AnGST: The Analyzer of Gene & Species Trees

User manual written by

Lawrence A. David

Last revised: December 2010

Abstract

AnGST, the Analyzer of Gene & Species Trees, is a phylogenetic algorithm that "reconciles" any observed differences between a gene tree and a reference tree (species tree). AnGST uses a generalized parsimony criterion to infer a minimal set of evolutionary events, including horizontal gene transfer (HGT), gene duplication (DUP), gene loss (LOS), speciation (SPC) and exactly one gene birth or genesis event (GEN). Inference errors due to phylogenetic uncertainty are minimized by incorporating tree construction into the reconciliation process. Multiple gene tree bootstraps can be provided to AnGST; the algorithm will retain and combine bootstrap subtrees which yield the most conservative reconciliation consistent with the sequence data. The AnGST software package is implemented in the Python programming language and can be downloaded from: almlab.mit.edu/angst/.

This user manual focuses on explaining how to run AnGST. For a more thorough discussion of the theory behind AnGST and algorithm benchmarking, see the following manuscript's Supplementary Information:

LA David & EJ Alm. "Rapid evolutionary innovation during an Archaean Genetic Expansion." *Nature*, 2010. doi:10.1038/nature09649.

About

Lawrence David wrote AnGST during the completion of his PhD in Computational and Systems Biology at the Massachusetts Institute of Technology (2005-2010). He is presently a Junior Fellow at the Harvard Society of Fellows. User questions, suggestions, or bug reports are welcomed and can be sent to Lawrence via the email address ldavid@fas.harvard.edu. Lawrence maintains a personal homepage at www.stinkpot.org.

Citation

AnGST is free for users to run and modify. If you use AnGST in a publication please cite:

LA David & EJ Alm. "Rapid evolutionary innovation during an Archaean Genetic Expansion." *Nature*, 2010. doi:10.1038/nature09649.

Acknowledgements

AnGST was developed in collaboration with many individuals. Eric Alm guided the development of the algorithm. Dirk Gevers, Abdoulaye Diallo, Ming-Chun (Miki) Lee, and David Robinson provided helpful initial user feedback. Albert Wang built the online AnGST server (almlab.mit.edu/angst). During his research, Lawrence David was supported by a National Defense Science & Engineering Graduate Fellowship (DoD) and a Whitaker Health Sciences Fund Fellowship.

Contents

1 Installation

To run AnGST on your own computer or cluster, you will need to have experience running programs on the command-line. If you haven't used command-line software before but would like to learn more about this computing environment, you can try perusing this online tutorial. Otherwise, please use our online AnGST server: almlab.mit.edu/angst/.

AnGST requires Python 2.X to run. AnGST is not compatible with Python 3.X. If you don't already have Python installed, you can download the Python installation package from the official Python website. After installing Python, decompress the AnGST tarball and you're ready to run AnGST.

2 Quick start

A toy dataset has been included with the AnGST distribution. To reconcile the gene and species trees in the toy dataset, navigate to the unzipped AnGST folder and execute on the command line:

```
>> python angst_lib/AnGST.py example/AnGST.input
```

The following sections walk through the various inputs (Section 3) and outputs (Section 4) associated with the toy reconciliation.

3 Inputs

Input file

The only argument directly passed to AnGST on the command-line is the `AnGST.input` filename. Tree filenames and reconciliation parameters are stored in this file. Here are the contents of the example `AnGST.input` file:

```
species=./example/species.txt
gene=./example/gene.txt
penalties=./example/penalty.file
output=./example/angst/
```

Species tree

Specify the species tree in `AnGST.input` using the following format:

```
species=species.newick
```

The input species (reference) tree to AnGST should be written in Newick format. Bootstrap values on the species tree will be disregared. Species names should not contain periods, underscores, or spaces. The input species tree must be rooted. All branches on the species tree, including the root, must have a branch length. Trees should end with a semicolon. Here is an example of the required species tree format:

```
(((1:1.0,(2:1.0,3:1.0):1.0):1.0,((4:1.0,5:1.0):1.0,6:1.0):1.0):0.001);
```

See Section 5 below if you have a time tree and would like to place temporal constraints on proposed HGT events.

Gene tree

Specify the gene tree in `AnGST.input` using the following format:

```
gene=gene.newick
```

The input gene tree(s) to AnGST should also be provided in Newick format. The gene tree should be unrooted (AnGST roots the tree while looking for the lowest scoring reconciliation scenario). All branches on the gene tree should have a branch length. Each leaf of the gene tree must correspond to one leaf on the species tree. All leaves must also have an identifier tag. Two leaves from the same genome should not have the same identifier tag. Identifier tags should be separated from species names by a period or an underscore. Here is an example of an acceptably formatted gene tree:

```
((2_0:1.0,(1_0:1.0,1_1:1.0):1.0):2.0,(4_0:1.0,5_0:1.0):1.0,6_0:1.0);
```

AnGST can minimize inference errors due to phylogenetic uncertainty by incorporating tree construction into the reconciliation process. Multiple gene tree bootstraps can be provided to AnGST; the algorithm will retain and combine bootstrap subtrees which yield the most conservative reconciliation consistent with the sequence data. See Section 5 below for details on how to use this feature.

Event penalties

AnGST will use event penalties to find the reconciliation with the lowest overall cost. The file enumerating event penalties are specified in `AnGST.input` with the line:

```
penalties=penalties.file
```

User penalties in `penalties.file` should take the following format for horizontal gene transfer (hgt), gene duplication (dup), gene loss (los), and speciation (spc):

```
hgt: 3.0
dup: 2.0
los: 1.0
spc: 0.0
```

Penalties should be real and non-negative. Different event penalties will lead to different reconciliation scenarios. Choosing event penalties is not an easy problem, and you may want to try a range of penalties. We found that when looking across a broad range of gene families and eukaryotic and prokaryotic genomes, the event penalties listed above minimized the divergence in genome size among related genomes [1].

Output directory

Assign the AnGST output directory in `AnGST.input` using the following syntax:

```
output=./angst_output/
```

4 Outputs

An AnGST run generates several output files:

```
>> /ls -lt example/angst/
total 48
-rw-r--r--  1 lad  staff   77 Dec 21 19:31 AnGST.counts
-rw-r--r--  1 lad  staff  140 Dec 21 19:31 AnGST.events
-rw-r--r--  1 lad  staff  378 Dec 21 19:31 AnGST.leaf
-rw-r--r--  1 lad  staff  100 Dec 21 19:31 AnGST.newick
-rw-r--r--  1 lad  staff    4 Dec 21 19:31 AnGST.score
-rw-r--r--  1 lad  staff  676 Dec 21 19:31 AnGST.stats
```

AnGST.counts

Records the number of gene copies inferred in lineages on the species tree. Ancestral lineages are denoted using dashed concatenations of their leaf species names.

AnGST.events

A list of the inferred evolutionary events: [brn], gene family birth; [spc], speciation; [los], gene loss; [dup], gene duplication; [hgt] horizontal gene transfer; and [cur], an extant gene copy. Ancestral lineages are denoted using dashed concatenations of their leaf species names.

AnGST.leaf

Chronicles the evolutionary history of each gene copy.

AnGST.newick

The reconciled gene tree in Newick format. If multiple bootstraps were provided as input for the gene tree (Section 5), this gene tree may not exactly match any of the individual bootstrap trees. Mappings from internal nodes on the gene tree to nodes on the species tree are recorded where node bootstrap values are conventionally written in Newick format. These mappings are expressed using a concatenation of leaf species names.

AnGST.score

Contains the AnGST reconciliation score.

AnGST.stats

A general log of the AnGST reconciliation. Includes statistics on AnGST running time and memory usage.

5 Optional

Bootstrap trees

Errors or uncertainty in gene phylogenies can lead to the inference of spurious macroevolutionary events [2] and is a particular concern for deeply branching phylogenies [3]. AnGST can account for phylogenetic uncertainty by simultaneously reconciling and reconstructing a gene tree; the tree with the lowest reconciliation cost can be constructed from an ensemble of trees consistent with the sequence data. Suitable tree ensembles can be generated by the non-parametric bootstrapping step of tree inference algorithms like PhyML. AnGST can merge subtrees from these bootstraps into a single "chimeric" tree that does not match any of the input bootstraps exactly, but whose bipartitions can be found in at least one of the input bootstraps. In simulations, we observed these trees to be significantly more accurate than trees based on sequence likelihood alone, although they generally have lower likelihood [1].

AnGST will incorporate multiple bootstraps into a reconciliation if there are multiple trees in the gene tree file. An example bootstrapped gene tree is provided in `./example/boot.txt`:

```
>> more ./example/boot.txt
((2_0:1.0,(1_0:1.0,3_0:1.0):1.0):2.0,(4_0:1.0,5_0:1.0):1.0,6_0:1.0);
((1_0:1.0,(2_0:1.0,3_0:1.0):1.0):2.0,(4_0:1.0,6_0:1.0):1.0,5_0:1.0);
```

Reconcile this tree by editing `AnGST.input` to read:

```
gene=./example/boot.txt
```

Ultrametric trees

If the branch lengths on the provided species tree represent times, AnGST can restrict the set of possible inferred gene transfers to only those between contemporaneous lineages. Add to `AnGST.input`:

```
ultrametric=True
```

Any non-zero chronological overlap is sufficient to allow transfers. But, if a gene transfer is inferred from node s_1 to node s_2, subsequent transfers of the gene copy in s_2 may only occur with lineages which exist during the range $T_1 \cap T_2$, where T_1 and T_2 are the times spanned by the parent edges of s_1 and s_2, respectively.

6 Currently unsupported

Several additional features have been built into AnGST but have not yet been extensively debugged. These include the ability to:

- Specify an outgroup for the gene trees, or provide rooted gene trees.
- Provide event-specific costs to AnGST (e.g. HGT from A $\rightarrow$ B costs exactly 2.5).
- Fix the node on the species tree at which the gene family was born.

If there is sufficient user interest, future versions of AnGST may include these features. Please feel free to ask!

Bibliography

[1] Lawrence A David and Eric J Alm. Rapid evolutionary innovation during an Archaean genetic expansion. *Nature*, Dec 2010.

[2] Matthew W Hahn. Bias in phylogenetic tree reconciliation methods: implications for vertebrate genome evolution. *Genome Biol*, 8(7):R141, Jan 2007.

[3] J Bergsten. A review of long-branch attraction. *Cladistics*, 21:163–193, Jan 2005.