three extent species. Then, as the only direct adjacency present at least twice in extent genomes is bc, the result of DirectAdj is a set of CARs with a and bc being in two separate CARs. However, as b is a 3-adjacency of a in species 1 and 2 (it is actually the only adjacency reaching the threshold up to $\alpha = 3$), GapAdj would end up with a CAR containing the sequence abc after iteration $\alpha = 3$.

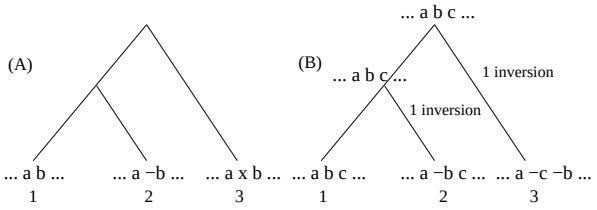


Figure 3: A species tree for the set of species $\Gamma = \{1, 2, 3\}$, with two different genome assignments at leaves. Example (B) depicts a most parsimonious inversion scenario leading to the observed genomes.

2.4.1 Algorithm

The full GapAdj algorithm is given in Supplementary Algorithm 1 (appendix). The output of GapAdj is the set of CARs C representing the ancestral genome at node ν of S. This set is first initialized to the set Σ_ν of genes at ν (each gene being assigned to its own CAR). The algorithm proceeds by iterating the two-step procedure described in Section 2.3 on increasing values of α, from 1 to a constant MAX_α. Step 1 consists in computing α-adjacency scores. The dynamic programming algorithms detailed in [2] for

computing the scores $LeftAdj(g, S|_{LA(g,G(u))=X})$ and $RightAdj(g, S|_{RA(g,G(u))=X})$ of left and right adjacencies of a gene g with a multiset X at a node u of S are directly generalizable to account for α-adjacencies, i.e. to compute the scores $LeftAdj(g, S|_{LA(g,\alpha,G(u))=X})$ and $RightAdj(g, S|_{RA(g,\alpha,G(u))=X})$. As for Step 2, we proceed by constructing a complete undirected graph Q where vertices are the two extremities of each CAR, and edges are weighted according to α-adjacencies scores, computed at Step 1, of the two genes at the extremities of each CAR. A heaviest Hamiltonian cycle through Q, where edges under a threshold τ are excluded, corresponds to an hypothetical ancestral genome characterized by a set of CARs C_α with $|C_\alpha| \leq |C_{\alpha-1}|$. This instance of the TSP is solved using the Chained Lin-Khernigan heuristic implemented in the Concord package [24].

2.4.2 Choice of parameters

An important parameter of our algorithm is the cut-off value τ used to filter out less reliable adjacencies from the solution produced by the TSP algorithm. Based on the simulations that we have performed in [2], we choose a fixed threshold allowing for the best balance between error rate and number of CARs produced. The chosen threshold τ corresponds roughly to keeping an adjacency if and only if it is conserved in at least 70% of the tree branches. Another important parameter of our algorithm is the constant MAX_α, corresponding to the maximum value of α to be considered, which affects both the running time, the final number of CARs and their accuracy. Clearly MAX_α does not need to be more than the size of the longest chromosome of Γ, as no improvement can be achieved for larger values. Unless explicitly indicated, we use $MAX_\alpha = 50$.

3 Results and Discussion

To evaluate the accuracy and running time of our approach, we first used data obtained from simulated genome evolution. This allows us to dissect the impact of each aspect of the method and of the data on the accuracy of the reconstructed ancestor. Our simulations are based on the phylogenetic tree of yeast species shown in Figure 4 (A), which is ideal for this type of study as it contains a phylum affected by a whole-genome duplication and another that remains non-duplicated. Each of the simulation-based results reported in this section are averaged over 50 repetitions.

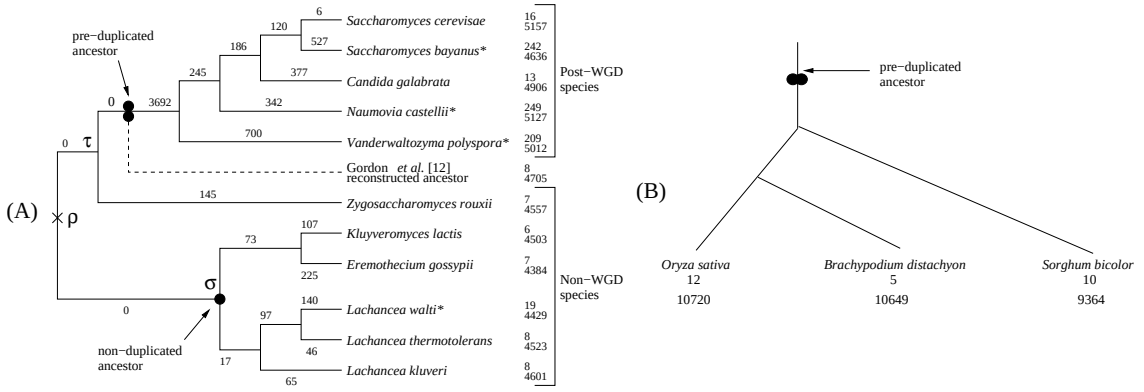


Figure 4: (A) Evolution of the 11 yeast species recorded in the Yeast Gene Order Browser, as given by [25]. The * indicates partially sequenced organisms. At leaves, the top number is the number of chromosomes, contigs or scaffolds. The bottom number is the number of genes, as reported in [20]. On each branch, the label is the number of gene losses, which is directly inferred from the gene content at leaves. The simple circle is the root of the monophyletic group of non-duplicated species, referred in the text by σ. (B) The phylogenetic tree for *Oryza sativa* (rice), *Brachypodium distachyon* (brachypodium) and *Sorghum bicolor* (sorghum). At leaves, the top number is the number of chromosomes. The bottom number is the number of markers used in the study of Section 3.4.

3.1 Simulations with no WGD

In the absence of WGD events, the method that is most comparable to ours is the one of Ma *et al.* [5], implemented in a program called InferCAR. As this method does not support gene losses, we first restrict our simulations to a model with rearrangements only. In addition, as a first validation, we consider single chromosomal genomes, and inversions as the only rearrangement events.

We simulated data sets based on the yeast phylogenetic tree but excluding the portion affected by the WGD. The tree contains six non-duplicated species. The node of interest is the root σ of the monophyletic group of five species (indicated by a simple circle in Figure 4 (A)). A simulated genome of two hundred genes is placed at the root ρ of the tree, and a number r of inversions are randomly performed on each branch of the tree, where r is chosen randomly in the interval $[\frac{rmax}{2}, rmax]$, for a given constant value $rmax$. Notice that the maximum value $rmax = 25$ considered in our simulations leads to some of the leaf genomes being almost completely shuffled, as four or five branches separate them from the root, which lead to the creation of about 160 to 200 breakpoints. The length of inverted segments follows a geometric distribution with $p = 0.5$, resulting in the majority of inversion involving a small number of genes, as previously suggested [23].

Figure 5 (two left diagrams) illustrates the two algorithms' error rates, computed as the fraction of inferred α-adjacencies (for $1 \leq \alpha \leq MAX_\alpha$) that are not present as α-adjacencies in the true simulated ancestor at σ, while the right diagram illustrates the number of CARs obtained (on average) for that ancestor. Both

algorithms show a high accuracy for adjacency prediction, as the error rate is always lower that 10%. Our GapAdj algorithm almost always recovers a complete genome (i.e. a single CAR), which is very rarely the case of InferCAR, which yields an average of 6 CARs for $rmax = 25$. However, this increase in CAR concatenation is obtained at the cost of a small loss of precision.

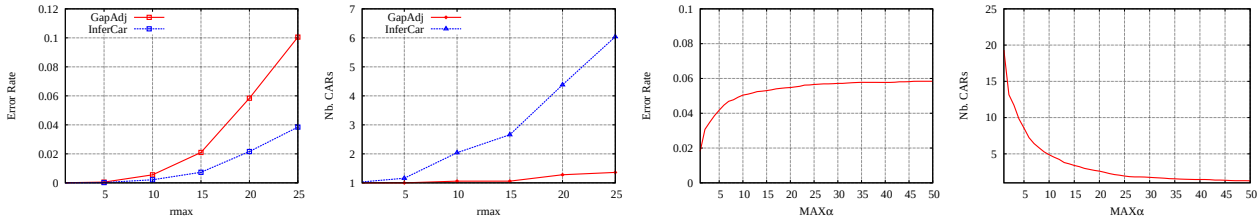


Figure 5: Simulations for a tree without WGD, and a maximum of $rmax$ inversions (x-axis on the two left diagrams) on each branch. Red curves are the results of GapAdj and the blue ones those of InferCAR. From left to right, (1st): Error rate for the inferred ancestral genome; (2nd) Number of inferred CAR; (3d) Error rate and (4th) Number of CARs obtained by *GapAdj*. For these two diagrams $rmax = 20$ and values on the x-axis correspond to the parameter MAX_α.

Figure 5 (two right diagrams) illustrates the progression of the error rate and CAR number for increasing values of α. It provides a comparison with the initial algorithm DirectAdj [2] that only considers direct adjacencies ($\alpha = 1$). From $\alpha = 1$ to $\alpha = 50$, the number of CARs drops from 20 to a single chromosome, while the error rate is increased by less than 4%. These preliminary results are promising as the initial goal of obtaining a completely assembled genome while keeping a low error rate is attained in this case.

We then consider an extended model of evolution for multichromosomal genomes that evolve through inversions, inter-chromosomal rearrangements (translocations, fusions, fissions) and gene losses. Based on the same six-leaf species tree described above, we simulate data sets starting with a 2-chromosome, 200-gene genome at the root ρ of the tree. The number of gene losses on each branch is proportional to that observed in actual yeast genomes, while the proportion of each type of rearrangement operation is chosen to be similar to that reported for *S. cerevisiae* in [20]: (Inv : Trans : Fus+Fiss) = (5 : 4 : 1). The results given in Figure 6 (two leftmost diagrams) reflect the difference in gapped-adjacencies and number of chromosomes between the real and predicted genome at node σ. Notice that chromosomal fusions and fissions may occur on the branch from ρ to σ, so the true number of chromosomes depicted in the second diagram of Figure 6 is not always 2. Interestingly, the curve for inferred CARs roughly follows the curve for true CARs. In addition, the error rate remains lower than 12% in all cases.

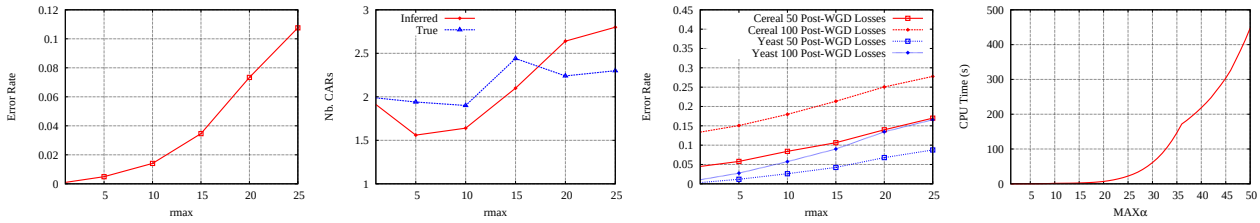


Figure 6: From left to tight, (1st) Error rate and (2nd) Number of CARs obtained by *GapAdj* on simulations following a model accounting for multichromosomal genomes evolving through gene losses, and a maximum of $rmax$ (x-axis) inversions and inter-chromosomal rearrangements per branch of the tree. (3d) Error rate obtained by *GapAdj* on simulations performed according to the cereal tree (Figure 4(B)) and the subtree of yeast rooted at τ (Figure 4(B)). The model accounts for inversions, inter-chromosomal rearrangements, gene losses and one WGD. The two red (resp. blue) curves correspond to the results for cereal (resp. yeast) by performing 50 and 100 losses just following the WGD. (4th) Running time of *GapAdj* for one data set following the "cereal 50" model, and with $rmax$=20.

3.2 Simulations with WGD

For simulations with WGD, we used two trees: one being the subtree of yeast (Figure 4 (A)) rooted at τ, and another (Figure 4 (B)) corresponding to the evolution of three cereals (rice, brachypodium and sorghum), that we will study in Section 3.4. We simulate data sets starting with a pre-duplication 2-chromosome, 200-gene genome at the root of the tree and performing a number of gene losses and a maximum $rmax$ of rearrangements on each branch. As WGD events are usually followed by extensive losses, we perform 50 or 100 random losses between the duplication and first speciation event, followed by 5 random losses on each branch of the tree. As for the rate of various rearrangements, we use the same as before. Error rates are given in Figure 6 (third diagram). The number of CARs produced by the algorithm typically slightly overshoots the correct number, varying from 2 to 4. Note that the losses that occurred immediately after the duplication event result in many false adjacencies inferred, as depicted by the difference in error rate between simulations with only 50 post-duplication losses and those with 100. Since those are ancient events, their effects are seen on many or all of the leaf gene orders, preventing us from inferring the right order in areas surrounding the lost genes in the ancestor. Interestingly, the fact that an outgroup (a genome that is not descendant of the WGD) is available for yeast allows to circumvent this problem as adjacencies can be grasped from this genome not affected by losses, which explains the better results obtained for yeast.

Figure 6 (last diagram) shows the running time of our algorithm for $rmax = 20$, as a function of MAX_α. Although the running time increases cubically with MAX_α, it remains quite manageable. In the absence of the WGD, the running time is significantly smaller, as it remains under 2 seconds even for $MAX_\alpha = 50$ (results not shown).

3.3 Study of yeast genome evolution

We applied our method to the full yeast species tree (Figure 4 (A)) with the gene data sets of the *Yeast Gene Order Browser* [20], to infer the pre-duplicated ancestral genome of *Sccharomyces cerevisiae*. We then compared our predicted ancestor with the 8-chromosome genome manually inferred by Gordon *et al.* [20]. Figure 7 (left) gives the fraction of α-adjacencies that we infer but are in contradiction with the genome inferred by Gordon *et al.* For all tested values of α, this fraction remains below 2%. Importantly, considering gapped adjacencies in addition to direct adjacencies allows to reduce the number of CARs from 23 to 12, which is significantly closer to the number of ancestral chromosomes predicted by Gordon *et al.* Among the 11 additional inferred 1-adjacencies, 7 are shared with the ancestor of Gordon *et al*.

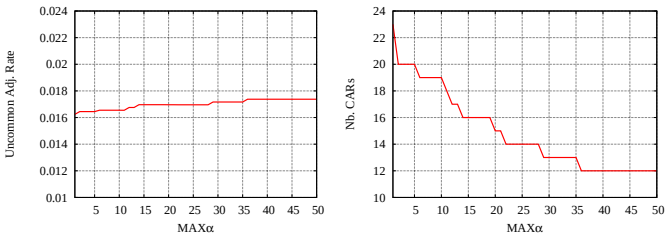


Figure 7: (Left) Fraction of adjacencies in disagreement between the pre-duplicated yeast ancestor inferred by *GapAdj* and that inferred by Gordon et al. in [20]. (Right) Number of CARs inferred with *GapAdj* algorithm.

3.4 Study of cereal genome evolution

We now focus on three of the four completely sequenced cereal crop genomes studied by Murat *et al.* [26], namely rice (*Oryza sativa*), sorghum (*Sorghum bicolor*) and brachypodium (*Brachypodium distachyon*). As demonstrated by various studies, these species have evolved following a whole genome duplication that has occurred about 60 million years ago (see Figure 4.(B)). Maize, the fourth species considered in [26] was excluded here to avoid noise due to an additional maize-specific WGD and ensuing massive gene loss. We used the sets of markers (10,720 from rice, 10,649 from brachypodium, and 9,364 from sorghum) and the homology relationships provided by Murat *et al.*, and the orders for these markers from the annotations in [27–29].

Supplementary Figure 1 shows the predicted pre-duplication genome and its extant descendants. Syntenic regions (homologous sets of genes with conserved order) are painted using the Cinteny web server [30]. Running GapAdj with a maximum value of α (up to the size of the largest chromosome which is about 3500), we end up with a set of 6 CARs (plain bars in Supplementary Figure 1), which is one more chromosome than that inferred by Murat *et al.* [26]. Looking carefully at the obtained results, we can see that the ancestral CARs 5 and 6 are clustered (and shuffled) into a single chromosome in Brachypodium

(chromosome 2), and in two chromosomes in rice and sorghum (chromosomes 1 and 5 in the rice, and 3 and 9 in sorghum). Moreover there is no other segment of the CARs 5 and 6 in any other extant chromosome. This observation suggests that these two CARs should be concatenated into a single and complete chromosome. This would be consistent with the results reported by Murat *et al.* [26], who infer that a single pre-duplicated chromosome C is the ancestor of the same chromosome in Brachypodium (2) and the same two chromosomes in rice (1 and 5) and sorghum (3 and 9). The reason our algorithm did not concatenate them is probably that the genes at both extremities of the ancestral CAR 5 are in two different chromosomes in rice and sorghum. This suggests a future extension of our algorithm that would consider the α-extremities of each current CAR for subsequent concatenations.

Comparing our observations with Murat *et al.*, we notice a number of striking similarities. In particular, one of the main discovery of the paper [26] is that some chromosomes have evolved following a particular evolutionary event, called nested fusion, resulting in the insertion of one chromosome inside another (non-telomeric fusion). Indeed, chromosome 2 of Brachypodium is explained in [26] as resulting from a nested chromosome fusion of the two copies of the chromosome C (introduced in the previous paragraph), that has occurred after the speciation leading to the Brachypodium lineage. Interestingly this nested fusion is clear in our results, as our chromosome painting is in agreement with chromosome 2 of Brachypodium being the result of an insertion of the ancestors of rice chromosome 5 in the middle of the ancestor of rice chromosome 1.

4 Conclusions

Any method for ancestral genome inference is debatable by nature, as it should be based on a model of evolution that is set *a priori*, even though we have no direct access to the history of genomes. Moreover, as real ancestors are not known, any validation method is open to criticism, and there is no direct way of evaluating one solution compared to another.

Based on the first observation, we opted for a synteny-based method that is based as much as possible on the observed data sets, without the need for explicitly defining the rearrangement events acting on these genomes. It is the first synteny-based method that fully capitalizes on the observed adjacencies in present day genomes in relation with their phylogenetic organization. It is flexible enough to apply to genomes that have evolved through whole genome duplication events, in addition to rearrangements and gene insertions and losses.

Based on the second observation, we first opted in [2] for a conservative approach concatenating two

ancestral genes g and h only if the direct adjacency (g, h) is observed in a large fraction of extant genomes and sufficiently supported by the phylogeny. The result was an algorithm with high accuracy for adjacency prediction, but with the counterpart being a high number of CARs. Our generalization to gapped adjacencies while maintaining a conservative strategy for each gap size has led to a reasonable compromise between accuracy in adjacency and karyotype reconstruction.

References

1. Bergeron A, Blanchette M, Chateau A, Chauve C: **Reconstruct. ancestral gene order using conserv. interv.** In *LNCS, vol. 3240*, WABI 2004:14- 25.
2. Bertrand D, Gagnon Y, Blanchette M, El-Mabrouk N: **Reconstruction of Ancestral Genome subject to Whole Genome Duplication, Speciation, Rearrangement and Loss.** In *LNCS, vol. 3240*, *Volume 6293 of* WABI 2010:78–89.
3. Bourque G, Pevzner P: **Genome-Scale Evolution: Reconstructing Gene Orders in the Ancestral Species**. *Genome Research* 2002, **12**:26 – 36.
4. Chauve C, Tannier E: **A methodological framework for the reconstruction of contiguous regions of ancestral genomes and its application to mammalian genomes**. *Plos Computational Biology* 2008, **4**(11).
5. Ma J, Zhang L, Suh B, Raney B, Burhans R, Kent W, Blanchette M, Haussler D, Miller W: **Reconstructing contiguous regions of an ancestral genome**. *Genome Research* 2007, **16**:1557- 1565.
6. Moret B, Wang L, Warnow T, Wyman S: **New approaches for reconstructing phylogenies from gene order data**. *Bioinformatics* 2001, **17**:S165–S173.
7. Sankoff D, Blanchette M: **Multiple genome rearrangement and breakpoint phylogeny**. *Journal of Computational Biology* 1998, **5**:555–570.
8. Zheng C, Sankoff D: **On the Pathgroups approach to rapid small phylogeny.** *BMC Bioinformatics* 2011, **12**:S4.
9. Pe'er I, Shamir R: **The median problems for breakpoints are NP-complete**. In *BMC Bioinformatics*, *Volume 5 of* Electronic colloquium on computational complexity 1998.
10. Chauve C, Gavranovic H, Ouangraoua A, Tannier E: **Yeast ancestral genome reconstruction**. *Plos Comput. Biol.* 2008, **4**(11).
11. El-Mabrouk N, Sankoff D: *Analysis of Gene Order Evolution beyond Single-Copy Genes*, Springer (Humana), *Volume Evolutionary Genomics: statistical and computational methods of* Methods in Mol. Biol. chap. Part II.
12. Muffato M, Louis A, Poisnel C, Crollius HR: **Genomicus: a database and a browser to study gene synteny in modern and ancestral genomes**. *Bioinformatics* 2011, **26**(8):1119- 1121.
13. Gavranovic H, Chauve C, Salse J, Tannier E: **Mapping ancestral genomes with massive gene loss...** *Bioinformatics* 2011, **27**(ISMB 2011):i257- i265.
14. Fulkerson D, Gross O: **Incidence matricesand interval graphs**. *Pac. J. Math.* 1965, **15**:835- 855.
15. Bryant D: **A Lower Bound for the Breakpoint Phylogeny Problem.** In *CPM'00* 2000:235–247.
16. El-Mabrouk N, Sankoff D: **The Reconstruction of Doubled Genomes**. *SIAM Journal on Computing* 2003, **32**:754-792.
17. Gavranović H, Tannier E: **Guided genome halving...** In *SIAM Journal on Computing*, *Volume 15 of* Pacific Symposium on Biocomputing 2010:21-30.
18. Zheng C, Zhu Q, Adam Z, Sankoff D: **Guided genome halving: hardness, heuristics and the history of the Hemiascomycetes**. In *SIAM Journal on Computing*, ISMB 2008:96 - 104.
19. Zheng C, Zhu Q, Sankoff D: **Descendants of Whole Genome Dup. within Gene Order Phylogeny**. *Journal of Computational Biology* 2008, **15**(8):947-964.

20. Gordon J, Byrne K, Wolfe K: **Additions, Losses, and Rearrangements on the Evolutionary Route from a Reconstructed Ancestor to the Modern Saccharomyces cerevisiae Genome**. *PloS Genetics* 2009, **5**(5).

21. Ouangraoua A, Tannier E, Chauve C: **Reconstructing the architecture of the ancestral amniote genome**. *Bioinformatics* 2011, **27**(19):2664- 2671.

22. Bergeron A, Chauve C, Gingras Y: **Formal models of gene clusters**. In *Bioinformatics algorithms: techniques and applications*, Wiley 2008.

23. Kent WJ, ⋯, Haussler D: **Evolution's cauldron: duplication, deletion, and rearrangement in the mouse and human genomes.** *Proc Natl Acad Sci U S A* 2003, **100**(20):11484–11489.

24. Lin S, Kernighan B: **An effective heuristic algorithm for the traveling salesman problem**. *Operations Research* 1973, **21**:498- 516.

25. Hedtke S, Townsend T, Hillis D: **Resolution of phylogenetic conflict in large data sets by increased taxon sampling**. *Systematic Biology* 2006, **55**:522- 529.

26. Murat F, Xu J, Tannier E, Abrouk M, Guilhot N, Pont C, Messing J, Salse J: **Ancestral grass karyotype reconstruction unravels new mechanisms of genome shuffling as a source of plant evolution**. *Genome Research* 2010.

27. *et al* SO: **The TIGR Rice Genome Annotation Resource: improvements and new features**. *Nucleic Acids Research* 2007, **35**:D883- D885.

28. *et al* AP: **The Sorghum bicolor genome and the diversification of grasses**. *Nature* 2009, **457**:551- 556.

29. Initiative IB: **Genome sequencing and analysis of the model grass *Brachypodium distachyon***. *Nature* 2010, **463**:763- 768.

30. Sinha A, Meller J: **Cinteny: flexible analysis and visualization of synteny and genome rearrangements...** *BMC Bioinformatics* 2009, **8**:82.

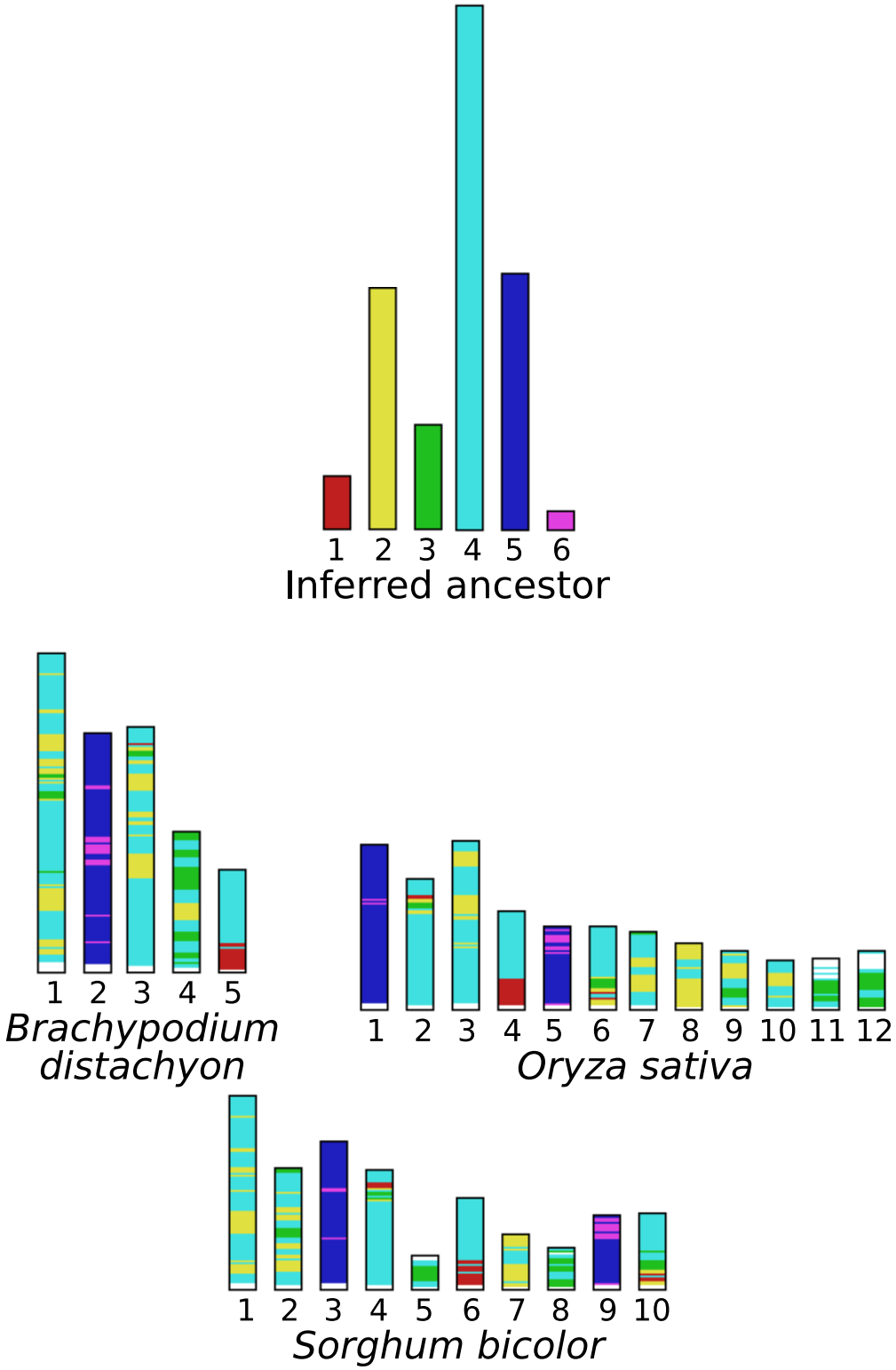
Additional Files

Supplementary algorithm 1

Algorithm Gapped-Adjacencies (GapAdj): (Σ, S, ν, τ, MAX_α)

Initialize the set C of CARs to Σ_ν;
For $\alpha = 1$ to MAX_α **Do**
STEP 1:
    For each internal node u of S (bottom-up traversal) **Do**
        For each $g \in \Sigma_u$ **Do**
            For each multiset X of possible adjacencies
            of g at u **Do**
                Compute $LeftAdj(g, \alpha, S|_{LA(g,\alpha,G(u))=X})$;
                Compute $RightAdj(g, \alpha, S|_{RA(g,\alpha,G(u))=X})$;
            End For
        End For
    End For
STEP 2:
    Construct the graph Q with vertices being the genes of
    Σ, and edges weighted according to computed α-adjacencies;
    By applying TSP-τ on Q, update the set C of CARs;
    Restrict Σ to the α-extremities of each CAR of C;
End For
Return (C);

Supplementary Figure



Supplementary Figure 1. Syntenic regions of three cereal species karyotype with respect to their ancestor inferred using our *GapAdj* algorithm.