Title:

Applications of Tree-Sequence Recording in SLiM 3
Tree-Sequence Recording with Practical Applications in SLiM 3
Practical Applications of Tree-Sequence Recording using SLiM 3 and msprime
Accelerating Forward Simulations with Tree-Sequence Recording in SLiM 3
Applied Tree-Sequence Recording in SLiM 3
...

Authors: Benjamin C. Haller, Jared Galloway, Jerome Kelleher, Philip W. Messer, Peter L. Ralph

We'll also do an "author contributions" bit so it's clear what everybody did, we've decided.
It's looking like I'll probably be first author on the meat of the paper (II and III below), but Philipp might write the introduction and/or discussion.

Target journal:

We have decided to pretend that our target is Molecular Biology & Evolution (MBE) for now, but Philipp points out that that may not be a great choice since he and I have an invited paper and a regular paper both already planned for them in the next couple of months, both SLiM-related

Other suggestions:
Molecular Ecology Resources, Genetics, Bioinformatics, MBE, PLoS Computational Biology

Abstract:

Forward simulations are increasingly important in popgen / ecology / evo, and there is an increasing desire to run large simulations involving full genome sequences, large populations, complex demography, realistic ecology, explicit space, spatial interactions, etc. However, such models can take an extremely long time to run – often so long that it is not possible to model the things we would like to model. This situation presents an obstacle to the field, and although increases in computing performance, etc., are gradually increasing our capabilities, there is a need for more rapid progress. The tree-sequence recording method of Kelleher et al. 2018 provides a path forward. This method allows neutral mutations to be omitted from forward-time simulations and added later, vastly more efficiently; it also allows more rapid computation of some key population genetics statistics along the chromosome, and provides a compact representation of a population's ancestry that can be analyzed in Python. We have extended this method to allow the recording of non-neutral mutations in the tree sequence during forward-time simulation, greatly broadening the applicability of the method. This extended method is now available in SLiM 3. We here present several examples of the practical application of this method, which we believe opens up whole new horizons of possibility in the modeling of evolutionary processes.

Outline:

I. Introduction (http://ecoevoevoeco.blogspot.com/2014/10/how-to-writepresent-science-baby.html)
    A. Baby
        i. Forward simulations are increasingly important in popgen / ecology / evo
        ii. Increasing desire to model full chromosomes, large populations, complex ecology
    B. Werewolf
        i. These models often take a very long time to run (memory usage too?)
        ii. This is an obstacle to the field, preventing realistic modeling
    C. Bullet

       i. The tree-sequence recording method of Kelleher et al. 2018 provides a way forward
       ii. We have extended this method to allow the recording of mutations during simulation
       iii. This extended method is now available in SLiM 3, free and open-source
       iv. We here present several examples of the practical application of this method

II. Methods
   A. Model execution details
       i. Software: SLiM version, msprime version, etc.; hopefully released versions by then
       ii. Hardware: my laptop's specs (I don't see any need for a cluster here)
   B. Test cases
       i. Simple neutral model: description, SLiM code, parameters
           a. Methods for running in SLiM, with and without neutral mutations
           b. Methods for adding neutral mutations to the tree in msprime
       ii. Two-locus beneficial mutation sweep: ditto
       iii. Background selection, fast stats computation in Python: ditto, plus stats
       iv. True local ancestry mapping with a two-subpop admixture model (like recipe 13.9)
           a. Methods for running the two-subpop model in SLiM and producing output
           b. Loading into Python, as in II.B.iii
           c. Assessing the mean local ancestry (pop 1 or pop 2 origin) at each position
           d. Producing a plot of that, as a practical example

III. Results
   A. Simple neutral model: comparison of total execution time
       i. Raw SLiM execution time with simulation of all neutral mutations
       ii. Total time for SLiM without mutations, plus msprime overlay of mutations
       iii. Time for the equivalent operation using the coalescent
   B. Two-locus beneficial mutation sweep: time comparison
       i. Raw SLiM execution time with simulation of all neutral mutations
       ii. Total time for SLiM without neutral mutations, plus msprime overlay
   C. Background selection: time comparison
       i. Raw SLiM execution time with simulation of all neutral mutations
       ii. Total time for SLiM without neutral mutations, plus msprime overlay
       iii. Computed statistics compared between the two cases to show correctness
   D. True local ancestry mapping: exposition of method
       i. Running the model in SLiM
       ii. Loading it into Python
       iii. The plot of the mean true local ancestry along the chromosome

IV. Discussion
   A. Recapitulation of the baby/werewolf/bullet from Introduction
   B. Performance implications
       i. Speedups are large
       ii. Memory usage comparison is whatever it is
   C. Utility implications
       i. Being able to load simulation results, with ancestral tree sequences, into Python
       ii. Fast statistics computation; more work will be published here soon
       iii. Fast calculation of ancestry-based metrics like true local ancestry in Python
       iv. Other possible manipulation / analysis in Python without difficult output file parsing
   D. Conclusion
       i. This new method makes it possible to run large models that were prohibitively expensive
       ii. We believe this will open up new and important areas of exploration
       iii. The software is available now for download and use (hopefully by then)