

The Evolution of Labiodental Articulation in Indo-European

Balthasar Bickel and Paul Widmer

July 2017

Contents

S4.1 Introduction	2
S4.2 Data and Phylogenies	2
S4.3 Modeling transition rates	6
S4.3.1 Procedures	6
S4.3.2 Parameters	10
S4.3.3 Evaluation of priors	10
S4.3.3.1 Speed of change allowed under various prior distributions on rates	11
S4.3.3.2 Maximum Likelihood estimates of rates	11
S4.3.4 Individual analyses	12
S4.3.4.1 Phylogenies from Chang et al.	12
S4.3.4.2 Phylogenies from Bouckaert et al.	13
S4.3.5 Results	13
S4.4 Ancestral state estimates	16
S4.4.1 Reconstruction comparing assumed proto-states	18
S4.4.1.1 Based on the phylogenies from Chang et al.	18
S4.4.1.2 Based on the phylogenies from Bouckaert et al.	19
S4.4.2 Summary	20
S4.5 Stochastic Character Mapping	21
S4.5.1 Procedures	21
S4.5.2 Parameters	22
S4.5.3 Individual analyses	22
S4.5.3.1 Reconstruction of labiodental reflexes for $*b^h$	22
S4.5.3.2 Reconstruction of labiodental reflexes for $*p$	25
S4.5.3.3 Reconstruction of labiodental reflexes for $*w$	27
S4.5.3.4 Reconstruction of labiodental reflexes for $*m$	29
S4.5.3.5 Reconstruction of labiodental reflexes for $*g^w$	31
S4.5.3.6 Reconstruction of labiodental reflexes for $*g^{wh}$	33
S4.5.3.7 Reconstruction of labiodental reflexes for $*k^w$	35
S4.5.3.8 Reconstruction of labiodental reflexes for $*d^h$	37
S4.5.3.9 Reconstruction of labiodental reflexes for $*k^j$	39
S4.5.3.10 Reconstruction of labiodental reflexes for $*b$	41
S4.5.4 Summary	43
References	48

S4.1 Introduction

Indo-European comparative linguistics has identified what is called *correspondence sets*: words with related meanings show systematic sound correspondences. For example, there is a large set of related words that begin with bilabial *w* in some languages, e.g. in English (*will*, *word*, *wind* etc.). These word-initial sounds systematically correspond to a bilabially articulated sound in some related languages (e.g. Latin *uolō*, *uerbum*, *uentus* etc.) and to a labiodentally articulated sound in others (e.g. German *wollen*, *Wort*, *Wind* etc.). This establishes a set of correspondences that contains the articulations $\{w, v\}$ (among others). Each set reflects descent (with or without modification) from an ancestral proto-sound, but the reconstruction of the phonetic quality of this sound is an unresolved problem. Despite this problem, correspondence sets are traditionally labeled by a symbol that makes a vague suggestion on the phonetics, e.g. the $\{w, v\}$ set is conventionally labelled “**w*” (with a distinctive asterisk), suggesting a bilabial realization. It is clear for experts, however, that these labels are mere suggestions and do not reflect an actual reconstruction of phonetics. To avoid confusion, we refer to the label of a correspondence set as a ‘correspondeme’. Unlike a phoneme, a correspondeme is not associated with a well-defined range of phonetic values. A correspondeme only represents a distinct (meaning-differentiating) proto-sound from which all modern variants in a set derive, whatever that sound might have been phonetically. This observation challenges any inference from correspondeme labels to the presence or absence of labiodental places of articulation (an inference that was explicitly made by Hockett (1)).

Traditional attempts at reconstructing actual phonetic values of correspondemes are chiefly based on qualitative considerations of maximum parsimony in the phylogeny, coupled with a tacit assumption that the values closely reflect what is reported in the earliest phonetic descriptions of well-investigated western languages (e.g. bilabial realization in Latin and Ancient Greek for descendants of the $\{w, v\}$ set). This approach incurs two problems. First, maximum-parsimony methods are problematic because sound changes can be completely reversed within relatively short time spans. For example the Germanic nasal *m* became *f* before *n* in North Germanic (Old Norse, early Old Swedish *nafn* ‘name’) and reverted to *m* in late Old Swedish (*namn*). Second, arguments based on earliest descriptions or even attestations are problematic because all current estimates show that the time that elapsed between Proto-Indo-European and the earliest attested languages is at least as long as the time between these languages and now. This is true even for relative young age estimates (2).

Here we take a fresh approach to the problem and model sound change as Continuous-Time Markov Chains (CTMC), allowing for reversals and different transition rates between states (Section 3). We fit CTMC models using the Markov Chain Monte Carlo Sampling (MCMC) implemented in *BayesTraits* (3). We then take the best fitting models to estimate ancestral values (Section S4.4) and to reconstruct values for each time interval of Indo-European (Section S4.5). For this we apply Stochastic Character Mapping (4–6), a method that has proven valid for linguistic reconstruction elsewhere (7).

S4.2 Data and Phylogenies

Etymological research in Indo-European linguistics has established ten major correspondemes which have a labiodental reflex in at least one daughter language: $*b^h$, $*p$, $*w$, $*m$, $*g^w$, $*g^{wh}$, $*k^w$, $*d^h$, $*k^j$, and $*b$. The phonetic values of correspondemes are justified in detail in Section S6:

```
labiodental.data <- read.csv('../data/ie_labiodental_data_d.csv', stringsAsFactors=F)
```

Extant languages have 76% labiodentals. Figures S5.1 and S5.2 show their distribution in two different phylogenies of Indo-European (using the *ggtree*(8)). We use phylogenies from Chang and colleagues(2) (identified by the suffix *.c* here) and from Bouckaert and colleagues(9) (suffix *.b*):

```
ie.c.sum.tree <- read.nexus('../data/indoeuropean-C-sum-matched.nex')
ie.c.sum.tree$tip.label <- gsub('(.*)(<.*>)', '\\1', ie.c.sum.tree$tip.label)
ie.c.sum.tree <- drop.tip(ie.c.sum.tree,
```

```

        setdiff(ie.c.sum.tree$tip.label, labiodental.data$language))

ie.b.sum.tree <- read.nexus('../data/indoeuropean-B-sum-matched.nex')
ie.b.sum.tree$tip.label <- gsub('(.*)(<.*>)', '\\1', ie.b.sum.tree$tip.label)
ie.b.sum.tree <- drop.tip(ie.b.sum.tree,
        setdiff(ie.b.sum.tree$tip.label, labiodental.data$language))

ie.c.trees <- read.nexus('../data/indoeuropean-C-20k-matched.nex')
ie.c.trees <- lapply(ie.c.trees, function(t) {
  t$tip.label <- gsub('(.*)(<.*>)', '\\1', t$tip.label)
  t <- drop.tip(t, setdiff(t$tip.label, labiodental.data$language))
  return(t)
})
class(ie.c.trees) <- 'multiPhylo'
attributes(ie.c.trees)$TipLabel <- ie.c.trees[[1]]$tip.label

ie.b.trees <- read.nexus('../data/indoeuropean-B-10k-matched.nex')
ie.b.trees <- lapply(ie.b.trees, function(t) {
  t$tip.label <- gsub('(.*)(<.*>)', '\\1', t$tip.label)
  t <- drop.tip(t, setdiff(t$tip.label, labiodental.data$language))
  return(t)
})
class(ie.b.trees) <- 'multiPhylo'
attributes(ie.b.trees)$TipLabel <- ie.b.trees[[1]]$tip.label

```

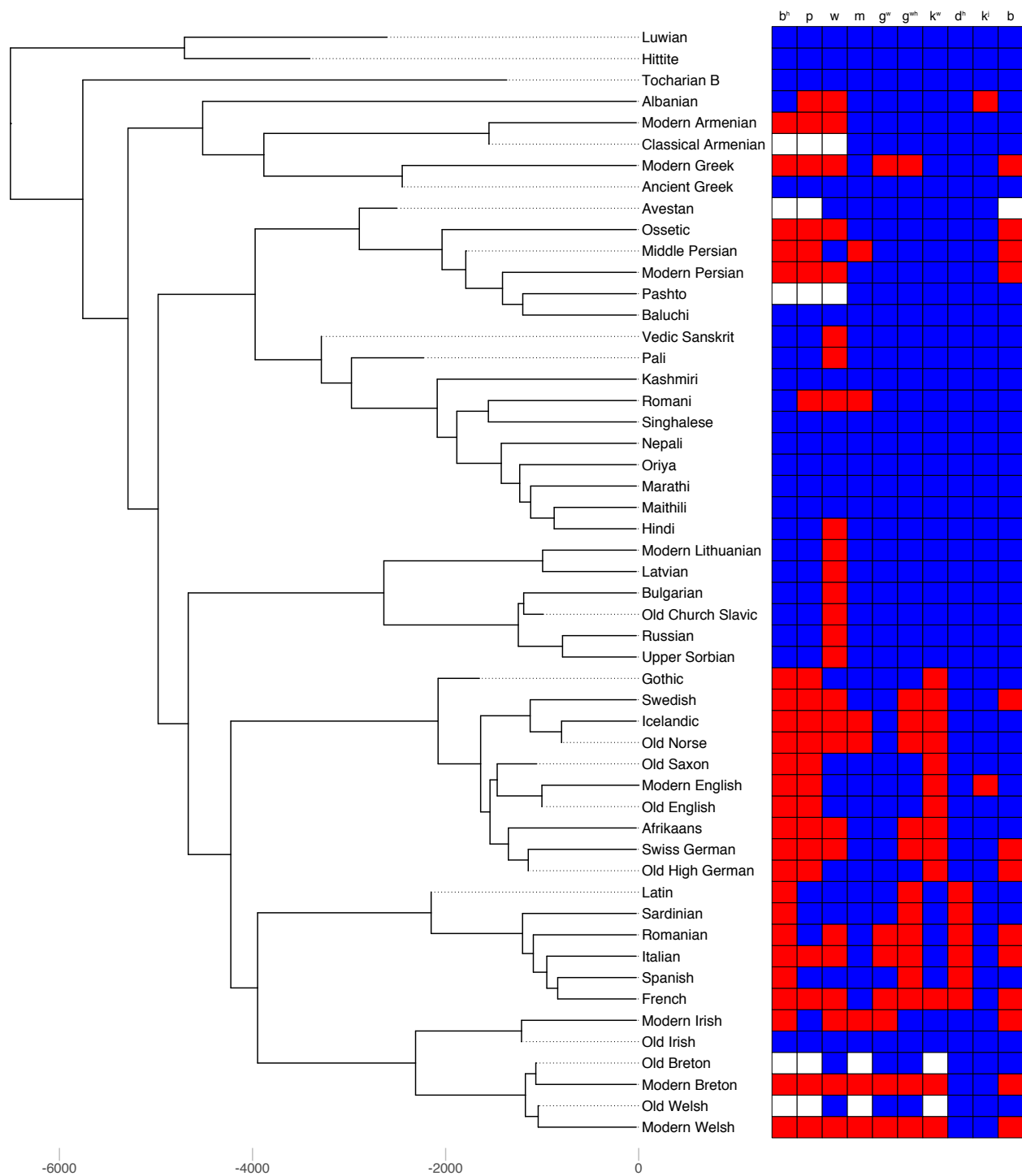
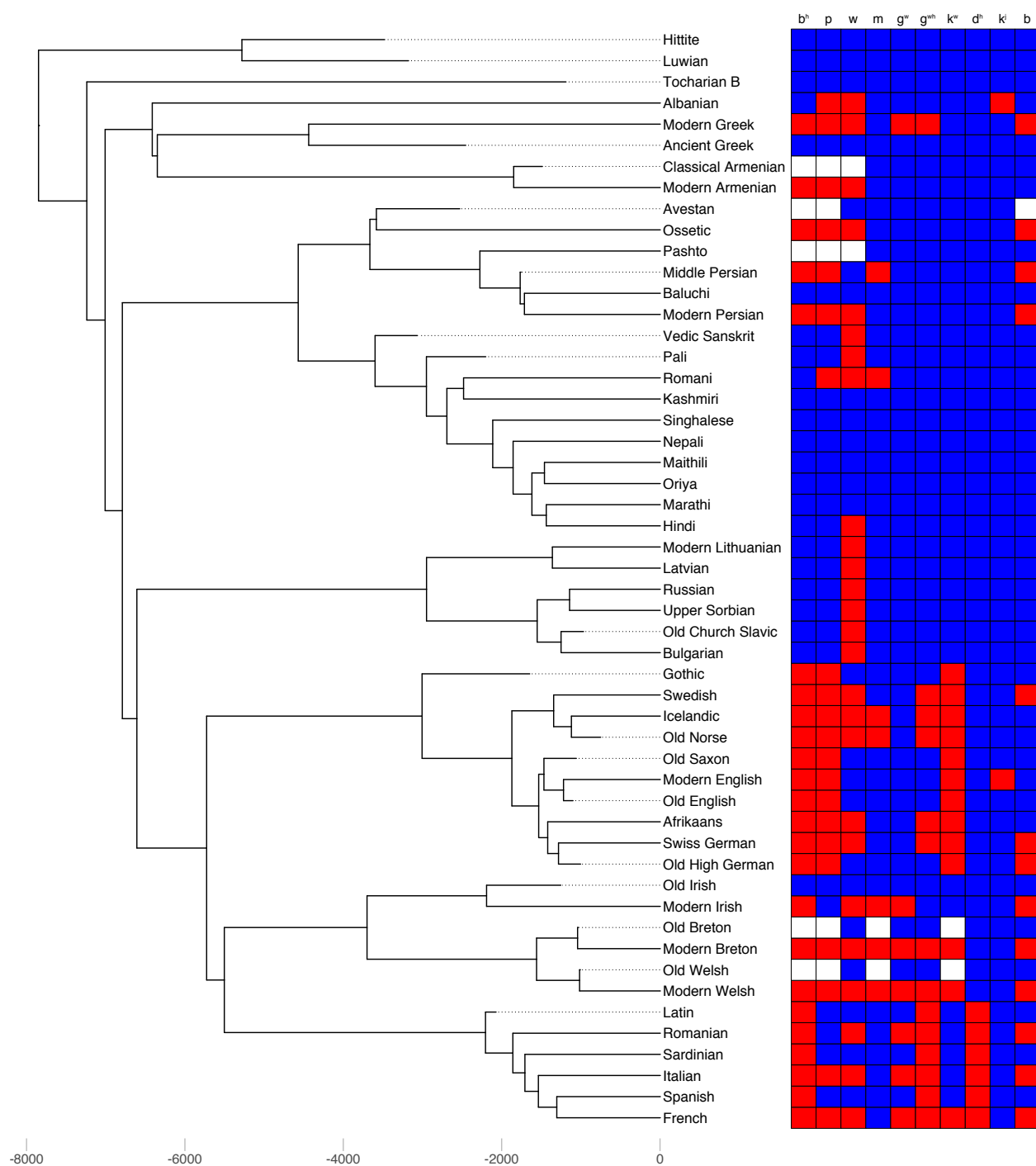


Figure S4.1: Labiodental reflexes in each correspondeme that contains at least one labiodental reflex. Red indicates presence, blue absence a labiodental reflex. White indicates unknown or uncertain data. Maximum clade consensus tree from Chang et al.



S4.3 Modeling transition rates

We model language change as CTMCs with the two states:

- Y: yes, the reflex of a given correspondeme is labiodental
- N: no, the reflex of a given correspondeme is not labiodental

We then estimate the rates of change q between two states in a Bayesian framework (5, 10–12):

- q_{YN} : instantaneous rate of loosing a labiodental realization (Yes>No)
- q_{NY} : instantaneous rate of gaining a labiodental realization (Yes>No)

In order to assess whether there is a diachronic bias in either direction, e.g. whether gains are favored over losses, we compare two models for each correspondeme:

- **ARD**: all rates different, i.e. rates of change towards vs away from a labiodental realization are different: $q_{YN} \neq q_{NY}$. This model therefore has two parameters.
- **ER**: equal rates, i.e. rates of change towards vs away from a labiodental realization are the same: $q_{YN} = q_{NY}$. This model therefore has one parameter.

In each case, the models are compared with Bayes Factors (BF), which are defined as double the difference between the log marginal likelihood of the more complex model, i.e. the ARD model, and the log marginal likelihood of the simpler model, i.e. the ER model. $BF > 2$ are considered to give “positive evidence” for the better-fitting model; between 5 and 10 “strong evidence” and > 10 “very strong evidence”.

We replicate all MCMC analyses several times(3) and report the median and the median absolute deviation across these replications. The parameters of each run of the analysis were determined on the basis of visual inspections of trace plots of the likelihoods and (in the case of posterior tree sample) of the tree mixing.

S4.3.1 Procedures

We use wrapper scripts which run `BayesTraitsV2` (3) in the background and allow various steps of post-processing the output:

```
fit.mcmc <- function(variable, tree=ie.sum.tree, data=labiodental.data,
  prior="uniform 0 .005", assumed.mrca=NULL,
  iterations=mcmc.iterations, burn.in=mcmc.burn.in,
  sampling.rate=mcmc.sampling.rate, covarion=F, reuse.tree=T) {
  test.ard <- Discrete(tree, data[,c('language', variable)], "Bayesian",
    pa=prior, it=iterations, bi=burn.in, sa=sampling.rate,
    rm=T, fo=assumed.mrca, cv=covarion, reuse.tree=reuse.tree)
  test.er <- Discrete(tree, data[,c('language', variable)], "Bayesian",
    resall='qNY', it=iterations, pa=prior, bi=burn.in, sa=sampling.rate,
    fo=assumed.mrca, cv=covarion, reuse.tree=reuse.tree)

  summary.er <- tail(test.er[,5],1)
  summary.ard <- tail(test.ard[,grep('q', colnames(test.ard), value=T)],1)
  bfres <- bftest(test.er, test.ard)
  bfres$Better <- ifelse(bfires$BetterModel %in% 'Model 1', 'ER', 'ARD')

  return(list(ARD=test.ard,
    ER=test.er,
    ER.vs.ARD=bfres[,c(1,3)],
    ER.rate=summary.er,
    ARD.rates=summary.ard
  ))
}
```

```

fit.ml <- function(variable, tree=ie.sum.tree, data=labiodental.data, assumed.mrca=NULL) {
  # make sure we don't use the wrong trees in case they are recycled in the mcmc runs:
  if (file.exists("BT.current.tree.nex")) { file.remove('BT.current.tree.nex') }
  test.ard <- Discrete(tree, data[,c('language', variable)], "ML",
    mlt=100, fo=assumed.mrca, reuse.tree=F)
  test.er <- Discrete(tree, data[,c('language', variable)], "ML",
    resall='qNY', mlt=100, fo=assumed.mrca, reuse.tree=F)
  return(list(ER=test.er, ARD=test.ard))
}

```

Function for tracing the MCMC iterations:

```

trace.plots <- function(mcmc.res, model=c('ARD','ER'), which=c('Likelihood', 'Rates', 'Trees')) {
  switch(model,
    ARD = {df <- do.call(rbind, lapply(seq_along(mcmc.res), function(r) {
      x <- mcmc.res[[r]][['ARD']]
      x$replicate <- r
      return(x)
    })),
    ER = {df <- do.call(rbind, lapply(seq_along(mcmc.res), function(r) {
      x <- mcmc.res[[r]][['ER']]
      x$replicate <- r
      return(x)
    })),
  )
  switch(which,
    Likelihood = { ggplot(df, aes(x=Iteration,y=Harmonic.Mean)) + geom_line(size=.2) +
      facet_wrap(~replicate) + theme_bw(base_size = 12) },
    Rates = { gather(df, rate, value, qNY:qYN) %>%
      ggplot(aes(x=Iteration, y=value, colour=rate)) +
      geom_line(size=.2) + theme(legend.position='top') +
      guides(colour = guide_legend(override.aes = list(size=2))) +
      facet_wrap(~replicate) + theme_bw(base_size = 12)
    },
    Trees = { #ggplot(df, aes(x=Iteration, y=Tree.No)) + geom_line(size=.2)
      ggplot(df, aes(x=Tree.No)) + geom_density(size=.2) +
      facet_wrap(~replicate) + theme_bw(base_size = 12) }
  )
}

```

Some helpers for exploring and plotting the results:

```

model.fits <- function(m) { do.call(rbind,
  lapply(m, function(x) { data.frame("["(x,"ER.vs.ARD")) } )
)}

fossil.fits <- function(m1, m2, model=c('ER','ARD')) { do.call(rbind,
  mapply(FUN=function(x, y) { bftest(x[[model]], y[[model]]), m1, m2, SIMPLIFY=F)
)}

rates <- function(m, model=c('ARD','ER')) {
  if(model %in% 'ARD') {mod <- paste0(model, ".rates")} else {mod <- paste0(model, ".rate")}
  do.call(rbind, lapply(m, function(x) { data.frame("["(x, mod)) } )
}

```

```

    })

ml.rates <- function(m) {
  x <- do.call(rbind, sapply(m, rbind))
  x$model <- rep(c('ER', 'ARD'), nrow(x)/2)
  return(x[,c('model',grep('q', colnames(x), value=T))])
}

transition.matrix <- function(m, aggr=c("median", "mad"), model=NULL, prob=T, years=100) {
  if(is.null(model)) {res <- sapply(m, function(x) unlist(tail(x,1)))} else {
    test_that('If you request a specific model, there should be a choice', {
      expect_true(any(names(m[[1]]) %in% c('ARD', 'ER'))
    })
    switch(model,
      ARD = {res <- sapply(m, function(x) unlist(tail(x$ARD,1)))},
      ER = {res <- sapply(m, function(x) unlist(tail(x$ER,1)))}
    )
  }
  switch(aggr,
    median = {rates <- apply(res[grep('q', rownames(res)),], 1, median)},
    mad = {rates <- apply(res[grep('q', rownames(res)),], 1, mad)}
  )
  # extract names, and do it in the right order so we can use it below for labeling:
  symbols <- unique(substr(gsub('q','',names(rates)),1,1))
  q.mat.diagonal <- sapply(split(rates, ceiling(seq_along(rates)/(length(symbols)-1))),
    function(x) {matrix(NA, ncol=length(symbols), nrow=length(symbols))
      diag(x.mat) <- q.mat.diagonal
      idx <- which(is.na(q.mat), arr.ind=T)
      q.mat[idx[order(idx[,1]),],] <- rates # reordered indices by rows first
      # perform matrix exponentiation if requested:
      if(prob) { q.mat <- matexpo(years*q.mat) }
      dimnames(q.mat) <- list(symbols,symbols)
      return(q.mat)
    })
  }

plot.matrix <- function(m, label='x', legend=T, colors=color.scheme[-1],
  node.size=20, padding=.8, weight.range=c(.5,1),
  label.dodge=-.1, label.size=3) {
  gg <- graph_from_adjacency_matrix(adjmatrix=m, weighted=TRUE, mode="directed", diag=T)
  myarrow <- arrow(angle = 30, length = unit(0.3, "cm"), ends = "last", type = "closed")
  if(ncol(m)==2) { directions=rep(c(180,0),2) } else
    { directions=rep(c(30,270,150),3) } # unelegant... but how else?
  p <- ggraph(gg, layout = 'linear', circular=ncol(m)>2) +
    geom_edge_fan(aes(width=round(weight,2), label=round(weight,2)), n=200,
      angle_calc='along', label_dodge=unit(label.dodge, "cm"),
      label_size=label.size,
      arrow=myarrow, end_cap = circle(padding, 'cm'),
      start_cap = circle(padding, 'cm')
    ) +
    geom_edge_loop(aes(width=round(weight,2), direction=directions, label=round(weight,2)),arrow=myarrow,
      label_size=label.size,
      end_cap = circle(padding, 'cm'),
      start_cap = circle(padding, 'cm')
    )
}

```



```

    ) +
    geom_node_point(aes(color=name), size=node.size) +
    theme_graph(base_family=my.font) + theme(plot.title=element_text(hjust=.5),
    plot.margin = unit(c(0.5,0.5,0.5,0.5), "cm"))
if(legend) { p + scale_edge_width(name='transitions:', range=weight.range) +
    scale_color_manual(name=paste0("Reflex of *", label, "\nis labiodental:"),
    values=colors) +
    guides(color = guide_legend(override.aes = list(size = 3)))
} else {
    p + scale_edge_width(range=weight.range, guide=F) +
    scale_color_manual(values=colors, guide=F) +
    ggtitle(label)
}
}

```

Functions for estimating ARD models and then using the resulting transition rates for Stochastic Character Mapping (see the explanation in Section 3.3 below):

```

years.per.change <- function(variable, trees=ie.trees, sum.tree=ie.sum.tree, data=labiodental.data,
    tree.simulations=nsim, rate.replications=replications,
    prior='uniform 0 .0025', reuse.tree=T) {

  # fit model with BayesTraits:
  model <- replicate(rate.replications,
    Discrete(trees, data[,c('language', variable)],
    mode="Bayesian", pa=prior, it=mcmc.iterations, bi=mcmc.burn.in,
    sa=mcmc.sampling.rate, cv=F, reuse.tree=reuse.tree),
    simplify=F)

  # prepare data for simmap:
  nas <- c('-', '?') # catch NAs
  states <- data[, variable]
  names(states) <- data$language
  state.symbols <- setdiff(as.character(unique(states)), nas)
  n.states <- length(state.symbols)

  if(any(states %in% nas | is.na(states))) { # see ?make.simmap
    states <- to.matrix(states, state.symbols)
    states[rowSums(states)==0,] <- rep(1/n.states) # equal probability
  }

  # extract Q matrix:
  q.mat <- transition.matrix(model, 'median', prob=F)

  # estimate simmaps with a flat root prior:
  treemaps <- make.simmap(sum.tree, states, Q=q.mat, pi='estimated',
    nsim=tree.simulations, message=F)

  # extract number of changes, translate into how many years a change takes:
  simmap.description <- describe.simmap(treemaps, message=F)
  median.n.changes <- median(simmap.description$count[, 'N']) # N = N(transitions)
  min.n.changes <- min(simmap.description$count[, 'N'])
  max.n.changes <- max(simmap.description$count[, 'N'])
  return(list(simmap.description=simmap.description,

```

```

        median.years = round(sum(sum.tree$edge.length)/median.n.changes),
        min.years = round(sum(sum.tree$edge.length)/max.n.changes),
        max.years = round(sum(sum.tree$edge.length)/min.n.changes)
    ))
}

est.changes <- function(simmap.description, sum.tree) {
  apply(sapply(simmap.description, function(x) x$count[, 'N']), 2, function(col) {
    sum(sum.tree$edge.length)/col
  }) %>% as.data.frame %>%
  gather(correspondeme, years)
}

```

S4.3.2 Parameters

Replications:

```
B=10
```

Replications of covarion models:

```
B.cv=3
```

MCMC iterations:

```
mcmc.iterations=1e+07
```

Burn in:

```
mcmc.burn.in=1e+05
```

Sampling rate:

```
mcmc.sampling.rate=1e+03
```

S4.3.3 Evaluation of priors

We defined the prior distribution for each model so that the estimated number of changes is within a realistic ballpark. To assess this ballpark we explored transition rate estimates obtained with priors of $\mathcal{U}(0, .0025)$, $\mathcal{U}(0, .005)$ and $\mathcal{U}(0, .01)$ by performing *Stochastic Character Mapping* (see Section 5 below). For this purpose we focus on models allowing for rate differences (ARD). This allows a larger range of rates to explore. We then also compare the rates obtained with these priors with rates estimated from Maximum-Likelihood models.

Parameters:

Stochastic character mappings:

```
nsim=100
```

Replications of rate estimates:

```
replications=5
```

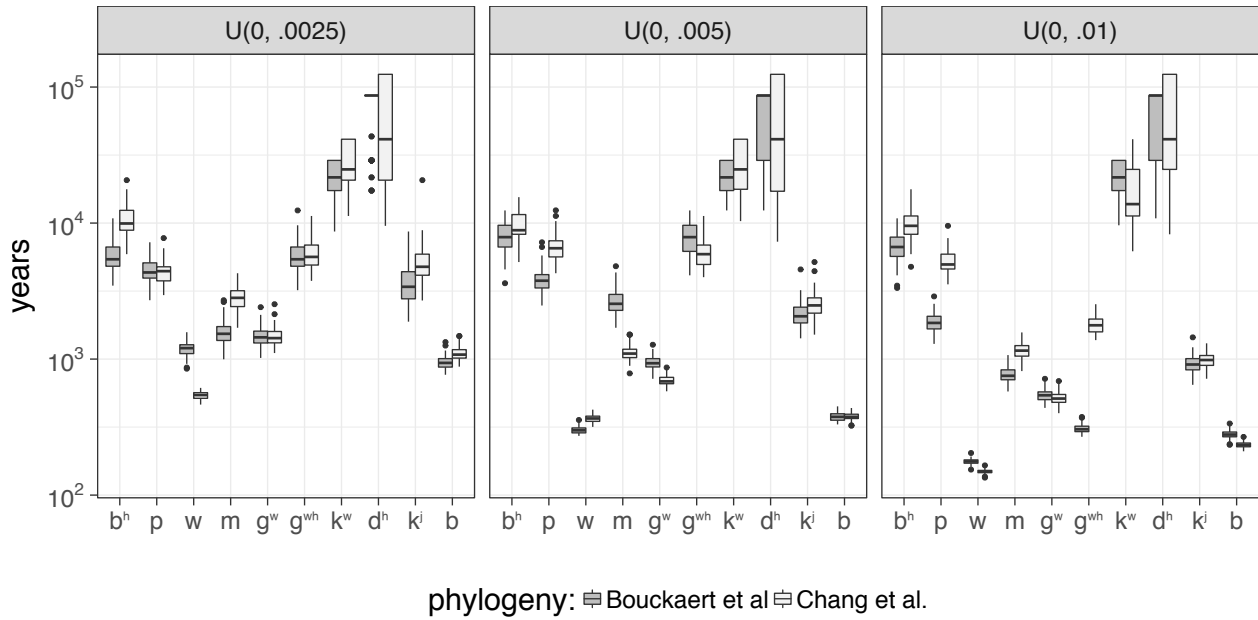


Figure S4.3: Estimated number of years per change per prior assumptions about rates of change

S4.3.3.1 Speed of change allowed under various prior distributions on rates

```
correspondemes <- c('b^h', 'p', 'w', 'm', 'g^w', 'g^wh', 'k^w', 'd^h', 'k^i', 'b')
```

Phylogenies from Chang et al.

```
ypc_0_.0025c <- sapply(correspondemes, function(crsp) {
  years.per.change(crsp, trees=ie.c.trees, sum.tree=ie.c.sum.tree,
    prior='uniform 0 .0025') })
ypc_0_.005c <- sapply(correspondemes, function(crsp) {
  years.per.change(crsp, trees=ie.c.trees, sum.tree=ie.c.sum.tree,
    prior='uniform 0 .005') })
ypc_0_.01c <- sapply(correspondemes, function(crsp) {
  years.per.change(crsp, trees=ie.c.trees, sum.tree=ie.c.sum.tree,
    prior='uniform 0 .01') })
```

Phylogenies from Bouckaert et al.

```
ypc_0_.0025b <- sapply(correspondemes, function(crsp) {
  years.per.change(crsp, trees=ie.b.trees, sum.tree=ie.b.sum.tree,
    prior='uniform 0 .0025') })
ypc_0_.005b <- sapply(correspondemes, function(crsp) {
  years.per.change(crsp, trees=ie.b.trees, sum.tree=ie.b.sum.tree,
    prior='uniform 0 .005') })
ypc_0_.01b <- sapply(correspondemes, function(crsp) {
  years.per.change(crsp, trees=ie.b.trees, sum.tree=ie.b.sum.tree,
    prior='uniform 0 .01') })
```

S4.3.3.2 Maximum Likelihood estimates of rates

Phylogenies from Chang et al.

```
(ml.estimates.c <- sapply(correspondemes, function(crsp) {
  ml <- replicate(B, fit.ml(crsp, tree=ie.c.sum.tree), simplify=F)
  rates <- unlist(ml.rates(ml)[-1])
  return(data.frame(median=round(median(rates),5), max=round(max(rates),5)))
})))
```

```
##          bh          p          w          m          gw          gwh          kw          dh          ky          b
## median 0.00012 0.00017 8e-04 0.00014 0.00012 1e-04 4e-05 1e-05 0.00011 0.00057
## max    3.11028 0.00017 0.00084 1.29729 0.07283 1e-04 2.72577 1e-05 4.23726 0.00269
```

Phylogenies from Bouckaert et al.

```
(ml.estimates.b <- sapply(correspondemes, function(crsp) {
  ml <- replicate(B, fit.ml(crsp, tree=ie.b.sum.tree), simplify=F)
  rates <- unlist(ml.rates(ml)[-1])
  return(data.frame(median=round(median(rates),5), max=round(max(rates),5)))
})))
```

```
##          bh          p          w          m          gw          gwh          kw          dh          ky          b
## median 7e-05 0.00012 0.00068 1e-04 7e-04 7e-05 3e-05 1e-05 1e-04 0.00292
## max    2.43028 0.76684 3.40766 2.95999 2.82243 1.48305 28.73511 1.03078 48.16946 1.34079
```

S4.3.4 Individual analyses

Given the results of the previous section, a prior of $\mathcal{U}(0, .01)$ seems reasonable, i.e. it is large enough without allowing unrealistically fast or unrealistically slow rates of change:

```
prior="uniform 0 .01"
```

S4.3.4.1 Phylogenies from Chang et al.

Reflexes of $*b^h$:

```
bh.mcmc.c <- replicate(B, fit.mcmc('bh', tree=ie.c.trees, prior=prior), simplify=F)
```

Reflexes of $*p$:

```
p.mcmc.c <- replicate(B, fit.mcmc('p', tree=ie.c.trees, prior=prior), simplify=F)
```

Reflexes of $*w$:

```
w.mcmc.c <- replicate(B, fit.mcmc('w', tree=ie.c.trees, prior=prior), simplify=F)
```

Reflexes of $*m$:

```
m.mcmc.c <- replicate(B, fit.mcmc('m', tree=ie.c.trees, prior=prior), simplify=F)
```

Reflexes of $*g^w$:

```
gw.mcmc.c <- replicate(B, fit.mcmc('gw', tree=ie.c.trees, prior=prior), simplify=F)
```

Reflexes of $*g^{wh}$:

```
gwh.mcmc.c <- replicate(B, fit.mcmc('gwh', tree=ie.c.trees, prior=prior), simplify=F)
```

Reflexes of $*k^w$:

```
kw.mcmc.c <- replicate(B, fit.mcmc('kw', tree=ie.c.trees, prior=prior), simplify=F)
```

Reflexes of $*d^h$:

```
dh.mcmc.c <- replicate(B, fit.mcmc('dh', tree=ie.c.trees, prior=prior), simplify=F)
```

Reflexes of $*k^j$:

```
ky.mcmc.c <- replicate(B, fit.mcmc('ky', tree=ie.c.trees, prior=prior), simplify=F)
```

Reflexes of $*b$:

```
b.mcmc.c <- replicate(B, fit.mcmc('b', tree=ie.c.trees, prior=prior), simplify=F)
```

S4.3.4.2 Phylogenies from Bouckaert et al.

Reflexes of $*b^h$:

```
bh.mcmc.b <- replicate(B, fit.mcmc('bh', tree=ie.b.trees, prior=prior), simplify=F)
```

Reflexes of $*p$:

```
p.mcmc.b <- replicate(B, fit.mcmc('p', tree=ie.b.trees, prior=prior), simplify=F)
```

Reflexes of $*w$:

```
w.mcmc.b <- replicate(B, fit.mcmc('w', tree=ie.b.trees, prior=prior), simplify=F)
```

Reflexes of $*m$:

```
m.mcmc.b <- replicate(B, fit.mcmc('m', tree=ie.b.trees, prior=prior), simplify=F)
```

Reflexes of $*g^w$:

```
gw.mcmc.b <- replicate(B, fit.mcmc('gw', tree=ie.b.trees, prior=prior), simplify=F)
```

Reflexes of $*g^{wh}$:

```
gwh.mcmc.b <- replicate(B, fit.mcmc('gwh', tree=ie.b.trees, prior=prior), simplify=F)
```

Reflexes of $*k^w$:

```
kw.mcmc.b <- replicate(B, fit.mcmc('kw', tree=ie.b.trees, prior=prior), simplify=F)
```

Reflexes of $*d^h$:

```
dh.mcmc.b <- replicate(B, fit.mcmc('dh', tree=ie.b.trees, prior=prior), simplify=F)
```

Reflexes of $*k^j$:

```
ky.mcmc.b <- replicate(B, fit.mcmc('ky', tree=ie.b.trees, prior=prior), simplify=F)
```

Reflexes of $*b$:

```
b.mcmc.b <- replicate(B, fit.mcmc('b', tree=ie.b.trees, prior=prior), simplify=F)
```

S4.3.5 Results

Figure S4.4 summarizes the models in terms of the logarithm of the Bayes Factor (logBF) across replications and phylogenies. logBF is defined as double the difference between the log marginal likelihood of the more complex model (assuming unequal rates) and the log marginal likelihood of the simpler model (assuming equal rates). Values above 2 are usually considered to support the complex model and this is indicated by light blue shading in the graphs. Only the correspondemes $*m$, $*g^w$, and $*b$ show evidence for more

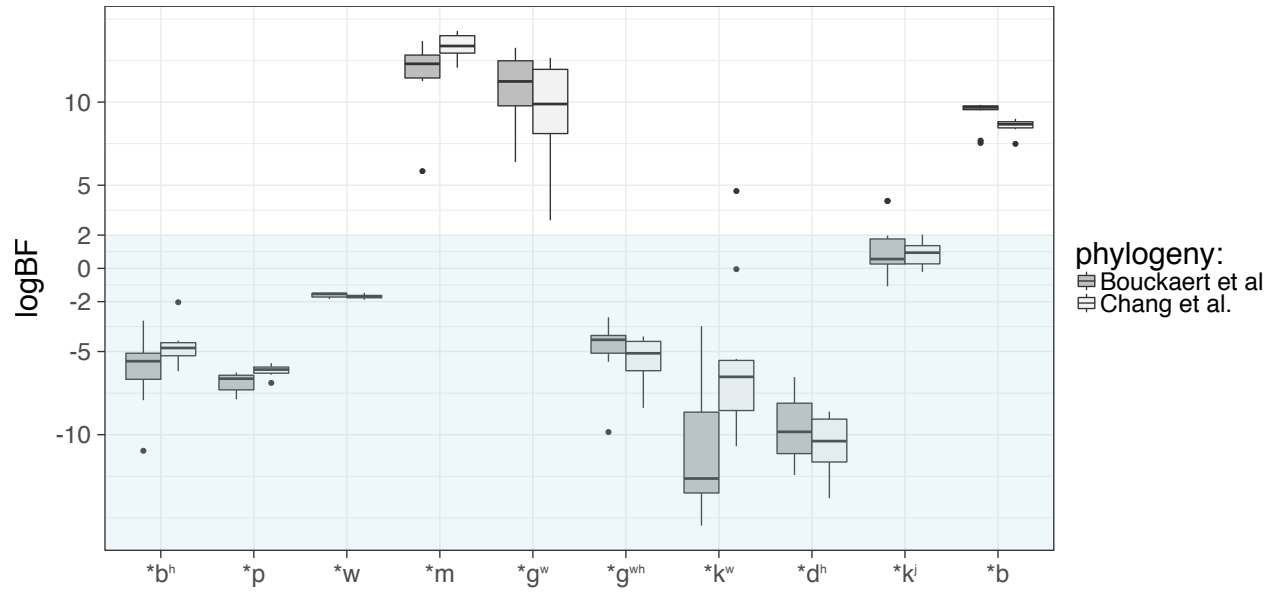


Figure S4.4: Bayes Factor evidence for different-rate model compared to equal-rate model. The blue shading covers the range conventionally considered not to give evidence.

complex models with unequal rates. The rate estimates themselves are plotted in Figure S4.5. The three correspondemes with unequal rates are more likely to lose labiodental reflexes than to gain them.

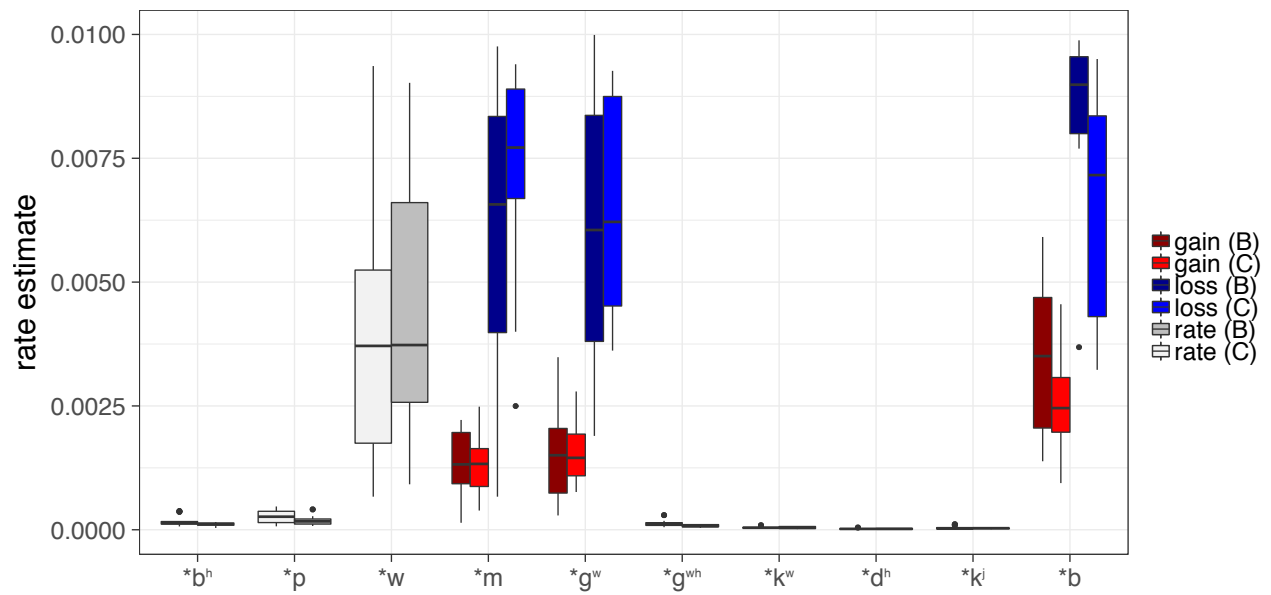


Figure S4.5: Estimated rates of gain and loss of labiodentals in the best models. Gray colors indicate that the best model assumes equal rates of gain and loss. Dark colors indicate phylogenies from Bouckaert et al (B), light colors phylogenies from Chang et al. (C).

S4.4 Ancestral state estimates

CTMC models include an estimate on the probability distribution of the root value. We extract this from the best-fitting models and report the results in Figure S4.6.

```
posterior.size=(mcmc.iterations-mcmc.burn.in)/mcmc.sampling.rate
ace.df <- data.frame(
  phylogeny=c(rep("Chang et al.'s phylogeny",
    B*length(correspondemes)*posterior.size),
    rep("Bouckaert et al.'s phylogeny",
    B*length(correspondemes)*posterior.size)),
  correspondeme=factor(rep(
    unlist(lapply(correspondemes,rep,B*posterior.size)),
    2),
    levels=correspondemes,
    labels=c("*b ", "*p", "*w", "*m", "*g ", "*g ", "*k ", "*d ", "*k ", "*b")),
  Prob=c(unlist(lapply(correspondemes, function(crsp) {
    sapply(get(paste0(crsp, '.mcmc.c')), function(x) {
      if(crsp %in% c('*m', '*b', '*g ')) { x$ARD$Root.P.Y. } else {
        x$ER$Root.P.Y.
      }
    }
  } )),
  unlist(lapply(correspondemes, function(crsp) {
    sapply(get(paste0(crsp, '.mcmc.b')), function(x) {
      if(crsp %in% c('*m', '*b', '*g ')) { x$ARD$Root.P.Y. } else {
        x$ER$Root.P.Y.
      }
    }
  } )),
  )
)
```

Figure S4.6 suggests that for 8 correspondemes, the probability of a labiodental realization in proto-Indo-European was very low. For two correspondemes — **w* and **b* — the ancestral state cannot be sufficiently resolved and is estimated at or close to 50% probabilities. In these two cases, there is no decisive evidence in favor of nor against a labiodental realization in proto-Indo-European. One way of resolving this issue is to fit and compare models with vs without assumed proto-states.

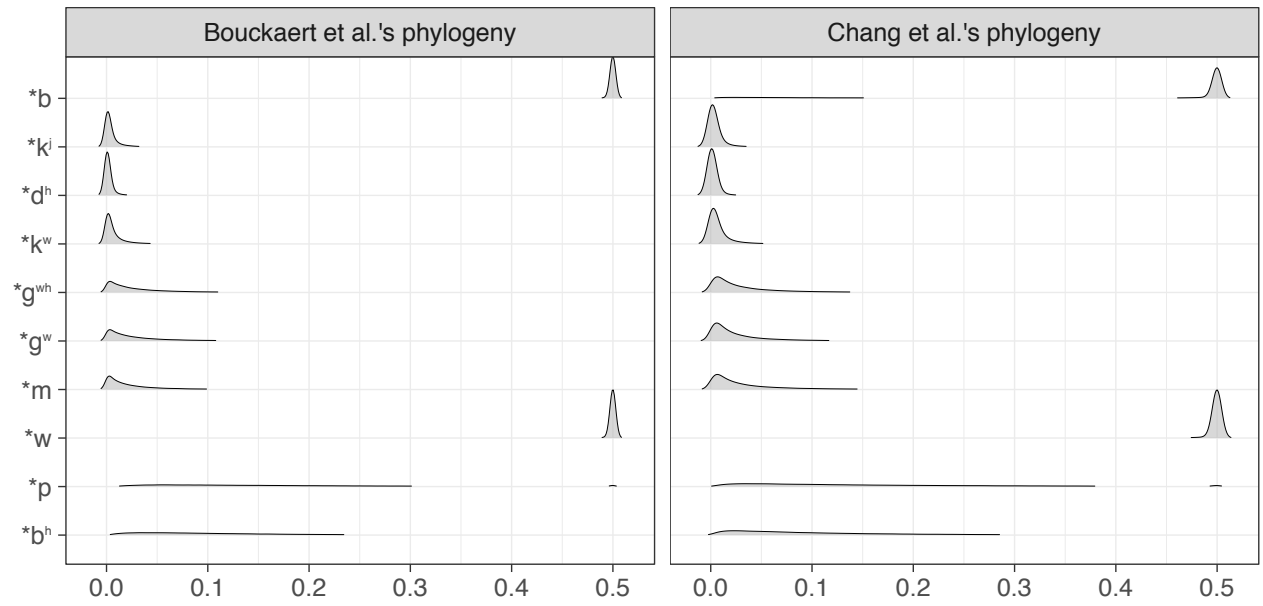


Figure S4.6: Posterior probabilities of labiodentals in PIE, based on the best-fitting models

S4.4.1 Reconstruction comparing assumed proto-states

S4.4.1.1 Based on the phylogenies from Chang et al.

Reconstruction of **w*:

```
PIE.had.labiodentals.c <- paste("Y", paste(ie.c.trees[[1]]$tip.label, collapse=" "))
PIE.lacked.labiodentals.c <- paste("N", paste(ie.c.trees[[1]]$tip.label, collapse=" "))

w.mcmc.c.Y <- replicate(B, fit.mcmc('w', tree=ie.c.trees, prior=prior,
                                   assumed.mrca=PIE.had.labiodentals.c), simplify=F)
w.mcmc.c.N <- replicate(B, fit.mcmc('w', tree=ie.c.trees, prior=prior,
                                   assumed.mrca=PIE.lacked.labiodentals.c), simplify=F)
```

The best fitting model is:

```
unique(model.fits(w.mcmc.c.Y)[,2])
```

```
## [1] "ER"
```

```
median(model.fits(w.mcmc.c.Y)[,1])
```

```
## [1] 1.791384
```

There is no evidence for or against a model that assumes labiodentals:

```
fossil.fits(w.mcmc.c.Y, w.mcmc.c.N, model='ER')
```

```
##      BayesFactor BetterModel
## 1      0.004066      Model 2
## 2      0.000848      Model 1
## 3      0.013824      Model 2
## 4      0.005306      Model 2
## 5      0.000482      Model 1
## 6      0.035176      Model 2
## 7      0.163120      Model 1
## 8      0.020978      Model 2
## 9      0.003164      Model 1
## 10     0.000472      Model 2
```

Reconstruction of **b*:

```
b.mcmc.c.Y <- replicate(B, fit.mcmc('b', tree=ie.c.trees, prior=prior,
                                   assumed.mrca=PIE.had.labiodentals.c), simplify=F)
b.mcmc.c.N <- replicate(B, fit.mcmc('b', tree=ie.c.trees, prior=prior,
                                   assumed.mrca=PIE.lacked.labiodentals.c), simplify=F)
```

The best fitting model is:

```
unique(model.fits(b.mcmc.c.Y)[,2])
```

```
## [1] "ARD"
```

```
median(model.fits(b.mcmc.c.Y)[,1])
```

```
## [1] 9.792657
```

There is no evidence for or against a model that assumes labiodentals:

```
fossil.fits(b.mcmc.c.Y, b.mcmc.c.N, model='ARD')
```

```
##      BayesFactor BetterModel
## 1      0.336638      Model 1
## 2      0.384274      Model 2
## 3      0.516128      Model 1
## 4      0.282814      Model 2
## 5      2.060142      Model 1
## 6      0.858620      Model 1
## 7      0.392786      Model 2
## 8      2.880922      Model 2
## 9      0.293846      Model 1
## 10     0.041236      Model 2
```

S4.4.1.2 Based on the phylogenies from Bouckaert et al.

Reconstruction of w :

```
PIE.had.labiodontals.b <- paste("Y", paste(ie.b.trees[[1]]$tip.label, collapse=" "))
PIE.lacked.labiodontals.b <- paste("N", paste(ie.b.trees[[1]]$tip.label, collapse=" "))

w.mcmc.b.Y <- replicate(B, fit.mcmc('w', tree=ie.b.trees, prior=prior,
                                   assumed.mrca=PIE.had.labiodontals.b), simplify=F)
w.mcmc.b.N <- replicate(B, fit.mcmc('w', tree=ie.b.trees, prior=prior,
                                   assumed.mrca=PIE.lacked.labiodontals.b), simplify=F)
```

The best fitting model is:

```
unique(model.fits(w.mcmc.b.Y)[,2])
```

```
## [1] "ER"
```

```
median(model.fits(w.mcmc.b.Y)[,1])
```

```
## [1] 1.622678
```

There is no evidence for or against a model that assumes labiodontals:

```
fossil.fits(w.mcmc.b.Y, w.mcmc.b.N, model='ER')
```

```
##      BayesFactor BetterModel
## 1      0.015978      Model 2
## 2      0.019660      Model 2
## 3      0.016286      Model 1
## 4      0.004662      Model 2
## 5      0.011198      Model 2
## 6      0.017590      Model 2
## 7      0.013098      Model 2
## 8      0.007294      Model 2
## 9      0.045020      Model 1
## 10     0.005688      Model 2
```

Reconstruction of b :

```
b.mcmc.b.Y <- replicate(B, fit.mcmc('b', tree=ie.b.trees, prior=prior,
                                   assumed.mrca=PIE.had.labiodontals.b), simplify=F)
b.mcmc.b.N <- replicate(B, fit.mcmc('b', tree=ie.b.trees, prior=prior,
                                   assumed.mrca=PIE.lacked.labiodontals.b), simplify=F)
```

The best fitting model is:

```
unique(model.fits(b.mcmc.b.Y)[,2])
```

```
## [1] "ARD"
```

```
median(model.fits(b.mcmc.b.Y)[,1])
```

```
## [1] 9.851225
```

There is no evidence for or against a model that assumes labiodentals:

```
fossil.fits(b.mcmc.b.Y, b.mcmc.b.N, model='ARD')
```

```
##      BayesFactor BetterModel
## 1      0.557948      Model 1
## 2      2.714010      Model 1
## 3      0.250344      Model 1
## 4      0.436856      Model 2
## 5      0.670000      Model 1
## 6      1.626904      Model 2
## 7      0.274046      Model 1
## 8      0.205004      Model 1
## 9      0.600036      Model 1
## 10     4.628180      Model 1
```

S4.4.2 Summary

In summary, there is no evidence for or against positing labiodentals as ancestral states of the **w* and **b* correspondemes. In the case of **b* we earlier observed a diachronic bias towards *loss* of labiodental realizations (Figure S4.5). The rate estimates for **w* suggest general instability, with equally high rates of gain and loss. This confirms the qualitative observation that this correspondeme varies substantially in closely related languages (e.g. German vs. English, Spanish vs. Gallo-Romance) and its realization is highly unstable in some extant languages. For example in Hindi, there is substantial variation between [w], [β], or [v] (13).

S4.5 Stochastic Character Mapping

Stochastic Character Mapping (4, 5) samples a posterior distribution of character histories given a phylogeny, the current states, and transition rates. We estimate maps using the rate estimates from the best-fitting models estimated in the previous section. The maps are estimated using the algorithm in `phytools::make.simmap` (6) and relies on Maximum Clade Credibility (MCC) summary trees for easier visualization as probability changes along each branch. Ancestral states at nodes however are estimates using posterior phylogeny samples and are plotted as pie charts.

In all models, we assume constant rates of change over time. This assumption is conservative and disfavors our hypothesis because it will over-estimate the age of innovations. Non-constant rates of change would allow for a late acceleration of innovations.

S4.5.1 Procedures

```
compute.simmap <- function(variable, tree=ie.c.sum.tree, data=labiodental.data,
  model=c('ER','ARD'), iterations=B, resolution=res) {
  # get the right model based on the tree
  if(grepl('c', deparse(substitute(tree)))) {type='c'} else {type='b'}

  # get Q matrix from BayesTraits results:
  mcmc.results <- get(paste0(tolower(variable), '.mcmc.', type))
  Q.matrix <- transition.matrix(mcmc.results, 'median', model, prob=F)
  # prepare data for simmap, catching NAs (which will be estimated)
  nas <- c('-', '?') # catch NAs
  states <- data[, variable]
  names(states) <- data$language
  state.symbols <- sort(setdiff(as.character(unique(states)), nas))
  n.states <- length(state.symbols)
  # estimating NAs with a uniform prior:
  if(any(states %in% nas | is.na(states))) {
    states <- to.matrix(states, state.symbols)
    states[rowSums(states)==0,] <- rep(1/n.states)
  }

  # make simmaps with a flat root prior:
  treemaps <- make.simmap(tree, states, model=model, Q=Q.matrix,
    pi='estimated', nsim=iterations, message=F)
}
```

The following function estimates SCMs on a single tree, but it adds node estimates based on a posterior tree sample:

```
plot.density <- function(variable='bh', tree=ie.c.sum.tree,
  nodes=T, multi.ace=T, legend=2000, lwd=c(8,8)) {
  if(grepl('c', deparse(substitute(tree)))) {type='c'} else {type='b'}
  # get the density map:
  density.map <- get(paste0(tolower(variable), '.density.', type))
  density.map$tree <- ladderize.simmap(density.map$tree)
  plot(density.map, outline=T, lwd=lwd,
    fsize=c(1,1), ftype='reg', legend=legend)
  if(nodes) {
    # get the nodes that are ancestral and different from ancient languages,
```

```

# depending on the tree:
switch(type,
  c = nodes <- c(53:59, 70, 79, 84, 99, 100, 103),
  # nodes <- c(53:61, 66:73, 75, 77, 80:84, 86:95, 97:100, 103)
  b = nodes <- c(53:59, 65, 70, 79, 84, 99, 100, 102:103)
)
# reconstruct based on stochastic mapping on a single tree or on a sample of trees:
if(multi.ace) {scm <- '.multi.simmmaps.'} else {scm <- '.simmmaps.'}
simmmaps <- get(paste0(tolower(variable), scm, type))
node.aces <- as.data.frame(describe.simmmap(simmmaps)$ace)
node.aces$node <- rownames(node.aces)
# and plot:
nodelabels(node=nodes, pie=as.matrix(node.aces[node.aces$node %in% nodes, 1:2]),
  piecol=c('blue','red'), cex=.7)
}
}

```

S4.5.2 Parameters

Number of simulated histories per tree:

```
B=100
```

Resolution (bins): the number of time intervals (bins) is fixed by the resolution parameter of `phytools::densityMap`, which we set to:

```
res=200
```

Setting the parameter to this value results in bins of about 30 years, roughly reflecting the state at one generation of speakers.

S4.5.3 Individual analyses

A full analysis on the posterior tree sample is computationally extremely expensive (B simulations \times 10 correspondemes \times 20,000 trees). In response to this, we use only the last 1000 trees from the posterior sample of each phylogeny:

```
ie.c.trees.sample <- tail(ie.c.trees, 1000)
ie.b.trees.sample <- tail(ie.b.trees, 1000)
```

The results of this is displayed as pie charts at the nodes. But the stochastic character maps along the branches are based on MCC summary trees.

S4.5.3.1 Reconstruction of labiodental reflexes for $*b^h$

With phylogenies estimated by Chang et al.:

```
correspondeme='bh'
bh.simmmaps.c <- compute.simmmap(correspondeme, tree=ie.c.sum.tree, model='ER')
bh.density.c <- densityMap(bh.simmmaps.c, plot=F, res=res)
bh.multi.simmmaps.c <- compute.simmmap(correspondeme, tree=ie.c.trees.sample, model='ER')
```

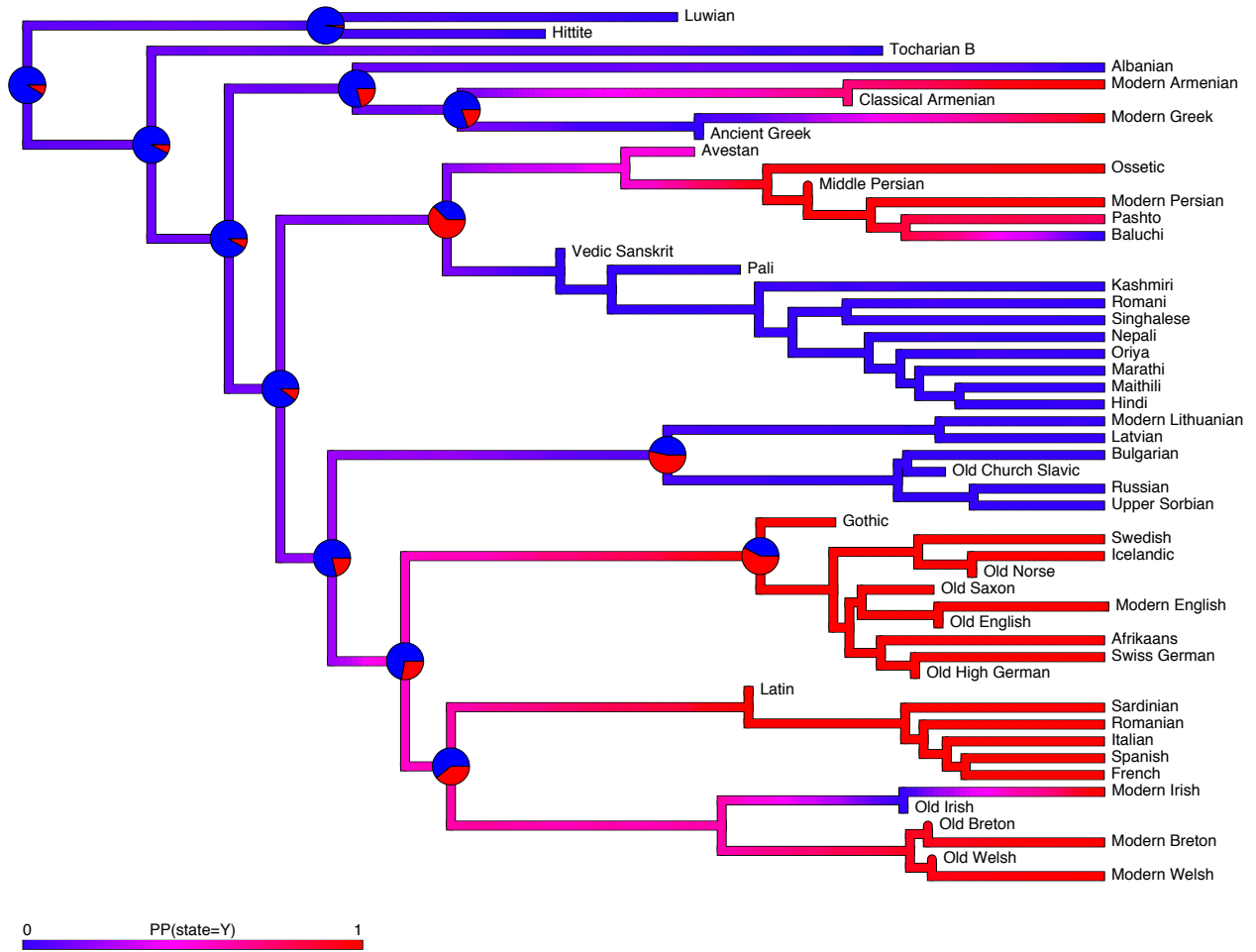


Figure S4.7: Posterior probability (PP) density of labiodental reflexes in some contexts of the correspondeme $*b^h$, assuming the phylogenies estimated by Chang et al.

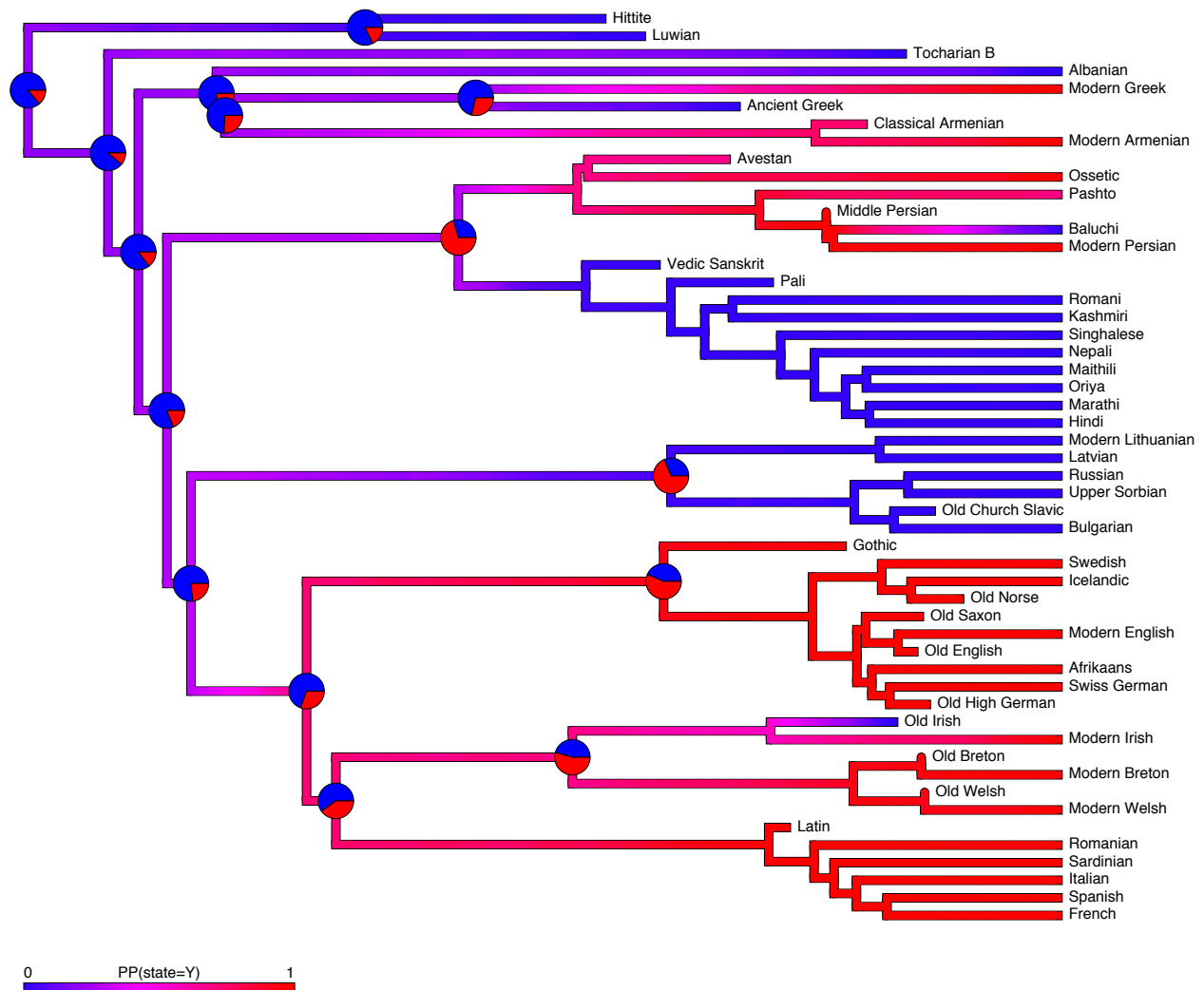


Figure S4.8: Posterior probability (PP) density of labiodental reflexes in some contexts of the correspondeme $*b^h$, assuming the phylogenies estimated by Bouckaert et al.

With phylogenies estimated by Bouckaert et al.:

```

bh.simmmaps.b <- compute.simmmap(correspondeme, tree=ie.b.sum.tree, model='ER')
bh.density.b <- densityMap(bh.simmmaps.b, plot=F, res=res)
bh.multi.simmmaps.b <- compute.simmmap(correspondeme, tree=ie.b.trees.sample, model='ER')

```

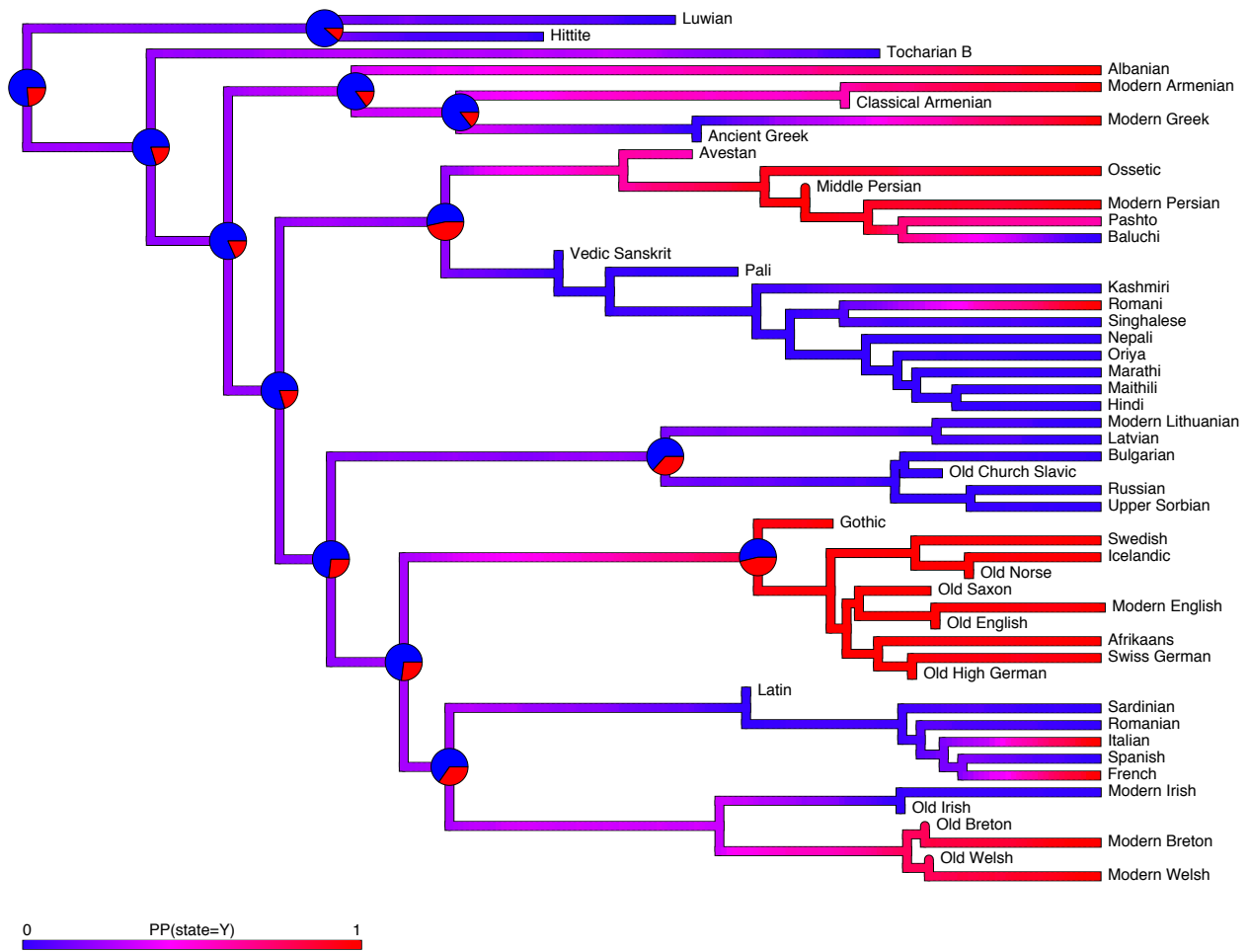



Figure S4.9: Posterior probability (PP) density of labiodental reflexes in some contexts of the correspondeme $*p$, assuming the phylogenies estimated by Chang et al.

S4.5.3.2 Reconstruction of labiodental reflexes for $*p$

With phylogenies estimated by Chang et al.:

```
correspondeme='p'
p.simmmaps.c <- compute.simmmap(correspondeme, tree=ie.c.sum.tree, model='ER')
p.density.c <- densityMap(p.simmmaps.c, plot=F, res=res)
p.multi.simmmaps.c <- compute.simmmap(correspondeme, tree=ie.c.trees.sample, model='ER')
```

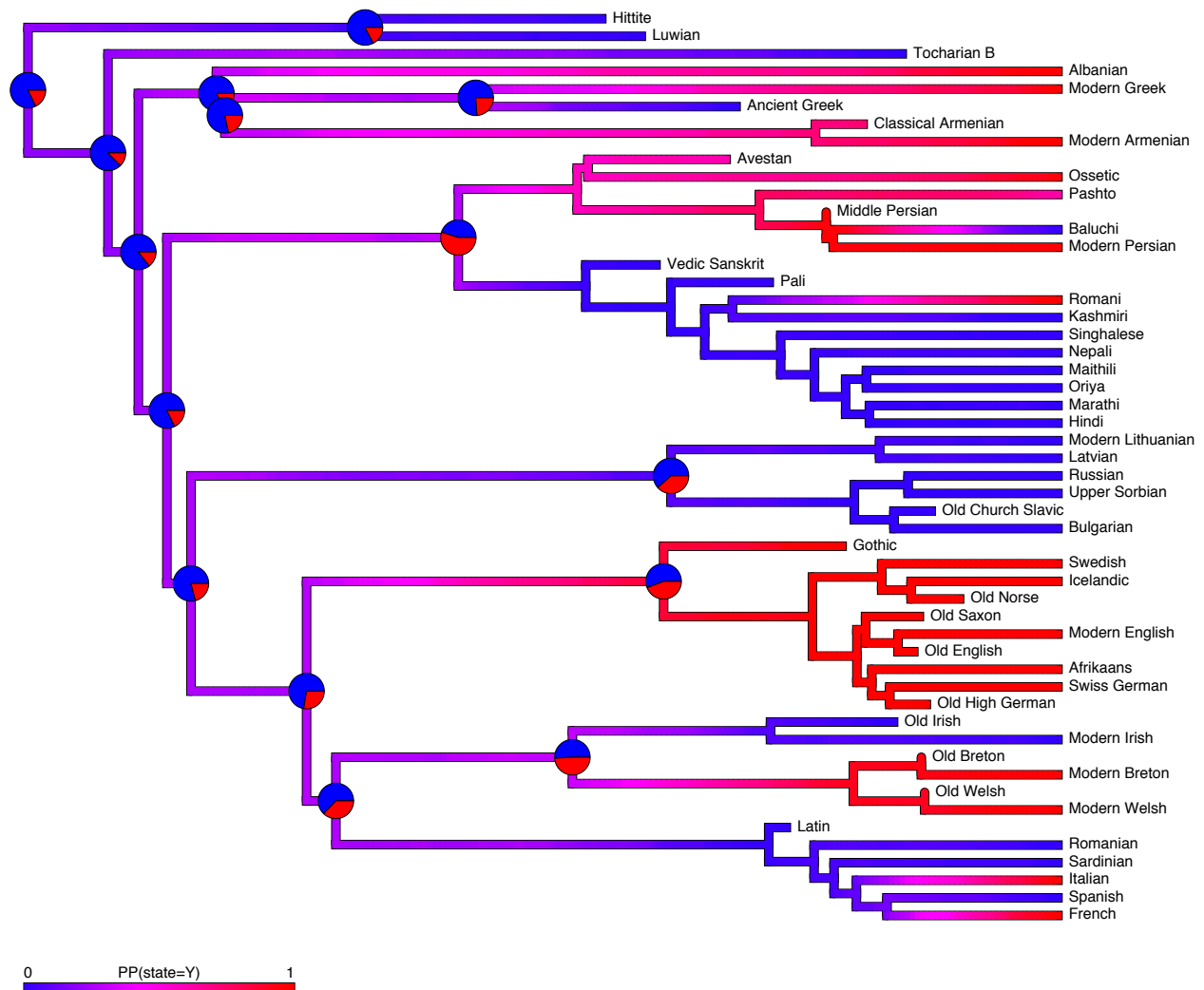


Figure S4.10: Posterior probability (PP) density of labiodental reflexes in some contexts of the correspondeme $*p$, assuming the phylogenies estimated by Bouckaert et al.

With phylogenies estimated by Bouckaert et al.:

```
p.simmmaps.b <- compute.simmmap(correspondeme, tree=ie.b.sum.tree, model='ER')
p.density.b <- densityMap(p.simmmaps.b, plot=F, res=res)
p.multi.simmmaps.b <- compute.simmmap(correspondeme, tree=ie.b.trees.sample, model='ER')
```



Figure S4.11: Posterior probability (PP) density of labiodental reflexes in some contexts of the correspondeme $*w$, assuming the phylogenies estimated by Chang et al.

S4.5.3.3 Reconstruction of labiodental reflexes for $*w$

With phylogenies estimated by Chang et al.:

```
correspondeme='w'
w.simmmaps.c <- compute.simmmap(correspondeme, tree=ie.c.sum.tree, model='ER')
w.density.c <- densityMap(w.simmmaps.c, plot=F, res=res)
w.multi.simmmaps.c <- compute.simmmap(correspondeme, tree=ie.c.trees.sample, model='ER')
```

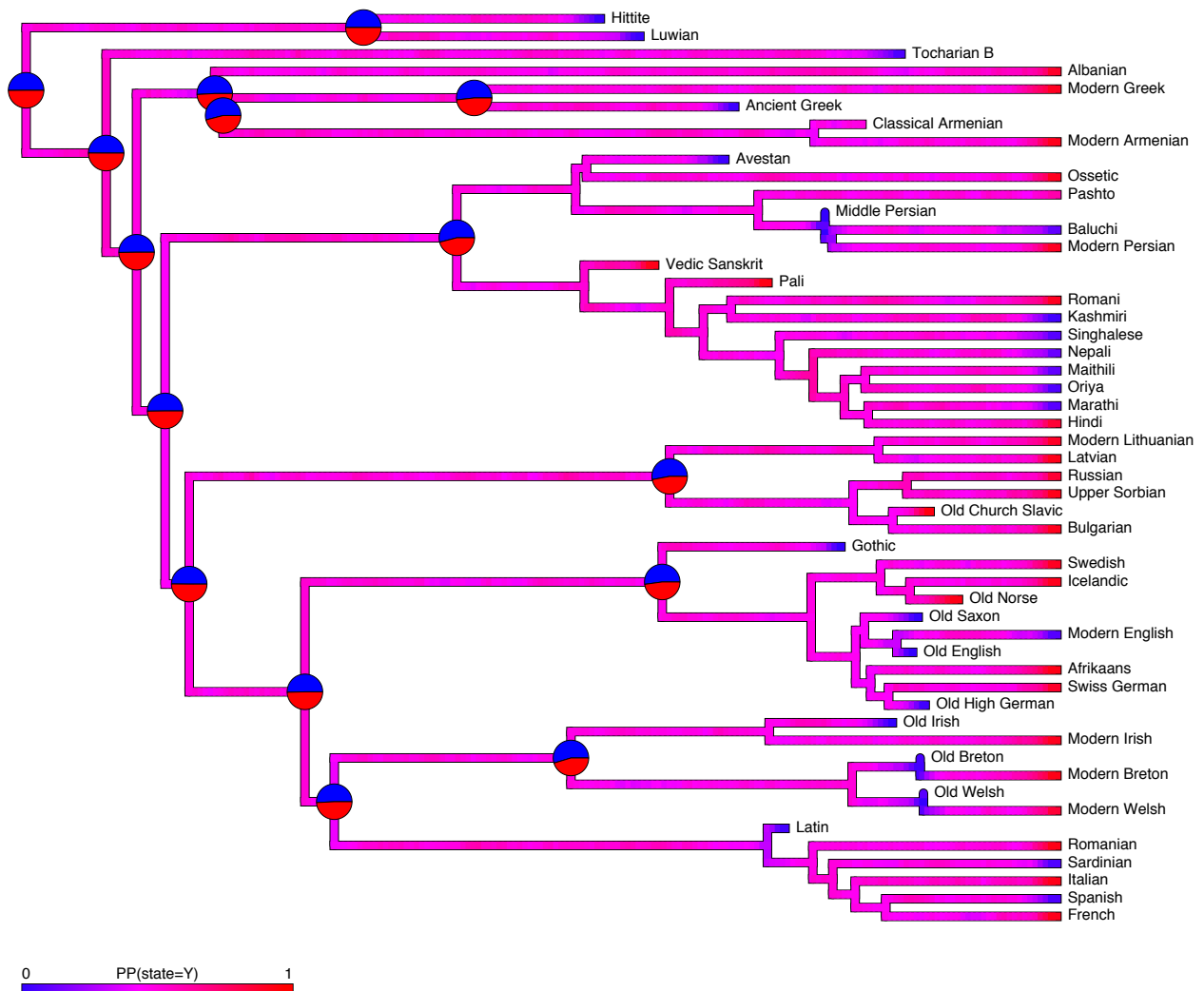


Figure S4.12: Posterior probability (PP) density of labiodental reflexes in some contexts of the correspondeme **w*, assuming the phylogenies estimated by Bouckaert et al.

With phylogenies estimated by Bouckaert et al.:

```
w.simmmaps.b <- compute.simmmap(correspondeme, tree=ie.b.sum.tree, model='ER')
w.density.b <- densityMap(w.simmmaps.b, plot=F, res=res)
w.multi.simmmaps.b <- compute.simmmap(correspondeme, tree=ie.b.trees.sample, model='ER')
```

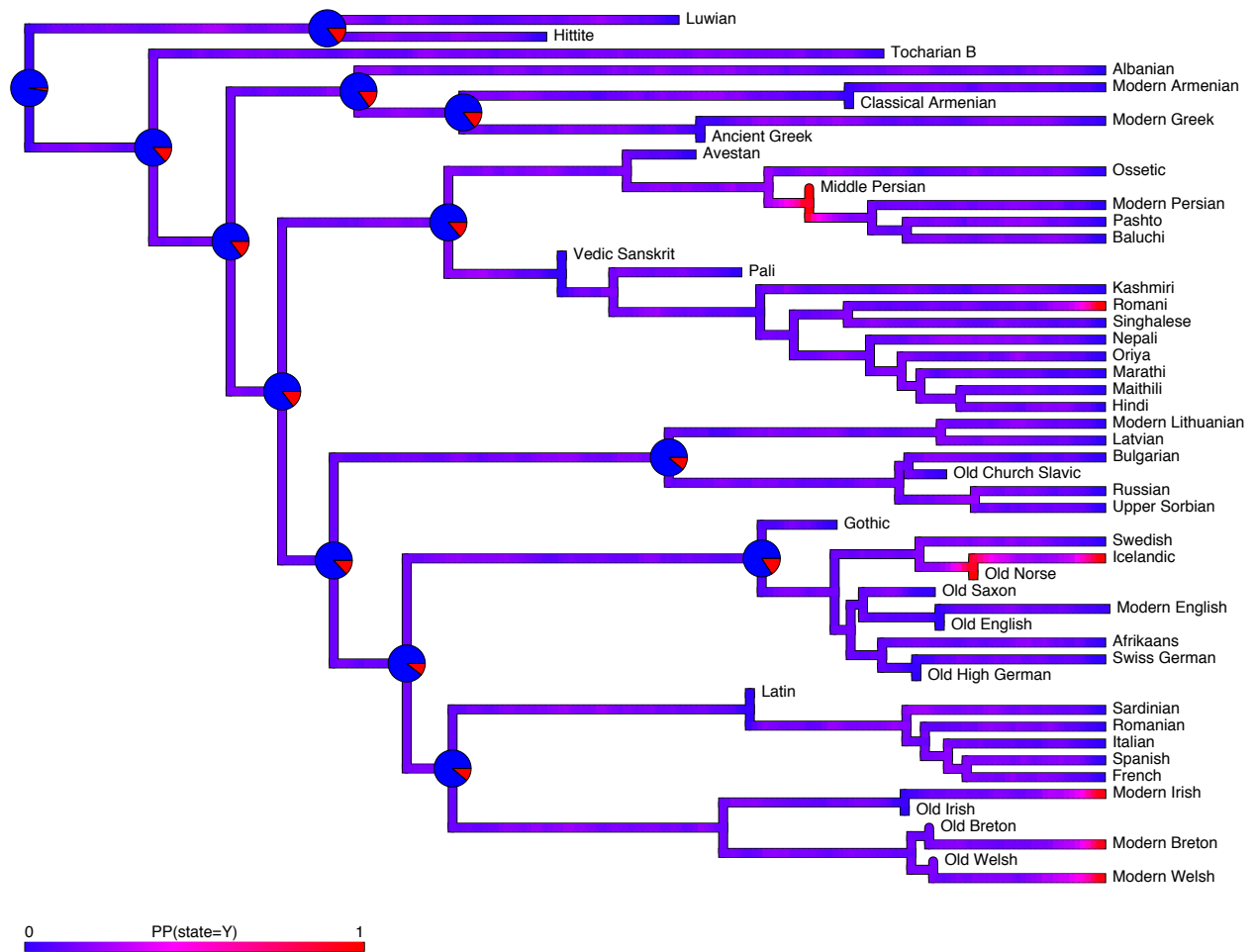


Figure S4.13: Posterior probability (PP) density of labiodental reflexes in some contexts of the correspondeme $*m$, assuming the phylogenies estimated by Chang et al.

S4.5.3.4 Reconstruction of labiodental reflexes for $*m$

With phylogenies estimated by Chang et al.:

```
correspondeme='m'
m.simmmaps.c <- compute.simmmap(correspondeme, tree=ie.c.sum.tree, model='ARD')
m.density.c <- densityMap(m.simmmaps.c, plot=F, res=res)
m.multi.simmmaps.c <- compute.simmmap(correspondeme, tree=ie.c.trees.sample, model='ARD')
```

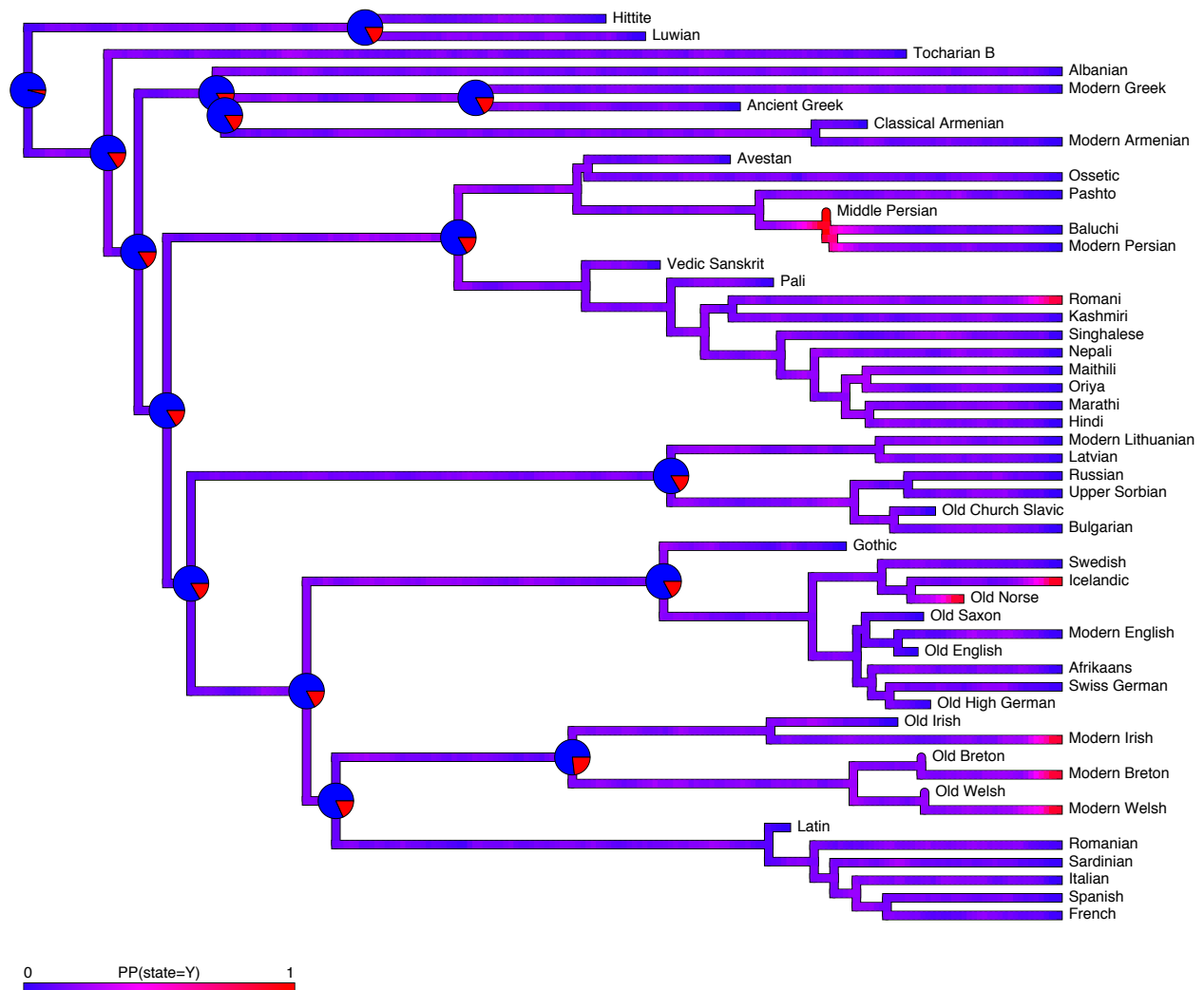


Figure S4.14: Posterior probability (PP) density of labiodental reflexes in some contexts of the correspondeme $*m$, assuming the phylogenies estimated by Bouckaert et al.

With phylogenies estimated by Bouckaert et al.:

```
m.simmmaps.b <- compute.simmmap(correspondeme, tree=ie.b.sum.tree, model='ARD')
m.density.b <- densityMap(m.simmmaps.b, plot=F, res=res)
m.multi.simmmaps.b <- compute.simmmap(correspondeme, tree=ie.b.trees.sample, model='ARD')
```

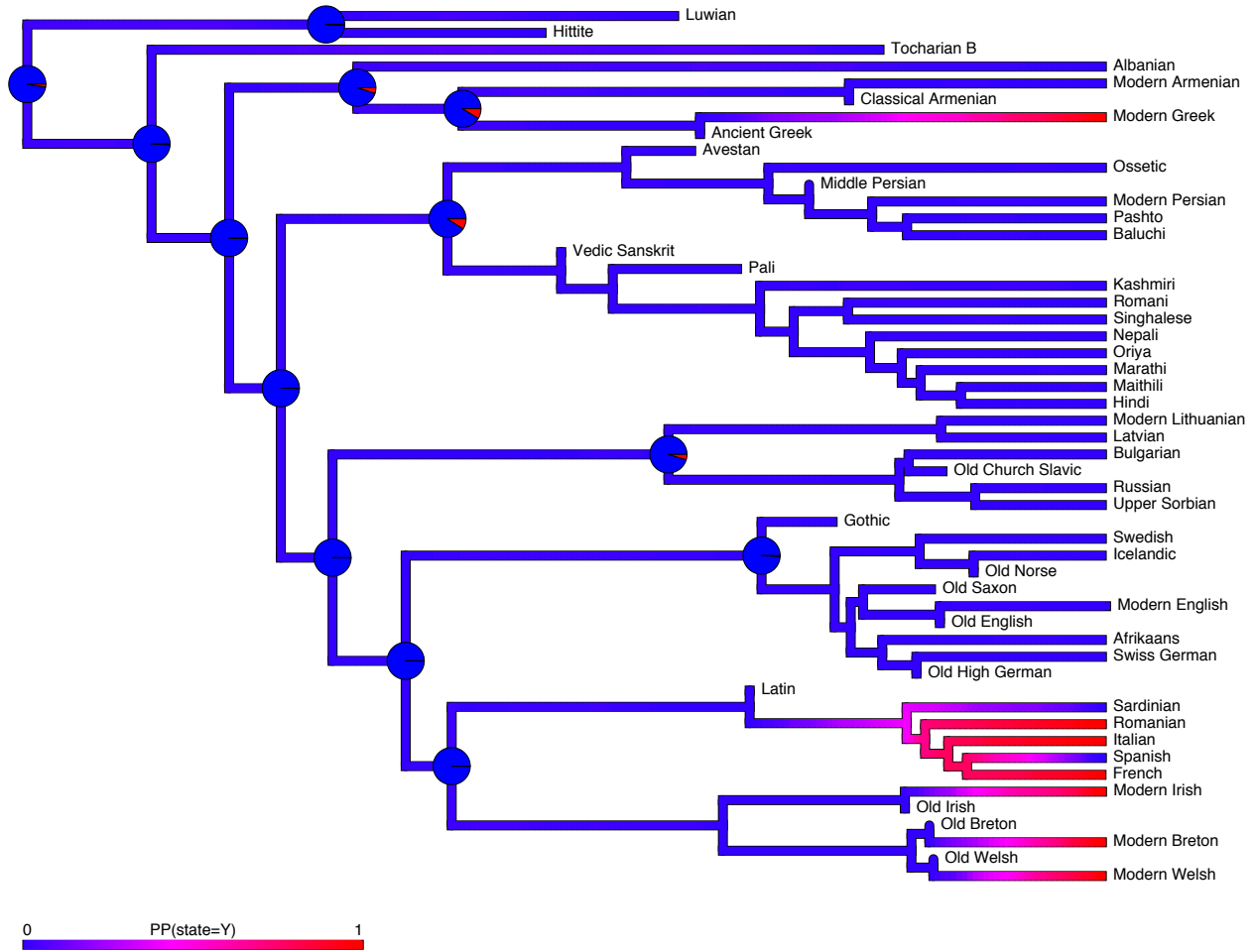


Figure S4.15: Posterior probability (PP) density of labiodental reflexes in some contexts of the correspondeme $*g^w$, assuming the phylogenies estimated by Chang et al.

S4.5.3.5 Reconstruction of labiodental reflexes for $*g^w$

With phylogenies estimated by Chang et al.:

```
correspondeme='gw'
gw.simmmaps.c <- compute.simmmap(correspondeme, tree=ie.c.sum.tree, model='ER')
gw.density.c <- densityMap(gw.simmmaps.c, plot=F, res=res)
gw.multi.simmmaps.c <- compute.simmmap(correspondeme, tree=ie.c.trees.sample, model='ER')
```

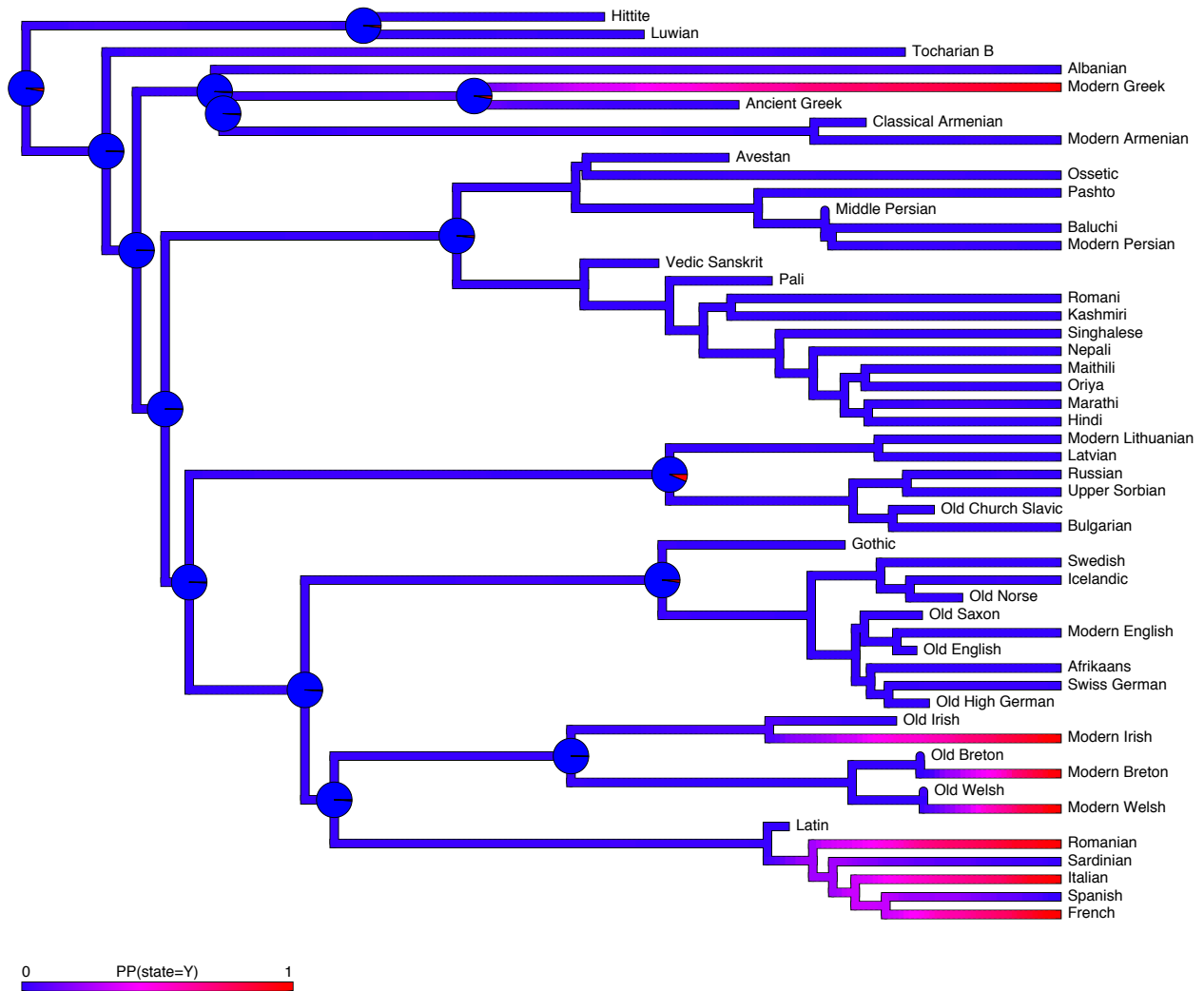


Figure S4.16: Posterior probability (PP) density of labiodental reflexes in some contexts of the correspondeme $*g^w$, assuming the phylogenies estimated by Bouckaert et al.

With phylogenies estimated by Bouckaert et al.:

```
gw.simmmaps.b <- compute.simmmap(correspondeme, tree=ie.b.sum.tree, model='ER')
gw.density.b <- densityMap(gw.simmmaps.b, plot=F, res=res)
gw.multi.simmmaps.b <- compute.simmmap(correspondeme, tree=ie.b.trees.sample, model='ER')
```

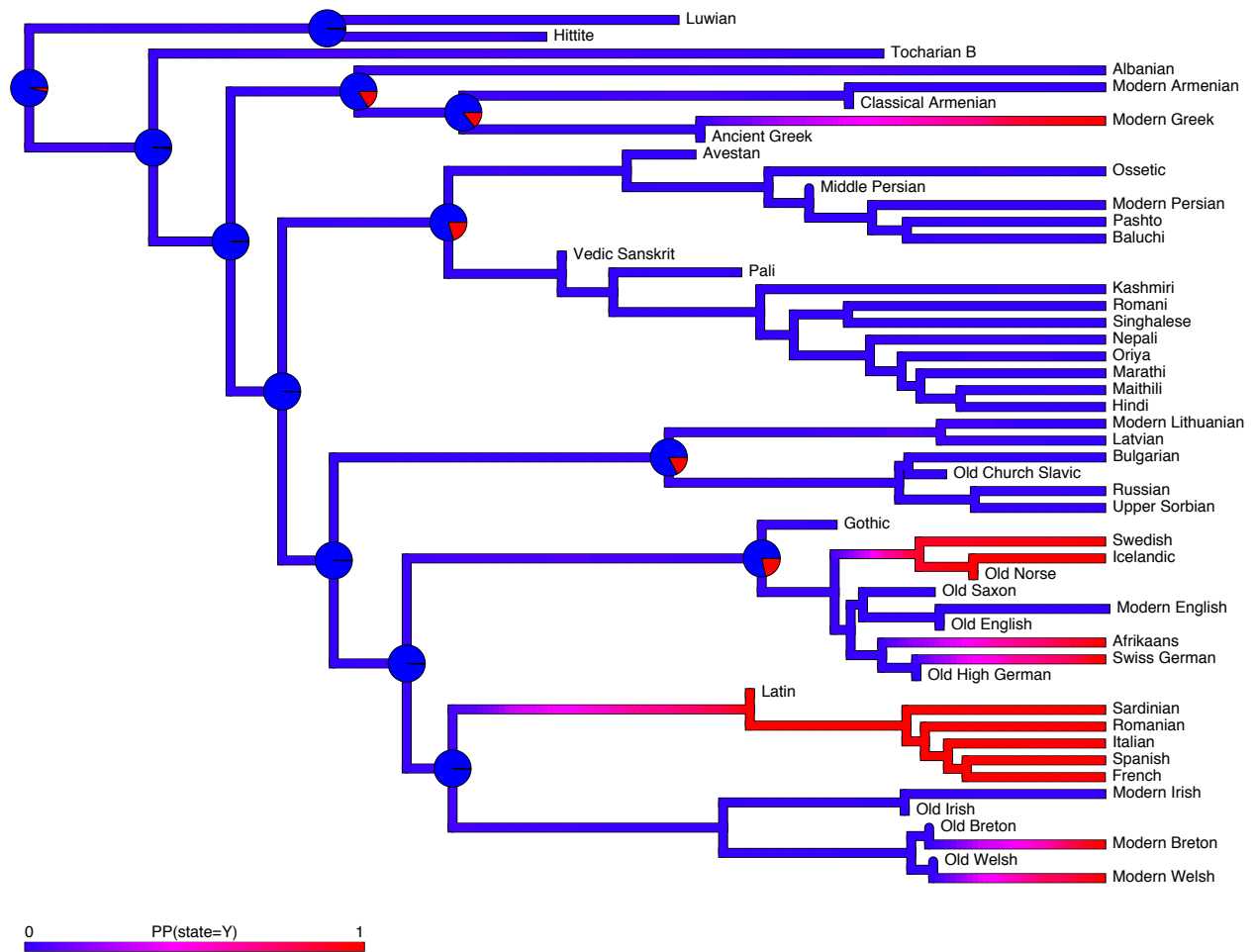



Figure S4.17: Posterior probability (PP) density of labiodental reflexes in some contexts of the correspondeme $*g^{wh}$, assuming the phylogenies estimated by Chang et al.

S4.5.3.6 Reconstruction of labiodental reflexes for $*g^{wh}$

With phylogenies estimated by Chang et al.:

```
correspondeme='gwh'
gwh.simmmaps.c <- compute.simmmap(correspondeme, tree=ie.c.sum.tree, model='ER')
gwh.density.c <- densityMap(gwh.simmmaps.c, plot=F, res=res)
gwh.multi.simmmaps.c <- compute.simmmap(correspondeme, tree=ie.c.trees.sample, model='ER')
```

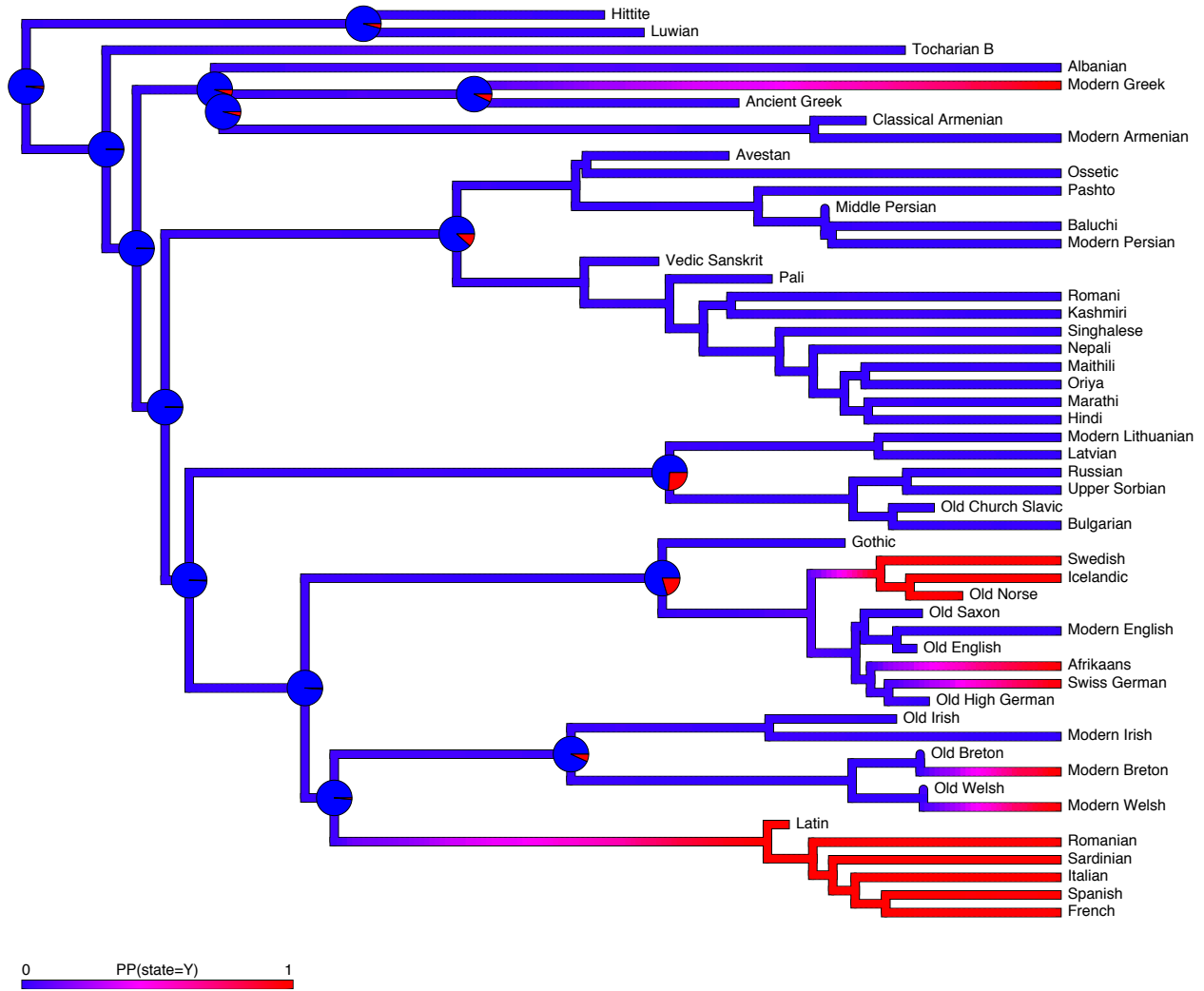


Figure S4.18: Posterior probability (PP) density of labiodental reflexes in some contexts of the correspondeme $*g^{wh}$, assuming the phylogenies estimated by Bouckaert et al.

With phylogenies estimated by Bouckaert et al.:

```
gwh.simmmaps.b <- compute.simmmap(correspondeme, tree=ie.b.sum.tree, model='ER')
gwh.density.b <- densityMap(gwh.simmmaps.b, plot=F, res=res)
gwh.multi.simmmaps.b <- compute.simmmap(correspondeme, tree=ie.b.trees.sample, model='ER')
```

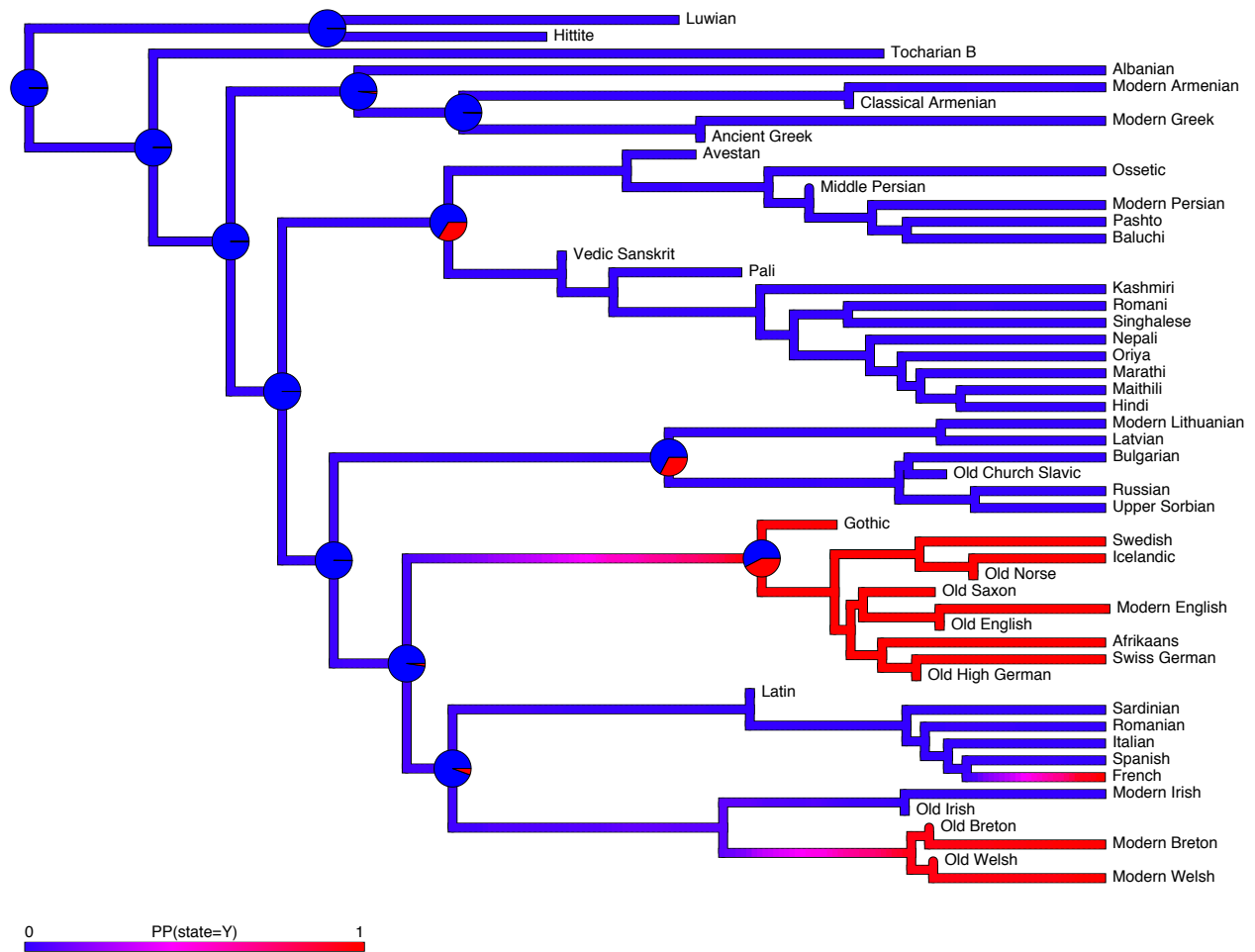


Figure S4.19: Posterior probability (PP) density of labiodental reflexes in some contexts of the correspondeme $*k^w$, assuming the phylogenies estimated by Chang et al.

S4.5.3.7 Reconstruction of labiodental reflexes for $*k^w$

With phylogenies estimated by Chang et al.:

```
correspondeme='kw'
kw.simmmaps.c <- compute.simmmap(correspondeme, tree=ie.c.sum.tree, model='ER')
kw.density.c <- densityMap(kw.simmmaps.c, plot=F, res=res)
kw.multi.simmmaps.c <- compute.simmmap(correspondeme, tree=ie.c.trees.sample, model='ER')
```

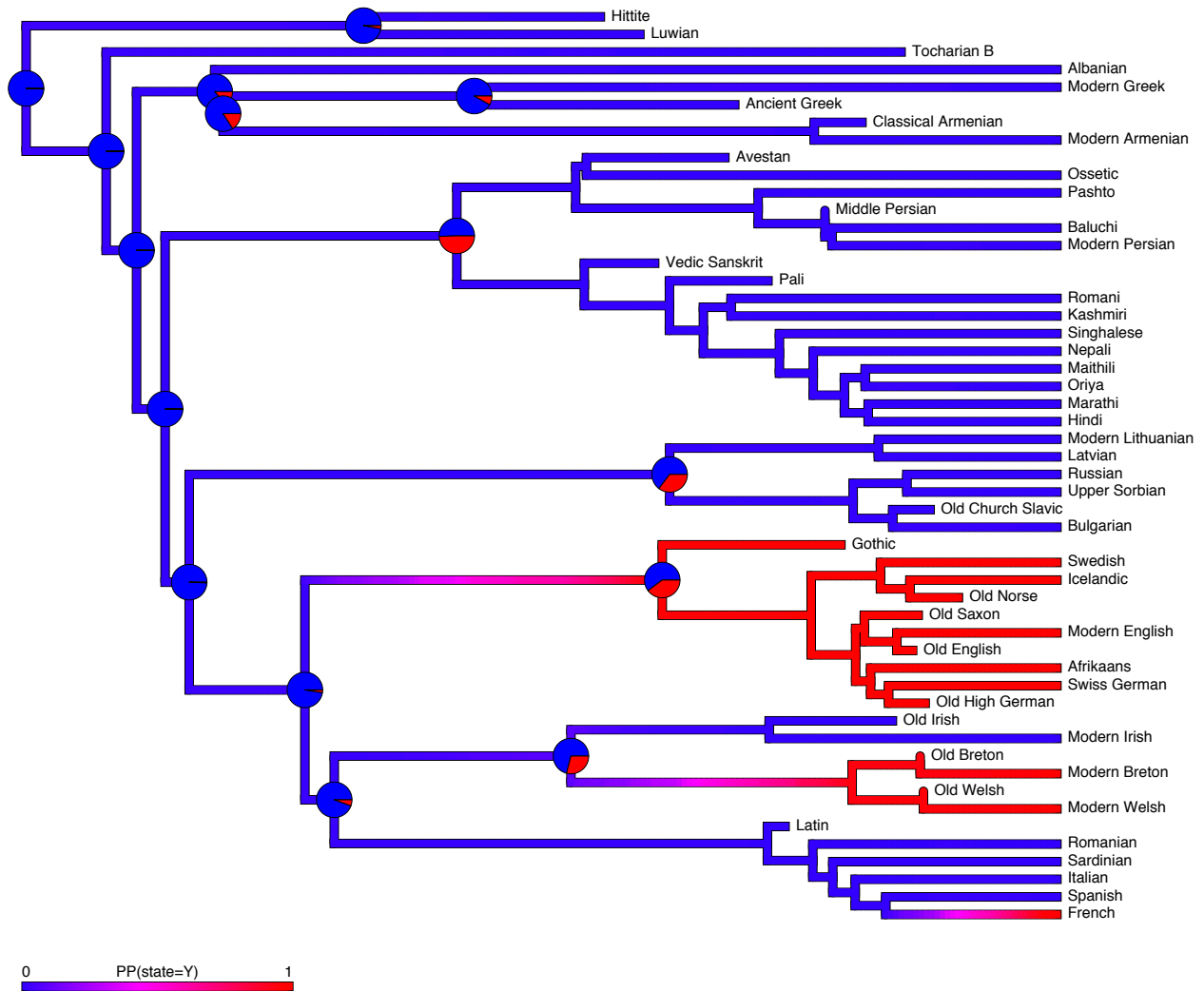


Figure S4.20: Posterior probability (PP) density of labiodental reflexes in some contexts of the correspondeme $*k^w$, assuming the phylogenies estimated by Bouckaert et al.

With phylogenies estimated by Bouckaert et al.:

```
kw.simmmaps.b <- compute.simmmap(correspondeme, tree=ie.b.sum.tree, model='ER')
kw.density.b <- densityMap(kw.simmmaps.b, plot=F, res=res)
kw.multi.simmmaps.b <- compute.simmmap(correspondeme, tree=ie.b.trees.sample, model='ER')
```

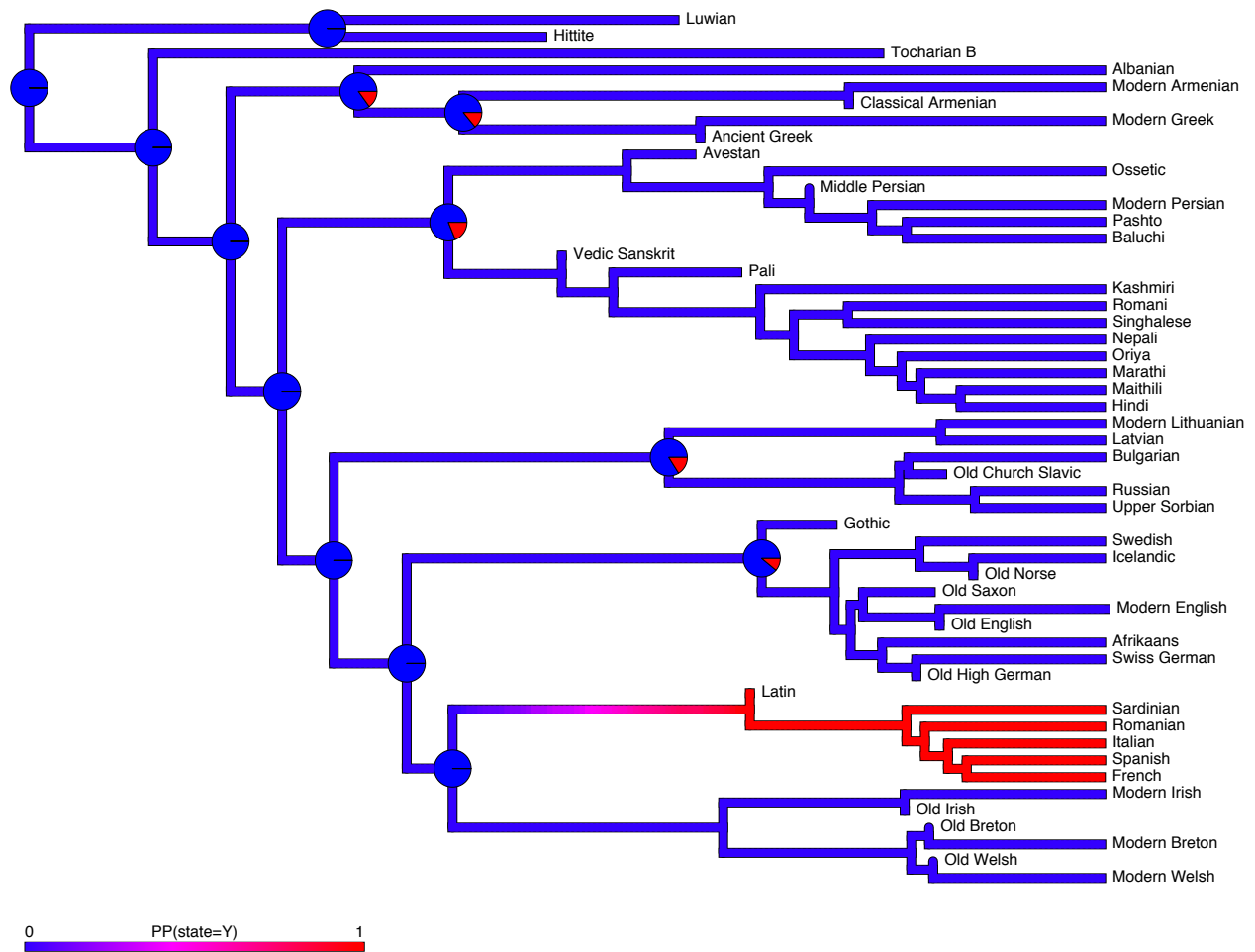


Figure S4.21: Posterior probability (PP) density of labiodental reflexes in some contexts of the correspondeme $*d^h$, assuming the phylogenies estimated by Chang et al.

S4.5.3.8 Reconstruction of labiodental reflexes for $*d^h$

With phylogenies estimated by Chang et al.:

```
correspondeme='dh'
dh.simmmaps.c <- compute.simmmap(correspondeme, tree=ie.c.sum.tree, model='ER')
dh.density.c <- densityMap(dh.simmmaps.c, plot=F, res=res)
dh.multi.simmmaps.c <- compute.simmmap(correspondeme, tree=ie.c.trees.sample, model='ER')
```

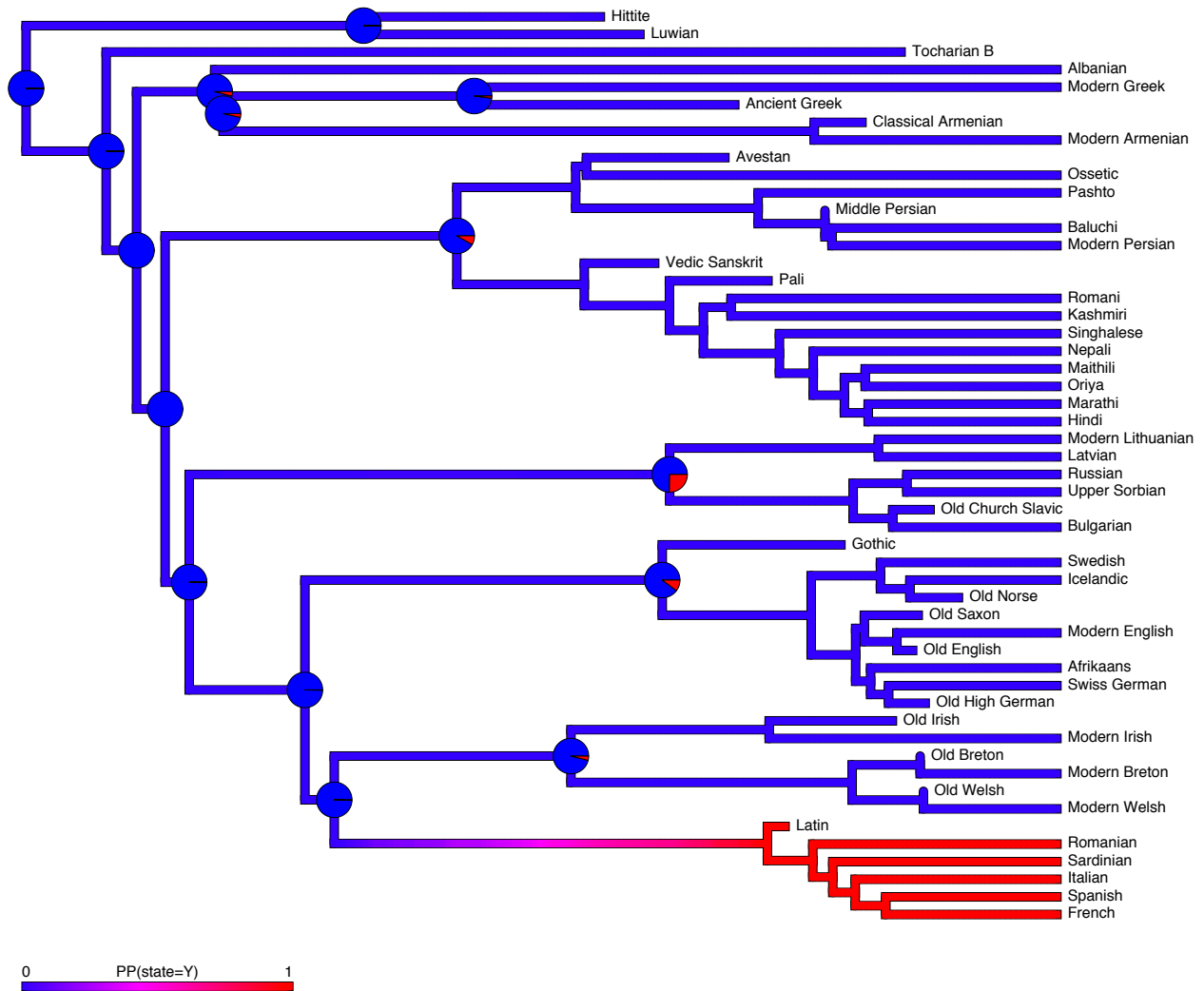


Figure S4.22: Posterior probability (PP) density of labiodental reflexes in some contexts of the correspondeme $*d^h$, assuming the phylogenies estimated by Bouckaert et al.

With phylogenies estimated by Bouckaert et al.:

```
dh.simmmaps.b <- compute.simmmap(correspondeme, tree=ie.b.sum.tree, model='ER')
dh.density.b <- densityMap(dh.simmmaps.b, plot=F, res=res)
dh.multi.simmmaps.b <- compute.simmmap(correspondeme, tree=ie.b.trees.sample, model='ER')
```

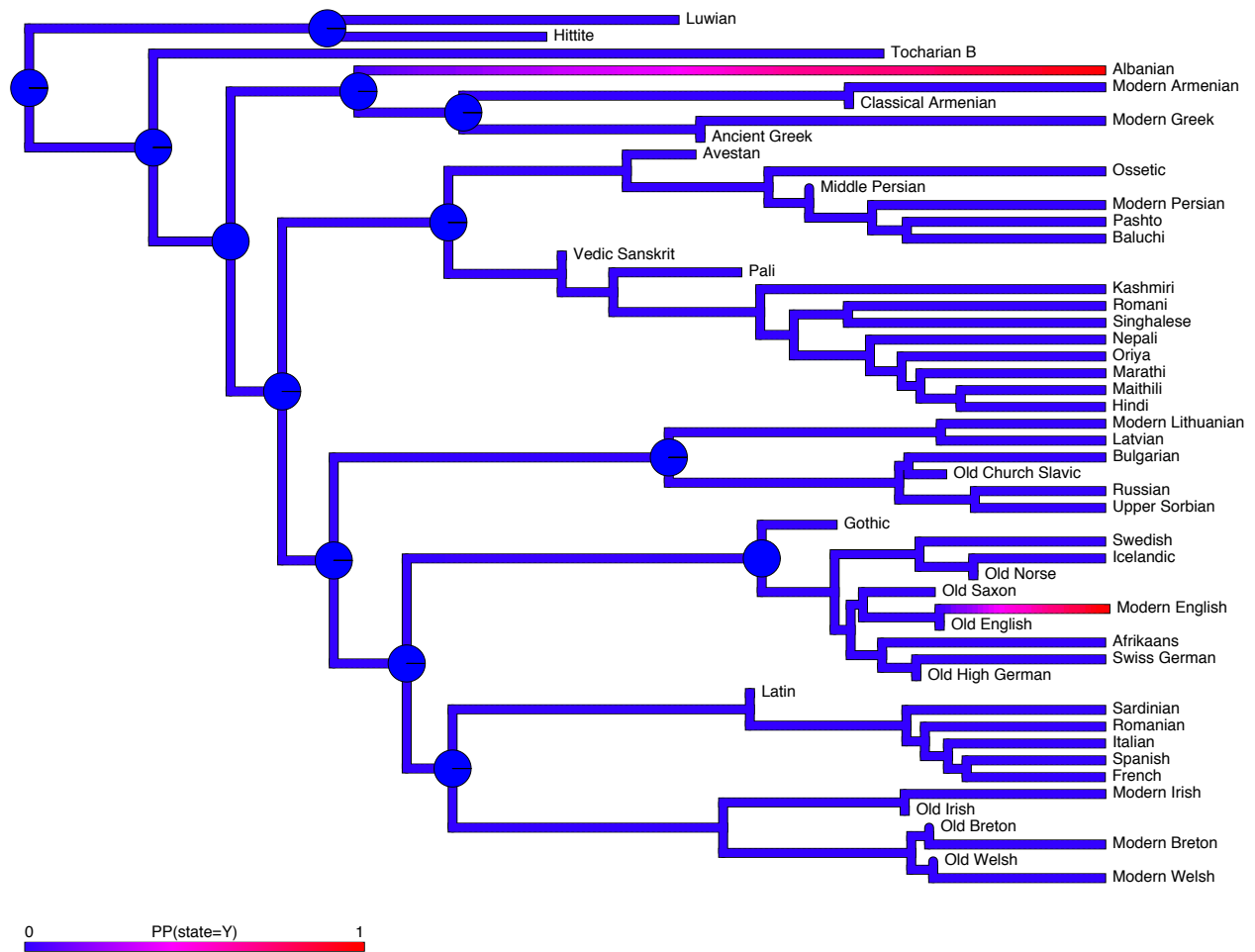


Figure S4.23: Posterior probability (PP) density of labiodental reflexes in some contexts of the correspondeme $*k^j$, assuming the phylogenies estimated by Chang et al.

S4.5.3.9 Reconstruction of labiodental reflexes for $*k^j$

With phylogenies estimated by Chang et al.:

```
correspondeme='ky'
ky.simmmaps.c <- compute.simmmap(correspondeme, tree=ie.c.sum.tree, model='ER')
ky.density.c <- densityMap(ky.simmmaps.c, plot=F, res=res)
ky.multi.simmmaps.c <- compute.simmmap(correspondeme, tree=ie.c.trees.sample, model='ER')
```

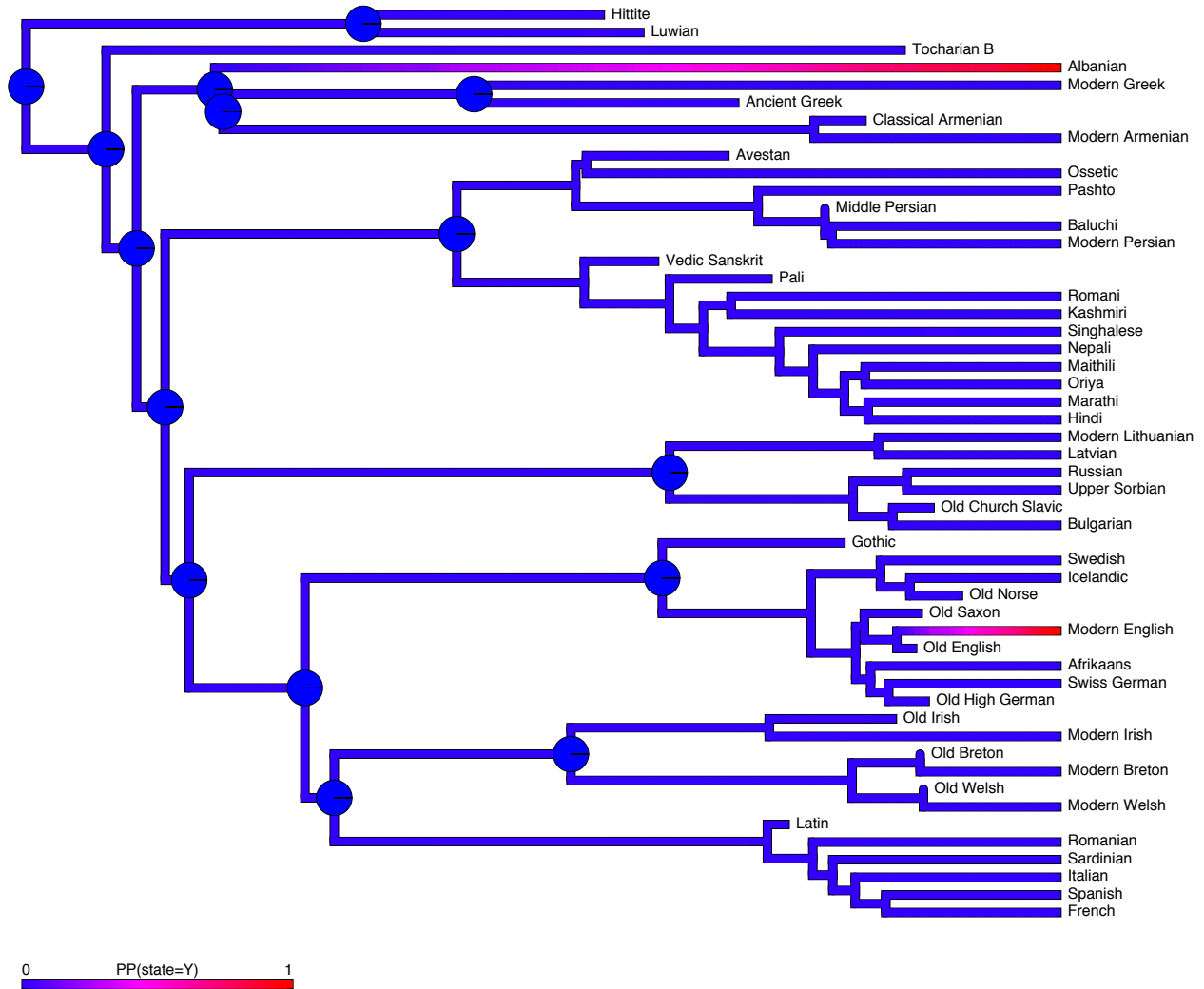


Figure S4.24: Posterior probability (PP) density of labiodental reflexes in some contexts of the correspondeme $*k^j$, assuming the phylogenies estimated by Bouckaert et al.

With phylogenies estimated by Bouckaert et al.:

```
ky.simmmaps.b <- compute.simmmap(correspondeme, tree=ie.b.sum.tree, model='ER')
ky.density.b <- densityMap(ky.simmmaps.b, plot=F, res=res)
ky.multi.simmmaps.b <- compute.simmmap(correspondeme, tree=ie.b.trees.sample, model='ER')
```

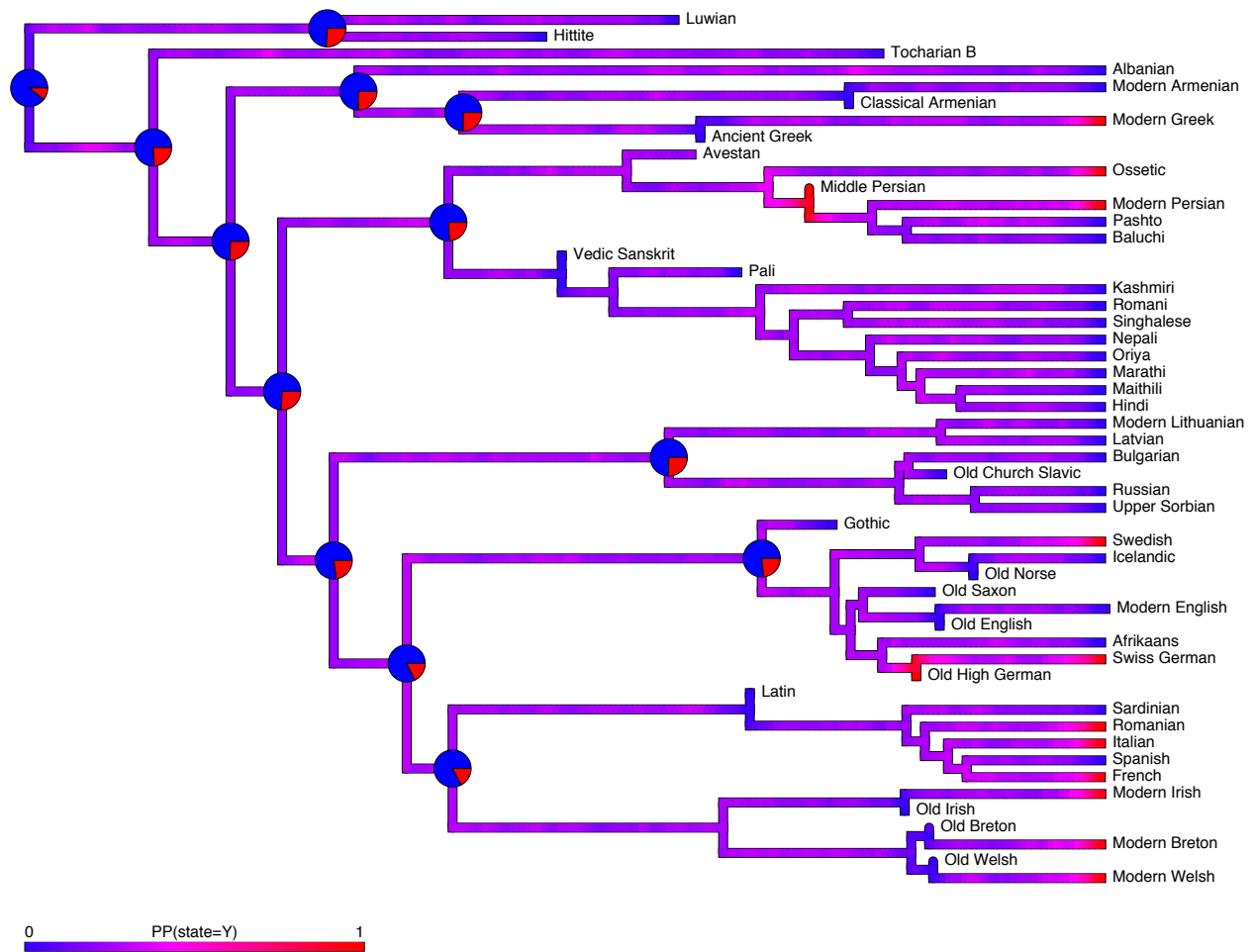



Figure S4.25: Posterior probability (PP) density of labiodental reflexes in some contexts of the correspondeme **b*, assuming the phylogenies estimated by Chang et al.

S4.5.3.10 Reconstruction of labiodental reflexes for **b*

With phylogenies estimated by Chang et al.:

```
correspondeme='b'
b.simmmaps.c <- compute.simmmap(correspondeme, tree=ie.c.sum.tree, model='ARD')
b.density.c <- densityMap(b.simmmaps.c, plot=F, res=res)
b.multi.simmmaps.c <- compute.simmmap(correspondeme, tree=ie.c.trees.sample, model='ARD')
```

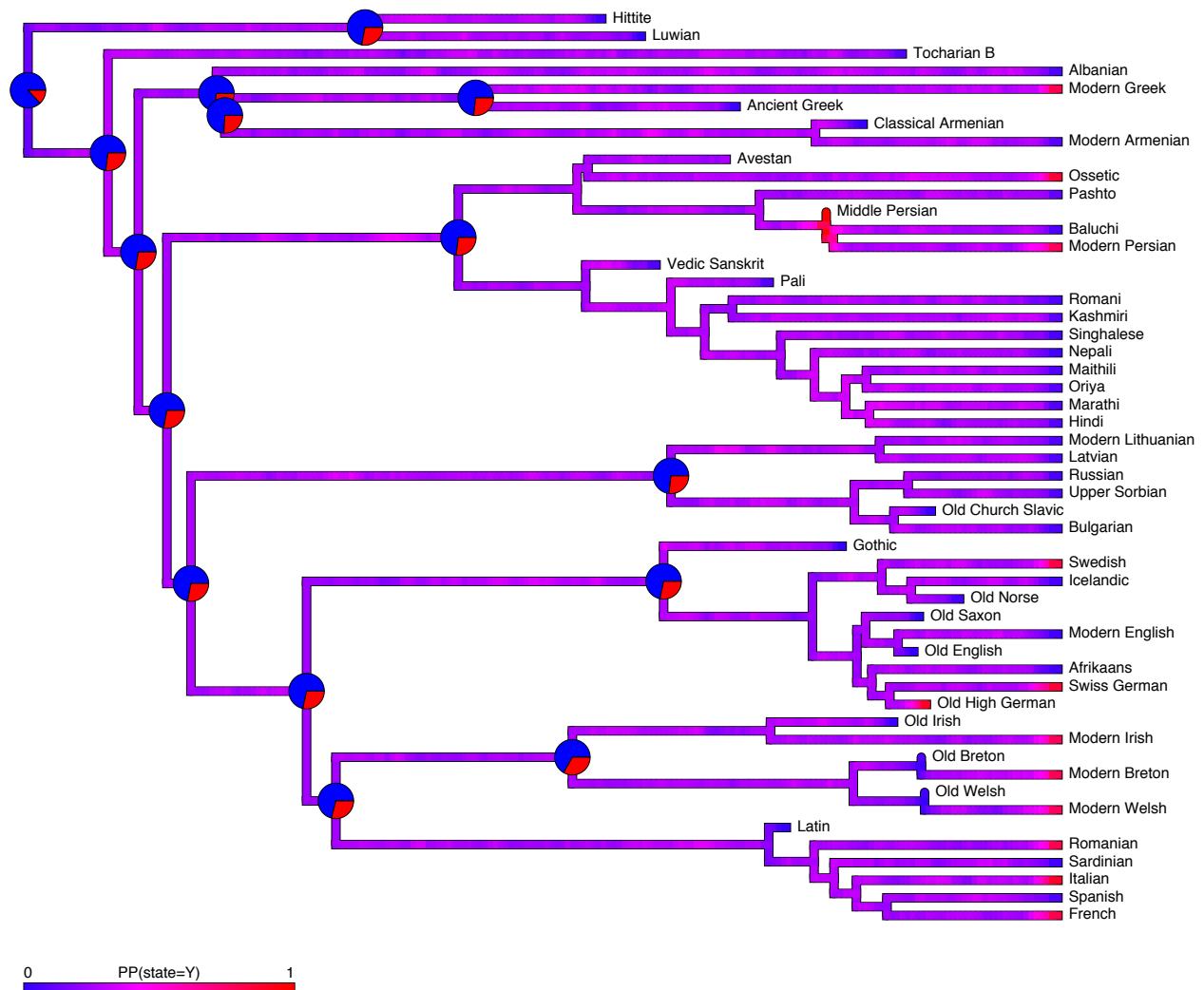


Figure S4.26: Posterior probability (PP) density of labiodental reflexes in some contexts of the correspondeme *b, assuming the phylogenies estimated by Bouckaert et al.

With phylogenies estimated by Bouckaert et al.:

```
b.simmmaps.b <- compute.simmmap(correspondeme, tree=ie.b.sum.tree, model='ARD')
b.density.b <- densityMap(b.simmmaps.b, plot=F, res=res)
b.multi.simmmaps.b <- compute.simmmap(correspondeme, tree=ie.b.trees.sample, model='ARD')
```

S4.5.4 Summary

The estimated root probabilities are as follows:

```
# doing this with lapply exceeds memory, so...; and let's do it externally...
# bh.c <- as.data.frame(describe.simap(bh.multi.simmaps.c)$ace)
# p.c <- as.data.frame(describe.simap(p.multi.simmaps.c)$ace)
# w.c <- as.data.frame(describe.simap(w.multi.simmaps.c)$ace)
# m.c <- as.data.frame(describe.simap(m.multi.simmaps.c)$ace)
# gw.c <- as.data.frame(describe.simap(gw.multi.simmaps.c)$ace)
# gwh.c <- as.data.frame(describe.simap(gwh.multi.simmaps.c)$ace)
# kw.c <- as.data.frame(describe.simap(kw.multi.simmaps.c)$ace)
# dh.c <- as.data.frame(describe.simap(dh.multi.simmaps.c)$ace)
# ky.c <- as.data.frame(describe.simap(ky.multi.simmaps.c)$ace)
# b.c <- as.data.frame(describe.simap(b.multi.simmaps.c)$ace)

load("all_aces_c.rda")

all.aces.c <- list(bh.c, p.c, w.c, m.c, gw.c, gwh.c, kw.c, dh.c, ky.c, b.c)
names(all.aces.c) <- correspondemes

(root.aces.c <- sapply(all.aces.c, '[', 1, 2))

##      bh      p      w      m      gw      gwh      kw      dh      ky      b
## 0.08591 0.23692 0.49831 0.02780 0.02537 0.03418 0.00374 0.00057 0.00100 0.10478

mean(root.aces.c)

## [1] 0.101858

median(root.aces.c)

## [1] 0.03099

# bh.b <- as.data.frame(describe.simap(bh.multi.simmaps.b)$ace)
# p.b <- as.data.frame(describe.simap(p.multi.simmaps.b)$ace)
# w.b <- as.data.frame(describe.simap(w.multi.simmaps.b)$ace)
# m.b <- as.data.frame(describe.simap(m.multi.simmaps.b)$ace)
# gw.b <- as.data.frame(describe.simap(gw.multi.simmaps.b)$ace)
# gwh.b <- as.data.frame(describe.simap(gwh.multi.simmaps.b)$ace)
# kw.b <- as.data.frame(describe.simap(kw.multi.simmaps.b)$ace)
# dh.b <- as.data.frame(describe.simap(dh.multi.simmaps.b)$ace)
# ky.b <- as.data.frame(describe.simap(ky.multi.simmaps.b)$ace)
# b.b <- as.data.frame(describe.simap(b.multi.simmaps.b)$ace)

load("all_aces_b.rda")

all.aces.b <- list(bh.b, p.b, w.b, m.b, gw.b, gwh.b, kw.b, dh.b, ky.b, b.b)
names(all.aces.b) <- correspondemes

(root.aces.b <- sapply(all.aces.b, '[', 1, 2))

##      bh      p      w      m      gw      gwh      kw      dh      ky      b
## 0.13783 0.17755 0.49897 0.03916 0.02567 0.01989 0.00596 0.00114 0.00768 0.13006

mean(root.aces.b)

## [1] 0.104391
```

```
## [1] 0.032415
```

Figures S4.27 and S4.28 give a synopsis of all stochastic character maps.

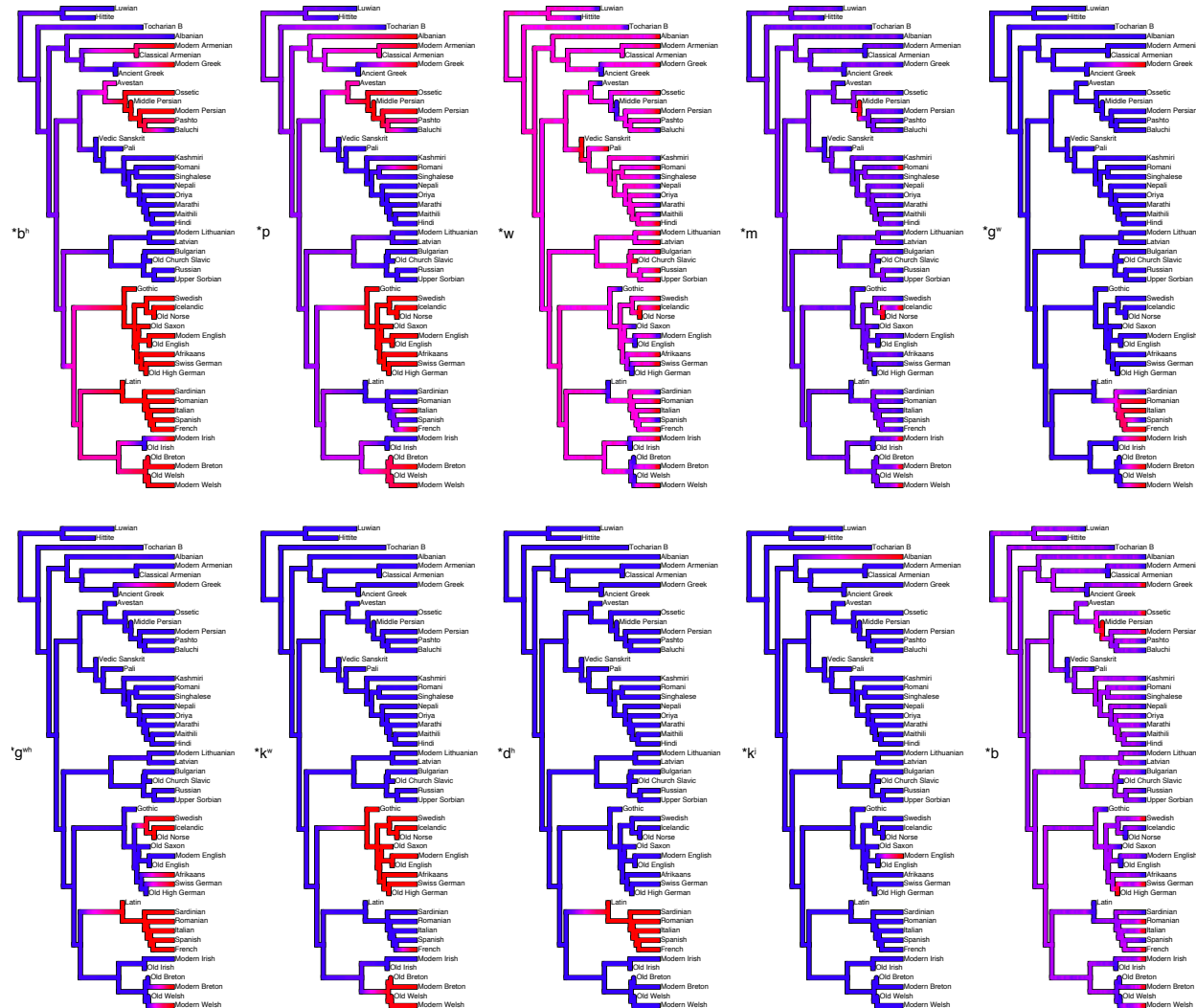


Figure S4.27: Stochastic character maps on the MCC trees estimated by Chang et al.

Figures S4.29 and S4.30 give a synopsis in terms of traitgrams. Each color represents the evolution of a correspondeme over all branches of the phylogeny. The position of the branch on the y-axis reflects the probability of a labiodental articulation.

```
all.maps.c <- list(
  bh=bh.density.c,p=p.density.c,w=w.density.c,m=m.density.c,
  gw=gw.density.c,gwh=gwh.density.c,kw=kw.density.c,
  dh=dh.density.c,ky=ky.density.c,b=b.density.c)

all.maps.values.c <- lapply(all.maps.c, function(density.map) {
  # get node and tip probabilities from densitymap:
  nodes.and.tip.values <- c(as.numeric(names(density.map$tree$maps[[1]])[1]), # the root
    sapply(density.map$tree$maps, function(m) {
```

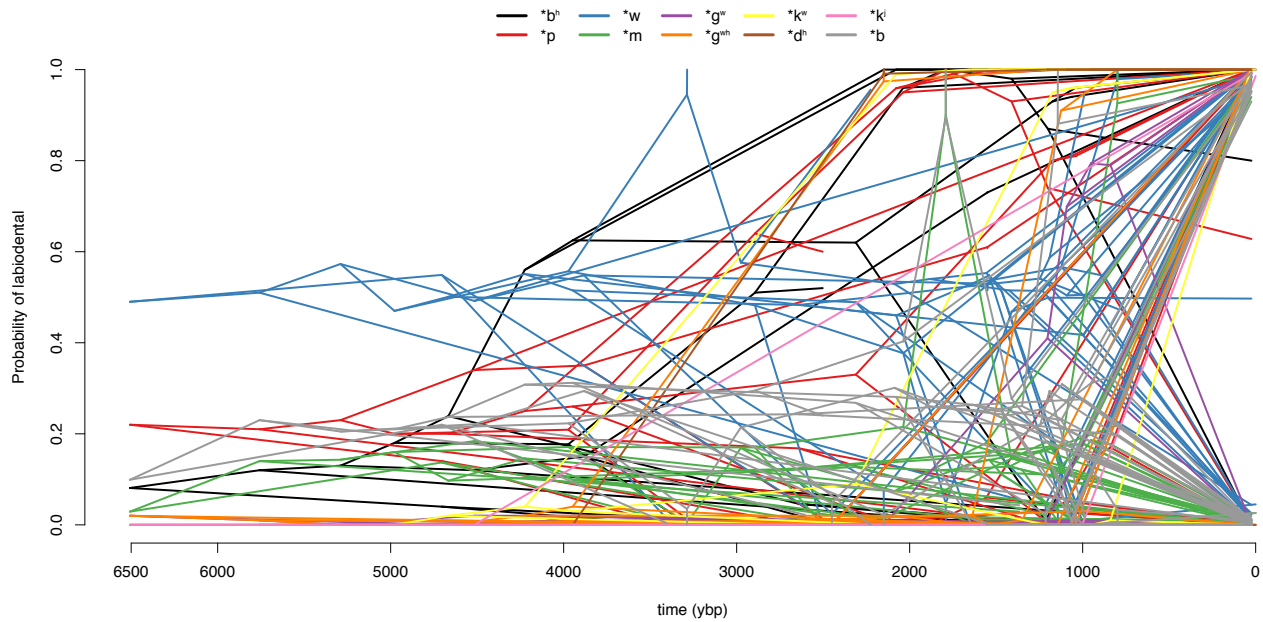



Figure S4.29: Evolution of labiodentals across correspondemes, based on the maximum clade credibility tree of the phylogeny in Chang et al.

```

        as.numeric(names(m)[length(m)])) } ) # the end states, i.e the internal nodes and tips
    )/1000
names(nodes.and.tip.values) <- as.character(c(53, # the root number
        density.map$tree$edge[,2]
    )) # the internal node and tip numbers
density.map$tree$tip.label <- as.character(1:length(density.map$tree$tip.label)) # name match
return(list(values=nodes.and.tip.values,tree=as.phylo(density.map$tree)))
})

tree.c <- all.maps.values.c[[1]]$tree

all.maps.b <- list(
    bh=bh.density.b,p=p.density.b,w=w.density.b,m=m.density.b,
    gw=gw.density.b,gwh=gwh.density.b,kw=kw.density.b,
    dh=dh.density.b,ky=ky.density.b,b=b.density.b)

all.maps.values.b <- lapply(all.maps.b, function(density.map) {
    # get node and tip probabilities from densitymap:
    nodes.and.tip.values <- c(as.numeric(names(density.map$tree$maps[[1]])[1]), # the root
        sapply(density.map$tree$maps, function(m) {
            as.numeric(names(m)[length(m)])) } ) # the end states, i.e the internal nodes and tips
    )/1000
names(nodes.and.tip.values) <- as.character(c(53, # the root number
        density.map$tree$edge[,2]
    )) # the internal node and tip numbers
density.map$tree$tip.label <- as.character(1:length(density.map$tree$tip.label)) # name match
return(list(values=nodes.and.tip.values,tree=as.phylo(density.map$tree)))
})

```

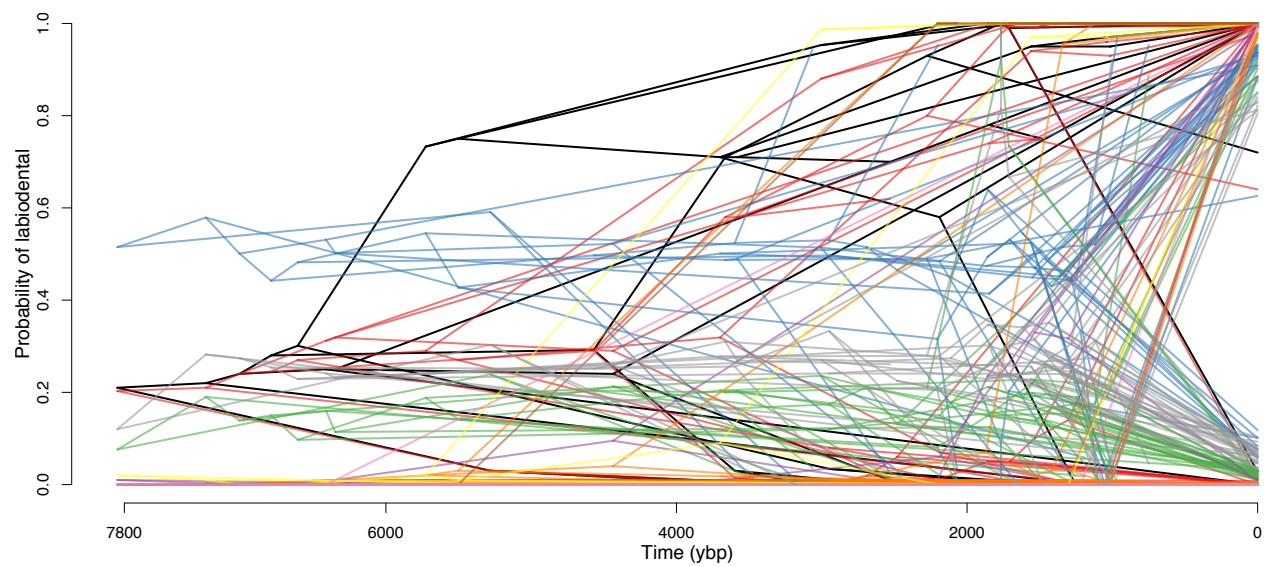


Figure S4.30: Evolution of labiodentals across correspondemes, based on the maximum clade credibility tree of the phylogeny in Bouckaert et al.

```
tree.b <- all.maps.values.b[[1]]$tree
```

References

1. C. F. Hockett, Distinguished lecture: F. *American Anthropologist*. **87**, 263–281 (1985).
2. W. Chang, C. Cathcart, D. Hall, A. Garrett, Ancestry-constrained phylogenetic analysis supports Indo-European steppe hypothesis. *Language*. **91**, 194–244 (2015).
3. M. Pagel, A. Meade, **BayesTraits V2.0** (2014).
4. R. Nielsen, Mapping mutations on phylogenies. *Systematic Biology*. **51**, 729–732 (2002).
5. J. P. Huelsenbeck, R. Nielsen, J. P. Bollback, Stochastic mapping of morphological characters. *Systematic Biology*. **52**, 131–158 (2003).
6. L. J. Revell, **phytools**: An R package for phylogenetic comparative biology (and other things). *Methods in Ecology and Evolution*. **3**, 217–223 (2012).
7. M. Widmer, S. Auderset, P. Widmer, J. Nichols, B. Bickel, NP recursion over time: Evidence from Indo-European. *Language*. **93**, 1–36 (2017).
8. G. Yu, D. K. Smith, H. Zhu, Y. Guan, T. T.-Y. Lam, **ggtree**: An R package for visualization and annotation of phylogenetic trees with their covariates and other associated data. *Methods in Ecology and Evolution*. **8**, 28–36 (2017).
9. R. Bouckaert *et al.*, Mapping the origins and expansion of the Indo-European language family. *Science*. **337**, 957–960 (2012).
10. M. Pagel, The maximum-likelihood approach to reconstructing ancestral character states of discrete characters on phylogenies. *Systematic Biology*. **48**, 612–622 (1999).
11. M. Pagel, A. Meade, D. Barker, Bayesian estimation of ancestral character states on phylogenies. *Systematic Biology*. **53**, 673–684 (2004).
12. F. Ronquist, Bayesian inference of character evolution. *Trends in Ecology and Evolution*. **19**, 475–481 (2004).
13. M. C. Shapiro, in *The Indo-Aryan languages*, G. Cardona, D. Jain, Eds. (Routledge, London, 2003), pp. 250–285.