Commands, parameters, and version numbers of programs to reproduce data simulation and phylogenetic analyses

1. Setting up the environment

1.1 Clone the GitHub repository for this paper and set the REPO variable.

```
git clone https://github.com/josephryan/Hernandez_Ryan_2021_Recoding

cd Hernandez_Ryan_2021_Recoding

export REPO=$(pwd)
```

1.2 Edit Servers.pm

The following commands run on a single modern processor would take more than 6 years to complete. We used 4 servers with more than 300 CPUs between them. To distribute jobs between servers, we developed scripts that will generate individual shell scripts. We provide commands to run these individual scripts on different servers. To distribute the jobs on multiple servers (or multiple CPUs on a single server), the following file must be edited.

$REPO/01-MODULES/Servers.pm

The following external programs are expected to be installed and in $PATH. The versions applied to our analyses are listed in parentheses.

Python (version 2.7.16)
P4 (version 1.2.0)
ete3 (version 3.1.1)
six (version 1.12.0)
RAxML (version 8.2.11)
Seq-Gen (version 1.3.2)
TOPD/FMTS (version 4.6)
R (version 3.3.1) - ggplot2 package (version 2.2.1)
RStudio (version 1.2.1335) - ggplot2 package (version 3.1.1)

All R scripts must be run in RStudio within their respective directories.

2. Testing Compositional Heterogeneity

2.1 Simulate datasets for null distribution (no induced compositional heterogeneity)

2.2.1 In the directory `02-COMPOSITIONAL_HETEROGENEITY/01-NULL_DISTRIBUTION/01-TREE0008`, make the directory 'scripts' and run the following command to generate several shell scripts that perform 1,000,000 simulations.

```
mkdir scripts

perl ../split_random_runs.pl TREE0008
```

2.2.2 The following command will execute the shell scripts to simulate amino acid data on hypothetical tree 0.008 (Figure 1A) under the GTR model with amino acid frequencies and transition rates estimated from the Chang dataset. Shell scripts will generate simulated sequences in PHYLIP format within each output directory, p4 scripts (p4.x) that were used to perform simulations, and ".out" files with comp-het index values for each simulated dataset.

```
ls -1 scripts | grep 'myserver*' | perl -ne 'chomp; print "sh scripts/$_
&\n";' | sh
```

2.2.3 Concatenate all ".out" files. The comp-het index values in these files are used as a null distribution to statistically compare to the comp-het indices from datasets that were simulated under gradients of compositional heterogeneity.

```
cat *.out >> all.out
```

2.3 Simulate compositionally heterogeneous datasets

2.3.1 In the directory `02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/01-TREE0008`, run the following commands to produce datasets with increasing compositional heterogeneity. The script `comphet.pl` prints out the comp-het index value for the simulated dataset and its associated p-value. The p-value is calculated by comparing the comp-het index of the simulated dataset to the null distribution of comp-het indices.

```
cd 01-TREE0008.1

perl ../../comphet.pl TREE0008 0.1 1000 ALPHABETICAL all.out > pvals.0.1.out

cd 02-TREE0008.5

perl ../../comphet.pl TREE0008 0.5 1000 ALPHABETICAL all.out > pvals.0.5.out

cd 03-TREE0008.9

perl ../../comphet.pl TREE0008 0.9 1000 ALPHABETICAL all.out > pvals.0.9.out
```

2.4 Relationship between length of the stem branches of clade-A and clade-B, as well as clade-C and clade-D, and compositional heterogeneity

We looked at the relationship between the lengths of the stem branches of clade-A and clade-B, as well as clade-C and clade-D for each tree (highlighted in red in Fig. 1A), and the comp-het index values to determine how branch lengths were impacting compositional heterogeneity.

2.4.1 Run the script `comphet_index_scatter.R` in RStudio to produce a scatter plot showing the relationship between the comp-het index values for each inflation parameter tested and the length of the stem branches of clade-A and clade-B, as well as clade-C and clade-D of each tree. For each inflation parameter, this script also calculates a linear regression for these relationships.

2.5 Recode datasets and create maximum-likelihood analysis shell scripts

2.5.1 In each of the directories `01-TREE0008.1`, `02-TREE0008.5`, and `03-TREE0008.9` make the directory 'scripts' and run the script `chunkify_comphet_trees.pl` to recode the compositionally heterogeneous dataset using Dayhoff 6-state recoding and S&R 6-state

recoding. This script also generates shell scripts to perform maximum-likelihood analyses in RAxML.

```
mkdir scripts
```

```
perl ../../chunkify_comphet_trees.pl
```

2.6 Phylogenetic reconstructions

2.6.1 Run the following command in each of the directories `01-TREE0008.1`, `02-TREE0008.5`, and `03-TREE0008.9` to execute the shell scripts generated from `chunkify_comphet_trees.pl` to perform maximum-likelihood analyses on both recoded and non-recoded datasets. Recoded datasets were reconstructed under the MULTIGAMMA multi-state model with GTR and non-recoded datasets were reconstructed under the Dayhoff and JTT models.

```
ls -1 scripts/ | grep 'myserver*' | perl -ne 'chomp; print "sh
scripts/$_&\n";' | sh
```

2.7 Score trees

2.7.1 Run the following command in each of the directories `01-TREE0008.1`, `02-TREE0008.5`, and `03-TREE0008.9` to determine the proportion of trees that do not correctly reconstruct a monophyletic group including taxa from clade-A and clade-B and a monophyletic group including taxa from clade-C and clade-D. The output file identifies which trees were incorrectly reconstructed and indicates the proportion of incorrect trees produced under each method (recoding and non-recoding).

```
perl ../../is_mono.pl DAYHOFF > is_mono_dayhoff.out
```

```
perl ../../is_mono.pl JTT > is_mono_jtt.out
```

2.8. Perform 2.1 – 2.7 using hypothetical trees 0.004, 0.002, and 0.001 (Figure 1A).

2.8.1 Repeat analyses 2.1-2.7 replacing TREE0008 with TREE0004, TREE0002, and TREE0001.

2.9 Statistics and visualization of scores

2.9.1 Run the script `comphet_bargraph.R` in RStudio to generate a bar graph showing the percentage of incorrect trees reconstructed under each method, tree used for simulation, and level of compositional heterogeneity tested.

2.9.2. Run the script `deepsplits_chisq.R` in RStudio to perform the chi-squared test on the number of incorrect trees reconstructed under recoding and non-recoding methods for each tree used for simulation and level of compositional heterogeneity tested.

2.9.3. In the directory `02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/`, run the following commands to calculate the Robinson-Foulds distance (RFD) values between the trees used for simulation and the recoded and non-recoded trees under each level of compositional heterogeneity tested. The script `rfd_comphet.pl` performs t-tests in R to determine if there are significant differences in RFDs between recoded and non-recoded trees, as well as generate boxplots to visualize the data.

```
cd 01-TREE_0008

perl ../rfd_comphet.pl TREE0008 DAYHOFF > rfd_dayhoff.out

perl ../rfd_comphet.pl TREE0008 JTT > rfd_jtt.out

cd 02-TREE_0004

perl ../rfd_comphet.pl TREE0004 DAYHOFF > rfd_dayhoff.out

perl ../rfd_comphet.pl TREE0004 JTT > rfd_jtt.out

cd 03-TREE0002

perl ../rfd_comphet.pl TREE0002 DAYHOFF > rfd_dayhoff.out

perl rfd_comphet.pl TREE0002 JTT > rfd_jtt.out

cd 04-TREE0001

perl rfd_comphet.pl TREE0001 DAYHOFF > rfd_dayhoff.out

perl rfd_comphet.pl TREE0001 JTT > rfd_jtt.out
```

3. Testing Saturation

3.1 Testing the association between the branch length scaling factor parameter and saturation

We performed a simulation experiment with a six-taxa bifurcating tree (test.tre) and ran five instances of Seq-Gen with incrementing “branch length scaling factor” parameters (i.e., 1.0, 2.0, 3.0, 4.0, and 5.0) and the "write ancestral sequences for each node" parameter (i.e., -w a) set, which printed ancestral sequences at each node. We then calculated the number of saturation events (i.e., the number of times a change in an amino acid led to the appearance of an amino acid that had been in that same position in a prior ancestral node) using the script `seq-gen_saturation_test.pl`.

3.1.1 In the directory `03-SATURATION/01-SATURATION_TEST`, run `seq-gen_saturation_test.pl` to perform the simulations outlined above.

```
perl seq-gen_saturation_test.pl
```

3.2 Simulations

3.2.1 In the directory `03-SATURATION/02-SEQ_GEN_CHANG/01-DAYHOFF`, use the following shell scripts to reproduce the simulations we performed in Seq-Gen using the Chang topology, the PAM250 (Dayhoff) model, and the branch length scaling factor parameters described in the manuscript. These scripts will generate a total of 20,000 datasets, create new directories for each topology/model/branch length scaling factor parameter tested, and parse the datasets into separate PHYLIP files within their corresponding directories.

```
sh run_seqgen_pam_chang.sh

sh divide_chang_pam.sh
```

3.2.2 Run the following shell scripts in the directory `03-SATURATION/02-SEQ_GEN_CHANG/02-JTT` to perform equivalent simulations under the JTT model.

```
sh run_seqgen_jtt_chang.sh

sh divide_chang_jtt.sh
```

3.2.3 The following commands and shell scripts reproduce the simulations we performed using the Feuda topology, the PAM250 model, and the branch length scaling factor parameters described in the manuscript. These scripts will generate a total of 20,000 datasets, create new directories, and parse data as in 3.2.1. Run these commands in the directory `03-SATURATION/03-SEQ_GEN_FEUDA/01-DAYHOFF`.

```
sh run_seqgen_pam_feuda.sh

sh divide_feuda_pam.sh
```

3.2.4 Running the following shell scripts in the directory `03-SATURATION/03-SEQ_GEN_FEUDA/02-JTT` performs equivalent simulations using the Feuda topology and the JTT model.

```
sh run_seqgen_jtt_feuda.sh

sh divide_feuda_jtt.sh
```

3.3 Recode datasets

3.3.1 In the directory `03-SATURATION`, create a new directory called ‘scripts.’ Run `chunkify.pl` to convert simulated datasets to Dayhoff 6-state recoded and S&R 6-state recoded datasets and generate shell scripts to perform maximum-likelihood analyses in RAxML.

```
mkdir scripts

perl chunkify.pl
```

3.4 Phylogenetic reconstructions

3.4.1 Execute the scripts generated from `chunkify.pl` to perform maximum-likelihood analyses on both recoded and non-recoded datasets. Recoded datasets were reconstructed under the MULTIGAMMA multi-state model with GTR and non-recoded datasets were reconstructed under the Dayhoff model.

```
ls -1 scripts | grep 'myserver*’ | perl -ne 'chomp; print "sh scripts/$_
&\n";' | sh
```

3.5. Score trees

3.5.1 Run the following script to calculate RFD values between the tree used for simulation and the recoded and non-recoded trees for each branch length scaling factor parameter tested. The script will also perform t-tests in R to determine if there are significant differences in RFDs between recoded and non-recoded trees, as well as generate box-plots to visualize the data.

```
perl compare_trees_to_sim.pl
```

3.6 Line graph of median RFDs for non-recoded and recoded datasets

2.6.1 Run the script `robinson_foulds_medians_fig.R` in RStudio to generate a line graph displaying median Robinson-Foulds distances for each tree, model, and branch length scaling factor parameter tested.

3.7 Phylogenetic reconstructions under the LG model

3.7.1 Datasets simulated over the Chang tree with branch length scaling factors 1, 5, 10, 15, and 20 were reconstructed using the LG model. To reproduce this analysis, in the directory `03-SATURATION` make a new directory called 'LG_scripts'. Run the script `chunkify_LG.pl` to generate shell scripts for phylogenetic reconstructions in RAxML.

```
mkdir LG_scripts
```

```
perl chunkify_LG.pl
```

3.7.2 Use the following command to execute the shell scripts generated from `chunkify_LG.pl` that perform maximum-likelihood analyses using the LG model.

```
ls -1 scripts | grep 'myserver*' | perl -ne 'chomp; print "sh scripts/$_
&\n";' | sh
```

3.8 Scoring and comparing trees reconstructed with the LG model

3.8.1 Run `compare_trees_to_sim_LG.pl` to score trees reconstructed under the LG model using the RFD in TOPD/FMTS. This script performs t-tests to compare the RFD values between the trees reconstructed with LG and those reconstructed with 6-state recoding. Box-plots are also generated to visualize the data.

```
perl compare_trees_to_sim_LG.pl
```

3.9 Simulating under GTR with estimated parameters from the Chang dataset

3.9.1 Use the following shell script in the directory `03-SATURATION/04-ESTIMATED_MODEL` to reproduce the simulations we performed in Seq-Gen using the Chang topology, amino acid rates of substitution, amino acid frequencies, and gamma rate heterogeneity estimated from the Chang dataset. Simulations were performed with branch length scaling factors 1, 5, 10, 15, and 20.

```
sh run_seqgen_estimated_model.sh
```

3.9.2 Run the following shell script to create directories for each branch length scaling factor parameter tested and parse the datasets into separate PHYLIP files within their corresponding directories.

```
sh divide_estimate.sh
```

3.9.3 Create a new directory called 'scripts' and convert the simulated datasets to Dayhoff 6-state recoded datasets using the script `chunkify_estimate.pl`. `chunkify_estimate.pl` also generates shell scripts for performing maximum-likelihood analyses in RAxML.

```
mkdir scripts
```

```
perl chunkify_estimate.pl
```

3.9.5 Execute the shell scripts generated from `chunkify_estimate.pl` to perform maximum-likelihood analyses on both recoded and non-recoded datasets. We performed maximum-likelihood analyses using the MULTIGAMMA multi-state model with GTR for recoded datasets and the Dayhoff model for non-recoded datasets.

```
ls -1 scripts | grep 'myserver*' | perl -ne 'chomp; print "sh scripts/$_
&\n";' | sh
```

3.9.6 Run `compare_trees_to_sim_estimate.pl` to calculate RFD values in TOPD/FMTS for trees produced from recoded and non-recoded datasets and perform t-tests to compare the RFD values between both approaches. This script also generates box-plots to visualize the data.

```
perl compare_trees_to_sim_estimate.pl
```

4. Testing Alternative Recoding Strategies on Compositional Heterogeneity

4.1 Test alternative recoding schemes for optimization

4.1.1 In the directory `02-COMPOSITIONAL_HETEROGENEITY/03-ALTERNATIVE_RECODING`, run the following command on each recoding scheme outlined in Table S1. The script will print a score for each binning strategy specified. Higher scores indicate improved optimization of the Dayhoff matrix.

```
perl score.pl 'AMINO ACID BINNING SCHEME'
```

4.2 Recode datasets

4.2.1 Recode the data generated in section 1.3.1 under hypothetical tree 0.002 and inflation parameters 0.1, 0.5, and 0.9 using the best scoring 9-, 12-, 15-, and 18-state binning strategies (Table 2). Perform the following commands in each of the listed directories within `02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/03-TREE0002`: `01-TREE0002.1`, `02-TREE0002.5`, and `03-TREE0002.9`. Run `chunkify_nscheme_recoding.pl` to generate shell scripts for maximum-likelihood analyses in RAxML using the MULTIGAMMA multi-state model with GTR.

```
mkdir 9scheme_scripts

mkdir 12scheme_scripts

mkdir 15scheme_scripts

mkdir 18scheme_scripts

perl ../../../03-ALTERNATIVE_RECODING/ chunkify_nscheme_recoding.pl 9
9scheme_scripts

perl ../../../03-ALTERNATIVE_RECODING/ chunkify_nscheme_recoding.pl 12
12scheme_scripts

perl ../../../03-ALTERNATIVE_RECODING/ chunkify_nscheme_recoding.pl 15
15scheme_scripts

perl ../../../03-ALTERNATIVE_RECODING/ chunkify_nscheme_recoding.pl 18
18scheme_scripts
```

4.3 Phylogenetic reconstructions

4.3.1 Execute the following commands in directories `01-TREE0002.1`, `02-TREE0002.5`, and `03-TREE0002.9` to run the shell scripts generated from `chunkify_nscheme_recoding.pl` that perform maximum-likelihood analyses on recoded datasets.

```
ls -1 9scheme_scripts | grep 'myserver*'| perl -ne 'chomp; print "sh
9scheme_scripts/$_ &\n";' | sh

ls -1 12scheme_scripts | grep 'myserver*'| perl -ne 'chomp; print "sh
12scheme_scripts/$_ &\n";' | sh

ls -1 15scheme_scripts | grep 'myserver*'| perl -ne 'chomp; print "sh
15scheme_scripts/$_ &\n";' | sh

ls -1 18scheme_scripts | grep 'myserver*'| perl -ne 'chomp; print "sh
18scheme_scripts/$_ &\n";' | sh
```

4.4 Score trees

4.4.1 Run the following commands in directories `01-TREE0002.1`, `02-TREE0002.5`, and `03-TREE0002.9` to determine what proportion of trees do not correctly reconstruct a monophyletic group of taxa from clade-A and clade-B and a monophyletic group of taxa from clade-C and clade-D. The output file identifies incorrectly reconstructed trees and calculates the proportion of incorrect trees produced.

```
perl ../../../02-HETEROGENEOUS_DATA/is_mono.pl DAYHOFF.9 >
is_mono_dayhoff9.out

perl ../../../02-HETEROGENEOUS_DATA/is_mono.pl DAYHOFF.12 >
is_mono_dayhoff12.out
```

```
perl ../../../02-HETEROGENEOUS_DATA/is_mono.pl DAYHOFF.15 >
is_mono_dayhoff15.out

perl ../../../02-HETEROGENEOUS_DATA/is_mono.pl DAYHOFF.18 >
is_mono_dayhoff18.out
```

4.5 Generate a bar graph for percentage of incorrect trees

4.5.1 Run the script `comphet_bargraph_nscheme.R` in RStudio to generate a bar graph showing the percentage of incorrect trees reconstructed under each alternative recoding strategy, as well as under Dayhoff 6-state recoding, and non-recoding using the Dayhoff model.

4.6 Chi-squared test comparing Dayhoff-18 recoding to non-recoding

4.6.1 Run the script `deepsplits_chisq_dayhoff_v_dayhoff18.R` in RStudio to perform the chi-squared test on the number of incorrect trees reconstructed under the Dayhoff model with non-recoding and the best scoring Dayhoff 18-state recoding strategy.

Post-review Supplementary analyses

1. Testing Dayhoff 6-state recoding under no induced compositional heterogeneity

1.1 Recode datasets and create maximum-likelihood analysis shell scripts

A reviewer requested that we test recoding on datasets that included no added compositional heterogeneity.

1.1.1 We selected the first 1,000 datasets (directory `7560001`) out of the 1,000,000 generated for the null distribution on hypothetical tree 0.002. We created the directory 'scripts' and ran the script `chunkify_comphet_trees.pl` to recode the data using Dayhoff 6-state and S&R-6 state recoding. The script `chunkify_comphet_trees.pl` also generates shell scripts for maximum-likelihood analyses in RAxML.

```
cd Hernandez_Ryan_2021_Recoding/02-COMPOSITIONAL_HETEROGENEITY/01-
NULL_DISTRIBUTION/03-TREE0002/7560001/

mkdir scripts

perl ../../../chunkify_comphet_trees.pl
```

1.2 Phylogenetic reconstructions

1.2.1 Execute the scripts generated from chunkify_comphet_trees.pl to perform maximum-likelihood analyses on both recoded and non-recoded datasets. Recoded datasets were reconstructed under the MULTIGAMMA multi-state model with GTR and non-recoded datasets were reconstructed under the Dayhoff model.

```
ls -1 scripts | grep 'myserver*' | perl -ne 'chomp; print "sh scripts/$_
&\n";' | sh
```

1.3 Score trees

1.3.1 Run the following command to determine the proportion of trees that do not correctly reconstruct a monophyletic group including taxa from clade-A and clade-B and a monophyletic group including taxa from clade-C and clade-D. The output file identifies which trees were incorrectly reconstructed and indicates the proportion of incorrect trees produced under each method (recoding and non-recoding).

```
perl ../../../02-HETEROGENEOUS_DATA/is_mono.pl DAYHOFF > is_mono_dayhoff.out

perl ../../../02-HETEROGENEOUS_DATA/is_mono.pl JTT > is_mono_jtt.out
```

1.4 Bar graph

1.4.1 Run the script `tree0.002_bargraph.R` in RStudio to generate a bar graph showing the percentage of incorrect trees reconstructed under each method and level of compositional heterogeneity tested (no added compositional heterogeneity compared to adjusted levels of compositional heterogeneity).

2. Testing 6-state Recoding Performance on Data Size

2.1 Varying data size simulations of compositionally heterogeneous datasets

2.1.1 In directory `04-DATA_SIZE/01-COMPOSITIONAL_HETEROGENEITY`, run the following commands to produce datasets with increasing compositional heterogeneity made up of 2,000, 3,000, 4,000, and 5,000 amino acid columns. The script `comphet.pl` prints out the comp-het index value for the simulated dataset and its associated p-value. However, for this analysis we did not compare the dataset to a null distribution (all.out includes only the number 0 and was used to meet the parameter settings to run the script), therefore the use of the p-value is not appropriate.

```
cd 01-TREE0002.1.2000

perl ../../../02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/comphet.pl
TREE0002 0.1 2,000 ALPHABETICAL all.out

cd 02-TREE0002.5.2000

perl ../../../02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/comphet.pl
TREE0002 0.5 2,000 ALPHABETICAL all.out

cd 03-TREE0002.9.2000

perl ../../../02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/comphet.pl
TREE0002 0.9 2,000 ALPHABETICAL all.out

cd 04-TREE0002.1.3000

perl ../../../02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/comphet.pl
TREE0002 0.1 3000 ALPHABETICAL all.out

cd 05-TREE0002.5.3000

perl ../../../02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/comphet.pl
TREE0002 0.5 3000 ALPHABETICAL all.out

cd 06-TREE0002.9.3000

perl ../../../02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/comphet.pl
TREE0002 0.9 3000 ALPHABETICAL all.out

cd 07-TREE0002.1.4000

perl ../../../02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/comphet.pl
TREE0002 0.1 4000 ALPHABETICAL all.out

cd 08-TREE0002.5.4000

perl ../../../02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/comphet.pl
TREE0002 0.5 4000 ALPHABETICAL all.out

cd 09-TREE0002.9.4000
```

```
perl ../../../02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/comphet.pl
TREE0002 0.9 4000 ALPHABETICAL all.out

cd 10-TREE0002.1.5000

perl ../../../02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/comphet.pl
TREE0002 0.1 5,000 ALPHABETICAL all.out

cd 11-TREE0002.5.5000

perl ../../../02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/comphet.pl
TREE0002 0.5 5,000 ALPHABETICAL all.out

cd 12-TREE0002.9.5000

perl ../../../02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/comphet.pl
TREE0002 0.9 5,000 ALPHABETICAL all.out
```

2.1.2 Make a new directory called 'scripts.' Convert all simulated datasets to Dayhoff 6-state recoded datasets and generate shell scripts to perform maximum-likelihood analyses in RAxML using the script `chunkify_estimate.pl`.

```
mkdir scripts

perl ../../03-SATURATION/04-ESTIMATED_MODEL/chunkify_estimate.pl
```

2.1.3 Execute the shell scripts generated from `chunkify_estimate.pl` to perform maximum-likelihood analyses on both recoded and non-recoded datasets. We performed maximum-likelihood analyses using the MULTIGAMMA multi-state model with GTR for recoded datasets and the Dayhoff model for non-recoded datasets.

```
ls -1 scripts | grep 'myserver*' | perl -ne 'chomp; print "sh scripts/$_
&\n";' | sh
```

2.1.4 To determine the proportion of trees that do not correctly reconstruct a monophyletic group including taxa from clade-A and clade-B and a monophyletic group including taxa from clade-C and clade-D, run the following command in each of the following subdirectories in directory `04-DATA_SIZE/01-COMPOSITIONAL_HETEROGENEITY`: `01-TREE0002.1.2000`, `02-TREE0002.5.2000`, `03-TREE0002.9.2000`, `04-TREE0002.1.3000`, `05-TREE0002.5.3000`, `06-TREE0002.9.3000`, `07-TREE0002.1.4000`, `08-TREE0002.5.4000`, `09-TREE0002.9.4000`, `10-TREE0002.1.5000`, `11-TREE0002.5.5000`, `12-TREE0002.9.5000`.

```
perl ../../../02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/is_mono.pl
DAYHOFF > is_mono.out
```

2.1.5 Run the script `comphet_bargraph_size.R` in RStudio to generate a bar graph showing the percentage of incorrect trees reconstructed under each method, data size, and level of compositional heterogeneity tested.

2.2 Varying data size simulations of saturated datasets

2.2.1 In the `04-DATA_SIZE/02-SATURATION`, use the following shell scripts to reproduce the simulations we performed in Seq-Gen for datasets made up of 2,000 and 5,000 amino acid columns using the Chang topology, the PAM250 (Dayhoff) model, and branch length scaling factor parameters set to 1, 5, 10, 15, and 20. Create new directories for each dataset size and branch length scaling factor parameter tested, and parse the datasets into separate PHYLIP files within their corresponding directories.

```
sh run_seqgen_pam_size.sh
```

```
sh divide_size.sh
```

2.2.2 Convert all simulated datasets to Dayhoff 6-state recoded datasets using the script `chunkify_estimate.pl`. `chunkify_estimate.pl` will also generate shell scripts for maximum-likelihood analyses in RAxML.

```
perl ../../03-SATURATION/04-ESTIMATED_MODEL/chunkify_estimate.pl
```

2.2.3 Execute the shell scripts generated from `chunkify_estimate.pl` to perform maximum-likelihood analyses on both recoded and non-recoded datasets. We performed maximum-likelihood analyses using the MULTIGAMMA multi-state model with GTR for recoded datasets and the Dayhoff model for non-recoded datasets.

```
ls -1 scripts/ | grep 'myserver*' | perl -ne 'chomp; print "sh scripts/$_
&\n";' | sh
```

2.2.3 Run the following script to calculate RFDs between the tree used for simulation (i.e., Chang tree) and the recoded and non-recoded trees for each dataset size and branch length scaling factor parameter tested. This script will also perform t-tests in R to determine if there are significant differences in RFDs between recoded and non-recoded trees, as well as generate box-plots to visualize the data.

```
perl compare_../../03-SATURATION/04-
ESTIMATED_MODEL/compare_trees_to_sim_estimate.pl
```

3. Testing 6-state Recoding Performance on Tree Shape and Compositional Heterogeneity

3.1 Simulate compositionally heterogeneous datasets on an asymmetrical tree

3.1.1 In directory `05-TREE_SHAPE`, run the following commands to produce compositionally heterogeneous datasets on the Chang topology. The script `comphet.pl` prints out the comp-het index value for the simulated dataset and its associated p-value. However, for this analysis we did not compare the dataset to a null distribution (all.out includes only the number 0 and was used to meet the parameter settings to run the script), therefore the use of the p-value is not appropriate.

```
cd 01-TREE000C_1_1000/

perl ../../02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/comphet.pl
TREE000C 0.1 1000 ALPHABETICAL all.out

cd 02-TREE000C_5_1000/

perl ../../02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/comphet.pl
TREE000C 0.5 1000 ALPHABETICAL all.out

cd 03-TREE000C_9_1000/

perl ../../02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/comphet.pl
TREE000C 0.9 1000 ALPHABETICAL all.out
```

3.1.2 Make a new directory called 'scripts.' Convert all simulated datasets to Dayhoff 6-state recoded datasets and generate shell scripts to perform maximum-likelihood analyses in RAxML using the script chunkify_estimate.pl.

```
mkdir scripts

perl ../03-SATURATION/04-ESTIMATED_MODEL/chunkify_estimate.pl
```

3.1.3 Execute the shell scripts generated from chunkify_estimate.pl to perform maximum-likelihood analyses on both recoded and non-recoded datasets. We performed maximum-likelihood analyses using the MULTIGAMMA multi-state model with GTR for recoded datasets and the Dayhoff model for non-recoded datasets.

```
ls -1 scripts/ | grep 'myserver*' | perl -ne 'chomp; print "sh scripts/$_
&\n";' | sh
```

3.1.4 To determine the proportion of trees that do not recover a monophyletic group including Porifera and ParaHoxozoa (i.e., Placozoa, Cnidaria, and Bilateria) or a monophyletic Metazoa, run the following command in each of the following subdirectories in directory `05-TREE_SHAPE`: `01-TREE000C_1_1000`, `02-TREE000C_5_1000`, `03-TREE000C_9_1000`.

```
perl ../../02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/is_mono.pl
DAYHOFF CHANG > is_mono.out
```

3.1.5 Run the script `comphet_bargraph_shape.R` in RStudio to generate a bar graph showing the percentage of incorrect trees reconstructed under each method and level of compositional heterogeneity tested.

4. Testing 6-state Recoding Performance with Randomly Sampled Amino Acid Frequencies and Compositional Heterogeneity

4.1 Simulate compositionally heterogeneous datasets with an algorithm that randomly pairs starting amino acid frequencies

A reviewer suggested that we repeat our compositional heterogeneity analyses from section 2.3.1

with starting amino acid frequencies randomly paired instead of alphabetically.

4.1.1 In the directory `06-AMINO_ACIDS`, run the following commands to produce datasets with randomly paired starting amino acid frequencies and increasing compositional heterogeneity.

```
cd 01-TREE0002_1_1000_RANDOM/

perl ../../02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/comphet.pl
TREE0002 0.1 1000 RANDOM all.out

cd 02-TREE0002_5_1000_RANDOM/

perl ../../02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/comphet.pl
TREE0002 0.5 1000 RANDOM all.out

cd 03-TREE0002_9_1000_RANDOM/

perl ../../02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/comphet.pl
TREE0002 0.9 1000 RANDOM all.out
```

4.2 Recode datasets and create maximum-likelihood analysis shell scripts

4.2.1 Make a new directory called 'scripts.' Convert all simulated datasets to Dayhoff 6-state recoded datasets and generate shell scripts to perform maximum-likelihood analyses in RAxML using the script chunkify_estimate.pl.

```
mkdir scripts

perl ../03-SATURATION/04-ESTIMATED_MODEL/chunkify_estimate.pl
```

4.2 Phylogenetic reconstructions

4.2.1 Execute the scripts generated from `chunkify_estimate.pl` to perform maximum-likelihood analyses on both recoded and non-recoded datasets. Recoded datasets were reconstructed under the MULTIGAMMA multi-state model with GTR and non-recoded datasets were reconstructed under the Dayhoff model.

```
ls -1 scripts/ | grep 'myserver*' | perl -ne 'chomp; print "sh scripts/$_
&\n";' | sh
```

4.3 Score trees

4.3.1 Run the following command in each of the directories `01-TREE0002_1_1000_RANDOM`, `02-TREE0002_5_1000_RANDOM`, and `03-TREE0002_9_1000_RANDOM` to determine the proportion of trees that do not correctly reconstruct a monophyletic group including taxa from clade-A and clade-B and a monophyletic group including taxa from clade-C and clade-D. The output file identifies which trees were incorrectly reconstructed and indicates the proportion of incorrect trees produced under each method (recoding and non-recoding).

```
perl ../../02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/is_mono.pl
DAYHOFF > is_mono.out
```

4.4 Generate a bar graph for percentage of incorrect trees

4.4.1 Run the script `comphet_bargraph_random.R` in RStudio to generate a bar graph showing the percentage of incorrect trees reconstructed under each method and level of compositional heterogeneity tested.

5. Compare levels of compositional heterogeneity from real datasets to simulated data

5.1 Calculate relative compositional frequency variability (RCFV) for simulated and real datasets

Given the decline in performance of non-recoding at very high inflation parameters, we assessed how our simulated levels of compositional heterogeneity compared to those present among real datasets.

5.1.1 In the directory `07-RCFV`, run the following commands to calculate RCFV scores for simulated datasets:

```
perl rcfv.pl --dir=/02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/01-
TREE0008/01-TREE0008.1 --phylip --pattern='phy$' > TREE0008.1_rcfv.out

perl rcfv.pl --dir=/02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/01-
TREE0008/02-TREE0008.5 --phylip --pattern='phy$' > TREE0008.5_rcfv.out
perl rcfv.pl --dir=/02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/01-
TREE0008/03-TREE0008.9 --phylip --pattern='phy$' > TREE0008.9_rcfv.out

perl rcfv.pl --dir=/02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/02-
TREE0004/01-TREE0004.1 --phylip --pattern='phy$' > TREE0004.1_rcfv.out

perl rcfv.pl --dir=/02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/02-
TREE_0004/02-TREE0004.5 --phylip --pattern='phy$' > TREE0004.5_rcfv.out

perl rcfv.pl --dir=/02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/02-
TREE_0004/03-TREE0004.9 --phylip --pattern='phy$' > TREE0004.9_rcfv.out

perl rcfv.pl --dir=/02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/03-
TREE0002/01-TREE0002.1 --phylip --pattern='phy$' > TREE0002.1_rcfv.out

perl rcfv.pl --dir=/02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/03-
TREE0002/02-TREE0002.5 --phylip --pattern='phy$' > TREE0002.5_rcfv.out

perl rcfv.pl --dir=/02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/03-
TREE0002/03-TREE0002.9 > TREE0002.9_rcfv.out

perl rcfv.pl --dir=/02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/04-
TREE0001/01-TREE0001.1 --phylip --pattern='phy$' > TREE0001.1_rcfv.out

perl rcfv.pl --dir=/02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/04-
TREE0001/02-TREE0001.5 --phylip --pattern='phy$' > TREE0001.5_rcfv.out

perl rcfv.pl --dir=/02-COMPOSITIONAL_HETEROGENEITY/02-HETEROGENEOUS_DATA/04-
TREE0001/03-TREE0001.9 --phylip --pattern='phy$' > TREE0001.9_rcfv.out
```

5.1.2 In the directory 07-RCFV, run the following command to calculate RCFV scores for real

datasets:

```
perl rcfv.pl –dir=/01-REAL_DATA --phylip --pattern='phy$' > REAL_DATA
_rcfv_phy.out
```

```
perl rcfv.pl –dir=/01-REAL_DATA --fa --pattern='fa$' > REAL_DATA_rcfv_fa.out
```

5.2 Statistics and data visualization

5.2.1 Run the script `normality_test.R` in RStudio to test if the RCFV scores from simulated and real data are normally distributed using the Shapiro-Wilk's test.

5.2.2 Run the script `Boxplot_Kruskal_Wallis.R` in RStudio to visualize the data using a box-plot and to determine if there were significant differences between RCFV scores from simulated and real data using the Kruskal-Wallis test. This script also performs the pairwise Wilcoxon rank sum test with the Bonferroni correction for multiple testing to determine which specific datasets significantly differed.