RESEARCH

The Small Parsimony Problem under the Single-Cut-or-Join with Single-Genes Duplication model

Aniket Mane[1], Pedro Feijão[2] and Cedric Chauve[1*]

[*]Correspondence: cedric.chauve@sfu.ca
[1]Department of Mathematics, Simon Fraser University, 8888 University Drive, Burnaby (BC), Canada
Full list of author information is available at the end of the article

Abstract

Background: Accurate reconstruction of ancestral gene orders is an important step towards understanding genome evolution. The Small Parsimony Problem has been extensively studied in this regard. The problem aims at finding the gene orders at internal nodes of a given phylogenetic tree such that the overall distance along the tree is minimized. However, this problem is intractable in most genome rearrangement models, especially when gene duplications are considered.

Results: In this work, we present a weighted version of the Small Parsimony Problem that accounts for the presence of duplicate genes. We study the problem in a variant of the Single-Cut or Join that accounts for gene duplication, assuming that duplicate genes originate from single-gene duplication events. We propose an Integer Linear Program to solve the problem that combines homology- and parsimony-based approaches to ancestral reconstruction and analyze its performance on simulated data sets and on a real data set of *Anopheles* mosquito genomes.

Keywords: Small Parsimony Problem; Gene order with duplicates; Single-Cut-or-Join

1 Background

Reconstruction of ancestral genomes is a well-studied problem in comparative genomics. Typically, ancestral genome reconstruction methods take as input a phylogenetic tree (a species tree) with extant gene orders at its leaves with the aim of computing the gene orders at the internal nodes of the tree, while optimizing a suitable criterion.

Approaches designed to address the Small Parsimony Problem can be widely divided into two types: homology-based and parsimony-based. Homology-based approaches were introduced in [1] and do not consider genome rearrangements directly. Instead, they rely on the use of *conserved genomic features*, such as gene adjacencies or common intervals, associated to specific internal nodes of the species tree and obtained by the comparison of the extant gene orders. In a following step, these genomic features can then be assembled into larger gene orders for the considered ancestral species, often called Contiguous Ancestral Regions (CARs) [2, 3].

On the other hand, parsimony-based approaches are guided by minimizing the evolutionary cost, in a given genome rearrangement model, along the branches of the given species tree. This approach builds upon many tractability results on the pairwise genome rearrangement distance problem [4] and is called the Small Parsimony Problem (SPP). The aim of these methods is to reconstruct all ancestral gene orders jointly so that the rearrangement scenarios between successive gene orders along the species tree can be explained using the minimum number of evolutionary events, in the chosen evolutionary model. However, even when restricted to its simplest instance, which translates to the median problem, the SPP has been proved to be NP-hard for most genome rearrangement models [5, 6, 7, 8]. The only strong tractability result for the SPP has been obtained in the Single-Cut-or-Join (SCJ) model [9] (see also [10] for limited tractability results on a variant of this model).

All results discussed above consider gene orders with no duplicated genes. However it is known that gene duplication plays an important role in genome evolution. This has motivated a large body of work on genome rearrangement problems accounting for duplicated genes. Unfortunately, in most cases, even computing the pairwise distance between genomes with duplicates is hard [11, 12]. As a result, there are very few methods aimed at reconstructing ancestral gene orders in a framework including gene duplication. The first work toward this goal was due to Sankoff and ElMabrouk [13] (see also [14]), who introduced the idea of using reconciled gene trees to define the gene content of ancestral genomes and orthology relations between genes; this idea was later used in the homology-based method DUPCAR [15] that requires however a dated species tree in order to order gene duplication events. Other works works include [16], GapAdj [17] that assumes that gene duplications originate from Whole-Genome Duplications (WGD), PMAG++ [18, 19], that relies on a binary encoding of adjacencies and a probabilistic evolutionary model of binary characters to infer ancestral adjacencies and the homology-based method MULTIRES [20] that requires a preliminary set of CARs as well as an upper bound on the number of duplications per gene. It follows that the problem of reconstructing ancestral gene orders with duplicated genes is still in need of novel methods, which motivates the work we present in this paper.

In [21], we introduced a tractable pairwise distance problem, the SCJ-TD-FD directed distance, that considers the distance between an ancestral genome and a descendant genome, in a model where genome rearrangement are SCJ events and gene duplications are single-gene events. We showed that in this model, the directed pairwise distance – the distance between an ancestral genome and a descendant genome – is tractable. More recently, we showed that, in the same model, the rooted median problem is NP-hard [22]. This latter result implies that the SPP is also intractable in the SCJ-TD-FD model. However, we also showed that the rooted median can be solved by a simple Integer Linear Program (ILP) and can reconstruct median gene order with high accuracy. In this paper, we study the weighted SPP in the SCJ-TD-FD model, that combines three ideas: (1) using reconciled gene trees to define the gene content of ancestral gene orders and orthology relations between genes; (2) working in the SCJ-TD-FD model in which the directed distance is tractable and easily described in an ILP framework; (3) combining both the homology-based and the parsimony-based approaches, as done in [10]. For this problem, we design an ILP that can solve very efficiently the weighted SPP, including also the ability to constraint optimal ancestral gene orders to be composed only of linear fragments. Our ILP can also create extant gene adjacencies, thus potentially improving the scaffolding of the provided extant genomes using a comparative approach [23].

2 Preliminaries and problem definition

2.1 Genome representation

A *genome* consists of a set of *chromosomes*, which are either linear or circular ordered sets of oriented genes. Each *gene* g has a pair of *gene extremities* g_h and g_t, respectively denoting the head and tail of the gene. An *adjacency* is a pair of gene extremities that are connected in the genome. From now, to make notations shorter, we denote an adjacency $\{x, y\}$ by xy.

The number of copies of a gene g in a genome G is called its *copy number*, denoted by $n_G(g)$. A genome in which each gene has a copy number of 1 is a *trivial genome*. A trivial genome can then be represented by the set of its adjacencies. On the contrary, a non-trivial genome may contain multiple genes from the same family. If genes from the same family are not distinguished (for example denoting the copies of g as $g^1, g^2, ..., g^k$, where k is the copy number of g), its adjacencies might form a multi-set if there is an adjacency between genes of the same two families that occur multiple time in the genome. In this context, such a multi-set of adjacencies might not represent the genome unambiguously, unless copies of the same gene have been distinguished as described previously, in which case the adjacencies form a set.

2.2 Evolutionary model

We use the model described in [21], which accounts for genome rearrangements and single-gene duplications, extended to account for gene gains. Genome rearrangements are modeled using *Single-Cut-or-Join* (SCJ) operations, which either delete an adjacency from the genome or form a new adjacency by joining a pair of free extremities, respectively. For duplication events, we consider two types of single-gene duplications, namely: *Tandem Duplications* (TD) and *Floating Duplications* (FD) (see Figure 1). A tandem duplication of an existing gene g introduces an extra copy of g, say g', by adding an adjacency $g_h g'_t$, and, if there was an adjacency $g_h x$ by replacing it by the adjacency $g'_h x$. A floating duplication introduces an extra copy g' of a gene g as a single-gene circular chromosome by adding the adjacency $g'_h g'_t$. Considering an ancestral genome A and a descendant genome D, a *gene gain* occurs when a gene in D belongs to a family absent in A (the gene family was gained along the branch from A to D).

2.3 The directed distance model

Let A and D be two genomes such that D is a descendant of A. We assume that for each gene in D we know which unique ancestral gene (if any) it evolved from along the branch from A to D (called *orthology relations* from now); note that several genes of D can have the same ancestor in A if this ancestral gene was duplicated during the evolution leading to D.

Assuming that each gene of D evolved from a gene in A and each gene of A is ancestor of at least one gene in D, we introduced in [21] the *SCJ-TD-FD distance* from A to D as the smallest number of SCJ, TD and FD needed to transform the adjacency set of A into the adjacency multi-set of D. We showed in [21] that this quantity can be computed in linear time. In the present work, we extend this directed distance to account for gene gains by modelling gene gains (losses) along a branch of the species tree by introducing (deleting) the gained (lost) gene. Thus, apart from the cost accounted through the SCJ distance, there is an additional cost of 2 - 1 for the floating duplication and 1 for the related cut (join). We denote by $d(A, D)$ the directed distance from A to D in this augmented model.

2.4 Problem statements

Let $S = (V, E)$ be a given species tree, augmented by gene orders at its leaves representing extant genomes. In our framework, we assume that for each gene family g, a *reconciled gene tree* T_g has been computed that provides the gene content for the family g at any ancestral species (internal node) u of S as well as orthology relations along each branch of S. As a consequence, we know the precise *gene content* of each species. We refer the reader to [24, 25, 26] for reviews on reconciled gene trees. Last, we assume that for each ancestral species u, we are given a set of candidate *ancestral adjacencies*, where each such candidate adjacency a is provided with a weight $w_{v,a} \in [0, 1]$ measuring the confidence in the hypothesis that adjacency a was actually present in ancestral species v. We denote by C_v the set of candidate adjacencies for species v. For extant species, candidate adjacencies are the provided extant adjacencies.

An *valid mixed* (resp. *linear*) set of candidate adjacencies for species v is a subset of C_v that admits a representation into a set of linear and circular (resp. linear) chromosomes. Consider we have for each species v a valid (either mixed or linear) set $A_v \subseteq C_v$ of adjacencies, where $a \in A_v$ is said to be *kept* while $a \in C_v - A_v$ is said to be *discarded*, encoded by binary variables $p_{v,a} = 1$ if adjacency a belongs to A_v (se way a is *kept* in v) and 0 otherwise (a is discarded). The *cost* of such an assignment of adjacencies to ancestral species is then given by the following formula:

$$\sum_{v \in V} \left(\gamma \left(\sum_{a \in C_v - A_v} w_{v,a} \right) + (1 - \gamma) d(A_{p(v)}, A_v) \right) \tag{1}$$

where $p(v)$ is the parent species of v in S and $\gamma \in [0, 1]$ is a user-defined factor aimed at balancing the weights of the discarded adjacencies and the total distance over the species tree S. We address the SPP in this context through two optimization problems

The Weighted Mixed SCJ-TD-FD Small Parsimony Problem: Compute a valid mixed set of candidate adjacencies for each species of S of minimum cost.

The Weighted Linear SCJ-TD-FD Small Parsimony Problem: Compute a valid linear set of candidate adjacencies for each species of S of minimum cost.

3 Methods

Both optimization problems introduced in the previous section generalize the Weighted SCJ Small Parsimony Problem introduced in the context of evolution without duplications [10] and shown to be NP-hard even in the mixed case. To solve them, we designed an Integer Linear Program (ILP) that we describe in this section.

3.1 An ILP for the Weighted Mixed SCJ-TD-FD Small Parsimony Problem

Consider an arbitrary, possibly non-optimal, solution of the Weighted Mixed SCJ-TD-FD Small Parsimony Problem. For a species v, we encode the valid subset of adjacencies A_v defining the solution using binary variables $p_{v,a}$ such that $p_{v,a} = 1$ if $a \in A_v$ and $p_{v,a} = 0$ if $a \in C_v - A_v$; note that an adjacency a can appear several time in C_v, in which case there is one decision variable $p_{v,a}$ per copy of a in C_v.

The first term of the objective function (1) can be encoded as

$$\gamma \sum_{v \in V} \sum_{a \in C_v} (1 - p_{v,a}) w_{v,a} \tag{2}$$

To encode the distance term of (1), we follow the technique described in [22] for the rooted median problem for the $d_{SCJTDFD}$ distance. Let uv be a branch of S, with $u = p(v)$. For a gene g we denote and a species v we denote by $n_v(g)$ the number of copies of g in v (known a priori from the gene content of species v, denoted by Γ_v). We then denote by $\delta(u, v)$ the absolute value of the difference in the number of genes in u and the number of genes in v, $\delta(u, v) = ||\Gamma_u| - |\Gamma_v||$. As shown in [22], the distance from u to v can then be encoded as

$$d(u, v) = |A_u - A_v| - |A_v - A_u| + 2\delta(u, v) - 2 \sum_{g \in \Gamma_u} t_{g,v} + 2 \sum_{g \in \Gamma_u} \alpha_{g,v} \tag{3}$$

where $t_{g,v}$ and $\alpha_{g,v}$ are defined, for any gene $g \in \Gamma_u$, by

$$\beta_{g,v} = \max \left(0, \max_{\substack{x_h y_t \in C_v \\ a(x) = a(y) = g}} p_{v, x_h y_t} \right)$$

$$\alpha_{g,v} = \min(\beta_{g,v}, \Lambda_{g,v} - \lambda_{g,v})$$

$$t_{g,v} = nc_v(g) - \lambda_{g,v}.$$

with

$$nc_v(g) = \sum_{\substack{x_h y_t \in C_v \\ a(x) = a(y) = g}} p_{v, x_h y_t}, \ \lambda_{g,v} = \left\lfloor \frac{nc_v(g)}{n_v(g)} \right\rfloor, \ \Lambda_{g,v} = \left\lceil \frac{nc_v(g)}{n_v(g)} \right\rceil.$$

Note that for $\Lambda_{g,v}$ above, the constraints of the type $x = \lceil y \rceil$ are not linear, but it can be replaced by the constraint $y \leq x \leq y + \epsilon$, where ϵ is very close to 1, say 0.999. A similar trick can be used for floor functions.

We now need to encode the term $|A_u - A_v| - |A_v - A_u|$. We call an adjacency $xy \in C_u$ as the *parent adjacency* of adjacency $x'y' \in C_v$ if $x = a(x')$ and $y = a(y')$, in which case $x'y'$ is a child (adjacency) of xy. When duplications are allowed from u to v, we might have multiple children of the same parent adjacency, so for every adjacency $a = xy \in C_u$, we denote by F_a the set of its child adjacencies in C_v. Further, using change variables $c_{u,v,a}$ and $c_{v,u,a}$ defined below, we obtain

$$|A_u - A_v| = \sum_{a \in A_u} c_{u,v,a}, \ \text{where } c_{u,v,a} = \max \left\{ 0, p_{u,a} - \sum_{b \in F_a} p_{v,b} \right\}$$

$$|A_v - A_u| = \sum_{a \in A_u} c_{v,u,a}, \text{ where } c_{v,u,a} = \max\left\{0, \sum_{b \in F_a} p_{v,b} - p_{u,a}\right\}.$$

It follows that the distance term of (3) can be encoded as

$$(1-\gamma)\sum_{v \in V}\left(\sum_{a \in A_u} c_{u,v,a} + \sum_{a \in A_u} c_{v,u,a} + 2\delta(u,v) - 2\sum_{g \in \Gamma_u} t_{g,uv} + 2\sum_{g \in \Gamma_u} \alpha_{g,uv}\right) \quad (4)$$

subject to:

$$\sum_{a=g_t y \in C_v} p_{v,a} \text{ and } \sum_{a=g_h y \in C_v} p_{v,a} \leq 1, \quad \forall g \in \Gamma_v, \forall v \in V,$$

these two constraints ensuring that, for each species, each gene extremity belongs to at most one adjacency, and thus the resulting kept adjacencies form a valid mixed set of adjacencies.

For $c_{u,v,a}$ and $c_{v,u,a}$ above, the constraints of the type $x = \max\{0, y\}$ are not linear, but it can be replaced by a pair of constraints, $x \geq y$ and $x \geq 0$, to give the desired value.

3.2 Solving the Weighted Mixed SCJ-TD-FD Small Parsimony Problem

The ILP described in the previous section only ensures that the set of kept adjacencies for each species is a valid mixed set. An approach to prevent the occurrence of circular chromosomes would be to add, for each possible circular chromosome, a new constraint that prevents the chromosome in question from being circular. Such constraints will be referred to as *linearity constraints*. However, in doing so, we get an exponential number of constraints. Thus, introducing all the constraints at once will be too expensive, in terms of both time and space. Furthermore, it is possible that many of the constraints might be redundant and will never impact the feasibility region.

To overcome this issue, we use the *delayed constraint generation* method. Typically, the use of this method involves a given linear programming (LP) instance with a large number of constraints. Instead of dealing with all the constraints at once to solve the Weighted Linear SCJ-TD-FD SPP, we start the initial optimization process by solving the Weighted Mixed SCJ-TD-FD SPP as described above. If the computed optimal solution contains circular chromosomes, then we select a random circular chromosome, say composed of k genes and add a new constraint that prevents all adjacencies of this chromosome to be all kept, which can be modelled by constraining the sum of the set of decision variables $p_{v,a}$ for these adjacencies to be strictly less than k. The resulting ILP is then solved and this process is iterated till all chromosomes in the optimal solution are found to be linear or a user-defined number of iterations is reached. Once this bound is reached, the remaining circular chromosomes, if any, are handled using a greedy approach, in which the least weight adjacency from each circular chromosome is discarded, thus linearizing each genome.

3.3 Implementation

We implemented the ILP described above using the Gurobi Optimizer software [27] (version 7.5.2), which has the interesting feature that, although the solver has to start the optimization process from scratch for each iteration, the problem size is reduced by removing redundant constraints as well as variables, identifying and merging of constraints that form cliques and appropriate rounding of bounds of integer variables. Such presolving techniques significantly improve the efficiency of the optimization process.

In order to compute reconciled gene trees we relied on the reconciliation software ecceTERA [28], that, for a given gene tree, computes a parsimonious reconciliation with the species tree S; ecceTERA was used with default parameters.

To compute and weight candidate adjacencies, we used DeCoStar [29]. For each pair of gene families such that at least two of their extant genes form an extant adjacency, we applied the Boltzmann sampling option of DeCoStar to the corresponding pair of reconciled gene trees. In this setting, DeCoStar samples adjacency evolutionary scenarios under the Boltzmann distribution and the weight of a potential ancestral adjacency is the frequency of sampled scenarios in which it appears; candidate adjacencies are then adjacencies of weight at least 0.5. While using DeCoStar, Boltzmann sampling with temperatures 0.1 and 1 was performed. In each setting, two runs (labelled a and b) were performed to check the consistency of the sampled adjacencies as it looks like there were overflow problems in the stochastic backtrack. In each run, 100 samples were generated by DeCoStar.

4 Results

In order to assess the accuracy of our algorithme for the SPP, we generated simulated data with the tool ZOMBI [30], that allows to simulate in the same process gene family evolution (gene duplication, gain and loss) and genome rearrangements.

Simulation protocol and parameters. In each of our experiments, we used ZOMBI to generate 20 datasets a random species tree with 10 extant genomes, evolved from a ancestral root genome composed of 100 genes with no duplicate (each gene family contains a single gene in the root genome), organized into a unique circular chromosome.

We generated simulated data in three evolutionary settings: no gene loss and moderate genome rearrangement rate (EXP1), no gene loss and high genome rearrangement rate (EXP2), with gene losses and moderate genome rearrangement rate (EXP3). We generated 20 datasets per setting.

For each dataset, ZOMBI generated a random species tree with 10 leaves and an ancestral root genome composed of 100 genes with no duplicate (each gene family contains a single gene in the root genome), organized into a unique circular chromosome. Then, for each dataset, we evolved 10 extant genomes as follows:

- 200 gene duplications occurred randomly over all branches of the species tree;
- in EXP1 and EXP3 the rearrangements were composed of 100 inversions and 100 transpositions, also spread randomly over all branches of the species tree; in EXP2 we simulated instead 500 inversions and 500 transpositions;
- finally, in EXP3 we also simulated 200 random gene losses.

For each experiment, we ran our ILP for the Weighted Mixed SCJ-TD-FD Small Parsimony Problem with five values of γ, $\{0, 0.25, 0.5, 0.75, 1\}$.

Using perfect data. In a first set of experiments, we analyzed our datasets considering as candidate adjacencies the true ancestral adjacencies provided by ZOMBI, all with weight 1. This experiment, that assumes the unrealistic setting of perfect input data, was intended to assess the impact of the simplifying assumptions of the SCJ-TD-FD evolutionary model (single-cut and single join rearrangement events, single-gene duplications) on its ability to recover accurate ancestral gene order from perfect input data. We measure this accuracy using the *precision*, *recall* and F_1 statistics, computed by comparing the ancestral adjacencies selected by the ILP to the true ancestral adjacencies. Table 1 shows these statistics, averaged over the 20 runs per setting.

Using perfect gene content and inferred candidate ancestral adjacencies. In the second set of experiments, we evaluated the impact of providing our ILP with potentially erroneous candidate ancestral adjacencies. To do so, we computed, from the true reconciled gene trees provided by ZOMBI (thus ensuring a perfect gene content at each ancestral species) candidate ancestral adjacencies using DeCoStar [29] as described in Section 3.3. Table 2 shows the resulting accuracy statistics, computed as in the previous experiment.

Using perfect gene trees and inferred reconciliations and candidate ancestral adjacencies.

In the final set of experiments, we evaluated the performance of the ILP in the presence of reconciled gene trees. Instead of using the gene trees provided by ZOMBI, we used ecceTERA to obtain reconciliations with the species tree. This increases the potential noise in the input as the gene content in the ancestral genomes is no longer perfect. Once again, we use DeCoStar to obtain candidate ancestral adjacencies. Table 3 shows the resulting statistics.

Results obtained with PMAG++. We compared our results with the ancestral gene orders obtained using PMAG++ [18, 19], that was recently shown to outperform existing methods to reconstruct ancestral gene orders with unequal gene content [31]. Table 4 shows the resulting accuracy statistics.

Discussion. Our results on simulated data show that overall our ILP provides accurate results, slightly better than the results obtained with PMAG++.

Table 1 shows that with perfect data, as expected we recover only true ancestral adjacencies, but more interesting, we also recover most of the ancestral adjacencies when $\gamma > 0$. The recall rate grows with γ, and it is interesting to observe that with $\gamma = 0$ (i.e. the objective function is based solely on evolutionary parsimony in the SCJ-TD-FD model), even in this perfect setting the recall rate is low in EXP2 and EXP3. This illustrates well that the SCJ-TD-FD model is a simplified evolutionary model that does not cope well with a large number of evolutionary events, which motivated the definition of the objective function of our ILP that also includes weighted candidate adjacencies.

The results obtained in the experiment where we know the ancestral gene content but infer ancestral gene adjacencies (Table 2) illustrate the expected positive impact of knowing an accurate ancestral gene content prior to reconstructing ancestral gene orders. Aside from $\gamma = 0, 1$ in EXP1, our ILP outperforms PMAG++ in precision and F_1 stastitics, although our method is obviously advantaged by being provided with an exact ancestral gene content. PMAG++ obtains a higher recall, at the expenses of a significantly lower precision; generally our precision is higher than the precision of PMAG++, with the exception of $\gamma = 1$. It is also interesting to comment on the balance between precision and recall depending on γ: it appears that in EXP1 (Table 3), at $\gamma = 0.75$ there is a significant increase of the recall together with a much lower decrease of the precision. This suggests that in this range of values for γ our method can recover a large number of true ancestral adjacencies at the expense of a low rate of false positives; actually, in all our experiments, the $F_{0.5}$ statistics that weighs recall lower than precision to penalize more incorrect ancestral adjacencies, reaches its maximum at $\gamma = 0.5$.

The influence of gene losses can be clearly seen in EXP3. In order to avoid the cost of cuts, the ILP tends to avoid selecting adjacencies involving genes that are lost further down the phylogeny. This results in a sparse reconstruction of ancestral genomes when losses are part of the evolution. The drop in F_1 statistics from table 2 to table 3 also highlights the reliance of the ILP on perfect gene content. The gene content in the final set of experiments was determined by reconciling gene trees with species tree. This step can potentially increase the noise in the data by mapping the genes to incorrect species, thereby losing any adjacencies involving to the mismatched genes. Moreover, these mismatched genes are akin to lost genes in terms of their influence on the ILP. Thus, the recall suffers significantly when gene trees are significantly different from the species trees.

5 Conclusion

In this work, we study the Small Parsimony Problem in a duplication-aware framework. We reconstruct the ancestral genomes as sets of adjacencies, under the SCJ-TD-FD evolutionary model that accounts for single gene duplications and using an optimality criterion that accounts both for the evolutionary distance along the given species tree and for prior weights on candidate ancestral adjacencies. Following the results of [22] on the rooted median problem in the SCJ-TD-FD problem and of [10] on the Weighted SCJ SPP, this variant of the SPP is NP-hard. In order to solve it, we provide an Integer Linear Program, that relies on a polynomial number of variables and constraints for the Weighted Mixed SCJ-TD-FD Small Parsimony Problem and on an iterative delayed constraints generation process for the Weighted Linear SCJ-TD-FD Small Parsimony Problem. Our ILP acts as the final step in a pipeline inferring ancestral gene orders from extant gene orders including duplicated genes, augmented with reconciled gene trees and candidate weighted gene adjacencies [23].

The ILP outperforms PMAG++ in the presence of perfect gene content and in the absence of gene losses. The ILP consistently offers better precision than PMAG++. This is partly due to careful selection of adjacencies by the ILP from the pool of candidate adjacencies. For lower values of $gamma$ however, this selection proves to be too restrictive resulting in a fragmented assembly. The inclusion of gene losses as well as the indirect loss of gene content through the process of reconciliation are both contributing factors for lower recall. However, it is the later that resulted in a significant drop in performance. This can be remedied by improving the determination of gene content from reconciled gene trees. As the final experiment is closer to mimicking the availability of information from real-life, refining the manner in which gene content is determined promises to be a significant step in improving the usability of the ILP.

Future research on the SCJ-TD-FD model could proceed along several directions.

An important observation is the substantial impact of gene gains and losses on the estimation of the overall evolutionary distance. In the current version of our ILP, gene gains and losses are penalized by the cost of the gain or loss together with the cost of the SCJ events required to extract/insert the gained or lost gene. Furthermore, gene gains and losses also affect the assembly as the ILP tends to avoid adjacencies involving such genes in order to reduce the SCJ cost. However, in terms of evolution, one can expect that gene losses are not the result of genome rearrangements, but correspond to duplicated genes accumulating mutations that makes them too distant in terms of sequence similarity to be identified as belonging to their original gene family (e.g. through neofunctionalization or pseudogenization [32]). An upstream approach to this issue could be to improve the clustering of extant genes into gene families, although this is a difficult problem. In the framework we introduced, it would be natural to try to avoid penalizing the apparent rearrangements associated to such events, based on the detection of conservation of the syntenic context around an apparently lost or gained gene. More generally, our work relies on the simplicity of the SCJ-TD-FD evolutionary model to obtain an efficient SPP algorithm. This should motivate looking for extension towards a more realistic evolutionary model (e.g. to account for segmental duplications or whole-chromosomes or whole-genome duplications) that can still lead to efficient ILP to solve the SPP.

Last, solvers such as Gurobi also provide the option to output multiple co-optimal solutions. It may be interesting to sample and evaluate co-optimal solutions. This was achieved by Luhmann *et al.* [10] through the use of an adaptation of the Sankoff-Rousseau dynamic programming algorithm. We plan to address the issue of exploring the set of co-optimal solutions to the SPP in a future research.

Competing interests
The authors declare that they have no competing interests.

Author's contributions
CC designed the work. All authors designed the methods and experiments. All authors analyzed the results ad wrote the paper.

Acknowledgements
CC acknowledges the support of NSERC and ComputeCanada. PCF was funded by Genome Canada. AM was funded by SFU.

Author details
[1]Department of Mathematics, Simon Fraser University, 8888 University Drive, Burnaby (BC), Canada. [2]School of Computing Science, Simon Fraser University, Burnaby (BC), Canada.

References

1. Bergeron, A., Blanchette, M., Chateau, A., Chauve, C.: Reconstructing ancestral gene orders using conserved intervals. In: Algorithms in Bioinformatics, 4th International Workshop, WABI 2004, Bergen, Norway. Lecture Notes in Computer Science, pp. 14–25 (2004). doi:10.1007/978-3-540-30219-3_2
2. Ma, J., Zhang, L., Suh, B.B., Raney, B.J., Burhans, R., Kent, W.J., Blanchette, M., Haussler, D., Miller, W.: Reconstructing contiguous regions of an ancestral genome. Genome research **16**, 1557–1565 (2006)
3. Chauve, C., Tannier, E.: A methodological framework for the reconstruction of contiguous regions of ancestral genomes and its application to mammalian genomes. PLoS Computational Biology **4**(11) (2008). doi:10.1371/journal.pcbi.1000234
4. Fertin, G., Labarre, A., Rusu, I., Tannier, E., Vialette, S.: Combinatorics of Genome Rearrangements. Computational molecular biology. MIT Press, ??? (2009). https://mitpress.mit.edu/books/combinatorics-genome-rearrangements
5. Pe'er, I., Shamir, R.: The median problems for breakpoints are NP-complete. Electronic Colloquium on Computational Complexity (ECCC) **5**(71) (1998)
6. Caprara, A.: The reversal median problem. INFORMS Journal on Computing **15**(1), 93–113 (2003). doi:10.1287/ijoc.15.1.93.15155
7. Tannier, E., Zheng, C., Sankoff, D.: Multichromosomal median and halving problems under different genomic distances. BMC Bioinformatics **10** (2009). doi:10.1186/1471-2105-10-120
8. Kovác, J.: On the complexity of rearrangement problems under the breakpoint distance. Journal of Computational Biology **21**(1), 1–15 (2014). doi:10.1089/cmb.2013.0004
9. Feijão, P., Meidanis, J.: SCJ: A breakpoint-like distance that simplifies several rearrangement problems. IEEE/ACM Transactions on Computational Biology and Bioinformatics **8**(5), 1318–1329 (2011). doi:10.1109/TCBB.2011.34
10. Luhmann, N., Lafond, M., Thevenin, A., Ouangraoua, A., Wittler, R., Chauve, C.: The scj small parsimony problem for weighted gene adjacencies. IEEE/ACM Transactions on Computational Biology and Bioinformatics (2017). doi:10.1109/TCBB.2017.2661761. In press; preliminary version in the proceedings of ISBRA 2016.
11. Blin, G., Chauve, C., Fertin, G., Rizzi, R., Vialette, S.: Comparing genomes with duplications: A computational complexity point of view. IEEE/ACM Transactions on Computational Biology and Bioinformatics **4**(4), 523–534 (2007). doi:10.1145/1322075.1322079
12. Angibaud, S., Fertin, G., Rusu, I., Thévenin, A., Vialette, S.: On the approximability of comparing genomes with duplicates. J. Graph Algorithms Appl. **13**(1), 19–53 (2009)
13. Sankoff, D., El-Mabrouk, N.: Duplication, rearrangement, and reconciliation. In: Comparative Genomics, pp. 537–550 (2000). doi:10.1007/978-94-011-4309-7_46
14. Chauve, C., El-Mabrouk, N., Guéguen, L., Semeria, M., Tannier, E.: Duplication, rearrangement and reconciliation: A follow-up 13 years later. In: Models and Algorithms for Genome Evolution, pp. 47–62 (2013)
15. Ma, J., Ratan, A., Raney, B.J., Suh, B.B., Zhang, L., Miller, W., Haussler, D.: DUPCAR: reconstructing contiguous ancestral regions with duplications. Journal of Computational Biology **15**(8), 1007–1027 (2008). doi:10.1089/cmb.2008.0069
16. Earnest-DeYoung, J.V., Lerat, E., Moret, B.M.E.: Reversing gene erosion - reconstructing ancestral bacterial genomes from gene-content and order data. In: Algorithms in Bioinformatics, 4th International Workshop, WABI 2004, Bergen, Norway, September 17-21, 2004, Proceedings. Lecture Notes in Computer Science, vol. 3240, pp. 1–13 (2004). doi:10.1007/978-3-540-30219-3_1
17. Gagnon, Y., Blanchette, M., El-Mabrouk, N.: A flexible ancestral genome reconstruction method based on gapped adjacencies. BMC Bioinformatics **13**(S-19), 4 (2012). doi:10.1186/1471-2105-13-S19-S4
18. Yang, N., Hu, F., Zhou, L., Tang, J.: Reconstruction of ancestral gene orders using probabilistic and gene encoding approaches. PLOS ONE **9**(10), 1–11 (2014). doi:10.1371/journal.pone.0108796
19. Zhou, L., Tang, J.: Ancestral reconstruction with duplications using binary encoding and probabilistic model. In: 7th International Conference on Bioinformatics and Computational Biology (BICoB 2015), pp. 97–104 (2015). http://www.proceedings.com/25603.html
20. Rajaraman, A., Ma, J.: Reconstructing ancestral gene orders with duplications guided by synteny level genome reconstruction. BMC Bioinformatics **17**(Suppl 14), 414 (2016). doi:10.1186/s12859-016-1262-8
21. Feijão, P., Mane, A.C., Chauve, C.: A tractable variant of the Single Cut or Join distance with duplicated genes. In: Comparative Genomics - 15th International Workshop, RECOMB CG 2017. Lecture Notes in Computer Science, vol. 10562, pp. 14–30 (2017). doi:10.1007/978-3-319-67979-2_2
22. Mane, A.C., Lafond, M., Feijão, P., Chauve, C.: The rooted SCJ median with single gene duplications. In: Comparative Genomics - 16th International Conference, RECOMB-CG 2018, Magog-Orford, QC, Canada, October 9-12, 2018, Proceedings, pp. 28–48 (2018). doi:10.1007/978-3-030-00834-5_2. https://doi.org/10.1007/978-3-030-00834-5_2

23. Anselmetti, Y., Duchemin, W., Tannier, E., Chauve, C., Bérard, S.: Phylogenetic signal from rearrangements in 18 anopheles species by joint scaffolding extant and ancestral genomes. BMC Genomics **19**(2), 96 (2018). doi:10.1186/s12864-018-4466-7
24. Doyon, J.-P., Ranwez, V., Daubin, V., Berry, V.: Models, algorithms and programs for phylogeny reconciliation. Briefings in Bioinformatics **12**(5), 392–400 (2011). doi:10.1093/bib/bbr045
25. Rusin, L.Y., Lyubetskaya, E.V., Gorbunov, K.Y., Lyubetsky, V.A.: Reconciliation of gene and species trees. BioMed Research International **2014**, 642089 (2014). doi:10.1155/2014/642089
26. Szöllösi, G.J., Tannier, E., Daubin, V., Boussau, B.: The Inference of Gene Trees with Species Trees. Systematic Biology **64**(1), 42–62 (2014). doi:10.1093/sysbio/syu048
27. Gurobi Optimization, L.: Gurobi Optimizer Reference Manual (2019). http://www.gurobi.com
28. Jacox, E., Chauve, C., Szöllösi, G.J., Ponty, Y., Scornavacca, C.: ecceTERA: comprehensive gene tree-species tree reconciliation using parsimony. Bioinformatics **32**(13), 2056–2058 (2016). doi:10.1093/bioinformatics/btw105
29. Duchemin, W., Anselmetti, Y., Patterson, M., Ponty, Y., Bérard, S., Chauve, C., Scornavacca, C., Daubin, V., Tannier, E.: DeCoSTAR: Reconstructing the ancestral organization of genes or genomes using reconciled phylogenies. Genome biology and evolution **9**(5), 1312–1319 (2017). doi:10.1093/gbe/evx069
30. Davin, A.A., Tricou, T., Tannier, E., de Vienne, D.M., Szollosi, G.J.: Zombi: A simulator of species, genes and genomes that accounts for extinct lineages. bioRxiv (2018). doi:10.1101/339473. https://www.biorxiv.org/content/early/2018/06/07/339473.full.pdf
31. Feng, B., Zhou, L., Tang, J.: Ancestral genome reconstruction on whole genome level. Current Genomics **18**(4), 306–315 (2017). doi:10.2174/1389202918666170307120943
32. Innan, H., Kondrashov, F.: The evolution of gene duplications: classifying and distinguishing between models. Nature Reviews Genetics **11**, 97 (2010). doi:10.1038/nrg2689

Figures

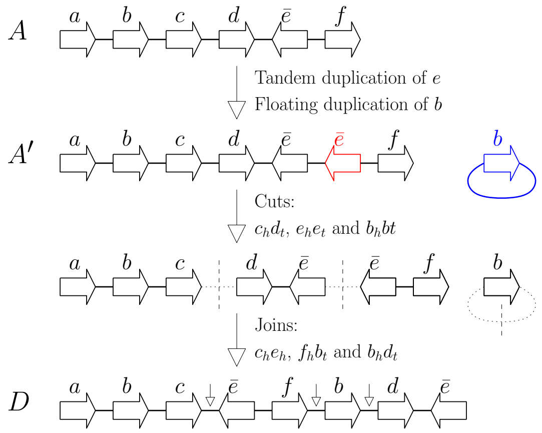

Figure 1 An example of the SCJ-TD-FD model: with tandem (red) and floating (blue) duplications taking place before cuts and joins.

Tables

Table 1 Average accuracy statistics with perfect input data

	EXP1			EXP2			EXP3		
γ	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
0	1.0	0.723	0.837	1.0	0.339	0.494	1.0	0.509	0.669
0.25	1.0	0.859	0.923	1.0	0.412	0.569	1.0	0.654	0.784
0.5	1.0	0.942	0.970	1.0	0.592	0.736	1.0	0.804	0.889
0.75	1.0	0.993	0.996	1.0	0.931	0.964	1.0	0.983	0.991
1	1.0	0.993	0.997	1.0	0.993	0.997	1.0	0.991	0.995

Table 2 Average accuracy statistics with perfect gene content and inferred candidate ancestral adjacencies

	EXP1			EXP2			EXP3		
γ	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
0	0.963	0.625	0.757	0.989	0.150	0.260	0.984	0.516	0.672
0.25	0.959	0.742	0.837	0.940	0.278	0.429	0.973	0.621	0.752
0.5	0.959	0.799	0.871	0.782	0.358	0.491	0.930	0.681	0.781
0.75	0.902	0.835	0.867	0.519	0.453	0.484	0.778	0.697	0.734
1	0.750	0.730	0.740	0.436	0.432	0.434	0.501	0.481	0.490

Table 3 Average accuracy statistics with perfect gene trees and inferred candidate ancestral adjacencies

	EXP1			EXP2			EXP3		
γ	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
0	0.978	0.052	0.098	0.547	0.011	0.022	0.999	0.039	0.082
0.25	0.962	0.069	0.129	0.453	0.024	0.046	0.948	0.052	0.100
0.5	0.962	0.071	0.131	0.453	0.024	0.046	0.948	0.053	0.101
0.75	0.827	0.640	0.722	0.641	0.328	0.434	0.893	0.203	0.328
1	0.762	0.743	0.753	0.438	0.434	0.436	0.449	0.250	0.319

Table 4 Average accuracy statistics of PMAG++

	EXP1	EXP2	EXP3
Precision	0.790	0.0107	0.832
Recall	0.837	0.0106	0.823
F1	0.813	0.0107	0.827