J. Math. Biol.
DOI 10.1007/s00285-017-1167-9

Mathematical Biology

The combinatorics of discrete time-trees: theory and open problems

Alex Gavryushkin[1] · Chris Whidden[2] · Frederick A. Matsen IV[2]

Received: 29 December 2016 / Revised: 14 July 2017

Abstract A *time-tree* is a rooted phylogenetic tree such that all internal nodes are equipped with absolute divergence dates and all leaf nodes are equipped with sampling dates. Such time-trees have become a central object of study in phylogenetics but little is known about the parameter space of such objects. Here we introduce and study a hierarchy of discrete approximations of the space of time-trees from the graph-theoretic and algorithmic point of view. One of the basic and widely used phylogenetic graphs, the NNI graph, is the roughest approximation and bottom level of our hierarchy. More refined approximations discretize the relative timing of evolutionary divergence and sampling dates. We study basic graph-theoretic questions for these graphs, including the size of neighborhoods, diameter upper and lower bounds, and the problem of finding shortest paths. We settle many of these questions by extending the concept of graph grammars introduced by Sleator, Tarjan, and Thurston to our

AG was supported by Marsden Fund of the Royal Society of New Zealand, Grant UOA1324, through the Centre for Computational Evolution, the University of Auckland, where this work has been carried out. This work was done in part while AG was visiting Fred Hutchinson Cancer Research Center. CW and FM were funded by National Science Foundation Awards 1223057 and 1564137. CW is a Simons Foundation Fellow of the Life Sciences Research Foundation. FM supported in part by a Faculty Scholar grant from the Howard Hughes Medical Institute and the Simons Foundation.

✉ Alex Gavryushkin
alex@gavruskin.com

Chris Whidden
cwhidden@fredhutch.org

Frederick A. Matsen IV
matsen@fredhutch.org

[1] Department of Biosystems Science and Engineering, ETH Zürich, 4058 Basel, Switzerland

[2] Program in Computational Biology, Fred Hutchinson Cancer Research Center, Seattle, WA 98109, USA

Published online: 29 July 2017

graphs. Although time values greatly increase the number of possible trees, we show that 1-neighborhood sizes remain linear, allowing for efficient local exploration and construction of these graphs. We also obtain upper bounds on the r-neighborhood sizes of these graphs, including a smaller bound than was previously known for NNI. Our results open up a number of possible directions for theoretical investigation of graph-theoretic and algorithmic properties of the time-tree graphs. We discuss the directions that are most valuable for phylogenetic applications and give a list of prominent open problems for those applications. In particular, we conjecture that the split theorem applies to shortest paths in time-tree graphs, a property not shared in the general NNI case.

Keywords Phylogenetic inference methods · Phylogenetic time-tree · Nearest neighbor interchange graph

Mathematics Subject Classification 92B05 · 68R10 · 68Q25

1 Introduction

The last 10 years have seen an explosion of methods using sequence data to infer demographic model parameters by sampling phylogenetic trees (Kuhner et al. 1995, 1998, 2000; Beerli and Felsenstein 2001; Kuhner 2006; Drummond et al. 2002, 2005, 2006; Minin et al. 2008). These methods have had an especially significant impact in inferring historical epidemic spreading rates of viruses and other evolutionary parameters of quickly evolving organisms. For example, such methods can be used to infer time to a most recent common ancestor of human HIV group M viruses (Worobey et al. 2008; Baele et al. 2013). Thus the *time-tree*—a rooted phylogenetic tree with all internal nodes equipped with absolute divergence dates—has become an important object of investigation. This interpretation of continuous parameters for time-trees stands in contrast to that for classical phylogenetic trees, in which branch lengths quantify the amount of molecular substitution along a branch. Posterior distributions on both time-trees and classical trees are estimated using Markov chain Monte Carlo (MCMC) (Mau and Newton 1997; Yang and Rannala 1997; Drummond et al. 2002), but with different transition kernels. This paper builds a foundation for a mathematical understanding of time-trees, such as understanding convergence properties of MCMC thereupon.

Although Markov chain Monte Carlo (MCMC) is guaranteed to sample from the true posterior given an infinite run time, it is important to understand mixing properties of the chain, which determine sampling properties for a finite time run. The mixing properties of phylogenetic MCMC have, in the classical unrooted case, been a major area of research from both the theoretical perspective of mixing time bounds (Mossel and Vigoda 2005, 2006; Štefankovič and Vigoda 2011; Spade et al. 2014) and the practical perspective of performance on real data (Beiko et al. 2006; Ronquist et al. 2006; Lakner et al. 2008; Whidden and Matsen IV 2015). This research has focused on mixing over the set of discrete phylogenetic tree graph structures, because mixing over these discrete structures is the primary obstruction to MCMC convergence.

Time-trees use different MCMC transition kernels than do classical phylogenetic trees. The discrete component of MCMC moves between classical phylogenetic trees are typically determined by their discretization. For example, common moves include *subtree prune and regraft* (SPR) moves, which cut a subtree off and reattach it at another location, and the subset of SPR moves called *nearest-neighbor interchange* (NNI) moves. Classically, these are applied without reference to branch lengths. On the other hand, the discrete component of moves between time-trees are defined also in terms of their timing information. For example, Hohna et al. (2008) show that SPR-like moves that reattach subtrees at the same divergence time are more effective than ones that do not. This sort of move cannot be expressed using the type of discretization used thus far in which all continuous information is lost, and so the previous work on MCMC mixing cannot be applied.

However, one can discretize time-trees in a way that does preserve some of the information. For example, by retaining the order of internal nodes backward in time one obtains a so-called *ranked tree* (Semple and Steel 2003). To make a less rough approximation of a time-tree, one can allow the time periods between nodes of the tree to take only finitely many possible values (Åkerborg et al. 2008). This results in an object we call a *discrete time-tree*. The graph built on such trees provides a discretization of the space of time-trees, which we call a *discrete time-tree graph*. By a *graph* on a set of trees here and throughout the paper we mean a graph consisting of trees as vertices, with edges connecting pairs of trees that are identical after a given tree rearrangement operation (Semple and Steel 2003). This graph-theoretic terminology is convenient and has become widespread in recent years (Spade et al. 2014; Whidden and Matsen IV 2015; Gavryushkin and Drummond 2016). A sequence of time-trees sampled using MCMC projects to a collection of movements on a graph in which each vertex is a discrete time-tree and each edge is a discrete version of an MCMC move on time-trees.

Although inferential algorithms have applied MCMC on time-trees for over a decade, and graphs corresponding to discretizations of unrooted tree space have been studied for even longer, we are not aware of any work defining graphs from discretization of time-tree spaces or analyzing random walks thereupon. Such a theory would provide a foundation for understanding the behavior of MCMC algorithms on time-trees, as has been done previously for graphs associated with unrooted phylogenetic trees. Up to now, the only discretization of time-tree space is that of ranked trees (Page 1991; Ford et al. 2009; Lambert and Stadler 2013), and the corresponding graphs have not been studied.

In this paper we initiate the mathematical study of discrete time-tree graphs, obtain basic geometric and graph-theoretic results, and compare these results to those in the classical phylogenetic setting. In particular, we focus on ranked trees, discrete time-trees (as defined above), and the ultrametric versions of those types of trees. We establish size bounds for neighborhoods and diameters in these tree spaces and show how those bounds can be used to develop efficient tree search algorithms such as MCMC. The importance of these basic geometric characteristics in phylogenetics and other areas of evolutionary biology is highlighted in Huber et al. (2011).

2 Technical introduction

Throughout the paper by a (phylogenetic) *tree* we mean a rooted binary tree with designated leaves, that is, an undirected acyclic graph with the following properties: (1) all nodes have degree 1, 2, or 3; (2) there exists exactly one node of degree 2, this node is called the *root* of the tree; (3) all nodes of degree 1 are labeled by distinct identifiers, these nodes are called *leaves* or *taxa* (singular *taxon*). The *parent* of a node x in a tree is the unique node y that is both adjacent to x and closer to the root of the tree than x. Every node of a tree has a parent except for the root, and is called the *child* of that parent.

We fix the number of leaves of the trees and denote this number by n throughout the paper. We also assume that each tree with n leaves uses the same fixed set of n labels to mark the leaves. Hence, we say that two trees are *isomorphic* if they are isomorphic as graphs and the isomorphism maps leaves marked by the same label to each other. We do not distinguish between isomorphic trees, i.e. we *identify* them.

By a *time-tree* (Fig. 1) we mean a phylogenetic tree with an absolute time associated with every node of the tree so that the time strictly increases along every path from a leaf to the root: for internal nodes the time is interpreted as (typically estimated) divergence time and for leaves it is the (typically known) sampling time. By "absolute" we clarify that these are not relative times, but rather actual times that can be put on a calendar. We assume that time progresses backward, from the leaves to the root.

A *discrete time-tree* is a tree such that all its nodes are assigned distinct times from the set of non-negative integers and every node has a smaller time than its parent (Fig. 1). Note that this implies that for every pair of nodes x, y if the shortest path from the root to x passes through y then the time of y is greater than the time of x. Each node is associated with a type of event: internal nodes represent divergence events, while leaf nodes represent the sampling of new taxa. The *rank* of a node is the number of nodes in the tree with strictly smaller time. We say that a pair of nodes x, y

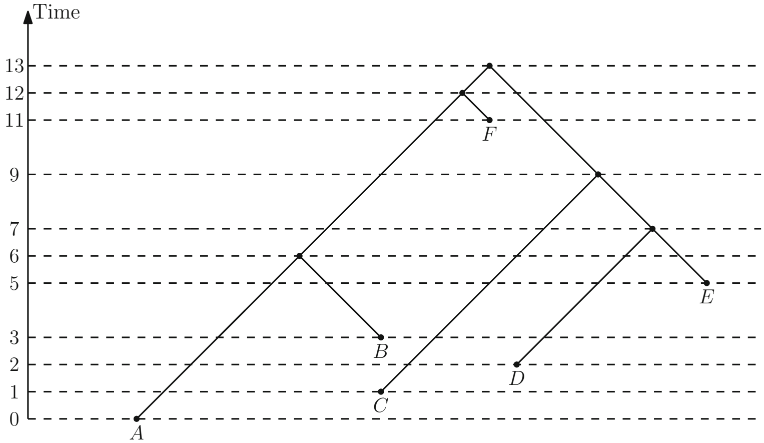

Fig. 1 Time-tree on 6 leaves. Time is measured by non-negative real numbers. If all times are integers, the tree is a discrete time-tree

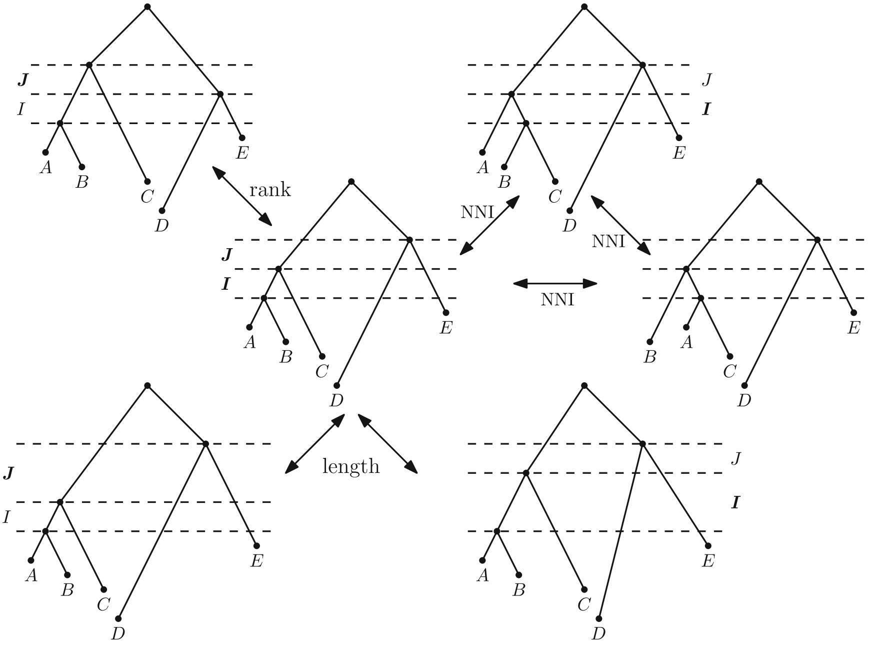

Fig. 2 All possible moves performed on intervals I and J. Assuming that the length of every event interval is either 1 or 2, the outer trees are all possible neighbors in DtT of the tree in the middle obtained by moves performed on intervals I and J. The trees on the right are obtained by moves performed on interval I and those on the left on J

of a discrete time-tree is an *event interval* if there exists no node z such that the time of z is between the time of x and y. Hence, an event interval is an interval between two divergence events, a taxon and a divergence event, or two taxa. Note that taxon events can be younger than divergence nodes, i.e. when the sampling time precedes divergence events on the tree. The difference between the times of x and y is called the *length* of the interval x, y. We identify two discrete time-trees if they are isomorphic as trees and the isomorphism preserves ranks of the nodes as well as event interval lengths.

We are now ready to introduce a hierarchy of discrete time-trees. At the bottom level of our hierarchy is the well-known NNI graph, which does not have timing information. NNI is a graph with the vertices being all trees on n leaves. Two trees T and R are adjacent in NNI if there exists an edge e in T and an edge f in R such that both edges are not adjacent to a leaf and the graph obtained from T by shrinking e to a vertex is isomorphic to the graph obtained from R by shrinking f. We denote this graph by DtT_0, where DtT stands for "discrete time-trees".

The following graph DtT_m forms the level $m > 0$ of the hierarchy. The set of vertices of the graph is the set of all discrete time-trees on n leaves such that every event interval has its length not greater than m. Two trees T and R are adjacent in DtT_m if R can be obtained from T by one of three operations: a *length move performed on interval I*, *swapping the rank of two nodes x and y on interval I* or an NNI *move performed on interval I* (Fig. 2). A length move changes the length of I by 1. Swapping

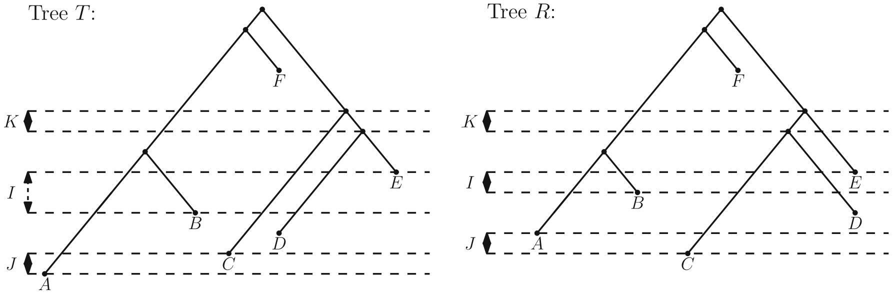

Fig. 3 Trees T and R are at DtT_2 distance 3. To move from T to R in DtT_2, one must decrease the length of interval I, swap the ranks of nodes A and C bounding interval J, and perform an NNI move on interval K, resolving nodes C and D

the rank of two nodes swaps the times of the nodes; such a swap is only possible when the nodes bound an event interval I of length 1. R can be obtained from T by an NNI move if there exist event intervals I_T and I_R of length 1 in T and R, respectively, such that the graphs obtained by shrinking I_T and I_R to vertices are isomorphic and the isomorphism preserves the lengths of the event intervals. In other words, by going from one tree to an adjacent tree in the DtT_m graph we can either change the length of one event interval by one unit, swap the rank of two nodes bounding an event interval of minimal length, or send the length of an event interval of minimal length down to zero and then resolve the multifurcation to either of the two possible trees. In the latter case, the new interval is of minimal length. See Fig. 3 for an example of the full variety of possible moves.

Note that DtT_0 can be obtained from DtT_1 by "forgetting" ranks and DtT_1—from DtT_m by forgetting lengths. In general, the graph structure for $m > 1$ can be understood by considering all four-tree diagrams as the four trees on the right in Fig. 2 and introducing an extra layer of nodes in every tail adjacent to the NNI triangle.

This hierarchy can be seen as a set of discrete refinements of the full space of phylogenetic time-trees, where event intervals can take an arbitrary real value. Indeed, if we allow the lengths of event intervals to take every possible non-negative real value and impose the Euclidean metric on trees with the same ranked topology, then we get the τ-space introduced by Gavryushkin and Drummond (2016). In this case, DtT_1 is the graph with the vertices being the orthants of τ-space and the adjacency relation being the relation of "have a shared facet of co-dimension 1." Similarly DtT_0 is the adjacency graph on the orthants of BHV space (Billera et al. 2001).

The DtT_m graph on ultrametric trees, that is the set of trees such that each leaf has the same time, is denoted by DtTu_m. The DtT_1 graph is denoted by RNNI and called the *ranked* NNI graph; on ultrametric trees we will denote it RNNIu. The R in RNNI stands for *ranked* and not for rooted. We emphasize that although this is the RNNI *graph*, NNI *moves* can be applied to trees in any of the spaces considered (under appropriate conditions). For simplicity, we use the generic identifier DtT to refer to a DtT_m with an arbitrary $m > 1$.

A graph can also be seen as a metric space where the distance is given by the length of a shortest path, so we will refer to DtT_m and other graphs introduced above as both, in agreement with Semple and Steel (2003).

3 Geometry and complexity of discrete time-trees

We begin by considering the shortest path distance on the graphs. It is well-known that computing distances is NP-hard in NNI (Dasgupta et al. 2000). Hence the following question is natural.

Problem 1 *What is the complexity of computing the distance between two discrete time-trees?*

Although this problem remains open, we make progress towards the solution by establishing several geometric and algorithmic properties of these graphs. First, we demonstrate in the following example that even for caterpillar trees the complexity cannot be derived from that of NNI distance. A *caterpillar tree* (sometimes called a *ladder tree*) has all leaves connected to a single path from the root. A *cherry* is a pair of taxa adjacent to a common internal node in the tree.

Example 2 (see Fig. 4) Let T be the ultrametric caterpillar tree denoted (((((((1, 2), 3), 4), 5), 6), 7) in Newick notation (Felsenstein et al. 1990) and R be the ultrametric tree ((((((1, 4), 5), 6), 2), 3), 7). Then a shortest NNI path is given by first making a cherry (2, 3), then moving the cherry up to the split 1456 | 7, and then resolving the cherry back. In RNNIu this path is not shortest, and one shortest path moves the parents of 2 and 3 up independently.

A set of trees A is called *convex* if for every pair of trees from A, there exists a shortest path between them such that every tree on the path belongs to A. Example 2 can be generalized to show that the set of trees of the form $(\dots(1, i_2), \dots, i_{n-1}), n)$, where $\{i_2, \dots, i_{n-1}\} = \{2, \dots, n-1\}$, is convex in DtT and is not convex in NNI. Indeed, the example shows that if two leaves have to be moved up, they can be moved independently along a shortest path. The generalization of this statement to an arbitrary number of leaves implies convexity in DtT, while the need to group leaves in NNI as in the previous example shows non-convexity in NNI space. This basic difference in

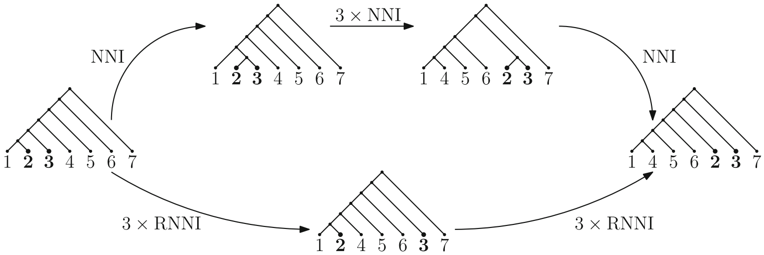

Fig. 4 Shortest paths in NNI may not be shortest in RNNI

the structure of the two graphs suggests that the complexity of computing the distance is likely to be different.

We proceed by establishing basic geometric properties of discrete time-trees.

3.1 Sizes of neighborhoods

We first bound the number of trees in each graph. As expected, times greatly expand the size of the graphs.

Lemma 3 (see Semple and Steel (2003)) *Let $|V|$ be the number of vertices in the graph, then $|V|$ is equal to*

$$\begin{array}{llll} \dfrac{(n-1)!\,n!\,n!\,m^{n-1}}{2^{n-1}} & \textit{in } \mathrm{DtT_m}, & \dfrac{(n-1)!\,n!\,m^{n-1}}{2^{n-1}} & \textit{in } \mathrm{DtTu}_m, \\ \dfrac{(n-1)!\,n!\,n!}{2^{n-1}} & \textit{in } \mathrm{RNNI}, & \dfrac{(n-1)!\,n!}{2^{n-1}} & \textit{in } \mathrm{RNNIu}, \\ (2n-3)!! & \textit{in } \mathrm{NNI}. & & \end{array}$$

Proof The proof for NNI can be found in Semple and Steel (2003). The rest follow using similar arguments. □

We continue by tightly bounding the sizes of 1-neighborhoods of various discretizations of the space of time-trees. Each has a 1-neighborhood size that is linear in n allowing for efficient local traversal and enumeration, a property of importance for phylogenetic algorithms, e.g. tree proposals in MCMC.

Lemma 4 *Let T be a tree on n leaves and* $\deg(T)$ *the number of trees adjacent to T. Then*

$$\begin{array}{llll} 2(n-1) \leq \deg(T) \leq 5n-6 & \textit{in } \mathrm{DtT}, & n-1 \leq \deg(T) \leq 3n-5 & \textit{in } \mathrm{DtTu}, \\ n-1 \leq \deg(T) \leq 3n-4 & \textit{in } \mathrm{RNNI}, & n-1 \leq \deg(T) \leq 2n-4 & \textit{in } \mathrm{RNNIu}, \\ \deg(T) = 2n-4 & \textit{in } \mathrm{NNI}. & & \end{array}$$

All bounds are tight.

Proof We prove the bounds by providing exemplar trees with the specified degrees and explaining why no tree can have a greater (or lesser) degree. Recall that $m > 1$ is the maximal possible length of an event interval in a tree from DtT. The lower bound in DtT is attained by any tree with all event intervals being of length m. In this case, $\deg(T)$ is simply the number of event intervals, since every event interval adds 1 to the total degree of the tree. Other trees have the same or more possible event interval changes, showing that this is a lower bound. The upper bound is attained by a caterpillar tree with all intervals short, every taxon being younger than every divergence event, and both taxa in the cherry being younger than at least one other taxon, that is, a caterpillar tree with internal nodes having ranks $n, \ldots, 2(n-1)$ and the ranks of the taxa in the cherry $< n-1$. In this case, $\deg(T)$ is bounded by the sum of the $2(n-1)$ possible interval length changes, at most $2(n-2)$ NNI neighbors, and at most n rank changes that can occur between the n taxa and the root of the cherry. In total, $\deg(T) \leq 2(n-1) + 2(n-2) + n = 5n - 6$. Note that this is an upper bound indeed, as each of the $2(n-2)$ intervals excluding the most recent can contribute either a rank change or 2 NNI moves. The caterpillar tree described above maximizes the number of intervals that contribute 2 NNI moves and enables the rest of intervals to contribute a rank change.

For ultrametric trees, the number of event intervals is $n-1$, hence they add $n-1$ to the degree of the caterpillar tree from interval length changes. The number of intervals on which an NNI move is possible is $n-2$ for ultrametric trees, hence they contribute $2(n-2)$ to the degree. In total, this gives the upper bound of $3n-5$ for ultrametric trees.

Every NNI move results in two neighbors, so the equality $\deg(T) = 2(n-2)$ follows for the NNI graph. Indeed, no NNI move can be performed on an edge adjacent to a taxon, exactly two NNI moves can be performed on every edge between internal nodes (the number of which is $n-2$), and no two trees obtained by an NNI move performed on different edges are identical.

The lower bound in RNNI is attained by the caterpillar-tree where taxa get ranks $0, 1, 3, 5, \ldots, 2n-3$ and internal nodes get ranks $2, 4, 6, \ldots, 2n-2$. In other words, the divergence events alternate with the taxa in the ranked topology of the tree so that if we parse the tree from the present to the past, we meet the nodes in the following order: taxon, taxon, coalescence, taxon, coalescence, taxon, coalescence, and so on. In this case, intervals bounded by a taxon from below add nothing to the degree of the tree and intervals bounded from below by an internal node add one each, hence $n-1$ in total. The upper bound for both RNNI and RNNIu is obtained in the same way as for DtT.

For the lower bound in RNNIu, we note that the oldest event interval (the one adjacent to the root) necessarily contributes 2 to the degree. Hence the lower bound can be reached by a tree such that no other interval contributes more than 1. The degree of such a tree is $n-1$ in RNNIu. □

Lemma 4 gives tight bounds for the number of trees in a 1-neighborhood, while the following theorem gives an upper bound for the number of trees in an r-neighborhood. The best known upper bound for NNI that we are aware of is $3^{n-2}2^{4r}$ (Li et al. 1996). We obtain a smaller bound for RNNIu and, hence, improve the result for NNI as well.

Theorem 5 *The number of trees within distance r from any given tree is at most*

$$\begin{array}{llll} 3^{n+2r-1} & \textit{in } \text{RNNIu}, & 3^{2n+2r-1} & \textit{in } \text{RNNI}, \\ 4^{n+2r-1} & \textit{in } \text{DtT}, & 4^{2n+2r-1} & \textit{in } \text{DtTu}. \end{array}$$

Proof We describe the proof in detail for RNNIu and then explain how to modify the proof for the other three graphs. The proof employs the technique by Sleator et al. (1992) for counting paths in a graph using a *graph grammar*. A graph grammar is a method of encoding each possible graph as a finite set of productions (described more formally below). Given the grammar, the number of possible productions to the power of the encoding length for a given neighborhood radius gives a simple upper bound on the number of trees in a neighborhood. Specifically, Theorem 2.3 from Sleator et al. (1992), adapted to trees, states the following:

> Let T be a tree with h nodes, including all internal nodes and taxa, Γ be a graph grammar, d be the number of vertices in left sides of Γ, and k be the maximum number of nodes in any right side of a production of Γ. Let $R(T, \Gamma, r)$ be the set of graphs obtainable from T by derivations in Γ of length at most r. Then $|R(T, \Gamma, r)| \leq (d+1)^{h+k\cdot r}$.

We first apply Theorem 2.3 from Sleator et al. (1992) and then improve the bound using specific properties of the RNNIu graph. We note that since our trees are ranked trees, they possess additional structure (ranking) on top of the tree graph structure, hence we will be applying the same style of argument as in Sleator et al. (1992) but to a somewhat different structure.

We now introduce the *graph grammar* for RNNIu. Since we will apply the graph grammar to modify trees, we use the graph-theoretic terminology here and recall that a tree is a graph. A graph grammar consists of a finite set of productions $\{L_i \to_i R_i\}$, where L_i and R_i are connected undirected edge-end labeled graphs and $\to_i$ is a one-to-one map between half-edges of L_i and those of R_i. Here, for every edge in the tree we distinguish between its two ends and refer to them as half-edges—this is necessary to be able to substitute one subgraph by another in a unique way by connecting corresponding half-edges—as in Sleator et al. (1992). The productions are then applied to the starting tree T to derive all possible trees at RNNIu distance up to r from T. By a *running tree at stage s* we mean the tree obtained after s applications of the production rules in the derivation. A production is said to be *ready* at a stage s of the derivation if the running tree at stage s has a subgraph isomorphic to the left side L_i of the production. A ready production can be applied to the running tree by destroying all nodes corresponding to the left side of the production under the isomorphism and replacing them with the right side R_i of the production. The map $\to_i$ of the production then says how to reconnect the right side of the production to the half-edges of the running tree that were created after the destruction. The obtained tree is the new running tree for the next stage $s+1$ of the derivation. See Sleator et al. (1992) for precise definitions and details.

The graph grammar for RNNIu is the grammar Γ shown on Fig. 5. Note that the definition of a graph grammar requires the left sides L_i to be connected and we have a disconnected left side in the third production. We can adapt our graph grammar to fit this requirement by noting that the nodes on the left side of the production must be of consecutive ranks, so we consider those nodes as being adjacent via a second type of adjacency relation by declaring nodes with consecutive ranks to be adjacent. This second type of relation is in addition to the first type of adjacency relation given by the branches of the tree. Furthermore, this second type of adjacency allows us to avoid considering two separate moves for the left and right sister nodes. Indeed, the half-edge labels order the edges to make the distinction between left and right, however since our nodes are ordered via ranking, this ordering of edges is irrelevant. We will further exploit this important property below, where we improve the bound obtained directly from grammar Γ.

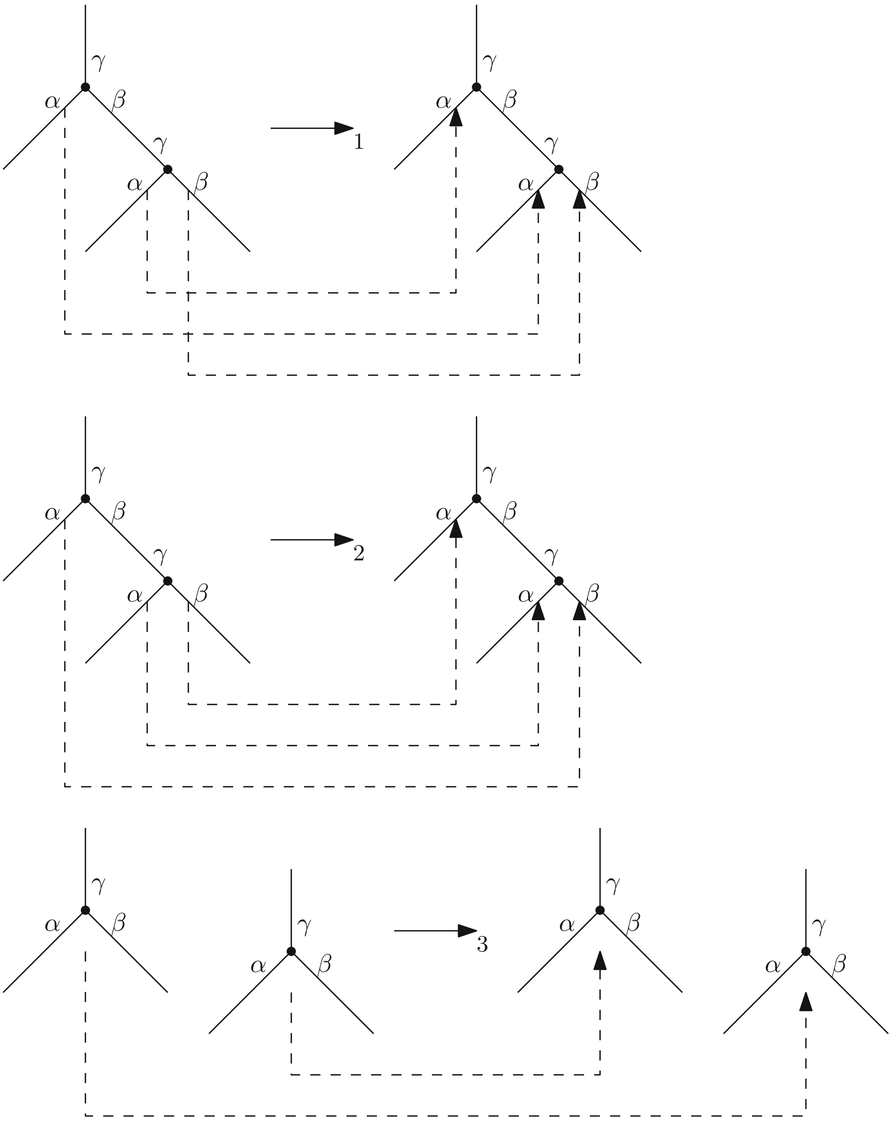

Fig. 5 Graph grammar Γ for RNNIu. Maps $\rightarrow_i$ are shown by *dashed lines*. In the first two productions, the edge-ends marked by β and γ without *dashed lines* are mapped to each other: *top* γ to *top* γ, β to β, *bottom* γ to *bottom* γ. In the last production, edge-ends marked by the same label are mapped to each other. All pairs of nodes on both sides of each production must be of consecutive ranks. The first two productions correspond to an NNI move, while the last production corresponds to rank change

The number of vertices in left sides of Γ is 6, the maximum number of vertices in any right side of a production of Γ is 2. Since the number of internal nodes of a tree on n leaves in RNNIu is $n - 1$, directly applying Theorem 2.3 from Sleator et al. (1992) we get the bound of 7^{n+2r-1}.

This bound can be improved following the techniques described in Sects. 3.2–3.4 of (Sleator et al. 1992, see also Sect. 5) by bounding the number of possible productions available at a given time by a number smaller than their total. Every node of a tree in any application of a production of grammar Γ can play either of two roles: top node or bottom node. Hence to indicate that a pair of nodes is ready for being destroyed and replaced using a production, we must specify which node is a top node and which is a bottom node. We use a total of three node-labels (Sleator et al. 1992 call these "labels of vertices") to identify these two roles and to distinguish which of the two possible NNI moves is applied. We claim that three node-labels a, b, c is enough to specify which productions can be applied to a given tree. We must redefine the notion of readiness to verify this claim. We say that a pair of consecutive nodes is *ready* if the node-label of the bottom node is a and the node-label of the top node is either b or c. If two consecutive nodes are ready and not connected by an edge in the tree, then only the rank move is possible. If they are, the type of the move is determined by the node-label of the top node. Hence, this notion of readiness uniquely identifies which of the three productions is applied.

Sleator et al. (1992) use a special node-label, called the "zero label", to mark nodes that should not be destroyed. They also explain how to eliminate the zero label in various cases. In our case, we can use one of the labels that are already in use, namely node-label a. That is, to indicate that a node v should never be destroyed, we label the node with an a. This labeling will indeed preserve the node v because of the following. If w is the node directly succeeding v at some step of the derivation then w can only be marked with an a and hence the pair v, w is not ready. If w is the node directly preceding v at some step of the derivation then the pair w, v is never ready. Hence v is preserved in either case.

Thus, all possible configurations encoded by the six nodes on the left sides of the productions of Γ can be encoded using three node-labels. This implies that the derivation can be encoded by a ternary sequence of length $n-1+2r$. Indeed, the first $n-1$ entries of the sequence are needed to encode all possible node-labelings of the initial tree, plus two entries are needed for each of the r applications of productions because every production creates two new nodes, each of which has to be node-labeled. The number of such strings, 3^{n-1+2r}, gives the desired improved bound.

The proof for the other three spaces follows similarly.

For RNNI, we extend the graph grammar Γ by two productions, namely, one for the move when two taxa swap their ranks and one for the move when a taxon and an internal node swap their ranks. Since in both of these productions the only possible type of move is the rank move, two node-labels is enough for these productions. Since the number of (internal and leaf) nodes in an RNNI tree is $2n-1$ and the maximum number of nodes on right sides of the productions is still 2, the desired bound is $3^{2n-1+2r}$.

For DtTu, we have to extend the grammar Γ by two productions, namely, one for increasing the length of the branch between two nodes in the production and one for decreasing. This increases the number of necessary node-labels by one. Indeed, if a production has a long event interval on the left side, three node-labels are enough to indicate whether the event interval length increases or decreases. Alternatively, if a production has a short event interval on the left side, three node-labels are needed for the top node to indicate the type of move to apply: a length increase move, one of the two possible NNI moves, or a rank swap move. Together, this requires 4 node-labels and the desired bound is 4^{n-1+2r}.

For DtT, we extend the grammar constructed for DtTu by four productions, namely, one for the move when two taxa either swap their ranks or increase the interval length between them, one for the move when two taxa either increase or decrease the interval length between them, one for the move when a taxon and an internal node either swap their ranks or increase the interval length between them, and one for the move when a taxon and an internal node either increase or decrease the interval length between them. All these moves require three node-labels, hence the desired bound is $4^{2n-1+2r}$. □

3.2 Diameter of the graphs

Since Theorem 5 bounds the sizes of r-neighborhoods by functions of the form $a^{f(n)+2r}$, the following corollary bounds the diameters of the graphs under consideration from below.

Corollary 6 *Let $\Delta(\mathcal{G})$, $|\mathcal{G}|$, and $\delta_r(\mathcal{G})$ be the diameter, the number of vertices, and the number of vertices within distance r from any given vertex in graph $\mathcal{G}$, respectively. If $\delta_r(\mathcal{G}) = a^{f(n)+2r}$ then*

$$\Delta(\mathcal{G}) \geq \frac{1}{2}\log_a \frac{|\mathcal{G}|}{b},$$

where $b = a^{f(n)}$. In particular, $\Delta(\text{RNNIu}) \geq \frac{1}{2}\log_3 \frac{(n-1)!\,n!}{6^{n-1}}$.

Proof We note that every r that satisfies the inequality $\delta_r(\mathcal{G}) < |\mathcal{G}|$ is smaller than the diameter, that is, $\Delta(\mathcal{G}) > r$ for all such r. By taking $\log_a(\cdot)$, the former inequality is equivalent to $\log_a b + 2r < \log_a |\mathcal{G}|$, that is, to $r < \frac{1}{2}\log_a \frac{|\mathcal{G}|}{b}$. Hence, the desired inequality. □

Now we estimate the diameter of the graphs from above, complementing the above lower bounds. Recall that $\Delta(\mathcal{G})$ is the diameter of graph $\mathcal{G}$.

Theorem 7 *For $n \geq 4$, $\Delta(\text{RNNIu}) \leq n^2 - 3n - \frac{5}{8}$.*

Proof Let T and R be trees in RNNIu. We show that there exists a path between them of length $n^2 - 3n - \frac{5}{8}$ or shorter. We denote the taxa on one side from the root of T by A and the rest of the taxa by B. Let $A = A_1 \cup A_2$ and $B = B_1 \cup B_2$ so that the root of R splits the taxa into $A_1 \cup B_1$ and $A_2 \cup B_2$. Note that all A_i's and B_i's have to be disjoint. We assume that $|A_1| = |A_2| = |B_1| = |B_2| = \frac{n-1}{4}$ and denote this number by s. We will see later in the proof that this assumption does not restrict the generality of our argument.

We construct the path from T to R proceeding in the following steps, which are illustrated on Fig. 6. Let $f(n)$ be the desired upper bound. For move counts in the

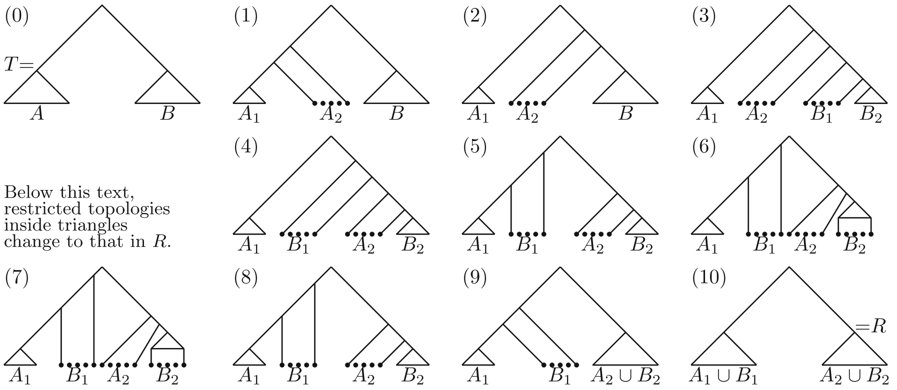

Fig. 6 An algorithm to compute a (not necessarily shortest) path between discrete time-trees. Although the trees at steps (6) and (7) look identical, the topologies inside the *triangles* are different: those at step (6) correspond to the restriction of T, at step (7)—of R

remainder of this proof, we will allow the "null move" in which no modification is made.

(1) Move all the nodes adjacent to taxa in A_2 on top of all the other nodes in T, apart from the root. The tree restricted to A_2 is then a caterpillar tree. This takes $3s^2$ moves.
(2) Move the caterpillar tree on A_2 to the other side of the root. This takes $\frac{1}{2}s^2 + \frac{1}{2}s$ moves.
(3) Move all the nodes adjacent to taxa in B_1 on top of all nodes apart from those adjacent to taxa in A_2 and the root. The tree restricted to B_1 is then a caterpillar tree. This takes $2s^2$ moves.
(4) Swap the caterpillars on A_2 and B_1. This takes s^2 moves.
(5) Move the caterpillar tree on B_1 to the other side of the root. This takes $\frac{1}{2}s^2 + \frac{1}{2}s$ moves.
(6) Translate the tree restricted to B_2 up to place it between the trees on A_1 and A_2. This takes s^2 moves.
(7) Convert the tree on $A_1 \cup B_2$ to the corresponding subtree in R in terms of topology and relative ranking of nodes within the subtree. This takes $2f(n/4)$ moves.
(8) Translate the tree on B_2 down so that the tree on $A_1 \cup B_2$ matches the corresponding subtree in R in terms of topology and relative ranking of nodes. This takes s^2 moves.
(9) Merge the tree on A_2 with the tree on B_2 so that the tree on $A_1 \cup A_2 \cup B_2$ coincides with that in R. This takes $2s^2$ moves.
(10) Merge the tree on B_1 with the tree on A_1 so that the tree coincides with R. This takes $3s^2$ moves.

In total, we have a recursive equation $f(n) = 14s^2 + s + 2f(n/4)$.

Assuming that the solution is a polynomial, we see that the polynomial must be of degree 2. It remains to apply the recursive equation to find the coefficients of the polynomial:

$$f(n) = n^2 - 3n - \frac{5}{8}.$$

It remains to note that the assumption $|A_1| = |A_2| = |B_1| = |B_2| = \frac{n-1}{4}$ does not increase the distance between T and R. Indeed, the sets can be chosen so that at most half of the nodes have to cross the root, that is, $|A_2 \cup B_1| \leq \frac{n-1}{2}$. Furthermore, the smaller the size of $A_2 \cup B_1$ the shorter the path. Finally, the total length of the path is maximized when $|A_2| = |B_1|$ and $|A_1| = |B_2|$. □

Although we stated and proved the theorem only for the RNNIu graph, the idea can be adapted to the other graphs considered here. However, this adaptation goes beyond the scope of this paper.

We also note that the proof of the theorem provides an efficient algorithm for computing (not necessarily shortest) paths between discrete time-trees. This algorithm can be used for efficient exploration of the space, e.g. via MCMC, where approximate paths can be used to improve mixing by making distant proposals. The algorithm might also be useful to determine if two trees appear to be separated by a valley (e.g. of posterior probability) or if they are rather located on a plateau.

A non-ranked tree has a symmetry associated with every internal node. Those symmetries can be employed to consider only a single representative from equivalent pairs of trees, such as by using tanglegrams (Matsen et al. 2015; Whidden and Matsen 2016), where a *tanglegram* is a graph formed by identifying the leaves of two phylogenetic trees. However, as the following proposition shows, ranked trees are free from all but one of those symmetries, hence the tanglegram approach is not applicable in this case.

Proposition 8 *Let T be a tree from* DtT, DtTu, RNNI, *or* RNNIu *and σ a permutation of taxa of T. Let T_σ be the tree obtained from T by permuting the taxa using σ. Then $T = T_\sigma$ if and only if σ is either the identity permutation or a transposition of a pair of taxa that form a cherry in T.*

Proof The sufficiency is obvious. For the necessity, assume that σ transpose taxa i, j such that i, j is not a cherry in T. This implies that i and j have different parents. Hence $T \neq T_\sigma$ because the ranks of the parent of i are different in T and T_σ. □

These observations suggest that the NNI graph is geometrically and algorithmically different from the other four graphs.

3.3 Efficient algorithm for generating RNNI graphs

We now introduce an algorithm to compute the RNNI graph on n leaves by extending an algorithm from Whidden and Matsen (2016). The input to our algorithm is a set S of RNNI trees in the format described by Gavryushkin and Drummond (2016) and implemented in Gavryushkin and Drummond (2015) (see also Semple and Steel 2003). The two key requirements of the algorithm are the ability to enumerate the neighbors of a given tree in the RNNI graph, and the ability to determine whether a given tree is already a vertex of the graph. Both of these requirements follow from an efficient unique representation for ranked trees.

A node of a tree is given by the set of taxa that are all descendants of the node. A tree is then given by a sequence of nodes ordered by their ranks. Clearly, this obtained representation is unique. For example, the representation of the tree on Fig. 1 is $(\{A\}, \{C\}, \{D\}, \{B\}, \{E\}, \{A, B\}, \{D, E\}, \{C, D, E\}, \{F\}, \{A, B, F\}, \{A, B, C, D, E, F\})$, or $(\{A, B\}, \{D, E\}, \{C, D, E\}, \{A, B, F\}, \{A, B, C, D, E, F\})$ if we make the tree ultrametric by placing all taxa below the most recent divergence event. Using this representation of trees we construct a map $\nu : S \to \omega$, where ω is the set of non-negative integers. This map allows us to determine whether or not a tree is already a vertex of the graph. This takes linear time if the map is implemented as a trie. We can also enumerate the neighbors of a tree in $O(n^2)$-time using Lemma 4. The algorithm therefore takes $O(|S| * n^2)$-time. The high-level steps of the algorithm are as follows. See Whidden and Matsen (2016) for an analogous proof of correctness.

```
Construct-RNNI-Graph($S$)

1. Let $G$ be an empty graph.
2. Let $\nu$ be an empty mapping from trees to integers.
3. Let $i = 0$.
4. For each of the $m$ trees:
   (a) Add a vertex $i$ to $G$ representing the current tree $T_i$.
   (b) Add $T_i \to i$ to $\nu$.
   (c) For each neighbor $T_j$ of $T_i$:
       (j) If $T_j$ is in $\mathrm{dom}(\nu)$ then add an edge $(i, \nu(T_j))$ to $G$.
   (d) $i = i + 1$.
```

4 Open problems and conjectures

In the previous section we posed Problem 1, which asks about the complexity of computing the distance in discrete time-tree graphs. This problem is the primary obstacle for actual biological applications of the introduced graphs. The history of research into computational complexity of phylogenetic graphs is very rich and exciting, in particular for the case of the NNI graph, with a number of erroneous results being published over the 25 years it took to settle the complexity. See Dasgupta et al. (2000) for a detailed discussion of those publications, some of which claimed that NNI distance is NP-hard while others claiming that NNI distance is decidable in polynomial time. The latter claims were mainly based on the so-called Split Theorem which states the following: If a partition of leaves given by an edge is shared between two trees then there exists a shortest NNI path between the trees such that every tree on the path maintains the partition. These claims were finally refuted by Li et al. (1996), who proved that the Split Theorem does not hold in the NNI graph. Hence, a natural question that would contribute to understanding Problem 1 is whether or not the Split Theorem holds in the RNNI graph and other time-tree graphs.

4.1 Split Theorem

Li et al. (1996) used properties of the diameter of the NNI graph and the sizes of its neighborhoods to provide an example of trees such that every shortest NNI path between the trees fails to maintain a partition of taxa shared by the origin and destination trees. Specifically, they showed that sorting two caterpillar trees simultaneously is more efficient than sorting them independently, provided the size of the trees is large enough. In the following, we conjecture that the ranked versions of NNI graph maintain splits along shortest paths. Our conjecture is based on the fact that RNNI does not satisfy the diameter bounds necessary for the argument in Li et al. (1996) to go through—see Theorems 5 and 7. Indeed, since sorting a caterpillar tree of size k in RNNI takes fewer moves than merging two such caterpillar trees and then separating them, the counter-example from Li et al. (1996) does not apply in RNNI. This counter-example using two trees encodes the basic way in which NNI shortest paths are non-trivial: that one can economize by first grouping leaves into bundles, moving the bundles, and then breaking the bundles apart. Because this basic operation does not provide an advantage in the simplest case of two caterpillar trees in RNNI, we do not believe that it will hold for more complex collections of moves.

Conjecture 9 *The following is true in all graphs* DtT, DtTu, RNNI, *and* RNNIu, *but not in* NNI. *If a partition of leaves given by an edge is presented in two trees T and R then the partition is presented in every tree on every shortest path between T and R.*

4.2 Computing the distance

In this section, we bound the number of neighbors of a tree x in the discrete time-tree graphs that are closer than x to a given tree y, *under the assumption that Conjecture 9 holds*. We denote trees by lowercase letters to stress the fact that we are considering graphs as metric spaces in this section. We show that the maximum fraction of neighbors that tend closer to y grows linearly with respect to $d(x, y)$. When the graph is seen as a metric space, this property is important for the study of the curvature of the space as well as convergence properties of random walks over the space. This will be useful in future studies analogous to Whidden and Matsen (2016). Furthermore, the result provides a further insight into possible approaches to Problem 1.

Theorem 10 *Assume that Conjecture 9 holds. Let x and y be two trees and $N(x)$ a one-neighborhood of x. Then the number of trees $u \in N(x)$ such that $d(u, y) \leq d(x, y)$ is at most*

(i) $3d(x, y)$ *in* RNNI*,*
(ii) $4d(x, y)$ *in* DtT_2*,*
(iii) $5d(x, y)$ *in* DtT_m *for* $m > 2$.

Proof We first prove the statement for DtT_2. Let U be the set of neighbors u of x such that $d(u, y) \leq d(x, y)$, as stated in the theorem. We partition U into three sets of trees—I, R, and L—obtained from x by a single NNI move, rank change, or length change, respectively. We then prove the theorem by bounding the size of each partition by $2d(x, y)$, $d(x, y)$, and $d(x, y)$, respectively.

We first consider some basic properties of minimal length paths between x and y. Observe that at most $d(x, y)$ bipartitions differ between x and y, as each NNI operation replaces one bipartition with another (and rank and length changes do not modify bipartitions). Now, Conjecture 9 implies that no minimal length path from x to y will replace a bipartition that is common to x and y.

We are now ready to bound the sizes of each partition I, R, and L. First, consider the subset I of closer neighbors obtained via NNI moves. By our observations above, $|I|$ is bounded by the number of NNI operations that modify one of the at most $d(x, y)$ bipartitions of x that are not a bipartition of y. There are two neighbors of x that lack any given bipartition (obtained by moving either the left or right subtree located below the bipartition). Therefore, $|I| \leq 2d(x, y)$.

Second, consider the subset R of closer neighbors obtained via rank change moves. These operate either on one of x's unique bipartitions or a bipartition common to x and y that differs in rank. As observed above, the number of unique bipartitions is bounded by the maximum number of NNI moves on any minimal x to y path. Any such path must fix the ranks of each common edge. In other words, if r_1 is the number of trees in U which are obtained from x by a rank move corresponding to a common edge and r_2 is the number of trees in U which are obtained from x by a rank move corresponding to a unique edge of x, then $r_1 + r_2 \leq d(x, y)$, because the r_1 moves have to be done along every shortest path from x to y. Thus, the total size of R is bounded by $d(x, y)$.

The bound of $d(x, y)$ on the number of length changes that can be applied to move x closer to y follows similarly. Therefore, there are at most $|I| + |R| + |L| \leq 2d(x, y) + d(x, y) + d(x, y) = 4d(x, y)$ trees $u \in N(x)$ such that $d(u, y) \leq d(x, y)$.

The statement for the other two graphs follows similarly: for RNNI we need to count only for trees from I and R, for DtT we will have to add $2d(x, y)$ instead of $d(x, y)$ for L. □

We conclude by noting that most of the results in this paper can be generalized to the class of sampled ancestor trees (Gavryushkina et al. 2014), where DtT trees are enriched by an extra event called *sampled ancestor*—an internal node marked by a taxon with exactly one child. However, currently only a few methods of sampling the space of sampled ancestor trees are known (Gavryushkina et al. 2016) and the geometry of the space is poorly understood (Gavryushkin and Drummond 2016), hence we do not include the generalization here.

5 Discussion

In this paper, we have introduced a hierarchy of discrete time-trees that naturally approximates the full space of phylogenetic time-trees. The first two levels of the hierarchy are the well-known classes of rooted phylogenetic trees and ranked phylogenetic trees. We extend this previous work by considering a novel graph structure, the RNNI graph, on the set of ranked trees. In the same way that the NNI graph is the discrete component of the full space of phylogenetic trees, the RNNI graph is the simplest discrete component of the full space of time-trees. Hence, the introduced hierarchy fits and refines naturally the classical picture of phylogenetic graphs.

Surprisingly, the geometry of RNNI and NNI graphs differ in many aspects including the diameter, the sizes of neighborhoods, and the convexity of caterpillar trees. This suggests that the computational complexity of computing distances in the two graphs is likely to be different, at least at the level of certain subgraphs such as the restriction to caterpillar trees. Although the size of discrete ranked graphs are much larger than their NNI graph counterparts, ranks provide an inherent structure that may serve as a useful avenue of attack for future algorithms.

We introduce and make some initial progress on the computational complexity of calculating the distance between two time-trees (Problem 1). This is an important step for further progress in theoretical research as well as computational and biological applications of time-trees. In particular, a better understanding of this problem would have the following applications.

5.1 Efficient tree search algorithms

Some of the most popular phylogenetic time-tree inference methods utilize the Markov Chain Monte Carlo (MCMC) algorithm to sample a probability distribution over the space of time-trees. The most efficient proposals on time-trees will take into account both local and global geometry of the corresponding space. Our hierarchy of discrete time-trees offers a natural discrete structure of the space used in such tree proposals.

Thus far, the most widely used modifications of trees are NNI, SPR, and their variants. The graph distance inherited from these modifications is known to be NP-hard to compute, and although sometimes NP-hard problems can be solved efficiently for practically interesting data sets, this is not the case for these phylogenetic graphs of ranked trees, where the size of tractable problems is an order of magnitude smaller than those that routinely arise in practice. Hence, a computationally tractable phylogenetic graph is a highly relevant and desirable object for the field, and we believe that the RNNI graph should have this property.

5.2 Convergence of phylogenetic algorithms

A tractable phylogenetic distance would greatly assist methods to assess convergence of Bayesian tree inference algorithms based on MCMC and related methods. In other areas of statistics, distance methods are widely used to test for convergence, but the computational complexity of phylogenetic distances makes those methods applicable to only very moderate data sizes. Phylogenetic distances are especially suitable for such applications because they correspond to the distance inherited from the inference algorithm, being based on the same sorts of tree rearrangements.

Another important property for phylogenetic applications is to compute the number of trees that are “equally good” under certain criterion. This intuition is formalized by Sanderson et al. (2011) in the notion of a terrace. To compute the terrace, an efficient tree metric is of crucial importance.

References

Åkerborg Ö, Sennblad B, Lagergren J (2008) Birth-death prior on phylogeny and speed dating. BMC Evol Biol 8(1):77. doi:10.1186/1471-2148-8-77

Baele G, Li WLS, Drummond AJ, Suchard MA, Lemey P (2013) Accurate model selection of relaxed molecular clocks in bayesian phylogenetics. Mol Biol Evol 30(2):239–243. doi:10.1093/molbev/mss243

Beerli P, Felsenstein J (2001) Maximum likelihood estimation of a migration matrix and effective population sizes in n subpopulations by using a coalescent approach. Proc Natl Acad Sci USA 98(8):4563–4568. doi:10.1073/pnas.081068098

Beiko RG, Keith JM, Harlow TJ, Ragan MA (2006) Searching for convergence in phylogenetic markov chain monte carlo. Syst Biol 55(4):553–565

Billera LJ, Holmes SP, Vogtmann K (2001) Geometry of the space of phylogenetic trees. Adv Appl Math 27(4):733–767

Dasgupta B, He X, Jiang T, Li M, Tromp J, Zhang L (2000) On computing the nearest neighbor interchange distance. In: Discrete mathematical problems with medical applications: DIMACS workshop discrete mathematical problems with medical applications, 8–10 Dec 1999, DIMACS Center, vol 55, p 19. American Mathematical Society

Drummond AJ, Ho SYW, Phillips MJ, Rambaut A (2006) Relaxed phylogenetics and dating with confidence. PLOS Biol 4(5):e88. doi:10.1371/journal.pbio.0040088

Drummond AJ, Nicholls GK, Rodrigo AG, Solomon W (2002) Estimating mutation parameters, population history and genealogy simultaneously from temporally spaced sequence data. Genetics 161(3):1307–20

Drummond AJ, Rambaut A, Shapiro B, Pybus OG (2005) Bayesian coalescent inference of past population dynamics from molecular sequences. Mol Biol Evol 22(5):1185–1192. doi:10.1093/molbev/msi103

Felsenstein J, Archie J, Day WH, Maddison W, Meacham C, Rohlf FJ, Swofford D (1986) The newick tree format. http://evolution.genetics.washington.edu/phylip/newicktree.html

Ford D, Matsen FA, Stadler T (2009) A method for investigating relative timing information on phylogenetic trees. Syst Biol 58(2):167–183. doi:10.1093/sysbio/syp018

Gavryushkin A, Drummond AJ (2015) Taugeodesic. doi:10.5281/zenodo.47152

Gavryushkin A, Drummond AJ (2016) The space of ultrametric phylogenetic trees. J Theor Biol 403:197–208. doi:10.1016/j.jtbi.2016.05.001

Gavryushkina A, Heath TA, Ksepka DT, Stadler T, Welch D, Drummond AJ (2016) Bayesian total-evidence dating reveals the recent crown radiation of penguins. Syst Biol. doi:10.1093/sysbio/syw060

Gavryushkina A, Welch D, Stadler T, Drummond AJ (2014) Bayesian inference of sampled ancestor trees for epidemiology and fossil calibration. PLOS Comput Biol 10(12):e1003919. doi:10.1371/journal.pcbi.1003919

Hohna S, Defoin-Platel M, Drummond AJ (2008) Clock-constrained tree proposal operators in bayesian phylogenetic inference. In: 8th IEEE international conference on 8th IEEE international conference on bioinformatics and bioengineering, BIBE 2008, pp 1–7. doi:10.1109/BIBE.2008.4696663

Huber KT, Spillner A, Suchecki R, Moulton V (2011) Metrics on multilabeled trees: interrelationships and diameter bounds. IEEE/ACM Trans Comput Biol Bioinform 8(4):1029–1040. doi:10.1109/TCBB.2010.122

Kuhner M K (2006) LAMARC 2.0: maximum likelihood and bayesian estimation of population parameters. Bioinformatics 22(6):768–770. doi:10.1093/bioinformatics/btk051

Kuhner MK, Beerli P, Yamato J, Felsenstein J (2000) Usefulness of single nucleotide polymorphism data for estimating population parameters. Genetics 156(1):439–447

Kuhner MK, Yamato J, Felsenstein J (1995) Estimating effective population size and mutation rate from sequence data using Metropolis-Hastings sampling. Genetics 140(4):1421–1430

Kuhner MK, Yamato J, Felsenstein J (1998) Maximum likelihood estimation of population growth rates based on the coalescent. Genetics 149(1):429–434

Lakner C, Van Der Mark P, Huelsenbeck JP, Larget B, Ronquist F (2008) Efficiency of Markov chain Monte Carlo tree proposals in Bayesian phylogenetics. Syst Biol 57(1):86–103

Lambert A, Stadler T (2013) Birth-death models and coalescent point processes: the shape and probability of reconstructed phylogenies. Theor Popul Biol 90:113–128. doi:10.1016/j.tpb.2013.10.002

Li M, Tromp J, Zhang L (1996) Some notes on the nearest neighbour interchange distance. In: Cai J-Y, Wong C (eds) Computing and combinatorics. Lecture notes in computer science, vol 1090. Springer, Berlin, pp 343–351. doi:10.1007/3-540-61332-3_168

Matsen IV FA, Billey S, Kas A, Konvalinka M (2015) Tanglegrams: a reduction tool for mathematical phylogenetics. arXiv:1507.04784 [q-bio.PE]

Mau B, Newton MA (1997) Phylogenetic inference for binary data on dendograms using Markov chain Monte Carlo. J Comput Graph Stat 6(1):122–131. doi:10.2307/1390728

Minin VN, Bloomquist EW, Suchard MA (2008) Smooth skyride through a rough skyline: Bayesian coalescent-based inference of population dynamics. Mol Biol Evol 25(7):1459–1471. doi:10.1093/molbev/msn090

Mossel E, Vigoda E (2005) Phylogenetic MCMC algorithms are misleading on mixtures of trees. Science 309(5744):2207–2209. doi:10.1126/science.1115493

Mossel E, Vigoda E (2006) Limitations of Markov chain Monte Carlo algorithms for Bayesian inference of phylogeny. Ann Appl Probab 16(4):2215–2234. doi:10.2307/25449847

Page RDM (1991) Random dendrograms and null hypotheses in cladistic biogeography. Syst Zool 40(1):54–62. doi:10.2307/2992221

Ronquist F, Larget B, Huelsenbeck J P, Kadane J B, Simon D, van der Mark P (2006) Comment on "Phylogenetic MCMC algorithms are misleading on mixtures of trees". Science 312(5772):367. doi:10.1126/science.1123622 (**author reply 367. ISSN: 0036-8075**).

Sanderson MJ, McMahon MM, Steel M (2011) Terraces in phylogenetic tree space. Science 333(6041):448–450. doi:10.1126/science.1206357

Semple C, Steel M (2003) Phylogenetics. Oxford University Press, Oxford

Sleator DD, Tarjan RE, Thurston WP (1992) Short encodings of evolving structures. SIAM J Discrete Math 5(3):428–450. doi:10.1137/0405034

Spade DA, Herbei R, Kubatko LS (2014) A note on the relaxation time of two Markov chains on rooted phylogenetic tree spaces. Stati Probab Lett 84:247–252

Štefankovič D, Vigoda E (2011) Fast convergence of Markov chain Monte Carlo algorithms for phylogenetic reconstruction with homogeneous data on closely related species. SIAM J Discrete Math 25(3):1194–1211. doi:10.1137/100790550

Whidden C, Matsen FA IV (2015) Quantifying MCMC exploration of phylogenetic tree space. Syst Biol 1063–5157:1076–836X. doi:10.1093/sysbio/syv006

Whidden C, Matsen IV FA (2016) Ricci-Ollivier curvature of the rooted phylogenetic Subtree-Prune-Regraft graph. In: Proceedings of the thirteenth workshop on analytic algorithmics and combinatorics (ANALCO16), pp 106–120. url: http://epubs.siam.org/doi/abs/10.1137/1.9781611974324.6

Worobey M, Gemmel M, Teuwen DE, Haselkorn T, Kunstman K, Bunce M, Muyembe J-J, Kabongo J-MM, Kalengayi RM, Van Marck E, Gilbert MTP, Wolinsky SM (2008) Direct evidence of extensive diversity of HIV-1 in kinshasa by 1960. Nature 455(7213):661–664. doi:10.1038/nature07390

Yang Z, Rannala B (1997) Bayesian phylogenetic inference using DNA sequences: a Markov chain Monte Carlo method. Mol Biol Evol 14(7):717–724