Extracting Few Representative Reconciliations with Host-Switches (Extended Abstract)

Mattia Gastaldello[(1,2)], Tiziana Calamoneri[(1)], Marie-France Sagot[(2)]

(1) Sapienza University of Rome, Computer Science Department, {gastaldello, calamo}@di.uniroma1.it
(2) Inria Grenoble - Rhône-Alpes & Université de Lyon, F-69000, Lyon; Université Lyon 1; CNRS, UMR5558, marie-france.sagot@inria.fr

Keywords: cophylogeny, reconciliations, equivalence relation.

Abstract. Phylogenetic tree reconciliation is the approach commonly used to investigate the coevolution of sets of organisms such as hosts and symbionts. Given a phylogenetic tree for each such set, respectively denoted by H and S, together with a mapping ϕ of the leaves of S to the leaves of H, a reconciliation is a mapping ϱ of the internal vertices of S to the vertices of H which extends ϕ with some constraints.

Given a cost for each reconciliation, a huge number of most parsimonious ones are possible, even exponential in the dimension of the trees. Without further information, any biological interpretation of the underlying coevolution would require that all optimal solutions are enumerated and examined. The latter is however impossible without providing some sort of high level view of the situation. One approach would be to extract a small number of representatives, based on some notion of similarity or of equivalence between the reconciliations.

In this paper, we define two equivalence relations that allow one to identify many reconciliations with a single one, thereby reducing their number. Extensive experiments indicate that the number of output solutions greatly decreases in general. By how much clearly depends on the constraints that are given as input.

1 Scientific Background

Given a directed binary tree T, we denote by $V(T)$ and $A(T)$ the set of its vertices and arcs, respectively. Given $v \in V(T)$, we denote by $p(v)$ its *parent* and by $s(v)$ its *sibling*.

Given two vertices $u, v \in V(T)$, u is an *ancestor* of v, denoted by $u \succeq_T v$, if either $u = v$ or there exists a directed path from u to v. If either $u \succeq_T v$ or $v \succeq_T u$, then we say that they are *comparable*. We say that u and v are *incomparable* if there is not a directed path between u and v.

If $u \succeq_T v$, we denote by $path_T(u, v) = (t_1, \ldots, t_j)$ the (unique) ordered sequence of vertices of T traversed along the directed path from u to v. Of course, $t_1 = u$ and $t_j = v$.

A *phylogenetic tree* T is a leaf-labelled rooted binary tree that models the evolution of a set of taxa (placed at the leaves) from their most recent common ancestor (placed at the root). The internal nodes of the tree correspond to the speciation events.

The model of host-symbiont evolution we rely on in this paper is the event-based one [1, 9]. Let H and S be the phylogenetic trees for the host and symbiont species, respectively. A function ϕ is defined from the leaves of S to the leaves of H that indicates the association between currently living host and symbiont species.

A *reconciliation* ϱ is a function from the set of internal vertices of S to the set of vertices of H that extends the mapping ϕ of the leaves under some constraints. Notice that each internal vertex of S can be associated to an event among: *cospeciation* (when both the parasite and the host speciate), *duplication* (when the parasite speciates but not

the host) and *host-switch* (when the parasite speciate and one of its children is associated to an incomparable host), while each arc (u, v) of S is associated to a certain number of loss events $l_{(u,v)} \geq 0$ that is equal to the length of $path_H(\varrho(u), \varrho(v))$ if $\varrho(u) \succeq_H \varrho(v)$. It is therefore possible to associate to each reconciliation ϱ a vector $E_\varrho = \langle e_c, e_d, e_s, e_l \rangle$ [2], that we call *event vector*, where e_c, e_d, e_s and e_l denote the number of cospeciations, duplications, host-switches and losses, respectively, that are in ϱ.

Given a vector $C = \langle c_c, c_d, c_s, c_l \rangle$ of real values that correspond to the cost of each type of event, the most parsimonious (or optimal) reconciliations are the ones that minimise the total cost, *i.e.* that minimise $cost(\varrho) = \sum_{i \in \{c,d,s,l\}} e_i \, c_i$.

We denote by $\mathcal{R}(H, S, \phi, C)$ the set of all optimal reconciliations from the tree S to the tree H whose leaves are connected by means of the mapping ϕ, and in which the costs of the events are given by C.

Phylogenetic tree reconciliation is the approach commonly used to investigate the coevolution of sets of organisms such as hosts and symbionts [6, 8].

However, a huge number of most parsimonious reconciliations are possible (see *e.g.* [4]). While any biological interpretation of the underlying coevolution would require that all optimal solutions are enumerated and examined, this is humanly unfeasible without providing some sort of high level view of the situation. One approach allowing this would be to extract a small number of representatives, based on some notion of similarity between reconciliations.

To the best of our knowledge, only a few such notions have been proposed in the literature. One of them is based on the comparison of the number of each one of the four events (cospeciation, duplication, loss and host-switch): two reconciliations are considered similar, and hence put in a same cluster, if they have the same number of each event, *i.e.* if they have the same event vector [2]. However, it is not difficult to find examples of very different reconciliations having the same number of each kind of event. Two of them are given in Figures 1.a and 1.b.

In [3], the authors define some operators which enable to go from one reconciliation to another, and from this provide a similarity measure between two reconciliations that is the smallest number of operations needed to change one reconciliation into another. Unfortunately, with this approach, it can happen that reconciliations that appear very similar have a rather high distance, as shown for example by Figures 1.c and 1.d. Moreover, the complexity of computing the similarity between reconciliations remains an open question, and there are thus no efficient algorithms for now.

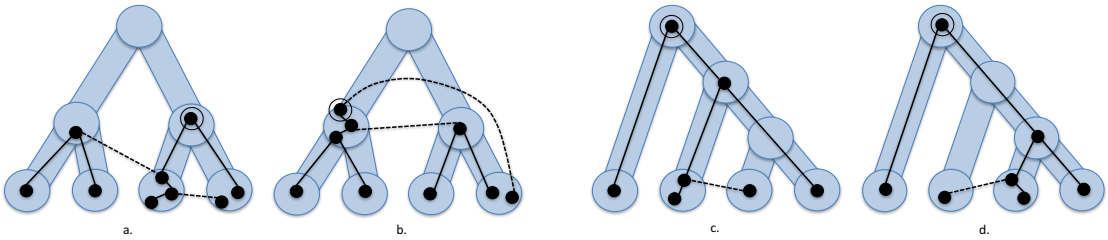

Figure 1: a. and b. Two reconciliations with the same event vector that nevertheless are rather different. The grey tubes represent the host tree, while the black (plain or dotted) lines inside the tubes represent the symbiont tree.
c. and d. Two reconciliations very similar with a possibly high distance (by adding arbitrarily many host vertices on the right path from the root) based on the operators.
The roots of the symbiont trees are double lined to facilitate their recognition.

In this work, we try to overcome the above problems by proposing, in Section 2, two equivalence relations that allow to identify many similar reconciliations with a single one, thereby substantially reducing the number of reconciliations that are enumerated.

In Section 3, we present some experimental results on real datasets which show that in most of the cases, these relations perform very well, especially when they are considered together. Finally, Section 4 proposes some future lines of research.

We call attention to the fact that in this extended abstract, due to lack of space, we will only give an intuition on which reconciliations we consider as equivalent and why, while we omit all technical details that make these results sound. These will be presented in the journal version of this paper.

2 Equivalent Reconciliations

2.1 Equivalence $\sim_1$

Given an optimal reconciliation $\varrho \in \mathcal{R}(H, S, \phi, C)$ and a vertex u of S such that arc (u, v) is mapped by ϱ as a host-switch, *i.e.* v is mapped to a vertex $\varrho(v)$ that is incomparable with $\varrho(u)$, we have that u can be mapped by ϱ to anyone of the vertices of $path_H(\varrho(p(u)), \varrho(s(v)))$ without changing the cost of ϱ, as proved by the following result.

Lemma 1. *Given any two reconciliations ϱ, σ, if:*
- *there exists an arc (u, v) mapped by both ϱ and σ as a host-switch, and*
- *$\varrho(w) = \sigma(w)$ for each $w \neq u$, and*
- *$\varrho(u) \neq \sigma(u)$ and $\varrho(u)$ and $\sigma(u)$ are mapped to two different vertices of $path_H(\varrho(p(u)), \varrho(s(v)))$, $\varrho(p(u)$ excluded*

then the costs associated to ϱ and σ are the same. In particular, ϱ will be optimal if and only if σ is.

The previous result leads us to consider as equivalent (using symbol $\sim_1$) all reconciliations that, for each host-switch (u, v), map u on a different vertex of $path_H(\varrho(p(u)), \varrho(s(v)))$. We call the latter a *sliding path* to highlight the idea that u can be moved anywhere inside this path without modifying the cost of the reconciliation.

The following result claims an interesting property of equivalent reconciliations w.r.t. relation $\sim_1$.

Theorem 1. *Given two reconciliations $\varrho, \sigma \in \mathcal{R}(H, S, \phi, C)$, if $\varrho \sim_1 \sigma$, then they have the same event vector, i.e. $E_\varrho = E_\sigma$.*

Observe that from the previous lemma, it follows that the partition of $\mathcal{R}(H, S, \phi, C)$ induced by $\sim_1$ is finer than the partition induced by the event vector, since two reconciliations that are equivalent w.r.t. $\sim_1$ are surely equivalent w.r.t. to the event vector partition, but the opposite is not true, and this is in agreement with the fact that two reconciliations with the same event vector can be very different: in such a case, our equivalence distinguishes them.

2.2 Equivalence $\sim_2$

We now propose a second equivalence relation between optimal reconciliations. This one is motivated by the following observation. Assume there are two siblings v and w in S that are mapped by ϕ on two incomparable vertices $\phi(v)$ and $\phi(w)$ in H. If host-switches are allowed, any reconciliation can equivalently map $p = p(v) = p(w)$ on a vertex that is either comparable with $\phi(v)$ and incomparable with $\phi(w)$ or vice-versa. All these solutions are equally feasible, and there is no reason to distinguish them. We can better explain this concept on the basis of the following result.

Lemma 2. *Given a reconciliation $\varrho \in \mathcal{R}(H, S, \phi, C)$ with $c_l > 0$, for each arc (u, v) mapped by ϱ as a host-switch s.t. $\varrho(u)$ and $\varrho(p(u))$ are incomparable, $\varrho(u) = \varrho(s(v))$.*

Given optimal reconciliations in which there are two adjacent vertices u and v of S (w.l.o.g. assume $u = p(v)$) that are both associated to a host-switch event, the previous result leads us to consider as equivalent (using symbol $\sim_2$) the reconciliations that map v to anyone of the vertices of H where its children are mapped. Figure 2.a illustrates this concept.

More formally, we have the following:

Theorem 2. *Given any two reconciliations ϱ, σ, if in both ϱ and σ:*
- *there exists a vertex v such that the mappings of v and of one of its children, let it be w, are incomparable, while v and $s(w)$ have the same mappings, and*
- *its parent $u = p(v)$ is such that its mapping and the one of one of its children (either v or $s(v)$) are incomparable, and*
- *$\varrho(t) = \sigma(t)$ for each $t \neq v$,*

then the costs associated to ϱ and σ are the same. In particular, ϱ will be optimal if and only if σ is.

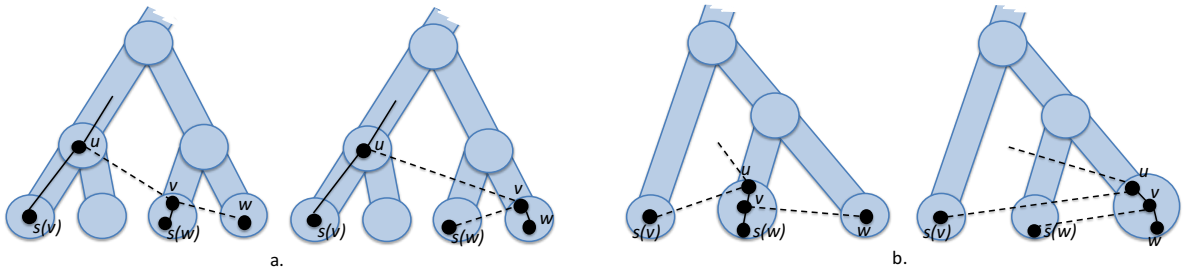

Figure 2: The two cases in which equivalence $\sim_2$ can be applied, focusing on vertex v.

Observe that, if $u = p(v)$ is incomparable with $s(v)$ (hence we are in the context of Figure 2.b), then if the two reconciliations are optimal, if $c_l > 0$ and $c_c \leq c_d$, either the arc $(u, p(u))$ is mapped as a host-switch while arc $(p(p(u)), p(u))$ is not mapped as a host-switch by ϱ or σ, or there must be an ancestor of u (and thus of v), let us denote it by x, such that the following is verified:
- arc $(p(x), x)$ is mapped as a host-switch by both ϱ and σ, and
- arc $(p(p(x)), p(x))$ is not mapped as a host-switch by both ϱ and σ (we reach the end of the ancestry recursion), and
- all the vertices y in the path $path_S(x, u)$ are such that:
  - they are mapped to the same host vertex as v, and
  - their child that is not in the path, let us denote it by z, is such that the arc (y, z) is mapped as a host-switch by both ϱ and σ.

3 Results

We now show the results of some experiments performed on real datasets.

To compute the numbers of $\sim_1$ and $\sim_2$ equivalence classes, we modified the code of a well known algorithm enumerating reconciliations, *i.e.* EUCALYPT. It works by computing a matrix by means of dynamic programming, and then exploiting it to enumerate or count all reconciliations in polynomial delay. For both equivalence classes, we operated only on the first part producing a different matrix in order to output or count one and only one reconciliation per class. Our modification therefore does not affect the computational time.

As concerns the first equivalence relation, we output for each class what can be considered as a canonical representative since the produced reconciliations have some identifying properties. On the contrary, for the second equivalence, we limit ourselves to count the number of classes without enumerating them.

We selected 13 datasets which correspond to those also used in [4] and that are indicated in that paper as GL, RH, FD, COG2085, COG3715, COG4964, COG4965, PP, SFC, EC, PMP, PML, and *Wolbachia*. The latter is a dataset of our own which corresponds to arthropod hosts and a bacterium genus, *Wolbachia*, living inside the cells of their hosts. It represents a larger set (each tree has 397 leaves) than the others that were taken from the literature and where the number of leaves varies between 13 to 100. We performed the experiments using the most commonly used cost vectors, namely (0, 1, 1, 1), (0, 1, 2, 1), and (0, 2, 3, 1) which correspond also to those presented in [4].

In all the tables, *# solutions* indicates the number of all optimal reconciliations, while $\#\sim_1$, $\#\sim_2$ and $\#\sim_2+\sim_1$ indicate the number of equivalence classes when relations $\sim_1$, $\sim_2$ or both are applied; the last column, called *NMR*, indicates the value of the Normalized Magnitude Reduction, rounded to two digits after the decimal point, which is given by $\frac{\log(\#sol)-\log(\#\sim_1+\sim_2))}{\log(\#sol)}$. Such value is one when all optimal solutions are reduced to a single parsimonious reconciliation when applying the two equivalences. Inversely, the closer this value is to zero, the less the two equivalences were able to reduce by similarity the number of solutions.

Observe that for *Wolbachia*, the number of solutions is so huge that, for space reason, we rounded the number to fit the table.

Table 1: Results for cost vector $(0, 1, 1, 1)$.

Dataset	# solutions	$\#\sim_1$	$\#\sim_2$	$\#\sim_2+\sim_1$	NMR
GL	2	2	2	2	0
RH	42	42	8	8	0,44
FD	25184	22752	224	180	0,49
COG2085	44544	36224	11	4	0,87
COG3715	1172598	777030	1888	872	0,52
COG4964	224	224	2	2	0,87
COG4965	17408	17408	4	4	0,86
PP	5120	4480	344	280	0,34
SFC	184	160	16	10	0,56
EC	16	16	13	13	0,07
PMP	2	2	1	1	1
PML	180	160	33	21	0,41
Wolbachia	$\sim 3.19 \cdot 10^{48}$	$\sim 5.72 \cdot 10^{47}$	$\sim 9.33 \cdot 10^{5}$	$\sim 7.68 \cdot 10^{4}$	0,90

Table 2: Results for cost vector $(0, 1, 2, 1)$.

Dataset	# solutions	$\#\sim_1$	$\#\sim_2$	$\#\sim_2+\sim_1$	NMR
GL	2	2	2	2	0
RH	2208	368	1608	268	0,27
FD	408	180	48	20	0,50
COG2085	37568	3200	226	14	0,75
COG3715	9	7	4	2	0,68
COG4964	36	4	9	1	1
COG4965	640	576	4	3	0,83
PP	72	72	36	36	0,16
SFC	40	16	10	4	0,62
EC	18	18	18	18	0
PMP	2	2	1	1	1
PML	2	2	1	1	1
Wolbachia	$\sim 1.01 \cdot 10^{47}$	$\sim 3.77 \cdot 10^{44}$	$\sim 2.92 \cdot 10^{8}$	$\sim 2.42 \cdot 10^{4}$	0,91

We now briefly comment the results presented in Tables 1 to 3. More detailed analyses will be provided in the journal version of this paper.

First, note that it is not surprising that in the case of the cost vector $(0, 1, 1, 1)$, there are on average more optimal solutions than with the other cost vectors. This is due

Table 3: Results for cost vector $(0, 2, 3, 1)$.

Dataset	# solutions	# $\sim_1$	# $\sim_2$	# $\sim_2$ + $\sim_1$	NMR
GL	2	2	2	2	0
RH	288	48	288	48	0,32
FD	80	16	10	2	0,84
COG2085	46656	1344	540	10	0,79
COG3715	33	2	33	2	0,80
COG4964	54	6	18	2	0,83
COG4965	6528	448	94	5	0,82
PP	72	72	36	36	0,16
SFC	40	16	10	4	0,62
EC	16	16	16	16	0
PMP	18	18	10	10	0,20
PML	11	6	7	4	0,42
Wolbachia	$\sim 4.08 \cdot 10^{42}$	$\sim 1.33 \cdot 10^{36}$	$\sim 4.18 \cdot 10^{10}$	$\sim 1.15 \cdot 10^{3}$	0,93

to the fact that the events that are different from cospeciation are indistinguishable in terms of cost, and this freedom of choice offers many alternatives for reaching a most parsimonious solution.

Given that both equivalence relations are primarily based on host-switch mappings, we would then expect that the higher is the number of host switches, the greater would be the chance of having a lower number of equivalence classes w.r.t. the total number of solutions. Equivalence $\sim_2$ depends further on the relative position of such host switches, that is, on whether the vertices involved in a host switch are ancestors of one another, and on how long is such ancestor path in H. It is better if such paths are very long rather than if they are frequent, as there is then more chance that each long one will lead to a collapse of many solutions into a single class.

Comparing the three tables, we observe that when the cost of a host-switch event is close to the cost of a loss, there is in general a smaller reduction of the number of optimal reconciliations when we pass to the $\sim_1$ equivalence classes. Intuitively, this is indeed because long sliding paths are more uncommon in this case. Inversely, the highest reductions from the total number of optimal solutions to the number of $\sim_1$ equivalence classes in general occur when the cost vectors are $(0, 1, 2, 1)$ or $(0, 2, 3, 1)$, *i.e.* when the cost of the host-switch event is higher w.r.t. the cost of a loss. In the other situations with many host switches (due either to the cost vector – *e.g.* $(0, 1, 1, 1)$ - or to the leaf-mapping, spreading close symbiont leaves to far host leaves – *e.g.* the dataset $COG2085$), equivalence $\sim_2$ performs better.

4 Perspectives

While the two equivalence relations introduced in this paper in general lead to very good results in terms of the overall goal of providing a more compact view of the solution space, we believe there are more such relations that could be explored in future. Alternatively, it is already possible – when less solutions are desired – to apply further clustering techniques based on a measure of similarity, or of distance among the equivalence classes identified in this paper as their number has become now much more reasonable even for large trees. One such approach has already been implemented and will be presented in the journal version of this paper.

Acknowledgments

This work has been partially supported by *Italian-French University* (Project "Algorithms and Models for Solving Complex Problems in Biology") and *Sapienza University of Rome* (Project "Combinatorial structures and Algorithms for Co-Phylogeny Problems"). The tables are filled also with the aids of the computing facilities of the CC

LBBE/PRABI.

References

[1] Bansal MS, Alm E, Kellis M. Efficient algorithms for the reconciliation problem with gene duplication, horizontal transfer and loss. Bioinformatics 2012;28(12):i283-91.

[2] Baudet C, Donati B, Sinaimeri B, Crescenzi P, Gautier C, Matias C, Sagot MF: Cophylogeny Reconstruction via an Approximate Bayesian Computation. *Syst. Biol.*, 64(3):416–431, 2015.

[3] Chan Yb, Ranwez V, Scornavacca C: Exploring the space of gene/species reconciliations with transfers. *J. Math. Biol.*, 71:1179–1209, 2015.

[4] Donati D, Baudet C, Sinaimeri B, Crescenzi P, Sagot MF. EUCALYPT: Efficient tree reconciliation enumerator. *Algorithms for Molecular Biology*, 10:3, 2015.

[5] Doyon JP, Ranwez V, Daubin V, Berry V. Models, algorithms and programs for phylogeny reconciliation. *Brief Bioinform.*, 12(5):392–400, 2011.

[6] Page RD, Charleston MA. Trees within trees: phylogeny and historical associations. *Trends Ecol Evol*,13(9):356–9, 1998.

[7] Merkle D, Middendorf M. Reconstruction of the cophylogenetic history of related phylogenetic trees with divergence timing information. *Theor Biosci*, 123(4):277–299, 2005.

[8] Charleston MA. Jungles: A new solution to the host/parasite phylogeny reconciliation problem. *Math Biosci*, 149(2):191–223, 1998.

[9] Tofigh A, Hallett M, Lagergren J. Simultaneous identification of duplications and lateral gene transfers. *IEEE/ACM Trans Comput Biol Bioinf*, 8(2):517–35, 2011.