Doc Start BIOINFORMATICS Vol. 00 no. 00 2015 Pages 1–10

ecceTERA: Comprehensive gene tree-species tree reconciliation using parsimony: supplementary material

Edwin Jacox[1], Cedric Chauve[2], Gergely J. Szöllősi[3], Yann Ponty[3,4,5], Celine Scornavacca[1,6*]

[1] ISE-M, Université Montpellier, CNRS, IRD, EPHE, Montpellier, France
[2] Department of Mathematics, Simon Fraser University, Burnaby, BC, Canada
[3] ELTE-MTA "Lendület" Biophysics Research Group, Budapest, Hungary
[4] Pacific Institute for the Mathematical Sciences, Vancouver, BC, Canada
[5] CNRS/Inria AMIB, Ecole Polytechnique, Palaiseau, France
[6] Institut de Biologie Computationnelle (IBC), Montpellier, France

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Associate Editor: XXXXXXX

S SUPPLEMENTARY MATERIAL

S.1 The ecceTERA model and dynamic programming algorithm.

The main methodological contribution of the present paper is a generic dynamic programming (DP) algorithm that allows the consideration of a comprehensive set of evolutionary events, as well as fully dated, partially dated or undated species trees, while ensuring time-consistency of the transfers regarding the provided speciation dates. The DP scheme underlying the `ecceTERA` algorithm is based on the equations described in Fig. S.4 and Fig. S.5.

Gene and species trees. We assume we are given a species tree S, with n leaves, and a gene tree G, with m leaves, both rooted and binary. For an internal node v of G or S, we denote by v_l, v_r, v_p and v_s respectively its left and right child, parent and sibling, where the order between sibling nodes is arbitrary. For a leaf u of G, corresponding to an extant gene, we denote by $s(u)$ the species whose genome contains this gene; similarly, for a leaf x of S, representing an extant species, we denote by $s(x)$ the corresponding species. For a vertex u of G, we denote by G_u the subtree of G rooted at u. For a vertex x of S, we denote by S_x the subtree of S rooted at x.

(Partially) dated species. We also assume that some internal nodes of S are dated, with dates increasing from the root to the leaves; without loss of generality, we also assume that no internal node can have the same date as any other node of S. For an internal dated node of S, the provided date corresponds to the date of the speciation associated with the corresponding ancestral species. The species tree S is fully dated if all its $n-1$ internal nodes are dated, partially dated if a strict subset of $k < n-1$ internal nodes are dated, and undated if no internal node is dated. Without loss of generality, we assume that the root of S is always dated, with date 0 and that, if $k \geq 1$ internal nodes of S are dated (including the root of S), the provided dates can be described by the integers from 0 to $k-1$. Finally we assume that all leaves, that represent extant species, receive the date k. For every dated node x of S, we denote its date by $t(x)$, an integer from $\{0, \ldots, k\}$; so if S is undated, $k = 1$ and the root r of S is the only internal dated node (with $t(r) = 0$) and all the leaves x of S have date $t(x) = 1$.

Moreover, we can associate with every non-root node x of S an *existence time-interval* $h(x)$ which represents the time interval during which the existence of x is consistent with the dated nodes of S. Namely, let x be a node of S other than the root, and let y be the first dated strict ancestor of x (i.e. $y \neq x$ even when x is dated) and z be the dated non-strict descendant of x with the lowest date (where non-strict means that we have $z = x$ if x is dated), $h_1(x) = t(y)$ and $h_2(x) = t(z)$. Then, if x is dated, one has $h(x) := (h_1(x), h_2(x)]$, and

*to whom correspondence should be addressed

$h(x) := (h_1(x), h_2(x))$ otherwise. Moreover, $h(root(S)) = [0, 0]$ and $h(x^d) = (0, k]$. Note also that, for any internal node x of S, one has $h_1(x_l) = h_1(x_r)$. For instance, for any internal node different from the root, $h(x) = (t(x_p), t(x)]$ if S is fully dated, and $h(x) = (0, 1]$ if S is undated.

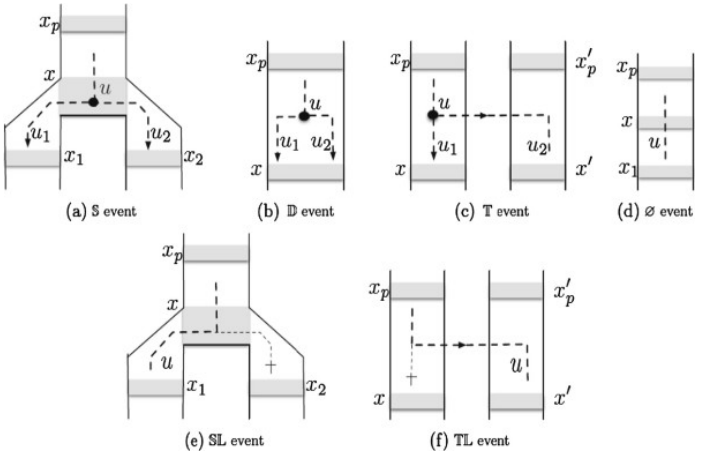

Fig. S.1: Illustration of all evolutionary events considered by the model of `ecceTERA`. Reprinted from Nguyen *et al.* (2013a).

The evolutionary model. The `ecceTERA` algorithm supports the evolutionary events of speciation (denoted by $\mathbb{S}$), gene duplication ($\mathbb{D}$), speciation-loss ($\mathbb{SL}$), horizontal gene transfer ($\mathbb{T}$) and transfer-loss ($\mathbb{TL}$), both between species present in the considered set of species and from/to unsampled/extinct species:

- Speciation and duplication events are self-explanatory;
- A speciation-loss is a speciation where the original gene is absent from one of the two species resulting from the speciation;
- A horizontal transfer corresponds to transferring the lineage of a child of a gene to another branch of the species tree (whose end node is called the *receiver of the transfer*), while the sibling lineage still evolves within the lineage of the parent;
- A transfer-loss is a horizontal transfer of one of the two descendants of a gene combined with the loss of its sibling lineage. Note that this event results in a single lineage leading to an extant gene. This implies that any algorithm to compute a parsimonious reconciliation that assumes that a transfer consists in one child of the current gene staying in its lineage and one child being transferred to another lineage cannot emulate a TL event since it implicitly assumes that a transfer results in two lineage toward extant genes. Currently, Mowgli and ecceTERA are the only parsimonious reconciliation algorithms that can model TL events (Doyon *et al.* 2011).
- A transfer or transfer-loss to (resp. from) an unsampled species is simply modeled as a transfer or transfer-loss to/from a node labeled x^d (more below).

Speciation-loss and transfer-loss events are called *unary events* as they do not result in the creation of two descendant genes, while speciations, duplications and transfers are called *binary events*. Note that each loss is coupled with either a speciation or a transfer event. Indeed, duplication-loss events – never parsimonious and leaving no trace in the data – are not taken into account. So, even if there is no specific loss event in our model, gene losses are considered in full generality through speciation-loss and transfer-loss. We refer the reader to (Ranwez *et al.* 2015) for a more precise description of this model and to Fig. S.1 for an illustration.

Transfers from the dead In phylogenetic studies, the evolution of molecular sequences is assumed to have taken place along the phylogeny traced by the ancestors of extant species. In the presence of lateral gene transfer, however, this may not be the case, because the species lineage from which a gene was transferred may have gone extinct or not have been sampled. As show in (Szöllősi *et al.* 2013), if the number of sampled species is small compared with the total number of existing species – which is almost always the case – the overwhelming majority of gene transfers involve speciation-to and evolution-along extinct or unsampled lineages, i.e. practically all transfers are "transfers from the dead". To model such transfer from extinct and unsampled species, we formally add an additional species lineage, represented by an extra branch from the root of S to a leaf denoted by x^d, which, is not considered as a child of the root. This branch can be seen as a representation of extinct and unsampled branches of the complete species phylogeny, formally reduced to a single branch. Transfers to this branch can occur at anytime from any branch of the species tree. This reflects the fact that the expected number of speciation events per branch of the species tree to extinct or unsampled linages is very large, i.e. that most linages that have existed have either gone extinct or have not been sampled. Fig. S.2 shows an example of a reconciliation where a transfer involving speciation-to and evolution-along extinct or unsampled lineages is parsimonious.

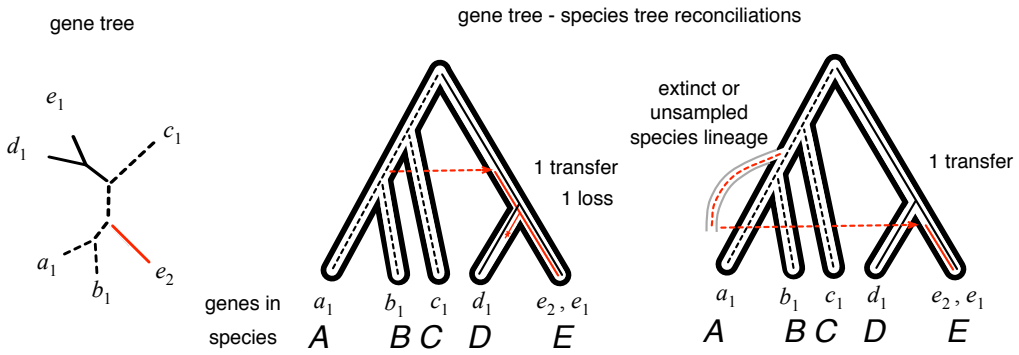

Fig. S.2: Illustration of "transfers from the dead". The reconciliation of gene tree in (left) involves one transfer and one loss event if "transfers from the dead" is not considered (center) because the the dating information does not allow a transfer directly to the extant species E; but a reconciliation only requiring one transfer event is possible if speciation-to and transfer-from extinct or unsampled lineages is considered (right). Species are indicated with upper case latin letters with corresponding genes in lower case, while the numbers adjacent to the internal nodes of the species tree correspond to dates.

Reconciliations of gene and species trees. A reconciliation α is a function that maps each node u of G onto an ordered sequence of nodes of S, denoted $\alpha(u) = (\alpha_1(u), \alpha_2(u), \ldots, \alpha_\ell(u))$. A reconciliation depicts an evolutionary history for a gene family with a given gene tree, evolving within a given species tree. Possible mappings are restricted by a few obvious conditions aimed at avoiding an inconsistent evolutionary history. More formally, for each pair of nodes u of G and $\alpha_i(u)$ of S (denoting $\alpha_i(u)$ by x below), α is said to be an *undated reconciliation* between G and S if and only if exactly one of the following events occurs for each pair of nodes u of G and $\alpha_i(u)$ of S (denoting $\alpha_i(u)$ by x below):

a) if x is the last node of $\alpha(u)$, one of the cases below is true:

1. $u \in L(G), x \in L(S)$ and $s(x) = s(u)$; (extant leaf)
2. $\{\alpha_1(u_l), \alpha_1(u_r)\} = \{x_l, x_r\}$; ($\mathbb{S}$ event)
3. $\alpha_1(u_l) = x$ and $\alpha_1(u_r) = x$; ($\mathbb{D}$ event)
4. $\alpha_1(u_l) = x$, and $\alpha_1(u_r)$ is any species node that is a not a descendant or ancestor of x;

or $\alpha_1(u_r) = x$, and $\alpha_1(u_l)$ is any species node that is a not a descendant or ancestor of x; ($\mathbb{T}$ event)

b) otherwise, one of the cases below is true:

5. $\alpha_{i+1}(u) \in \{x_l, x_r\}$; ($\mathbb{SL}$ event)
6. $\alpha_{i+1}(u)$ is any node that is a not a descendant or ancestor of x; ($\mathbb{TL}$ event)

Note that this definition is a simplified version of the definition of a reconciliation given in (Doyon *et al.* 2010), as our definition does not require the species tree to be dated. The notion of reconciliation is illustrated in Fig. S.3.

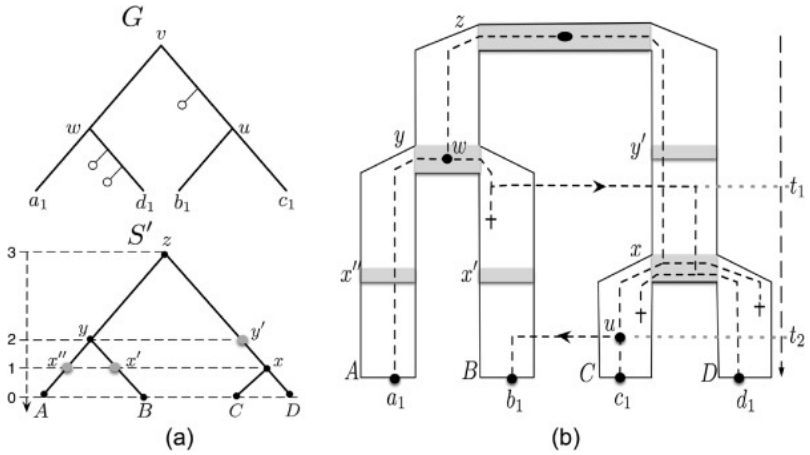

Fig. S.3: Illustration of the notion of reconciliation with a fully dated species tree. (a) A gene tree G and a fully dated species tree S. The branch to x^d is implicit and not represented on this tree. (b) A scenario including two speciation events (nodes w and z, two speciation-loss at node x, one transfer-loss from branch wB to the parent branch of x, at time t_1, and one transfer from node gene u to the branch wB at time t_2. Reprinted from Nguyen *et al.* (2013a).

We say that a reconciliation is *time-consistent* if, for each node u of G, we can associate a precise time t_e with each event in $\alpha(u)$, such that:

1. $t_e(\alpha_i(u)) \in h(\alpha_i(u))$;
2. $t_e(\alpha_i(u)) < t_e(\alpha_j(u)))$, for each $1 \leq i < j \leq |\alpha(u)|$;
3. if u is an internal node of G, $t_e(\alpha_{|\alpha(u)|}(u)) < t_e(\alpha_1(u_c))$, for each child u_c of u;
4. finally, in the case of a transfer (resp. transfer-loss) from a node $a := \alpha_{|\alpha(u)|}(u)$ (resp. $a := \alpha_i(u)$) to a node $b := \alpha_1(u_{r|l})$ (resp. $b := \alpha_{i+1}(u)$), then one must have $t_e(a) = t_e(b)$.

In other words, the evolution of a gene can only go forward in time. Time-consistency is difficult to achieve for species trees that are non-fully dated because of transfers (Tofigh *et al.* 2011). Indeed, transfers imply time constraints other than the dates given by the function t, and thus a dating t_e satisfying the conditions given above may not exist (necessary and sufficient conditions for the existence of an adequate t_e for undated species tree are given in (Stolzer *et al.* 2012; Donati *et al.* 2015)).

In the presence of a fully dated species tree, the above notion of time-consistency is equivalent to the notion of global time-consistency of transfers followed in the `Mowgli` software (Doyon *et al.* 2010), and is stronger

than the notion of local consistency followed in RANGER-DTL (Bansal *et al.* 2012) (note that global time-consistency of transfers can be ensured in polynomial time for fully dated species trees).

We associate a cost with each evolutionary event except for speciations: δ, τ, λ are respectively the user-defined costs associated with duplication, transfer and loss events (note that we define a speciation-loss (resp. transfer-loss) event to cost λ (resp. $\tau + \lambda$)). However, we consider that a transfer or transfer-loss to an extinct/unsampled lineage occurs at no cost (because it models a speciation involving an extinct/unsampled lineage), and duplications along the branch of x^d also occur at no cost (see Szöllősi *et al.* 2013, for more detail on this aspect of the model).

The algorithm and its analysis. We now describe the reconciliation algorithm underlying `ecceTERA`. For the sake of exposition we describe the Dynamic Programming algorithm to compute the cost of a parsimonious reconciliation. In order to compute an actual parsimonious reconciliation in the desired model, a standard backtracking is applied that traces back parsimonious events in the DP tables and thus maps every gene u onto a sequence $\alpha(u)$.

Note also that the DP algorithm introduced in this work is very close to the one used in `Mowgli` (Doyon *et al.* 2010) to compute a reconciliation with a fully dated species tree. The main differences are:

1. the introduction of x^d for the transfer to/from the dead and;
2. the formulation of the DP equations without the notion of subdivision, which is replaced here by the introduction of time-intervals, allowing us to also handle undated and partially dated species trees in a more natural and compact way.

The cost of a reconciliation is the sum of the costs of the events associated with its internal nodes. Here, we are interested in finding a reconciliation having minimum cost. To do so, for each node u from G, each node x from S and each date t from the interval $\{0 \dots, k+1\}$, the `ecceTERA` DP equations fill four main DP tables $c(u,x,t)$, $\bar{c}(u,x,t)$, $r(u,x,t)$ and $\bar{r}(u,x,t)$, which are defined as follows:

- $c(u,x,t)$ is the cost of a parsimonious reconciliation of the subfamily G_u with the species tree S_x, under the assumption that the reconciliation event associated with gene u took place at time t;
- $\bar{c}(u,x,t)$ is the cost of a parsimonious reconciliation of G_u with S_x, in which the gene u is not subject to a transfer-loss event, again under the assumption that the event associated with u took place at time t;
- $r(u,x,t)$ (resp. $\bar{r}(u,x,t)$) is the parsimony cost associated with a most parsimonious reconciliation of G_u *outside of* S_x in which gene u is associated with a transfer (resp. a transfer-loss) – so the gene u is kept (resp. lost) in the species x – and the transfer (resp. transfer-loss) occurs at time t to a species existing at this time and which is neither x nor a descendant or ancestor of x (*i.e.* is incomparable to x in terms of the partial order defined by S).

An important aspect of our algorithm is thus to associate a precise time interval with the evolutionary events of a reconciliation, taken among all the k time intervals defined by the k dated internal nodes of the species tree; in particular, in the case of an undated species tree, all events at the root occur at time $t = 0$, while all other events occur at time $t = k$.

In this framework, the cost of a reconciliation is $\min_{x \in V(S)} c(root(G), x)$ (Doyon *et al.* 2010, among others).

We describe in Fig. S.4 and Fig. S.5 the Dynamic Programming equations to fill in the four tables introduced above. In Fig. S.4 we present the main equations, using the additional notation $\theta_{y \neq z} := \begin{cases} 1 & \text{if } y = z, \\ 0 & \text{if } y \neq z. \end{cases}$

Before describing how the tables r and $\bar{r}$ are computed, we comment on the above DP equations:

- The first equation captures the basic fact that a gene u in a species x cannot be the object of an evolutionary event at time t if t is not in the existence time-interval of x;
- The second equation splits the search for a parsimonious reconciliation cost in two cases: (1) x is subject to a transfer-loss, (2) all other cases. The latter case is addressed in the third equation, where all other possible events are considered: speciation or speciation-loss if the current date t is the one at which the children of the species x appear, duplication (with no time constraint as no new species is considered),

If $t \notin h(x)$:

$$c(u,x,t), \bar{c}(u,x,t) := +\infty$$

Otherwise:

$$c(u,x,t) := \min \begin{cases} \bar{r}(u,x,t) + \lambda \times \theta_{x \neq x^d} & \{\mathbb{TL}\} \\ \bar{c}(u,x,t) & \{\textbf{no } \mathbb{TL} \textbf{ event}\} \end{cases}$$

$$\bar{c}(u,x,t) := \min \begin{cases} \left.\begin{array}{l} c(u_l,x_l,h_1(x_l)) + c(u_r,x_r,h_1(x_r)) \\ c(u_l,x_r,h_1(x_r)) + c(u_r,x_l,h_1(x_l)) \end{array}\right| & \text{if } \mathsf{Binary}(u) \wedge \mathsf{Binary}(x) \wedge t = h_1(x_l) = h_1(x_r) \quad \{\mathbb{S}\} \\ c(u_l,x,t) + c(u_r,x,t) + \delta \times \theta_{x \neq x^d} & \text{if } \mathsf{Binary}(u) \quad \{\mathbb{D}\} \\ \left.\begin{array}{l} c(u_l,x,t) + r(u_r,x,t) \\ r(u_l,x,t) + c(u_r,x,t) \end{array}\right| & \text{if } \mathsf{Binary}(u) \quad \{\mathbb{T}\} \\ c(u,x,t') & \text{where } t' \text{ is the smallest } t' > t \text{ such that } t' \in h(x) \ \{\textbf{no-event}\} \\ \left.\begin{array}{l} c(u,x_l,h_1(x_l)) + \lambda \\ c(u,x_r,h_1(x_r)) + \lambda \end{array}\right| & \text{if } \mathsf{Binary}(u) \wedge \mathsf{Binary}(x) \wedge t = h_1(x_l) = h_1(x_r) \quad \{\mathbb{SL}\} \\ 0 & \text{if } \mathsf{Leaf}(u) \wedge \mathsf{Leaf}(x) \wedge s(u) = s(x) \quad \{\textbf{extant}\} \\ +\infty & \text{otherwise} \quad \{\textbf{Invalid}\} \end{cases}$$

Fig. S.4: Dynamic-programming recurrences for computing the most-parsimonious reconciliation in the $\mathbb{DTL}$ model: part 1.

transfer – referring to the table r – absence of event (the current gene "ages" and we move to the next date in the current species existence time interval), and correct matching of genes/species at the leaves of the trees S and G;

- The introduction of the DP table $\bar{c}$ prevents two transfer-loss events from being consecutively applied to a given gene u. This restriction prevents infinite sequences of evolutionary events that would be caused by cycles of transfer-loss events;
- The equations used to fill the $\bar{c}$ table are very close to the ones of the algorithm of `Mowgli` that were developed for a fully dated species tree. The main difference is that, instead of using the notion of a time-slice, we rely on the existence time-intervals, which allow us – in a partially dated tree – to associate to each undated node an interval corresponding to several time slices, in an arguably more natural way;
- The no-event equation that allows us to let a gene evolve during a unitary time interval is then equivalent to a time-slice change in the model of Doyon *et al.* (2010);
- The constraint that a speciation or speciation-loss event can only take place at the earliest possible date at which the resulting gene existed (i.e. the beginning of the existence time-interval of the species they have been assigned to) ensures a date to all binary events that is consistent with the dates of the species tree.

The correctness of the algorithm then follows if $r(u,x,t)$ (resp. $\bar{r}(u,x,t)$) indeed records the parsimonious cost of a reconciliation with a transfer (resp. transfer-loss) of u from species x. So, in order complete the description of the algorithm and of its correctness, we need to describe how $r(u,x,t)$ and $\bar{r}(u,x,t)$ are computed, which we do in Fig. S.5, and to ensure that they indeed provide the cost of a parsimonious reconciliation associated respectively to a transfer or a transfer-loss of u.

In the equations of Fig. S.5, we introduce fours additional DP tables: $d^{\downarrow}, \bar{d}^{\downarrow}, d^{\uparrow}, \bar{d}^{\uparrow}$, all indexed as the tables c and r. Each cell of these tables contains the cost of a most parsimonious reconciliation associated respectively with a transfer (tables d) or a transfer-loss (tables $\bar{d}$) of u. The semantics associated to these tables are as follows:

- $d^{\uparrow}(u,x,t)$ (resp. $\bar{d}^{\uparrow}(u,x,t)$) records the cost of the best reconciliation of G_u with a subtree of S that belongs to $S - S_x$, the species tree obtained by pruning S_x from S (*i.e.* this subtree does not contain either x or one of its descendants) where u is subject to a transfer (resp. transfer-loss) – these tables are thus identical to r and $\bar{r}$, but we introduce them in order to clarify the tree exploration process of looking for the best receiver for a transfer (see discussion below);

$$r(u,x,t) = d^{\uparrow}(u,x,t) \qquad \bar{r}(u,x,t) = \bar{d}^{\uparrow}(u,x,t)$$

$$d^{\uparrow}(u,x,t) := \min \begin{cases} d^{\uparrow}(u,x_p,t) & \text{if } x \neq \mathsf{Root} \\ d^{\downarrow}(u,x_s,t) & \text{if } \mathsf{Binary}(x_p) \end{cases} \qquad \bar{d}^{\uparrow}(u,x,t) := \min \begin{cases} \bar{d}^{\uparrow}(u,x_p,t) & \text{if } x \neq \mathsf{Root} \\ \bar{d}^{\downarrow}(u,x_s,t) & \text{if } \mathsf{Binary}(x_p) \end{cases}$$

$$d^{\downarrow}(u,x,t) := \min \begin{cases} c(u,x,t) + \tau \times \theta_{x \neq x^d} & \text{if } t \in h(x) \\ d^{\downarrow}(u,x_l,t) & \text{if not } \mathsf{Leaf}(x) \\ d^{\downarrow}(u,x_r,t) & \text{if } \mathsf{Binary}(x) \end{cases} \qquad \bar{d}^{\downarrow}(u,x,t) := \min \begin{cases} \bar{c}(u,x,t) + \tau \times \theta_{x \neq x^d} & \text{if } t \in h(x) \\ \bar{d}^{\downarrow}(u,x_l,t) & \text{if not } \mathsf{Leaf}(x) \\ \bar{d}^{\downarrow}(u,x_r,t) & \text{if } \mathsf{Binary}(x) \end{cases}$$

Fig. S.5: Dynamic-programming recurrences for computing the most-parsimonious reconciliation in the $\mathbb{DTL}$ model: part 2.

- $d^{\downarrow}(u,x,t)$ (resp. $\bar{d}^{\uparrow}(u,x,t)$) records the cost of the two best reconciliations of G_x with a subtree of S rooted at a species that belongs to S_x where u is subject to a transfer (resp. transfer-loss).

Note that the different cases on the right-hand sides of the DP equations might not be exclusive: it is possible that both x is not the root of S and x_p is binary; this is not a problem as the operator $\min$ combines all values of the valid right-hand terms of the DP equations into a single set and selects the best one out of this set.

The proof of the DP equations within the four tables d in Fig. S.5 is straightforward. The base case corresponds to the case where $t \in h(x)$, in which case we rely on tables c and $\bar{c}$. Otherwise, for tables $d^{\uparrow}$ and $\bar{d}^{\uparrow}$ then we need to consider the scenarios associated with its parent x_p (i.e. the costs associated to the receivers in $S - S_{x_p}$) and to its sibling x_s (receivers in S_{x_s}), and the correctness of the semantics associated with the tables follows by induction. For tables $d^{\downarrow}$ and $\bar{d}^{\downarrow}$ we need to consider the receivers located in the subtree S_x, *i.e.* in the subtrees S_{x_l} and S_{x_r}. The computation of the tables $d^{\uparrow}$ and $\bar{d}^{\uparrow}$ can then be seen as based on a standard bottom-up backtracking process in a tree while the tables $d^{\downarrow}$ and $\bar{d}^{\downarrow}$ follow a recursive top-down exploration of a tree: to compute $d^{\uparrow}(u,x,t)$ for example, which finds a receiver for a transfer in $S - S_x$, one considers receivers in $S - S_{x_p}$ (bottom-up backtracking) and in S_{x_p} (top-down exploration).

The consistency of the time associated with evolutionary events follows from the fact that, whenever a recursive call considers a child x' of x, the date associated with the corresponding recursive call (*i.e.* right-hand member of the DP equation) is $h_1(x')$ which is the first possible date for an event associated with species x'. While this does not prevent that a cycle of events including speciations, duplications and transfers could create a cycle (and then violate global time-consistency) in the case of a non-fully dated species tree, it ensures a local time-consistency that generalizes the one enforced in the `RANGER-DTL-D` algorithm. In the case of a fully dated species tree, as every species is associated with a unitary existence time-interval and every speciation event implies a time progression, global time consistency is ensured.

The time complexity of the algorithm is given by the number of triples (u,x,t) to consider, as every DP equation fills in a DP table cell indexed by such a triple in constant time, assuming all members of the right-hand side of the equation have been computed. So the time and space complexity of the algorithm is in $O((k+1)|S||G|)$, which, in the case of a fully dated species tree is $O(|S|^2|G|)$ and in the case of an undated species tree in $O(|S||G|)$ thus matching the lowest complexities of existing implementations. Note that, to the best of our knowledge, the $O(|S|\log(|S|)|G|)$ time algorithm described in (Bansal *et al.* 2012) for fully dated species trees has not been implemented in `Ranger-DTL`.

Also note that, when computing an actual reconciliation, and not only its cost, the evolutionary event associated with a gene is given by the labels of the DP equations followed during the backtracking phase, and the time interval of an event is the interval $(t-1,t]$ defined uniquely by the third parameter t of the DP equations.

Amalgamation and supports. For the sake of readability, we presented only the reconciliation dynamic-programming recurrences, but they can be extended to consider gene tree clades of a set of gene trees $\mathcal{G}$ instead

of nodes of a single gene tree G – as done for example in Scornavacca *et al.* (2015) – to be able to amalgamate a set of gene trees for a given gene family, or to consider unrooted and non-binary gene trees to output a single binary rooted gene tree. Another extension implemented in `ecceTERA` is to consider nearly-parsimonious solutions (either via suboptimal reconciliations or optimal reconciliations under different sets of event costs). This aspect is motivated by the problem of computing the support of parsimonious evolutionary events (To *et al.* 2015) and the observation, in realistic simulations, that the most parsimonious reconciliation might not always be the true one (Doyon *et al.* 2012). Nearly-optimal reconciliation can be obtained by considering variations of the scoring scheme in a Pareto framework described in To *et al.* (2015). We also refer the reader to Section 4 of the `ecceTERA` manual.

Time-consistency. When the species tree is undated or not fully-dated, some reconciliations computed by `ecceTERA` may fail to be time-consistent. As with Notung (Stolzer *et al.* 2012) and EUCALYPT (Donati *et al.* 2015), our software can be used to output only reconciliations that are time-consistent, as discussed in (Donati *et al.* 2015)).

S.2 Experimental results.

We used three biological datasets to test the speed of our method:

1. The COG dataset, i.e. the biological dataset described in Bansal *et al.* (2012), consisting of around 4700 gene trees on 100 species sampled across the tree of life (but essentially prokaryotes).
2. The CYANO dataset, consisting of 1,099 simulated gene trees, from the 36 cyanobacteria present in version 5 of the HOGENOM database (Penel *et al.* 2009), which spans all fully sequenced organisms (bacteria, archea, and eukarya) in publicly available databases. These data were generated and analyzed in Szöllősi *et al.* (2013): topology, branch length and alignment lengths were taken from cyanobacteria trees obtained from real sequence data, and alignments of the extant genes were obtained by simulation. Only families with a number of genes between 10 and 150 were used (Szöllősi *et al.* 2013).
3. The HOGENOM dataset, contains 1,242 gene trees selected from all those available in release 6 of the HOGENOM database as follows: first, only gene trees with more than two leaves were retained. In this set, for each group of gene trees sharing the same number of leaves, only one tree was kept. The species tree on 1,460 taxa was obtained by applying the `SSMUL` tools (Scornavacca *et al.* 2011) to the HOGENOM database, and applying the SuperTriplets method (Ranwez *et al.* 2010) to the obtained gene trees.

In Tables S.1 and S.2, we report the running times for `RANGER-DTL-U`, `ecceTERA` and dated `ecceTERA`, respectively on the CYANO and COG datasets.

	mean	σ	median
RANGER-DTL-U	0	0	0
ecceTERA	0.002	0.004	0
dated ecceTERA	0.009	0.006	0.01

Table S.1. Running times (in seconds) on the CYANO data set for `RANGER-DTL-U`, `ecceTERA` and dated `ecceTERA`. We ran the experiments on a 3.40GHz $\times$ 8 Intel Core i7 processor with 8 GB of RAM.

	mean	σ	median
RANGER-DTL-U	0.001	0.003	0
ecceTERA	0.004	0.006	0
dated ecceTERA	0.039	0.066	0.01

Table S.2. Running times (in seconds) on the COG data set for `RANGER-DTL-U`, `ecceTERA` and dated `ecceTERA`. We ran the experiments on a 3.40GHz $\times$ 8 Intel Core i7 processor with 8 GB of RAM.

In Figure S.6, we report their running times and memory requirements (VmPeak values) for the HOGENOM dataset. From this figure, it is evident that RANGER-DTL-U needs more memory than ecceTERA on this data set, when the gene trees contain more than 15 taxa.

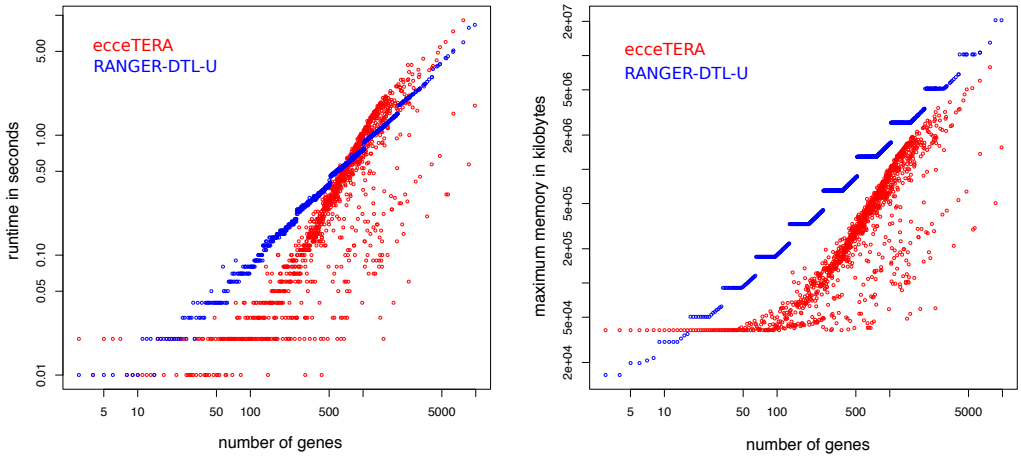

Fig. S.6: (Left) Running times (in seconds) on the HOGENOME data set for RANGER-DTL-U and ecceTERA (undated). (Right) VmPeak values for RANGER-DTL-U and ecceTERA on the HOGENOME dataset. Experiments were run on a 3.40GHz $\times$ 8 Intel Core i7 processor with 8 GB of RAM.

Figure S.7(a, b) shows an example of two reconciled gene trees for the same gene family within the CYANO dataset, and illustrates the impact of reconciliation algorithms based on a comprehensive evolutionary model. The reconciled tree depicted in (b) contains seven gene transfers, while the one illustrated in (a) contains only one transfer. Recently, genome scale sequence data from dozens of species have shown (Szöllősi *et al.* 2013) that 2 out of 3 transfers inferred by reconciliation algorithms are due to reconstruction errors; thus we consider the tree in (a) to be more likely. Note that the orthology relationships implied by the two trees are different. Indeed in the tree on the left, all genes, apart from the ones in *Acaryochloris marina* and *Trichodesmium erythraeum*, are pairwise orthologous, while in the tree depicted in (b), several pairs of genes have their lowest common ancestor involved in a gene transfer, and thus are not orthologous but *pseudoparalogous*.

A remark on time-consistency. When undated species trees are considered, since finding a time-consistent solution is NP-complete (Hallett *et al.* 2004; Ovadia *et al.* 2011; Tofigh *et al.* 2011), polynomial algorithms – such as those implemented in ecceTERA and RANGER-DTL-U – cannot guarantee time-consistent reconciliations. We ran ecceTERA using two undated models, one when direct transfers between all branches are allowed (M_1) and one where transfers to ancestral branches are not allowed (M_2, recall that this method is the one implemented in RANGER-DTL-U (Bansal *et al.* 2012)), on the HOGENOME data sets. We found that, under model M_1, 80% of the time, among the three reconciliations returned by ecceTERA (the symmetric and asymmetric medians and a random reconciliation (Nguyen *et al.* 2013b)), at least one is time-consistent; the percentage drops to 41% for model M_2. This shows that, when direct transfers to ancestral branches are forbidden, time-inconsistent solutions are more frequent.

REFERENCES

Bansal, M. S., Alm, E. J., and Kellis, M. (2012). Efficient algorithms for the reconciliation problem with gene duplication, horizontal transfer and loss. *Bioinformatics*, **28**(12), i283–91.

Chevenet, F., Doyon, J.-P., Scornavacca, C., Jacox, E., Jousselin, E., and Berry, V. (2016). Sylvx: a viewer for phylogenetic tree reconciliations. *Bioinformatics*, **32**(4), 608–610.

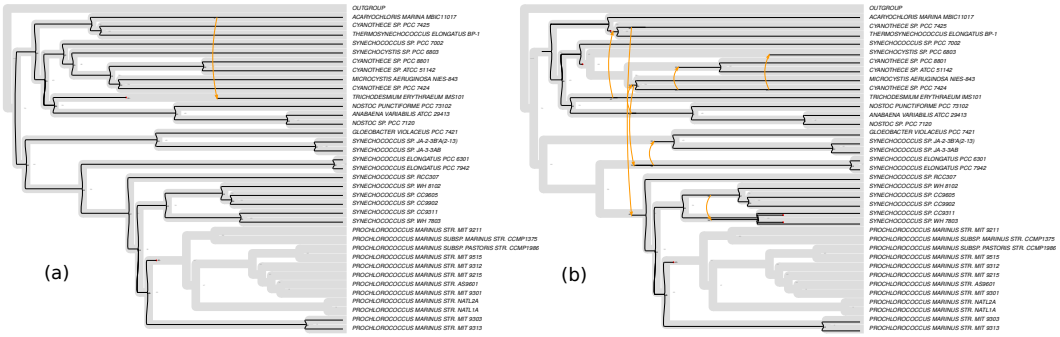

Fig. S.7: (a, b) Two reconciled gene trees, both obtained from the gene family HBG284208 of the HOGENOM database v 5 (Penel *et al.* 2009), drawn using SylvX (Chevenet *et al.* 2016). The tree illustrated in (b) has been obtained by PhyML and reconciled by `ecceTERA` using a simple model where the species tree is considered as undated and transfers to ancestral branches of the species phylogeny and transfer-losses are forbidden; the one depicted in (a) has been obtained by applying the method presented in (Scornavacca *et al.* 2015) and reconciled by `ecceTERA` using a more complex model, where the species tree is considered as dated and transfer-loss and transfer from/to extinct or unsampled species are allowed.

Donati, B., Baudet, C., Sinaimeri, B., Crescenzi, P., and Sagot, M. (2015). EUCALYPT: efficient tree reconciliation enumerator. *Algorithms for Molecular Biology*, **10**, 3.

Doyon, J., Scornavacca, C., Gorbunov, K. Y., Szöllosi, G. J., Ranwez, V., and Berry, V. (2010). An efficient algorithm for gene/species trees parsimonious reconciliation with losses, duplications and transfers. In *Comparative Genomics - International Workshop, RECOMB-CG 2010, Ottawa, Canada, October 9-11, 2010. Proceedings*, volume 6398 of *Lecture Notes in Computer Science*, pages 93–108. Springer.

Doyon, J., Ranwez, V., Daubin, V., and Berry, V. (2011). Models, algorithms and programs for phylogeny reconciliation. *Briefings in Bioinformatics*, **12**(5), 392–400.

Doyon, J., Hamel, S., and Chauve, C. (2012). An efficient method for exploring the space of gene tree/species tree reconciliations in a probabilistic framework. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, **9**(1), 26–39.

Hallett, M., Largergren, J., and Tofigh, A. (2004). Simultaneous identification of duplications and lateral transfers. In *Proceedings of the Eight International Conference on Research in Computational Molecular Biology (RECOMB)*, pages 347–356. ACM Press.

Nguyen, T. H., Ranwez, V., Pointet, S., Chifolleau, A.-M. A., Doyon, J.-P., and Berry, V. (2013a). Reconciliation and local gene tree rearrangement can be of mutual profit. *Algorithms for Molecular Biology*, **8**(1), 12.

Nguyen, T.-H., Ranwez, V., Berry, V., and Scornavacca, C. (2013b). Support measures to estimate the reliability of evolutionary events predicted by reconciliation methods. *PloS ONE*, **8**(10), e73667.

Ovadia, Y., Fielder, D., Conow, C., and Libeskind-Hadas, R. (2011). The cophylogeny reconstruction problem is np-complete. *Journal of Computational Biology*, **18**(1), 59–65.

Penel, S., Arigon, A.-M., Dufayard, J.-F., Sertier, A.-S., Daubin, V., Duret, L., Gouy, M., and Perrière, G. (2009). Databases of homologous gene families for comparative genomics. *BMC Bioinformatics*, **10 Suppl 6**(Suppl 6), S3.

Ranwez, V., Criscuolo, A., and Douzery, E. J. (2010). Supertriplets: a triplet-based supertree approach to phylogenomics. *Bioinformatics*, **26**(12), i115–i123.

Ranwez, V., Scornavacca, C., Doyon, J.-P., and Berry, V. (2015). Inferring gene duplications, transfers and losses can be done in a discrete framework. *Journal of mathematical biology*, pages 1–34.

Scornavacca, C., Berry, V., and Ranwez, V. (2011). Building species trees from larger parts of phylogenomic databases. *Information and Computation*, **209**(3), 590–605.

Scornavacca, C., Jacox, E., and Szöllosi, G. J. (2015). Joint amalgamation of most parsimonious reconciled gene trees. *Bioinformatics*, **31**(6), 841–848.

Stolzer, M., Lai, H., Xu, M., Sathaye, D., Vernot, B., and Durand, D. (2012). Inferring duplications, losses, transfers and incomplete lineage sorting with nonbinary species trees. *Bioinformatics*, **28**(18), i409–i415.

Szöllősi, G. J., Rosikiewicz, W., Boussau, B., Tannier, E., and Daubin, V. (2013). Efficient exploration of the space of reconciled gene trees. *Systematic Biology*, **62**(6), 901–912.

Szöllősi, G. J., Tannier, E., Lartillot, N., and Daubin, V. (2013). Lateral gene transfer from the dead. *Systematic Biology*, **62**(3), 386–397.

To, T.-H., Jacox, E., Ranwez, V., and Sconavacca, C. (2015). A fast method for calculating reliable event supports in tree reconciliations via Pareto optimality. *BMC Bioinformatics*, **16**(1), 384.

Tofigh, A., Hallett, M. T., and Lagergren, J. (2011). Simultaneous identification of duplications and lateral gene transfers. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, **8**(2), 517–535.