ASTRAL III

Polynomial time species tree reconstruction from partially resolved gene trees. Zhang et al., 2018

Presented by:
Shayan Tabe Bordbar
Problem formulation

• Input: set of \( k \) input gene trees
• Output: binary tree \( T \), with maximum quartet similarity with input gene trees.
• Proved to be NP-hard by Lafond and Scornavacca.
• ASTRAL solves this problem under a constrained search space.
• Set \( X \) constrains the bipartitions of \( T \).
• Further improvements on ASTRAL, have been mainly focused on improving the running time by using different heuristics to form set \( X \).
ASTRAL III

• Aim: improving the running time of ASTRAL II without sacrificing accuracy.
• In the original ASTRAL implementation (ASTRAL I), the set of all bipartitions from all input gene trees forms $X$.
• ASTRAL II computes several extended majority consensus trees from input gene trees and resolves polytomies using three methods:
  (a) UPGMA starting from sides of polytomies, based on a pre-computed distance matrix.
  (b) Greedy consensus of gene trees sub-sampled to one taxon from each side of a polytomy.
  (c) Compute $d$ caterpillar trees, from samples obtained one from each side of a polytomy of degree $d$. 
ASTRAL III improvements

ASTRAL III improves on ASTRAL II in two aspects:

- Handling of tens of thousands of input gene trees, by first limiting the growth of set $X$ to $O(nk)$, where $n$ is the number of species and $k$ is the number of gene trees, and second by exploiting similarities in input gene trees.

- Handling polytomies such that scoring high degree nodes takes the same time as binary nodes.
Search space improvement

• While in ASTRAL I and II, set $X$ can grow exponentially, ASTRAL III ensures that set $X$ grows in $O(nk)$.
  • Modifying one of the three heuristics used to resolve a polytomy of extended majority consensus trees whose bipartitions form $X$ in ASTRAL II, such that $|X|$ grows linearly with the degree of the polytomy.

• In ASTRAL III, gene tree polytomies contribute more bipartitions to set $X$ compared to ASTRAL II, however $|X|$ still grows in $O(nk)$.
  • This is done by repeating the sampling procedure from each side of the gene tree polytomy multiple times. Extracted samples are used to form new bipartitions using a pre-computed guide tree, to be added to set $X$. 
Other improvements

• More efficient computations regarding polytomy nodes, in the dynamic programming scheme.
• Keeping track of nodes that appear in multiple trees and performing computations for those nodes only once.
•Trimming of solution space using an upper bound on the best achievable score for a clade.
Research questions

• How does contracting low support branches affect accuracy of ASTRAL?

• How does ASTRAL-III do in terms of running time and search space as compared to ASTRAL-II?

• How does the guarantee on the size of search space implemented in ASTRAL-III affect accuracy as compared to ASTRAL-II?
Datasets

• Avian biological dataset
  • 14,446 loci
  • 48 species
  • Results compared with two previously constructed species trees

• Simulated avian-like dataset
  • Emulates Avian dataset
  • Varying ILS levels
  • Varying gene tree estimation error levels

• SimPhy-homogeneous (S100)
  • 101 taxa (50 replicates), 1000 genes per replicate
  • Used Indelible to create nucleotide seqs of four fixed lengths

• SimPhy-ASTRAL2 (S200)
  • 201 taxa
  • Higher ILS compared to S100
Fig. 1 Properties of the S100 dataset. a The density plot of the amount of true gene discordance measured by the FN rate between the true species tree and the true gene trees. b The density plot of gene tree estimation error measured by FN rate between true gene trees and estimated gene trees for different sets of sequence lengths.
Contraction effect on S100

Fig. 2 Impact of contraction on the S100 dataset. The FN error of ASTRAL-III species trees is shown on the S100 dataset given $k = 50, 200, 500, \text{or } 1000$ genes (boxes) run on the original FastTree gene trees (non) or gene trees with branches with $\leq \{0, 3, 5, 7, 10, 20, 33, 50\}%$ support contracted ($x$-axis). Average FN error and standard error bars (200 replicates) are shown with the four alignment lengths combined (black solid line). Average FN error broken down by alignment length (50 replicates) is also shown (dashed colored lines).
Fig. 3 Impact of contraction on the avian simulated dataset. The FN error of ASTRAL-III species trees is shown on the avian simulated dataset given $k = 1000$ genes with (left) fixed sequence lengths = 500 and varying levels of ILS, or (right) fixed ILS (1X) and varying sequence length, in each case both with full FastTree gene trees (non) or trees with branches with $\leq \{0, 3, 5, 7, 10, 20, 33, 50\}$% support contracted (x-axis). Average and standard error bars are shown for all conditions combined (black solid line) and also for each model condition separately (dashed color lines). Each model condition has 20 replicates.
Contraction effect on avian
Running time

Fig. 5 Running time versus k. Average running times (4 replicates) are shown for ASTRAL-II and ASTRAL-III on the avian dataset with 500bp or 1500bp alignments with varying numbers of gens (k), shown in log scale (see Additional file 1: Figure S2 for normal scale). A line is fit to the data points in the log/log space and line slopes are shown. ASTRAL-II did not finish on $2^{14}$ genes in 48 hours.
Fig. 6 Weight calculation and $|X|$ on S100. Average and standard error of (a) the time it takes to score a single tripartition using Eq. 3 and (b) search space size $|X|$ are shown for both ASTRAL-II and ASTRAL-III on the S100 dataset. Running time is in log scale. We vary numbers of gene trees (boxes) and sequence length (200 and 1600). See Additional file 1: Figure S3 for similar patterns for with 400 and 800bp alignments.
Conclusion

ASTRAL III improves the running time of ASTRAL II without sacrificing accuracy. In particular, ASTRAL III enhances:

- Scalability to datasets with large number of gene trees.
- Handling polytomies in input gene trees.

Mentioned improvements enable authors to investigate the effect of support-based branch removal. They conclude that removing very low support branches generally improves accuracy, but more aggressive elimination of branches can hurt accuracy.