algorithms

MDPI

Review

Predicting the Evolution of Syntenies—An Algorithmic Review

Nadia El-Mabrouk

Département d'Informatique et de Recherche Opérationnelle, Université de Montréal, CP 6128 succ Centre-Ville, Montreal, QC H3C 3J7, Canada; mabrouk@iro.umontreal.ca

Abstract: Syntenies are genomic segments of consecutive genes identified by a certain conservation in gene content and order. The notion of conservation may vary from one definition to another, the more constrained requiring identical gene contents and gene orders, while more relaxed definitions just require a certain similarity in gene content, and not necessarily in the same order. Regardless of the way they are identified, the goal is to characterize homologous genomic regions, i.e., regions deriving from a common ancestral region, reflecting a certain gene co-evolution that can enlighten important functional properties. In addition of being able to identify them, it is also necessary to infer the evolutionary history that has led from the ancestral segment to the extant ones. In this field, most algorithmic studies address the problem of inferring rearrangement scenarios explaining the disruption in gene order between segments with the same gene content, some of them extending the evolutionary model to gene insertion and deletion. However, syntenies also evolve through other events modifying their content in genes, such as duplications, losses or horizontal gene transfers, i.e., the movement of genes from one species to another. Although the reconciliation approach between a gene tree and a species tree addresses the problem of inferring such events for single-gene families, little effort has been dedicated to the generalization to segmental events and to syntenies. This paper reviews some of the main algorithmic methods for inferring ancestral syntenies and focus on those integrating both gene orders and gene trees.

Keywords: synteny; gene tree; duplication; horizontal gene transfer; reconciliation

Citation: El-Mabrouk, N. Predicting the Evolution of Syntenies—An Algorithmic Review. *Algorithms* **2021**, *14*, 152. https://doi.org/10.3390/a14050152

Academic Editors: Hélène Touzet and Aïda Ouangraoua

Received: 9 April 2021
Accepted: 8 May 2021
Published: 11 May 2021

1. Introduction

Genes are the basic units of heredity containing the genetic information responsible for the functioning of a cell. During evolution, they are mutated, duplicated, lost and passed to organisms through speciation, the evolutionary process by which a population evolves to become a distinct species, or Horizontal Gene Transfers (HGT), largely shaping the evolution of bacteria, where genes are passed from one species to another. In addition, their order on the genome is modified through various rearrangement events, such as inversions, transpositions or translocations. See Figure 1(1) for an evolutionary history of gene sequences involving a variety of rearrangement, duplication and loss events, and Figure 1(2) for an evolutionary history of a single gene family also involving a HGT event.

Although mutations modifying genomic contents (gene gain and loss) and rearrangements modifying gene orders play a concerted role in shaping gene families, they are usually considered separately: gene gain and loss in the context of inferring the evolution of a given gene family, and rearrangements in the context of understanding genome evolution. In other words, in contrast to rearrangements, gain and loss events are usually considered to be single gene events.

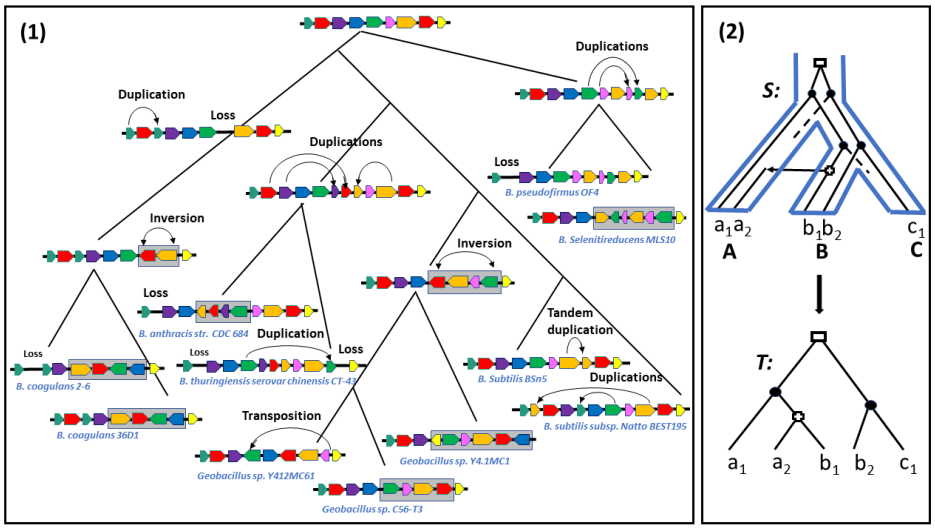

Figure 1. (**1**) An evolutionary history of syntenic regions, inspired by Figure 4 in [1], representing the evolution of tRNA repertoires in the *Bacillus* genus. The tree represents the speciation history of a set of *Bacillus* species. Each colored arrow represents a block of tRNAs, following the operon subdivision available for *B. cereus*. Two arrows of the same color represents a duplicated block. Gray rectangles indicate the segment affected by an inversion. Notice that blocks orientation (indicated by the orientation of the arrow) does not reflect the reality, it is just given to illustrate the effect of an inversion, which not only inverts the order, but also the orientation of the blocks. (**2**) An evolutionary history of a single-gene family (for example, a set of arrows of one given color in the set of bacterial genomes) belonging to the set of genomes $\Sigma = \{A, B, C\}$. The gene family $\Gamma = \{a_1, a_2, b_1, b_2, c_1\}$ is such that a gene x_i belongs to the genome X. The evolution of the gene family inside the species tree S is represented up, and the induced gene tree T is represented bellow. This evolutionary history involves a duplication (represented by a rectangle), losses (dotted lines) and a HGT event (represented by a horizontal line in S and a cross in T).

For a given gene family Γ with gene copies located in a set Σ of genomes, a gene tree T for Γ (representing the evolution of the gene sequences through nucleotide or amino acid mutations) and a species tree S for Σ, the reconciliation approach [2] consists of inferring the evolution of Γ by embedding T into S and explaining the incongruence between the two trees from duplications, losses or HGT events that would have obscured the speciation scenario. Reconciliation is based on the assumption that each gene family evolves independently. Although this hypothesis holds for genes that are far apart in the genome, it is clearly too restrictive for those grouped into syntenies, i.e., forming a set of *homologous* chromosomal regions, meaning that they are deriving from a common ancestral interval, with approximately the same gene content and order. Although convergent evolution should not be excluded, such co-linear sequences of genes are more plausibly the result of a concerted evolution from a common ancestral region, rather than of an independent set of gene duplications that would have generated the same gene organization in different genomic regions.

The neuropeptide Y-family receptors [3], the Homeobox gene clusters [4–6], the FGFR fibroblast growth factor receptors [7,8], the genes of the opioid system [9–11] or the major histocompatibility complex encoding numerous immunologically vital genes playing an imperative role in controlling the vertebrate adaptive immunity [12], are a few examples of genes organized in syntenies in human, as well as in numerous vertebrate genomes. Many of these gene families, appearing in potentially quadruplicated regions in human and other mammalian genomes, have been considered to be evidence of the "2R hypothesis" [13] assessing two rounds of whole genome duplication events in the evolution leading to the contemporary vertebrate genomes. Transposed duplications copying genes or chromosomal segments from an original locus to a new one also play an important role in the evolution of syntenies. Being able to make the difference between the two modes of evolution is also important [14].

Operons in bacteria, containing adjacent genes that are transcribed together into a single mRNA sequence, is another example of genes organized in syntenies [15]. This organization provides a valuable source of information. For example, genes belonging to the same metabolic pathway were found to be organized in similar operons in microorganisms of different phylogenetic lineages, such as Escherichia coli and the Gram-positive Bacillus subtilis [16]. Notice that as horizontal transfers between bacteria of the same or different proteobacterial branches play a major role in shaping bacterial operons, an evolutionary model for studying the origin and evolution of operons cannot avoid considering transfer events.

From an algorithmic point of view, research has focused mainly on the evolution of single-gene families based on sequence divergence and single-gene gain/loss on one side [17], and on the inference of ancestral genomes based on gene content and order of extant genomes on the other side [18]. For the latter branch of research, the considered methods can be grouped into distance-based methods labeling ancestral nodes in a way minimizing total branch length over the phylogeny, and synteny-based (or mapping) methods first inferring a collection of relations between ancestral genes in terms of adjacencies, and then assembling this collection into Contiguous Ancestral regions (CARs) [19]. This latter method can be seen as generating ancestral syntenies (conserved regions) from a set an extant genomes.

What about inferring the evolution of a set of syntenies? In other words, what about the intermediate stage between gene family evolution and genome evolution? In this paper, we review some of the strategies that can be used for this purpose, that combine both information on gene order and gene trees. This review can be seen as a follow-up on a previous review of the evolution of gene families [20], and another presenting the state-of-the-art on algorithmic methods accounting for all different types of evolutionary events (sequence, order and content) [21]. Another relevant review is that of Anselmetti et al. [18] on the reconstruction of ancestral genomes. However, the present review has a specific focus on the evolution of syntenies, rather than on single genes on one extremity, and on whole genomes on the other.

I begin by introducing the concept of syntenies, and the general notations on trees in the next section. In Section 3, I briefly review the sorting by rearrangement problem on two permutations and on a phylogeny, and extend the review to the methods accounting for gene gain and loss in Section 4. The main part of this paper is Section 5 where I review, in more details, algorithms for predicting synteny evolution, accounting for both gene trees and gene orders in a unifying framework. I finally conclude with a discussion on open problems.

2. Syntenies Defined as Gene Orders

The term "synteny", first introduced in 1971 [22], arose from the need to refer to Human genes located on the same chromosome, but with a genetic distance that could not be determined by the frequency of recombination inferred from the new gene mapping methods. As recalled in [23], *synteny* means "same thread" (or ribbon), a state of being together in location, as synchrony means being together in time. Thus, according to the original definition, saying that two genes are syntenic only means that they are located on the same chromosome. Today however, the term is largely used by biologists in an evolutionary meaning to design genes or chromosomal segments with a common evolutionary ancestry, i.e., *homologous* genes, or regions of contiguous genes.

For example, CoGe (https://genomevolution.org/wiki/index.php/Synteny (accessed on 8 April 2021)), a platform for performing comparative genomics research, defines a synteny as a valid deduction that two or more genomic regions derived from a single ancestral region. Inferring "syntenic blocks" usually relies on inferring pairs of chromosomal regions with a similar gene content and order. The SynMap tool of CoGe identifies such blocks by finding sets of homologous gene pairs and merging them into regions.

Such *synteny blocks* or *regions* that are more conserved than average in the genomes can reveal regulatory or functional interactions between the involved genes, or combination of alleles that are advantageous when inherited together. Conversely, breakage of conserva-

tion in gene order or gene content is an important footprint of the evolution of genomes through global rearrangements [24–26] that can be used to infer phylogenetic trees [27].

Two chromosomal regions with identical gene content and order can clearly be labeled as syntenic. However, because syntenic regions are largely remodeled during evolution, it is usually necessary to relax this strict conservation requirement, allowing for a certain gene content or gene order disruption. Notice that genes are usually represented as signed ("+" for the $5' \rightarrow 3'$ strand and "–" for the $3' \rightarrow 5'$ strand) units, where the sign or *orientation of a gene* indicates on which of the two complementary DNA strands the gene is located.

Thus, ranging from a strict definition in terms of conserved segments with identical gene content, order and orientation [25] to the most relaxed one in terms of being located on the same chromosome, the notion of two regions being syntenic has been defined in several ways, also depending on the evolutionary events being considered. In fact, during evolution, syntenic regions evolve independently through local gene rearrangements or local events modifying their gene content, such as tandem duplications adding genes or, conversely, losses removing genes. They also evolve collectively through transpositions and translocations splitting a single synteny into two syntenies, or conversely joining two syntenies into one; new syntenies are created through transposed duplications [28] or whole genome duplication, or conversely lost [29]. They are also passed to organisms through speciation or HGTs (see Figure 1).

From a combinatorial point of view, various formal definitions of synteny blocks, also called *gene clusters* have been introduced to allow identifying them in a set of genomes [20,30] (see Figure 2). Notice first that although we define syntenies as sequences of genes, from a combinatorial or an algorithmic point of view, any other marker or unit can be considered instead of genes. The notion of *common intervals* [31–33] refers to conserved segments in which we relax the conditions that genes appear in the same order or the same orientation. Formally, given K genomes represented as permutations on an alphabet Σ, a *common interval* is a subset $\mathcal{S}$ of Σ such that in each genome, all the genes in $\mathcal{S}$ are contiguous, i.e., grouped together with no other gene in between them, but not necessarily in the same order. In particular, *strong common intervals*, defined as common intervals that do not overlap with any other common interval [34], have rich combinatorial properties [30]. A more relaxed definition of synteny blocks account for possible gaps between genes. A first formal model of max-gap clusters was introduced in [35] under the name of gene teams: Given K genomes, a *gene team* is a maximum subset A of a set of genes Γ such that in each genome, any gene in A is separated by at most δ genes from another gene of A. Common intervals and max-gap clusters completely ignore gene orders. A compromise between gene content and gene order conservation is given in [36,37] where two genes adjacent in one genome are required to be separated by at most δ genes in another genome.

We now introduce some terminology and notations on gene families and trees that we will use in this paper.

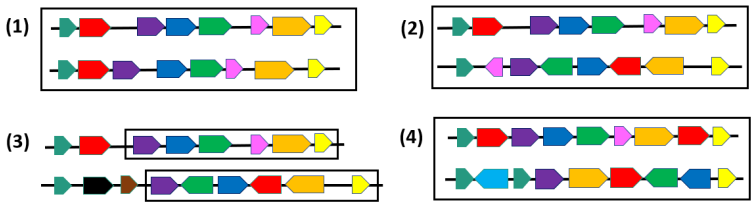

Figure 2. Pairs of regions where genes (or blocks) of the same family are represented by the same color. (**1**) Two identical permutations; (**2**) Two common intervals; (**3**) A δ-team (maximum chain of common genes separated by at most δ foreign genes) with $\delta = 1$; (**4**) Two gene orders on different alphabets and with gene duplicates.

2.1. Gene Families

Two homologous genes or regions X_1 and X_2 are said to be *orthologous* if the last event that has led to the creation of X_1 and X_2 from a common ancestor is a speciation, *paralogous* if it is a duplication and *xenologous* if it is a HGT event. For example, in Figure 1(2), $\{a_1, b_1\}$ are orthologous, $\{a_1, b_2\}$ are paralogous and $\{a_2, b_1\}$ are xenologous. A *gene family* refers to

a set of homologous gene copies (orthologous, paralogous or xenologous) in one or many genomes. Gene families are usually inferred from gene sequence identity. In this paper, the alphabet Γ_X of a chromosomal region X is the set of gene families with loci in X; a sequence X of genes is called a *permutation* of Γ if it contains exactly one copy from each gene family of Γ. Two sequences X and Y are said to have *the same gene content* if they are defined on the same alphabet and have the same number of gene copies for each gene family. If gene orientations are known, then the elements of Γ appear in X accompanied with a sign + or $-$; we talk about *signed syntenies* (e.g., *signed permutations*).

2.2. Trees

If not specified differently, all trees are considered rooted and binary, where a *binary tree* is a tree with all internal (i.e., non-leaf) nodes being binary. We denote by $r(T)$ the root, by $V(T)$ the node set, by $\mathcal{L}(T) \subset V(T)$ the leafset and by $E(T)$ the edge set of T. An edge of $E(T)$ is written as a pair (x, y) of two adjacent nodes, where x, the closest to the root, is called *the parent* of y and y is called *the child* of x. In a binary tree, each internal node has two children. For an internal node x of a tree T, we denote by T_x the subtree of T rooted at x.

The *lowest common ancestor* (LCA) in T of a subset L' of $\mathcal{L}(T)$, denoted by $lca_T(L')$, is the ancestor common to all nodes in L' that is the most distant from the root.

Given a binary tree T, an *extension of* T is a tree T' obtained from T by grafting edges to T, where *grafting* consists of subdividing an edge xy of T, therefore creating a new node z between x and y, then adding a leaf w with parent z.

A tree S is a *species tree* for a set Σ of species if its leafset is in bijection with Σ. A species tree represents an ordered set of speciation events that have led to Σ.

A *gene family* is a set Γ of genes where each gene g belongs to a given species $S = s(g)$ of Σ. A tree T is a *gene tree* for a gene family Γ if its leafset is in bijection with Γ.

3. The Sorting by Rearrangement Problem

In 2003, Pevzner and Tesler [38] developed the notion of synteny blocks as chromosomal segments represented as permutations, that can be converted to identical permutations through micro-rearrangements. The GRIMM-Synteny algorithm [39] constructs synteny blocks from a dot-plot of anchors representing similarities between genes or non-coding regions, and chaining them ignoring micro-rearrangements.

The SORTING BY REARRANGEMENT PROBLEM consists of inferring a rearrangement history of minimum cost, for a given model of evolution, allowing the transformation of a permutation X into another permutation Y. For a unitary cost of operations, we call *Rearrangement Distance* between X and Y the minimum number of allowed operations transforming one synteny into the other.

Given two permutations, the SORTING BY REARRANGEMENT PROBLEM has been shown to be solvable in linear time for the inversion, translocation (including chromosomal fusion and fission), inversions+translocation distances [40–42], as well as for the SCJ (Single-Cut-or-Join) [43] and the DCJ (Double-Cut-and-Join) distance [44], where an SCJ event breaks or creates an adjacency, and a DCJ event breaks two adjacencies and reconnects their extremities in any possible manner. SCJs and DCJs are artificial events unifying most rearrangement events (inversions, transpositions and translocations) in a single model. On the other hand, computing the transposition distance between two permutations has been shown NP-hard [45], although efficient bounded heuristics exist, the best algorithm so far having an approximation factor of 1.375 [46].

The Small Phylogeny Problem

Inferring the evolutionary history of a set of syntenies represented as gene orders has mainly been handled as a SMALL PHYLOGENY PROBLEM [47]. Given a single synteny per genome (i.e., no paralogous syntenies in the same genome are allowed), and given a known phylogenetic tree for the set of considered species, the problem is to infer the ancestral syntenies at the internal nodes of the tree in a way optimizing certain mathematical criteria according to the chosen evolutionary model. Those criteria are usually related to minimizing

the number or cost of evolutionary events leading to the extant syntenies, although maximum likelihood criteria have also been considered.

Given a node-labeled tree S, where labels are syntenies, and given two adjacent nodes u and v in S where u is the parent of v, u is labeled by X_u and v is labeled by X_v, the length of the branch (u, v) of S is the minimum cost of an evolutionary scenario transforming X_u to X_v. Then the general problem can be formulated as Algorithms 1.

Algorithm 1 Small Phylogeny Problem

SMALL PHYLOGENY PROBLEM:
Input: A phylogenetic tree S for a set Σ of species, a set Γ of gene families, a set X of syntenies on Γ labeling the leaves of S and a model of evolution;
Output: A synteny labeling of the internal nodes of S minimizing the total branch length over the phylogeny.

This problem has been most often considered in the context of inferring ancestral genomes, i.e., where syntenies are actually entire chromosomes or genomes. For most formulations in terms of different kinds of genomes (circular, multichromosomal, single or multiple gene copies, signed or unsigned genes) and different cost or distance metrics, even the simplest restriction in terms of the median of three genomes (an unrooted three leaf phylogeny) has been shown NP-hard [48].

Based on the breakpoint graph of two permutations [49], and considering three genomes at a time, the MGR [50] algorithm infers the median by iteratively performing "good" reversals, i.e., reversals diminishing the distance between the three considered genomes. The MGRA [51] algorithm uses a generalization of the breakpoint graph, called multi-colors graph, to more than two permutations, and performs 2-breaks (corresponding to the standard reversals, translocations, fissions, and fusions) "consistent" with the given species tree.

On the other hand, the steinerization method is probably the most popular heuristic for the small phylogeny problem. First assigning an initial synteny labeling to each internal node of the phylogeny, the solution is then refined iteratively by decomposing the phylogeny into a set of overlapping median configurations, updating the median at each step only if it diminishes the sum of the lengths of the branches incident to the median, and iterating until eventually converging to a minimum. The quality of the solutions largely depends on the initialization of the ancestral gene orders. Various initialization strategies have been considered, with the purpose of avoiding local minima. In particular, based on a divide-and-conquer heuristic for finding a median of three permutations minimizing the inversion distance, GASTS [52] uses an accurate initialization step, allowing for an efficient algorithm running several orders of magnitude faster than existing approaches. Another approach, the Pathgroup approach [53], is based on storing partially completed breakpoint graphs on each node of the phylogeny and greedily completing them, following a priority list, in a bottom-up traversal of the species tree. The partial graphs eventually accumulate enough edges in their pathgroups so that cycles can be formed and so that fragments of the ancestral genome can be reconstructed. Other strategies for an initial assignment may consist of a lifted labeling, i.e., taking, for an internal node x, one gene order (synteny) among those labeling the leaves of S_x, or considering all gene orders that are in a certain neighborhood of the extant ones [54].

4. Accounting for Gene Gain and Loss

In the above section, we restricted the review to the papers considering syntenies (or genomes) as permutations on the same alphabet (same set of genes). However, gene loss and gene duplication can also modify the content of synteny blocks.

As for gene losses, they are relatively easy to integrate in the sorting by rearrangement algorithms. More precisely, for the case of syntenies represented as two permutations on two different alphabets (some genes occurring exclusively in one of the two sequences), the inversion+indel problem which consists of computing the minimum number of inversions, insertions and deletions (indels) transforming one synteny into the other, has been shown equivalent to the DCJ+indel distance computation when the breakpoint graph representing the two syntenies has no "bad components" [55,56]. Moreover, linear time

extensions of the DCJ distance computation to the DCJ+indel distance computation have been developed [57,58]. In addition, an extension of the MGRA algorithm, which reconstructs the ancestral genome of multiple genomes using a multi-color breakpoint graph, has been extended to MGRA2 [59] allowing for indels.

However, when duplicates are allowed in syntenies, an extra degree of difficulty is introduced as the one-to-one correspondence between gene copies is not established in advance. In this case, all pairwise rearrangement problems become hard [60]. A review of the methods used for comparing two ordered gene sequences with duplicates can be found in [61]. These methods are grouped into two main classes: those following the *Match-and-Prune model*, aiming at transforming strings into permutations to minimize a rearrangement distance between the resulting permutations [62–65], and those following the *Block Edit model*, introduced in its most general form by Lopresti and Tomkins [66], which consists of covering the two compared syntenies with pairs of blocks to minimize several certain block operations. Such operations can be substitutions, inversions, transpositions, but also duplications. To maintain the symmetry of the resulting distance, a "block uncopy" (symmetrical to a duplication) is also considered.

As reviewed in [61], almost all versions of the Block Edit model are NP-hard. Moreover, even ignoring rearrangements and asking for an optimal sequence of duplications and losses transforming a synteny into another is shown APX-hard even if the number of occurrences of a gene inside a genome is bounded by 2 [67]. Exact exponential-time algorithms based on Integer Linear Programming (ILP) [68,69] and a polynomial-time heuristic based on dynamic programming [70] have been developed for this model, the latter being extended to rearrangements (inversions and transpositions), in addition to duplications and losses. The implemented OrthoALign software [1] has been applied, in a phylogenetic framework, to infer the evolution of transfer RNA repertoires in the Bacillus genus. Recently, an ILP formulation for the DCJ-Indel distance of "natural genomes", i.e., where any marker may occur an arbitrary number of times in any of the two genomes, has been developed [71]. Notice that the problem is slightly easier to handle for balanced syntenies, (i.e., two syntenies containing the same number of occurrences of each gene) though still NP-hard. For computing the DCJ distance of balanced genomes, an integer linear programming (ILP) formulation has been developed [72], as well as a linear time approximation algorithm using the adjacency graph (an alternative representation of the breakpoint graph) [73], with approximation factor O(k) where k is the maximum number of occurrences of any gene in the input genomes.

Finally, more complex evolutionary models have been considered [74,75] unifying the study of various problems on sequence alignment (nucleotide substitutions), rearrangements, duplications and homologous recombinations. These models are tractable only under some strict conditions, such as the hypothesis of no breakpoint re-used in [74], or under strict combinatorial constraints of the "history graph" introduced in [75].

5. Accounting for Gene Trees

The aforementioned methods for inferring the evolution of a set of syntenies only consider syntenies' contents and gene arrangements, while ignoring the evolution of each gene family through nucleotide and amino acid substitutions and indels affecting their sequences. A plethora of methods exist for reconstructing gene trees from sequence divergence. Classical methods use a distance, maximum likelihood or Bayesian approach for inferring the gene tree best representing a sequence alignment (e.g., PhyML [76], RAxML [77], MrBayes [78]), while others use a species tree, in addition to a multiple sequence alignment, to model gene gains and losses inferred from the reconciliation between gene and species trees (e.g., TreeBeST [79], PhylDog [80], ALE [81]). Several gene tree databases from whole genomes are available, including Ensembl Compara [82], PhylomeDB [83] or Panther [84].

In the following, we review the computational approaches that have been considered to integrate gene trees, in addition to gene order and/or gene gain and loss, in a unifying framework for synteny evolution.

5.1. The Reconciliation Approach

Given a gene tree T and a species tree S representing the true bifurcation histories of the considered gene family and set of species, respectively, inferring the scenario of gene gain and loss explaining the difference between the two trees is the purpose of the gene-tree-species-tree-reconciliation approach [2,17]. A *Reconciliation* of T with respect to S is usually defined as an *event-labeled* extension of T, where an internal node label represents the event at the origin of the bifurcation, and grafted branches represent lost (or missing) genes. The considered events are most often speciation, duplications and possibly HGTs. In particular, a most parsimonious reconciliation minimizing the number $D(T,S)$ of Duplications (the D-distance) or the number $DL(T,S)$ of Duplications and Losses (the DL-distance) can be found in linear time using the LCA (Last Common Ancestor) mapping [85–87].

Given a tree T, an extension R of T (R can be equal to T), and a mapping s from $\mathcal{L}(T)$ to $V(S)$ (indicating the genome to which the gene associated with each leaf of T belongs), *an extension of* s is a function $\tilde{s}$ from $V(R)$ to $V(S)$ such that for each leaf x of T, $\tilde{s}(x) = s(x)$. Considering an evolutionary model for a gene family accounting for Duplications (D) and Losses (L) in addition to speciation, the *DL-reconciliation* of T with respect to S is defined as follows.

Definition 1 (DL-reconciliation)**.** *Let Γ be a gene family where each $x \in \Gamma$ belongs to the species $s(x)$ of Σ. Let T be a binary gene tree for Γ and S be a binary species tree for Σ. A* DL-reconciliation *is a triplet $\langle R, \tilde{s}, \tilde{e}\rangle$ where R is an extension of T and $\tilde{s}$ is an extension of s such that for each binary node x of R with two children x_l and x_r, one of the following cases holds:*

1. *$\tilde{s}(x_l)$ and $\tilde{s}(x_r)$ are the two children of $\tilde{s}(x)$ in S in which case $\tilde{e}(x) = Spe$;*
2. *$\tilde{s}(x_l) = \tilde{s}(x_r) = \tilde{s}(x) = \sigma$ in which case $\tilde{e}(x) = Dup$ representing a duplication in σ;*

A grafted leaf on a newly created node x corresponds to a loss in $\tilde{s}(x)$.

As R is as an extension of T, each node in T has a corresponding node in R. In other words, we can consider that $V(T) \subseteq V(R)$. In particular, the species labeling on R induces a species labeling on T.

Given a cost function c on duplications and losses and a reconciliation $\mathcal{R} = \langle R, \tilde{s}, \tilde{e}\rangle$, the cost $c(\mathcal{R})$ is the sum of costs of the induced events.

Given a cost of 0 for speciation and a unitary cost for duplications and losses, the DL-reconciliation $\langle R, \tilde{s}, \tilde{e}\rangle$ of minimum cost $DL(T,S)$ is unique and obtained from $\tilde{s}$ being the LCA-mapping, i.e., verifying, for any internal node x of $V(R) \cap V(T)$, $\tilde{s}(x) = lca_S(s(\mathcal{L}(T[x])))$. We also refer to this reconciliation as the *lca-reconciliation*. See Figure 3 for an illustration.

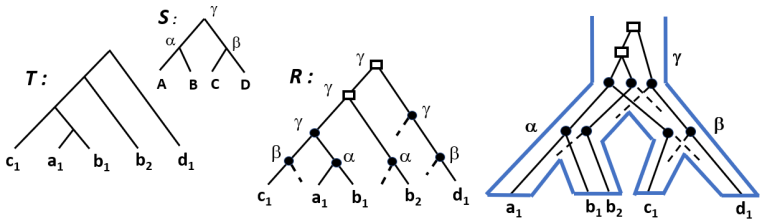

Figure 3. A gene tree T for the gene family $\Gamma = \{a_1, b_1, b_2, c_1, d_1\}$, where a gene x_i belongs to the genome X; a species tree S for the genomes $\Sigma = \{A, B, C, D\}$; a DL-reconciliation $\langle R, \tilde{s}, \tilde{e}\rangle$ of cost $DL(T,S)$, where $\tilde{s}$ is represented by the letter and $\tilde{e}$ by the label (rectangle or plain circle) assigned to each internal node of R. On the right, a representation of R embedded in the species tree S.

5.2. Adjacency Evolution

Rather than considering each gene family independently, a first approach towards integrating dependency information between genes is to account for gene adjacencies. This is the goal of the DeCo algorithm [88] integrating gene tree and gene order information for the purpose of inferring the evolution of syntenies, where syntenies are restricted to adjacencies, i.e., segments of two genes.

More precisely, given a set of gene families, each represented by a reconciled gene tree, i.e., a gene tree node labeled by the event (D or S) at the origin of the bifurcation represented by each internal node, and given a set of adjacencies between genes, the algorithm seeks for an evolutionary scenario of the adjacencies in agreement with the reconciled gene trees, and minimizing adjacency gain and breakage.

Considering an appropriate clustering of adjacencies, the problem is shown to reduce to a set of problems with exactly two gene trees where all adjacencies are between those two trees. Therefore, the input of the DeCo algorithm is a pair of event-labeled gene trees (reconciled gene trees) and a set of adjacencies between the two trees, and the result is a forest of adjacency trees (called *adjacency forest*) of minimum cost (determined by adjacency gain and breakage), where an adjacency tree describes the descent pattern of adjacencies: for adjacency AB to descent from an adjacency CD, gene A should descent from gene C and gene B from gene D. An adjacency tree is event-labeled, where an event-labeling an internal node may be a speciation, an adjacency duplication or a gene duplication. Adjacency breakage and gene loss are represented by grafted edges, and adjacency gains are the roots of new trees (see Figure 4, inspired from Figure 2 in [88], for an illustration).

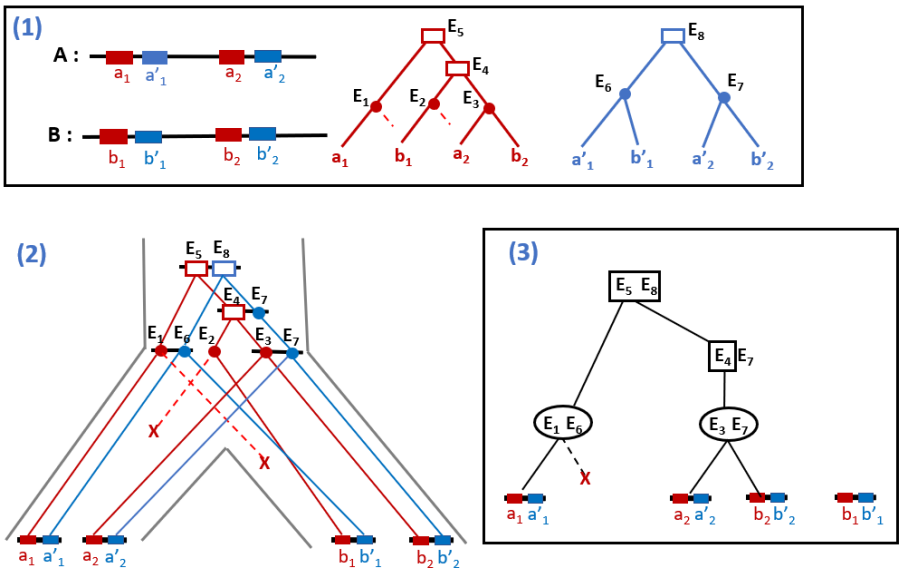

Figure 4. (**1**) Input of DeCo: a set of adjacencies between two gene families, the red and blue ones, in two genomes A and B; reconciled gene trees for the red and blue families, where rectangles are gene duplications, plain circles are speciation and dotted lines are grafted edges for gene losses; (**2**) An evolutionary history embedded in the species tree $S = (A, B)$ explaining the evolution of the adjacencies through gene duplication, speciation and gene loss; (**3**) The output of DeCo in the form of a forest of two trees, representing the evolutionary history of (**2**). The simultaneous duplication of the two adjacent genes corresponding to E_5 and E_8 is an adjacency duplication. The dotted line represents a gene loss. For a unitary cost of adjacency gain and breakage and a 0 cost for other events, the left tree in (**3**) has cost 0, while the right one has a cost of 1, for the gain of adjacency $b_1 b'_1$.

A polynomial-time dynamic programming algorithm is developed, based on a set of recurrences detailing all the cases of adjacency breakage or gain, depending on whether only one or the two genes of an adjacency are duplicated or lost, together or separately.

Applied on an arbitrary set of adjacencies between an arbitrary set of gene trees, the DeCo approach infers a set of ancestral adjacencies for the ancestral species of a species tree. However, as each adjacency is considered independently from the others, inferred ancestral adjacencies are not guaranteed to be compatible with a linear structure. Although this may be seen as a drawback of the method, it can also be used as benchmark for correcting gene trees [89].

As reviewed in [18], following the initial model, several extensions of DeCo have been considered, such as accounting for horizontal gene transfers (see Section 5.6) [90], or handling fragmented extant genome assemblies (ARt-DeCo) [91]. These extensions are

implemented in a unique software called DeCoSTAR [92]. A global score accounting for the gene tree likelihood, the reconciliation cost and the adjacency gain and breakage cost was also developed [93].

5.3. Evolution through Segmental Duplications and Losses

Another strategy for considering the co-evolution of adjacent genes in syntenies is to generalize the reconciliation model to account for segmental duplications and losses rather than single-gene events (see Figure 5).

In [94,95], the DL-reconciliation of a gene tree has been generalized to the DL-reconciliation of a "synteny tree" (defined bellow) accounting for segmental duplications and losses. In this study, a *synteny* X is an ordered sequence of genes belonging to a genome $s(X)$. The genes of a synteny all belong to different gene families and thus, in particular, tandem duplications are not allowed.

We say that a set $\mathcal{G} = \{\Gamma_1, \Gamma_2, \ldots, \Gamma_t\}$ of gene families are *organized into a set* $\mathcal{X}$ *of syntenies* iff there is a bijection between the genes of $\mathcal{G}$ and the genes in $\mathcal{X}$ (each gene of $\mathcal{G}$ belongs to exactly one synteny of $\mathcal{X}$). A *synteny tree* is a tree with leaves mapped to syntenies. In particular, a synteny tree T for $\mathcal{X}$ is a tree with a one-to-one mapping between $\mathcal{L}(T)$ and $\mathcal{X}$.

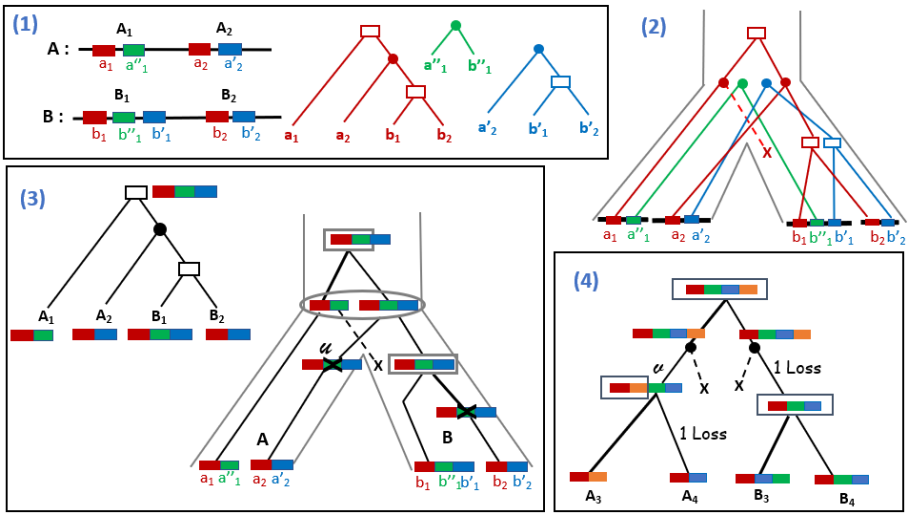

Figure 5. (**1**) Three gene families (red, blue and green) organized into four syntenies $\mathcal{X} = \{A_1, A_2, B_1, B_2\}$ located in two genomes A and B, and the reconciled gene trees for each gene family (node labels are represented as in Figure 4). Gene orders, as well as gene trees are consistent; (**2**) The gene trees embedded in the species tree $S = (A, B)$, illustrating an independent evolutionary history for each gene family; (**3**) A synteny tree for $\mathcal{X}$ (left) and a Super-Reconciliation, i.e., an evolutionary scenario of segmental duplications and losses (right), embedded in the species tree S. Each node of the Super-Reconciliation is labeled by the type of event (duplication represented by a rectangle, speciation by an oval and loss by a cross) and the segment affected by a duplication (inside the rectangle) or a loss (marked by the cross). The bold edge incident to a duplication node is the one leading to the duplicated synteny; (**4**) An unordered Super-Reconciliation for a new set of syntenies $\{A_3, A_4, B_3, B_4\}$ that are not order-consistent. The minimum number of duplications and losses explaining this synteny tree is 5, but rearrangements are still required on some branches as orders are not conserved.

The syntenies of a *synteny family* $\mathcal{X}$ are considered to have evolved from a single ancestral synteny through speciation (defined as for single genes), segmental duplications and segmental losses, where:

- a speciation $Spe(X, [1, l])$ acting on a synteny $X = g_1 \cdots g_l$ belonging to a genome $s(X)$ has the effect of reproducing X in the two genomes s_l and s_r children of $s(X)$ in S;
- a (segmental) duplication $Dup(X, [i, j])$ acting on a synteny X belonging to a genome $s(X)$ is an operation that copies a substring $g_i \cdots g_j$ of $X = g_1 g_2 \cdots g_i \cdots g_j \cdots g_l$

somewhere else into the genome $s(X)$, creating a new *copied synteny* $X' = g'_i \cdots g'_j$, where each g'_k, for $i \leq k \leq j$, belongs to the same gene family as g_k;

- a (segmental) loss $Loss(X, [i, j])$ acting on a $X = g_1 \cdots g_l$ is an operation that removes a substring $g_i \cdots g_j$ of X, leading to the *truncated synteny* $X' = g_1 \cdots g_{i-1} g_{j+1} \cdots g_l$. A loss is called *full* if X' is the empty string (i.e., all genes of X are removed) and *partial* otherwise. A partial loss event is denoted $pLoss$.

Thus, in contrast to a single-gene family, a tree representing the evolution of a set of syntenies is not only labeled by the type of event $\tilde{e}(x)$ corresponding to each internal node, but also by the segment of the synteny affected by the event.

Now, given a set $\mathcal{T} = \{T_1, T_2, \cdots, T_t\}$ of gene trees for the gene families $\mathcal{G} = \{\Gamma_1, \Gamma_2, \cdots, \Gamma_t\}$ organized into a set $\mathcal{X}$ of syntenies belonging to a set Σ of taxa, and given a species tree S for Σ, the goal is to infer a history of segmental duplications and losses that gave rise to the extant syntenies from a unique ancestral synteny. Clearly, the problem can be subdivided into two parts:

1. Given the set $\mathcal{T}$ of gene trees for $\mathcal{G}$, find a synteny tree T for $\mathcal{X}$;
2. Given a species tree S for Σ, find a *Super-reconciliation* $\langle R, \tilde{s}, \tilde{e} \rangle$ of T with S, i.e., an event-labeled synteny tree which is a "partial extension" of T, representing a valid history for $\mathcal{X}$, of minimum *DL-distance*. Here, the DL-distance of $\langle R, \tilde{s}, \tilde{e} \rangle$ is the number of induced segmental duplications and losses.

Notice that due to partial losses, a valid history for $\mathcal{X}$ is not necessarily a binary tree, but rather a *partially binary tree*, i.e., a tree with each internal node having one or two children. In fact, nodes representing partial losses have a unique child (see for example Figure 5(3), left, node u). This is the reason for "partial tree extension" rather than tree extension, where a *partial extension* of T is a tree T' obtained from T by grafting edges or nodes to T, where *grafting a node* simply consists of subdividing an edge xy of T, therefore creating a new node z between x and y.

It is important to notice that, ignoring rearrangements, an evolutionary history of duplications (only creating new gene orders, i.e., not modifying existing ones) and losses does not always exist for an arbitrary set of gene orders, and thus for an arbitrary set of syntenies $\mathcal{X}$, regardless of the trees linking them. If this holds, the syntenies are said to be *order-consistent*. As explained in [95], this can be verified in linear time by representing the gene orders as a directed graph and verifying if it is acyclic. For example, the syntenies $\{A_1, A_2, B_1, B_2\}$ in Figure 5(1)–(3) are order-consistent, while the syntenies $\{A_3, A_4, B_3, B_4\}$ in Figure 5(4) are not order-consistent,

5.3.1. Synteny Tree Reconstruction

The first problem can be handled as a classical phylogenetic reconstruction problem using an alignment of the sequences composing $\mathcal{X}$, each synteny considered to be a single sequence obtained from the concatenation of its gene sequences. However, with this method, the specificity of each gene family is ignored, in addition to being unsuitable in the case of gene rearrangements obscuring the initial alignment. Rather, if a gene tree T_i is available for each gene family Γ_i, then a synteny tree may be obtained from those individual trees. In fact, if the set of trees are "consistent", i.e., do not present contradictory phylogenetic information, then a synteny tree may be represented by a "supertree". The consistency problem of rooted trees has been widely studied. The BUILD algorithm [96] can be used to test, in polynomial time, whether a collection of rooted trees is consistent, and if so, construct a compatible, not necessarily fully resolved, supertree, i.e., a tree displaying them all. This algorithm has been generalized to output all compatible minimally resolved supertrees [97–99], which may be exponential in the number of genes.

If the gene trees are not consistent, a synteny tree may be obtained from a greedy consensus tree method (strict, majority rule or singular majority rule consensus) [100] reconstructing a "consensus tree", i.e., a tree minimizing a given distance with the set of input trees. Alternatively, we may want to minimally correct the input gene trees in a way they become consistent. In the case of gene families likely containing multiple copies in the same synteny, a way of doing can be to keep a single copy in each synteny. This is actually required for the super-reconciliation model considered in [94,95], as tandem

duplications are not allowed in this model. Formally, given a set of gene trees represented as MUL-trees, i.e., trees with potentially repeated leaf-labels, the problem is to prune them in an appropriate way, keeping a single copy of each label.

MUL-trees remain relatively little studied compared with single-labeled phylogenetic trees, mainly due to the fact that many problems that are tractable for phylogenetic trees become NP-hard when extended to MUL-trees. For example, most generalizations of the tree distances [101], as well as generalizations of consensus tree methods are exponential in the case of MUL-trees [102–104]. In a recent publication [105], we consider two problems related to pruning MUL-trees. First, given a set of MUL-trees, the SET PRUNING FOR CONSISTENCY, or MULSETPC, problem asks for a leaf-pruning of each tree leading to a set of consistent trees. Second, proceeding each gene tree at a time, the MUL-TREE PRUNING FOR RECONCILIATION (MULPR) problem asks for a pruning minimizing a reconciliation cost with a given species tree. Both problems are shown NP-hard. Nevertheless, an accurate greedy heuristic for MULPR has been developed.

5.3.2. Super-Reconciliation

Given a set of gene families $\mathcal{G} = \{\Gamma_1, \Gamma_2, \cdots, \Gamma_t\}$ organized into a set of syntenies $\mathcal{X}$, a synteny tree T for $\mathcal{X}$ and a species tree S for the set Σ of species containing the syntenies, the problem is to infer a scenario of segmental duplications and losses explaining T with respect to S. Therefore, at this stage, the super-reconciliation problem can be seen as a generalization of the classical reconciliation problem allowing for segmental events.

In [94,95], this problem is handled by a two-step algorithm: First label the internal nodes of T as duplications or speciation following the LCA-mapping (Figure 5(3), left), and then infer an optimal scenario of losses in agreement with this event-labeled tree $\tilde{T}$ (Figure 5(3), right). This two-steps method has been shown exact, i.e., leading to a super-reconciliation of minimum cost (DL-distance).

The main problem is the second step which consists of extending the tree $\tilde{T}$ with losses and infer the actual event at each node (i.e., the corresponding synteny and segment being duplicated or lost). As losses and segments affected by the events are fully determined by gene orders assigned to internal nodes, the problem actually reduces to the small phylogeny problem, i.e., the problem of assigning syntenies to internal nodes of $\tilde{T}$.

For $x \in V(\tilde{T})$, define $d(x, X)$ as the minimum number of segmental duplications and losses induced by a synteny assignment of $\tilde{T}_x$ with X being the assignment at x. The SMALL PHYLOGENY FOR SYNTENIES problem is to find an optimal assignment, i.e., an assignment leading to $d(\tilde{T}) = \min_X d(r(\tilde{T}), X)$ for X belonging to the set of syntenies that are order-consistent with $\mathcal{X}$.

Let x be an internal node of $\tilde{T}$ and x_l, x_r be its two children. Let X, X_l, X_r be valid assignments for respectively x, x_l and x_r. Then X_l and X_r are subsequences of X. If x is a speciation, then all missing genes in X_l and X_r result from losses. Otherwise, if x is a duplication, then for at most one of X_l and X_r, the missing prefix or suffix can be due to the partial duplication of a segment of X, and all other missing genes should be losses. Therefore, we define two distances: $D^T(X, Y)$ for the minimum number of segmental losses required to transform X to Y and $D^P(X, Y)$ for the minimum number of segmental losses required to transform a substring of X to Y.

Based on the above observations, recurrence relations are defined for a dynamic programming algorithm computing $d(x, X)$, for each $x \in V(\tilde{T})$ and each possible synteny X. The exponential-time complexity of the algorithm is due to the exponential size of the set of syntenies that should be considered at each internal node of $\tilde{T}$.

5.3.3. Unordered Super-Reconciliation

As ignoring rearrangements is usually much too restrictive and asking for consistent gene orders is not very realistic, a variant of the above model is to allow for rearrangements, yet only consider minimizing duplications and losses. Alternatively, this can be seen as a variant of the Super-Reconciliation problem, ignoring gene orders. For example, Figure 5(4) reflects a history for the syntenies $\{A_3.A_4, B_3, B_4\}$ that are not order-consistent, with three duplications and two losses. However, rearrangements are still required on some branches, for example on the branch leading to B_3 or the one leading to v.

Reducing each synteny X to its set $Set(X)$ of genes (i.e., ignoring the order of genes), an *unordered evolutionary history* of a set of syntenies can be represented as a partially binary tree where each internal node x corresponds to an event $e(\text{Set}(X))$ with X being the synteny at x and $e \in \{Spe, Dup, pLoss\}$ such that if e is:

1. Spe, then x is a binary node with two children corresponding to syntenies Y and Z such that $\text{Set}(X) = \text{Set}(Y) = \text{Set}(Z)$ and $s(Y)$ and $s(Z)$ are the two children of $s(X)$ in S.
2. Dup, then x is a binary node with two children corresponding to syntenies Y and Z such that $\text{Set}(Y) = \text{Set}(X)$, $\text{Set}(Z) \subseteq \text{Set}(X)$ and $s(X) = s(Y) = s(Z)$.
3. $pLoss$, then x is a unary node with a child corresponding to a synteny Y such that $\text{Set}(Y) \subsetneq \text{Set}(X)$ and $s(X) = s(Y)$.

An Unordered Super-Reconciliation (USR) is a labeled synteny tree representing a valid unordered evolutionary history for $\mathcal{X}$. The UNORDERED SUPER-RECONCILIATION problem then consists of inferring the USR of minimum cost. As for the ordered version of the problem, the USR Problem reduces to a small phylogeny problem which consists of inferring internal node gene contents of a tree $\tilde{T}$ leading to a minimal duplication and loss cost. As duplications are already determined by the node labeling of $\tilde{T}$, only loss events remain to be minimized, which is done by a programming algorithm running in time $O(|V(T)||\mathcal{G}|)$.

5.4. Minimizing Duplication Episodes

Another strategy to account for multiple gene duplications is to infer duplication scenarios minimizing duplication episodes, i.e., locations in the species tree where a series of duplications may have occurred. This strategy has been used for the purpose of inferring whole genome duplication events, but it can as well be used for inferring a multiple duplication scenario for the gene families of a set of syntenies.

In the literature dedicated to this problem, a *multiple gene duplication* refers to a set of single duplications occurring at the same location of the species tree. The most general formulation is the following: Given a set of gene trees $\mathcal{T} = \{T_1, T_2, \cdots T_t\}$ and a species tree S, find evolutionary scenarios for the collection of gene trees that yields the minimum number of multiple gene duplication events.

Most methods presented in the literature start by labeling internal nodes of the gene trees as duplication or speciation nodes according to the LCA-mapping. Two problems are then considered: (1) According to **an interval model**, assign to each duplication node d of a gene tree $T \in \mathcal{T}$ an interval $Int(d)$ corresponding to the positions in S where the duplication d may have occurred; (2) According to **a clustering model**, cluster the duplications into a set of *minimum episodes* (see Figure 6 for an illustration).

In [106], Paszek and Górecki provides an overview of the interval and clustering models considered in the literature.

5.4.1. The Interval Model

It ranges from the most constrained one restricting $Int(d)$ to the node of S corresponding to $\tilde{s}(d)$ (the LCA-Mapping of d in S as defined above), to the most relaxed one, the *FHS-model* introduced by Fellows et al. [107], where d can be mapped to any node between $\tilde{s}(d)$ and $\tilde{s}(r(T))$. Notice that the *FHS-model* may lead to converting speciation nodes of T to duplication nodes.

Between these two interval models, the *PG-model* by Pazek and Górecki [108] is the most relaxed one leading to the minimum number of duplications for each gene tree T, i.e., the most relaxed interval model that does not lead to converting a speciation into a duplication in an lca-reconciliation of T. Formally, for the PG-model, if d has no ancestor which is a speciation node then $Int(d) = [\tilde{s}(d), \tilde{s}(r(T))]$, otherwise $Int(d) = [\tilde{s}(d), \tilde{s}(p) - 1]$ where p is the ancestor of d closest to the root (possibly the root) such that all nodes between d and p (p excluded) are duplication nodes.

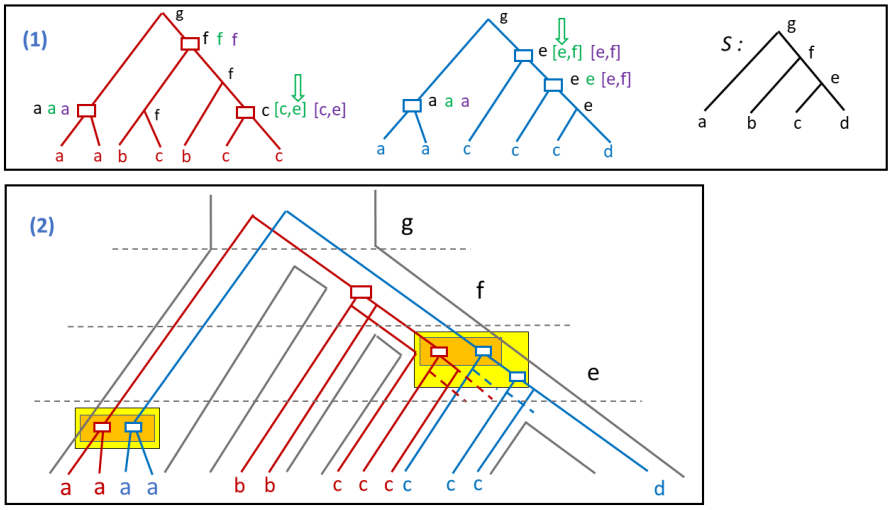

Figure 6. (**1**) Two gene trees (red and blue) and a species tree S. Leaves of the gene trees are labeled according to the genome the corresponding gene belongs to. The event-labeling of internal nodes correspond to the LCA-mapping (rectangles are duplications and the other nodes are speciation). The interval labeling $Int(d)$ is given for each duplication node d (a single letter l corresponding to the interval $[l, l]$): the black label corresponds to the LCA-mapping, the green interval to the GMS-model and the purple interval to the PG-model; (**2**) The two gene trees embedded in the species tree S. The duplication positions are chosen from the GMS-model intervals, and indicated by a green arrow in (**1**). The orange boxes correspond to the multiple duplications according to the ME model, while the yellow boxes correspond to the multiple duplications according to the EC model.

Finally, according to the *GMS-model* proposed by Guigó et al. [109], let p be the node preceding d (or d if d is the root), then $Int(d) = [\tilde{s}(d), \tilde{s}(p)]$, if $\tilde{s}(d) = \tilde{s}(p)$, otherwise $Int(d) = [\tilde{s}(d), \tilde{s}(p) - 1]$.

Except the FHS-model, all these interval models are examples of the general interval models presented in [110].

5.4.2. The Clustering Model

Given an interval assignment of each duplication node of the set of gene trees, three different duplication clustering models have been proposed in the literature. The *Episode Clustering (EC) model* allows clustering any two duplications that can be mapped to the same location in the species tree (yellow boxes in Figure 6), while a slightly more constrained model, called the *Minimum Episodes (ME) model*, excludes cases in which a duplication and an ancestor of this node (from the same gene tree) are clustered together (orange boxes in Figure 6).

The two problems were introduced by Guigó et al. [109] with the interval model being the GMS-model. These problems can be formulated as NP-hard set cover problems [111]. Alternatively, representing them as a Tree Interval Cover (TIC) Problem, polynomial-time algorithms can be designed for the EC and ME models under the GMS interval model [112,113]. Moreover, linear time and space algorithms for the TIC Problem that applies to the EC and ME models have been developed by Luo et al. [114] under every interval model. Finally, Paszek and Górecki [106] proposed a variant of the algorithm for general interval models presented in [110] that runs in linear time for the ME Problem. Solutions to the EC and ME problems for unrooted gene trees and for the PG-model were also studied [108,115].

Finally, in addition to the EC and ME clustering models, the *Gene Duplication Clustering (GD) model* [107] is similar to EC except that only duplications from different gene trees can be clustered in a single episode.

Notice that the EC Problem for the FHS-model has a trivial outcome with one cluster. On the other hand, the GD and ME problems for the FHS interval model have been shown NP-hard [107,116]. The unconstrained ME model has also been extended to gene losses for different costs of duplications and losses [116].

5.5. Evolution of Tandemly Arrayed Gene Clusters

An important class of syntenic regions is constituted by Tandemly Arrayed Gene clusters (TAGs). Slippage during recombination, a mechanism at the origin of tandem repeats which is favored by the presence of repetitive sequences, induces a chain reaction eventually leading to the creation of large TAGs, i.e., groups of paralogous genes that are adjacent on a chromosome. TAGs account for about one third of the duplicated genes in eukaryotes [117]. In human, they represent about 15% of all genes [118] forming a number of complex gene clusters. Those repeated regions are however extremely difficult to study or even to assemble correctly due to the fact that during evolution, the duplication status of segments is obscured by subsequent deletions, breaks and rearrangements. When the step of determining the linear gene composition of a TAG cluster is completed, inferring an evolutionary scenario for the tandemly repeated genes is further complicated by the fact that the phylogenetic signal is often obscured by gene conversion.

Methods based on a preprocessing of a self-alignment dot-plot of a cluster or the dot-plot of a pairwise-alignment of two clusters have been developed for reconstructing a hypothetical ancestral sequence and a duplication scenario leading to an observed gene cluster [119–121]. Although these methods are useful to infer recent evolutionary events, they are less appropriate for longer timescales as alignment of the nonfunctional regions becomes impossible due to mutations continuously affecting each duplicated segment.

Assuming correct gene orders and gene trees have been obtained, the problem of inferring the evolutionary scenario of a set of TAG clusters can be handled using the *tandem-duplication model of evolution* first introduced by Fitch in 1977 [122]. This model assumes that from a single ancestral gene at a given position in the chromosome, the locus grows through a series of consecutive duplications placing the newly created copy next to the original one. Such tandem duplications may be *simple* (duplication of a single-gene) or *multiple* (simultaneous duplication of neighboring genes). Based on this idea, several theoretical studies have considered the problem of reconstructing the tandem-duplication history of a single TAG cluster through tandem duplications only (which is not always possible) [123]. The model has been extended to the study of a set of orthologous TAG clusters, with an evolutionary model accounting for losses and rearrangements, in addition to simple tandem duplications [124,125]. Later, Lajoie et al. [126] developed the DILTAG algorithm inferring all most parsimonious evolutionary histories for a single gene cluster, according to a general cost model involving simple and multiple tandem duplications, deletions and inversions.

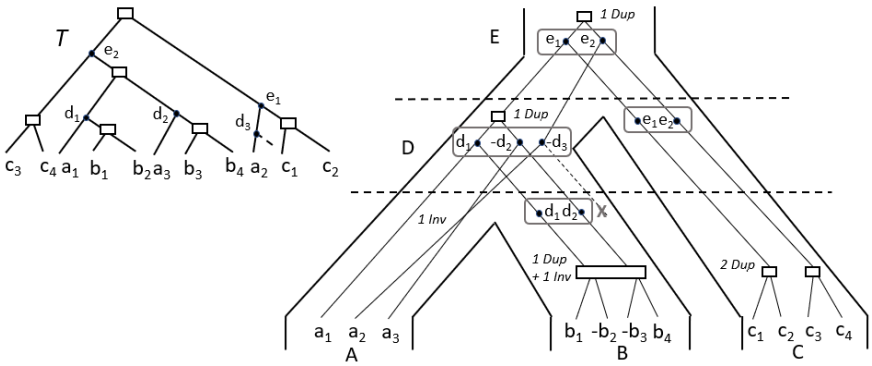

Figure 7. The gene tree T is event-labeled according to the lca-mapping with the species tree $S = ((A, B), C)$. A leaf x_i of T denotes a gene copy in genome X, its index i corresponding to its position in the TAG cluster of genome X, as illustrated on the tips of the species tree S on the right.

DILTAG proceeds by exploring a "history graph" (search space), where vertices correspond to *ordered gene trees*, i.e., gene trees with ordered leaves (gene orders), and edges to evolutionary events. The size of the whole search-space being exponential, a greedy heuristic was developed that only conserves, in a queue, the most promising partial evolutionary histories obtained after exploring a given depth of the history graph.

In Tremblay-Savard et al. [127], DILTAG was then used for inferring the evolution of a set of orthologous TAGs (see Figure 7 for an example). The developed MULTIDILTAG algorithm proceeds in two steps. First, ignoring gene orders, an lca-reconciliation of the gene tree T with the species tree S is computed, leading to a set $\Gamma(x)$ of ancestral genes at each internal node x of S. Then, reinserting gene order and sign information on the leaves of S, the order and sign of genes at the internal nodes of S are inferred by traversing S bottom-up and applying DILTAG on each branch (x, y) of S, with the exception that instead of reaching a single gene, the algorithm stops when it reaches the expected number of gene copies. All orders leading to the optimal solution are then conserved in a set $\mathcal{S}_x$.

The set $\mathcal{S}_A$ of a leaf A is just the TAG cluster corresponding to A.

Now, denote by $\Gamma(x)$ the set of speciation vertices of the lca-reconciliation of T mapping to x. For example, in Figure 7, $\Gamma(D) = \{d_1, d_2, d_3\}$. Let $s \in \{l, r\}$, denoting the left or the right child of a node. The pre-speciation genome set $PG(x_s)$ is the subset of $\Gamma(x)$ with a child in the branch (x, x_s), i.e., the genes in $\Gamma(x)$ that have not been lost after speciation on the branch going to x_s. For example in Figure 7, $PG(D_l) = \{d_1, d_2, d_3\}$ while $PG(D_r) = \{d_1, d_2\}$.

The set $\mathcal{S}_x$ at each internal node x of S is computed by MULTIDILTAG as follows: (1) For each of $s \in \{l, r\}$, DILTAG is executed on each element of $\mathcal{S}_{x_s}$, and stops as soon as the attained gene order contains $|PG(x_s)|$ genes. The set of all ancestral gene orders obtained (output of DILTAG) form an initial *pre-speciation* set $\mathcal{PS}_{x_s}$ (further refined by removing the elements that do not lead to a minimum cost). For example, in Figure 7, $\mathcal{PS}_{D_l} = \{(d_1, -d_2, -d_3)\}$ and $\mathcal{PS}_{D_r} = \{(d_1, d_2)\}$; (2) For each of $s \in \{l, r\}$, construct the set $\mathcal{PS}'_{x_s}$ by reinserting, in each gene order, and in all possible orders, the genes lost on the branch (x, x_s). Then, $\mathcal{S}_x = \mathcal{PS}'_{x_l} \cup \mathcal{PS}'_{x_r}$. For example, in Figure 7, $\mathcal{S}_D = \{(d_1, -d_2, -d_3), (d_1, d_2, d_3), (d_1, d_2, -d_3), (d_1, d_3, d_2), (d_1, -d_3, d_2), (d_3, d_1, d_2,), (-d_3, d_1, d_2)\}$.

During the execution of the algorithm, a solution graph is incremented by adding the appropriate "speciation edges" from the elements of $\mathcal{S}_x$ to those of $\mathcal{PS}_{x_l}$ and $\mathcal{PS}_{x_r}$ giving rise to the minimum distance.

Although showing good performance in inferring the total number and size distribution of duplication events on simulated datasets, a limitation of the MULTIDILTAG heuristic is however in dealing with multiple gene deletions, as the algorithm is highly exponential in this case, and becomes quickly intractable.

5.6. Accounting for Horizontal Gene Transfers

Horizontal gene transfer, largely involved in shaping bacterial gene content, has been included later in the reconciliation analysis of gene families in the purpose of inferring scenarios of duplications, losses and transfers and, identifying xenologous gene copies, in addition to orthologs and paralogs. In this context, the *DTL distance* is the minimum number of Duplications, Transfers and Losses explaining a gene tree T given a species tree S. The following review of DTL-reconciliation is largely inspired from [17].

For a transfer to happen from a source genome $\tilde{s}(x)$ to a target genome $\tilde{s}(y)$ (x and y being two nodes of the gene tree T), both genomes should have coexisted. Therefore, a *time-consistent* HGT scenario should allow ordering the internal nodes of the species tree S. This problem of finding a most parsimonious time-consistent (or acyclic) DTL-scenario, is NP-hard [128–131]. However, the DTL-distance problem becomes polynomial if consistency requirement is relaxed [132,133]. The main idea is to consider all possible mapping of T nodes to S nodes, using a dynamic programming approach.

More precisely, let $c(x, s)$ be the minimum cost of a reconciliation of T_x with S such that x is mapped to $s \in V(S)$. The gene tree T is processed bottom-up, with the base-case corresponding to leaves $x \in \mathcal{L}(T)$ treated as follows:

$$For\ \ x \in \mathcal{L}(T),\ c(x,s) = \begin{cases} 0, & \text{if } s = s(x), \\ +\infty, & \text{otherwise.} \end{cases}$$

As for an internal node x with children y and z, we must consider the three possibilities of x being labeled as a speciation, duplication or HGT node, with $c_s(x,s)$, $c_d(x,s)$ and $c_t(x,s)$ representing, respectively, those three mutually exclusive cases. Then, $c(x,s) = \min\{c_s(x,s), c_d(x,s), c_t(x,s)\}$. Finally, the minimum cost of a reconciliation of T with S is $\min_{s \in V(S)} c(r(T), s)$.

Ignoring losses and considering the cost of a reconciliation as being the number of duplications and HGT, the following recurrences hold [133].

$$c_s(x,s) = \begin{cases} \min\{c(y,t) + c(z,u) \text{ for all } t,u \text{ incomparable and s.t.} \\ \qquad lca_S(t,u) = s\}, \quad \text{if } s \text{ is an internal node of } S, \\ +\infty, \qquad\qquad\qquad\qquad\qquad\qquad \text{otherwise.} \end{cases}$$

$$c_d(x,s) = \min\{1 + c(y,t) + c(z,u) \text{ for all descendants } t,u \text{ of } s \text{ in } S\}$$

$$c_t(x,s) = \min\{1 + c(y,t) + c(z,u) \text{ for all } t \text{ being descendant of } s \\ \text{in } S \text{ and all } u \text{ being incomparable to } s\}$$

A straightforward implementation of these recurrences leads to an algorithm in $O(mn^2)$ time, where $m = |V(T)|$ and $n = |V(S)|$. This time complexity has been further improved to $O(mn)$ [134].

The above recurrences may be adapted to handle losses. David and Alm [135] described AnGST, an algorithm for the DTL distance running in $O(mn^2)$, while Bansal et al. [132] described RANGER-DTL, an algorithm for the DTL distance running in $O(mn)$.

When divergence time information, or a temporal ordering of internal nodes, is available for S, then the DTL-scenario must respect this ordering (i.e., HGT events are constrained to occur only between co-existing species). A DTL-scenario respecting a dated tree is called a *date-respecting DTL-scenario*. Bansal et al. [132] show how the definition of a reconciliation and the above recurrences can be adapted to solve this problem. They give an algorithm with $O(mn \log n)$ time complexity. Notice that a date-respecting DTL-scenario is not necessarily time-consistent. In fact, scenarios may be locally consistent (i.e., HGT events occurring between co-existing species), but globally inconsistent. Global consistency may be obtained by subdividing the species tree S into slices and exploring slices one after the other. This strategy has been first used in [136], leading to an $O(nm^4)$ algorithm. Later, Doyon et al. [128] have improved the computation of a most parsimonious DTL-reconciliation for dated trees to $O(mn^2)$.

Recently, we extended the reconciliation approach to a special case of horizontal gene transfers, namely *Endosymbiotic Gene Transfers (EGT)*, where genes are transferred solely between the mitochondrial and nuclear genome of the same species [137]. Such transfers from the mitochondrion to the nucleus have been a driven force in the evolution of eukaryotes since the unique ancestral endosymbiotic event integrating an α-proteobacterium into a host eukaryotic cell. The *DLE-distance* (DLE for Duplication, Loss and EGT) is easier to compute than the DTE-distance, as there is no need for exploring a set of source genomes, nor there is a risk of time inconsistency. The linear-time algorithm developed in [137] for an arbitrary cost of operations can be seen as an adaptation of the quadratic-time DTL algorithm for dated trees, which allows transfer between any co-existing species [128].

Extending the DTL-reconciliation to super-reconciliation, i.e., allowing for segmental gene transfers, raises many questions that must be deeply explored. Given a synteny tree $\tilde{T}$ and a species tree S, the super-reconciliation method presented in Section 5.3 runs in two steps: (1) Infer the event-labeling of the internal nodes of $\tilde{T}$, this labeling being the one minimizing duplication nodes; (2) Infer an optimal scenario of losses in agreement with this event-labeling. This two-step methodology has been shown exact in the case of the DL-reconciliation, i.e., it has been shown that this method leads to the DL-distance. However, this is not true anymore for the DTL distance. In fact, a reconciliation minimizing duplications, transfers and losses is not necessarily a reconciliation minimizing duplications and transfers. Indeed, considering a node as a transfer rather than as a speciation node

may decrease the number of required losses. We are presently investigating this problem of generalizing super-reconciliation to handle transfers, in addition to duplications and losses.

6. Conclusions

Despite the large effort dedicated to the development of methods for inferring the evolution of synteny blocks, a lot remains to be done towards a unifying model allowing consideration of the variety of evolutionary events shaping the genomes. As reviewed in this paper, most of the algorithmic effort has been invested in the genome rearrangement field on one side, and in the reconciliation field for inferring individual gene gain and loss on the other side. Combining order and content information in the purpose of inferring a co-evolutionary history of genes remains a largely under-explored field.

In most studies on gene gain and loss, segmental movements are indirectly inferred from single-gene movements, by combining co-occurring events or concatenating individual adjacencies. For example, although allowing for a wide variety of events (rearrangements, tandem duplications, losses, HGTs, etc.), the DeCo collection of algorithms stand on minimizing single adjacencies' gain or breakage with no direct link to segmental events. The same holds for algorithms dedicated to minimizing episodes of events, only indirectly referring to multiple duplications and ignoring order information. As for the reviewed algorithms for tandemly arrayed gene clusters, although accounting for both rearrangements and segmental duplications and losses, duplications are limited to tandem duplications, and they do not allow for the study of paralogous syntenies, and do not account for the gain or loss of syntenies. Conversely, the Super-Reconciliation approach only accounts for transposed duplications, ignoring tandem duplications.

This latter approach, which is a direct generalization of the reconciliation approach to segmental events, opens the door to the same algorithmic perspectives as the classical reconciliation approach, such as generalization to HGT events. However, its main limitation is the difficulty of generating an accurate synteny tree, though a solution may actually be to choose a tree optimizing such a Super-Reconciliation cost. In any case, a better use of individual gene trees, including the inconsistency among those trees, should be considered, ways of including tandem duplications should be explored, as well as better solutions to integrate rearrangement inference algorithms.

To summarize, accounting at the same time for sequence divergence (gene trees), gene order and gene content, evolving through punctual mutations (substitutions and indels), rearrangements and segmental gain and loss events, remain a challenge. Beyond the difficulty of designing an appropriate algorithmic method outpassing the limitations of existing methods, the problem is to design appropriate scores accounting for the variety of evolutionary events. Designing such a general score has been undertaken in [93], but a lot remains to do in this field.

Of course, all the algorithmic methods presented in this paper suffer from the standard limitations of parsimony methods [138,139], such as impossibility of accounting for multiple state changes along a branch of a phylogeny, or uncertainty in phylogenetic reconstructions. An overall statistical framework for evaluating evolutionary hypothesis [140] is surely better than a heuristic outputting a single solution with no measure of reliability. However, the largest the list of considered evolutionary events, the more difficult is the problem of an appropriate sampling for such a statistical study. Being able to generate the whole set of most parsimonious scenarios remains an interesting approach for a probabilistic evaluation of the solutions.

Another related problem lies on the possibility of performing the appropriate simulations for testing the accuracy of the developed algorithms. This goal is also complicated with the increase of the considered data types and evolutionary events. Moreover, some events remain largely unexplored for this purpose, such as the Endosymbiosis gene transfer, a special case of HGTs where genes are exchanged only between the mitochondrial and nuclear genomes of the same species. This event is known to have played a major role in the evolution of eukaryotes [141,142]. Although prior work provides useful insights to understand the parameters influencing such an event [143,144], designing an appropriate model for the simulation of EGT evolutionary histories that can be used to assess the accuracy of our algorithm in [137] remains to be done.

Funding: This research was funded by Natural Sciences and Engineering Research Council of Canada and Fonds de recherche Nature et technologies, Quebec.

Data Availability Statement: Not applicable.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Tremblay-Savard, O.; Benzaid, B.; Lang, B.F.; El-Mabrouk, N. Evolution of tRNA Repertoires in Bacillus Inferred with OrthoAlign. *Mol. Biol. Evol.* **2015**, *32*, 1643–1656. [CrossRef]
2. Goodman, M.; Czelusniak, J.; Moore, G.; Romero-Herrera, A.; Matsuda, G. Fitting the gene lineage into its species lineage, a parsimony strategy illustrated by cladograms constructed from globin sequences. *Syst. Zool.* **1979**, *28*, 132–163. [CrossRef]
3. Larsson, T.; Olsson, F.; Sundstrom, G.; Lundin, L.; Brenner, S.; Venkatesh, B.; Larhammar, D. Early vertebrate chromosome duplications and the evolution of the neuropeptide Y receptor gene regions. *BMC Evol. Biol.* **2008**, *8*, 1–22. [CrossRef] [PubMed]
4. Abbasi, A.; Grzeschik, K. An insight into the phylogenetic history of HOX linked gene families in vertebrates. *BMC Evol. Biol.* **2007**, *7*, 1–15. [CrossRef] [PubMed]
5. Ferrier, D. Evolution of Homeobox Gene Clusters in Animals:The Giga-Cluster and Primary vs. Secondary Clustering. *Front. Ecol. Evol.* **2016**, *4*, 36. [CrossRef]
6. Garcia-Fernàndez, J. The genesis and evolution of Homeobox gene clusters. *Nat. Rev. Genet.* **2005**, *6*, 881–892. [CrossRef]
7. Ajmal, W.; Khan, H.; Abbasi, A. Phylogenetic investigation of human FGFR-bearing paralogons favors piecemeal duplication theory of vertebrate genome evolution. *Mol. Phylogenet. Evol.* **2014**, *81*, 49–60. [CrossRef]
8. Hafeez, M.; Shabbir, M.; Altaf, F.; Abbasi, A. Phylogenomic analysis reveals ancient segmental duplications in the human genome. *Mol. Phylogenet. Evol.* **2016**, *94*, 95–100. [CrossRef]
9. Dreborg, S.; Sundstrom, G.; Larsson, T.; Larhammar, D. Evolution of vertebrate opioid receptors. *Proc. Natl. Acad. Sci. USA* **2008**, *105*, 15487–15492. [CrossRef]
10. Stevens, C. The evolution of vertebrate opioid receptors. *Front. Biosci. J. Virtual Libr.* **2009**, *14*, 1247–1269. [CrossRef] [PubMed]
11. Sundstrom, G.; Dreborg, S.; Larhammar, D. Concomitant Duplications of Opioid Peptide and Receptor Genes before the Origin of Jawed Vertebrates. *PLoS ONE* **2010**, *5*. [CrossRef]
12. Naz, R.; Tahir, S.; Abbasi, A. An insight into the evolutionary history of human MHC paralogon. *Mol. Phylogenet. Evol.* **2017**, *110*, 1–6. [CrossRef] [PubMed]
13. Hughes, A. Phylogenies of developmentally important proteins do not support the hypothesis of two rounds of genome duplication early in vertebrate history. *J. Mol. Evol.* **1999**, *48*, 565–576. [CrossRef]
14. Wang, Y.; Li, J.; Paterson, A. MCScanX-transposed: Detecting transposed gene duplications based on multiple colinearity scans. *Bioinformatics* **2013**, *29*, 1458–1460. [CrossRef] [PubMed]
15. Moreno-Hagelsieb, G.; Trevino, V.; Pérez-Rueda, E.; Collado-Vides, T.S.J. Transcription unit conservation in the three domains of life: A perspective from Escherichia coli. *Trends Genet.* **2001**, *17*, 175–177. [CrossRef]
16. Fani, R.; Brilli, M.; Liò, P. The Origin and Evolution of Operons: The Piecewise Building of the Proteobacterial Histidine Operon. *J. Mol. Evol.* **2005**, *60*, 378–390. [CrossRef] [PubMed]
17. El-Mabrouk, N.; Noutahi, E. *Bioinformatics and Phylogenetics, Seminal contributions of Bernard Moret*; Computational Biology, Chapter Gene Family Evolution: An Algorithmic Framework; Springer: Berlin/Heidelberg, Germany, 2019.
18. Anselmetti, Y.; Luhmann, N.; Bérard, S.; E, E.T.; Chauve, C. Comparative Genomics. In *Methods in Molecular Biology*; Chapter Comparative Methods for Reconstructing Ancient Genome Organization; Humana Press: New York, NY, USA, 2017; Volume 1704.
19. Gagnon, Y.; Blanchette, M.; El-Mabrouk, N. A flexible ancestral genome reconstruction method based on gapped adjacencies. *BMC Bioinform.* **2012**, *13*, 1–12.
20. El-Mabrouk, N.; Sankoff, D. Methods in Molecular Biology. In *Evolutionary Genomics: Statistical and Computational Methods*; Chapter Analysis of Gene Order Evolution Beyond Single-Copy Genes; Springer (Humana): Totowa, NJ, USA, 2012; Volume 855, pp. 397–429.
21. Chauve, C.; El-Mabrouk, N.; Gueguen, L.; Semeria, M.; Tannier, E. *Models and Algorithms for Genome Evolution*; Computational Biology, Chapter Duplication, Rearrangement and Reconciliation: A Follow-Up 13 Years Later; Springer: London, UK, 2013.
22. Renwick, J. *Human Genetics*; Grouchy, J., Ebling, F., Henderson, I., Eds.; Excerpta Medica: Amsterdam, The Netherlands, 1972; pp. 443–444.
23. Passarge, E.; Horsthemke, B.; Farber, R. Incorrect use of the term synteny. *Nat. Genet.* **1999**, *23*, 387. [CrossRef] [PubMed]
24. Eisen, J.; Heidelberg, J.; White, O.; Salzberg, S. Evidence for symmetric chromosomal inversions around the replication origin in bacteria. *Genome Biol.* **2000**, *1*, 1101–1109. [CrossRef] [PubMed]
25. Nadeau, J.; Taylor, B. Lengths of chromosomal segments conserved since divegence of man and mouse. *Proc. Natl. Acad. Sci. USA* **1984**, *81*, 814–818. [CrossRef]
26. Simison, W.B.; Parham, J.; Papenfuss, T.; Lam, A.; Henderson, J. An Annotated Chromosome-Level Reference Genome of the Red-Eared Slider Turtle. *Genome Biol. Evol.* **2020**, *12*, 456–462. [CrossRef] [PubMed]
27. Drillon, G.; Champeimont, R.; Oteri, F.; Fischer, G.; Carbone, A. Phylogenetic Reconstruction Based on Synteny Block and Gene Adjacencies. *Mol. Biol. Evol.* **2020**, *37*, 2747–2762. [CrossRef] [PubMed]
28. Johnson, M.; Cheng, Z.; Morrison, V.; Scherer, S.; Ventura, M.; Gibbs, R.; Green, E.; Eichler, E. Recurrent duplication-driven transpositionof DNA during hominoid evolution. *Proc. Natl. Acad. Sci. USA* **2006**, *103*, 17626–17631. [CrossRef]

29. Mangal, M.; Srivastava, A.; Sharma, R.; Kalia, P. Conservation and Dispersion of Genes Conferring Resistance to Tomato Begomoviruses between Tomato and Pepper Genomes. *Front. Plant Sci.* **2017**, *8*, 1803. [CrossRef] [PubMed]
30. Bergeron, A.; Chauve, C.; Gingras, Y. Formal models of gene clusters. In *Bioinformatics Algorithms: Techniques and Applications*; Mandoiu, I., Zelikovsky, A., Eds.; Wiley: Hoboken, NJ, USA, 2008; Chapter 8.
31. Bergeron, A.; Stoye, J. On the similarity of sets of permutations and its applications to genome comparison. *J. Comput. Biol.* **2003**, *13*, 1340–1354. [CrossRef]
32. Heber, S.; Stoye, J. Finding all common intervals of *k* permutations. Lecture Notes in Computer Science. In *Combinatorial Pattern Matching*; Amir, A., Landau, G.M., Eds.; Springer: Berlin/Heidelberg, Germany, 2001; Volume 2089, pp. 207–218.
33. Uno, T.; Yagiura, M. Fast algorithms to enumerate all common intervals of two permutations. *Algorithmica* **2000**, *26*, 290–309. [CrossRef]
34. Landau, G.; Parida, L.; Weimann, O. Gene proximity analysis across whole genomes via PQ trees. *J. Comput. Biol.* **2005**, *12*, 1289–1306. [CrossRef]
35. Bergeron, A.; Corteel, S.; Raffinot, M. The algorithmic of gene teams. In *Workshop on Algorithms in Bioinformatics*; Lecture Notes in Computer Science; Guigó, R., Gusfield, D., Eds.; Springer: Berlin/Heidelberg, Germany, 2002; Volume 2452, pp. 464–476.
36. Yang, Z.; Sankoff, D. Natural Parameter Values for Generalized Gene Adjacency. *J. Comput. Biol.* **2010**, *17*, 1113–1128. [CrossRef]
37. Zhu, Q.; Adam, Z.; Choi, V.; Sankoff, D. Generalized Gene Adjacencies, Graph Bandwidth, and Clusters in Yeast Evolution. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2009**, *6*, 213–220.
38. Pevzner, P.; Tesler, G. Genome rearrangements in mammalian evolution: Lessons from human and mouse genomic sequences. *Genome Res.* **2003**, *13*, 13–26. [CrossRef]
39. Pham, S.; Pevzner, P. DRIMM-Synteny: Decomposing genomes into evolutionary conserved segments. *Bioinformatics* **2010**, *26*, 2509–2516. [CrossRef]
40. Bader, D.; Moret, B.; Yan, M. A Linear-Time Algorithm for Computing Inversion Distance between Signed Permutations with an Experimental Study. *J. Comput. Biol.* **2001**, *8*, 483–491. [CrossRef] [PubMed]
41. Bergeron, A.; Mixtacki, J.; Stoye, J. Reversal Distance without Hurdles and Fortresses. In *Combinatorial Pattern Matching*; Lecture Notes in Computer Science; Sahinalp, S., Muthukrishnan, S., Dogrusoz, U., Eds.; Springer: Berlin/Heidelberg, Germnay, 2004; Volume 3109, pp. 388–399.
42. Tesler, G. Efficient algorithms for multichromosomal genome rearrangements. *J. Comput. Syst. Sci.* **2002**, *65*, 587–609. [CrossRef]
43. Feijao, P.; Meidanis, J. SCJ: A breakpoint-like distance that simplifies several rearrangement problem. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2011**, *8*, 1318–1329. [CrossRef] [PubMed]
44. Yancopoulos, S.; Attie, O.; Friedberg, R. Efficient sorting of genomic permutations by translocation, inversion and block interchange. *Bioinformatics* **2005**, *21*, 3340–3346. [CrossRef] [PubMed]
45. Bulteau, L.; Fertin, G.; Rusu, I. Sorting by transpositions is difficult. *SIAM J. Discret. Math.* **2012**, *26*, 1148–1180. [CrossRef]
46. Silva, L.; Kowada, L.; Rocco, N.; Walter, M. *An Algebraic 1.375-Approximation Algorithm for the Transposition Distance Problem*; Elsevier: Amsterdam, The Netherlands, 2021; Submitted.
47. Sankoff, D. Minimal mutation trees of sequences. *SIAM J. Appl. Math.* **1975**, *28*. [CrossRef]
48. Pe'er, I.; Shamir, R. The median problems for breakpoints are NP-complete. *BMC Bioinform.* **1998**, *5*. Available online: https://www.researchgate.net/profile/Ron-Shamir/publication/220138763_The_median_problems_for_breakpoints_are_NP-complete/links/02bfe50e41b4bbed55000000/The-median-problems-for-breakpoints-are-NP-complete.pdf (accessed on 8 May 2021).
49. Hannenhalli, S.; Pevzner, P.A. Transforming cabbage into turnip (polynomial algorithm for sorting signed permutations by reversals). *J. ACM* **1999**, *48*, 1–27. [CrossRef]
50. Bourque, G.; Pevzner, P. Genome-Scale Evolution: Reconstructing Gene Orders in the Ancestral Species. *Genome Res.* **2002**, *12*, 26–36.
51. Alekseyev, M.; Pevzner, P. Breakpoint graphs and ancestral genome reconstructions. *Genome Res.* **2009**, *19*, 943–957. [CrossRef]
52. Xu, A.; Moret, B. GASTS: Parsimony Scoring under Rearrangements. In *Algorithms in Bioinformatics*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 351–363.
53. Zheng, C.; Sankoff, D. On the PATHGROUPS approach to rapid small phylogeny. *BMC Bioinform.* **2011**, *12*, S4. [CrossRef]
54. Kovac, J.; Brejova, B.; Vinar, T. A practical algorithm for ancestral rearrangement reconstruction. In *International Workshop on Algorithms in Bioinformatics*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 163–174.
55. Shao, M.; Lin, Y. Approximating the edit distance for genomes with duplicate genes under DCJ, insertion and deletion. *BMC Bioinform.* **2012**, *13*, S13. [CrossRef]
56. Willing, E.; Zaccaria, S.; Braga, M.; Stoye, J. On the inversion-indel distance. *BMC Bioinform.* **2013**, *14*, S3. [CrossRef]
57. Braga, M.; Willing, E.; Stoye, J. Double cut and join with insertions and deletions. *J. Comput. Biol.* **2011**, *18*, 1167–1184. [CrossRef] [PubMed]
58. Compeau, P. DCJ-indel sorting revisited. *Algorithms Mol. Biol.* **2013**, *8*, 1–9. [CrossRef]
59. Avdeyev, P.; Jiang, S.; Aganezov, S.; Hu, F.; Alekseyev, M. Reconstruction of Ancestral Genomes in Presence of Gene Gain and Loss. *J. Comput. Biol.* **2016**, *23*, 150–164. [CrossRef] [PubMed]
60. Lyubetsky, V.; Gershgorin, R.; Gorbunov, K. Chromosome structures: Reduction of certain problems with unequal gene content and gene paralogs to integer linear programming. *BMC Bioinform.* **2017**, *18*, 1–8. [CrossRef]
61. Fertin, G.; Labarre, A.; Rusu, I.; Tannier, E.; Vialette, S. *Combinatorics of Genome Rearrangements*; Istrail, S., Pevzner, P., Waterman, M., Eds.; The MIT Press: Cambridge, MA, USA; London, UK, 2009.

62. Bryant, D. *Comparative Genomics*; Chapter the Complexity of Calculating Exemplar Distances; Kluwer Academic: Dordrecht, The Netherlands, 2000; pp. 207–211.
63. Bulteau, L.; Jiang, M. Inapproximability of (1,2)-exemplar distance. In *IEEE/ACM Transactions on Computational Biology and Bioinformatics*; Institute of Electrical and Electronics Engineers and Association for Computing Machinery: New York City, NY, USA, 2013; pp. 1384–1390.
64. Sankoff, D. Genome rearrangement with gene families. *Bioinformatics* **1999**, *15*, 909–917. [CrossRef]
65. Yin, Z.; Tang, J.; Schaeffer, S.; Bader, D. Exemplar or matching: Modeling DCJ problems with unequal content genome data. *J. Comb. Optim.* **2016**, *32*, 1165–1181. [CrossRef]
66. Lorespi, D.; Tomkins, A. Block edit models for approximate string matching. *Theor. Comput. Sci.* **1997**, *181*, 159–179.
67. Dondi, R.; El-Mabrouk, N. Aligning and Labeling Genomes Under the Duplication-Loss Model. In *Computability in Europe (CiE)*; Lecture Notes in Computer Science; IOS Press: Amsterdam, The Netherlands, 2013; Volume 7921, pp. 97–107.
68. Holloway, P.; Swenson, K.; Ardell, D.; El-Mabrouk, N. Ancestral Genome Organization: An Alignment Approach. *J. Comput. Biol.* **2013**, *20*, 280–295. [CrossRef]
69. Andreotti, S.; Reinert, K.; S, S.C. The Duplication-Loss Small Phylogeny Problem: From Cherries to Trees. *J. Comput. Biol.* **2013**, *20*, 643–659. [CrossRef] [PubMed]
70. Benzaid, B.; Dondi, R.; El-Mabrouk, N. Duplication-Loss Genome Alignment: Complexity and Algorithm. In Proceedings of the 13th International Conference, LATA 2019, Petersburg, Russia, 26–29 March 2019.
71. Bohnenkamper, L.; Braga, M.; Doerr, D.; Stoye, J. Computing the Rearrangement Distance of Natural Genomes. *J. Comput. Biol.* **2021**, *28*, 1–22. [CrossRef]
72. Shao, M.; Lin, Y.; Moret, B. An exact algorithm to compute the double-cut-and-join distance for genomes with duplicate genes. *J. Comput. Biol.* **2015**, *22*, 425–435. [CrossRef] [PubMed]
73. Rubert, D.; Feijao, P.; Braga, M.; Stoye, J.; Martinez, F.V. Approximating the DCJ distance of balanced genomes in linear time. *Algorithms Mol. Biol.* **2017**, *12*, 1–13. [CrossRef] [PubMed]
74. Ma, J.; Ratan, A.; Raney, B.; Suh, B.; Miller, W.; D, D.H. The infinite sites model of genome evolution. *Proc. Natl. Acad. Sci. USA* **2008**, *105*, 14254–14261. [CrossRef]
75. Paten, B.; Zerbino, D.; Hickey, G.; Haussler, D. A unifying model of genome evolution under parsimony. *BMC Bioinform.* **2014**, *15*, 1–31. [CrossRef]
76. Guindon, S.; Gascuel, O. A simple, fast and accurate algorithm to estimate large phylogenies by maximum likelihood. *Syst. Biol.* **2003**, *52*, 696–704. [CrossRef] [PubMed]
77. Stamatakis, A. RAxML-VI-HPC: Maximum likelihood-based phylogenetic analysis with thousands of taxa and mixed models. *Bioinformatics* **2006**, *22*, 2688–2690. [CrossRef] [PubMed]
78. Ronquist, F.; Huelsenbeck, J. MrBayes3: Bayesian phylogenetic inference under mixed models. *Bioinformatics* **2003**, *19*, 1572–1574. [CrossRef] [PubMed]
79. Schreiber, F.; Patricio, M.; Muffato, M.; Pignatelli, M.; Bateman, A. TreeFam v9: A new website, more species and orthology-on-the-fly. *Nucleic Acids Res.* **2013**. [CrossRef]
80. Boussau, B.; Szöllősi, G.; Duret, L.; Gouy, M.; Tannier, E.; Daubin, V. Genome-scale coestimation of species and gene trees. *Genome Res.* **2013**, *23*, 323–330. [CrossRef] [PubMed]
81. Szöllősi, G.J.; Rosikiewicz, W.; Boussau, B.; Tannier, E.; Daubin, V. Efficient exploration of the space of reconciled gene trees. *Syst. Biol.* **2013**, *62*, 901–912. [CrossRef] [PubMed]
82. Vilella, A.; Severin, J.; Ureta-Vidal, A.; Heng, L.; Durbin, R.; Birney, E. EnsemblCompara gene trees: Complete, duplication-aware phylogenetic trees in vertebrates. *Genome Res.* **2009**, *19*, 327–335. [CrossRef]
83. Huerta-Cepas, J.; Capella-Gutierrez, S.; Pryszcz, L.; Denisov, I.; Kormes, D.; Marcet-Houben, M.; Gabald'on, T. PhylomeDB v3.0: An expanding repository of genome-wide collections of trees, alignments and phylogeny-based orthology and paralogy predictions. *Nucleic Acids Res.* **2011**, *39*, D556–D560. [CrossRef] [PubMed]
84. Mi, H.; Muruganujan, A.; Thomas, P. PANTHER in 2013: Modeling the evolution of gene function, and other gene attributes, in the context of phylogenetic trees. *Nucleic Acids Res.* **2012**, *41*, D377–D386. [CrossRef]
85. Chen, K.; Durand, D.; Farach-Colton, M. Notung: Dating Gene Duplications using Gene Family Trees. *J. Comput. Biol.* **2000**, *7*, 429–447. [CrossRef]
86. Zhang, L. On a Mirkin-Muchnik-Smith conjecture for comparing molecular phylogenies. *J. Comput. Biol.* **1997**, *4*, 177–187. [CrossRef]
87. Zmasek, C.M.; Eddy, S.R. A simple algorithm to infer gene duplication and speciation events on a gene tree. *Bioinformatics* **2001**, *17*, 821–828. [CrossRef]
88. Bérard, S.; Gallien, C.; Boussau, B.; Szollosi, G.; Daubin, V.; Tannier, E. Evolution of gene neighborhoods within reconciled phylogenies. *Bioinformatics* **2012**, *28*, 382–388. [CrossRef]
89. Noutahi, E.; Semeria, M.; Lafond, M.; Seguin, J.; Boussau, B.; Gueguen, L.; El-Mabrouk, N.; Tannier, E. Efficient Gene Tree Correction Guided by Genome Evolution. *PLoS ONE* **2016**, *11*, e0159559. [CrossRef]
90. Patterson, M.; Szollosi, G.; Daubin, V.; Tannier, E. Lateral gene transfer, rearrangement, reconciliation. *BMC Bioinform.* **2013**, *14*, S4. [CrossRef] [PubMed]
91. Anselmetti, Y.; Berry, V.; Chauve, C.; Chateau, A.; Tannier, E.; Bérard, S. Ancestral gene synteny reconstruction improves extant species scaffolding. *BMC Genom.* **2015**, *16*, S11. [CrossRef]
92. W, W.D.; Anselmetti, Y.; Patterson, M.; Ponty, Y.; Berard, S.; Chauve, C.; Scornavacca, C.; Daubin, V.; Tannier, E. DeCoSTAR: Reconstructing the ancestral organization of genes or genomes using reconciled phylogenies. *Genome Biol. Evol.* **2017**, *9*, 1312–1319.

93. Duchemin, W. Phylogeny of Dependencies and Dependencies of Phylogenies in Genes and Genomes. Ph.D. Thesis, Université de Lyon, Lyon, France, 2017.
94. Delabre, M.; El-Mabrouk, N.; Huber, K.; Lafond, M.; Mouton, V.; Noutahi, E.; Castellanos, M. Reconstructing the History of Syntenies Through Super-Reconciliation. In *RECOMB-CG*; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2018; pp. 179–195.
95. Delabre, M.; El-Mabrouk, N.; Huber, K.; Lafond, M.; Mouton, V.; Noutahi, E.; Castellanos, M. Evolution through segmental duplications and losses: A super-Reconciliation approach. *Algorithms Mol. Biol.* **2020**, *15*, 499–506. [CrossRef]
96. Aho, A.; Yehoshua, S.; Szymanski, T.; Ullman, J. Inferring a tree from lowest common ancestors with an application to the optimization of relational expressions. *SIAM J. Comput.* **1981**, *10*, 405–421. [CrossRef]
97. Constantinescu, M.; Sankoff, D. An efficient algorithm for supertrees. *J. Classif.* **1995**, *12*, 101–112. [CrossRef]
98. Ng, M.; Wormald, N. Reconstruction of rooted trees from subtrees. *Discret. Appl. Math.* **1996**, *69*, 19–31. [CrossRef]
99. Semple, C. Reconstructing minimal rooted trees. *Discret. Appl. Math.* **2003**, *127*, 489–503. [CrossRef]
100. Bryant, D. A classification of consensus methods for phylogenetics. *DIMACS Ser. Discret. Math. Theor. Comput. Sci.* **2003**, *61*, 163–184.
101. Lafond, M.; El-Mabrouk, N.; Huber, K.; Moulton, V. The complexity of comparing multiply-labelled trees by extending phylogenetic-tree metrics. *Theor. Comput. Sci.* **2019**, *760*, 15–34. [CrossRef]
102. Jansson, J.; Lemence, R.; Lingas, A. The complexity of inferring a minimally resolved phylogenetic supertree. *SIAM J. Comput.* **2012**, *41*, 272–291. [CrossRef]
103. Huber, K.; Moulton, V.; Spillner, A. Computing a consensus of multilabeled trees. In Proceedings of the 14th Workshop on Algorithm Engineering and Experiments (ALENEX 2012), Kyoto, Japan, 16 January 2012; pp. 84–92.
104. Lott, M.; Spillner, A.; Huber, K. Inferring polyploid phylogenies from multiply-labeled gene trees. *BMC Evol. Biol.* **2009**, *9*, 216. [CrossRef]
105. Gascon, M.; Dondi, R.; El-Mabrouk, N. Complexity and algorithm for MUL-tree pruning. In Proceedings of the IWOCA 2021—32nd International Workshop on Combinatorial Algorithms, Ottawa, ON, Canada, 5–7 July 2021; Lecture Notes in Computer Science.
106. Paszek, J.; Gorecki, P. Efficient Algorithms for Genomic Duplication Models. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2017**, *15*, 1515–1524. [CrossRef]
107. Fellows, M.; Hallet, M.; Stege, U. On the multiple gene duplication problem. In Proceedings of the 9th International Symposium on Algorithms and Computation, Taejon, Korea, 14–16 December 1998; pp. 347–356.
108. Paszek, J.; Górecki, P. Genomic duplication problems for unrooted gene trees. *BMC Genom.* **2016**, *17*, 165–175. [CrossRef]
109. Guigo, R.; Muchnik, I.; Smith, T. Reconstruction of Ancient Molecular Phylogeny. *Mol. Phylogenet. Evol.* **1996**, *6*, 189–213. [CrossRef] [PubMed]
110. Czabarka, E.; Szkély, L.; Vision, T. Minimizing the number of episodes and Gallai's theorem on intervals. *arXiv* **2012**, arXiv:1209.5699.
111. Page, R.; Cotton, J. Vertebrate Phylogenomics: Reconciled Trees and Gene Duplications. *Pac. Symp. Biocomput.* **2002**, 536–547. [CrossRef]
112. Bansal, M.S.; Eulenstein, O. The multiple gene duplicationproblem revisited. *Bioinformatics* **2008**, *24*, i132–i138. [CrossRef] [PubMed]
113. Burleigh, J.G.; Bansal, M.S.; Wehe, A.; Eulenstein, O. Locating multiple gene duplications through reconciled trees. In Proceedings of the Research in Computational Molecular Biology, Singapore, 30 March–2 April 2008; Volume 4955, pp. 273–284.
114. Luo, C.; Chen, M.; Chen, Y.; Yang, R.W.L.; Liu, H.; Chao, K. Linear-time algorithms for the multiple gene duplication problems. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2011**, *8*, 260–265.
115. Paszek, J.; Górecki, P. Inferring duplication episodes from unrooted gene trees. *BMC Genom.* **2018**, *19*, 288. [CrossRef]
116. Dondi, R.; Lafond, M.; Scornavacca, C. Reconciling Multiple Genes Trees via Segmental Duplications and Losses. *Algorithms Mol. Biol.* **2019**, *14*, 1–9. [CrossRef]
117. Zhou, L.; Huang, B.; Meng, X.; Wang, G.; Wang, F.; Xu, Z.; Song, R. The amplification and evolution of orthologous 22-kDa α-prolamin tandemly arrayed genes in *coix*, sorghum and maize genomes. *Plant Mol. Biol.* **2010**, *74*, 631–643. [CrossRef]
118. Shoja, V.; Zhang, L. A Roadmap of Tandemly Arrayed Genes in the Genomes of Human, Mouse, and Rat. *Mol. Biol. Evol.* **2006**, *23*, 2134–2141. [CrossRef] [PubMed]
119. Song, G.; Zhang, L.; Vinar, T.; Miller, W. Inferring the recent duplication history of a gene cluster. In *Comparative Genomics*; Lecture Notes in Computer Science; Ciccarelli, F., Miklós, I., Eds.; Springer: Berlin/Heidelberg, Germany, 2009; Volume 5817.
120. Vinař, T.; Brejová, B.; Song, G.; Siepel, A. Reconstructing Histories of Complex Gene Clusters on a Phylogeny. *J. Comput. Biol.* **2010**, *17*, 1267–1269. [CrossRef]
121. Zhang, Y.; Song, G.; Hsu, C.; Miller, W. Simultaneous History Reconstruction for Complex Gene Clusters in Multiple Species. Available online: https://www.worldscientific.com/doi/abs/10.1142/9789812836939_0016 (accssed on 8 April 2021).
122. Fitch, W. Phylogenies constrained by cross-over process as illustrated by human hemoglobins and a thirteen-cycle, eleven amino-acid repeat in human apolipoprotein A-I. *Genetics* **1977**, *86*, 623–644. [CrossRef]
123. Bertrand, D.; Gascuel, O. Topological rearrangements and local search method for tandem duplication trees. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2005**, *2*, 15–28. [CrossRef] [PubMed]
124. Bertrand, D.; Lajoie, M.; El-Mabrouk, N. Inferring Ancestral Gene Orders for a Family of Tandemly Arrayed Genes. *J. Comput. Biol.* **2008**, *15*, 1063–1077. [CrossRef] [PubMed]
125. Ma, J.; Ratan, A.; Raney, B.J.; Sush, B.B.; Zhang, L.; Miller, W.; Haussler, D. DUPCAR: Reconstructing Contiguous Ancestral Regions with Duplication. *J. Comput. Biol.* **2008**, *15*, 1007–1027. [CrossRef] [PubMed]

126. Lajoie, M.; Bertrand, D.; El-Mabrouk, N. Inferring the Evolutionary History of Gene Clusters from Phylogenetic and Gene Order Data. *Mol. Biol. Evol.* **2010**, *27*, 761–772. [CrossRef]
127. Savard, O.T.; Bertrand, D.; El-Mabrouk, N. Evolution of orthologous tandemly arrayed gene clusters. *BMC Bioinform.* **2011**, *12*, S2.
128. Doyon, J.P.; Scornavacca, C.; Gorbunov, K.Y.; Szöllősi, G.J.; Ranwez, V.; Berry, V. An efficient algo. for gene/species trees parsimonious reconciliation with losses, duplications and transfers. In *RECOMB-CG*; Lecture Notes in Computer Science; Springer: Berlin, Germany, 2010; Volume 6398, pp. 93–108.
129. Doyon, J.; Ranwez, V.; Daubin, V.; Berry, V. Models, algorithms and programs for phylogeny reconciliation. *Briefings Bioinform.* **2011**, *12*, 392–400. [CrossRef] [PubMed]
130. Hallett, M.; Lagergren, J. Efficient algorithms for lateral gene transfer problems. In Proceedings of the the Fifth Annual International Conference on Computational Biology, Montreal, QC, Canada, 22–25 April 2001; pp. 149–156.
131. Ovadia, Y.; Fielder, D.; Conow, C.; Libeskind-Hadas, R. The cophylogeny reconstruction problem is NP-complete. *J. Comput. Biol.* **2011**, *18*, 59–65. [CrossRef]
132. Bansal, M.; Alm, E.; Kellis, M. Efficient algorithms for the reconciliation problem with gene duplication, horizontal transfer and loss. *Bioinformatics* **2012**, *28*, 283–291. [CrossRef] [PubMed]
133. Tofigh, A.; Hallett, M.; Lagergren, J. Simultaneous identification of duplications and lateral gene transfers. *IEEE/ACM Trans. Comput. Bioinform.* **2011**, *8*, 517–535. [CrossRef]
134. Tofigh, A. Using Trees to Capture Reticulate Evolution: Lateral Gene Transfers and Cancer Progression. Ph.D. Thesis, KTH Royal Institute of Technology, Stockholm, Sweden, 2009.
135. David, L.; Alm, E. Rapid evolutionary innovation during an Archaean genetic expansion. *Nature* **2011**, *469*, 93–96. [CrossRef] [PubMed]
136. Libeskind-Hadas, R.; Charleston, M. On the computational complexity of the reticulate cophylogeny reconstruction problem. *J. Comput. Biol.* **2009**, *16*, 105–117. [CrossRef]
137. Anselmetti, Y.; El-Mabrouk, N.; Lafond, M.; Ouandraoua, A. Gene Tree and Species Tree Reconciliation with Endosymbiotic Gene Transfer. Available online: http://www-labs.iro.umontreal.ca/~mabrouk/Publications/ISMB2021.pdf (accessed on 8 April 2021).
138. Bollback, J. SIMMAP: Stochastic character mapping of discrete traits onphylogenies. *BMC Bioinform.* **2006**, *7*, 1–7. [CrossRef]
139. Huelsenbeck, J.; Nielsen, R.; Bollback, J. Stochastic mapping of morphological characters. *Syst. Biol.* **2003**, *52*, 131–158. [CrossRef]
140. Simon, D.; Larget, B. *Bayesian Analysis to Describe Genomic Evolution by Rearrangement (BADGER)*; Version 1.02 Beta; Department of Mathematics and Computer Science, Duquesne University: Pittsburgh, PA, USA, 2004.
141. Roger, A.; Munoz-Gomez, S.; Kamikawa, R. The Origin and Diversification of Mitochondria. *Curr. Biol.* **2017**, *27*, R1177–R1192. [CrossRef] [PubMed]
142. Sloan, D.B.; Warren, J.M.; Williams, A.M.; Wu, Z.; Abdel-Ghany, S.E.; Chicco, A.J.; Havird, J.C. Cytonuclear integration and co-evolution. *Nat. Rev. Genet.* **2018**, *19*, 635–648. [CrossRef]
143. Brandvain, Y.; Wade, M. The Functional Transfer of Genes From the Mitochondria to the Nucleus: The Effects of Selection, Mutation, Population Sizeand Rate of Self-Fertilization. *Genetics* **2009**, *182*, 1129–1139. [CrossRef] [PubMed]
144. Kelly, S. The economics of endosymbiotic gene transfer and the evolution of organellar genomes. *bioRxiv* **2020**. [CrossRef]