2 Phylogenetic Networks II: Enumerative and Algorithmic Aspects

This set of lecture notes is in two parts. The first part considers various enumeration problems associated with phylogenetic networks. In the second part, we consider a particular algorithmic problem which, together with its variants, sparked the mathematical interest in phylogenetic networks.

2.1 Parameters of networks

A phylogenetic tree with ℓ leaves has exactly $2\ell - 3$ vertices and $2\ell - 2$ edges regardless of its shape. The next theorem describes an analogue of this for phylogenetic networks.

Theorem 2.1. Let $\mathcal{N}$ be a phylogenetic network on n vertices with ℓ leaves, r reticulations, and t tree vertices. Then $t = \ell + r - 2$ and $n = 2t + 3$. Furthermore, $\mathcal{N}$ has $3r + 2\ell - 2$ edges.

Proof.

For phylogenetic trees, the number of leaves bounds the total number of vertices.

Question. Does the number of leaves bound the total number of vertices of a phylogenetic network?

Theorem 2.1 says that the total number of vertices of a phylogenetic network is bounded either by the number of tree vertices or by the sum of the number of leaves and the number of reticulations.

For tree-child networks, we can do better than this.

Theorem 2.2. Let $\mathcal{N}$ be a tree-child network on n vertices with ℓ leaves and r reticulations. Then

$$r < \frac{n}{4} < \ell.$$

Theorem 2.2 says that the total number of vertices of a tree-child network is at most 4 times the number of leaves, so the number of leaves bounds the total number of vertices of a tree-child network.

Lemma 2.3. Let $\mathcal{N}$ be a tree-child network with ℓ leaves and r reticulations. Then $r \leq \ell - 1$.

Proof.

Proof of Theorem 2.2.

Question. In the proof of Theorem 2.2, we showed that $r \leq \ell - 1$. Is this sharp? In other words, for all $\ell \geq 1$, is there a tree-child network with ℓ leaves and $\ell - 1$ reticulations?

2.2 Counting phylogenetic trees

Possibly the oldest result in mathematical phylogenetics is an enumeration result. This result dates back to Schröder (1870):

Theorem 2.4. For all $\ell \geq 2$, the number t_ℓ of phylogenetic trees with leaf set $\{1, 2, \ldots, \ell\}$ is

$$t_\ell = 1 \times 3 \times 5 \times \cdots \times (2\ell - 3) = \frac{(2\ell-2)!}{(\ell-1)!2^{\ell-1}}.$$

Proof. Exercise.

Corollary 2.5. For all odd $n \geq 3$, the number t_n of (phylogenetic) trees on vertex set $\{1, 2, \ldots, n\}$ with ℓ leaves is

$$t_n = \binom{n}{\ell} \frac{(n-1)!}{2^{\ell-1}}.$$

It is important to note that t_n counts the number of (phylogenetic) trees whose vertex set is labelled $1, 2, \ldots, n$.

Proof of Corollary 2.5.

2.3 Counting phylogenetic networks

We have seen an exact count for the number t_ℓ of phylogenetic trees with leaf set $\{1, 2, \ldots, \ell\}$ as well as an exact count for the number t_n of (phylogenetic) trees with vertex set $\{1, 2, \ldots, n\}$. In contrast, the number of phylogenetic networks on n labelled vertices is unknown. In fact, the number of tree-child networks with leaf set $\{1, 2, \ldots, \ell\}$ is also unknown. To date, the best counts for these last two numbers are approximate counts.

For the purposes of making comparisons, lets first write each of t_ℓ and t_n in a much cruder form. By first approximating t_ℓ and t_n for large ℓ and n, respectively, using Stirling's approximation for factorials, we can write t_ℓ and t_n as

$$t_\ell = 2^{\ell \log \ell + O(\ell)} \tag{1}$$

and

$$t_n = 2^{n \log n + O(n)}. \tag{2}$$

Here logarithms are to the base 2. Furthermore, $O(\ell)$, for example, stands for a value that is no more than a constant times ℓ. This constant is independent of ℓ and so, for *large* ℓ, the contribution of $O(\ell)$ to the exponent will be *small* in comparison to $\ell \log \ell$.

In contrast to (2) and (1), respectively, we have the following recent theorem. For all odd $n \geq 3$, let gn_n denote the number of (general) phylogenetic networks with vertex set $\{1, 2, \ldots, n\}$ and, for all $\ell \geq 2$, let tc_ℓ denote the number of tree-child networks with leaf set $\{1, 2, \ldots, \ell\}$.

Theorem 2.6. For all odd $n \geq 3$,

$$gn_n = 2^{\frac{3}{2} n \log n + O(n)}$$

and, for all $\ell \geq 2$,

$$tc_\ell = 2^{2\ell \log \ell + O(\ell)}.$$

2.4 The Minimum Hybridisation problem

For the rest of the notes, we investigate the following problem. Let $\mathcal{P}$ be a collection of phylogenetic trees on the same set of species, where each tree in $\mathcal{P}$ *correctly* represents the tree-like evolution of different parts of the species genomes. What is the smallest number of reticulations to explain the evolution of the species? This smallest number provides an indication of the extent that reticulation has had on the evolutionary history of the species under consideration.

In its various interpretations, the problem dates back at least to Hein (1990). The interpretation we investigate here dates back to 2005. In general, the problem is NP-hard even when the initial collection consists of just two phylogenetic trees. However, there is an attractive and algorithmically crucial characterisation of the problem in this simplest case. We explore this characterisation and its algorithmic consequences.

For a phylogenetic network $\mathcal{N}$, the *reticulation number* of $\mathcal{N}$, denoted $h(\mathcal{N})$, is the number of reticulations in $\mathcal{N}$. Let $\mathcal{T}$ and $\mathcal{T}'$ be two phylogenetic trees. The *reticulation number* of $\mathcal{T}$ and $\mathcal{T}'$, denoted $h(\mathcal{T}, \mathcal{T}')$, is

$$\min\{h(\mathcal{N}) : \mathcal{N} \text{ is a phylogenetic network on } X \text{ that displays } \mathcal{T} \text{ and } \mathcal{T}'\}.$$

Note. In the literature, the reticulations are typically referred to as 'hybridisations', hence the notation. For consistency, we will stick with 'phylogenetic networks' and 'reticulations' rather than 'hybridisation networks' and 'hybridisations', respectively.

Mathematically, the problem we are interested in is the following.

MINIMUM RETICULATION
Instance: Two rooted phylogenetic X-trees $\mathcal{T}$ and $\mathcal{T}'$.
Question: Find $h(\mathcal{T}, \mathcal{T}')$?

Example 2.1.

2.5 Characterising Minimum Reticulation

Before formally stating the characterisation, lets think about what we are trying to do. Let $\mathcal{T}$ and $\mathcal{T}'$ be two phylogenetic X-trees, and let $\mathcal{N}$ be a phylogenetic network on X that displays $\mathcal{T}$ and $\mathcal{T}'$. If, for each reticulation in $\mathcal{N}$, we delete its two incoming reticulation edges (and repeatedly contract any resulting degree-two vertex and delete any resulting non-leaf degree-one vertex), we obtain a collection of phylogenetic trees whose leaf sets are subsets of X. Importantly, what do you notice about about the trees in this collection? These trees are (disjoint) 'subtrees' of both $\mathcal{T}$ and $\mathcal{T}'$. Thus the reticulation edges in $\mathcal{N}$ correspond to different paths of genetic inheritance (reticulation events). The fewer such edges deleted, the smaller the number of such events. It is this intuition that underlies the definition of an 'agreement forest', the concept that is central to the characterisation. We next define agreement forests.

Let $\mathcal{T}$ be a phylogenetic X-tree and let X' be a subset of X. The minimal rooted subtree of $\mathcal{T}$ that connects the leaves in X' is denoted by $\mathcal{T}(X')$. Additionally, the *restriction* of $\mathcal{T}$ to X', denoted $\mathcal{T}|X'$, is the phylogenetic X'-tree obtained from $\mathcal{T}(X')$ by contracting all non-root vertices of degree two.

Let $\mathcal{T}$ and $\mathcal{T}'$ be two phylogenetic X-trees. For technical reasons, view the roots of $\mathcal{T}$ and $\mathcal{T}'$ as a vertex ρ adjoined via a new edge to the original roots. An *agreement forest* $\mathcal{F} = \{\mathcal{L}_\rho, \mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_k\}$ for $\mathcal{T}$ and $\mathcal{T}'$ is a partition of $X \cup \{\rho\}$ with $\rho \in \mathcal{L}_\rho$ satisfying the following properties:

(i) for all $i \in \{\rho, 1, 2, \ldots, k\}$, we have $\mathcal{T}|\mathcal{L}_i \cong \mathcal{T}'|\mathcal{L}_i$; and

(ii) the tres in $\{\mathcal{T}(\mathcal{L}_i) : i \in \{\rho, 1, 2, \ldots, k\}\}$ and $\{\mathcal{T}'(\mathcal{L}_i) : i \in \{\rho, 1, 2, \ldots, k\}\}$ are vertex disjoint subtrees of $\mathcal{T}$ and $\mathcal{T}'$, respectively.

Example 2.2.

Given what we have seen so far, it seems plausible that if we have an agreement forest for $\mathcal{T}$ and $\mathcal{T}'$ of size $k+1$, then we can construct a phylogenetic network with k reticulations that displays $\mathcal{T}$ and $\mathcal{T}'$. The problem with this is that, no matter how we go about this construction, the resulting phylogenetic network is not necessarily acyclic!

To avoid this problem, we extend the definition of an agreement forest to an acyclic-agreement forest. Let $\mathcal{F} = \{\mathcal{L}_\rho, \mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_k\}$ be an agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. Let $G_\mathcal{F}$ be the directed graph whose vertex set is $\mathcal{F}$ and $(\mathcal{L}_i, \mathcal{L}_j)$ is a directed edge precisely if $i \neq j$ and either

(i) the root of $\mathcal{T}(\mathcal{L}_i)$ in $\mathcal{T}$ is an ancestor of the root of $\mathcal{T}(\mathcal{L}_j)$ in $\mathcal{T}$ or

(ii) the root of $\mathcal{T}'(\mathcal{L}_i)$ in $\mathcal{T}'$ is an ancestor of the root of $\mathcal{T}'(\mathcal{L}_j)$ in $\mathcal{T}'$.

We say that $\mathcal{F}$ is an *acyclic-agreement forest* for $\mathcal{T}$ and $\mathcal{T}'$ if $G_{\mathcal{F}}$ is has no directed cycles, that is, $G_{\mathcal{F}}$ is acyclic.

Note. As $\mathcal{F}$ is an agreement forest, the roots of $\mathcal{T}(\mathcal{L}_i)$ and $\mathcal{T}(\mathcal{L}_j)$, and the roots of $\mathcal{T}'(\mathcal{L}_i)$ and $\mathcal{T}'(\mathcal{L}_j)$ are not the same. (Why?)

If $\mathcal{F}$ is an acyclic-agreement forest and it has the smallest k over all acyclic-agreement forests for $\mathcal{T}$ and $\mathcal{T}'$, we say $\mathcal{F}$ is a *maximum-acyclic-agreement forest* for $\mathcal{T}$ and $\mathcal{T}'$, in which case, we denote the number k by $m_a(\mathcal{T}, \mathcal{T}')$.

Example 2.3.

At last we state the characterisation.

Theorem 2.7. Let $\mathcal{T}$ and $\mathcal{T}'$ be two phylogenetic X-trees. Then $h(\mathcal{T}, \mathcal{T}') = m_a(\mathcal{T}, \mathcal{T}')$.

Note. Theorem 2.7 'reduces' the Minimum Reticulation problem to minimising the number of edges to delete from $\mathcal{T}$ and $\mathcal{T}'$ until there is 'agreement' amongst the resulting subtrees.

Question. Let $\mathcal{T}$ and $\mathcal{T}'$ be two phylogenetic X-trees, and let $\mathcal{F} = \{\mathcal{L}_\rho, \mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_k\}$ be a maximum-acyclic-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$. Show that $|\mathcal{L}_\rho| \geq 2$, that is, in addition to ρ, there is at least one other element in $\mathcal{L}_\rho$.

Question. If $\mathcal{F}$ is an acyclic-agreement forest for $\mathcal{T}$ and $\mathcal{T}'$ of size $k+1$, then there is a polynomial-time algorithm for constructing a reticulation network $\mathcal{N}$ that displays $\mathcal{T}$ and $\mathcal{T}'$ with $h(\mathcal{N}) \leq k$. Can you describe such an algorithm?

2.6 Algorithmic implications

In the literature, it now seems like there are a zillion different approaches to nullifying the NP-hardness of Minimum Reticulation in determining $h(\mathcal{T}, \mathcal{T}')$. Nevertheless, every one of these approaches relies on Theorem 2.7. The next result and its proof highlight how Theorem 2.7 is used.

For a phylogenetic X-tree $\mathcal{T}$, a *cluster* of $\mathcal{T}$ is a subset A of X such that there is a vertex in $\mathcal{T}$ whose descendants in X are precisely the elements in A.

Theorem 2.8. (Cluster-Reduction Theorem) Let $\mathcal{T}$ and $\mathcal{T}'$ be two phylogenetic X-trees, and let A be a cluster of both $\mathcal{T}$ and $\mathcal{T}'$. Then

$$h(\mathcal{T}, \mathcal{T}') = h(\mathcal{T}|A, \mathcal{T}'|A) + h(\mathcal{T}_a, \mathcal{T}'_a),$$

where $\mathcal{T}_a$ and $\mathcal{T}'_a$ are obtained from $\mathcal{T}$ and $\mathcal{T}'$, respectively, by replacing $\mathcal{T}(A)$ and $\mathcal{T}'(A)$ with a single new leaf a.

Note. As A is a cluster of $\mathcal{T}$, the minimal subtree $\mathcal{T}(A)$ is a 'pendant' subtree of $\mathcal{T}$, that is, it can be obtained from $\mathcal{T}$ by simply deleting the edge directed into the vertex corresponding to A. Similarly, $\mathcal{T}'(A)$ is a pendant subtree of $\mathcal{T}'$.

Proof of the Cluster-Reduction Theorem.

In addition to the cluster reduction, there is the more complicated 'chain reduction'. Together they show that Minimum Reticulation is a fixed-parameter tractable. A theoretical notion, fixed-parameter tractability in practice means that if the size of X is large but $h(\mathcal{T}, \mathcal{T}')$ is small, then a solution to Minimum Reticulation can be found reasonably quickly.