randtip, a generalized framework to expand incomplete phylogenies using non-molecular phylogenetic information

Ignacio Ramos-Gutiérrez[1], Herlander Lima[2] & Rafael Molina-Venegas[2*]

1. *Department of Biology (Botany), Univ. Autónoma de Madrid, Madrid, Spain.*
2. *Universidad de Alcalá. GLOCEE - Global Change Ecology and Evolution Group, Department of Life Sciences, Alcalá de Henares, 28805, Spain.*

*Correspondence: rafmolven@gmail.com

Abstract

1. The increasing availability of molecular information has lifted our understanding of species evolutionary relationships to unprecedent levels. However, current estimates of the world's biodiversity suggest that about a fifth of all extant species are yet to be described, and we still lack molecular information for many of the known species. Hence, evolutionary biologists will have to tackle phylogenetic uncertainty for a long time to come.
2. This prospect has urged the development of software to expand phylogenies based on non-molecular phylogenetic information, and while the available tools provide some valuable features, major drawbacks persist and some of the proposed solutions are hardly generalizable to any group of organisms.
3. Here, we present a completely generalized and flexible framework to expand incomplete molecular phylogenies. The framework is implemented in the R package 'randtip', a toolkit of functions that was designed to randomly bind phylogenetically uncertain taxa in backbone phylogenies through a fully

customizable and automatic procedure that uses taxonomic ranks as a major source of phylogenetic information.

4. Although randtip is capable of automatically generating fully operative phylogenies for any group of organisms using just a list of species and a backbone tree, we stress that the 'blind' expansion of phylogenies (using randtip or any other available software) often leads to suboptimal solutions. Thus, we discuss a variety of circumstances that may require customizing simulation parameters beyond default settings to optimally expand the trees, including a detailed step-by-step workflow.
5. Phylogenetic uncertainty should be tackled with caution, assessing potential pitfalls and opportunities to optimize parameter space prior to launch any simulation. Used judiciously, our framework will help evolutionary biologists to efficiently expand incomplete molecular phylogenies and thereby account for phylogenetic uncertainty in quantitative analyses.

Kewwords: backbone phylogeny, most derived consensus clade, most recent common ancestor, phylogenetic uncertainty, taxonomic rank.

1. Introduction

The past two decades have seen an explosive interest in incorporating evolutionary history into ecological analyses (Webb et al., 2002; Cavender-Bares et al., 2009; Mouquet et al., 2012), boosting several disciplines such as community ecology (Davies, 2021), macroecology (Lamsdell & Congreve, 2021) and conservation biology (Molina-Venegas et al., 2020). This eco-phylogenetic revolution was driven by the increased availability of molecular information (Sayers et al., 2020) and sophisticated tools for inferring phylogenetic trees (Smith & Walker, 2018), which have lifted our understanding of species evolutionary relationships to unprecedent levels. However, and despite the phylogeny of certain groups is nearly completed (e.g. mammals; Upham et al., 2019), phylogenetic relationships remain vastly uncertain –particularly shallow ones (i.e. infra-family)– for many groups. For example, one of the largest global phylogenies of angiosperm plants published to date includes only ~12.5% of the species in the group (Janssens et al., 2020), and recent accounts of terrestrial arthropod biodiversity showed that up to 80% of insect species are yet to be discovered (Stork, 2018). These bleak figures suggest that evolutionary biologists will have to tackle phylogenetic uncertainty for a long time to come.

Conscious of the limited extent of molecular phylogenetic information, Rangel et al. (2015) developed a theoretical foundation to systematically account for phylogenetic uncertainty in quantitative analyses. Roughly, the procedure starts with the identification of *phylogenetically uncertain taxa* (PUTs), this is, taxonomic units (e.g. species, subspecies) that are well delineated in the continuum of biodiversity but remain missing from available molecular phylogenies. Then, all acceptable taxonomic, morphological, or behavioral information on the PUTs is used to conservatively define their *most derived consensus clades* (MDCCs), this is, the less inclusive phylogenetic nodes that most

70 certainly contain them. Finally, each PUT is assigned to a random point along one branch
71 of its corresponding MDCC, and the procedure is replicated a high number of times to
72 obtain a distribution of possible trees that can be used in downstream phylogenetic
73 analyses iteratively. While the 'true' phylogenetic hypothesis will most certainly remain
74 unsampled, the workflow allows exploring the parameter space, thereby quantifying the
75 extent to which phylogenetic uncertainty has a significant impact in the analyses (e.g.
76 Calatayud et al., 2019; Molina-Venegas et al., 2021). Rangel et al. (2015) accompanied
77 their framework with the software SUNPLIN, a set of algorithms for randomly expanding
78 molecular phylogenies using the aforementioned procedure (Martins et al., 2013).

79 Although Rangel et al. (2015) suggested that the identification of MDCCs should
80 be based on expert taxonomic evaluation, such knowledge is in practice beyond the reach
81 of most researchers, particularly when dealing with very large phylogenies that often
82 encompass a wide spectrum of taxonomic groups and thousands of species. In a valuable
83 attempt to automatize the identification of MDCCs, Jin & Qian (2019) developed
84 V.PhyloMaker, a R package that generates very large phylogenies of vascular plants.
85 V.PhyloMaker builds on the idea of the classical software Phylomatic (Webb &
86 Donoghue, 2005), which uses a taxonomically informed backbone mega-tree to
87 automatically define MDCCs (genus or family nodes in case the former are not available)
88 and bind the PUTs to the selected clades. Beyond covering features that were already
89 implemented in Phylomatic, V.PhyloMaker provides an option to insert PUTs in
90 randomly chosen nodes below the crown node of the corresponding MDCCs, so that a
91 distribution of possible phylogenies can be generated with relatively little effort (Jin &
92 Qian, 2019).

93 However, we note that current available tools for the insertion of PUTs, while
94 valuable, have some important drawbacks. For example, V.PhyloMaker uses a pure node-

based approach to insert PUTs, and thus the simulations often lead to the formation of polytomies even if a fully bifurcated backbone tree is used. In contrast, SUNPLIN allows the insertion of PUTs along randomly selected branches, but the user must manually set all the MDCCs for the simulations (Martins et al., 2013). V.PhyloMaker circumvents this limitation at the cost of requiring an 'annotated' backbone mega-tree (a linkage between all the species represented in the backbone tree and their taxonomic genus and family) that is provided by the developers of the software, and thus the user is forced to use the backbone tree for which the software was implemented –currently the GBOTB.extended tree for vascular plants (see Jin & Qian, 2019)–. Also, the definition of MDCCs on the basis of fixed taxonomic ranks (e.g. genera and otherwise families) might be excessively conservative and hence suboptimal under certain circumstances. For example, large taxonomic families often include taxonomic ranks between the family and genus level that may represent putative MDCCs (e.g. subfamilies, tribes and subtribes in the Asteraceae, Poaceae and Fabaceae plant families). Finally, there are shortcomings that are transversal to all available software for PUT binding, including the disregard of natural paraphyletic groups (Hörandl & Stuessy, 2010) and the impossibility to fully customize the space of branch lengths for the insertion of PUTs among other issues.

Here, we present a completely generalized and flexible framework to expand incomplete phylogenies. The framework is implemented in the R package 'randtip', a toolkit of functions that was designed to randomly bind PUTs in backbone phylogenies through a fully customizable and automatic procedure that uses taxonomic ranks as a major source of phylogenetic information. Although randtip is capable of automatically generating fully operative phylogenies for any group of organisms using just a list of species and a backbone tree, we discuss a variety of circumstances that may require

customizing simulation parameters beyond default settings to optimally expand the trees, including a detailed step-by-step workflow.

2. General workflow

In this section, we describe the general workflow of randtip to expand phylogenies. Roughly, given a list of taxa (typically Linnean binomials) for which a phylogeny is to be obtained and a backbone tree (provided by the user), the software identifies putative MDCCs for the PUTs in the list. MDCCs are defined based on taxonomic ranks, including genus, subtribe, tribe, subfamily, family, superfamily, order and class, and by default the software will select the less inclusive among the available. Once each PUT is assigned to a MDCC, randtip will automatically bind the former to the backbone tree according to the parameters that are set for the simulations, and a phylogeny including all the species in the user's list is returned (Fig. 1). The workflow can be customized using a variety of parameters that are either passed through the whole simulation or adjusted independently for each PUT (see Appendix for detailed step-by-step examples).

2.1. *Input files*

The workflow of randtip is guided by a data frame R object (hereafter '*info*') and the instructions that are passed through the main function of the package (*rand.tip*). The data frame *info* is a template with 21 columns –20 variables of type character or logical plus one integer variable for internal use– that must contain, as a minimum, all the taxa in the user's list (column 1) and their genus rank (column 2). Optionally, the user may provide supra-generic taxonomic ranks and set parameter values specifically for individual PUTs. For simplicity, we will consider the most common scenario in the ecological literature

144 where the operative taxa represent Linnean binomials (genus and species with or without
145 subspecific epithets), although genus-level phylogenies and taxon lists are also supported.
146 The *info* template can be created automatically using the auxiliar function *build.info*,
147 which is fed with species names in a character vector or single column data frame.
148 Besides, *build.info* can interact with a suite of taxonomic repositories –currently
149 implemented for 'ncbi' (default), 'itis', 'gbif' and 'bold' via the *classification* function of
150 'taxize' R package (Chamberlain et al., 2020)– to retrieve taxonomic information that
151 will be used to identify putative supra-generic MDCCs for the PUTs (note that
152 information to define genus-level MDCCs is intrinsically contained in the scientific
153 names of the species). This can be done by setting the argument 'find.ranks' of *build.info*
154 to TRUE (default). We recommend providing at least one supra-generic rank (e.g.
155 taxonomic family) for all the species in *info*, which will be used to define MDCCs
156 whenever the genera of the PUTs are missing in the phylogeny (otherwise the PUTs will
157 not be bound). Often the user will need to further edit *info* once the template is
158 automatically created with *build.info* (for example, to customize binding parameters for
159 certain PUTs or to amend taxonomic mistakes in web repositories). This can be done
160 directly in R using the auxiliar function *edit.info* or exporting the data frame as a
161 spreadsheet (e.g. csv or xlsx) and importing it back into R once all the edits are completed.
162 The user must provide a backbone phylogeny as a *phylo* R object. Although
163 randtip can identify MDDCs on the sole basis of taxonomic information of the species
164 that are included in both the user's list and the backbone tree (hereafter 'taxon list' mode),
165 MDDCs can also be identified based on taxonomic ranks of all the species that are
166 represented in the latter regardless of their presence in the former (hereafter 'backbone'
167 mode). Both approaches have pros and cons, and they will behave identically whenever
168 the genera of the PUTs are minimally represented in the backbone tree. To use the

169 ‘backbone’ mode of randtip, the argument ‘mode’ of *build.info* must be set to “backbone”
170 (default) to additionally include all the species in the phylogeny as rows in the *info* data
171 frame, so that their taxonomic information can also be retrieved (if the argument
172 ‘find.ranks’ of *build.info* is set to TRUE).

173 Once the data frame *info* is assembled, we strongly recommend the user to check
174 the incidence of PUTs in the user’s list and their putative MDCCs. This can be done with
175 the auxiliar function *check.info*, which will inform on the PUT status of the species, the
176 presence of spelling errors, putative MDCCs, and the phyletic nature of the set of species
177 that are included in each MDCC and share taxonomic ranks (e.g. congenerics, contribals,
178 confamiliars) with the corresponding PUT –hereafter *phylogenetically placed and co-*
179 *ranked* (PPCR) species–. Also, the tip labels of the backbone tree are checked out for
180 duplicates (e.g. *Ziziphora taurica taurica* and *Ziziphora taurica*) and the ultrametric
181 nature of the tree is assessed. The auxiliar functions *get.clade* and *plot.clade* can in turn
182 be used to extract and plot respectively any subtree representing putative MDCCs, so that
183 the user can visually explore them using the R graphic window. Exploring MDCCs is
184 particularly recommended to optimize PUT binding, particularly when PPCR species
185 form polyphyletic groups. Alternatively, subtrees can be exported in Newick format to
186 visualize them using auxiliar software such as Dendroscope (Huson & Scornavacca,
187 2012), which may be very convenient for very large clades. Once the MDCCs are defined
188 and the user has optionally customized parameter values for individual PUTs, the
189 wrapping function *info2input* is fed with the data frame *info* and the backbone tree to
190 create a final dataset that will be passed through the *rand.tip* function to expand the
191 phylogeny. This final dataset ensures consistent structure for use in *rand.tip* and allows
192 generating as many trees as desired without the need to search for putative MDCCs in

193 *info* repeatedly (this is done by *info2input* just once), which is a computationally intense
194 task.
195
196 2.2. *Selecting MDCCs and binding PUTs*
197 The binding of PUTs is conducted with the function *rand.tip*, which includes a
198 variety of parameters that are passed through the whole simulation (Table S1). However,
199 all the parameter arguments of *rand.tip* can be adjusted independently for each PUT by
200 editing in the corresponding slots of *info*, which makes the framework completely flexible
201 and customizable.
202 Randtip will always try to find the less inclusive MDCC for each PUT according
203 to the taxonomic ranks that are provided in *info*, starting from genus level and up to class
204 level until a MDCC for the PUT is found. Regardless of the mode of randtip that is set by
205 the user ('taxon list' or 'backbone'), the software will always first attempt to define
206 genus-level MDCCs as the *most recent common ancestor* (MRCA) of all the species in
207 the backbone tree that are congeneric to the PUTs. However, the definition of MDCCs
208 above the genus level may differ between the two modes of randtip. On 'taxon list' mode,
209 supra-generic MDCCs are defined as the MRCA of all the species in the user's list that
210 are PPCR with the target PUT (e.g. contribals, consubfamiliars, confamiliars). In contrast,
211 the 'backbone' mode (default) defines supra-generic MDCCs as the MRCA of all the
212 species in the backbone phylogeny (regardless of their presence in the user's list) that are
213 PPCR with the target PUT (see Fig. 3 and section 3 for an extended discussion).
214 By default, *rand.tip* will bind each PUT to a randomly selected branch below the
215 crown node of the corresponding MDCC, the probability of being added along any branch
216 being directly proportional to the length of the branch –argument 'prob' set to TRUE
217 (default)–. Alternatively, branches can be selected on the basis of equal probability, and

218 in either case the user can decide to add the stem branch of the clade to the pool of
219 candidate branches –if the argument 'use.stem' is set to TRUE (default is FALSE)–. The
220 exact point to insert the PUT in the selected branch is sampled from a uniform
221 distribution. Importantly, the extent to which the default behavior of *rand.tip* to insert
222 PUTs represents an optimal scenario may depend on the phyletic nature of their PPCR
223 species. These can represent monophyletic (whenever the MDCC is exclusively shaped
224 by species that are PPCR with the target PUT), singleton (terminal branch), paraphyletic
225 (whenever the species that map within the MDCC but are not PPCR with the PUT form
226 either a monophyletic or a singleton group), or polyphyletic (set of PPCR species that do
227 not fit any of the previous categories) groups (see section 4.1 and Fig. 2). For example,
228 the PPCR species of a given PUT could form a polyphyletic clade simply because one of
229 them maps clearly away from the main (monophyletic) cluster of PPCR species –e.g.
230 because the outlying PPCR species is labelled in error (Pentinsaari et al., 2020)–, in which
231 case the default behavior of *rand.tip* to bind the PUT (i.e. any branch below the crown
232 node of the largest monophyletic cluster) would be reasonable. In contrast, the
233 polyphyletic nature of PPCR species could be due to 'intruder' taxa that map within an
234 otherwise monophyletic cluster, in which case the default behavior of *rand.tip* could be
235 suboptimal because the evidence that the largest monophyletic cluster of the group
236 include the PUT may be less clear (Fig. 2). As we discuss in section 4, randtip allows the
237 user to optimize the binding of PUTs according to the specifics of each case.

238 It is important to note that the user can always decide to what extent they want to
239 rely on the retrieved taxonomic ranks for the automatic identification of MDCCs. For
240 example, if the taxonomic affiliation of a PUT to a given genus is controversial, the user
241 may edit the data frame *info* to change the genus-rank of the PUT into 'NA', in which
242 case randtip will use the taxonomic rank immediately above to find a new MDCC.

243 **3. Comparison between 'taxon list' and 'backbone' modes of randtip**

244 The first decision the user will have to tackle is choosing between the 'taxon list' and
245 'backbone' modes of randtip. As we stated earlier, both approaches will perform
246 identically as long as the genera of the PUTs are minimally represented in the backbone
247 phylogeny, yet the definition of supra-generic MDCCs may differ between the two
248 approaches. For example, it might happen that some of the PPCR species of a given PUT
249 (let's say confamiliars) are missing in the user's list but are represented in the backbone
250 phylogeny. Thus, in case these PPCR species were phylogenetically external to the
251 confamiliars that are included in the user's list, the 'backbone' mode of randtip would
252 define an older MDCC than 'taxon list' (Fig. 3). It follows that the extent of the
253 divergence in the functioning between both modes (whenever a supra-generic MDCC is
254 to be defined) depends on the phylogenetic placement of the PPCR species that are
255 included in the user's list. In sum, the 'backbone' mode works based on the 'true' supra-
256 generic MDCCs (but note that these may neither represent the actual MDCCs as the
257 backbone phylogenies are often not fully comprehensive) with the trade-off that it is a
258 more time-consuming approach than 'taxon list'. In contrast, the latter might define
259 younger supra-generic MDCCs (meaning more restricted parameter space to bind PUTs)
260 under some circumstances (Fig. 3). We recommend considering the 'backbone' mode as
261 a first option (default) and use 'taxon list' only when there is a low incidence of PUTs
262 requiring supra-generic binding and/or low mismatch in the nodes defining supra-generic
263 MDCCs between both approaches (see Fig. 3 and Appendix for an extended discussion).
264

265 **4. Newly designed features for PUT binding**

266 As discussed above, *rand.tip* will by default bind PUTs to randomly selected
267 branches below the crown node of the corresponding MDCCs. However, this default

behavior can be modified using a variety of arguments that are implemented in *rand.tip*. For example, if the user is not interested in generating a distribution of possible phylogenies but one single tree without randomizing the PUTs, the argument 'rand.type' of *rand.tip* can be set to "polytomy" (default is "random") for the function to insert the PUTs as polytomies at the crown nodes of their corresponding MDCCs instead. This is the only binding option that was implemented in the seminal software Phylomatic (Webb & Donoghue, 2005), and it might still be convenient for extremely resource-consuming phylogenetic analyses where using a distribution of possible trees could be computationally prohibitive. Alternatively, the user may want to bind the PUTs following the default behavior of *rand.tip* but still inserting *some* of them as polytomies in their corresponding MDCCs. To do so, the user can set the corresponding slots of the column 'rand.type' of *info* to "polytomy" (default is NA) while keeping the argument 'rand.type' of *rand.tip* to "random".

4.1 *Polyphyletic, paraphyletic and singleton groups of PPCR species*

While PUT randomizations within monophyletic groups of PPCR species will always follow the same scheme (i.e. by default, randomly selected branches below the crown node of the corresponding MDCCs), the user must choose between different scenarios for polyphyletic, paraphyletic and singleton groups. In case MDCCs are shaped by polyphyletic groups of PPCR species, the user must choose between three different scenarios using the 'polyphyly.scheme' argument of *rand.tip*. If the default option "largest" is set, *rand.tip* will pick the largest monophyletic cluster of PPCR species among the available to insert the PUT (less conservative scenario; Fig. 2). If the option "frequentist" is set, *rand.tip* will first pick one of the constituent clusters of PPCR species that conform the polyphyletic group, the probability of being selected being proportional

293 to the size of the cluster, and then the PUT will be inserted in the selected cluster. If the
294 option "complete" is set, *rand.tip* will bind the PUT to a randomly selected branch below
295 the crown node of the MDCC (most conservative scenario).

296 In case MDCCs are defined by paraphyletic groups of PPCR species, two different
297 scenarios are eligible. If the argument 'use.paraphyletic' is set to TRUE (default), *rand.tip*
298 will only bind PUTs to branches so that the paraphyletic nature of the group remains
299 unchanged (Fig. 2). Otherwise, the randomization will be conducted as if the MDCCs
300 were defined by a monophyletic group of species. Importantly, certain taxonomic groups
301 such as the Olacaceae s.l. plant family are paraphyletic (Chase et al., 2016), and thus
302 randomizing PUTs at any point below the crown node of this family (i.e. setting
303 'use.paraphyletic' to FALSE) may result in an excessively conservative parameter space
304 that would encompass almost the entire Santalales order (Malécot & Nickent, 2008).

305 In case the MDCC of a PUT is defined by one single PPCR species (Fig. 2),
306 *rand.tip* will by default bind the PUT to the terminal branch subtending the only PPCR
307 species, and whenever the MDCC is no longer singleton (because at least one PUT was
308 already bound), *rand.tip* will consider the entire newly formed clade (same height as the
309 original singleton clade) to sample for candidate branches. We will refer to this procedure
310 as 'bind-to-singleton' hereafter. However, if the argument 'use.singleton' is set to FALSE
311 (default is TRUE), the parent node of the singleton PPCR species will be defined as the
312 MDCC of the PUT instead (Fig. 2). Although the latter scheme is more conservative than
313 the former, it may lead to suboptimal solutions under some circumstances. For example,
314 the parameter space to randomize a PUT whose MDCC is shaped by one single species
315 that is the only representative of a subfamily in the phylogeny can be drastically increased
316 in case the subfamily is the sister group to the rest of the family. Note that all these

317 parameters can be specifically set for individual PUTs by filling in the corresponding
318 slots of *info*.
319
320 4.2. *Manual definition of MDCCs*
321 Although randtip was conceived to automatize the definition of MDCCs based on
322 taxonomic ranks, the user can manually define MDCCs for the PUTs. This can be done
323 by filling in the corresponding slots of the columns 'taxa1' and 'taxa2' of *info*. As long
324 as these slots are not set to 'NA' (default), the MDCCs of the PUTs will be defined on
325 the basis of this information instead. For example, if the slots 'taxa1' and 'taxa2' of a
326 PUT are filled in with different species names, the PUT will be bound to a randomly
327 selected branch below the MRCA of the two given species. If both slots are filled in with
328 the same species name, randtip will follow the bind-to-singleton procedure to insert the
329 PUT as sister to the so defined species, and in case the same genus is provided the PUT
330 will be inserted as sister to the clade defined by the MRCA of all the species in that genus.
331
332 4.3. *Respecting monophyletic and paraphyletic clades*
333 By default, *rand.tip* will never bind a PUT to a branch that results in breaking the
334 monophyletic or paraphyletic nature of a group (of any taxonomic rank) unless the
335 arguments 'respect.mono' and 'respect.para' of the function are set to FALSE (default is
336 TRUE). Thus, while previous software followed either approach (e.g. V.Phylomaker
337 always respects monophyletic genera but SUNPLIN does not), randtip offers the user the
338 possibility to choose between both options, either by setting the general arguments of the
339 *rand.tip* function or on a customized basis for individual PUTs by filling in the
340 corresponding slots of *info*.
341

342 4.4. *Clumping PUTs*

343 Some genera may not be represented in the phylogeny, and thus their representative
344 species will likely form a polyphyletic group if they are to be bound randomly below the
345 crown node of the corresponding supra-generic MDCC. However, the user could be
346 certain in that a group of congeneric PUTs whose genus is missing in the phylogeny is
347 monophyletic. Thus, if the argument 'clump.puts' is set to TRUE (default), *rand.tip* will
348 first bind one of the congeneric PUTs, and then the rest will be bound to the former
349 following the bind-to-singleton procedure. Similarly, it may happen that supra-generic
350 taxonomic groups are not represented in the phylogeny, in which case *rand.tip* will clump
351 the PUTs as described above and following the taxonomic hierarchy so that the missing
352 taxonomic groups will form monophyletic clusters once all the PUTs are bound. As any
353 other randomization parameter of randtip, the user may decide the PUTs that will be
354 clumped in this way by setting the 'clump.puts' option individually in the corresponding
355 slots of *info*.

356 Trinomials representing infra-specific taxa (e.g. subspecies) are also supported. If
357 'clump.puts' is set to TRUE, *rand.tip* will clump PUTs with infra-specific information
358 according to their specific epithets (i.e. second name in the trinomial). To do so, *rand.tip*
359 will first check if any of the trinomial PUTs that share specific epithet are represented in
360 the phylogeny. This search also takes into account the type subspecies of the species,
361 which will be detected in either trinomial (e.g. *Ablepharus chernovi chernovi*) or binomial
362 (e.g. *Ablepharus chernovi*) nomenclature. In case one or more PPCR subspecies are found
363 in the backbone tree for a given species, *rand.tip* will define a MDCC for the infra-
364 specific PUTs following the standard procedures described in section 4.1. Finally, if none
365 of the trinomials in the group are found, *rand.tip* will first bind any of them to the tree,

366 and then all the others will be bound to the former following the bind-to-singleton
367 procedure.
368 We note that some available phylogenies use, likely in error, both the binomial
369 and trinomial form of a species to label different tips. For example, the GBOTB.tre mega-
370 tree (Smith & Brown, 2018) includes *Ziziphora taurica taurica* and *Ziziphora taurica* as
371 two different tips. In this case, *rand.tip* will randomly select either tip as the actual type
372 subspecies and ignore the other. Although the *check.info* function will warn the user about
373 the existence of possible duplicate taxa in the backbone tree, we strongly recommend the
374 user to visually explore tip labelling before expanding any backbone tree.
375
376 4.5. *Non-ultrametric phylogenies*
377 Previous software for PUT binding were conceived to be used with either ultrametric
378 phylogenies (trees with branch lengths where all tips are equidistant from the root) or
379 phylogenies without branch lengths. However, non-ultrametric trees where branch-length
380 are not proportional to time but character distance are also subject of ecological analyses
381 (e.g. Mishler et al., 2014; Mienna et al., 2020). The *check.info* function will warn the user
382 in case the backbone phylogeny is non-ultrametric, and *rand.tip* will force non-
383 ultrametric trees to be ultrametric –following the nnls method as implemented in
384 'phytools' R package (Revell, 2012)– if the argument 'forceultrametric' is set to TRUE
385 (default is FALSE). It is important to note that forcing phylogenies to be ultrametric in
386 this way should not be taken as a formal statistical approach for inferring an ultrametric
387 tree but a method to be deployed whenever a genuinely ultrametric phylogeny read from
388 file fails due to issues related to numerical precision (Revell, 2012). Thus, we strongly
389 recommend the user to visually explore phylogenetic trees that fail the ultrametricity test

390 of *check.info* before assuming the failure is due to numerical precision of computer
391 machinery.
392 If the backbone tree is non-ultrametric and the 'forceultrametric' argument is set
393 to FALSE, *rand.tip* will simulate the new branch lengths subtending the PUTs by
394 sampling from a uniform distribution U (0, L], where L is the length of the longest branch
395 in the backbone tree. In case a backbone phylogeny without branch lengths is provided,
396 *rand.tip* will output a phylogeny without branch lengths as well (i.e. topological
397 information only). Thus, the only condition for *rand.tip* to accept a phylogeny is that it is
398 rooted.
399
400 4.6 *Customizing a subset of branches to randomize PUTs*
401 The node-based workflow of randtip should suffice to cover most situations in PUT
402 binding exercises. However, the distribution of possible branches for the simulation might
403 not be drawn via MDCCs under some circumstances. For example, taxa of hybrid origin
404 often appear as the sister species or either parent depending on the set of molecular
405 markers that are used for the inference (Wang et al., 2014), in which case phylogenetic
406 uncertainty may pertain to only two singleton putative MDCCs (assuming that the
407 identity of the parents is known and both are represented in the backbone tree). Using the
408 auxiliar function *custom.branch*, the user will be able to customize specific subsets of
409 branches to bind PUTs across any segment of the phylogeny.
410
411 **5. Concluding remarks**
412 Randtip is, to our knowledge, the only framework for PUT binding that is completely
413 flexible and generalized, thus addressing several shortcomings of previous designs and
414 offering new opportunities to optimize parameter space in tree expansion exercises.

415 Although randtip can generate fully operative phylogenies using default settings, we
416 stress that accounting for phylogenetic uncertainty should not be conceived as a 'black
417 box' procedure for the immediate generation of phylogenies. Indeed, previous studies
418 have documented inaccuracies in the generation of such rough phylogenies due to the
419 'blind' use of software packages (Gastauer & Meira-Neto, 2013). As described above,
420 there is a variety of circumstances that may require customizing simulation parameters
421 for specific PUTs if we are to avoid suboptimal solutions. Beyond providing newly
422 designed tools to expand phylogenetic trees, the framework presented here will help
423 evolutionary biologists to get the most out of the evolutionary information that can be
424 used to guide tree expansion exercises.
425
426 **Author contribution statement**
427 RM-V conceived the ideas with inputs from IRG, IRG developed the code with the help
428 of RM-V and HL, RM-V led the writing. All the authors read and commented on the
429 manuscript.
430
431 **Glossary of terms**
432 *Phylogenetically uncertain taxon* (PUT): taxonomic unit (e.g. species, subspecies) that is
433 well delineated in the continuum of biodiversity but remain missing from available
434 molecular phylogenies (Rangel et al. 2015).
435 *Most derived consensus clade* (MDCC): the less inclusive phylogenetic clade that most
436 certainly contains a PUT (Rangel et al. 2015).
437 *Most recent common ancestor* (MRCA): the youngest phylogenetic node from which a
438 set of taxa descend.

Taxonomic rank: level of a group of organisms in the taxonomic hierarchy (e.g. genus, family, order).

Phylogenetically placed and co-ranked species (PPCR): set of species that are represented in the phylogeny and share one or more taxonomic ranks with a given PUT (e.g. congenerics, contribals, confamiliars).

Figure 1. Workflow of the randtip R package. The user provides a backbone phylogeny and a list of species that are to be bound to the former (some of these are already placed in the tree while others represent *phylogenetically uncertain taxa* or PUTs). The function *build.info* creates the template *info* and retrieves taxonomic information for the listed species (and for those represented in the phylogeny if the 'backbone' mode of randtip is set) from web repositories. The resultant data frame (*info*) can be evaluated with the function *check.info*. Once the user has edited *info* according to the particularities of each PUT, the data frame is passed through *info2input* to create the input object for the *rand.tip* function, which in turn will expand the backbone phylogeny.

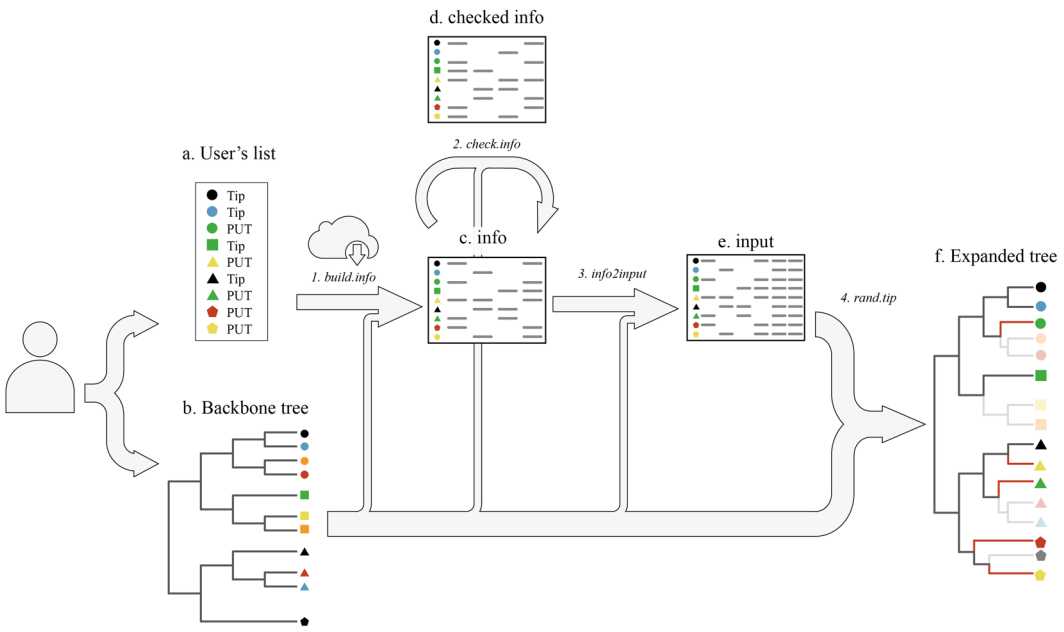


Figure 2. Types of phyletic groups formed by *phylogenetically placed and co-ranked* (PPCR) species (green circle symbols) and possible scenarios for PUT binding within each type. Non-PPCR species are colored in grey, and non-PPCR species that cause an otherwise monophyletic group of PPCR species to be paraphyletic or polyphyletic are colored in red. The candidate branches to bind the PUT in each scenario are colored in green. The fractions close to the phylogenetic nodes indicate the probability for the candidate clades to be selected under the polyphyletic scenario "frequentist".

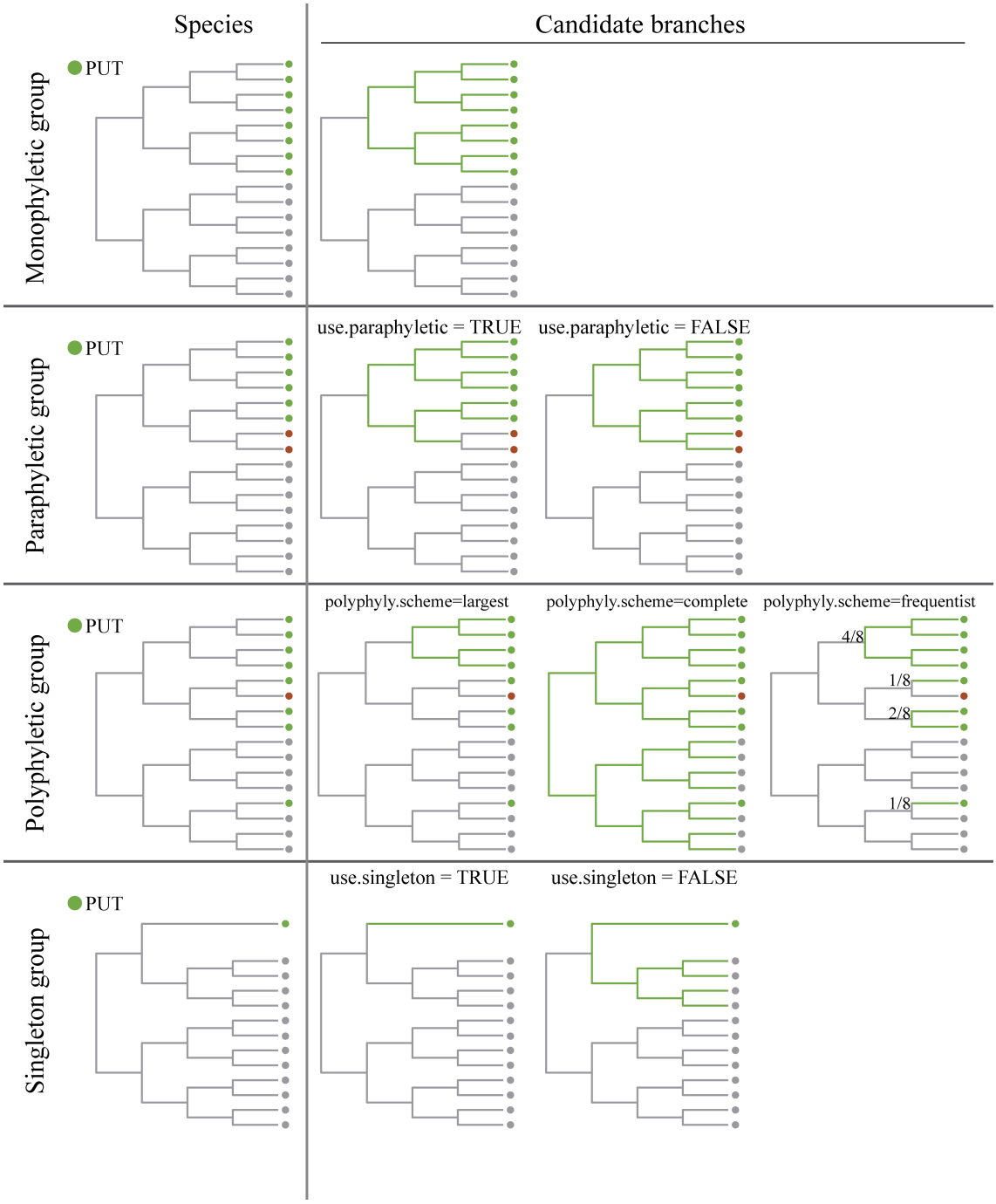
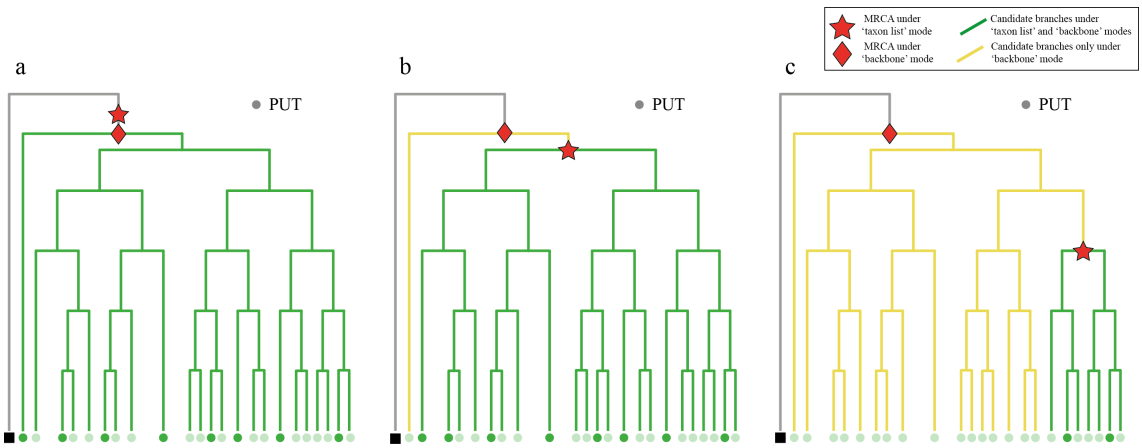


Figure 3. Scenarios of increasing divergence in the performance between the 'taxon list' and 'backbone' modes of randtip. The circle symbols on the phylogenetic tips represent *phylogenetically placed and co-ranked* (PPCR) species (e.g. confamiliars) of the PUT that is to be bound in each case, and the highlighted ones are those included in the user's list. The diamond red symbol (hereafter 'diamond node') indicates the crown node of the *most derived consensus clade* (MDCC) that is identified for the PUT when taxonomic information is available for all the species in the backbone phylogeny (i.e. under 'backbone' mode), and the star red symbol (hereafter 'star node') indicates the crown node of the MDCC that is identified when taxonomic information is available only for the species in the backbone phylogeny that are also included in the user's list (i.e. under 'taxon list' mode). In the first scenario (a), the diamond and star nodes are coincident, and thus both modes of randtip will use the same space of branch lengths (in green) to bind the PUT. In the second scenario (b), *the most recent common ancestor* (MRCA) of the subset of PPCR species that are represented in the user's list includes all PPCR species but one, and therefore the branch subtending the latter (in yellow) will never be selected under 'taxon list' mode. In the third scenario (c), a higher number of PPCR species are missing from the user's list, resulting in a smaller space of branch lengths to bind the PUT under 'taxon list' mode. Note that under 'backbone' mode, both the green and yellow branches would be candidates to bind the PUT.



References

Calatayud, J., Rodríguez, M. Á., Molina-Venegas, R., Leo, M., Horreo, J. L., & Hortal, J. (2019). Pleistocene climate change and the formation of regional species pools. *Proceedings of the Royal Society B*, *286*, 20190291. https://doi.org/10.1098/rspb.2019.0291

Cavender-Bares, J., Kozak, K. H., Fine, P. V. A., & Kembel, S. W. (2009). The merging of community ecology and phylogenetic biology. *Ecology Letters*, *12*, 693–715. https://doi.org/10.1111/j.1461-0248.2009.01314.x

Chamberlain, S., Szoecs, E., Foster, Z., Arendsee, Z., Boettiger, C., Ram, K., Bartomeus, I., Baumgartner, J., O'Donnell, J., Oksanen, J., Greshake Tzovaras, B., Marchand, P., Tran, V., Salmon, M., Li, G., & Grenié, M. (2020) *taxize: Taxonomic information from around the web*. R package version 0.9.98. https://github.com/ropensci/taxize

Chase, M. W., Christenhusz, M. J. M., Fay, M. F., Byng, J. W., Judd, W. S., Soltis, D. E., Mabberley, D. J., Sennikov, A. N., Soltis, P. S., & Stevens, P. F. (2016). An update of the Angiosperm Phylogeny Group classification for the orders and families of flowering plants: APG IV. *Botanical Journal of the Linnean Society*, *181*, 1–20. https://doi.org/10.1111/boj.12385

Davies, T. J. (2021). Ecophylogenetics redux. *Ecology Letters*, *24*, 1073–1088. https://doi.org/10.1111/ele.13682

Gastauer, M., & Meira-Neto, J. A. A. (2013). Avoiding inaccuracies in tree calibration and phylogenetic community analysis using Phylocom 4.2. *Ecological Informatics*, *15*, 85–90. https://doi.org/10.1016/j.ecoinf.2013.03.005

Hörandl, E., & Stuessy, T. F. (2010). Paraphyletic groups as natural units of biological classification. *TAXON*, *59*, 1641–1653. https://doi.org/10.1002/tax.596001

Huson, D. H., & Scornavacca, C. (2012). Dendroscope 3: An interactive tool for rooted phylogenetic trees and networks. *Systematic Biology*, *61*, 1061–1067. https://doi.org/10.1093/sysbio/sys062

Janssens, S., Couvreur, T. L. P., Mertens, A., Dauby, G., Dagallier, L.-P., Abeele, S. V., Vandelook, F., Mascarello, M., Beeckman, H., Sosef, M., Droissart, V., Bank, M. van der, Maurin, O., Hawthorne, W., Marshall, C., Réjou-Méchain, M., Beina, D., Baya, F., Merckx, V., … Hardy, O. (2020). A large-scale species

level dated angiosperm phylogeny for evolutionary and ecological analyses. *Biodiversity Data Journal*, *8*, e39677. https://doi.org/10.3897/BDJ.8.e39677

Jin, Y., & Qian, H. (2019). V.PhyloMaker: An R package that can generate very large phylogenies for vascular plants. *Ecography*, *42*, 1353–1359. https://doi.org/10.1111/ecog.04434

Lamsdell, J. C., & Congreve, C. R. (2021). Phylogenetic paleoecology: Macroecology within an evolutionary framework. *Paleobiology*, *47*, 171–177. https://doi.org/10.1017/pab.2020.61

Malécot, V., & Nickrent, D. L. (2008). Molecular phylogenetic relationships of Olacaceae and related Santalales. *Systematic Botany*, *33*, 97–106. https://doi.org/10.1600/036364408783887384

Martins, W. S., Carmo, W. C., Longo, H. J., Rosa, T. C., & Rangel, T. F. (2013). SUNPLIN: Simulation with Uncertainty for Phylogenetic Investigations. *BMC Bioinformatics*, *14*, 324. https://doi.org/10.1186/1471-2105-14-324

Mishler, B. D., Knerr, N., González-Orozco, C. E., Thornhill, A. H., Laffan, S. W., & Miller, J. T. (2014). Phylogenetic measures of biodiversity and neo-and paleo-endemism in Australian Acacia. *Nature Communications*, *5*, 4473. https://doi.org/10.1038/ncomms5473

Molina-Venegas, R., Ramos-Gutiérrez, I., & Moreno-Saiz, J. C. (2020). Phylogenetic patterns of extinction risk in the endemic flora of a Mediterranean hotspot as a guiding tool for preemptive conservation actions. *Frontiers in Ecology and Evolution*, *8*, 373. https://doi.org/10.3389/fevo.2020.571587

Molina-Venegas, R., Rodríguez, M. Á., Pardo-de-Santayana, M., Ronquillo, C., & Mabberley, D. J. (2021). Maximum levels of global phylogenetic diversity efficiently capture plant services for humankind. *Nature Ecology & Evolution*, *5*, 583–588. https://doi.org/10.1038/s41559-021-01414-2

Mouquet, N., Devictor, V., Meynard, C. N., Munoz, F., Bersier, L.-F., Chave, J., Couteron, P., Dalecky, A., Fontaine, C., Gravel, D., Hardy, O. J., Jabot, F., Lavergne, S., Leibold, M., Mouillot, D., Münkemüller, T., Pavoine, S., Prinzing, A., Rodrigues, A. S. L., … Thuiller, W. (2012). Ecophylogenetics: Advances and perspectives. *Biological Reviews*, *87*, 769–785. https://doi.org/10.1111/j.1469-185X.2012.00224.x

Pentinsaari, M., Ratnasingham, S., Miller, S. E., & Hebert, P. D. (2020). BOLD and GenBank revisited–Do identification errors arise in the lab or in the sequence libraries? *PLoS One*, *15*, e0231814.

Rangel, T. F., Colwell, R. K., Graves, G. R., Fučíková, K., Rahbek, C., & Diniz-Filho, J. A. F. (2015). Phylogenetic uncertainty revisited: Implications for ecological analyses. *Evolution*, *69*, 1301–1312. https://doi.org/10.1111/evo.12644

Revell, L. J. (2012). phytools: An R package for phylogenetic comparative biology (and other things). *Methods in Ecology and Evolution*, *3*, 217–223. https://doi.org/10.1111/j.2041-210X.2011.00169.x

Sayers, E. W., Beck, J., Brister, J. R., Bolton, E. E., Canese, K., Comeau, D. C., Funk, K., Ketter, A., Kim, S., Kimchi, A., Kitts, P. A., Kuznetsov, A., Lathrop, S., Lu, Z., McGarvey, K., Madden, T. L., Murphy, T. D., O'Leary, N., Phan, L., … Ostell, J. (2020). Database resources of the National Center for Biotechnology Information. *Nucleic Acids Research*, *48*, D9–D16. https://doi.org/10.1093/nar/gkz899

Smith, S. A., & Brown, J. W. (2018). Constructing a broadly inclusive seed plant phylogeny. *American Journal of Botany*, *105*, 302–314. https://doi.org/10.1002/ajb2.1019

Smith, S. A., & Walker, J. F. (2019). PyPHLAWD: A python tool for phylogenetic dataset construction. *Methods in Ecology and Evolution*, *10*, 104–108. https://doi.org/10.1111/2041-210X.13096

Stork, N. E. (2018). How many species of insects and other terrestrial arthropods are there on Earth? *Annual Review of Entomology*, *63*, 31–45. https://doi.org/10.1146/annurev-ento-020117-043348

Upham, N. S., Esselstyn, J. A., & Jetz, W. (2019). Inferring the mammal tree: Species-level sets of phylogenies for questions in ecology, evolution, and conservation. *PLOS Biology*, *17*, e3000494. https://doi.org/10.1371/journal.pbio.3000494

Wang, Z., Du, S., Dayanandan, S., Wang, D., Zeng, Y., & Zhang, J. (2014). Phylogeny reconstruction and hybrid analysis of *Populus* (Salicaceae) based on nucleotide sequences of multiple single-copy nuclear genes and plastid fragments. *PloS One*, *9*, e103645. https://doi.org/10.1371/journal.pone.0103645

Webb, C. O., Ackerly, D. D., McPeek, M. A., & Donoghue, M. J. (2002). Phylogenies and community ecology. *Annual Review of Ecology and Systematics*, *33*, 475–505. https://doi.org/10.1146/annurev.ecolsys.33.010802.150448

Webb, C. O., & Donoghue, M. J. (2005). Phylomatic: Tree assembly for applied phylogenetics. *Molecular Ecology Notes*, *5*, 181–183. https://doi.org/10.1111/j.1471-8286.2004.00829.x