Phylogenetic Placement & Taxonomic Identification

with

SEPP and TIPP

Michael Nute

STAMPS 2018

Woods Hole, MA

Agenda

- Step-by-step Tutorial: **https://github.com/MGNute/stamps-tutorial**
  - Includes link to pdf of these slides
- Quick Review & Practical Considerations
  - SEPP vs. TIPP comparison
  - SEPP/TIPP: for Shotgun or Amplicon Data?
  - SEPP: interpreting branch length
- Additional Reference Material
  - TIPP: BLAST pipeline for shotgun data
  - TIPP reference package link & contents
  - Guide to SEPP command-line arguments
  - Guide to TIPP command-line arguments
  - Contents of TIPP/SEPP placement JSON file
  - Additional Resources

Step-by-Step Tutorial:

- **Part I**: Taxonomic Identification using TIPP
  - Test Data: Lemur vaginal swab, 16S amplicon sequencing with 454.
  - Reference Data: 11,988 full-length sequences from RDP
    - *For the tutorial: Clostridia only*
  - **Part I.a**: visualizing the placements from TIPP
- **Part II**: Phylogenetic Placement using SEPP
  - Test Data: 5 reads selected from the data above
  - Reference Data: (*same as above*)
- **https://github.com/MGNute/stamps-tutorial**
  - Follow the link for the “Hands-On Worksheet” to get started.
  - Feel free to get started and let us know if you have questions!

TIPP vs. SEPP

	SEPP *"Placement in a Phylogeny"*	TIPP *"Placement in a Taxonomy"*
Output	• Attachment point and branch length for each read. *(esoteric)*	• Taxonomic Identification for each read. *(concrete)*
Reference Tree	• Phylogeny *(based on gene, may not agree with taxonomy)*	• Taxonomy
Used for	• Downstream analysis, data exploration (visualization), etc…	• Taxonomic Identification • Abundance Profiling
Advantage	• Placement provides richer data for metrics like Unifrac (Jansenn, et. al, *mSystems*, 2018) • Branch length can be informative	• More accurate than BLAST when novel sequences are present

TIPP & SEPP: Shotgun or Amplicon?

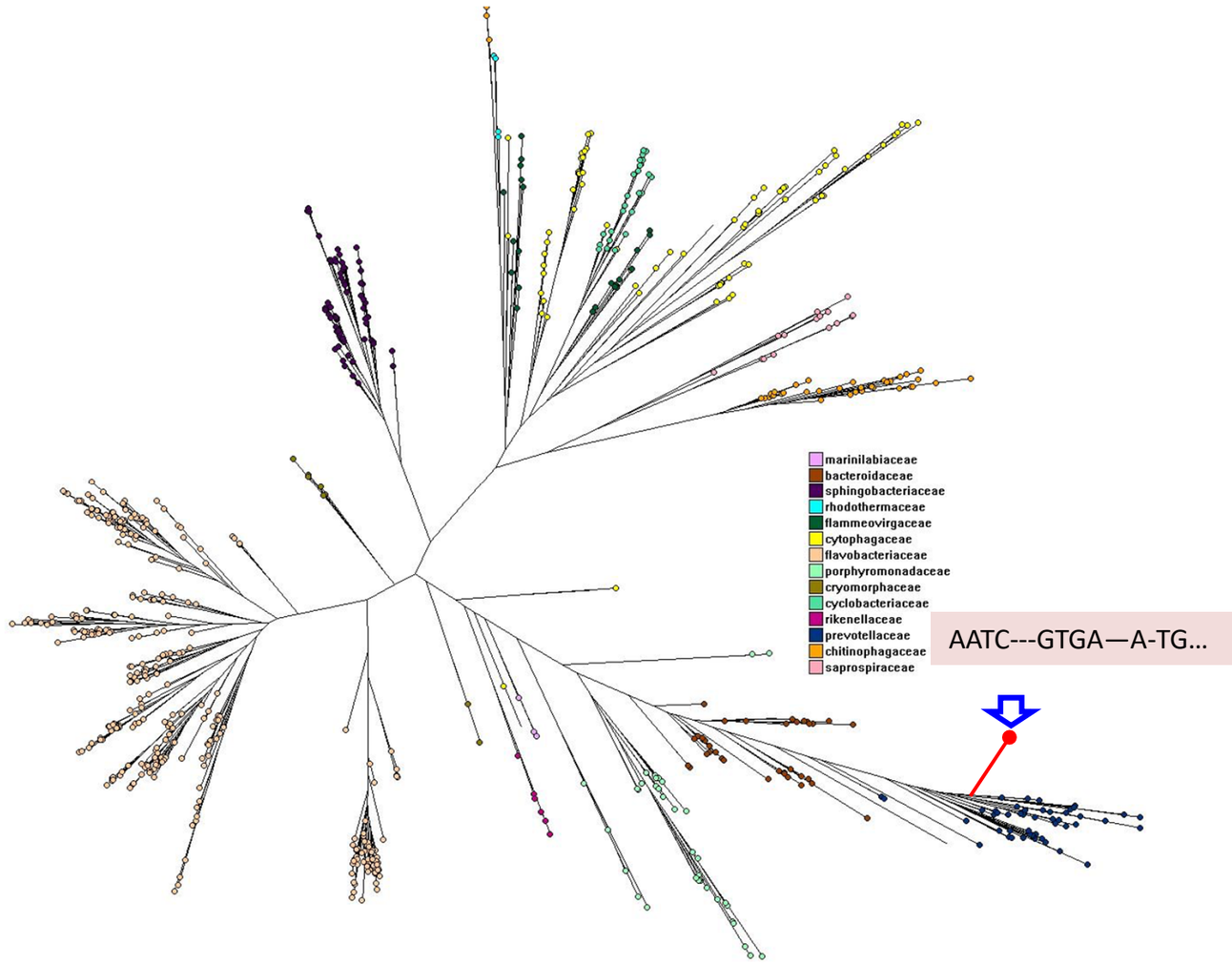
Both!!

- **Shotgun Sequencing**
  - Reads should be filtered down to subset that hits the reference tree (using BLAST, e.g.)
  - Reads should be corrected to go the same direction as the reference, and trimmed where edges are overlapping (TIPP will do this automatically via run_abundance.py, *see slides 8-9*).
  - For TIPP, can be done for many references separately and the results combined
    - *See run_abundance.py example later (again, slides 8-9)*
  - Reference Packages:
    - 40 COGS usable as marker genes (single-copy, universal) available with full reference packages (*link on slide 10*)
- **Amplicon Sequencing**
  - No need to filter reads (they already come from a single known gene)
  - Reference Packages:
    - In tipp.zip, package “RDP_Bact_2016” is recent and high-quality for 16S amplicon data.
    - For other genes, roll your own reference package (or contact me for help at nute2@Illinois.edu)

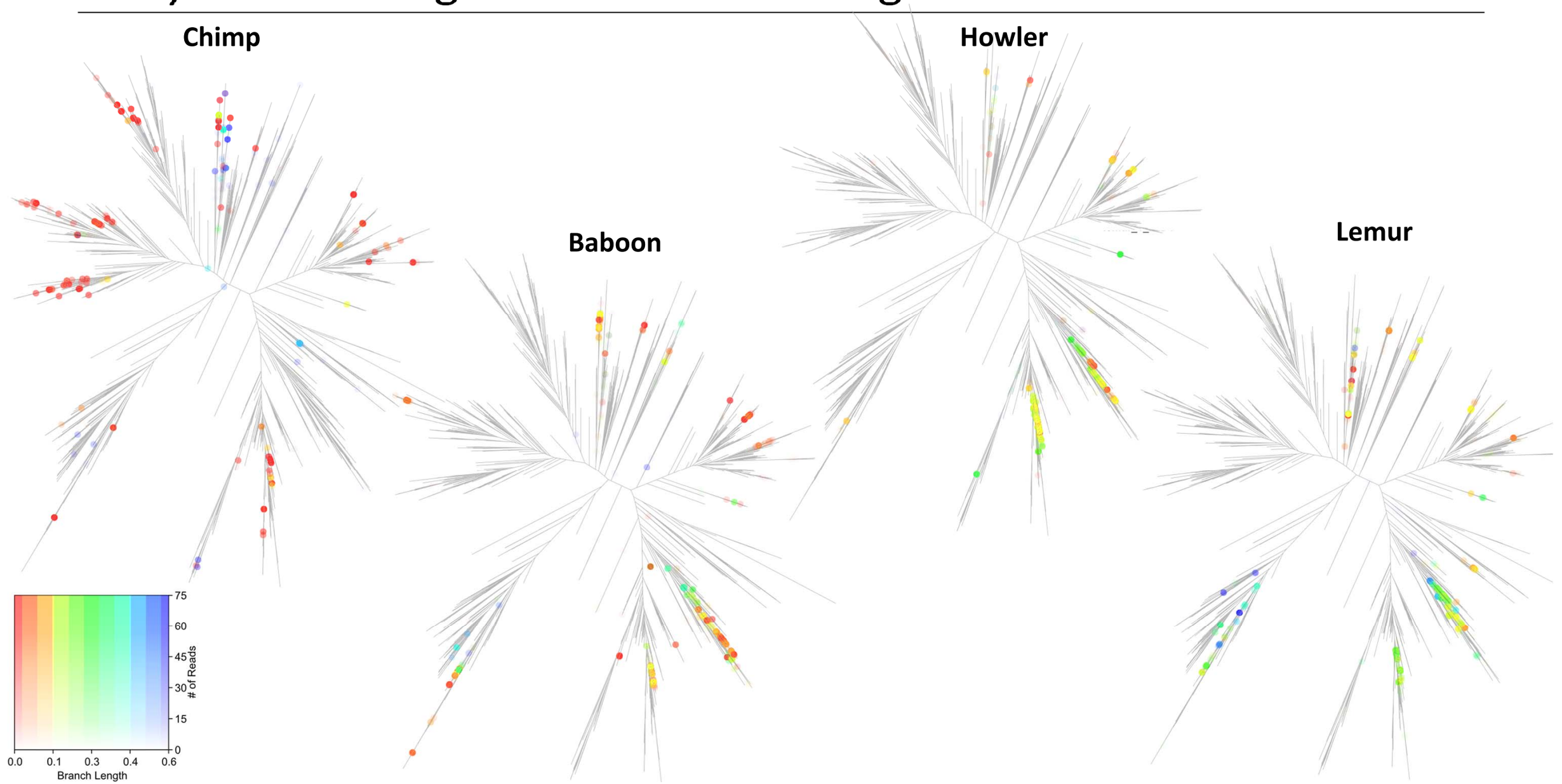
SEPP (or TIPP): Interpreting Branch Length

- For each read, SEPP gives the best ***new branch*** in the tree.

- Includes:
  1. Attachment Point
     - *Attaches near the closest matching sequence(s) in the alignment.*
  2. Branch Length
     - *How far it is from its closest match in the database.*
     - *In some sense, a measure of* ***novelty****.*



Why Branch Length Can be Interesting



TIPP: Pipeline for Shotgun Data

- For future reference (already loaded on STAMPS server):
  - TIPP Reference Package Download: (TBA)
- For shotgun data, use 40 single-copy marker genes as references:
  - First step: use BLAST to filter out reads that hit any of the 40.
  - Second step: use TIPP to place into the taxonomy & classify
  - Third step: summarize
- run_abundance.py script executes the entire pipeline. (NOTE: this will mean up to 40 separate TIPP runs, so it may be best to divide the inputs.)
- Sample command:

```
python3 $sepppath/run_abundance.py \
      -c $sepppath/.sepp/tipp.config \
      -G cogs \
      -A 10 -P 2500 \
      -at 0.00 -pt 0.00 \
      -f <input_seqs> \
      -d <output_folder> -o <job_name> -p <temp_dir>
```

TIPP: Pipeline for Shotgun Data (continued)

- First output from run_abundance.py is BLAST results.
- To re-start, or re-run on old BLAST results: add ‘-b’ option:

```
python3 $sepppath/run_abundance.py \
      -c $sepppath/.sepp/tipp.config \
      -G cogs \
      -A 10 -P 2500 \
      -at 0.00 -pt 0.00 \
      -f <input_seqs> \
      -d <output_folder> -o <job_name> -p <temp_dir> \
      -b <blast_output_file>
```

TIPP Reference Package Contents

File Name	Description
all_taxon.taxonomy	*(used internally within TIPP)*
original_sequence_name_map.txt	Sequences in reference data get renamed to avoid issues with special characters. This is the mapping to their original names from NCBI.
pasta.fasta	Multiple sequence alignment of reference sequences.
pasta.hmm	*(used internally within TIPP)*
pasta.size	*(used internally within TIPP)*
pasta.taxonomy	Refined taxonomy on reference sequences. Conforms to the taxonomy described by "taxonomy.table".
pasta.taxonomy.RAxML_info	RAxML info file for the refined taxonomy "pasta.taxonomy"
pasta.tree	Phylogeny estimated on reference sequences. Does not conform to the taxonomy.
pasta.tree.RAxML_info	RAxML info file for the phylogeny in "pasta.tree"
species.mapping	Tab delimited file with sequence name and NCBI Taxomy ID for all sequences in the reference alignment.
species.txt	*(used internally within TIPP)*
taxonomy.table	Taxonomy structure for taxa covered by reference alignment.

Beta versions of updated TIPP reference packages are located at: https://www.dropbox.com/s/98r7r9ccf1zgyqt/tipp-2017.zip

SEPP Command Line Tips

Use command `run_sepp.py –h` to show help and all command line options

Categories:

- Decomposition Options:
  - It is ok to leave these all blank
  - SEPP has sensible defaults
- Input Options:
  - **Mandatory**:
    - Input Tree (-t)
    - Input Alignment (-a)
    - RAxML model file (-r)
    - Query sequences, a.k.a. fragment file (-f)
- Output Options:
  - Good idea to specify all of these:
  - Temp directory (-p)
    - Needs to be cleaned out periodically.
  - Output folder (-d)
    - (Good practice to know where output will go)
  - Output prefix (-o)
    - Also good practice, to avoid confusing one run with another.

TIPP Command Line Tips:

Use command `run_tipp.py –h` to show help and all command line options

Categories:

- Decomposition Options:
  - (Same as SEPP, leaving blank is fine)
  - Can affect output and running time though.
- Input Options:
  - **Two Options**:
    - Specify same inputs as SEPP (on previous page)
    - Specify a reference package (-R)
      - Must be in the folder "$REFERENCE" at install time.
- TIPP-specific options:
  - Taxonomy (-tx): called "taxonomy.table" in the reference packages
  - Taxonomy-Name Mapping (-txm): called "species.mapping" in the reference packages
- Another Option: use run_abundance.py to implement the full pipeline on a set of shotgun data.

Additional Resources

- GitHub repository:
  - https://github.com/smirarab/sepp
  - Maintained by Siavash Mirarab, may not always have latest development (but well-tested and stable)
- TIPP Reference Packages:
  - https://www.dropbox.com/s/98r7r9ccf1zgyqt/tipp-2017.zip
- SEPP Visualization (warning: very preliminary):
  - https://github.com/MGNute/phylogenetic_histograms
- Contact the TIPP developers:
  - Tandy Warnow (warnow@Illinois.edu)
  - Mike Nute (nute2@Illinois.edu)
  - Siavash Mirarab (smirarabbaygi@eng.ucsd.edu)