II

Theory

In this second part of the book we develop the theoretical foundations for phylogenetic networks. First we study splits and unrooted phylogenetic networks (Chapter 5). The mathematical and computational aspects of such networks have been worked out in detail over the last 20 years. The existing algorithms are routinely used in numerous publications in systematic biology. Then we turn to clusters and rooted phylogenetic networks (Chapter 6). Their theory is still under development and widely used methods for the computation of rooted phylogenetic networks have yet to emerge.

Splits and unrooted phylogenetic networks

The concept of a *split* plays an important role in the mathematics of phylogeny. It is motivated by the simple, but crucial, observation that every edge e in an unrooted phylogenetic tree T defines a bipartition of the underlying taxon set $\mathcal{X}$ into two non-empty and disjoint subsets, A and B, known as a *split*. The splits of an unrooted phylogenetic tree uniquely define the topology of the tree and splits are used, for example, to compare different trees or to compute consensus trees. Any set of splits that is *compatible* corresponds to a phylogenetic tree and so one possible way to generalize from trees to networks is to consider sets of splits that are *incompatible*.

Splits provide the basis of unrooted phylogenetic trees and a large class of unrooted phylogenetic networks, namely *split networks*, just as clusters provide the basic building blocks for rooted phylogenetic trees and networks (see Chapter 6). The foundation for the theory of split networks was laid down in a seminal paper by Bandelt and Dress [9].

5.1 Overview

Figure 5.1 shows the relationships between some of the main concepts introduced in this chapter. The focus is on *splits*, and on *split networks* as an important type of *unrooted phylogenetic networks*. We compare splits to clusters, since the two concepts are closely related to each other. We describe the *Buneman graph* construction that provides a canonical split network for an arbitrary set of splits. We shall see that splits can be *compatible*, *circular* or *weakly compatible*. A nice feature of circular splits is that they can be represented by split networks that are *outer-labeled planar*. Weakly compatible splits are of interest because they can be efficiently calculated using the *split decomposition* algorithm. We also give a brief introduction to *T-theory*, a related mathematical theory.

Tree inference approaches (described in Chapter 3) can be thought of as methods for computing compatible sets of splits. The *split decomposition method* provides

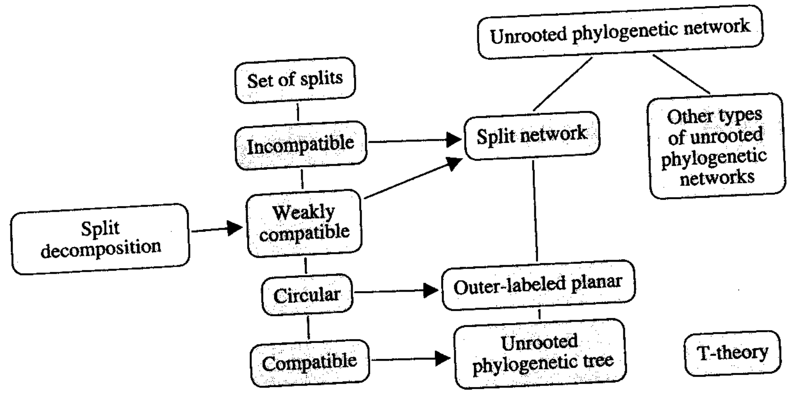


Figure 5.1 Overview of the main concepts introduced in this chapter. On the left, we list the different properties that a set of splits can have, in order of decreasing generality, and in the middle, we list the corresponding types of split networks.

one way of computing a set of incompatible splits on $\mathcal{X}$. Other methods for computing a set of splits that is not necessarily compatible are presented in Part III of the book.

5.2 Splits

Splits and clusters are closely related concepts. While clusters *group* taxa to emphasize their common features, splits *divide* taxa to emphasize their distinctive features. Here is the formal definition of a split:

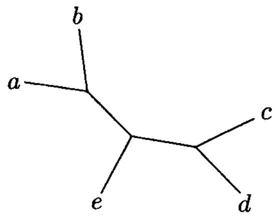
Definition 5.2.1 (Split) *A split $S = A \mid B$ is a bipartition of a set of taxa $\mathcal{X}$ into two non-empty subsets A, B with $A \cap B = \emptyset$ and $A \cup B = \mathcal{X}$. We also use the notation $\frac{A}{B}$ and $\frac{B}{A}$ equivalently, to improve readability. A* weighted split *is a split S that has been assigned a* weight $\omega(S) \geq 0$.

Note that there is no ordering in a split and so $A \mid B$ equals $B \mid A$. Consider a split $S = A \mid B$ on $\mathcal{X}$. We call A and B the two *split parts* of S. The *size* of a split $S = A \mid B$ is defined as the minimal cardinality of the two parts A and B. More formally, $\mathrm{size}(S) = \min\{|A|, |B|\}$. A split of size one is called a *trivial split*. For any taxon $x \in \mathcal{X}$, we use $S(x)$ to denote the split part that contains x and we use $\bar{S}(x)$ to denote the other part.

Let $\mathcal{S}$ be a set of splits on $\mathcal{X}$ and let $\mathcal{X}' \subset \mathcal{X}$ be a subset of taxa. We define the set of splits *induced* by $\mathcal{X}'$, or the *restriction* of $\mathcal{S}$ to the subset $\mathcal{X}'$, as

$$\mathcal{S}|_{\mathcal{X}'} = \left\{ \frac{A \cap \mathcal{X}'}{B \cap \mathcal{X}'} : \frac{A}{B} \in \mathcal{S},\ A \cap \mathcal{X}' \neq \emptyset \text{ and } B \cap \mathcal{X}' \neq \emptyset \right\}. \tag{5.1}$$

5.2 Splits



$\{a\}|\{b,c,d,e\}$
$\{b\}|\{a,c,d,e\}$
$\{c\}|\{a,b,d,e\}$
$\{d\}|\{a,b,c,e\}$
$\{e\}|\{a,b,c,d\}$
$\{a,b\}|\{c,d,e\}$
$\{a,b,e\}|\{c,d\}$

(a) Unrootedtree T (b) Split encoding of T

e 5.2 **(a) An unrooted phylogenetic tree T on $\mathcal{X} = \{a, \ldots, e\}$. (b) The seven splits represented by T.**

As already mentioned, any edge e of a phylogenetic tree T defines a split of the underlying taxon set $\mathcal{X}$ as follows: Deletion of e produces precisely two subtrees, T' and T'', say; the split defined by e is then the split $A \mid B$, where A and B are the taxon labels occurring in T' and T'' respectively. We use $\sigma_T(e) = A \mid B$ to denote the *split represented by e*. Because every leaf in a phylogenetic tree T is labeled by some taxon, it follows that both A and B are non-empty. Further, as each taxon occurs precisely once in T, it follows that $A \cap B = \emptyset$ and $A \cup B = \mathcal{X}$. If the edges of the tree have lengths or weights, then these can be assigned to the corresponding splits, as well.

If T does not contain any unlabeled nodes of degree two, then any two different edges e and f always represent two different splits, that is, $\sigma_T(e) \neq \sigma_T(f)$ must hold. The only situation in which a phylogenetic tree can contain an unlabeled node of degree 2 is when it has a root with outdegree two. In this case, the two edges e and f that originate at the root ρ give rise to the same split $\sigma_T(e) = \sigma_T(f)$, but to two complementary clusters.

A common construction to avoid this special case at the root ρ is to attach an additional leaf to ρ that is labeled by a special taxon o, which we call a (formal) *outgroup*. Then the root of a phylogenetic tree is specified as the node to which the leaf edge of o attaches. All phylogenetic trees that are discussed in this chapter are unrooted. However, by using this *outgroup trick* much of what we discuss concerning unrooted trees can also be adapted to rooted phylogenetic trees.

Let T be an unrooted phylogenetic tree on $\mathcal{X}$. We define the *split encoding* $\mathcal{S}(T)$ to be the set of all splits represented by T, that is,

$$\mathcal{S}(T) = \{\sigma(e) \mid e \text{ is an edge in } T\}. \tag{5.2}$$

The term *encoding* is justified by the observation that the tree T can be uniquely reconstructed from $\mathcal{S}(T)$, as we see in the next section.

Figure 5.2 shows an unrooted phylogenetic tree that has seven edges and thus gives rise to seven different splits. We can determine all splits associated with any given unrooted phylogenetic tree T in quadratic time using the following algorithm:

Algorithm 5.2.2 (Splits from tree) *The set $\mathcal{S}(T)$ of all splits associated with an unrooted phylogenetic tree T on $\mathcal{X}$ can be computed as follows:*

(i) *Choose a start leaf ρ and assume that all edges of T are directed away from ρ.*
(ii) *In a postorder traversal of T, for each node v compute the set $L(v)$ of taxon labels that are encountered in the subtree rooted at v.*
(iii) *For each edge $e = (u, v)$ of T, add the split $\sigma(e) = \frac{L(v)}{\mathcal{X}-L(v)}$ to $\mathcal{S}(T)$.*

Exercise 5.2.3 (Splits on a tree) *Consider the following set of splits*

$$\mathcal{S} = \left\{ \frac{\{a\}}{\{b,c,d,e\}}, \frac{\{b\}}{\{a,c,d,e\}}, \frac{\{c\}}{\{a,b,d,e\}}, \frac{\{d\}}{\{a,b,c,e\}}, \frac{\{e\}}{\{a,b,c,d\}}, \frac{\{a,b\}}{\{c,d,e\}}, \frac{\{a,e\}}{\{b,c,d\}} \right\}. \quad (5.3)$$

Does there exist an unrooted phylogenetic tree T on $\mathcal{X} = \{a, \ldots, e\}$ whose split encoding equals $\mathcal{S}$?

5.3 Compatibility and incompatibility

Suppose we are given an arbitrary set of splits $\mathcal{S}$ on $\mathcal{X}$. We would like to know whether $\mathcal{S}$ can be represented by some unrooted phylogenetic tree T, that is, does there exist some tree T with $\mathcal{S} = \mathcal{S}(T)$? The answer is given by the concept of compatibility [37].

Definition 5.3.1 (Compatibility of splits) *Two splits $S_1 = A_1 \mid B_1$ and $S_2 = A_2 \mid B_2$ on $\mathcal{X}$ are called* compatible, *if one of the following four possible intersections of their split parts is empty:*

$$A_1 \cap A_2,\ A_1 \cap B_2,\ B_1 \cap A_2 \text{ or } B_1 \cap B_2. \quad (5.4)$$

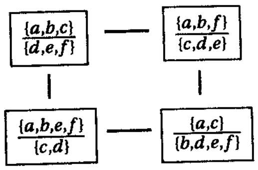
Otherwise, the two splits are called incompatible. *A set of splits $\mathcal{S}$ is called* compatible *if all pairs of splits in $\mathcal{S}$ are compatible.*

The following result states that the local condition that every pair of splits in $\mathcal{S}$ is compatible suffices to ensure that we have the global property that all splits in $\mathcal{S}$ can be realized simultaneously on a single unrooted phylogenetic tree:

Theorem 5.3.2 (Compatibility Theorem) *Let $\mathcal{S}$ be a set of splits on $\mathcal{X}$ and assume that $\mathcal{S}$ contains all trivial splits on $\mathcal{X}$. There exists a unique unrooted phylogenetic tree T that realizes $\mathcal{S}$, that is, with $\mathcal{S}(T) = \mathcal{S}$, if and only if $\mathcal{S}$ is compatible.*

This result was first formulated and proved in [37]. One way to see that the result holds is simply to apply the outgroup trick and then the result follows from the analogous result for clusters and rooted phylogenetic trees, which is shown in Section 6.4.

$\{a,b,c\}|\{d,e,f\}$,
$\{a,b,e,f\}|\{c,d\}$,
$\{a,b,f\}|\{c,d,e\}$,
$\{a,c\}|\{b,d,e,f\}$



(a) Set of splits (b) Incompatibility graph

Figure 5.3 (a) A set of four splits $\mathcal{S}$ on $\mathcal{X} = \{a, \ldots, f\}$. (b) The corresponding incompatibility graph $IG(\mathcal{S})$, which has four nodes and four edges.

In the above theorem we require that the set of splits $\mathcal{S}$ contains all trivial splits on $\mathcal{X}$. This assumption can be dropped if we use $\mathcal{X}$-trees rather than phylogenetic trees in the theorem.

It is often useful to represent the incompatibilities among a set of splits $\mathcal{S}$ by a graph that is defined as follows [9] (see Figure 5.3):

Definition 5.3.3 (Incompatibility graph) *The* incompatibility graph $IG(\mathcal{S})$ *of a set of splits* $\mathcal{S}$ *is the graph* (V, E) *that has node set* $V = \mathcal{S}$ *and edge set* $E = \big\{\{S_1, S_2\} \mid S_1$ *and* S_2 *are incompatible*$\big\}$.

A split $S \in \mathcal{S}$ is compatible with all other splits in $\mathcal{S}$, if and only if it is an isolated node in the incompatibility graph. In consequence, a set of splits $\mathcal{S}$ is compatible if and only if the incompatibility graph $IG(\mathcal{S})$ has no edges.

5.4 Splits and clusters

Splits are the unrooted counterparts of clusters and the two concepts are closely related. As discussed in Section 3.16 and Chapter 6, clusters can be represented by rooted phylogenetic trees or networks. In this section, we discuss how to transform splits to clusters and vice versa.

To obtain a set of clusters $\mathcal{C}$ from a set of splits $\mathcal{S}$ on $\mathcal{X}$, we must first choose an *outgroup* taxon $o \in \mathcal{X}$. Then, for each split $S = \frac{A}{B}$ in $\mathcal{S}$, we define the cluster C associated with S to be the split part that does not contain o, that is, we set $C = \bar{S}(o)$. We usually also consider $\{o\}$ as a cluster, to ensure that all trivial clusters are present in $\mathcal{C}$.

Exercise 5.4.1 (Splits to clusters) *Prove that this assignment preserves incompatibilities, in other words, that* $\mathcal{S}$ *is compatible if and only if* $\mathcal{C}$ *is compatible (see Definition 6.2.2).*

Exercise 5.4.2 (Splits to clusters example) *Using* c *as an outgroup, list all clusters for the set of splits shown in Figure 5.3(a).*

Let $\mathcal{S}$ be a set of splits on $\mathcal{X}$. If $\mathcal{S}$ is compatible, then there exists an unrooted phylogenetic tree T that represents $\mathcal{S}$. In this case, an alternative method to define the associated set of clusters $\mathcal{C}$ is to choose a root ρ for T and then to let $\mathcal{C}$ be the set of all clusters represented by the rooted version of T. If we choose a node of T to be the root, then there is a simple one-to-one correspondence between the splits and clusters. On the other hand, if the root is chosen so as to subdivide some edge e of T, then the split $S = \frac{A}{B}$ associated with e gives rise to precisely two clusters, namely A and B.

Now let us look at the opposite problem of defining a set of splits $\mathcal{S}$ for a given set of clusters $\mathcal{C}$ on $\mathcal{X}$. For a given cluster C, we could simply define the associated split S as C versus the complement of C, that is, as $S = \frac{C}{(\mathcal{X}-C)}$. Unfortunately, this assignment does not preserve incompatibilities. For example, the clusters $\{a, b\}$ and $\{b, c\}$ on $\mathcal{X} = \{a, b, c\}$ are incompatible, whereas the two splits associated in the manner suggested, $\frac{\{a,b\}}{\{c\}}$ and $\frac{\{a\}}{\{b,c\}}$, are not. To address this problem, we add a new (formal) *outgroup* taxon $o \notin \mathcal{X}$ that is then always placed in the split part that contains the complement of a cluster. In other words, for every cluster C we define the associated split as $S = \frac{C}{(\mathcal{X}-C)\cup\{o\}}$ on $\mathcal{X}' = \mathcal{X} \cup \{o\}$.

In the special case that the set of clusters $\mathcal{C}$ is compatible, and thus corresponds to some rooted phylogenetic tree T, we can obtain the set of associated splits directly from T as $\mathcal{S}(T)$, after first unrooting the tree.

Exercise 5.4.3 (Number of splits and clusters) *How many different splits and clusters are possible on a set $\mathcal{X}$ of n taxa?*

5.4.1 Optimal compatible subsets

Let $\mathcal{S}$ be a set of splits on $\mathcal{X}$. If $\mathcal{S}$ is incompatible, then there are two basic computational problems that are sometimes of interest. The first problem is to remove a minimum number of splits such that the remaining set of splits is compatible:

Problem 5.4.4 (Maximum compatibility problem) *Determine a maximum-size subset of splits $\mathcal{S}' \subset \mathcal{S}$ that is compatible.*

The second problem is to remove a minimum number of taxa such that the set of splits induced on the remaining taxa is compatible:

Problem 5.4.5 (Maximum compatible subset problem) *Determine a maximum-size subset of taxa $\mathcal{X}' \subset \mathcal{X}$ such that the set of splits $S|_{\mathcal{X}'}$ induced on $\mathcal{X}'$ is compatible.*

It follows from the NP-completeness of the two analogous problems formulated for clusters in Section 6.2.1 that these two problems are NP-complete.

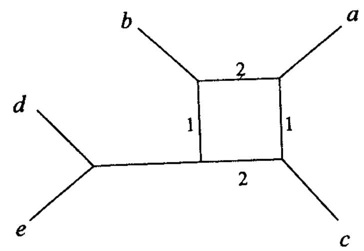


Figure 5.4 A split network N on $\mathcal{X} = \{a, \dots, e\}$. Every leaf edge, and, in this example, also the edge separating the taxa d and e from the rest of the taxa, each represents a split in the same way that they would in an unrooted phylogenetic tree. The two edges labeled 1 together represent the split $\frac{\{a,b\}}{\{c,d,e\}}$, whereas the two edges labeled 2 represent the split $\frac{\{a,c\}}{\{b,d,e\}}$.

5.5 Split networks

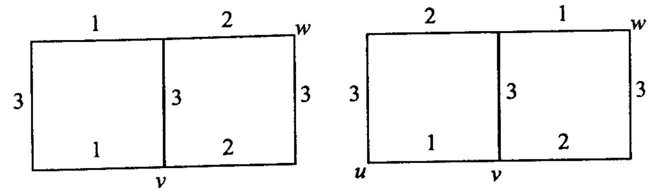
As we have seen, any set of compatible splits (containing all trivial splits) corresponds to an unrooted phylogenetic tree on $\mathcal{X}$. In this section, we introduce a mathematical generalization of the concept of an unrooted phylogenetic tree, called a *split network*, which can be used to represent an arbitrary, in particular, incompatible, set of splits.

In a split network, we use one or more edges to represent a split. The set of edges used to represent a given split S has the property that deletion of all these edges produces exactly two connected components, and, as in the case of phylogenetic trees, the two parts of S are given by the sets of taxa that occur as labels of one component, or the other, respectively. For an example, see Figure 5.4.

First we define the concept of a split graph. Think of this as the underlying graph-theoretical concept, just like phylogenetic trees are based on the underlying graph-theoretical concept of a tree, as defined in Chapter 1. A split network N is then obtained from such a graph by specifying a labeling of the leaves by taxa and a labeling of the edges by splits.

Below, we define a split graph as a finite, connected graph, together with an edge coloring whose most important property is that deleting all edges of a fixed color produces precisely two connected components. In a split network, all edges of a given color represent one split S and things are arranged such that the two connected components separated by the edges are each labeled by the taxa of one of the two parts of the split.

To develop the necessary formal concepts, let $G = (V, E)$ be a finite connected graph. Further, let K denote a finite set of labels, which we call *colors*. An *edge coloring* is a mapping $\sigma : E \to K$ that has the property that no two adjacent edges are given the same color. Such a mapping is called *surjective*, if it uses all colors in K.



(a) Isometric coloring (b) Non-isometric coloring

Figure 5.5 (a) A split graph with an isometric coloring of its edges using the colors $K = \{1, 2, 3\}$. All shortest paths between any two nodes, for example v and w, contain the same set of colors. (b) The coloring indicated here is not isometric; for example, one shortest path between v and w has colors 1 and 3, whereas the other one has colors 2 and 3. Additionally, there exists a shortest path connecting the nodes u and w that is not properly colored, as it contains two edges with color 1.

Consider an undirected path,

$$P = (u_0, e_1 = \{u_0, u_1\}, u_1, e_2 = \{u_1, u_2\}, \ldots, e_t = \{u_{t-1}, u_t\}, u_t), \tag{5.5}$$

containing exactly $\text{len}(P) = t$ different edges. We denote the set of colors that occur in P by $\sigma(P)$, defined as

$$\sigma(P) = \{\sigma(e_1), \ldots, \sigma(e_t)\} \tag{5.6}$$

and call P *properly colored*, if all edges in P have different colors, that is, if $|\sigma(P)| = t$. We call an edge coloring σ of G an *isometric coloring*, if for any two nodes, all shortest paths between them are properly colored and use exactly the same set of colors (see Figure 5.5).

Definition 5.5.1 (Split graph) *A* split graph *consists of a finite, simple, connected, bipartite graph $G = (V, E)$, together with an edge coloring $\sigma : E \to K$ that is surjective and isometric.*

With this definition, we get the crucial property that deletion of all edges of any given color produces precisely two connected components [62]:

Theorem 5.5.2 (Deletion of split produces two components) *Let $(G = (V, E), \sigma : E \to K)$ be a split graph. For any color $c \in K$ we have: the graph G_c obtained by deleting all edges of color c consists of precisely two connected components, which we denote by $G_c^0 = (V_c^0, E_c^0)$ and $G_c^1 = (V_c^1, E_c^1)$.*

Proof For two nodes v and w we use $d(v, w)$ to denote the minimal number of edges in any path from v to w, and we use $\sigma(v, w)$ to denote the set of colors that occur in every shortest path from v to w. Note that σ is an isometric coloring if $d(v, w) = |\sigma(v, w)|$ for all $v, w \in V$.

First we show that *any* path P from v to w uses all colors from $\sigma(v, w)$, by induction on $\mathrm{len}(P) \geq d(v, w)$.

To start the induction, assume $\mathrm{len}(P) = d(v, w)$. In this case P is a shortest path from v to w and so the claim follows directly from the definition of $\sigma(v, w)$.

Now, assume that $\mathrm{len}(P) > d(v, w)$ and so $P = (v, e_1, \ldots, w', e_t, w)$, for some $t > d(v, w)$. Because G is bipartite, we cannot have $d(v, w) = d(v, w')$ and so we must either have

$$d(v, w) = d(v, w') - 1 \quad \text{or} \quad d(v, w) = d(v, w') + 1. \tag{5.7}$$

In the first case every shortest path from v to w can be extended to a shortest path from v to w', implying

$$\sigma(v, w) \subseteq \sigma(v, w'). \tag{5.8}$$

Now, because $\mathrm{len}(v, e_1, \ldots, e_{t-1}, w') < \mathrm{len}(P)$, we can apply the induction hypothesis and obtain

$$\sigma(v, w) \subseteq \sigma(v, w') \subseteq \sigma(v, e_1, \ldots, e_{t-1}, w') \subseteq \sigma(P), \tag{5.9}$$

as claimed.

In the other case, we have

$$\sigma(v, w) = \sigma(v, w') \cup \{\sigma(e_t)\}. \tag{5.10}$$

Moreover, because $\mathrm{len}(v, e_1, \ldots, e_{t-1}, w') < \mathrm{len}(P)$, we also have

$$\sigma(v, w') \subseteq \sigma(v, e_1, \ldots, e_t, w'). \tag{5.11}$$

$t\text{-}1$

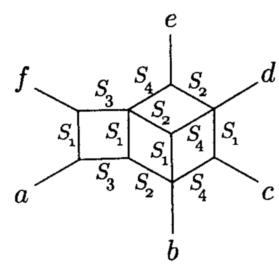
Putting these two observations together, we see that

$$\sigma(v, w) = \sigma(v, w') \dot{\cup} \{\sigma(e_t)\} \subseteq \sigma(v, e_1, \ldots, e_{t-1}, w') \dot{\cup} \{\sigma(e_t)\} = \sigma(P), \tag{5.12}$$

as claimed.

The proved statement implies that any two nodes v and w are in the same connected component of G_c if and only if $c \notin \sigma(v, w)$ holds. This implies that G_c must have *at least* two components, as σ is surjective. In the remainder of the proof, we show that G_c has *at most* two components.

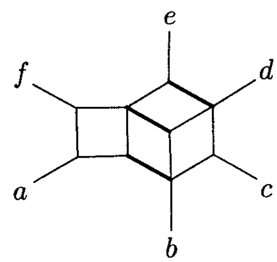
Consider an edge $e = \{w', w\}$ that has color c and assume that $P = (v, e_1, \ldots, e_t, w')$ is a shortest path from some node v to w', then, as above, there are two possible cases: (1) If $d(v, w) = d(v, w') + 1$, then v and w' must lie in the same component of G_c, because $(v, e_1, \ldots, e_t, w', e, w)$ is a shortest path from v to w, which in turn implies (by the isometric property) that $c = \sigma(e) \notin \{\sigma(e_1), \ldots, \sigma(e_t)\}$ holds. (2) If $d(v, w) = d(v, w') - 1$, then any shortest path



(a) Split network N

$S_1 = \{a,b,c\}|\{d,e,f\}$
$S_2 = \{a,e,f\}|\{b,c,d\}$
$S_3 = \{a,f\}|\{b,c,d,e\}$
$S_4 = \{a,b,f\}|\{c,d,e\}$

(b) Non-trivial splits



(c) Split $S_2 = \frac{\{a,e,f\}}{\{b,c,d\}}$

Figure 5.6 (a) A split network N representing all trivial splits on $\mathcal{X} = \{a, \ldots, f\}$ and the four non-trivial splits listed in (b). All the edges representing a particular non-trivial split are labeled by that split. However, the labeling of edges by splits is usually omitted, as shown in (c). Edges representing the same split, such as the three edges shown in bold lines representing S_2, are drawn parallel and with the same length.

$(v, e'_1, \ldots, e'_{t-1}, w)$ from v to w gives rise to a shortest path $(e'_1, \ldots, e'_{t-1}, e)$ from v to w'. This implies, as above, that $c = \sigma(e) \notin \{\sigma(e'_1), \ldots, \sigma(e'_{t-1})\} = \sigma(v, w)$ holds.

So, in G_c every node v is connected either to w or to w', but not to both of these nodes. This implies that G_c has at most two components. □

A split network N on $\mathcal{X}$ is obtained from a split graph by labeling its nodes by a taxon set $\mathcal{X}$ and labeling its edges by a set of splits $\mathcal{S}$ on $\mathcal{X}$ [62]:

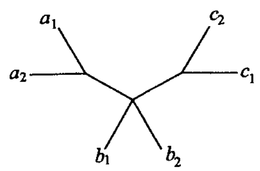
Definition 5.5.3 (Split network) *Let $\mathcal{S}$ be a set of splits on $\mathcal{X}$. A* split network $N = (V, E, \sigma, \lambda)$ *that represents $\mathcal{S}$ is given by a split graph $(G = (V, E), \sigma : E \to \mathcal{S})$ and a node labeling $\lambda : \mathcal{X} \to V$, with the property that for every split $S = A \mid B$ in $\mathcal{S}$ we have:*

$$A = \bigcup_{v \in V_S^0} \lambda^{-1}(v) \quad \text{and} \quad B = \bigcup_{v \in V_S^1} \lambda^{-1}(v), \tag{5.13}$$

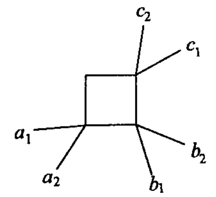
(or vice versa), in other words, deletion of all edges of color S produces a graph consisting of precisely two connected components, one containing all nodes labeled with elements of A and the other containing all nodes labeled with elements of B.

Formally, all edges representing a given split S are labeled by that split. In drawings, however, the labeling or coloring of the edges is usually omitted and it is customary, instead, to draw all edges representing the same split as parallel lines of the same length, so as to help make clear which edges belong together, see Figure 5.6. Such a representation is always possible, see Section 13.8.

Note that any unrooted phylogenetic tree T can be regarded as a split netw
Define $\sigma : E \to \mathcal{S}$ as $\sigma(e) := \sigma_T(e)$, then it is obvious that deletion of all e



(a) Split network N_1



(b) Split network N_2

Figure 5.7 The two split networks N_1 and N_2 shown in (a) and (b) both represent the same set of splits $\mathcal{S}$, consisting of the two non-trivial splits $\frac{\{a_1,a_2\}}{\{b_1,b_2,c_1,c_2\}}$ and $\frac{\{a_1,a_2,b_1,b_2\}}{\{c_1,c_2\}}$, together with all trivial splits on $\mathcal{X} = \{a_1, a_2, b_1, b_2, c_1, c_2\}$. Because $\mathcal{S}$ is compatible, there exists a corresponding split network that is a tree, namely N_1. The split network N_2 also represents $\mathcal{S}$, but is not a tree.

(that is, in fact, only one edge) of color $S = \sigma_T(e)$ produces two connected components which correspond to the two parts of S. So in this sense, split networks are a generalization of unrooted phylogenetic trees. Moreover, we have:

Lemma 5.5.4 (Split networks and compatiblity) *A set of splits $\mathcal{S}$ on $\mathcal{X}$ is compatible if and only if there exists a split network N representing $\mathcal{S}$ that is a tree.*

Figure 5.7 demonstrates that the split network associated with a set of splits is not uniquely defined and also that a split network representation of a compatible set of splits need not necessarily be a tree, although a tree representation for a compatible set of splits always exists. We would like to point out, however, that both algorithms described in Chapter 7 for constructing a split network for a given set of splits produce a tree, when run with a compatible set of splits as input.

Exercise 5.5.5 (Draw a split network) *Consider the set of splits $\mathcal{S}$ on $\mathcal{X} = \{a, \ldots, g\}$ given by*

$$\frac{\{a, b\}}{\{c, d, e, f, g\}}, \quad \frac{\{a, b, c\}}{\{d, e, f, g\}}, \quad \frac{\{a, b, e, f, g\}}{\{c, d\}}, \quad \frac{\{a, f, g\}}{\{b, c, d, e\}}, \tag{5.14}$$

and all trivial splits on $\mathcal{X}$. Draw a split network N that represents $\mathcal{S}$.

5.6 The canonical split network

Let $\mathcal{S}$ be a set of splits on $\mathcal{X}$. The definition of a split network (Definition 5.5.3) gives no indication of how one might obtain such a split network for $\mathcal{S}$. To address this, in this section we discuss how to define a *canonical split network* N that represents $\mathcal{S}$, which is also called the *Buneman graph* [17].

Definition 5.6.1 (Projection) *Let $\mathcal{S} = \{S_1, \ldots, S_k\}$ be a set of splits on $\mathcal{X}$. We call a vector $p = (D_1, \ldots, D_k)$ of length k a* projection *of $\mathcal{S}$, if it has the two following properties:*

(i) *The i-th component of p consists of one of the two split parts of the i-th split $S_i = \frac{A_i}{B_i}$ in $\mathcal{S}$, that is, we have either $D_i = A_i$ or $D_i = B_i$, for all $i = 1, \ldots k$.*
(ii) *Any two components of p have a non-empty intersection, that is, we have $D_i \cap D_j \neq \emptyset$ for all $i, j = 1, \ldots, k$.*

The core *of a projection $p = (D_1, \ldots, D_k)$ is defined as the set $\mathring{p} = \bigcap_{i=1}^{k} D_i$ of all such taxa that are contained in all components of p.*

We have the following easy result:

Lemma 5.6.2 (Every taxon in exactly one core) *Let $\mathcal{S}$ be a set of splits on $\mathcal{X}$. Then every taxon x in $\mathcal{X}$ is contained in the core $\mathring{p}$ of exactly one projection p of $\mathcal{S}$.*

Proof Let $\mathcal{S} = \{S_1, \ldots, S_k\}$ be a set of splits on $\mathcal{X}$. Consider any taxon x in $\mathcal{X}$. For $i = 1 \ldots, k$ let D_i be the split part of split S_i that contains x. Then x is contained in every component of $p = (D_1, \ldots, D_k)$ and so p is a projection and x is contained in $\mathring{p}$.

Now, assume that there exists a second projection $q \neq p$ whose core also contains x. Then p and q differ on at least one component, say the i-th component. The i-th component of p contains one part of the split S_i and q contains the other part. As the two split parts are disjoint, they cannot both contain x and so x cannot be contained in both cores, implying that such a second projection q does not exist. □

We can now define the canonical split network for a given set of splits:

Definition 5.6.3 (Buneman graph) *Let $\mathcal{S}$ be a set of splits on $\mathcal{X}$. The* canonical split network *or* Buneman graph *associated with $\mathcal{S}$ is the split network $N = (V, E, \sigma, \lambda)$ that is defined as follows [17]:*

(i) *The node set V is given by the set of all projections $\mathcal{P}(\mathcal{S})$ of $\mathcal{S}$.*
(ii) *Any two nodes p and p' are connected by an edge e of color $\sigma(e) = i$ in E if and only if p and p' differ precisely in their i-th component.*
(iii) *For each taxon x in $\mathcal{X}$, we set $\lambda(x) = p$, where p is the unique projection whose core contains x, i.e. with $x \in \mathring{p}$.*

As an example, consider the set of splits

$$\mathcal{S} = \{S_1, S_2, S_3, S_4\} = \left\{ \frac{\{a\}}{\{b, c, d, e\}}, \frac{\{a, b, c\}}{\{d, e\}}, \frac{\{a, b, d\}}{\{c, e\}}, \frac{\{a, c, e\}}{\{b, d\}} \right\} \quad (5.1$$

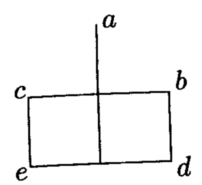


Figure 5.8 The canonical split network (Buneman graph) N for a set of splits $\mathcal{S}$ on $\mathcal{X} = \{a, b, c, d, e\}$ described in the text.

on the taxon set $\mathcal{X} = \{a, b, c, d, e\}$. There are exactly seven projections for $\mathcal{S}$ and these provide the set of nodes for the canonical split network N associated with $\mathcal{S}$:

$$\begin{array}{lccccl}
p_1 = (& \{a\}, & \{a,b,c\}, & \{a,b,d\}, & \{a,c,e\} &), \\
p_2 = (& \{b,c,d,e\}, & \{a,b,c\}, & \{a,b,d\}, & \{a,c,e\} &), \\
p_3 = (& \{b,c,d,e\}, & \{a,b,c\}, & \{a,b,d\}, & \{b,d\} &), \\
p_4 = (& \{b,c,d,e\}, & \{a,b,c\}, & \{c,e\}, & \{a,c,e\} &), \\
p_5 = (& \{b,c,d,e\}, & \{d,e\}, & \{a,b,d\}, & \{a,c,e\} &), \\
p_6 = (& \{b,c,d,e\}, & \{d,e\}, & \{a,b,d\}, & \{b,d\} &), \quad \text{and} \\
p_7 = (& \{b,c,d,e\}, & \{d,e\}, & \{c,e\}, & \{a,c,e\} &).
\end{array} \tag{5.16}$$

The cores of these projections are: $\mathring{p}_1 = \{a\}$, $\mathring{p}_2 = \emptyset$, $\mathring{p}_3 = \{b\}$, $\mathring{p}_4 = \{c\}$, $\mathring{p}_5 = \emptyset$, $\mathring{p}_6 = \{d\}$, and $\mathring{p}_7 = \{e\}$. Thus, the mapping of taxa to nodes is $\lambda(a) = p_1, \lambda(b) = p_3, \lambda(c) = p_4, \lambda(d) = p_6$ and $\lambda(e) = p_7$. The set of edges is given by all pairs of nodes that disagree on precisely one component: $e_1 = \{p_1, p_2\}$, $e_2 = \{p_2, p_3\}$, $e_3 = \{p_2, p_4\}$, $e_4 = \{p_2, p_5\}$, $e_5 = \{p_3, p_6\}$, $e_6 = \{p_4, p_7\}$, $e_7 = \{p_5, p_6\}$ and $e_8 = \{p_5, p_7\}$. Each edge is colored by the component on which the two incident nodes disagree on. In this example, the coloring of edges is $\sigma(e_1) = 1, \sigma(e_2) = \sigma(e_7) = 4$, $\sigma(e_3) = \sigma(e_8) = 3$ and $\sigma(e_4) = \sigma(e_5) = \sigma(e_6) = 2$. The resulting split network is shown in Figure 5.8.

Exercise 5.6.4 (Buneman graph and compatibility) *Show that the Buneman graph of a compatible set of splits $\mathcal{S}$ is an unrooted phylogenetic tree.*

Lemma 5.6.5 (Buneman graph is a split network) *Let $\mathcal{S}$ be a set of splits on $\mathcal{X}$. The Buneman graph for $\mathcal{S}$ is a split network for $\mathcal{S}$.*

Proof We must show that the Buneman graph is a split graph as defined in Definition 5.5.1 and then the claim follows with the help of Theorem 5.5.2.

Let $\mathcal{S} = \{S_1, \ldots, S_k\}$ be a set of splits on $\mathcal{X}$ and let N be the corresponding Buneman graph.

We first show that the edge coloring on N is surjective. Consider any number $i = 1, \ldots, k$. We must show that N contains an edge that has color i, that is, there exist two projections p and q that differ only on their i-th component. Let p_0 be some projection of $\mathcal{S}$. Let $A = \{D_j \mid D_j \subset D_i\}$ be the set of components of p_0 that are subsets of D_i, not including D_i, and let $B = \{D_j \mid D_j \not\subset D_i\}$ be the set of components that are not subsets of D_i. Define $\bar{A} = \{\mathcal{X} - D_j \mid D_j \in A\}$ as the set of complements of all the components in A. By definition of A and B we have the following: Each set in $\bar{A}$ intersects every component in B. Moreover, each set in $\bar{A}$ intersects all other sets in $\bar{A}$. Finally, D_i and $\mathcal{X} - D_i$ intersects all sets in $\bar{A}$ and B. Hence, the vector p obtained by replacing each component of p_0 that is listed in A by its complement is a projection. Similarly, the vector q obtained by replacing each component of p_0 that $\mathcal{S}$ in the set $A \cup \{D_i\}$ by its complement is a projection. As p and q differ only by their i-th component, we are done.

Our goal is to show that the the graph is connected, bipartite and the edge coloring is isometric, by proving the following statement: For any two nodes p and q that differ on exactly d components, the length of any shortest path P (and such a path always exists) from p to q is d and each of the indices $i_1, \ldots, i_d$ on which p and q differ occurs as an edge color in P.

The proof is by induction on the number d of components on which p and q differ. If $d = 1$, then there is indeed a path from p to q consisting of a single edge e and that edge is colored by the index of the component on which the two nodes differ.

Now, assume that $d > 1$ and that any two nodes that differ on $d - 1$ components are connected by a path of length $d - 1$ colored by the indices of the components on which the two nodes differ. Let p and q be two nodes that differ on exactly d components.

Let D_i be a split part of some split S_i that occurs as the i-th component of p but not of q, and assume that D_i has minimum size with this property. Consider the vector p' that is obtained from p by by replacing D_i by the complementary split part $\mathcal{X} - D_i$.

As D_i is minimal, none of the other components of p can be completely contained in D_i and so they all must contain at least one element that is not contained in D_i. This implies that p' is a valid projection and hence a node of the Buneman graph.

Because p and q disagree on d components and because $\mathcal{X} - D_i$ is the i-th component of q, it follows that p' and q disagree on only $d - 1$ components. By assumption, all shortest paths from p' to q have length $d - 1$ and the edges are colored by the $d - 1$ indices of the components on which the two nodes differ, none colored by the index of D_i since p' and q do not differ on D_i.

Because p and p' differ by only one component, they are connected by an edge in N, which can be used to extend any shortest path from p' to q by one edge to obtain a path from p to q that has d edges and d different colors. Hence, there *exists* a path from v to w with the desired properties.

It remains to be shown that *all* shortest paths from v to w have the desired properties. To this end, consider an arbitrary path P of length d from p to q and let p' be the second node in the path. By assumption, p' is connected to q by a path containing $d-1$ edges, each colored by a different index and none colored by the index of the component on which p and p' differ. Hence, P has d different colors.

Let us demonstrate now that N is bipartite. Chose a fixed taxon x in $\mathcal{X}$. The *parity* of a projection p is set to 0, if the number of components of p that contain the taxon x is even, and is set to 1, otherwise. Using this measure, we can partition the set of nodes V of the Buneman graph in two disjoint subsets V_1 and V_2 such that for every edge $e \in E$ one of the endpoints lies in V_1 and the other endpoint lies in V_2, in the following way: Let V_1 and V_2 be the set of nodes with odd and even parity, respectively. Then, for each pair of nodes u, v in V_1 (or in V_2), there does not exist an edge connecting them since they differ on more than one component. This completes the proof that the Buneman graph is a split graph as defined in Definition 5.5.1.

To prove that the Buneman graph is a split network, we need to show for the node labeling $\lambda : \mathcal{X} \to V$ and for every split $S = A \mid B$ in $\mathcal{S}$ that the deletion of all edges of color S produces a graph consisting of precisely two connected components, one containing all nodes labeled with elements of A and the other containing all nodes labeled with elements of B (see Definition 5.5.1). Theorem 5.5.2 ensures that, for each split S of a split set $\mathcal{S}$ represented by N, by deleting all edges of color S we obtain a graph N_S consisting of precisely two connected components N_S^0 and N_S^1. We need to prove that all nodes labeled with elements of A (or of B) are contained in N_S^0 (or N_S^1, respectively).

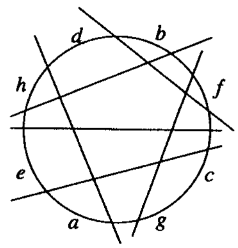
In the proof of Theorem 5.5.2, we showed that any two nodes p and q are contained in the same connected component of N_S if and only if $S \notin \sigma(p,q)$ holds. For every pair of nodes p and q such that $\lambda^{-1}(p) = a$ and $\lambda^{-1}(q) = b$, with a in A and b in B, it holds that p and q differ on the split S. As demonstrated above, this implies that $S \in \sigma(p,q)$, so p and q are not in the same connected component of N_S^0 and N_S^1. Since this holds for each pair of nodes p and q in V and for every split, this concludes the proof. □

Let $\mathcal{S}$ be a set of splits on $\mathcal{X}$. How many nodes and edges might the canonical split network contain? If the incompatibility graph $IG(\mathcal{S})$ contains a clique of size k, then any one of the 2^k possible choices of split parts gives rise to a node in the network and thus the number of nodes and edges of the network is exponential in the number of splits in the worst case.

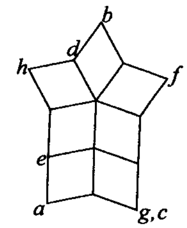
In Chapter 7 we discuss how to compute the Buneman graph using the convex hull algorithm.

$\{a,b,d,e,h\} \mid \{c,f,g\}$
$\{a,c,d,e,g,h\} \mid \{b,f\}$
$\{a,c,e,g\} \mid \{b,d,f,h\}$
$\{a,c,g\} \mid \{b,d,e,f,h\}$
$\{a,c,e,f,g\} \mid \{b,d,h\}$
$\{a,e,h\} \mid \{b,c,d,f,g\}$

(a) Circular splits



(b) Circular ordering

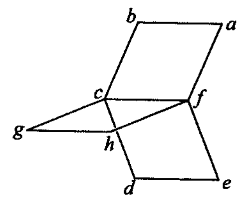


(c) Planar network

Figure 5.9 (a) A set of six circular splits $\mathcal{S}$ on $\mathcal{X} = \{a, b, \ldots, h\}$. (b) An arrangement of the taxa around a circle such that every split $S = A \mid B \in \mathcal{S}$ can be realized by a straight line through the circle that separates the two split parts A and B. A circular ordering is given by (a, g, c, f, b, d, h, e). (c) An outer-labeled planar split network representing $\mathcal{S}$.

$\{a,b\} \mid \{c,d,e,f,g,h\}$
$\{a,b,c,d,e,f\} \mid \{g,h\}$
$\{a,b,c,f,g,h\} \mid \{d,e\}$
$\{a,e,f,h\} \mid \{b,c,d,g\}$

(a) Non-circular splits



(b) Non-planar network

Figure 5.10 (a) A set of four non-circular splits $\mathcal{S}$ on $\mathcal{X} = \{a, b, \ldots, h\}$. (b) A non-planar split network representing $\mathcal{S}$.

5.7 Circular splits and planar split networks

One practical problem that arises when working with split networks is that the networks can be very complicated and thus difficult to visualize in a comprehensible way. Hence, a number of restricted classes of sets of splits have been introduced in an attempt to avoid overly complicated networks. The two most important are *circular* splits, which are the focus of this section, and *weakly compatible* splits, which we introduce in the next section.

Informally, a set of splits $\mathcal{S}$ on $\mathcal{X}$ is called *circular*, if the taxa in $\mathcal{X}$ can be placed around a circle in such a way that each split $S = \frac{A}{B}$ can be realized by a line through the circle that separates the plane into two half-planes, one containing all taxa in A and the other containing all taxa in B (see Figure 5.9). An example of a non-circular set of splits and the corresponding split network is shown in Figure 5.10. More formally [9]:

Definition 5.7.1 (Circular splits) *A set of splits $\mathcal{S}$ on $\mathcal{X}$ is called* circular, *if there exists a linear ordering $(x_1, \ldots, x_n)$ of the elements of $\mathcal{X}$* for $\mathcal{S}$ *such that each*