Received: 8 February 2023 | Revised: 25 May 2023 | Accepted: 30 June 2023

DOI: 10.1111/geb.13733

Global Ecology and Biogeography | A Journal of Macroecology | WILEY

METHOD

A generalized framework to expand incomplete phylogenies using non-molecular phylogenetic information

Ignacio Ramos-Gutiérrez[1,2] | Herlander Lima[3] | Bruno Vilela[4] | Rafael Molina-Venegas[2,5]

[1]Department of Biology (Botany), Universidad Autónoma de Madrid, Madrid, Spain

[2]Biodiversity and Global Change Research Center (CIBC-UAM), Universidad Autónoma de Madrid, Madrid, Spain

[3]GLOCEE—Global Change Ecology and Evolution Group, Department of Life Sciences, Universidad de Alcalá, Alcalá de Henares, Spain

[4]Instituto de Biologia, Universidade Federal da Bahia, Salvador, Brazil

[5]Department of Ecology, Universidad Autónoma de Madrid, Madrid, Spain

Correspondence
Rafael Molina-Venegas and Ignacio Ramos-Gutiérrez, Biodiversity and Global Change Research Center (CIBC-UAM), Universidad Autónoma de Madrid, Madrid 28049, Spain.
Email: rafmolven@gmail.com and ignacio.ramosgutierrez@uam.es

Funding information
Ministry of Science and Innovation of Spain, Grant/Award Number: CGL2017-86926-P; Regional Government of Madrid, Spain, Grant/Award Number: CM/JIN/2019-005

Handling Editor: Franziska Schrodt

Abstract

Aim: 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. 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.

Innovation: 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 phylogenetically uncertain taxa in backbone phylogenies through a fully customizable and automatic procedure that uses taxonomic ranks as a major source of phylogenetic information. Although randtip can generate 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 "quick-and-dirty" approaches 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 tutorial that was designed to provide guidelines to non-specialist users.

Main Conclusions: 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 phylogenies and thereby account for phylogenetic uncertainty in quantitative analyses.

KEYWORDS
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 (Cavender-Bares et al., 2009; Mouquet et al., 2012; Webb et al., 2002), 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, 2019), which have lifted our understanding of species evolutionary relationships to unprecedent levels. However, and despite the phylogeny of certain groups, such as mammals, is nearly completed (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), that is, taxonomic units (e.g., species, subspecies) that are well delineated in the continuum of biodiversity but remain missing from available phylogenies. Then, all acceptable taxonomic, morphological, or behavioural information on the PUTs is used to conservatively define their *most derived consensus clades* (MDCCs), that is, the less inclusive phylogenetic nodes that most certainly contain them. Finally, each PUT is assigned to a random point along one randomly selected branch of its corresponding MDCC, and the procedure is replicated a high number of times to obtain a distribution of possible trees that can be used in downstream analyses iteratively. While the "true" phylogenetic hypothesis will most certainly remain unsampled, the workflow allows exploring the parameter space, thereby quantifying the extent to which phylogenetic uncertainty has a significant impact in the analyses (e.g., Calatayud et al., 2019; Molina-Venegas et al., 2021). Rangel et al. (2015) accompanied their framework with the software SUNPLIN, a set of algorithms for randomly expanding phylogenies using the aforementioned procedure (Martins et al., 2013).

Although Rangel et al. (2015) suggested that the identification of MDCCs should be based on expert taxonomic evaluation, such knowledge is in practice beyond the reach of most researchers, particularly when dealing with very large phylogenies that often encompass a wide spectrum of taxonomic groups and thousands of species. In a valuable attempt to automatize the identification of MDCCs, Jin and Qian (2019) developed V.PhyloMaker, an R package that can generate large phylogenies of vascular plants (recently updated as U.PhyloMaker to include vertebrate animals; Jin & Qian, 2023). PhyloMaker is based on the seminal idea of the classical software Phylomatic (Webb & Donoghue, 2005), which uses a taxonomically informed backbone mega-tree to automatically define MDCCs (in the case of PhyloMaker, genus or family nodes in case the former are not available) and bind the PUTs to the selected clades. Beyond covering features that were already implemented in Phylomatic, PhyloMaker provides an option to insert PUTs in randomly chosen nodes below the crown node of the corresponding MDCCs, so that a distribution of possible phylogenies can be generated with relatively little effort (Jin & Qian, 2019).

However, we note that current available tools for the insertion of PUTs, while valuable, have some important drawbacks. For example, 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). 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 trees for which the software was implemented. Also, the definition of MDCCs on the basis of a few taxonomic ranks (e.g., PhyloMaker only considers genus or family nodes otherwise) 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 paraphyletic groups (Hörandl & Stuessy, 2010) and the impossibility to fully customize the space of phylogenetic edges 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 procedure that uses automatically retrieved and arranged taxonomic data as a major source of phylogenetic information. Although randtip can generate 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 tutorial that was designed to provide guidelines to non-specialist users (see Supporting Information).

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 them to the backbone tree according to the parameters that are set for the simulations, and a phylogeny including all the taxa in the user's list is returned (Figure 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 Supporting Information for detailed step-by-step examples).

2.1 | Input files

The workflow of randtip is guided by a dataframe R object (hereafter "*info*") and the instructions that are passed through the main function of the package (*rand_tip*). The dataframe *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 where the operative taxa represent Linnean binomials (genus and species with or without subspecific epithets), although genus-level phylogenies are also supported. The *info* template can be created automatically using the auxiliary function *build_info*, which is fed with species names in a character vector or single-column dataframe. Besides, *build_info* can interact with a suite of taxonomic repositories—currently implemented for "ncbi" (default), "itis", "gbif" and "bold" via the *classification* function of "taxize" R package (Chamberlain et al., 2020)—to automatically retrieve and arrange taxonomic information that will be used to identify putative supra-generic MDCCs for the PUTs (note that information to define genus-level MDCCs is intrinsically contained in the scientific names of the species). This can be done by setting the argument "find.ranks" of *build_info* to TRUE (default). We recommend providing at least one supra-generic rank (e.g., taxonomic family) for all the species in *info*, which will be used to define MDCCs whenever the genus of the PUTs is missing in the phylogeny (otherwise the PUTs will not be bound). Often the user will need to further edit *info* once the template is created (for example, to customize binding parameters for certain PUTs or to amend taxonomic mistakes in web repositories).

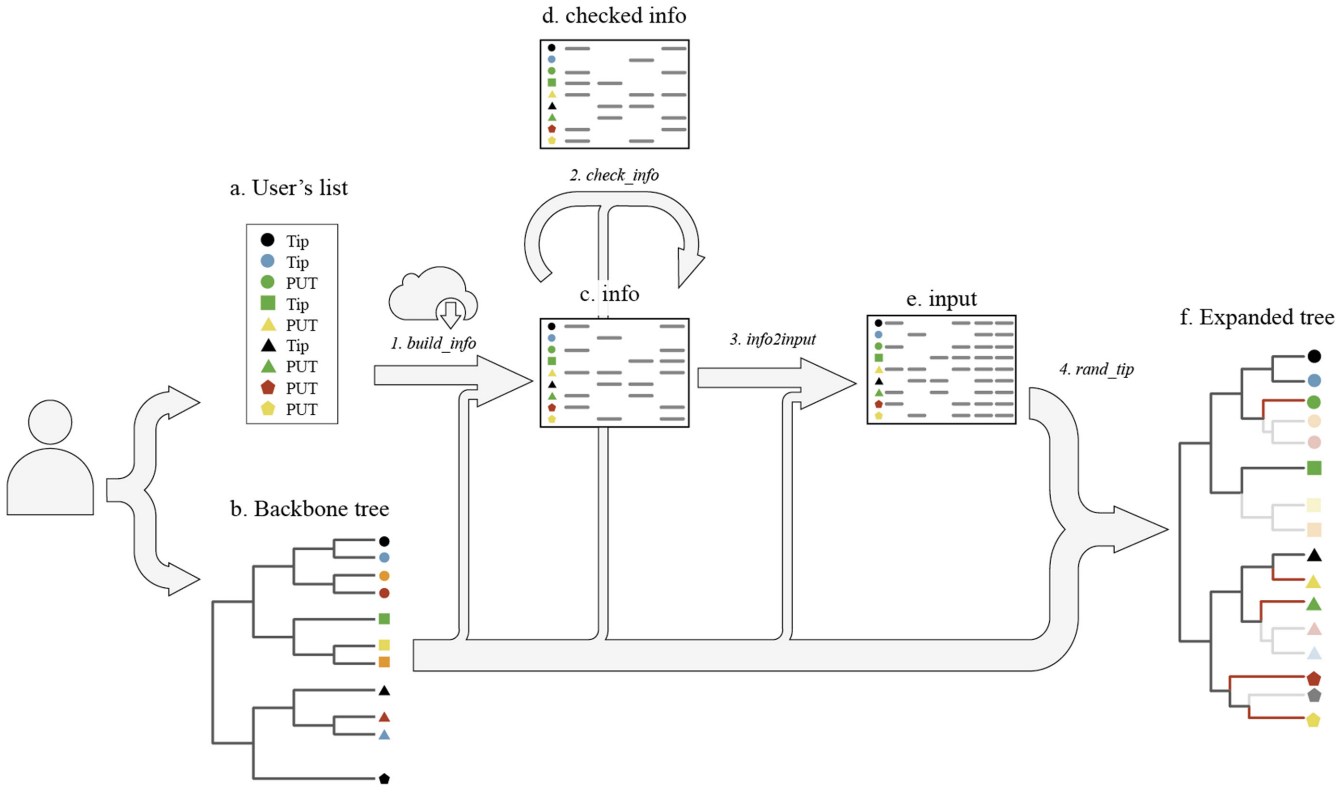

FIGURE 1 Schematic workflow of randtip. The user provides a backbone phylogeny and a list of taxa that are to be bound to the former (some of these are already placed in the tree while others represent *phylogenetically uncertain taxa* or PUTs). The function *build_info* creates the template *info* and retrieves taxonomic information for the listed taxa (and for those represented in the phylogeny if the "backbone" mode of randtip is set to TRUE) from web repositories. The resultant dataframe (*info*) can be evaluated with the function *check_info*. Once the user has edited *info* according to the particularities of each PUT, the dataframe is passed through *info2input* to create the input object for the *rand_tip* function, which in turn will expand the backbone phylogeny (in red, terminal branches subtending newly bound PUTs).

This can be done directly in R using the auxiliary function *edit_info* or exporting the dataframe as a spreadsheet (e.g., csv or xlsx) and importing it back into R once all the edits are completed.

The user must provide a backbone phylogeny as a *phylo* R object. Although randtip can identify MDDCs on the sole basis of taxonomic ranks of the species that are included both in the user's list and the backbone tree (hereafter "taxon list" mode), MDDCs can also be identified based on taxonomic ranks of all the species that are represented in the tree regardless of their presence in the user's list (hereafter "backbone" mode). Both approaches have pros and cons (see Section 3), but they will perform identically whenever the genus of the PUTs is represented by at least one species in the backbone tree. To use the "backbone" mode of randtip, the argument "mode" of *build_info* must be set to "backbone" (default) for the software to include all the species in the phylogeny as rows in the *info* dataframe (otherwise, only the species that appear both in the phylogeny and the user's list will be included), so that their taxonomic information can be automatically retrieved and arranged (if the argument "find. ranks" of *build_info* is set to TRUE).

Once the dataframe *info* is assembled, we strongly recommend checking the incidence of PUTs in the user's list and their putative MDCCs. This can be done with the auxiliary function *check_info*, which will inform on the PUT status of the species, the presence of possible spelling errors, putative MDCCs, and the phyletic nature of the set of species that are included in each MDCC and share taxonomic ranks (e.g., congenerics, contribals, confamiliars) with the corresponding PUT—hereafter *phylogenetically placed and co-ranked* (PPCR) species. Also, the tip labels of the backbone tree are checked out for duplicates (e.g., *Ziziphora taurica taurica* and *Ziziphora taurica*), and the software evaluates if the tree is ultrametric or not. By default, *check_info* will make use of parallel processing to speed up the search for possible spelling errors and the identification of the phyletic nature of PPCR species, which is convenient for very large datasets. The auxiliary functions *get_clade* and *plot_clade* can in turn be used to extract and plot any subtree representing putative MDCCs, so that the user can visually explore them using the R graphic window (PPCR and non-PPCR species of the PUT are shown in contrasting colours, see Supporting Information for examples). Exploring MDCCs is particularly recommended to optimize PUT binding, and particularly when PPCR species form polyphyletic groups (see Section 2.2). Alternatively, subtrees can be exported in Newick format to visualize them using auxiliary software such as Dendroscope (Huson & Scornavacca, 2012), which may be convenient for very large clades. Once the MDCCs are defined and the user has optionally customized parameter values for individual PUTs, the wrapping function *info2input* is fed with the dataframe *info* and the backbone phylogeny to create a final dataset that will be passed through the *rand_tip* function to expand the tree. This final dataset ensures consistent structure for use in *rand_tip* and allows generating as many trees as desired without the need to search for putative MDCCs in *info* repeatedly. This is done by *info2input* just once, a computationally intense task that is, by default, expedited using parallel processing.

2.2 | Selecting MDCCs and binding PUTs

The binding of PUTs is conducted with the function *rand_tip*, which includes a variety of parameters that are passed through the whole simulation (Table S1). However, all the parameter arguments of *rand_tip* can be adjusted independently for each PUT by editing in the corresponding slots of *info*, which makes the framework completely flexible and customizable.

Randtip will always try to find the less inclusive MDCC of each PUT according to the taxonomic ranks that are provided in *info*, starting from genus level and up to class level until a MDCC is found. Regardless of the mode of randtip that is set by the user ("backbone" or "taxon list"), the software will always first attempt to define genus-level MDCCs as the *most recent common ancestor* (MRCA) of all the species in the backbone tree that are congeneric to the PUTs. However, MDCCs above the genus level may differ between the two modes of randtip. On "taxon list" mode, supra-generic MDCCs are defined as the MRCA of all the species in the user's list that are PPCR with the target PUT (e.g., contribals, consubfamiliars, confamiliars). In contrast, the "backbone" mode (default) defines supra-generic MDCCs as the MRCA of all the species in the backbone phylogeny (regardless of their presence in the user's list) that are PPCR with the target PUT (see Figure 2 and Section 3 for an extended discussion).

By default, *rand_tip* will bind each PUT to a randomly selected branch below the crown node of the corresponding MDCC, the probability of being added along any branch being directly proportional to the length of the branch—if the argument "prob" is set to TRUE (default). Alternatively, branches can be selected on the basis of equal probability, and in either case the user can decide to add the stem branch of the clade to the pool of candidate branches—if the argument "use.stem" is set to TRUE (default is FALSE). The exact point to insert the PUT in the selected branch is sampled from a uniform distribution. Importantly, the extent to which the default behaviour of *rand_tip* to insert PUTs represents an optimal scenario may depend on the phyletic nature of their PPCR species. These can represent monophyletic (whenever the MDCC is exclusively shaped by species that are PPCR with the target PUT), singleton (terminal branch), paraphyletic (whenever the species that map within the MDCC but are not PPCR with the PUT form either a monophyletic or singleton group) or polyphyletic (set of PPCR species that does not fit any of the previous categories) groups (see Section 4.1 and Figure 3). The PPCR species of a given PUT could form a polyphyletic group simply because one of them maps clearly away from the main (monophyletic) cluster of PPCR species—for example, because the outlying PPCR species is labelled in error (Pentinsaari et al., 2020)—in which case the default behaviour of *rand_tip* to bind the PUT (i.e., any branch below the crown node of the largest monophyletic cluster) would be reasonable. However, the polyphyletic nature of the PPCR species could also be due to "intruder" species that map within an otherwise monophyletic cluster, in which case the default behaviour of *rand_tip* could be suboptimal because the evidence that the largest monophyletic cluster of the group includes the PUT is less conclusive (Figure 3).

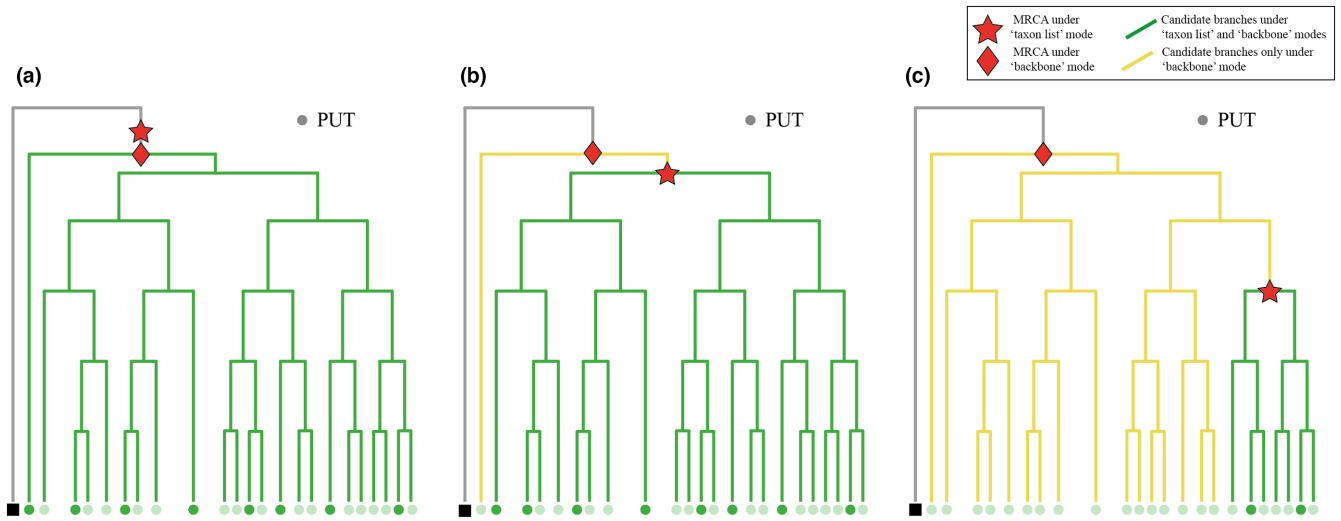

FIGURE 2 Scenarios of increasing divergence in the performance between the "taxon list" and "backbone" modes of randtip. The circle symbols on the phylogenetic tips represent *phylogenetically placed and co-ranked* (PPCR) species (e.g., confamiliars) of the PUT, and the highlighted ones are those included in the user's list in each scenario. 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.

As we discuss in Section 4, randtip allows the user to optimize the binding of PUTs according to the specifics of each case.

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

3 | THE "TAXON LIST" AND "BACKBONE" MODES OF RANDTIP

The first decision the user will have to tackle is choosing between the "taxon list" and "backbone" modes of randtip. As we stated earlier, both approaches will perform identically as long as the genus of the PUTs is represented by at least one species in the backbone phylogeny, yet supra-generic MDCCs may differ between the two modes. For example, it might happen that some of the PPCR species of a given PUT (let us say confamiliars) are missing in the user's list but are represented in the backbone phylogeny. Thus, in case these PPCR species were phylogenetically external to the confamiliars of the PUT that are included in the user's list, the "backbone" mode of randtip would define an older MDCC than "taxon list" (Figure 2). It follows that the extent of the divergence in the functioning between both modes (whenever a supra-generic MDCC is to be defined) depends on the phylogenetic placement of the PPCR species that are included in the user's list. In sum, the "backbone" mode works based on the "true" supra-generic MDCCs (but note that these may neither represent the actual MDCCs as the backbone phylogenies are often not fully comprehensive) with the trade-off that it is a more time-consuming approach than "taxon list". In contrast, the latter might define younger supra-generic MDCCs (meaning more restricted parameter space to bind PUTs) under some circumstances (Figure 2). We recommend considering the "backbone" mode as a first option (default) and use "taxon list" only when there is a low incidence of PUTs requiring supra-generic binding and/or low mismatch in the nodes defining supra-generic MDCCs between both approaches (see Figure 2 and Supporting Information for an extended discussion).

4 | NEWLY DESIGNED FEATURES FOR PUT BINDING

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

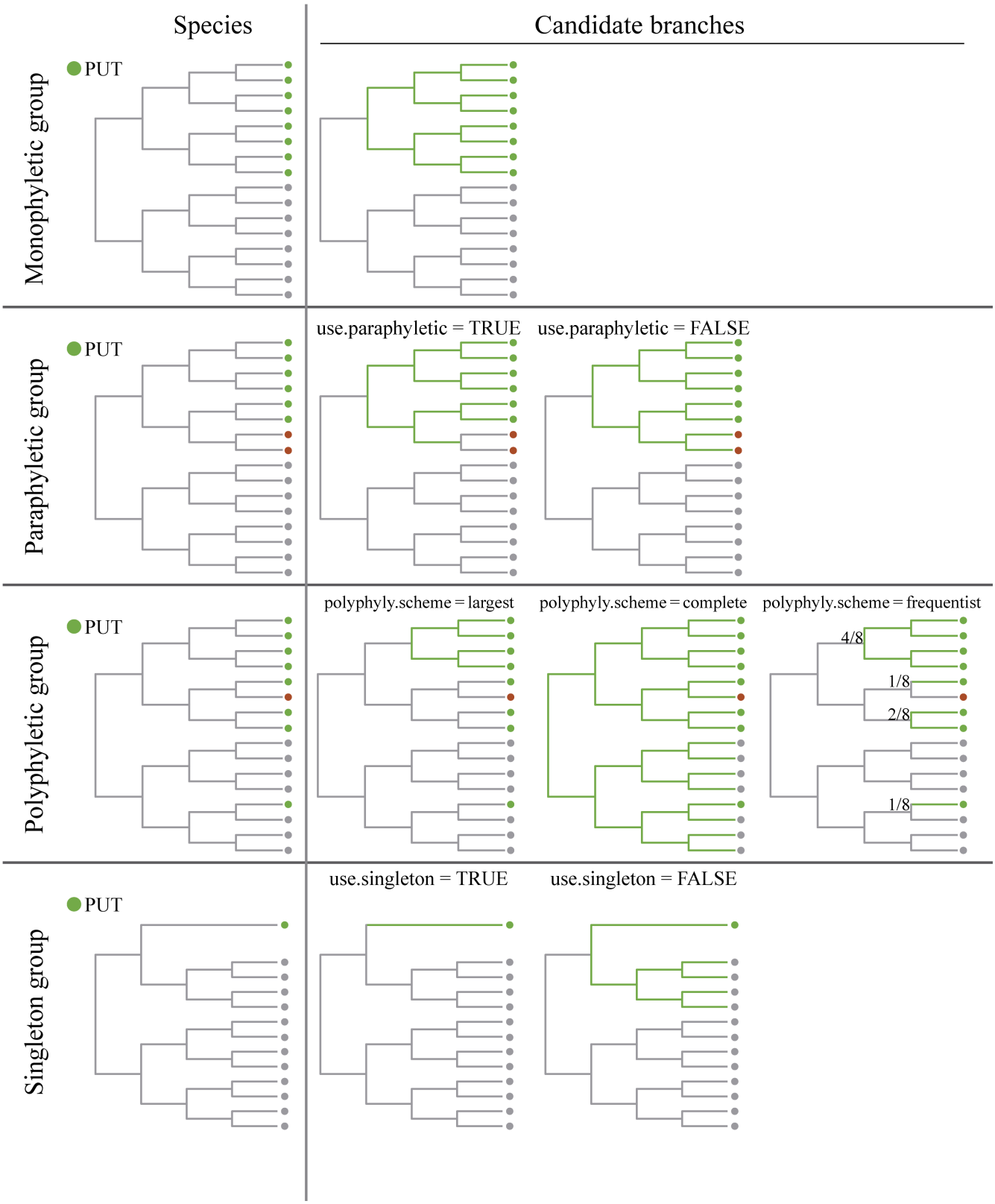

FIGURE 3 Types of phyletic groups formed by *phylogenetically placed and co-ranked* (PPCR) species (green circle symbols) and possible scenarios for PUT binding within each type. Non-PPCR species are in grey, and non-PPCR "intruder" species are in red. The candidate branches to bind the PUT in each scenario are in green (the vertical segments of the trees are purely aesthetic and were coloured to ease clade visualization). The fractions close to the phylogenetic nodes indicate the probability for the candidate clades to be selected under the scenario "frequentist".

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 behaviour 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" 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 the MDCC of a PUT is shaped by a polyphyletic group of PPCR species, the software allows the user to choose between three different binding scenarios using the "polyphyly.scheme" argument. 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; Figure 3). 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 to the size of the cluster, and then the PUT will be inserted in the selected cluster. If the option "complete" is set, *rand_tip* will bind the PUT to a randomly selected branch below the crown node of the MDCC (most conservative scenario).

In case the MDCC of a PUT is defined by a paraphyletic group of PPCR species, two different scenarios are eligible. If the argument "use.paraphyletic" is set to TRUE (default), the candidate branches are those that keep the paraphyletic nature of the group unchanged after the binding (Figure 3). Otherwise, the randomization will be conducted as if the MDCC were defined by a monophyletic group of species. Importantly, certain taxonomic groups such as the Olacaceae s.l. plant family are paraphyletic (Chase et al., 2016), and thus randomizing PUTs at any point below the crown node of this family (i.e., setting "use.paraphyletic" to FALSE) may result in an excessively conservative parameter space that would encompass almost the entire Santalales order (Malécot & Nickrent, 2008).

In case the MDCC of a PUT is defined by one single PPCR species (Figure 3), *rand_tip* will by default bind the PUT to the terminal branch subtending the only PPCR species, and whenever the MDCC is no longer singleton (because at least one PUT was already bound), *rand_tip* will consider the entire newly formed clade (same height as the original singleton clade) to sample candidate branches. We will refer to this procedure as "bind-to-singleton" hereafter. However, if the argument "use.singleton" is set to FALSE (default is TRUE), the parent node of the singleton PPCR species will be defined as the MDCC of the PUT instead (Figure 3). Although the latter scheme is more conservative than the former, it may lead to suboptimal solutions under some circumstances. For example, the parameter space to randomize a PUT whose MDCC is shaped by one single species that is the only representative of a subfamily in the phylogeny can be drastically increased in case the subfamily is the sister group to the rest of the family. Note that all these parameters can be specifically set for individual PUTs by filling in the corresponding slots of *info*.

4.2 | Manual definition of MDCCs

Although randtip was conceived to automatize the definition of MDCCs based on taxonomic ranks, the user can manually define MDCCs for the PUTs. This can be done by filling in the corresponding slots of the columns "taxa1" and "taxa2" of *info*. As long as these slots are not set to "NA" (default), the MDCCs of the PUTs will be defined on the basis of this information instead. For example, if the slots "taxa1" and "taxa2" of a PUT are filled in with different species names, the PUT will be bound to a randomly selected branch below the MRCA of the two given species. If both slots are filled in with the same species name, *rand_tip* will follow the bind-to-singleton procedure to insert the PUT as sister to the so defined species, and in case the same genus is provided the PUT will be inserted as sister to the clade defined by the MRCA of all the species in that genus.

4.3 | Respecting monophyletic and paraphyletic groups

By default, *rand_tip* will never bind a PUT to a branch that results in breaking the monophyletic or paraphyletic nature of a group (of any taxonomic rank) unless the arguments "respect.mono" and "respect.para" are set to FALSE (default is TRUE). Thus, while previous software followed either approach (e.g., Phylomaker always respects monophyletic genera but SUNPLIN does not), randtip offers the user the possibility to choose between both options, either by setting the arguments of the *rand_tip* function or on a customized basis for individual PUTs by filling in the corresponding slots of *info*.

4.4 | Clumping PUTs

Some genera may not be represented in the phylogeny, and thus their representative species will likely form a polyphyletic group if they are to be bound randomly below the crown node of the corresponding supra-generic MDCC. However, the user could be certain in that a group of congeneric PUTs whose genus is missing in the phylogeny is monophyletic. Thus, if the argument "clump.puts" is set

to TRUE (default), *rand_tip* will first bind one of the congeneric PUTs, and then the rest will be bound following the bind-to-singleton procedure. Similarly, it may happen that supra-generic taxonomic groups are not represented in the phylogeny, in which case *rand_tip* will clump the PUTs as described above and following the taxonomic hierarchy so that the missing taxonomic groups will form monophyletic clusters once all the PUTs are bound. As any other randomization parameter of randtip, the user may decide the PUTs that will be clumped in this way by setting the "clump.puts" option individually in the corresponding slots of *info*.

Trinomials representing infra-specific taxa (e.g., subspecies) are also supported. If "clump.puts" is set to TRUE, *rand_tip* will clump PUTs with infra-specific information according to their specific epithets (i.e., second name in the trinomial). To do so, *rand_tip* will first check if any of the trinomial PUTs that share specific epithet are represented in the phylogeny. This search also takes into account the type subspecies of the species, which will be detected in either trinomial (e.g., *Ablepharus chernovi chernovi*) or binomial (e.g., *Ablepharus chernovi*) nomenclature. In case one or more PPCR subspecies are found in the backbone tree, *rand_tip* will define a MDCC for the infra-specific PUTs following the standard procedures described in Section 4.1. Finally, if none of the trinomials in the group are found, *rand_tip* will first bind any of them to the tree, and then all the others will be bound following the bind-to-singleton procedure.

We note that some available phylogenies use, likely in error, both the binomial and trinomial form of a species to label different tips. For example, the GBOTB.tre mega-tree (Smith & Brown, 2018) includes *Ziziphora taurica taurica* and *Ziziphora taurica* as two different tips, and the GBOTB.extended.tre mega-tree (Jin & Qian, 2019) includes both *Saxifraga serpyllifolia* and *Saxifraga serpyllifolia serpyllifolia*. In these cases, *rand_tip* will randomly select either tip as the actual type subspecies and ignore the other. Although the *check_info* function will warn the user about the existence of possible duplicate taxa in the backbone tree (see Supporting Information for an example), we strongly recommend the user to visually revise tip labelling before expanding any backbone tree.

4.5 | Non-ultrametric phylogenies

Previous software for PUT binding were conceived to be used with either ultrametric phylogenies (trees with branch lengths where all tips are equidistant from the root) or phylogenies without branch lengths. However, non-ultrametric trees where branch length is not proportional to time but character distance are also subject of ecological analyses (e.g., Mishler et al., 2014). The *check_info* function will warn the user in case the backbone phylogeny is non-ultrametric, and *rand_tip* will force non-ultrametric trees to be ultrametric—following the nnls method as implemented in "phytools" R package (Revell, 2012)—if the argument "forceultrametric" is set to TRUE (default is FALSE). It is important to note that forcing phylogenies to be ultrametric in this way should not be taken as a formal statistical approach for inferring an ultrametric tree but a method to be deployed whenever a genuinely ultrametric phylogeny read from file fails due to issues related to numerical precision (Revell, 2012). Thus, we strongly recommend the user to visually explore phylogenetic trees that fail the ultrametricity test of *check_info* before assuming the failure is due to numerical precision of computer machinery.

If the backbone tree is non-ultrametric and the "forceultrametric" argument is set to FALSE, *rand_tip* will simulate the new branch lengths by sampling from a negative exponential distribution *EX* ($1/\lambda$), where λ is the inverse of the mean terminal branch length in the backbone tree. In case a backbone phylogeny without branch lengths is provided, *rand_tip* will output a phylogeny without branch lengths as well (i.e., topological information only). Hence, the only condition for *rand_tip* to accept a phylogeny is that it is rooted.

4.6 | Customizing a subset of branches to randomize PUTs

The node-based workflow of randtip should suffice to cover most situations in PUT binding exercises. However, the distribution of possible branches for the simulation might not be drawn via MDCCs under some circumstances. For example, taxa of hybrid origin often appear as the sister species of either parent depending on the set of molecular markers that are used for the inference (Wang et al., 2014), in which case phylogenetic uncertainty may pertain to only two singleton putative MDCCs (assuming that the identity of the parents is known and both are represented in the backbone tree). Using the auxiliary function *custom_branch*, the user can customize specific subsets of branches to bind PUTs across any segment of the phylogeny.

5 | CONCLUDING REMARKS

Randtip is, to our knowledge, the only framework for PUT binding that is completely flexible and generalized, thus addressing several shortcomings of previous designs and offering new opportunities to optimize parameter space in tree expansion exercises. Although randtip can generate fully operative phylogenies using default settings, we stress that accounting for phylogenetic uncertainty should not be conceived as a "black box" procedure for the immediate generation of phylogenies. Indeed, previous studies have documented inaccuracies in the generation of such "quick-and-dirty" phylogenies due to the "blind" use of software packages (Gastauer & Meira-Neto, 2013). Phylogenetic uncertainty should always be tackled with caution and restraint, for there is a variety of circumstances that may require customizing simulation parameters for specific PUTs if we are to avoid suboptimal solutions. Beyond providing newly designed tools to expand phylogenetic trees, the framework presented here will help evolutionary biologists to get the most out of the evolutionary information that can be used to guide tree expansion exercises.

AUTHOR CONTRIBUTIONS

Rafael Molina-Venegas conceived the ideas with inputs from Ignacio Ramos-Gutiérrez; Ignacio Ramos-Gutiérrez developed the code with the help of Rafael Molina-Venegas, Herlander Lima and Bruno Vilela; and Rafael Molina-Venegas led the writing. All the authors read the manuscript and approved submission.

ACKNOWLEDGEMENTS

IR-G was primarily supported by the project CM/JIN/2019-005 entitled "Plant evolutionary history and human well-being in a changing world; assessing theoretical foundations using empirical evidence and new phylogenetic tools", granted to R.M.-V. (Regional Government of Madrid, Spain) and also by the project CGL2017-86926-P (Ministry of Science and Innovation of Spain).

CONFLICT OF INTEREST STATEMENT

The authors declare no conflicts of interest.

DATA AVAILABILITY STATEMENT

No data were used to produce this manuscript.

CODE AVAILABILITY STATEMENT

All the code can be sourced from the GitHub repository (https://github.com/iramosgutierrez/randtip) as explained in the supplementary material, and the software will be delivered as a formal R package in CRAN upon acceptance of the manuscript.

ORCID

Ignacio Ramos-Gutiérrez https://orcid.org/0000-0002-8675-0114
Rafael Molina-Venegas https://orcid.org/0000-0001-5801-0736

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., van der Bank, M., 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

Jin, Y., & Qian, H. (2023). U.PhyloMaker: An R package that can generate large phylogenetic trees for plants and animals. *Plant Diversity*, *45*, 347–352. https://doi.org/10.1016/j.pld.2022.12.007

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

BIOSKETCH

Ignacio Ramos-Gutiérrez is a PhD candidate interested in exploring floristic and biogeographic patterns of Iberian vascular plants, with a major focus on the phylogenetic dimension of biodiversity for conservation.

SUPPORTING INFORMATION

Additional supporting information can be found online in the Supporting Information section at the end of this article.

How to cite this article: Ramos-Gutiérrez, I., Lima, H., Vilela, B., & Molina-Venegas, R. (2023). A generalized framework to expand incomplete phylogenies using non-molecular phylogenetic information. *Global Ecology and Biogeography*, *32*, 1707–1716. https://doi.org/10.1111/geb.13733