Constructing consensus polyploid phylogenies

Katharina T. Huber[1], Vincent Moulton[1], Andreas Spillner[2], Sabine Storandt[3], Radosław Suchecki[1]





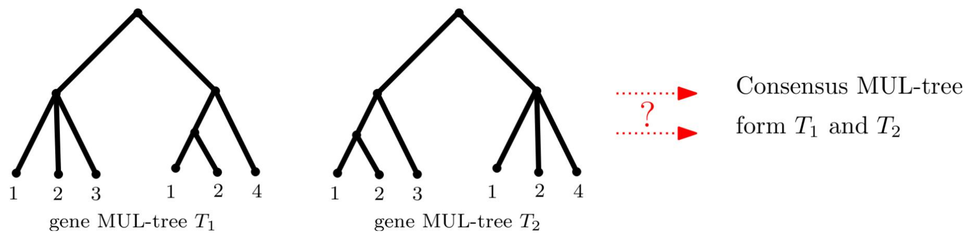
1. School of Computing Sciences, University of East Anglia, Norwich, UK
2. Department of Mathematics and Computer Science, University of Greifswald, Germany
3. Institut für Formale Methoden der Informatik, University of Stuttgart, Germany

Summary

Multi-labeled trees (or MUL-trees) along with phylogenetic networks are used to reconstruct evolutionary past of polyploid species. The key feature of a MUL-tree is that more than one of its leaves can be labelled by a single species. Consequently the problem of constructing a consensus MUL-tree from a set of multi-labeled gene trees is much more complex than in the case of standard phylogenetic trees. Furthermore, for a given input, multiple distinct consensus MUL-trees can be constructed, and thus each of these MUL-trees needs to be scored, so that a most parsimonious one (in a well defined sense) can be identified. We present an improved implementation of an established algorithm that constructs a consensus MUL-tree from a set of multi-labeled gene trees. We have prepared an experimental pipeline which allowed us to identify certain limitations of the original approach and also to evaluate several speed-ups and improvements. The reduction in memory consumption as well as run-time makes it feasible to construct consensus MUL-trees from much larger and more complex input datasets.

Motivation

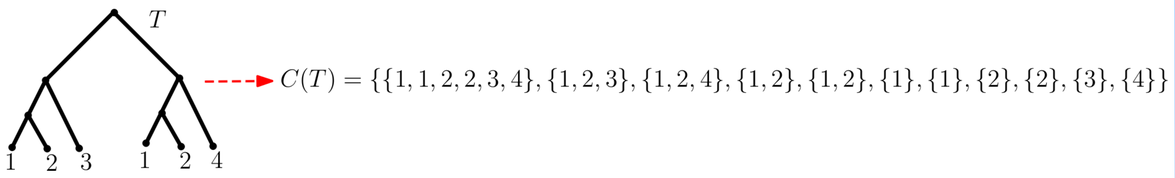
In the context of understanding the evolutionary past of polyploidy species, the following problem is encountered:



- Note that T_1 and T_2 are MUL-trees rather than phylogenetic trees i.e. some of the leaves in T_1 (and in T_2) share the same label

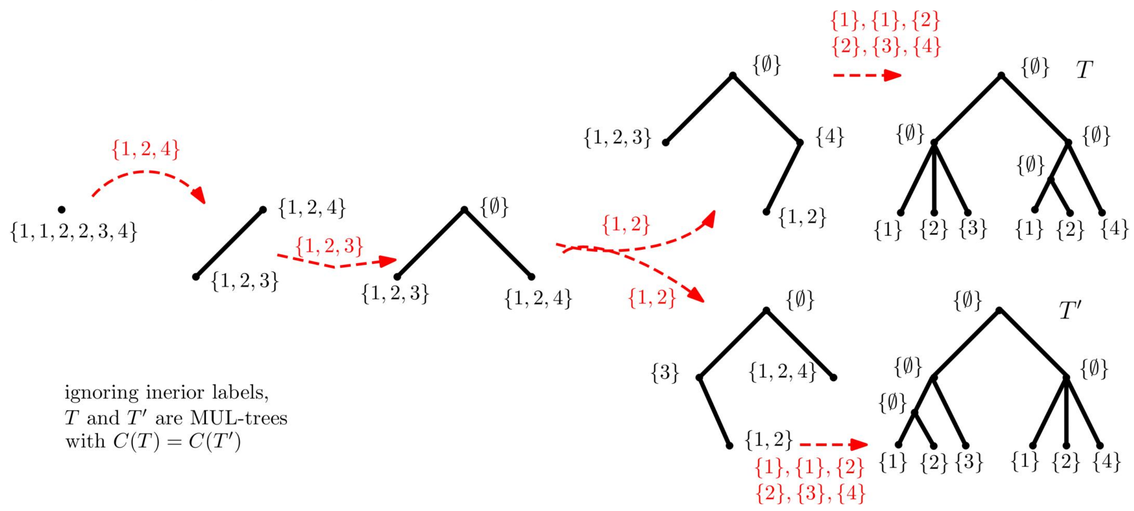
A first approach

Analogous to how a phylogenetic tree induces a set of clusters, a MUL-tree induces a multiset *C(T)* of multiclusters:



Idea

Reconstruct consensus MUL-tree *T(S)* from the multiset of multiclusters from the trees in the set *S* of MUL-trees, consider for example *C(T)*



However,

- It is NP-hard to decide whether a multiset C of multiclusters can be represented as a MUL-tree [1].
- A MUL-tree constructed from C may not be unique.
- A first MUL-tree heuristic "Padre Consensus" [2] is implemented in PADRE [3]. However, this implementation finds it difficult to handle larger datasets.

A new heuristic for constructing consensus MUL-trees (outline) [4]

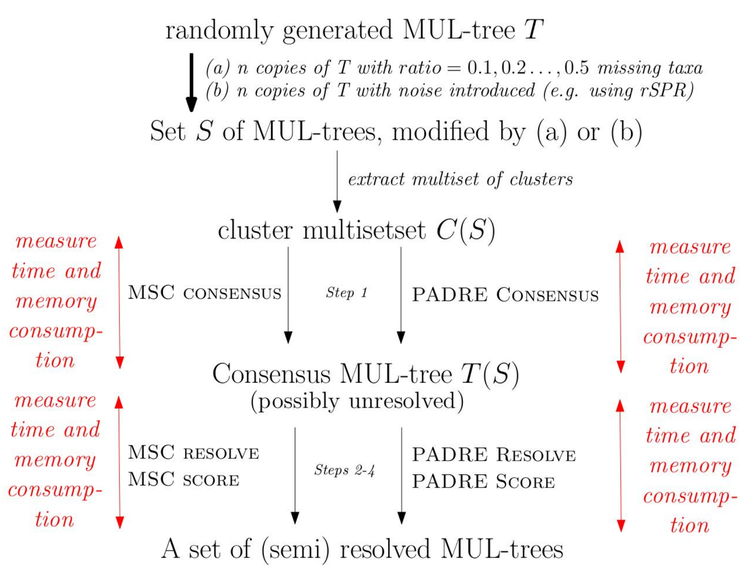
Input: Set S of gene MUL-trees.
Output: Consensus MUL-tree *T(S)*.

Step 1: Identify a compatible set C_{COMP} of multiclusters (uses more sophisticated data structures than Padre)
Step 2: Construct a set W of MUL-trees that display C_{COMP}, i.e. for each $T \in W$, $C(T) = C_{COMP}$
Step 3: For all $T \in W$ carefully select interior vertices of high outdegree to be resolved (PADRE resolves all of them)
Step 4: Score each tree obtained by following previous steps.

Experiment outline

We have constructed a pipeline that allows us to identify any shortcomings of the original implantation (PADRE) and evaluate the new implementation (MSC) .

Main goal was to be able to tackle larger and more complex datasets

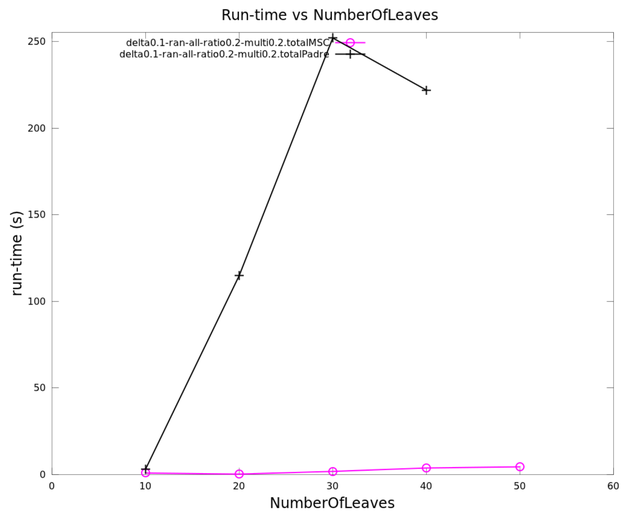


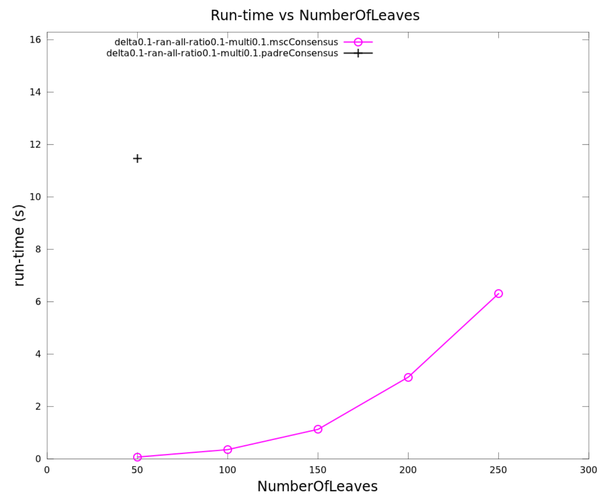
Some results

These selected results have been obtained from various experiments on datasets where a proportion of leaves have been randomly removed from the input gene MUL-trees.

Comparison of average run-times of the old (PADRE) and the new (MSC) implementation.

Note that the fall in average run time for PADRE on 40 leaves datasets is an artefact of the implementation reaching it's limit, it failed to produce results for some of the datasets on 50 leaves.

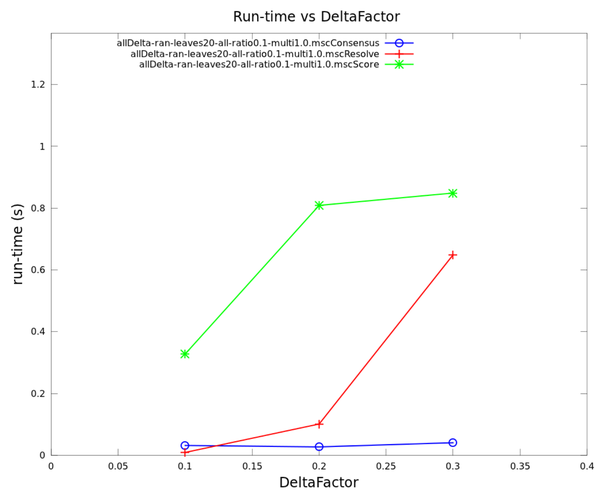


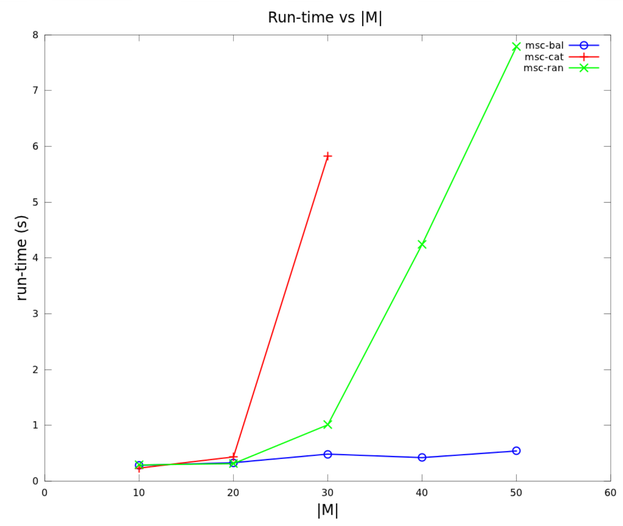


Employing more sophisticated data structures significantly cuts average run-time of MSC CONSENSUS in comparison to PADRE CONSENSUS (notice only a single datapoint for PADRE as it was unable to compute some of the datasets on more than 50 leaves in the allocated time or because it ran out of memory).

The DeltaFactor is a measure of complexity of the input. The higher the value, the more multiplicity in the leaf set of the input gene MUL-trees.

Here, we observe that although DeltaFactor does not visibly influence the run-time of the consensus method, it strongly affects resolving and scoring.





This chart illustrates preliminary findings on the influence of topology of the input gene MUL-trees on the average run-time of the new implementation.

The x-axis is labelled by the number of leaves in the input.

In addition to randomly generated topologies (ran), we have generated fully balanced (bal) and caterpillar (cat) trees .

It appears that balanced trees are easier to reconstruct than those of caterpillar topology, with random ones falling somewhere in the middle.

Conclusions and further work

- New implementation able to deal with much larger and more complex datasets than PADRE.
- Resolving high outdegree nodes is the main bottleneck.
- Further improvements to be tested experimentally.
- Experiments to be repeated with noise randomly introduced into the data.

References

[1] Huber, K., Lott, M., Moulton, V. and Spillner, A. (2008): The complexity of deriving multi-labeled trees from bipartitions. *Journal of Computational Biology* 15(6), 639-651.

[2] Lott, M., Spillner, A., Huber, K. T. , Petri A., Oxelman, B., and Moulton, V. (2009): Inferring polyploid phylogenies from multiply-labeled gene trees. BMC Evolutionary Biology, 9:216, 2009

[3] Lott, M., Spillner, A., Huber, K. T., and Moulton, V. (2009). Padre: A package for analyzing and displaying reticulate evolution. BIO, 25(9):1199–1200.

[4] Huber, K.T., Moulton, V., Spillner, A., Storandt, S., Suchecki, R.: Constructing consensus polyploid phylogenies - in preparation.