Compositions of series-parallel graphs

Brian Kell

March 29, 2006

Abstract

A composition of a graph is a partition of the vertex set such that the subgraph induced by each part is connected. In this paper we shall survey past results about compositions of graphs and present a new result which yields a linear-time method for computing the number of compositions of a series-parallel graph.

Contents

Acknowledgements

I am very grateful to my research advisor, Dr. Jamie Radcliffe, without whose guidance and support over the past few months this thesis would never have been written. His enthusiasm for mathematics is contagious, and his insight has been invaluable.

Dr. Jonathan Cutler has also been of much assistance. He has read several revisions of this thesis and given me helpful opinions and suggestions.

I also wish to thank Dr. Judy Walker for suggesting that I make compositions of graphs my topic of research, for teaching my graph theory class this semester, and for encouraging me to apply for the Nebraska Mentoring through Critical Transition Points program, which has graciously funded my work.

Further thanks go to Dr. Gordon Woodward, who among other things finally succeeded in persuading me to become a math major; Dr. Glenn Ledder, whose confidence in my work is much appreciated; Dr. Sohrab Asgarpoor, who provided answers about the electrical engineering implications of this topic; and Lori Mueller, who has helped to keep me on track and has taken care of many important administrative tasks.

Finally, I must thank my parents, Tom and Patti Kell, for their unceasing support and encouragement, and for lending me their basement for a few days as I typed this paper.

Chapter 1

Introduction

A *graph* is a mathematical object consisting of a set of *vertices*, or points, and a set of *edges* connecting pairs of vertices. Typically graphs are drawn with dots representing the vertices and straight or curved lines representing the edges. For instance, Figure 1.1 shows a graph with seven vertices and nine edges.

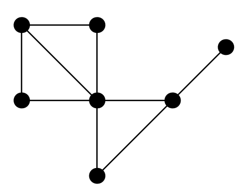

Figure 1.1: A graph.

Graphs are the focus of the branch of mathematics known as graph theory. They are often used to model networks, such as electrical circuits, computer networks, or highway systems. Results from graph theory can be used to match jobs to qualified people, to design efficient methods for representing data in computers, to analyze chemical compounds, and to determine the optimal order in which to perform a series of interdependent tasks. As computers increase in importance, more applications of graph theory are appearing, as graphs provide a convenient mathematical model for real-world problems.

One question that might be asked about a graph is how many different ways the graph can be cut up into pieces. There are several ways to interpret this question; the interpretation we shall examine here considers only the vertices which make up each part, and ignores the edges, except that the

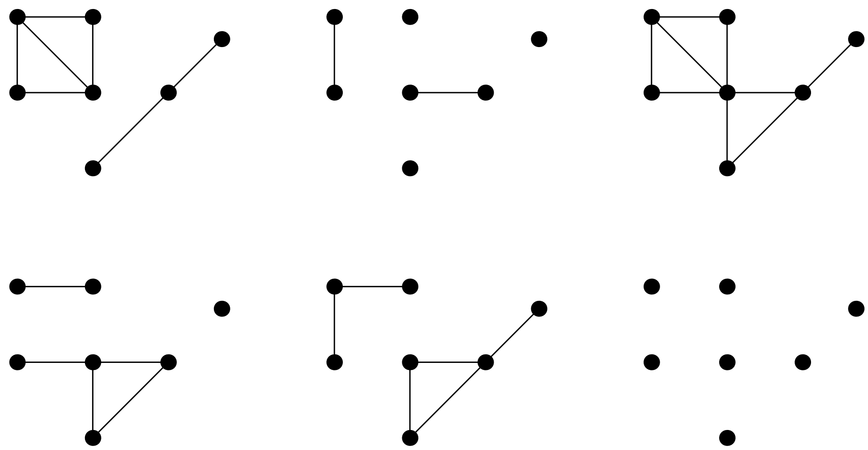

Figure 1.2: Some compositions of the graph in Figure 1.1.

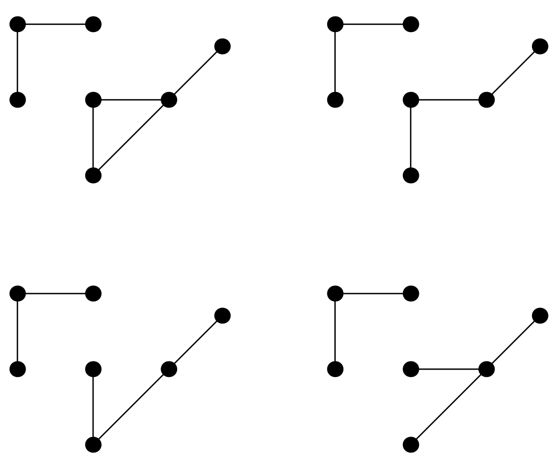

Figure 1.3: Four “cuttings” that yield the same composition.

vertices in each part must be connected. Each such separation of the graph into pieces is called a *composition* of the graph. For example, the graph of Figure 1.1 has 130 compositions. Six of these compositions are shown in Figure 1.2.

Note in particular that one composition contains all the edges of the graph; this is produced by not "cutting" any of the edges at all. Another composition is formed by cutting all the edges in the graph, making one piece for each vertex. An important point is that the four "cuttings" in Figure 1.3 are considered to be the same composition, since the pieces produced contain the same vertices.

If G is an arbitrary graph, then, how many compositions does G have? This is an open question in graph theory. The answer is known for certain basic types of graphs, but no efficient technique for computing the number of compositions of a general graph is known.

In this paper we consider a class of graphs called *series-parallel graphs*. Such a graph has two distinguished vertices, called the *terminals*. The simplest series-parallel graph has only two vertices and a single edge connecting them, as shown in Figure 1.4. (The terminals of a series-parallel graph will be drawn as unfilled circles to distinguish them from the other vertices.)

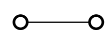

Figure 1.4: The simplest series-parallel graph.

More complicated series-parallel graphs may be constructed from simpler series-parallel graphs by joining them in *series* or in *parallel*. Joining graphs in series can be imagined as "chaining" the graphs together, connecting them at their terminals. For example, the graph in Figure 1.5 is formed by chaining together four copies of the simple graph of Figure 1.4.

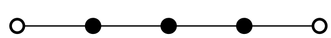

Figure 1.5: A graph formed by joining simpler graphs in series.

When two or more graphs are joined in parallel, on the other hand, they are joined at both terminals. For example, the three graphs in Figure 1.6 can be joined in parallel to form the graph in Figure 1.7.

Each of the graphs in Figure 1.6 are series-parallel graphs, so the graph in Figure 1.7 is also a series-parallel graph. This larger graph can itself be joined with another graph in series or in parallel. These graphs can of course become arbitrarily large. Figure 1.8 shows a series-parallel graph with 40 vertices and 53 edges. Not all graphs are series-parallel, however.

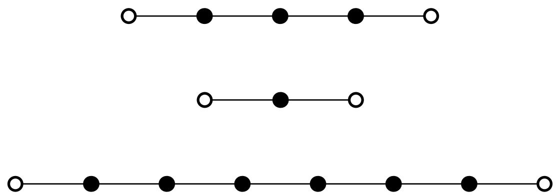

Figure 1.6: Three series-parallel graphs.

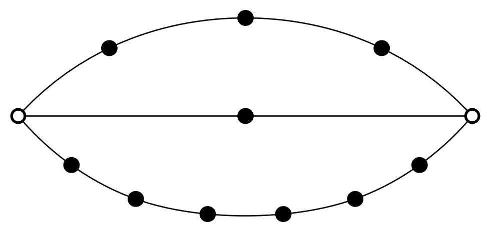

Figure 1.7: The graphs of Figure 1.6 joined in parallel.

The graph in Figure 1.9 is a common example of a graph that is not series-parallel, because it cannot be constructed from simpler series-parallel graphs by joining them in series or in parallel. In the context of electrical networks, this graph is often called the Wheatstone bridge.

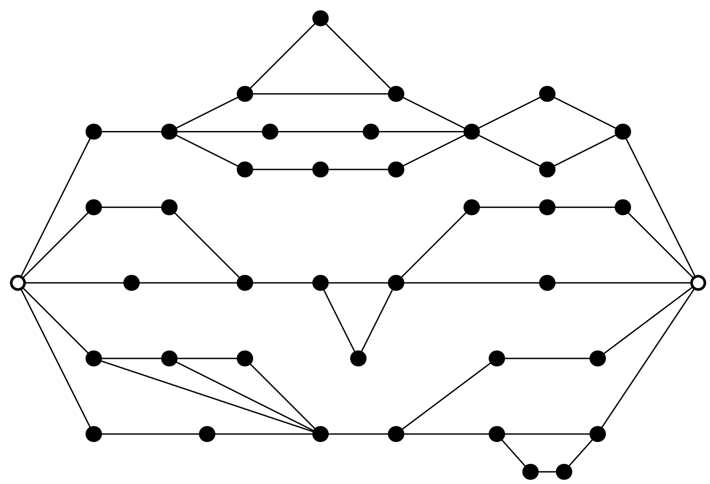

Figure 1.8: A larger series-parallel graph.

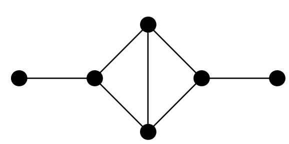

Figure 1.9: A graph that is not series-parallel.

Chapter 2

Definitions and notation

In this chapter we shall set forth some definitions and notation which will be used throughout the rest of this paper. Many of these definitions are common graph theory terms; the reader may refer to Clark and Holton [6], Gould [10], Thulasiraman and Swamy [24], Wilson [25], or any number of other graph theory texts for further information.

2.1 Fundamental concepts in graph theory

A *graph* G is an ordered pair (V, E) of sets. The set V, called the *vertex set* of G and alternately denoted by $V(G)$, is a nonempty set of elements called the *vertices* of G. The set E, called the *edge set* of G and alternately denoted by $E(G)$, is a possibly empty set of elements called the *edges* of G, such that each edge e in $E(G)$ is an unordered pair $\{u, v\}$ of vertices of G; these vertices are called the *end vertices* or *endpoints* of e.

For example, consider again the graph of Figure 1.1. It has been redrawn in Figure 2.1 with the vertices labeled v_1 through v_7 and the edges labeled e_1 through e_9. The vertex set of this graph is $\{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$, and the edge set is $\{e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8, e_9\}$. Each edge has two endpoints; for instance, the endpoints of e_3 are v_1 and v_5. Note that this mathematical definition of a graph does not include information about how the graph is drawn; for many purposes, graph theory is concerned only with the interconnectedness of the vertices and the edges.

In general, the endpoints of an edge need not be distinct. If the two endpoints of an edge are identical, then that edge is called a *loop*. Similarly, it is possible that two or more edges have the same endpoints; if this is the case, these edges are called *parallel*. For example, in the graph in Figure 2.2,

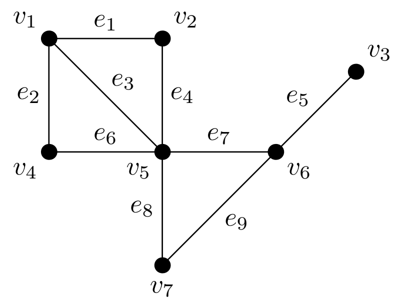

Figure 2.1: A graph with vertices and edges labeled.

the edge e_1 is a loop, because the vertex v_1 serves as both of its endpoints, and the edges e_4 and e_5 are parallel, since the vertices v_2 and v_4 are the endpoints of both of these edges. In this paper, however, we shall consider only graphs which have no loops and no parallel edges. Such graphs are called *simple*. This is really not a major constraint, since loops can be discarded and sets of parallel edges replaced by a single edge (producing what is called the *underlying simple graph*), and the number of compositions will not be affected. We shall also restrict our scope to graphs with finitely many vertices and edges.

If two vertices u and v are the endpoints of an edge e, we say that u and v are *joined* by e, that u and v are *adjacent* (written $u \operatorname{adj} v$), and that e is *incident* on u and v. Two edges that share an endpoint are also called *adjacent*. The *degree* of a vertex v is the number of edges incident on v (counting loops twice, if the graph is not simple).

It may be the case that two graphs have the same structure, differing only in how the vertices or edges are labeled or how the graph is drawn. In such a case we say that the two graphs are *isomorphic*. More precisely, a graph G is isomorphic to a graph H if there exists a one-to-one correspondence (a

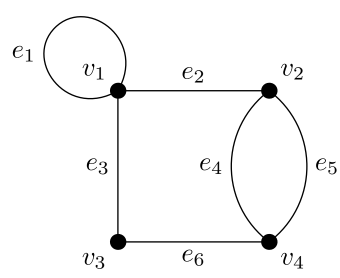

Figure 2.2: A graph with a loop and parallel edges.

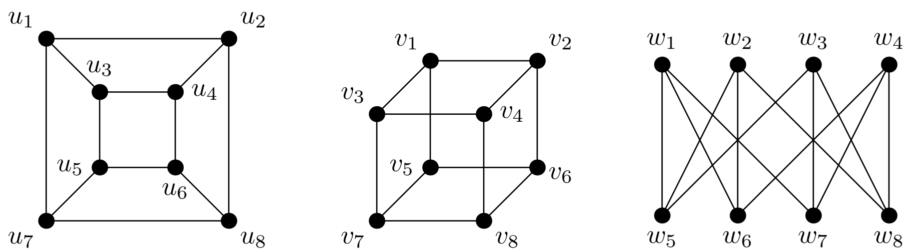

Figure 2.3: Three mutually isomorphic graphs.

bijection) between the vertices of G and the vertices of H, and a one-to-one correspondence between the edges of G and the edges of H, such that if e is an edge in G with endpoints u and v, then the corresponding edge f in H has as its endpoints the vertices w and x that correspond to u and v respectively. Such a pair of bijections is called a *graph isomorphism*. In this paper we shall not distinguish between isomorphic graphs, since if a graph G is isomorphic to a graph H, then G and H have the same number of compositions.

To give an example, the three graphs in Figure 2.3 are mutually isomorphic. For each pair of these graphs, there are several different isomorphisms. The fact that the graph on the left and the graph in the center are isomorphic is readily apparent; one isomorphism between them is as follows.

$$\begin{aligned} u_1 &\leftrightarrow v_1, & u_5 &\leftrightarrow v_7, \\ u_2 &\leftrightarrow v_2, & u_6 &\leftrightarrow v_8, \\ u_3 &\leftrightarrow v_3, & u_7 &\leftrightarrow v_5, \\ u_4 &\leftrightarrow v_4, & u_8 &\leftrightarrow v_6. \end{aligned}$$

However, it is not so easy to see that the graph on the left is isomorphic to the graph on the right. One isomorphism between these two graphs is given below.

$$\begin{aligned} u_1 &\leftrightarrow w_5, & u_5 &\leftrightarrow w_6, \\ u_2 &\leftrightarrow w_3, & u_6 &\leftrightarrow w_4, \\ u_3 &\leftrightarrow w_1, & u_7 &\leftrightarrow w_2, \\ u_4 &\leftrightarrow w_7, & u_8 &\leftrightarrow w_8. \end{aligned}$$

An *empty graph* is one that has no edges. The *singleton graph* is the graph with one vertex and no edges.

If G and H are graphs with $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$, then H is said to be a *subgraph* of G. A subgraph H is called a *proper subgraph* of G if $H \neq G$, that is, if $V(H) \subsetneq V(G)$ or $E(H) \subsetneq E(G)$, so that there exists a vertex in G that does not exist in H, or an edge in G that does not exist in H.

If U is a nonempty subset of the vertices of G, then the subgraph of G *induced* by U is the subgraph whose vertex set is U and whose edge set consists of all edges of G with both endpoints in U. Similarly, if F is a nonempty subset of the edges of G, then the subgraph of G induced by F is the subgraph whose edge set is F and whose vertex set consists of every vertex of G that is an endpoint of at least one edge in F. For instance, if we consider the graph of Figure 2.1, then the subgraph induced by the vertices $\{v_1, v_2, v_5, v_7\}$ is the subgraph shown on the left in Figure 2.4, and the subgraph induced by the edges $\{e_1, e_2, e_4, e_6, e_7\}$ is shown on the right.

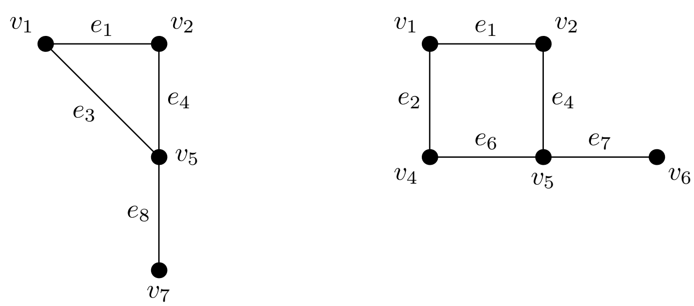

Figure 2.4: Subgraphs induced by a subset of vertices and a subset of edges.

If e is an edge of G, then $G - e$ denotes the subgraph of G obtained by deleting the edge e, that is, the subgraph whose vertex set is $V(G)$ and whose edge set is $E(G) \setminus \{e\}$. Analogously, if v is a vertex of G, then $G - v$ denotes the subgraph of G formed by deleting the vertex v and all edges incident on it, that is, the subgraph of G induced by $V(G) \setminus \{v\}$. Figure 2.5 shows two subgraphs of the graph G in Figure 2.1. The subgraph on the left is $G - e_9$; that on the right is $G - v_5$.

Two useful operations we can perform on a graph are the identification of vertices and the contraction of edges. We *identify* two vertices u and v of a graph when we replace them by a new vertex w such that all edges that were previously incident on either u or v are now incident on w. (Any edges incident on both u and v become loops incident on w.) We *contract* an edge e of a graph when we delete it and then identify its endpoints. For instance, if we identify the vertices v_2 and v_5 in Figure 2.1, we obtain the graph illustrated on the left in Figure 2.6. Note that the edges e_1 and e_3 are now parallel, and e_4 has become a loop. If instead we contract the edge e_7, we produce the graph on the right.

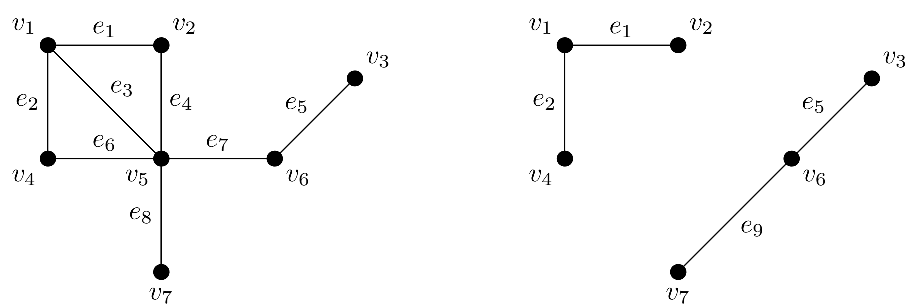

Figure 2.5: Subgraphs produced by deleting an edge and by deleting a vertex.

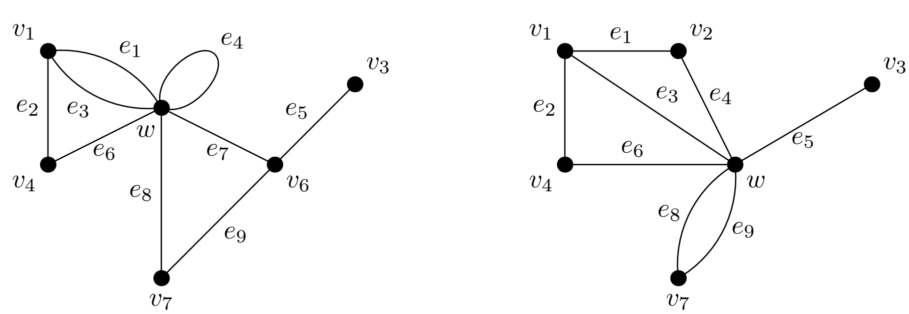

Figure 2.6: Graphs produced by the identification of vertices and the contraction of an edge.

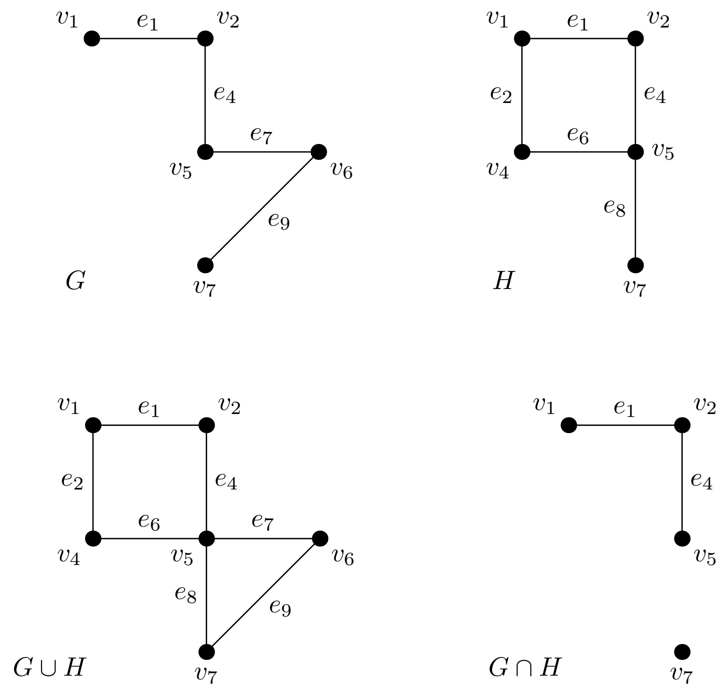

Figure 2.7: The union and intersection of two graphs.

The *union* of two graphs G and H, denoted $G \cup H$, is the graph whose vertex set is $V(G) \cup V(H)$ and whose edge set is $E(G) \cup E(H)$. In the same way, the *intersection* of two graphs G and H with at least one vertex in common, written $G \cap H$, is the graph whose vertex set is $V(G) \cap V(H)$ and whose edge set is $E(G) \cap E(H)$. (The requirement that G and H have at least one vertex in common is necessary since the vertex set of a graph must be nonempty.) Figure 2.7 shows two subgraphs of the graph in Figure 2.1, which are labeled G and H, and the union and intersection of these subgraphs. The union contains all vertices and edges that are in G or H, or both; the intersection contains only those vertices and edges that are in both G and H. Note that v_7 is an isolated vertex in the intersection.

The *join* of two graphs G and H, denoted $G + H$, is the graph whose vertex set is $V(G) \cup V(H)$ and whose edge set is $E(G) \cup E(H) \cup J$, where

$$J = \{\, (v_i, v_j) \mid v_i \in V(G) \text{ and } v_j \in V(H) \,\}.$$

In other words, the join of two graphs is their union together with edges that join every vertex in G to every vertex in H. Figure 2.8 illustrates the join of two graphs.

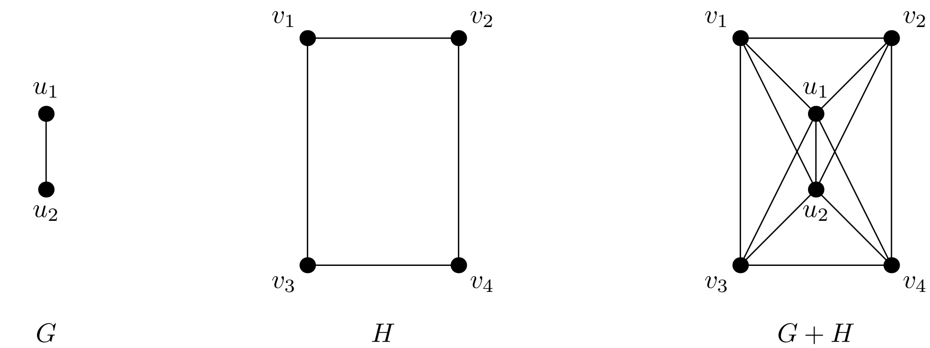

Figure 2.8: The join of two graphs.

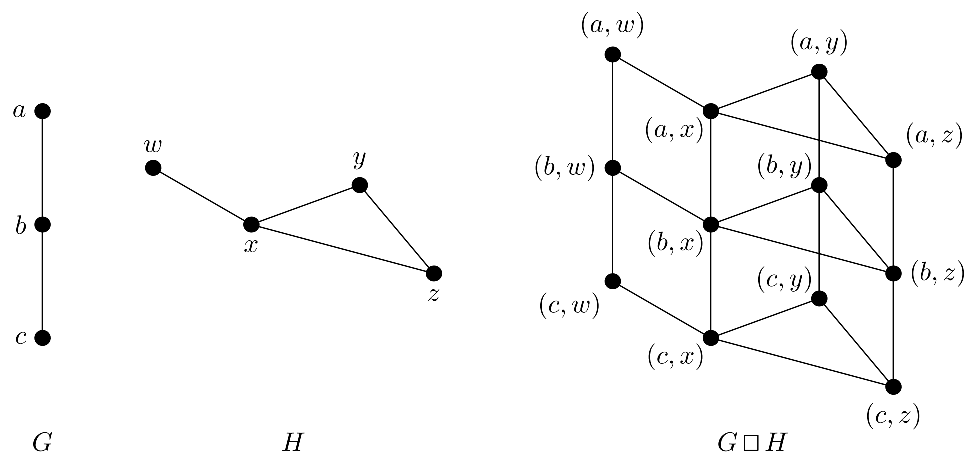

Figure 2.9: The Cartesian product of two graphs.

The *Cartesian product* of two graphs G and H, which we shall denote by $G \,\square\, H$ after Nešetřil [16], is the graph whose vertex set is $V(G) \times V(H)$ and where two vertices (u, v) and (x, y) are adjacent if and only if

$$(u = x) \text{ and } (v \operatorname{adj} y), \qquad \text{or} \qquad (u \operatorname{adj} x) \text{ and } (v = y).$$

Figure 2.9 depicts the Cartesian product of two graphs. The reader may refer to Imrich and Klavžar [11] for a more thorough discussion of the Cartesian product and its properties.

A finite sequence

$$v_0 e_1 v_1 e_2 v_2 \ldots v_{k-1} e_k v_k$$

is called a *walk* (from v_0 to v_k) in a graph G if for $0 \leq i \leq k$, v_i is a vertex in $V(G)$, and for $1 \leq i \leq k$, e_i is an edge in $E(G)$ with endpoints v_{i-1} and v_i.

The vertex v_0 is called the *origin* of the walk, and the vertex v_k is called the *terminus*. The vertices v_1, v_2, $\ldots$, v_{k-1} are called the *internal vertices* of the walk. A walk is called *closed* or *open* according as $v_0 = v_k$ or $v_0 \neq v_k$.

If all of the vertices v_0, v_1, v_2, $\ldots$, v_k of a walk are distinct, then the walk is called a *path*. It is easily seen that, given any two vertices u and v of a graph, any walk from u to v contains a path from u to v [6, Theorem 1.3]. For example, the heavy edges in Figure 2.10 form a walk from v_3 to v_4. There are several other walks from v_3 to v_4 in this graph.

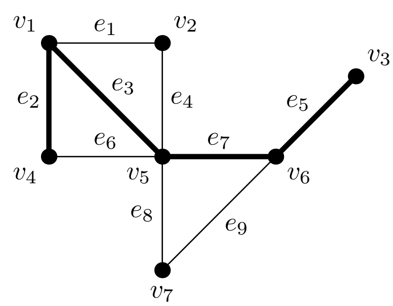

Figure 2.10: A path from v_3 to v_4.

Two vertices u and v of a graph G are said to be *connected* if there exists a path in G from u to v. A graph is called *connected* if every two of its vertices are connected; otherwise it is called *disconnected*. A connected subgraph of a graph G is called a *connected component* of G if it is not a proper subgraph of any other connected subgraph of G. We shall denote the number of connected components of a graph G by $k(G)$. For instance, the subgraphs G and H in Figure 2.7 are both connected, as is their union. However, their intersection is disconnected, since (to take an example) there does not exist a path from v_2 to v_7. The intersection has two connected components, so $k(G \cap H) = 2$.

If all of the edges e_1, e_2, $\ldots$, e_k of a walk are distinct, then the walk is called a *trail*. A nontrivial closed trail in which the origin and internal vertices are distinct is called a *cycle*. Figure 2.11 shows a graph with two cycles, which are drawn with heavy edges. A graph is called *acyclic* if it contains no cycles. A connected acyclic graph is called a *tree*. The three graphs in Figure 2.12 are trees.

A *bridge* is an edge e of a graph G such that $k(G - e) > k(G)$. In other words, a bridge is an edge such that its deletion increases the number of connected components of the graph. In fact, if e is a bridge, then $k(G-e) = k(G)+1$; that is, its deletion increases the number of connected components

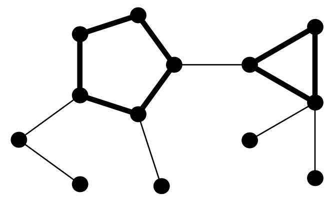

Figure 2.11: A graph with two cycles.

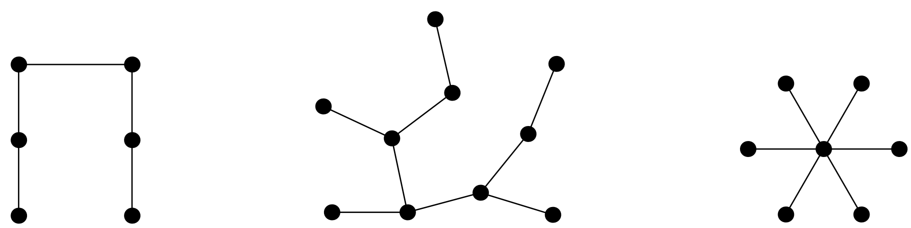

Figure 2.12: Three trees.

by exactly one [6, Theorem 2.6]. Similarly, a *cut vertex* is a vertex v of a graph G such that $k(G-v) > k(G)$. That is to say, a cut vertex is a vertex such that its deletion increases the number of connected components of the graph. For example, in the graph of Figure 2.1, the edge e_5 is a bridge, since if it were deleted the vertex v_3 would be isolated, forming two connected components. The vertex v_5 is a cut vertex, since its deletion disconnects the graph, as seen in the right-hand graph of Figure 2.5. The vertex v_6 is also a cut vertex in this graph.

2.2 Compositions of graphs

The intuitive concept of cutting a graph into pieces has been formalized by Arnold Knopfmacher and M. E. Mays in [13]. A *composition* of a graph is a partition of the vertex set into one or more parts, such that the subgraph induced by each part is connected.

This definition has several important implications. First, since a composition of a graph G is a partition of $V(G)$, every vertex of G is in exactly one of the parts of the composition. In other words, a composition of G yields a set of connected subgraphs of G, and every vertex of G is in exactly one of these subgraphs. Second, as shown in Figure 1.3, not every set of connected subgraphs yields a distinct composition. In particular, when a composition is illustrated as a set of subgraphs, the edges exist solely to indicate the connected components; there may be other choices of edges that would imply the same composition. Third, in general the vertices of G cannot be arbitrarily partitioned, since the subgraph induced by each part must be connected. This means, for example, that the partition

$$\{\,\{v_1, v_5\},\ \{v_2, v_3, v_6\},\ \{v_4, v_7\}\,\}$$

of the vertices of the graph in Figure 2.1 is not a composition of the graph. This partition is illustrated in Figure 2.13; it is clear that not all of the subgraphs induced by each part are connected.

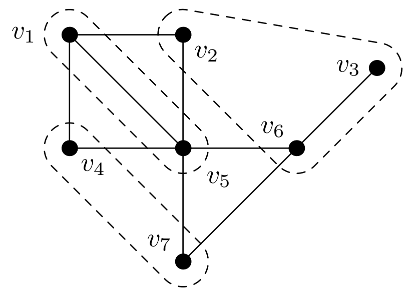

Figure 2.13: A partition of the vertices of a graph which does not yield a composition.

We shall use the notation $C(G)$, introduced in [13], to denote the number of compositions of a graph G.

A composition of a graph is a partition of the vertex set, which can naturally be viewed as an equivalence relation. For a given composition $\mathcal{C}$, then, we shall say that the vertices u and v are *together* in $\mathcal{C}$ (written $u \sim v$) if u and v are in the same part. We shall sometimes refer to the parts of the composition, which are subsets of the vertex set of the graph, as the *components* of the composition.

We shall say that an edge with endpoints u and v *belongs to* a particular composition if $u \sim v$ in that composition.

2.3 Standard families of graphs

Certain graphs arise so often in graph theory that they have a standard notation.

A *complete graph* is a simple graph in which every pair of vertices is joined by an edge. Any two complete graphs with the same number of vertices are isomorphic, so we refer to *the* complete graph on n vertices, denoted K_n. Figure 2.14 shows K_4, K_6, and K_{10}. Note that K_1 is the singleton graph.

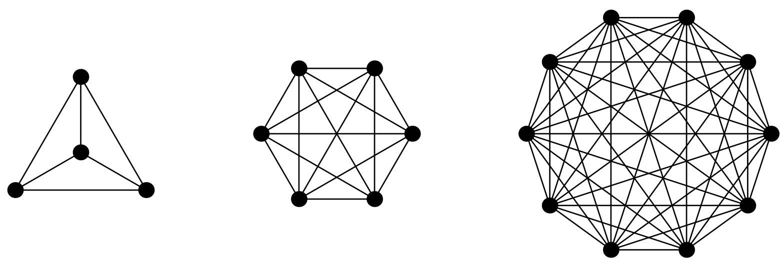

Figure 2.14: The complete graphs K_4, K_6, and K_{10}.

If the vertices of a graph G can be partitioned into two nonempty subsets X and Y (that is, $X \cup Y = V(G)$ and $X \cap Y = \varnothing$) in such a way that every edge of G joins a vertex in X to a vertex in Y, then G is said to be a *bipartite graph*, and the partition $V(G) = X \cup Y$ is called a *bipartition* of G. For example, the graph in Figure 2.3 is bipartite, which is most easily seen in the rightmost drawing; a bipartition of this graph is $X = \{w_1, w_2, w_3, w_4\}$, $Y = \{w_5, w_6, w_7, w_8\}$.

A simple bipartite graph G with bipartition $V(G) = X \cup Y$ is called a *complete bipartite graph* if every vertex in X is joined to every vertex in Y. In this case, if X contains m vertices and Y contains n vertices, we denote this graph by $K_{m,n}$. The graph in Figure 2.3 is not a complete bipartite graph, since (for example) there is no edge joining w_1 and w_8. Figure 2.15 shows the complete bipartite graph $K_{3,5}$. Note that $K_{m,n}$ is always isomorphic to $K_{n,m}$.

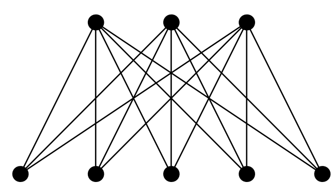

Figure 2.15: The complete bipartite graph $K_{3,5}$.

A simple graph with vertex set $\{v_1, v_2, \ldots, v_n\}$, where v_i is joined to v_j if and only if $|i - j| = 1$, is called a *path graph*. The path graph with n vertices is denoted P_n. Figure 2.16 shows P_5. It is clear that P_1 is the singleton graph, and that P_2 is the same as the complete graph K_2 and the complete bipartite graph $K_{1,1}$.

Figure 2.16: The path graph P_5.

A simple graph with vertex set $\{v_1, v_2, \ldots, v_n\}$, where v_i is joined to v_j if and only if $i - j \equiv \pm 1 \pmod{n}$, is called a *cycle graph*. The cycle graph with n vertices is denoted C_n. Figure 2.17 illustrates C_4, C_6, and C_9. Note that by this definition C_1 is the singleton graph and C_2 is the path graph P_2. Additionally, C_3 is the same as the complete graph K_3.

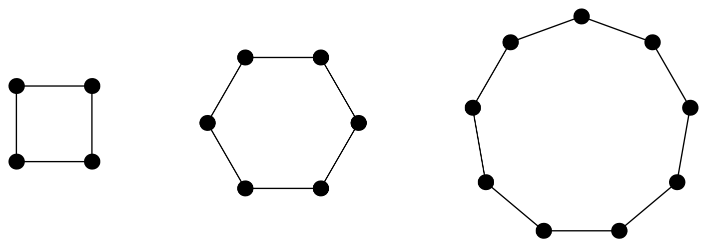

Figure 2.17: The cycle graphs C_4, C_6, and C_9.

The complete bipartite graph $K_{1,n-1}$ is called the *star graph* with n vertices and is denoted S_n. This name comes from the fact that such a graph is often drawn with one vertex in the center, surrounded by the other vertices, giving the appearance of a star. Figure 2.18 shows S_4, S_7, and S_{12}. By this definition S_2 and S_3 are the path graphs P_2 and P_3, respectively.

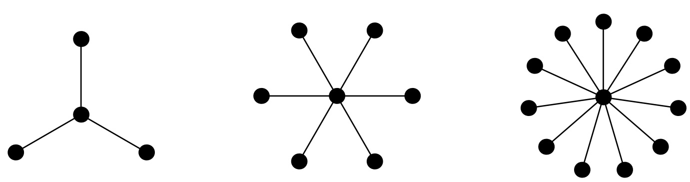

Figure 2.18: The star graphs S_4, S_7, and S_{12}.

The *wheel graph* with n vertices, denoted W_n, can be formed from the cycle graph C_{n-1} by adding another vertex and joining this new vertex with every vertex in the original cycle graph; so W_n is the join of C_n and the singleton graph. By this definition, W_2 is the path graph P_2, W_3 is the cycle graph C_3, and W_4 is the complete graph K_4. For completeness, we shall take the wheel graph W_1 to be the singleton graph, as it is the only simple graph with exactly one vertex. Figure 2.19 illustrates W_4, W_5, and W_8.

The vertices contributed by the cycle graph are called the *outer vertices*,

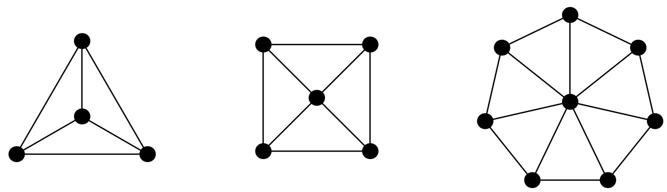

Figure 2.19: The wheel graphs W_4, W_5, and W_8.

while the vertex contributed by the singleton graph is called the *central vertex*. The edges joining the central vertex to each of the outer vertices are called the *spokes* of the wheel graph.

The *grid graph* $G_{m,n}$ is the Cartesian product $P_m \,\square\, P_n$ of a path graph with m vertices and a path graph with n vertices. The grid graph $G_{4,6}$ is presented in Figure 2.20. Clearly path graphs are a special case of grid graphs; the path graph P_n is simply the grid graph $G_{1,n}$. Another special case is the class of *ladder graphs*. The ladder graph L_n is defined to be the grid graph $G_{2,n}$. Figure 2.21 depicts the ladder graphs L_1, L_2, and L_5.

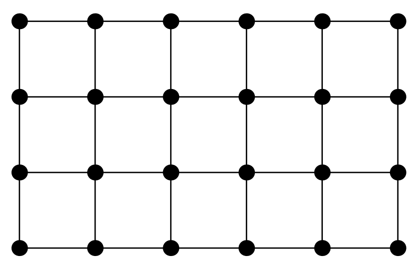

Figure 2.20: The grid graph $G_{4,6}$.

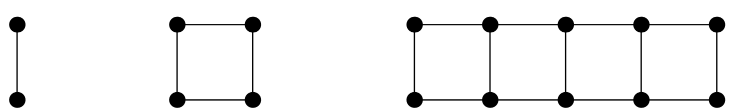

Figure 2.21: The ladder graphs L_1, L_2, and L_5.

Chapter 3

Past work

In this chapter we shall explore some previous work in the area of graph compositions. Knopfmacher and Mays focused on finding formulas for the number of compositions of various families of graphs, such as the wheel graphs and the complete bipartite graphs [13]. These results were extended by Ridley and Mays in [19], where they considered the unions of two graphs.

3.1 Path graphs and complete graphs

The term *composition* was introduced by Knopfmacher and Mays in [13] and was motivated by the well-known combinatorial concept of a *composition* of a positive integer n, which is a representation of n as an ordered sum of positive integers [5]. (Some authors allow zeroes to appear in this sum [22]; the definition we use here may unambiguously be called a *weak composition* [1].) For example, several compositions of 7 are $1+3+3$, $2+1+2+1+1$, $6+1$, and 7. (Note that $6+1$ and $1+6$ are distinct compositions of 7.) The relationship between compositions of graphs and compositions of positive integers becomes apparent when we consider the path graph P_n.

It is easy to see that any subset of the edges of P_n yields a distinct composition. Since there are $n-1$ edges in P_n, there are 2^{n-1} subsets of these edges, and consequently we have the following theorem.

Theorem 1. $C(P_n) = 2^{n-1}$.

Knopfmacher and Mays arrive at this result by noting that any connected component of a composition of P_n must itself be a path graph, and that the sum of the number of vertices in the components must of course be n. Therefore, a composition of P_n yields a composition of n, and vice versa.

The number of compositions of n is 2^{n-1} [1, Corollary 5.2]. This gives Theorem 1.

If we consider the complete graph K_n, we see that any partition of the vertices yields a composition; the subgraph induced by each part must necessarily be connected, since there is an edge between any two vertices in K_n. Therefore we see that the number of compositions of K_n is equal to the number of ways to partition n elements into nonempty subsets. This is exactly the definition of the *Bell number* B_n, which gives the following result.

Theorem 2. $C(K_n) = B_n$.

The series of Bell numbers $B_1, B_2, B_3, \ldots$ begins 1, 2, 5, 15, 52, 203, 877, 4140, 21 147, 115 975, ... Much is known about this sequence; for further information, see [2] and [3]. In 1977 Gould compiled a bibliography of some 178 references on the Bell numbers [9].

The importance of the path graph P_n and the complete graph K_n lies in the following two theorems.

Theorem 3. *If G is a connected graph with n vertices, then it has at least $C(P_n) = 2^{n-1}$ compositions.*

We shall give a proof of Theorem 3 on page 25, after we have proved Theorem 7.

Theorem 4. *If G is a connected graph with n vertices, then it has at most $C(K_n) = B_n$ compositions.*

Proof. Any composition of G is a distinct partition of the vertex set of G. But there are only n elements in this set, and hence B_n distinct partitions. Thus there can be at most B_n compositions of G. □

Of course, this lower bound does not apply to disconnected graphs; the empty graph with n vertices has exactly one composition.

3.2 Counting principles for compositions

In order to determine the number of compositions of a graph, it is necessary to know more than the number of vertices and edges in the graph. Knopfmacher and Mays give the example shown in Figure 3.1, in which both G_1 and G_2 have four vertices and four edges. It can be seen that $C(G_1) = 10$ but $C(G_2) = 12$.

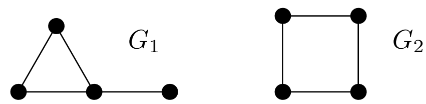

Figure 3.1: Two graphs with the same number of vertices and edges but different numbers of compositions.

A basic theorem gives the number of compositions of a disconnected graph in terms of the number of compositions of each of its connected components. (It also provides a useful result in the case when a graph has a cut vertex.)

Theorem 5. *Let G be a graph that is the union of two subgraphs G_1 and G_2, that is, $G = G_1 \cup G_2$. If there are no edges joining a vertex in $G_1 \setminus G_2$ to a vertex in $G_2 \setminus G_1$, and G_1 and G_2 have at most one vertex in common, then $C(G) = C(G_1) \cdot C(G_2)$.*

Proof. This is an application of the multiplication principle [2, Section 3.1]; we form compositions of G by pairing compositions of G_1 with compositions of G_2 in all possible ways. □

A similarly useful result pertains to graphs with a bridge.

Theorem 6. *Let G be a graph, and suppose that the vertices of G can be partitioned into two nonempty, disjoint subsets V_1 and V_2 such that there exists exactly one edge e that joins a vertex in V_1 to a vertex in V_2. Let G_1 be the subgraph of G induced by V_1, and let G_2 be the subgraph of G induced by V_2. Then $C(G) = 2 \cdot C(G_1) \cdot C(G_2)$.*

Proof. Let E be the subgraph of G consisting of the edge e with its two endpoints. First we see that G_1 and E have exactly one vertex in common, and so by Theorem 5 we have that $C(G_1 \cup E) = C(G_1) \cdot C(E)$. But clearly E is the path graph P_2, so $C(E) = 2$, and hence $C(G_1 \cup E) = 2 \cdot C(G_1)$. Now we note that $G = G_1 \cup E \cup G_2$ and that $G_1 \cup E$ and G_2 have exactly one vertex in common. Accordingly we apply Theorem 5 again, and find that $C(G) = C(G_1 \cup E \cup G_2) = 2 \cdot C(G_1) \cdot C(G_2)$. □

The proof given by Knopfmacher and Mays uses the multiplication principle more directly. Let u be the endpoint of e in V_1, and let v be the other endpoint of e (so $v \in V_2$). We see that e is a bridge, since u and v are in the same connected component in G, but in different connected components in $G - e$. For any composition of G_1 and any composition of G_2, there are exactly two compositions of G: one in which e is included, combining the component containing u in G_1 and the component containing v in G_2, and one in which e is not included. This gives $2 \cdot C(G_1) \cdot C(G_2)$ compositions of G.

Note that if a graph has a bridge, then the vertices of the graph can be partitioned in the manner described in Theorem 6; this is true even if the graph is disconnected.

3.3 Trees and almost complete graphs

As a consequence of Theorem 6 and the fact that every edge in a tree is a bridge [6, Theorem 2.8], we have the following theorem.

Theorem 7. *Let T_n be any tree with n vertices. Then $C(T_n) = 2^{n-1}$.*

Proof. The proof is by induction on n. Plainly T_1 is the singleton graph, which has exactly one composition, and $1 = 2^0$. Now suppose that this result is true for all n less than some integer k, where $k \geq 2$. Fix any edge e in T_k; this edge is a bridge, and so its deletion produces two connected components T_1 and T_2, each of which must be a tree. Suppose that T_1 has m vertices; then T_2 has $k - m$ vertices. Since $m \geq 1$ and $k - m \geq 1$, we have that $m < k$ and $k - m < k$, so by our inductive hypothesis $C(T_1) = 2^{m-1}$ and $C(T_2) = 2^{k-m-1}$. Because e is a bridge, Theorem 6 implies that $C(T) = 2 \cdot C(T_1) \cdot C(T_2) = 2 \cdot 2^{m-1} \cdot 2^{k-m-1} = 2^{k-1}$. □

The path graph P_n is a tree with n vertices, so Theorem 1 is a special case of Theorem 7. Similarly, the star graph S_n is a tree with n vertices, so $C(S_n) = 2^{n-1}$.

Let T_n be a tree with n vertices, as in Theorem 7. Then T_n has 2^{n-1} compositions, by Theorem 7; moreover it has exactly $n-1$ edges [6, Theorem 2.4]. Consider the *power set* of $E(T_n)$, that is, the set of all subsets of $E(T_n)$. There are 2^{n-1} such subsets. Each composition of T_n can be mapped to one of these subsets, namely, the set of edges which belong to T_n, and plainly this mapping must be injective (one-to-one). But there are 2^{n-1} compositions of T_n and the same number of subsets of $E(T_n)$, so this mapping must also be surjective (onto). Consequently we see that the power set of the edge set of a tree yields in a natural way the set of compositions of the tree.

Using Theorem 7, we can now give a proof of Theorem 3, which states that a connected graph G with n vertices has at least $C(P_n) = 2^{n-1}$ compositions.

Proof of Theorem 3. Since G is connected, we can construct a *spanning tree* of G, which is a subgraph of G that contains all vertices of G and is a tree [6, Theorem 2.12]. This spanning tree has n vertices, and so by Theorem 7 it has 2^{n-1} compositions; that is, there exist 2^{n-1} partitions of the vertex set of the spanning tree which are compositions. But each of these compositions is of course a partition of the vertex set of G, and so we have at least 2^{n-1} compositions of G. □

Knopfmacher and Mays also consider the case of a complete graph with one edge deleted, and arrive at the following result.

Theorem 8. *Let K_n^- denote the complete graph on n vertices with one edge deleted. Then $C(K_n^-) = B_n - B_{n-2}$.*

Proof. Let e be the deleted edge, and let u and v be the endpoints of e. The only compositions of K_n which are not compositions of K_n^- are those in which one component consists only of u and v; in all other compositions of K_n in which u and v are in the same component, there exists a third vertex w in that component, and the path from u to w to v still exists in K_n^-. Therefore, to count the number of compositions of K_n^- we must subtract from the number of compositions of K_n all those compositions in which one component is $\{u, v\}$. By Theorem 2, there are B_n compositions of K_n. The vertices other than u and v form K_{n-2}, of which there are B_{n-2} compositions. Hence, $C(K_n^-) = B_n - B_{n-2}$. □

The analysis becomes more complicated when more than one edge is deleted from a complete graph; the number of compositions of the resulting graph depends on whether the deleted edges are adjacent or not. The number of compositions of the complete graph K_n with two edges deleted is shown in Table 3.1 for several values of n.

n	3	4	5	6	7	8
deleted edges adjacent	2	10	40	168	758	3682
deleted edges not adjacent	—	12	43	175	778	3749

Table 3.1: Number of compositions of the complete graph K_n with two edges deleted.

3.4 Cycle graphs

The number of compositions of the cycle graph C_n can also be expressed with a simple formula, given in Theorem 9.

Theorem 9. $C(C_n) = 2^n - n$.

Proof. Choose any edge e and delete it. This produces the path graph P_n. By Theorem 1, there are 2^{n-1} compositions of P_n. Each of these compositions is also a composition of C_n. Moreover, for each of the compositions of P_n, another composition of C_n can be obtained by reinserting e, unless the composition of P_n was formed by deleting no edges, or by deleting exactly one edge. In these cases, reinserting e results in the composition consisting of only one part, containing all n vertices. This composition is induced by $n+1$ different subsets of the edges: the entire edge set, and each of the n subsets containing all but one edge. Therefore, there are $2 \cdot 2^{n-1} - n = 2^n - n$ compositions of C_n. $\square$

Knopfmacher and Mays give in [13] a very brief sketch of a second proof of Theorem 9 which uses the concept of Lyndon compositions. A fuller exposition of this proof follows.

A *Lyndon composition* of the positive integer n is an aperiodic composition that is lexicographically least among its cyclic permutations [13]. This means that $4+2+4+2$, $1+3+1+3+1+3$, and $3+1+4+2+2$ are not Lyndon compositions of 12, since the first two are periodic, and the last is not lexicographically least among its cyclic permutations (in fact, neither is the first). The cyclic permutations of the last composition are

$$\begin{aligned}&3+1+4+2+2,\\&1+4+2+2+3,\\&4+2+2+3+1,\\&2+2+3+1+4, \text{ and}\\&2+3+1+4+2,\end{aligned}$$

which arranged in lexicographic order are

$$\begin{aligned}&1+4+2+2+3,\\&2+2+3+1+4,\\&2+3+1+4+2,\\&3+1+4+2+2,\\&4+2+2+3+1.\end{aligned}$$

Therefore $1+4+2+2+3$ is a Lyndon composition of 12, since it is aperiodic and lexicographically least among its cyclic permutations.

Knopfmacher and Mays give, without proof, the number $L(n)$ of Lyndon compositions of the integer n as

$$L(n) = \frac{1}{n} \sum_{d|n} \mu\left(\frac{n}{d}\right) 2^d. \tag{3.1}$$

They also note that by this formula $L(1) = 2$.

First, it is clear that the value $L(1) = 2$ is incorrect, since there is but a single composition of 1 (namely, $1 = 1$). With the exception of this discrepancy, however, the above formula does give the correct number of Lyndon compositions of n. We shall now examine why this is so.

The concept of Lyndon compositions is related to the concept of *Lyndon words*, which are aperiodic strings that are lexicographically least among their cyclic permutations [15]. This means that, for instance, the strings `0101` and `101101101` are not Lyndon words, since they are periodic; and the string `10010` is not a Lyndon word, despite being aperiodic, because it is not lexicographically least among its cyclic permutations (however, its least cyclic permutation, `00101`, is a Lyndon word).

We may consider any binary string S of length n to be "built" from a unique Lyndon word, in the following sense: If S is aperiodic, then its corresponding Lyndon word is simply its lexicographically least cyclic permutation. Otherwise, S is periodic, and its corresponding Lyndon word is the lexicographically least cyclic permutation of the shortest repeating binary substring in S.

To give an example, the string `11011101` is periodic; its smallest repeating substring is `1101`. The lexicographically least cyclic permutation of this substring is `0111`, which is the Lyndon word from which the original string is "built."

A Lyndon word W of length d can be used to "build" a binary string S of length n only if d divides n, since the Lyndon word must repeat an integral number of times in S. Suppose that d does divide n. There are d distinct cyclic permutations of W, each of which, when repeated n/d times, will yield a distinct binary string of length n.

Hence, the number of binary strings of length n is

$$\sum_{d|n} d\ell(d),$$

where $\ell(d)$ is the number of Lyndon words of length d. Since the number of binary strings of length n is 2^n, we see that

$$2^n = \sum_{d|n} d\ell(d).$$

By the Möbius inversion formula [3, Section 12.7], we obtain

$$\ell(n) = \frac{1}{n} \sum_{d|n} \mu \left(\frac{n}{d} \right) 2^d. \tag{3.2}$$

Interestingly, Equation 3.2 is identical to Equation 3.1. Therefore, it is claimed that there is a one-to-one correspondence between Lyndon words of length n and Lyndon compositions of n (except when $n = 1$, as noted above). Finding such a correspondence will prove Equation 3.1.

Consider a row of n circles. A composition of n can be represented by separating this row of circles into one or more pieces. For example, the composition $11 = 3 + 1 + 5 + 2$ can be represented as shown in Figure 3.2.

Figure 3.2: The composition $11 = 3 + 1 + 5 + 2$.

Now consider the binary string `00110000101`, which is of length 11. We can express this string with circles, with ○ representing `0` and ● representing `1`, as shown in Figure 3.3. If we separate this row of circles after each `1` (that is, after each ●), we obtain the composition $11 = 3+1+5+2$, as seen in Figure 3.4.

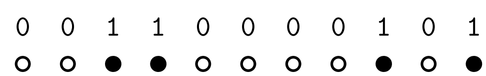

Figure 3.3: The binary string `00110000101`.

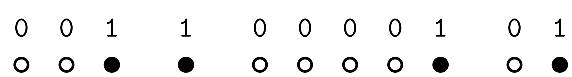

Figure 3.4: The correspondence between `00110000101` and $11 = 3+1+5+2$.

Note that the last digit of the binary string does not matter; the same composition could just as well have been represented by `0011000100`, as seen in Figure 3.5. Thus, for $n \geq 1$, each each composition of n corresponds

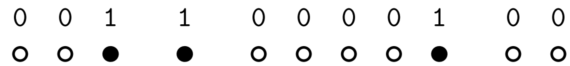

Figure 3.5: The string `00110000100` also yields $11 = 3 + 1 + 5 + 2$.

to exactly two binary strings of length n. This "one-to-two" correspondence occurs because the last digit of the binary string is irrelevant.

When we consider Lyndon words, the situation is different. With the sole exception of the Lyndon word `0`, the last digit of every Lyndon word is `1`, for otherwise the string could be cyclically permuted to rotate the final `0` to the beginning, creating a lexicographically lesser string (because all Lyndon words other than `0` must contain the digit `1`). This fact is convenient for us, since we can effectively ignore the last digit of Lyndon words when $n \geq 2$.

Since Lyndon words are aperiodic, their corresponding compositions are also aperiodic. Moreover, since Lyndon words are lexicographically least among their cyclic permutations, their corresponding compositions will be lexicographically *greatest*. This is because a Lyndon word will begin with as many `0`s as possible, thus placing the largest part of the composition first. So we see that the compositions corresponding to Lyndon words are exactly those aperiodic compositions that are lexicographically greatest among their cyclic permutations. Each of these greatest compositions will correspond to exactly one Lyndon composition. Therefore, for $n \geq 2$ there is a one-to-one correspondence between Lyndon words of length n and Lyndon compositions of n, thus providing a justification for Equation 3.1.

In the case where $n = 1$, we have two Lyndon words (`0` and `1`), but only one Lyndon composition ($1 = 1$). This is the reason that Equation 3.1 yields $L(1) = 2$ instead of $L(1) = 1$.

Now that we have established the validity of Equation 3.1, let us see how this can be used to prove Theorem 9.

Let d be a divisor of n. To make a composition of the cycle graph C_n, we may delete any n/d edges spaced regularly around C_n, and, in each of the n/d segments of C_n thus produced, delete edges so as to create a Lyndon composition of d, with the same Lyndon composition in each segment. (Note that there are d distinct ways in which one may delete n/d regularly spaced edges of C_n.) Moreover, every such composition of C_n can be produced in this way. The composition consisting solely of one part can be made by setting $d = n$ and using the Lyndon composition $d = d$. All other compositions must have more than one part. If such a composition is aperiodic, then it can be made by setting $d = n$ and using the appropriate Lyndon composition of d. If instead it is periodic, let k be the number of times the smallest repeating unit repeats (called the *primitive period* of the composition [14]), set $d = n/k$, and choose the appropriate Lyndon composition of n/k. One minor problem remains, however, which is that the composition consisting of a single part is counted n times (once for the deletion of each edge of the graph, when $d = n$). So we must subtract $n - 1$ from our final sum.

Let $\underline{L}(d)$ denote the number of Lyndon compositions of d; so $\underline{L}(d) = L(d)$ for all d, except that $\underline{L}(1) = 1$ but $L(1) = 2$. Then by the above argument we see that

$$C(C_n) = \sum_{d|n} d\underline{L}(d) - n + 1.$$

Notice that because $L(1) = \underline{L}(1) + 1$, but $L(d) = \underline{L}(d)$ for all other values of d, we may instead write

$$C(C_n) = \sum_{d|n} dL(d) - n,$$

which is indeed the formula given in [13]. By the Möbius inversion of Equation 3.1 we obtain

$$2^n = \sum_{d|n} dL(d),$$

so $C(C_n) = 2^n - n$, which is Theorem 9.

The sequence $L(1)$, $L(2)$, $L(3)$, $\ldots$, as given by Equation 3.1, begins 2, 1, 2, 3, 6, 9, 18, 30, 56, $\ldots$.

3.5 Wheel graphs

If we take the wheel graph W_1 to be the singleton graph, W_2 to be the path graph P_2, and W_3 to be the cycle graph C_3, then the sequence $\{C(W_n)\}$ begins

1, 2, 5, 15, 43, 118, 316, 836, 2199, 5769, 15 117, 39 592, 103 670, $\ldots$.

Lyndon compositions also arise in a formula for the number of compositions of the wheel graph W_n. Here they are used as an index of summation. Note that in this formula $C(P_0)$ is taken to be 1.

Theorem 10. *For $n > 1$,*

$$C(W_n) = 2^{n-1} - n + 2 + \sum_{1<d|n-1} d \sum_{a_1+\cdots+a_k=d}' \prod_{i=1}^{k} C(P_{a_i-1})^{(n-1)/d},$$

where $\sum'$ indicates a sum over Lyndon compositions of d.

Proof. We consider two cases. In the first case, the central vertex is together with no outer vertices. These compositions of the wheel graph correspond to compositions of the cycle graph C_{n-1}, which by Theorem 9 has $2^{n-1}-(n-1)$ compositions.

In the second case, the central vertex is together with one or more outer vertices. Consider the spokes that belong to such a composition. They separate the "gaps" between the outer vertices into one or more clusters. For example, if two adjacent spokes belong to the composition, then the gap between them is isolated from the other gaps. Since there are $n-1$ gaps in total, there may be a single cluster of gaps (if only one spoke belongs to the composition), or as many as $n-1$ clusters (if every spoke belongs to the composition).

The arrangement of these gaps may be periodic (for example, if every second spoke belongs to the composition) or aperiodic. In either case, however, the arrangement corresponds to some Lyndon composition, possibly repeated an integral number of times. Since there are $n-1$ gaps in all, this must be a Lyndon composition of some integer d which divides $n-1$. This is the reason for the two summations: the first is a summation over all possible divisors of $n-1$, and the second is a summation over all Lyndon compositions of each divisor. There are d distinct cyclic permutations of this Lyndon composition, which will yield d distinct separations of the gaps; this gives the factor of d.

For each cyclic permutation of this Lyndon composition, which must be repeated $(n-1)/d$ times to go all the way around the outside of the wheel, the clusters of gaps contain within them clusters of outer vertices that are not together with the central vertex. If there are a_i gaps in a cluster, then that cluster contains $a_i - 1$ such outer vertices. There are $C(P_{a_i-1})$ compositions of these outer vertices, and each cluster of vertices may be partitioned into a composition independently of the other clusters. This produces the product over the lengths a_i of the clusters. The exponent of $(n-1)/d$ exists because of the repetition of the Lyndon composition; the cluster of length a_i will reappear $(n-1)/d$ times around the wheel.

Finally, note that if $d = 1$, then there is but one Lyndon composition (namely, $1 = 1$). Since $C(P_0) = 1$ we get only one composition of the wheel graph in this case (this is the composition in which every outer vertex is together with the central vertex). Therefore, we restrict d to be strictly greater than 1 in the first summation, and simply add 1 explicitly to the formula. This is why the formula begins $2^{n-1} - n + 2 + \cdots$ rather than $2^{n-1} - n + 1 + \cdots$. This is also the reason that the formula does not work for $n = 1$, since in the case of W_1 the composition where the central vertex is not together with any outer vertex is the same as the composition where the central vertex is together with all outer vertices. $\square$

The value of $C(W_n)$ is much easier to obtain using a recurrence relation noted by Knopfmacher and Mays, which is

$$\begin{aligned} &C(W_1) = 2, \\ &C(W_2) = 2, \\ &C(W_n) = 3 \cdot C(W_{n-1}) - C(W_{n-2}) + n - 2 \qquad \text{for } n \geq 3. \end{aligned}$$

Of course, this value for $C(W_1)$ is incorrect, for the reason noted in the proof.

In [19], Ridley and Mays show that, for $n \geq 4$,

$$C(W_n) = F_{2n-1} + F_{2n-3} - n + 1 = L_{2n-2} - n + 1,$$

where F_k denotes the kth Fibonacci number and L_k denotes the kth Lucas number.

3.6 Ladder graphs

The ladder graph L_1 is simply the path graph P_2, so $C(L_1) = 2$ by Theorem 1. The ladder graph L_2 is the cycle graph C_4, so $C(L_2) = 12$ by Theorem 9. Further values of $C(L_n)$ can be described by a recurrence relation.

Theorem 11. *The number of compositions of the ladder graph L_n satisfies the recurrence relation*

$$\begin{aligned} &C(L_1) = 2, \\ &C(L_2) = 12, \\ &C(L_n) = 6 \cdot C(L_{n-1}) + C(L_{n-2}) \qquad \textit{for } n \geq 3. \end{aligned}$$

Proof. Denote the vertices of the ladder graph L_k by $v_{1,1}$, $v_{1,2}$, $v_{2,1}$, $v_{2,2}$, $\ldots$, $v_{k,1}$, $v_{k,2}$. We shall call $v_{k,1}$ and $v_{k,2}$ the *tail vertices* of L_k. If the tail

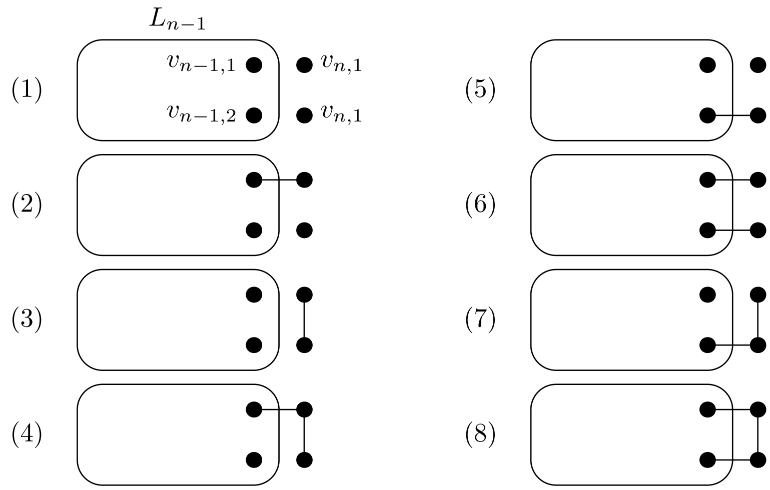

Figure 3.6: The eight cases considered in the proof of Theorem 11.

vertices are together in a composition of a ladder graph, we shall say that the composition is *united*; otherwise, we shall say that the composition is *divided*. Let D_k represent the number of divided compositions of L_k, and let U_k represent the number of united compositions.

To make L_n from L_{n-1}, we add two new tail vertices $v_{n,1}$ and $v_{n,2}$, and three new edges which join $v_{n,1}$ to $v_{n-1,1}$, $v_{n,2}$ to $v_{n-1,2}$, and $v_{n,1}$ to $v_{n,2}$. Any combination of these three edges can belong to a composition of L_n. This results in eight cases to consider, as seen in Figure 3.6.

Suppose we wish to make a divided composition of L_n. For any divided composition of L_{n-1} in which the tail vertices are not together, cases (1), (2), (5), and (6) will result in a divided composition of L_n. If we start with a united composition of L_{n-1}, then cases (1), (2), and (5) will yield a divided composition of L_n. Therefore $D_n = 4 \cdot D_{n-1} + 3 \cdot U_{n-1}$.

On the other hand, suppose we wish to construct a united composition of L_n. From a divided composition of L_{n-1}, cases (3), (4), and (7) yield distinct united compositions of L_n; case (8) would make the composition of L_n united. Starting with a united composition of L_{n-1}, we can make a united composition of L_n via case (3) or case (8); cases (4) and (7) yield the same composition as case (8). Hence $U_n = 3 \cdot D_{n-1} + 2 \cdot U_{n-1}$.

Thus, we have

$$C(L_n) = D_n + U_n = 7 \cdot D_{n-1} + 5 \cdot U_{n-1}.$$

But we also have

$$D_{n-1} - U_{n-1} = D_{n-2} + U_{n-2} = C(L_{n-2}).$$

Consequently,

$$C(L_n) = 6(D_{n-1} + U_{n-1}) + (D_{n-1} - U_{n-1}) = 6 \cdot C(L_{n-1}) + C(L_{n-2}).$$

$\square$

Knopfmacher and Mays observe that this recurrence relation is the same as that for the denominators in the continued fraction expansion of $\sqrt{10}$.

3.7 Unions of graphs

A general result for the number of compositions of a graph expressed as a union of two subgraphs is given by Ridley and Mays in [19]. They consider a graph expressed as a union of edge-disjoint graphs, but in fact their results hold even if the subgraphs have one or more edges in common.

Let G be a graph expressed as a union $G = A \cup B$, and let $\sim_{\mathcal{C}}$ be the equivalence relation defined by some composition $\mathcal{C}$ of G. If we restrict the application of $\sim_{\mathcal{C}}$ to the vertices of A, the equivalence classes so produced do not necessarily make a composition of A, since one or more of them may not be connected when we use edges from A only. However, we do obtain a unique composition $\mathcal{C}_A$ of A defined by all edges in A with endpoints u and v such that $u \sim_{\mathcal{C}} v$, that is, all edges in A that belong to $\mathcal{C}$. We obtain a similar composition $\mathcal{C}_B$ for B. We shall say that $\mathcal{C}$ *induces* the pair $(\mathcal{C}_A, \mathcal{C}_B)$. (Ridley and Mays say that $\mathcal{C}$ is *valid* for $\mathcal{C}_A$ and $\mathcal{C}_B$ if $\mathcal{C}$ induces $\mathcal{C}_A$ and $\mathcal{C}_B$.)

The procedure for counting distinct compositions of G, then, is to consider the pairing of each composition $\mathcal{C}_A$ of A with each composition $\mathcal{C}_B$ of B, and to count the resulting composition of G only if it induces $\mathcal{C}_A$ and $\mathcal{C}_B$. Since each composition of G induces exactly one pair $(\mathcal{C}_A, \mathcal{C}_B)$, this procedure will count each composition of G exactly once.

For a given composition $\mathcal{C}$ of a graph G, we shall classify each pair $\{u, v\}$ of vertices in G into one of three categories. If $u \sim v$, that is, if u and v are in the same component of $\mathcal{C}$, then we shall say that u and v are *together* in $\mathcal{C}$. If u and v are not together in $\mathcal{C}$, but there exist vertices x and y adjacent in G such that u and x are together and v and y are together, then we shall say that u and v are *nearby* in $\mathcal{C}$. If u and v are neither together nor nearby in $\mathcal{C}$, we shall say that they are *distant* in $\mathcal{C}$. Ridley and Mays use the terms *type (0)*, *type (1)*, and *type (2)* to refer to pairs of vertices that are together, nearby, and distant, respectively, when considering only the subgraph A of a graph $G = A \cup B$. Since these numbered types seem unnecessary, we shall avoid them.

Using this terminology, we can present the following two theorems.

Theorem 12. *Let G be a graph expressed as a union $G = A \cup B$, and consider a composition $\mathcal{C}$ of G that induces some pair of compositions $\mathcal{C}_A$ and $\mathcal{C}_B$ of A and B, respectively. If two vertices u and v in A are nearby in $\mathcal{C}_A$, then u and v are not together in $\mathcal{C}$.*

Proof. Since u and v are nearby in $\mathcal{C}_A$, there exist vertices x and y in A, joined by an edge e in A, such that u and x are together in $\mathcal{C}_A$ and v and y are together in $\mathcal{C}_A$. Clearly then each of the pairs $\{u, x\}$ and $\{v, y\}$ is together in $\mathcal{C}$. Suppose now that u and v are together in $\mathcal{C}$. Since togetherness is an equivalence relation, it is transitive, and so we see that x and y are together in $\mathcal{C}$. We are given that $\mathcal{C}$ induces $\mathcal{C}_A$ and $\mathcal{C}_B$, so e belongs to $\mathcal{C}_A$; in other words, x and y are together in $\mathcal{C}_A$. Hence u and v are together in $\mathcal{C}_A$, which is a contradiction. $\square$

Theorem 13. *Let G be a graph expressed as a union $G = A \cup B$, and consider a composition $\mathcal{C}$ of G that induces some pair of compositions $\mathcal{C}_A$ and $\mathcal{C}_B$ of A and B, respectively. If no two vertices in A are distant in $\mathcal{C}_A$, then the restriction of $\sim_{\mathcal{C}}$ to $V(A)$ is $\sim_{\mathcal{C}_A}$.*

Proof. If the restriction of $\sim_{\mathcal{C}}$ to $V(A)$ is not $\sim_{\mathcal{C}_A}$, then there exists some pair $\{u, v\}$ of vertices in A that are together in $\mathcal{C}$ but not in $\mathcal{C}_A$. Since they are together in $\mathcal{C}$, Theorem 12 implies that they cannot be nearby in $\mathcal{C}_A$. Hence u and v must be distant in $\mathcal{C}_A$, and thus we have proved the contrapositive. $\square$

Ridley and Mays point out that if two vertices u and v in $A \cap B$ are distant in $\mathcal{C}_A$ but together in $\mathcal{C}_B$, then it is still possible that $\mathcal{C}$ induces $\mathcal{C}_A$ and $\mathcal{C}_B$, even though the restriction of $\sim_{\mathcal{C}}$ to A is not $\sim_{\mathcal{C}_A}$. In this case, u and v are together in $\mathcal{C}$, which in general affects the togetherness or nearbyness in $\mathcal{C}$ of other pairs of vertices. Therefore, whether $\mathcal{C}$ induces $\mathcal{C}_A$ and $\mathcal{C}_B$ cannot be determined without examining all pairs of vertices that are distant in $\mathcal{C}_A$.

To place an upper bound on the number of cases one must examine, consider the subgraph H of G induced by the vertices in $A \cap B$. Suppose that H has p vertices and has q edges that are in $E(A)$. Then the number of pairs of vertices in H that are not adjacent in A is $q' = \frac{1}{2}p(p-1) - q$. Each of the q pairs of vertices adjacent in A can be either together or nearby, but cannot be distant. Each of the q' pairs of vertices not adjacent in A can be together, nearby, or distant. Therefore there are at most $2^q 3^{q'}$ cases to be considered. As Ridley and Mays observe, this is not a good bound, since the transitivity of $\sim_{\mathcal{C}_A}$ means that in general the categories of the different pairs of vertices cannot all be independent.

Our general procedure for counting compositions of $G = A \cup B$, which is to consider all pairs $(\mathcal{C}_A, \mathcal{C}_B)$ of compositions of A and B and to count the corresponding composition of G only if it induces $\mathcal{C}_A$ and $\mathcal{C}_B$, is summarized in the following theorem.

Theorem 14. *Let $C_0, C_1, \ldots, C_{k-1}$ denote the numbers of compositions $\mathcal{C}_A$ of A in the k cases determined by the possible categories of pairs of vertices in $A \cap B$, and for each case $r = 0, 1, \ldots, k-1$ let M_r denote the number of compositions $\mathcal{C}_B$ of B such that the resulting composition $\mathcal{C}$ of G induces $\mathcal{C}_A$ and $\mathcal{C}_B$. Then $C(G) = M_0C_0 + M_1C_1 + \cdots + M_{k-1}C_{k-1}$.*

It is useful to identify some conditions which are sufficient to show that a composition induces two others. Suppose as before that $G = A \cup B$. From the definition of induced compositions, it is clear that a composition $\mathcal{C}$ of G induces compositions $\mathcal{C}_A$ and $\mathcal{C}_B$ of A and B, respectively, if and only if all edges in $E(A)$ that belong to $\mathcal{C}$ also belong to $\mathcal{C}_A$, and all edges in $E(B)$ that belong to $\mathcal{C}$ also belong to $\mathcal{C}_B$. The following theorems are not given by Ridley and Mays, but they follow easily.

Theorem 15. *Let G be a graph expressed as a union $G = A \cup B$. Let $\mathcal{C}_A$ and $\mathcal{C}_B$ be compositions of A and B, respectively, and let $\mathcal{C}$ be the resulting composition of G. If the restriction of $\sim_{\mathcal{C}}$ to $V(A)$ is $\sim_{\mathcal{C}_A}$, and the restriction of $\sim_{\mathcal{C}}$ to $V(B)$ is $\sim_{\mathcal{C}_B}$, then $\mathcal{C}$ induces $\mathcal{C}_A$ and $\mathcal{C}_B$.*

Proof. Suppose that $\mathcal{C}$ does not induce $\mathcal{C}_A$ and $\mathcal{C}_B$. Then there exists an edge in $E(A)$ that belongs to $\mathcal{C}$ but does not belong to $\mathcal{C}_A$, or there exists an edge in $E(B)$ that belongs to $\mathcal{C}$ but does not belong to $\mathcal{C}_B$. If the first case holds, then the endpoints of this edge are not together in $\mathcal{C}_A$, but are together in $\mathcal{C}$; so the restriction of $\sim_{\mathcal{C}}$ to $V(A)$ is not $\sim_{\mathcal{C}_A}$. Likewise, if the second case holds, then the restriction of $\sim_{\mathcal{C}}$ to $V(B)$ is not $\sim_{\mathcal{C}_B}$. This proves the contrapositive. □

Theorem 16. *Let G be a graph expressed as a union $G = A \cup B$. Let $\mathcal{C}_A$ and $\mathcal{C}_B$ be compositions of A and B, respectively, and let $\mathcal{C}$ be the resulting composition of G. If no two vertices in $A \cap B$ are nearby in $\mathcal{C}_A$, and no two such vertices are nearby in $\mathcal{C}_B$, then $\mathcal{C}$ induces $\mathcal{C}_A$ and $\mathcal{C}_B$.*

Proof. As in the proof of Theorem 15, we shall prove the contrapositive. Suppose that $\mathcal{C}$ does not induce $\mathcal{C}_A$ and $\mathcal{C}_B$. Then there exists an edge in $E(A)$ that belongs to $\mathcal{C}$ but does not belong to $\mathcal{C}_A$, or there exists an edge in $E(B)$ that belongs to $\mathcal{C}$ but does not belong to $\mathcal{C}_B$. Call the endpoints of this edge u and v. In the first case, we see that u and v are not together in $\mathcal{C}_A$, and yet they are adjacent in A; thus they are nearby in $\mathcal{C}_A$. By the same argument, in the second case u and v are nearby in $\mathcal{C}_B$. $\square$

It is interesting to note that Theorem 5 applies to a particular type of union of graphs: that in which the intersection contains no vertices, or just one vertex. In this case the hypothesis of Theorem 16 is obviously true, which allows us simply to multiply numbers of compositions without needing to consider whether the pairs of compositions are induced by the resulting composition of the union.

Chapter 4

Series-parallel graphs

While Knopfmacher and Mays have explored compositions of several families of graphs in [13], and Ridley and Mays have investigated the general properties of compositions of unions of graphs in [19], there has not been work done on the particular case of series-parallel graphs. This class of graphs includes the path graphs, the cycle graphs, and the ladder graphs as special cases.

This chapter will explore several definitions of series-parallel graphs and their importance in electric circuit analysis. The next chapter will present a new result which allows the linear-time computation of the number of compositions of any series-parallel graph.

4.1 Definitions of series-parallel graphs

One definition of series-parallel graphs, which takes a "bottom-up" approach, comes from [24]. First, we define a *one-terminal-pair graph* to be a graph with two distinct vertices specially designated as the *terminals* of the graph. (In our illustrations of one-terminal-pair graphs, we shall show the terminals as unfilled circles to distinguish them from the other vertices.)

Now a *series-parallel graph* can be defined as a one-terminal-pair graph with the following recursive definition. As a starting point, the graph shown in Figure 4.1, which is seen to be the path graph P_2, is a series-parallel graph. Moreover, if G_1 and G_2 are series-parallel graphs, then all series

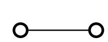

Figure 4.1: The path graph P_2 is a series-parallel graph.

combinations of G_1 and G_2 and all parallel combinations of G_1 and G_2 are series-parallel graphs. A *series combination* of one-terminal-pair graphs G_1 and G_2 is formed by identifying one of the terminals of G_1 with one of the terminals of G_2; the terminal of G_1 and the terminal of G_2 which were not identified become the terminals of the series combination. A *parallel combination* of one-terminal-pair graphs G_1 and G_2 is produced by arbitrarily labeling the terminals in each graph so that each graph has one terminal labeled A and one labeled B, and then identifying the two terminals labeled A and identifying the two terminals labeled B.

An alternate definition of series-parallel graphs, taking more of a "top-down" approach, is found in [23]. Here we define two edges of a graph to be *series* if they are incident on a vertex of degree 2, and *parallel* if they join the same pair of distinct vertices. A *series-parallel graph* is constructed by starting with two vertices joined by two parallel edges, and repeatedly replacing edges of the graph by series or parallel edges. (This replacement step may be performed zero times, in which case the resulting series-parallel graph is two copies of P_2 in parallel combination, or it may be repeated any finite number of times.)

One difference between this definition and the definition in [24] is that this definition does not allow the resulting graph to have bridges or cut vertices. Nor does it designate terminals of the resulting graph. When two terminals are designated so that the graph can be formed through series and parallel combinations as in the former definition, then it may be called a *two-terminal series-parallel graph*. It is known that every series-parallel graph in the latter sense is a two-terminal series-parallel graph for an appropriate choice of terminals [7].

Using this second definition of series-parallel graphs, it can also be shown that a graph G is series-parallel if and only if it contains no K_4 as a subcontraction, that is, the complete graph K_4 cannot be obtained from G by the repeated deletion and contraction of edges [7]. In fact, this property itself has been used in a third definition of series-parallel graphs [18].

Finally, a fourth definition, one given by Riordan and Shannon [20], says that a connected graph is series-parallel with respect to two terminals u and v if through every edge there exists a path from u to v, and no two paths from u to v pass through any edge in opposite directions.

We shall adopt the first definition of series-parallel graphs, that found in [24], so that all of our series-parallel graphs will have two distinguished terminals. This definition can be shown to be equivalent to that given by Riordan and Shannon [20]; it may be easier to use their definition when examining a graph to determine whether it is series-parallel.

For example, we claimed in Chapter 1 that the Wheatstone bridge, illustrated in Figure 1.9, is not a series-parallel graph. Using the definition given by Riordan and Shannon, we see that if this graph were series-parallel, then the terminals must be the leftmost and rightmost vertices, since every edge must be used in some path from one terminal to the other. Now it is easy to see that one path from the left terminal to the right terminal goes from top to bottom through the edge drawn vertically, while another such path goes from bottom to top. Therefore this graph is not series-parallel.

By the nature of their construction, series-parallel graphs may have parallel edges. However, in this paper we shall concern ourselves only with the underlying simple graph of series-parallel graphs, since the existence of parallel edges does not change the number of compositions of the graph. Hence, for the remainder of this paper we shall assume that series-parallel graphs have no parallel edges.

4.2 Applications to electric circuits

Series-parallel graphs are of special interest in the study of electrical networks. A *resistor* is an element of an electric circuit that impedes the flow of electric current. Resistors are shown in circuit diagrams using the symbol shown in Figure 4.2. The unit of measurement of resistance is the *ohm*, abbreviated Ω.

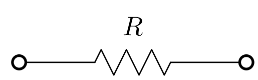

Figure 4.2: A resistor.

Resistors are often connected in series, as shown in Figure 4.3, or in parallel, as in Figure 4.4. It is well known that the equivalent resistance R_{eq} of the two resistors in Figure 4.3 is $R_{\text{eq}} = R_1 + R_2$. In other words, any two resistors connected in series, of resistances R_1 and R_2, can be replaced by a single resistor of resistance R_{eq} without affecting the operation of the circuit. In general, for resistances R_1, R_2, $\ldots$, R_k connected in series, we have

$$R_{\text{eq}} = \sum_{i=1}^{k} R_i.$$

In a similar way, two resistors connected in parallel can be replaced by an equivalent resistance. When resistors are connected in parallel, however, it is

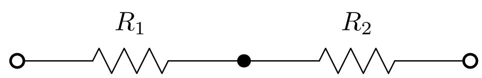

Figure 4.3: Two resistors connected in series.

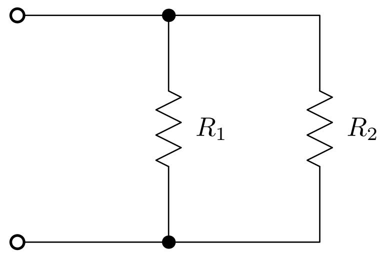

Figure 4.4: Two resistors connected in parallel.

not their resistances that add together, but their *conductances*. Conductance is the reciprocal of resistance. Therefore, for two resistances R_1 and R_2 connected in parallel, we have

$$\frac{1}{R_{\text{eq}}} = \frac{1}{R_1} + \frac{1}{R_2},$$

which yields

$$R_{\text{eq}} = \frac{R_1 R_2}{R_1 + R_2}.$$

In general, for resistances R_1, R_2, ..., R_k connected in parallel, we have

$$\frac{1}{R_{\text{eq}}} = \sum_{i=1}^{k} \frac{1}{R_i}.$$

Some other circuit elements act in a similar manner. An *inductor* acts to oppose a change in current; it consists of a coil of wire wound about a core. Equivalent inductances are computed in exactly the same way as equivalent resistances. A *capacitor* is an element used to store an electric charge. Equivalent capacitances can also be computed, but it is when capacitors are connected in parallel that capacitances add together; in series the sum of the reciprocals of the capacitances is the reciprocal of the equivalent capacitance [4, 17].

Duffin [8] uses the notation $A + B$ for two resistors A and B in series, and $A : B$ for two resistors in parallel. This notation is well chosen, since

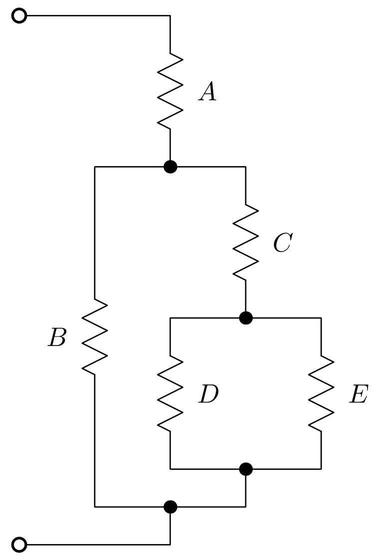

Figure 4.5: A series-parallel connection of resistors.

resistances add in the series connection. By defining

$$A : B = \frac{AB}{A + B},$$

one can write a concise description of both a resistive circuit and the equivalent resistance. Duffin gives the example $R = A + B : (C + D : E)$, which corresponds to the combination of resistors shown in Figure 4.5.

However, the numbers of compositions of graphs do not add together when the graphs are connected in series, so Duffin's notation is somewhat inappropriate in our case. For this reason we shall introduce a different notation. Suppose that G and H are one-terminal-pair graphs. We shall use $G \bullet H$ to mean a series combination of G and H, and $G \parallel H$ to mean a parallel combination of G and H.

Chapter 5

Our results

Consider $G \bullet H$, a series combination of the one-terminal-pair graphs G and H. Since G and H have exactly one vertex in common, Theorem 5 gives the following result.

Theorem 17. *If G and H are one-terminal-pair graphs, then $C(G \bullet H) = C(G) \cdot C(H)$.*

Thus our choice of notation $G \bullet H$ represents both the vertex that G and H have in common and the fact that multiplication is the key to computing the number of compositions of a series combination.

However, when we consider $G \parallel H$, a parallel combination of the one-terminal-pair graphs G and H, we must know more than just the number of compositions of G and the number of compositions of H.

For example, consider the four graphs in Figure 5.1. Both G_1 and G_2 have 34 compositions, and both H_1 and H_2 have 89 compositions. However, the four parallel combinations $G_1 \parallel H_1$, $G_1 \parallel H_2$, $G_2 \parallel H_1$, and $G_2 \parallel H_2$ all have different numbers of compositions, as shown below.

$$\begin{aligned} C(G_1 \parallel H_1) &= 1772, \\ C(G_1 \parallel H_2) &= 1786, \\ C(G_2 \parallel H_1) &= 1909, \\ C(G_2 \parallel H_2) &= 1928. \end{aligned}$$

Therefore, to compute the number of compositions of $G \parallel H$, we need to know more about the internal structure of G and H.

We shall call a composition $\mathcal{C}$ of a one-terminal-pair graph G *closed* or *open* according as the two terminals of G are together or not in $\mathcal{C}$. Moreover,

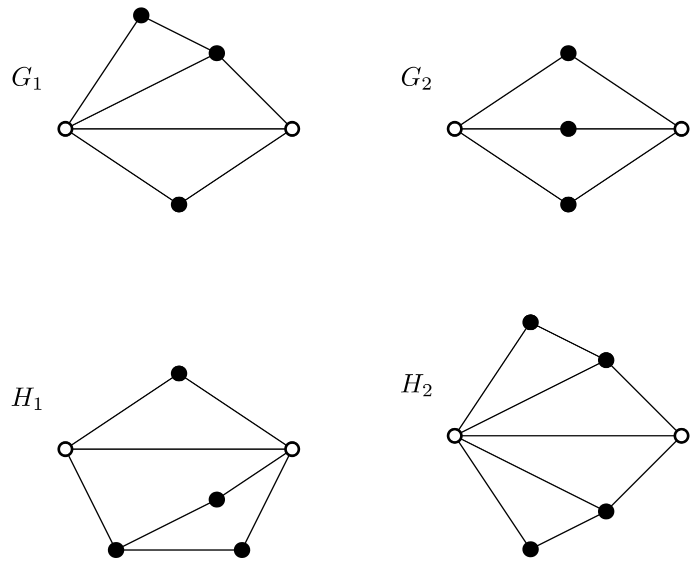

Figure 5.1: Four one-terminal-pair graphs.

call $\mathcal{C}$ an *almost closed composition* if the terminals of G are nearby in $\mathcal{C}$. We shall denote the number of closed compositions and almost closed compositions of a one-terminal-pair graph G by $\overline{C}(G)$ and $A(G)$, respectively.

We now introduce the following theorem.

Theorem 18. *If G and H are one-terminal-pair graphs, then* $C(G \parallel H) = C(G) \cdot C(H) - \overline{C}(G) \cdot A(H) - A(G) \cdot \overline{C}(H)$.

Proof. Let $\mathcal{C}_G$ and $\mathcal{C}_H$ be compositions of G and H, respectively, and let $\mathcal{C}$ be the corresponding composition of $G \parallel H$. Theorem 12 implies immediately that $\mathcal{C}$ does not induce $\mathcal{C}_G$ and $\mathcal{C}_H$ if the terminals are together in $\mathcal{C}_G$ and nearby in $\mathcal{C}_H$, or vice versa. On the other hand, Theorem 16 implies that $\mathcal{C}$ does induce $\mathcal{C}_G$ and $\mathcal{C}_H$ if the terminals are not nearby in $\mathcal{C}_G$ and not nearby in $\mathcal{C}_H$. The only cases that remain are those in which the terminals are nearby in both $\mathcal{C}_G$ and $\mathcal{C}_H$; or nearby in $\mathcal{C}_G$ and distant in $\mathcal{C}_H$, or vice versa.

Now the only two cases in which $\mathcal{C}$ does not induce $\mathcal{C}_G$ and $\mathcal{C}_H$ is when there are two vertices, both in $V(G)$, that are together in $\mathcal{C}$ but not in $\mathcal{C}_G$, and when there are two vertices, both in $V(H)$, that are together in $\mathcal{C}$ but not in $\mathcal{C}_H$. But we observe that the terminals are together in $\mathcal{C}$ if and only if they are together in $\mathcal{C}_G$ or in $\mathcal{C}_H$. Therefore in these remaining cases the terminals are not together in $\mathcal{C}$, and so $\mathcal{C}$ induces $\mathcal{C}_G$ and $\mathcal{C}_H$.

Consequently any composition $\mathcal{C}_G$ of G can be paired with any composition $\mathcal{C}_H$ of H to produce a composition of $G \parallel H$ that induces $\mathcal{C}_G$ and $\mathcal{C}_H$, except in the case when the terminals are together in $\mathcal{C}_G$ and nearby in $\mathcal{C}_H$, or vice versa. There are $C(G) \cdot C(H)$ pairings of a composition of G with a composition of H. From this number we must subtract the $\overline{C}(G) \cdot A(H)$ pairings in which the terminals are together in $\mathcal{C}_G$ and nearby in $\mathcal{C}_H$, and the $A(G) \cdot \overline{C}(H)$ pairings in which the terminals are nearby in $\mathcal{C}_G$ and together in $\mathcal{C}_H$. $\square$

The proof of Theorem 18 is quite similar to the proof of Theorem 9, where we essentially counted the possible subsets of the edges of the cycle graph and subtracted those subsets that contain all but one edge. If we think of a cycle graph as two path graphs connected in parallel, then the subsets we subtracted were those compositions in which the terminals were nearby in the composition of one subgraph and together in the composition of the other.

Theorems 17 and 18 allow us to compute the number of compositions of any series combination or parallel combination if we know several things about each of the graphs being combined, namely, the number of compositions, the number of closed compositions, and the number of almost closed compositions. In order to have a method for computing the number of compositions of an arbitrary series-parallel graph, however, we should also like to have formulas to find the numbers of closed and almost closed compositions of series and parallel combinations of one-terminal-pair graphs.

The desired formulas are simple for series combinations, as seen in the theorems below.

Theorem 19. *If G and H are one-terminal-pair graphs, then $\overline{C}(G \bullet H) = \overline{C}(G) \cdot \overline{C}(H)$.*

Proof. Let u and v be the terminals of G, and let w and x be the terminals of H. Without loss of generality, suppose that in $G \bullet H$ the vertices v and w are identified as the vertex z. Then the terminals of $G \bullet H$ are u and x. Plainly any path from u to x in $G \bullet H$ must include z, and thus any composition of $G \bullet H$ in which u and x are together must also have u and z together, and x and z together. Of course, it is also true by the transitivity of togetherness that any closed composition of G can be paired with any closed composition of H to produce a closed composition of $G \bullet H$. Therefore the number of closed compositions of $G \bullet H$ is the number of such pairings; this number is $\overline{C}(G) \cdot \overline{C}(H)$. $\square$

Theorem 20. *If G and H are one-terminal-pair graphs, then $A(G \bullet H) = A(G) \cdot \overline{C}(H) + \overline{C}(G) \cdot A(H)$.*

Proof. Let u and v be the terminals of G, and let w and x be the terminals of H. Without loss of generality, suppose that in $G \bullet H$ the vertices v and w are identified as the vertex z.

Consider any almost closed composition $\mathcal{A}$ of $G \bullet H$, resulting from a composition $\mathcal{C}_G$ of G and a composition $\mathcal{C}_H$ of H. The terminals u and x are nearby in $\mathcal{A}$. This means that there exists no path in $G \bullet H$ from u to x with all of its edges belonging to $\mathcal{A}$, but there does exist a path P from u to x with all but one of its edges belonging to $\mathcal{A}$. Plainly P must include z. Let e be the edge in P that does not belong to $\mathcal{A}$.

If e is in G, then u and z are nearby in $\mathcal{C}_G$, since one of the endpoints of e is together with u, and the other endpoint of e is together with z. (If u and z were together in $\mathcal{C}_G$, then there would exist a path in $G \bullet H$ from u to z with all edges belonging to $\mathcal{A}$; but then this path and the portion of P from z to x would form a path from u to x with all edges belonging to $\mathcal{A}$, and so $\mathcal{A}$ would not be an almost closed composition.) Moreover, if e is in G, then z and x are together in $\mathcal{C}_H$. A similar argument shows that if e is in H, then u and z are together in $\mathcal{C}_G$, and z and x are nearby in $\mathcal{C}_H$.

Thus any almost closed composition of $G \bullet H$ results from the pairing of a closed composition with an almost closed composition. The number of such pairings is $A(G) \cdot \overline{C}(H) + \overline{C}(G) \cdot A(H)$. □

When we consider parallel combinations, the formulas become slightly longer.

Theorem 21. *If G and H are one-terminal-pair graphs, then*

$$\begin{aligned}\overline{C}(G \parallel H) = {} & \overline{C}(G) \cdot C(H) + C(G) \cdot \overline{C}(H) \\ & - \overline{C}(G) \cdot A(H) - A(G) \cdot \overline{C}(H) - \overline{C}(G) \cdot \overline{C}(H).\end{aligned}$$

Proof. We are seeking the number of closed compositions of $G \parallel H$, that is, the number of compositions of $G \parallel H$ in which the terminals are together. Let $\mathcal{C}$ be a composition of $G \parallel H$ that induces some pair $(\mathcal{C}_G, \mathcal{C}_H)$ of compositions of G and H, respectively. As noted in the proof of Theorem 18, the terminals are together in $\mathcal{C}$ if and only if they are together in $\mathcal{C}_G$ or together in $\mathcal{C}_H$.

If the terminals are together in $\mathcal{C}_G$ (that is, if $\mathcal{C}_G$ is a closed composition of G), then $\mathcal{C}_G$ can be paired with any composition $\mathcal{C}_H$ of H, except almost closed compositions of H (in accordance with Theorem 12), to form a composition of $G \parallel H$ that induces $\mathcal{C}_G$ and $\mathcal{C}_H$ in which the terminals are together. This approach yields $\overline{C}(G) \cdot \big[C(H) - A(H)\big]$ closed compositions of $G \parallel H$. Similarly, we can pair closed compositions of H with any composition of G except almost closed compositions, and form $\overline{C}(H) \cdot \big[C(G) - A(G)\big]$ closed compositions of $G \parallel H$ in this way.

But if we simply add these together, we shall have counted some closed compositions of $G \parallel H$ twice, namely those in which the terminals are together in both $\mathcal{C}_G$ and $\mathcal{C}_H$. Therefore we must subtract $\overline{C}(G) \cdot \overline{C}(H)$ from this sum, so the total number of closed compositions of $G \parallel H$ is

$$\overline{C}(G) \cdot C(H) + C(G) \cdot \overline{C}(H) - \overline{C}(G) \cdot A(H) - A(G) \cdot \overline{C}(H) - \overline{C}(G) \cdot \overline{C}(H).$$

□

Theorem 22. *If G and H are one-terminal-pair graphs, then*

$$\begin{aligned} A(G \parallel H) = {} & A(G) \cdot C(H) + C(G) \cdot A(H) \\ & - A(G) \cdot \overline{C}(H) - \overline{C}(G) \cdot A(H) - A(G) \cdot A(H). \end{aligned}$$

Proof. Let $\mathcal{C}$ be a composition of $G \parallel H$ that induces some pair $(\mathcal{C}_G, \mathcal{C}_H)$ of compositions of G and H, respectively. If $\mathcal{C}$ is an almost closed composition of $G \parallel H$, then there does not exist a path in $G \parallel H$ from one terminal to the other with all of its edges belonging to $\mathcal{C}$, but there does exist a path P across the terminals with all but one of its edges belonging to $\mathcal{C}$.

If P is in G, then $\mathcal{C}_G$ is an almost closed composition of G, and $\mathcal{C}_H$ can be any composition of H, except a closed composition. This gives us $A(G) \cdot \big[C(H) - \overline{C}(H)\big]$ almost closed compositions of $G \parallel H$. On the other hand, if P is in H, then $\mathcal{C}_H$ is an almost closed composition of H, and $\mathcal{C}_G$ can be any open composition of G. We get $A(H) \cdot \big[C(G) - \overline{C}(G)\big]$ almost closed compositions of $G \parallel H$ in this way.

But as in the proof of Theorem 21, simply adding these together counts some almost closed compositions of $G \parallel H$ twice: those in which both G and H contain a path across the terminals with all but one edge belonging to $\mathcal{C}$. This necessitates the subtraction of $A(G) \cdot A(H)$ from this sum, so that the total count of almost closed compositions of $G \parallel H$ is

$$A(G) \cdot C(H) + C(G) \cdot A(H) - A(G) \cdot \overline{C}(H) - \overline{C}(G) \cdot A(H) - A(G) \cdot A(H).$$

□

Theorems 17 through 22 give us a way to recursively compute the number of compositions of any series-parallel graph. In the end the question comes

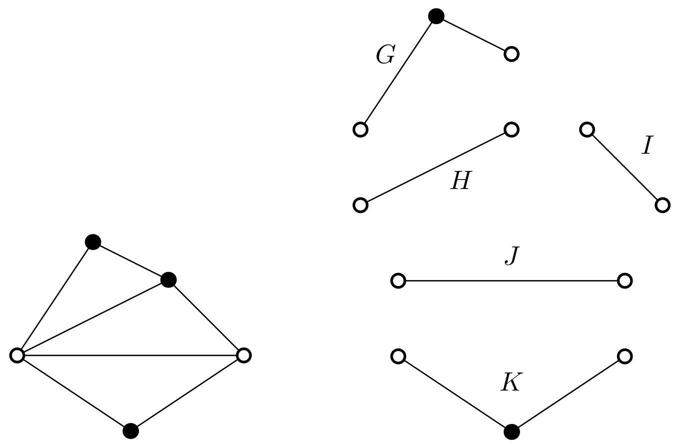

Figure 5.2: The graph G_1 of Figure 5.1.

down to the number of compositions of the path graph P_2. But of course it is easy to see that $C(P_2) = 2$, $\overline{C}(P_2) = 1$, and $A(P_2) = 1$.

For example, suppose we wish to verify that the graph G_1 in Figure 5.1 really has 34 compositions as claimed. We observe that the graph is a combination of five subgraphs, as seen in Figure 5.2. First graphs G and H are combined in parallel; the resulting graph is combined in series with graph I; and then this graph is combined in parallel with graphs J and K in turn.

We begin by noting that each of the graphs G, H, I, J, and K is a path graph. In fact, H, I, and J are P_2, so we have

$$C(H) = C(I) = C(J) = 2,$$
$$\overline{C}(H) = \overline{C}(I) = \overline{C}(J) = 1,$$
$$A(H) = A(I) = A(J) = 1.$$

In addition, G and K are P_3, so

$$C(G) = C(K) = 4,$$
$$\overline{C}(G) = \overline{C}(K) = 1,$$
$$A(G) = A(K) = 2.$$

(This can be seen by inspection, or by applying Theorems 1, 19, and 20.)

Now, from Theorems 18, 21, and 22, we have

$$\begin{aligned}
C(G \parallel H) &= 4 \cdot 2 - 1 \cdot 1 - 2 \cdot 1 = 5, \\
\overline{C}(G \parallel H) &= 1 \cdot 2 + 4 \cdot 1 - 1 \cdot 1 - 2 \cdot 1 - 1 \cdot 1 = 2, \\
A(G \parallel H) &= 2 \cdot 2 + 4 \cdot 1 - 2 \cdot 1 - 1 \cdot 1 - 2 \cdot 1 = 3.
\end{aligned}$$

Let $L = G \parallel H$ for ease of notation. The next step is to compute these three values for $L \bullet I$, using Theorems 17, 19, and 20; we obtain

$$\begin{aligned}
C(L \bullet I) &= 5 \cdot 2 = 10, \\
\overline{C}(L \bullet I) &= 2 \cdot 1 = 2, \\
A(L \bullet I) &= 3 \cdot 1 + 2 \cdot 1 = 5.
\end{aligned}$$

Setting $M = L \bullet I$, we now consider $M \parallel J$ and get

$$\begin{aligned}
C(M \parallel J) &= 10 \cdot 2 - 2 \cdot 1 - 5 \cdot 1 = 13, \\
\overline{C}(M \parallel J) &= 2 \cdot 2 + 10 \cdot 1 - 2 \cdot 1 - 5 \cdot 1 - 2 \cdot 1 = 5, \\
A(M \parallel J) &= 5 \cdot 2 + 10 \cdot 1 - 5 \cdot 1 - 2 \cdot 1 - 5 \cdot 1 = 8.
\end{aligned}$$

Lastly, taking $N = M \parallel J$, we examine $N \parallel K$, which is our original graph, and find that

$$C(N \parallel K) = 13 \cdot 4 - 5 \cdot 2 - 8 \cdot 1 = 34.$$

Of course, we could also compute $\overline{C}(N \parallel K)$ and $A(N \parallel K)$ if desired.

This method for computing the number of compositions of a series-parallel graph can easily be described as an algorithm. For a given series-parallel graph with q edges, this algorithm can run in $O(q)$ time, that is, in an amount of time proportional to the number of edges in the graph. A series-parallel graph can be recognized in $O(q)$ time [23]. Ultimately a series-parallel graph with q edges is composed of q copies of P_2 combined in series or in parallel; this requires $q - 1$ combinations. Hence this algorithm requires $q-1$ computations of C and $q-2$ computations of each of $\overline{C}$ and A.

Chapter 6

Future work

From the point of view of electric circuit analysis, the natural next step after determining how graphs combine in series and in parallel is to work out the star-delta transformation, first introduced by Kennelly [12]. In this transformation, a subgraph isomorphic to C_3 within a larger graph can be replaced by the star graph S_4, or vice versa, without affecting the properties of the graph as a whole. This technique is commonly used in the analysis of electric circuits which cannot be solved using only series and parallel calculations.

In fact, both the star-delta transformation and the substitution of an equivalent circuit for two circuits in series are particular cases of a more general theorem given by Rosen [21], which states that any subgraph isomorphic to the star graph S_n can be replaced by an equivalent "mesh," or complete graph K_{n-1}, without affecting the properties of the graph as a whole. (Since for $n \geq 5$ there are more edges in K_{n-1} than in S_n, it is not possible in general to go in the opposite direction, that is, to replace a subgraph isomorphic to K_{n-1} by S_n.) Rosen's theorem is discussed in more depth in [4].

In the problem of counting the number of compositions of a graph, the star-delta transformation would allow us to replace any "essentially delta" subgraph with an "essentially star" subgraph, or vice versa, without changing the number of compositions, closed compositions, or almost closed compositions. However, our initial work on this problem seems to indicate that this transformation does not exist: the set of equations that the equivalent subgraphs would need to satisfy seems to be inconsistent.

Perhaps a fruitful direction of study would be to generalize the results described in this paper by considering the class of graphs of a given tree width w. Intuitively, tree width can be understood by considering how a graph may be constructed. Any tree (other than the singleton graph) can be produced by starting with K_2 and repeatedly adding vertices, with each new vertex joined to exactly one existing vertex. To extend this procedure, we may begin with the complete graph K_{w+1} and repeatedly add vertices, joining each new vertex to exactly w existing vertices, every pair of which is adjacent. At the end, we may optionally delete some of these edges. Any graph which can be made in this way for some w, but cannot be made in this way for any smaller w, is said to be of tree width w. (Hence, trees are of tree width 1.) For instance, the ladder graph L_n is of tree width 2, since each of the squares in the ladder graph can be built from two triangles joined along an edge, with the diagonal edge being deleted at the end. Moreover, it is easily seen that every series-parallel graph is of tree width at most 2.

Another area for further research is in the area of grid graphs. The path graphs and ladder graphs have been discussed in this paper; Ridley and Mays [19] examine also the $3 \times n$ grid graph, deriving the following formula.

Theorem 23. *Let $G_{3,n}$ denote the $3 \times n$ grid graph $P_3 \,\square\, P_n$. Then*

$$C(G_{3,n}) = c_1\alpha_1^{n-1} + c_2\alpha_2^{n-1} + c_3\alpha_3^{n-1} + c_4\alpha_4^{n-1} + c_5\alpha_5^{n-1},$$

where the approximate numerical values of the constants are as follows.

i	c_i	α_i
1	0.157 907	−0.221 357
2	0.000 396 095	0.191 796
3	3.821 52	19.3717
4	$0.010\,0861 - 0.012\,6952i$	$1.828\,95 - 1.232\,29i$
5	$0.010\,0861 + 0.012\,6952i$	$1.828\,95 + 1.232\,29i$

The first few values for $C(G_{3,n})$, which have been verified by exhaustively examining all possible subsets of the edges, are given in Table 6.1. (Note that Ridley and Mays list $C(G_{3,4})$ incorrectly as 27 080; the correct value is 27 780.)

n	1	2	3	4
$C(G_{3,n})$	4	74	1434	27 780

Table 6.1: The first several values of $C(G_{3,n})$.

Other than these special cases, however, the number of compositions of grid graphs is not known. The problem which began our research into compositions of graphs involved finding the number of compositions of the 11×11 grid graph. At the time, it seemed as though the answer should be easily computable by a well known formula. But to the contrary, the study of compositions of graphs appears to be a relatively new field, with many open questions awaiting further research.

Bibliography

[1] Miklós Bóna. *A Walk Through Combinatorics: An Introduction to Enumeration and Graph Theory*. World Scientific, Singapore, 2002.

[2] Richard A. Brualdi. *Introductory Combinatorics*. Pearson Prentice Hall, Upper Saddle River, N. J., fourth edition, 2004.

[3] Peter J. Cameron. *Combinatorics: Topics, Techniques, Algorithms*. Cambridge University Press, 1996.

[4] G. W. Carter and A. Richardson. *Techniques of Circuit Analysis*. Cambridge University Press, 1972.

[5] Charalambos A. Charalambides. *Enumerative Combinatorics*. Chapman & Hall/CRC, Boca Raton, Fla., 2002.

[6] John Clark and Derek Allan Holton. *A First Look at Graph Theory*. World Scientific, Singapore, 1991.

[7] R. J. Duffin. Topology of series-parallel networks. *Journal of Mathematical Analysis and Applications*, 10(2):303–318, April 1965.

[8] R. J. Duffin. Electrical network models. In D. R. Fulkerson, editor, *Studies in Graph Theory, Part I*, volume 11 of *Studies in Mathematics*, pages 94–138. Mathematical Association of America, 1975.

[9] H. W. Gould. *Research Bibliography of Two Special Number Sequences*. Combinatorial Research Institute, Morgantown, W. Va., revised edition, January 1977.

[10] Ronald Gould. *Graph Theory*. Benjamin/Cummings, Menlo Park, Calif., 1988.

[11] Wilfried Imrich and Sandi Klavžar. *Product Graphs: Structure and Recognition*. Wiley-Interscience Series in Discrete Mathematics and Optimization. John Wiley & Sons, New York, 2000.

[12] A. E. Kennelly. The equivalence of triangles and three-pointed stars in conducting networks. *Electrical World and Engineer*, 34(12):413–414, September 1899.

[13] A. Knopfmacher and M. E. Mays. Graph compositions I: Basic enumeration. *Integers: Electronic Journal of Combinatorial Number Theory*, 1(A04):1–11, 2001.

[14] V. Krishnamurthy. *Combinatorics: Theory and Applications*. Ellis Horwood Series in Mathematics and Its Applications. Ellis Horwood, Chichester, West Sussex, 1986.

[15] Conrado Martínez and Xavier Molinero. An efficient generic algorithm for the generation of unlabelled cycles. In Michael Drmota, Philippe Flajolet, Danièle Gardy, and Bernhard Gittenberger, editors, *Mathematics and Computer Science III: Algorithms, Trees, Combinatorics and Probabilities*, Trends in Mathematics, pages 187–197, Basel, 2004. Birkhäuser Verlag.

[16] Jaroslav Nešetřil. Representations of graphs by means of products and their complexity. In J. Gruska and M. Chytil, editors, *Mathematical Foundations of Computer Science 1981*, Lecture Notes in Computer Science, pages 94–102, Berlin, 1981. Springer-Verlag.

[17] James W. Nilsson and Susan A. Riedel. *Electric Circuits*. Prentice Hall, Upper Saddle River, N. J., sixth edition, 2001.

[18] T. Nishizeki and N. Chiba. *Planar Graphs: Theory and Algorithms*. Number 32 in Annals of Discrete Mathematics. North-Holland, Amsterdam, 1988.

[19] J. N. Ridley and M. E. Mays. Compositions of unions of graphs. *Fibonacci Quarterly*, 42(3):222–230, August 2004.

[20] John Riordan and C. E. Shannon. The number of two-terminal series-parallel networks. *Journal of Mathematics and Physics*, 21:83–93, 1942.

[21] A. Rosen. A new network theorem. *Journal of the Institution of Electrical Engineers*, 62:916–918, 1924.

[22] Steven Skiena. *Implementing Discrete Mathematics: Combinatorics and Graph Theory with Mathematica*. Addison-Wesley, Redwood City, Calif., 1990.

[23] K. Takamizawa, T. Nishizeki, and N. Saito. Linear-time computability of combinatorial problems on series-parallel graphs. *Journal of the Association for Computing Machinery*, 29(3):623–641, July 1982.

[24] K. Thulasiraman and M. N. S. Swamy. *Graphs: Theory and Algorithms*. John Wiley & Sons, New York, 1992.

[25] Robin J. Wilson. *Introduction to Graph Theory*. Longman, Harlow, Essex, third edition, 1985.

Index