Convex recoloring as an evolutionary marker

Zeev Frenkel a, Yosef Kiat b, Ido Izhaki c, Sagi Snir a,⇑

aDepartment of Ecology and Evolutionary Biology, University of Haifa, Israel
bIsraeli Bird Ringing Center, Society for the Protection of Nature in Israel, Israel

corresponding author.

ARTICLE INFO

Article history:
Received 21 May 2016
Revised 16 October 2016
Accepted 25 October 2016
Available online 3 November 2016

Keywords:
Phylogenetics
Maximum parsimony
Character compatibility
Perfect phylogeny
Statistical significance
Supertree
Optimal convex recoloring cost

ABSTRACT

With the availability of enormous quantities of genetic data it has become common to construct very accurate trees describing the evolutionary history of the species under study, as well as every single gene of these species. These trees allow us to examine the evolutionary compliance of given markers (characters). A marker compliant with the history of the species investigated, has undergone mutations along the species tree branches, such that every subtree of that tree exhibits a different state. Convex recoloring (CR) uses combinatorial representation to measure the adequacy of a taxonomic classifier to a given tree. Despite its biological origins, research on CR has been almost exclusively dedicated to mathematical properties of the problem, or variants of it with little, if any, relationship to taxonomy. In this work we return to the origins of CR. We put CR in a statistical framework and introduce and learn the notion of the statistical significance of a character. We apply this measure to two data sets - Passerine birds and prokaryotes, and four examples. These examples demonstrate various applications of CR, from evolutionary relatedness, through lateral evolution, to supertree construction. The above study was done with a new software that we provide, containing algorithmic improvement with a graphical output of an optimally recolored tree.

Availability: A code implementing the features and a README is available at http://research.haifa.ac.il/ssagi/software/convexrecoloring.zip.

© 2016 Elsevier Inc. All rights reserved.

1. Introduction

The practice of constructing a tree depicting the evolutionary history of a set of organisms is nowadays common to almost every phylogenomic study - an area combining genomic data and techniques for the study of evolution (Eisen and Fraser, 2003; Delsuc et al., 2005). In particular, the deluge of the molecular data accumulating constantly, allows us to gauge the accuracy of the constructed trees. A character, genetic or morphological, classifies the species set into several character classes. If we consider each class as a different color, then every species is colored by the state of the character it possesses, and the given character induces a coloring over the tree leaves. We say that the coloring is convex on the given tree if every color class induces a clade or a subtree and these subtrees do not overlap (Moran and Snir, 2008) (or equivalently, do not intersect). Convexity is a desirable and natural property in classification. When a character is convex on a tree, it is denoted as homoplasy free meaning it displays no reversals or convergence (Zhang and Kumar, 1997). The well-founded and widespread phylogenetic approach maximum parsimony (Fitch, 1971) seeks a tree with minimal changes on its edges, summed over all input characters. A minimum can be obtained when a perfect phylogeny exists in which case each input character is homoplasy-free on that phylogeny (Fernandez-Baca, 2001). Such a tree not necessarily exists, and even finding it is computationally intractable (Bodlaender et al., 1992). In the above setting, the characters are given and assumed to be reliable, and a plausible tree is sought. In other settings, the tree is also given, along with the characters, but one or more characters are not convex on that tree. In this case, we may question about the reliability of that tree.

Alternatively, in a setting where the tree provides enough confidence, the question shifts to the reliability of the input characters. Moreover, we may wonder if the character under examination has evolutionary traces or is influenced by other factors such as environment or simply randomness. In both cases, questioning the tree while assuming character reliability or questioning the character evolutionary meaningfulness, we look for the recoloring distance that counts the minimum number of tree nodes we need to recolor in order to arrive at convexity. This value indicates the level of disagreement between the tree and the coloring. The notion of the recoloring distance was coined in Moran and Snir (2008) where the problem, convex recoloring (CR), was defined and studied for several types of trees and input colorings. Despite its biological
origin, due to its mathematical cleanliness, mainly combinatorial/algorithmic aspects of the problem and its derivatives, that have little if at all biological relevance, were studied. These include extensions to certain graph types rather than a tree, specific input colorings, constrained recoloring schemes, and alike (see e.g. Kanj and Kratsch, 2009; Kammer and Tholey, 2012; Campêlo et al., 2013 and references therein, but see also Matsen, 2015 for a classification oriented study).

In this work we bring back the high level theory of CR down to the biological ground in several aspects. For a taxonomist, it would be desirable to determine quantitatively and statistically, the relevance of a character (i.e. any classification) to the tree at hand. The recoloring distance is an absolute, context-less, number. We therefore introduce the notion of a coloring significance, indicating how likely we are to see, a coloring of this distance or less, by chance on the given tree. In the Results section we demonstrate the use of the coloring significance measure by applying CR to several examples. First, in order to obtain an intuition regarding this measure we show a simulation study. The results reveal that the recoloring distance is more structured than expected. Next, using two data sets, we demonstrate the various uses of CR as an evolutionary marker. The first data set is over eighty Passerine birds, and the second is over a hundred prokaryotes, with few colorings (characters) for each data set. The results obtained concern not only questions of phylogeny/character reliability, but also intensity of non-tree-like activity in prokaryotes and the power of supertree methods.

Importantly, we provide a software that implements the features we describe in this work. To this respect, in the Method section we describe an algorithmic improvement to the algorithm presented in Moran and Snir (2008). The improvement is achieved by reducing the average number of colors checked at a node. We do not give an asymptotic analysis for this improvement but do provide rigorous proof for its correctness. We are aware that since the appearance of the algorithm of Moran and Snir (2008), there have been further improvements (e.g. Bar-Yehuda et al., 2008) to that first algorithm, and there might be other algorithms with better complexity than the one presented here. However a basic property of this algorithm, which to the best of our knowledge was not used before, is a local view that allows a dynamic calculation of the set of candidate colors of each tree node. Accordingly, we believe that the algorithmic improvements provided here, accompanied with more fundamental theoretical improvements to CR, viewing it as a fixed parameter tractable problem (Bodlaender et al., 2011), will allow application of CR to data sets of orders of thousands of species and hundreds of colors.

2. Results

We now show four examples for the application of convex recoloring to synthetic and real data. The first one is a simple example based on random colorings of a binary tree, demonstrating the distribution of optimal convex recoloring cost in one simple case. The other three are applications to real biological examples of colored trees where the colorings represent a different classification each time. In each case we compute the optimal recoloring and its associated p-value, signifying how much the given coloring complies with the evolutionary history of the given species set (that is also given as input, and is represented by the tree topology).

2.1. Example 1: Statistical distribution of the recoloring distance

Our first example shows how the recoloring distance distributes for a given tree size and number of colors. We constructed a set of random binary trees with 50 leaves. Next, we randomly and uniformly colored the tree leaves by 4 colors (no uncoldored leafs, all internal nodes are uncoldored). This is simply done by choosing for every leaf each color with probability 1/4. Therefore, the trees obtained are different in topology and also by the proportions between color sets. For each of these trees a convex recoloring was calculated. The distribution of cost of recoloring is presented in Fig. 1(a). We note that a naive upper bound to the expected value of this statistic, is the value of 3n/4 where n is the number of leaves. This is achieved by recoloring all the leaves with the most common color. As this must have at least n/4, the bound is trivially obtained. However, as we see in the figure, a much smaller value (from n/2 to 3n/5) is usually obtained, signifying existence of a more profound structure in this question than that naive bound. Notwithstanding, a more precise bound is not trivial to obtain and is beyond the scope of this work. Distribution of colors frequencies on the resulted convex trees is presented in Fig. 1(b). The results are divided into three cases (three bar chart in the figure) representing cases in which the most common color had (i) below 25 members (Blue bars), (ii) between 25 and 28 members (Brown bars), and (iii) above 28 members (Green bars). As shown, this difference in the prevalence of the most common color, affects minimally over the distribution of the final colors, where the most common color colors around 70% of the leaves. We note that as there are many (possibly even exponentially many) optimal recolorings, this distribution might be biased according to the strategy employed by the algorithm. One may observe that in a tree, every color is preserved at least by a single leaf as this does not violate convexity of the tree. This observation is explained by the three short bars in the right of Fig. 1(b).

2.2. Example 2: Birds moult strategies

In this example compatibility of adult/juvenile moult strategy of birds with their evolutionary history was examined. We took a tree over 80 bird taxa representing 29 of the 46 Passerine families (Treplin et al., 2008). The leaves of this phylogeny were classified by their main moult strategies in adult/juvenile life stages as described in Jenni and Winkler (1994), Cramp et al. (1993), and Ginn and Melville (1983). Such characterization was made only for 43 of these genus and species and was expressed by one, two or even three of three observed moult strategy types: “Summer complete/summer partial”, “Summer complete/summer complete”, and “Winter complete/winter complete”. Such characterization induces the following coloring of phylogenetic tree’s leaves: leaves corresponding to non-characterized species and species characterized by more than one strategy type - uncolored; leaves corresponding to species characterized by only one moult strategy type are colored by Blue, Red and Green (26, 7 and 4 leafs respectively). Based on our program we found that this coloring is not convex: $P_{opt} = 8$, $p$-value = 0.26 (see Fig. 2). Excluding the green color results in non-convex coloring with $P_{opt} = 5$, $p$-value = 0.46. Unifying colors Red and Blue (in the initial coloring) results in $P_{opt} = 3$, $p$-value = 1.0. The latter means the following. After unifications of Red and Blue, we are left with two colors - Red/Blue and Green - where the Green comprises of only 4 members, that are dispersed. A cost of 3 means that in order to arrive at convexity we must uncolor all but one of the Green leaves. As shown in previous section (Section 2.1), any tree recoloring retains at least one leaf of any color class intact. The latter implies that this is not only the minimum cost possible, rather also the maximum cost for the given configuration of 4 Green leaves. Moreover, since any other, random or not, input coloring with 4 Green leaves cannot achieve a cost higher than that (i.e. a cost greater than 3), all colorings attain this (3) or smaller cost, explaining the p-value of 1 of that result. The
biological interpretation of the results above, is that adult/juvenile moult strategies of birds are not evolutionary compliant. It can be explained by the hypothesis that similar adult/juvenile moult strategies were formed independently for different bird species and/or changed in different directions during the process of evolution (e.g., caused by changing of climatic niches).

2.3. Example 3: Birds migration strategies

Birds genus and species from Example 2 above were also classified by subdivision into three overlapping classes based on main migration strategy: “residents”, “short-“ and “long-distance migrants” (Hall and Tullberg, 2004; Cramp et al., 1993). In total, 41 out of 80 genus were classified. Such a classification induces the following coloring on the tree leaves: 15 “pure” residents (Red), 8 short-distance migrants (Blue), 6 long-distance migrants (Green) and 12 having various strategies. We found that such a coloring is also non convex: Removing genus with various strategies yields $P_{opt} = 9$, $p$-value $= 0.2$ (see Fig. 3). Combining Blue (short-distance migrants) and Green (long-distance migrants) into “migrants” gives a bi-colored tree that is non convex with $P_{opt} = 11$, $p$-value $= 0.18$. Removal of the Blue color (nodes) results in $P_{opt} = 4$, $p$-value $= 0.51$. Finally, combining Blue to Red results in $P_{opt} = 6$, $p$-value $= 1.0$. The above means that migration distance is, similarly to moult strategy, also not evolutionary compliant (presumably like many of ecological/geographical/behavior characters). Such estimation can be explained by the hypothesis that ability and preference to migrate on long distance changed in both direction during the process of evolution and was caused by multiple internal and environmental traits.

2.4. Example 4: Evolutionary classes among prokaryotes

In this part, we study convexity among prokaryotes. Our species set is composed of 41 archaeal and 59 bacterial genomes, representing the forest of life (Puigbó et al., 2009), and that were studied in Puigbó et al. (2009). The characters used for colorings represent three different classifications: (i) domain based (2 colors, archaeal/bacterial), (ii) phylum based (24 colors), and (iii) order based (57 colors). The underlying approach here is different from the examples above as these characters are considered accurate and largely representing the main trend of evolution of the given species set. Under this setting, the given tree is under scrutiny. Here, trees represent gene specific histories, dubbed gene trees. These histories are substantially different as many genes are subjected to the phenomena of horizontal gene transfer (HGT), the passage of genetic material between organisms by means other than lineal descent (Doolittle, 1999; Ochman et al., 2000). Evolution in light of HGT tangles the traditional universal Tree of Life, turning it into a network of relationships (Gogarten et al., 2002; Zhaxybayeva et al., 2004; Gogarten and Townsend, 2005; Bapteste et al., 2005).

To put the above discussion in the context of color convexity, we did the following. First we considered a tree representing the evolution of the Isoleucyl-tRNA synthetase (IleS, COG0060) gene, henceforth the IleS-tree, that is present in all 100 considered prokaryotes. The IleS-tree and the corresponding colorings of (domain-, phylum-, and order-based) are depicted in Fig. 4. Leaf coloration follows order classification. Our results show that none of the colorings is convex on the IleS-tree.

In order to delve deeper into the meaning of this result, we analyzed each category (coloring) separately. Starting with the domain level, the tree from Fig. 4 can be perceived as an unrooted quartet tree (Avni et al., 2015) over four large clades (subtrees) pertaining almost exclusively to either bacteria and archaea. If we ignore the outliers and color these clades as depicted in the figure: archaea - Red, and bacteria - Green, we see a quartet colored Red, Green|Red, Green. Obviously, this coloring is not convex, suggesting a very early HGT between archaea and bacteria of the IleS gene.

At the phylum level, there can be seen several violations to convexity that can be evidenced by the presence of members of a single phyla populating two of the four domain clades indicated above. This by definition is a violation to convexity as we require all members of a phyla to be present in a single domain clade. Specifically, in the figure (Fig. 4), we point at three members of the Proteobacteria-Alpha phylum (index 22, green arrows), present in the two bacteria clades.

Finally, there are many violations of convexity at the level of orders. One such violations that we also mention in the figure is of orders with indices 20 (Desulfurococcales, indicated by blue arrows) and 55 (Thermoproteales, indicated by red arrows). There can be found several quartets over these orders composed of a pair from the index 20 order and another pair from the index 55 order that exhibit a quartet colored 20,55/20, 55 arrangement. It can be shown that such an arrangement requires a recoloring of at least one leaf (see Fig. 4).

Despite the deep discordance between individual gene histories, the belief in an underlying, vertical trend of evolution even among prokaryotes, yields a major challenge of finding this tree. Normally, this underlying phylogeny is inferred by constructing gene trees for genes thought to be immune to HGT, typically
Fig. 2. Adult/juvenile moult strategy of birds. Leaf coloring is as follows: “Summer complete/summer complete” – Blue, “Winter complete/winter complete” – Red, “Summer and winter complete/winter complete” – Green; non-characterized species and species characterized by more than one strategy type – uncolored (black). Optimal (convex) recoloring is schematically shown by lines of corresponding colors. Note that only one green and two red colored leaves remained. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
ribosomal RNA genes. Nevertheless, even such genes are subjected to HGT, obfuscating the central trend of evolutionary relationships (Berkum et al., 2003; Dewhirst et al., 2005; Schouls et al., 2003; Yap et al., 1999). Therefore, it was suggested to construct the underlying species tree by a two-stage approach as follows: First, gene trees such as the IleS-tree above, are constructed separately...
for a multitude of genes. These trees do not necessarily span the entire taxa set rather overlap at subsets of it. Subsequently these trees are amalgamated together to produce a big tree over the complete taxa set. This approach is denoted the supertree construction and the resulted tree is denoted a supertree (Bininda-Emonds et al., 2002; Creevey and McInerney, 2005).

In light of the above, the task we pursue here is how much the supertree “corrects” the non convexity of individual gene trees. In Puigbò et al. (2010), a set of 6901 orthologous gene families (COGS Tatusov et al., 2001) was selected and for each such family, its gene tree was reconstructed. From this set, a subset of around a hundred fairly conserved, ubiquitous genes, denoted nearly universal trees (or NUTs), were taken. A tree spanning the entire taxa set was constructed by a supertree method, based on the NUTs trees. We denote it as the NUTs-tree. We wanted to measure the convexity of the NUTs-tree with respect to each of our three colorings. Applying our program we found that all are convex on this tree. For illustration, the tree, leaf-colored according to phylum, is shown in Fig. 5.

To summarize this part, we start with the IleS-tree. We note that the fact that all the three colorings were found highly insignificant (high $p$-values) suggests an intensive HGT activity. Nevertheless, in the case of HGT, one caveat should be raised. HGT operates in scale of subtrees while recoloring counts single nodes and therefore a

![Fig. 4. A tree over 100 prokaryotes based on genes Isoleucyl-tRNA synthetase (IleS, COG0060) from Puigbò et al. (2009). For convenience, organism names are appended by three numbers separated by an underline (representing domain, phylum, and order indices respectively; order indices 1 and 57 correspond to organisms with questionable order). Leaf coloration follows the coloring defined by order. It can be seen that the three colorings- domain, phylum, and order - are not convex on the IleS tree. On the domain level, one can see two pairs of large clades (subtrees), a pair for each domain, corresponding to domains 1 (Archaea, red lines) and 2 (Bacteria, green lines), intertwined along the tree, yielding non convexity of the domain coloring. At the phylum level, phylum 22 (Proteobacteria-Alpha, pointed by green arrows in the figure) was found in both bacteria clades and hence yielding non convexity also at the level of phylum. We remark that one can find few additional such examples for bad classified phylums according to this gene tree. The archaea domain was also found non convex by the order coloring: carriers of colors correspond to order 20 (Desulfurococcales, pointed by blue arrows) and 55 (Thermoproteales, pointed by red arrows) overlap. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)](https://www.nature.com/articles/s41598-017-09903-0)
recoloring distance cannot, on its face value, be indicative to intensity of HGT. Our domain-level coloring illustrates that. Recall we had a tree over four large clades, two colored with Green and two with Red. In order to turn this coloring to convex on that tree, a whole clade needs to be recolored. In contrast, one Subtree Pruning and Regrafting (SPR) operation, that cuts an entire clade from its current location and joins it in another, would have fixed this situation, yielding a convex tree. However, this SPR move would have modified the tree topology - an operation that stands in contradiction to the CR philosophy that keeps the tree topology intact.

Therefore, while intensity of HGT is normally measured by the SPR-distance to the species tree (Hein, 1990), it is important to mention that finding such a distance is computationally intractable (NP-hard) (Bordewich and Semple, 2005) (but exponential in the number of SPR events), finding the recoloring distance may provide some intuition and is exponential only in the number of colors.

The second example with prokaryotic data, dealt with the power of the supertree approach and how this is related to CR. We have shown that the supertree approach can "correct" all coloring violations as exhibited by IleS-tree. We note that convexity.
with respect to these classifications, is not the only criterion. Therefore, we can frequently find trees that are not convex with respect to this classification yet provide other, insightful relationships.

3. Conclusions

In this work we studied convex recoloring (CR) and focused on relevant biological aspects of it. Since its introduction in 2005 (Moran and Snir, 2005; Moran and Snir, 2005), CR was almost entirely studied in the context of theoretical computer science while the biological relevance of it was neglected. We believe that this is the prime importance of the work presented here. Specifically, we used CR as a marker for character compliance with organismal evolution, by fitting it to the tree nodes and measuring compatibility. We augmented the parameterless value of the recoloring distance with a statistical framework that provides the (statistical) significance of the given input coloring in terms of a p-value, allowing determination of the evolutionary relatedness of the character under study.

On a more technical level, we provided algorithmic improvements to the basic algorithm for CR introduced in Moran and Snir (2008). The improvement is achieved by reducing the set of possible recolorings and considering a more local, instead of a global, view of the problem. In general, when the input coloring is near random and has a big recoloring distance, this improvement appears to be of little benefit over the asymptotic bound. Nevertheless this improvement is more pronounced in the case of a coloring close to convexity. It appears that our heuristic bears some similarity to the principles implemented in Bar-Yehuda et al. (2008). While we do not have theoretical asymptotic analysis for this improvement, it was experimentally demonstrated in our simulations and real data examples.

Importantly, we also provide software implementation for the algorithm, containing the features discussed above and providing an output that can be used conveniently in tree viewing software as demonstrated in our examples. To the best of our knowledge, no such software exists.

In the experimental realm, we applied our software to four examples, two from Ornithology and two from Microbiology. The examples from Ornithology addressed the topic of character compatibility, as demonstrated in our examples. To the best of our knowledge, no such software exists.

In the experimental realm, we applied our software to four examples, two from Ornithology and two from Microbiology. The examples from Ornithology addressed the topic of character compatibility, as demonstrated in our examples. To the best of our knowledge, no such software exists.

In the experimental realm, we applied our software to four examples, two from Ornithology and two from Microbiology. The examples from Ornithology addressed the topic of character compatibility, as demonstrated in our examples. To the best of our knowledge, no such software exists.

In the experimental realm, we applied our software to four examples, two from Ornithology and two from Microbiology. The examples from Ornithology addressed the topic of character compatibility, as demonstrated in our examples. To the best of our knowledge, no such software exists.

In the experimental realm, we applied our software to four examples, two from Ornithology and two from Microbiology. The examples from Ornithology addressed the topic of character compatibility, as demonstrated in our examples. To the best of our knowledge, no such software exists.

In the experimental realm, we applied our software to four examples, two from Ornithology and two from Microbiology. The examples from Ornithology addressed the topic of character compatibility, as demonstrated in our examples. To the best of our knowledge, no such software exists.

In the experimental realm, we applied our software to four examples, two from Ornithology and two from Microbiology. The examples from Ornithology addressed the topic of character compatibility, as demonstrated in our examples. To the best of our knowledge, no such software exists.

In the experimental realm, we applied our software to four examples, two from Ornithology and two from Microbiology. The examples from Ornithology addressed the topic of character compatibility, as demonstrated in our examples. To the best of our knowledge, no such software exists.

In the experimental realm, we applied our software to four examples, two from Ornithology and two from Microbiology. The examples from Ornithology addressed the topic of character compatibility, as demonstrated in our examples. To the best of our knowledge, no such software exists.

In the experimental realm, we applied our software to four examples, two from Ornithology and two from Microbiology. The examples from Ornithology addressed the topic of character compatibility, as demonstrated in our examples. To the best of our knowledge, no such software exists.

In the experimental realm, we applied our software to four examples, two from Ornithology and two from Microbiology. The examples from Ornithology addressed the topic of character compatibility, as demonstrated in our examples. To the best of our knowledge, no such software exists.

In the experimental realm, we applied our software to four examples, two from Ornithology and two from Microbiology. The examples from Ornithology addressed the topic of character compatibility, as demonstrated in our examples. To the best of our knowledge, no such software exists.

In the experimental realm, we applied our software to four examples, two from Ornithology and two from Microbiology. The examples from Ornithology addressed the topic of character compatibility, as demonstrated in our examples. To the best of our knowledge, no such software exists.

In the experimental realm, we applied our software to four examples, two from Ornithology and two from Microbiology. The examples from Ornithology addressed the topic of character compatibility, as demonstrated in our examples. To the best of our knowledge, no such software exists.

In the experimental realm, we applied our software to four examples, two from Ornithology and two from Microbiology. The examples from Ornithology addressed the topic of character compatibility, as demonstrated in our examples. To the best of our knowledge, no such software exists.

In the experimental realm, we applied our software to four examples, two from Ornithology and two from Microbiology. The examples from Ornithology addressed the topic of character compatibility, as demonstrated in our examples. To the best of our knowledge, no such software exists.

In the experimental realm, we applied our software to four examples, two from Ornithology and two from Microbiology. The examples from Ornithology addressed the topic of character compatibility, as demonstrated in our examples. To the best of our knowledge, no such software exists.

In the experimental realm, we applied our software to four examples, two from Ornithology and two from Microbiology. The examples from Ornithology addressed the topic of character compatibility, as demonstrated in our examples. To the best of our knowledge, no such software exists.
color can be significantly smaller than the entire set of bad colors (as there can be a situation where a certain bad color is candidate only for a small subset of nodes colored by bad colors, see example presented in Fig. 8a). An additional reduction of the set of candidate colors for a node can be achieved by dynamically recalculating the set of candidate colors for subtrees, while considering previous decisions made for other nodes affecting the node in question (such decisions can break down chains of overlapping carriers of candidate colors that can result in splitting a cluster of candidate colors into smaller parts or even singletons, see Fig. 8b for example). In the Appendix we provide rigorous arguments why the restriction to candidate colors indeed guarantees optimal convex recoloring. Our algorithm follows along the lines induced by Lemma 4.8 of Moran and Snir (2008) however instead of considering the fixed set of bad colors, we use the smaller sets of candidate colors, that are calculated dynamically during the run of the algorithm.

4.3. Coloring significance - estimation of p-value

An important feature of CR that was not explored so far, is the significance of a given input (non convex) coloring. A relatively low cost \( P_{opt}(C) \) of some coloring \( C \) is not necessarily a proof for the goodness of the input tree coloring \( C \). For example, it might be that this optimal cost is attained by many random recolorings. Hence, the significance of a given coloring gives an estimate on how likely we are to find by random another coloring with the same cost. The biological meaning of this value can be interpreted as follows. Assume we believe in the given tree topology (this is also the underlying assumption in CR in general, as opposed to perfect phylogeny, where the tree is built based on the given set of discrete characters). We also believe in the coloring on the tree (i.e., the color assignment to the tree nodes). Then this significance value can be interpreted as a means to measure statistically the compliance of this character with the evolutionary history of the taxa set at hand (that is depicted by the tree).

Therefore in addition to the \( P_{opt}(C) \) value we provide an estimation for the quality of \( C \) by the probability to obtain \( P_{opt}(C) \leq P_{opt}(C') \) for a “random” coloring \( C' \). As analytic calculation of this measure appears to be hard and presumably computationally intractable, the straightforward way to proceed is via simulations (a method known as permutation test or bootstrap Wasserman, 2004). Hence, to estimate this probability (that can be dubbed as a p-value) we calculate the frequency of events \( R = (P_{opt}(C) \leq P_{opt}(C')) \) out of \( N \) random colorings of the tree (e.g., \( N = 10,000 \)). In order to maintain the initial properties of the input coloring \( C \), we preserve the proportions between colors of the original coloring. Hence, each random coloring of the tree is simulated by a reshuffling of the input colors of \( C \) between the nodes set. As the initial coloring \( C \) can be considered as a realization of random tree coloring, we get: \( p\text{-value} = (N_R + 1)/(N + 1) \). Uncolored nodes are not affected by the reshuffling similarly as they do not affect the cost function. The software implementation associated with this article provides this value along with the absolute cost of the optimal convex recoloring for the given coloring \( C \).

4.4. Implementation

The algorithm is implemented in Python and receives as input a colored tree in either Newick or NEXUS formats. Colors are given to nodes either as part of their names by some convention, or by a separate table. It is also possible to indicate colors to internal nodes, and these colors are interpreted by the program as part of the input. The output of the program is an optimally recolored tree (one of the many possible, saved both in Newick or NEXUS formats), the list of recoloring of the nodes, the cost of the optimal convex recoloring, and p-value. This output can be used by several tree viewer softwares (e.g., FigTree Rambaut, 2010) as is demonstrated in our Results section below. Our experiments showed that the program was able to find optimal convex recoloring for trees with 200 leaves, randomly colored by up to 60 colors, in a few seconds. Recall that by Moran and Snir (2008), the algorithm runs in time that is linear in the number of vertices and even in the number of good colors and exponential (that is, fixed parameter tractable Downey and Fellows, 1999) in the number of bad colors. Consequently, it can also handle larger trees (e.g. of 1000 leaves) however with relatively small number of “bad” colors (e.g., 20).

More implementation details can be found in the Appendix.

Acknowledgements

We wish to acknowledgements Lana Martin for valuable edit on the manuscript.
Appendix A. Rigorous proofs for optimal recoloring via candidate colors

Here we present a formal proof that there exists an optimal convex recoloring (one of all possible) that rewrite (recolors) nodes only by their cores candidate colors (similar to Lemma 4.7 of Moran and Snir, 2008). In fact, in our algorithm of optimal convex recoloring searching candidate colors are calculated dynamically with taking into account restrictions caused by current decision not use some colors in recoloring of subtree. Any partial convex coloring can be naturally extended in such a way that there will remained no uncolored nodes situated in the carrier of some used color (let \( v \in \text{carrier}(d) \); we can define \( C(v) := d \); this definition is correct because initially coloring \( C \) was convex, carriers of colors were not changed, hence \( C \) remained to be convex). We consider only recoloring with zero cost of coloring for uncolored nodes, hence, for the simplicity, all convex coloring and recoloring will be considered as already after all these extensions (i.e., if some node is uncolored in considered convex coloring then it is not belonging to carrier of any used color). Pair of colors \((d_1, d_2)\) considered as neighbor in coloring \( C \) (not necessary convex) if there exist vertices \( v_1 \) and \( v_2 \) such that \( C(v_1) = d_1 \), \( C(v_2) = d_2 \), and \( u \) connected with \( v \) by single edge or by path (going on edges of tree) visiting only uncolored (in \( C \)) nodes.

Claim A. 1. Let \( C \) be input coloring of tree \( T \). There exist a convex recoloring \( C' \) of minimal possible cost such that for each \( d \in C(T) \) there exists node \( v \) such that \( C(v) = C'(v) = d \).

Proof. Situation with convex input coloring \( C \) (e.g., no colored nodes) is trivial. Let \( C \) a convex recoloring of minimal possible cost with minimal possible number of colors (it always exists because the set of nodes is finite). Let there exists color \( d \in C(T) \) such that for any \( v \in T \) is uncolored in \( C \) or \( C(v) \neq d \). Let \( d' \) be one of neighbor colors of \( d \) in \( C \). Then recoloring \( C' \) coincidental to \( C \) out of \( C^{-1}(d) \) and coloring all nodes from \( C^{-1}(d) \) by \( d' \) is convex (because \( C' \) is convex) and use less colors (does not use \( d \)). This contradicts to definition of \( C' \).

Claim A. 2. Let \( d \) be a candidate color for nodes \( u \) and \( v \) in respect to input coloring \( C \). Then \( d \) is a candidate for all nodes in the path (going on edges of tree, without returns) from \( u \) to \( v \).

Proof. Color \( d \) is candidate for \( u \) and \( v \), hence there exist colors \( d^{(0)}_u, \ldots, d^{(u)}_u \) and \( d^{(0)}_v, \ldots, d^{(v)}_v \) such that \( u \in \text{carrier}(C^{-1}(d^{(u)}_u)) \), \( v \in \text{carrier}(C^{-1}(d^{(v)}_v)) \), \( d^{(u)}_u = d^{(v)}_v = d \), \( \text{carrier}(C^{-1}(d^{(u)}_u)) \cap \text{carrier}(C^{-1}(d^{(v)}_v)) \neq \emptyset \) for all \( i = 1, \ldots, n^{(u)} \) and \( j = 1, \ldots, n^{(v)} \). Hence, there exists path from \( u \) to \( v \) (going on edges of tree) such that color \( d \) is a candidate for all visited nodes. A path from \( u \) to \( v \) going on edges of tree without returns is unique, hence color \( d \) as candidate for all its nodes.

Recoloring any colored \( n \) nodes to any individually-other colors (i.e., \( C'(v) \neq C(v) \)), but it can be that \( C'(v) = C(u) \) costs \( n \). This enables to make a following observation:

Observation A. 3. Let \( T' \) be a subtree of tree \( T \). Let \( C' \) be a recoloring of \( T \) in respect to input coloring \( C \). Let all nodes in \( T' \) are recolored by \( C' \) only by its non-candidate colors. Then any recoloring of \( T' \) coincidental to \( C' \) in \( T \setminus T' \) has cost not higher than \( C' \).

Claim A. 4. Let \( C' \) be a convex recoloring (after all extensions, see above) of \( T \) in respect to input coloring \( C \). Assume nodes \( u \) and \( v \) are connected by single edge or by a path going on tree edges and visiting only nodes uncolored in \( C' \) (and hence not in carrier of any color in \( C' \)). Assume also \( C'(u) \) is candidate for nodes \( u \) and \( v \), but \( C' \) is not candidate for \( v \). Then there exists a convex recoloring \( C' \) with \( \text{cost}(C') \leq \text{cost}(C') \) coloring more nodes by its candidate colors than \( C' \).

Proof. Let \( T_0^{(C')} \) be the minimal subtree of \( T \) containing all nodes colored in \( C' \) by color \( C'(v) \), i.e., \( T_0^{(C')} = \text{carrier}(C^{-1}(C'(v))) \). Denote by \( T_0^{(C')} \) the maximal subtree of \( T^{(C')} \) containing node \( v \) and not containing nodes such that \( C'(v) \) is its candidate color. Based on Claim 2, set \( T_0^{(C')} \setminus T_0^{(C')} \) (it can be empty) is connected (see example presented in Fig. 9). \( C' \) is convex, hence all nodes of \( T_0^{(C')} \) in \( C' \) are colored by \( C'(v) \). Therefore, recoloring \( C' \) coincidental with \( C \) in \( T \setminus T_0^{(C')} \) and coloring nodes of \( T_0^{(C')} \) by color \( C(u) \) is convex. Now \( C'(v) \) is candidate (in \( C \)) for \( v \). Color \( C'(v) \) was not candidate (in \( C \)) for all nodes of \( T_0^{(C')} \), hence all nodes that were colored by its candidate color in \( C' \) remained colored by the same candidate color in \( C' \). Following to Observation 1, the cost of \( C' \) is not higher than the cost of \( C' \).

Claim A. 5. There exists recoloring of the minimum possible cost such that all nodes are recolored by its candidate colors or remained uncolored.

Proof. Let \( C' \) be a convex recoloring (with all possible extensions, see above) of the minimum possible cost such that for each \( d \in C(T) \) there exist node \( v \) such that \( C(v) = C'(v) = d \) (see Claim 1), recoloring the most possible number of nodes by its candidate color. Now we will show that if some nodes are colored in \( C' \) by
non-candidate color then these nodes were uncolored in \( C \) (hence \( C \) coincidental with \( C \) on all other nodes and not coloring these nodes is like searched for) or there exists convex recoloring \( C \) having cost (relatively to input coloring \( C \)) not higher than \( C \) and coloring more nodes by its candidate colors. This contradicts to definition of \( C \) and enough for the proof. Assume node \( v \) is colored in \( C \) and \( C \) such that \( C(v) \) is not candidate for \( v \) (in \( C \)).

1. Let color \( C(v) \) is not presented in \( C \). Using designations from the proof of Claim 4, convex recoloring \( C \) coincidental with \( C \) in \( T[V(C) \setminus T''_C] \) and coloring nodes of \( T''_C \) by color \( C(v) \) is convex and has cost lower than \( C \) (because \( C(C) = C(v) \neq C(v) \)) that contradicts to definition of \( C \).

2. Assume color \( C(v) \) is already presented in \( C \) (hence the recoloring \( C \) from (1) can be non convex). Let \( u \) is such that \( C(u) = C(v) \) (it exists by definition of \( C \)). Color \( C(v) \) is a candidate for \( u \) and \( v \), hence, following Claim 2, color \( C(v) \) is candidate for all nodes in the path from \( u \) to \( v \) (going on edges of tree without repeats). Let node \( u' \) be the first node in this path (starting from \( u \)) colored in \( C \) by non-candidate color \( d \). Let node \( u' \) be the last node in the path from \( u \) to \( v' \) colored by its candidate color (e.g., \( v' \) can coincident with \( u \)). Node \( u' \) belongs to overlap of \( \text{carrier}(C^{-1}(C(u'))) \) and \( \text{carrier}(C'(C(u'))) \geq \text{carrier}(u,v) \), hence color \( C(u) \) is candidate for \( u \) and \( v \), and, based on Claim 2, it is candidate for \( v' \). Hence, based on Claim 4 there exist a convex recoloring \( C \) having cost not higher than \( C \) and coloring more nodes by candidate colors. That contradicts our definition of \( C \).

We now consider a special case of a convex recoloring. A partial convex recoloring \( C \) is conservative relative to initial coloring \( C \) if it satisfies the following: (1) only vertices uncolored by \( C \) can be colored by \( C \); (2) A node uncolored by \( C \) can be colored in \( C \) only by a bad color of \( C \) or remain uncolored; (3) A vertex can change its color only to bad color of \( C \); and (4) For every color \( d \) used in coloring \( C \), set \( C^{-1}(d) \) is connected. In Moran and Snir (2008) it is shown that an optimal conservative recoloring is also a general optimal convex recoloring. By our next claim, optimality holds even if we replace “bad color” (in the definition of Moran and Snir (2008)) by “candidate color” in the definition of conservative recoloring (see above). We refer to such a conservative recoloring as candidate conservative recolorings.

Claim A.6. An optimal candidate conservative recoloring is an optimal convex recoloring in general.

Proof. Let \( C \) be an optimal convex recoloring from Claim 5. Using all possible extensions of \( C \) on nodes uncolored in \( C \) one can obtain an optimal convex recoloring satisfying conditions of candidate conservative convex recoloring. □

Observation A.7. Let \( T \) be a subtree of tree \( T \). Let \( C \) be an optimal recoloring for subtree \( T \) restricted to some set of colors. In this case we can use Claim 6 with restriction to a set of candidate colors such that this set is used by uncoloring nodes of excluded colors (see Fig. 10).

Observation A.8. Using the definition of good colors in the sense of Moran and Snir (2008), i.e., color \( d \) is good in partial coloring \( C \) if \( \text{carrier}(C^{-1}(d)) \) contains no nodes with other colors and no uncolored nodes from carriers of other colors. Then there exists an optimal candidate conservative recoloring with no recoloring of good colors by other good colors.

Using Observation 8 we can improve the algorithm searching for optimal convex recoloring.

Algorithm for searching for optimal convex recoloring: For a rooted tree \( T \), we denote by \( T' \) the subtree rooted at vertex \( v \). We designate by \( P_{t}(v,d,D) \) the minimal cost of a candidate conservative convex recoloring \( C \) of \( T' \) such that \( C(v) = d \) and \( C \) uses only colors from \( D \) for the descendants of \( v \). For convenience, we use symbol \( \star \) to denote a “color” of an uncolored vertex and assume a cost infinity for uncoloring a colored vertex. Denote \( P_{t}(v,d,D) := \min_{d \in D \setminus \{d\}} P_{t}(v,d,D) \) the minimum cost convex recoloring that uses colors from \( D \). We also designate by \( P_{t}(v,d,D) \) the minimal cost of convex recoloring under which \( v \) is either \( C(v) = d \) or \( C(T_v) \) does not contain \( d \). This means that \( P_{t}(v,d,D) = \min_{d \in D \setminus \{d\}} \min_{D \cup \{\star\} \setminus \{d\}} P_{t}(v,d,D) \). Using Observation 8 we can improve the algorithm searching for optimal convex recoloring.

Assume \( T \) is rooted at some vertex \( v \). Let \( \delta_{uv} \) denote the inverse Cronecker delta, such that \( \delta_{uv} = 0 \) if \( u \neq v \), and \( \delta_{uv} = 1 \) otherwise. Denote by \( C = C_{C}(T) \) the set of all node colors used in coloring \( C \). Then analogously to Lemma 4.8 from Moran and Snir (2008), the cost of a minimal convex recoloring of the entire tree \( T \) can be written as \( P_{t}(v,C) \) and is calculated recursively: \( P_{t}(v,d,D) = \delta_{t(v)} + \min_{d \in D \setminus \{d\}} P_{t}(v,d,D) \) where \( t(v) \) are children of vertex \( v \), \( \delta_{t(v)} = 0 \), \( D_{D} = D \), and \( D \setminus D_{D} = \emptyset \) for \( i \neq j \). The restriction to candidate conservative recolorings rather than all conservative convex recolorings has no asymptotic implication on the running time of the algorithm however it allows us to discard a large fraction of valid color partitions \( \{D_1, \ldots, D_k\} \) and reduce the running time dramatically. The implementation of the candidate is done recursively one child after the other, while the color assignment of \( i \)-th child is checked only after it is guaranteed that for all \( j < i \) the color assignment for the \( j \)-th child satisfy the candidate criterion.

Appendix B. Implementation details

This algorithm is implemented in Python and receives as input a colored tree in either Newick or NEXUS formats. In standard Newick format node names (captions that can include color) can be specified only for leaves. A color is assigned to a leaf by specification immediately after the leaf name: $Color$. It is also possible to set initial coloring of leaves by a separate table in the following format: (LeafInsTree) (NameOfLeafToDisplay) (Color) (see examples in ReadMe.txt file). In NEXUS format it is also possible to assign input
of the nodes, and the cost of the optimal convex recoloring.

References