Submitted for recommendation to *Peer Community in Evolutionary Biology*

MaxTiC: Fast ranking of a phylogenetic tree by Maximum Time Consistency with lateral gene transfers

Cédric Chauve[1], Akbar Rafiey[1], Adrian A. Davin[3], Celine Scornavacca[4], Philippe Veber[3], Bastien Boussau[3], Gergely J Szöllősi[5,6], Vincent Daubin[3], and Eric Tannier[2,3]

[1]Department of Mathematics, Simon Fraser University, Burnaby (BC), Canada
[2]Inria Grenoble Rhône-Alpes, F-38334 Montbonnot, France
[3]Univ Lyon, Université Lyon 1, CNRS, Laboratoire de Biométrie et Biologie Évolutive UMR5558, F-69622 Villeurbanne, France
[4]Institut des Sciences de l'Evolution, Université de Montpellier, CNRS, IRD, EPHE 34095 Montpellier Cedex 5, France
[5]MTA-ELTE "Lendulet" Evolutionary Genomics Research Group, Budapest Hungary
[6]Deptartment of Biological Physics, Eötvos Loránd University, Budapest Hungary

Abstract

Lateral gene transfers (LGTs) between ancient species contain information about the relative timing of species diversification. Specifically, the ancestors of a donor species must have existed before the descendants of the recipient species. Hence, the detection of a LGT event can be translated into a time constraint between nodes of a phylogeny if donors and recipients can be identified. When a set of LGTs are detected by interpreting the phylogenetic discordance between gene trees and a species tree, the set of all deduced time constraints can be used to order totally the internal nodes and thus produce a ranked tree. Unfortunately LGT detection is still very challenging and current methods produce a significant proportion of false positives. As a result the set of time constraints is not always compatible with a ranked species tree. We propose an optimization method, which we call MaxTiC (Maximum Time Consistency), for obtaining a ranked species tree compatible with a maximum number of time constraints. The problem in general inherits NP-completeness from feedback arc sets. However we give an exact polynomial time method based on dynamic programming to compute an optimal ranked binary tree supposing that its two children are ranked. We turn this principle into a heuristic to solve the general problem and test it on simulated datasets. Under a wide range of conditions, which we compare to biological datasets, the obtained ranked tree is very close to the real one, confirming the theoretical possibility of dating in the history of life with transfers by maximizing time consistency. MaxTiC is available within the ALE package: `https://github.com/ssolo/ALE/tree/master/misc`

I. Introduction

Telling the evolutionary time [4] is usually achieved by combining molecular clocks and the fossil record. It was pointed out by Gogarten [6] and demonstrated by Szöllősi *et al* [13] that there existed a third source of information about evolutionary time in ancient lateral gene transfers.

Indeed, suppose an ancient species A transfers a gene to another species B, and the latter has descendants $\mathcal{B}$ that are sampled in a phylogenetic study, which is a necessary condition for the transfer to be detected. It is not necessary to assume the same for A, that is, that A has descendants in the same phylogeny [15]. If we call X the most recent common ancestor of A and sampled species, and Y the most recent common ancestor of the species in $\mathcal{B}$, then X must be older than Y because a gene from a descendant of X has been transferred to an ancestor of Y (see Figure 1).

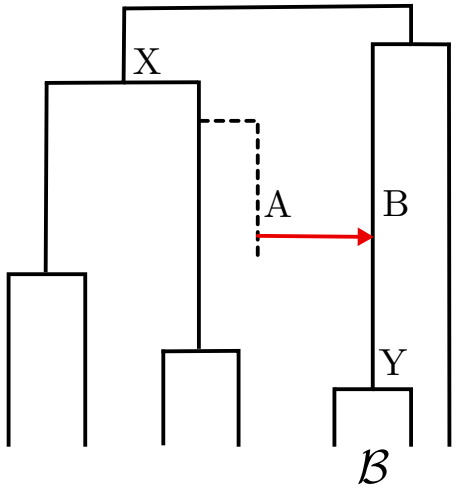

Figure 1: The dating information in transfers. A species tree is depicted, with a transfer from species A to contemporaneous species B. The donor species A possibly belongs to a lineage with no sampled descendants (dotted line in the phylogeny). The transfer from A to B informs that speciation X is older than speciation Y. This precedence relation between X and Y is the time constraint *associated with the transfer from A to B.*

While a single transfer can provide a time constraint between two nodes of a phylogeny, many transfers combined can provide a multitude of time constraints that can be used to determine the time order of the internal nodes of a phylogeny and obtain a *ranked phylogeny* [11].

Such an approach, however, requires that the direction of lateral gene transfers events be specified, which can be challenging [10]. The method by Szöllősi *et al* [13] consisted in searching in the space of ranked trees the one that has the best likelihood according to a model of gene tree species tree reconciliation taking lateral gene transfers into account. Due to the size of the space, it does not scale up to more than a few dozen species.

Here, we describe a fast method to compute a ranked species tree from LGTs detected on an unranked tree. Several pieces of software are available to detect transfers using phylogenetic incongruence between species trees and gene trees without the need of a ranked species tree [2, 12, 16, 7]. We transform transfers into time constraints, and the set of time constraints into a total order.

If all detected transfers were real, this would be the end of the story. Indeed, all transfers would be compatible with the real chronology. Finding a ranked tree compatible with a set of constraints is easy if there is one order compatible with *all* constraints. It is just a matter of ordering the nodes of a directed acyclic graph. However, due to errors or uncertainties in the output of any method, the set of time constraints inferred from transfers is not necessarily entirely compatible with a total order of the species tree nodes. In practice it is never the case. Conflict may be due to errors in either the species tree, the gene tree reconstructions or in species tree gene tree reconciliations, the latter resulting potentially from not taking into account

processes such as incomplete lineage sorting or transfers with replacement of an homologous gene. We then propose to compute a ranked tree which maximizes consistency with a set of constraints, a particular case of the Feedback Arc Set problem. We describe a method called MaxTiC, for Maximal Time Consistency, based on a divide and conquer principle. The divide step consists in solving the problem for subtrees of the species tree. The conquer step consists in exactly solving by dynamic programming the particular case in which a total order on internal nodes is given for each of the two children subtrees.

This conquer step can also be seen as a general method to mix two parts of a species tree which have been independently dated, provided transfers have been detected between the two clades.

We test the MaxTiC method with transfers detected by ALE [16] on simulated data generated with SimPhy [9]. To test the limits of the principle, we use a wide range of transfer rates, population sizes (which has an effect on the gene tree species tree incongruence through incomplete lineage sorting), variations in the species tree. We show that under most conditions, including conditions mimicking biological datasets (we use a fungi and a cyanobacteria datasets for comparison), the ranked tree recovered by the method is very close to the true one (A normalized Kendall τ close to 0.95), but is never exactly the true one because of false transfers inferred by ALE.

We first describe the protocol, including simulations, transfer detection, conversion of each transfer into a time constraint. Then we describe our main algorithm, the exact dynamic programming procedure on a subproblem and how we use it as a heuristic for the general problem. We finally present the results on the simulated datasets and discuss the possibility to date a tree of life with transfers.

II. Simulation and inference of transfers

Generalities. We consider that phylogenetic trees are binary and species trees are rooted. In a species tree a node x is the descendant of a node y, or equivalently y is an ancestor of x if y is on the path from the root to x. We note this relation $x \leq y$, and it defines a partial order on the nodes. A species tree is *ranked* if there is a total order of its internal nodes that agrees with the partial order given by the tree. In that total order, we say from a pair (x, y) with $x < y$ that x is younger than y and y is older than x. Gene trees can be rooted or not, and each of their leaves maps to a leaf of the species tree. Reconciled gene trees are rooted and annotated gene trees, where every node maps to nodes or branches of the species tree and is annotated with speciation, duplication or transfer events [17].

Simulation by SimPhy. We generated simulated datasets with an independent tool[1]. For all sets of parameters, we used SimPhy [9] to generate a ranked species tree with 500 leaves. Along this species tree, we generated typically 1000 gene trees with a population size between 2 and 10^6, null rates of duplications and losses, and a rate of transfers from 10^{-9} to 10^{-5}.

Then we pruned each leaf of the species tree with a probability 0.8, so that the final species tree has approximately 100 leaves. Gene trees are pruned accordingly by removing leaves belonging to the removed species. This simulates a sampling of sequenced species, accounting for species extinction or species absence in the study.

Detection of transfers. Transfers are detected by ALEml_undated, a program from the ALE suite [16]. It takes as input an unranked rooted species tree and an unrooted gene tree, and produces a sample of 100 reconciled gene trees, sampled according to their likelihood under a

[1]Independent meaning that it was developed by an independent team, with other purposes than to test our method. However, it has been developed to validate evolutionary inference methods in general, which is somewhat a dependency [3].

model of duplication, loss, and transfers. Duplication, transfer and loss rates are estimated with a maximum likelihood objective, for each gene family independently, and the 100 reconciled gene trees are sampled according to these ML rates. Transfers from unsampled lineages are handled [14]. We kept transfers found in at least 5% of the reconciliations in one gene family, in order to reduce the noise from improbable transfers.

From transfers to constraints. Each transfer inferred by ALE has an ancestor of the donor species and descendants of the recipient species in the phylogeny. The most recent species in the phylogeny which is an ancestor of the donor is called X, the first descendant of the recipient is called Y, and a constraint is inferred as $X > Y$, which means X is older than Y. We assign to the constraint $X > Y$ the support of the transfer, which is the frequency at which the constraint $X > Y$ is found in the 100 reconciled gene trees, summed across all gene families.

III. Finding a maximum consistent set of constraints with MaxTiC.

Definition. We suppose we have as input a rooted, but otherwise unranked, species tree S and a large set of weighted constraints $\mathcal{C}$, which are directed pairs of nodes of S. We call a constraint *informative* if its two nodes are internal nodes not related by an ancestor/descendant relationship, and we suppose without loss of generality that $\mathcal{C}$ contains only informative constraints.

Some constraints might be conflicting, for example like in Figure 2: Y is found to be older than X, and Z is found to be older than T, but T is an ancestor of Y and X is an ancestor of Z. The two constraints $Y > X$ and $Z > T$ cannot be true at the same time in the context of the drawn species tree.

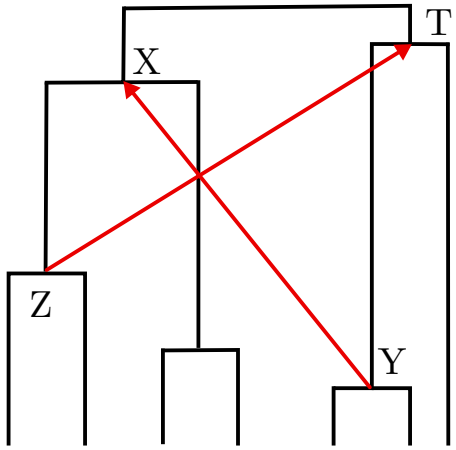

Figure 2: A set of two conflicting constraints. Each of the constraints $Y > X$ and $Z > T$ can be fulfilled by some ranked version of the species tree, but not both.

In general we say that a constraint is *compatible* with a ranked tree if it is directed from an older to a younger node. A subset of constraints on the internal nodes of a tree S is *consistent* if there exists a ranked tree based on S such that all constraints from this subset are compatible with the ranked tree. Otherwise it is *conflicting*. We search for a maximum weight consistent subset of $\mathcal{C}$.

Relation with the Feedback Arc Set. If we see the branches of the unranked species tree as arcs of a directed graph with infinite weight, and the constraints as weighted arcs in this graph, then this problem translates exactly into an instance of the Feedback Arc Set problem. This

classical problem is NP-complete [5], and cannot be approximated with a constant factor. The best algorithms to solve it in practice are local search heuristics. It is equivalent to finding a total order of the nodes of a directed graph, which maximizes the total sum of the arcs xy such that $x > y$ in this order.

NP-hardness. Note that as we have a species tree with infinite weight arcs, we are not in the general case of the Feedback Arc Set problem, so the NP-completeness of our variant is not immediate. However it is easy to reduce the Feedback Arc Set to our problem, leading to the NP-hardness property.

Theorem 1. *The maximum time consistency problem is NP-hard.*

Proof. Let us take any instance of the Feedback Arc Set in the form of a weighted graph with n vertices. Construct a species tree with $2n$ leaves, connected by n cherry nodes (i.e. nodes having two leaves as children), and complete the rest of the tree by a comb. The cherry nodes are identified with the nodes of the graph, so that any arc can be assimilated to a constraint, and a ranked species tree maximizing the set of consistent constraints yields a total order of the vertices of the initial graph maximizing the consistency with the arcs. Any algorithm finding a maximum time consistent set of contraints, applied on the comb with cherries, would find the solution to the feedback arc set. This proves NP-hardness of the maximum time consistency problem. □

A heuristic principle based on divide-and-conquer approximations. Specificities of our problem compared to the Feedback Arc Set can be harnessed to design specific heuristics. In general Feedback Arc Set is approximable within a factor of $\log n$ where n is the size of the graph. The approximation factor is obtained by a divide and conquer strategy. First the graph is cut into two balanced parts. The problem is recursively solved on the two parts and then the two subsolutions are mixed [8]. The presence of an underlying tree for the graph (the species tree) provides a "natural" way to recursively cut the graph into two. Indeed, let r be the root of the species tree. It is always the highest node in any ranked tree. Then define c_1 and c_2 the two children of r (descendants separated by only one edge), and t_1 and t_2 the two subtrees rooted at c_1 and c_2. Define three sets of constraints: those having two extremities in t_1, those having two extremities in t_2, and those having one extremity in t_1 and one in t_2. The subtree t_1 and the first set of constraints, as well as the subtree t_2 and the second set of constraints, define new instances of the problem. So the divide step is to solve independently and recursively the problem on these two instances. This results in ranked trees for t_1 and t_2, that is, two independent total orders of the internal nodes of t_1 and t_2. Constructing an order of all the internal nodes, that is, containing r, the internal nodes of t_1 and the internal nodes of t_2, according to the third set of constraints, is the mixing (conquer) step.

The mixing principle. The mixing step of the algorithm consists in obtaining a ranked tree from two ranked subtrees. In two ways this step alone also gives the solution to more general problems. First it can be applied to general approximation algorithms for the Feedback Arc Set. In Leighton and Rao [8], the mixing step for the Feedback Arc Set was achieved by simply concatenating the two orders obtained from the solutions to the two subproblems. We propose here a better (optimal) way to achieve this mixing by dynamic programming. Thus it improves on the approximation solutions to the general Feedback Arc Set problem (the approximation ratio however is not improved).

Second, it can be viewed as the solution to a more general problem in phylogenetic dating, that we can call Mixing ranks, defined as follows. Suppose we are given a rooted binary tree S, with root r, where the two subtrees t_1 and t_2 rooted by the children of r are ranked. It can be a common situation where two disjoint clades have been dated independently by

any method, including, but not limited to, a recursive application of the divide and conquer principle. Suppose also that possibly conflicting propositions for relative time constraints between nodes of t_1 and t_2 are given, which can be the result of a fossil calibration or transfer detection for example. Then we have to construct a ranked tree for S that contains the input ranks of the children subtrees of the root, and is compatible with a maximum weight subset of time constraints.

The algorithm described below proves that this particular situation of the Feedback Arc Set can be solved in polynomial time.

Theorem 2. MIXING RANKS *can be solved in polynomial running time.*

Indeed, call $a_1, \ldots, a_k$, resp. $b_1, \ldots, b_l$, the sequence of internal nodes of t_1, resp. t_2, decreasingly ordered by their position in the ranked subtree (by convention a_1 and b_1 are the oldest nodes). Call $\mathcal{C}$ the set of weighted constraints between internal nodes of t_1 and t_2. Given a subset N of the internal nodes of the species tree, note $\mathcal{C}_N$ the set of constraints which have both their extremities in N.

Note $N_{ij} = \{a_i, \ldots, a_k, b_j, \ldots, b_l\}$. Let then $s(i,j)$ be the size of the maximum set of consistent constraints in $\mathcal{C}_{N_{ij}}$, also compatible with the orders $a_i, \ldots, a_k$ and $b_j, \ldots, b_l$. It is easy to see that the value of the optimal solution to MIXING RANKS is $s(1,1)$. We compute it recursively with the following equations,

- $s(k+1, j) = s(i, l+1) = 0$ for all i, j,
- $s(i,j) = min(s(i+1,j) + incoming(a_i), s(i, j+1) + incoming(b_j))$, if $i \leq k$ and $j \leq l$,

where $incoming(x)$ is the total weight of the constraints ending on x.

This translates into a dynamic programming scheme. Backtracking along the matrix of $s(i,j)$ gives the optimal mixing of the two orders $a_1, \ldots, a_k$ and $b_1, \ldots, b_l$. Putting r before the mixed order gives an optimal solution.

Applying the mixing algorithm as a conquer step yields a recursive heuristic for the general problem.

Implementation. In our software MaxTiC, we implemented in Python the heuristic recursive principle described above along with a greedy heuristic and a local search approach. The greedy heuristic consists in progressively adding to the species tree the constraints in decreasing order of their weight, provided that they do not create conflict with what has already been added. The local search consists in proposing moves of the total order of the species tree nodes, by taking one node at random and changing its position in the total order to a randomly chosen alternative one, and accepting the move if it is compatible with the partial order given by the species tree and if it increases the value of the solution.

We tested this program on simulated data, taking the best solution out of the greedy one and the heuristic one, and applying on it the local search for a fixed runtime limit of three minutes.

IV. RESULTS

Transfer rate and number of inferred transfers. We first tested the ability of the ALE method to infer a likely number of transfers, as well as the effect of inferring transfers in a phylogeny which is a small subtree of the one on which transfers have been simulated. On Figure 3, we can see that up to a very high transfer rate, which is far above the ones measured in biological datasets, the number of inferred transfers follows a regular function of the transfer rate. Measures of transfer numbers on biological datasets were done for comparison purposes from the cyanobacteria dataset from Szöllősi *et al* [18], and from the fungi dataset from Szöllősi *et al* [19]. They show that the range of the simulation contains the published biological conditions.

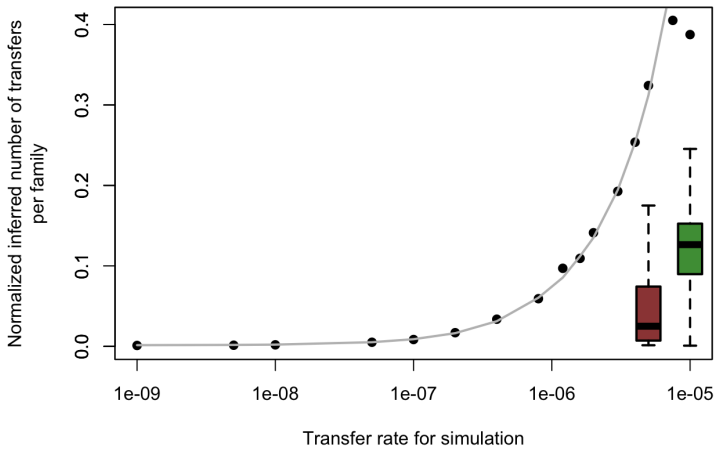

Figure 3: Mean number of inferred transfers (number of transfers per family and per branch of the species tree), as a function of the transfer rate in the simulation ($\log_{10}$ scale). Each point is one simulation of a species tree and 1000 gene trees with its own transfer rate. The right boxplots show the distribution of the number of inferred transfers on gene families from two published biological datasets: 28 *Fungi [19] (red) and* 36 *cyanobacteria [18] (green). For each gene family the number of inferred transfers per branch is computed. It shows that comparable numbers are found in simulated and biological datasets.*

Henceforth we use the mean number of inferred transfers as the reference instead of the transfer rate, to relate our measures to numbers comparable with what is found in biological dataset, for which we don't know the transfer rate on the complete phylogeny containing extinct and unsampled species.

Number of conflicting constraints. We then measured the fraction of constraints computed from transfers that has to be discarded to get a consistent set of constraints (Figure 4). We compare this value to the fraction of the constraints not compatible with the true (simulated) ranked species tree (red points). We see that the values on reconstructed node orders are close and always a bit under the true values. This justifies the minimizing approach: the true conflict is close to the minimum. However as the optimum is always lower than the true value, it also shows that discrepancies to the truth are not due to limitations in the optimization algorithm but to limitations in the model itself. The small difference is probably due to overfitting of artefactual constraints. Another lesson from this figure is that for biologically relevant transfer rates between 5% and 15% of constraints must be removed to get a consistent subset. As these constraints are necessarily deduced from false transfers, this places a lower bound on the rate of false positive transfers. It has already been observed that current transfer detection methods usually infer an accurate number of transfers but the precise identification of donors and receivers is more approximate [1].

Similarity between inferred and true ranked trees. We come to the main result, that is, the measure of the accuracy of the method to find the right ranking. We compare the true (simulated) ranked tree with the obtained ranked tree and compute the Kendall τ distance between different total orders. The Kendall τ distance between two orders is the number of pairs i, j of elements of the two orders such that i is before j in one order, and j is before i in another. We normalize this number by the maximum possible Kendall distance given that the

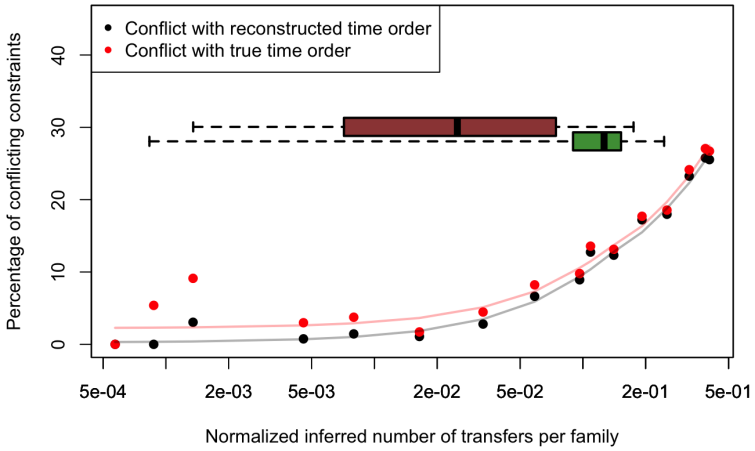

Figure 4: Fraction of constraints that have to be removed in order to get a consistent set, as a function of the mean number of inferred transfers ($\log_{10}$ scale). Red dots are for the fraction of constraints in conflict with the true (simulated) tree, and black dots are for the fraction of constraints in conflict with the reconstructed tree, minimizing the conflicts. Horizontal boxplots show the number of inferred transfers from two biological datasets: 28 *Fungi [19] (red) and* 36 *cyanobacteria [18] (green).*

two orders have to be derived from the species tree, to get a number between 0 and 1 (0 for the maximum distance between orders given a species tree, 1 for two equal orders). To compute the maximum Kendall distance between two linear extensions of a partial order determined by a tree we use the following property.

Property 1. *Given a rooted tree T inducing a partial order P on its internal nodes, two depth first searches of T, ordering the children of any node in, respectively lexicographical and anti-lexicographical order, output two linear extensions of P such that their Kendall distance is maximum, among all pairs of linear extensions of P.*

This property is easy to demonstrate: take any pair i, j of internal nodes of a rooted tree. Either one is the ancestor of the other and they appear in the same order in any pair of linear extensions. Or they are incomparable, with a last common ancestor a, having children a_1, the ancestor of i, and a_2, the ancestor of j. In one depth first search a_1 and its descendants, including i, appear before a_2 and its descendants, including j, and in the other it is the opposite. So all incomparable pairs appear in a different order, contributing to the Kendall distance. This obviously gives the maximum possible Kendall distance.

Sensitivity to the number of families. We give an idea of how many gene trees (and in consequence how many transfers) are necessary to get a good dating information. In Figure 5 (bottom), we plot the Kendall similarity between the true tree and the obtained tree, as a function of the number of gene trees, for a constant transfer rate of 1.6×10^{-6}, corresponding to approximately 5 inferred transfers per family (all families have approximately 100 genes).

We see that the method starts with a very low similarity if there are not enough gene trees, which is expected as in the absence of transfers there is no information to infer the ranked tree. Then the similarity rapidly increases, almost reaching a plateau at about 400 families, then slowly increasing up to 5000. This means that the more gene trees are available, the best the result will be, but with little gain after 1000 gene trees. On the top panel of the Figure 5, we

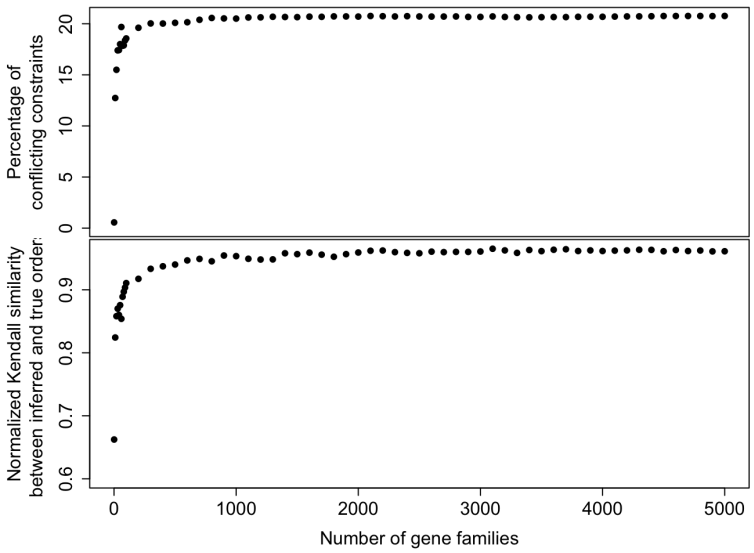

Figure 5: Top: Fraction of the constraints removed by MaxTiC to get a consistent set as a function of the number of gene trees. Bottom: Normalized Kendall similarity of the true ranked tree and the obtained ranked tree, as a function of the number of gene trees in the experiment.

see that the conflict (ratio of removed constraints to obtain a consistent set) also grows quickly and then stays remarkably stable. This shows that the rate of signal and conflict is relatively constant in all families.

Sensitivity to the transfer rate. We then investigated the effect of the transfer rate on the accuracy of the result. We measured the normalized Kendall similarity as a function of the average number of transfers per gene family. The results are shown on Figure 6. As expected, too few transfers give a low quality result, because of a lack of signal, and too many transfers make the similarity to the true node order decrease. However the slopes are very different: whereas a reasonable number of transfers are sufficient to give a good ranked tree, the ranked tree stays reasonably good even with a huge number of transfers (several dozens per family).

Note, however, that in any conditions, the normalized Kendall similarity to the true ranked tree remains bounded slightly above 95%, and under almost all conditions, it is between 90% and 95%. So it is possible, with ALE to detect transfers, to get a result close to the real order of speciations in a wide range of conditions, but the real order seems never to be found.

Note finally that the amount of conflict (that can be measured on real data) is not necessarily a good proxy for the similarity (that requires the knowledge of the true ranked tree): as shown by a comparison of Figures 2 and 6, the evolutions of the two has no evident correlation when the transfer rate increases.

Sensitivity to non modeled processes and uncertainties in the gene trees. We examine the effect of non modeled processes or gene tree uncertainties (Figure 7). In Simphy it is possible to vary the population size, and with the population size the probability of incomplete lineage sorting (ILS) increases. ALE does not model ILS, thus any statistically supported deviation from the species tree topology resulting from ILS will be interpreted as a series of DTL events. Indeed it can be seen on Figure 7 (middle) that for a same transfer rate, the number of inferred transfers increases with population size, thus with the amount of ILS. On The top panel of Figure 7 it can be seen that these supernumerary transfers are not time compatible as the minimum frequency of conflicting transfers increases also with population size. However on the bottom panel, we

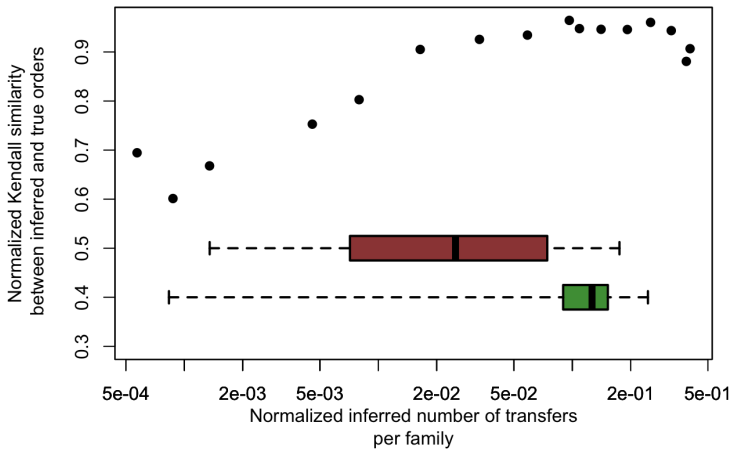

Figure 6: Normalized Kendall similarity of the true ranked tree and the obtained ranked tree, as a function of number of transfers, per branch and per family ($\log_{10}$ scale). The boxplots show the distribution of the number of inferred transfers on gene families from two published biological datasets: 28 Fungi [19] (red) and 36 cyanobacteria [18] (green). For each gene family the number of inferred transfers per branch is computed.

can see that nonetheless, and despite a decrease in the Kendall similarity with the true ranked tree, it is still possible to reasonably rank a species tree even in the presence of a high rate of false transfers due to ILS or phylogenetic uncertainty.

Sensitivity to uncertainties in the species tree. Finally, the topology of the species tree is in general not known with a high precision, so we tested the robustness of the method to errors in the species tree. We compared the normalized Kendall similarity of 5 simulations with the true species tree (blue dots in Figure 8) for a fixed transfer rate of 10^{-6}, with 5 simulations for 5 different conditions: re-rooting the species tree at a grand-child of the root (red dots), and respectively applying 5, 10, 15 and 20 random "nearest neighbor interchanges" (NNIs) in the species tree (green dots). We plot in Figure 8 the normalized Kendall similarity in function of the obtained Robinson-Foulds distance to the true tree (A certain number of random NNIs leads to a Robinson-Foulds distance of at most this number).

The tendency of Figure 8 shows a good robustness to errors in the species tree, showing that even with quite distant species trees, the rank of true clades is well preserved.

V. Conclusion

We give a proof of principle of a method to get a ranked species tree with the information provided by transfers. We present a method and a software, called MaxTiC for Maximum Time Consistency, taking an unranked species tree as input, together with a set of possibly conflicting weighted time constraints, and outputting a ranked tree maximizing the total weight of a compatible subset of constraints. We validate this principle for dating on simulations from an independent (developed by an independent team, with different aims) genome simulator Simphy, under conditions comparable to published biological datasets. The results confirm the principle of the possibility to date with transfers, under a wide range of conditions including uncertainties in gene trees and species trees. Thus we introduce an additional source of

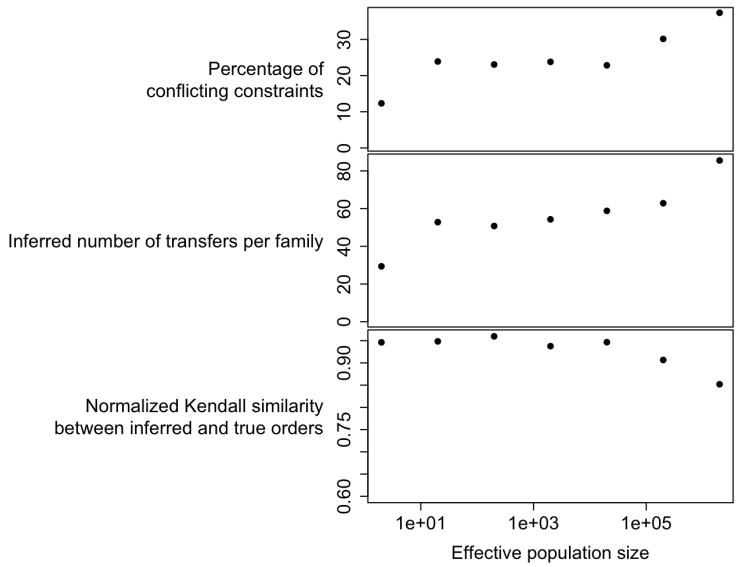

Figure 7: Minimum fraction of conflicting constraints, mean number of inferred transfers per family and normalized Kendall similarity of the true ranked tree and the obtained ranked tree, as three functions of population size ($\log_{10}$ scale), for a fixed transfer rate (10^{-6} in the simulation). Population size favors incomplete lineage sorting in SimPhy, as such it is used here to measure the effect of non modeled processes or as a proxy for errors in phylogenetic reconstruction.

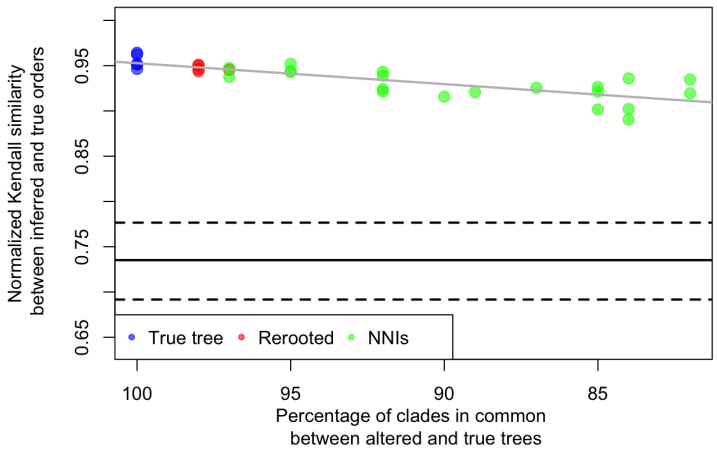

Figure 8: Normalized Kendall similarity of the true ranked tree and the obtained ranked tree, for different species trees in which errors have been introduced. The Kendall similarity is computed on the common clades, that is, on the fraction of true clades in the modified species tree. The three horizontal lines show the normalized Kendall similarity of a distribution of randomly generated ranked trees, independently from transfers. This shows that transfers give a robust dating information even in the presence of a highly uncertain species tree.

information compared to dated fossils and the (relaxed) molecular clock. It is all the more important since the fossil record is poor or difficult to interpret precisely in clades where

transfers are abundant.

VI. Acknowledgments

G.J.Sz. received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme under grant agreement No. 714774. This project was supported by the French Agence Nationale de la Recherche (ANR) through grant no. ANR-10-BINF-01–01 'Ancestrome'.

References

[1] Sophie S. Abby, Eric Tannier, Manolo Gouy, and Vincent Daubin. Lateral gene transfer as a support for the tree of life. *Proc Natl Acad Sci U S A*, 109(13):4962–4967, Mar 2012.

[2] Mukul S. Bansal, Eric J. Alm, and Manolis Kellis. Efficient algorithms for the reconciliation problem with gene duplication, horizontal transfer and loss. *Bioinformatics*, 28(12):i283–i291, Jun 2012.

[3] Priscila Biller, Carole Knibbe, Guillaume Beslon, and Eric Tannier. Comparative genomics on artificial life. In *Computability in Europe*, Lecture Notes in Computer Science, 2016.

[4] PCJ Donoghue and MP Smith, editors. *Telling the evolutionary time*. CRC press, 2003.

[5] Michael R. Garey and David S. Johnson. *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA, 1990.

[6] J P Gogarten, R D Murphey, and L Olendzenski. Horizontal gene transfer: pitfalls and promises. *The Biological bulletin*, 196:359–61; discussion 361–2, June 1999.

[7] Edwin Jacox, Cedric Chauve, Gergely J. Szöllősi, Yann Ponty, and Celine Scornavacca. ecceterа: comprehensive gene tree-species tree reconciliation using parsimony. *Bioinformatics*, Feb 2016.

[8] T. Leighton and S. Rao. An approximate max-flow min-cut theorem for uniform multicommodity flow problems with applications to approximation algorithms. In *Proc. 29th Annual Symp. Foundations of Computer Science*, pages 422–431, October 1988.

[9] Diego Mallo, Leonardo De Oliveira Martins, and David Posada. Simphy: Phylogenomic simulation of gene, locus, and species trees. *Syst Biol*, 65(2):334–344, Mar 2016.

[10] Matt Ravenhall, Nives Škunca, Florent Lassalle, and Christophe Dessimoz. Inferring horizontal gene transfer. *PLoS Comput Biol*, 11(5):e1004095, May 2015.

[11] C. Semple and M.A. Steel. *Phylogenetics*. Oxford lecture series in mathematics and its applications. Oxford University Press, 2003.

[12] Maureen Stolzer, Han Lai, Minli Xu, Deepa Sathaye, Benjamin Vernot, and Dannie Durand. Inferring duplications, losses, transfers and incomplete lineage sorting with nonbinary species trees. *Bioinformatics*, 28(18):i409–i415, Sep 2012.

[13] Gergely J. Szöllosi, Bastien Boussau, Sophie S. Abby, Eric Tannier, and Vincent Daubin. Phylogenetic modeling of lateral gene transfer reconstructs the pattern and relative timing of speciations. *Proc Natl Acad Sci U S A*, 109(43):17513–17518, Oct 2012.

[14] Gergely J. Szöllõsi, Wojciech Rosikiewicz, Bastien Boussau, Eric Tannier, and Vincent Daubin. Efficient exploration of the space of reconciled gene trees. *Syst Biol*, 62(6):901–912, Nov 2013.

[15] Gergely J. Szöllosi, Eric Tannier, Nicolas Lartillot, and Vincent Daubin. Lateral gene transfer from the dead. *Syst Biol*, 62(3):386–397, May 2013.

[16] Gergely J. Szöllősi, Adrián Arellano Davín, Eric Tannier, Vincent Daubin, and Bastien Boussau. Genome-scale phylogenetic analysis finds extensive gene transfer among fungi. *Philos Trans R Soc Lond B Biol Sci*, 370(1678):20140335, Sep 2015.

[17] Gergely J. Szöllősi, Eric Tannier, Vincent Daubin, and Bastien Boussau. The inference of gene trees with species trees. *Syst Biol*, 64(1):e42–e62, Jan 2015.

[18] Gergely J Szöllosi, Bastien Boussau, Sophie S Abby, Eric Tannier, and Vincent Daubin. Phylogenetic modeling of lateral gene transfer reconstructs the pattern and relative timing of speciations. *Proceedings of the National Academy of Sciences of the United States of America*, 109:17513–17518, October 2012.

[19] Gergely J Szöllősi, Adrián Arellano Davín, Eric Tannier, Vincent Daubin, and Bastien Boussau. Genome-scale phylogenetic analysis finds extensive gene transfer among fungi. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 370:20140335, September 2015.