ORIENTABILITY OF PHYLOGENETIC NETWORK GRAPHS

ETHAN CECCHETTI

ABSTRACT. Traditionally, the genetic history of species has been modeled using phylogenetic trees. Recently, scientists have begun using phyolgenetic networks to model more complex occurrences, such as hybridization, which cannot be displayed by trees. Phylogenetic networks are represented by network graphs which are trivalent directed graphs without directed circuits. In this paper we discuss the mathematics of network graphs. Given an unoriented trivalent graph, we determine a necessary and sufficient condition for orienting the graph as a network graph.

1. INTRODUCTION

Evolution has traditionally been modeled using trees with each vertex representing a different species. An edge connecting two vertices indicates that one species is directly descended from the other. One species splitting into two or more species is represented by multiple edges emanating from the vertex. Leaves (or terminal vertices) usually represent either currently living species or extinct ones, but there can also be one leaf that represents the root, or common ancestor to all of the other species. These trees are known as *phylogenetic trees*.

Phylogenetic trees, however, are constrained by a major restriction: they can only account for speciation, or the splitting of one species into two or more species. They cannot account for hybridization, where two species come together to form a single new species. Recently however, scientists have begun to use more complicated graphs known as "network graphs" to account for these occurrences. *Phylogenetic networks* display the same data as phylogenetic trees, but are represented by network graphs instead of trees. In a network graph, circuits are allowed, edges are given directions, and every internal vertex is trivalent (has three edges connected to it). Speciation is represented by a vertex with one edge oriented inward and the other two out, whereas hybridization is represented by a vertex with two edges in and one out [6]. A circuit occurs when a species splits into two species and at some later time, two species descended from the original hybridize. Such a circuit will always have both edges directed outward at the speciation node and both edges directed inward at the hybridization node. The edges in a circuit will never be oriented to form a continuously directed circle as this would make no sense biologically.

Recently, much work has been done concerning phylogenetic networks and several variations on the definition appear in the literature. Most of this work is related to developing computer algorithms to construct networks from DNA data [1, 2, 3, 5]. These graphs are
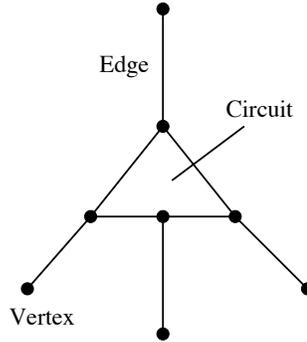
FIGURE 1. Graph

often highly restricted network graphs, such as galled-network graphs in which no single edge can belong to more than one circuit [4, 7].

In this paper we discuss the mathematics of general network graphs. We look at when undirected trivalent graphs can be given orientations to become valid network graphs. We show that there exist certain trivalent graphs that cannot be oriented in this manner. We call these "illegal" graphs and we prove that a trivalent graph can be oriented as a network graph if and only if it has at least two leaves and contains no illegal subgraph.

In Section 2 we define terms necessary for use in this paper. Section 3 discusses an obstruction to orienting a trivalent graph as a network graph. In Section 4 we prove the main theorem.

2. TERMINOLOGY

This section contains definitions and short explanations of the mathematical concepts used in this paper. The first, and underlying definition, is that of a graph. A *graph* is a finite set of points, or *vertices* and pairs of vertices known as *edges*. We often represent edges as lines connecting the two vertices. Edges are called *adjacent* if they share a common vertex. A *circuit* is a sequence of adjacent edges returning to the original vertex without traversing the same edge twice. The length of a circuit is defined as the number of edges forming that circuit. In our graphs, we do not allow circuits of length one where a single edge is connected to the same vertex at both ends. These circuits are known as *loops*. Graphs, however, can have longer circuits including *digons*, or circuits of length two. A graph without any circuits is known as a *tree*.

In a graph, there are two types of vertices: leaves and nodes. The difference between a leaf and a node is the valence of the vertex. The *valence* of a vertex is the number of edges connected to it. A *leaf*, or external vertex, is a vertex with valence one, and a *leaf edge* is any edge connected to a leaf. A *node*, or internal vertex, is a vertex with valence more than one. That is, all nodes have at least two edges connected to them. A connected
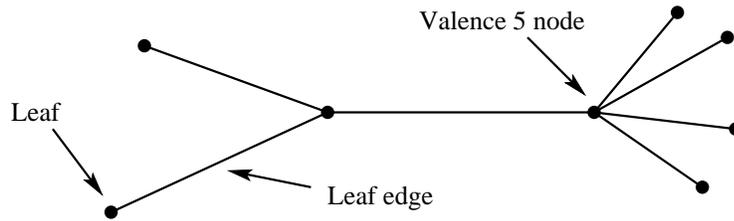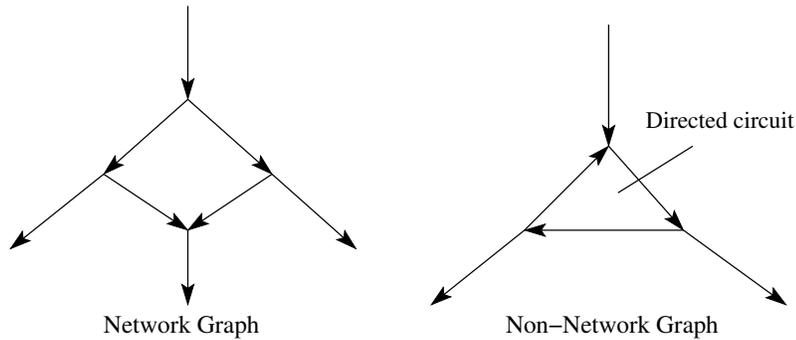
FIGURE 2. Tree

FIGURE 3. Directed Graphs

graph in which all nodes are valence three will be referred to in this paper as a *trivalent graph*. Note that trivalent graphs can also have leaves.

Graphs can also be directed. A *directed graph* is a graph where each edge has an orientation or direction associated with it. With directed graphs, there is a possibility of a *directed circuit*, or a circuit where it is possible to reach the same vertex on the circuit twice simply by following the orientation of the circuit's edges. There is also a special type of directed graph known as a network graph. In a *network graph*, the following conditions must be satisfied:

(1) The graph must be trivalent.
(2) There cannot be any directed circuits.
(3) Every node must have either one edge directed inward and two out, or two edges in and one out. There can never be three edges directed inward or three outward at a single node.

Note that loops are not permitted because any directed loop would create a directed circuit.

3. Non-orientable graphs

In the section we will discuss the answer to the following question: Given an undirected trivalent graph, can the graph always be oriented as a network graph?

Theorem 3.1. *A trivalent graph Γ with no leaves can never be oriented as a network graph.*

Proof. This will be a proof by contradiction, so we assume that there exists an orientation on the edges of Γ such that Γ is a network graph. Because there are no nodes on Γ with three edges directed inward, we can always move away from a node while following the direction of an edge. Therefore, on Γ, starting at any node, we can continue moving infinitely along the edges. Because there are only a finite number of edges on Γ, we must eventually revisit a node. If we can revisit a node following directed edges, there must be a directed circuit. This contradicts the assumption that a valid orientation of the graph Γ exists. $\qquad\square$

We will need the following lemma to prove that there are restrictions on orienting trivalent graphs with leaves.

Lemma 3.2. *For any network graph, if the orientation of every edge is reversed, the graph will still meet the conditions of a network graph.*

Proof. If there are no directed circuits in a graph, reversing the orientation of every edge cannot create any directed circuits. If there are no nodes with three edges pointing inward, there will be no nodes with three edges pointing outward if all of the edges are reversed, and vice versa. $\qquad\square$

We define an *inward pointing* leaf edge to be a leaf edge directed away from the valence one vertex and an *outward pointing* leaf edge is directed toward the valence one vertex.

Theorem 3.3. *A network graph cannot have all leaf edges pointing inward or all leaf edges pointing outward.*

Proof. If all leaf edges are pointing inward, then, when starting at any node, it is impossible to get to a leaf by following the direction of the edges. Therefore, if we begin at any vertex and follow the orientation of the edges, we can always continue without getting stuck at a leaf. By the same argument as Theorem 3.1, there must be a directed circuit and therefore the graph cannot have a valid orientation. Also, by Lemma 3.2, we cannot create a network graph with all leaf edges directed outward. This is because, if one existed, we could simply reverse the orientation of every edge and create a valid orientation with every leaf edge directed inward. $\qquad\square$

This theorem is of particular importance when a graph has only one leaf.

Corollary 3.4. *Any trivalent graph with only one leaf cannot be oriented as a network graph.*

For example, there is no way to orient the graph in Figure 4 as a network graph.

Theorem 3.5. *There exist trivalent graphs with any number of leaves that cannot be oriented as network graphs.*
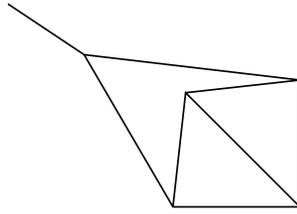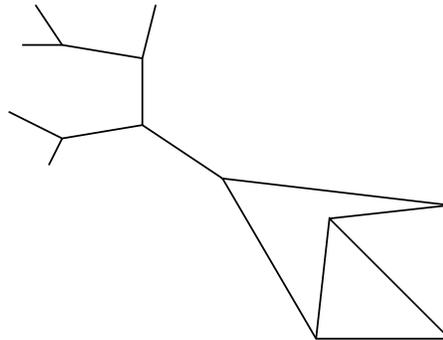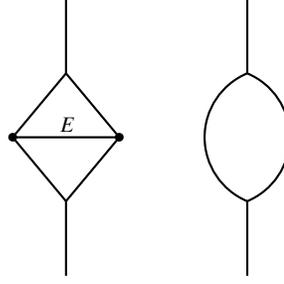
FIGURE 4

FIGURE 5

Proof. To construct a graph with n leaves that cannot be oriented, we begin by constructing a trivalent graph Γ with n leaves. We can then construct a trivalent graph with only one leaf Γ' and attach the leaf of Γ' to Γ in the middle of any edge. The result is a trivalent graph with n leaves. If we could orient the combined graph as a network graph, it would give a valid orientation to the single-leaf subgraph Γ'. This is impossible due to Corollary 3.4. □

For example, in Figure 5 we see a graph with five leaves that cannot be oriented as a network graph.

4. Orientable graphs

In this section we give necessary and sufficient conditions for orienting a trivalent graph as a network graph. We remind the reader that all graphs are assumed to be connected and do not contain single-edge loops. Given a trivalent graph Γ and any non-leaf edge E in Γ, we can form a new graph Γ' by removing E and both its vertices V and W. We combine the other two edges meeting at V into a single edge and do the same for W. An example is shown in Figure 6.

FIGURE 6. Γ and Γ'

Lemma 4.1. *Suppose Γ' is formed from Γ by removing any non-leaf edge E from Γ as described above. If Γ' can be oriented as a network graph, then Γ can also be oriented as a network graph.*

Proof. We begin by orienting Γ' as a network graph. Now we can add the removed vertices, V and W, back in, orienting the split edges in the same direction they were oriented before. By doing this, we insure the neither V nor W will have three edges in or three edges out.

Now we must orient E either from V to W or from W to V. If orienting E from V to W would create a directed circuit, then there must be a directed path from W to V in Γ', and if orienting E from W to V would create a directed circuit, then there must be a directed path from V to W in Γ'. (Although V and W are not vertices in Γ', we can view them as points on an edge.) If there is a directed path both from V to W and from W to V, then Γ' already has a directed circuit. Because Γ' was oriented as a network graph, this cannot be the case. Therefore there is an orientation for E such that Γ is a network graph. $\square$

Suppose Γ_1 is a trivalent graph with exactly two leaves and Γ_2 is any trivalent graph. The operation $\Gamma_1 \# \Gamma_2$ denotes a combination of the two graphs. To combine the graphs, we must add two vertices to Γ_2 either by adding two vertices to one edge, or one vertex to each of two edges. We then fuse the leaves of Γ_1 with these new vertices, making them trivalent nodes and creating a new graph. We will denote the resulting graph (regardless of choices) as $\Gamma_1 \# \Gamma_2$. For example, see Figure 7.

Lemma 4.2. *Suppose Γ_1 and Γ_2 are trivalent graphs and Γ_1 has exactly two leaves. If Γ_1 and Γ_2 can both be oriented as network graphs, then $\Gamma_1 \# \Gamma_2$ can also be oriented as a network graph.*

Proof. We begin by labeling the two vertices where we joined Γ_1 and Γ_2 as V and W. Choose an orientation on Γ_2. We now add V and W to Γ_2 leaving the orientation of the split edge(s) unchanged. This insures that neither of these vertices will have all edges directed inward or all edges directed outward. We now orient Γ_1 as a network graph, and attach it to Γ_2 as described above. If Γ_2 has a directed path from V to W, we must insure that Γ_1 does not have a directed path from W to V. If it does, Γ_1 cannot have a directed
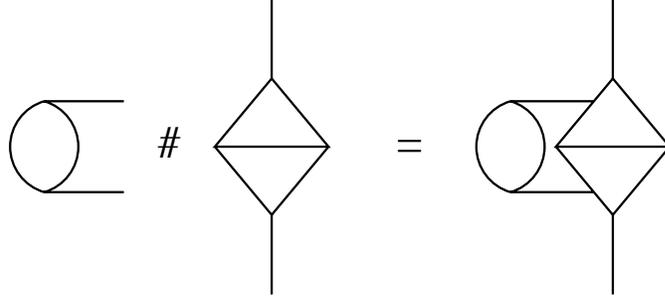
FIGURE 7. $\Gamma_1 \# \Gamma_2$

path from V to W since this would create a directed circuit. So by Lemma 3.2, we can reverse the orientation of every edge in Γ_1 and insure there is no longer a directed path from W to V. The result is a valid orientation on $\Gamma_1 \# \Gamma_2$.   □

For the purpose of the next theorem, we will define an *illegal subgraph* of Γ as a subgraph which is trivalent subgraph with one leaf and is connected to the rest of Γ only at the leaf. A *separating edge* of Γ is an edge which, if removed, will disconnect the graph. Note that the leaf edge of an illegal subgraph is a separating edge. For example, Figure 5 contains an illegal subgraph. It is clear from Corollary 3.4 that any graph containing an illegal subgraph cannot be oriented as a network graph.

Theorem 4.3. *Any trivalent graph with at least two leaves and no illegal subgraphs can be oriented as a network graph.*

Proof. We will prove this by induction on the number of edges. A graph with only one edge (a simple straight line) has two leaves and no nodes. We can clearly orient this graph as a network graph. Now let Γ be a trivalent graph with k edges for $k \geq 2$ and assume by induction that the theorem is true for all graphs with less than k edges. There are three cases to consider.

Case 1: Suppose Γ is a tree. We can begin at one leaf V, orienting its leaf edge toward the node. Now we can orient every other edge such that it points from the vertex closer to V toward the vertex farther from V. Because there are no circuits, we cannot create a directed circuit, and because there is a unique directed path from V to any vertex, at any node, exactly one edge points inward, and the other two point outward.

Case 2: Suppose Γ contains at least one circuit of length two (a digon) or three (a triangle). Let Γ' be the graph obtained from removing an edge of this circuit and its two vertices as described in Lemma 4.1. Because Γ is trivalent and has leaves, it is easy to see that removing an edge from a triangle or digon cannot create a loop. We see in Figure 8 the result of removing an edge in this manner. If any of the resulting edges are separating edges, then one of the corresponding edges in the original graph was also a separating

FIGURE 8

edge. Therefore, if Γ is a legal graph, so is Γ'. Since Γ' is smaller than Γ, by induction, it can always be oriented as a network graph. By Lemma 4.1, so can Γ.

Case 3: Suppose Γ is not a tree and contains no digons or triangles. Γ must therefore contain a circuit with at least four edges. We can remove an edge E_1 and its vertices from the circuit in Γ to create a smaller graph. Because Γ contains no digons, this operation cannot create a loop. By induction, if the resulting graph contains no illegal subgraphs, it can be oriented as a network graph and, by Lemma 4.1, so can Γ.

If the above operation creates an illegal subgraph, with separating edge E_2, then E_1 and E_2 must form a pair of edges which, if both removed, would separate the graph. Also, one of the resulting smaller graphs, Γ_{ill}, has no leaves, while the other, Γ_2, has all the leaves from Γ. We can attach E_1 and E_2 to Γ_{ill} in their original positions to create a new graph, Γ_1 with two leaves. It is now clear that Γ can be expressed as $\Gamma_1 \# \Gamma_2$.

Because Γ has no digons or triangles, removing the two vertices to form Γ_2 cannot create any loops. Neither Γ_1 nor Γ_2 can contain any illegal subgraphs because, if they did, so would Γ. Therefore, Γ_1 and Γ_2 must both be legal so, by induction, they can both be oriented as network graphs. By Lemma 4.2, Γ can also be oriented as a network graph. $\square$

By combining the results of Section 3 and Theorem 4.3, we obtain the following theorem.

Theorem 4.4. *A trivalent graph can be oriented as a network graph if and only if it has at least two leaves and contains no illegal subgraph.*

REFERENCES

[1] D. GUSFIELD, S. EDDHU, C. LANGLEY, *Optimal, efficient reconstruction of phylogenetic networks with constrained recombination.* J. Bioinformatics Comput. Biol. 2 (1) (2004) 173–213.
[2] L. NAKHLEH, J. SUN, T. WARNOW, C. R. LINDER, B. M. E. MORET, A. THOLSE, *Towards the development of computational tools for evaluating phylogenetic network reconstruction methods*, in: Proc. 8th Pacific Symp. Biocomputing, 2003.

[3] L. Nakhleh, T. Warnow, C. R. Linder, K. S. John, *Reconstructing reticulate evolution in species: Theory and practice*, J. Comput. Biol. 12 (6) (2005) 796–811.

[4] Y. S. Song, *A concise necessary and sufficient condition for the existence of a galled-tree.* IEEE/AMC Trans. Comput. Biol. and Bioinformatics 3 (2) (2006) 186–191.

[5] Y. S. Song, J. Hein, *Constructing minimal ancestral recombination graphs*, J. Comput. Biol. 12 (2) (2005) 147–169.

[6] L. Wang, K. Zhang, L. Zhang, *Perfect phylogenetic networks and recombination.* J. Comput. Biol. 8 (1) (2001) 69–78.

[7] S. Willson, *Unique restrictions of the tree-like phylogenetic networks from distances between leaves.* Bulletin of Math. Biol. 64 (4) (2006) 919–944.