Empirical analysis of conifers

Jeremy M. Beaulieu and Brian C. O'Meara

```
## Loading required package: ape
## Loading required package: deSolve
## Loading required package: GenSA
## Loading required package: subplex
## Loading required package: nloptr
```

Conifer phylogeny

The phylogeny used in this study comes from Leslie et al. (2018), which expands upon the phylogeny of Leslie et al. (2012) by including more species and additional fossil calibration points. In particular, this analysis improves taxon sampling in the previously undersampled genera Abies (increased from 26 to 55 spp.), *Callitris* (from 4 to 10 spp.), and *Podocarpus* (from 58 to 74 spp.). We also notably expanded sampling for *Agathis* (increased from 13 to 16 spp.), *Cupressus* (from 7 to 11 spp.), *Picea* (from 32 to 35 spp.), *Pinus* (from 102 to 116 spp.), and *Prumnopitys* (from 5 to 8 spp.). The phylogeny of Leslie et al. (2018) includes 578 species, or around 90% of recognized extant diversity. It is based on sequences from two chloroplast genes (*rbc*L, *mat*K) and one nuclear ribosomal gene (18S). More specific details about the dating procedures and the fossil calibrations used are found within the main text and supplemental information provided by Leslie et al. (2018).

Background code

A default `MiSSE` run uses birth-death parameter estimates obtained from `ape` as the starting values for the optimization search. However, in this analysis, we used the default settings, as well as a set of random parameters values. We sampled 19 random values for turnover, τ, drawn from a lognormal distribution with the mean being the log of the ML estimate of turnover under a single-rate birth-death model; the random values for extinction fraction, ϵ, was drawn from a uniform distribution bound from 0 to 1. In total, we tried 20 random starts.

Initially, we set set the maximum number of rate classes to 12, and we assumed that ϵ was either fixed within a given rate class or allowed to vary among them. In total, we ran 22 total models each with 20 different starting points. Below are the functions used.

```
library(hisse)
library(parallel)

GetModelPars <- function(shift.no, weps = FALSE) {
    turnover <- 1:shift.no
    if (weps == TRUE) {
        eps <- 1:shift.no
    } else {
        eps <- rep(1, length(shift.no))
    }
    par.settings <- NULL
    par.settings$turnover <- turnover
    par.settings$eps <- eps
    return(par.settings)
```

```
}

RunMiSSE <- function(phy, f, weps = FALSE, n.cores = 10) {
    # Step 1: Get a set of 20 random starting points.
    start.vals <- hisse:::starting.point.generator(phy, samp.freq.tree = f, k = 1)
    turn.tries <- c(sum(start.vals[1:2]), exp(rnorm(19, log(sum(start.vals)), 0.25)))
    eps.tries <- c(start.vals[2]/start.vals[1], runif(19, 0.01, 0.99))

    print(turn.tries)
    print(eps.tries)

    # This begins looping through the model space
    for (model.index in 2:12) {
        print(model.index)
        RandomStartVal <- function(iteration, model.index, phy, f, turnover, eps,
            turn.tries, eps.tries, trans.tries, weps) {
            print(iteration)
            tmp <- MiSSE(phy = phy, f = f, turnover = turnover, eps = eps, starting.vals = c(turn.tries
                eps.tries[iteration], trans.tries), root.type = "herr_als")
            if (weps == TRUE) {
                save(tmp, file = paste(model.index, iteration, ".e.Rsave", sep = "."))
            } else {
                save(tmp, file = paste(model.index, iteration, "Rsave", sep = "."))
            }
        }
        par.set <- GetModelPars(model.index, weps = weps)
        startTries <- mclapply(1:20, RandomStartVal, model.index = model.index, phy = phy,
            f = f, turnover = par.set$turnover, eps = par.set$eps, turn.tries = turn.tries,
            eps.tries = eps.tries, trans.tries = 0.001, weps = weps, mc.cores = n.cores)
    }
}
phy <- read.tree("Leslieetal2018.tre")
RunMiSSE(phy = phy, f = 0.9, n.cores = 20)
```

One other thing to note is that we forgot to include the single rate class model into the set. We ended up doing this after the fact, using the following code:

```
phy <- read.tree("Leslieetal2018.tre")
tmp <- MiSSE(phy = phy, f = 0.9, turnover = c(1), eps = c(1))
save(tmp, file = "rateclass.1.weps.Rsave")
recon
```

Summarizing the model fits

The final fit for a given rate class was determined by taking the starting values that maximized the likelihood. We used the following code to summarize these fits:

```
logliks <- c()
aics <- c()
weps <- TRUE
for (rate.class in 2:12) {
    if (weps == TRUE) {
        model.restarts <- system(paste("ls -1 ", rate.class, "*..e.Rsave", sep = ""),
            intern = TRUE)
```

```
        res <- c()
        res2 <- c()
        for (file.index in 1:length(model.restarts)) {
            load(model.restarts[file.index])
            res <- c(res, tmp$loglik)
            res2 <- c(res2, tmp$AIC)
        }
        logliks <- c(logliks, res[which.max(res)])
        aics <- c(aics, res2[which.min(res2)])
        load(model.restarts[which.max(res)])
        save(tmp, file = paste("rateclass.", rate.class, ".weps.Rsave", sep = ""))
    } else {
        model.restarts.weps <- system(paste("ls -1 ", rate.class, "*..e.Rsave", sep = ""),
            intern = TRUE)
        model.restarts.all <- system(paste("ls -1 ", rate.class, "*.Rsave", sep = ""),
            intern = TRUE)
        model.restarts <- model.restarts.all[which(!model.restarts.all %in% model.restarts.weps)]
        res <- c()
        res2 <- c()
        for (file.index in 1:length(model.restarts)) {
            load(model.restarts[file.index])
            res <- c(res, tmp$loglik)
            res2 <- c(res2, tmp$AIC)
        }
        logliks <- c(logliks, res[which.max(res)])
        aics <- c(aics, res2[which.min(res2)])
        load(model.restarts[which.max(res)])
        save(tmp, file = paste("rateclass.", rate.class, ".noweps.Rsave", sep = ""))
    }
}
```

From a purely model selection perspective, models that assume four, five, and six rate classes of τ rates (and assume a single global ϵ) are fairly indistinguishable from one another, despite the five rate classes having the overall best AIC. Models that allowed ϵ to freely estimated among the different rate classes did not significantly improve the fit of the model relative to assuming ϵ was fixed across all τ rate classes.

Obtaining the reconstructions

In order to reconstruct the rate history across the tree, we used the following code:

```
RunMiSSERecon <- function(with.eps = FALSE, num.models, n.cores) {

    for (rate.class in 2:num.models) {

        if (with.eps == TRUE) {
            model.restarts <- system(paste("ls -1 ", rate.class, "*..e.Rsave", sep = ""),
                intern = TRUE)
        } else {
            model.restarts.weps <- system(paste("ls -1 ", rate.class, "*..e.Rsave",
                sep = ""), intern = TRUE)
            model.restarts.all <- system(paste("ls -1 ", rate.class, "*.Rsave", sep = ""),
                intern = TRUE)
            model.restarts <- model.restarts.all[which(!model.restarts.all %in% model.restarts.weps)]
```

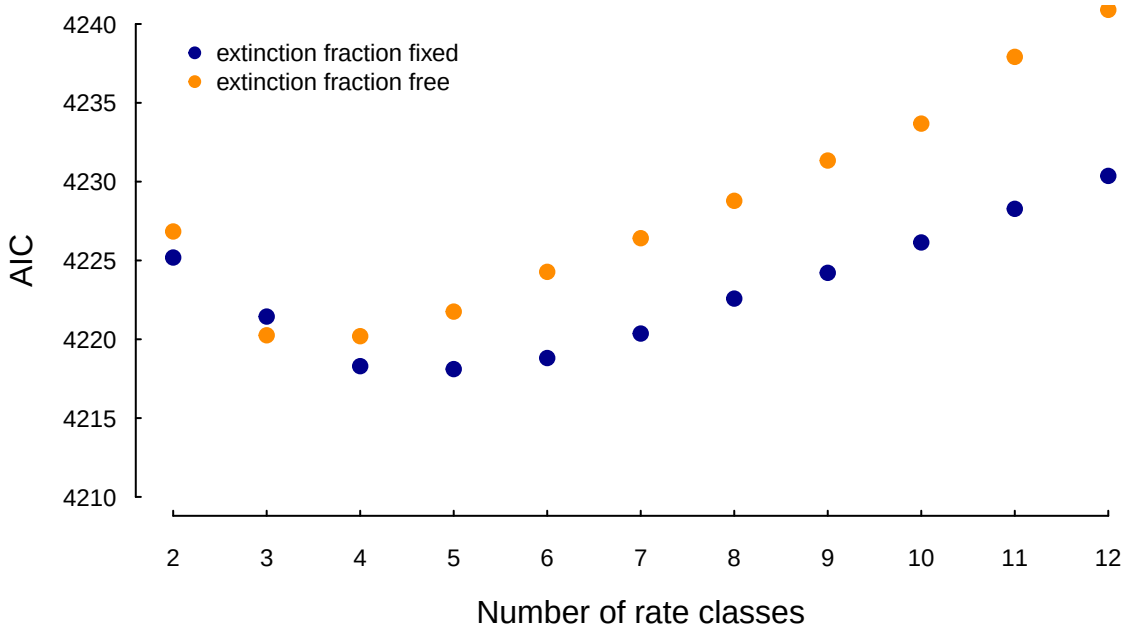

Figure 1: The fit of an incremental increase in the number of rate classes estimated under a MiSSE analysis of the conifer phylogeny of Leslie et al. (2018). There is a clear reduction in AIC from one rate class to two (not shown), which levels off between four, five, and six rate classes. Note, the distinction is made between whether the model allows extinction fraction to fixed (blue) or free across regimes (orange).

```
        }

        res <- c()
        for (file.index in 1:length(model.restarts)) {
            load(model.restarts[file.index])
            res <- c(res, tmp$loglik)
        }

        load(model.restarts[which.max(res)])
        pp <- MarginReconMiSSE(phy = tmp$phy, f = tmp$f, pars = tmp$solution, hidden.states = tmp$hidde
            condition.on.survival = TRUE, root.type = "herr_als", aic = tmp$AIC,
            verbose = TRUE, n.cores = n.cores)
        save(pp, file = "recon.bestmodel.2.Rsave")
    }
}
RunMiSSERecon(with.eps = FALSE, num.models = 12, n.cores = 20)
```

As before, we forgot to include the single rate class model reconstruction, and so did it after the fact:

```
load("rateclass.1.weps.Rsave")
pp <- MarginReconMiSSE(phy = tmp$phy, f = tmp$f, pars = tmp$solution, hidden.states = tmp$hidden.states
    condition.on.survival = TRUE, root.type = "herr_als", aic = tmp$AIC, verbose = TRUE,
    n.cores = 3)
save(pp, file = "recon.bestmodel.1.Rsave")
```

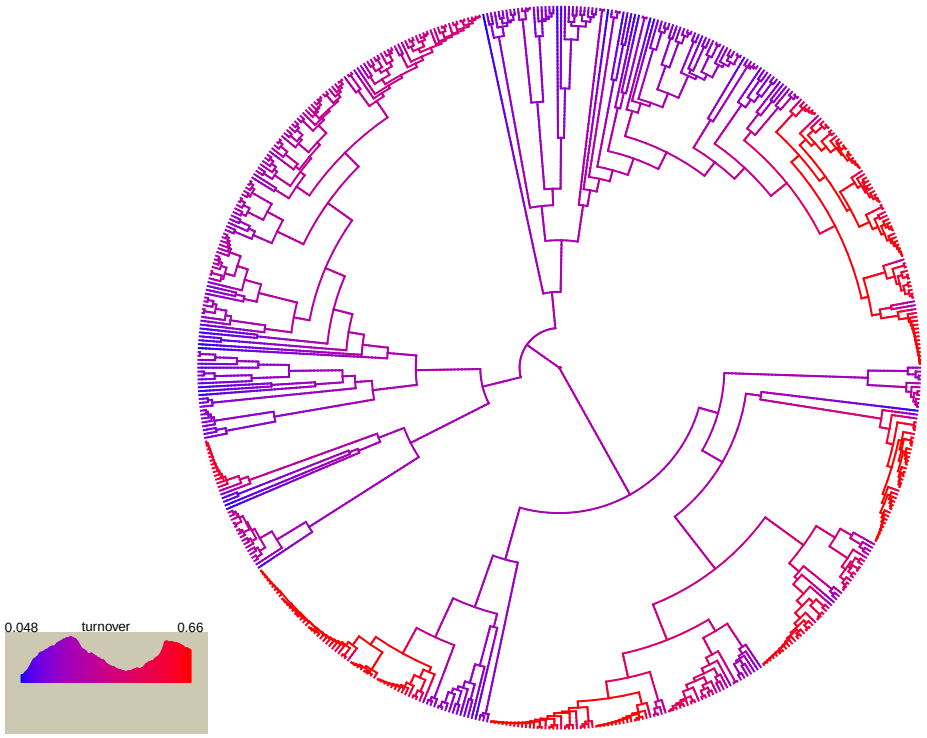

Figure 2: A model-averaged MiSSE analysis of the conifer phylogeny of Leslie et al. (2018) showing the distribution of turnover through time and across taxa.

Model-averaging the reconstructions

Since no single model is substantially supported over all others, and since there is uncertainty in the distribution of rate classes within a given model, we employ the model-averaging procedure described in Beaulieu and O'Meara (2016) and Caetano et al. (2018). The first step for this is to simple take the reconstructions from all the models and put them into a list. This list can then be supplied to several functions to obtain various model-averaged rate estimates.

```
setwd("~/missecomparison/conifer_MISSE")
best.recons <- system(paste("ls -1 ", "recon.bestmodel.*", sep = ""), intern = TRUE)
recon.list = list()
for (index in sequence(length(best.recons))) {
    load(best.recons[index])
    recon.list[[index]] <- pp
}
```

Here we will show the reconstruction of the τ and ϵ across the conifer tree.

```
## $rate.tree
## Object of class "contMap" containing:
##
## (1) A phylogenetic tree with 578 tips and 577 internal nodes.
##
## (2) A mapped continuous trait on the range (0.048018, 0.661904).
```

```
## $rate.tree
## Object of class "contMap" containing:
##
```

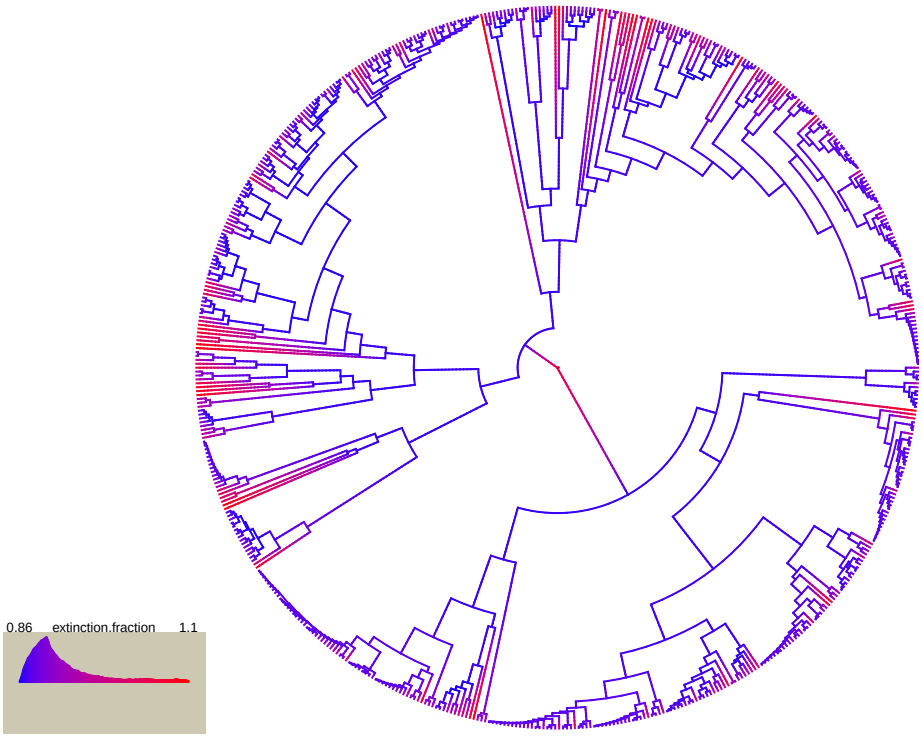

Figure 3: A model-averaged MiSSE analysis of the conifer phylogeny of Leslie et al. (2018) showing the distribution of extinction fraction through time and across taxa.

```
## (1) A phylogenetic tree with 578 tips and 577 internal nodes.
##
## (2) A mapped continuous trait on the range (0.864582, 1.126699).
```

More stuff??