Neighbourhoods of Phylogenetic Trees: Exact and Asymptotic Counts

A thesis submitted in partial fulfilment

of the requirements for the Degree of

Master of Science in Mathematics

at the University of Canterbury

by Jamie de Jong

2015

Contents

Acknowledgements

I would like to thank my supervisors Jeanette McLeod and Mike Steel for their ideas and support throughout my research, and for their assistance with editing my thesis. I would also like to thank Simone Linz (University of Auckland) for her input during the early stages of my research, and for reading a draft of my thesis.

This research was supported by a University of Canterbury Masters Scholarship.

Abstract

A central theme in phylogenetics is the reconstruction and analysis of evolutionary trees from a given set of data. To determine the optimal search methods for the reconstruction of trees, it is crucial to understand the size and structure of neighbourhoods of trees under tree rearrangement operations. The diameter and size of the immediate neighbourhood of a tree has been well-studied, however little is known about the number of trees at distance two, three or (more generally) k from a given tree. In this thesis we explore previous results on the size of these neighbourhoods under common tree rearrangement operations (NNI, SPR and TBR). We obtain new results concerning the number of trees at distance k from a given tree under the Robinson-Foulds (RF) metric and the Nearest Neighbour Interchange (NNI) operation, and the number of trees at distance two from a given tree under the Subtree Prune and Regraft (SPR) operation. We also obtain an exact count for the number of pairs of binary phylogenetic trees that share a first RF or NNI neighbour.

1 Introduction

Phylogenetics is the study of evolutionary relationships between species. These relationships are represented as phylogenetic trees, where the leaves correspond to extant species, and interior vertices correspond to ancestral species. A branch between two species in a tree indicates an evolutionary relationship between them (Semple and Steel, 2003; Felsenstein, 2004). Central to phylogenetics is the problem of finding the optimal tree to fit a given data set, with the aim of determining the evolutionary history of the species being studied. However the number of possible phylogenetic trees grows rapidly with the number of leaves, so for data sets with a large number of leaves, the optimal tree is commonly found by searching the set of phylogenetic trees (tree space) via tree rearrangement operations (Kubatko, 2007; Whelan and Money, 2010). Tree rearrangement operations are also used to compare phylogenetic trees, by looking at the distance (smallest number of tree rearrangement operations) between the trees. These could be trees obtained from the same data set using different search methods, or from different data sets on the same set of species (DasGupta et al., 1997a,b).

In order to effectively search tree space using tree rearrangement operations it is crucial to understand the size and structure of the neighbourhood (set of trees obtained) of a phylogenetic tree under these operations. In this thesis we investigate the size of the neighbourhoods of trees arising from three commonly used tree rearrangement operations; Nearest Neighbour Interchange (NNI), Subtree Prune and Regraft (SPR), and Tree Bisection and Reconnection (TBR) as well as the Robinson-Foulds (RF) distance. Fig. 1 shows examples of the RF, NNI and SPR distances between trees. Expressions for the number of trees at distance one, two or three from a given tree under NNI, and distance one under SPR and TBR are already known (Robinson, 1971; Allen and Steel, 2001; Humphries and Wu, 2013). We will consider each of these neighbourhoods in detail, and provide independent proofs for these expressions. In addition we provide new asymptotic expressions for the number of trees at distance k from a given tree under NNI and the RF distance, and show that unlike NNI and RF, the number of trees at distance two from a given tree under SPR is dependent on the shape of the tree, and cannot be expressed solely in terms of the number of leaves and cherries of the tree.

The literature on the structure of tree neighbourhoods and tree space has included results regarding the distribution of distances between trees (Bryant and Steel, 2009), and the smallest number of NNI operations required to reach every tree in the set (Gordon et al., 2013; Caceres et al., 2013). Bryant (2004) characterised the splits appearing in trees within a certain distance of a given tree under the four distance measures we investigate here (RF, NNI, SPR and TBR). Our work on the number of trees at distance one or two from a given tree produces an exact count for the num-

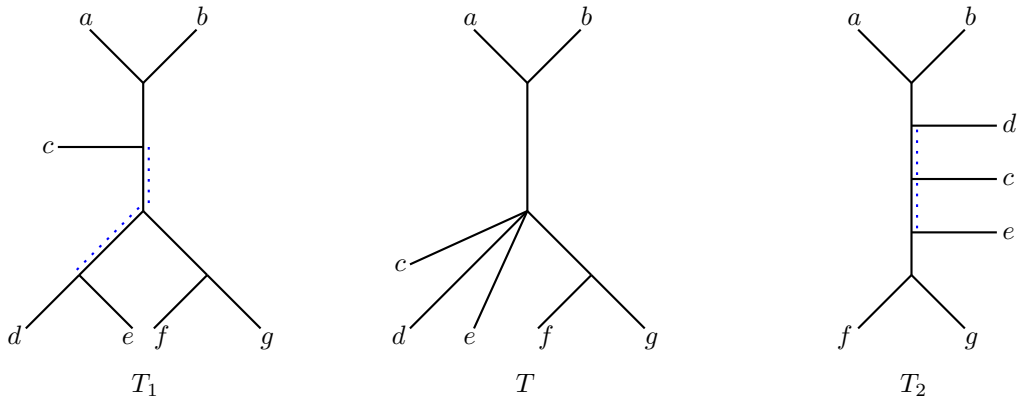

Figure 1: Here T_1 and T_2 are unrooted binary phylogenetic trees with 7 leaves. They are (i) distance two apart under the RF metric, (ii) distance two apart under the NNI metric, and (iii) distance one apart under the SPR metric. Tree T is obtained from T_1 or T_2 by contracting the two internal edges indicated by dotted lines.

ber of pairs of binary phylogenetic trees with n leaves that share a first neighbour under NNI and RF.

Unless otherwise stated results in this thesis are my own, and in all cases where results were originally stated elsewhere I have proved them independently, and without reference to the original source.

2 Definitions

A *graph* G is an ordered pair $(V(G), E(G))$ consisting of a vertex set $V(G)$ and an edge set $E(G)$. For any vertices $x, y \in V(G)$, x and y are *adjacent* if there is an edge $e \in E(G)$ such that $e = \{x, y\}$. We call x and y the *endpoints* of e, and x and e are said to be *incident*. Two distinct edges $e, f \in E(G)$ are *adjacent* if they have an endpoint in common. Edges $e, f \in E(G)$ are *parallel edges* if they have the same endpoints. An edge $f = \{x, x\}$ where $x \in V(G)$ is called a *loop*. A graph is *simple* if it has no loops or parallel edges. All of the graphs referred to in this thesis are simple.

The *degree* of a vertex $v \in V(G)$ is the number of vertices in $V(G)$ that are adjacent to v, and is denoted $deg(v)$. The Handshaking Lemma is a well-known result stating that for a graph G, $\sum_{v \in V(G)} deg(v) = 2|E(G)|$ (Bollobas, 1998).

A graph H is a *subgraph* of G if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. If $V(H) \subset V(G)$ or $E(H) \subset E(G)$ then H is a *proper subgraph* of G. A *path* P in G of length k is a subgraph of G which consists of a sequence of distinct vertices $v_0, v_1, ..., v_k$ such that for all $i \in \{0, 1, ..., k-1\}$, v_i and v_{i+1} are adjacent in P. We may also refer to P as a $(v_0{-}v_k)$-path or an $(e{-}f)$-path where $e = \{v_0, v_1\}$ and $f = \{v_{k-1}, v_k\}$. A *cycle* is a path in which the first and last vertices are the same, that is, $v_0 = v_k$. The subgraph of a graph G *induced* by the vertex set $V \subseteq V(G)$ is the subgraph with vertex set V,

and edge set $E \subseteq E(G)$ where E consists of all the edges of G that have both endpoints in V.

Two vertices $x, y \in V(G)$ are *connected* if there is an $(x-y)$-path in G. A graph G is *connected*, if all pairs of vertices $x, y \in V(G)$ are connected. A *component* of G is a maximal connected subgraph of G.

The *distance* between two vertices $x, y \in V(G)$, denoted $d_G(x, y)$, is the length of the shortest $(x-y)$-path in G. We define the *distance* between two vertex sets, $U = \{u_1, u_2, ...\}$ and $V = \{v_1, v_2, ...\}$ to be $d_G(U, V)$ where

$$d_G(U, V) = \min\{d_G(u_i, v_j) : 1 \leq i \leq |U|, 1 \leq j \leq |V|\}.$$

The diameter M of G is given by

$$M = \max\{d_G(v_i, v_j) : v_i, v_j \in V(G)\}.$$

Two graphs G and G' are *isomorphic* if there is a bijection $\sigma : V(G) \to V(G')$ such that for all pairs of vertices $x, y \in V(G)$, x and y are adjacent in G if and only if $\sigma(x)$ and $\sigma(y)$ are adjacent in G'.

2.1 Trees

A *tree* T is a connected graph containing no cycles. A *forest* is a graph whose components are trees. A tree is *rooted* if it has a distinguished root vertex, otherwise it is *unrooted*. A *leaf* of a tree T is a vertex of T that has degree one. The *leaf set* $\mathcal{L}(T) \subseteq V(T)$ of a tree T is the set of all leaves in T. Vertices of T that are not leaves, are called *internal vertices*. If an edge of T is incident to a leaf we call it a *pendant edge* of T, otherwise it is an *internal edge* of T.

A *binary tree* is a tree in which all internal vertices have degree three. A *binary phylogenetic tree* T is a tree with a bijection $\phi : X \to \mathcal{L}(T)$ where X is a set of n labels (see Fig. 1). Let $UB(n)$ be the set of all unrooted binary phylogenetic trees with n leaves. In this thesis we shall restrict our attention to unrooted binary phylogenetic trees unless otherwise stated.

A *cherry* in a tree T is a path of length two in which both end points are leaves of T. Let $UB(n, c)$ be the set of all unrooted binary phylogenetic trees with n leaves and c cherries. For example, in Fig. 1, $T_1 \in UB(7, 3)$ and $T_2 \in UB(7, 2)$, while T is not a binary tree.

For trees $T_1, T_2 \in UB(n)$, we say that T_1 and T_2 are *equal* ($T_1 = T_2$) if they are isomorphic by a map that preserves the leaf labelling.

2.2 Subtrees

A *subtree* of a graph G is a subgraph of G that is a tree. All connected subgraphs of a tree T are subtrees. The *distance* in T between a subtree T' of T and a set of vertices $V \subseteq V(T)$ is $d_T(V(T'), V)$. Throughout this thesis, we will simply write this as $d_T(T', V)$. Throughout this thesis we assume that all subtrees are proper subtrees, and have the property that if T' is a subtree of $T \in UB(n)$ then $\mathcal{L}(T') \subseteq \mathcal{L}(T)$. This ensures that T' has at least one vertex of degree two. If T' has exactly one vertex of degree two then it is a *pendant subtree*, else it is an *internal subtree*. An edge e in a tree T is *incident* to a subtree T' of T if e is incident to a vertex of degree two in T. Unless otherwise specified, we use the term 'subtree' to mean 'pendant subtree'. All subtrees in this thesis are maximal unless otherwise stated.

A tree T is a *caterpillar* if the subtree T' induced by the internal vertices of T is a path. A *balanced tree* is a tree in which all leaves are equidistant from a single vertex or edge. Fig. 2 shows a caterpillar and the two structures for a balanced tree.

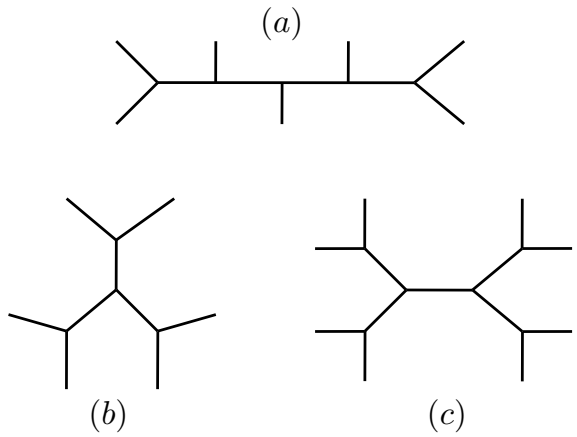

Figure 2: Examples of (a) a caterpillar and (b), (c) balanced trees.

Define $P_k(T)$ to be the number of paths of length k in T. An *internal path* P of a tree T is a path in which all vertices of P are internal vertices of T. We denote by $p_k(T)$ the number of internal paths of length k in T.

2.3 Edge and Vertex Operations

Given a tree T, if we *delete* an edge $e \in E(T)$, we obtain the forest $T \setminus e$ where $V(T \setminus e) = V(T)$ and $E(T \setminus e) = E(T) - \{e\}$. We *contract* an edge $e = \{x, y\}$ of T to obtain a new non-binary tree, denoted T/e, by deleting e and combining x and y into a single vertex w, such that all vertices adjacent to x or y in T are adjacent to w in T/e. Fig. 1 shows the tree T resulting from the

contraction of two internal edges of a tree T_1.

Let T be a tree with edge $e = \{x, y\}$. We *subdivide* e, by deleting e and inserting a vertex u and edges $e_1 = \{x, u\}$ and $e_2 = \{u, y\}$ to obtain a non-binary tree T'. We *suppress* a vertex u in a non-binary tree T where $deg(u) = 2$, by deleting u and its incident edges $e_1 = \{x, u\}$ and $e_2 = \{u, y\}$, and inserting a single edge $e' = \{x, y\}$ to obtain a tree T'. Edge subdivision and vertex suppression are inverse operations.

In this thesis, when we perform any of the operations detailed in this subsection, we assume that all edge and vertex labels in the original tree are preserved by the operation, except those explicitly deleted or inserted.

2.4 Splits

Given a set X, a *partition* of X is a set of disjoint, non-empty subsets $\{X_1, X_2, ..., X_m\}$, $m \geq 1$, such that $X = \cup_{k=1}^{m} X_k$. A partition of X is a *bipartition* if $m = 2$. Consider a tree $T \in UB(n)$. A bipartition $\{L_1, L_2\}$ of $\mathcal{L}(T)$ is a *split* if there exists an edge $e \in E(T)$ such that $T \setminus e$ has components T_1 and T_2 with $\mathcal{L}(T_1) = L_1$ and $\mathcal{L}(T_2) = L_2$. We define $S(T, e) = \{L_1, L_2\}$ as the split of T *associated with* e. A split $S(T, e)$ is *trivial* if e is a pendant edge of T. We define $\Sigma(T) = \{S(T, e) : \text{ where } e \text{ is an internal edge of } T\}$ as the set of all non-trivial splits of T. Two trees T_1 and T_2 are *equal* if and only if $\Sigma(T_1) = \Sigma(T_2)$ (Buneman, 1971).

We are now able to determine expressions for the number of internal edges of a tree $T \in UB(n)$, and for $|UB(n)|$. We provide independent proofs of these well known results (Semple and Steel, 2003).

Lemma 2.1. *Let $T \in UB(n)$, $n \geq 3$. Then T has $n - 3$ internal edges.*

Proof. We proceed by induction on the number of leaves. The only possible unrooted binary tree with three leaves has precisely one internal vertex and three edges, all of which are pendant edges. Therefore there are $n-3 = 0$ internal edges. Assume that the lemma holds for some $n \geq 3$. Consider $T \in UB(n+1)$. We delete a leaf ℓ of T and its incident pendent edge e, and suppress the resulting vertex of degree 2, to obtain a tree T'. There are two cases to consider. Either the two edges adjacent to e in T are both internal edges, or one of them is a pendant edge and the other an internal edge.

1. If both of the edges adjacent to e in T are internal edges then in T' they have been replaced with a single internal edge f. Therefore $T' \in UB(n)$. By our induction assumption, T' has $n-3$ internal edges. In order to insert e (and incident leaf ℓ) into T' at f, we subdivide f with a vertex x into the two internal edges f_1 and f_2, as well as inserting e as a pendant edge

incident to x. Therefore there are $n-3+1 = n-2$ internal edges.

2. If exactly one of the edges adjacent to e in T is a pendant edge and the other is an internal edge, then in T' they have been replaced with a single pendant edge p. Therefore $T' \in UB(n)$. Again, by the induction assumption T' has $n-3$ internal edges. In order to insert e (and incident leaf ℓ) into T' at p we subdivide p with a vertex x into an internal edge g and a pendant edge p', as well as inserting e as a pendant edge incident to x. Therefore there are $n-3+1 = n-2$ internal edges.

Therefore all unrooted binary trees on $n+1$ leaves have $n-2$ internal edges, and the induction assumption holds for all n. □

Lemma 2.2. *For all $n \in \mathbb{Z}^+$, $n \geq 3$ we have*

$$|UB(n)| = \frac{(2n-4)!}{(n-2)!2^{n-2}}.$$

Proof. We proceed by induction. When $n = 3$ there is only one unrooted binary tree, so the base case holds. Assume the lemma holds for some $n \geq 3$. Consider a tree $T \in UB(n+1)$ and label the leaves from 1 to $n+1$ in any order. If we delete the leaf labelled $n+1$ and its incident pendant edge, and suppress the resulting vertex of degree 2, we obtain a tree $T' \in UB(n)$.

Given any tree $T'' \in UB(n)$, by Lemma 2.1 there are $2n-3$ different locations at which we could insert an edge e, and incident leaf ℓ, to obtain a tree in $UB(n+1)$. It is easy to see that each of these locations produces a different tree. What is less obvious is that inserting an edge (and incident leaf) into two different trees in $UB(n)$ never produces the same tree in $UB(n+1)$. To see this, consider two distinct trees $T_1, T_2 \in UB(n)$. Let e' be an internal edge of T_1. Suppose that $S(T_1, e') = \{A, B\}$ is not a split of T_2. Insert an edge e and incident leaf ℓ into T_1 and T_2 to obtain trees $T_1', T_2' \in UB(n+1)$ respectively. Clearly $S' = \{A \cup \ell, B\}$ or $S'' = \{A, B \cup \ell\}$ is a split of T_1' (or both are). Since $S(T_1, e')$ is not a split of T_2, neither S' or S'' is a split of T_2', so $T_1' \neq T_2'$.

Therefore, by the induction assumption we have

$$\begin{aligned}|UB(n+1)| &= \frac{(2n-4)!}{(n-2)!2^{n-2}}(2n-3) \\ &= \frac{(2n-2)!}{(n-1)!2^{n-1}} \\ &= \frac{(2(n+1)-4)!}{((n+1)-2)!2^{(n+1)-2}},\end{aligned}$$

and so the assumption holds for all n.

□

2.5 Neighbourhoods

In this thesis we consider four metrics: Robinson-Foulds (RF), Nearest Neighbour Interchange (NNI), Subtree Prune and Regraft (SPR), and Tree Bisection and Reconnection (TBR), which are defined in their respective sections.

Given one of these four metrics δ_θ, $\theta \in \{\text{RF, NNI, SPR, TBR}\}$, on $UB(n)$, the k^{th} *neighbourhood* of a tree T, denoted $N_\theta^k(T)$, is given by

$$N_\theta^k(T) = \{T' \in UB(n) : \delta_\theta(T, T') = k\}.$$

A tree $T' \in N_\theta^k(T)$ is called a k^{th} *neighbour* of T. Note that T is also a k^{th} neighbour of T'.

3 Robinson-Foulds Metric

The *Robinson-Foulds (RF) distance* between two trees $T_1, T_2 \in UB(n)$ is defined by

$$\delta_{RF}(T_1, T_2) = \frac{1}{2}|\Sigma(T_1) - \Sigma(T_2)| + \frac{1}{2}|\Sigma(T_2) - \Sigma(T_1)|.$$

Alternatively the Robinson-Foulds distance between T_1 and T_2 can be seen as the minimum m for which there exist $E_1 \subseteq E(T_1)$ and $E_2 \subseteq E(T_2)$ where $|E_1| = |E_2| = m$, such that $T_1/E_1 = T_2/E_2$. This is illustrated in Fig. 1, where $\delta_{RF}(T_1, T_2) = 2$.

The k^{th} *RF neighbourhood* of a tree $T \in UB(n)$ is the set of trees in $UB(n)$ that are exactly RF distance k from T. So in terms of edge contraction, this neighbourhood consists of all trees $T' \in UB(n)$ such that the minimum j for which we could contract j edges of T and j edges of T' and obtain the same (non-binary) tree, is k.

The RF distance was originally introduced by Bourque (1978) and was generalised by Robinson and Foulds (1981). Unlike the metrics induced by NNI, SPR, and TBR that we will see in later sections, the RF distance between two trees is computationally easy to calculate. (Day (1985) provided a linear-time algorithm.) Much of the literature on the RF distance has focused on calculating the RF distance between two trees, and on the distribution of the distances between trees. There has been little work on the size of the neighbourhood of a tree under RF. Bryant and Steel (2009) gave a polynomial-time algorithm for finding the distribution of trees around a given tree T, and showed that this distribution can be approximated by a Poisson distribution determined by the proportion of leaves of T that are in cherries. In this section we investigate the size of the first, second and k^{th} RF neighbourhood. Our main result is an asymptotic expression for the size of the k^{th} RF neighbourhood, which is presented in Theorem 3.1.

Theorem 3.1. *Let $T \in UB(n)$ $(n \geq 4)$. Then for each fixed $k \in \mathbb{Z}^+$ there exists a constant $C_{T,k}$ such that,*

$$N^k_{RF}(T) = \frac{2^k n^k}{k!}\left(1 + C_{T,k}n^{-1} + O(n^{-2})\right) \tag{1}$$

where

$$-\frac{5k^2+7k}{4} \leq C_{T,k} \leq 4k^2 - 7k.$$

The proof of Theorem 3.1 comprises two steps. First, given a tree $T \in UB(n)$ we determine the number of binary phylogenetic trees whose splits differ from $\Sigma(T)$ by exactly the k splits associated with a given subset of k internal edges of T. Then we determine the number of subsets of k internal edges in T. We consider three cases:

1. The k edges are pairwise non-adjacent.
2. Exactly two of the k edges are adjacent.
3. More than two of the k edges are adjacent.

The term of order n^k in Equation (1) is completely determined by Case 1 above, while the term of order n^{k-1} is determined by Cases 1 and 2. We show that all other possibilities for the k edges, (covered by Case 3) only contribute to terms of order n^{k-2} or lower.

Neighbours with Different Splits over k Given Edges

Let Σ_k be a given set of k splits of $T \in UB(n)$ $(k \geq 1)$. We define

$$\Delta(T, \Sigma_k) = |\{T' \in UB(n) : (\Sigma(T) - \Sigma_k) \subset \Sigma(T')\}|,$$

as the number of trees containing the splits $\Sigma(T) - \Sigma_k$; and

$$\mathring{\Delta}(T, \Sigma_k) = |\{T' \in UB(n) : (\Sigma(T) \cap \Sigma(T')) = \Sigma(T) - \Sigma_k\}|,$$

as the number of trees containing the splits $\Sigma(T) - \Sigma_k$, and no other splits of T.

In Lemma 3.2 we obtain an expression for $\Delta(T, \Sigma_k)$ and show that once T and Σ_k are specified, $\mathring{\Delta}(T, \Sigma_k)$ is independent of n.

Lemma 3.2. *Let $T \in UB(n)$ ($n \geq 4$), let e_1, ..., e_k ($1 \leq k \leq n-3$) be distinct internal edges of T, and let Σ_k be the set of k splits of T associated with these edges. Define F to be the subgraph of T consisting of the edges e_1,..., e_k. Then*

(i)

$$\Delta(T, \Sigma_k) = \prod_{m=1}^{k} \left(\frac{(2m+2)!}{(m+1)!2^{m+1}} \right)^{C_m}$$

where C_m is the number of components with exactly m edges of F, and

(ii) once T and Σ_k are specified, $\mathring{\Delta}(T, \Sigma_k)$ is constant with respect to n.

Proof.

(i) We determine the number of unrooted binary trees with at least the splits $\Sigma(T) - \Sigma_k$ by considering the edge contraction definition of RF. Let C be a component of F with m edges, and let A be the (possibly internal) subtree of T consisting of the corresponding m edges and their adjacent edges in T. Then A has $m+3$ leaves. Contracting the m internal edges of A produces a tree T_m with a single internal vertex of degree $m+3$. The set of unrooted binary trees where it is possible to contract m internal edges and obtain T_m is $UB(m+3)$. By Lemma 2.2

$$|UB(m+3)| = \frac{(2(m+3)-4)!}{((m+3)-2)!2^{(m+3)-2}} = \frac{(2m+2)!}{(m+1)!2^{m+1}}.$$

The choice of tree for one component does not restrict the number of choices for any other component, so applying the same principle to every component in F, we obtain

$$\Delta(T, \Sigma_k) = \prod_{m=1}^{k} \left(\frac{(2m+2)!}{(m+1)!2^{m+1}} \right)^{C_m}.$$

(ii) This is similar to (i), except that none of the splits in Σ_k can be in any of the trees in $\mathring{\Delta}(T, \Sigma_k)$. We consider again A and T_m. Some number of the $\frac{(2m+2)!}{(m+1)!2^{m+1}}$ trees in $UB(m+3)$ have some splits in common with A, and hence are not counted by $\mathring{\Delta}(T, \Sigma_k)$. However, the number of such trees is dependent on the shape and size of A, which itself depends on the choice of the k edges of T and not on the shape or number of leaves of T. Hence given T and Σ_k, $\mathring{\Delta}(T, \Sigma_k)$ is constant with respect to n.

□

We can use Lemma 3.2 to find expressions for the sizes of the first and second RF neighbourhood of a tree $T \in UB(n)$. However, first we need to know how many pairs of adjacent and non-adjacent edges there are in T.

Lemma 3.3. *Let $T \in UB(n, c)$ with $n \geq 4$. Then T has*

1. *$n - 2$ internal vertices,*
2. *c internal vertices that have exactly one incident internal edge,*
3. *$n - 2c$ internal vertices that have exactly two incident internal edges, and*
4. *$c - 2$ internal vertices that have three incident internal edges.*

Proof.

1. By Lemma 2.1, T has $n - 3$ internal edges, so in total T has $2n - 3$ edges. Therefore, by the Handshaking Lemma, the sum of the degrees of all the vertices in T is $2(2n - 3) = 4n - 6$. Since there are n leaves, the sum of the degrees of the internal vertices is $4n - 6 - n = 3(n - 2)$. Every internal vertex of T has degree three, so there are $n - 2$ internal vertices.

2. If an internal vertex has only one incident internal edge, then it has two incident pendant edges. These pendant edges are adjacent to each other, and therefore form part of a cherry. Since T has c cherries, it has at most c internal vertices with exactly one incident internal edges.

   A cherry in T has an internal vertex with at most one incident internal edge in T. For $n \geq 4$, T has no internal vertices that are incident to zero internal edges. Therefore, every cherry in T has an internal vertex with exactly one incident internal edge in T. Therefore, there are c internal vertices in T that have exactly one incident internal edge.

3. Given that there are c cherries, and each cherry contains two leaves of T, there are $n - 2c$ leaves in T that are not part of a cherry. Each of these leaves is incident to a pendant edge. These pendant edges must each be adjacent to two internal edges in T, else they would be part of a cherry. Therefore there are exactly $n - 2c$ internal vertices in T with two incident internal edges.

4. Since $n \geq 4$ there are no internal vertices in T that are incident to zero internal edges. Therefore all remaining internal vertices of T have three incident internal edges. There are

   $$(n - 2) - c - (n - 2c) = c - 2$$

   of these vertices in T.

□

Corollary 3.4. *A tree $T \in UB(n, c)$ $(n \geq 4)$ has $n + c - 6$ pairs of adjacent internal edges and $\frac{1}{2}(n^2 - 9n + 24) - c$ pairs of non-adjacent internal edges.*

Proof. For a pair of internal edges to be adjacent, they must both be incident to the same internal vertex. Therefore an internal vertex with two incident internal edges will result in one pair of adjacent internal edges, an internal vertex with three incident internal edges will result in three pairs, and an internal vertex with less than two incident internal edges will result in none. Therefore, by Lemma 3.3, T has

$$(n - 2c) + 3(c - 2) = n + c - 6$$

pairs of adjacent internal edges.

All remaining pairs of internal edges must be non-adjacent. In total, T has $n-3$ internal edges, and the number of pairs of these edges is

$$\sum_{k=1}^{n-4} k = \frac{1}{2}(n - 4)(n - 3).$$

Therefore the number of pairs of non-adjacent internal edges in T is

$$\frac{1}{2}(n - 4)(n - 3) - (n + c - 6) = \frac{1}{2}(n^2 - 9n + 24) - c.$$

□

Lemma 3.5. *Let $T \in UB(n, c)$ $(n \geq 3)$ and suppose that T has c cherries. Then*

(i) $|N_{RF}(T)| = 2(n - 3)$, *and*

(ii) $|N^2_{RF}(T)| = 2n^2 - 8n + 6c - 12$.

Proof.

(i) For $n = 3$, T has no internal edges, and so the result is trivially true. Now assume that $n \geq 4$. Then

$$|N_{RF}(T)| = \sum_{\substack{\Sigma_1 \subseteq \Sigma(T) \\ |\Sigma_1|=1}} \mathring{\Delta}(T, \Sigma_1).$$

Let $\Sigma_1 = \{S(T, e)\}$ where e is an internal edge of T. By Lemma 3.2, $\Delta(T, \Sigma_1) = 3$, so there are three trees in $UB(n)$ with the splits $\Sigma(T) - \Sigma_1$. However, one of these trees is T, so there are two trees T' and T'' in $UB(n)$, aside from T, with the splits $\Sigma(T) - \Sigma_1$. Since $T' \neq T$ and $T'' \neq T$, $S(T, e_1)$ is not a split of T' or T''. Therefore $\mathring{\Delta}(T, \Sigma_1) = 2$. Hence if we sum over all internal edges of T, we obtain $|N_{RF}(T)| = 2(n - 3)$, by Lemma 2.1.

(ii) For $n = 3$ and $n = 4$, T has fewer than two internal edges, and so the result is trivially true. Now assume that $n \geq 5$. Similarly to (i),

$$|N^2_{RF}(T)| = \sum_{\substack{\Sigma_2 \subseteq \Sigma(T) \\ |\Sigma_2|=2}} \mathring{\Delta}(T, \Sigma_2).$$

Let $\Sigma_2 = \{S(T, e_1), S(T, e_2)\}$, where e_1 and e_2 are internal edges of T. Similarly to the proof of (i), the set of trees counted by $\Delta(T, \Sigma_2)$ includes some trees with one or more of the splits in Σ_2 in common with T. So to obtain $\mathring{\Delta}(T, \Sigma_2)$, we subtract from $\Delta(T, \Sigma_2)$ the number of trees in $UB(n)$ that have exactly one split different to T associated with either e_1 or e_2, or the same splits as T. Hence

$$\mathring{\Delta}(T, \Sigma_2) = \Delta(T, \Sigma_2) - \mathring{\Delta}(T, S(T, e_1)) - \mathring{\Delta}(T, S(T, e_2)) - 1.$$

Therefore, by Lemma 3.2, if e_1 and e_2 are not adjacent, $\mathring{\Delta}(T, \Sigma_2) = 9 - 5 = 4$, and if e_1 and e_2 are adjacent then $\mathring{\Delta}(T, \Sigma_2) = 15 - 5 = 10$.

To see that different choices of the edges e_1 and e_2 cannot produce any duplicate trees, consider trees T' and T'' which are RF distance two from T. Suppose that the splits that differ between T and T' are associated with distinct edges e and e', while the splits that differ between T and T'' are associated with distinct edges f and f', $\{e, e'\} \neq \{f, f'\}$. Without loss of generality, we assume that $e \notin \{f, f'\}$. Then $S(T'', e) = S(T, e) \neq S(T', e)$, and so $T \neq T'$.

Hence, by Corollary 3.4,

$$N^2_{RF}(T) = 10(n + c - 6) + 4\left(\frac{1}{2}(n^2 - 9n + 24) - c\right) = 2n^2 - 8n + 6c - 12.$$

□

We can now determine the size of $\mathring{\Delta}(T, \Sigma'_k)$ for a tree T where Σ'_k is the set of splits associated with k pairwise non-adjacent edges of T.

Corollary 3.6. *Let $T \in UB(n)$ $(n \geq 4)$ and let Σ'_k $(1 \leq k \leq n - 3)$ be the set of splits associated with distinct, pairwise non-adjacent internal edges e_1, ..., e_k of T. Then $\mathring{\Delta}(T, \Sigma'_k) = 2^k$.*

Proof. By Lemma 3.2, $\Delta(T, \Sigma'_k) = 3^k$. Similar to the proof of Lemma 3.5, the set of trees counted by $\Delta(T, \Sigma'_k)$ includes some trees that have a subset of the splits Σ'_k in addition to the splits $\Sigma(T) - \Sigma'_k$. As in the proof of Lemma 3.5 (i), for each edge e_j, $j = 1, ..., k$, there are three trees with the splits $\Sigma(T) - S(T, e_j)$, however one of these trees also has the split $S(T, e_j)$. Hence there are precisely two trees with all of the splits in $\Sigma(T)$ except $S(T, e_j)$. Therefore, taking the product over all k edges we have $\mathring{\Delta}(T, \Sigma'_k) = 2^k$. □

The Number of Subsets of k Internal Edges

Lemma 3.7. *Let $T \in UB(n)$ $(n \geq 4)$. Then*

(i) The number of sets of k distinct, pairwise non-adjacent internal edges e_1,..., e_k $(1 \leq k \leq n-3)$ in T, denoted $A_{T,k}$, satisfies

$$\frac{1}{k!}n^k - \frac{k(5k+1)}{2k!}n^{k-1} + O(n^{k-2}) \leq A_{T,k} \leq \frac{1}{k!}n^k - \frac{k(k+2)}{k!}n^{k-1} + O(n^{k-2}).$$

(ii) The number of sets of k distinct internal edges e_1,..., e_k $(2 \leq k \leq n-3)$ in T where exactly two edges are adjacent, denoted $B_{T,k}$, satisfies

$$\frac{1}{2(k-2)!}n^{k-1} + O(n^{k-2}) \leq B_{T,k} \leq \frac{2}{(k-2)!}n^{k-1} + O(n^{k-2}).$$

(iii) The number of sets of k distinct internal edges e_1,..., e_k $(3 \leq k \leq n-3)$ in T where more than two edges are adjacent, is $O(n^{k-2})$.

Proof.

(i) We calculate the bounds by considering the best and worst case scenarios for the choice of each edge. There are $n-3$ choices for the first edge e_1. There are at most $(n-3)-2$ choices for e_2 (this can occur when e_1 has exactly one adjacent internal edge in T). Then there are at most $(n-3)-4$ choices for e_3 (this can occur when e_1 and e_2 each have exactly one adjacent internal edge in T), and so on. Therefore

$$\begin{aligned} A_{T,k} &\leq \frac{1}{k!}(n-3)(n-3-2)(n-3-2(2))\cdots(n-3-2(k-1)) \\ &= \frac{1}{k!}n^k - \frac{1}{k!}n^{k-1}\sum_{i=0}^{k-1}(3+2i) + O(n^{k-2}) \\ &= \frac{1}{k!}n^k - \frac{k(k+2)}{k!}n^{k-1} + O(n^{k-2}). \end{aligned}$$

On the other hand, there are at least $(n-3)-5$ choices for e_2 (this can occur when e_1 has four adjacent internal edges in T). Then there are at least $(n-3)-10$ choices for e_3 (this can occur when e_1 and e_2 each have four adjacent internal edges in T), and so on. Therefore

$$\begin{aligned} A_{T,k} &\geq \frac{1}{k!}(n-3)(n-3-5)(n-3-5(2))\cdots(n-3-5(k-1)) \\ &= \frac{1}{k!}n^k - \frac{1}{k!}n^{k-1}\sum_{i=0}^{k-1}(3+5i) + O(n^{k-2}) \\ &= \frac{1}{k!}n^k - \frac{k(5k+1)}{2k!}n^{k-1} + O(n^{k-2}). \end{aligned}$$

(ii) We will prove this in the same way as (i), assuming without loss of generality that e_1 and e_2 are the adjacent pair of edges. There are $n-3$ choices for e_1. There are at most four choices

for e_2 (this can occur if e_1 has four adjacent internal edges in T). For e_3 there are at most $(n-3)-3$ choices (this can occur if e_1 and e_2 each have two adjacent pendant edges in T). The remaining edges follow in the same way as in (1). Therefore

$$\begin{aligned} B_{T,k} &\leq \frac{4}{2(k-2)!}(n-3)(n-6)(n-6-2(1))...(n-6-2(k-3)) \\ &= \frac{2}{(k-2)!}n^{k-1} + O(n^{k-2}). \end{aligned}$$

On the other hand, there is at least one possible choice for e_2 (this can occur if e_1 has exactly one adjacent internal edge in T). For e_3 there are at least $(n-3)-7$ choices (this can occur if e_1 and e_2 each have no adjacent pendant edges in T). The remaining edges are chosen in the same way as in (1). Hence

$$\begin{aligned} B_{T,k} &\geq \frac{1}{2(k-2)!}(n-3)(n-10)(n-10-5(1))...(n-10-5(k-3)) \\ &= \frac{1}{2(k-2)!}n^{k-1} + O(n^{k-2}). \end{aligned}$$

(iii) Let F be the subgraph of T consisting of the edges e_1,..., e_k. Then F has $m \leq k-2$ components. Suppose we first choose m internal edges of T corresponding to one edge in each component of F. By (i) the number of such choices is $O(n^m)$, as each of these edges will contribute a linear factor to the total number of ways of choosing the k edges. However, the remaining $k-m \geq 2$ edges can be chosen in such a way that we always choose an edge adjacent to at least one of those already chosen. The number of these choices depends only on the number and location of the edges already chosen, and not on n. Hence the number of possible sets is $O(n^m)$ where $m \leq k-2$.

□

Note that in the proof of Lemma 3.7, it may not be possible to maximise (or minimise) the number of choices for each individual edge in T, however this is not a problem as we only require bounds on the number of choices of the k edges of T.

We now know the number of binary phylogenetic trees whose splits differ from those of $T \in UB(n)$ by exactly k splits over a given set of k edges, and the number of subsets of k internal edges. Combining this information, we can prove Theorem 3.1.

Proof of Theorem 3.1

Proof. We break down the calculation of the size of the k^{th} RF neighbourhood of T into two steps. We consider how many trees there are whose splits differ from those of T by exactly the k splits

corresponding to a given set of k distinct internal edges of T. Then we consider how many ways these k edges can be chosen in T. By Lemma 3.2, given T and a set of k distinct internal edges of T with associated split set Σ_k, the number of trees with splits $\Sigma(T) - \Sigma_k$ and none of the splits in Σ_k, is independent of n. Hence the only factor dependent on the size of n is the number of ways of choosing the k edges in T.

By Lemma 3.7, when we count the number of ways of choosing k distinct internal edges of T, the case where the k edges are pairwise non-adjacent (Case 1 from the beginning of this section) gives a term of order n^k and a term of order n^{k-1}. The case where exactly two of the k edges are adjacent (Case 2) produces a term of order n^{k-1}, but does not have a term of order n^k. If more than two of the k edges are adjacent (Case 3) then the highest order term is $O(n^{k-2})$.

Now we consider how many trees there are whose splits differ from those of T by exactly the k splits corresponding to a given set of k distinct internal edges of T. From the information above, the only two cases we need to consider are those where the k edges are pairwise non-adjacent, or exactly two of the k edges are adjacent. By Corollary 3.6, the case where all edges are pairwise non-adjacent produces 2^k k^{th} RF neighbours with splits that differ from the splits of T over precisely the k given internal edges. In the case where exactly two edges are adjacent, the $k-1$ pairwise non-adjacent edges give 2^{k-2} neighbours, by Corollary 3.6. The adjacent pair result in 10 neighbours, by the proof of Lemma 3.5 (ii). Hence in total there are $10 \cdot 2^{k-2}$ neighbours. Therefore, by Lemma 3.7,

$$\begin{aligned} |N^k_{RF}(T)| &\geq \left(\frac{1}{k!}n^k - \frac{k(5k+1)}{2k!}n^{k-1}\right)2^k + 10\left(\frac{1}{2(k-2)!}n^{k-1}\right)2^{k-2} + O(n^{k-2}) \\ &= \frac{2^k}{k!}n^k - \frac{5k^2+7k}{4k!}2^k n^{k-1} + O(n^{k-2}). \end{aligned}$$

$$\begin{aligned} |N^k_{RF}(T)| &\leq \left(\frac{1}{k!}n^k - \frac{k(k+2)}{k!}n^{k-1}\right)2^k + 10\left(\frac{2}{(k-2)!}n^{k-1}\right)2^{k-2} + O(n^{k-2}) \\ &= \frac{2^k}{k!}n^k + \frac{4k^2-7k}{k!}2^k n^{k-1} + O(n^{k-2}). \end{aligned}$$

□

4 Nearest Neighbour Interchange

In this section we provide proofs for the expressions for the size of the first and second NNI neighbourhoods of an unrooted binary tree, originally found by Robinson (1971). We extend Robinson

(1971)'s result for the size of the third NNI neighbourhood by finding an explicit expression in terms of the number of leaves, cherries and internal paths of length three in the tree. These results were proved independently, without reference to the original proofs. Finally we provide a new asymptotic expression for the size of the k^{th} NNI neighbourhood of an unrooted binary tree.

Let $T \in UB(n)$ and let $e = \{x, y\}$ be an interior edge of T. Let A_1 and A_3 be subtrees of T that are distance one from e and distance three apart (see Fig. 3). Then A_1 and A_3 are *swappable* across e. Let vertex z_1 adjacent to x be the root of A_1, and z_3 adjacent to y be the root of A_3. A *nearest neighbour interchange* (NNI) on T is performed by deleting the edges $\{x, z_1\}$ and $\{y, z_3\}$, and inserting edges $\{x, z_3\}$ and $\{y, z_1\}$. We will also refer to this process as *swapping* the subtrees A_1 and A_3 *across* e. The resulting tree is a *first NNI neighbour* of T. To make it clear which edge of a tree T two subtrees are swapped across in an NNI operation on T, we will refer to such an operation as an *NNI operation on edge e in T*.

The two distinct first NNI neighbours resulting from an NNI operation on edge e in T can be seen in Fig. 3. We have four subtrees A_1, A_2, A_3 and A_4 that are all distance one from e. To obtain T' we swap subtrees A_2 and A_3, and to obtain T'' we swap subtrees A_2 and A_4. Note that swapping subtrees A_1 and A_4 produces a tree isomorphic to T'. Although there are four different pairs of subtrees that could be swapped across e, there are only two distinct neighbours that can be obtained from NNI operations on e.

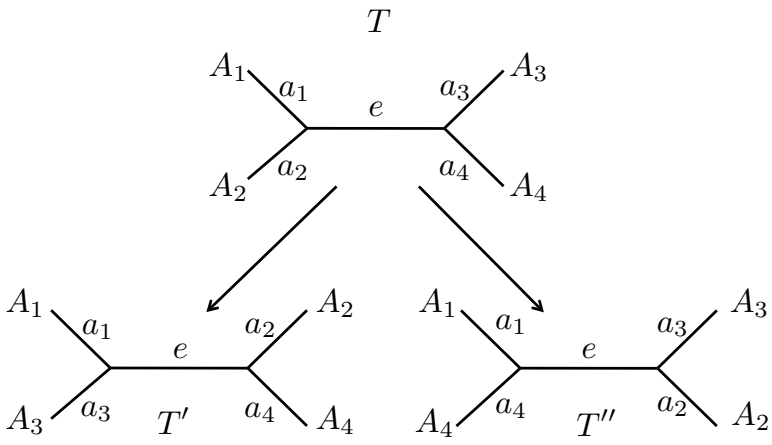

Figure 3: The two first NNI neighbours of T resulting from an NNI operation on the edge e.

We see that in T' and T'', all four subtrees A_1, A_2, A_3 and A_4 are distance one from e. Given a labelling of the edges of the original tree T, we preserve this labelling by assigning the label a_i to the edge incident to subtree A_i, in T and in the two first NNI neighbours of T resulting from an NNI operation on edge e. Note that T can also be obtained from T' by an NNI operation, which we

call the *inverse* of the operation used to obtain T' from T.

Consider a graph G in which each vertex represents a tree in $UB(n)$ and there is an edge between the vertices representing trees T_1 and T_2 if they are first NNI neighbours. The *NNI distance* between T_1 and T_2, $\delta_{NNI}(T_1, T_2)$, is the distance between the two vertices representing trees T_1 and T_2 in G.

Throughout this section we will consider the trees resulting from a series of NNI operations beginning with a tree $T \in UB(n)$. Let $NNI(T; e_1, e_2, ..., e_k) \subseteq \cup_{j=0}^{k} N_{NNI}^{j}(T)$ be the set of trees that can be obtained by performing an NNI operation on internal edge e_1 in T to give T_1, followed by an NNI operation on internal edge e_2 in T_1 to give T_2, and so on until we have completed k NNI operations. Note that if $T' \in NNI(T; e_1, ..., e_k)$, T' is not necessarily a k^{th} NNI neighbour of T. It may instead be a j^{th} NNI neighbour of T for some $j < k$ ($j \in \mathbb{N}$).

Robinson (1971) determined the size of the first and second NNI neighbourhoods of any unrooted binary tree, and found an upper bound on the size of the third NNI neighbourhood. In this section we independently investigate each of these three neighbourhoods, and obtain an explicit expression for the third NNI neighbourhood of a tree in terms of the number of leaves, cherries and internal paths of length three in the tree. Robinson (1971) also gave an upper bound for the size of the k^{th} NNI neighbourhood of a tree T in terms of the size of the $(k-1)^{th}$ NNI neighbourhood of T. Our main result for this section is the asymptotic expression for the size of the k^{th} NNI neighbourhood of a binary tree given in Theorem 4.1. The proof of this theorem appears in Section 4.4, although it relies on many of the results in Sections 4.1-4.3 pertaining to the sizes of the first, second and third NNI neighbourhoods.

Theorem 4.1. *Let $T \in UB(n)$ ($n \geq 4$). Then for each fixed $k \in \mathbb{Z}^+$ there exists a constant $D_{T,k}$ such that,*

$$|N_{NNI}^{k}(T)| = \frac{2^k n^k}{k!}\left(1 + D_{T,k}n^{-1} + O(n^{-2})\right) \tag{2}$$

where

$$-\frac{3k(k+1)}{2} \leq D_{T,k} \leq 3k(k-2).$$

As mentioned previously, tree rearrangement operations are also used to compare trees produced by different tree reconstruction methods, or trees obtained from different data sets. This can be achieved by determining the NNI distance (smallest number of operations) between the two trees. DasGupta et al. (1997b) showed that the problem of computing the NNI distance between two trees in $UB(n)$ is NP-complete. Culik and Wood (1982) found an upper bound of $4n\log(n)$ on the NNI

distance between two trees in $UB(n)$, which was later improved to $n\log(n)$ by Li et al. (1996).

It is also useful to understand the structure of $UB(n)$, and the first and second NNI neighbourhoods of a tree (e.g. how the first NNI neighbours of a tree relate to each other). A *walk* in a graph G is a sequence of vertices and edges, in which the vertices are not necessarily distinct. Consider a graph G in which each vertex represents a tree in $UB(n)$ and there is an edge between the vertices representing trees T_1 and T_2 if they are first NNI neighbours. Bryant (2008) posed the question; what is the length of the shortest walk that visits every vertex of G? Gordon et al. (2013) provided a constructive proof that this walk is a Hamiltonian path (a path that visits every vertex of G exactly once). So by a series of NNI operations beginning from a tree $T \in UB(n)$, it is possible to visit each tree in $UB(n)$ exactly once. We refer to this series of NNI operations as an *NNI walk*. In Section 5 we investigate the structure of $UB(n)$ by determining the number of pairs of trees that share a first NNI neighbour (the number of pairs of trees that are within NNI distance two of each other).

4.1 First Neighbourhood

Determining the size of the first NNI neighbourhood of a tree is not simply a matter of counting all of the NNI operations that could be performed on the tree. We need to consider whether or not it is possible for two different NNI operations on T to produce the same tree. The following theorem is due to Robinson (1971).

Theorem 4.2. *Let $T \in UB(n)$ $(n \geq 3)$. Then $|N_{NNI}(T)| = 2(n-3)$.*

Our proof requires the following lemma, which shows how the non-trivial splits of two trees that are first NNI neighbours compare.

Lemma 4.3. *Let $T \in UB(n)$ $(n \geq 4)$ and $T' = NNI(T; e)$ where e is an internal edge of T. Then*

$$|\Sigma(T) - \Sigma(T')| = |\Sigma(T') - \Sigma(T)| = 1.$$

Furthermore $\Sigma(T) - \Sigma(T') = \{S(T, e)\}$, and for all internal edges $e' \neq e$ in T, $S(T', e') = S(T, e')$.

Proof. Note that $|\Sigma(T)| = |\Sigma(T')|$ as $T, T' \in UB(n)$. Let the subtrees distance one from e in T be A, B, C and D, where $d_T(A, B) = 2$. We have $S(T, e) = \{\mathcal{L}(A) \cup \mathcal{L}(B), \mathcal{L}(C) \cup \mathcal{L}(D)\}$. Either A or B is one of the two subtrees that are swapped by the NNI operation, so $d_{T'}(A, B) = 3$, and $\mathcal{L}(A)$ and $\mathcal{L}(B)$ are in different parts of $S(T', e)$. Hence $S(T, e) \neq S(T', e)$.

Suppose there exists an internal edge e' of T, such that $e' \neq e$. Let $S(T, e') = \{L_1, L_2\}$. In T, either e' is adjacent to e, or e' is in one of the subtrees A, B, C or D. Therefore either L_1 or L_2 is a subset of the leaves in one of the subtrees A, B, C or D. Since A, B, C and D are the four subtrees of T'

that are distance one from e, $S(T', e') = S(T, e')$.

Therefore $\Sigma(T) - \Sigma(T') = \{S(T, e)\}$ and $\Sigma(T') - \Sigma(T) = \{S(T', e)\}$. Hence

$$|\Sigma(T) - \Sigma(T')| = |\Sigma(T') - \Sigma(T)| = 1.$$

□

Proof of Theorem 4.2

Proof. If $n = 3$ then T has no internal edges, so the result is trivially true. Assume that $n \geq 4$. By the definition of an NNI operation, there are two distinct first NNI neighbours of T resulting from an NNI operation on an internal edge e in T. By Lemma 2.1, T has $n - 3$ internal edges. If we perform NNI operations on each of these edges we obtain $2(n - 3)$ first NNI neighbours, provided that operations on two different internal edges of T do not produce the same tree.

Let $T_1 \in NNI(T; e)$ and $T_2 \in NNI(T; e')$ where e and e' are internal edges in T, and $e \neq e'$. By Lemma 4.3, $S(T_1, e) \neq S(T, e)$, but $S(T_2, e) = S(T, e)$. Hence $S(T_1, e) \neq S(T_2, e)$ and so $T_1 \neq T_2$. Therefore there are $2(n - 3)$ distinct first NNI neighbours of T. □

4.2 Second Neighbourhood

In this subsection we independently prove the following result due to Robinson (1971) for the size of the second NNI neighbourhood. Recall that $UB(n, c)$ is the set of unrooted binary trees with n leaves and c cherries.

Theorem 4.4. *Let $T \in UB(n, c)$ $(n \geq 3)$. Then*

$$|N^2_{NNI}(T)| = 2n^2 - 10n + 4c.$$

Before proving Theorem 4.4, we present several general results regarding trees obtained by a series of k NNI operations. These results will help to determine exactly when NNI operations over different sets of edges produce the same tree, and are used to determine the size of the second, third and k^{th} neighbourhoods of a tree. In Lemma 4.3, we saw the impact of a single NNI operation on the non-trivial splits of a tree. Now we compare the non-trivial splits of trees that are k^{th} NNI neighbours.

Lemma 4.5. *Let $T \in UB(n)$ $(n \geq 4)$, and let e_1, ..., e_k $(k \geq 1)$ be internal edges of T such that there exists e_m $(1 \leq m \leq k)$ for which $e_m \notin \{e_1, ..., e_{m-1}, e_{m+1}, ..., e_k\}$. Let $T' \in NNI(T; e_1, ..., e_k)$. Then $S(T, e_m)$ is not a split of T'. Furthermore, if e' is an internal edge of T and $e' \notin \{e_1, ..., e_k\}$, then $S(T', e') = S(T, e')$.*

Proof. Let $T_{m-1} \in NNI(T; e_1, ..., e_{m-1})$ such that $T' \in NNI(T_{m-1}; e_m, ..., e_k)$. Since $e_m \notin \{e_1, ..., e_{m-1}\}$, $S(T_{m-1}, e_m) = S(T, e_m)$ by Lemma 4.3. Let A and B be two subtrees in T_{m-1} that are distance one from e_m, where $d(A, B) = 2$. Then

$$S(T_{m-1}, e_m) = S(T, e_m) = \{\mathcal{L}(A) \cup \mathcal{L}(B), \mathcal{L}(T) - (\mathcal{L}(A) \cup \mathcal{L}(B))\}.$$

The NNI operation over edge e_m in T_{m-1} swaps either A or B with one of the other two subtrees that are distance one from e. Let $T_m \in NNI(T_{m-1}; e_m)$ such that $T' \in NNI(T_m; e_{m+1}, ..., e_k)$. Then $\mathcal{L}(A)$ and $\mathcal{L}(B)$ are in different parts of $S(T_m, e_m)$. Since $e_m \notin \{e_{m+1}, ..., e_k\}$, $S(T', e_m) = S(T_m, e_m)$ by Lemma 4.3. Hence all leaves in A are in a different component of $T' \setminus e_m$ to the leaves of B. Therefore $S(T, e_m)$ is not a split of T'.

Let e' be an internal edge of T, $e' \notin \{e_1, ..., e_k\}$. To see that $S(T', e') = S(T, e')$, consider trees T'_1, ..., T'_{k-1} where $T'_1 \in NNI(T; e_1)$, $T'_2 \in NNI(T'_1; e_2)$, ..., $T'_{k-1} \in NNI(T'_{k-2}; e_{k-1})$, and $T' \in NNI(T'_{k-1}; e_k)$. By Lemma 4.3,

$$S(T', e') = S(T'_{k-1}, e') = \cdots = S(T'_1, e') = S(T, e').$$

□

Corollary 4.6. *Let $T \in UB(n)$ ($n \geq 4$) and let e_1, ..., e_k ($k \geq 1$) be internal edges of T such that there exists e_m ($1 \leq m \leq k$) for which $e_m \notin \{e_1, ..., e_{m-1}, e_{m+1}, ..., e_k\}$. Let $P = NNI(T; e_1, ..., e_j)$ and $Q = NNI(T; e_{j+1}, ..., e_k)$ where $1 \leq j \leq k$. Then*

$$P \cap Q = \emptyset.$$

Proof. Without loss of generality suppose that $1 \leq m \leq j$. By Lemma 4.5, $S(T, e_m)$ is not a split of any of the trees in P.

Also by Lemma 4.5, for all $T' \in Q$, $S(T', e_m) = S(T, e_m)$, since $e_m \notin \{e_{j+1}, ..., e_k\}$. Therefore, since two trees are equal if and only if they have the same set of splits, we have $P \cap Q = \emptyset$. □

Now we consider whether or not two consecutive operations occurring on the same edge of a tree has any impact on the neighbours of that tree.

Lemma 4.7. *Let $T \in UB(n)$ ($n \geq 4$), and let e_1,..., e_k ($k \geq 2$) be internal edges of T. Suppose there exists an m ($1 \leq m \leq k-1$) for which $e_m = e_{m+1}$. If*

$$P = NNI(T; e_1, ..., e_m, e_{m+1}, e_{m+2}, ..., e_k),$$

then $P \cap N^k_{NNI}(T) = \emptyset$.

Proof. Let $T_{m-1} \in NNI(T; e_1, ..., e_{m-1})$ and let $T_m, T'_m \in NNI(T_{m-1}; e_m)$, $T_m \neq T'_m$. Now suppose we perform an operation on edge $e_{m+1} = e_m$ in T_m to obtain a tree T_{m+1}. Let $T' \in NNI(T_{m+1}; e_{m+2}, ..., e_k)$.

First, suppose the operation on edge e_{m+1} is the inverse of the operation on edge e_m. Then $T_{m+1} = T_{m-1}$. Hence

$$T' \in NNI(T_{m-1}; e_{m+2}, ..., e_k),$$

and so $T' \notin N^k_{NNI}(T)$.

Now suppose that the operation on edge e_{m+1} is not the inverse of the operation on edge e_m. Then $T_{m+1} = T'_m$. Hence

$$T' \in NNI(T'_m; e_{m+2}, ..., e_k),$$

and so $T' \notin N^k_{NNI}(T)$. Therefore $P \cap N^k_{NNI}(T) = \emptyset$.

□

We now consider the number of k^{th} NNI neighbours resulting from a series of k NNI operations over a given set of distinct edges.

Lemma 4.8. *Let $T \in UB(n)$ ($n \geq 4$), let e_1, e_2,..., e_k ($1 \leq k \leq n-3$) be distinct internal edges of T. Then $NNI(T; e_1, ..., e_k)$ is a subset of $N^k_{NNI}(T)$ of size 2^k.*

Proof. For each edge e of T there are two distinct first NNI neighbours resulting from NNI operations on e. Since we perform NNI operations on k different edges in T, there are 2^k k^{th} neighbours, provided that none of the resulting trees are equivalent, or in the j^{th} NNI neighbourhood of T for some $j < k$.

The latter follows from Corollary 4.6 since $e_1, ..., e_k$ are distinct. This means that

$$NNI(T; e_1, ..., e_k) \subseteq N^k_{NNI}(T).$$

To show that none of the resulting 2^k trees are equivalent we consider the splits of these trees. Let T_k and T'_k be two trees in $NNI(T; e_1, ..., e_k)$, where at least one operation produced a different first neighbour in each case. In other words, there exist trees T_{m-1}, T_m and T'_m such that $T_{m-1} \in NNI(T; e_1, ..., e_{m-1})$, $T_m, T'_m \in NNI(T_{m-1}; e_m)$, $T_m \neq T_{m+1}$, $T_k \in NNI(T_m; e_{m+1}, ..., e_k)$, and $T'_k \in NNI(T'_m; e_{m+1}, ..., e_k)$. Note that since T_m and T'_m are the two distinct first NNI neighbours of T_{m-1} obtained by an NNI operation on e_m, $T'_m \in NNI(T_m; e_m)$.

Now we consider the splits of T, T_m, T'_m, T_k, and T'_k. By Lemma 4.5, $S(T_m, e_m) \neq S(T, e_m)$ and $S(T'_m, e_m) \neq S(T, e_m)$ because we performed a single NNI operation on edge e_m. Also by Lemma

4.5, $S(T_m, e_m) \neq S(T'_m, e_m)$, as T_m and T'_m are first NNI neighbours (by an operation on edge e_m). Since $e_m \notin \{e_{m+1}, ..., e_k\}$, $S(T_k, e_m) = S(T_m, e_m) \neq S(T'_m, e_m) = S(T'_k, e_m)$ by Lemma 4.5. Hence $T_k \neq T'_k$.

Therefore we obtain 2^k distinct k^{th} NNI neighbours from k NNI operations over distinct edges e_1,...,e_k in order. □

For the remainder of this subsection we restrict our attention to performing NNI operations on two different edges of a tree. It is natural to consider whether or not the distance between two edges in the tree affects the resulting second NNI neighbours. The following result is due to Robinson (1971), and was originally proved by exhaustion, leaving the details to the reader. Here we provide an alternate proof.

Lemma 4.9. *Let $T \in UB(n)$ ($n \geq 5$), and let e_1 and e_2 be distinct internal edges of T. Let $P = NNI(T; e_1, e_2)$ and $Q = NNI(T; e_2, e_1)$. If e_1 and e_2 are adjacent then $P \cap Q = \emptyset$, otherwise $P = Q$.*

Proof. First suppose that edges e_1 and e_2 are non-adjacent in T. Let A be the subtree containing e_2 such that $d_T(A, e_1) = 1$. Let the other three subtrees distance one from e_1 be B, C and D. First we consider $NNI(T; e_1, e_2)$. The first operation swaps two of the subtrees incident to e_1 to obtain $T_1 \in NNI(T; e_1)$. We then perform an NNI operation on edge e_2 in T_1. We obtain a tree T_2 with a subtree A' such that $d_{T_2}(A', e_1) = 1$ and B, C and D are the other three subtrees distance one from e_1. Now consider $NNI(T; e_2, e_1)$. First we perform an NNI operation on edge e_2 in A (in T), and one of the two distinct trees produced is $T'_1 \in NNI(T; e_2)$ with subtree A' where $d_{T'_1}(A', e_1) = 1$ and B, C and D are the other three subtrees distance one from e_1. The second operation swaps two of the subtrees distance one from e_1 in T'_1, which are A', B, C and D. One of the two distinct trees obtained is T_2, and so $T_2 \in NNI(T; e_2, e_1)$. This is true for all $T_2 \in NNI(T; e_1, e_2)$, so $P \subseteq Q$. Similarly $Q \subseteq P$ and so $P = Q$.

Now suppose that e_1 and e_2 are adjacent. Let A and B be subtrees such that $d_T(A, e_1) = d_T(B, e_1) = 1$ and $d_T(A, e_2) = d_T(B, e_2) = 2$. Let C and D be subtrees such that $d_T(C, e_2) = d_T(D, e_2) = 1$ and $d_T(C, e_1) = d_T(D, e_1) = 2$. Let E be the subtree such that $d_T(E, e_1) = d_T(E, e_2) = 1$. This can be seen in Fig. 4.

First we consider $NNI(T; e_1, e_2)$. Let $T_1 \in NNI(T; e_1)$ and $T_2 \in NNI(T_1; e_2)$. The first operation is over e_1, so either $d_{T_1}(A, E) = 2$ or $d_{T_1}(B, E) = 2$. Without loss of generality suppose $d_{T_1}(A, E) = 2$. Then $d_{T_1}(E, e_2) = d_{T_1}(A, e_2) = 2$. Therefore after the second operation, $d_{T_2}(E, A) = 2$. Now we consider $NNI(T; e_2, e_1)$. Let $T'_1 \in NNI(T; e_2)$ and $T'_2 \in NNI(T'_1; e_1)$. The first NNI operation is

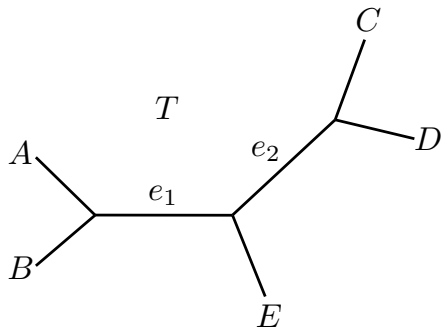

Figure 4: A general structure for an unrooted binary tree T, showing subtrees A, B, C, D, and E.

over e_2, so $d_{T_2'}(A, E) = 4$. Therefore $d_{T_2'}(A, E) \geq 3$. Hence $T_2' \neq T_2$. The choice of $T_2 \in P$ and $T_2' \in Q$ were arbitrary, so $P \cap Q = \emptyset$.

□

Lemma 4.9 tells us how the distance between the two edges we perform NNI operations on affects the resulting second NNI neighbours. Corollary 4.6 justifies that different choices of edges for the two NNI operations do not produce any duplicate second NNI neighbours. Corollary 4.8 tells us the number of second NNI neighbours resulting from NNI operations over a given set of internal edges of a tree in order. We now have sufficient information to determine the size of the second NNI neighbourhood.

Proof of Theorem 4.4

Proof. In this proof we consider all possible choices of two internal edges e_1 and e_2 of T, and the second NNI neighbours obtained by two NNI operations on these edges. Clearly if $n = 3$, T has no internal edges and the result is trivially true. If $e_1 = e_2$, then by Lemma 4.7, $NNI(T; e_1, e_2) \cup NNI(T; e_2, e_1) \subseteq N_{NNI}(T) \cup \{T\}$. It follows that if $n = 4$, then $N^2_{NNI}(T) = 0$, and so the result holds.

Now assume that $n \geq 5$, and suppose that e_1 and e_2 are distinct. By Lemma 4.8,

$$|NNI(T; e_1, e_2)| = |NNI(T; e_2, e_1)| = 2^2 = 4.$$

We know from Lemma 4.9 that if e_1 and e_2 are not adjacent then $NNI(T; e_1, e_2) = NNI(T; e_2, e_1)$. Hence $NNI(T; e_1, e_2) \cup NNI(T; e_2, e_1) = 4$. Lemma 4.9 also tells us that if e_1 and e_2 are adjacent then $NNI(T; e_1, e_2) \cap NNI(T; e_2, e_1) = \emptyset$. Hence $NNI(T; e_1, e_2) \cup NNI(T; e_2, e_1) = 8$.

By Corollary 4.6, $NNI(T; e_1, e_2) \cap NNI(T; f_1, f_2) = \emptyset$ if $\{e_1, e_2\} \neq \{f_1, f_2\}$. Therefore

$$|N^2_{NNI}(T)| = \sum_{\{e_1, e_2\} \in E(T)} |NNI(T; e_1, e_2) \cup NNI(T; e_2, e_1)|.$$

We know from Lemma 3.4 that T has $n+c-6$ pairs of adjacent internal edges and $\frac{1}{2}(n^2-9n+24)-c$ pairs of non-adjacent internal edges. Therefore summing over all possible choices of edges e_1 and e_2, we have

$$\begin{aligned}|N^2_{NNI}(T)| &= 8(n+c-6)+4\left(\frac{1}{2}(n^2-9n+24)-c\right)\\ &= 2n^2-10n+4c.\end{aligned}$$

□

It is interesting to compare these results for the size of the first and second NNI neighbourhoods with the corresponding results for the RF distance. In both cases the size of the first neighbourhood is dependent only on the number of leaves, while the size of the second neighbourhood is determined by the number of leaves and cherries.

4.3 Third Neighbourhood

In this subsection we determine the size of the third NNI neighbourhood, extending the work of Robinson (1971) who found an upper bound. Robinson (1971)'s results can be summarised as

$$|N^3_{NNI}(T)| = 8x+16y+24z+36p_3(T)+2t, \tag{3}$$

where x, y, z and t are integers, $p_3(T)$ is the number of internal paths of length three, and

$$x+y+z+p_3(T) = \frac{(n-3)(n-4)(n-5)}{6}, \tag{4}$$

$$t = n+c-6 \leq \frac{3(n-4)}{2},$$

$$p_3(T) \leq 2n-12 \text{ for } n \geq 7\ ,$$

$$z = c-2 \leq \frac{n-4}{2} \text{ for } n \geq 4\ ,$$

$$y \leq \begin{cases} \frac{3}{2}n^2-16n+42 & \text{if } n \text{ is odd} \\ (\frac{3}{2}n^2-\frac{3}{2}n+45 & \text{if } n \text{ is even.}\end{cases}$$

Our result for the size of the third NNI neighbourhood is presented in Theorem 4.10, and will be proved later in this subsection.

Theorem 4.10. *Let $T \in UB(n,c)$ $(n \geq 4)$. Then*

$$|N^3_{NNI}(T)| = \frac{4}{3}n^3-8n^2-\frac{70}{3}n+8cn-46c+12p_3(T)+164.$$

The following result is a corollary of Lemma 4.9.

Corollary 4.11. *Let $T \in UB(n)$ $(n \geq 4)$, and let e_1, ..., e_k $(k \geq 1)$ be internal edges of T. Let*

$$P = NNI(T; e_1, ..., e_m, e_{m+1}, ..., e_k),$$

$$Q = NNI(T; e_1, ..., e_{m+1}, e_m, ..., e_k).$$

If e_m and e_{m+1} are distinct and non-adjacent then $P = Q$.

Proof. Let T_{m-1} be a tree in $NNI(T; e_1, ..., e_{m-1})$. Then

$$NNI(T_{m-1}; e_m, e_{m+1}) = NNI(T_{m-1}; e_{m+1}, e_m)$$

by Lemma 4.9. This is true for any choice of T_{m-1}, so

$$NNI(T; e_1, ..., e_{m-1}, e_m, e_{m+1}) = NNI(T; e_1, ..., e_{m-1}, e_{m+1}, e_m).$$

Therefore $P = Q$. □

Corollary 4.12. *Let $T \in UB(n)$ $(n \geq 4)$, and let e_1, ..., e_k $(k \geq 1)$ be internal edges of T. Suppose that $e_m = e_j$ for some m, j where $1 \leq m < j \leq k$. Let*

$$P = NNI(T; e_1, ..., e_m, e_{m+1}, ..., e_{j-1}, e_j, ..., e_k),$$

$$Q = NNI(T; e_1, ..., e_{m-1}, e_{m+1}, ..., e_{j-1}, e_j, ..., e_k)$$

$$R = NNI(T; e_1, ..., e_{m-1}, e_{m+1}, ..., e_{j-1}, e_{j+1}, ..., e_k).$$

Suppose that the edges $e_{m+1}, ..., e_{j-1}$ are non-adjacent to e_m. If the operation on edge e_j is the inverse of the operation on edge e_m, then $P = R$, otherwise $P = Q$.

Proof. By Corollary 4.11,

$$\begin{aligned} P &= NNI(T; e_1, ..., e_{m-1}, e_{m+1}, e_m, ..., e_{j-1}, e_j, ..., e_k) \\ &= NNI(T; e_1, ..., e_{m-1}, e_{m+1}, e_{m+2}, e_m, ..., e_{j-1}, e_j, ..., e_k) \\ &\vdots \\ &= NNI(T; e_1, ..., e_{m-1}, e_{m+1}, ..., e_{j-1}, e_m, e_j, ..., e_k). \end{aligned}$$

So by the proof of Lemma 4.7, if the operation on edge e_j is the inverse of the operation on edge e_m, $P = R$, otherwise $P = Q$.

□

In the proof of Theorem 4.4 we used Corollary 4.6 to determine that performing NNI operations over two different pairs of edges never produces the same tree. Now we prove a similar result for sets of three edges.

Lemma 4.13. *Let $T \in UB(n)$ $(n \geq 5)$, $P = NNI(T; e_1, e_2, e_3)$, and $Q = NNI(T; f_1, f_2, f_3)$ where $e_1 \neq e_2$, $e_2 \neq e_3$, $f_1 \neq f_2$, and $f_2 \neq f_3$. Assume that the NNI operation on edge e_3 is not the inverse of the NNI operation on edge e_1, and the NNI operation on edge f_3 is not the inverse of the operation on edge f_1. If $\{e_1, e_2, e_3\} \neq \{f_1, f_2, f_3\}$, $P \cap Q = \emptyset$.*

Proof. Assume that $\{e_1, e_2, e_3\} \neq \{f_1, f_2, f_3\}$. Then either there exists e_i $(1 \leq i \leq 3)$ such that $e_i \notin \{f_1, f_2, f_3\}$, or there exists f_j $(1 \leq j \leq 3)$ such that $f_j \notin \{e_1, e_2, e_3\}$ (or both). Without loss of generality assume that there exists e_i $(1 \leq i \leq 3)$ such that $e_i \notin \{f_1, f_2, f_3\}$. There are two cases to consider.

1. First suppose that $e_i \neq e_j$ for all $j \neq i$ $(1 \leq j \leq 3)$. Then by Corollary 4.6, $P \cap Q = \emptyset$.

2. Now suppose that $e_i = e_j$ for some $j \neq i$ $(1 \leq j \leq 3)$. Then $e_i = e_1 = e_3$. If $e_2 \notin \{f_1, f_2, f_3\}$, Case 1 applies. Hence assume that $e_2 \in \{f_1, f_2, f_3\}$. Now $|\{e_1, e_2, e_3\}| = 2$. If $|\{f_1, f_2, f_3\}| = 3$ then Corollary 4.6 applies and $P \cap Q = \emptyset$. Assume that $|\{f_1, f_2, f_3\}| < 3$. Then $f_1 = f_3$. If $f_2 \notin \{e_1, e_2, e_3\}$ then again, Corollary 4.6 applies and $P \cap Q = \emptyset$. Assume that $f_2 \in \{e_1, e_2, e_3\}$. Then $f_2 = e_2$. Since $\{e_1, e_2, e_3\} \neq \{f_1, f_2, f_3\}$, $f_1 \notin \{e_1, e_2, e_3\}$.

   If e_2 is not adjacent to e_1, then $NNI(T; e_1, e_2, e_3) = NNI(T; e_2, e_3)$ by Corollary 4.12. By Case 1, $NNI(T; e_2, e_3) \cap NNI(T; f_1, f_2, f_3) = \emptyset$, so $P \cap Q = \emptyset$. Likewise, if e_2 is not adjacent to f_1, Case 1 applies and $P \cap Q = \emptyset$.

   Now suppose that that e_2 is adjacent to both e_1 and f_1. By Lemma 4.5, for all $T' \in Q$, $S(T', e_1) = S(T, e_1)$, as no NNI operation has been performed on the edge e_1. We will show that none of the trees in P have the split $S(T, e_1)$.

   Let the four subtrees adjacent to e_1 in T be A, B, C and D such that $d_T(A, B) = 2$ and neither A or B is incident to e_2. Then

$$S(T, e_1) = \{\mathcal{L}(A) \cup \mathcal{L}(B), \mathcal{L}(T) - (\mathcal{L}(A) \cup \mathcal{L}(B))\}.$$

   The first operation on edge e_1 will result in a tree T_1 where $d_{T'}(A, B) = 3$ and either A or B is incident to both e_1 and e_2. Without loss of generality assume this subtree is A. Then $\mathcal{L}(A)$ and $\mathcal{L}(B)$ are in different parts of $S(T_1, e_1)$. Let T_2 be a tree resulting from an NNI operation on edge e_2 in T_1. In T_2, B is adjacent to e_1, but A is not, and $d_{T_2}(A, B) = 4$. The final NNI operation is on e_1 and produces a tree T_3, where $d_{T_3}(A, B) \geq 3$. Hence T_3 does not have the split $S(T, e_1)$. Therefore none of the trees in P contain the split $S(T, e_1)$, and $P \cap Q = \emptyset$.

□

Proof of Theorem 4.10

Proof. We perform NNI operations on three internal edges e_1, e_2 and e_3 of T. These edges may be distinct, or not all distinct.

First, suppose that all three of these edges are distinct. Then there are four cases to consider;

1. The edges are pairwise non-adjacent.
2. Exactly two of the edges are adjacent.
3. The edges form an internal path of length three.
4. The edges share a common endpoint.

1. Suppose we choose three pairwise non-adjacent internal edges of T and perform NNI operations on each of these three edges in any order. By Corollary 4.8 and Corollary 4.11, there are $2^3 = 8$ third NNI neighbours of T resulting from these NNI operations.

2. Suppose we choose three internal edges e_1, e_2 and e_3 of T, such that exactly two of these three edges are adjacent. Without loss of generality let the adjacent pair be e_1 and e_2. Now suppose that we perform NNI operations on each of these three edges in any order. There are $3! = 6$ ways to order the NNI operations on the three edges. By Corollary 4.11,

$$NNI(T; e_1, e_2, e_3) = NNI(T; e_1, e_3, e_2) = NNI(T; e_3, e_1, e_2)$$

$$NNI(T; e_2, e_1, e_3) = NNI(T; e_2, e_3, e_1) = NNI(T; e_3, e_2, e_1).$$

   By Lemma 4.9,

$$NNI(T; e_1, e_2, e_3) \cap NNI(T; e_2, e_1, e_3) = \emptyset,$$

   as e_1 and e_2 are adjacent. Therefore by Corollary 4.8, there are

$$|NNI(T; e_1, e_2, e_3) \cup NNI(T; e_2, e_1, e_3)| = 8 + 8 = 16$$

   third NNI neighbours of T resulting from these NNI operations.

3. Suppose we have an internal $(e_1 - e_3)$-path of length three, with edge e_2 adjacent to both e_1 and e_3. There are $3! = 6$ ways to order the operations on these three edges. By Corollary 4.11,

$$NNI(T; e_1, e_3, e_2) = NNI(T; e_3, e_1, e_2),$$

   since e_1 and e_3 are non-adjacent. Now consider $NNI(T; e_2, e_1, e_3)$ and $NNI(T; e_2, e_3, e_1)$. Since e_1 and e_3 are non-adjacent we might also expect these sets to be equivalent. However,

in one of the first NNI neighbours $T' \in NNI(T; e_2)$, e_1 and e_3 are adjacent. In the other first NNI neighbour $T'' \in NNI(T; e_2)$, $T'' \neq T'$, e_1 and e_3 are non-adjacent. By Lemma 4.9 and Corollary 4.11,

$$NNI(T'; e_1, e_3) \cap NNI(T'; e_3, e_1) = \emptyset, \text{ and}$$

$$NNI(T''; e_1, e_3) = NNI(T''; e_3, e_1).$$

Since $|NNI(T''; e_1, e_3)| = 4$,

$$NNI(T; e_2, e_1, e_3) \cap NNI(T; e_2, e_3, e_1) = 4.$$

By considering distances between pairs of subtrees distance one from one or more of the three edges e_1, e_2 and e_3, it can be shown that these are the only duplicate trees obtained. Since each ordering of the operations produces $2^3 = 8$ neighbours, we have

$$6(8) - |NNI(T; e_1, e_3, e_2)| - 4 = 48 - 8 - 4 = 36$$

third NNI neighbours for each choice of the three edges.

4. Suppose we choose three internal edges e_1, e_2 and e_3 of T that share a common endpoint, and perform NNI operations on each of these three edges in any order. Without loss of generality suppose that the first NNI operation is over edge e_1, and let $T' \in NNI(T; e_1)$. In T' edges e_2 and e_3 are not adjacent. Therefore by Lemma 4.11,

$$NNI(T; e_1, e_2, e_3) = NNI(T; e_1, e_3, e_2).$$

Similarly,

$$NNI(T; e_2, e_1, e_3) = NNI(T; e_2, e_3, e_1), \text{ and}$$

$$NNI(T; e_3, e_1, e_2) = NNI(T; e_3, e_2, e_1).$$

However, because all three edges are pairwise adjacent in T, the choice of the first edge is important, and

$$NNI(T; e_1, e_2, e_3) \cap NNI(T; e_2, e_1, e_3) = \emptyset,$$

$$NNI(T; e_1, e_2, e_3) \cap NNI(T; e_3, e_1, e_2) = \emptyset, \text{ and}$$

$$NNI(T; e_2, e_1, e_3) \cap NNI(T; e_3, e_1, e_2) = \emptyset.$$

Therefore there are $|NNI(T; e_1, e_2, e_3)| + |NNI(T; e_2, e_1, e_3)| + |NNI(T; e_3, e_1, e_2)| = 24$ resulting third NNI neighbours.

Now suppose that we choose three internal edges e_1, e_2 and e_3 of T, at least two of which are the same edge. We consider $NNI(T; e_1, e_2, e_3)$. There are two cases to consider.

1. If $e_1 = e_2 = e_3$, then by Lemma 4.7 we obtain no third NNI neighbours.

2. Suppose that exactly one of the edges e_1, e_2 and e_3 is distinct. By Lemma 4.7, if $e_1 = e_2$ or $e_2 = e_3$ then $NNI(T; e_1, e_2, e_3) \cap N^3_{NNI}(T) = \emptyset$.

   Now suppose that e_2 is the distinct edge, so $e_1 = e_3$. Then either e_2 is adjacent to e_1, or not.

   (a) Suppose that e_2 is not adjacent to e_1. Then by Corollary 4.12, $NNI(T; e_1, e_2, e_3) \cap N^3_{NNI}(T) = \emptyset$.

   (b) Suppose that e_2 is adjacent to e_1. Consider different arrangements of the five subtrees A, B, C, D and E distance one from one or both of the two adjacent internal edges, and not containing either of them (see Fig. 4 in the proof of Lemma 4.9). This is the same as arrangements of the binary phylogenetic tree where $n = 5$. By Lemma 2.2 there are 15 different binary trees with $n = 5$, and one of these is the tree T_5 corresponding to the arrangement in T. There are $2(n-3) = 4$ first NNI neighbours of T_5 by Theorem 4.2 and $2n^2 - 10n + 4c = 8$ second NNI neighbours of T_5 by Theorem 4.4. Therefore there are $15 - 8 - 4 - 1 = 2$ trees in $UB(5)$ that are not T_5, or in the first or second neighbourhood of T_5. Similarly there are 2 trees in $UB(n)$ with pendant subtrees A, B, C, D and E that are not T or in the first or second neighbourhood of T' (recall that E is the subtree that is distance one from both e_1 and e_2). These two trees T_1 and T_2, are those for which $d_{T_1}(E, e_1) = d_{T_1}(E, e_2) = d_{T_2}(E, e_1) = d_{T_2}(E, e_2) = 1$.

   Now we show that T_1 and T_2 are third NNI neighbours of T. The first NNI operation on edge e_1 swaps subtrees so that in the resulting tree T', $d_{T'}(E, e_1) = 1$ and $d_{T'}(E, e_2) = 2$. Let T'' be the tree resulting from the second NNI operation on edge e_2 of T'. Since E is not adjacent to e_2 in T', $d_{T''}(E, e_1) = 1$ and $d_{T''}(E, e_2) = 2$. Let X be the subtree of T'' such that $d_{T''}(X, e_1) = d_{T''}(X, e_2) = 1$. Note that either $X = C$ or $X = D$. Then E and X are a swappable pair for the third NNI operation on edge e_1 in T''. Hence there exists $T''' \in NNI(T''; e_3)$ such that $d_{T'''}(E, e_1) = d_{T'''}(E, e_2) = 1$. The proof of Lemma 4.13 justifies that $T''' \neq T$. Whether $T''' = T_1$ or $T''' = T_2$ depends on whether the first NNI operation on edge e_1 in T swapped subtrees so that $d_{T'}(E, A) = 2$, or so that $d_{T'}(E, B) = 2$. Hence T_1 and T_2 are both third NNI neighbours of T, so

$$|NNI(T; e_1, e_2, e_3) \cap N^3_{NNI}(T)| = 2.$$

By Lemma 4.13 all distinct choices of edges e_1, e_2 and e_3 produce distinct first neighbours.

Recall that $p_3(T)$ is the number of ways to select three internal edges of T so that they form an internal path of length three. Let x be the number of ways of choosing three internal edges of T so that they are pairwise non-adjacent, let y be the number of ways of choosing three internal edges of T so that exactly one pair is adjacent, and let z be the number of ways of choosing three internal edges of T so that they share a common endpoint. By Corollary 3.4, there are $n + c - 6$ adjacent pairs of edges in T.

Therefore combining all of the cases (where the three edges are distinct, or not all distinct),

$$|N^3_{NNI}(T)| = 8x + 16y + 36p_3(T) + 24z + 2(n + c - 6). \tag{5}$$

By Lemma 3.3, $z = c - 2$.

To determine y, we note that there are $n + c - 6$ pairs of adjacent internal edges in T. Therefore there are $(n - 5)(n + c - 6)$ ways of choosing three internal edges of T such that are least two of these edges are adjacent. Removing all cases where we have an internal path of length three, or the three edges share an endpoint, we have

$$y = (n - 5)(n + c - 6) - 2p_3(T) - 3z.$$

In total there are $\frac{1}{6}(n - 3)(n - 4)(n - 5)$ ways to choose three internal edges of T, so the number where all are pairwise non-adjacent is

$$x = \frac{1}{6}(n - 3)(n - 4)(n - 5) - y - z - p_3(T).$$

Hence substituting into (5) we have

$$|N^3_{NNI}(T)| = \frac{4}{3}n^3 - 8n^2 - \frac{70}{3}n + 8cn - 46c + 12p_3(T) + 164.$$

□

Now we consider how we might calculate the value of $p_3(T)$ for a tree $T \in UB(n)$.

Theorem 4.14. *Let $T \in UB(n)$. Then $p_1(T) = n - 3$ and*

$$p_k(T) = 4p_{k-2}(T) - h_k(T) - m_k(T),$$

where for all k, $m_k(T)$ is the number of paths of length k in T where both end points are leaves of T, and $h_k(T)$ is the number of paths of length k in T where exactly one end vertex is a leaf of T.

Proof. The number of internal edges in T is $n-3$, so $p_1(T) = n-3$.
The number of paths of length k in T is $P_k(T) = p_k(T) + m_k(T) + h_k(T)$. Now in a binary tree, $P_k(T) = 4p_{k-2}(T)$. Therefore

$$p_k(T) = P_k(T) - m_k(T) - h_k(T) = 4p_{k-2}(T) - h_k(T) - m_k(T).$$

□

It follows that $p_3(T) = 4(n-3) - h_3(T) - m_3(T)$ for a tree $T \in UB(n,c)$, and therefore

$$|N^3_{NNI}(T)| = \frac{4}{3}n^3 - 8n^2 + \frac{74}{3}n + 8cn - 46c - 12h_3(T) - 12m_3(T) + 20.$$

Note that $m_k(T)$ and $h_k(T)$ can both be counted using a breadth first search in polynomial time.

4.4 Asymptotic Result for the k^{th} Neighbourhood

In this subsection we prove Theorem 4.1. Similarly to the proof of Theorem 3.1 we consider the number of k^{th} NNI neighbours resulting from NNI operations over a given set of k internal edges. From Lemma 3.7 we know the number sets of k internal edges of T. Combining these gives us the total number of k^{th} NNI neighbours. The four different cases that are relevant are:

1. The k edges are distinct and pairwise non-adjacent.
2. The k edges are distinct and exactly two are adjacent.
3. The k edges are distinct and more than two are adjacent.
4. The k edges are not all distinct.

These are the same cases as for RF, with the additional possibility that the k edges are not all distinct (Case 4). In Equation 2 of Theorem 4.1, the term of order n^k is completely determined by Case 1, while the term of order n^{k-1} is determined by Cases 1 and 2. We show that all other possibilities for the k edges (covered by Cases 3 and 4) only contribute to terms of order n^{k-2} or lower.

Neighbours Resulting from NNI Operations over k Given Edges

We consider how many k^{th} NNI neighbours result from k NNI operations on a given set of k internal edges of T in the cases outlined above.

Lemma 4.15. *Let $T \in UB(n)$ $(n \geq 4)$.*

(i) For any given set of k distinct, pairwise non-adjacent internal edges ($1 \leq k \leq n-3$), there are 2^k k^{th} neighbours of T resulting from NNI operations on this sequence of edges in any order.

(ii) For any given set of k distinct internal edges ($2 \leq k \leq n-2$) where exactly one pair is adjacent, there are 2^{k+1} k^{th} neighbours of T resulting from NNI operations on this sequence edges in any order.

(iii) For a given T and a given sequence of k (not necessarily distinct) edges of T ($k \geq 1$), the number of k^{th} NNI neighbours resulting from NNI operations on this sequence edges in any order is constant with respect to n.

Proof.

(i) Suppose we perform NNI operations on k distinct, pairwise non-adjacent internal edges e_1, ..., e_k of T. Lemma 4.8 tells us that if the NNI operations are performed in a given order we obtain 2^k neighbours. Since the edges are pairwise non-adjacent, by Corollary 4.11, changing the order of the operations does not change the set of trees produced. Hence there are 2^k neighbours of T resulting from NNI operations on this set of edges in any order.

(ii) The only difference between this and (i) is the pair of adjacent edges e_i and e_j ($1 \leq i < j \leq k$). By Corollary 4.11,

$$NNI(T; e_1, ..., e_i, ..., e_j, ..., e_k) = NNI(T; e_1, ..., e_{i-1}, e_{i+1}, ..., e_{j-1}, e_{j+1}, ..., e_k, e_i, e_j).$$

As in (i), by Lemma 4.8 and Corollary 4.11, performing NNI operations on the edges e_1, ..., e_{i-1}, e_{i+1}, ..., e_{j-1} , e_{j+1}, ..., e_k in any given order produces the set of trees

$$NNI(T; e_1, ..., e_{i-1}, e_{i+1}, ..., e_{j-1}, e_{j+1}, ..., e_k),$$

where

$$|NNI(T; e_1, ..., e_{i-1}, e_{i+1}, ..., e_{j-1}, e_{j+1}, ..., e_k)| = 2^{k-2}.$$

Let $T_{k-2} \in NNI(T; e_1, ..., e_{i-1}, e_{i+1}, ..., e_{j-1}, e_{j+1}, ..., e_k)$. By Lemma 4.9,

$$NNI(T_{k-2}; e_i, e_j) \cap NNI(T_{k-2}; e_j, e_i) = \emptyset.$$

Therefore since

$$|NNI(T_{k-2}; e_i, e_j) \cup NNI(T_{k-2}; e_j, e_i)| = 4 + 4 = 8,$$

we have $8(2^{k-2}) = 2^{k+1}$, k^{th} NNI neighbours of T.

(iii) Let F be the subgraph of T consisting of the edges $e_1, ..., e_k$. Then F has m components, $C_1, ..., C_m$ ($1 \leq m \leq k$). Edges in different components of F are not adjacent, so by Corollary 4.11 the order in which we perform NNI operations on them does not change the resulting neighbours. However, by Lemma 4.9 the order of NNI operations on the edges that form a

component of F, does change the resulting neighbours. Therefore the number of neighbours resulting from NNI operations on the k edges is

$$\prod_{\ell=1}^{m} f(C_\ell),$$

where $f(C_\ell)$ is the number of distinct k^{th} NNI neighbours resulting from NNI operations in T on the edges from $e_1, ..., e_k$ that are in component C_ℓ of F (more than one NNI operation may be on the same edge). We consider each component separately.

Let C_p, $1 \leq p \leq m$ be a component of F with q edges and consider calculating $f(C_p)$. Let $f_1, ..., f_j$ $(j \leq k)$ be the subsequence of the edges $e_1, ..., e_k$ that are in C_p. Note that the edges $f_1, ..., f_j$ are not necessarily distinct. Add pendant edges incident to vertices in $V(C_p)$, so that all of the vertices in $V(C_p)$ have degree three. The resulting tree C'_p is an unrooted binary tree with $q+3$ leaves. The internal edges of C'_p are the distinct edges of the sequence $f_1, ..., f_j$. Then $f(C_p)$ is equivalent to the number of distinct k^{th} NNI neighbours of C'_p resulting from NNI operations on the edges f_1, ..., f_j of C'_p. The number of k^{th} neighbours $f(C'_p)$ from these operations depends only on the shape and size of C'_p, the number of times we perform an NNI operation on each internal edge of C_p, and the order in which the operations are performed. All of these factors are determined by the choice of the edges $e_1, ..., e_k$ of T. Therefore, given a tree T, and internal edges $e_1, ..., e_k$ of T, the number of k^{th} NNI neighbours of T resulting from NNI operations on the edges $e_1, ..., e_k$ in any order is independent of n.

□

Now we have all of the information required to prove Theorem 4.1.

Proof of Theorem 4.1

Proof. We break down the calculation of the size of the k^{th} NNI neighbourhood of T into two steps. First we consider how many neighbours result from k NNI operations on a given sequence of k edges of T. Then we consider how many ways these k edges can be chosen in T. By Lemma 4.15 the number of k^{th} NNI neighbours of a given tree T resulting from operations over a given sequence of k edges is not dependent on n. Hence the only factor dependent on n is the number of ways of choosing these k edges. We consider two cases.

First, assume that the k edges are all distinct, and consider how many ways they can be chosen in T. By Lemma 3.7 the case where the k edges are pairwise non-adjacent (Case 1 from the beginning of this subsection) gives a term of order n^k and a term of order n^{k-1}. The case where exactly two of the k edges are adjacent (Case 2), produces a term of order n^{k-1}, but not a term of order n^k. If

more than two of the k edges are adjacent then the highest order term is $O(n^{k-2})$.

Now suppose that the k edges are not all distinct. By Lemma 3.7 if $k-1$ of the k edges are distinct and pairwise non-adjacent, the highest order term is $O(n^{k-1})$. However, by Corollary 4.12, the trees produced by this are not k^{th} NNI neighbours of T. By Lemma 3.7, if more than two of the k edges are the same, or more than two are adjacent, the highest order term is $O(n^{k-2})$.

In the case where the edges are pairwise non-adjacent, by Lemma 4.15 there are 2^k k^{th} NNI neighbours of T resulting from NNI operations on a given set of k edges. In the case where exactly two edges are adjacent there are 2^{k+1} resulting k^{th} NNI neighbours. Hence by Lemma 3.7,

$$\begin{aligned}|N^k_{NNI}(T)| &\geq \left(\frac{1}{k!}n^k - \frac{k(5k+1)}{2k!}n^{k-1}\right)2^k + \frac{1}{2(k-2)!}n^{k-1}2^{k+1} + O(n^{k-2}) \\ &= \frac{2^k}{k!}n^k - \frac{3k(k+1)}{2k!}2^k n^{k-1} + O(n^{k-2}).\end{aligned}$$

$$\begin{aligned}|N^k_{NNI}(T)| &\leq \left(\frac{1}{k!}n^k - \frac{k(k+2)}{k!}n^{k-1}\right)2^k + \frac{2}{(k-2)!}n^{k-1}2^{k+1} + O(n^{k-2}) \\ &= \frac{2^k}{k!}n^k + \frac{3k(k-2)}{k!}2^k n^{k-1} + O(n^{k-2}).\end{aligned}$$

□

We can see that this result is very similar to the size of the k^{th} RF neighbourhood, as $D_{T,k}$ and $C_{T,k}$ are both quadratic in k.

5 Pairs of Trees with Shared Neighbours

Now that we have expressions for the size of the first and second NNI and RF neighbourhoods, it is possible to find an exact count for the number of pairs of binary phylogenetic trees with n leaves that share a first NNI or RF neighbour. This is the same as the number of pairs of trees that are within at most distance two of each other, and tells us more about the structure of $UB(n)$.

We can calculate the number of pairs of trees that share a first neighbour by summing the size of the first and second neighbourhoods of a tree, over all binary phylogenetic trees. This counts each pair twice, so we halve the result. However, since the size of the second neighbourhood for both NNI and RF is dependent on the number of cherries, it is necessary to know how many binary phylogenetic

trees there are with n leaves and c cherries, which is $|UB(n,c)|$. Hendy and Penny (1982) found an expression for $|UB(n,c)|$, which they proved using induction on the number of leaves. Here we present a constructive proof of their result.

Proposition 5.1. *For all $n \geq 4$,*

$$|UB(n,c)| = \frac{n!(n-4)!}{c!(c-2)!(n-2c)!2^{2c-2}},$$

for $2 \leq c \leq \frac{n}{2}$, and $|UB(n,c)| = 0$ otherwise.

Proof. The tree with the smallest number of cherries is a caterpillar, which has two cherries. Since there are two leaves in a cherry, the maximum number of cherries a tree can have is $\frac{n}{2}$. Hence for $c < 2$ or $c > \frac{n}{2}$ we have $|UB(n,c)| = 0$.

Let $2 \leq c \leq \frac{n}{2}$. Each $T \in UB(n,c)$ has $2c$ leaves that are in cherries. The number of ways of choosing the $2c$ leaves of T to form the c cherries is $\binom{n}{2c}$. From those $2c$ leaves we choose two for each cherry. Since the order of the cherries is not important, we divide by $c!$, the number of ways to order the c cherries. Therefore the number of ways of choosing c cherries from n leaves is

$$\begin{aligned} M &= \frac{1}{c!}\binom{n}{2c}\binom{2c}{2}\binom{2c-2}{2}\cdots\binom{2}{2} \\ &= \frac{1}{c!}\left(\frac{n!}{(2c)!(n-2c)!}\frac{(2c!)}{2!(2c-2)!}\frac{(2c-2)!}{2!(2c-4)!}\cdots\frac{2!}{2!}\right) = \frac{n!}{c!(n-2c)!2^c}. \end{aligned}$$

Now consider each cherry as a single leaf with the labels of both leaves. There are c of these double-labelled leaves and $n-2c$ other leaves. We determine the number of trees that can be formed with these leaves. We have the restriction that no pair of the $n-2c$ single-labelled leaves can be in a cherry. Therefore we will first consider the number of trees we can form with only the c double-labelled leaves. This number, P, is given in Lemma 2.2,

$$P = |UB(c)| = \frac{(2c-4)!}{(c-2)!2^{c-2}}.$$

Now let T be one of these trees with c double-labelled leaves. We insert the remaining $n-2c$ single-labelled leaves. Each single-labelled leaf can only be joined to edges in $E(T)$, so as not to create another cherry. There are $2c-3$ edges in $E(T)$ to which the single-labelled leaves could be joined. Since there are no other restrictions on where these single-labelled leaves must be inserted, we simply need to count the number of distinct trees resulting from joining the $n-2c$ single labelled

edges to edges in $E(T)$. The number of distinct trees is given by

$$\begin{aligned} Q &= (n-2c)!\binom{(n-2c)+(2c-3)-1}{(2c-3)-1} \\ &= (n-2c)!\binom{n-4}{2c-4} \\ &= \frac{(n-4)!(n-2c)!}{(2c-4)!(n-2c)!} = \frac{(n-4)!}{(2c-4)!}. \end{aligned}$$

Combining M, P, and Q, we have

$$\begin{aligned} |UB(n,c)| = MPQ &= \frac{n!}{c!(n-2c)!2^c} \cdot \frac{(2c-4)!}{(c-2)!2^{c-2}} \cdot \frac{(n-4)!}{(2c-4)!} \\ &= \frac{n!(n-4)!}{c!(c-2)!(n-2c)!2^{2c-2}}. \end{aligned}$$

□

Now we can use this result to find the number of pairs of binary phylogenetic trees in $UB(n)$ that are within at most distance two of each other under NNI and RF. For $\theta \in \{NNI, RF\}$, define

$$N_\theta^{\leq k}(n) = \{(T,T') : T,T' \in UB(n), d_\theta(T,T') \leq k\}.$$

Corollary 5.2. *Let $n \geq 3$, Then*

(i) $|N_{NNI}^{\leq 2}(n)| = \sum_{c=2}^{\lfloor \frac{n}{2} \rfloor} |UB(n,c)|(n^2-4n+2c-3)$.

(ii) $|N_{RF}^{\leq 2}(n)| = \sum_{c=2}^{\lfloor \frac{n}{2} \rfloor} |UB(n,c)|(n^2-3n+3c-9)$.

Proof.

(i) For $T \in UB(n,c)$, the number of first and second NNI neighbours is

$$\begin{aligned} N_{NNI}(T) + N_{NNI}^2(T) &= 2(n-3) + 2n^2 - 10n + 4c \\ &= 2n^2 - 8n + 4c - 6. \end{aligned}$$

To find the number of pairs of trees in $UB(n)$ that are within NNI distance two, we simply sum the number of first and second neighbours over all trees in $UB(n)$, and then halve the result as each pair will be counted twice. So,

$$\begin{aligned} |N_{NNI}^{\leq 2}(n)| &= \frac{1}{2}\sum_{c=2}^{\lfloor \frac{n}{2} \rfloor} |UB(n,c)|(2n^2-8n+4c-6) \\ &= \sum_{c=2}^{\lfloor \frac{n}{2} \rfloor} |UB(n,c)|(n^2-4n+2c-3). \end{aligned}$$

Proposition 5.1 gives us $|UB(n,c)|$.

(ii) For each unrooted binary tree T, the number of first and second RF neighbours is

$$\begin{aligned} N_{RF}(T) + N^2_{RF}(T) &= 2(n-3) + 2n^2 - 8n + 6c - 12 \\ &= 2n^2 - 6n + 6c - 18. \end{aligned}$$

Therefore

$$\begin{aligned} |N^{\leq 2}_{RF}(n)| &= \frac{1}{2} \sum_{c=2}^{\lfloor \frac{n}{2} \rfloor} |UB(n,c)|(2n^2 - 6n + 6c - 18) \\ &= \sum_{c=2}^{\lfloor \frac{n}{2} \rfloor} |UB(n,c)|(n^2 - 3n + 3c - 9). \end{aligned}$$

□

6 Subtree Prune and Regraft

A *subtree prune and regraft (SPR)* operation on a tree $T \in UB(n)$ is defined by the following process:

1. Select an edge $e = \{u, v\} \in E(T)$ and delete it, leaving two components T_u (containing the vertex u) and T_v (containing the vertex v).

2. Select an edge $f \in E(T_v)$, and subdivide f with a new vertex w to obtain two edges f_1 and f_2. The vertex w has degree two.

3. Insert the edge $g = \{w, u\}$, and suppress the vertex v to obtain a binary tree $T' \in UB(n)$.

Essentially we *prune* the subtree T_u, and *regraft* it onto edge f. We refer to e as the *cut edge* and f as the *join edge* of the SPR operation (see Fig. 5). The tree T' is a *first SPR neighbour* of T. We will use the notation $SPR(T, (e, f))$ to refer to the tree obtained by an SPR operation on tree T with cut edge e and join edge f.

Note that if $d_T(e, f) = 1$ then T' is a first NNI neighbour of T (Semple and Steel, 2003). In Fig. 1, T_2 is obtained from T_1 by a single SPR operation, with cut edge incident to the leaf d and join edge incident to the root of the cherry with leaves a and b.

Consider a graph G in which each vertex represents a tree in $UB(n)$ and there is an edge between the vertices representing trees T_1 and T_2 if they are first SPR neighbours. The *SPR distance* between T_1 and T_2, $\delta_{SPR}(T_1, T_2)$, is the distance between the two vertices representing T_1 and T_2 in G.

The size of the first SPR neighbourhood of a given binary phylogenetic tree was determined by Allen and Steel (2001). No other SPR neighbourhood sizes are known. In this section we independently

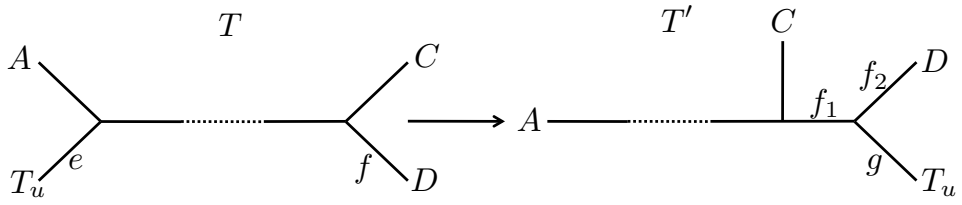

Figure 5: An example of an SPR operation with cut edge e and join edge f.

investigate the first and second SPR neighbourhoods. We obtain the same expression as Allen and Steel (2001) for the size of the first SPR neighbourhood, and we show that unlike RF and NNI, the size of the second SPR neighbourhood of a binary tree T cannot be determined solely by the number of leaves and cherries of T.

In relation to the structure of the SPR neighbourhood, Caceres et al. (2013) provided tight bounds on the length of the shortest NNI walk that visits all trees in the first SPR neighbourhood of a tree T. Allen and Steel (2001) found upper and lower bounds for the maximum SPR distance between any two trees in $UB(n)$.

6.1 First Neighbourhood

Allen and Steel (2001) calculated an expression for the size of the first SPR neighbourhood of a tree $T \in UB(n)$. This is stated below, along with an independent proof.

Theorem 6.1. *Let $T \in UB(n)$, $n \geq 3$. Then $|N_{SPR}(T)| = 2(n-3)(2n-7)$.*

Proof. Suppose that we perform a single SPR operation on T with cut edge $e = \{u, v\}$ and join edge f. Call the resulting tree T'. Given the cut edge e, there are three cases to consider for the choice of the join edge f;

1. $d_T(e, f) = 0$,
2. $d_T(e, f) = 1$, and
3. $d_T(e, f) > 1$.

1. Assume that $d_T(e, f) = 0$, that is, f is adjacent to e. Without loss of generality let $f = \{v, v_1\}$. Edge f is subdivided by vertex w, and vertex v is suppressed by the SPR operation, so f and v are not in T'. In T', vertex w is adjacent to the three vertices that are adjacent to v in T. Hence we have essentially replaced v in T with w in T', and so $T' = T$.

2. Since $d_T(e, f) = 1$ there is exactly one other edge on the $(e - f)$-path in T, which we call h. Then T' is a first NNI neighbour of T obtained by swapping subtrees across h (see Fig. 6).

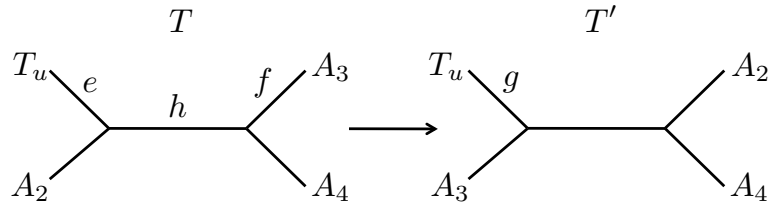

Figure 6: Tree T' is a first neighbour of T obtained by an SPR operation with cut edge e and join edge f, or by an NNI operation swapping subtrees A_2 and A_3 across h.

Let h' be an internal edge of T, and let A and B be a pair of subtrees of T that are swappable across h' (see Fig. 7). Let T'' be a first NNI neighbour of T obtained by swapping subtrees A and B across edge h'. Then T'' is a first SPR neighbour of T obtained by an operation with cut edge e' and join edge f', where $d_T(e', A) = 1$, $d_T(e', h') = 0$ and $d_T(f', B) = 0$. Therefore we have exactly the first NNI neighbours of T, of which there are $2(n - 3)$.

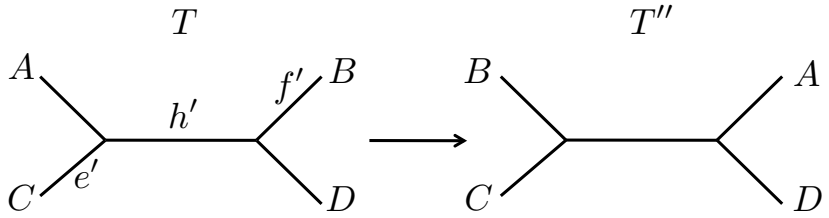

Figure 7: Tree T'' is a first neighbour of T obtained by an NNI operation swapping subtrees A and B across edge h', or by an SPR operation with cut edge e' and join edge f'.

3. Now we consider the case where $d_T(e, f) > 1$. Let $h = \{v_1, v_2\}$ be an edge on the $(e - f)$-path in T, such that h is adjacent to e. Let A, B, C and D be the four subtrees distance one from h in T, with $d_T(A, B) = 2$, and let a, b, c, and d be their respective incident internal edges (see Fig. 8). If edge f is in subtree C or D, then either $e = a$ or $e = b$. If f is in subtree A or B, then either $e = c$ or $e = d$. There are $(2n - 3) - 5 = 2n - 8$ edges in the four subtrees A, B, C and D. Therefore given h, there are $2(2n - 8) = 4(n - 4)$ first SPR neighbours of T. By Lemma 2.1, T has $n - 3$ internal edges, giving a total of $4(n - 3)(n - 4)$ first SPR neighbours, provided that all are distinct, and none are trees from Case 2.

Now we justify that all of the $4(n-3)(n-4)$ trees from Case 3 are distinct and none are trees from Case 2. We consider the set of trees obtained by SPR operations when $e = a$ and f is an

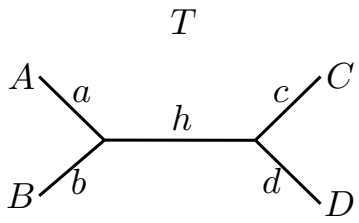

Figure 8: The four subtrees A, B, C and D distance one from internal edge h in T.

edge in C, and show that no other SPR operations (from Case 2 or Case 3) can produce any of these trees.

Let $e = a$, let f be an edge in C, and let $N_1, ..., N_p$ be the first SPR neighbours resulting from SPR operations with these cut and join edges, where p is the number of edges in C. Since a is the edge deleted from T by the SPR operation, in the neighbour N_i, $i = 1, ..., p$, the edges h and b have been replaced with a single edge h'. We have $d_{N_i}(A, B) \geq 4$, $d_{N_i}(A, D) \geq 4$, and $d_{N_i}(B, D) = 2$, $i = 1, ..., p$. Note that C is not a subtree of N_i.

First we note that each different choice of f in C produces a different tree so $N_1, ..., N_p$ are distinct. What is not so easy to see is that no other SPR operation (from Case 2 or Case 3) can produce a tree in $\{N_1, ..., N_p\}$. Let e' and f' be internal edges of T such that $d_T(e', f') \geq 1$ (Cases 2 and 3). Assume that $e' \neq a$ or f' is not an edge of C. Let $T'' = SPR(T, (e', f'))$. We justify that $T'' \notin \{N_1, ..., N_p\}$ by considering all possible choices of e' and f'.

First suppose that e' is an edge in A, B or D. Then this subtree is not a subtree of T'', so $T'' \notin \{N_1, ..., N_p\}$. Similarly, if f' is an edge in A, B or D then $T'' \notin \{N_1, ..., N_p\}$. As noted above, C is not a subtree of N_i, $i = 1, ..., p$. Suppose that neither e' or f' is an edge of C. Then C is a subtree of T'', and so $T'' \notin \{N_1, ..., N_p\}$. We now assume that either e' or f' is an edge in C (or both).

Suppose that e' is an edge in C. If $f' = a$ or $f' = b$ then $d_{T''}(A, B) = 3$, so $T'' \notin \{N_1, ..., N_p\}$. If $f' = c$, $f' = d$, $f' = h$, or f' is an edge of C, then $d_{T''}(A, B) = 2$, so $T'' \notin \{N_1, ..., N_p\}$.

Now suppose that e' is not an edge in C. Therefore f' is an edge in C. The remaining choices for e are a, b, c, d and h. If $e' = c$, $e' = d$ or $e' = h$ then $d_{T''}(A, B) = 2$ so $T'' \notin \{N_1, ..., N_p\}$. Let $e' = b$. Then $d_{T''}(A, D) = 2$, and so $T'' \notin \{N_1, ..., N_p\}$. Hence $T'' \in \{N_1, ..., N_p\}$ only if

$e' = a$ and f' is an edge of C. Therefore Case 3 produces $4(n-3)(n-4)$ distinct first SPR neighbours, none of which are trees in Case 2.

Combining the three cases we have $2(n-3)$ first SPR neighbours from Case 2, and $4(n-3)(n-4)$ first SPR neighbours from Case 3. From Case 1 there were none. Therefore in total there are

$$\begin{aligned} N &= 2(n-3) + 4(n-3)(n-4) \\ &= 2(n-3)(2n-7) \end{aligned}$$

first SPR neighbours of T. □

6.2 Second Neighbourhood

As with NNI and RF, the size of the first SPR neighbourhood of a tree depends only on the number of leaves in the tree. However, unlike NNI and RF, the size of the second SPR neighbourhood of a tree cannot be expressed solely in terms of the number of leaves and cherries of the tree. In this subsection we show that these two parameters are not sufficient to determine even the highest order term of the size of the second SPR neighbourhood. At the end of this subsection we prove our main results, which are presented in Theorems 6.2 and 6.3.

Theorem 6.2. *Let $T \in UB(n)$.*

(i) If T is a caterpillar then

$$|N^2_{SPR}(T)| = \frac{1}{2}n^4 + O(n^3).$$

(ii) If T is a balanced tree then

$$|N^2_{SPR}(T)| = \frac{1}{3}n^4 + O(n^3).$$

It is evident from Theorem 6.2 that the size of the second SPR neighbourhood of a tree T is not uniquely determined by the number of leaves of T. However, every caterpillar has exactly two cherries, while a balanced tree with at least six leaves has at least three cherries. Therefore for $n \geq 6$ a caterpillar and a balanced tree, each with n leaves, have different numbers of cherries. Therefore Theorem 6.2 does not justify that the size of the second SPR neighbourhood of T cannot be uniquely determined by the number of leaves and cherries of T. To show this, we consider two different structures of an unrooted binary tree T with $n = 3m$ ($m \geq 3$) leaves, and 3 cherries. These two tree structures (Type 1 and Type 2) can be seen in Fig. 9 and Fig. 10 respectively. Similarly to Theorem 6.2, we show that trees of Type 1 and Type 2 also have a different highest order term in the expression for the size of the second SPR neighbourhood. This result is presented in Theorem 6.3.

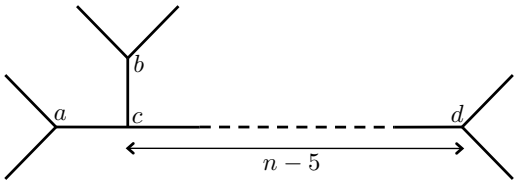

Figure 9: A Type I tree with three cherries and $n = 3m$ leaves ($m \geq 3$).

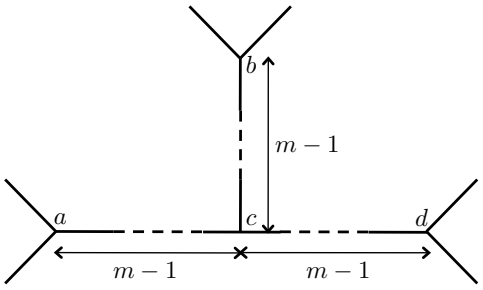

Figure 10: A Type II tree with three cherries and $n = 3m$ leaves ($m \geq 3$).

Theorem 6.3. *Let T_1 and T_2 be unrooted binary trees with $n = 3m$ leaves ($m \geq 3$) and three cherries, and suppose that T_1 is of Type I and T_2 is of Type II. Then*

$$|N^2_{SPR}(T_1)| = \frac{1}{2}n^4 + O(n^3), \text{ and}$$

$$|N^2_{SPR}(T_2)| = \frac{23}{54}n^4 + O(n^3).$$

We will use the notation

$$SPR(T, (c_1, j_1), (c_2, j_2), ..., (c_k, j_k))$$

to denote the tree obtained by k successive SPR operations starting with tree T, where c_1 and j_1 in T are the cut and join edges respectively of the first operation, c_2 and j_2 in $SPR(T, (c_1, j_1))$ are the cut and join edges of the second operation, and so on. When $k = 2$ we refer to the two operations that result in the set of trees $SPR(T, (c_1, j_1), (c_2, j_2))$, as a *pair of SPR operations.*

First we determine an upper bound on the size of the second SPR neighbourhood. This follows directly from the expression for the size of the first SPR neighbourhood given in Theorem 6.1.

Corollary 6.4. *Let $T \in UB(n)$ ($n \geq 3$). Then*

$$|N^2_{SPR}(T)| \leq 4(n-3)^2(2n-7)^2 = O(n^4).$$

The first step in proving Theorems 6.2 and 6.3 is to determine whether or not all pairs of SPR operations contribute to the term of order n^4 in the expression for the size of the second SPR neighbourhood of a tree.

Let $T \in UB(n)$ and let

$$\mathbb{T}(T) = \{(c_1, c_2, j_1, j_2) : c_1, j_1 \in E(T), c_1 \neq j_1; c_2, j_2 \in E(SPR(T, (c_1, j_1))), c_2 \neq j_2\}.$$

This is the set of all possible choices for the four cut and join edges of two SPR operations starting with tree T.

We could break down the possible choices of the edges c_1, j_1, c_2 and j_2 into many cases by considering whether or not they are distinct, and the pairwise distances between them. Here the case we will consider is the one for which the four edges c_1, j_1, c_2 and j_2 are distinct edges of the original tree T, and are pairwise at least distance three apart.

Let $\mathbb{S}(T)$ be the subset of $\mathbb{T}(T)$ where $c_2, j_2 \in E(T)$, and the four edges c_1, j_1, c_2, j_2 are pairwise at least distance three apart in T.

The following lemma shows that in order to prove Theorems 6.2 and 6.3 it suffices to consider only pairs of SPR operations with cut and join edges in $\mathbb{S}(T)$.

Lemma 6.5. *Let $T \in UB(n)$. Then*

$$|\mathbb{S}(T)| = \frac{2}{3}n^4 + O(n^3)$$
$$|\mathbb{T}(T) - \mathbb{S}(T)| = O(n^3).$$

Proof. For n sufficiently large, it is possible to choose the edges c_1, j_1, c_2 and j_2 in T such that $(c_1, c_2, j_1, j_2) \in \mathbb{S}(T)$. To determine the size of $\mathbb{S}(T)$, we count the number of sets of four internal edges of T, where all pairs of edges in the set are at least distance three apart. There are $2n-3$ choices for edge c_1, since this is the number of edges in T (this follows from Lemma 2.1). The maximum number of choices for j_1 is $(2n-3-7)$ (this can occur if c_1 is a pendant edge). The minimum number of choices for edge j_1 is $(2n-3-29)$ (this can occur if c_1 is an internal edge). The maximum number of choices for c_2 is $(2n-3-7-6)$ (this can occur if c_1 and j_1 are both pendant edges). The minimum number of choices for c_2 is $(2n-3-2(29)$ (this can occur if both c_1 and j_1 are internal edges). A similar process determines upper and lower bounds on the number of choices for edge j_2. We divide by the number of ways to order the four edges. Therefore

$$|\mathbb{S}(T)| \geq \frac{1}{4!}(2n-3)(2n-3-29)(2n-3-2(29))(2n-3-3(29)) = \frac{2}{3}n^4 + O(n^3), \text{ and}$$

$$|\mathbb{S}(T)| \leq \frac{1}{4!}(2n-3)(2n-3-7)(2n-3-7-6)(2n-3-7-2(6)) = \frac{2}{3}n^4 + O(n^3).$$

Now we consider $\mathbb{T}(T) - \mathbb{S}(T)$. Determining $|\mathbb{T}(T) - \mathbb{S}(T)|$ is similar to determining $|\mathbb{S}(T)|$, however for at least one of the four cut and join edges, instead of counting the number of edges at least distance three from those already chosen, we count the number within distance two of those already chosen, and therefore obtain a constant factor instead of a linear factor. Let M be a maximal subset of the the edges $\{c_1, c_2, j_1, j_2\}$ such that the edges in M are pairwise distance at least three apart in T, where $|M| = m < 4$. Suppose we first choose the edges in M. From the argument above we can see that the number of such choices is $O(n^m)$. The remaining $4 - m \geq 1$ edges must be chosen from edges within distance two of those already chosen. The number of these choices depends only on the number and location of the m edges already chosen, and not on n. Hence

$$|\mathbb{S}(T)| = \frac{2}{3}n^4 + O(n^3), \text{ and } |\mathbb{T}(T) - \mathbb{S}(T)| = O(n^3).$$

□

Lemma 6.5 tells us that the highest order term in the expression for the size of $\mathbb{S}(T)$ is $O(n^4)$. Note that instead of requiring the edges in $\mathbb{S}(T)$ to be at least distance three apart we could have made them distance k apart for any $k \in \mathbb{Z}^+$ and Lemma 6.5 would still hold. We have chosen to consider distance three, because if pairs of these four edges are within distance two of each other, then there are more cases to consider in order to determine exactly when two different pairs of SPR operations produce the same tree. To determine only the $O(n^4)$ term in the expression for the size of the second SPR neighbourhood, we can ignore all cases where there exist edges $e, f \in \{c_1, c_2, j_1, j_2\}$ such that $d_T(e, f) \leq 2$.

However, we can't simply take the highest order term in the expression for the size of $\mathbb{S}(T)$ as the highest order term in the expression for the size of the second SPR neighbourhood of a tree T, as there may be cases where two different pairs of SPR operations produce the same tree (duplicates), or when a pair of SPR operations produces a first SPR neighbour of T. To prove Theorem 6.2 and Theorem 6.3 we need to know precisely when these two situations arise.

In Lemma 6.7 and 6.8 we show that there are no cases where a pair of SPR operations with cut and join edges in $\mathbb{S}(T)$ yield a first SPR neighbour, and that whether two different pairs of operations produce the same tree is dependent on whether or not the cut and join edges j_2, c_1, c_2, and j_1 lie on a path in this order. Note that if the edges do lie on a path in this order, they also lie on a path in the reverse order. We first require a result about how many ways one SPR operation on a tree $T \in UB(n)$ can reduce the distance between two subtrees of T.

Lemma 6.6. *Let $T \in UB(n)$. Suppose there exist subtrees A and B of T (not necessarily pendant or maximal), such that $d_T(A, B) = k$, and if A or B is an internal subtree, then it has at least one internal edge. Let a and b be vertices of degree two in A and B respectively, such that $d_T(a, b) = k$. Call the two pendant edges of the $(a - b)$-path P, e and f respectively. Let $T' \in UB(n)$ with the same leaf set as T, such that A and B are subtrees of T' (with vertices a and b respectively of degree two), $d_{T'}(A, B) = 2$, and $d_{T'}(a, b) = 2$. Then*

(i) for $k \geq 4$, if $T' = SPR(T, (c_1, j_1))$ then $\{c_1, j_1\} = \{e, f\}$, and

(ii) for $k \geq 5$, if $T' = SPR(T, (c_1, j_1), (c_2, j_2))$ with $(c_1, c_2, j_1, j_2) \in \mathbb{S}(T)$ then $\{c_1, j_1\} = \{e, f\}$ or $\{c_2, j_2\} = \{e, f\}$.

Proof. Fig. 11 shows trees T and T'.

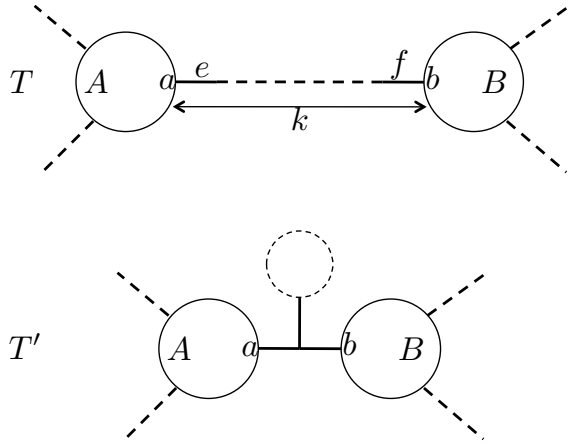

Figure 11: Trees T and T'.

(i) Let $T'' = SPR(T, (c_1, j_1))$. We show that $T'' \neq T'$ unless $\{c_1, j_1\} = \{e, f\}$. First suppose that edge c_1 is in subtree A or B. Then either $T'' = T$ (if we regraft in the same place), or $T'' \neq T$ since the subtree (A or B) is not in T''. Likewise for edge j_1.

Now suppose that c_1 and j_1 are not edges of A or B. Hence A and B are both subtrees of T''. If we assume that edge c_1 is not in P or incident to P, then $d_{T''}(A, B) \geq 5$ if j_1 is an edge of P, else $d_{T''}(A, B) \geq 4$. Therefore $T'' \neq T'$. If c_1 is incident to P then deleting edge c_1 creates a vertex of degree two in P, which is suppressed by the SPR operation. Hence $d_{T''}(A, B) \geq 4$ if j_1 is an edge of P, otherwise $d_{T''}(A, B) \geq 3$, and so $T'' \neq T'$.

Now suppose that j_1 is not an edge of A or B, and c_1 is an edge of P. If $c_1 \notin \{e, f\}$ then $d_{T''}(A, B) \geq 3$ if j_1 is incident to A or B, otherwise $d_{T''}(A, B) \geq 4$, and so $T'' \neq T'$. Finally

suppose that $c_1 \in \{e, f\}$, and j_1 is not incident to A or B. Then $d_{T''}(A, B) \geq 3$ and $T'' \neq T'$. If j_1 is adjacent to A or B but $j_1 \notin \{e, f\}$ (which can occur if A or B is an internal subtree) then $d_{T''}(A, B) = 2$ but $d_{T''}(a, b) > 2$, since the internal subtree has at least one internal edge. Hence if $T'' = T'$ then $\{c_1, j_1\} = \{e, f\}$.

(ii) Let $T'' = SPR(T, (c_1, j_1)(c_2, j_2))$, where $(c_1, c_2, j_1, j_2) \in \mathbb{S}(T)$. We show that $T'' \neq T'$ unless $\{c_1, j_1\} = \{e, f\}$ or $\{c_2, j_2\} = \{e, f\}$. As in (i) if any of the four cut and join edges are in the subtrees A or B in T, then that subtree is not a subtree of T'', so $T'' \neq T'$. In (i) we saw that if the cut edge of an operation is not in P or incident to P in T then the operation does not reduce the distance between A and B. As in (i), an operation with a cut edge incident to P, reduces the distance between A and B by at most one. Hence if neither cut edge c_1 or c_2 is in P, we have $d_{T''}(A, B) \geq 3$, and so $T'' \neq T'$.

Suppose that c_1 is not an edge of P, but c_2 is. Then by (i), if $T_1 = SPR(T, (c_1, j_1))$ then $d_{T_1}(A, B) \geq 4$. By (i), if $T'' = T'$ then $\{c_2, j_2\} = \{e, f\}$.

Now suppose that $c_1 \notin \{e, f\}$ is an edge of P. Then by (i), in the tree $T_1 = SPR(T, (c_1, j_1))$ we have $d_{T_1}(A, B) \geq 3$ if j_1 is incident to A or B, otherwise $d_{T_1}(A, B) \geq 4$. If $d_{T_1}(A, B) \geq 4$ then by (i), the second operation cannot result in T' unless $\{c_2, j_2\} = \{e, f\}$. If $d_{T_1}(A, B) = 3$, then since $(c_1, c_2, j_1, j_2) \in \mathbb{S}(T)$, the edges c_2 and j_2 cannot be in or incident to the shortest path between A and B in T_1. Hence $d_{T''}(A, B) = 3$, and $T'' \neq T'$.

Finally, suppose that $c_1 \in \{e, f\}$. If j_1 is not incident to A or B then in the tree $T_1 = SPR(T, (c_1, j_1))$ we have $d_{T_1}(A, B) \geq 3$. Again, if $d_{T_1}(A, B) \geq 4$ then by (i), the second operation cannot result in T' unless $\{c_2, j_2\} = \{e, f\}$. If $d_{T_1}(A, B) = 3$ then since $(c_1, c_2, j_1, j_2) \in \mathbb{S}(T)$, the edges c_2 and j_2 cannot be in or incident to the shortest path between A and B in T_1. Hence $d_{T''}(A, B) = 3$, and $T'' \neq T'$. If j_1 is adjacent to A or B, but $j_1 \notin \{e, f\}$ then $d_{T_1}(A, B) = 2$ but $d_{T_1}(a, b) > 2$. Again, c_2 and j_2 cannot be edges on or incident to the path between A and B in T_1, so $d_{T''}(a, b) > 2$. The only remaining case is $\{c_1, j_1\} = \{e, f\}$.

Therefore $T'' \neq T'$ unless $\{c_1, j_1\} = \{e, f\}$ or $\{c_2, j_2\} = \{e, f\}$.

□

Lemma 6.7. *Let $T \in UB(n)$, and suppose that $T' = SPR(T, (c_1, j_1))$ and $T'' = SPR(T, (c_1, j_1), (c_2, j_2))$ where $(c_1, c_2, j_1, j_2) \in \mathbb{S}(T)$. Suppose that the edges j_2, c_1, c_2, and j_1 lie on a path in T in this order.*

Then

(i) $T'' \notin N_{SPR}(T)$, *and*

(ii) for all other choices of edges $(c_1', c_2', j_1', j_2') \in \mathbb{S}(T)$ *where* $(c_1', c_2', j_1', j_2') \neq (c_1, c_2, j_1, j_2)$, *we have*

$$T'' \neq SPR(T, (c_1', j_1'), (c_2', j_2')).$$

Proof. Since the four cut and join edges lie on a path in T, the rest of the tree can be partitioned into five subtrees (two pendant and three internal) connected by these four edges.

Consider the forest $T \setminus \{c_1, j_1, c_2, j_2\}$. It has components A, B, C, D and E which are subtrees of T. Edge j_2 is incident to A and B, edge c_1 is incident to B and C, edge c_2 is incident to C and D, and edge j_1 is incident to D and E. Fig. 12 shows T, T' and T''. Each of the internal subtrees B, C and D have at least three internal edges, as all pairs of the four cut and join edges are at least distance three apart. Let b be the endpoint of c_1 that is in B, and c be the endpoint of c_2 that is in C.

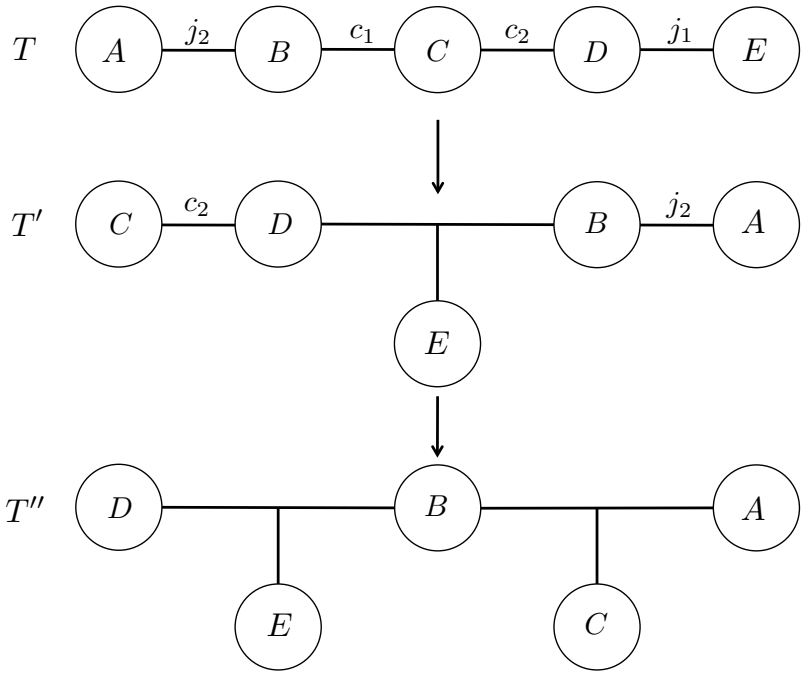

Figure 12: Tree $T' = SPR(T, (c_1, j_1))$ and $T'' = SPR(T, (c_1, j_1), (c_2, j_2))$.

(i) From the above we have $d_T(B, E) = d_T(b, E) \geq 9$ and $d_{T''}(B, E) = d_{T''}(b, E) = 2$. Therefore if T'' is a first SPR neighbour of T, then either $T'' = SPR(T, (c_1, j_1)) = T'$ or $T'' =$

$SPR(T,(j_1,c_1))$ by Lemma 6.6 [1]. We also have $d_{T''}(A,C)=2$ and $d_{T'}(A,C)\geq 10$, so $T''\neq T'$. In $T_1=SPR(T,(j_1,c_1))$, $d_{T_1}(A,C)\geq 6$, so $T''\neq T_1$. Therefore T'' is not a first SPR neighbour of T.

(ii) Considering $(c_1',c_2',j_1',j_2')\in\mathbb{S}(T)$, let $T_1=SPR(T,(c_1',j_1'))$ and $T_2=SPR(T,(c_1',j_1'),(c_2',j_2'))$. We show that $T_2=T''$ implies that $(c_1',c_2',j_1',j_2')=(c_1,c_2,j_1,j_2)$. As before, we have $d_{T''}(B,E)=d_{T''}(b,E)=2$. Since $d_T(B,E)=d_T(b,E)\geq 9$, if $T''=T_2$, then the cut and join edges for one of the operations must be c_1 and j_1 by Lemma 6.6. There are four cases to consider.

(a) First suppose that $(c_1',j_1')=(c_1,j_1)$. Then $T_1=T'$. Since all SPR operations on T' result in distinct neighbours (by Lemma 6.1), $T''=SPR(T,(c_1,j_1),(c_2',j_2'))$ only if $(c_2',j_2')=(c_2,j_2)$.

(b) Now suppose that $(c_1',j_1')=(j_1,c_1)$. Then $d_{T_1}(B,C)=d_{T_1}(B,E)=d_{T_1}(C,E)=2$. The edges c_2' and j_2' must be distance three or more from c_1 and j_1 in T. If c_2' or j_2' are in one of the subtrees B, C and E, then this subtree is not a subtree of T_2 and $T_2\neq T''$. If neither c_2' or j_2' are in one of the subtrees B, E or C, then $d_{T_2}(B,C)=d_{T_2}(B,E)=d_{T_2}(C,E)=2$. However, $d_{T''}(C,E)\geq 7$ so $T_2\neq T''$.

We now assume that $\{c_2',j_2'\}=\{c_1,j_1\}$. We have $d_T(A,C)\geq 5$ and $d_{T''}(A,C)=2$. If $T_2=T''$, then by the proof of Lemma 6.6, $c_1'\in\{c_2,j_2\}$ and j_1' is incident to either A or C. Since $j_1'\neq c_2'$, we have $j_1'\neq c_1$ which means that $j_1'\in\{c_2,j_2\}$. Therefore $\{c_1',j_1'\}=\{c_2,j_2\}$.

(c) Suppose that $(c_1',j_1')=(j_2,c_2)$. Then $d_{T_1}(A,C)=d_{T_1}(A,D)=2$. Regardless of whether the second SPR operation involves pruning B or E in T, $d_{T_2}(A,C)=d_{T_2}(A,D)=2$, and $d_{T_2}(A,B)>2$. However, $d_{T''}(A,B)=2$, so $T_2\neq T''$.

(d) Finally suppose that $(c_1',j_1')=(c_2,j_2)$. Then $d_{T_1}(A,C)\geq 6$. In T_1, the subtrees at the ends of the (c_1-j_1)-path are C and E. So $d_{T_2}(C,E)=2$, and $T_2\neq T''$.

Therefore $T_2=T''$ implies that $(c_1',c_2',j_1',j_2')=(c_1,c_2,j_1,j_2)$.

□

[1] Note that Lemma 6.6 applies when $d_T(B,E)=d_T(b,x)\geq 9$ and $d_{T''}(B,E)=d_{T''}(b,x)=2$ where x is a vertex of degree two in E. However since E is a pendant subtree with only one vertex of degree two we simply use $d_T(b,E)$ instead of $d_T(b,x)$ for simplicity. This occurs in other places throughout the proofs of Lemma 6.7 and Lemma 6.8.

Lemma 6.8. *Let $T \in UB(n)$ and suppose that $T' = SPR(T, (c_1, j_1))$ and $T'' = SPR(T, (c_1, j_1), (c_2, j_2))$ where $(c_1, c_2, j_1, j_2) \in \mathbb{S}(T)$. Suppose that there is no path in T in which the edges j_2, c_1, c_2, and j_1 appear in this order. Then*

(i) $T'' \notin N_{SPR}(T)$, and

(ii) for all choices of edges $(c_1', c_2', j_1', j_2') \in \mathbb{S}(T)$, $(c_1', c_2', j_1', j_2') \neq (c_1, c_2, j_1, j_2)$, we have

$$T'' = SPR(T, (c_1', j_1'), (c_2', j_2'))$$

iff $(c_1', c_2', j_1', j_2') = (c_2, c_1, j_2, j_1)$.

Proof. Let C_1 and D_1 be the subtrees rooted at the endpoints c and d respectively of the (c_1, j_1)-path in T. Then $d_{T'}(C_1, D_1) = 2$. Now let C and D be subtrees of C_1 and D_1 respectively for which $d_{T''}(C, D) = 2$. Because neither c_2 or j_2 is within distance two of c_1 or j_1, C and D each have at least three internal edges. Therefore C and D are (not necessarily pendent) subtrees such that $d_T(C, D) = d_T(c, d) \geq 5$ and $d_{T''}(C, D) = d_{T''}(c, d) = 2$.

Let A_1 and B_1 be subtrees rooted at the endpoints a and b respectively of the (c_2, j_2)-path in T. Let the subtrees at the endpoints of the (c_2, j_2)-path in T' be A_2 and B_2 respectively. Note that a and b are the endpoints of this path in T'. Now let $A = A_1 \cap A_2$ and $B = B_1 \cap B_2$. Since $(c_1, c_2, j_1, j_2) \in \mathbb{S}(T)$, A and B have at least three internal edges. So A and B are (not necessarily pendant) subtrees of T rooted at either end of the (c_2, j_2)-path in T. We have $d_T(A, B) = d_T(a, b) \geq 5$. Since c_1 can't be within distance two of either c_2 or j_2, $d_{T'}(A, B) = d_{T'}(a, b) \geq 5$. Finally $d_{T''}(A, B) = d_{T''}(a, b) = 2$.

(i) From above we have $d_T(C, D) = d_T(c, d) \geq 5$, but $d_{T''}(C, D) = d_{T''}(c, d) = 2$. By Lemma 6.6, $T'' \in N_{SPR}(T)$ implies $T'' = SPR(T, (c_1, j_1)) = T'$ or $T'' = SPR(T, (j_1, c_1))$. However $d_{T'}(A, B) = d_{T'}(a, b) \geq 5$ and $d_{T''}(A, B) = d_{T''}(a, b) = 2$, so by Lemma 6.6, if $T'' \in N_{SPR}(T)$ then either $T'' = SPR(T, (c_2, j_2))$ or $T'' = SPR(T, (j_2, c_2))$. Since $(c_1, c_2, j_1, j_2) \in \mathbb{S}(T)$, these four trees are distinct, and $T'' \notin N_{SPR}(T)$.

(ii) As in Lemma 6.7, if

$$T'' = SPR(T, (c_1', j_1'), (c_2', j_2'))$$

for $(c_1', c_2', j_1', j_2') \in \mathbb{S}(T)$, $(c_1', c_2', j_1', j_2') \neq (c_1, c_2, j_1, j_2)$, then $\{\{c_1', j_1'\}, \{c_2', j_2'\}\} = \{\{c_1, j_1\}, \{c_2, j_2\}\}$ by Lemma 6.6. We consider all possible cases. Let $T_1 = SPR(T, (c_1', j_1'))$ and $T_2 = SPR(T, (c_1', j_1'), (c_2', j_2'))$.

(a) First let $(c_1', j_1') = (c_2, j_2)$ and $(c_2', j_2') = (c_1, j_1)$. The first SPR operation on T prunes and regrafts A_1 so that $d_{T_1}(A_1, B_1) = 2$. Because the edges j_1, c_2, c_1, j_2 do not lie on a path in T in this order, the endpoints of the $(c_1 - j_1)$-path in T_1 are c and d. Hence $d_{T_2}(A, B) = d_{T_2}(a, b) = d_{T_2}(C, D) = d_{T_2}(c, d) = 2$ and case analysis shows that $T_2 = T''$. So

$$T'' = SPR(T, (c_1, j_1), (c_2, j_2)) = SPR(T, (c_2, j_2), (c_1, j_1)).$$

(b) Now consider the case where $(c_1', j_1') = (c_1, j_1)$. Then $T_1 = T'$. Since we know that SPR operations on T with different cut and join edges result in distinct trees (by Theorem 6.1), we have

$$SPR(T, (c_1, j_1), (j_2, c_2)) \neq SPR(T, (c_1, j_1), (c_2, j_2)) = T''.$$

Similarly,

$$SPR(T, (c_2, j_2), (j_1, c_1)) \neq SPR(T, (c_2, j_2), (c_1, j_1)) = T''.$$

(c) Let X be the subtree of T such that $d_T(X, D) = 2$ and X does not contain edge c_1. Then $d_{T'}(C, D) = 2$ and $d_{T'}(C, X) = d_{T'}(D, X) = 3$. Since the cut and join edges for the second SPR operation must be at least distance three from c_1 and j_1 in T there is a subtree of X which we denote X', such that $d_{T''}(C, X') = d_{T''}(D, X') = 3$. Suppose that $(c_1', j_1') = (j_1, c_1)$. Then $d_{T_1}(C, X) = d_{T_1}(D, X) \geq 4$. Again, there exists a subtree X'' of X such that $d_{T_2}(C, X'') = d_{T_2}(D, X'') \geq 4$. Since $(c_1, c_2, j_1, j_2) \in \mathbb{S}(T)$, the intersection between X' and X'' is non-empty. Therefore $T_2 \neq T''$. The same argument applies if we consider $(c_1', j_1') = (j_2, c_2)$.

Therefore

$$T'' = SPR(T, (c_1, j_1), (c_2, j_2)) = SPR(T, (c_2, j_2), (c_1, j_1)),$$

but for all other choices of edges $(c_1', c_2', j_1', j_2') \in \mathbb{S}(T)$, $(c_1', c_2', j_1', j_2') \neq (c_1, c_2, j_1, j_2)$, we have $T'' \neq SPR(T, (c_1', j_1'), (c_2', j_2'))$.

□

We have now established that there are no pairs of SPR operations that produce a first SPR neighbour of a tree T. The only case where two different pairs of SPR operations produce the same tree is when there is no path in T with the edges j_2, c_1, c_2, j_1 in the order listed, and

$$SPR(T, (c_1, j_1), (c_2, j_2)) = SPR(T, (c_2, j_2), (c_1, j_1)).$$

We now count how many ways the edges j_2, c_1, c_2 and j_1 can appear in a path in a binary tree T in the order given, with the four edges pairwise at least distance three apart. Let this quantity be $P(T)$. We need to know the number of paths of all lengths greater than or equal to thirteen in T, which is dependent on tree shape. However for a caterpillar and a balanced tree the number of paths of any length is completely determined by the number of leaves of the tree.

Lemma 6.9. *For $n \geq 4$:*

(i) A caterpillar with n leaves, has $4(n-k)$ paths of length k for $3 \leq k \leq n-1$.

(ii) Let

$$f(k) = \begin{cases} 3\left(2^{\frac{k}{2}-1}\right)\left(n - 2^{\frac{k}{2}}\right), & k \text{ even;} \\ 2^{\frac{k+1}{2}}\left(n - 3\left(2^{\frac{k-3}{2}}\right)\right), & k \text{ odd.} \end{cases}$$

A balanced tree with $n = 2^i$ leaves ($i \geq 2$) has $f(k)$ paths of length k for $3 \leq k \leq 2i-1$, and a balanced tree with $n = 3 \cdot 2^i$ leaves ($i \geq 1$) has $f(k)$ paths of length k for $3 \leq k \leq 2(i+1)$.

Proof.

(i) A caterpillar T has a single path of $n-3$ internal edges. Now $p_{k-2}(T)$ is the number of ways to select $k-2$ of these internal edges so that they are adjacent. This is given by $p_{k-2}(T) = (n-3) - (k-2) + 1 = n-k$. Then, because T is binary, $P_k(T) = 4(n-k)$, for $k \geq 3$.

(ii) If T is a balanced tree with n leaves, then it has $c = \frac{n}{2}$ cherries. Let $\bar{P}_k(n)$ be the number of paths of length k in a balanced tree with n leaves, and $\bar{p}_k(n)$ be the number of internal paths of length k in a balanced tree with n leaves. The number of internal paths of length k in T is given by the number of paths of length k in T' where T' is the subtree induced by the internal vertices of T. Since T' has $\frac{n}{2}$ leaves,

$$\bar{p}_k(n) = \bar{P}_k\left(\frac{n}{2}\right),$$

provided $n \geq 6$. As in (i),

$$\bar{P}_k(n) = 4\bar{p}_{k-2}(n).$$

We have $\bar{p}_2(n) = n + c - 6 = 3\left(\frac{n}{2} - 2\right)$ by Corollary 3.4, and so if k is even then

$$\begin{aligned} \bar{P}_k(n) &= 3\left(2^{k-2}\right)\left(\frac{n}{2^{\frac{k}{2}-1}} - 2\right) \\ &= 3\left(2^{\frac{k}{2}-1}\right)\left(n - 2^{\frac{k}{2}}\right). \end{aligned}$$

We have $\bar{p}_1(n) = n - 3$ by Lemma 2.1, so if k is odd then

$$\begin{aligned} \bar{P}_k(n) &= 2^{k-1}\left(\frac{n}{2^{\frac{k-1}{2}-1}} - 3\right) \\ &= 2^{\frac{k+1}{2}}\left(n - 3\left(2^{\frac{k-3}{2}}\right)\right). \end{aligned}$$

Now if $n = 2^i$, the maximum path length in the tree is given by $2i - 1$, and if $n = 3 \cdot 2^i$ then the maximum path length in the tree is given by $2(i + 1)$.

□

Now that we know the number of paths in a caterpillar or balanced tree of any given length, we can determine the size of $P(T)$. We are now ready to to prove Theorem 6.2.

Proof of Theorem 6.2

Proof. Suppose that T has a path P of length k, $k \geq 13$. Fix the two pendant edges of P as j_2 and j_1 so that j_2 is the first edge in P, and j_1 is the k^{th} edge in P. All pairs of the edges j_2, c_1, c_2, and j_1 must be distance three or more apart and in the order given. So $d_T(c_1, j_2) \geq 3$ and $d_T(c_1, j_1) \geq 7$. If c_1 is the m^{th} edge in P then $5 \leq m \leq k-8$. Now if c_2 is the j^{th} edge in P, then $m+4 \leq j \leq k-4$, so there are $(k-4)-(m+4)+1 = k-m-7$ possible choices for the location of c_2. Finally, it does not matter at which endpoint of P we begin counting. So the number of ways of arranging the four edges on this path is

$$R_k = 2\sum_{m=5}^{k-8}(k-m-7) = (k-11)(k-12).$$

(i) By Lemma 6.9, T has $4(n-k)$ paths of length k for $k \geq 3$. Hence for a caterpillar,

$$\begin{aligned} P(T) &= \sum_{k=13}^{n-1} 4(n-k)(k-11)(k-12) \\ &= \frac{1}{3}n^4 + O(n^3). \end{aligned}$$

We know by Lemma 6.7 and Lemma 6.8 that if we count the number of ways to choose the edges $(c_1, c_2, j_1, j_2) \in \mathbb{S}(T)$, then in the cases not counted by $P(T)$ we count every second neighbour twice. For the cases that are counted by $P(T)$ we obtain no duplicate trees. So by Lemma 6.5,

$$\begin{aligned} |N^2_{SPR}(T)| &= \frac{1}{2}\left(\frac{2}{3}n^4 + O(n^3) - P(T)\right) + P(T) \\ &= \frac{1}{2}\left(\frac{2}{3}n^4 + P(T)\right) + O(n^3) \\ &= \frac{1}{2}\left(\frac{2}{3}n^4 + \frac{1}{3}n^4\right) + O(n^3) = \frac{1}{2}n^4 + O(n^3). \end{aligned}$$

(ii) Similarly for a balanced tree T with $n = 3(2)^i$ leaves ($i \geq 1$), we can sum over even and odd

path lengths (see Lemma 6.9) to obtain

$$\begin{aligned}
P(T) &= \sum_{k=13}^{n-1} P_k(T)(k-11)(k-12) \\
&= \sum_{m=7}^{\log_2(\frac{n}{3})+1} \left(3\left(2^{m-1}\right)(n-2^m)(2m-11)(2m-12)\right) + \\
&\quad \sum_{m=7}^{\log_2(\frac{n}{3})+1} \left(2^m\left(n-3\left(2^{m-2}\right)\right)(2m-12)(2m-13)\right) \\
&= \frac{8}{\ln(2)^2} n^2 \ln(n)^2 + O(n^2 \ln(n)) \\
&= O(n^2 \ln(n)^2) = O(n^3).
\end{aligned}$$

If T is a balanced tree with $n = 2^i$ leaves ($i \geq 2$), then we instead have

$$\begin{aligned}
P(T) &= \sum_{m=7}^{\log_2(\frac{n}{4})+1} \left(3\left(2^{m-1}\right)(n-2^m)(2m-11)(2m-12)\right) + \\
&\quad \sum_{m=7}^{\log_2(\frac{n}{4})+2} \left(2^m\left(n-3\left(2^{m-2}\right)\right)(2m-12)(2m-13)\right) \\
&= \frac{8}{\ln(2)^2} n^2 \ln(n)^2 + O(n^2 \ln(n)) = O(n^3).
\end{aligned}$$

Therefore for any balanced tree T,

$$|N^2_{SPR}(T)| = \frac{1}{2}\left(\frac{2}{3}n^4 + P(T)\right) + O(n^3) = \frac{1}{3}n^4 + O(n^3).$$

□

This shows that the size of the second SPR neighbourhood of a tree cannot be uniquely determined by the number of leaves of the tree. To show that the number of leaves and cherries is insufficient we consider Theorem 6.3.

Proof of Theorem 6.3

Proof. Suppose that $n = 3m$ and $c = 3$, where $m \geq 7$. Consider the tree T_1 of Type 1, with n leaves and c cherries (see Fig. 9). Let C_{xy} be the caterpillar formed by the path between vertices x and y in T_1 and all of the edges incident to vertices on that path. Let a, b and d be the roots of the three cherries of T_1, such that $d_{T_1}(a, b) = 2$. Let c be the vertex in T_1 that is not adjacent to a leaf. Both of the caterpillars C_{ad} and C_{bd} have $n - 1$ leaves. If we find $P(C_{ad})$ and $P(C_{bd})$ then we will have

found every way of selecting the edges c_1, c_2, j_1 and j_2 so that all four edges are on a path in the order j_2, c_1, c_2, j_1. Eliminating double counting, we have

$$P(T_1) = P(C_{ad}) + P(C_{bd}) - P(C_{cd}) = 2P(C_{ad}) - P(C_{cd}).$$

We do not consider the caterpillar C_{ab} because it is too short to have any paths of length thirteen or more. So by Theorem 6.2,

$$P(T_1) = \frac{2}{3}(n-1)^4 - \frac{1}{3}(n-2)^4 + O(n^3) = \frac{1}{3}n^4 + O(n^3).$$

Now let T_2 be the tree of Type 2 with n leaves, c cherries and maximum path length $2m$ (see Fig. 10). Let a, b and d be the roots of the three cherries of T_2, and let c be the vertex in T_2 that is not adjacent to a leaf. Then by the same process as above,

$$P(T_2) = P(C_{ad}) + P(C_{bd}) + P(C_{ab}) - P(C_{ac}) - P(C_{bc}) - P(C_{cd}) = 3P(C_{ad}) - 3P(C_{ac}).$$

Now C_{ad} has $2m+1$ leaves and C_{ac} has $m+2$ leaves, so

$$\begin{aligned} P(T_2) &= (2m+1)^4 - (m+2)^4 + O(n^3) \\ &= (\frac{2}{3}n+1)^4 - (\frac{1}{3}n+2)^4 + O(n^3) \\ &= \frac{5}{27}n^4 + O(n^3). \end{aligned}$$

Therefore $|N^2_{SPR}(T_1)| = \frac{1}{2}n^4 + O(n^3)$ and $|N^2_{SPR}(T_2)| = \frac{23}{54}n^4 + O(n^3)$.

□

Since T_1 and T_2 have the same number of leaves and cherries, it is clear that other properties of the tree T would be required to get an exact formula for the highest order term of $|N^2_{SPR}(T)|$.

7 Tree Bisection and Reconnection

A *tree bisection and reconnection (TBR)* operation on a tree $T \in UB(n)$ is performed by:

1. Deleting an internal edge $e = \{x, y\}$ in T, leaving two components X (containing the vertex x) and Y (containing the vertex y).

2. Suppressing the vertices x and y to give new edges e_x and e_y respectively. We call the resulting components X' and Y' respectively.

3. Inserting an edge f connecting a pair of edges f_X in X' and f_Y in Y' to give a tree $T' \in UB(n)$.

Tree T' is a first *TBR neighbour* of T. A single TBR operation can be seen in Fig. 13. We call e the *cut edge* of the TBR operation, and f_X and f_Y are the *join edges*. Note that if either $e_x = f_X$ or $e_y = f_Y$ then the TBR operation is an SPR operation.

Consider a graph G in which each vertex represents a tree in $UB(n)$ and there is an edge between the vertices representing trees T_1 and T_2 if they are first TBR neighbours. The *TBR distance* between T_1 and T_2, $\delta_{TBR}(T_1, T_2)$, is the distance between the two vertices representing T_1 and T_2 in G.

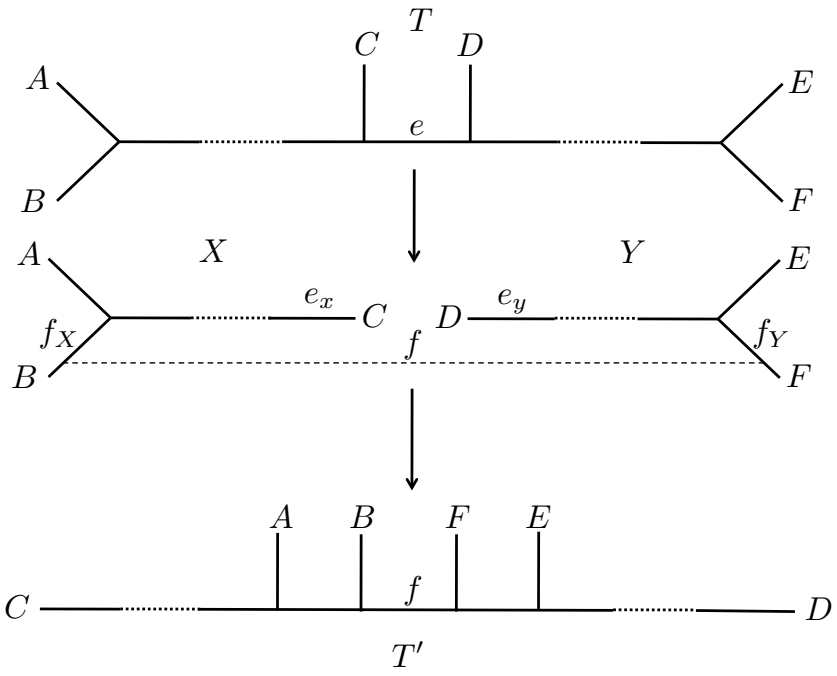

Figure 13: Tree T' is a first TBR neighbour of T resulting from a TBR operation with cut edge e and join edges f_X and f_Y.

7.1 First Neighbourhood

Humphries and Wu (2013) showed that the size of the first TBR neighbourhood of a tree $T \in UB(n)$ ($n \geq 4$) is given by

$$|N_{TBR}(T)| = 4 \sum |A||B| - (4n-2)(n-3)$$

where the sum is over all non-trivial splits $\{A, B\}$ of T. In Theorem 7.1 we take a slightly different approach to calculating the size of the first neighbourhood by considering paths of different lengths in T. We will then show in Theorem 7.2 that our result is equivalent to that of Humphries and Wu

(2013).

Theorem 7.1. *Let $T \in UB(n)$ $(n \geq 3)$. Then*

$$|N_{TBR}(T)| = 2(n-3)(2n-7) + \sum_{k=5}^{M}(k-4)P_k(T)$$

where M is the diameter of T and $P_k(T)$ is the number of paths of length k in T.

Proof. If $n = 3$ then T has no internal edges (by Lemma 2.1), so the result is trivially true. Now assume that $n \geq 4$. Let T' be a TBR neighbour of T. We consider the possible choices of join edges f_X and f_Y in relation to e_x and e_y. There are three cases to consider.

1. Let $f_X = e_x$ and $f_Y = e_y$. Then $T' = T$.

2. Suppose that either $f_X = e_x$ or $f_Y = e_y$, but not both. Then this is an SPR operation, so there are $2(n-3)(2n-7)$ neighbours by Theorem 6.1.

3. Finally suppose that $f_X \neq e_x$ and $f_Y \neq e_y$. Then $f_X, f_Y \in E(T)$, and e is an edge on the $(f_X - f_Y)$-path P in T. Suppose that e is adjacent to f_X, then deleting edge e results in a vertex of degree two incident to f_X in X. Hence, after suppressing x in X to obtain X', $f_X \notin E(X')$. However by definition $f_X \in E(X')$ so e is not adjacent to f_X. Similarly e is not adjacent to f_Y. Consider choosing the edges f_X and f_X so that $d_T(f_X, f_Y) \geq 3$. Let k be the length of P. Since e is not equal to or adjacent to f_X or f_Y, there are $k-4$ possible choices for e in the path P. We then sum over all possible paths of length five or greater in T.

Therefore, provided no two TBR operations produce the same tree, we have

$$|N_{TBR}(T)| = 2(n-3)(2n-7) + \sum_{k=5}^{M}(k-4)P_k(T)$$

where M is the maximum length of any such path (the diameter of T).

We now show that given a particular TBR operation on T from Case 3, there is no other TBR operation that yields the same neighbour.

Suppose we perform a TBR operation on T where $f_X \neq e_x$ and $f_Y \neq e_y$ (Case 3), and call the resulting tree T'. Let T_X and T_Y be the subtrees of T for which $T_X \subseteq X$, $T_Y \subseteq Y$, $d_T(T_X, e) = d_T(T_Y, e) = 1$, $f_X \notin E(T_X)$, and $f_Y \notin E(T_Y)$. Let T'_X and T'_Y be the maximal internal subtrees of T incident to f_X and f_Y respectively, for which $d_T(T'_X, e) = d_T(T'_Y, e) = 1$. We let A denote the

pendant subtree incident to f_X, and B denote the pendant subtree incident to f_Y. Note that any of the subtrees T_X, T_Y, A and B may consist of only a single vertex, while T'_X and T'_Y may contain only one leaf and one internal vertex. Both T and T' can be seen in Fig. 14. All six of these labelled subtrees are subtrees of T'. We have $d_{T'}(T_X, T_Y) \geq 5$ and $d_{T'}(T_X, T'_X) = d_{T'}(T_Y, T'_Y) = 1$.

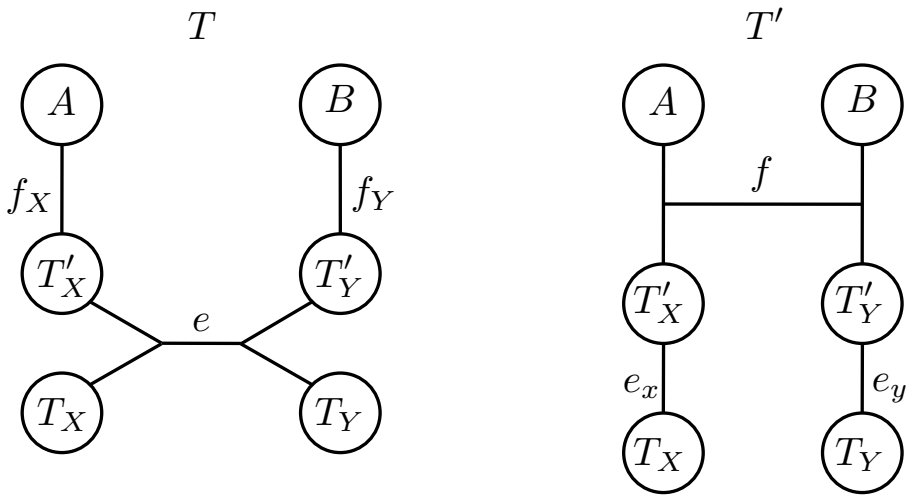

Figure 14: Trees T and T' where $\delta_{TBR}(T, T') = 1$.

Now suppose we perform a TBR operation on T to obtain T'', with cut edge e' and join edges g_X and g_Y. We will show that $T'' \neq T'$ for any choice of the edges e', g_X, g_Y except $e' = e$, $\{g_X, g_Y\} = \{f_X, f_Y\}$, by considering the distances between subtrees.

First suppose that the cut edge e' is in one of the six labelled subtrees. Then either $T'' = T$ or the corresponding subtree is not a subtree of T''. Hence $T'' \neq T'$.

Suppose that $e' = f_X$. Then in the forest F obtained by deleting e', $d_F(T_X, T_Y) = 3$. If either g_X or g_Y lies on the path between T_X and T_Y then $d_{T''}(T_X, T_Y) = 4$, else $d_{T''}(T_X, T_Y) = 3$. Hence $T'' \neq T'$. The same is true if $e' = f_Y$.

Suppose that e' is adjacent to e. If e' is incident to either T'_X or T'_Y then in the forest F resulting from the deletion of e', $d_F(T_X, T_Y) = 2$. Hence $d_{T''}(T_X, T_Y) \leq 3$, and $T'' \neq T'$. If e' is incident to T_X then $d_{T''}(T_Y, T'_Y) \geq 2$ and if e' is incident to T_Y then $d_{T''}(T_X, T'_X) \geq 2$. Hence $T'' \neq T'$.

Finally suppose that $e' = e$. If either g_X or g_Y is in one of the six labelled subtrees then that subtree is not a subtree of T'', so $T'' \neq T'$.

Suppose that $g_X = e_x$ or $g_Y = e_x$. Then $d_{T''}(T_X, T'_X) = 2$. Similarly if $g_X = e_y$ or $g_Y = e_y$ then $d_{T''}(T_Y, T'_Y) = 2$. Hence neither of these cases give $T'' = T'$.

Therefore $T'' = T'$ implies that $e' = e$ and $\{g_X, g_Y\} = \{f_X, f_Y\}$. Hence

$$|N_{TBR}(T)| = 2(n-3)(2n-7) + \sum_{k=5}^{M}(k-4)P_k(T)$$

where M is the maximum length of any such path (the diameter of T).

□

We now show that the result in Theorem 7.1 is equivalent to that of Humphries and Wu (2013).

Theorem 7.2. *Let $T \in UB(n)$ $(n \geq 4)$. Then*

$$4\sum |A||B| - (4n-2)(n-3) = 2(n-3)(2n-7) + \sum_{k=5}^{M}(k-4)P_k(T)$$

where the first sum is over all non-trivial splits $\{A, B\}$ of T and M is the diameter of T.

Proof. Let e be an internal edge of T, and let e_1 and e_2 be edges of T such that e lies on the $(e_1 - e_2)$-path in T and $d_T(e, e_1), d_T(e, e_2) \geq 1$. Then $d_T(e_1, e_2) \geq 3$. The number of possible choices of these edges is given by $\sum_{k=5}^{M}(k-4)P_k(T)$.

We now calculate the number of possible choices of the edges e, e_1 and e_2, by considering the splits of T. Let $S = \{A, B\}$ be the non-trivial split of T corresponding to edge e in T. Let T_X and T_Y be the subtrees incident to e, where $e_1 \in E(T_X)$ and $e_2 \in E(T_Y)$. We consider how many possible choices there are for the edges e_1 and e_2. Note that $|A| = |\mathcal{L}(T_X)|$ and $|B| = |\mathcal{L}(T_Y)|$. The number of edges in T_X is $2|A| - 2$, because there is one vertex (incident to e) of degree two. Since $d_T(e, e_1) \geq 1$ there are only $2|A| - 4$ possible choices of e_1. The same is true of e_2 and $|B|$. Hence

$$\sum_{k=5}^{M}(k-4)P_k(T)) = \sum_{\{A,B\} \in \Sigma(T)} (2|A| - 4)(2|B| - 4).$$

Therefore

$$
\begin{aligned}
(4 \sum_{\{A,B\}\in\Sigma(T)} (|A||B|) - (4n-2)(n-3)) - (2(n-3)(2n-7) + \sum_{k=5}^{M}(k-4)P_k(T)) \\
&= 4 \sum_{\{A,B\}\in\Sigma(T)} (|A||B|) - \sum_{k=5}^{M}((k-4)P_k(T)) - 8(n-3)(n-2) \\
&= 4 \sum_{\{A,B\}\in\Sigma(T)} (|A||B|) - \sum_{(A,B)\in\Sigma(T)} ((2|A|-4)(2|B|-4)) - 8(n-3)(n-2) \\
&= 8 \sum_{\{A,B\}\in\Sigma(T)} (|A|+|B|-2) - 8(n-3)(n-2) \\
&= 8(n-3)(n-2) - 8(n-3)(n-2) \\
&= 0.
\end{aligned}
$$

□

We now find explicit formulae for the size of the TBR neighbourhood of a caterpillar and both types of balanced tree (where $n = 2^i$ or $n = 3 \cdot 2^i$, $i \in \mathbb{Z}^+$). Our expression for the size of the neighbourhood of a caterpillar is the same as that obtained by Humphries and Wu (2013). They also found an asymptotic expression for the size of the TBR neighbourhood of a 'complete' tree, which is a more general structure than a balanced tree. Our expression for the size of the TBR neighbourhood of a balanced tree in Corollary 7.3 agrees with their result.

Corollary 7.3. *Let $T \in UB(n)$.*

(i) If T is a caterpillar ($n \geq 6$), then

$$|N_{TBR}(T)| = \frac{2}{3}n^3 - 4n^2 + \frac{16}{3}n + 2.$$

(ii) If T is a balanced tree with $n = 3 \cdot 2^i$ leaves ($i \geq 2$), then

$$|N_{TBR}(T)| = (4i - \frac{20}{3})n^2 + 22n - 6.$$

(iii) If T is a balanced tree with $n = 2^i$ leaves ($i \geq 3$), then

$$|N_{TBR}(T)| = (4i - 13)n^2 + 22n - 6.$$

Proof.

(i) For $k \geq 3$, a caterpillar T has $4(n-k)$ paths of length k by Lemma 6.9. Hence

$$\begin{aligned}
|N_{TBR}(T)| &= 2(n-3)(2n-7) + \sum_{k=5}^{n-1}(k-4)P_k(T) \\
&= 2(n-3)(2n-7) + \sum_{k=5}^{n-1} 4(k-4)(n-k) \\
&= 2(n-3)(2n-7) + \frac{2}{3}(n-3)(n-4)(n-5) \\
&= \frac{2}{3}n^3 - 4n^2 + \frac{16}{3}n + 2.
\end{aligned}$$

(ii) For a balanced tree T with $n = 3(2)^i$ leaves ($i \geq 2$), we sum over all of the even and odd paths in T. By Lemma 6.9 we obtain

$$\begin{aligned}
|N_{TBR}(T)| &= 2(n-3)(2n-7) + \sum_{k=5}^{n-1}(k-4)P_k(T) \\
&= 2(n-3)(2n-7) + \sum_{m=3}^{\log_2(\frac{n}{3})+1} 6(m-2)(2^{m-1})(n-2^m) \\
&+ \sum_{m=3}^{\log_2(\frac{n}{3})+1} (2m-5)2^m(n-3(2^{m-2})) \\
&= 4n^2\log_2(n) - \left(\frac{20}{3} + 4\log_2(3)\right)n^2 + 22n - 6 \\
&= \left(4i - \frac{20}{3}\right)n^2 + 22n - 6.
\end{aligned}$$

(iii) For a balanced tree T with $n = (2)^i$ leaves ($i \geq 3$), we again sum over all of the even and odd paths in T. By Lemma 6.9 we obtain

$$\begin{aligned}
|N_{TBR}(T)| &= 2(n-3)(2n-7) + \sum_{k=5}^{n-1}(k-4)P_k(T) \\
&= 2(n-3)(2n-7) + \sum_{m=3}^{\log_2(\frac{n}{4})+1} 6(m-2)(2^{m-1})(n-2^m) \\
&+ \sum_{m=3}^{\log_2(\frac{n}{4})+2} (2m-5)2^m(n-3(2^{m-2})) \\
&= 4n^2\log_2(n) - 13n^2 + 22n - 6 \\
&= (4i-13)n^2 + 22n - 6.
\end{aligned}$$

□

8 Concluding Comments

In this thesis, we derived new results for the sizes of the first and second RF neighbourhoods of an unrooted binary tree. We independently verified the expressions for the sizes of the first and second

NNI neighbourhoods, originally calculated by Robinson (1971), and extended Robinson (1971)'s result for the third NNI neighbourhood of an unrooted binary tree. In addition, we calculated new asymptotic results for the sizes of the k^{th} RF and NNI neighbourhoods of a binary phylogenetic tree. We also found an expression for the number of pairs of binary trees that share a first neighbour under the RF and NNI metrics.

In our results for the size of the k^{th} RF and NNI neighbourhoods of an unrooted binary tree T (Theorems 3.1 and 4.1), the term of order n^{k-1} contains a parameter dependent on T and k. We have calculated bounds on the value of this parameter; for RF, $-\frac{5k^2+7k}{4} \leq C_{T,k} \leq 4k^2 - 7k$, and for NNI, $\frac{-3k(k+1)}{2} \leq D_{T,k} \leq 3k(k-2)$. These bounds are not strict, so it would be interesting to investigate ways of improving them. A natural question is whether or not both positive and negative values of $C_{T,k}$ and $D_{T,k}$ are possible for any given value of k, and if so, can we find examples of such trees.

We independently verified the expression for the size of the first SPR neighbourhood, originally calculated by Allen and Steel (2001), and showed that in contrast to RF and NNI, the size of the second SPR neighbourhood is not solely dependent on the number of leaves and cherries of the tree. Humphries and Wu (2013) showed that for TBR even the first neighbourhood depends on variables other than the number of leaves and cherries. We calculated an expression for the size of the first TBR neighbourhood, that is equivalent to that of Humphries and Wu (2013).

In this thesis we have considered neighbourhoods of unrooted binary trees under the four metrics; RF, NNI, SPR and TBR. There are, however, many other metrics that can be used to compare trees, that would be interesting to investigate. For example, Moulton and Wu (2015) recently defined a new metric d_p, similar to the TBR metric. (The same metric was also independently defined by Kelk and Fischer (2014).) Using the result of Humphries and Wu (2013) they calculated the size of the first neighbourhood of an unrooted binary tree under this metric. Given the difficulty of calculating the size of the second SPR neighbourhood it is possible that similar problems would arise in calculating the size of the second neighbourhood under TBR or d_p. However, this would be interesting to investigate, and it may be possible to find the size of the second TBR or d_p neighbourhood of a particular type of tree, such as a caterpillar or a balanced tree.

References

Allen, B. L. and Steel, M. (2001). Subtree transfer operations and their induced metrics on evolutionary trees. *Annals of Combinatorics*, 5(1):1–13.

Bollobas, B. (1998). *Modern Graphy Theory*. Springer Science and Business Media.

Bourque, M. (1978). *Arbres de Steiner et reseaux dont varie l'emplagement de certains sommets*. PhD thesis, University of Montreal.

Bryant, D. (2004). The splits in the neighbourhood of a tree. *Annals of Combinatorics*, 8(1):1–11.

Bryant, D. (2008). Penny ante: A mathematical challenge.
Available at: http://www.math.canterbury.ac.nz/bio/events/kaikoura09/penny.shtml.

Bryant, D. and Steel, M. (2009). Computing the distribution of a tree metric. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 6(3):420–426.

Buneman, P. (1971). The recovery of trees from measures of dissimilarity. In Kendall, D. G. and Tautu, P., editors, *Mathematics in the Archaeological and Historical Sciences*, pages 387–395. Edinburgh University Press.

Caceres, A. J. J., Castillo, J., Lee, J., and St. John, K. (2013). Walks on SPR neighbourhoods. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 10(1):236–239.

Culik, K. and Wood, D. (1982). A note on some tree similarity measures. *Information Processing Letters*, 15(1):39–42.

DasGupta, B., He, X., Jiang, T., Li, M., Tromp, J., and Zhang, L. (1997a). On computing the nearest neighbour interchange distance. In *In: Proc. Dimacs Workshop on Discrete Problems with Medical Applications*, pages 125–143. Press.

DasGupta, B., He, X., Jiang, T., Li, M., Tromp, J., and Zhang, L. (1997b). On distances between phylogenetic trees. *Proceedings of the Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 427–436.

Day, W. H. E. (1985). Optimal algorithms for comparing trees with labeled leaves. *Journal of Classification*, 2(1):7–28.

Felsenstein, J. (2004). *Inferring Phylogenies*. Sinauer Associates Inc.

Gordon, K., Ford, E., and St. John, K. (2013). Hamiltonian walks of phylogenetic treespaces. *IEEE Transactions on Computational Biology and Bioinformatics*, 10(4):1076–1079.

Hendy, M. D. and Penny, D. (1982). Branch and bound algorithms to determine minimal evolutionary trees. *Mathematical Biosciences*, 59(2):277–290.

Humphries, P. J. and Wu, T. (2013). On the neighbourhoods of trees. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 10(3):721–728.

Kelk, S. and Fischer, M. (2014). On the complexity of computing MP distance between binary phylogenetic trees. Available at: http://arxiv.org/abs/1412.4076.

Kubatko, L. (2007). Inference of phylogenetic trees. In Friedman, A., editor, *Tutorials in Mathematical Biosciences IV: Evolution and Ecology*, pages 1–38. Springer-Verlag.

Li, M., Tromp, J., and Zhang, L. (1996). Some notes on the nearest neighbour interchange distance. In *Lecture Notes in Computer Science*, volume 1090, pages 343–351. Springer-Verlag.

Moulton, V. and Wu, T. (2015). A parsimony-based metric for phylogenetic trees. *Advances in Applied Mathematics*, 66:22–45.

Robinson, D. F. (1971). Comparison of labeled trees with valency three. *Journal of Combinatorial Theory, Series B*, 11(2):105–119.

Robinson, D. F. and Foulds, L. R. (1981). Comparison of phylogenetic trees. *Mathematical Biosciences*, 53(1-2):131–147.

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

Whelan, S. and Money, D. (2010). The prevalence of multifurcations in tree-space and their implications for tree-search. *Molecular Biology and Evolution*, 27(12):2674–2677.