ecceTERA: Tree Estimating using Reconciliation and Amalgamation
version 1.2.4, 14/09/2016

Given *a species tree* **S**, *a set of gene trees* **G**, and *a set of costs for gene events* (duplications, transfers and losses), ecceTERA can compute:

(1) for each tree G of **G**, the MPR score of a most parsimonious reconciliation (MPR) between S and G (if G is unrooted/non-binary, a rooted binary resolution of G is computed beforehand).
(2) the MPR score for the "amalgamation" of all the gene trees in **G** (considered as unrooted), as well as an optimal gene tree according to the combinatorial model defined in [2].

ecceTERA can also compute a reconciliation graph describing all reconciliations according to the combinatorial model defined in [1], as well as the median reconciliations, defined as representative reconciliations with the most common events (see [4] for details), and a random reconciliation (each reconciliation is printed along with its score and event supports [4]). More details are given below.

The basic parameters are described in Section 1. The only required parameters are the species tree (species.file) and the gene tree (gene.file) in Newick format.

The default parameters assume an ultrametric dated (i.e., with branch lengths), binary species tree. Alternatively, an explicit ordering of species tree nodes can be given in the bootstrap fields (dates.as.bootstrap=true). A partially dated species tree is input by specifying dated=1 and providing partial dates in the bootstrap fields. An undated tree is input using dated=0. For undated or partially dated species tree, only transfers to non-descendant/non-ancestral nodes are allowed, which aims to prevent transfers between nodes whose dates are incompatible. Note that the solutions for undated or partially dated species tree are not guaranteed to be consistent. Set check.time.consistency or check.time.consistency.all to true to check if the solutions are consistent. To summarize, the species tree must be binary and one of the following:
- dated, branch lengths given (ultrametric): dated=2 (default)
- dated (dates given in the bootstrap field): dated=2 dates.as.bootstrap=true,
- partially dated (dates given in the bootstrap field): dated=1,
- undated: dated=0

The gene tree file can contain multiple gene trees, which are processed individually unless the trees are amalgamated (see Section 3). If a gene tree is unrooted, it will either be rooted randomly or with an optimal rooting if resolve.trees=1 [manuscript in preparation]. Similarly, non-binary gene trees will be made binary by choosing a random resolution per each polytomy unless resolve.trees=0 or resolve.trees=1 in which case all possible binary resolutions of the gene tree will be considered in an efficient manner [manuscript in preparation].

The parameters specific to the reconciliation graph and the "amalgamation" are described in Sections 2 and 3, respectively. By default, reconciliations are not printed. Specifying print.reconciliations=1 will build the graph and print the median reconciliations, along with a random optimal reconciliation. The gene trees will be amalgamated if amalgamate=true. Note that reconciliations are only defined for rooted, binary gene trees. Thus, if amalgamate=false and the input gene tree is either unrooted or non-binary, first, a rooted, binary resolution of the gene tree (the best one if resolve.trees=0 or 1 and a random one otherwise) is computed. This new gene tree will be used to compute the reconciliations. If the trees are amalgamated (amalgamate=true), the resulting optimal gene tree [2] (that is always rooted and binary) will be used to compute the reconciliations.

Section 4 describes the parameters used to find Pareto-optimal solutions as described in [5].

Section 5 describes our format for a printed reconciliation. The reconciliation can be printed in the SylvX format [7] (software available at http://www.sylvx.org) by specifying sylvx.reconciliation=true. Alternatively, the scripts folder contains a script (recon2xml.pl) to convert an ecceTERA reconciliation into the PhyloXML format [6]. This script can be run from the command-line as 'recon2xml.pl <reconciliation_file>'. Run 'recon2.xml.pl -h' for more details.

Section 6 describes the parameters used to find reconciliations with respect to phylogenetic networks as described in [8-9].

1. RECONCILIATION PARAMETERS

The ecceTERA program can be used to compute the score of a most parsimonious reconciliation between a species tree and a gene tree – given a set of costs for gene duplications, horizontal transfers, and losses (DTL costs). The species file and gene file are the only required arguments. All arguments are specified as <name>=<value>. **NOTE: There must be no spaces between the parameter name (e.g. species.file), the equal sign and the value (e.g. tests/Stree.tree)**. Boolean values are given as true/false (or 1/0).

As an example, to run ecceTERA with default options on the test files, use:
bin/ecceTERA species.file=tests/Stree.tree gene.file=tests/HBG284008_1.gtrees

INPUT PARAMETERS:

species.file   [path, required]
> The path to the species file in Newick format. The tree must be binary. The tree can either be undated (dated=0), fully dated (dated=2) or partially dated (dated=1). If the tree is fully dated, there are two ways to specify its dates: either the species tree has branch lengths and is ultrametric (note that branch lengths are given after the a colon; see the tree in tests/Stree.tree for an example), in which case an ordering of species nodes will be deduced from the branch lengths, or an ordering of species nodes can be given directly in the bootstrap fields (dates.as.bootstrap=true, see the tree in tests/StreeBS.tree for an example). In case the tree is partially dated, only the latter option is possible.

gene.file   [path, required]
> The path to a gene file in Newick format. This file can contain more than one gene tree. A reconciliation with the species tree will be performed on each gene tree, unless the gene trees are amalgamated (amalgate=true, see Section 3). If a gene tree is unrooted, it will either be rooted randomly or with an optimal rooting if resolve.trees=1 [manuscript in preparation]. Similarly non-binary gene trees will be binary by choosing a random resolution per each polytomy unless resolve.trees=0 or 1, in which case all possible binary resolutions will be considered in an efficient manner [manuscript in preparation].

char.sep   [char] default = _
> The character used as separator for gene names in the gene trees. For example, if char.sep=_ is given as an option, then the gene HOMO_2 corresponds to the taxa HOMO.

parameter.file   [path] default=none
> If a valid file is given, parameters will also be read from the file in the same format <name>=<value>. Any parameters given on the command line will take precedence and the duplicated parameter will be ignored.

gene.mapping.file   [path] default = none
> A file with mapping between gene names and species names using a space as a separator (i.e., each line is <gene_name> <species_name>. Otherwise, gene names must start with a valid species name followed by the separator char.sep.

BASIC COSTS:

dupli.cost   [double] default = 2
> The cost of a gene duplication.

HGT.cost   [double] default = 3
> The cost of a horizontal gene transfer (HGT).

loss.cost   [double] default = 1
> The cost of a gene loss.

OPTIONS:

collapse.mode   [int] default = -1
  If given a value of 0 or 1, gene tree nodes whose values are strictly below collapse.threshold are suppressed to create polytomies. The values are either the distances to the father node (collapse.mode=0) or bootstrap values (collapse.mode=1). These polytomies will be resolved depending on the value of the option resolve.trees (see below). The default value of -1 indicates that no collapsing should be done.

collapse.threshold   [double] default = 0.5
  Threshold used to collapse gene tree nodes if collapse.mode is 0 or 1.

compute.T   [bool] default = true
  Consider transfers as a possible event. If false, transfers and transfer losses are not considered.

compute.TD   [bool] default = true
  If true, horizontal gene transfers are allowed with species other than those in the species tree (thus extinct or unsampled species) [3].

compute.TL   [bool] default = true
  Consider transfer loss as a possible event.

consistency.limit   [int] default = 10000
  The maximum number of reconciliations to check if check.time.consistency.all=true (undated and partially dated species trees only).

dates.as.bootstrap   [bool] default = false
  If true, the species node ordering is read directly from the bootstrap fields (see the tree in tests/StreeBS.tree for an example), the root having the highest value. Leaves are considered to have a value of 0.

dated   [int] default = 2
  If dated=0, the species tree is considered to be undated. If 1, the species tree is considered to be partially dated. Note that the dates must be given in the bootstrap fields (thus dates.as.bootstrap is automatically set to true). If 2 (default), the species tree is considered to be fully dated and thus it either has branch lengths and is ultrametric (and in this case an ordering of species nodes is deduced from the branch lengths), or fully ordered (and in this case dates.as.bootstrap has to be set as true).

degree.limit   [int] default = 12
  Maximum out degree for the collapsed polytomic trees (0=no limit).

gene.origination.species.root [bool] default = false
  Force the reconciliation to originate at the species root.

resolve.trees   [int] default = -1
  If resolve.trees=0, the algorithm will consider all possible resolutions of non-binary nodes (polytomies), as described in [10]. If resolve.trees=1, in addition to resolving all non-binary nodes, the algorithm will consider the input gene tree as unrooted and search for the best possible rooting [manuscript in preparation]. For the default value (-1), polytomies will not be resolved and the gene tree must be binary.

trim.species.tree   [bool] default = false
  If true, unneeded nodes of the species tree (i.e. nodes that will never be considered in an optimal reconciliation) will be removed before calculating the matrix. Trimming the species tree can improve the performance of the matrix calculation.

ultrametric.only   [bool] default = true
  Return an error if the dated species tree is not ultrametric.

use.bootstrap.weighting   [bool] default = false
  If true, polytomies are resolved using bootstrap values to weigh splits, as described in Section 3.1 of [10]. Default is true if collapse.mode is set to 1.

OUTPUT OPTIONS:

check.time.consistency   [bool] default = false
> If print.reconciliations=1, check whether the median and random reconciliations are time consistent and print this information to the screen (undated and partially dated species trees only). Median and random reconciliations are printed, regardless whether they are time consistent or not. If print.reconciliations=2, only time consistent reconciliations will be printed to the file, which will have '_consistent' added to the file name.

check.time.consistency.all   [bool] default = false
> Check the time consistency of all reconciliations up to the limit consistency.limit (undated and partially dated species trees only) and print to the screen the number of consistent reconciliations encountered.

orthology.output   [path] default = none
> Print all pairs of orthologous genes to the given file for the symmetric median reconciliation.

output.dir   [path] default = .
> The directory for all output files. The default is the current directory.

output.prefix   [path] default = none
> A prefix to prepend to all output files.

print.newick   [bool] default = false
> The species tree and the binary, rooted gene tree used in the calculation of the MPR are printed in Newick format to the file specified by print.newick.species.tree.file and print.newick.gene.tree.file respectively. If the given gene trees are amalgamated, or the given gene tree is unrooted/non-binary, the resulting (optimal) rooted binary gene tree is printed. The internal node ids corresponding to the printed reconciliations are printed in the bootstrap fields. A number identifying the gene tree will be appended to the file name if there are multiple gene trees.

print.newick.gene.tree.file   [path] default = geneTree
> Filename of the printed newick gene tree. If a name other than the default is given, print.newick will be set to true. The `output.prefix` will be preprended to the filename if given. The file will be placed in the `output.dir` if given, otherwise in the current directory.

print.newick.species.tree.file   [path] default = speciesTree
> Filename of the printed newick species tree. If a name is specified, print.newick will be set to true. The `output.prefix` will be preprended to the filename if given. The file will be placed in the `output.dir` if given, otherwise in the current directory.

verbose   [bool] default = false
> If true, detailed information on the run, such as running time, will be printed, much of which is intended for those interested in the inner working of the program.

2. RECONCILIATION GRAPH PARAMETERS

All optimal MPRs can be represented by a graph by specifying print.info=true. The program will output the number of reconciliations and the symmetric and asymmetric median scores [4]. The median reconciliations and a random one can be printed by specifying print.reconciliations=1. Use print.newick=true (see previous section) to print gene and species files with the corresponding node ids. The ids in these files are the ones used in the printed reconciliations. Use suboptimal parameters to include non-optimal reconciliations, see Section 4.

An example of a run of ecceTERA with default options on the test files is:
bin/ecceTERA species.file=tests/Stree.tree gene.file=tests/HBG284008_1.gtrees print.info=1

keep.only.canonical.reconciliations   [bool] default = true
> When the species tree is dated, ecceTERA can return several *equivalent* reconciliations in the sense of [1]. We can thus define a canonical reconciliation as the member of this equivalent set with some desirable properties, see [1] for more details. If true, the program simplifies the graph of all possible reconciliations by removing non-canonical

reconciliations. Only canonical reconciliations will be printed.

print.graph   [bool] default = 'false'
  Print the graph representation of all possible MPRs to the file given by print.graph.file, with a number identifying the gene tree, if there are multiple gene trees. If keep.only.canonical.reconciliations is true, the canonical graph [1] will be printed with the suffix "_canonical.sif", with a prefix if given by output.prefix to the directory specified by output.dir. These graphs can be visualized with Cytoscape (http://www.cytoscape.org/).

print.graph.file   [path] default = 'DTLGraph'
  Filename for the printed graph. If a name is specified, print.graph will be set to true.

print.info   [bool] default = false
  Print reconciliation information (see above).

print.reconciliations   [int] default = 0
  Print the median reconciliations and a random optimal reconciliation if the value is 1. Print all reconciliations if the value is 2, up to the limit given by solution.limit. If check.time.consistency=true, then only consistent reconciliations will be printed for a value of 2 and '_consistent' will be added to the file name. If the value is 0, no reconciliations will be printed.

print.reconciliations.file   [path] default = 'reconciliationsFile'
  The file name for printed reconciliations. Suffixes will be added to the file name for median reconciliations (symmetric or asymmetric) and if canonical reconciliations are requested. If a name is specified, print.reconciliations will be set to 1.

solution.limit   [int] default = 100000
  The maximum of reconciliations to print if print.reconciliations=2.

sylvx.reconciliation   [bool] default = false
  Print the reconciliations using the Sylvx format [7] (http://www.sylvx.org). Additionally, an accompanying species tree with the suffix "_sylvx_species.txt" will be printed.

3. AMALGAMATION PARAMETERS

A reconciliation between **S** and a set of gene trees **G** can be performed by "amalgamating" the gene trees in **G** (considered as unrooted) [2]. To do this, specify amalgamate=true. The optimal MPR cost is computed using all of the possible clades found in the gene trees. One of the optimal gene trees corresponding to the best cost can be printed using the print.newick option.

As an example, to run an amalgamation with default options on the test files, use:
bin/ecceTERA species.file=tests/Stree.tree gene.file=tests/HBG284008_10.gtrees amalgamate=1

ale   [bool] default = false
  The gene input file is in the ALE format (thus amalgamate is automatically set to true)

amalgamate   [bool] default = false
  If true, an amalgamation of the gene trees is used to compute the reconciliation.

fix.dtl.costs   [bool] default = true
  If true, when costs are updated while iterating (see max.iterations), only the amalgamation weight will be updated, see [2].

max.iterations   [int] default = 1
  If greater than one, the costs and amalgamation weight will be updated at most max.iterations times, until the costs and amalgamation weight converge (to know how the costs and the amalgamation weight are updated, see [2]).

weight.amalgamation   [double] default = 0

The weight by which the clade supports are multiplied in the amalgamation.

4. PARETO-OPTIMALITY PARAMETERS

By default, the reconciliation graph contains all optimal MPRs. However, the *real* reconciliation is not necessarily a parsimonious one for the given input cost vector – due, among other things, to potential inaccuracies in the input costs. Furthermore, event supports (the proportion of reconciliations that include a given event [1]) have been proven to be more robust when the space around the optimal solutions is considered [4]. If the parameter 'pareto.mod' is set to any integer between 1 and 3, ecceTERA will first explore the space around the optimal solutions by considering all Pareto optimal reconciliations: regardless of their costs, it will compute the triplets <number of duplications, number of transfers, number of losses> that are not *included* in any other one (e.g. <2,1,3> will be excluded if there is a triplet <2,1,2> because <2,1,3> only adds a loss). Furthermore, other conditions can be added so that the printed reconciliations are forced to be close to the optimal ones: either a) by keeping those that are MPR for a cost vector within a given range from the input one (strategy s3 in [5]), or b) by keeping those that do not exceed a given over-cost (strategies s4 and s5 in [5]). Depending on the chosen algorithm, ecceTERA will clean the triplet list to generate solutions for strategy s3 (pareto.mod=1), strategy s4 (pareto.mod=2) and strategy s5 (pareto.mod=3) in [5]. Additional parameters to tune the Pareto optimal reconciliations computation are described below.

Example parameters for strategy s3:
bin/ecceTERA species.file=tests/Stree.tree gene.file=tests/HBG284008_1.gtrees compute.TD=0 pareto.mod=1 nD=0.4 nL=0.4 nDL=0.4

Example parameters for strategy s4:
bin/ecceTERA species.file=tests/Stree.tree gene.file=tests/HBG284008_1.gtrees compute.TD=0 pareto.mod=2 suboptimal.epsilon=1 real.epsilon=1

Example parameters for strategy s5:
bin/ecceTERA species.file=tests/Stree.tree gene.file=tests/HBG284008_1.gtrees compute.TD=0 pareto.mod=3 suboptimal.epsilon=1 real.epsilon=1

min.recs   [double] default = 0
        The reconciliation calculation is repeated with increasing values of epsilon until the number of reconciliations in the solution space is greater than min.recs.

min.recs.increment   [double] default = 1
         To be used in combination with the min.recs parameter; it sets the increments of epsilon.

max.epsilon   [double] default = 30
        To be used in combination with the min.recs parameter; it sets the maximum epsilon value to try.

nD   [double] default = 0, valid values are between 0 and 1, inclusive.
        Optional parameters for pareto.mod 1. It is the percentage of the ratio r=duplication cost/transfer cost that defines the interval for this ratio cost, which is [r-nD*r;r+nD*r]. This is taken as input of strategy s3 in [5].

nDL   [double] default = 0,valid values are between 0 and 1, inclusive
        Similar to nD, but for the ratio duplication cost/loss cost.

nL   [double] default = 0, valid values are between 0 and 1, inclusive
        Similar to nD, but for the ratio loss cost/transfer cost.

pareto.mod   [int] default = 0, valid values are 0-3
        Pareto-optimality algorithm to use (see above). 1 corresponds to "strategy s3" in [5], 2 corresponds to "strategy s4" in [5], and 3 corresponds to "strategy s5" in [5]. The Pareto-optimal algorithm is not used if pareto.mod=0.

real.epsilon   [bool] default = false
        The epsilon value given by suboptimal.epsilon will be considered a real value rather than a percentage.

suboptimal.epsilon   [double] default = 0

        Additional parameter for pareto.mod=2,3. If greater than 0, then non-optimal reconciliations will be included in the graph [5] if their cost is within *suboptimal.epsilon%* of the optimal. For example, if suboptimal.epsilon=2 and the optimal cost found is 36, then not only will all reconciliations with a cost of 36 will be generated, but also any Pareto-optimal reconciliations with a cost less than or equal to 36.72, i.e. those within 2% of the optimal. If this parameter is omitted, or the given value is negative, then no constraint on the over-cost is imposed, i.e. suboptimal.epsilon=+infinity.

5. RECONCILIATION FILE FORMAT

This reconciliation is a mapping from the gene tree to the species tree. For a formal definition of a reconciliation, see [1]. Separate files for the gene and species trees can be generated in Newick format with the ids used specified in the bootstrap fields using the print.newick.file option.

As noted earlier, the reconciliation can be printed in the SylvX format [7] (software available at HYPERLINK "http://www.sylvx.org/"http://www.sylvx.org) by specifying sylvx.reconciliation=true. Alternatively, the scripts folder contains a script (recon2xml.pl) to convert an ecceTERA reconciliation into the PhyloXML format [6]. This script can be run from the command-line as 'recon2xml.pl <reconciliation_file>'. Run 'recon2.xml.pl -h' for more details.

In our reconciliation format, each line corresponds to a clade/gene id and the events occurring to the clade, ordered from earliest to latest, as follows:

clade/geneId:[splitEvent];[noSplitEvents];<splitEvent>:cladeId_l,cladeId_r

The first split event is the one that led to the clade. This event does not exist for the root. Following that, there are zero or more events that do not split the clade, for example a speciation loss event. The last event is a clade split event (transfer, duplication, or speciation). The final two clade ids, cladeId_l and cladeId_r, are the two clade/gene child ids. For leaves, the gene leaf name is used. Using these two ids, the entire gene tree can be reconstructed. The first clade listed in the file is always the gene/clade root.

Bifurcation (Split) Events
The basic event structure of <splitEvent> is "speciesId,event,auxId_l,auxId_r@support", where auxId_l and auxid_r consist of the species ids describing the event, with auxId_l corresponding to cladeId_l and auxId_r to cladeId_r, where appropriate, e.g. speciations (see below for details). The possible events, and their auxiliary values are as follows:

D: duplication, aux=x,x.
```
 |   u,x   |
 | ul,x  ur,x|              u:x,D,x,x@0.5:ul,ur
```

      DD: duplication within dead (outgroup) appears as u:-1,DD,-1,-1@support.

S: speciation, aux=xl, xr. The xl species event is the species id containing cladeId_l, and species id_r contains cladeId_r.
```
  |  u,x  |
 /   /\   \          u:x,S,xl,xr@0.5:ul,ur
|ul,xl||ur,xr|
```

T: transfer: aux=xl, xr. The id_l species event is the species id containing cladeId_l, and species id_r contains cladeId_r. For a transfer event, xl or xr will be the same as the event species id, speciesId. Additionally, x=xl or x=xr, which follows from the definition of a reconciliation.
```
 | u,x |
 |ul,xl| -->|ur,xr| u:x,T,xl,xr@0.5:ul,ur
```

TTD: transfer to dead, aux=x,-1, where x is the transfer from id

TFD: transfer from dead, aux=-1,x, where x is the transfer to id.

No Split Events

The basic event structure of <noSplitEvent_i> is "speciesId,event,lostId,keptId@support". The possible events, and their auxillary values are as follows:

SL: speciation loss, aux=lostId,keptId. A speciation event where the gene is lost in one of the two descendant species (lostId), and kept in the other (keptId). The clade/gene id does not change and the id of the species which keeps the gene is also the species id of the following event (if it exists)

```
   |  u,x   |
  /   /\    \          u:<splitEvent>;x,SL,xLost,xKept@0.5.
 |xLost| |u,xKept|
```

TL: transfer loss, aux=x,x'. The gene is transferred from species x to species x' and lost in x. The clade/gene id does not change and the id of the species to which the gene is transferred, x', is the species id of the following event (if it exists).

```
 |u,x|   |   |
 |  |  -->|u,x'|        u:<splitEvent>;x,TL,x,x'@0.5
```

TLTD: transfer loss to unsampled/extinct species, aux=x,-1. The gene is transferred from species x to the dead (alpha/outgroup) and lost in x. The clade/gene id does not change.

```
      |u,x|   |      |
      |  | -->|u,alpha|     u:<splitEvent>;x,TLTD,x,-1@0.5
```

TLFD: transfer from unsampled/extinct species, aux=-1,x. The gene is transferred from the dead (alpha/outgroup) to species x and lost in the dead. The clade/gene id does not change.

```
      |u,alpha|   |  |
      |       | -->|u,x|     u:<splitEvent>;x,TLFD,-1,x@0.5
```

6. RECONCILIATIONS WITH SPECIES NETWORKS

In [8-9], the authors presented algorithms to reconcile gene trees with species networks: the "minimum switching algorithm" (solving Problem 1 in [8-9]) and the "minimum network reconciliation algorithm" (solving Problem 2 in [8-9]). Our implementation solves respectively Problem 1 and 2 in [8] when HGT.cost is set to 0, and Problem 1 and 2 in [9] otherwise.

The species network has to be specified in eNewick (for "extended Newick") format. For more details on this format, see http://dmi.uib.es/~gcardona/BioInfo/enewick.html. Apart from the "gene.mapping.file" parameter, all non-optional parameters described in Section 1 can be used in the network mode. The other parameters specific to this mode are:

input.network   [bool] default = false
        The input species phylogeny is a network (in extended newick)

best.switch   [bool] default = true
        Run minimum switching algorithm

min.recon   [bool] default = false
        Run minimum network reconciliation algorithm

print.newick.best.switching.file   [path] default = 'bestSwitching'
        The file name for the best species switching.

Note that the best species switching is a tree in Newick format and can be used as species tree for all algorithms described in Sections 1-4 of this manual.

REFERENCES

[1] Celine Scornavacca, Wojciech Paprotny and Vincent Berry and Vincent Ranwez. Representing a set of reconciliations in a compact way. *Journal of Bioinformatics and Computational Biology* Vol. 11, No. 2 (2013) p. 1250025.

[2] Celine Scornavacca, Edwin Jacox, and Gergely Szöllősi. Joint Amalgamation of Most Parsimonious Reconciled Gene Trees. *Bioinformatics* (2014): btu728.

[3] Gergely Szöllősi, Eric Tannier, Nicolas Lartillot, Vincent Daubin. Lateral gene transfer from the dead. *Systematic Biology* Vol. 62, No. 3 (2013) pp. 386-97.

[4] Thi-Hau Nguyen, Vincent Ranwez, Vincent Berry, Celine Scornavacca. Support measures to estimate the reliability of evolutionary events predicted by reconciliation methods. *PloS One* Vol. 8, No. 10 (2013) p. e73667.

[5] Thu-Hien To, Edwin Jacox, Vincent Ranwez, Celine Scornavacca. A Fast Method for Calculating Reliable Event Supports in Tree Reconciliations via Pareto-optimality. *BMC bioinformatics* Vol. 16, No. 1 (2015) p. 384.

[6] Mira V Han and Christian M Zmasek. PhyloXML: XML for evolutionary biology and comparative genomics. *BMC Bioinformatics* Vol. 10 (2009) p. 356.

[7] François Chevenet, Jean-Philippe Doyon, Celine Scornavacca, Edwin Jacox, Emmanuelle Jousselin and Vincent Berry. SylvX: a viewer for phylogenetic tree reconciliations. *Bioinformatics* (2015): btv625.

[8] Thu-Hien To and Celine Scornavacca. Efficient algorithms for reconciling gene trees and species networks via duplication and loss events. BMC genomics Vol 16, Suppl 10 (2015) p. S6.

[9] Celine Scornavacca, Joan Carles Pons Mayol, Gabriel Cardona. Fast algorithm for the reconciliation between gene trees and LGT networks. Submitted.

[10] Edwin Jacox, Mathias Weller, Eric Tannier and Celine Scornavacca. Resolution and reconciliation of non-binary gene trees with transfers, duplications and losses. Submitted.