Foundations for Undergraduate Research in Mathematics

Aaron Wootton
Valerie Peterson
Christopher Lee
Editors

A Primer for Undergraduate Research

From Groups and Tiles to Frames and Vaccines

Birkhäuser

Foundations for Undergraduate Research in Mathematics

Series editor
Aaron Wootton
Department of Mathematics, University of Portland, Portland, USA

More information about this series at http://www.springer.com/series/15561

Aaron Wootton • Valerie Peterson
Christopher Lee
Editors

A Primer for Undergraduate Research

From Groups and Tiles to Frames and Vaccines

Birkhäuser

Editors
Aaron Wootton
Department of Mathematics
University of Portland
Portland, OR, USA

Valerie Peterson
Department of Mathematics
University of Portland
Portland, OR, USA

Christopher Lee
Department of Mathematics
University of Portland
Portland, OR, USA

ISSN 2520-1212 ISSN 2520-1220 (electronic)
Foundations for Undergraduate Research in Mathematics
ISBN 978-3-319-66064-6 ISBN 978-3-319-66065-3 (eBook)
DOI 10.1007/978-3-319-66065-3

Library of Congress Control Number: 2017959749

Mathematics Subject Classification: 00A05, 00A07, 00A08, 00B10

Printed on acid-free paper

This book is published under the trade name Birkhäuser, www.birkhauser-science.com
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Contents

Coxeter Groups and the Davis Complex

Timothy A. Schroeder

Suggested Prerequisites. *Group theory, Graph theory, Combinatorial topology.*

1 Introduction

This chapter presents topics in an area of mathematics at the intersection of geometry, topology, and algebra called Geometric Group Theory. It is likely that students have been exposed to geometry and abstract algebra topics as undergraduates. Some reading this may have also been introduced to topology, as well. In this chapter, we will be using terms and concepts from each of these areas, the point being to develop a working knowledge of these terms and concepts that allows us to progress toward our goal: A reflection group acting on a topological space. We will not spend much time on topological details (students are certainly encouraged to pursue that subject formally). Geometric details, referenced throughout, but specifically in Section 3.2, are suggested as a project in Section 6. That leaves algebra. Perhaps the best, and most familiar place to begin. We recall the definition of a group G:

Definition 1. A *group* G is a set G, together with a closed binary operation, denoted $\cdot$, such that the following hold:

- Associativity: For all $a, b, c \in G$, $(a \cdot b) \cdot c = a \cdot (b \cdot c)$.
- Identity: There is an element e in G such that for all $x \in G$, $e \cdot x = x \cdot e = x$.
- Inverse: For every $a \in G$, there is an element $a' \in G$ such that $a \cdot a' = a' \cdot a = e$. Such an element a' is unique, and is denoted a^{-1}.

T.A. Schroeder (✉)
Murray State University, Murray, KY 42071, USA
e-mail: tschroeder@murraystate.edu

A. Wootton et al. (eds.), *A Primer for Undergraduate Research*, Foundations for Undergraduate Research in Mathematics, DOI 10.1007/978-3-319-66065-3_1

Students in an abstract algebra course begin with this definition, and likely embark on a journey through group theory: element orders, subgroups, cosets, homomorphisms, isomorphisms, cyclic groups, generators, etc. Eventually, students may be introduced to group actions, along with their stabilizers and orbits. All the while, most likely, using the favorite dihedral groups as prototypes.

As useful and correct and edifying as that exposure is, it may strike students as somewhat sterile. The groups are axiomatically presented, they are thought of abstractly, and pictures (aside from regular polygons or the occasional Cayley graph of a finite group) may be few and far between. It is our intention that the groups we present in this context will have a decidedly constructive, and even geometric, flavor to them.

In this chapter, we will study *finitely presented* groups, specifically *Coxeter groups*, and we will present a constructive view of a topological space on which these Coxeter groups act. We begin with an overview of group presentations and graph theory, then define Coxeter groups and the associated spaces. The reader should note that many technical terms are italicized, and can be referenced in the included citations or elsewhere.

2 Group Presentations and Graphs

Let A be a set and define the set $A^{-1} = \{a^{-1} \mid a \in A\}$. Define W_A to be the collection of all finite length *words* in $A \cup A^{-1}$. A "word" being a finite string of elements from the set $A \cup A^{-1}$. (We are thinking of the set A as our "alphabet.") For example, if $A = \{a, b\}$, then a, ab, ba, aba^{-1}, $bb^{-1}ab$ are all elements of the set of words W_A. Perhaps you sense that this list is redundant. That is, you may already be thinking that two of the words are actually equivalent. You're right, of course, for as the notation suggests, elements of the set A^{-1} are to play the role of inverses. To make this clear, let's get a little more formal.

First, to the set W_A, we include the word comprised of no elements, the so-called *empty word*, denoted by the symbol 1.

Now, to have a group, we must have an associative operation, usually described and understood as some sort of multiplication. So, on the set W_A, we define multiplication by *concatenation*. (In other words, "put next to each other.") In the example above, to multiply ba and aba^{-1}, we have $ba \cdot aba^{-1} = baaba^{-1} = ba^2ba^{-1}$. You should verify that, in general, concatenation is in fact associative.

Next, we say $w_2 \in W_A$ is obtained from $w_1 \in W_A$ by an *elementary reduction* (or *expansion)* if w_2 is obtained from w_1 by deleting (or inserting) a sub-word of the form aa^{-1} or $a^{-1}a$, for some element $a \in A$. We say that two words w and w' are *equivalent* if we may pass from w to w' by a finite sequence of elementary reductions or expansions (both are allowed), and we write $w \sim w'$. (As you may have guessed, this means that in our list above, $ab \sim bb^{-1}ab$.) The relation $\sim$ defines an equivalence relation on the set W_A. (See Exercise 1.)

Finally, let F_A be the collection of equivalence classes of $\sim$, where for $w \in W_A$, $[w]$ denotes the equivalence class containing w. Concatenation on W_A induces a well-defined operation on F_A: For $u, v \in W_A$, $[u] \cdot [v] = [uv]$. (See Exercise 2.) In fact, F_A is a group with identity the equivalence class containing 1. It is called the *free group on A*.

2.1 Group Presentations

It could be the case that within a set of words, or within a group, there could be many ways, besides using elementary reductions or expansions, to represent a given element. To handle this more complicated situation, we add what are called *relators* to our group, and study *presentations* of groups.

Let A be a set and consider again F_A, the free group on A. Let R be a set of words in the alphabet $A \cup A^{-1}$, and define $N(R)$ to be the smallest normal subgroup of F_A containing the equivalence classes of the elements of R. (This normal subgroup is formed by taking the collection of all finite products of conjugates of elements of R and their inverses in F_A.) We have the following definition.

Definition 2. Let A, R and $N(R)$ be as above. The *group defined by the presentation* $\langle A \mid R \rangle$ is the quotient group $F_A/N(R)$.

That is, a group defined by a presentation is the quotient of a free group by the normal subgroup generated by the words in some set R. We equate a group with its presentation, writing $F_A/N(R) = \langle A \mid R \rangle$, but note that it is the case that a given group can have multiple presentations.

If A is a finite set (for it could be the case that A is an infinite set), we say the corresponding group is *finitely generated*. If R is also finite, we say that the group is *finitely presented*. For reference on group presentations, see [6] or [10].

In practice we often take a slightly more constructive approach to defining and working with the elements of a finitely presented group. We highlight this perspective next.

2.1.1 A Constructive Approach

Resetting the table, consider the set W_A of words in the alphabet $A \cup A^{-1}$, where A is some finite set, and let R be a finite set of words contained in W_A. The elements of R, again called "relators," will serve to identify certain words in W_A, besides those identified in the free group F_A. Indeed, we say a word w_2 in W_A is obtained from $w_1 \in W_A$ by a *simple R-reduction* (or *R-expansion*) if w_2 is obtained from w_1 by deleting (or inserting) a sub-word r, where $r \in R$. We then say two words w and w' are *R-equivalent*, and write $w \sim_R w'$, if there exist a finite sequence of simple R-reductions, simple R-expansions, elementary reductions, and elementary expansions leading from w to w'. As before, concatenation induces an operation on the set of equivalence classes of R-equivalent words of W_A, and we have the structure of a group G with presentation $\langle A \mid R \rangle$. We again equate the group with its presentation and write $G = \langle A \mid R \rangle$.

Now, for an element $w \in W_A$, we think of w as representing its entire equivalence class of R-equivalent words and drop the equivalence class notation (or the associated coset), understanding that other words may be equivalent to w. That is, we think of the words themselves as elements of the group; but in that set, there is much redundancy. In particular, we write $w = w'$ *as group elements* when $w \sim_R w'$ as words (or when w and w' are in the same coset of $N(R)$). Multiplication is still represented by concatenation, and the identity element still represented by 1. This means that the generating set A is considered a subset of the group itself. Also, R-equivalence means that we are able to insert or delete each $r \in R$ into any part of a word w without changing the group element. This amounts to equating each relator r with the identity element 1, a perspective often indicated in the presentation as each $r \in R$ will often be equated to 1 in the right hand side of the presentation. Finally, note that if $A = \{a_1, a_2, \ldots, a_n\}$ and $R = \{r_1, r_2, \ldots, r_k\}$, then we may write $\langle a_1, a_2, \ldots, a_n \mid r_1, r_2, \ldots, r_k \rangle$ to denote $\langle A \mid R \rangle$.

2.2 Some Basic Graph Theory

In order to fully explore group presentations, and the structures of the resulting groups, it does us well to review (or introduce) some basic graph theory.

Let V be a set, and let E be a collection of two element subsets of V, where an individual set can be repeated in the collection, and an individual element can be repeated to form a two-element subset. Such sets define a graph $\Gamma = (V, E)$ where the elements of V are the *vertices* of the graph and the elements of E the *edges*. A set $\{v, w\} \in E$ indicates an edge between vertices v and w, and $\{v, v\} \in E$ defines an edge in Γ from v to itself, i.e. a *loop* in Γ. If a subset in E has multiplicity n, then we include n edges between the associated vertices in Γ. Here, Γ is said to have *multi-edges*.

We define a *directed graph* by taking $E \subseteq V \times V$ where the ordered pair (v, w) indicates a directed edge between the vertices v and w. We indicate this pictorially by placing an arrow on the associated edge. Of course, this sort of structure can indicate an orientation to the edges in the graph, where traversing the associated edge in different directions has different implications. For our purposes, it is possible to have a graph with both directed edges and undirected edges. In this case, the edge set E will denote undirected edges by two element sets, and directed edges by ordered pairs.

A graph or directed graph $\Gamma = (V, E)$ is said to be *labeled*, or *weighted*, if there is a function from the set of edges E to a set of labels.

A graph Γ is said to be *finite* if both V and E are finite sets. Γ is said to be *simple* if Γ includes no loops nor multi-edges.

Two graphs $\Gamma_1 - (V_1, E_1)$ and $\Gamma_2 = (V_2, E_2)$ are *isomorphic* if there exists a bijection $f : V_1 \to V_2$ such that if $\{u, v\} \in E_1$, then $\{f(u), f(v)\} \in E_2$. A similar definition exists for directed edges, that is if $(u, v) \in E_1$, then $(f(u), f(v)) \in E_2$. If the bijection f maps V_1 to itself, and $E_1 = E_2$, then the map f defines an automorphism of the graph $\Gamma_1(V_1, E_1)$.

Example 1. Let $V = \{a, b, c, d, e, f\}$, and consider edge sets

$$E_1 = \{\{a, b\}, \{b, c\}, \{c, d\}, \{d, e\}, \{e, f\}, \{f, g\}\}\,, \textit{and}$$

$$E_2 = \{(a, b), (b, c), (c, a), (d, e), (e, f), (f, d), \{a, d\}, \{c, e\}, \{b, f\}\}\,.$$

The corresponding graphs $\Gamma_1 = (V, E_1)$ and $\Gamma_2 = (V, E_2)$ are shown Figure 1.

Example 2. Let $V = \mathbb{Z}$, and $E = \{\{n, n+1\} \mid n \in \mathbb{Z}\}$. Then $\Gamma = (V, E)$ can be understood as the real line, with vertices at every integer.

2.3 Cayley Graphs for Finitely Presented Groups

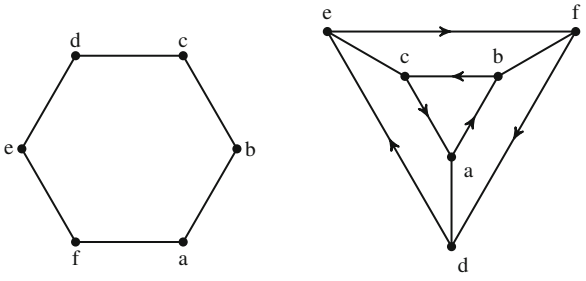
We now present the contruction of the *Cayley graph* associated to a finitely presented group. The Cayley graph is a very useful graph, endowed with some of the additional structure discussed above. Let $G = \langle A \mid R \rangle$, we create the corresponding labeled Cayley graph Γ as follows:

- $V = G$; that is, we include one vertex for every element of G.
- E: Given any $v \in G$ and $a \in A$, then there is a directed edge (v, va) from v to the element $va \in G$. We label this edge by the generator a.

This means that in Γ, each vertex has one edge emanating from it for each element of A, and it will also have one edge entering for each element of A. It should be noted that we view the vertices of the graph as elements of the group, that is, we view $G \subseteq \Gamma$. So, from a given vertex v, traversing an edge labeled a with the orientation corresponds to multiplying v on the right by a, and traversing an edge labeled a against the orientation corresponds to multiplying v on the right by a^{-1}.

Example 3. Let $G = \langle a \mid a^6 \rangle$. Then the Cayley graph of G can be viewed as the graph Γ_1 in Figure 1, but replace the edges there with directed edges all oriented to flow counter-clockwise around the hexagon, and with $a = a$, $b = a^2$, $c = a^3$, ... etc.

Fig. 1 Γ_1 and Γ_2.



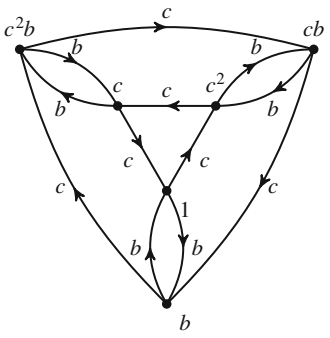


Fig. 2 The directed Cayley graph for G in Example 4.

Example 4. Let $G = \langle b, c \mid b^2, c^3, bcb^{-1}c \rangle$. G has Cayley graph shown in Figure 2.

In Example 4, due to the generator b having order 2 (we have $b^2 = 1$ in G), the description of the Cayley graph given above prescribes multi-edges. For at any $g \in G$, since $gb^2 = g$, there is an edge labeled b emanating from g to gb, and an edge entering g from gb. To avoid these multi-edges, and to reflect the fact that $b^{-1} = b$, it is our convention that when a generator b has order 2 we will identify incoming and outgoing edges corresponding to b; and instead of two directed edges, we will include one undirected edge. Thus, the Cayley graph for the group in Example 4 will actually be viewed as the graph Γ_2 in Figure 1, with directed and undirected edges.

Finally, we remark that the vertices of a Cayley graph correspond to elements of the group, hence *equivalence classes* of words. As a result, it can be very difficult to tell when two words represent the same element. So, constructing the Cayley graph is not as straightforward as one might expect. However, the description above enables us to understand a local picture of the Cayley graph at any vertex.

Exercise 1. Show that the relations $\sim$ and $\sim_R$ on the set W_A described above generate *equivalence relations*.

Exercise 2. Show that concatenation is well-defined on equivalence classes. Then, show that F_A is a group, with identity element 1. (For a given $u \in W_A$, what is $[u]^{-1}$?) In general, is F_A abelian?

Exercise 3. Let $A = \{a, b\}$, $R = \{aba^{-1}b^{-1}\}$. Show that $ab \sim_R ba$.

Exercise 4. Construct the Cayley diagram for the following presented groups.

(a) $\langle a \mid \ \rangle$
(b) $\langle a, b \mid \ \rangle$
(c) $\langle a, b \mid aba^{-1}b^{-1} \rangle$
(d) $\langle a, b \mid aba^{-1}b \rangle$
(e) $\langle a, b \mid b^2 \rangle$

(f) $\langle a, b \mid a^2 b^{-2} \rangle$
(g) $\langle a, b \mid a^4, b^2 \rangle$
(h) $\langle a, b \mid a^2 b^{-3} \rangle$
(i) $\langle a, b, c, d \mid aba^{-1}b^{-1}, cdc^{-1}d^{-1} \rangle$

Challenge Problem 1. A very strong connection between group theory and topology is now at hand. Indeed, given a group $G = \langle A \mid R \rangle$ with Cayley graph Γ (with single edges for generators of order 2), Γ can be given the "path-metric" by making each edge isometric to the unit interval. Define then the distance d between two points to be the minimum path length (or infimum) over all paths connecting the two points. Show that this defines a metric on Γ. Notice that for vertices which correspond to $u, v \in G$, $d(u, v) \in \mathbb{Z}$. Show that in this case, $d(u, v)$ is the minimum length of words w in the alphabet $A \cup A^{-1}$ representing the element $u^{-1}v$. The restriction of the metric to $G \subseteq \Gamma$ is the so called "word-length metric."

Challenge Problem 2. A free group has a more 'universal' definition: Let G be a group and let $A \subseteq G$. We say that G is *free on* A, denoted G_A, if for every group H, and any function $f : A \to H$, then there exists a unique homomorphism $h : G \to H$ s.t. $h|_A = f$.

For a given set A, show that

(a) If G is free on A, then A generates G. (That is, every element of G is a product of elements of A and their inverses.)
(b) If G is free on A, then A contains no elements of finite order.
(c) The group F_A (defined above) is free on A. (That is, for any group H and function $f : A \to H$, we must verify that f extends uniquely to a homomorphism.)
(d) Let G_A be a group that is free on A, then $F_A \cong G_A$. (That is, the definitions are equivalent!)

3 Coxeter Groups

In this section, we'll explore a special type of finitely presented group called a *Coxeter* group. Put succinctly, Coxeter groups are groups that are generated by elements of order 2, often viewed as reflections in some geometric space. As you may recall from an undergraduate geometry course, or can find in a standard geometry text such as [15, Chapter 10], isometries of geometric spaces can be understood as compositions of reflections. Therefore, one can see the importance of groups generated by reflections, and their natural connection to geometry. It should also be noted that the study of such *finite* groups, generated by reflections acting on $\mathbb{R}^2$, are essential in classifying Lie groups and Lie algebras, and the classification of regular polytopes. Coxeter groups are generalizations of this idea, where the order 2 generators are not necessarily viewed as reflections on $\mathbb{R}^n$, but will almost certainly be viewed as some sort of homeomorphism of a topological space.

As we'll see, the review of some basic graph theory in Section 2.2 will be useful, since the presentations of Coxeter groups can be encoded as finite, simple, labeled graphs; and because the topological spaces on which we will have these groups act are intimately related to their Cayley graphs.

3.1 The Presentation of a Coxeter Group

Let $\Gamma = (S, E)$ be a finite simple graph with vertex set S and with edges labeled by integers ≥ 2. Denote by m_{st} the label on the edge $\{s, t\}$. Γ encodes the data for a presentation of a Coxeter group W_Γ

$$W_\Gamma = \left\langle S \mid s^2 = 1 \text{ for each } s \in S \text{ and } (st)^{m_{st}} = 1, \text{ for each edge } \{s, t\} \text{ of } \Gamma \right\rangle . \tag{1}$$

The pair (W_Γ, S) (or simply (W, S) when the graph Γ is clear) is called a *Coxeter system*. We call such a labeled graph Γ a *Coxeter graph*. Throughout this chapter, we will take such a graph as the defining data for our Coxeter groups, noting that the vertices of the graph correspond to the generators of the group. This is standard convention: To simultaneously view each $s \in S$ as a generator of the group, an element of the group, and a vertex of the defining Coxeter graph. See [4, 9] for further reference on Coxeter groups and Coxeter systems. See [14] for further treatment on the defining Coxeter graphs.

Observe that the Coxeter graph Γ is not the Cayley graph associated to the group. Rather, it is an efficient way to encode the presentation of the group. There are other ways of defining Coxeter groups; for example Coxeter matrices or Dynkin diagrams. But, any such effort is just encoding the above type of presentation in another way. Our focus will be on the so-called Coxeter graphs.

In summary, a Coxeter group is generated by a set of elements that have order 2, and the only other relators are of the form $(st)^{m_{st}}$, where $s \neq t$, where m_{st} is the order of the element (st). Also, since the generators each have order two, they are their own inverses and we refer to them as *reflections* or *involutions*. That is to say, a Coxeter group is a group that is generated by reflections.

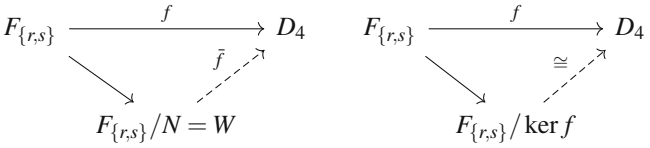
As noted in Section 2.1, the relators of the type r^2 and $(st)^{m_{st}}$ amount to equating these words to the identity element of the corresponding Coxeter group. See Exercises 5, 6, 7, and 8 to explore the implications of these relators, and the ensuing structure of the Coxeter group. Further, the reader should note that if two vertices s and t are not connected by an edge in Γ, then they do not define a relator and themselves generate an infinite subgroup, as Example 5 illustrates.

Example 5. Let Γ be a graph with two vertices, and no edges. Then

$$W = \left\langle r, s \mid r^2, s^2 \right\rangle ,$$

and its elements can be algorithmically listed: $1, r, s, rs, sr, rsr, srs, rsrs, srsr, \ldots$. Is it clear why the generators alternate within each word in this list? This group is called the infinite dihedral group, and denoted D_∞. Is it clear why the group is infinite?

Example 6. Let D_4 denote the dihedral group of order 8, that is, D_4 is the group of isometries of a square, and let $W = \langle r, s \mid r^2, s^2, (rs)^4 \rangle$. We will show $W \cong D_4$. To do this, recall the definition of a group defined by a presentation in Definition 2, and let $F_{\{r,s\}}$ denote the free group on generators r and s, and N the normal subgroup generated by the relators $\{r^2, s^2, (rs)^2\}$. That is, N is the smallest normal subgroup of $F_{\{r,s\}}$ containing the relators. Define a map $f : F_{\{r,s\}} \to D_4$ by mapping r to a reflection across a diagonal of a square, s to a reflection across an adjacent side bisector, and extending f to the rest of $F_{\{r,s\}}$ by requiring that f be a homomorphism. That is, a word in the alphabet $\{r, s, r^{-1}, s^{-1}\}$ is mapped to the corresponding composition of the reflections on the square described above. In particular, the student can verify that f is surjective and that all the relators are all in $\ker f$. From these observations, we can conclude two things: (1) Since $\ker f$ is a normal subgroup of $F_{\{r,s\}}$ containing the relators, we must have that $N \leq \ker f$; and (2) By the universal property of quotient groups, [10, 5.6], f descends to a map $\bar{f} : F_{\{r,s\}}/N = W \to D_4$. We have two commuting diagrams:



The diagram on the right being the classical 'first' or 'fundamental' isomorphism theorem. It gives us that $\left[F_{\{r,s\}} : \ker f\right] = 8$. Since $N \leq \ker f$, and we have that $\left[F_{\{r,s\}} : N\right] = \left[F_{\{r,s\}} : \ker f\right] \cdot [\ker f : N]$, we know that

$$|W| = \left[F_{\{r,s\}} : N\right] \leq 8.$$

But, by simply observing words in the alphabet $\{r, s, r^{-1}, s^{-1}\}$ subject to the relators, it is clear that any element of the group has as a representative a word of length at most 4. Indeed, since $(rs)^4 = 1$, we know $rsrsrsrs = 1$, $rsrsrsr = s$, $rsrsrs = sr$, $rsrsr = srs$, $rsrs = srsr$, and so on. This means that $\{1, r, s, rs, sr, rsr, srs, rsrs = srsr\}$ exhausts the set of words that represent the distinct elements of W, and so $|W| \leq 8$. Thus, $|W| = 8$, which means that $[\ker f : N] = 1$, and there is actually only one diagram. In particular, we have that $\bar{f} : W \to D_4$ is an isomorphism.

Example 7. One can show, using an argument similar to that in Example 6, that the Coxeter group $W = \langle r, s \mid r^2, s^2, (rs)^n \rangle$ is isomorphic to D_n, the dihedral group of order $2n$.

Example 8. Let $W = \langle r, s, t \mid r^2, s^2, t^2, (rs)^2, (st)^2, (rt)^2 \rangle$. It is a Coxeter group and in a similar manner to that above, one can show that W is isomorphic to $\mathbb{Z}_2 \oplus \mathbb{Z}_2 \oplus \mathbb{Z}_2$ (See Figure 3).

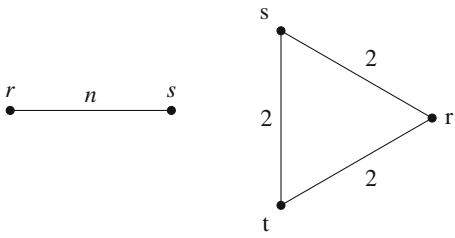


Fig. 3 Coxeter graphs for Examples 7 and 8 respectively.

Example 9. Consider the group

$$W = \left\langle r, s, t, u, v \mid r^2, s^2, t^2, u^2, v^2, (rs)^2, (st)^2, (tu)^2, (uv)^2, (rv)^2 \right\rangle .$$

Verify that W has corresponding Coxeter graph Γ where Γ is a pentagon with each edge labeled 2. Note that W is infinite, as it contains copies of D_∞ as subgroups.

3.2 Coxeter Groups and Geometry

To set up a discussion of the relationship between geometry and Coxeter groups, we recall the following facts, often covered in an undergraduate transformational geometry course.

Fact 1 Let P be a (2-dimensional) geometric space meeting the axioms of so-called "Neutral geometry." Given a line $l \subset P$, the reflection over the line l, denoted r_l is an isometry of the space. That is, the distance between two points p and q is the same as the distance between their reflected images p' and q'.

Fact 2 With P as above, if $\gamma : P \to P$ is an isometry, then γ can be understood as a composition of 1, 2, or 3 reflections over a line.

Fact 3 The composition of reflections over lines whose (acute) angle between them is α is a rotation through an angle of 2α.

(The student is encouraged to recall the construction of a reflection in neutral, plane geometry and to verify that the composition of a reflection with itself is the identity isometry on the geometric plane. In other words, a reflection defined in this way has order 2. For reference, see [15, Chapter 10].)

The first two facts provide great motivation for the study of Coxeter groups and their inherent connection to geometry. Indeed, they give us that the group of isometries of a geometric plane is generated by elements of order 2. The third fact gives context for the other relators present in the presentation of a Coxeter group.

In later chapters, and in the suggested project 2, we give a Coxeter group as our given data, and from it try to determine an appropriate geometric model. Here and in Challenge problem 5, we turn this around. Namely, we present some geometric models and reflections, and ask the reader to determine the associated group.

3.2.1 Euclidean Space and Reflections

View $\mathbb{R}^2$ as the set of 2-dimensional vectors over $\mathbb{R}$ equipped with the usual dot product

$$\langle \mathbf{u}, \mathbf{v} \rangle = u_1 v_1 + u_2 v_2$$

where $\mathbf{u} = (u_1, u_2)$ and $\mathbf{v} = (v_1, v_2)$ are vectors in $\mathbb{R}^2$. A line l in $\mathbb{R}^2$ through $\mathbf{x}_o$ in the direction of $\mathbf{v}$ has parametric equation $\mathbf{x} = \mathbf{x}_0 + \mathbf{v}t$, where $t \in \mathbb{R}$. Any such line defines a reflection r_l with formula

$$r_l(\mathbf{x}) = \mathbf{x} - 2 \langle \mathbf{u}, \mathbf{x} - \mathbf{x_0} \rangle \mathbf{u}, \tag{2}$$

for any $\mathbf{x} \in \mathbb{R}^2$ and where $\mathbf{u}$ is a unit vector orthogonal to $\mathbf{v}$.

3.2.2 Spherical Geometry and Reflections

Denote by $\mathbb{S}^2$ the subset of $\mathbb{R}^3$ of points (x_1, x_2, x_3) for which $x_1^2 + x_2^2 + x_3^2 = 1$ and call it the "2-sphere." With the usual dot product defined above extended to $\mathbb{R}^3$, we see that $\mathbb{S}^2$ can be viewed as the set of vectors $\mathbf{x}$ for which $\langle \mathbf{x}, \mathbf{x} \rangle = 1$.

A line l in $\mathbb{S}^2$ is defined as the intersection of $\mathbb{S}^2$ with a plane through the origin in $\mathbb{R}^3$. Any such line defines a reflection r_l of $\mathbb{S}^2$ in a similar way to that above. Indeed, for any $\mathbf{x} \in \mathbb{S}^2$, we have

$$r_l(\mathbf{x}) = \mathbf{x} - 2 \langle \mathbf{u}, \mathbf{x} \rangle \mathbf{u}, \tag{3}$$

where $\mathbf{u} \in \mathbb{R}^3$ is unit vector orthogonal to the plane defining the line l.

3.2.3 Hyperbolic Geometry and Reflections

Define a modified inner product on $\mathbb{R}^3$ by

$$\langle \mathbf{x}, \mathbf{y} \rangle_M = x_1 y_1 + x_2 y_2 - x_3 y_3.$$

(The "M" stands for Minkowski, and the student should note that this formula does not technically define an "inner product," as it's not the case that $\langle \mathbf{x}, \mathbf{x} \rangle_M \geq 0$ for all vectors $\mathbf{x}$.) Let $\mathbb{H}^2$ denote the subset of $\mathbb{R}^3$ for which $\langle \mathbf{x}, \mathbf{x} \rangle_M = -1$ and $x_3 > 0$. This is just the upper sheet of the two-sheeted hyperboloid in $\mathbb{R}^3$ defined by equation $x^2 + y^2 - z^2 = -1$, and called the *hyperboloid model* for hyperbolic space.

A line l in $\mathbb{H}^2$ is defined as the intersection of $\mathbb{H}^2$ with a plane through the origin in $\mathbb{R}^3$, and, as above, any such line defines a reflection in $\mathbb{H}^2$. In particular, let $\mathbf{u} \in \mathbb{R}^3$ denote a unit vector orthogonal to plane defining l, with respect to the modified inner product $\langle \quad , \quad \rangle_M$. (This means that $\langle \mathbf{u}, \mathbf{v} \rangle_M = 0$ for any vector $\mathbf{v}$ in the plane defining l.) For any $\mathbf{x} \in \mathbb{H}^2$, the reflection r_l is defined by

$$r_l(\mathbf{x}) = \mathbf{x} - 2 \langle \mathbf{u}, \mathbf{x} \rangle_M \mathbf{u}. \tag{4}$$

3.2.4 The Poincaré Disk Model for Hyperbolic Space

There is another model we'll consider for hyperbolic space called the "Poincaré disk model," denoted by $\mathbb{H}^2_P$. In terms of its points, $\mathbb{H}^2_P$ consists of $(x, y) \in \mathbb{R}^2$ for which $x^2+y^2 < 1$, the interior of the unit disk in $\mathbb{R}^2$. However, distance in $\mathbb{H}^2_P$ is calculated in such a way that the distance from the origin to the boundary of the disk is infinite. As a result, lines come in two forms:

1. The portion of lines through the origin in $\mathbb{R}^2$ contained in the interior of the unit disk, or
2. The portion of circles in $\mathbb{R}^2$ orthogonal to the boundary circle $x^2 + y^2 = 1$ contained in the interior of the unit disk.

Since lines come in two forms, the corresponding reflections come in two forms.

1. If l is a line of type (1) above, then r_l is a restriction to the interior of the unit disk of the standard Euclidean reflection defined in Equation 2 with $\mathbf{x}_0 = (0, 0)$.
2. If l is a line of type (2), then r_l is the inversion in the circle orthogonal to the boundary circle, applied to the points in the interior of the unit disk.

The student may recall that the inversion $\mathbf{x}'$ of a point $\mathbf{x}$ in a circle of radius k with center $\mathbf{x}_0$ is given by

$$\mathbf{x}' = \mathbf{x}_0 + \frac{(\mathbf{x} - \mathbf{x}_0)}{\langle \mathbf{x} - \mathbf{x}_0, \mathbf{x} - \mathbf{x}_0 \rangle}. \tag{5}$$

From the above formulas, one can easily see the how the hyperboloid model is completely analogous to the Euclidean and spherical model. An advantage of the Poincaré model over the hyperboloid model is that it is *conformal*. That is, angles between lines in $\mathbb{H}^2_P$ are equal to the angles between the corresponding lines or circles in $\mathbb{R}^2$.

Exercise 5. Let $r \in A$ and suppose $rr = r^2 \in R$. Show that $r \sim_R r^{-1}$.

Exercise 6. Let $r, s \in A$ and suppose $r^2, s^2, (rs)^2 \in R$. Show that $rs \sim_R sr$.

Exercise 7. Let $r, s \in A$ and suppose $(rs)^3 = rsrsrs, r^2, s^2 \in R$. Show that $rsr \sim_R srs$.

Exercise 8. Let $r, s \in A$ and suppose $r^2, s^2, (rs)^n \in R$, for some $n \in \mathbb{Z}$, $n \geq 2$. Then $(sr)^n = 1$. (That is, if you have the relator $(rs)^n$. You also have the relation $(sr)^n$.)

Exercise 9. Write the presentation of a Coxeter group for each of the three Coxeter graphs shown in Figure 4.

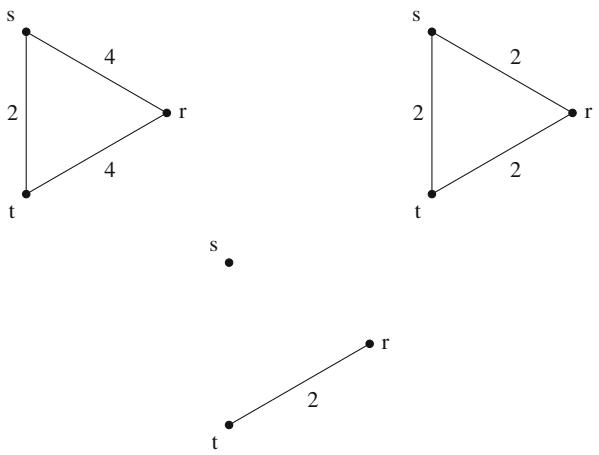


Fig. 4 The Coxeter graphs for Exercise 9.

Exercise 10. Sketch the Coxeter graph that defines the following presentations.

(a) $\langle r, s \mid r^2, s^2 \rangle$
(b) $\langle r, s, t \mid r^2, s^2, t^2, (rs)^2, (st)^3, (rt)^6 \rangle$
(c) $\langle r, s, t \mid r^2, s^2, t^2, (rs)^2, (st)^4, (rt)^4 \rangle$
(d) $\langle r, s, t \mid r^2, s^2, t^2, (rs)^4, (st)^4, (rt)^4 \rangle$
(e) $\langle r, s, t, u, v \mid r^2, s^2, t^2, u^2, v^2, (rs)^2, (st)^2, (tu)^4, (uv)^2, (rv)^4 \rangle$

Exercise 11. Cayley graphs for Coxeter groups: Recall the construction of Cayley graphs described in Section 2.2. Note that the generators of a Coxeter group always have order 2, so at each vertex of the Cayley graph of a Coxeter group, the incoming and outgoing edges are identified to reflect the idea that for a given generator r, $r = r^{-1}$. Construct the Cayley graph for the groups in Examples 7,8, and 9. Do the same for the groups presented in Exercises 9 and 10.

Exercise 12. By using an argument similar to that in Example 6, identify each of the presented groups with a familiar group. Namely the student should attempt to find a map from an appropriate free group to a familiar group, then show that this map descends to an isomorphism of the given presented group to the familiar group.

(a) $\langle r, s \mid r^2, s^2, (rs)^3 \rangle$
(b) $\langle r, s \mid r^2, s^2, (rs)^n \rangle$ (See Example 7)
(c) $\langle r, s, t \mid r^2, s^2, t^2, (rs)^2, (st)^2, (rt)^2 \rangle$ (See Example 8)
(e) $\langle r, s, t \mid r^2, s^2, t^2, (rs)^3, (st)^3, (rt)^2 \rangle$ (Hint: This finite Coxeter group is quite famous. Make an educated guess at its order and try to think of a group with the same order.)

Challenge Problem 3. Given a Coxeter graph $\Gamma = (V, E)$, show that an automorphism of Γ which preserves edge labels (that is, if m is the label on $\{r, s\}$, then m is the label on the edge defined by the images of r and s) induces an automorphism of the corresponding Coxeter group.

Challenge Problem 4. Consider the formulas for reflections in Euclidean, spherical, and hyperbolic space given in equations 2, 3, 4, and 5. Let r_l generically denote one of these reflections. Show that

(a) r_l is appropriately defined for the non-Euclidean models. (Specifically, show that it sends points on the sphere to points on the sphere, points on the hyperboloid to points on the hyperboloid, and points in the interior of the unit disk, to points in the interior of the unit disk.)
(b) For all the models, verify that $r_l \circ r_l =$ the identity map on the space.

Challenge Problem 5. Consider again the 2-dimensional geometric models discussed above.

(a) Find a presentation for the group generated by a reflection over the x-axis, and the line $y = \sqrt{3}x$ in Euclidean space.
(b) Find a presentation for the group generated by reflections over the lines formed by the planes $z = 0$, $y = 0$ and $y = x$ in $\mathbb{S}^2$.
(c) Find a presentation for the group generated by reflections over the lines formed by the planes $y = 0$ and $y = x$ in $\mathbb{H}^2$.
(d) Find a presentation for the group generated by reflections over the x-axis, the line $y = x$, and the hyperbolic line connecting the points

$$\left(\frac{\sqrt{\cos^3\frac{\pi}{4}}}{\cos\frac{\pi}{8}}, 0\right) \quad \text{and} \quad \left(\frac{\sqrt{\cos^5\frac{\pi}{4}}}{\cos\frac{\pi}{8}}, \frac{\sqrt{\cos^5\frac{\pi}{4}}}{\cos\frac{\pi}{8}}\right)$$

in $\mathbb{H}^2_P$.

For each of these, besides considering the angle between the given reflections as a clue to appropriate group element orders, the student is also encouraged to use a computer algebra system to calculate the order of the composition of reflections directly.

4 Group Actions on Complexes

In Geometric Group Theory, the idea is to study the interplay between a finitely presented group G, and a corresponding topological, or geometric, space X. In particular, we view the elements of a group G as *homeomorphisms* of the space X.

That is, each element of the group corresponds to a continuous, bijective function $X \to X$. For example, if $G = D_3$, the dihedral group of order 6, then each element of the group can be viewed as a function that maps a regular 3-gon X to itself. Perhaps, the element reflects the triangle over an altitude, or rotates the triangle by $120°$. In any case, each element of D_3 is viewed as a function of the triangle to itself.

The language we use to encompass this sort of relationship between a group G and a topological space X is to say that the group G *acts* on X. Group actions are one of the richest areas of mathematics, for if the action is appropriate, many of the properties of the group (orders of elements, subgroups, cosets, etc.) manifest themselves in the topological space; and properties of the topological space (e.g. any geometric structure) manifest themselves in the group. In this chapter, the approach we take is to focus on finitely presented groups, mostly Coxeter groups, acting on a special type of topological space called a *CW-complex*. For an overview of Geometric Group Theory, see [2]. For more references on topological spaces, see [12], and for references on CW-complexes, see [8] or [7].

4.1 CW-Complexes

CW-complexes are topological spaces whose construction can be done in a step-by-step manner, using very simple topological spaces as building blocks. The blocks are referred to as n-balls, and the steps in the process correspond to dimension. The idea is to build a space out of points, then edges, then disks/squares/triangles/5-gons/etc., then tetrahedron/cubes/prisms/etc.,... and so on. Let us get a bit more specific.

Let D^n denote a *topological n-ball*, with *boundary* $\partial D^n = S^{n-1}$. In particular,

- D^0 is just a point, S^{-1} is the empty set.
- D^1 is just a line segment, S^0 is the two endpoints of the edge.
- D^2 is a disk, S^1 is the circle bounding the disk.
- D^3 is a (filled) ball, S^2 is the surface of the ball.

$$\vdots$$

In general, D^n can be viewed as the set $\{(x_1, x_2, x_3, \ldots, x_n) \in \mathbb{R}^n \mid x_1^2 + x_2^2 + x_3^2 + \ldots + x_n^2 \leq 1\}$, with boundary S^{n-1} viewed as the set of points $\{(x_1, x_2, x_3, \ldots, x_n) \in \mathbb{R}^n \mid x_1^2 + x_2^2 + x_3^2 + \ldots + x_n^2 = 1\}$. It is important to note, however, that these balls and their boundaries are thought to be completely independent of any sort of Cartesian space. They are considered to be their own spaces. This isn't difficult to picture in low dimensions, for it is easy to think of collections of points, segments, disks, and balls as independent of any sort of coordinate axes. Though harder to picture (or even impossible?), the same applies in higher dimensions. But for the sake of this chapter, picturing things in dimensions ≤ 3 will suffice.

We should also note that the n-balls used as building blocks do not have to be round. For example, D^2 could be a traditional disk, or it could be a square, or a pentagon, or a very strange non-convex shape. See Figure 5. These are all

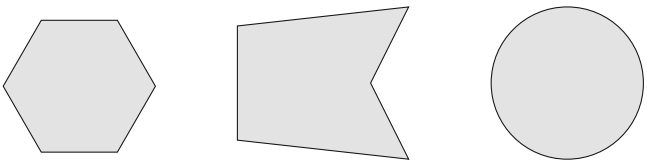


Fig. 5 Homeomorphic examples of a 2-ball.

topological 2-balls because any one of them can be stretched, bent, squashed, etc…, but not torn; in a such a way as to match any of the others. This is the idea of a homeomorphism. The study of homeomorphisms in topology is as central as the study of isomorphisms in group theory, or even as central as the study of differentiable functions in Calculus I. For a nice introductory reference on the centrality of homeomorphisms to the study of topology, see [11].

The idea of a CW-complex, then, is to take n-balls of various dimensions, and use them as the building blocks to construct a topological space. It's almost as if we are playing LEGOs®, and the n-balls are our pieces. Within the context of the larger space, we refer to the individual n-balls as *n-cells*. Now, just as a LEGO® construction can be given in steps, the construction of a CW-complex X, can be described in steps, corresponding to dimension:

0. Start with a (discrete) set of 0-cells. Denote this set X^0, called the 0-skeleton.
1. To X^0, attach a set of 1-cells along their boundaries; forming X^1 called the 1-skeleton. (At this stage, your complex looks like a graph.)
2. To X^1, attach a set of 2-cells along their boundaries; forming X^2 called the 2-skeleton.
3. To X^2, attach a set of 3-cells along their boundaries; forming X^3 called the 3-skeleton.

$$\vdots$$

The "attaching" described above should be described a bit more formally. Inductively, for $n \geq 1$, we form the n-skeleton X^n from the $(n-1)$-skeleton X^{n-1} by attaching n-balls via functions $f : \partial D^n = S^{n-1} \to X^{n-1}$ (called attaching maps), one function for each of the attached n-balls mapping its boundary to the lower dimensional skeleton. If $\{D^n\}$ is the collection of n-balls attached at step n, then the n-skeleton X^n is understood as the disjoint union $X^{n-1} \cup \{D^n\}$ under some identifications. In particular, for each attached n-ball D^n, each point x in the boundary is identified with its image under f. In other words, new cells are "glued" along their boundaries to the existing space. More formally, we have

$$X^n := \left(X^{n-1} \cup \{D^n\}\right) / \sim,$$

where $x \sim f(x)$ for each attached n-ball D^n, each x in ∂D^n, and corresponding attaching map f. Precisely, X^n is defined as a *quotient space*. Set $X = \cup_n X^n$, the

union of all n-skeleta. It is a CW-complex. The "CW" stands for "Closure-Weak" in reference to the topology of such a space. We will not get specific with the topology of these spaces, only noting that each cell carries its own topology homeomorphic to the unit ball in $\mathbb{R}^n$.

If the process described above stops at some dimension n (that is, if there are no m-balls attached for $m > n$), than we say X is finite dimensional. In particular, if the process stops after X^1, that is $X = X^1$ and there are no m-balls for $m > 1$, then X is a graph with vertices the 0-cells, and edges the 1-cells of X. (It may help to think of CW-complexes as graphs generalized to higher dimensions.)

If X contains finitely many cells, we say X is finite. Note, however, that in any given step, there may be infinitely many cells, or there could be no cells to attach. While we will look at some constructions in Examples 10 and 11, we often are given the total space X, and understand the above process to have taken place, and refer to the resulting *cellulation* of X.

Example 10. We can view a 2-sphere S^2 as a CW-complex in many ways. One way is with one 0-cell, no 1-cells, and one 2-cell attached to the 0-cell by identifying all of its boundary to the 0-cell – like pulling a draw string on a bag to tighten the opening. The attaching map is indicated with a solid arrow in Figure 6. Here we see that since there are no 1-cells, $X^0 = X^1$. In general, it can be the case in a CW-complex that the i-skeleton can equal the $i + 1$-skeleton.

Example 11. Figure 7 depicts a cellulation of a torus by one 0-cell, two 1-cells, and one 2-cell. The attaching maps are indicated with solid arrows in the figure, resulting skeleta indicated with dashed arrows. Two 1-cells are attached to the indicated 0-cell. One 2-cell, viewed as a rectangle, is attached to the 1-skeleton where the corners are all identified with the 0-cell, and the edges are identified to the 1-cells with orientations as shown. (Though it is not shaded, the rectangle shown represents a 2-cell.)

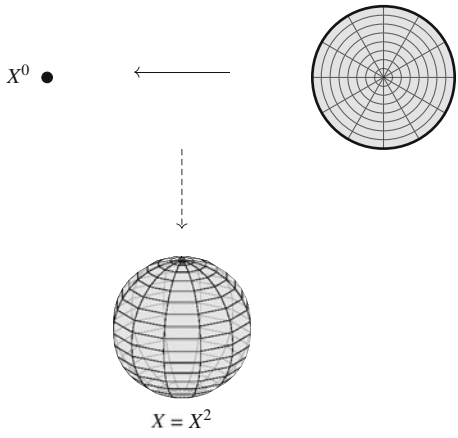


Fig. 6 A cellulation of a 2-sphere.

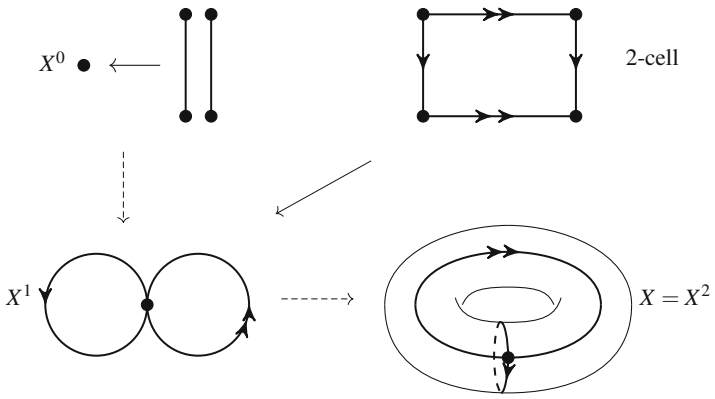


Fig. 7 The cellular decomposition of the torus.

The Euler Characteristic To a finite CW-complex X we can attach a number called the *Euler characteristic* of X, denoted $\chi(X)$, where

$$\chi(X) = \sum_{\text{cells } \sigma} (-1)^{\dim \sigma}. \tag{6}$$

That is, $\chi(X)$ is the alternating sum of the number of cells in each dimension. It is an interesting result of Algebraic Topology that the Euler characteristic of a space X does not depend on the specific cellulation one uses, rather only on the *homotopy class* of the space. In particular, homeomorphic spaces (regardless of cellulation) will have the same Euler characteristic. See [8] for reference. For example, using the cellulation of the sphere in Example 10, we get $\chi(S^2) = 2$. But it is the case that no matter the cellulation of S^2, we still get $\chi(S^2) = 2$. In fact, it is a fundamental result of algebraic topology that the Euler characteristic classifies surfaces, both orientable and non-orientable. The interested student should refer to Project Idea 3 that further investigates the Euler characteristic.

4.2 Group Actions on CW-Complexes

As discussed in the introductory comments of Section 4, an important aspect of group theory (and in fact all of mathematics) is the study of *group actions*. In an undergraduate abstract algebra course, you may have studied group actions on generic sets, or in the context of the Sylow Theorems. The automorphism groups of graphs are commonly studied in this environment, as group elements can be viewed as bijections of the vertex set or edge set of an associated graph. Some students may have even studied how a group can "act" on its own Cayley graph. But as we noted above, graphs are just 1-dimensional CW-complexes, and hence topological spaces.

So a group "acting" on a graph is really an example of the more general sort of group action we study here. That is, a group acting on a topological space.

Definition 3. An *action* of a group G on a topological space X is a homomorphism $\phi : G \to \mathrm{Homeo}(X)$. Where $\mathrm{Homeo}(X)$ is the set of homeomorphisms $X \to X$, under composition.

Once again, this is very similar to the definition of group action you may see in an abstract algebra text like [6], however, instead of each element of the group corresponding to simply a bijection of a set X, we carry a topological requirement that each element $g \in G$ corresponds to a homeomorphism $\phi_g : X \to X$. For $g \in G$, we write $g \cdot x$ for $\phi_g(x)$. That is, we think of g itself as sending $x \in X$ to some other point of the space. Similarly, for a subset $Y \subseteq X$, $g \cdot Y$ stands for "the points to which g sends all of the points of the set Y."

Since the topological spaces we consider in this chapter are CW-complexes, we require our actions be *cellular*. That is, we consider the action on the level of cells and require that, for $g \in G$, and for any n-cell σ in X, $g \cdot \sigma$ is another n-cell of X. So a cellular action is one that sends n-cells to n-cells, and all adjacency relationships are maintained. In other words, if two cells are adjacent before being acted upon, then they are adjacent after being acted upon. In this context, the familiar definitions of *orbit* and *stabilizer* take on this cellular theme.

Definition 4. For an n-cell σ, the *stabilizer* of σ is

$$\mathrm{Stab}_G(\sigma) = \{g \in G \mid g \cdot \sigma = \sigma\}$$

and the *G-orbit* of σ is the set of n-cells γ for which $\tau = g \cdot \sigma$ for some $g \in G$.

In the examples we consider, the actions are also be *proper* and *co-compact.*

Definition 5. An action of a group G on a complex X is *proper* if $|\mathrm{Stab}_G(\sigma)| < \infty$ for each cell σ of X.

The next term we would like to define, "co-compact", requires some set-up. For a cellular action of a group G on a complex X, we define the *quotient space* X/G to be a CW-complex where each cell σ of X is identified with its orbit. There are some topological issues that we are bypassing, but for reference, the space X/G is the endowed with the *quotient topology*. (See [12] or [8].) Put simply, X/G is a CW-complex comprised of an n-cell representing each G-orbit of n-cells (for each n), and containment relationships from the parent space X are maintained: If two n-cells are in the same orbit, then they will identify as a single cell in X/G. If an n-cell σ contains an m-cell τ in X, then they will have corresponding n- and m-cells in the same containment relationship in X/G. We are now ready to define a co-compact action.

Definition 6. An action of a group G on a complex X is *co-compact* if the quotient space X/G is a finite complex.

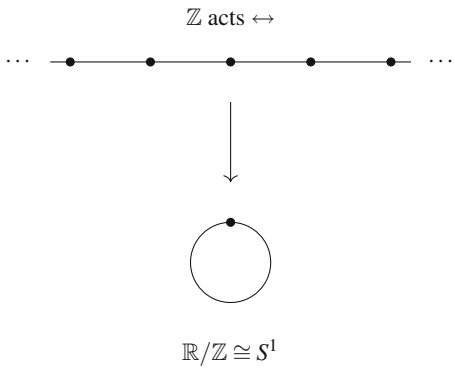


Fig. 8 $\mathbb{Z}$ acts via translations, $\mathbb{R}/\mathbb{Z}$ is shown.

Before exploring some examples, we have one last, closely related, term to define for the action of a group G on a complex X.

Definition 7. Let G act on the CW-complex X. A closed subset C of X is a *fundamental domain* for the action if $G \cdot C$, the union of all orbits of cells in C, contains X. A fundamental domain C of X is a *strict* fundamental domain if the G-orbit of each cell intersects C in exactly one cell.

Example 12. The real line $\mathbb{R}$ can be thought of as a CW-complex, where each integer point on the line corresponds to a 0-cell, and each resulting interval corresponds to a 1-cell attached at two endpoints. The group $(\mathbb{Z}, +)$ acts on $\mathbb{R}$ by translation. For example, if we take $5 \in \mathbb{Z}$, and $x \in \mathbb{R}$, $5 \cdot x = 5 + x$. That is, the action of 5 on $\mathbb{R}$ slides every point on the line 5 units right. $-3 \in \mathbb{Z}$ slides each point 3 units left. This action is cellular, with (non-strict) fundamental domain any interval; it is proper, the only cell-stabilizer is the identity $0 \in \mathbb{Z}$; and it is co-compact. There is one orbit of 0-cells and one orbit of 1-cells. Thus, the space $\mathbb{R}/\mathbb{Z}$ should consist of exactly two cells: one 0-cell, and one 1-cell. And, since each 1-cell is connected to 0-cells on both ends, in the quotient space, the one 1-cell must be connected to the one 0-cell at both ends. Thus, $\mathbb{R}/\mathbb{Z}$ is a circle as shown in Figure 8.

The idea of a strict fundamental domain of an action is that it is a subset of the space whose orbit covers the whole space, but does so efficiently. For example, the interval $[0, 1]$ is a fundamental domain for the action of $\mathbb{Z}$ on $\mathbb{R}$ described in example 12, but it is not strict, since the endpoints of the interval are in the same orbit. A similar situation occurs in the next example.

Example 13. The real plane $\mathbb{R} \times \mathbb{R}$ can be thought of as a CW-complex, where each integer grid point corresponds to a 0-cell, horizontal and vertical segments connecting the grid points correspond to 1-cells, and 2-cells correspond to the resulting squares. $\mathbb{Z} \times \mathbb{Z}$ acts on $\mathbb{R} \times \mathbb{R}$ by horizontal and vertical translation. For example, for $(3, -2) \in \mathbb{Z} \times \mathbb{Z}$ and $(x, y) \in \mathbb{R} \times \mathbb{R}$, $(3, -2) \cdot (x, y) = (3 + x, -2 + y)$. This action is cellular, proper, and co-compact. To see the quotient space, realize that the square of the form $[0, 1] \times [0, 1]$ can be translated to cover the whole plane.

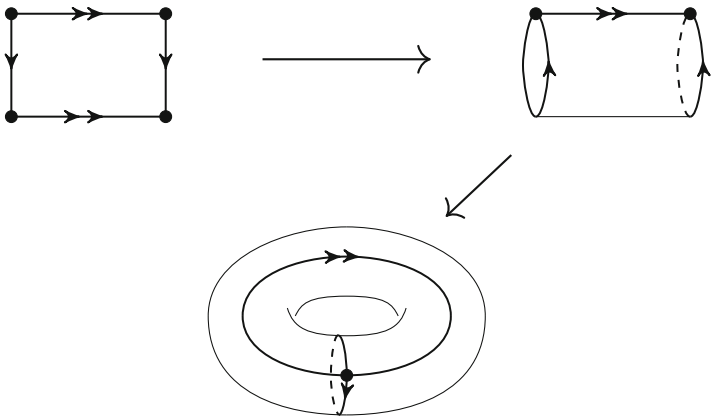


Fig. 9 $\mathbb{R}/(\mathbb{Z} \times \mathbb{Z}) \cong$ a torus.

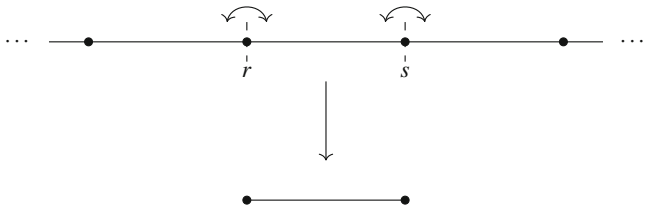


Fig. 10 D_∞ acting on $\mathbb{R}$ by reflections. $\mathbb{R}/D_\infty$ is shown.

That is, this square is a fundamental domain of the action, but it is not strict. There is one orbit of 2-cells, and thus one 2-cell in the quotient space. Within this one square, the top and bottom edges are identified, as they are within the same orbit, and the left and right edges are identified, as they are within the same orbit. However, the horizontal edges are not translated to the vertical edges under this action. So there are two orbits of 1-cells and thus two 1-cells in the quotient space. Finally, the four corners are identified as a single point, as there is one orbit of 0-cells, and so the quotient space has one 0-cell. $(\mathbb{R} \times \mathbb{R})/(\mathbb{Z} \times \mathbb{Z})$ is the torus, the identifications shown in Figure 9. Note that this is a slightly different perspective on the cellulation of the torus described in Example 11.

Example 14. The infinite dihedral group $D_\infty = \langle r, s \mid r^2, s^2 \rangle$ acts on $\mathbb{R}$ where we take r to correspond to a reflection about $0 \in \mathbb{R}$, and s to be a reflection about $1 \in \mathbb{R}$. See Figure 10. This action is cellular, proper (the stabilizers of the 0-cells have order 2), and co-compact. There are two orbits of 0-cells, one orbit of 1-cells. The quotient space then is a closed interval. Note that it can be identified with a strict fundamental domain of the action.

Orbihedral Euler Characteristic If a group G acts properly and co-compactly on a complex X, then the *orbihedral Euler characteristic* of X/G is the rational number

$$\chi^{\text{orb}}(X/G) = \sum_{\sigma} \frac{(-1)^{\dim \sigma}}{|\text{Stab}_G(\sigma)|}, \tag{7}$$

where the sum is over the cells of X/G. (See [4] or [14] for reference on the orbihedral Euler characteristic.) Note that (1) The orbihedral Euler characteristic is the usual Euler characteristic in the case all cell stabilizers are trivial and (2) The orbihedral Euler characteristic is multiplicative. That is, if $H \leq G$ of index m, then

$$\chi^{\text{orb}}(X/H) = m\chi^{\text{orb}}(X/G). \tag{8}$$

(See Exercise 15 below.)

In Example 14 with $\mathbb{R}/D_\infty$, we have a 1-cell stabilized by the trivial subgroup, and two 0-cells stabilized by order two subgroups. So

$$\chi^{\text{orb}}(\mathbb{R}/D_\infty) = \underbrace{1/2 + 1/2}_{0-\text{cells}} - \underbrace{1}_{1-\text{cell}} = 0.$$

Exercise 13. Calculate $\chi(X)$ in the case X is

(a) An empty tetrahedron.
(b) An empty octahedron.
(c) An empty cube.
(d) The torus described in Example 11.

The cube, octahedron, and tetrahedron are "regular" cellulations of $\mathbb{S}^2$. Can you think of any others? Calculate the corresponding Euler characteristic. Sketch an non-regular cellulation of $\mathbb{S}^2$ and calculate χ.

Exercise 14. Calculate $\chi^{orb}(X/G)$ for each of the quotient spaces in Examples 12, 13, and 14.

Exercise 15. Prove Equation 8 above: If $H \leq G$ of index m, then $\chi^{\text{orb}}(X/H) = m\chi^{\text{orb}}(X/G)$.

Exercise 16. Using the idea of Example 13, describe an action of $D_\infty \times D_\infty$ on $\mathbb{R} \times \mathbb{R}$. Sketch the quotient space $\mathbb{R} \times \mathbb{R}/(D_\infty \times D_\infty)$ and calculate

$$\chi^{\text{orb}}\left((\mathbb{R} \times \mathbb{R})/(D\infty \times D\infty)\right).$$

Exercise 17. A group $G = \langle A \mid R \rangle$ acts on its Cayley graph Γ in the following way: For $g \in G$, v a vertex (which is also an element of the group), we have $g \cdot v = gv$. This is a generalization of the (left) action of a group on itself. Verify that this defines

a cellular action on the Cayley graph. (That is, edges are sent to edges). Consider the following examples.

(a) Describe the action of $F_{a,b} = \langle a, b \mid \ \rangle$ on its Cayley graph Γ. Sketch $\Gamma/F_{a,b}$.
(b) Describe the action of $G = \langle a, b \mid aba^{-1}b^{-1} \rangle$ on its Cayley graph Γ. Sketch Γ/G. (Compare with Example 13).
(c) Describe the action of $W = \langle r, s \mid r^2, s^2, (rs)^3 \rangle$ on its Cayley graph Γ. Sketch Γ/W.
(d) Describe the action of $W = \langle r, s, t \mid r^2, s^2, t^2, (rs)^2, (st)^2, (rt)^2 \rangle$ on its Cayley graph Γ. Sketch Γ/W.
(e) Describe the action of

$$W = \langle r, s, t, u, v \mid r^2, s^2, t^2, u^2, v^2, (rs)^2, (st)^2, (tu)^2, (uv)^2, (rv)^2 \rangle$$

on its Cayley graph Γ. Sketch Γ/W.

Exercise 18. Look up a cellular decomposition of a two-holed torus X (sometimes called a genus 2 surface). It can be understood as an "identification space" similar to the torus in Example 13, though beginning with an octagon rather than a rectangle. Calculate $\chi(X)$. Do the same with a "Klein bottle" and any "genus g-surface."

Challenge Problem 6. Consider the element $rs \in D_\infty$ and the subgroup generated by it, $\langle rs \rangle \leq D_\infty$.

(a) What is the order of $\langle rs \rangle$?
(b) What is the index of $\langle rs \rangle$ in D_∞.
(c) Under the action of D_∞ on $\mathbb{R}$ described in example 10, describe the action of the element rs on $\mathbb{R}$. That is, for $x \in \mathbb{R}$, what is $(rs) \cdot x$? What is $(rs)^{-1} \cdot x$? Deduce the action of $(rs)^n$ on $\mathbb{R}$.
(d) Sketch the quotient space $\mathbb{R}/\langle rs \rangle$ and verify equation 8.

The group $\langle rs \rangle$ is called a *finite index torsion free subgroup* of D_∞. This means that it contains no elements of finite order, besides the identity. Project idea 3 asks the student to consider such subgroups and similar questions in the context of different Coxeter groups.

Challenge Problem 7. Repeat Problem 6 for $D_\infty \times D_\infty$ acting on $\mathbb{R} \times \mathbb{R}$. That is, find a finite index subgroup in $D_\infty \times D_\infty$ acting on $\mathbb{R} \times \mathbb{R}$, and answer the included questions.

5 The Cellular Actions of Coxeter Groups: The Davis Complex

In several papers (e.g., [3], [4], and [5]), M. Davis describes a construction which associates to any Coxeter system (W, S), a complex $\Sigma(W, S)$, or simply Σ when the Coxeter system is clear, on which W acts properly and co-compactly. This is the Davis complex. We describe the construction here.

5.1 Spherical Subsets and the Strict Fundamental Domain

Let (W, S) be a Coxeter system with defining graph Γ. For a subset of generators U, denote by W_U the subgroup of W generated by the elements of U. Of interest are subsets of generators (vertices of the graph) that generate finite groups. We call these *spherical subsets*. These spherical subsets will be the key to defining an action of the corresponding Coxeter group on a complex.

5.1.1 Spherical Subsets

Finite Coxeter groups are completely classified, codified by the so-called "Dynkin diagrams;" and in general, one can detect if a given subset of generators of a Coxeter group defines a finite subgroup. But for us to work through our low-dimensional examples (dimension ≤ 3), we need only detect spherical subsets with three or fewer elements, and we can do that in a way that doesn't directly rely on knowing Dynkin diagrams.

Let (W, S) be a Coxeter system with corresponding Coxeter graph Γ. First note that every vertex of Γ corresponds to an order 2 generator, so every vertex defines a spherical subset of order 1. Next, recall that any two vertices connected by an edge generate a finite group, so all edges define a spherical subset of order 2. Furthermore, these are the only spherical subsets of order 2, since any two vertices not connected by an edge generate D_∞. Also, from this we deduce that the vertices of *any* spherical subset must be pairwise connected. Finally, we present the following fact for spherical subsets of order 3: For pairwise connected vertices r, s, and t of Γ with edge labels m_{rs}, m_{st}, and m_{rt}, the subgroup $\{r, s, t\}$ is finite if and only if

$$\frac{1}{m_{rs}} + \frac{1}{m_{st}} + \frac{1}{m_{rt}} > 1.$$

The reader is invited to check this fact against the Coxeter group examples and exercises worked at the end of Section 3, and investigate the geometric implications of such an inequality.

5.1.2 The Strict Fundamental Domain

With (W, S) a Coxeter system with Coxeter graph Γ, we define a finite complex K that will be the strict fundamental domain of the action of W on the Davis complex. The perspective we take is in some ways the inverse of the process laid out in Section 4.2, where given a group acting on a space X, we calculated the quotient space X/G. Here, we will construct the strict fundamental domain first and use it

to construct a space on which the Coxeter group acts. We do this by 'reflecting' K around in a tiling pattern that respects the structure of the group. Consider Example 14. There we have a Coxeter group acting on the real line, with (strict) fundamental domain an interval. Note that we can turn the perspective around, starting with the interval between the reflection points and reflect it repeatedly over its endpoints and develop (or cover) the full real line. The real line in this example, with the cellulation and indicated action, is the Davis complex corresponding to the Coxeter group D_∞. (A fact the reader should confirm after the construction process is spelled out below.) In this case, as will be the case with all of the Davis complex examples, that the strict fundamental domain corresponds to the quotient space of the action.

The strict fundamental domain K is a complex with the following cell structure:

- 0-cells: Each spherical subset, including $\emptyset$, corresponds to a 0-cell.
- 1-cells: If U and V are spherical subsets with $U \subseteq V$, then this inclusion relationship corresponds to a 1-cell connecting vertices U and V.
- 2-cells: If U, V, and W are spherical subsets with $U \subseteq V \subseteq W$, then this inclusion relationship corresponds to a 2-cell attached to the edges $U \subseteq V$, $V \subseteq W$, and $U \subseteq W$. (It is a triangle with edges corresponding to $U \subseteq V$, $V \subseteq W$, and $U \subseteq W$.)
- 3-cells: If U, V, W, Y are spherical subsets with $U \subseteq V \subseteq W \subseteq Y$, then this relationship corresponds to a 3-cell connected appropriately. (It is a tetrahedron with triangle faces corresponding to the 4 triangles determined by the previous step.)

$\vdots$

In general, this process terminates and one forms the finite complex K. This process actually forms K into what is called a *simplicial complex*, which is a special type of CW-complex. We will also say that K is the *geometric realization* of the partially ordered set of spherical subsets. Figure 11 shows K for the group in Example 9.

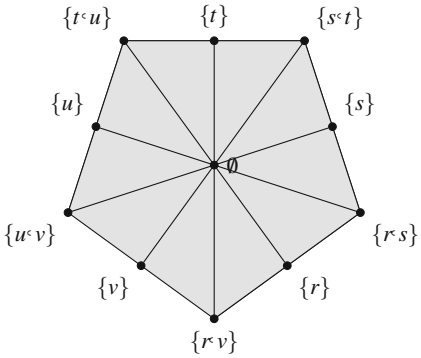


Fig. 11 K for the group defined in Example 9.

5.2 The Davis Complex

We are now ready for the construction of the Davis complex for a Coxeter system (W, S), with strict fundamental domain K.

For each generator $s \in S$, the set of cells of K for which each corresponding spherical subset contains $\{s\}$ is called the *s-mirror*. (In Figure 11, these 'mirrors' correspond to five 'faces' of K.) We form the Davis complex Σ by taking a copy of K for each element of the group W and "gluing" them together according to mirrors. This corresponds to "reflecting" K in the associated mirror. More formally, we define an equivalence relation $\sim$ on the set $W \times K$ (a copy of K for each element of the group) by $(w, x) \sim (v, y)$ if and only if for some generator s, $x = y$ in the s-mirror of K and $v = ws$. Then define the Davis complex Σ as the quotient space

$$\Sigma = (W \times K)/\sim . \tag{9}$$

See [4] for reference. The (left) W-action on the set $W \times K$ defined by $w \cdot (v, k) = (wv, k)$ respects the equivalence relation and passes to a cellular action on Σ with strict fundamental domain K. (See Exercise 20.)

We write wK for the union of all the equivalence classes of elements in $w \times K$. This notation is indicative of the group action, as wK stands for the w-translate of the copy of K corresponding to the identity element of the Coxeter group. Thus, for any generator s, in the equivalence relation above wK is 'glued' to wsK along the s-mirror.

Example 15. Let Γ be a segment labeled with 3. Then

$$W = \langle r, s \mid r^2, s^2, (rs)^3 \rangle$$

and the Davis Complex is the hexagon shown in Figure 12. In this case, the Davis complex is 2-dimensional, so all of the hexagon is "filled in," but for emphasis, we only shade the strict fundamental domain K. In the figure, the center vertex corresponds to the spherical subset $\{r, s\}$ and is thus in both the r- and s-mirrors. So all copies of K are glued at this point. Finally, note that since $rsr = srs$ in W, $rsrK = srsK$ in Σ. To better understand the term "mirror," observe that Σ can be developed by reflecting K repeatedly in the lines formed by extending the r- and s-mirrors. Indeed, one should trace the orbit of the vertex denoted $\emptyset$ under these reflections.

5.3 The Mirror Cellulation of Σ

If a defining Coxeter graph Γ is an m-gon, then we have simpler cellulation of K that makes more clear the term "mirror." So let Γ be an m-gon and take as the set of 0-cells the spherical subsets of the form $\{r, s\}$, that is, pairs of adjacent vertices in Γ. For each generator r, the spherical subset $\{r\}$ is included in exactly two other

spherical subsets, namely $\{r, s\}$ and $\{r, t\}$ where s and t are the adjacent to r in Γ. We then declare the two edges $\{r\} \subseteq \{r, s\}$ and $\{r\} \subseteq \{r, t\}$ forming the r-mirror as one 1-cell with endpoints attached to $\{r, s\}$ and $\{r, t\}$. Take these as the 1-cells of this new cellulation of K. Finally, use one 2-cell to represent the interior. This is a coarser, but simpler cellulation of K than that described above. The full Davis complex is formed as above, with the same identifications along mirrors. The group action on the finer cellulation described above descends to a cellular action here.

Example 16. Let W be the Coxeter group defined in Example 9. We simplify the cell structure of the strict fundamental domain shown in Figure 11 by taking five 0-cells, five 1-cells, and one 2-cell. Now, using this cellulation, we think of developing the space Σ by reflecting K successively in each of its mirrors, respecting the group relations. Namely reflecting K by r and then s is equivalent to reflecting by s and then r since $(rs)^2 = 1$ implies $rs = sr$ in W. This yields a copy of K for each element of the group, as before, with the reflection actions very clearly defined along the faces of K. See Figure 13 for an incomplete image of Σ, drawn suggestively in the Poincaré disk model of hyperbolic space. Note that reflections in the five bold lines correspond to the five generators of the group, and the action of W on Σ is achieved by reflecting over these lines. For reference on this simpler cellulation of Σ, see [13]. For an investigation as to why this image is depicted in hyperbolic space, see Project 2.

5.4 The Coxeter Cellulation

We finally present a cellulation of the Davis Complex that is a generalization of the Cayley graph. We will see that for a given system (W, S), the 1-skeleton of this cellulation of the corresponding Davis Complex is the (non-directed) Cayley graph of (W, S). Moreover, the combinatorial data from the defining Coxeter graph will be on display. We should be clear: We are not constructing a new space. This is a different cellulation of the space Σ under the same group action.

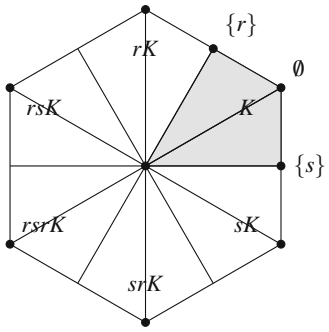


Fig. 12 Σ in the case $W = D_3$, the dihedral group of order 6, with fundamental chamber $K = \Sigma/W$ shaded.

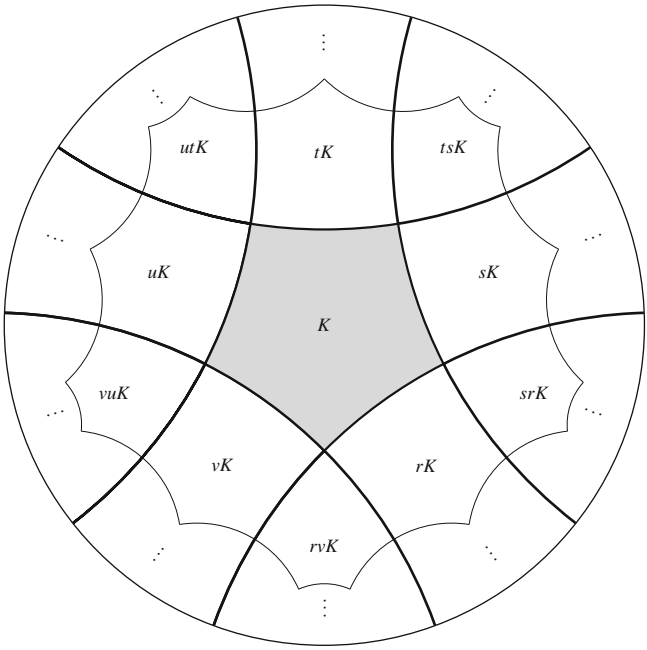


Fig. 13 $\Sigma \cong \mathbb{H}^2$, with fundamental chamber $K = \Sigma/W$ shaded.

5.4.1 Euclidean Representations

Let (W, S) be a Coxeter system. For each spherical subset $T \subseteq S$, there is an action of W_T on $\mathbb{R}^{|T|}$. We examine some low-dimensional examples:

- $T = \emptyset$: Then $W_T = \{1\}$, and we have a trivial action of the identity group on a point ($\mathbb{R}^0$ is a point).
- $T = \{s\}$, for some $s \in S$: Then $W_T \cong \mathbb{Z}_2$, and we have a reflection action on $\mathbb{R}^1$. Considering a reflection about the point $x = 0$, the non-negative real line forms a strict fundamental domain.
- $T = \{s, t\}$, for some $s, t \in S$ with edge label m_{st}: Then $W_T \cong D_n$, and we have a group action on $\mathbb{R}^2$. Specifically, consider reflections over the x-axis, say this corresponds to s, and a reflection corresponding to t over the line through the origin making an angle with π/m_{st} with the x-axis. The student should verify that the order of the composition of these reflections is m_{st}. The region between, and including, the two reflection lines is a strict fundamental domain for this action.
- $T = \{r, s, t\}$, for some $r, s, t \in S$: This is left for the reader.

$\vdots$

5.4.2 The Coxeter Cell of Type T

Now, given a spherical subset T, and a representation like that described above, we form the *Coxeter cell of type T*: Begin with a point in the interior of the strict fundamental domain described above, and calculate its orbit. The points in the orbit are in 1–1 correspondence with the elements of W_T. We then take the *convex hull* of the resulting points. This means we form a CW-complex where the 0-cells are the points, 1-cells are edges between these points, 2-cells are attached to cycles of edges, and so on. The resulting space is a cellulation of a topological $|T|$-ball we call a Coxeter cell of type T. In the examples we considered above, the student should confirm that we have:

- $T = \emptyset$: A Coxeter cell of type $\emptyset$ is a point.
- $T = \{s\}$, for some $s \in S$: A Coxeter cell of type $\{s\}$ is an edge.
- $T = \{s, t\}$, for some $s, t \in S$ with edge label m_{st}: A Coxeter cell of type $\{s, t\}$ is a filled-in $(2m_{st})$-gon.
- $T = \{r, s, t\}$, for some $r, s, t \in S$: A Coxeter cell of type $\{r, s, t\}$ is a 3-ball. A solid polytope. The details are left to the reader.

  $\vdots$

We then have a new cellulation of Σ called the *Coxeter cellulation*. To emphasize the Davis complex equipped with this cellulation, we write Σ_{cc}. The n-cells are the Coxeter cells of type T where $|T| = n$, one cell of type T for each coset of W_T in W. Specifically, the 0-cells of Σ_{cc} correspond to the cosets of $W_\emptyset$, i.e. the group elements. The 1-cells correspond to the cosets of W_s, for each generator s. (This means that an edge of type $\{s\}$ has endpoints w and ws, for arbitrary $w \in W$). The 2-cells correspond to cosets of $W_{\{s,t\}}$, etc. As mentioned above there is a nice relationship between this cellulation, and the defining graph Γ. The student is invited to explore the Coxeter cellulation in Project 1.

Example 17. Consider the case $W = \langle r, s \mid r^2, s^2, (rs)^3 \rangle$ with Σ shown in Figure 12. The Coxeter cellulation of the same space has six 0-cells, corresponding to the translates of the 0-cell corresponding to $\emptyset$; six 1-cells, coming in two types, $\{r\}$ and $\{s\}$, connecting the 0-cells; and one 2-cell of type $\{r, s\}$. The student should calculate the cosets of $W_{\{r\}}$ and $W_{\{s\}}$ in W, and check these against the vertex sets of edges in Σ_{cc}. Finally, note that the entirety of Σ_{cc} is a prototype for a Coxeter cell of type $\{r, s\}$ when $m_{rs} = 3$.

Exercise 19. Show that the relation $\sim$ on $W \times K$ described above generates an equivalence relation.

Exercise 20. Show that if $(v, x) \sim (u, y)$ as described above, then for any $w \in W$, $w \cdot (v, x) \sim w \cdot (u, y)$. That is, show that the left action of W on $W \times K$ respects the equivalence relation and descends to an action of W on Σ. Show that this action is cellular, proper, co-compact, and has strict fundamental domain K. Conclude that $\Sigma/W = K$.

Exercise 21. Calculate $\chi^{\text{orb}}(K)$ for the K depicted in Figure 13. Calculate $\chi^{\text{orb}}(K)$ for K depicted in Figure 11. Do you get the same answer?

Exercise 22. Sketch the Davis complex Σ associated to the infinite dihedral group D_∞. Sketch the Cayley graph associated to D_∞ and consider the action of D_∞ on this graph. How does it compare to Σ?

Exercise 23. Sketch the Davis complex Σ associated to each of the Coxeter graphs in Figure 4. Also consider the case Γ is three points, no edges.

Exercise 24. Sketch Σ using the group defined in Example 9, using the cellulation described in Section 5.3 and Figure 13, but the relator $(rv)^2$ is changed to $(rv)^4$. Calculate $\chi^{\text{orb}}(K)$.

Exercise 25. Sketch Σ using the cellulation described in Section 5.3 Figure 13 for the group

$$W = \langle r, s, t, u \mid r^2, s^2, t^2, u^2, (rs)^2, (st)^2, (tu)^2, (ru)^2 \rangle .$$

Does the geometry in your picture need to be distorted in any way? Calculate $\chi^{\text{orb}}(K)$.

Exercise 26. Sketch Σ for the Coxeter group defined by Coxeter graph Γ a hexagon, with each edge labeled 4. Calculate $\chi^{\text{orb}}(K)$.

6 Closing Remarks and Suggested Projects

Working through the above exercises and examples, along with checking and verifying the references, should provide good material for undergraduate projects or even comprise part or all of an independent study course. In addition, we invite you to consider some of the following related ideas.

Research Project 1. Investigation of the cellulations Σ and Σ_{cc}.

Students are invited to explore the *Coxeter cellulation* described in Section 5.4 by consulting [4]. In particular, the student should confirm the assertions made above by considering the following:

- The construction of Σ_{cc} for the Coxeter groups presented throughout this chapter.

- The relationship between the fine (simplicial) cellulation of Σ and the Coxeter cellulation Σ_{cc}.
- The poset of Coxeter cells in Σ_{cc}.
- Verify that the 1-skeleton of Σ_{cc} is isomorphic to the Cayley graph of (W, S).
- Look up the definition of the "Cayley 2-complex" and investigate its relationship to Σ_{cc}.
- Look up the definition of the *nerve* of a Coxeter system, and the *link* of a vertex in a CW-complex. Show that the link of every vertex in Σ_{cc} is isomorphic to the nerve, and that the 1-skeleton of the link is the defining graph Γ.
- For the case when the nerve of (W, S) is a sphere, determine the relationship between the Coxeter cellulation and "mirror" cellulation described above. In particular, note that Σ and Σ_{cc} are the same space, admitting the same group action.

Furthermore, with just a few exceptions, it is known (and can be demonstrated by the student) that if the defining graph is homeomorphic to S^1, then the Davis complex is a 2-*manifold*. But this relationship generalizes to higher dimensions. Namely, if the *nerve* of the Davis complex is homeomorphic to S^{n-1}, then Σ is an n-manifold. Students are invited to explore these definitions, see why there are a "few exceptions" to Σ a 2-manifold in the case the defining graph is an m-gon, and use this strategy as a vehicle to develop examples of group actions on manifolds. But the connection between the nerve and the Coxeter cellulation also enables the efficient construction of the Davis complex (and therefore the Cayley graph) in cases when the defining graph or nerve are not isomorphic to cellulations of $\mathbb{S}^{n-1}$. For starters, consider the case when a defining graph is a triangle with an extra edge extending from one vertex, and all edges labeled with 2.

This survey activity is intended to be exploratory for the student, and may serve to connect some concepts about Cayley graphs, groups, and cosets presented in an undergraduate abstract algebra course.

Research Project 2. Geometry of Σ.

Refer to the geometry discussion in Section 3.2. Noting the deformity of the Davis complex in Figure 13 (is that really a right-angled pentagon?), students are invited to explore the relationship between the relators defining the Coxeter group and the corresponding geometry of the Σ. Given a Coxeter graph Γ an m-gon, and considering the so-called "mirror" cellulation, students should understand what angle should be present between the r- and s-mirrors to respect a relator of the form $(rs)^n$. Then, using some non-Euclidean, Euclidean, and Neutral geometry results from a standard undergraduate geometry course, students are invited to explore the geometry that must be imposed on various examples of Σ if these angles are assigned between mirrors of K. For this exploration, students are encouraged to take

a defining graph Γ an edge or an m-gon and classify the geometry of the resulting Davis complexes. For an investigation of the 3-dimensional geometry of Σ, students are directed to [13].

Research Project 3. Starting with Challenge Problems 6 and 7, find finite index, torsion free subgroups and classify the resulting surfaces covered by Σ.

The surface in Figure 9 is familiarly described as a "torus," but it is also called a *genus 1 surface,* due to the surface having one "hole." Similarly, a sphere is a genus 0 surface, a two-holed torus is a genus 2 surface, etc. As in Example 13, such surfaces arise as quotient spaces of Group actions on manifolds. (In that case, $\mathbb{Z}\times\mathbb{Z}$ acting on $\mathbb{R}^2$.) Moreover, it is known that compact, closed surfaces are uniquely determined by their Euler characteristic: A compact, closed Surfaces of genus g have $\chi = 2-2g$. Indeed, we see in Example 11 that the torus has $\chi = 2-2(1) = 0$.

As discussed in Project Idea 1, with Γ an m-gon, we have, with few exceptions, that Σ is a 2-manifold. The space $K = \Sigma/W$ is compact (it is a finite complex), but it is not a closed surface like the torus. It has a boundary along the edge of K, whereas, the torus has no edge or boundary to it at all. (We are dealing with just the surface of the torus. Not the inside.) The issue with K is that in the action of W on Σ we have non-trivial cell-stabilizers, whereas the action of $\mathbb{Z}\times\mathbb{Z}$ on $\mathbb{R}^2$, the identity element is the only group element that fixes any cell. In this case, $K = \Sigma/W$ is called an "orbifold" and it is for spaces like this that the orbihedral Euler characteristic is meant.

As investigated in Challenge problems 6 and 7, it is the case that we can find finite index subgroups H of W for which Σ/H is a compact, closed surface and therefore has genus determined by its (regular) Euler characteristic. In other words, we can find subgroups $H \leq W$ that act on Σ with all cell-stabilizers being trivial. These are called "torsion-free" subgroups. In general, a group is *torsion free* if it contains no, non-trivial, elements with finite order.

The suggested project for the student is for a given m-gon Γ, use the first isomorphism theorem to detect finite index, torsion-free subgroups. In particular, find a homomorphism $\phi : W \to G$ where G is a finite group, and where the identity element of W is the only element in the kernel that stabilizes any cell of Σ. (That is, all cell-stabilizers inject.) If such a map is found, $H = \ker\phi$ is such a subgroup.

With that set-up, let Γ be an m-gon for which Σ is a 2-manifold. Develop a map $\phi : W \to G$ described above and verify in examples that $H = \ker\phi$ is a finite-index, torsion free subgroup. Verify, in general, that such an H is always a finite index, torsion-free subgroup and that Σ/H is a compact, closed surface (see the hint). Finally, use the multiplicative property of the orbihedral Euler characteristic in Equation 8 to identify the resulting surface Σ/H. For small genus, students may also be able to construct the surface as an identification space in the mode of Figure 9.

Hint: H being finite index is clear. To verify H is torsion free, note that any torsion element $h \in H$ must stabilize a point of σ, hence a cell of Σ (this needs a fixed point theorem and an averaging argument that allows one to find a "centroid" for an orbit. See [1, Corollary 2.8] for reference.) Since this element h is in $H = \ker \phi$, it maps to the identity in G and is in some cell stabilizer. But since ϕ is injective on cell stabilizers, h must be trivial.

References

1. Bridson, M.R., Häfliger, André.: Metric Spaces of Non-Positive Curvature. Springer, Berlin (1999)
2. Cannon, J.: Geometric Group Theory. In: Daverman, R.J., Sher, R.B. (eds.), chap. 6, pp. 261–305. Elsevier, Amsterdam (2001)
3. Davis, M.W.: Groups generated by reflections and aspherical manifolds not covered by Euclidean space. Ann. Math. **117**, 293–324 (1983)
4. Davis, M.W.: The Geometry and Topology of Coxeter Groups. Princeton University Press, Princeton (2007)
5. Davis, M.W., Moussong, G.: Notes on nonpositively curved polyhedra. In: Broczky, K., Neumann, W., Stipicz, A. (eds.) Low Dimensional Topology, pp. 11–94. Janos Bolyai Mathematical Society, Budapest (1999)
6. Fraleigh, J.B.: A First Course in Abstract Algebra, 7th edn. Pearson, Boston (2002)
7. Geoghegan, R.: Topological Methods in Group Theory. Springer, New York (2008)
8. Hatcher, A.: Algebraic Topology. Cambridge University Press, Cambridge (2002)
9. Humphreys, J.: Reflection Groups and Coxeter Groups. Cambridge University Press, Cambridge (1990)
10. Hungerford, T.W.: Algebra. Springer, New York (1974)
11. Messer, R., Straffin, P.: Topology Now! Mathematical Association of America, Washington, DC (2006)
12. Munkres, J.: Toplogy, 2nd edn. Prentice Hall, Upper Saddle River, NJ (2000)
13. Schroeder, T.A.: Geometrization of 3-dimensional Coxeter orbifolds and Singer's conjecture. Geom. Dedicata. **140**(1), 163ff (2009). doi:10.1007/s10711-008-9314-5
14. Schroeder, T.A.: ℓ^2-homology and planar graphs (2013). Colloq. Math. doi:10.4064/cm131-1-11
15. Venema, G.A.: Foundations of Geometry, 2nd edn. Pearson, Boston (2012)

A Tale of Two Symmetries: Embeddable and Non-embeddable Group Actions on Surfaces

Valerie Peterson and Aaron Wootton

Suggested Prerequisites. *An introductory course in group theory and a basic understanding of three-dimensional Euclidean space.*

1 Introduction

Mathematics has long been fascinated by symmetry, so it is perhaps unsurprising that the topic continues to suffuse much of the current undergraduate curriculum. Indeed, examining the symmetries of an object—say, the rigid rotations and reflections of regular polygons in the plane or polytopes in space (such as the Platonic solids in $\mathbb{R}^3$)—provides an intuitive context for a student's first foray into group theory. Here we expand on that notion by extending symmetries of regular polytopes to *compact Riemann surfaces*, or g-holed doughnuts. These surfaces are objects possessed of deeper mathematical structure than regular polytopes but, we claim, a rich source of problems tractable to those with only a modest amount of algebraic experience.

In what follows, we distill the modern approach to finding finite (conformal) symmetry groups of compact Riemann surfaces down to a few manageable tools in an effort to provide a treatment accessible to a wide array of students, faculty, and other enthusiasts.The beauty of this perspective is that jumping into the process of finding symmetry groups does not require a deep understanding of conformal maps or covering space theory. Rather, many of the foundational ideas can be translated into relatively straightforward problems in computational finite group theory using a contemporary adaption of "Riemann's existence theorem," which

V. Peterson (✉) • A. Wootton
Department of Mathematics, University of Portland, 5000 N. Willamette Blvd, Portland, OR 97203, USA
e-mail: petersov@up.edu; wootton@up.edu

A. Wootton et al. (eds.), *A Primer for Undergraduate Research*, Foundations for Undergraduate Research in Mathematics, DOI 10.1007/978-3-319-66065-3_2

verifies exactly when a finite group can act on a Riemann surface in terms of two concrete conditions, one group theoretic and one arithmetic. The introduction of this powerful tool has mustered somewhat of a renaissance for the study of symmetry groups of compact Riemann surfaces in the past thirty years, with significant input provided by undergraduates via summer research programs, independent studies, and honors projects. References [1, 2, 4], and [23] are just a sample of the many recent such works in this area. We suspect there are many more hidden gems in the field yet to be unearthed.

A typical way to introduce compact Riemann surfaces and their symmetry groups is via so-called classical surfaces, due to their visual appeal. Intuitively speaking, a *classical surface* is a smooth, genus g surface in three-dimensional Euclidean space, the *genus* being the number of holes in the surface.[1] One can now visualize a symmetry (or symmetry group) by choosing a classical surface S in $\mathbb{R}^3$ with the appropriate shape and positioning so that a group of rigid rotations about some axis (or set of axes) naturally acts on S. For example, take any smooth genus g surface in $\mathbb{R}^3$, embedded nicely so that its holes line up symmetrically along some fixed axis (illustrated in Figure 1). The group generated by rotating the surface through π radians about this axis produces an action of $\mathbb{Z}_2$, the cyclic group of order 2, on this surface. In this case we say that $\mathbb{Z}_2$ is a *symmetry group* for this surface.[2]

Definition 1. When a rigid rotation of $\mathbb{R}^3$ produces a symmetry of a classical surface S embedded in $\mathbb{R}^3$, as in Figure 1, we say that it is an *embeddable symmetry of S*. We then say that a group of symmetries G of a classical surface S is *embeddable* if it can be realized as the restriction of a rotation subgroup of the full group of isometries of $\mathbb{R}^3$ to the embedded surface S.

We note that this definition excludes reflections because, though they may indeed produce embedded symmetries in a certain context, such maps are necessarily "anti-conformal" and we wish to consider only those actions that preserve the underlying structure of S (the so-called *conformal automorphisms*). Examining anti-conformal symmetries could make an interesting (advanced) undergraduate project, however; readers are referred to [5, 24] for relevant background.

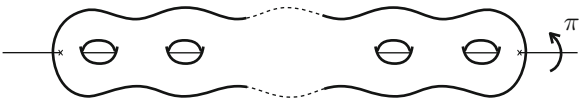


Fig. 1 The generator of the cyclic rotational symmetry group of order 2 on a genus g classical surface.

[1] Stated more carefully, the extra structure that makes a classical surface a *Riemann surface* is a complex analytic structure resulting from the presence of a collection of local C^∞-charts, which make the surface a one-dimensional complex manifold. As our approach here does not make direct use of the complex analytic structure of these surfaces, however, we may safely eschew such technical details.

[2] Our use of the word "symmetry" here is a colloquial choice to replace the more precise but less widely known "conformal automorphism."

Further, we note that although every compact Riemann surface can be realized as a classical surface (see [6] or [19]), very few groups can actually be embedded. This follows from the fact that finite subgroups of the isometry group of $\mathbb{R}^3$ are either cyclic, dihedral, or isomorphic to one of the alternating groups A_4 or A_5 or to the symmetric group S_4 [16, Thm 17.10]. However, a classical result of Hurwitz [10] guarantees that *every* finite group acts on some compact Riemann surface of sufficiently high genus; thus, indeed, "most" finite groups are not embeddable.

In the course of mentoring multiple undergraduate student research projects in the area, we have formulated what we feel is a coherent and friendly overview of the study of (conformal) symmetries of surfaces, one which assumes little more than a basic grounding in finite group theory and Euclidean geometry. The primary purpose of this article is, therefore, to share this analysis and perspective so that others might appreciate the beauty and simplicity of the subject. As an application, we employ the techniques and tools described below in order to detect all A_4-actions on Riemann surfaces of genus $g \geq 2$. Further, necessary and sufficient conditions are given characterizing exactly which of those A_4-actions are embedded. More precisely, we construct all possible surfaces for which an action of A_4, together with its *signature data*, is embeddable, the distinction being that we are also interested in preserving properties of the quotient prescribed by that action. Regarding the other possible finite isometry subgroups, we speculate that determining conditions for the embeddability of A_5 or S_4 analogous to the conditions that follow for A_4 would make fruitful student research projects. Finally, the theory developed herein holds specifically for surfaces of genus $g \geq 2$, so we fix this assumption for the remainder of the article.

2 Determining the Existence of a Group Action

With the goal of presenting necessary and sufficient conditions for a finite group to act by symmetry on a compact Riemann surface S of genus $g \geq 2$, we first introduce and illustrate the needed terminology with some concrete and easy-to-visualize examples. Though the examples we present all involve classical surfaces in $\mathbb{R}^3$, the ideas discussed also apply to group actions on more abstract surfaces.

2.1 Realizing A_4 as a Group of Rotations

The group of isometries (or "rigid symmetries") of $\mathbb{R}^3$ can be described in various ways, such as products of reflections. We refer the interested reader to [16] for more. The subgroup of the full group of isometries that consists of orientation-preserving symmetries (or "rigid motions") is generated by translations and rotations. Ultimately, we are interested in the different ways that A_4 can act on surfaces as a group of rotations, so we will begin by examining a familiar action of A_4 on one of the Platonic solids: a tetrahedron embedded in $\mathbb{R}^3$.

First recall that A_4 is the alternating group on four elements; it is the subgroup of the symmetric group S_4 consisting of all even permutations. It is often convenient to denote the elements of A_4 using the following permutation notation:

$$A_4 = \{(1), (12)(34), (13)(24), (14)(23), (123), (132),$$
$$(124), (142), (134), (143), (234), (243)\}$$

Note that A_4 has many cyclic subgroups. Subgroups of order 2 are generated by permutations of the form $(ab)(cd)$ (the products of transpositions) and order 3 subgroups are generated by elements of the form (abc) (the 3-cycles). When taken together with the identity, the three products of transpositions form a subgroup of order four isomorphic to the Klein 4-group, V_4. Readers may refer to [9], for example, for a treatment of alternating and symmetric groups and their structure.

Now consider how A_4 can act on an embedded regular tetrahedron $\mathbf{T}$ whose center lies at the origin (see Figure 2). We view the permutation cycles of A_4 listed above as permuting the vertices of $\mathbf{T}$ and now seek to describe those permutations as rotations of $\mathbb{R}^3$.

Consider an axis passing through the uppermost vertex of $\mathbf{T}$ and the origin. Rotating $\mathbf{T}$ about this axis through an angle of $\frac{2\pi}{3}$ yields a rotational symmetry of $\mathbf{T}$ that cyclically permutes the remaining three vertices of $\mathbf{T}$, as shown in Figure 3 (left). If we label the vertices as shown, this symmetry is represented by the order 3 permutation $(1)(234) = (234)$. We obtain similar rotational symmetries about lines passing through the origin and each of the other vertices and thereby obtain actions on T represented by the remaining 3-cycles in A_4.

In addition to these rotational symmetries, if we skewer any two non-adjacent (opposite) edges of $\mathbf{T}$ by a line passing through their midpoints, as in Figure 3

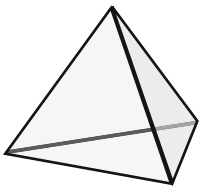


Fig. 2 A regular tetrahedron $\mathbf{T}$ in $\mathbb{R}^3$.

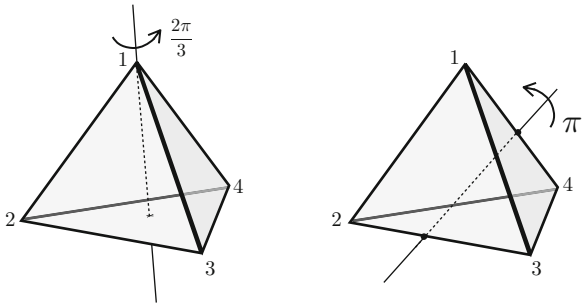


Fig. 3 Rotations of the tetrahedron in $\mathbb{R}^3$ of order 3 (left) and order 2 (right).

(right), rotation about this axis through an angle of π also yields a symmetry of $\mathbf{T}$, this time of order 2. Representing this symmetry as a permutation of the vertices as numbered above, rotation about the line depicted in Figure 3 yields $(14)(23)$. Rotating about similar lines through the other two pairs of opposing sides yields the permutations $(12)(34)$ and $(13)(24)$. The rotational symmetry group of $\mathbf{T}$, then, contains A_4. (In fact, there are no other rotations of $\mathbf{T}$, so A_4 is the full group of rotations. See [16, Thm 17.2].)

Soon we will use the tetrahedron $\mathbf{T}$ to construct a number of different surfaces on which A_4 acts. Henceforth, when we refer to A_4 as a group of rotations acting on one of those surfaces, we shall be referring to the specific rotations just described.

2.2 Preliminary Examples

To learn more about group actions, we turn now to examples. These will also prepare us for the construction of new surfaces on which A_4 acts, wherein handles and surfaces of higher genus are added to the tetrahedron.

Example 1. Let S be the classical genus 2 surface illustrated in Figure 4 with its center at the origin, situated symmetrically across all axes.

If α is a rotation in $\mathbb{R}^3$ through π radians about the x-axis, then the group $G_1 = \langle\alpha\rangle$ will be an embedded group of symmetries of S isomorphic to the cyclic group of order 2. Likewise, if β is a rotation through π radians about the y-axis, then the group $G_2 = \langle\beta\rangle$ will be an embedded group of symmetries of S also isomorphic to the cyclic group of order 2. It is not hard to see that the group $G = \langle\alpha, \beta\rangle$, which is isomorphic to the Klein 4-group, is also an embedded group acting on S wherein the element $\alpha\beta = \gamma$ accomplishes rotation through π about the z-axis.

Observe that the groups G_1 and G_2 from Example 1 act very differently on the genus 2 surface despite being isomorphic groups; α takes each hole to itself, for example, whereas β swaps the holes. This should clarify the earlier comment that we do not want simply to classify how to embed A_4 itself, but to classify how to embed all the different ways A_4 can act. To determine exactly when a symmetry group acts on a surface, we need to develop a coherent way of describing the effects of group actions on a surface in order to distinguish between the actions themselves. As motivation, let's consider a few more examples; we'll refer back to each of these in the next section.

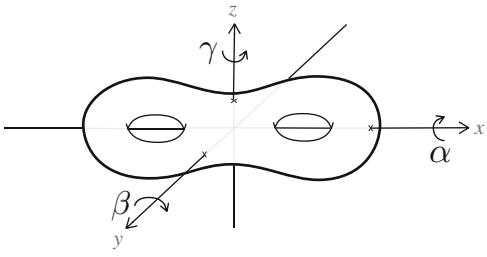


Fig. 4 Various $\mathbb{Z}_2$-actions on a genus 2 surface.

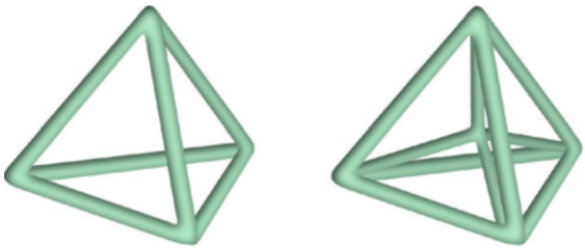


Fig. 5 A_4 acts on an inflated tetrahedron (left), and on an inflated tetrahedron with a "jack" added (right).

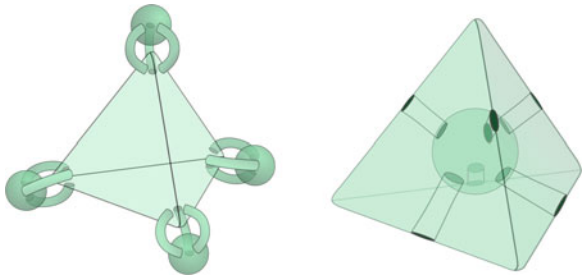


Fig. 6 A_4 acts on a tetrahedron with tripods (left) and on the boundary of a punctured solid tetrahedron (right).

Example 2. Rather than considering the entire tetrahedron, now consider the skeleton of $\mathbf{T}$ (*i.e.*, just the vertices and the edges of $\mathbf{T}$). If we "inflate" these edges into tubes and smooth out the corners appropriately, we obtain a compact Riemann surface on which A_4 acts as an embedded group exactly as before; see Figure 5 (left). Observe that this is a surface of genus 3: if we flatten the inflated tetrahedron by pushing down on the top vertex while expanding the base, the bottom triangle becomes the outer ring of a three-holed torus. We denote this (pre-squashed) inflated skeleton of $\mathbf{T}$ by $T_{(0;2,2,3,3)}$. The seemingly strange notation will become self-explanatory in due time.

Example 3. Now consider the skeleton of the tetrahedron $\mathbf{T}$ with four additional edges added as follows: an edge emanates from each vertex, and all four edges meet and terminate at the center of $\mathbf{T}$. As in the previous example, inflating these edges into tubes and smoothing out the corners appropriately produces a compact Riemann surface on which A_4 acts; see Figure 5 (right). Observe that this is a surface of genus 6: though trickier to visualize, if we push down on the center of the jack to produce three windowpanes below and then push the top vertex of the tetrahedron through one of those panes, we will end up with a handle attached to 5 panes, for a total of 6 holes. (For a combinatorial argument that this surface has genus 6, see [7].) We denote this "tetrahedron with a jack" surface by $T_{(0;2,2,2,3,3)}$.

Example 4. For a genus 8 surface on which A_4 acts, consider the following. Begin with the original tetrahedron and glue in a "tripod," consisting of three cylinders emanating symmetrically from a sphere, around each vertex of the tetrahedron. This surface, denoted $T_{(0;2,3,3,3,3)}$, is pictured in Figure 6 (left).

Example 5. For one last foundational example, we now return to the original tetrahedron $\mathbf{T}$ acted on by A_4 but now also place a smaller sphere inside $\mathbf{T}$ (this construction works just as well by placing a smaller tetrahedron, or any surface homeomorphic to a sphere, inside $\mathbf{T}$ but is easiest to visualize with the sphere). Construct a new surface by drilling holes through the tetrahedron and sphere (at the midpoints of the tetrahedron's edges, say) and connecting the inner and outer surfaces by gluing a cylinder into each pair of holes, so that the interior of the new surface lies between the two original surfaces. Smoothing corners and edges appropriately yields a compact Riemann surface on which A_4 once again acts as an embedded group; see Figure 6 (right). Note that this new surface has genus 5: even though 6 holes were drilled, it is possible to deform the surface by stretching one of these cylinders out and around the larger tetrahedron, leaving just 5 handles. (The moral is that one must be careful when counting holes to determine genus. This was also illustrated in Example 2, where the inflated tetrahedron skeleton—with 4 apparent holes—was deformed to reveal a surface of genus of 3.) We denote this (pre-deformed) surface by $T_{(0;3,3,3,3)}$.

2.3 Signatures

As previously noted, the groups G_1 and G_2 from Example 1 are isomorphic but act very differently as symmetry groups of the genus 2 surface in that example. In order to capture the differences between groups actions, we now introduce the notion of a *signature*. Signatures record information about the *orbits* and *branch points* of an action. We will develop these notions via examples and then give more formal definitions.

Recall that if a group G acts on a set X, then the *orbit* of an element $x \in X$ is the set of all points in X to which x is mapped by elements of G; that is, the orbit of x is the set $\{y \in X \mid y = g(x) \text{ for some } g \in G\}$. For such an action, we then define the *quotient space* X/G to be the set of all orbits of X under the action of G. Now, if G is acting on a compact Riemann surface S, the quotient space S/G can be made into a compact topological space after endowing it with the quotient topology. (In fact, with appropriate local charts chosen, S/G is itself a compact Riemann surface, see [17, Thm 3.4].) It will be convenient here to think of S/G as a subset of S consisting of exactly one member from each possible orbit (*i.e.*, as a *fundamental domain* of S).

Example 6. To illustrate this, let S be the classical genus 2 surface from Example 1 and let $G_1 = \langle \alpha \rangle$, the order 2 group of rotations of S about the x-axis. The quotient surface S/G_1 can be realized by taking half of the surface S (obtained by slicing S with the xz-plane) and identifying the top and the bottom halves of each circle that results from the intersection of S with the xz-plane; see Figure 7. Observe that the surface S/G_1 has genus 0.

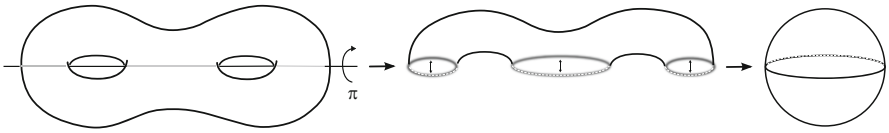


Fig. 7 Acting by α on this genus 2 surface produces a genus 0 quotient

The previous example illustrates that sometimes it is not hard to describe the quotient space S/G, or its genus, explicitly. It isn't always so easy. Fortunately, we shall see in the next section that there is a formula for determining the genus of a quotient.

Definition 2. If G is a finite group acting on S, let $\pi_G : S \to S/G$ denote the natural quotient map between the surfaces S and S/G. That is, π_G takes a point in S and returns the orbit of that point, now thought of as a single element in the quotient S/G. In the other direction, given an orbit $y \in S/G$, then $\pi_G^{-1}(y)$ is that same orbit, now thought of as a set of points in S. We say an orbit $y \in S/G$ is a *branch point* of the map π_G if $|\pi_G^{-1}(y)| < |G|$. We define the *order* of y to be $|G|/|\pi_G^{-1}(y)|$.

Note that if no nontrivial element of G fixes a point $x \in S$, the G-orbit of x will consist of $|G|$ distinct points. Thus, in some sense, "most" orbits (as a set of points in S) have size $|G|$ and *branch points* are simply those orbits whose size is strictly smaller. We use the term branch "points" rather than branch "orbits" merely to highlight the fact that we are referring to elements of S/G.

As it turns out, the order of a branch point is always an integer. For any point $x \in S$, the *stabilizer* of x is the subgroup of G that leaves x fixed; we denote this by $\mathrm{Stab}_G(x)$. The orbit-stabilizer theorem (see [9]) implies that the stabilizer subgroups for points in the same orbit are conjugate, and that the size of an orbit is the (shared) index of any (conjugate) stabilizer. Thus, the size of an orbit divides $|G|$. This also gives us a way to determine the order of a branch point: choose any point $x \in S$ from a given orbit, and count the number of elements of G that fix that point. That is, the order of the branch point $\pi_G(x)$ coincides with $|\mathrm{Stab}_G(x)|$, the order of the stabilizer subgroup of x. So, orbits of elements that are not fixed by G (*i.e.*, orbits of elements whose stabilizer is the trivial subgroup $\{e\}$) have size $|G|$ and we do not refer to $\pi_G(x)$ as a branch point in this case.

Example 7. Again, let S be the classical genus 2 surface from Example 1 and let $G_1 = \langle \alpha \rangle$. Observe that the map π_{G_1} has six branch points, each corresponding to an intersection point of S with the x-axis; more precisely, the intersection of the x-axis with the fundamental domain shown in the middle of Figure 8. Since $|G_1| = 2$ and there is exactly one point in each preimage $\pi_{G_1}^{-1}(y)$ for each branch point y, the order of each branch point will be $2/1 = 2$ (Figure 8).

We are now ready to introduce the main tool we shall use to describe and help differentiate between group actions.

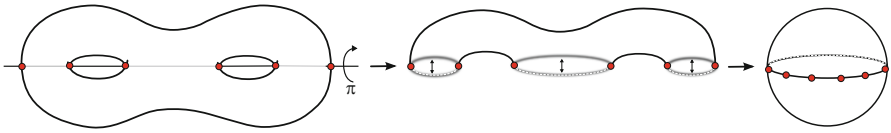


Fig. 8 Branch points and quotient for the action of $\langle\alpha\rangle$ on a genus 2 surface.

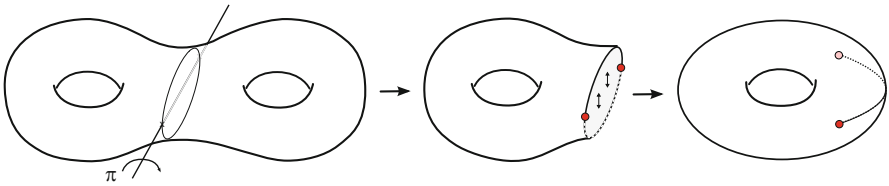


Fig. 9 Acting by β on this genus 2 surface produces a genus 1 quotient.

Definition 3. Suppose that a finite group G acts on a surface S. We define the *signature* of the action to be the tuple $(h; m_1, \ldots, m_r)$ where h is the genus of S/G and r is the number of branch points of π_G, the order of these branch points being $m_1, m_2, \ldots, m_r$. The order in which we list the m_i does not matter, though they typically appear in non-decreasing order.

The signature of a group action encodes much of the topological and geometric structure of how the group acts on a surface S. We illustrate with a couple of examples using classical surfaces.

Example 8. Once again, let S be the classical genus 2 surface from Example 1 and let $G_1 = \langle\alpha\rangle$. As illustrated in Example 6, the quotient space S/G_1 has genus 0. In Example 7, we showed that the map π_{G_1} has 6 branch points, each of order 2. Therefore, G_1 acts with signature $(0; 2, 2, 2, 2, 2, 2)$.

Now consider $G_2 = \langle\beta\rangle$. The quotient surface S/G_2 can be identified with the half of S obtained by slicing S with the yz-plane and identifying the top and the bottom halves of the circle where S intersects the yz-plane; see Figure 9. The surface S/G_2 still contains a hole, so has genus 1. Also observe that the map π_{G_2} has two branch points, each again corresponding to the intersection point of S (or, precisely, of the fundamental domain of S/G shown in the middle of Figure 9) with the y-axis. Since $|G_2| = 2$ and there is exactly one point in each preimage $\pi_{G_2}^{-1}(y)$ for each branch point y, the order of each branch point will be $2/1 = 2$. Therefore, G_2 acts with signature $(1; 2, 2)$.

We can use similar techniques to determine the signatures of the other groups acting on S from Example 8.

Exercise 1. Show that the signature of $\langle\alpha\beta\rangle$ (the z-axis rotation from Example 1) is also $(1; 2, 2)$.

Exercise 2. Show that the group $G = \langle \alpha, \beta \rangle$ is isomorphic to the Klein 4-group and acts with signature $(0; 2, 2, 2, 2, 2)$.

2.4 Generating Vectors and Riemann's Existence Theorem

We introduced the notion of the signature of a group action to encode topological data associated with the action. However, it is also one of the fundamental tools used to provide necessary and sufficient conditions for the existence of a group action on a surface S of genus $g \geq 2$. Before we provide these conditions, we present a few needed definitions.

Definition 4. For a fixed group G and elements $x, y \in G$, we define the *commutator* of x and y to be $xyx^{-1}y^{-1}$ and denote it by $[x, y]$. We call the subgroup of G generated by all its commutators the *commutator subgroup*.

Definition 5. Given a finite group G with identity element e_G, we say a vector $(a_1, b_1, a_2, b_2, \ldots, a_h, b_h, c_1, \ldots, c_r)$ of elements of G is an $(h; m_1, \ldots, m_r)$-*generating vector* for G if all of the following hold:

1. $G = \langle a_1, b_1, a_2, b_2, \ldots, a_h, b_h, c_1, \ldots, c_r \rangle$
2. The order of c_j is m_j for $1 \leq j \leq r$
3. $\prod_{i=1}^{h} [a_i, b_i] \prod_{j=1}^{r} c_j = e_G$

We are now ready to state the main tool used in modern day group action classification, "Riemann's existence theorem."

Theorem 1. *[3, Prop. 2.1] A finite group G acts on a compact Riemann surface S of genus $g \geq 2$ with signature $(h; m_1, \ldots, m_r)$ if and only if:*

1. *the Riemann–Hurwitz formula is satisfied:*

$$g - 1 = |G|(h - 1) + \frac{|G|}{2} \sum_{j=1}^{r} \left(1 - \frac{1}{m_j}\right),$$

   and
2. *there exists an $(h; m_1, \ldots, m_r)$-generating vector for G.*

Put another way, this theorem says we can guarantee the existence of some group action simply by finding a generating vector and checking that the Riemann-Hurwitz formula produces $g \geq 2$. As we will now see with a few examples, constructing a generating vector with the desired properties is often a straightforward task.

Example 9. The vector (x, x, x, x, x, x) is a $(0; 2, 2, 2, 2, 2, 2)$-generating vector for the cyclic group $\mathbb{Z}_2 = \langle x \rangle = \{e, x\}$ of order 2. The three conditions for being a

generating vector are satisfied: clearly $G = \langle x, x, x, x, x, x \rangle$, the order of each element is 2, and $x \cdot x \cdot x \cdot x \cdot x \cdot x = x^6 = e$. The Riemann-Hurwitz equation becomes

$$g - 1 = 2(0 - 1) + \frac{2}{2} \sum_{j=1}^{6} \left(1 - \frac{1}{2}\right) = -2 + 6 \cdot \frac{1}{2} = -2 + 3 = 1,$$

from which we deduce that $g = 2$. Theorem 1 then implies that there is an action of $\mathbb{Z}_2$ with signature $(0; 2, 2, 2, 2, 2, 2)$ on a surface of genus 2. We exhibited such an action in Example 8.

Example 10. The vector (e, e, x, x) is a $(1; 2, 2)$-generating vector for the same cyclic group $\mathbb{Z}_2 = \langle x \rangle$ of order 2. In this case, the Riemann-Hurwitz equation yields

$$g - 1 = 2(1 - 1) + \frac{2}{2} \sum_{j=1}^{2} \left(1 - \frac{1}{2}\right) = 0 + 2 \cdot \frac{1}{2} = 1,$$

and again we obtain $g = 2$. Theorem 1 tells us there is an action of $\mathbb{Z}_2$ with signature $(1; 2, 2)$ on a surface of genus 2. This action, too, was exhibited in Example 8.

Example 11. The vector $\big((12)(34), (13)(24), (123), (124)\big)$ is a $(0; 2, 2, 3, 3)$-generating vector for the alternating group A_4. Verifying that that these four elements indeed generate A_4 is left to the reader. The Riemann–Hurwitz equation becomes

$$g - 1 = 12(0 - 1) + \frac{12}{2}\left(1 - \frac{1}{2} + 1 - \frac{1}{2} + 1 - \frac{1}{3} + 1 - \frac{1}{3} + \right)$$

$$= -12 + 6 \cdot \frac{7}{3} = -12 + 14 = 2,$$

giving $g = 3$. As such, Theorem 1 guarantees the existence of an action of A_4 on a surface of genus 3 with signature $(0; 2, 2, 3, 3)$. An example of such an action was presented in Example 2 wherein A_4 acted on an inflated tetrahedron skeleton.

Exercise 3. Use Theorem 1 to show that there exists an action of the Klein 4-group V_4 on a surface of genus 2 with signature $(0; 2, 2, 2, 2, 2)$. That is, construct a $(0; 2, 2, 2, 2, 2)$-generating vector for V_4 and then determine genus by using the Riemann-Hurwitz formula. Note that an example of such an action was presented in Example 1 with signature determined in Exercise 2.

Exercise 4. Use Theorem 1 to show that A_4 acts with signatures $(0; 2, 2, 2, 3, 3)$, $(0; 2, 3, 3, 3, 3)$ and $(0; 3, 3, 3, 3)$ by constructing explicit generating vectors. Use the Riemann-Hurwitz formula to determine the genus of the corresponding surface in each case. As we shall see shortly, such actions were described explicitly in Examples 3, 4, and 5 respectively.

Theorem 1 can also be used in the opposite direction to determine the genus of a quotient. If one already has already identified a group action of G on a surface S of genus $g \geq 2$—not necessarily an embedded action but an action defined explicitly enough to determine the branching data for the quotient map—then Theorem 1 allows you to compute the genus of S/G and therefore the signature of the G-action without ever having to construct a generating vector for G. Here follows an example.

Example 12. In Example 2 we constructed the "inflated tetrahedron" surface $\mathbf{T}_{(0;2,2,3,3)}$ on which the group A_4 acts (as an embedded group, in fact). Notice that for this action, the only fixed points (*i.e.*, points fixed by at least one non-identity group element) are the intersection points between the surface and the axes of rotation. From this we can extract the data on branch points necessary to apply Theorem 1.

First suppose that L is a line passing through a vertex (inflated) and the origin. Then the two points of intersection of L with the surface (one "inside" the vertex, closer to the origin, and one "outside") are the only points fixed by the rotation about L. Next note that the action of A_4 permutes the vertices of $\mathbf{T}_{(0;2,2,3,3)}$, so in particular, all fixed points on the outside of a vertex of $\mathbf{T}_{(0;2,2,3,3)}$ belong to the same orbit and all points on the inside belong to the same orbit (and these two orbits are distinct). Thus there are two such orbits of these vertex fixed points, each orbit having four points, and therefore two branch points of order $12/4 = 3$ for the action of rotation about L.

Now suppose that L is any one of the three lines that intersect the (inflated) midpoints of a pair of opposite edges of the inflated tetrahedron. Then the four points of intersection of L with $\mathbf{T}_{(0;2,2,3,3)}$ (through the inner and outer midpoints of the two inflated edges through which L passes) are the only points fixed by the rotation about L. Note that the action of A_4 permutes the midpoints of each of the edges of $\mathbf{T}_{(0;2,2,3,3)}$ so, as was the case with vertices, all fixed points on the outside of a midpoint of $\mathbf{T}_{(0;2,2,3,3)}$ belong to the same orbit and all points on the inside belong to the same orbit and these two orbits are distinct. Thus there are two such orbits, each with six points, implying two branch points of order $12/6 = 2$.

Since there are no other branch points, and we know the genus of $\mathbf{T}_{(0;2,2,3,3)}$ is 3, we can now use the Riemann–Hurwitz formula to find the genus of $\mathbf{T}_{(0;2,2,3,3)}/A_4$ and hence find the signature of this A_4-action. Specifically, we have

$$3 - 1 = 12(h-1) + \frac{12}{2}\left(1 - \frac{1}{2} + 1 - \frac{1}{2} + 1 - \frac{1}{3} + 1 - \frac{1}{3}\right)$$

$$= 12(h-1) + 6 \cdot \frac{7}{3} = 12(h-1) + 14.$$

Therefore $12(h-1) = -12$ so $h = 0$. It follows that A_4 acts with signature $(0;2,2,3,3)$ on $\mathbf{T}_{(0;2,2,3,3)}$ (explaining our choice of subscript!). Note that in Example 11 we proved the existence of an action with a given signature by constructing an explicit generating vector for that action, whereas here we used

our knowledge of an explicit action to compute branching data and then deduce the signature for the action. This demonstrates two different ways of applying Theorem 1.

Exercise 5. Repeat the process of Example 12 to recover the signatures for the actions described in Examples 3, 4, and 5. Specifically, use the geometric descriptions given for each action to identify branch points (and their respective orders) and then apply the Riemann-Hurwitz formula to find the genus of each quotient surface.

3 Actions of the Alternating Group A_4

To illustrate an application of the techniques presented above, we now deduce a complete description of all possible actions of the alternating group A_4. Moreover, in addition to providing necessary and sufficient conditions for when a signature arises as the signature of some A_4-action via Riemann's existence theorem, we shall go one step further and determine exactly which of these signatures correspond to *embedded* actions.

3.1 Signatures for A_4-Actions

In order to determine necessary and sufficient conditions for when an action of A_4 on a compact Riemann surface of genus $g \geq 2$ has a given signature we need to apply Theorem 1. First note that since A_4 only contains (nontrivial) elements of order 2 and 3, any signature for such an action will be of the form $(h; 2^P, 3^Q)$ where by "$2^P, 3^Q$" we mean P copies of 2 and Q copies of 3. In light of the fact that we are restricting to genus $g \geq 2$, the requirement that a signature satisfy the Riemann-Hurwitz formula (condition (1) of Theorem 1) allows us to exclude a number of signatures immediately.[3] A bit of arithmetic shows that any other signature of the form $(h; 2^P, 3^Q)$ satisfies the Riemann–Hurwitz formula for some $g \geq 2$, so to determine which of the remaining signatures are actual signatures, we must either construct a generating vector (by condition (2) of Theorem 1) or show that one can not be constructed. Since part of the definition of a generating vector involves group commutators and group generators, we'll need two facts (left as exercises) about commutators and generators to do this.

Lemma 1. *The commutator subgroup of A_4 is V_4, the Klein 4-group.*

[3] Specifically, the signatures $(0; -)$, $(1; -)$, $(0; 2)$, $(0; 2, 2)$, $(0; 2, 2, 2)$, $(0; 2, 2, 2, 2)$, $(0; 3)$, $(0; 3, 3)$, $(0; 3, 3, 3)$, $(0; 2, 3)$, $(0; 2, 2, 3)$, and $(0; 2, 3, 3)$, can all be excluded, where $(h; -)$ denotes an action with no branch points.

Lemma 2. *A subset $\{x_1, \ldots, x_n\} \subseteq A_4$ generates A_4 if and only if:*

1. *at least one of the x_i has order 3; and,*
2. *if all x_i have order 3, then $n \geq 2$ and there exist i, j such that $x_i \notin \langle x_j \rangle$.*

We are now ready to provide necessary and sufficient conditions for a signature to be realizable as the signature of an A_4-action.

Theorem 2. *A signature of the form $(h; 2^P, 3^Q)$ is the signature for some A_4-action on a surface of genus $g \geq 2$ if and only if:*

1. *$Q \neq 1$; and,*
2. *if $h = 0$ then $Q \neq 0$; and,*
3. *$(h; 2^P, 3^Q)$ does not belong to this list of exceptions: $(1; -)$, $(0; 3, 3)$, $(0; 3, 3, 3)$, $(0; 2, 3, 3)$, where $(h; -)$ denotes an action with no branch points.*

Proof. First, we introduce some convenient notation. Since we are trying to construct generating vectors for A_4, and we know they must have the form $(h; 2^P, 3^Q)$, we shall denote such a generating vector by:

$$(a_1, b_1, a_2, b_2, \ldots, a_h, b_h, c_1, \ldots, c_P, d_1, \ldots d_Q)$$

where $a_1, b_1, a_2, b_2, \ldots, a_h, b_h$ are as originally defined, but $c_1, \ldots, c_P$ are all elements of order 2 and $d_1, \ldots d_Q$ are all of order 3 (so we have split the last r terms up according to order).

For the forward direction, assume $(a_1, b_1, a_2, b_2, \ldots, a_h, b_h, c_1, \ldots, c_P, d_1, \ldots d_Q)$ is a generating vector with $Q = 1$. Then we must have

$$\prod_{i=1}^{h} [a_i, b_i] \left(\prod_{j=1}^{P} c_j \right) d_1 = e$$

or equivalently

$$\prod_{i=1}^{h} [a_i, b_i] \left(\prod_{j=1}^{P} c_j \right) = d_1^{-1}.$$

By Lemma 1, however, we know $[a_i, b_i] \in V_4$ for any a_i and b_i, and we know $\prod_{j=1}^{P} c_j \in V_4$, so it follows that $\prod_{i=1}^{h} [a_i, b_i] \left(\prod_{j=1}^{P} c_j \right) \in V_4$, a contradiction since we are assuming d_1 has order 3. Thus a signature of the form $(h; 2^P, 3^1)$ cannot be the signature of an A_4-action.

A vector with $h = 0$ and $Q = 0$ will be of the form $(c_1, \ldots, c_P)$. In particular, it will contain only elements of order 2, so by Lemma 2 cannot generate A_4. It follows that there is no generating vector for A_4 with $h = 0$ and $Q = 0$, and hence if $h = 0$, we must have $Q > 0$. The Riemann–Hurwitz formula ensures that the signature $(h; 2^P, 3^Q)$ does not have the form of any of the exceptions listed in part 3 of the theorem.

To prove the reverse direction, we start constructing generating vectors that satisfy the conditions of Theorem 1. We can construct a generating vector of the form $(h; 2^P, 3^Q)$ with $P, Q \geq 2$ as follows. We set $a_i = b_i = e$ for all i. If P is even, set $c_i = (12)(34)$ for all i, and if P is odd, set $c_1 = (12)(34)$, $c_2 = (13)(24)$, $c_3 = (14)(23)$ and $c_i = (12)(34)$ for all $i > 3$. For Q even, we set $d_{2i-1} = (123)$ and $d_{2i} = (321)$ for $1 \leq i \leq Q/2$ and for Q odd, we set $d_1 = d_2 = d_3 = (123)$ and $d_{2i} = (123)$ and $d_{2i+1} = (321)$ for $2 \leq i \leq (Q-1)/2$. By construction, it is easy to see that each of these generating vectors satisfy the relation $\left(\prod_{i=1}^{h}[a_i, b_i]\right)\left(\prod_{j=1}^{P} c_j\right)\left(\prod_{j=1}^{Q} d_j\right) = e$ and since $P, Q \geq 2$, there is always an element of order 2 and element of order 3 in this generating vector, so the elements generate A_4 by Lemma 2. Thus these signatures always give rise to some A_4-action.

To finish, we need to consider the case $Q = 0$ and the cases $P = 0, 1$ ($Q = 1$ is excluded by assumption). We shall proceed by cases, though in each case, our argument will be the same: we provide an explicit generating vector for each signature. To avoid unnecessary repeated arguments, we note that in each case these generating vectors will contain elements of order 2 and 3, so by Lemma 2 will generate A_4. We also leave it as an easy exercise to the reader to verify that the product $\left(\prod_{i=1}^{h}[a_i, b_i]\right)\left(\prod_{j=1}^{P} c_j\right)\left(\prod_{j=1}^{Q} d_j\right) = e$ is satisfied.

Suppose that $Q = 0$. By the observations above, we must have $h > 0$. Now if $P = 0$ as well, we must have $h \geq 2$, and so we define a generating vector by

$$(a_1, b_1, a_2, b_2, a_3, b_3, \ldots, a_h, b_h) = ((12)(34), e, (123), e, e, e, \ldots, e, e).$$

If $P = 1$, we define a generating vector by

$$(a_1, b_1, a_2, b_2, \ldots, a_h, b_h, c_1) = ((123), (234), e, e, \ldots, e, e, (14)(23)).$$

If $P \geq 2$, we define a generating vector by $a_1 = (123)$, $b_1 = e$, $a_i = b_i = e$ for all $i > 1$ and if P is even, set $c_i = (12)(34)$ for all i, and if P is odd, set $c_1 = (12)(34)$, $c_2 = (13)(24)$, $c_3 = (14)(23)$ and $c_i = (12)(34)$ for all $i > 3$.

Now suppose that $P = 0$. By our arguments above, we may suppose $Q \geq 2$. If $Q = 2$ or $Q = 3$, we must have $h \geq 1$ in which case we can define generating vectors by

$$(a_1, b_1, a_2, b_2, \ldots, a_h, b_h, d_1, d_2) = ((12)(34), e, e, e, \ldots, e, e, (123), (321))$$

and

$$(a_1, b_1, a_2, b_2, \ldots, a_h, b_h, d_1, d_2, d_3) = ((12)(34), e, e, e, \ldots, e, e, (123), (123), (123))$$

respectively. If $Q \geq 4$, we define a generating vector by $a_i = b_i = e$ for all i, $d_1 = (234)$, $d_2 = (432)$, and for Q even, we set $d_{2i-1} = (123)$ and $d_{2i} = (321)$ for $3 \leq i \leq Q/2$, and for Q odd, we set $d_3 = (123)$, $d_4 = (123)$, $d_5 = (123)$ and $d_{2i} = (123)$ and $d_{2i+1} = (321)$ for $3 \leq i \leq (Q-1)/2$.

Finally suppose that $P = 1$. By our arguments above, we may again suppose $Q \geq 2$. Now, if $Q = 2$, we must have $h \geq 1$ in which case we can define a generating vector by

$$(a_1, b_1, a_2, \ldots, a_h, b_h, c_1, d_1, d_2) = (e, e, e, \ldots, e, e, (12)(34)), (123), (234)).$$

If $Q = 3$, we can define a generating vector by replacing the element $d_1 = (123)$ above with the pair $(321), (321)$. This has the effect of leaving the product fixed while adding an element of order 3:

$$(a_1, b_1, \ldots, a_h, b_h, c_1, d_1, d_2, d_3) = (e, e, \ldots, e, e, (12)(34), (321), (321), (234)).$$

If $Q \geq 4$, we define a generating vector by $a_i = b_i = e$ for all i, $c_1 = (12)(34)$, $d_1 = (123)$, $d_2 = (234)$, and for Q even, we set $d_{2i-1} = (123)$ and $d_{2i} = (321)$ for $3 \leq i \leq Q/2$, and for Q odd, we set $d_3 = (123)$, $d_4 = (123)$, $d_5 = (123)$ and $d_{2i} = (123)$ and $d_{2i+1} = (321)$ for $3 \leq i \leq (Q-1)/2$. □

For a fixed genus g, the Riemann-Hurwitz formula can be used to find all the possible signatures with which A_4 might act on a surface of genus g. Theorem 2 now allows us to determine for which of those signatures there is an action. We illustrate.

Example 13. We determine all signatures for which there exists an A_4-action on a surface of genus 16. Any such signature $(h; 2^P, 3^Q)$ must satisfy the Riemann-Hurwitz formula which simplifies as follows

$$27 = 12h + 3P + 4Q.$$

Since $h, P, Q \geq 0$ are integers, we can determine all possible solutions to this equation. We get the following list of potential signatures: $(0; 2^1, 3^6)$, $(0; 2^5, 3^3)$, $(0; 2^7)$, $(1; 2^1, 3^3)$, $(1; 2^5)$, $(2; 2^1)$. Applying Theorem 2, we see that all of these signatures correspond to an A_4-action on a surface of genus 16 except the signature $(0; 2^7)$.

Exercise 6. Determine all signatures for which there exists an A_4-action on a surface of genus 35.

4 Embeddable A_4-Actions

Theorem 2 provides necessary and sufficient conditions for the existence of any A_4-action on a compact Riemann surface S in terms of the signature of the action. Though the mathematics underlying this result is quite deep, only a basic understanding of finite group theory is required to apply the result. Accordingly, many problems in the field are made accessible to researchers with diverse backgrounds and, in particular, are amenable to student investigations. To illustrate, we now apply this result (together with a few additional results) to address the embeddability of A_4-actions. Specifically, we shall do the following: characterize exactly when a signature $\mathscr{S}$ is the signature of an *embedded* action of A_4 on some compact Riemann surface.

4.1 Necessary and Sufficient Conditions for Embeddability of A_4

In work by Thomas Tucker [22], conditions are given identifying all possible genera for surfaces admitting embeddable finite group actions, followed by constructions for realizing these; in the case of A_4, the construction involves drilling holes in or attaching various gadgets to an inflated tetrahedron. As it turns out, although not mentioned in [22], Tucker's methods allow one to classify all possible signatures for embedded actions, not just genera. (The distinction is that a single genus may give rise to multiple signatures.) We will now exemplify Tucker's methods and make the construction in [22] explicit in the context of classifying all *signatures* of embedded A_4-actions. While the constructions in [22] are simpler than the ones that follow here, we note that the goals are different: we describe signatures, rather than just genera, which requires some technicalities that Tucker is able to avoid.

Necessary conditions for a signature to correspond to an embedded A_4-action can be deduced from [22], wherein Tucker characterizes the possible number (and parity) of branch points for an action depending on whether the origin lies inside or outside of the surface. In order to prove sufficiency, we present a schema for constructing group actions for each possible signature satisfying the appropriate conditions. Specifically, we first introduce eight different *base cases*—embedded surfaces on which A_4 acts as a group of rigid symmetries—and then describe three different operations that can be applied to these base cases (and their corresponding signatures) to construct new embedded surfaces on which A_4 acts (with new signatures). Four of the base cases are the surfaces $\mathbf{T}_{(0;2,2,3,3)}$, $\mathbf{T}_{(0;2,2,2,3,3)}$, $\mathbf{T}_{(0;2,3,3,3,3)}$, and $\mathbf{T}_{(0;3,3,3,3)}$ constructed in Examples 2, 3, 4, and 5, respectively and are reproduced in Figure 10 for reference.

The remaining four base cases are built by adding genus to the surfaces $\mathbf{T}_{(0;2,2,3,3)}$, $\mathbf{T}_{(0;2,2,2,3,3)}$ in such a way as to preserve symmetries of the surface. For brevity, Table 1 summarizes all the relevant information we need regarding these latter four cases: it provides notation for each new case, the genus of the corresponding surface (which is calculated by summing any holes added during construction from Examples 2 and 3), a brief description of how the case is constructed from previous cases, and a visual representation of each.

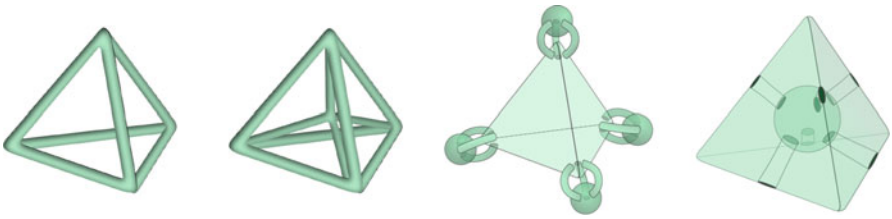


Fig. 10 From left to right: the inflated tetrahedron, also known as $\mathbf{T}_{(0;2,2,3,3)}$, the tetrahedron with jack, $\mathbf{T}_{(0;2,2,2,3,3)}$, the tetrahedron with tripods, $\mathbf{T}_{(0;2,3,3,3,3)}$, and the doubled tetrahedron with drilled holes, $\mathbf{T}_{(0;3,3,3,3)}$.

Table 1 Base cases for construction of embeddable A_4-actions.

Notation	Genus	Construction	Picture
$\mathbf{T}_{(1;2,2)}$	7	Drill holes through all vertices of $\mathbf{T}_{(0;2,2,3,3)}$ so every axis of rotation through a vertex no longer intersects the surface	
$\mathbf{T}_{(1;3,3)}$	9	Drill holes through all midpoints of edges of $\mathbf{T}_{(0;2,2,3,3)}$ so every axis of rotation through midpoints of edges no longer intersects the surface	
$\mathbf{T}_{(1;2,3,3)}$	12	Drill holes through all midpoints of edges of $\mathbf{T}_{(0;2,2,2,3,3)}$ so every axis of rotation through midpoints of edges now intersects the surface only through the central "jack"	
$\mathbf{T}_{(2;-)}$	13	Drill holes through all midpoints of edges and vertices of $\mathbf{T}_{(0;2,2,3,3)}$ so no axis of rotation intersects the surface	

Next we introduce the operations we will use to build group actions with specified signatures. We describe the geometric construction for each of these operations as well as the algebraic implications for the signature.

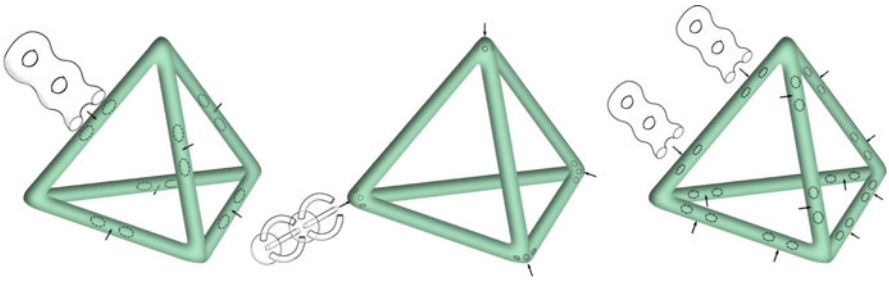


Fig. 11 Operations for building new surfaces that admit embedded A_4-actions. Depicted from left to right are the handle operator $\mathscr{H}^3$, the tripod operator $\mathscr{T}^2$, and the generalized handle operator $\mathscr{G}^2$.

Handle Operator, $\mathscr{H}$ Starting with any base case $\mathscr{B}$ and $n \geq 1$ we define the *handle operator* of order n, denoted $\mathscr{H}^n$, as follows. Let M_n denote n conjoined handles (*i.e.*, an n-holed torus with one end removed to reveal a pair of pant legs), as pictured in Figure 11 (left). Attach a copy of M_n to each edge of $\mathscr{B}$, centered about the midpoint, in such a way that any element of A_4 that switches two midpoints also maps copies of M_n onto each other identically and any rotation about the midpoint of an edge rotates the multi-handle onto itself. By construction, A_4 acts on this surface, though the genus of $\mathscr{H}^n(\mathscr{B})$ is larger than the genus of $\mathscr{B}$ by $6n$ (since we have added a genus n handle at each of 6 edges). We have also added $2n$ branch points of order 2, orbit representatives being the intersection points of a specific M_n with the axis of rotation passing through the midpoint of the corresponding edge of $\mathscr{B}$. Since the number of branch points of order 3 has not changed, we have all numerical information needed to apply the Riemann–Hurwitz formula. One calculates that the quotient genus of $\mathscr{H}^n(\mathscr{B})/A_4$ is the same as the genus of $\mathscr{B}/A_4$. Therefore, given the signature $\mathscr{S}$ of any base case, the handle operator $\mathscr{H}^n$ constructs a surface on which A_4 acts with signature $\mathscr{R}$ obtained by appending $2n$ 2's to the signature $\mathscr{S}$. By abuse of notation, we let $\mathscr{H}^n$ also denote the corresponding operator on signatures, *i.e.*, $\mathscr{H}^n(\mathscr{S}) = \mathscr{R}$.

Tripod Operator, $\mathscr{T}$ Starting with any base case $\mathscr{B}$, we define the *tripod operator* of order n for $n \geq 1$, denoted $\mathscr{T}^n$, as follows. Let P_n denote n conjoined tripods, as in Figure 11 (center), which depicts P_2. Attach a copy of P_n at every vertex of $\mathscr{B}$ in such a way that any element of A_4 that switches two vertices maps the corresponding tripods onto each other and any rotation through a vertex rotates each P_n onto itself. By construction, A_4 acts on this new surface, though $\mathscr{T}^n$ increases the genus of $\mathscr{B}$ by $8n$ (a single tripod P_1 adds genus two at each vertex, so P_n adds genus $2n$ at each vertex). We have also added $2n$ branch points of order 3, orbit representatives being the intersection points of a specific attached tripod with the axis of rotation passing through the corresponding vertex of $\mathscr{B}$. Since the number of branch points of order 2 has not changed, we again apply the Riemann–Hurwitz formula and find that the genus of the quotient $\mathscr{T}^n(\mathscr{B})/A_4$ is the same as the genus of $\mathscr{B}/A_4$. Therefore,

given any base case $\mathscr{B}$ and corresponding signature $\mathscr{S}$, the tripod operator $\mathscr{T}^n$ constructs an embedded surface on which A_4 acts with signature $\mathscr{R}$ obtained by appending $2n$ 3's to the signature $\mathscr{S}$. As before, we write $\mathscr{T}^n(\mathscr{S}) = \mathscr{R}$.

Generalized Handle Operator, $\mathscr{G}$ Our last operation generalizes the first. Starting with any of the base cases $\mathscr{B}$, we again use copies of M_n (n conjoined handles) and define the *generalized handle operator* of order n for $n \geq 1$, denoted $\mathscr{G}^n$, as follows: we attach twelve copies of M_n to $\mathscr{B}$ positioned so they do not intersect any of the axes of rotation of A_4 and such that any element of A_4 identically maps the copies of M_n onto each other as in Figure 11 (right). By construction, A_4 still acts on this surface, but $\mathscr{G}^n$ has increased the genus of $\mathscr{B}$ by $12n$ (since we have added n handles at each of 12 described points). Since no copy of M_n intersects an axis of rotation for A_4, no additional branch points have been added and we can again apply the Riemann–Hurwitz formula to find that the genus of $\mathscr{G}^n(\mathscr{B})/A_4$ is $h + 1$ where h is the genus of $\mathscr{B}/A_4$. Therefore, given any base case $\mathscr{B}$ and its corresponding signature $\mathscr{S}$, the generalized handle operator $\mathscr{G}^n$ constructs an embedded surface on which A_4 acts with signature $\mathscr{R}$ obtained by adding 1 to the genus in the signature $\mathscr{S}$. As before we write $\mathscr{G}^n(\mathscr{S}) = \mathscr{R}$.

Remark 1. These three operators can be applied repeatedly to base cases and hence can be applied to surfaces constructed from the base cases in the ways described above. Moreover, these operations are independent in the sense that changing the order of composition of operations does not change the resulting surface. The schema we shall use to construct surfaces on which A_4 acts will cite this fact. Notationally, we represent consecutive application of operations using standard function composition notation.

Before we provide sufficient conditions, we consider an explicit example to illustrate the basic argument at hand.

Example 14. Consider the surface $\mathbf{T}_{(2;-)}$ (pictured last in Table 1). If we apply each of the operators $\mathscr{H}$, $\mathscr{T}$, and $\mathscr{G}$ consecutively to this surface, we obtain a new surface $\mathscr{G}(\mathscr{T}(\mathscr{H}(\mathbf{T}_{(2;-)})))$ obtained by adding a handle at each midpoint of each edge, a tripod at each vertex, and an additional twelve handles, each straddling the point halfway between a vertex and the midpoint of an edge. Adding all these additional handles to the original genus of $\mathbf{T}_{(2;-)}$, we see that the genus of $\mathscr{G}(\mathscr{T}(\mathscr{H}(\mathbf{T}_{(2;-)})))$ is $13 + 6 + 8 + 12 = 39$. The addition of branch points and genus to the quotient means that A_4 acts on this surface with signature $(3; 2, 2, 3, 3)$.

We are now ready to provide necessary conditions on the signature of an embeddable A_4 group action.

Theorem 3. *There is an embedded A_4-action on a compact Riemann surface of some genus $g \geq 2$ with signature $(h; 2^P, 3^Q)$ satisfying Theorem 2 if and only if:*

1. *Q is even; and,*
2. *if P is odd, then $Q \geq 2$.*

Proof. As noted above, necessity follows from [22]. To show sufficiency, we construct an embedded surface on which A_4 acts for each allowable signature. We proceed by cases depending upon the parity of P and the value of h.

First suppose that P is even. Then for any $h \geq 2$ and any P, A_4 acts as an embedded group on the surface $\mathscr{G}^{h-2}(\mathscr{H}^{\frac{P}{2}}(\mathscr{T}^{\frac{Q}{2}}(\mathbf{T}_{(2;-)})))$ with signature $(h; 2^P, 3^Q)$. If $h = 1$ and $P \neq 0$, then A_4 acts as an embedded group on the surface $\mathscr{H}^{\frac{P-2}{2}}(\mathscr{T}^{\frac{Q}{2}}(\mathbf{T}_{(1;2,2)}))$ with signature $(1; 2^P, 3^Q)$, and if $h = 1$ and $Q \neq 0$, then A_4 acts as an embedded group on the surface $\mathscr{H}^{\frac{P}{2}}(\mathscr{T}^{\frac{Q-2}{2}}(\mathbf{T}_{(1;3,3)}))$ with signature $(1; 2^P, 3^Q)$. Note that when $h = 1$, we cannot have both $P = 0$ and $Q = 0$, as this signature was excluded by Theorem 2. For $h = 0$, we first note that by Theorem 2, we cannot have $Q = 0$. Now if $P > 0$, then the surface $\mathscr{H}^{\frac{P-2}{2}}(\mathscr{T}^{\frac{Q-2}{2}}(\mathbf{T}_{(0;2,2,3,3)}))$ is an embedded surface on which A_4 acts with signature $(0; 2^P, 3^Q)$. Finally, if $P = 0$, then we must have $Q \geq 4$, and in this case, the surface $\mathscr{T}^{\frac{Q-4}{2}}(\mathbf{T}_{(0;3,3,3,3)})$ is an embedded surface on which A_4 acts with signature $(0; 3^Q)$.

Now suppose that P is odd. By Theorem 2 we know that $Q > 0$. Now if $h \geq 1$ then A_4 acts as an embedded group on the surface $\mathscr{G}^{h-1}(\mathscr{H}^{\frac{P-1}{2}}(\mathscr{T}^{\frac{Q-2}{2}}(\mathbf{T}_{(1;2,3,3)})))$ with signature $(1; 2^P, 3^Q)$. If $h = 0$, to obtain a valid signature, we must have either $Q \geq 4$ or $P \geq 3$. If $Q \geq 4$, then A_4 acts as an embedded group on the surface $\mathscr{H}^{\frac{P-1}{2}}(\mathscr{T}^{\frac{Q-4}{2}}(\mathbf{T}_{(0;2,3,3,3,3)}))$ with signature $(0; 2^P, 3^Q)$ and if $P \geq 3$, then A_4 acts as an embedded group on the surface $\mathscr{H}^{\frac{P-3}{2}}(\mathscr{T}^{\frac{Q-2}{2}}(\mathbf{T}_{(0;2,2,2,3,3)}))$ with signature $(0; 2^P, 3^Q)$.

Since this includes all possible signatures given in Theorem 3, it follows that these conditions are also sufficient conditions for the signature to correspond to an embedded action of A_4 on a classical surface S, thus completing the proof. □

We note that for a given signature satisfying Theorem 3, it is not necessary that *all* actions producing that signature embed, only that there exists at least one action that does embed. We finish with an example comparing signatures for embeddable and non-embeddable actions in the case of a fixed genus.

Example 15. In Example 13 we showed that the signatures $(0; 2^1, 3^6)$, $(0; 2^5, 3^3)$, $(1; 2^1, 3^3)$, $(1; 2^5)$, $(2; 2^1)$ are all signatures for the action of A_4 on a surface of genus 16. Applying Theorem 3, we see that only one of these signatures, $(0; 2^1, 3^6)$, is a valid signature for an embedded A_4-action.

Exercise 7. Use Theorem 3 to determine which (if any) of the allowable signatures you generated in Exercise 6 actually correspond to embedded actions.

Challenge Problem 1. Use Theorem 3 to recreate Tucker's result from [22]. Namely, that embeddable A_4-actions exist only on surfaces of genus $g = 6m+8n+k$, where m, n are positive integers and $k = 0, 3, 5, 7$.

5 Suggested Projects

There are a number of interesting projects for which our exposition lays much of the necessary groundwork. For example, in Theorem 2 we provided necessary and sufficient conditions for a signature to be a signature for an action of A_4 on some compact Riemann surface (not necessarily embedded). Similar results have been developed for other classes of groups, such as cyclic groups (see [8]). Recreating such results would substantially aid in developing the tools and techniques required to work effectively with signatures of group actions.

Challenge Problem 2. Provide necessary and sufficient conditions for a signature to be associated with the action of a cyclic group $\mathbb{Z}_n$ of order n on some compact Riemann surface.

There are, of course, many families of groups for which such conditions are not known, and this same question could be asked of any such family.

Research Project 1. Fix a specific group G, or a family of groups. What signatures can be realized as resulting from an action of this group (or family of groups) on a Riemann surface?

Additionally, one could examine questions of embeddability of actions of the other finite subgroups of the isometry group of $\mathbb{R}^3$. Much is already known about the embeddability of cyclic and dihedral groups [13,20] but one could explore these questions for the symmetric group, S_4, or the alternating group, A_5.

Research Project 2. Starting with its action on a cube (or octahedron), describe all embeddable signatures of S_4.

Research Project 3. Starting with its action on a dodecahedron (or icosahedron), describe all embeddable signatures of A_5.

Colloquially speaking, Theorems 2 and 3 taken together suggest that about half of all possible signatures for A_4-actions are actually embedabble (modulo a few genus $h = 0$ cases). In addition, in [22], Tucker determined the *genus spectrum of embeddable A_4-actions*, or the sequence of all possible genera admitting embedded

A_4-actions, but did not consider how many allowable signatures embed for a given genus. It would be interesting to explore this further. For example, we found earlier in Example 13 that there are five allowable signatures corresponding to A_4-actions on a surface of genus 16, but only one of these signatures corresponds to an embedded action. One might want to ask, do approximately half of the allowable signatures in each genus embed, on average? Or does the ratio of embedded to non-embedded actions depend a great deal on the specific genus one begins with?

Research Project 4. Examine how the signatures for embedded A_4-actions are distributed across the genus spectrum of embeddable A_4-actions.

Of course, a similar investigation can be undertaken for S_4 and A_5 upon the completion of Projects 1 and 2.

Research Project 5. Examine how the signatures for embedded S_4 and A_5-actions are distributed across the genus spectrum of embeddable S_4 and A_5-actions

Considered more generally, the *genus spectrum* of an arbitrary group G is the set of all genera for which there exists a G-action on some surface of that genus. In [11], it is shown that eventually (*i.e.*, as the genus grows large), the genus spectrum for a group G can be completely predicted. For relatively small genus though, it is more difficult, and determining the genus spectrum up to the point where it stabilizes is still of significant interest. Currently there are only a few classes of groups for which the entire genus spectrum is known, see for example [12] for cyclic groups of prime power order, [21] for semi-direct products of cyclic primes groups and [15] for other certain special groups of prime power order. It would be a very useful exercise in applying the techniques we have described to recreate some of these initial results on genus spectra.

Challenge Problem 3. Determine the genus spectrum of the cyclic group $\mathbb{Z}_2$.

Challenge Problem 4. For a fixed prime p, determine the genus spectrum of the cyclic group $\mathbb{Z}_p$.

Using the techniques developed in these challenge problems, it might be possible to take a specific fixed group for which the genus spectrum is not known and study in detail its genus spectrum.

Research Project 6. For a given group G, what is the genus spectrum for G (*i.e.*, the possible genera of surfaces that admit a G-action)?

Additionally, though we have restricted our focus to conformal automorphisms here (*e.g.*, rigid rotations of a tetrahedron), reflections also give rise to symmetries, though in this case they are *anticonformal*. For a more ambitious project, it would be interesting to pursue similar questions about embeddability for groups admitting both conformal and anticonformal automorphisms. A starting point for such a project would be a closer look at the finite transformation groups of $\mathbb{R}^3$, for which we suggest [16] as a good starting reference. Some progress has been made toward this goal for cyclic and dihedral groups (see [5,13,14,18–20]) and we suggest using these articles as a starting point for such an analysis.

Research Project 7. Examine the embeddability questions above in the context of both conformal and anticonformal symmetries.

References

1. Arbo, M., Benkowski, K., Coate, B., Nordstrom, H., Peterson, C., Wootton, A.: The genus level of a group. Involve **2**(3), 323–340 (2009)
2. Benim, R.: Classification of quasiplatonic Abelian groups and signatures. Rose-Hulman Undergrad. Math. J. **9**(1), Article 1 (2008)
3. Broughton, S.A.: Classifying finite group actions on surfaces of low genus. J. Pure Appl. Algebra **69**, 233–270 (1990)
4. Broughton, S.A., Haney, D.M., McKeough, L.T., Mayfield, B.S.: Divisible tilings in the hyperbolic plane. New York J. Math. **6**, 237–283 (2000)
5. Costa, A.F.: Embeddable anticonformal automorphisms of Riemann surfaces. Comment. Math. Helv. **72**(2), 203–215 (1997)
6. Garsia, A.M.: An embedding of closed Riemann surfaces in euclidean space. Comment. Math. Helv. **35**, 93–110 (1961)
7. Ghrist, R., Peterson, V.: The geometry and topology of reconfiguration. Adv. Appl. Math. **38**(3), 302–323 (2007)
8. Harvey, W.J.: Cyclic groups of automorphisms of a compact Riemann surface. Q. J. Math. Oxford Ser. (2) **17**, 86–97 (1966)
9. Hillman, A.P., Alexanderson, G.L.: Abstract Algebra: A First Undergraduate Course. Waveland Press, Inc., Prospect Heights, IL (1994)
10. Hurwitz, A.: Ueber algebraische Gebilde mit eindeutigen Transformationen in sich. Math. Ann. **41**(3), 403–442 (1892)
11. Kulkarni, R.S.: Symmetries of surfaces. Topology **26**(2), 195–203 (1987)
12. Kulkarni, R.S., Maclachlan, C.: Cyclic p-groups of symmetries of surfaces. Glasgow Math J. **33**, 213–221 (1991)
13. Lin, F.L.: Embeddable dihedral groups of Riemann surfaces. Chinese J. Math. **7**(2), 133–152 (1979)

14. Lin, F.L.: Conformal symmetric embeddings of compact Riemann surfaces. Tamkang J. Math. **10**(1), 97–103 (1979)
15. Maclachlan, C., Talu, Y.: p-groups of symmetries of surfaces. Michigan Math. J. **45**, 315–332 (1998)
16. Martin, G.: Transformation Geometry: An Introduction to Symmetry. Undergraduate Texts in Mathematics. Springer, New York (1982)
17. Miranda, R.: Algebraic Curves and Riemann Surfaces. Graduate Studies in Mathematics, vol. 5. American Mathematical Society, Providence, RI (1995)
18. Rüedy, R.A.: Deformations of embedded Riemann surfaces. In: Advances in the Theory of Riemann Surfaces (Proceedings of Conference, Stony Brook, NY, 1969) Annal of Mathematics Studies, vol. 66, pp. 385–392. Princeton University Press, Princeton, NJ (1969)
19. Rüedy, R.A.: Embeddings of open Riemann surfaces. Comm. Math. Helv. **46**, 214–225 (1971)
20. Rüedy, R.A.: Symmetric embeddings of Riemann surfaces. In: Discontinuous Groups and Riemann Surfaces (Proceedings Conference, University of Maryland, College Park, MD, 1973), vol. 79, pp. 409–418. Annals of Mathematical Studies. Princeton University Press, Princeton, NJ (1974)
21. Sullivan, C.O., Weaver, A.: A Diophantine Frobenius problem related to Riemann surfaces. Glasgow Math J. **53**, 501–522 (2011)
22. Tucker, T.W.: Two notes on maps and surface symmetry. In: Rigidity and Symmetry. Fields Institute Communications, vol. 70, pp. 345–355. Springer, New York (2014)
23. Vinroot, C.R.: Symmetry and tiling groups for genus 4 and 5. Rose-Hulman Undergrad. Math. J. **1**(1), Article 5 (2000)
24. Zarrow, R.: Anticonformal automorphisms of compact Riemann surfaces. Proc. Am. Math. Soc. **54**, 162–164 (1976)

Tile Invariants for Tackling Tiling Questions

Michael P. Hitchman

Suggested Prerequisites. *Number theory, Group theory.*

1 Prologue

We begin with a question: For which values of n do translations of the two tiles in Figure 1 tile the staircase region S_n with n steps?

The staircase problem, which appeared in a 1990 paper by Conway and Lagarias [5], exhibits all that we find engaging about the mathematics of tiling: it is simple to state, simple to investigate, yet challenging to answer. Moreover, the inventive solution in [5], which makes use of combinatorial group theory and planar topology, sparked considerable research in the mathematics of tiling, as we shall see.

We can begin investigating the staircase problem by trying to tile some small staircase regions. In short order, one can find tilings of S_2, S_9, S_{11}, and S_{12}. Do it! A tiling of S_9 appears in Figure 1. On the other hand, regions like S_7 can be eliminated immediately for having the wrong area: since each tile has area 3, any tileable region must have area divisible by 3. Other regions, such as S_8, do not appear to be tileable, even though they meet this area requirement.

We might then write down some generalities. The area of S_n is $n(n+1)/2$ which is divisible by 3 if and only if $n \equiv 0$ or 2 (mod 3); that is, n has remainder 0 or 2 when divided by 3. So S_n is not tileable if $n \equiv 1$ (mod 3). We also see that the 2×3 and 3×2 rectangles are tileable, from which we can work out that an $n \times 12$ rectangle is also tileable if $n \equiv 0$ or 2 (mod 3), because such rectangles can be built from these 2×3 and 3×2 bricks. From this fact we can show that if S_n is tileable,

M.P. Hitchman (✉)
Linfield College, 900 SE Baker St., McMinnville, OR 97128-6894, USA
e-mail: mhitchm@linfield.edu

A. Wootton et al. (eds.), *A Primer for Undergraduate Research*, Foundations for Undergraduate Research in Mathematics, DOI 10.1007/978-3-319-66065-3_3

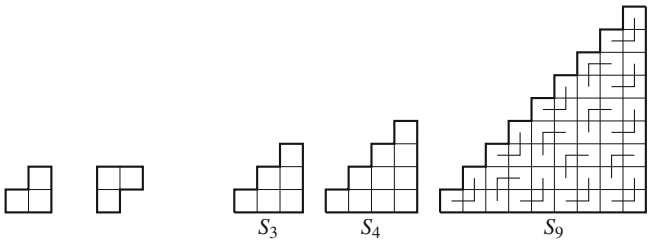
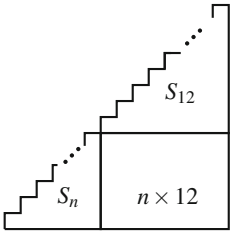


Fig. 1 Copies of these two L-shaped tiles can tile the staircase region S_9.

Fig. 2 Partitioning S_{n+12}.



then so is S_{n+12}, because we can partition S_{n+12} into three tileable subregions as in Figure 2. Take the time to check these facts.

With tilings of S_2, S_9, S_{11}, and S_{12} in hand, along with the fact that if S_n is tileable then so is S_{n+12}, we have a proof that if $n \equiv 0, 2, 9$, or 11 (mod 12) then S_n is tileable by the L-shaped tiles in Figure 1.

Conway and Lagarias prove that the converse is also true by ruling out tilings in the other cases. We know that if $n \equiv 1, 4, 7$, or 10 (mod 12) then S_n is not tileable because its area is not divisible by 3. The nut of the problem appears to be when $n \equiv 3, 5, 6$, or 8 (mod 12). We can imagine providing a brute-force proof that S_8, say, is not tileable, one that tracks each attempt at a tiling until it runs into some impossibility due to the shape of the remaining untiled squares. But how about S_{1208}? A brute-force nonexistence proof is out of the question. What we require is another tile invariant, some necessary feature of any tiling of S_n that can be used to rule out tilings in these remaining cases.

Here is where Conway and Lagarias' approach to this problem becomes so inventive. As Figure 3 shows, if we expand the tile set to include the straight tiles t_1 and t_4, then each of the regions S_3, S_5, S_6 and S_8 can be tiled. To match notation we develop later, we call this expanded tile set T_3 (the subscript refers to the number of squares in each tile), and the original tiles in the staircase problem are t_2 and t_3.

Conway and Lagarias then introduce combinatorial group-theoretic techniques to this scene, which we present in Section 3.2. For now we state their key result:

The Conway/Lagarias Invariant In any tiling of a staircase region S_n by the set T_3, the number of t_2 tiles used minus the number of t_3 tiles used is constant.

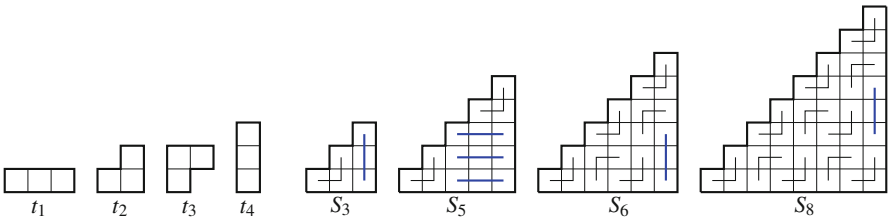


Fig. 3 The larger tile set T_3 tiles S_3, S_5, S_6 and S_8.

For instance, the tiling of S_8 in Figure 3 contains 7 copies of t_2 and 4 copies of t_3 giving us a difference of 3. The Conway/Lagarias invariant says any other tiling of S_8 by T_3 must produce this difference of 3. Similarly, any tiling of S_6 would produce the difference 2, because this is the difference in the tiling of S_6 in Figure 3.

We can use this invariant to prove S_8 is not tileable by $\{t_2, t_3\}$. If this set did tile S_8 it would do so with 12 tiles, giving us $a_2 + a_3 = 12$, where a_i equals the number of t_i tiles used in the tiling. On the other hand, we know $a_2 - a_3 = 3$, thanks to the Conway/Lagarias invariant. But no integer solutions exist to this system of equations, so no tiling of S_8 by $\{t_2, t_3\}$ exists.

This argument can be generalized to prove nonexistence of a tiling for all $n \equiv 3, 5, 6, 8 \pmod{12}$. In fact, we will provide a quick test in Section 4 (with Lemma 1) which will allow us to see immediately from the tilings in Figure 3 that S_n is not tileable by $\{t_2, t_3\}$ in all these cases.

Although the Conway/Lagarias invariant was discovered in the service of the staircase problem, considerable effort has been expended on the question of finding tile invariants associated to a tile set and a family of regions, for their own sake, independent of tileability questions. We discuss methods for finding tile invariants in Section 3 and reconnect these invariants to tileability questions in Section 4. In Section 3.4 we discuss the tile counting group, an algebraic structure that encodes information about tile invariants, first considered in [15]. In Section 5 we touch briefly on questions of enumeration, before offering some concluding remarks. Project ideas, questions, exercises and challenge problems appear throughout the paper to engage the reader in the mathematics of tiling.

2 Tiling Basics

All the tile sets and regions considered here are polyominoes. A *polyomino* is the result of joining one or more equal squares edge to edge. The *area* of a region R, denoted $|R|$, equals its number of squares. *Dominoes*, *trominoes*, *tetrominoes*, and *pentominoes* have area two, three, four, and five, respectively. Figure 4 shows three 8-ominoes.

We assume our polyominoes live in the integer lattice, the plane tiled by unit squares whose corners have integer coordinates. We let $[a \times b]$ denote a rectangle of height a and width b in the lattice. To identify a particular cell in the lattice, let $s_{x,y}$ denote the cell whose lower left corner is the point (x, y).

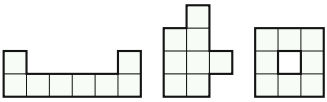


Fig. 4 Three 8-ominoes.

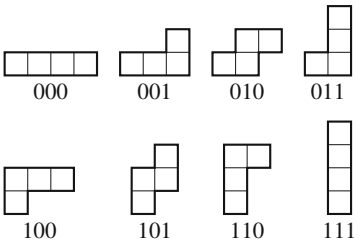


Fig. 5 The set T_4 of ribbon tile tetrominoes, labelled by their binary signatures.

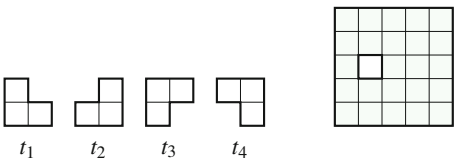


Fig. 6 L-trominoes do not tile this deficient rectangle.

A *ribbon tile* of area n is a polyomino consisting of n squares laid out in a path, such that from an initial square, each step either goes up or to the right. We let T_n denote the set of area n ribbon tiles, and note that T_n has 2^{n-1} elements. The tiles in T_n can be indexed by binary signatures of length $n-1$, where each 0 represents a step to the right, and 1 represents a step up. The set T_4 of ribbon tile tetrominoes appears in Figure 5 along with each tile's binary signature. The set T_3 is the (expanded) set of tiles considered in the staircase problem.

Suppose T is a tile set. We say a region R is *tileable by* T, or T *tiles* R, if R can be covered without gaps or overlaps by a collection of translations of some (or all) of the tiles in T. We note that rotations or reflections of the tiles are not allowed in any of the problems in this article. (If we want to allow for rotations or reflections, we ought to put these shapes in the tile set.)

Example 1. A *deficient rectangle* is a rectangle with a single cell missing. In [7] Golomb showed that the set of four L-trominoes in Figure 6 can tile any deficient $[2^k \times 2^k]$, where $k \geq 1$. This result has an elegant proof by induction, which is now commonly found in texts teaching this proof technique.

Certainly not every deficient $[a \times b]$ is tileable by L-trominoes. For starters, they don't all have area divisible by 3. This area condition is not sufficient, though, as the deficient $[5 \times 5]$ in Figure 6 has area 24 but isn't tileable. Try covering the cell to the left of the missing cell with an L-tromino, and you immediately run into trouble in either the upper left or lower left corner. It turns out that L-trominoes do, in fact, tile most deficient rectangles of the right area. The details can be found in [2].

Roughly speaking, the tile invariants of the next section are meant to help us prove nonexistence without making use of local geometric constraints of the region itself as we did here.

Exercise 1. Show that for each $k \geq 1$ the set of four L-trominoes in Figure 6 tiles any deficient $[2^k \times 2^k]$.

3 Tile Invariants

Suppose the tile set $T = \{t_1, t_2, \ldots, t_n\}$ has n tiles, and that $\mathscr{R}$ denotes a family of regions. There may be different ways in which T tiles a region $R \in \mathscr{R}$. Suppose α represents one such tiling. We let $a_i(\alpha)$ equal the number of times tile t_i appears in the tiling α. A linear combination of the $a_i(\alpha)$ whose value depends only on the region R and not the particular tiling of it, for all the regions in the family $\mathscr{R}$, is called a *tile invariant with respect to the tile set T and the family of regions* $\mathscr{R}$, or simply a *tile invariant for T over* $\mathscr{R}$.

A tile invariant has the form

$$k_1 \cdot a_1(\alpha) + k_2 \cdot a_2(\alpha) + \cdots + k_n \cdot a_n(\alpha) = c(R)$$

where each k_i is an integer and $c(R)$ is the value of the linear combination that depends only on R. Furthermore, the equality may be taken modulo m for some positive integer m. Since tile invariants are independent of the particular tiling, we often omit the α from their notational descriptions.

Three commonly studied families of regions are $\mathscr{R}_{\text{all}}$, the set of all regions; $\mathscr{R}_{\text{sc}}$, the set of simply connected regions; and $\mathscr{R}_{\text{rect}}$, the set of rectangular regions. Simply connected regions in the lattice are those that are connected (in one piece) and have connected boundary. Regions that enclose a hole such as the region in Figure 7 are not simply connected. Note that $\mathscr{R}_{\text{rect}} \subset \mathscr{R}_{\text{sc}} \subset \mathscr{R}_{\text{all}}$.

We emphasize that tile invariants depend on the tile set as well as the family of regions. The Conway/Lagarias invariant for T_3 applies to the family of simply connected regions $\mathscr{R}_{\text{sc}}$, but not $\mathscr{R}_{\text{all}}$. Figure 7 shows two tilings of a non-simply connected region that generate different values for the linear combination $a_2 - a_3$: $a_2(\alpha) - a_3(\alpha) = 2$, while $a_2(\beta) - a_3(\beta) = -1$. We note that this example does not rule out the possibility that $a_2 - a_3$ is constant modulo 3 over $\mathscr{R}_{\text{all}}$. In fact this turns out to hold, as we see in Example 3.

Example 2. The area invariant.

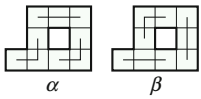


Fig. 7 The Conway/Lagarias invariant doesn't hold for T_3 over all regions.

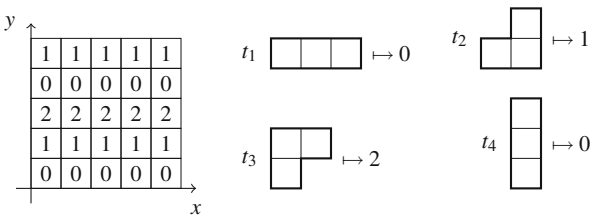


Fig. 8 Coloring the lattice with horizontal strips of like colors.

Suppose the area of each tiles in $T = \{t_1, \ldots, t_n\}$ is m. Then we have the following tile invariant, called the *area invariant*: For each region $R \in \mathscr{R}_{\text{all}}$, if α is a tiling of R,

$$a_1(\alpha) + a_2(\alpha) + \cdots + a_n(\alpha) = \frac{|R|}{m}.$$

3.1 Coloring Invariants

Imagine "coloring" each cell in the lattice with an integer. Any region R then has a color sum, found by summing the numbers in the cells of R. Now suppose we color the lattice in such a way that for each $t \in T$, the color sum of t is independent of where the tile t is placed in the lattice, perhaps modulo m for some m, (remembering once more that we cannot rotate or reflect t before placing it in the lattice). Such a coloring would determine a tile invariant for T over $\mathscr{R}_{\text{all}}$.

Example 3. Color the lattice with horizontal strips of 0s, 1s and 2s as in Figure 8 (the cell $s_{x,y}$ in the lattice gets color y, modulo 3), and consider the tile set T_3. The color sum of t_1, wherever it is placed, is equivalent to 0 (mod 3). Tile t_2 has color sum 1 (mod 3); t_3 has color sum 2 (mod 3), and t_4 has color sum 0 (mod 3).

Let $c(R)$ represent the color sum of a region R tileable by T_3. This value is independent of any tiling of R, but given a tiling α, $c(R)$ will be the sum of the color sums of the tiles in α. It follows that $0a_1 + a_2 + 2a_3 + 0a_4 \equiv c(R)$ (mod 3). Since $2 \equiv -1$ (mod 3) we may express the tile invariant determined by this coloring as

$$a_2 - a_3 \equiv c(R) \pmod 3.$$

Note that If we restrict to simply connected regions, this coloring tile invariant is an immediate consequence of the celebrated Conway/Lagarias invariant.

Example 4. We establish three tile invariants with respect to the set L of 4 L-trominoes in Figure 6 over $\mathscr{R}_{\text{all}}$.

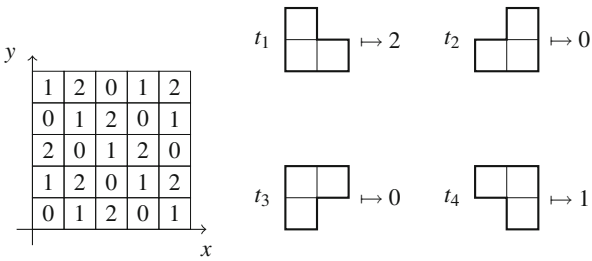


Fig. 9 Coloring cell $s_{x,y}$ with $x + y$ (mod 3).

$$a_1 + a_2 + a_3 + a_4 = \frac{|R|}{3} \tag{1}$$

$$a_4 - a_1 \equiv c(R) \pmod 3 \tag{2}$$

$$a_1 + a_2 + 2a_3 + 2a_4 \equiv d(R) \pmod 3 \tag{3}$$

The first invariant is the area invariant. The second invariant follows from the coloring in which $s_{x,y}$ gets color $x + y$, taken modulo 3 as pictured in Figure 9. The coloring in Figure 8 yields the third invariant $a_1 + a_2 + 2a_3 + 2a_4 \equiv c \pmod 3$ with respect to L.

We may form new tile invariants by combining these three in various ways. For instance, we obtain $a_2 - a_3 \equiv c_*(R) \pmod 3$ by adding invariants (2) and (3). A natural question to ask is whether there are tile invariants in this setting that are not consequences of the three above. In Section 3.4 we investigate the question of determining all tile invariants when we consider the tile counting group.

3.2 Boundary Word Invariants

In this section we introduce the boundary word approach first used by Conway and Lagarias in [5]. We assume throughout the section that all tiles and regions are simply connected.

Begin by orienting and labelling the edges in the lattice as follows: horizontal edges are labelled x and oriented to the right; vertical edges are labelled y and oriented up. Any path in the lattice determines a word, using the alphabet $\{x, y, x^{-1}, y^{-1}\}$, by writing the letters you encounter as you proceed along the path, following the convention that if you traverse an edge against its orientation, you record the inverse of the label. These words determine elements of the free group F on two generators x and y.

We associate a boundary word $w(R)$ to any simply connected region R by traversing a clockwise path around the boundary of R from a chosen basepoint. In this way, we give each tile t in a tile set T a tile boundary word $w(t)$. For instance, the tile boundary words of the tiles in T_3 (we choose bottom left-most basepoints) are as indicated in Figure 10.

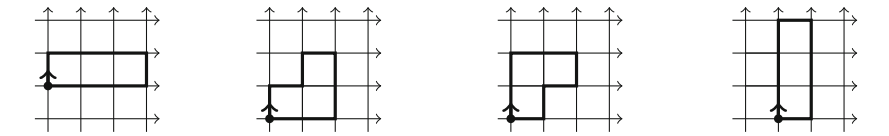


Fig. 10 Boundary words of the tiles in T_3.

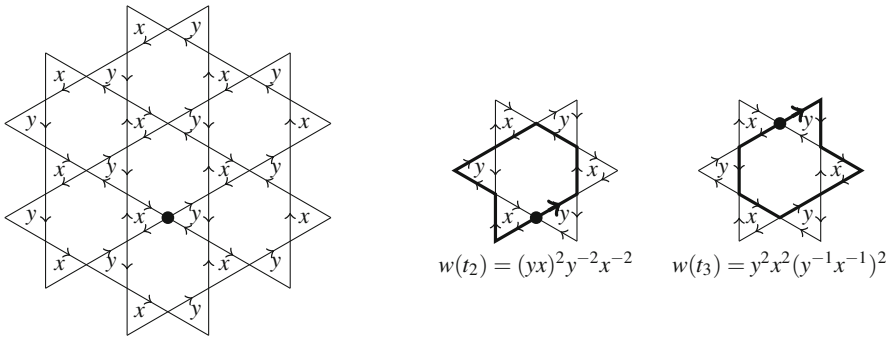


Fig. 11 The Cayley Graph of F/H.

Conway and Lagarias then discovered an alternate grid of streets labelled x and y along which words in F trace new paths. Instead of streets forming rectangular blocks as in the integer lattice, the blocks in the new setting are made of hexagons and triangles, a portion of which is shown in Figure 11. The key feature of this new grid is that the tile boundary words from Figure 10 still trace closed paths. The images of $w(t_2)$ and $w(t_3)$ appear in Figure 11. Check that the images of $w(t_1)$ and $w(t_4)$ are also closed paths.

We remark that the diagram in Figure 11 is called the Cayley graph of F/H, where H is the normal subgroup of F consisting of all words that determine closed paths in this diagram. (H is the normal subgroup of F generated by the words x^3, y^3, and $(yx)^3$.) Vertices in the graph correspond to elements in F/H, and a directed edge labelled x runs from each element $g \in F/H$ to the element xg. Similarly, a directed edge labelled y runs from each $g \in F/H$ to yg. We do not need this group-theoretic interpretation of the diagram in Figure 11 in what follows, but it helps to explain how Conway and Lagarias found it: They wanted a quotient group F/H in which the tile boundary words were all trivial. Several good texts, including [12], develop the combinatorial group theory at play here.

Now comes a dash of planar topology. Conway and Lagarias define a group homomorphism $h : H \to \mathbb{Z}$ via the winding number $h(w)$ that counts the number of hexagons that the closed path w in H encloses in the counterclockwise direction minus the number of hexagons it encloses in the clockwise direction. Note that $h(w(t_2)) = +1$ because the path in Figure 11 traced by $w(t_2)$ encloses one hexagon in the counterclockwise direction, while $w(t_3) = -1$. One can check that $h(w(t_1)) = h(w(t_4)) = 0$.

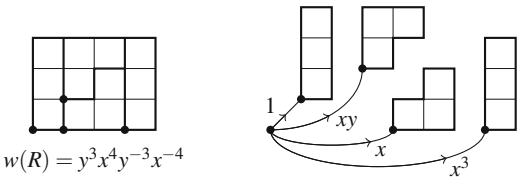


Fig. 12 A bouquet of balloons. Here $w(R) = w(t_4)\left(xyw(t_3)(xy)^{-1}\right)\left(xw(t_2)x^{-1}\right)\left(x^3w(t_4)x^{-3}\right)$.

If R is tileable by a tile set T, then it turns out that $w(R)$ can be expressed as a product of conjugates of tile boundary words. The idea is that a tiling of R by T may be visualized as a bouquet of tile-shaped balloons whose strings run from the basepoint of R to the tile basepoints, as suggested in Figure 12. The boundary word of R will then be equivalent to the word recorded by following each string in turn, going around the tile boundary, then returning along the string in order to move to the next string. This fact is a result in combinatorial group theory called van Kampen's Lemma.

So, if R is tileable by T, then $w(R)$ is in H and we can determine the integer $h(w(R))$ which is independent of any tiling of R. It counts the net number of hexagons enclosed by $w(R)$. On the other hand, if we have a tiling α of R having m tiles, then

$$w(R) = \prod_{i=1}^{m} w_i r_i w_i^{-1}$$

where each $w_i \in F$ and each r_i is a tile boundary word. Because h is a group homomorphism $h(w(R))$ will equal $\sum_{i=1}^{m} h(r_i)$. Each copy of t_2 in α will add 1 to this sum, and each copy of t_3 in α will subtract 1 from the sum, while t_1 and t_4 contribute zero. So we have arrived at the Conway/Lagarias invariant:

$$a_2(\alpha) - a_3(\alpha) = h(w(R))$$

is constant for any tiling α of a given simply connected region R. In general, this process gives us the following group-theoretic necessary condition for tileability.

Theorem 1. *Suppose H is a normal subgroup of F containing the tile boundary words of the tile set T, for some choice of tile basepoints. If a simply connected region R is tileable by T then $w(R) \in H$.*

Conway and Lagarias' choice of H is particular to the tile set T_3, but the general approach of using boundary words to generate invariants has been applied with great success elsewhere. For instance, in a very well-written and accessible paper [16], Propp applies a boundary word approach to the solution of a tiling question involving skew tetrominoes and regions called aztec diamonds. Propp found his tile invariant by first considering a larger tile set, just as Conway and Lagarias had done to solve the staircase problem.

In [10], Korn uses boundary words to find many tile invariants, including the invariant that for any simply connected region, $a_1 - a_2$ is constant for the set $\{t_1, t_2, t_7, t_8\}$ of the tetrominoes in Figure 24 (see [10, Theorem 8.3]). Other boundary word arguments can be found in [13, 14, 18], and [20]. We also note that the Conway/Lagarias invariant can be derived without the use of winding numbers by appealing instead to a topological property of a 2-complex corresponding to the quotient group F/H in their construction. This approach is worked out in [8].

Example 5. In [13] Pak and Moore use boundary words to generate tile invariants for each tile set T_n over $\mathscr{R}_{\mathrm{sc}}$. They orient and label the edges of the lattice with elements in the finite cyclic group $\mathbb{Z}_n$ and then associate to each element a vector in the complex plane in such a way that the boundary word of each tile in T_n generates a closed polygonal path in $\mathbb{C}$ whose edges are these vectors. They then use signed areas enclosed by these paths to determine tile invariants. To state the result we need a bit more notation. Recall, ribbon tiles in T_n have a binary signatures. Let t_ϵ denote the tile with signature $\epsilon = \epsilon_1\epsilon_2\cdots\epsilon_{n-1}$, where each $\epsilon_i \in \{0, 1\}$, and let $a_\epsilon(\alpha)$ denote the number of times the tile $t_\epsilon \in T_n$ appears in a tiling α.

Theorem 2 ([13, Theorem 1.2]). *Suppose $n \geq 2$. For each $1 \leq i < n/2$ we have the following tile invariants for the set T_n over $\mathscr{R}_{\mathrm{sc}}$:*

$$\sum_{\epsilon:\epsilon_i=0,\epsilon_{n-i}=1} a_\epsilon(\alpha) - \sum_{\epsilon:\epsilon_i=1,\epsilon_{n-i}=0} a_\epsilon(\alpha) = c_i(R).$$

Furthermore, if n is even, we have the following tile invariant:

$$\sum_{\epsilon:\epsilon_{n/2}=1} a_\epsilon(\alpha) \equiv c_*(R) \pmod 2.$$

When $n = 2$, this theorem gives one invariant: In any tiling of R by dominoes, the number of vertical dominoes is constant, modulo 2. When $n = 3$, the theorem gives the Conway/Lagarias invariant. When $n = 4$, the theorem provides two invariants, and these were proved in [14]. Moreover, we note that the invariant in the $n = 2$ case can be derived from the coloring $s_{x,y} \mapsto y \pmod 2$, but for $n \geq 3$, the invariants in this theorem cannot be found by coloring arguments [15, Theorem 1.8].

3.3 Invariants from Local Connectivity

Suppose R is a region tileable by T, and $\mathscr{L}$ is a finite set of small regions, each of which can be tiled in at least two ways. A *local move in $\mathscr{L}$* consists of replacing one tiling of a region L in $\mathscr{L}$ (that appears within a tiling of R) with another tiling of L. That is, a local move changes one tiling of R to another by removing a few tiles and then tiling the now uncovered subregion in a different way. We say R has *local connectivity with respect to $\mathscr{L}$ and T* if any tiling of R can be converted to any

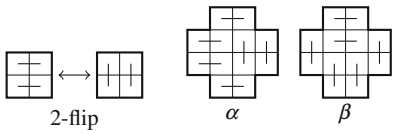


Fig. 13 Any two domino tilings of a simply connected region can be made to agree by a finite number of 2-flips.

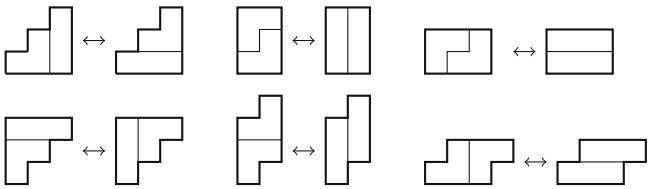


Fig. 14 $\mathscr{R}_{\text{sc}}$ has a local move property with respect to T_3 consisting of these six 2-flips.

other by performing a finite number of local moves in $\mathscr{L}$. Finally, we say there is a *local move property for T and* $\mathscr{R}$ if there exists a set of local moves $\mathscr{L}$ such that all regions $R \in \mathscr{R}$ have local connectivity with respect to $\mathscr{L}$ and T.

For instance, consider the set T_2 of dominoes and let $\mathscr{L} = \{[2 \times 2]\}$. This square has two tilings: one by two horizontal dominoes, the other by two vertical dominoes, as shown in Figure 13. A local move in $\mathscr{L}$ then consists of swapping two horizontal dominoes that cover a 2×2 square with two vertical dominoes covering that same square, or vice versa. We call such a move a 2-flip. Thurston proved in [20] that any two tilings of a simply connected region can be made to agree by a finite sequence of 2-flips. That is, there is a local move property for dominoes and simply connected regions. The reader may check that the tiling α in Figure 13 can be converted to the tiling β by performing three 2-flips.

Example 6. In [19, Theorem 1.1] Sheffield generalized Thurston's result by proving the remarkable theorem that for each set T_n of ribbon tiles the set $\mathscr{R}_{\text{sc}}$ has a local move property with respect to a finite set of 2-flips, thus resolving a conjecture appearing in [13]. For instance, for T_3, any simply connected region R has local connectivity with respect to the set of six local moves in Figure 14.

With this theorem in hand, the Conway/Lagarias invariant over $\mathscr{R}_{\text{sc}}$ follows immediately: Suppose α is a tiling of a region R. Let $a_2(\alpha) - a_3(\alpha) = c(R)$. If β is any other tiling of R then we can get to β from α by performing a finite number of local moves. Since the tile count $a_2 - a_3$ doesn't change in any of these six local moves, it follows that $a_2(\beta) - a_3(\beta) = a_2(\alpha) - a_3(\alpha)$. Thus, $a_2 - a_3 = c(R)$ is a tile invariant over $\mathscr{R}_{\text{sc}}$. The other tile invariants in Theorem 2 also follow from Sheffield's result, and with the following theorem, Sheffield effectively solves every tiling question of the form: "Does T_n tile a simply connected region R?"

Theorem 3 ([19, Theorem 1.4]). *There is a linear-time (i.e., linear in the number of squares of R) algorithm for determining whether there exists a tiling by T_n of a simply connected region R, and producing such a tiling when one exists.*

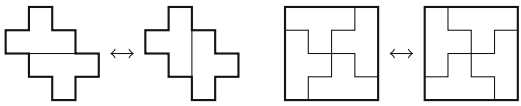


Fig. 15 Any two tilings of a rectangle by T-tetrominoes can be made to agree by a finite number of these two local moves.

Example 7. Korn proves in [10, Theorem 4.1] that the set of four T-tetrominoes has a local move property for rectangular regions. In particular, any rectangle has local connectivity with respect to the two local moves given in Figure 15.

The strongest statement one can hope to make about how any two tilings of a given region must be related is to establish local connectivity, should it exist. If we have a set $\mathscr{L}$ of local moves, it is straightforward to write down tile invariants. However, local connectivity can be difficult to prove in general. Objects called height functions have been a helpful tool in this endeavor. Height functions were first used by Thurston in [20], and were used by both Sheffield and Korn in the above examples. We do not pursue height functions here, but refer the interested reader to [10, 13, 19], and [20] for introductions to this technique.

We mention one way to prove that a family of regions $\mathscr{R}$ does not have a local move property for a given tile set. Any region in $\mathscr{R}$ that has exactly two tilings must be a local move. So, if one can demonstrate an infinite family of regions in $\mathscr{R}$, each having exactly two tilings, then a local move property does not hold for the family $\mathscr{R}$ with the tile set.

Returning to Example 7, Korn uses this approach to show that there is not a local move property for the set of T-tetrominoes and $\mathscr{R}_{\text{sc}}$. In particular, with [10, Theorem 4.13] he provides an infinite family of simply connected regions, each having exactly two tilings by T-tetrominoes. Like tile invariants, the existence of a local move property for a tile set depends on the family of regions under consideration.

3.4 The Tile Counting Group

Given a tile set T and a family or regions $\mathscr{R}$, the tile counting group $G(T, \mathscr{R})$ was introduced in [15] as a way to record information about the tile invariants associated with T and $\mathscr{R}$. This group is a quotient of the free abelian group $\mathbb{Z}^n$, where n is the size of the tile set. We think of elements of $\mathbb{Z}^n$ as n-tuples of integers, and addition is done coordinate-wise: if $w = (a_1, \ldots, a_n)$ and $v = (b_1, \ldots, b_n)$ and $k \in \mathbb{Z}$, then $w + v = (a_1 + b_1, \ldots, a_n + b_n)$, and we let kw denote $(ka_1, \ldots, ka_n)$. The tile counting group is defined as follows. To any tiling α of a region $R \in \mathscr{R}$ we associate a *tile vector* $v_\alpha \in \mathbb{Z}^n$ given by $v_\alpha = (a_1(\alpha), a_2(\alpha), \ldots, a_n(\alpha))$. Two tilings α and β of a region R determine the *difference vector* $v_\alpha - v_\beta$. Let H denote the normal subgroup of $\mathbb{Z}^n$ generated by all possible difference vectors obtainable from our family of regions $\mathscr{R}$ and the tile set T. The tile counting group is the quotient group

$$G(T, \mathscr{R}) = \mathbb{Z}^n / H.$$

In this context a tile invariant may be viewed as a linear function f of the form $f(a_1, a_2, \ldots, a_n) = \sum_{i=1}^{n} k_i a_i$ that maps elements of $\mathbb{Z}^n$ into $\mathbb{Z}$ or $\mathbb{Z}_m$, the finite cyclic group of order m, such that $f(v) = 0$ for each $v \in H$.

Example 8. With respect to the tile set L of 4 L-trominoes and the family $\mathscr{R}_{\text{all}}$ we have three tile invariants from Example 4. We show that these effectively supply all the tile invariants in this setting by using them as coordinate functions in a map that establishes the tile counting group.

Theorem 4. $G(L, \mathscr{R}_{\text{all}}) \simeq \mathbb{Z} \times \mathbb{Z}_3 \times \mathbb{Z}_3$.

Proof. Consider the group homomorphism $\phi : \mathbb{Z}^3 \to \mathbb{Z} \times \mathbb{Z}_3 \times \mathbb{Z}_3$ defined by

$$\phi(a_1, a_2, a_3, a_4) = (a_1 + a_2 + a_3 + a_4, a_4 - a_1, a_1 + a_2 + 2a_3 + 2a_4),$$

where the second and third coordinates in the image are taken modulo 3. To see that ϕ is surjective, note that $\phi(0, -1, 2, 0) = (1, 0, 0)$, $\phi(0, 0, -1, 1) = (0, 1, 0)$, and $\phi(0, -1, 1, 0) = (0, 0, 1)$. Since ϕ maps to the generators of $\mathbb{Z} \times \mathbb{Z}_3 \times \mathbb{Z}_3$, ϕ is surjective.

Since the coordinate functions of ϕ are tile invariants, $\phi(v_\alpha - v_\beta) = (0, 0, 0)$ for any difference vector, and it follows that $H \subseteq \ker \phi$. To see the reverse containment, suppose $v = (a_1, a_2, a_3, a_4) \in \ker \phi$. Then we have three equations:

$$a_1 + a_2 + a_3 + a_4 = 0 \tag{4}$$

$$a_4 - a_1 = 3k \text{ for some integer } k \tag{5}$$

$$a_1 + a_2 + 2a_3 + 2a_4 = 3l \text{ for some integer } l. \tag{6}$$

Equation (5) tells us $a_1 = a_4 - 3k$. Subtracting (4) from (6) gives $a_3 = 3l - a_4$, and substituting these expressions into Equation (4) gives us a_2 in terms of a_4. So, an arbitrary element v of $\ker \phi$ has the form

$$(a_4 - 3k, 3k - 3l - a_4, 3l - a_4, a_4) = a_4(1, -1, -1, 1) + k(-3, 3, 0, 0) + l(0, -3, 3, 0).$$

Figure 16 shows tilings of rectangles that generate these three difference vectors:

$$v_\alpha - v_\beta = (1, -1, -1, 1), v_\gamma - v_\zeta = (-3, 3, 0, 0), \text{ and } v_\eta - v_\theta = (0, -3, 3, 0),$$

(assuming these tilings agree on corresponding untiled $[2 \times 3]$ or $[3 \times 2]$ blocks). Thus, any element of $\ker \phi$ is a linear combination of difference vectors, and so is in H. Thus, ϕ is onto with kernel equal to H, and by the first isomorphism theorem of groups, $G(L, \mathscr{R}_{\text{all}}) = \mathbb{Z}^4/H \simeq \mathbb{Z} \times \mathbb{Z}_3 \times \mathbb{Z}_3$. □

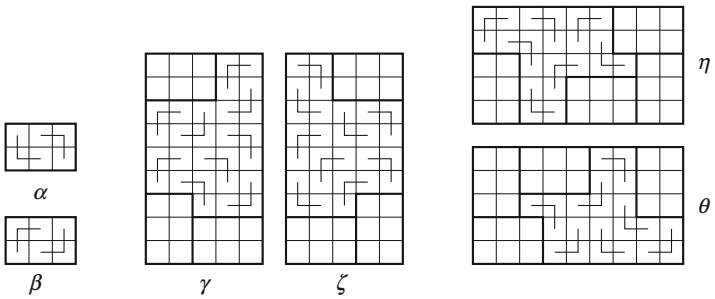


Fig. 16 Generating difference vectors for the set L.

We remark that since the regions in Figure 16 are rectangles, and the tile invariants apply to rectangles, we have also proved $G(L, \mathscr{R}_{\text{rect}}) \simeq \mathbb{Z} \times \mathbb{Z}_3 \times \mathbb{Z}_3$.

Exercise 2. Why can't we use Figure 11 to establish that $a_2 - a_3$ is invariant for the set L over $\mathscr{R}_{\text{sc}}$?

Challenge Problem 1. Prove that $G(T_3, \mathscr{R}_{\text{sc}}) \simeq \mathbb{Z}^2$ by using the area and Conway/Lagarias invariants as coordinate functions.

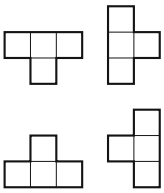
Pak demonstrated in [15, Section 6] that the tile invariants in Theorem 2 together with the area invariant generate the tile counting group for T_n over $\mathscr{R}_{\text{sc}}$.

Theorem 5. *If $n = 2m+1$ then $G(T_n, \mathscr{R}_{\text{sc}}) \simeq \mathbb{Z}^{m+1}$. If $n = 2m$ then $G(T_n, \mathscr{R}_{\text{sc}}) \simeq \mathbb{Z}^m \times \mathbb{Z}_2$.*

4 Tile Invariants and Tileability

The colorings in Section 3.1 generate tile invariants. We can look at coloring arguments in a different way to state necessary conditions for tileability of a given region. Suppose we color each cell in the lattice with an element of an abelian group G whose identity element is e. (G is often $\mathbb{Z}$ or $\mathbb{Z}_n$.) A *coloring map* $\phi : \mathscr{R}_{\text{all}} \to G$ is a function that assigns to each region R its color sum $\phi(R)$ (the sum in G of the elements covered by the region R). A *T-coloring* is a coloring map ϕ in which $\phi(t) = e$ for each tile t in the tile set T, wherever t is placed in the lattice. We then have the following necessary condition for the tileability of a region R:

Theorem 6. *Suppose $\phi : \mathscr{R}_{\text{all}} \to G$ is a T-coloring. If a region R is tileable by T then $\phi(R) = e$.*



5	1	5	1	5	1	5	1
1	5	1	5	1	5	1	5
5	1	5	1	5	1	5	1
1	5	1	5	1	5	1	5
5	1	5	1	5	1	5	1
1	5	1	5	1	5	1	5
5	1	5	1	5	1	5	1
1	5	1	5	1	5	1	5

Fig. 17 Can T-tetrominoes tile a chessboard with a $[2 \times 2]$ missing?

Example 9. Can the set of T-tetrominoes in Figure 17 tile an $[8 \times 8]$ with a $[2 \times 2]$ removed? We may color the cells of the lattice according to $s_{x,y} \mapsto 1$ if $x+y$ is even and $s_{x,y} \mapsto 5$ if $x+y$ is odd. Notice that the color sum of each T-tetromino, wherever it is placed, is equivalent to 0, modulo 8. In other words, we have a T-coloring where G is the abelian group $\mathbb{Z}_8$. Now, any $[2 \times 2]$ square covers two 1s and two 5s, so the color sum of the $[8 \times 8]$ with a $[2 \times 2]$ removed is

$$30 \cdot 1 + 30 \cdot 5 \equiv 4 \pmod 8,$$

meaning its color sum is not the identity in $\mathbb{Z}_8$, so this region is not tileable by T-tetrominoes.

Example 10. We show that $[a \times b]$ with a $[2 \times 2]$ removed is not tileable by T_4 when $a, b > 2$ and $a \equiv 0 \pmod 4$.

Consider the group $\mathbb{Z}^3$ whose identity element is $e = (0,0,0)$. Let $v_0 = (1,0,0)$, $v_1 = (0,1,0), v_2 = (0,0,1)$ and $v_3 = (-1,-1,-1)$. Color the lattice according to $\phi(s_{x,y}) = v_{x+y \pmod 4}$. Any tile t in T_4, wherever it is placed in the lattice, has color sum $\phi(t) = v_0 + v_1 + v_2 + v_3 = e$. Since $a = 4k$, where $k > 0$, $[a \times b]$ is tileable by kb copies of the vertical tile 111, and hence $\phi([a \times b]) = e$, wherever $[a \times b]$ is placed in the lattice. Now suppose a $[2 \times 2]$ has been removed and the resulting region R has been placed in the lattice so that the bottom left corner of the missing $[2 \times 2]$ is at that origin. Then $\phi(R) = \phi([a \times b]) - \phi([2 \times 2]) = e - (v_0 + 2v_1 + v_2) = -(1,2,1) \neq e$. Thus R is not tileable by T_4, by Theorem 6.

Exercise 3. Use a coloring argument to show that the set of 8 L-tetrominoes found by rotations and reflections of the ribbon tile 001 in Figure 18 does not tile an $[8 \times 8]$ with a $[2 \times 2]$ removed.

We return to ribbon tiles once more. Though Theorem 3 resolves tiling questions for any simply connected region and the full set T_n, it remains interesting to investigate tileability questions with subsets of T_n. Define the *height of a ribbon tile* having binary signature $\epsilon = \epsilon_1 \epsilon_2 \cdots \epsilon_{n-1}$ to be the sum $\epsilon_1 + \epsilon_2 + \cdots + \epsilon_{n-1}$, taken modulo 2. Heights partition T_n into two subsets of equal size, the set T_n^0 of height-0 tiles and the set T_n^1 of height-1 tiles. Figure 18 shows the set T_4^1.

The following tile invariant, called the height invariant in [15], is obtained by considering the sum, taken modulo 2, of the $\lfloor n/2 \rfloor$ tile invariants in Theorem 2.

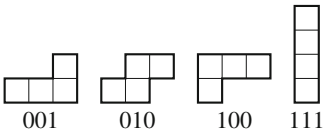


Fig. 18 The set T_4^1 of height-1 ribbon tile tetrominoes.

The Height Invariant If R is a simply connected region tileable by T_n, then the number of height-1 tiles used in any tiling of R is constant, modulo 2. We let $h(R)$ denote the value of this constant.

We can now prove the following useful test for nonexistence of a tiling by height-1 ribbon tiles.

Lemma 1 (Tileability Test for T_n^1). *Suppose there exists a tiling of a simply connected region R by T_n in which an odd number of height-0 tiles are used. Then the set T_n^1 of height-1 tiles does not tile R.*

Proof. Suppose T_n tiles R with $a + b$ tiles where a counts the number of height-1 tiles and b counts the number of height-0 tiles, and suppose $b \geq 1$ is odd. Then $h(R) \equiv a \cdot 1 + b \cdot 0 \equiv a \pmod 2$.

If T_n^1 tiles R then it does so with $a + b$ tiles, in which case we would also have $h(R) \equiv a + b \pmod 2$. But these two descriptions of $h(R)$ would imply $b \equiv 0 \pmod 2$, a contradiction since b is odd. Thus, no tiling of R by height-1 tiles exists. □

Example 11. Finishing off the Staircase Problem.

For ribbon tile trominoes, $T_3^1 = \{t_2, t_3\}$ (the L-trominoes in the staircase problem), and $T_3^0 = \{t_1, t_4\}$ (the straight tiles). Figure 3 shows tilings of S_3, S_5, S_6 and S_8, and each one uses an odd number of height-0 tiles. Lemma 1 ensures that the set $\{t_2, t_3\}$ does not tile these regions. Moreover, if the set T_3 can tile S_n with an odd number of height-0 tiles, then it can tile S_{n+12} with an odd number of height-0 tiles, since S_{n+12} can be partitioned as in Figure 2, and the regions $[n \times 12]$ and T_{12} can be tiled without any height-0 tiles. So, S_n is not tileable by $\{t_2, t_3\}$ if $n \equiv 3, 5, 6,$ or 8 (mod 12), and we have completed the proof that S_n is tileable by $\{t_2, t_3\}$ if and only if $n \equiv 0, 2, 9,$ or 11 (mod 12).

Example 12. In [15], Pak used this approach to answer several tiling questions involving rectangles and the sets T_n^1, for various n. In [8] we consider which modified rectangles $M(a, b)$ (obtained from $[a \times b]$ by removing its upper-left and lower-right cells) can be tiled by the set T_4^1 in Figure 18. We show that for $a, b > 1$, $M(a, b)$ is tileable by T_4^1 if and only if $a \equiv 2 \pmod 4$ and b is odd, or a is odd and $ab \equiv 2 \pmod 8$. With Lemma 1 in hand, the format of the proof follows much as the one used in the staircase problem. We find tilings of small, base case $M(a, b)$s, and then show inductively that tilings exist in all the cases listed above. We then rule out tilings of other $M(a, b)$ of the right area by demonstrating inductively that there

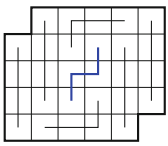


Fig. 19 $M(5,6)$ cannot be tiled by height-1 ribbon tile tetrominoes alone.

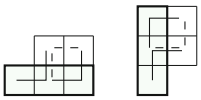


Fig. 20 Signed tiling of $[1 \times 3]$ and $[3 \times 1]$ by the subset $\{t_2, t_3\}$ of T_3. The solid tiles are $+1$ weighted and the dashed tiles are -1 weighted.

exist tilings of them in which an odd number of height-0 tiles are used. For instance, by Lemma 1, the height-1 ribbon tile tetrominoes do not tile $M(5,6)$ because T_4 tiles it with an odd number of height-0 tiles, as in Figure 19.

The boundary word arguments used to find the tile invariants for T_n are significantly more involved than coloring arguments, and it is reasonable to ask whether coloring arguments might unearth them. In fact, they cannot. Conway and Lagarias prove that no coloring argument could have been used to solve the staircase problem, and the same holds for the results in Example 12. To establish this fact they turn to signed tilings.

A *signed tiling* of a region R by a tile set T consists of a placement of tiles from T, each given a weight of $+1$ or -1, in such a way that the sum of the weights of the tiles is +1 for every cell inside the region R, and 0 for every cell outside the region. Figure 20 shows signed tilings of $[1 \times 3]$ and $[3 \times 1]$ by the subset $\{t_2, t_3\}$ of T_3. It follows that any tiling of a region by T_3 can be converted to a signed tiling by $\{t_2, t_3\}$. For instance, since T_3 tiles S_8, we know there is a signed tiling of S_8 by $\{t_2, t_3\}$. Thanks to the following result, which appears in [5, Theorem 5.2] and [15, Theorem 8.1], we know that if a tile set T cannot tile a region R *and* there exists a signed tiling of R by T, then no coloring argument could have been used to prove nonexistence of a tiling.

Theorem 7. *Any T-coloring that proves nonexistence of a tiling of a region also proves nonexistence of any signed tiling of the region.*

Proof. We prove the contrapositive. Suppose $\phi : \mathscr{R} \to G$ is a T-coloring, and $\{(t_i', \delta_i) \mid i = 1, \cdots, m\}$ represents a signed tiling of R by T, where each t_i' is a translation of a tile in T and each $\delta_i \in \{-1, 1\}$ gives the weight of the tile t_i'. Then $\phi(R) = \phi(s_1) + \cdots + \phi(s_{|R|})$ where the sum (in G) is over all the cells in R. But this sum equals $\sum_{i=1}^{n} \delta_i \phi(t_i')$ because the net weight from the signed tiling of each cell in R is $+1$ and the net weight of each cell outside R is 0. But as a T-coloring, $\phi(t_i') = e$ and it follows that $\phi(R) = e$. Thus, the T-coloring cannot be used to prove nonexistence of a tiling of R. □

It is not hard to extend the scene of Figure 20 to the general case $n \geq 3$. Namely, each tile in T_n has a signed tiling by the subset T_n^1 of height-1 tiles. Applying Theorem 7 and Lemma 1 to this setting we have the following fact: if a simply connected region R can be tiled by T_n in such a way that an odd number of tiles are height-0 tiles, then the set T_n^1 does not tile R, and no coloring argument could have been used to prove this.

Research Project 1. The height-1 ribbon tile pentominoes, T_5^1, have binary signatures $\{0001, 0010, 0100, 1000, 0111, 1011, 1101, 1110\}$. Pak proved that T_5^1 tiles $[a \times b]$ (where $a, b > 1$) if and only if $ab = 0 \pmod{10}$, [15, Theorem 7.2]. Which modified rectangles $M(a, b)$ (see Example 12) can be tiled by T_5^1?

Research Project 2. We have focused on the height invariant and its usefulness for tiling problems involving the set T_n^1, but other subsets of T_n have been considered as well. For instance, [15, Theorems 0.2, 0.3] uses the $i = 1$ invariant for T_5 from Theorem 2 to determine which rectangles and which staircase regions are tileable by the subset $\{0001, 0101, 0111, 1000, 1010, 1110\}$ of T_5. Explore other subsets of T_n that may lend themselves to tiling questions requiring the use of some of the ribbon tile invariants of Theorem 2.

Challenge Problem 2. Let $\Gamma_m(a, b)$ be the "gamma"-shaped region obtained by removing from $[a \times b]$ the square $[m \times m]$ from its lower right corner. Use Lemma 1 to prove that for odd $m \geq 1$ and $a, b > m$, the set T_4^1 tiles $\Gamma_m(a, b)$ if and only if $a \equiv m \pmod 4$ and $b \equiv m \pmod 8$; or $a \equiv 3m \pmod 4$ and $b \equiv 3m \pmod 8$. (This unpublished result was established by the author and Ben Coate as part of a student-faculty research project in 2008.)

Research Project 3. Consider again the regions $\Gamma_m(a, b)$ and the tile set T_4^1. What can be said in the case m is even? A coloring argument may apply.

5 Enumeration

Counting the number of distinct tilings of a region R by T is a question in enumerative combinatorics. A recent survey article by Propp [17] discusses general techniques that can be brought to bear on the matter. Rather than talk generalities, we present a few examples to give the reader a feel for enumeration questions.

Example 13 (Domino Tilings). The rectangle $[2 \times n]$ has F_{n+1} tiling by dominoes, where F_n is the n^{th} term in the Fibonacci sequence $1, 1, 2, 3, 5, 8, \ldots$. This result has a nice proof by induction. A more complicated formula, which we mention in passing, enumerates the domino tilings of the rectangle $[m \times n]$:

$$\prod_{j=1}^{\lceil \frac{m}{2} \rceil} \prod_{k=1}^{\lceil \frac{n}{2} \rceil} (4\cos^2 \frac{j\pi}{m+1} + 4\cos^2 \frac{k\pi}{n+1}).$$

Yes, this expression is always an integer (and gives 0 if m and n are odd), and this surprising result was established in 1961, independently in [6] and [9].

Exercise 4. Show that (a) $[2 \times n]$ has F_{n+1} tilings by dominoes, and (b) $[2 \times 3k]$ has F_{2k+1} tilings by T_3.

Example 14. In [21] Ueno provides a recursive process for enumerating tilings of rectangles by the set of four L-trominoes. The process involves matrices whose entries correspond to the number of tilings of thin strips in the lattice. For instance, Ueno shows there are 162 ways to tile the square $[6 \times 6]$. One can check that just two of these tilings avoid tiling any subrectangle within $[6 \times 6]$. One of these tilings appears in Figure 21, the other is its mirror image. The number of tilings grows rapidly with the area of the rectangle. The square $[9 \times 9]$ has 1,193,600 tilings. Which of these tilings, if any, avoid tiling a subrectangle? Perhaps none.

Here's a question that appeared in [2]. Suppose we remove two cells at random from a $[2 \times n]$. What is the probability that the remaining region (if its area is divisible by 3) can be tiled by the 4 L-trominoes in Figure 6? For instance, if $n = 4$ we have $\binom{8}{2}$ possible pairs of points to remove. Perhaps 10 of these pairs leave the remaining region untileable, in which case the probability of being able to tile a randomly generated board of this size is $9/14$. Can we say anything about the probability as

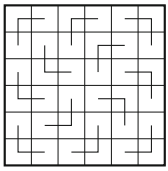


Fig. 21 A tiling of $[6 \times 6]$ by L-trominoes.

5	0	1	2	3	4
4	5	0	1	2	3
3	4	5	0	1	2
2	3	4	5	0	1
1	2	3	4	5	0
0	1	2	3	4	5

Fig. 22 The colors of the cells of $[6 \times 6]$. Any ribbon tile from T_6 covers one cell of each color.

$n \to \infty$? Is this question tractable? Work done in [1] may help. This question may not be a good one to pursue ultimately, but perhaps investigating it will lead to one that is.

Research Project 4. The set of 4 L-trominoes tiles the rectangle $[2\times 3k]$ in 2^k ways. Is there a closed-form expression for the number of tilings of $[4 \times 3k]$? How about other families of rectangles? The question of enumerating tilings of small height rectangles with Ls and $[1 \times 1]$ squares was considered in [4].

Example 15. The ribbon tile set T_n tiles the square $[n \times n]$ in $n!$ ways.

This fact was stated in [17], where Propp cites it as an unpublished result of Cristopher Moore's. We provide some notation to help us work through a proof of this nice result. First, place $[n \times n]$ in the lattice so that its lower left corner is at the origin, and color cell $s_{x,y}$ with the value $x + y$, taken modulo n (see Figure 22 for $n = 6$). The *initial cell* of any ribbon tile is its lower left most cell, and the *level* of a ribbon tile placed in $[n \times n]$ is the color of its initial cell. Note that any ribbon tile in T_n will cover one cell of each color, wherever it is placed. Furthermore, the initial cell of any tile in a tiling of $[n \times n]$ must be on or below the main diagonal at level $n - 1$.

One can prove that no two tiles in any tiling of $[n \times n]$ by T_n can have the same level, which implies there is exactly one tile of level i, for each $i = 0, 1, \ldots, n - 1$.

To enumerate the tilings we show there is a one-to-one correspondence between the number of tilings of $[n \times n]$ and the number of permutations of the set $\{0, 1, \ldots, n - 1\}$.

To establish the correspondence we focus on the cells on the main diagonal (the cells $s_{x,y}$ such that $x + y = n - 1$). Suppose we begin with a tiling of $[n \times n]$. Record,

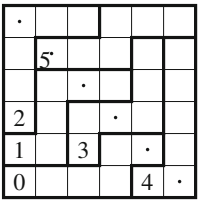


Fig. 23 The tiling of $[6 \times 6]$ by T_6 corresponding to permutation (2,5,1,3,0,4).

in sequence, the level of the tile covering the cells on the main diagonal, proceeding from upper left to lower right. (In Figure 23 the main diagonal cells are dotted.) The result will be a permutation of $\{0, 1, \ldots, n-1\}$, because each cell on this diagonal is covered by a different tile, and in any tiling we have one tile of each level from 0 to $n-1$.

Conversely, any permutation uniquely determines a tiling. The first number in the permutation gives the level of the tile that covers the upper left cell of the main diagonal, the cell $s_{0,n-1}$. Once this level has been chosen, there is a unique tile in T_n that can be placed at that level *and* cover $s_{0,n-1}$. Once that tile is placed, move to the second number in the permutation, which gives the level of the tile that covers $s_{1,n-2}$. Again, there is a unique tile in T_n that can work. In general, the i^{th} number in the permutation gives the level of the tile used to cover cell $s_{i-1,n-i}$, and this level uniquely determines the tile that can work. For instance, the permutation (2,5,1,3,0,4) uniquely determines the tiling of $[6 \times 6]$ in Figure 23. We encourage the reader to work through the details.

6 Concluding Remarks

In this introduction to the mathematics of tiling we have placed an emphasis on tile invariants and their role in tileability questions. Finding tile invariants for a given tile set, especially those that provide us with nonexistence results that would otherwise have been difficult to find, continues to be an interesting research opportunity. Many of the papers referenced here make some customized use of coloring or boundary word arguments to find tile invariants for tackling a tiling question; and these papers tend to generate as many questions as they solve. We encourage students to visit these papers, carefully work through the arguments presented, then begin tinkering with different questions. A fine place to begin would be in Korn's thesis [10], which offers an engaging and accessible treatment of tiling techniques to solve many types of tiling questions.

We have focused exclusively on regions living in a square lattice of the Euclidean plane because our experiences in student-centered research lie here, but we mention that the mathematics of tiling extends to other settings as well. For instance, in [5] and [20] the authors apply boundary word techniques to regions living in a triangular lattice, and also a hexagonal lattice. Much work on tilings has also been done in the

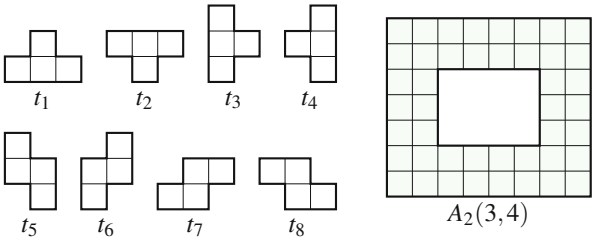


Fig. 24 The annular region $A_2(3,4)$ can be tiled by the set of skew and T-tetrominoes in 64 ways.

hyperbolic plane, on the surface of a sphere, and in higher dimensions. We leave it to the interested reader to explore work done in these other settings.

Example 16. We close with a set of tiling questions initially investigated in an REU in 2013. Let T consist of the skew and T-tetrominoes in Figure 24. The annular region $A_n(a,b)$ consists of a rectangular annulus of width n surrounding a rectangle $[a \times b]$. Figure 24 shows $A_2(3,4)$. The following results are established in [3].

- Every $A_2(a,b)$ is tileable by T, and $A_2(a,b)$ can be tiled in 2^{a+b-1} ways.
- The set $\mathscr{A}_2$ of all width-2 annuli has no local move property with respect to T.
- The tile counting group $G(T,\mathscr{A}_2) \simeq \mathbb{Z}^3 \times \mathbb{Z}_2$ is generated by these tile invariants:
  $\sum a_i = \frac{|R|}{4}$ (area invariant)
  $a_2 - a_1 = c_1(R)$
  $a_4 - a_3 = c_2(R)$
  $a_1 + a_2 + a_7 + a_8 \equiv c_3(R) \pmod 2$

Several open questions in this setting remain. Do analogous results hold for other annuli? Tilings of $A_3(1,3)$ and $A_3(2,2)$ appearing in [3] show that except for the area invariant, the tile invariants above do not apply for width-3 annuli. Furthermore, the region $A_3(4,4)$ does not appear to be tileable, though a non brute-force proof has not been found. Since there is a signed tiling of $A_3(4,4)$ by skews and Ts, no coloring argument can prove nonexistence. Finally, we remark that this tile set does not tile $[6 \times 6]$ or $[6 \times 10]$, as demonstrated in [11], though a brute-force approach was used there. Can a more elegant solution be found?

Challenge Problem 3. Use a coloring argument to show that if $a \equiv 3 \pmod 4$ and $b \equiv 2 \pmod 4$ then the subset $\{t_3, t_4, t_7, t_8\}$ of skews and Ts in Figure 24 does not tile the modified rectangle $M(a,b)$. This result appeared in [11].

Research Project 5. Determine the tile counting group for the set of 4 skews and 4 T-tetrominoes and the family of all width-3 annuli. Can the number of tilings of $A_3(a, b)$ be enumerated?

Research Project 6. If a cell (or two) is randomly removed from an $[a \times b]$, what is the probability that the remaining region is tileable by a given tile set?

References

1. Aanjaneya, M.: Tromino tilings of domino-deficient rectangles. Discret. Math. **309**(4), 937–944 (2009)
2. Ash, J.M., Golomb S.W.: Tiling deficient rectangles with trominoes. Math. Mag. **77**(1), 46–55 (2004)
3. Bright, A., Clark, J.G., Dunn C., Evitts, K., Hitchman, M.P., Keating, B., Whetter, B.: Tiling annular regions with skew and T-tetrominoes. Involve J. Math. **10**(3), 505–521 (2017)
4. Chinn, P., Grimaldi, R., Heubach, S.: Tiling with Ls and squares. J. Integer Seq. **10**, Article 07.2.8 (2007)
5. Conway, J.H., Lagarias, J.C.: Tiling with polyominoes and combinatorial group theory. J. Combin. Theory Ser. A **53**, 183–208 (1990)
6. Fisher, M., Temperley, H.: Dimer problem in statistcal mechanics - an exact result. Philos. Mag. **6**, 1061–1063 (1961)
7. Golomb, S.W.: Checker boards and polyominoes. Am. Math. Mon. **61**, 675–682 (1954)
8. Hitchman, M.P.: The topology of tile invariants. Rocky Mt. J. Math. **45**(2), 539–564 (2015)
9. Kasteleyn, P.: The statistics of dimers on a lattice I. The number of dimer arrangements on a quadratic lattice Physica **27**, 1209–1225 (1961)
10. Korn, M.: Geometric and algebraic properties of polyomino tilings. Ph.D. thesis, MIT (2004). http://hdl.handle.net/1721.1/16628
11. Lester, C.: Tiling with T and skew tetrominoes. Querqus Linfield J. Undergrad. Res. **1**(1), Article 3 (2012)
12. Magnus, W., Karrass, A., Solitar, D.: Combinatorial Group Theory, 2nd edn. Dover, New York (1976)
13. Moore C., Pak, I.: Ribbon tile invariants from signed area. J. Combin. Theory Ser. A **98**, 1–16 (2002)
14. Muchnik R., Pak, I.: On tilings by ribbon tetrominoes. J. Combin. Theory Ser. A **88**, 188–193 (1999)
15. Pak, I.: Ribbon tile invariants. Trans. Am. Math. Soc. **352**(12), 5525–5561 (2000)
16. Propp, J.: A pedestrian approach to a method of Conway, or, a tale of two cities. Math. Mag. **70**(5), 327–340 (1997)
17. Propp, J.: Enumeration of tilings. In: Bona M. (ed.) Handbook of Enumerative Combinatorics. CRC Press, Boca Raton (2015)

18. Reid, M.: Tile homotopy groups. Enseign. Math. **49**(1/2), 123–155 (2003)
19. Sheffield, S.: Ribbon tilings and multidimensional height functions. Trans. Am. Math. Soc. **354**(12), 4789–4813 (2002)
20. Thurston, W.P.: Conway's tiling groups. Am. Math. Mon. **95**, 757–773 (1990). special geometry issue
21. Ueno, C.: Matrices and tilings with right trominoes. Math. Mag. **81**(5), 319–331 (2008)

Forbidden Minors: Finding the Finite Few

Thomas W. Mattman

Suggested Prerequisites. *There are no serious prerequisites for this material. A course on graph theory would be helpful, but there are many books for self-study, including Marcus [18], that would quickly bring a student up to speed.*

1 Introduction

Kuratowski's Theorem [16], a highlight of undergraduate graph theory, classifies a graph as planar in terms of two forbidden subgraphs, K_5 and $K_{3,3}$. (We defer a precise statement to the next paragraph.) We will write Forb($\mathscr{P}l$) = $\{K_5, K_{3,3}\}$ where $\mathscr{P}l$ denotes the planarity property. We can think of the Graph Minor Theorem of Robertson and Seymour [23] as a powerful generalization of Kuratowski's Theorem. In particular, their theorem, which has been called the "deepest" and "most important" result in all of graph theory [15], implies that each graph property $\mathscr{P}$, *whatsoever*, generates a corresponding finite list of graphs. This scaffolding allows students to devise their own Kuratowski type theorems. As an example, we will determine the seven forbidden minors for a property that we call strongly almost-planar.

To proceed, we must define graph minor, which is a generalization of subgraph. We will assume familiarity with the basic terminology of graph theory; West's [27] book is a good reference at the undergraduate level. While Diestel [8] is at a higher level, it includes an accessible approach to graph minor theory. For us, graphs are simple (no loops or double edges) and not directed. We can define the notion of a minor using graph operations. We obtain subgraphs through the operations of edge and vertex deletion. For minors, we allow an additional operation: *edge contraction*.

T.W. Mattman (✉)
Department of Mathematics and Statistics, CSU, Chico, Chico, CA 95929-0525, USA
e-mail: TMattman@CSUChico.edu

A. Wootton et al. (eds.), *A Primer for Undergraduate Research*, Foundations for Undergraduate Research in Mathematics, DOI 10.1007/978-3-319-66065-3_4

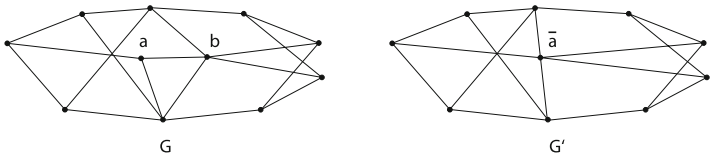


Fig. 1 Edge contraction.

As in Figure 1, when we contract edge ab in G, we replace the pair of vertices with a single vertex $\bar{a}$ that is adjacent to each neighbor of a or b. The resulting graph G' has one less vertex and at least one fewer edge than G. (If a and b share neighbors, even more edges are lost.) A *minor* of graph G is any graph obtained by contracting (zero or more) edges in a subgraph of G. Recall that K_5 is the complete graph on five vertices and $K_{3,3}$ the complete bipartite graph with two parts of three vertices each. We can now state Kuratowski's Theorem, using the formulation in terms of minors due to Wagner.

Theorem 1 (Kuratowski-Wagner [16, 26]). *A graph G is planar if and only if it has no K_5 nor $K_{3,3}$ minor.*

Robertson and Seymour's theorem can be stated as follows.

Theorem 2 (The Graph Minor Theorem [23]). *In any infinite sequence of graphs $G_1, G_2, G_3, \ldots$ there are indices $i \neq j$ such that G_i is a minor of G_j.*

This yields two important corollaries, which we now describe.

Planarity is an example of a property that is *minor closed*: If H is a minor of a planar graph G, then H must also be planar. If $\mathscr{P}$ is minor closed, the Graph Minor Theorem implies a finite set of forbidden minors.

Corollary 1. *Let $\mathscr{P}$ be a graph property that is minor closed. Then there is a finite set of forbidden minors Forb($\mathscr{P}$) such that G has $\mathscr{P}$ if and only if it has no minor in Forb($\mathscr{P}$).*

In honor of the theorem for planar graphs, we call Forb($\mathscr{P}$) the *Kuratowski set* for $\mathscr{P}$.

Even if $\mathscr{P}$ is not minor closed, the Graph Minor Theorem determines a finite set. For this, note that K_5 and $K_{3,3}$ are *minor minimal nonplanar*; each is nonplanar with every proper minor planar. More generally, for graph property $\mathscr{P}$, a graph G is *minor minimal for* $\mathscr{P}$ or MM$\mathscr{P}$ if G has $\mathscr{P}$, but no proper minor does.

Corollary 2. *Let $\mathscr{P}$ be a graph property. The set of MM$\mathscr{P}$ graphs is finite.*

Every $\mathscr{P}$ graph has a MM$\mathscr{P}$ minor. However, if the negation, $\neg\mathscr{P}$, is not minor closed, there will be graphs that are not $\mathscr{P}$ even though they have a MM$\mathscr{P}$ minor. In addition to providing a necessary condition for $\mathscr{P}$, the finite MM$\mathscr{P}$ list, therefore, constitutes a certificate or sufficient condition for $\neg\mathscr{P}$; a graph with no MM$\mathscr{P}$ minor definitely does not have property $\mathscr{P}$. For these reasons, when $\mathscr{P}$ is not minor closed, determining the list of MM$\mathscr{P}$ graphs is worthwhile, even if it does not completely characterize $\mathscr{P}$ the way that a Kuratowski set would.

In the next section we summarize the graph properties $\mathscr{P}$ with known MM$\mathscr{P}$ or Kuratowski set. In Section 3 we illustrate how students might develop their own Kuratowski type theorem through an explicit example: we determine Forb($\mathscr{P}$) for a property that we call strongly almost-planar. In the final section we propose several concrete research projects and provide some suggestions about how to choose graph properties $\mathscr{P}$.

Throughout the paper, we present a list of 'challenge problems' and 'research projects.' Challenges are warm up exercises for talented undergraduates. In most cases, the solution is known and can be found through a web search or in the references at the end of the paper. Research projects, on the other hand, are generally open problems (as far as we know). Some are quite difficult, but, we hope, all admit openings. Indeed, we see this as a major theme in this area of research. Even if we know MM$\mathscr{P}$ and Kuratowski sets are finite, a complete enumeration is often elusive. However, it is generally not too hard to capture graphs that belong to the set. These problems, then, promise a steady diet of small successes along the way in our hunt to catch all of the finite few.

2 Properties with Known Kuratowski Set

In this section we summarize the graph properties with known MM$\mathscr{P}$ or Kuratowski set. First, an important caveat. While the Graph Minor Theorem ensures these sets are finite, the proof is not at all constructive and gives no guidance about their size. It makes for nice alliteration to talk of the 'finite few,' but some finite numbers are really rather large. A particularly dramatic cautionary tale is Y∇Y reducibility (we omit the definition) for which Yu [28] has found more than 68 billion forbidden minors.

On the other hand, bounding the *order* (number of vertices) or *size* (number of edges) of a graph is a minor closed condition. For example, whatever property you may be interested in, appending the condition "of seven or fewer vertices," ensures that the set of MM$\mathscr{P}$ graphs is no larger than 1044, the number of order seven graphs. In general, it is quite difficult to predict the size of a Kuratowski set in advance and researchers in this area often do resort to restricting properties by simple graph parameters such as order, size, or connectivity.

We will focus on results that generalize planarity in various ways. However, we briefly mention graphs of bounded tree-width as another important class of examples. Let $\mathscr{T}_k$ denote the graphs of tree-width at most k. For the sake of brevity we omit the definition of tree-width (which can be found in [8], for example) except for noting that $\mathscr{T}_1$ is the set of forests, i.e., graphs whose components are trees. For small k, the obstructions are quite simple: Forb($\mathscr{T}_1$) $= \{K_3\}$, Forb($\mathscr{T}_2$) $= \{K_4\}$, and Forb($\mathscr{T}_3$) has four elements, including K_5 [5, 25]. However, for $k \geq 4$ the Kuratowski set for $\mathscr{T}_k$ is unknown.

Research Project 1. Find graphs in Forb($\mathscr{T}_k$) for $k \geq 4$. Is K_{k+2} always forbidden? Is there always a planar graph in Forb($\mathscr{T}_k$)?

We can think of a planar graph as a 'spherical' graph since it can be embedded on a sphere with no edges crossing. More generally, the set of graphs that embed on a particular compact surface (orientable or not) is also minor closed. However, to date, (in addition to the sphere) only the Kuratowski set for embeddings on a projective plane is known; there are 35 forbidden minors [3, 12, 22]. The next step would be *toroidal graphs*, those that embed on a torus; Gagarin, Myrvold, and Chambers remark that there are at least 16 thousand forbidden minors [11]. In the same paper they show only four of them have no $K_{3,3}$ minor. This is a good example of how a rather large Forb($\mathscr{P}$) can be tamed by adding conditions to the graph property $\mathscr{P}$. While observing that it's straightforward to determine the toroidal obstructions of lower connectivity, Mohar and Škoda [21] find there are 68 forbidden minors of connectivity two. Explicitly listing the forbidden minors of lower connectivity would be a nice challenge for a strong undergraduate.

Challenge Problem 1. Determine the forbidden minors of connectivity less than two for embedding in the torus. Find those for a surface of genus two.

The Kuratowski sets for more complicated surfaces are likely even larger than the several thousand known for the torus.

Outerplanarity is a different way to force smaller Kuratowski sets. A graph is *outerplanar* (or has property $\mathscr{OP}l$) if it can be embedded in the plane with all vertices on a single face. The set of forbidden minors for this property, Forb($\mathscr{OP}l$) is well known and perhaps best attributed to folklore (although, see [7]).

Challenge Problem 2. Determine Forb($\mathscr{OP}l$). (Use Kuratowski's theorem!)

Similarly, one can define outerprojective planar or outertoroidal graphs as graphs that admit embeddings into those surfaces with all vertices on a single face. There are 32 forbidden minors for the outerprojective planar property [4].

Challenge Problem 3. Find forbidden minors for the outerprojective planar property.

Research Project 2. Find forbidden minors for the outertoroidal property.

Apex vertices also lead to minor closed properties. Let $v \in V(G)$ be a vertex of graph G. We will use $G - v$ to denote the subgraph obtained from G by deleting v (and all its edges). Given property $\mathscr{P}$, we say that G has $\mathscr{P}'$ (or is apex-$\mathscr{P}$) if there is a vertex v (called an *apex*) such that $G - v$ has $\mathscr{P}$. If $\mathscr{P}$ is minor closed, then $\mathscr{P}'$ is as well. For example, Ding and Dziobiak determined the 57 graphs in Forb($\mathscr{O}\mathscr{P}l'$) [9]. In the same paper, they report that there are at least 396 graphs in the Kuratowski set for apex-planar.

Research Project 3. Find graphs in Forb($\mathscr{P}'$) when $\mathscr{P}$ is toroidal with no $K_{3,3}$, $\mathscr{T}_1$, $\mathscr{T}_2$, $\mathscr{T}_3$, or for some other property with small Forb($\mathscr{P}$).

The set of linklessly embeddable graphs are closely related to those that are apex-planar. We say a graph is *linklessly embeddable* (or has property $\mathscr{L}$) if there is an embedding in $\mathbb{R}^3$ that contains no pair of nontrivially linked cycles. (See [1] for a gentle introduction to this idea). An early triumph of graph minor theory was the proof that Forb($\mathscr{L}$) has exactly seven graphs [24]. An apex-planar graph is also $\mathscr{L}$ and, as part of an undergraduate research project, we showed that Forb($\mathscr{L}$) $\subset$ Forb($\mathscr{P}l'$) [6]. The related idea of knotlessly embeddable (which, like $\mathscr{L}$, is minor closed) has more than 240 forbidden minors [13].

As a final variation on properties related to planarity, rather than vertex deletion (which gives apex properties), let's think about the other two operations for graph minors, edge deletion and contraction. For graph G and edge $ab \in E(G)$, let $G - ab$ denote the subgraph resulting from deletion and G/ab the minor obtained by edge contraction. Unlike apex properties, in general edge operations do not preserve closure under taking minors. This is why we frame some results below in terms of MM$\mathscr{P}$ sets.

As we've mentioned, there are at least several hundred graphs in Forb($\mathscr{P}l'$). In an undergraduate research project [17] we found that there are also large numbers of graphs that are not simply an edge away from planar. Call a graph G ND (not edge deletion apex) if there is no edge ab with $G - ab$ planar and similarly NC (not contraction apex) if no G/ab is planar. We showed that the are at least 55 MMND and 82 MMNC graphs. On the other hand, if we switch from the existential to the universal quantifier, we obtain properties that are minor closed with reasonably small Kuratowski sets; in the next challenge, each Forb($\mathscr{P}$) has at most ten elements. Say that a graph G is CA (completely apex) if $G - v$ is planar for every vertex v, CD (completely edge deletion apex) if every $G - ab$ is planar, and CC (completely contraction apex) if every G/ab is planar.

Challenge Problem 4. For $\mathscr{P} = CA$, show that $\mathscr{P}$ is minor closed and determine Forb($\mathscr{P}$). Repeat for $\mathscr{P} = CD$ and CC.

Instead of flipping quantifiers, we can think about combing operations with other logical connectives. For example, Gubser [14] calls G *almost-planar* if, for every $ab \in E(G)$, $G - ab$ or G/ab is planar. There are six forbidden minors for this property [10]. In the next section we determine the Kuratowski set for a property that we call *strongly almost-planar* or SAP: for every $ab \in E(G)$, both $G - ab$ and G/ab are planar. Note that every strongly almost-planar graph is almost-planar.

3 Strongly Almost–Planar Graphs

In this section we model how a research project in this area might play out through an explicit example, the strongly almost-planar or SAP property: G is SAP if, $\forall ab \in E(G)$, both $G - ab$ and G/ab are planar.

Our first task is to determine whether or not this property (or its negation) is minor closed. If not, there is no hope of getting a Kuratowski set. Instead, we would target the list of MMSAP graphs, as that would provide a necessary condition for SAP and a sufficient condition for SAP to fail. However, as we will now show, SAP is minor closed, meaning our goal is instead Forb(SAP).

Lemma 1. *SAP is minor closed*

Proof. It is enough to observe that SAP is preserved by the three operations used in constructing minors, vertex or edge deletion and edge contraction.

Suppose G is SAP and $v \in V(G)$. Let $G' = G - v$. We must show that for each $ab \in E(G')$, both $G' - ab$ and G'/ab are planar. Since $V(G') \subset V(G)$, we can think of ab as an edge in $E(G)$. Then it's easy to identify $G' - ab$ as a subgraph of the planar graph $G - ab$, which shows $G' - ab$ is also planar. Similarly, we'll know that G'/ab is planar once we show that it is a subgraph of G/ab. There are a few cases to think about (Is a or b or both adjacent to v?) but it always turns out that $G'/ab = (G/ab) - v$.

For this property, the argument for edge contraction and deletion is quite simple. For any $ab \in E(G)$, by assumption $G - ab$ and G/ab are planar. Then any minor of these graphs is again planar, including those given by deleting or contracting an edge. □

Next, we must generate examples of forbidden minors, meaning graphs that are minor minimal for not SAP. We are looking for graphs G that are just barely not SAP: although G is not SAP, every proper minor is. Most likely, there's only a single edge ab with $G - ab$ or G/ab nonplanar. And that graph is probably minor minimal nonplanar, so one of the *Kuratowski graphs* K_5 or $K_{3,3}$. We will use K to represent a generic Kuratowski graph, that is $K \in \{K_5, K_{3,3}\}$. In summary, we are looking for graphs of the form K 'plus an edge,' where adding an edge includes the idea of reversing an edge contraction.

We encourage you to take a minute to see what graphs you can discover that have the form K 'plus an edge'. Hopefully, you will find five G for which $G - ab$ is nonplanar. Perhaps you have even more? Remember we want minor minimal examples, so check if any pair are minors one of the other.

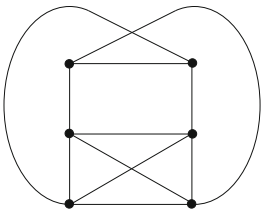


Fig. 2 The graph $K_{3,3} + 2e$.

Since edge contraction may be a new idea for the reader, let's delve a little further into examples where G/ab is nonplanar. The reverse operation of edge contraction is called a *vertex split* and defined as follows. Replace a vertex $\bar{a}$ with two vertices a and b connected by an edge. Each neighbor of $\bar{a}$ becomes a neighbor of at least one of a and b.

Suppose $G/ab = K_{3,3}$. There are essentially two ways to make a vertex split and recover G. One is to make one of the new vertices, say a, adjacent to no neighbor of $\bar{a}$ and the other, b, adjacent to all three. Then, in G, a has degree one (its only neighbor is b) and b will have degree four. The other option is to make a adjacent to one neighbor of $\bar{a}$ and let b have the other two. There are other possibilities since we may choose to make both a and b adjacent to one of $\bar{a}$'s neighbors; but such graphs will have one of the two we described earlier as a minor.

If $G/ab = K_5$, there are three ways to split the degree four vertex $\bar{a}$. Two are similar to the ones just described for $K_{3,3}$ where we make a adjacent to zero or to one neighbor of $\bar{a}$. The third option, split up the four neighbors of $\bar{a}$ by making a and b adjacent to two each, results in the graph $K_{3,3} + 2e$ shown in Figure 2. However, you should observe that this graph has a proper subgraph among those found by adding an edge to $G - ab = K_{3,3}$.

A little experimentation along these lines should lead to the seven graphs of Figure 3. Note that the six graphs in the top two rows occur in pairs where we perform similar operations on K_5 and $K_{3,3}$. We'll write $K \sqcup K_2$, $K \dot{\cup} K_2$, and $\bar{K}$ for the pairs at left, center, and right, respectively and $K_{3,3} + e$ for the seventh graph at the bottom of the figure. The five graphs with $G - ab$ nonplanar are $K_{3,3} + e$, and the pairs $K \sqcup K_2$ and $K \dot{\cup} K_2$. The graphs obtained from $G/ab = K$ where a is made adjacent to a single neighbor of $\bar{a}$ are the $\bar{K}$ pair. When a shares no neighbors with $\bar{a}$, we construct the graphs $K \dot{\cup} K_2$ for a second time. The graph $K_{3,3} + 2e$ (see Figure 2), obtained from K_5 by splitting a vertex so that a and b are each adjacent to two neighbors of $\bar{a}$, is not SAP. But, it has another non SAP graph, $K_{3,3} + e$, as a proper subgraph and cannot be minor minimal. The others are both non SAP and minor minimal, as we now verify.

Lemma 2. *The seven graphs of Figure 3 are minor minimal for not SAP.*

Proof. As we noticed, the two $\bar{K}$ graphs become Kuratowski graphs after an edge contraction and the rest have an edge deletion that leaves a nonplanar graph. This shows that none of the graphs are SAP.

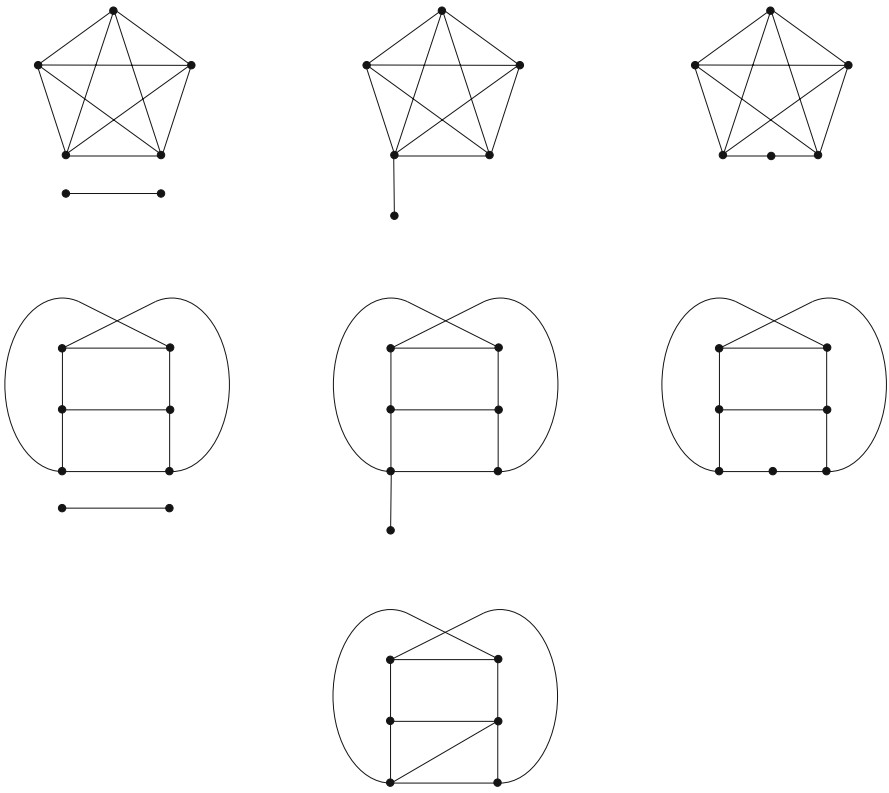


Fig. 3 Forbidden minors for SAP.

It remains to show that every proper minor of each graph is SAP. Since SAP is minor closed, it's enough to verify this for the three basic operations vertex or edge deletion and edge contraction. Actually, since none of our graphs has an isolated vertex, we need only check edge deletion and contraction. For once we have those in hand, then any graph of the form $G - a$ is automatically SAP as it's a subgraph of one of the $G - ab$ graphs formed by deleting an edge on a. (Recall that we just proved that SAP is closed under taking minors.)

Note that planar graphs are SAP, so we can reduce to the case where an edge deletion or contraction gives a nonplanar graph. Let G be a graph in the figure and suppose G' is a nonplanar minor obtained by an edge deletion or contraction. Observe that, up to isolated vertices, G' is simply a Kuratowski graph K. In particular, $E(G') = E(K)$. Since K is minor minimal nonplanar, any further edge deletion or contraction leaves a planar graph, which shows G' is SAP, as required. This completes the argument that the seven graphs in the figure are in Forb(SAP).

□

We will argue that there are no other graphs in Forb(SAP). We begin with graphs that are not connected.

Lemma 3. *If $G \in Forb(SAP)$ is not connected, then $G = K \sqcup K_2$ with $K \in \{K_5, K_{3,3}\}$.*

Proof. Let $G = G_1 \sqcup G_2$ in Forb(SAP) be the disjoint union of (nonempty) graphs G_1 and G_2. Since planar graphs are SAP, at least one of G_1 and G_2, say G_1, is not planar.

We first observe that G_2 must have an edge, $E(G_2) \neq \emptyset$. Otherwise, since G is not SAP, there is an edge $ab \in E(G)$ and a nonplanar minor G', formed by deleting or contracting ab. Since G_2 has no edges, it is planar and $ab \in E(G_1)$. If follows that deleting or contracting ab in G_1 already gives a nonplanar graph. That is, G_1 is a proper minor of G that is not SAP. This contradicts our assumption that G is minor minimal not SAP.

By Kuratowski's theorem, G_1 has a Kuratowski graph minor, K. We claim that $G_1 = K$. Further, since G_2 has an edge, we must have $G_2 = K_2$. For, if either of these fail, the graph $K \sqcup K_2$, which is not SAP by the previous lemma, is a proper minor of G. This contradicts our assumption that G is minor minimal for not SAP. □

We can now complete the argument.

Theorem 3. *The seven graphs of Figure 3 are precisely the elements of Forb(SAP).*

Proof. Using the previous two lemmas, it remains only to verify that if G is connected and in Forb(SAP), then it is one of the five connected graphs in the figure. Suppose G is connected and minor minimal not SAP. Since G is not SAP, there is an $ab \in E(G)$ such that G', a minor formed by deleting or contracting ab, is not planar. Then G' has a Kuratowski graph K as a minor. In fact K must appear as a subgraph of G'. If not, one of the two $\bar{K}$ graphs is a minor of G' and, hence also of G. This contradicts our assumption that G is minor minimal for not SAP.

Suppose that the nonplanar G' is formed by edge deletion: $G' = G - ab$. There are several cases depending on the size of $V(K) \cap \{a, b\}$. If there is no common vertex, then G has a $K \sqcup K_2$ minor. Since we assumed G is minor minimal for not SAP, $G = K \sqcup K_2$, but this contradicts our assumption that G is connected. Suppose there is one vertex in the intersection. Then G has a $K \dot{\cup} K_2$ minor. By minor minimality, $G = K \dot{\cup} K_2$ and appears in Figure 3 as required. Finally, if $\{a, b\} \subset V(K)$, then K must be $K_{3,3}$ and, by minor minimality, $G = K_{3,3} + e$ is one of the graphs in the figure.

If instead $G' = G/ab$, let $\bar{a}$ denote the vertex that results from identifying a and b. If $\bar{a} \in V(K)$, there are two possibilities. It may be that G has one of the $\bar{K}$ or $K \dot{\cup} K_2$ graphs of Figure 3 as a minor. But then, by minor minimality, G is one of those graphs in the figure, as required. The other possibility is G has the $K_{3,3} + 2e$ graph of Figure 2 as a minor. Then, $K_{3,3} + e$ is a proper minor, contradicting the minor minimality of G. On the other hand, if $\bar{a} \notin V(K)$, G must have a $K \sqcup K_2$ minor. By minor minimality, $G = K \sqcup K_2$, which contradicts our assumption that G is connected. □

While it's difficult to convey the hard work that went into finalizing the list of seven graphs, we hope this account gives some of the flavor of a project in this area. This argument is, in fact, not so different from what appears in (soon to be) published research, see [10, 17]. Recall that CA, CD, and CC all have Kuratowski sets with at most 10 members (see Challenge 4). We can think of almost-planar as CD or CC and SAP as CD and CC. This suggests that other combinations of the three C properties are also likely to be minor closed with a small number of forbidden minors. For example, here are two ways to combine CA and CD.

Research Project 4. Say graph G has property CACD if, for every edge ab and every vertex $v \notin \{a, b\}$, either $G - v$ or $G - ab$ is planar. Determine whether or not CACD is minor closed and find the Kuratowski set or MMCACD set. Repeat for strongly CACD, which requires both $G - v$ and $G - ab$ planar.

4 Additional Project Ideas

In this section we propose several additional research projects along with general strategies to develop even more.

Let $\mathscr{E}_k$ denote the graphs of size k or less. We have mentioned that this property is minor closed.

Challenge Problem 5. Determine the Kuratowski set for edge-free graphs, Forb$(\mathscr{E}_0)$ and that for the corresponding apex property, Forb$(\mathscr{E}_0')$. What is Forb$(\mathscr{E}_k)$ for $k > 0$?

However, $\mathscr{E}_1'$ is already interesting and general observations about higher k would be worth pursuing.

Research Project 5. Find graphs in Forb$(\mathscr{E}_1')$. Find forbidden minors for $\mathscr{E}_k'$ when $k \geq 2$. Can you formulate any conjectures about Forb$(\mathscr{E}_k')$?

In a different direction, if $\mathscr{P}$ is minor closed, then so too are all $\mathscr{P}^{(k)}$ where $\mathscr{P}^{(k+1)} = (\mathscr{P}^{(k)})'$.

Research Project 6. Find graphs in Forb($\mathscr{E}_0''$). We might call $\mathscr{E}_0''$ graphs 2-apex edge free. Any conjectures about k-apex edge free?

How about working with order instead of size?

Research Project 7. Find forbidden minors for graphs of order at most k. What about apex versions of these Kuratowski sets? Any conjectures?

Naturally, one can combine these. What is the Kuratowski set for graphs that have at least two edges and three vertices? What of graphs that have either an edge or four vertices?

These project ideas encourage you to formulate your own conjectures. As examples of the kinds of conjectures that might arise, we refer to Research project 1. There we noticed that K_{k+2} is a forbidden minor in $\mathscr{T}_k$ for $k = 1, 2, 3$, which led us to ask if the pattern persists. That research project also includes a guess about planar graphs, again based on what is known for small k. Recently, we made similar observations about forbidden minors for $\mathscr{P}l^{(k)}$ which is also called k-apex [19]. While proving that $K_{k+5} \in$ Forb($\mathscr{P}l^{(k)}$) we were unable to confirm a stronger conjecture that all graphs in the K_{k+5} family are forbidden. (Please refer to [19] for the definition of a graph's family.) We have a similar conjecture for graphs in the family of $K_{3^2,1^k}$, a $k+2$-partite graph with two parts of three vertices each and the remainder having only one vertex.

Research Project 8. Prove the conjecture of [19]: The K_{k+5} and $K_{3^2,1^k}$ families are in Forb($k-$apex).

So far we have focused attention on graph properties that are minor closed and most of the discussion in Section 2 described techniques for generating such properties. The meta-problem of finding additional minor closed graph properties is also worthwhile.

Research Project 9. Find a minor closed graph property $\mathscr{P}$ different from those described to this point. Find graphs in Forb($\mathscr{P}$).

A survey by Archdeacon [2] includes a listing of several more problems on forbidden graphs; many of them would be great undergraduate research projects.

As in Corollary 1, minor closed $\mathscr{P}$ are attractive because Forb($\mathscr{P}$) then precisely characterizes graphs with the property. On the other hand, as discussed following Corollary 2, even if $\mathscr{P}$ is not minor closed, it is worth finding the finite list of MM$\mathscr{P}$ graphs which provide both a necessary condition for $\mathscr{P}$ and a sufficient condition for its negation. The possible projects in this direction are virtually endless. Take your favorite graph invariant (e.g., chromatic number, girth, diameter, minimum or maximum degree, degree sequence, etc.) and see how many MM$\mathscr{P}$ graphs you can find for specific values of the invariant. Of course, if you choose a graph property at random, you run the risk of stumbling onto a MM$\mathscr{P}$ list that, while finite, is rather large. In that case, you can simply restrict by graph order or size, for example.

If you're fortunate enough to be working with a student with some computer skills, you might let her loose on the graph properties that are built into many computer algebra systems. With computer resources, even the 300 thousand or so graphs of order nine or less are not out of the question, see for example [20].

Finally, let us note that the recent vintage of the Graph Minor Theorem and the rather specific interests of graph theorists leave a virtually untouched playing field open to those of us working with undergraduates. To date, serious researchers have focused on finding forbidden minors for a fairly narrow range of properties deemed important in the field. For those of us who needn't worry overly about the significance of the result, there is tremendous freedom to pursue pretty much any idea that comes to mind and see where it takes us. These are early days in this area and whichever path you choose to follow, there's an excellent chance of capturing a Kuratowski type theorem of your very own.

References

1. Adams, C.C.: The Knot Book: An Elementary Introduction to the Mathematical Theory of Knots. American Mathematical Society, Providence, RI (2004). Revised reprint of the 1994 original
2. Archdeacon, D.: Variations on a theme of Kuratowski. Discret. Math. **302** 22–31 (2005)
3. Archdeacon, D.: A Kuratowski theorem for the projective plane. J. Graph Theory **5**, 243–246 (1981)
4. Archdeacon, D., Hartsfield, N., Little, C.H.C., Mohar, B.: Obstruction sets for outer-projective-planar graphs. Ars Combin. **49**, 113–127 (1998)
5. Arnborg, S., Proskurowski, A., Corneil, D.G.: Forbidden minors characterization of partial 3-trees. Discret. Math. **810**, 1–19 (1990)
6. Barsotti, J., Mattman, T.W.: Graphs on 21 edges that are not 2-apex. Involve **9**, 591–621 (2016)

7. Chartrand, G., Harary, F.: Planar permutation graphs. Ann. Inst. H. Poincar. Sect. B (N.S.) **3**, 433–438 (1967)
8. Diestel, R.: Graph Theory, 4th edn. Graduate Texts in Mathematics, vol. 173. Springer, Heidelberg (2010)
9. Ding, G., Dziobiak, S.: Excluded-minor characterization of apex-outerplanar graphs. Graphs Combin. **32**, 583–627 (2016)
10. Ding, G., Fallon, J., Marshall, E.: On almost-planar graphs. 2016 (Preprint)
11. Gagarin, A., Myrvold, W., Chambers, J.: The obstructions for toroidal graphs with no K3,3's. Discret. Math. **309** 3625–3631 (2009)
12. Glover, H., Huneke, J.P., Wang, C.S.: 103 graphs that are irreducible for the projective plane. J. Combin. Theory Ser. B **27**, 332–370 (1979)
13. Goldberg, N., Mattman, T.W., Naimi, R.: Many, many more intrinsically knotted graphs. Algebr. Geom. Topol. **14**, 1801–1823 (2014)
14. Gubser, B.S.: A characterization of almost-planar graphs. Combin. Probab. Comput. **5**, 227–245 (1996)
15. Kawarabayashi, K., Mohar, B.: Some recent progress and applications in graph minor theory. Graphs Combin. **23**, 1–46 (2007)
16. Kuratowski, K.: Sur le problème des courbes gauches en topologie. Fundam. Math. **15**, 271–283 (1930)
17. Lipton, M., Mackall, E., Mattman, T.W., Pierce, M., Robinson, S., Thomas, J., Weinschelbaum, I.: Six variations on a theme: almost planar graphs. Involve (2017, to appear)
18. Marcus, D.A.: Graph Theory. A Problem Oriented Approach. MAA Textbooks. Mathematical Association of America, Washington, DC (2008)
19. Mattman, T.W., Pierce, M.: The K_{n+5} and $K_{3^2,1^n}$ families and obstructions to n-apex. In: Knots, Links, Spatial Graphs, and Algebraic Invariants. Contemporary Mathematics, vol. 689, pp. 137–158. American Mathematical Society, Providence (2017)
20. Mattman, T.W., Morris, C., Ryker, J.: Order nine MMIK graphs. In: Knots, Links, Spatial Graphs, and Algebraic Invariants. Contemporary Mathematics, vol. 689, pp. 103–124. American Mathematical Society, Providence (2017)
21. Mohar, B., Škoda, P.: Obstructions of connectivity two for embedding graphs into the torus. Canad. J. Math. **66**, 1327–1357 (2014)
22. Mohar, B., Thomassen, C.: Graphs on Surfaces. Johns Hopkins University Press, Baltimore, MD (2001)
23. Robertson, N., Seymour, P.: Graph minors. XX. Wagner's conjecture. J. Combin. Theory Ser. B **92**, 325–357 (2004)
24. Robertson, Seymour, P., Thomas, R.: Sachs' linkless embedding conjecture. J. Combin. Theory Ser. B **64**, 185–227 (1995)
25. Satyanarayana, A., Tung, L.: A characterization of partial 3-trees. Networks **20**, 299–322 (1990)
26. Wagner, K.: Über eine Eigenschaft der ebenen Komplexe. Math. Ann. **114**, 570–590 (1937)
27. West, D.B.: Introduction to Graph Theory. Prentice Hall, Inc., Upper Saddle River, NJ (1996)
28. Yu, Y.: More forbidden minors for wye-delta-wye reducibility. Electron. J. Combin. **13** Research Paper 7, 15 pp. (2006)

Introduction to Competitive Graph Coloring

C. Dunn, V. Larsen, and J.F. Nordstrom

Suggested Prerequisites. *Ideally a first course in Graph Theory or Discrete Mathematics. However, mathematical maturity and experience writing proofs will suffice.*

1 Introduction

The *map-coloring game* was first presented in Martin Gardner's "Mathematical Games" column in *Scientific American* in 1981 [24]. Invented by Steven J. Brams, the game involves two players, Alice and Bob, alternating coloring the countries on a map such that two countries that share a nontrivial border must receive different colors. The first player, Alice, wants to ensure that the map eventually gets colored with the finite set of colors with which the players begin. The second player, Bob, however, wants to ensure that there comes a time in the game when there is an uncolored country for which none of the existing colors can be used. The interesting question is then, for a given map what is the least number of colors necessary such that Alice has a winning strategy? Unfortunately, this game did not receive any attention from the graph theory community at the time. Ten years later, Bodlaender reintroduced the *r-coloring game* [4] within the broader context of graphs. In the original formulation of the game on graphs, we begin with a finite graph, G, and a set X of r colors. Two players, Alice and Bob, alternate coloring the uncolored vertices of G using colors from X. At each step of the game, the players must choose to color

C. Dunn (✉) • J.F. Nordstrom
Linfield College, McMinnville, OR 97128, USA
e-mail: cdunn@linfield.edu; jfirkins@linfield.edu

V. Larsen
Kennesaw State University, Marietta, GA 30060, USA
e-mail: vlarsen@kennesaw.edu

A. Wootton et al. (eds.), *A Primer for Undergraduate Research*, Foundations for Undergraduate Research in Mathematics, DOI 10.1007/978-3-319-66065-3_5

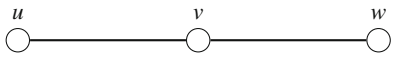


Fig. 1 Alice and Bob playing with two colors on G.

an uncolored vertex with a legal color. Alice goes first. In the basic formation of the game, a color $\alpha \in X$ is *legal* for an uncolored vertex u if u has no neighbors already colored α. Alice wins the game if all vertices of the graph are colored; otherwise, Bob wins. Although both players must use a legal color, Alice is trying to ensure that at every stage of the game all vertices have a legal color available, while Bob would like to force a situation in which an uncolored vertex exists for which there is no legal color.

For example, suppose Alice and Bob are playing the game on the graph G in Figure 1 with two colors which we will call α and β. If u is colored α, then only β is legal for v. However, if u is colored α and w is colored β, then v will have no legal color.

Notice, on this small example, we can see that if Alice first colors v, with α, then β is always legal for u and w. Thus Alice wins on G. However, if Alice first colors u with α, then Bob can color w with β, leaving v with no legal color. Hence Bob wins.

The least r such that Alice has a winning strategy for this game on G is called the *game chromatic number of* G, denoted $\chi_{\mathrm{g}}(G)$. The game chromatic number was first introduced by Bodlaender in [4]. It has since been studied extensively, including in [11, 23, 27].

In our example, it should be clear that Bob will win the 1-coloring game on G. We have demonstrated a strategy for Alice to win the 2-coloring game on G. Thus, $\chi_{\mathrm{g}}(G) = 2$.

1.1 Trees and Forests

We begin by looking at the game chromatic number for trees and forests and comparing it to the usual chromatic number of a graph, denoted $\chi(G)$.

Consider the rooted tree T in Figure 2. The chromatic number of T, $\chi(T)$, is 2 since we can alternate colors on each level. In particular, each vertex will have a different color from its parent and from its children. All children of a vertex v can have the same color since no children of v are adjacent to each other.

Since every tree can be represented by a rooted tree, we can generalize the method in the example to show that for any tree, T, $\chi(T) \leq 2$.

Now consider the game chromatic number of a tree T. If Bob is always able to color two children of a vertex, v, different colors, then Alice cannot color T with two colors, since v will have no legal color. The reader can check that the tree in Figure 2 has game chromatic number greater than 2. No matter what Alice does, Bob is able to force a vertex to have no legal color.

Fig. 2 Tree T.

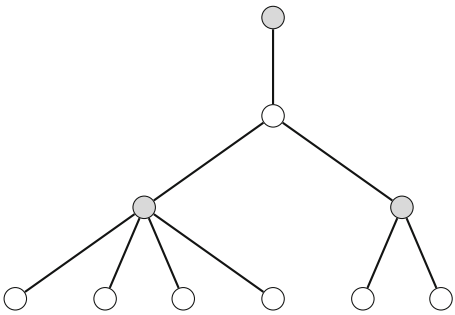
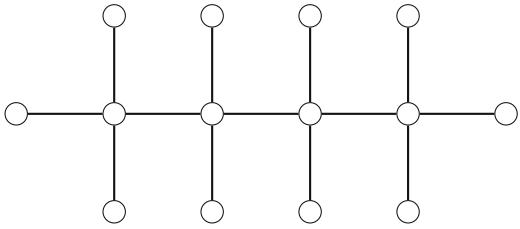


Fig. 3 The smallest tree T with $\chi_g(T) = 4$.



The following theorems give some known results for the game chromatic number of trees and forests, which will be explored in more detail in Section 2.

Theorem 1 (Faigle, et al. [23]). *If F is a forest, then $\chi_g(F) \leq 4$. Moreover, there exists a forest F with $\chi_g(F) = 4$.*

Theorem 2 (Dunn et al. [18]). *Let F be a forest and let $\ell(F)$ be the length of the longest path in F. Then $\chi_g(F) = 2$ if and only if:*

1. $1 \leq \ell(F) \leq 2$, *or*
2. $\ell(F) = 3$, $|V(F)|$ *is odd, and every component with diameter* 3 *is a path.*

Consider the tree in Figure 3. It is the smallest tree T with $\chi_g(T) = 4$.

Theorem 3 (Dunn et al. [18]). *If F is a forest with $|V(F)| \leq 13$ then $\chi_g(F) \leq 3$.*

Although we know a bound for the game chromatic number of trees and forests, classifying trees with game chromatic number 3 or 4 remains open.

It has been well established that the parameter $\chi_g(G)$ has some interesting and possibly unexpected properties. For example, while it is true that if H is a subgraph of G then $\chi(H) \leq \chi(G)$, it is not necessarily the case that $\chi_g(H) \leq \chi_g(G)$. For example, as discussed in [3], if $G = K_{n,n}$ and $n \geq 2$ we have that $\chi_g(G) = 3$. However, if M is any perfect matching in G, then $\chi_g(G - M) = n$. So for $n \geq 4$, $\chi_g(G - M) > \chi_g(G)$.

We now briefly look at some variations of the coloring game.

1.2 The (r, d)-Relaxed Coloring Game

Let r be a positive integer, d be a nonnegative integer, and G be a finite graph. Two players, Alice and Bob, play a game on G by coloring the uncolored vertices with colors from a set X of r colors. At all times, the subgraph induced by a color class must have maximum degree at most d. Alice wins the game if all vertices are eventually colored; otherwise, Bob wins. We call this the (r, d)-relaxed coloring game. In particular, we have modified the original coloring game by changing the definition of a legal color. At any point in the game for any $\alpha \in X$, let C_α be the set of vertices colored α at that point. We say that a color $\alpha \in X$ is *legal* for the uncolored vertex u, if after u is colored α we have that $\Delta(G[C_\alpha]) \leq d$, where $G[C_\alpha]$ is the subgraph of G induced by the vertices in C_α.

The least r for which Alice has a winning strategy for this game on G is called the *d-relaxed game chromatic number of* G, denoted ${}^{d}\chi_{\mathrm{g}}(G)$. When $d = 0$, we have the usual coloring game and the game chromatic number of G, $\chi_{\mathrm{g}}(G)$. The relaxed game chromatic number has been considered in [6, 14–16, 26]. We will examine the d-relaxed game chromatic number further in Section 3.

Suppose that Alice and Bob are playing the (r, d)-relaxed coloring game on a graph G. For the purposes of analyzing strategies for Alice or Bob, we note that a color α is legal for an uncolored vertex u if the following two conditions hold:

1. The vertex u has at most d neighbors already colored α.
2. If v is a neighbor of u and v is already colored α, then v has at most $d - 1$ neighbors already colored α.

It might seem that if we increase the defect, then Alice could more easily win with fewer colors. It is known that if $G = K_{n,n}$ with $n \geq 2$, then ${}^{1}\chi_{\mathrm{g}}(G) = n$. But as we saw in Section 1.1, if $G = K_{n,n}$ and $n \geq 2$ we have that $\chi_{\mathrm{g}}(G) = 3$. Thus, for $n \geq 4$, there is a class of graphs for which ${}^{0}\chi_{\mathrm{g}}(G) = 3$ but ${}^{1}\chi_{\mathrm{g}}(G) \geq 4$.

Although the (r, d)-relaxed game is a common way to change the definition of a legal color, there are many other ways to relax the conditions of the game. For example, we also explore the notion of a *clique-relaxed* game in Section 4. In the *k-clique-relaxed r-coloring game* on a finite graph G, a color α is *legal* for an uncolored vertex u if coloring u with α does not result in a monochromatic $(k + 1)$-clique. As before, we can find the least number of colors required for Alice to have a winning strategy on G. We call this the *k-clique-relaxed game chromatic number of* G, denoted ${\chi_{\mathrm{g}}}^{(k)}(G)$.

1.3 Edge Coloring and Total Coloring

Another variation of the coloring game is to consider the game on edges rather than vertices. The rules are similar to the vertex coloring game. We begin with a finite graph, G, and a set X of r colors. Two players, Alice and Bob, alternate coloring the

uncolored edges of G using colors from X. At each step of the game, the players must choose to color an uncolored edge with a legal color. Alice goes first. A color $\alpha \in X$ is *legal* for an uncolored edge e if e has no incident edges already colored α. Alice wins the game if all edges of the graph are colored; otherwise, Bob wins. The least number of colors required for Alice to have a winning strategy on G is called the *game chromatic index of* G, denoted $\chi_g{}'(G)$. This is explored in Section 5.

Once we have considered edge coloring and vertex coloring, a natural extension might be to consider *total coloring*. In a total coloring game, explored in Section 6, Alice and Bob alternate coloring vertices and edges.

2 Classifying Forests by Game Chromatic Number

In this section we examine the classic coloring game played on forests and trees. In [23], it was shown that the game chromatic number of a forest is at most 4. In this section, we focus on the question "Do there exist simple criteria for determining the game chromatic number of a forest?"

It is trivial to see that only edgeless forests have game chromatic number 1. Further exploration of this question is in the paper [18]. We will prove some results from the paper to highlight strategies which are helpful in exploring game coloring problems. First, we show that forests with game chromatic number 2 have determining criteria. We also demonstrate how a separator strategy is used to prove that small trees have game chromatic number at most 3. There is not any known criteria for differentiating between forests with game chromatic number 3 and 4.

2.1 Forests with Game Chromatic Number 2

We begin by showing the proof of Theorem 2, restated below.

Theorem 2 (Dunn et al. [18]). *Let F be a forest and let $\ell(F)$ be the length of the longest path in F. Then $\chi_g(F) = 2$ if and only if:*

1. $1 \leq \ell(F) \leq 2$, *or*
2. $\ell(F) = 3$, $|V(F)|$ *is odd, and every component with diameter* 3 *is a path.*

Proof. Suppose that the condition is not met by some forest F. If $\ell(F) < 1$, then it is easy to see that $\chi_g(F) = 1$. Thus $\ell(F) \geq 3$. First, assume that $\ell(F) > 3$ and that Alice colors x with α as her first move. If there is a vertex y at distance 2 from x then Bob can color y with β. The common neighbor of x and y has no legal color available, so Bob wins. Thus we assume that there is no such y. Then, some component of F not containing x must contain a path subgraph with 5 vertices, P_5. Let v_1, v_2, v_3, v_4, v_5 be the vertices of this P_5. Bob will color v_3 with α on his first turn, and on his second he can color either v_1 or v_5 with β and win the 2-coloring game on F.

Now assume that $\ell(F) = 3$ and that Alice colors x with α as her first move; if there is a vertex y at distance 2 from x, then Bob will win the 2-coloring game as above. Thus x is not in a component with diameter 3. If there is a component with diameter 3 that is not a path, then this component has a vertex, u, of degree 3 or more that is adjacent to at least two leaves, v_1 and v_2. Bob colors v_1 with α. Unless Alice colors u with β, Bob will win on his second move by coloring an uncolored neighbor of u with β. If Alice colors u with β, then because the component has diameter 3, there exists a vertex at distance 2 from u. Bob colors this with α and wins the 2-coloring game on F.

Thus every component with diameter 3 is a path, but $|V(F)|$ is even. Let T_1 be a path component with diameter 3. Because $|V(F)|$ is even Bob can play so that Alice is the first to color a vertex in T_1 (unless Bob wins the game before that point). When Alice colors a vertex x in T_1, Bob immediately colors a vertex at distance 2 from x with a different color and wins the 2-coloring game on F.

Now suppose that a forest F satisfies the condition of the theorem. We will show that Alice can win the 2-coloring game on F. If $\mathrm{diam}(F) \leq 2$ then note that Bob can only win if, in a component of diameter 2, two leaves are colored using different colors before the central vertex is colored. Therefore, Alice will win the 2-coloring game by using the following strategy. (1) If possible, color the central vertex in the component Bob most recently played in. (2) Otherwise, color the central vertex in a component with no colored vertices. (3) If neither of those are possible, color any uncolored vertex.

If there are components of diameter 3, then they are all paths and $|V(F)|$ is odd. By parity, Alice can always choose a vertex to color so that either it is not in a P_4 component or it is in the same P_4 component that Bob just played in. Alice's strategy is as follows: if Bob colors x in a P_4 component using color α, Alice colors the unique vertex at distance 2 from x with α. If it is Alice's first turn, or if Bob did not color in a P_4 component, then she follows her strategy from the case where $\mathrm{diam}(F) \leq 2$ with the additional restriction that she does not choose a vertex from a P_4 component. Using this strategy, no uncolored vertex will ever have two differently colored neighbors. Therefore, Alice will win the 2-coloring game on F.

In the proof above, we identify configurations which allow Bob to win the r-coloring game (in this case, an uncolored vertex with two differently colored neighbors), and implement a strategy for Bob (or Alice) which forces (or avoids) these configurations. These two methods are at the heart of many proofs in game coloring, so finding such configurations is both a tractable and useful exercise for students.

2.2 Smallest Tree with Game Chromatic Number 4

Now that forests with game chromatic number 2 are classified, the investigation turns to the differences between forests with game chromatic numbers 3 and 4. Some simple observations can be made: in order for Bob to win the 3-coloring game on a

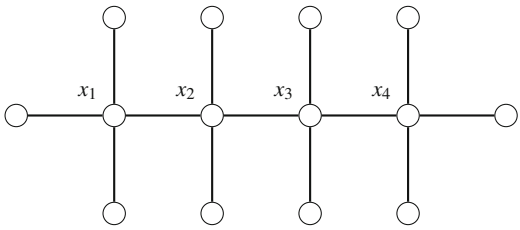


Fig. 4 The tree T' has game chromatic number 4 by Theorem 4

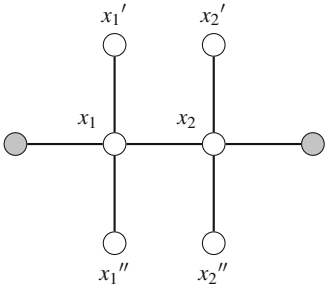


Fig. 5 The tree T_0; the two shaded vertices have color α and unshaded vertices are uncolored.

forest F there must be a vertex of degree 3 or more. Thus linear forests have game chromatic number at most 3. It is a natural question to ask how small a forest with game chromatic number 4 can be.

Using small configurations on which Bob can win the 3-coloring game, we show that the graph T' in Figure 4 is the smallest tree with game chromatic number 4.

Lemma 1. *Let T_0 be the partially colored tree shown in Figure 5. Bob can win the 3-coloring game on T_0.*

Proof. Suppose it is Alice's turn. If she colors x_1 or x_2, or if she colors any uncolored leaf with a color other than α, Bob can immediately color so that either x_1 or x_2 has no legal color available. Therefore, we may assume without loss of generality that Alice colors x_1' with α. Bob will color x_2' with β. Now if Alice colors x_2 with γ, then Bob can color x_1'' with β and win. If Alice does not color x_2 with γ, then Bob will color either x_1 or x_2'' with γ and win because x_2 has no legal color available.

Suppose instead that it is Bob's turn. He colors x_2' with β. Exactly as above, Bob has a winning response to any move Alice can make.

Theorem 4 (Dunn et al. [18]). *The tree T' (Figure 4) has game chromatic number 4.*

Proof. We show that $\chi_g(T') = 4$ by showing that Bob has a winning strategy when playing the 3-coloring game on T'. Bob's goal is to attain the configuration T_0 from Lemma 1. If Alice colors (with α) one of x_1, x_2, x_3, x_4 or a leaf adjacent to either x_1 or x_4, then Bob colors one of the x_i at distance 3 away from Alice's move using α. By Lemma 1, Bob will win the 3-coloring game. Otherwise, we may assume without

loss of generality that Alice colors a leaf adjacent to x_2 with α; Bob responds by coloring the other leaf adjacent to x_2 with β. Unless Alice now colors x_2, Bob can make it uncolorable on his next turn. However, if Alice colors x_2 (necessarily with γ), then Bob can color a leaf of x_4 with γ and win by Lemma 1.

The proof of Theorem 4 highlights some of the potential difficulty in using partially colored subgraphs in a larger graph. When creating a strategy for one player that focuses on a subgraph, an opponent can either color a vertex in that subgraph, a vertex that is not on that subgraph but affects color choices for uncolored vertices in the subgraph, or a vertex that has no effect on the subgraph in question.

To show that trees with fewer vertices than T' have game chromatic number at most 3, we need some lemmas regarding small trees. More detailed proofs can be found in [18].

Lemma 2. *If T is a tree with $|V(G)| \leq 13$, then there exists a vertex $v \in V(T)$ such that every component of $T - v$ has at most 6 vertices.*

Proof. Choose v so that the order of the largest component in $T - v$ is minimized.

Lemma 3. *If T is a tree on at most 7 vertices, then T has at most two vertices of degree 3 or more and at most one vertex of degree 4 or more.*

Proof. This can be shown by examining the degree sum of a counterexample.

In order to prove Theorem 3, it is useful to describe a *separator strategy*. At any point in the game, the forest F will be partially colored. Using the partial coloring, we define a collection of *trunks* as follows. For each colored vertex x with degree d, split x into d colored vertices, say $x_1, x_2, \ldots, x_d$, so that each x_i is colored the same as x and is adjacent to exactly one neighbor of x. After each colored vertex is split in this fashion, the resulting graph will be a collection of trees, called *trunks*. Any colored vertex in a trunk must be a leaf of that trunk; if every vertex in a trunk is colored then it is a K_2 component. Defined a different way, given a forest F and a partial coloring, a *trunk* R is a maximal connected subgraph of F such that every colored vertex in R is a leaf of R. It is important to note that coloring a vertex in one trunk has no effect on available colors in another trunk. We now restate and prove Theorem 3.

Theorem 3 (Dunn et al. [18]). *If F is a forest with $|V(F)| \leq 13$ then $\chi_g(F) \leq 3$.*

Proof. Let F be a forest with $|V(F)| \leq 13$ and let T be the component with the most vertices. By Lemma 2, there exists a vertex $v \in V(T)$ such that every component of $T - v$ has size at most 6. Alice colors this vertex, and now each trunk of F has order at most 7 and at most 1 colored vertex.

We call a vertex v *dangerous* if v has at least as many uncolored neighbors as legal colors available for it. Alice will win the 3-coloring game if there are no dangerous vertices in F. We show that Alice has a strategy which eliminates dangerous vertices in each trunk of F, and therefore has a strategy that will win the

3-coloring game on F. Alice will always play in the same trunk as Bob; if there are no uncolored vertices in that trunk Alice will play in a new trunk as if Bob just colored v.

Let R be the trunk of F where Bob just played. By Lemma 3, R has at most 2 dangerous vertices. If there are fewer than 2 dangerous vertices then Alice will color any remaining dangerous vertex on her turn, and then no dangerous vertices exist in R. If there are two dangerous vertices in R, then we consider the following cases:

Case 1 Some dangerous vertex x has no colored neighbors.

Alice colors the other dangerous vertex with any legal color. After Bob's next move in this trunk, Alice colors x if it is still uncolored.

Case 2 Some dangerous vertex x has two colored neighbors.

Alice colors x with any available color. Since the colored vertices of R lie on a path, the other dangerous vertex x' has at most one colored neighbor. After Bob's next move in this trunk, Alice colors x' if it is still uncolored.

Case 3 Each dangerous vertex has exactly one colored neighbor.

By Lemma 3 at least one dangerous vertex, say x, has degree 3; let the other dangerous vertex be x'. If x and x' are not adjacent then Alice can color x first and then x' later, as in Case 2. If x and x' are adjacent, then x must have an uncolored neighbor u distinct from x'. Alice colors u with the same color as the previously colored neighbor of x. Therefore, x is no longer a dangerous vertex, and x' has only 1 colored neighbor. After Bob's next move in this trunk, Alice can color x' if it is still uncolored.

In all cases, Alice is able to eliminate the dangerous vertices before Bob is able to surround any vertex with 3 distinct colors. Therefore, Alice can win the 3-coloring game on F.

By using this trunk coloring strategy and considering the different cases on maximum degree and diameter, it can be shown that T' is in fact the unique forest on 14 vertices with game chromatic number 4 [18]. A main idea in this proof is that if there are few dangerous vertices, then Alice will have an easier time creating a winning strategy. Further utilizing this idea, one can prove some conditions under which a forest F has $\chi_g(F) \leq 3$.

Theorem 5 (Dunn et al. [18]). *A forest F has game chromatic number at most 3 if there exists a vertex b such that, if Alice starts by coloring b, every trunk R of F either*

1. *has no neighboring vertices whose degrees are both 3 or more, or*
2. *has a vertex $v \in R$ with degree 3 or more which is adjacent to every vertex in R with degree 3 or more, and v is not adjacent to a colored vertex.*

In fact, [18] proves something slightly stronger, but we omit that statement here in order to avoid a technical definition.

The proofs of Theorems 4 and 3 might lead one to believe that, if we can restrict the maximum degree of a forest, then Bob will not have the flexibility required to win the 3-coloring game. However, the following result shows that this is false.

Theorem 6 (Dunn et al. [18]). *There exists a tree T with $\Delta(T) = 3$ and $\chi_g(T) = 4$.*

When outlining a strategy for Alice in the 3-coloring game on trees with maximum degree 3, the authors of [18] found a partially colored subgraph for which the strategy did not work. In fact, Bob could win the 3-coloring game on this subgraph. Working backwards, the authors pieced together a tree on which Bob can force this partially colored subgraph to occur.

As an aside, this is not an uncommon path towards interesting results. Recently, Steinberg's Conjecture (every planar graph without 4-cycles and 5-cycles is 3-colorable) was proven to be false [7] using this "method". As a team of mathematicians were working on a lemma needed to prove the conjecture, a gap in the proof turned into a counterexample to that lemma, which led to a counterexample to the entire conjecture!

Every known example of trees with $\Delta(T) = 3$ and $\chi_g(T) = 4$ have even order; the proof of Theorem 6 relies on the fact that Alice cannot avoid coloring first in a particular subgraph. One unanswered question regarding forests with game chromatic number 3 and 4 is if it is possible that maximum degree 3 implies $\chi_g(T) \leq 3$ for trees with odd order.

3 Relaxed-Coloring Games

In this section we will consider a variation of the game in which the subgraphs induced by each color class must satisfy the condition of having maximum degree bounded by a predetermined constant d. In this version of the game, the players are in the process of creating a *defect coloring* or *relaxed coloring* of the graph. Such colorings have been examined in [8–10,21]. To highlight some of the strategies that have been employed to provide upper bounds on the associated parameter, we will provide results with trees. However, these strategies have been useful with many classes of graphs.

First we will define the game, which we will refer to as the (r, d)-*relaxed coloring game*. Let G be a finite graph and let X be a set of r colors, for some positive integer r. Let d be a nonnegative integer. We call d the *defect*. For each $\alpha \in X$, we call the set of all vertices colored α the *color class of* α, denoted C_α. In this version of the game, a color $\alpha \in X$ is *legal* for an uncolored vertex u if, after u has been colored α, the subgraph induced by color class C_α must have maximum degree at most d. More succinctly, at every point in the game, for every $\alpha \in X$, it must be true that $\Delta(G[C_\alpha]) \leq d$. Note that if $d = 0$, this is the original version of the coloring game. For a fixed d, the least r such that Alice has a winning strategy for this game is called the *d-relaxed game chromatic number of* G and is denoted by ${}^d\chi_g(G)$. For a fixed r, the least d for which Alice has a winning strategy for this game is called the *r-game defect of* G and is denoted by $\mathrm{def}_g(G, r)$.

At any time in the game, we define the *defect* of a colored vertex $x \in C_\alpha$ to be the number of neighbors of x already colored α. We denote this value by $\mathrm{def}(x)$.

In terms of the analysis of a strategy, one must evaluate two things when considering the color α for the uncolored vertex u:

1. the vertex u must be adjacent to at most d vertices already colored α; and
2. if v is adjacent to u and v has already been colored α, then v can be adjacent to at most $d-1$ vertices already colored α.

Theorem 7 (Chou et al. [6]). *If T is a tree, then ${}^1\chi_g(T) \leq 3$.*

We will present two proofs of this theorem. First, we will provide the argument used by Chou, Wang, and Zhu in [6]. This uses a separator strategy as introduced in Section 2. The second result uses an *activation strategy*. Activation strategies have proven to be quite useful for many classes of graphs, and can be seen as the culmination of the work in a number of papers [11,23,27,28,30,31]. We will present a number of activation strategies in later sections.

Proof. (Separator Argument) Suppose in the process of the game, the tree T is partially colored. We obtain a collection of subtrees as follows: for each colored vertex x with degree d, we split x into d colored vertices, say $x_1, x_2, \ldots, x_d$, so that each x_i is colored the same color as x and is incident with exactly one of the original edges incident to x in T. After splitting each of the colored vertices of T, we obtain a collection of smaller partially colored trees, say $T_1, T_2, \ldots, T_m$, which we will call *trunks*, such that $\cup_{i=1}^{m} E(T_i) = E(T)$. Note that in each T_i, it is the case that only some of the leaves may be colored.

Alice's goal in selecting the vertices to color is simply to ensure that after she has colored her chosen vertex, each of the trunks of the partially colored T has at most two colored leaves. Suppose Alice can achieve this goal. Then after Bob colors a vertex, each trunk T_i has at most two colored vertices, with the exception of one trunk which may have three colored vertices. Moreover, if the trunk T_i has three colored leaves, then one of those leaves was just colored by Bob. We will call this the *new colored leaf of* T_i.

It is easy to prove inductively that Alice can achieve her goal. If after Bob's move there is a trunk T_i containing three colored leaves, then Alice will choose the vertex that lies at the intersection of the paths joining these three colored vertices. Suppose Alice has chosen vertex $u \in V(T_i)$ by this process. She will then choose a color for u as follows: if the colored leaves of T_i use at most two colors, Alice will choose a color that has not been used on T_i. Otherwise, T_i has three distinctly colored leaves. Alice will color u with the color of the new colored leaf, v, of T_i.

To show that this works, it suffices to show that in the case that T_i has three distinctly colored leaves, then the color of v is legal for u. This is obvious as v has no other colored neighbors. Thus, Alice can color u with the color assigned to v. This will affect the defect of at most two vertices: u and v. Each will then have defect of at most 1. Thus, following this strategy, all vertices of T will be colored and Alice will win.

For the second proof of this result, we will define what we will call the *Tree Strategy* for Alice. Suppose Alice and Bob are playing the coloring game on the tree $T = (V, E)$. Alice picks a vertex r and orients all edges in the tree toward r. The resulting orientation on T has the property that every vertex $v \in V \setminus \{ r \}$ has a unique outneighbor which we denote $p(v)$. We call $p(v)$ the *parent* of v and refer to v as a *child* of $p(v)$. Continuing this notation, let $p^2(v) = p(p(v))$. Inductively, if $p^i(v)$ is defined, let $p^{i+1}(v) = p(p^i(v))$. We define the set of *descendants* of a vertex v by

$$G(v) := \{ w \in V \mid v = p^k(w) \text{ for some } k \}.$$

We then define $G[v] := G(v) \cup \{ v \}$.

Throughout the game, vertices go from uncolored to colored. At any time in the game, let U be the set of uncolored vertices and C be the set of colored vertices. Alice will maintain a set A of *active* vertices. One way of viewing the notion of an active vertex is to think of it like a post-it note that Alice places on vertices that are becoming dangerous, yet not requiring immediate attention. When her strategy leads her to consider a vertex that already has such a post-it note, she knows that it is now time to color this vertex. Alice will only color vertices that are active; however, she may activate a vertex and color it on the same turn. For simplicity, we will assume that any vertex that Bob colors also becomes active; thus $C \subseteq A$. Whenever a vertex is colored it is added to C, and whenever a vertex is activated it is added to A. When a vertex v is colored, let $c(v)$ be that color. For any color $\alpha \in X$, we say that α is *eligible* for a vertex v if either $p(v) \in U$ or $c(p(v)) \neq \alpha$. Note that if $|X| \geq 2$, every uncolored vertex must have at least one eligible color.

Tree Strategy

On her first move, Alice will color the vertex r with any color. Suppose Bob has just colored vertex b. Alice's strategy will have two stages: a search stage and a coloring stage. In the search stage, she seeks the vertex u that she will color. In the coloring stage, she selects the color for u.

Search Stage

- Initial Step
  - If $p(b) \in U$, then set $x := p(b)$ and move to the recursive step.
  - If $p(b) \in C$, $c(b) = c(p(b))$, and $p^2(b) \in U$, then set $u := p^2(b)$ and move to the coloring stage.
  - Otherwise, let u be any uncolored vertex such that $p(u) \in C$ and move to the coloring stage, activating u if u is inactive.
- Recursive Step
  - If $x \notin A$ and $p(x) \in U$, then activate x, set $x := p(x)$ and repeat the recursive step.
  - Otherwise, activate x if it is inactive, set $u := x$, and move to the coloring stage.

Coloring Stage

- Choose an eligible color for u which minimizes $\mathrm{def}(u)$.

We are now ready to provide our second proof of Theorem 7.

Proof. (Activation Argument) Let T be a tree and let $|X| = 3$. Alice will employ the Tree Strategy defined above. Suppose $u = p(v)$ and u is uncolored. Then the first time a vertex in $G[v]$ is activated, necessarily by Bob, Alice will take action at u by coloring (if u is already active) or by activating.

Suppose that Alice has chosen to color vertex u. Suppose x and y are the first two children of u to be activated, and x is activated first. Note that on the turn that x is activated, Alice takes action at u by activating it. Assuming that she does not also color u on this turn, suppose that w is the first vertex in $G[y]$ to be activated. Then when w is activated (by Bob coloring it), y is activated and Alice moves to color u.

Now suppose that Alice is choosing a color for u. First note that u has at most three colored neighbors: $p(u)$ and at most two children, again say x and y. If the number of colors used on these three neighbors is at most two, then Alice's strategy dictates that she will use a color not used in the set of neighbors of u, $N(u)$. Otherwise, it must be the case that $p(u)$, x, and y are all colored distinctly. Since there is only one vertex in $G[y]$ colored, then Alice can safely color u with $c(y)$ as this results in $\mathrm{def}(u) = \mathrm{def}(y) = 1$, with the defect of no other vertex affected. We note that Bob can borrow this strategy at any time. So at any time in the game, it is possible to color any uncolored vertex with a legal color. Thus, all vertices of T will eventually be colored, and Alice will win.

We note that while the bound in Theorem 7 is known to be sharp [6], we do not know the necessary and sufficient characteristics of a tree T to determine whether ${}^1\chi_g(T) = 3$ or ${}^1\chi_g(T) = 2$.

4 The Clique-Relaxed Game

Earlier, we examined the graph coloring game using d-relaxed coloring rules on vertices and on edges. Exploring different modifications to standard coloring rules can open up a wide range of interesting competitive coloring problems. Another relaxation of standard coloring rules is *clique-relaxed coloring*, where the aim is to avoid large monochromatic cliques (complete subgraphs).

In this section, we examine the game coloring version of clique-relaxed coloring; we focus on planar and outerplanar graphs. A *planar graph* is a graph that can be drawn in the plane such that no edges cross each other. Each region bounded by the edges is called a face and the infinitely large unbounded face is called the *outer face*. One specific type of planar graph is an *outerplanar graph*, which is a planar graph that can be drawn so that every vertex belongs to the outer face.

Recall that $\omega(G)$ denotes the size of the largest clique in G. We say that a vertex coloring of a graph G is a *proper k-clique-relaxed coloring* if $\omega(H) \leq k$ for each subgraph H induced by one of the color classes. The *k-clique-relaxed chromatic number of* G, denoted $\chi^{(k)}(G)$, is the smallest k such that G has a proper k-clique-relaxed coloring.

Although there appears to be very little proven about this parameter in the literature, it seems like a natural variation of other coloring relaxations and has interesting properties with regards to competitive graph coloring. Notice that for any graph G, we get $\chi^{(1)}(G) = {}^0\chi_{\mathrm{g}}(G) = \chi(G)$ and that $\chi^{(k)}(G) \leq \chi^{(k-1)}(G)$ for every positive integer k. The following theorem gives an upper bound for $\chi^{(k)}(G)$ in terms of the standard chromatic number $\chi(G)$.

Theorem 8 (Dunn et al. [17]). *Let G be a graph. Then* $\chi^{(k)}(G) \leq \left\lceil \frac{\chi(G)}{k} \right\rceil$ *for any positive integer k.*

Proof. Let G be a graph where $\chi(G) = r$. Color the vertices of G properly with r colors which we denote $\alpha_1, \ldots, \alpha_r$. We divide the r colors into $s := \lceil \frac{r}{k} \rceil$ groups, all of size exactly k except possibly the last group. We label these groups $A_1, \ldots, A_s$ and recolor all vertices in A_i with color β_i. There are s colors used in this new coloring. Suppose, for the sake of contradiction, that $H \subseteq G$ is a $(k+1)$-clique where each vertex of H receives color β_i. Then by the pigeonhole principle, at least two vertices x and y in A_i were colored α_j originally for some j. Thus xy cannot be an edge of G, which contradicts the fact that H is a $(k+1)$-clique.

Using this theorem we are able to give bounds for $\chi^{(k)}(G)$ for some classes of graphs. The Four Color Theorem [2] shows that $\chi(G) \leq 4$ for all planar graphs G. Also, it is also easy to show that $\chi(G) \leq 3$ for outerplanar graphs G. These observations lead to the following corollary.

Corollary 1. *If G is a planar graph, then* $\chi^{(k)}(G) \leq 2$ *when* $2 \leq k \leq 3$ *and* $\chi^{(k)}(G) = 1$ *when* $k \geq 4$. *Moreover, if G is an outerplanar graph, then* $\chi^{(2)}(G) \leq 2$ *and* $\chi^{(k)}(G) = 1$ *when* $k \geq 3$.

These bounds are sharp as K_4 is a planar graph with $\chi^{(k)}(K_4) = 2$ when $2 \leq k \leq 3$ and K_3 is an outerplanar graph with $\chi^{(2)}(K_3) = 2$.

To play the *k-clique-relaxed r-coloring game* on a graph G, two players (Alice and Bob) will take turns coloring uncolored vertices of G with legal colors coming from a fixed set X of r colors. A color $\alpha \in X$ is *legal* for an uncolored vertex u if coloring u with α does not create a monochromatic $(k+1)$-clique. Said another way, α is not a legal color for u if $G[N(u)]$ contains a k-clique H where each vertex of H is colored α. Alice always colors first, and she wins the game when all the vertices are colored. Therefore, Bob will win if there is at least one uncolored vertex u in G whose neighborhood contains monochromatic k-cliques in each of the r colors. The *k-clique-relaxed game chromatic number of* G, denoted ${\chi_{\mathrm{g}}}^{(k)}(G)$, is the least r such that Alice has a winning strategy in the k-clique-relaxed r-coloring game.

The first observation is that ${}_{\chi_g}{}^{(1)}(G) = {}^0\chi_g(G)$. In [17], the authors investigate the k-clique-relaxed game chromatic number on outerplanar graphs. Because the maximum clique size in an outerplanar graph is 3, it follows that ${}_{\chi_g}{}^{(k)}(G) = 1$ when G is an outerplanar graph and $k \geq 3$. Therefore, the focus is on the 2-clique-relaxed coloring game. In the following result, Alice's strategy is to use a vertex ordering given by Guan and Zhu [25] to implement a separator strategy similar to the one used on trees (see Section 3).

Theorem 9 (Dunn et al. [17]). *Let G be an outerplanar graph. Then ${}_{\chi_g}{}^{(2)}(G) \leq 4$.*

Proof. Let G be an outerplanar graph. Alice's strategy is to define auxiliary graphs G' and T, and to use these graphs to choose which vertex to color. First, let G' be the graph obtained by adding edges to G until the graph is maximally outerplanar. Guan and Zhu [25] showed that for every maximally planar graph, there is a linear ordering $L := v_1, \ldots, v_n$ of the vertices of H such that v_1v_2 is on the outer face and, for all $i \geq 3$, the vertex v_i is adjacent to exactly two vertices $v_{a(i)}$ and $v_{b(i)}$ such that $a(i) < b(i) < i$. We call $v_{a(i)}$ and $v_{b(i)}$ the *major parent* and *minor parent* (respectively) of v_i.

To create the graph T, Alice deletes from G' all the edges of the form $v_iv_{b(i)}$. Note that v_1v_2 is still an edge and for each $i \geq 3$ the vertex v_i is adjacent to exactly one vertex with lower index; thus, T must be a tree. As in the separator strategy of Section 3, Alice can ensure that after her turn each trunk of T has at most two colored vertices. Bob may possibly color a third vertex in a trunk, so each uncolored vertex v_i that Alice chooses has at most 3 colored neighbors in T. It is possible that v_i has more colored neighbors in G, as $v_iv_{b(i)}$ could be an edge in G. Further, [25] showed that each vertex in G' is the minor parent to at most two vertices. Therefore v_i is adjacent to at most six colored vertices in the original graph G. In the 2-clique relaxed 4-coloring game, a vertex will have a legal color unless it is adjacent to monochromatic K_2 subgraphs in each of the 4 colors. Because v_i is adjacent to at most 6 colored vertices, there is a legal color for Alice to use.

Bob will also always have a legal move, because Alice's strategy leaves at most two colored vertices in each trunk of T. Therefore, uncolored vertices have at most 5 colored neighbors in G on Bob's turn.

The bound in Theorem 9 has no sharpness example; in [17] an example is given where Bob has a winning strategy for the 2-clique-relaxed 2-coloring game on an outerplanar graph. Bob's strategy uses the symmetry of the graph to create a particular partially colored subgraph on which an uncolored vertex can be made uncolorable. It remains open whether or not there exists an outerplanar graph where Bob has a winning strategy for the 2-clique-relaxed 3-coloring game.

Theorem 10 (Dunn et al. [17]). *There exists an outerplanar graph G with $\chi^{(2)}(G) \geq 3$.*

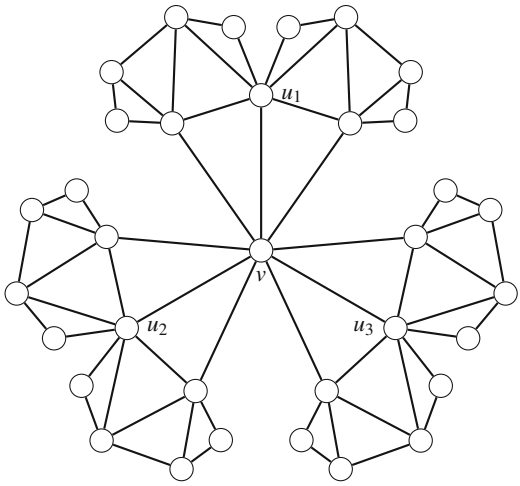


Fig. 6 An outerplanar graph with $\chi^{(2)}(G) \geq 3$.

Figure 6 provides an example of an outerplanar graph with $\chi^{(2)}(G) \geq 3$. The proof can be found in [17].

This raises the question of whether there exists an outerplanar graph G such that ${\chi_g}^{(2)}(G) = 4$. Further results in [17] show a subclass of outerplanar graphs in which no such example can be found.

The question of clique-relaxed coloring games on planar graphs also remains open. Because the maximum clique size in a planar graph is four, the games of interest on planar graphs are the 2- and 3-clique-relaxed games.

5 Edge Coloring

The focus of this section is a variation of the game in which Alice and Bob alternate coloring edges rather than vertices. The most obvious consequence of analyzing this version of the game is that the maximum degree of the graph becomes an important component of the upper bounds for the corresponding parameters.

Let G be a finite graph and let r be a positive integer and d be a nonnegative integer. As used before, d is the *defect* and X is a set of r colors. The players alternate coloring, with Alice coloring an edge first. We say that a color $\alpha \in X$ is *legal* for an uncolored edge e if the following conditions are satisfied:

1. the edge e is incident with at most d edges already colored α; and
2. if e' is an edge incident to e and e' has already been colored α, then e' is adjacent to at most $d - 1$ edges already colored α.

Note that if e is colored α, then at this point in the game, every edge has at most d neighbors colored α. Alice wins if every edge is eventually legally colored. Bob wins if there comes a time in the game when there is an uncolored edge for which

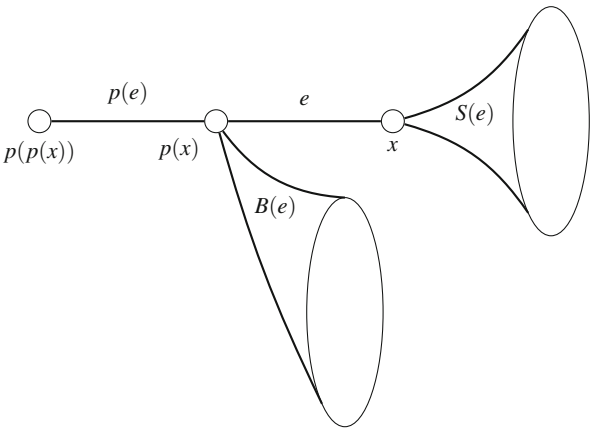


Fig. 7 For an edge $e = xp(x)$, the vertex $p(p(x)) = p^2(x)$, the edge $p(e)$, and the sets $B(e)$ and $S(e)$.

no legal color exists. For a fixed defect d, the least r such that Alice has a winning strategy for this game is called the *d-relaxed game chromatic index of* G, denoted ${}^d\chi_g'(G)$. Similarly, for a fixed r, the *r-edge-game defect of* G, denoted $\mathrm{def}_g'(G, r)$, is the least d such that Alice has a winning strategy. This game was first introduced in [13].

We will present some of the work done in [19] restricted to trees, which generalized the results in [13]. Further work in this area can be seen in [1,5,22,29]. We begin by defining terminology and notation, and by providing a winning strategy for Alice in the edge coloring game on trees. Let T be a tree with $\Delta(T) = \Delta$ for some positive integer Δ. For her strategy, Alice chooses an arbitrary leaf $r \in V(T)$ at which she roots T. She then regards all edges in T as oriented toward r. Let e_0 be the unique edge in T that is incident to r. For each vertex $v \in V \setminus \{r\}$, define $p(v)$ to be the unique outneighbor of v. Then for each edge $e \in E$, there is a unique vertex $x \in V$ such that $e = xp(x)$. We now introduce some terminology, as illustrated in Figure 7.

For every edge $e = xp(x)$ with $e \neq e_0$, define the *parent of* e, denoted $p(e)$, to be the edge $p(x)p^2(x)$, where $p^2(x) = p(p(x))$. We say that e is a *child of* $p(e)$. Note that, because $p(x)$ is well defined, $p(e)$ is also well defined. Whenever $p^i(e)$ is defined and $p^i(e)$ is not incident with the root, define $p^{i+1}(e) = p(p^i(e))$. As in the second (activation) proof of Theorem 7, we define the *descendants of* e to be

$$G(e) = \{e' \in E \mid e = p^k(e') \text{ for some positive integer } k\}.$$

For each edge $e = xp(x)$, define the *siblings of* e to be

$$B(e) = \{yp(y) \in E \mid p(y) = p(x) \text{ and } y \neq x\}$$

and $B[e] = B(e) \cup \{e\}$. Define the *children of* e to be

$$S(e) = \{yp(y) \in E \mid x = p(y)\}.$$

We call the set of all edges incident to an edge e the *neighborhood of* e, denoted $N(e)$.

Fix $j \in [\Delta - 3]$ and let X be a set of $\Delta - j$ colors. Note that $|X| \geq 3$. At any point in the game, let C and U be the set of colored and uncolored edges, respectively. For each $\alpha \in X$, we call the set of all edges colored α the *color class of* α, denoted C_α. For a colored edge e, denote the color of e by $c(e)$.

Similar to the notion with vertices, for each colored edge e, define the *defect of* e to be the number of neighbors of e colored with $c(e)$. If e is uncolored, we set the defect of e to be zero. We denote the defect of e by $\mathrm{def}(e)$. Thus

$$\mathrm{def}(e) = \begin{cases} \left|N(e) \cap C_{c(e)}\right|, & \text{if } e \in C; \\ 0, & \text{otherwise.} \end{cases}$$

We say that color $\alpha \in X$ is *eligible* for edge e if $p(e) \notin C_\alpha$. We denote the set of eligible colors for e by $X(e)$. When coloring an edge e, Alice always chooses an eligible color. Note that since $|X| \geq 3$, this is always possible.

In most strategies for vertex-coloring games, Alice avoids increasing the defect of a vertex when she can. The interesting aspect of this edge strategy is that in some cases, it will be necessary for her to do just the opposite. She will attempt to reach a minimum threshold for the defect of some edges. For any edge e, we say that $B[e]$ is *secure* if there exist edges $e_1, e_2, \ldots, e_{j+1} \in B(e)$ and a color α such that $c(e_i) = \alpha$ for $i \in [j+1]$. In other words, $B[e]$ is secure if e has $j+1$ siblings colored with the same color. Note that if $B[e]$ is secure, then the number of distinctly colored siblings of e is at most

$$|B(e) \setminus \{e_1, e_2, \ldots, e_j\}| \leq \Delta - j - 2.$$

As $|X(e)| \geq \Delta - j - 1$, there is always a legal eligible color for an uncolored edge e when $B[e]$ is secure.

We will now define the strategy that Alice will use for this game with trees. This strategy is a modification of the activation strategy developed in [13]. In response to Bob's moves, Alice designates certain edges *active*, as above in the second proof of Theorem 7; precisely how she chooses these edges will be explained below. When an edge e becomes active, we say that e has been *activated*. In addition, all colored edges are active. We denote the set of active edges by A, and note that $C \subseteq A$. This set has the property that once an edge e is in A, e will remain active for the remainder of the game.

Tree Strategy for Edges

Alice begins the game by coloring e_0 with any color. Suppose now that Bob has just colored edge $b = xp(x)$ in T for some $x \in V \setminus \{r\}$ and some $i \in [n]$. Alice's response has two stages: a Search Stage and a Coloring Stage. In the Search Stage, Alice finds an edge e to color. In the Coloring Stage, Alice chooses a color for e.

Search Stage

- If $p(b) \in U$, then activate each edge along the (x, r)-path until reaching an edge g with $p(g) \in A$. [Note that this includes Alice activating the edge b.] If $p(g) \in C$, set $e := g$. Otherwise, set $e := p(g)$.
- If $p(b) \in C$ with $c(p(b)) = c(b)$ and $p^2(b) \in U$, then set $e := p^2(b)$.
- If $p(b) \in C$ with $c(p(b)) = c(b)$, $p^2(b) \in C$, and $B(p(b)) \cap U \neq \emptyset$, then set e to be any uncolored sibling of $p(b)$.
- Otherwise, set e to be any uncolored edge whose parent is colored.

Coloring Stage

- If $B[e]$ is secure, then color e with an eligible color for e that does not appear among the siblings of e.
- Otherwise, $B[e]$ is not secure. Let f be the last edge to be colored with a color eligible for e such that $c(f) = c(p(f))$ and $p(f) \in B(e)$. If such an edge exists, then color e with $c(f)$. If no such edge exists, then color e with any eligible color for e that minimizes $\mathrm{def}(e)$.

We now present the theorem and proof from [19] for trees.

Theorem 11 (Dunn et al. [19]). *Let T be a tree and $\Delta(T) = \Delta$ for some positive integer Δ. Let j be an integer with $0 \leq j \leq \Delta - 1$, and define $h(j) = 2j + 2$. Then $\mathrm{def}_{\mathrm{g}}{}'(T, \Delta - j) \leq h(j)$. Moreover, if $d \geq h(j)$ then ${}^{d}\chi_{\mathrm{g}}{}'(T) \leq \Delta - j$.*

Proof. Suppose that Alice and Bob are playing the $(\Delta - j, d)$-relaxed edge coloring game on T for some $d \geq h(j)$. Note that when either $j = \Delta - 1$ or $j = \Delta - 2$, the result is immediate. Hence, it will suffice to consider the game with color set X with $|X| \geq 3$. We will assume that Alice uses the Tree Strategy for Edges.

Claim. If $e \in U$, then e has at most two active children. Furthermore, when e has two active children, Alice colors e.

Proof. Let f be the first active child of e. When Alice activates f, she also activates e. Note that while e is uncolored, Alice never colors an edge in $G(e) \setminus G(f)$ before Bob. If Bob colors an edge $b \in G(e) \setminus G(f)$, Alice activates $p(b), p^2(b), \ldots$, and so on, until she reaches e. Since e is active, Alice colors e.

Claim. Suppose that Alice has chosen to color edge e with $\alpha \in X$. Then at the end of Alice's turn, $\mathrm{def}(e) \leq j + 2$.

Proof. Since $\alpha \in X(e)$, then $p(e)$ does not contribute to the defect of e. By Claim 5, e has at most two active children; hence, e has at most two children colored α. If $B[e]$ is secure, then Alice would have chosen a color that does not appear among the siblings of e. In this case, $\mathrm{def}(e) \leq 2$. Otherwise, when $B[e]$ is not secure, there are at most j siblings of e colored α. Thus, $\mathrm{def}(e) \leq j + 2$.

Claim. Suppose that e is about to be colored α and $p(e) \in U$. Then e has at most one child colored α. Furthermore, if e has a child colored α, then Bob colors e and Alice colors $p(e)$.

Proof. If some sibling $f \in B(e)$ is the first active child of $p(e)$, then Alice colors $p(e)$ when the first edge in $G(p(e)) \setminus G(f)$ is activated. Since $p(e)$ is uncolored and e is to be colored, we conclude that e has no active children and hence no children colored. So assume that e is the first active child of $p(e)$. Note that $p(e) \in A$. If an edge in $G(p(e)) \setminus G(e)$ is then activated, Alice colors $p(e)$. Otherwise, we may assume that e has no colored siblings at the time when e is colored. Before e is colored, it is incident with at most two colored edges, which are children of e. Since $|X| \geq 3$, there is a color that does not appear on any child of e. Then, because Alice will choose a color to minimize $\mathrm{def}(e)$, Alice never chooses to color e with α if a child of e has already been colored α. So, if e has two active children before e is colored, then Alice colors e with α only when neither child is colored α. Thus, if e has a child colored α, then Bob must be coloring e with α, and since $p(e) \in A$, Alice responds by coloring $p(e)$.

Suppose $f \in S(e) \cap C_\alpha$. By Claim 5, if f has a child colored α before f is colored, then Bob must have colored f and Alice responds by coloring e. Thus $\mathrm{def}(f) = 2$ once e is colored. Otherwise, f has no children colored α before f is colored. Since e has at most two active children before e is colored, f has at most one sibling colored α before e is colored. Then $\mathrm{def}(f) \leq 2$ once e is colored.

Now consider the siblings of e. If $B[e]$ is secure, since Alice is choosing to color e with α, then α does not appear among the siblings of e. Hence, coloring e with α does not affect the defect of any edge in $B(e)$.

Finally, we consider the case where $B[e]$ is not secure.

Claim. Suppose Alice has chosen to color edge e with $\alpha \in X$ and $B[e]$ is not secure. If there exists an edge $f \in B(e) \cap C_\alpha$, then $\mathrm{def}(f) \leq 2j + 2$ once e is colored.

Proof. Let $E' = B(e) \cap C_\alpha$. Since $B[e]$ is not secure, we have that $|E'| \leq j$. Let $f \in E'$ such that $|S(f) \cap C_\alpha|$ is maximal, and let

$$S(f) \cap C_\alpha = \{s_1, s_2, \ldots, s_m\},$$

where $i < j$ implies that s_i is colored before s_j. We show that $m \leq |E'| + 2$. By Claim 5, f has at most two active children before it is colored. Hence, f has at most two children colored α before f is colored. So only the following cases need be considered:

Case 1 The edge f is colored before s_1.

Since $p(s_i) = f$ for each $i \in [n]$ and $c(f) = \alpha$, Alice does not color any s_i. For each s_i that Bob colors, Alice then colors $p(e)$ if $p(e) \in U$, an edge in $E' \setminus \{f\}$ if $p(e) \in C$, or e if $E' \cap U = \emptyset$. Then at most $|E'| + 1$ children of f are colored α before Alice colors e. Hence, $m \leq |E'| + 1$.

Case 2 The edge f is colored after s_1 and before s_2.

Alice does not color s_i for any $i \geq 2$. As in the previous case, when Bob colors s_i with $i \geq 2$, Alice then colors $p(e)$, an edge in $E' \setminus \{f\}$, or e. Thus, once $f \in C$, at most $|E'| + 1$ children of f are colored α. Including s_1, we have that $m \leq |E'| + 2$.

Case 3 The edge f is colored after s_2.

If $p(e) \in U$, then Claim 5 implies that f has at most one child colored α before f is colored. Since f has two children colored α before f is colored, $p(e)$ must be colored before f. Furthermore, Alice colors f immediately after s_2 is colored, as s_1 and s_2 must be the first two active children of f. Once f is colored, each time Bob colors a child of f with α, Alice colors an edge in $E' \setminus \{f\}$. Therefore, once f is colored, Bob can color at most $|E'|$ children of f with α before e is colored. So $m \leq |E'| + 2$.

Thus, in all cases, we have that

$$m = |S(f) \cap C_\alpha| \leq |E'| + 2 \leq j + 2$$

once e is colored. Since f was chosen to maximize $|S(f) \cap C_\alpha|$ and each $f' \in E'$ has at most j siblings colored α before e is colored, we have that

$$\mathrm{def}(f') \leq |S(f) \cap C_\alpha| + j \leq 2j + 2$$

for all $f' \in E'$.

Note now that if Alice is coloring edge e with α, according to the Tree Strategy for Edges, Claim 5 guarantees that $\mathrm{def}(e) \leq h(j)$. We have also shown that for any edge $f \in N(e) \cap C_\alpha$, immediately after e is colored, $\mathrm{def}(f) \leq h(j)$. As Bob may adopt Alice's strategy at any point in the game, every edge is eventually colored, and Alice wins the game. Thus

$$\mathrm{def_g}'(T, \Delta - j) \leq 2j + 2 = h(j).$$

Moreover, if the game is being playing with some defect $d > 2j + 2$, and an edge e eventually has defect at least d, then it must be through the actions of Bob that this occurs. At the time that e is uncolored, the above arguments show that it is possible to color e with an eligible color α such that coloring e does not increase the defect of any edge e' with $\mathrm{def}(e') > 2j + 2$. Thus, for any $d \geq h(j)$, we have that

$${}^{d}\chi_{\mathrm{g}}'(T) \leq \Delta - j,$$

as desired.

In addition to possible improvements to the above bounds, there are many properties of $\chi_{\mathrm{g}}'(G)$ and ${}^{d}\chi_{\mathrm{g}}'(G)$ that remain to be studied. This variation of the coloring game would be ideal for asking additional questions and examining additional classes of graphs. While the above result is generalized in [19] to a larger class of graphs (k-degenerate), no other classes have yet been considered.

6 Total Coloring

The total coloring game is a variation of the original coloring game in which Alice and Bob are free to color vertices or edges on their turns. For the remainder of this section, we will refer to vertices and edges as *elements* of the graph. The components of the game are a finite graph G and a finite set of colors X with $|X| = r$. At any point in the game, a color $\alpha \in X$ is legal for an uncolored vertex u if u has no neighbors colored α and is incident with no edges colored α. Similarly, α is *legal* for an uncolored edge e if e is not incident with any edges colored α and neither endpoint of e is colored α. Alice wins the game if all of the elements of G are colored. Bob wins otherwise. In other words, Bob wins if there comes a time in the game when there is an uncolored element for which no legal color exists. The least r such that Alice has a winning strategy is called the *total game chromatic number of* G and is denoted $\chi_{\mathrm{g}}''(G)$.

The following work is currently in preparation [20]. To prove the following theorem, we will again have Alice employ an activation strategy, which we will refer to as the *Total Activation Strategy*. Let $G = (V, E)$ be a graph and fix a linear ordering L of V. Note that L lexicographically induces a linear ordering $\overline{L}$ of E. For any element a we use L and $\overline{L}$ to separate $N(a)$ into two sets, $N^+(a)$ and $N^-(a)$; an element $b \in N(a)$ is only in $N^+(a)$ if $b < a$. Let $N^+[a] := N^+(a) \cup \{a\}$ and $N^-[a] := N^-(a) \cup \{a\}$. At any time in the game, let U_{v} and U_{e} be the sets of uncolored vertices and edges of G, respectively. Alice will maintain two sets, A_{v} and A_{e} of active vertices and edges, respectively. Once an element has become active, it will remain active for the remainder of the game. For a given vertex v, at any given time in the game let $m(v) = \min_L (N^+[v] \cap U_{\mathrm{v}})$ be the *mother* of v. Similarly, for a given edge e, we define $m(e) = \min_L (N^+[e] \cap U_{\mathrm{e}})$ to be the mother of e. We note that the mother of any uncolored element must exist, as the element itself is a candidate.

Total Activation Strategy
On Alice's first turn, she activates and colors the first vertex in L. Now suppose that Bob has just colored element b of type t, where $t \in \{v, e\}$ and $\{t, \bar{t}\} = \{v, e\}$. (We again assume that Bob also activates b if it is not already active.) Alice must now search for the element she will color and choose a color for this element.

Search Stage

- Initial Step
  - If $m(b)$ exists, then set $x := m(b)$ and move to the recursive step.
  - If $m(b)$ does not exist and $U_t \neq \emptyset$, let u be the least element of U_t and move to the coloring stage.
  - Otherwise, let u be the least element of $U_{\bar{t}}$ and move to the coloring stage.
- Recursive Step
  - If x is active, set $u := x$ and move to the coloring stage.
  - Otherwise, activate x, set $x := m(x)$, and repeat the recursive step.

Coloring Stage

- Color u with any color legal for u.

Theorem 12 (Dunn et al. [20]). *If F is a forest with $\Delta(F) = \Delta$, then $\chi_g''(F) \leq \Delta + 4$.*

Proof. Suppose Alice and Bob are playing the total coloring game on F with color set X where $|X| = \Delta + 4$. We will assume that Alice will use the activation strategy outlined above. We will now show that for any uncolored element, there is always a legal color available at any point in the game.

Suppose v is an uncolored vertex. At any time in the game, v is incident with at most Δ edges, possibly all distinctly colored before v is activated. Additionally, at most one vertex in $N^+(v)$ can be colored before v becomes active. Finally, at most two vertices in $N^-(v)$ can be activated (or colored) before v is colored: one which activates v, and one which forces Alice to immediately color v. Thus, v is incident or adjacent to at most $\Delta + 3$ uniquely colored elements before v is colored.

Now suppose that $e = xy$ is an uncolored edge and $x > y$ in L. At any point in the game, e can be incident with at most two distinctly colored vertices before e becomes active. In addition, at most $\Delta - 1$ edges in $P(e) \cup B(e)$ can be distinctly colored before e becomes active. Finally, two children of e can be activated (or colored) before e is colored: one to activate e and one to trigger the coloring of e. Thus, e is incident with at most $\Delta + 3$ colored elements before it is colored.

Since $\Delta + 4$ colors are available, and given that Bob may utilize Alice's strategy, Alice will win. Thus, $\chi_g''(F) \leq \Delta + 4$.

A *chordal graph* is a graph in which every cycle subgraph with 4 or more vertices has a chord; in other words, a chordal graph does not have C_n as an induced subgraph for any $n \geq 4$.

Theorem 13 (Dunn et al. [20]). *If G is a chordal graph with $\Delta(G) = \Delta$ and $\omega(G) = k + 1$, then $\chi_g''(G) \leq \Delta + 3k + 2$.*

Proof. Let X be a set of colors with $|X| = \Delta + 3k + 2$ and suppose that Alice and Bob are playing the total coloring game on G with X. We will assume that Alice employs the Total Activation Strategy defined above. We will show that at any time in the game, any uncolored element has an eligible legal color available.

Suppose that v is an uncolored vertex. First, it is clear that there are at most Δ edges incident with v that could be colored before v is activated (or colored). Also, every vertex in $N^+(v)$ may be colored before v is colored. Let $S \subseteq N^-(v)$ be the set of all children of v that are activated before v is colored. Note that for any vertex $w \in S$, it must be the case that $m(w) \in N^+[v]$ since G is a chordal graph. Thus, every time a vertex in S is activated, Alice will respond by either activating or coloring a vertex in $N^+[v]$. So initially, this yields $|S| \leq 2\,|N^+[v]| \leq 2(k + 1) = 2k + 2$. However, we can improve this bound slightly by considering the turn on which Alice activates v. Suppose that Alice activates v in response to an action taken at

$w \in S$. Alice will then take another action in $N^+[v]$. So Alice has responded with two actions in $N^+[v]$ due to the action taken at w. Thus, $|S| \leq 2k+1$. So we have that before v is colored, the number of distinctly colored elements to which v may be adjacent or incident is at most

$$\Delta + \left|N^+(v)\right| + |S| \leq \Delta + 3k + 1.$$

Now suppose that $e = xy$ is an uncolored edge with $x < y$ in L. At any point in the game e can be incident to at most two colored vertices before e is activated. Additionally, at most $\Delta - 1$ edges in $P(e) \cup B(e)$ can be colored before e becomes active. Similar to the argument above for vertices, let $S \subseteq S(e)$ be the set of children of e that are activated before e is colored. Note that each time an edge in S is activated or colored, Alice will take action at an edge in $H[e]$. Thus, $S \leq 2\,|H[e]| \leq 2k$. Thus, before e is colored, the number of distinctly colored elements to which e may be incident is at most

$$\begin{aligned}\Delta - 1 + |\{x, y\}| + |H(e)| + |S| &\leq \Delta - 1 + 2 + (k-1) + 2k \\ &= \Delta + 3k.\end{aligned}$$

We note that there are multiple avenues for creating "defect" versions of this game, depending on whether you allow for adjacent vertices to receive the same color (or not), and similarly for edges. Each of these variations can lead to different results for different classes of graphs.

7 Conclusions and Problems to Consider

The area of competitive graph coloring is rich with open problems. We have presented the standard vertex coloring game along with variations in the definition of a legal color. Additionally, we have presented variations in which the players color edges. Each of these variations still have problems to consider, but we have also given a framework which invites researchers to consider their own variations of a legal color.

Questions in competitive graph coloring are ideally suited to undergraduate research. Not only are there many interesting open questions, but students can begin exploring these questions with very little background. For many questions, students need only to be introduced to basic definitions in graph theory, the rules of the game and the relevant parameters, along with strategies for Alice and Bob. Experimentation with game variations and specific classes of graphs can lead students to ask their own questions. We would note that the work in each of papers in [17–20] was done with undergraduates and each paper has undergraduates listed as coauthors.

We summarize some open questions stemming from the games presented above, but we also expect this paper to provide a model for asking and answering additional questions in the area of competitive graph coloring.

Regarding the standard vertex coloring game, a number of problems still remain open.

Research Project 1. Find criteria which differentiates between forests with game chromatic numbers 3 and 4.

Research Project 2. Does there exists a forest F of odd order with $\Delta(F) = 3$ where $\chi_g(F) = 4$?

The game chromatic number has also been studied on other common classes of graphs, and there are many open questions coming from this.

Research Project 3. For planar graphs G, the best known upper [32] and lower [28] bounds show that $8 \leq \chi_g(G) \leq 17$. For outerplanar graphs G, the current bounds are $6 \leq \chi_g(G) \leq 7$ [25]. Progress for either class would be of interest.

Research Project 4. Another interesting question, asked by Zhu [30], is whether $\chi_g(G) = k$ implies that Alice has a winning strategy for the $(k+1)$-coloring game on G.

Regarding the d-relaxed chromatic number ${}^d\chi_g(G)$, Theorem 7 has been shown to be sharp, and it was shown in [26] that if $d \geq 2$ then for any tree T, ${}^d\chi_g(T) \leq 2$. However, similar to the classification problem discussed in Section 2, we do not know criteria to distinguish those trees T for which ${}^1\chi_g(T) = 3$ and those for which ${}^1\chi_g(T) = 2$.

Research Project 5. Find criteria to distinguish trees where ${}^1\chi_g(T) = 3$ from those where ${}^1\chi_g(T) = 2$.

Chou et al. [6] also showed that, where G is an outerplanar graph and $d \geq 1$, ${}^d\chi_g(G) \leq 6$.

Research Project 6. Improve this bound, or provide an example which shows that it is sharp. This has not been done for any d.

Clique-relaxed game chromatic number has not been as widely studied. In particular, there are no current results for planar graphs. For outerplanar graphs G, Theorems 9 and 10 show that $3 \leq {\chi_g}^{(2)}(G) \leq 4$.

Research Project 7. Determine bounds for the clique-relaxed game chromatic number for planar graphs. Once a sharp upper bound is known, the question of classifying graphs with particular clique-relaxed game chromatic number arises.

The edge coloring game also has many open questions. For forests F with maximum degree Δ, it has been shown in [1, 22] that ${\chi_g}'(F) \leq \Delta + 1$ except when $\Delta = 4$, where the best result shows ${\chi_g}'(F) \leq 6$. Lam et al. [29], who showed that ${\chi_g}'(F) \leq \Delta + 2$ for forests, also asked if there could be a constant c such that ${\chi_g}'(G) \leq \Delta + c$ for all graphs.

Research Project 8. Is there a constant c such that ${\chi_g}'(G) \leq \Delta + c$ for all graphs? A similar question can be asked of the d-relaxed edge coloring game, or for the total coloring game.

Finally, in [12] it is shown that for every $m \in \mathbb{N}$, there exists a graph G such that $m \leq \chi_g(G) < {}^1\chi_g(G)$, which seems counterintuitive.

Research Project 9. Is it true that for every nonnegative integer d, there exists a graph G such that ${}^{d}\chi_{\mathrm{g}}(G) < {}^{d+1}\chi_{\mathrm{g}}(G)$? This question was first proposed by Kierstead and would seem an interesting one to settle.

References

1. Andres, S.D.: The game chromatic index of forests of maximum degree $\Delta \geq 5$. Discret. Appl. Math. **154**(9), 1317–1323 (2006)
2. Appel, K., Haken, W., Koch, J.: Every planar map is four colorable. Part II: reducibility. Ill. J. Math. **21**, 491–567 (1977)
3. Bartnicki, T., Grytczuk, J., Kierstead, H.A., Zhu, X.: The map-coloring game. Am. Math. Mon. **114**(9), 793–803 (2007)
4. Bodlaender, H.: On the complexity of some coloring games. In: Möhring, R. (ed.) Graph Theoretical Concepts in Computer Science, vol. 484, pp. 30–40. Lecture notes in Computer Science. Springer, Berlin (1991)
5. Cai, L., Zhu, X.: Game chromatic index of k-degenerate graphs. J. Graph Theory **36**(3), 144–155 (2001)
6. Chou, C., Wang, W., Zhu, X.: Relaxed game chromatic number of graphs. Discret. Math. **262**(1–3), 89–98 (2003)
7. Cohen-Addad, V., Hebdige M., Král', D., Li, Z., Salgado, E.: Steinberg's conjecture is false. Preprint arXiv:1604.05108 (2016)
8. Cowen, L., Cowen, R., Woodall, D.: Defective colorings of graphs in surfaces: partitions into subgraphs of bounded valency. J. Graph Theory **10**, 187–195 (1986)
9. Cowen, L., Goddard, W., Jesurum, C.: Defective coloring revisited. J. Graph Theory **24**, 205–219 (1997)
10. Deuber, W., Zhu, X.: Relaxed coloring of a graph. Graphs Combin. **14**, 121–130 (1998)
11. Dinski, T., Zhu, X.: A bound for the game chromatic number of graphs. Discret. Math. **196**, 109–115 (1999)
12. Dunn, C.: Complete multipartite graphs and the relaxed coloring game. Order **29**(3), 507–512 (2012)
13. Dunn, C.: The relaxed game chromatic index of k-degenerate graphs. Discret. Math. **307**, 1767–1775 (2007)
14. Dunn, C., Kierstead, H.A.: A simple competitive graph coloring algorithm II. J. Combin. Theory Series B, **90**, 93–106 (2004)
15. Dunn, C., Kierstead, H.A.: A simple competitive graph coloring algorithm III. J. Combin. Theory Ser. B **92**, 137–150 (2004)
16. Dunn, C., Kierstead, H.A.: The relaxed game chromatic number of outerplanar graphs. J. Graph Theory **46**, 69–78 (2004)
17. Dunn, C., Naymie, C., Nordstrom, J.F., Pitney, E., Sehorn, W., Suer, C.: Clique-relaxed graph coloring. Involve **4**(2), 127–138 (2011)
18. Dunn, C., Larsen, V., Lindke, K., Retter, T., Toci, D.: The game chromatic number of trees and forests. Discret. Math. Theor. Comput. Sci. **17**(2), 31–48 (2015)
19. Dunn, C., Morawski, D., Nordstrom, J.F.: The relaxed edge-coloring game and k-degenerate graphs. Order **32**(3), 347–361 (2015)
20. Dunn, C., Hays, T., Naftz, L., Nordstrom, J.F., Samelson, E., Vega, J., Total coloring games (in preparation)
21. Eaton, N., Hull, T.: Defective list colorings of planar graphs. Bull. Inst. Combin. Appl. **25**, 79–87 (1999)

22. Erdős, P., Faigle, U., Hochstättler, W., Kern, W.: Note on the game chromatic index of trees. Theor. Comput. Sci. **313**(3), 371–376 (2004)
23. Faigle, U., Kern, W., Kierstead, H.A., Trotter, W.: On the game chromatic number of some classes of graphs. Ars Combinatoria **35**, 143–150 (1993)
24. Gardner, M.: Mathematical games. Sci. Am. **23**, (1981)
25. Guan, D., Xhu, X.: Game chromatic number of outerplanar graphs. J. Graph Theory **30**, 67–70 (1999)
26. He, W., Wu, J., Zhu, X.: Relaxed game chromatic number of trees and outerplanar graphs. Discret. Math. **281**(1–3), 209–219 (2004)
27. Kierstead, H.A.: A simple competitive graph coloring algorithm. J. Combin. Theory Ser. B **78**, 57–68 (2000)
28. Kierstead, H.A., Trotter, W.: Planar graph coloring with an uncooperative partner. J. Graph Theory **18**, 569–584 (1994)
29. Lam, P., Shiu, W., Xu, B.: Edge game-coloring of graphs. Graph Theory Notes N. Y. **XXXVII**, 17–19 (1999)
30. Zhu, X.: The game coloring number of planar graphs. J. Combin. Theory Ser. B **75**, 245–258 (1999)
31. Zhu, X.: The game coloring number of pseudo partial k-trees. Discret. Math. **215**, 245–262 (2000)
32. Zhu, X.: Refined activation strategy for the marking game. J. Combin. Theory Ser. B **98**, 1–18 (2008)

Matroids

Erin McNicholas, Nancy Ann Neudauer, and Colin Starr

Suggested Prerequisites. *Linear Algebra is the primary prerequisite for this material. Some familiarity with Graph Theory and finite fields is also desirable; in particular, we rely on operations in* $\mathbb{Z}_2$ *a great deal.*

1 Introduction

In recent years graph theory has emerged as one of the most popular fields for undergraduate research. It is accessible, beautiful, and a powerful way to model relationships. As in other areas of discrete mathematics, problems that seemed intractable have become approachable with advances in computing power. Graph theory and discrete mathematics more generally have opened up as fruitful areas for research.

Matroids are an abstract generalization of graphs. Given any graph, we can define an associated *cycle matroid*. Because all graphs can be represented as matroids, we may use the more abstract setting of matroids to draw conclusions about graphs. The converse, however, is not true; not all matroids are the cycle matroid of some graph. Thus, in working with matroids we must guard against blindly treating them like graphs.

Matroids also incorporate a notion of *independence* that generalizes linear independence, so we can consider matroids a generalization of linear algebra; again,

E. McNicholas (✉) • C. Starr
Willamette University, 900 State St., Salem, OR 97306, USA
e-mail: emcnicho@willamette.edu; cstarr@willamette.edu

N. Ann Neudauer
Pacific University, 2043 College Way, Forest Grove, OR 97116, USA
e-mail: nancy@pacificu.edu

A. Wootton et al. (eds.), *A Primer for Undergraduate Research*, Foundations for Undergraduate Research in Mathematics, DOI 10.1007/978-3-319-66065-3_6

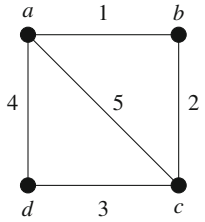


Fig. 1 A Graph

however, not all matroids can be represented this way, either. For those that can, we can draw on the tools and techniques of Linear Algebra to further our cause.

Tutte said, "If a theorem about graphs can be expressed in terms of edges and circuits only, it probably exemplifies a more general theorem about matroids." This has driven work in matroid theory and is the approach we take in this chapter to extend the work on the problem of uniquely pancyclic graphs to matroids. In turn, results about the matroids may solve open problems for the graphs.

2 Graphs

We start with a brief review of several common terms from Graph Theory. For a more detailed introduction to the subject see [12] or [13]. A **graph** G is defined by a set of vertices $V(G) = \{v_1, \ldots, v_n\}$ and a list of pairs of vertices called the **edges** $E(G)$ of G. Given an edge pair $e = \{v_i, v_j\}$, we say the vertices v_i and v_j are **adjacent** and **joined** by the edge e; a vertex is **incident** with an edge it belongs to. For example, take $V(G) = \{a, b, c, d\}$ and $E(G) = \{\{a, b\}, \{b, c\}, \{c, d\}, \{d, a\}, \{a, c\}\}$. This graph is illustrated in Figure 1. Vertices a and b are adjacent, but vertices b and d are not. Vertex a is incident with edge 5, but vertex b is not. For small graphs, it is usually easier to draw them than to describe them as a set of vertices and an edge list.

For the research projects we discuss in this chapter, we need a few additional terms and concepts associated with graphs.

First, a **cycle** in a graph is a sequence of adjacent vertices $v_1, v_2, \ldots, v_k, v_1$ such that $v_i \neq v_j$ unless $i = j$. Alternatively, we can describe a cycle by the edges $v_1v_2, v_2v_3, \ldots, v_{k-1}v_k, v_kv_1$, with the same restriction $v_i \neq v_j$ unless $i = j$. (Note that we abbreviated the edge $\{v_i, v_j\}$ to v_iv_j; this is common notation.) The **length** of a cycle is the number of edges appearing in it. In our previous example, using the edge description of a cycle, G has cycles 125, 534, and 1234 of lengths 3, 3, and 4, respectively. **Paths** are defined similarly except that the last vertex does not match the first: $v_1v_2, v_2v_3, \ldots, v_{k-1}v_k$.

A graph G is **simple** if the edges $E(G)$ are distinct and each edge consists of two distinct vertices (that is, there is at most one edge between any two vertices, and there are no loops connecting a vertex to itself). A graph is **planar** if it can be embedded in the plane, or equivalently, it can be drawn in such a way that the edges only meet at vertices. All the graphs presented in this chapter are simple and planar. While graphs (e), (f), and (g) in Figure 2 are drawn with intersecting edges, edge a could be drawn above edges 1 and 2 to demonstrate the planarity of these graphs.

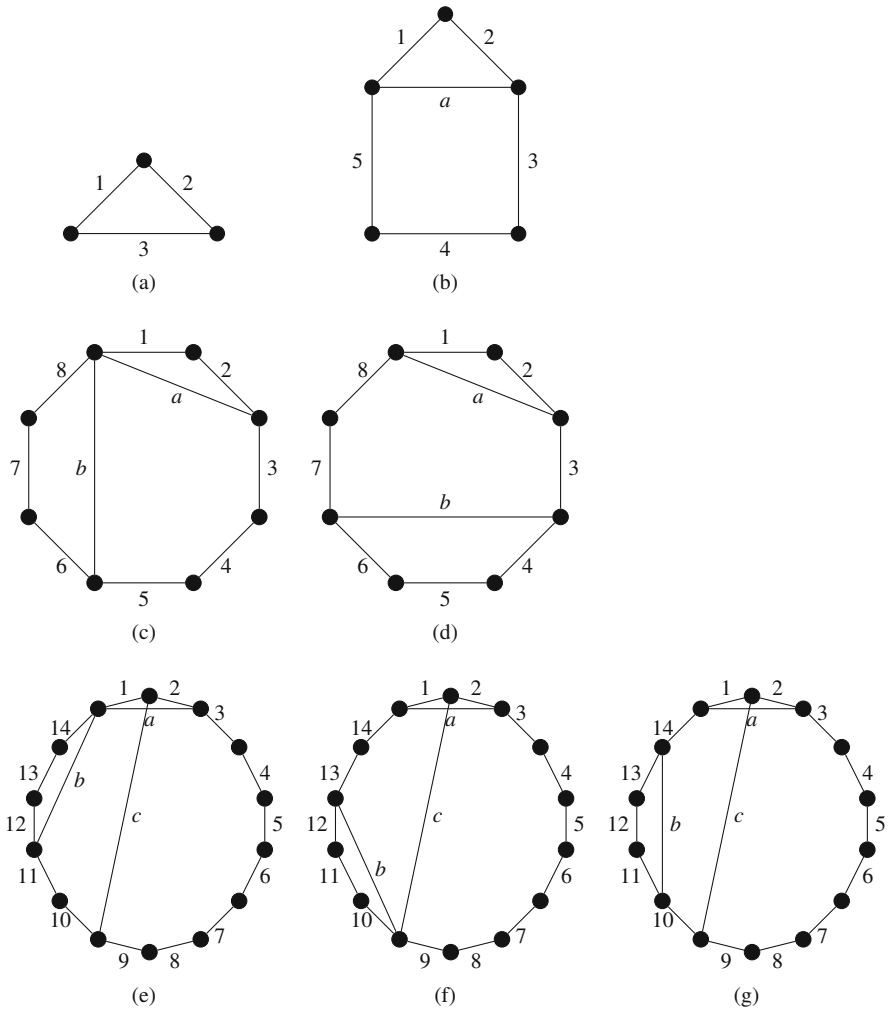


Fig. 2 The known UPC graphs

2.1 UPC Graphs

Graph Theory is rich with interesting open problems. In this chapter, we consider the following problem posed by Entringer in 1973:

Question. Which simple graphs on n vertices have exactly one cycle of size ℓ for each ℓ such that $3 \leq \ell \leq n$?

Following Bondy[1], we call a simple graph G on n vertices **uniquely pancyclic**, or **unipancylic**, if it has exactly one cycle of each possible length, $3, \ldots, n$. We usually abbreviate unipancyclic to UPC. The graphs in Figure 2 are the only known UPC graphs on up to 59 vertices [6, 10, 11]; for example, the 8-vertex graph in Figure 2(c) has cycles $12a, 678b, 345ba$, $12345b, 345678a$, and 12345678, and those are its only cycles.

Exercise 1. Verify that the graphs shown in Figure 2 are all UPC.

It is not known whether there are any more UPC graphs beyond those shown in Figure 2.

Research Project 1. Find another UPC graph not shown in Figure 2 or show that there are no more.

A UPC graph is necessarily **Hamiltonian**; that is, it contains a cycle that passes through every vertex. Given a simple graph with a unique Hamiltonian cycle H, the edges that are not part of H are called **chords** of the graph.

In 1986, Shi proved that there are no UPC graphs having exactly 4 chords [10] and that the UPC graphs in Figure 2 are the only ones with three or fewer chords. Markström [6] proved in 2009 that there are no other UPC graphs on 59 or fewer vertices and that there are no UPC graphs with exactly 5 chords. Little other progress has been made on this problem since it was first posed in 1973.

Some possible avenues for progress include determining structural characteristics of UPC graphs. Project 3 requires ideas in Graph Theory that we have not introduced. See [12] for a general discussion of graph theoretic concepts.

Research Project 2. Determine whether a UPC graph must be planar. (The known ones are.)

Research Project 3. Determine the possible chromatic numbers of UPC graphs.

In 2001, Starr and Turner generalized the notion of UPC graphs to UPC matroids (unpublished work). Below, we discuss this generalization and the interesting discovery of a non-graphic UPC matroid on 4 chords.

3 Matroids

3.1 Introduction to Matroids

In 1935, Whitney published the seminal paper on matroid theory, generalizing familiar notions of *dependence* and *independence* that arise in several areas of mathematics [4]. Since then, the field has risen in importance due to its generality and its wide application to other branches of mathematics.

Matroids are an axiomatically-defined structure (as we will see in the definition below) abstracting notions of algebraic dependence, linear dependence, and geometric dependence (e.g., collinearity of points) that unify several areas of discrete mathematics. Their abstract properties can sometimes be represented more concretely, like with a graph or a geometry. The usefulness of matroids to pure mathematical research is similar to that of groups: by studying an abstract version of phenomena that occur in different realms of mathematics, we learn something about all those realms simultaneously. Matroids are the objects we study in order to understand dependence properties of discrete sets of points in space. There are also applications of matroid theory to practical problems, and matroids are essential in combinatorial optimization. Rota said that "Anyone who has worked with matroids has come away with the conviction that matroids are one of the richest and most useful ideas of our day." [9]

Definition 1. Let E be a finite set of elements, and let $\mathscr{I}$ be a family of subsets of E. We say the ordered pair $M = (E, \mathscr{I})$ is a **matroid** if $\mathscr{I}$ has the following properties:

1. $\emptyset \in \mathscr{I}$;
2. If $I_1 \in \mathscr{I}$ and $I_2 \subseteq I_1$, then $I_2 \in \mathscr{I}$;
3. If $I_1, I_2 \in \mathscr{I}$ and $|I_1| < |I_2|$, then there exists $e \in I_2 - I_1$ such that $I_1 \cup e \in \mathscr{I}$.

Property 3 is known as the **independence augmentation axiom**.

The set E is referred to as the **ground set** of the matroid and $\mathscr{I}$ as the set of **independent sets** [8]. We say that a set is **dependent** if it is not independent. A minimally dependent set (i.e., a dependent set such that the removal of *any* element from it renders it independent) is called a **circuit**.

These structures in the matroid capture the essential notion of *independence* as we encounter it in other contexts. Note that we adopt the common convention from Graph Theory that if e is a single element, then $A \cup e$ abbreviates $A \cup \{e\}$.

For a general introduction to Matroid Theory see [7] or [8].

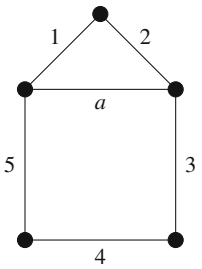


Fig. 3 A Graphic Matroid

3.2 The Cycle and Vector Matroids

You might ask how graphs can be matroids. Or how matroids are related to Linear Algebra. There are several ways to define a matroid on a graph; the most common we describe here. Below, we show how you can define a matroid on a set of vectors.

Let G be a graph. Let $E = E(G)$ be the set of edges of G, and let $\mathscr{I}$ be the set of all subsets of E that do not contain a cycle in the graph. Is this a matroid? To check that this is in fact a matroid we merely need to check the three properties from Definition 1. Certainly the empty set does not contain a cycle, so $\mathscr{I}$ satisfies Property 1 of the definition. Any subset of a cycle-free subgraph of G is also cycle-free, giving us Property 2. Property 3 is much harder to prove, but it does hold. We defer further discussion until we develop an alternate definition of a matroid based on circuits. Since it does satisfy the three axioms, we can define a matroid in this way starting with any graph. This matroid is called the **cycle matroid** of the graph, denoted $M(G)$. Any matroid that is the cycle matroid of some graph is a **graphic matroid**, or **graphic**. Recall from our earlier discussion that not all matroids are graphic.

Example 1. Given the graph in Figure 3, we can define the cycle matroid as follows: the ground set of the cycle matroid on G is {1,2,3,4,5,a}; the maximal independent sets, called the **bases** of the matroid, are $\{1,2,3,4\}$, $\{1,2,3,5\}$, $\{1,2,4,5\}$, $\{1,3,4,5\}$, $\{2,3,4,5\}$, $\{1,a,3,4\}$, $\{1,a,3,5\}$, $\{1,a,4,5\}$, $\{2,a,3,4\}$, $\{2,a,3,5\}$, and $\{2,a,4,5\}$. $\mathscr{I}$ is the set of all subsets of these sets. The circuits of this matroid are $\{1,2,3,4,5\}$, $\{1,2,a\}$, and $\{3,4,5,a\}$. As in all cycle matroids, the circuits are precisely the cycles in the corresponding graph.

We see here how matroids abstract graphs, but what about their connection to linear algebra?

We can define a matroid on any collection of vectors. Suppose that A is an $m \times n$ matrix. To define a matroid, we need to decide what the ground set and the independent sets are, and then to check that these satisfy the axioms. Let E be the set of columns of A, and let $\mathscr{I}$ be the set of all linearly independent subsets of E. Since the empty set is linearly independent, $\mathscr{I}$ satisfies Property 1 of the definition. Every subset of a linearly independent set of vectors is also linearly independent, so $\mathscr{I}$ satisfies Property 2, as well. Finally, we know that if $I_1, I_2 \in \mathscr{I}$ with $|I_1| < |I_2|$,

then I_1 spans a subspace of smaller dimension than does I_2, so I_2 contains a vector e not in the span of I_1. Therefore, $I_1 \cup e$ is also linearly independent, and $\mathscr{I}$ satisfies Property 3.

A matroid that arises in this way is called a **vector matroid**. A matroid is **representable** over a field F if it can be expressed as the vector matroid of some matrix with entries in F. The entries could be real numbers or elements of $\mathbb{Z}_2$, for example.

Example 2. As a specific example of a vector matroid, consider the matrix

$$\begin{array}{c} 1\ 2\ 3\ 4\ \ 5\ a \\ \left[\begin{array}{cccc|cc} 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{array}\right] \end{array}$$

over $\mathbb{Z}_2$. That means that when we consider linear combinations, the coefficients are either 0 or 1 and any arithmetic is performed modulo 2. Thus each linear combination can be thought of a subset of the set of columns – each column either appears or it doesn't. Second, note that the rank of the matrix is 4 – that is, it takes four linearly independent column vectors to create a basis for the column space, which we are now thinking of as a basis of the matroid. We can see that columns 1 through 4 are linearly independent, while columns 1 through 5 are linearly dependent. Since removing any one of columns 1 through 5 leaves a linearly independent set, columns $\{1, 2, 3, 4, 5\}$ are *minimally* linearly dependent and hence a circuit of this matroid. Similarly, $\{1, 2, a\}$ and $\{3, 4, 5, a\}$ are also circuits, and these three circuits are the only circuits of the matroid.

Having identified the collection of circuits, we see that this vector matroid is isomorphic to the cycle matroid of the graph in Figure 3!

What do we mean by isomorphic? Two matroids are **isomorphic** if there is a bijection between their ground sets that also induces a bijection between their sets of independent sets (or equivalently their circuits). Essentially, one is just a relabeling of another.

It turns out that every graphic matroid is representable over $\mathbb{Z}_2$. Recall that this means we can find a collection of vectors over $\mathbb{Z}_2$ so that the graphic matroid is the vector matroid on those vectors. We refer to any matroid representable over $\mathbb{Z}_2$ as **binary**. In fact, graphic matroids are representable over *every* field. See [8], for example, for a proof. A matroid representable over every field is called **regular**. As you may suspect, there are both binary matroids and regular matroids that are not graphic.

The previous example justifies the language of independence for matroids from its use in linear algebra. However, there are other notions of independence that are also included in this definition, as we saw in the cycle matroid of Example 1. In that case, independence refers to an absence of cycles in a subset of edges of the graph.

3.3 Beyond the Basics

Notice that in the examples above, the bases of each matroid were all the same size. This is not a coincidence! It is always the case that for a given matroid the bases all have the same size. The size of a basis is called the **rank** of the matroid, by analogy with the rank of a matrix. We can, in fact, define the rank of any subset $S \subseteq E$ to be the size of a largest possible independent set contained in S. Instead of defining a matroid in terms of its independent sets, a matroid can also be defined by its set of bases or by a rank function; see Oxley (for example) for further details [8].

The **closure** of a set D of elements is the set $Cl(D)$ consisting of D and all elements that can be added to D without changing the rank. For the cycle matroid in Figure 3, the closure of $\{1, 2, 3\}$ is $\{1, 2, 3, a\}$ since a completes the cycle $\{1, 2, a\}$ but does not increase the size of the largest contained independent set, and therefore the rank of $\{1, 2, 3\}$ and $\{1, 2, 3, a\}$ are the same (namely, 3).

In Example 2, the columns with these labels exhibit the same properties: columns $1, 2, 3$, and a are linearly dependent over $\mathbb{Z}_2$, and they span a subspace of ${\mathbb{Z}_2}^4$ of dimension 3 – the same subspace spanned by just columns $1, 2$, and 3. The closure of the set of columns $\{1, 2, 3\}$ is thus $\{1, 2, 3, a\}$.

The previous examples illustrate matroids in the familiar settings of Graph Theory and Linear Algebra. You may be asking yourself whether there are any examples of matroids that are not graphic or representable over $\mathbb{Z}_2$. We consider one now.

Example 3. Let $E = \{1, 2, 3, 4\}$, and let $\mathscr{I} = \{S \subseteq E : |S| \leq 2\}$. In other words, *all* sets of size two or smaller are independent sets. Then $(E, \mathscr{I})$ is a kind of matroid known as a **uniform matroid**. This particular matroid is known as $U_{2,4}$. More generally, $U_{k,n} = (E, \mathscr{I})$ refers to the matroid with $|E| = n$ and $\mathscr{I} = \{S \subseteq E : |S| \leq k\}$.

Is $U_{2,4}$ really a new type of matroid? It is natural to wonder whether there is a graphic or vector representation that captures the same dependencies on four elements. That is the kind of question that stimulates mathematical research! As we see below (and you can show), $U_{2,4}$ is not graphic, but it does have a vector representation over fields of order 3 or more.

Exercise 2. Verify that the three independence axioms hold for $U_{2,4}$.

Exercise 3. Show that $U_{2,4}$ is not graphic.

Exercise 4. Verify that $U_{2,4}$ is not representable over $\mathbb{Z}_2$ and find a representation of $U_{2,4}$ over $\mathbb{Z}_3$.

There are many equivalent ways of defining matroids. As we mentioned before, we could define them in terms of the bases or the rank function instead of the independent sets. For our purposes, the most useful alternative to the independent sets definition is in terms of the circuits.

Definition 2. The ordered pair $M = (E, \mathscr{C})$ is called a **matroid** if E is a finite set, and $\mathscr{C}$ is a family of subsets of E satisfying:

1. $\emptyset \notin \mathscr{C}$;
2. if $C \in \mathscr{C}$, then no proper subset of C is in $\mathscr{C}$;
3. if $C_1, C_2 \in \mathscr{C}$ and $e \in C_1 \cap C_2$, then there exists $C_3 \in \mathscr{C}$ such that $C_3 \subseteq C_1 \cup C_2 - \{e\}$.

Property 3 is known as the **circuit elimination axiom**.

We can show that the two definitions of a matroid are equivalent, or, more precisely, cryptomorphic [3]. Let $\mathscr{I}$ be the set of all subsets of E that do not contain (as a subset) any member of $\mathscr{C}$.

Challenge Problem 1. Given a set $\mathscr{C}$ satisfying the properties in Definition 2, show that the set $\mathscr{I}$ constructed as above fulfills the properties and axioms of Definition 1, and that the set of circuits of $(E, \mathscr{I})$ is precisely $\mathscr{C}$. Conversely, show that if $M = (E, \mathscr{I})$ is a matroid according to Definition 1, then its set $\mathscr{C}$ of circuits satisfies the properties above and the set of all subsets of E that do not contain any member of $\mathscr{C}$ is precisely the given set $\mathscr{I}$.

Example 4. Graphic matroids are often naturally examined in light of the circuit axioms above: rather than show that graphs satisfy the independence axioms, we show they satisfy the circuit axioms. The set $\mathscr{C}$ of cycles of a graph certainly does not include the empty set, and no proper subset of a cycle is also a cycle. We offer an informal argument that $\mathscr{C}$ satisfies the circuit elimination axiom: suppose that e belongs to two cycles C_1 and C_2. Say $e = uv$. Travel from u toward v along C_1 (not going directly from u to v). Along the way, pay attention to any vertices shared by C_1 and C_2. Similarly, travel from u toward v along C_2. Since C_1 and C_2 both include paths from u to v that avoid e, there exist vertices a and b such that C_1 and C_2 stop sharing vertices at a and start sharing vertices again at b. Then traveling from a to b along C_1 and from b back to a along C_2 gives a cycle in $C_1 \cup C_2 - e$.

Exercise 5. Consider the graphic matroid given in Figure 2(e), and the circuits C_1 and C_2 containing edges $\{1, b, 11, 10, c\}$ and $\{1, 2, a\}$, respectively. Find the circuit C_3 whose existence is guaranteed by the circuit elimination axiom.

Each matroid has a **dual** defined on the same ground set. The bases of the dual of a matroid M on E are precisely the complements of the bases of M. If you are familiar with Graph Theory, you may wonder whether the notions of duality for cycle matroids and graphs coincide. (If you are less familiar, see [12] or [13] for an exploration of these ideas.) In fact, they do: if M is the cycle matroid of a planar graph G, the dual of M is isomorphic to the cycle matroid of the dual of G. The dual of the cycle matroid of G is called the **cutset matroid** of G due to the fact that the cycles of the graph become the cutsets in the dual. To learn more about this see [13].

Notice that the matroid $U_{2,4}$ in Example 3 is its own dual. Its bases are all subsets of size 2 (the maximal independent sets), and, since the ground set has 4 elements, their complements are also the subsets of size 2.

Exercise 6. Find the dual of $U_{2,5}$.

We can operate on matroids by deleting or contracting elements or sets of elements. If M is a matroid and $e \in E(M)$, the **deletion** of e from M is the matroid $M \setminus e$ on $E(M) - \{e\}$ whose circuits are the circuits of M not involving e. For example, $U_{2,4} \setminus 4$ gives the matroid on $\{1, 2, 3\}$ with just the one circuit $\{1, 2, 3\}$.

The **contraction** of e in M is the matroid M/e on $E(M) - \{e\}$ whose circuits are the minimal[1] non-empty members of $\{C - e | C$ is a circuit of $M\}$. Contracting 4 in $U_{2,4}$, for example, changes the set of circuits of M into the set $\{\{1, 2, 3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}\}$. Since $\{1, 2, 3\}$ contains the set $\{1, 2\}$, it is not minimal in this collection and thus is not a circuit of $M/4$, but the other three are. That is, $U_{2,4}/4$ is $U_{1,3}$.

Interestingly, the operations of deletion and contraction commute with themselves and with each other, so if S is a set of elements to delete and T is a set of elements to contract, the order in which they are deleted and contracted does not matter. The result is the following.

Definition 3. A **minor** of a matroid M is a matroid obtained by a sequence of deletions and contractions; that is, a matroid of the form $M/T \setminus S$ for some $S, T \subseteq E(M)$.

4 In Search of UPC Matroids

4.1 UPC Matroids

We generalize the notion of a UPC graph to that of a UPC matroid: a matroid M of rank r is UPC if it has exactly one circuit each of sizes $3, 4, \ldots, r + 1$. We will refer to the $(r + 1)$-circuit as the **Hamiltonian circuit** by analogy with graphs.

Note that all of the UPC graphs above can also be viewed as UPC matroids. Given the difficulty in finding another UPC graph, Starr and Turner expanded the question: Are there any non-graphic UPC matroids? Indeed, there is at least one. The following is the rank-24, non-graphic, binary UPC matroid M_{24} found by Starr and Turner in 2001:

[1] in the set-inclusion sense

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	a	b	c	d
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0

There are many ways to see that this is non-graphic; the simplest is probably to try to find a graph it represents.

Note that columns 1 through 25 give the Hamiltonian circuit, and columns a, b, c, and d are the chords. Deleting[2] the chord d and contracting[3] elements 14 through 24 shows that the UPC graph on 14 vertices is a minor of this UPC matroid.

In fact, all of the known UPC graphs are minors of the rank-24 UPC matroid: contracting columns 3 through 24 and deleting a, b, c, and d gives the triangle; contracting 6 through 24 and deleting b, c, and d gives the 5-vertex UPC graph; contracting 8 through 24 and deleting c and d gives *both* 8-vertex UPC graphs depending on how we choose to label the graph edges; and contracting 14 through 24 and deleting d gives all three 14-vertex UPC graphs, again depending on where we place different edges. For example, when we delete d and contract 14 through 24, we are left with a 13-cycle. Since there is no requirement that we label its edges in numerical order, this leaves some choice for the placement of chords.

[2]To delete an edge from a matroid represented as a matrix, simply delete the column corresponding to that element.

[3]To contract an edge from a matroid represented as a matrix, first pivot on a given element so that it is (a) 1 and (b) the only nonzero element in its column, then delete the row and column of that entry. See [8] for further details.

Exercise 7. Find the rank-7 minor of M_{24} described above and then find two non-isomorphic graph realizations of it based on the matrix labeling. Compare to Figure 2.

Interestingly, although there are multiple non-isomorphic UPC graphs on 8 and 14 vertices, the corresponding UPC matroids of a given rank are unique. That is, there is only one rank-13 UPC matroid even though it has three non-isomorphic graph realizations. It is not clear whether this holds in general. Part of the difficulty is that no other UPC matroids are known at the time of this writing. Perhaps you can find one!

Research Project 4. Determine whether two UPC matroids of the same rank must be isomorphic.

Research Project 5. Find another UPC matroid.

Research Project 6. We know that the triangle (0 chords) is a UPC minor of every UPC matroid, and, in fact, each of the known UPC matroids is a minor of all of the higher-rank UPC matroids known. Does this continue for larger UPC matroids? For example, must a UPC matroid on k chords contain a UPC minor on $k-1$ chords? $k-2$?

4.2 Binary Matroids

We turn our attention to binary matroids. Recall that a matroid is **binary** if it is representable over $\mathbb{Z}_2$. There are at least two reasons for looking at binary matroids in particular: (1) every graphic matroid is also binary, so understanding binary UPC matroids can help us understand UPC graphs; (2) binary matroids are relatively easy to think about since the matrix representation is made up entirely of 0s and 1s.

The following definitions and theorems demonstrate the ease of working with binary matroids. Several graph-theoretic definitions extend directly to binary matroids, and identifying circuits is much easier than in representable matroids in general.

Definition 4. A matroid M is **Hamiltonian** if it has a circuit of size $r+1$, where r is the rank of M.

In this definition, we follow Borowiecki [2] rather than Lalithambal and Sridharan [5], who have a slightly different definition of Hamiltonian matroid (although the definitions do coincide in the case of binary matroids).

Definition 5. In a matroid, a **chord** of a circuit C is an element of $Cl(C) - C$. If the matroid is Hamiltonian with a unique Hamiltonian circuit H, we will reserve the word chord to refer to a chord of H unless specified otherwise.

Definition 6. The **support** of column $\mathbf{a}$, denoted $S(\mathbf{a})$, in a binary matrix M is the set of row indices labeling where $\mathbf{a}$ has a non-zero entry. The **weight** of column $\mathbf{a}$, denoted $wt(\mathbf{a})$, is the cardinality of $S(\mathbf{a})$.

The following definition leads to a way of determining when a set of elements generates a circuit in a binary matroid.

Definition 7. Let $\mathscr{K}$ be a collection of vectors. Then $\mathscr{K}$ is *covered* if

$$\left(\sum_{v\in P_1} v\right)\cdot\left(\sum_{v\in P_2} v\right)\neq 0$$

for every partition $\{P_1, P_2\}$ of $\mathscr{K}$ into two sets. Here, the dot refers to the usual dot product over the reals, while the vector sums are calculated over $\mathbb{Z}_2$.

Example 5. For example, consider the following matrix over $\mathbb{Z}_2$:

$$\begin{array}{c} \begin{array}{cccc|cccc} 1 & 2 & 3 & 4 & 5 & a & b & c \end{array} \\ \left[\begin{array}{cccc|cccc} 1&0&0&0&1&1&0&0\\ 0&1&0&0&1&1&1&0\\ 0&0&1&0&1&0&1&1\\ 0&0&0&1&1&0&0&1 \end{array}\right] \end{array}$$

We see that a and b are covered since $a\cdot b = 1$, while a and c are not covered since $a\cdot c = 0$. The set $\{a,b,c\}$ is covered since $(a+b)\cdot c = 1, (a+c)\cdot b = 2$, and $(b+c)\cdot a = 1$.

Notice our vector matroid examples have been in the form $[I_r|D]$, where I_r is the $r\times r$ identity matrix and the matroid M is of rank r. All representable matroids can be expressed in this form, a fact we will use frequently; see [8] for a proof.

Theorem 1. *Let M be a binary matroid represented by $[I_r|D]$, and let $A = \{f_1,\ldots,f_\ell\}$ be a set of columns from D. Let I index the columns from I_r such that $\sum_{i\in I} e_i = \sum_{i=1}^{l} f_i$. Then $A\cup\{e_i|i\in I\}$ is a circuit if and only if A is covered. If A is covered, we refer to the circuit $A\cup\{e_i|i\in I\}$ as the circuit **generated** by A.*

The following chain of equivalences includes two important characterizations of binary matroids. See [8] for a proof.

Theorem 2. *Given a matroid M, the following are equivalent:*

1. *M is binary*
2. *If C_1 and C_2 are distinct circuits, their* symmetric difference $C_1 \triangle C_2 = (C_1 - C_2) \cup (C_2 - C_1)$ *is a disjoint union of circuits*
3. *M has no minor isomorphic to $U_{2,4}$*

Thus, for example, there is no way to represent $U_{2,4}$ as a matrix over $\mathbb{Z}_2$ in the way we did for the graph in Example 1, as you showed in an earlier exercise.

4.3 Binary UPC Matroids

Given the difficulty of working with matroids in general, the ease of working with binary matroids, and the success of Starr and Turner in discovering a non-graphic binary UPC matroid, it may be worth investigating binary UPC matroids in particular. Should we restrict ourselves to considering only binary matroids in seeking a UPC matroid, or is it worth looking elsewhere? This brings us to our next research project.

Research Project 7. Determine whether a UPC matroid can have a minor isomorphic to $U_{2,4}$. That is, must all UPC matroids be binary?

Consider a binary Hamiltonian matroid M of rank r with a Hamiltonian circuit H. We know that we can represent M in the form $[I_r|D]$, but the presence of H allows us to refine this a little. We can label r elements of H with the columns of I_r and the last with a column of all ones, $\mathbf{1}$. Then M has the binary representation $[I_r|\mathbf{1}K]$ where the columns of I_r and $\mathbf{1}$ form H, and the columns of K are the chords of M. The following illustrate some of the results we find when we limit our scope to binary matroids. If $x_1, \ldots, x_k$ form a *covered* subset of $\{\mathbf{1}\} \cup K$, we will use the notation $< x_1, \ldots, x_k >$ to denote the circuit generated by $x_1, \ldots, x_k$. While the circuit generated by a covered subset of $\{\mathbf{1}\} \cup K$ is unique in the binary case, this is not necessarily the case for representable matroids over other fields.

Lemma 1. *Let M be a binary Hamiltonian matroid represented by $[I_r|\mathbf{1}K]$. Then there are exactly two* 1*-chord circuits for each chord* $\mathbf{k}$ *in K.*

Proof. Let H be the Hamiltonian circuit represented by $[I_r|\mathbf{1}]$. For each $\mathbf{k}$ in K there are certainly two circuits: one that uses $\mathbf{1}$ and one that does not. By Corollary 9.3.5 of Oxley [8], for any edge $e \in E$, there are at most two circuits contained in $H \cup e$.

Lemma 2. *In a binary Hamiltonian matroid M, every set of two chords creates at least one circuit.*

Proof. Given a Hamiltonian matroid M with binary representation $[I_r|\mathbf{1}K]$, let $\mathbf{b}$ and $\mathbf{c}$ be two distinct chords in K. If $\{\mathbf{b}, \mathbf{c}\}$ is covered, it generates a two-chord circuit. If it is not covered there must be some row i where $\mathbf{b}$ has a 1 and $\mathbf{c}$ has a 0 and some row j where $\mathbf{b}$ has a 0 and $\mathbf{c}$ has a 1. Consider the set $\{\mathbf{1}, \mathbf{b}, \mathbf{c}\}$. These i^{th} and j^{th} rows ensure that $(\mathbf{1}+\mathbf{c})\cdot\mathbf{b} \neq 0$, $(\mathbf{1}+\mathbf{b})\cdot\mathbf{c} \neq 0$, and $(\mathbf{b}+\mathbf{c})\cdot\mathbf{1} \neq 0$. Thus, the set $\{\mathbf{1}, \mathbf{b}, \mathbf{c}\}$ is covered. In either case, the chords $\mathbf{b}$ and $\mathbf{c}$ induce a circuit.

These results lead to the following bound on the number of chords.

Theorem 3. *The number of chords k in a binary UPC matroid of rank r is bounded by*

$$k \leq \frac{\sqrt{8r-7}}{2} - \frac{3}{2}.$$

Proof. Consider a binary UPC matroid M of rank r with k chords. The total number of circuits in M is $r-1$. By Lemmas 1 and 2, there are exactly $2k$ single-chord circuits and at least $\binom{k}{2}$ two-chord circuits in addition to the Hamiltonian circuit. Thus $1 + 2k + \frac{k(k-1)}{2} \leq r - 1$. Using the quadratic formula to simplify, we find $k \leq \frac{\sqrt{8r-7}}{2} - \frac{3}{2}$.

Markström has a refinement of this that takes into account 3-chord cycles for UPC graphs [6].

Note that Theorem 3 gives a lower bound for the rank in terms of the number of chords. Can we find an upper bound?

Research Project 8. Find an upper bound on the rank r in terms of the number of chords k in a binary UPC matroid.

Shi considers particular drawings of UPC graphs to make arguments about their properties by setting the vertices on a circle and drawing chords of the graph as chords of the circle, so the chords often cross. Using structural properties about the number of crossing chords in the 3-cycles and 4-cycles of UPC graphs, Shi has shown that the known UPC graphs with three chords are the only ones, and that there are no UPC graphs with exactly four chords [10]. Can we find analogous results for binary UPC matroids? Recall the *support* of a column $\mathbf{d}$ in a binary matrix

M, denoted $S(\mathbf{d})$, refers to the set of row indices labeling where $\mathbf{d}$ has a non-zero entry. We can use this idea of support to define the chordal relationship analogous to *crossing* in the drawings we have of UPC graphs.

Definition 8. Two chords of a binary matroid M are *skew* if the intersection of their supports is non-empty and they both contain support elements not in the intersection.

Note that if two chords $\mathbf{b}$ and $\mathbf{c}$ are skew, then they generate *two* circuits: $\{\mathbf{b}, \mathbf{c}\}$ and $\{\mathbf{1}, \mathbf{b}, \mathbf{c}\}$ are both covered in this case (following the reasoning in Lemma 2).

The following are a few structural results for binary UPC matroids. While the results themselves may be useful, the proofs also demonstrate how one might study the structural properties of binary UPC matroid circuits.

Lemma 3. *If M is a binary UPC matroid with chord $\mathbf{b}$ having weight equal to 3 (i.e., $\mathbf{b}$ induces a 1-chord 4-circuit), then $\mathbf{b}$ is not skew to other chords in M.*

Proof. By way of contradiction, suppose $\mathbf{b}$ and $\mathbf{c}$ are chords in a UPC Matroid M such that $wt(\mathbf{b}) = 3$ and $\mathbf{b}$ is skew to $\mathbf{c}$. There exists a labeling of the Hamiltonian circuit such that the matroid is representable as follows with chord $\mathbf{c}$ having either form $\mathbf{c_1}$ or $\mathbf{c_2}$.

$$\begin{pmatrix} & \mathbf{b} & \mathbf{c_1} & \mathbf{c_2} & \\ & 1 & 1 & 1 & \\ & 1 & 0 & 1 & \\ & 1 & 0 & 0 & \\ & 0 & 1 & & \\ I_r\ \mathbf{1} & \vdots & 1 & x & \cdots \\ & 0 & 1 & & \\ & 0 & 0 & 0 & \\ & \vdots & \vdots & \vdots & \\ & 0 & 0 & 0 & \end{pmatrix}$$

Recall the circuit $C =< \mathbf{x}, \mathbf{y}, \mathbf{z} >$ generated by $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$ (where $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$ are chords or $\mathbf{1}$) is the circuit consisting of these elements as well as any necessary elements of the Hamiltonian circuit. If chord $\mathbf{c}$ has the form $\mathbf{c_1}$, then circuits $< \mathbf{1}, \mathbf{c} >$ and $< \mathbf{1}, \mathbf{b}, \mathbf{c} >$ both have length $r - x + 1$, and M is not UPC. If $\mathbf{c}$ has the form $\mathbf{c_2}$ then circuits $< \mathbf{c} >$ and $< \mathbf{b}, \mathbf{c} >$ both have length $x + 3$, and M is not UPC. Since these are the only ways for a chord $\mathbf{c}$ to be skew to the chord of weight 3, the lemma follows.

Note that the pancyclicity of M was not used in Lemma 3. Similar techniques prove the more general result:

Lemma 4. *If M is a binary UPC matroid, the chords in the 4-circuit of M can not be skew to each other.*

This gives us a nice structural result on binary UPC matroids:

Corollary 1. *A binary matroid consisting entirely of a Hamiltonian circuit and two skew chords is not UPC.*

Proof. Suppose M is a binary matroid consisting of a Hamiltonian circuit and two skew chords. It follows that M has exactly 7 circuits, and thus the rank must be 8. By Lemma 3, it follows that the 4-circuit must involve the 2 skew chords. Thus, by Lemma 4, M is not UPC.

Research Project 9. What other structural requirements can you find for binary UPC Matroids?

Research Project 10. Use structural requirements to find or to argue against the existence of binary UPC matroids with five or more chords.

5 Further Reading

Although the literature on this problem is sparse, there are a few resources for further reading that we have found helpful.

1. Cornuejols, G., Guenin, C.: Ideal binary clutters, connectivity, and a conjecture of Seymour. SIAM J. Discret. Math. **15**(3), 329–352 (2002)
2. George, J.C., Khodkar, A., Wallis, W.D.: Pancyclic and Bipancyclic Graphs, pp. 49–67. Springer, New York (2016)
3. Oxley, J., Whittle, G.: A note on the non-spanning circuits of a matroid. Eur. J. Comb. **12**, 259–261 (1999)
4. Piotrowski, W.: Chordal characterizations of graphic matroids. Discret. Math. **68**, 273–279 (1988)
5. Wild, M.: Axiomatizing simple binary matroids by their closed circuits. Appl. Math. Lett. **6**(6), 39–40 (1993)

References

1. Bondy, J.A., Murty, U.S.R.: Graph Theory with Applications. North-Holland, Amsterdam (1976)
2. Borowiecki, M., On Hamiltonian matroids. In: Graph Theory: Proceedings of a Conference, held in Lagow, February 10–13 (1981). Lecture Notes in Mathematics, vol. 1018. Springer, Berlin (2006)
3. Gordon, G., McNulty, J.: MATROIDS: A Geometric Introduction. Cambridge University Press, Cambridge (2012)
4. Hassler, W.: On the abstract properties of linear dependence. Am. J. Math. **57**(3), 509–533 (1935). The Johns Hopkins University Press, Baltimore
5. Lalithambal, R., Sridharan, N.: A study on hamiltonian matroids. Electron Notes Discret. Math. **15**, 201 (2003)
6. Markström, K.: A note on uniquely pancyclic graphs. Aust. J. Combin. **44**, 105–110 (2009)
7. Neel, D.L., Neudauer, N.A.: Matroids you have known. Math. Mag. **82**(1), 26–41 (2009)
8. Oxley, J.: Matroid Theory. Oxford University Press, New York (1992)
9. Rota, G.-C.: Indiscrete Thoughts. Birkhäuser, Boston, MA (1996)
10. Shi, Y.B.: Some theorems of uniquely pancyclic graphs. Discret. Math. **59**, 167–180 (1986). MR 87j:05103
11. Shi, Y.B., Yap, H.P., Teo, S.K.: On uniquely r-pancyclic graphs. In: Graph Theory and Its Applications: East and West, (Jinan, 1986). Annals of the New York Academy of Sciences, vol. 576, pp. 487–499. New York Academy of Sciences, New York (1989). MR 93d:05088
12. West, D.B.: Introduction to Graph Theory, 2nd edn. Prentice Hall, Upper Saddle River (2000)
13. Wilson, R.J.: Graph Theory, 4th edn. Addison Wesley Longman Limited, Harlow, Essex (1996)

Finite Frame Theory

Somantika Datta and Jesse Oldroyd

Suggested Prerequisites. *Linear algebra: Bases, Eigenvalues, Eigenvectors, Inner products. Complex variables: Basic properties of complex numbers.*

1 Introduction

Given a signal, whether it is a discrete vector or a continuous function, one desires to write it in terms of simpler components. Typically, these components or "building blocks" form what is called a *basis*. A basis is an optimal set, having the minimal number of elements, such that any vector (signal) in the underlying space can be written *uniquely* as a linear combination of the basis vectors. Bases play an important role in the analysis of vector spaces, since the characteristics of the signal can be read off from the coefficients in the basis representation. However, if the coefficients get corrupted by noise or if some get lost during transmission then valuable signal information can get lost beyond recovery. The main problem with bases is this lack of flexibility - even a slight modification of a basis can leave us with a set that is no longer a basis. Since the basis representation is typically nonredundant one can try to bring in more flexibility by adding some extra elements and sacrificing the uniqueness property of a basis representation. This leads to the notion of a *frame*. A frame can be thought of as a *redundant* basis, having more elements than needed. In fact, in any finite dimensional vector space every finite spanning set is a frame. Although the redundancy of a frame leads to non-unique

S. Datta (✉)
University of Idaho, Moscow, ID 83844-1103, USA
e-mail: sdatta@uidaho.edu

J. Oldroyd
West Virginia Wesleyan College, Buckhannon, WV 26201, USA
e-mail: oldroyd.j@wvwc.edu

A. Wootton et al. (eds.), *A Primer for Undergraduate Research*, Foundations for Undergraduate Research in Mathematics, DOI 10.1007/978-3-319-66065-3_7

representations, this also makes the corresponding signal representations resilient to noise and robust to transmission losses. In applications, such robustness might be more desirable than having a unique representation.

It is widely acknowledged that the idea of frames originated in the 1952 paper by Duffin and Schaeffer [11], but frames have only gained significant popularity relatively recently due to work such as [10]. Frames are now standard tools in signal processing and are of great interest to mathematicians and engineers alike. This chapter focuses on frames in finite dimensional spaces. The notion of frames for infinite dimensional spaces like function spaces is far more subtle [6, 9, 26] and will not be discussed here. Along with introducing finite frame theory, this chapter discusses a special highly desirable class of frames called *equiangular tight frames.* Possible research ideas suitable for an undergraduate curriculum are also discussed.

Throughout the chapter, many of the well known results will be stated without proofs and the reader will be provided with the necessary references. It is assumed that readers have taken some undergraduate linear algebra course, and are familiar with the notion of a basis and its fundamental properties. For an in depth study of linear algebra, readers may refer to [12, 17]. A nominal knowledge of complex numbers is also assumed. In all that follows, $\mathbb{R}$ will denote the set of real numbers, and $\mathbb{C}$ will denote the set of complex numbers. For a given $c \in \mathbb{C}$, the complex conjugate of c is denoted by $\overline{c}$, and the modulus of c is denoted by $|c|$. In a setting that applies to both $\mathbb{R}$ and $\mathbb{C}$ we will use the notation $\mathbb{F}$, the elements of which are called *scalars*.

We first recall one of the most significant properties of a basis in the following result [12].

Theorem 1 ([12]). *Let $\mathcal{V}$ be a vector space and $B = \{v_1, v_2, \ldots, v_n\}$ be a subset of $\mathcal{V}$. Then B is a basis for $\mathcal{V}$ if and only if each $v \in \mathcal{V}$ can be uniquely expressed as a linear combination of vectors of B, that is, there exist unique scalars $c_1, c_2, \ldots, c_n$ such that*

$$v = c_1 v_1 + c_2 v_2 + \cdots + c_n v_n. \tag{1}$$

The scalars c_i in (1) are called the *coefficients* of v with respect to B. Note that each basis for a given vector space has the same number of elements, and this number is the dimension of the underlying vector space. As will become apparent later, in the case of a frame there is an added flexibility in that frames for the same space can differ in the number of elements. The computation of the coefficients in (1) is important since this allows us to represent v in terms of the basis elements. However, this process can be cumbersome. The concept of an inner product can greatly simplify these calculations. For the convenience of the reader, we next recall some standard definitions and properties pertinent to inner products.

Definition 1. Let $\mathcal{V}$ be a vector space over $\mathbb{F}$. An inner product on $\mathcal{V}$ is a function that assigns to every ordered pair of vectors $u, v \in \mathcal{V}$, a scalar in $\mathbb{F}$ denoted by $\langle u, v \rangle$, such that for all $u, v, w \in \mathcal{V}$ and all $c \in \mathbb{F}$, the following hold:

(a) $\langle u + w, v\rangle = \langle u, v\rangle + \langle w, v\rangle$.
(b) $\langle cu, v\rangle = c\langle u, v\rangle$.
(c) $\overline{\langle u, v\rangle} = \langle v, u\rangle$, where the bar denotes complex conjugation.
(d) $\langle u, u\rangle \geq 0$, with equality if and only if $u = 0$.

A vector space endowed with an inner product is called an *inner product space*. If $\mathbb{F} = \mathbb{R}$, it is called a *real inner product space* and if $\mathbb{F} = \mathbb{C}$, it is called a *complex inner product space*.

Example 1. In $\mathbb{F}^n$, an inner product of two vectors $u = (a_1, a_2, \ldots, a_n)$ and $v = (b_1, b_2, \ldots, b_n)$ can be defined as

$$\langle u, v\rangle = \sum_{i=1}^{n} a_i \overline{b}_i.$$

This is the standard inner product of $\mathbb{F}^n$. When $\mathbb{F} = \mathbb{R}$, the conjugations are not needed, and we have what is commonly referred to as the *dot product*, often written as $u \cdot v$.

The definition of an inner product is used to generalize the notion of length in a vector space. Recall that in $\mathbb{R}^3$, the Euclidean length of a vector $v = (a, b, c)$ is given by $\sqrt{a^2 + b^2 + c^2} = \sqrt{\langle v, v\rangle}$. This leads to the following.

Definition 2. Let $\mathscr{V}$ be an inner product space. For $v \in \mathscr{V}$, the *norm* or *length* of v is defined by

$$\|v\| = \sqrt{\langle v, v\rangle}.$$

If $\|v\| = 1$ then v is called *unit normed*.

Definition 3. Two distinct vectors u, v in an inner product space $\mathscr{V}$ are said to be *orthogonal* if $\langle u, v\rangle = 0$. A subset S of $\mathscr{V}$ is called an *orthogonal set* if any two distinct vectors in S are orthogonal.

Recall that in $\mathbb{R}^2$ or $\mathbb{R}^3$ two vectors that are mutually perpendicular to each other have dot product equal to zero. The notion of an inner product can thus be used to infer the (angular) distance between vectors in a vector space, and leads to a special kind of basis called an *orthonormal basis*.

Definition 4. A basis B of a vector space $\mathscr{V}$ is called an *orthonormal basis* (ONB) if B is an orthogonal set in which every vector is unit normed.

Every finite dimensional vector space has an orthonormal basis. This is a consequence of the Gram-Schmidt orthogonalization process [12]. One of the main advantages of an ONB is that the coefficients in the basis representation, when using an ONB, are very easy to compute. This is due to the following result.

Theorem 2 ([12]). *Let $B = \{u_i\}_{i=1}^n$ be an ONB of $\mathcal{V}$. Any vector $v \in \mathcal{V}$ can be written in terms of the vectors in B as*

$$v = \sum_{i=1}^{n} \langle v, u_i \rangle u_i.$$

Theorem 2 shows that the unique coefficients in the basis representation in (1), when using an ONB, are just given by the inner products $\langle v, u_i \rangle$, and therefore very simple to calculate. Theorem 2 leads to Parseval's Formula which can be thought of as a generalization of the Pythagorean Identity.

Proposition 1 (Parseval's Formula). *[15] Let $\{u_i\}_{i=1}^n$ be an ONB of a vector space $\mathcal{V}$. Then for every $v \in \mathcal{V}$*

$$\|v\|^2 = \sum_{i=1}^{n} |\langle v, u_i \rangle|^2. \tag{2}$$

Parseval's Formula is particularly intimate to frame theory, and the importance of this can be understood by taking a closer look at (2). It says that the norm of the signal v is completely determined by the orthonormal basis coefficients $\{\langle v, u_i \rangle\}$. Suppose that the signal v cannot be analyzed directly but one can measure the coefficients $\{\langle v, u_i \rangle\}$. Since both sides of (2) have the meaning of energy, this suggests that some valuable information of the signal can be obtained solely from its coefficients even if one does not know what the signal is.

A natural question is: why do we wish to generalize bases or why do we want to look beyond bases? First of all, it might be worth pointing out that once a basis for a vector space $\mathcal{V}$ has been fixed, for each $v \in \mathcal{V}$, one can just work with the coefficients of v that appear in the basis representation. This means that if $B = \{v_1, v_2, \ldots, v_n\}$ is a basis of $\mathcal{V}$ and if $c_1, c_2, \ldots, c_n$ are the unique coefficients representing v, that is,

$$v = c_1 v_1 + c_2 v_2 + \cdots + c_n v_n, \tag{3}$$

then in order to store or transmit v, one only uses the vector $(c_1, c_2, \ldots, c_n)$. Once these coefficients are known, one can recover v using (3).

Now suppose that in transmitting v, the coefficients $\{c_i\}_{i=1}^n$ get corrupted by noise, and as a result what is received is $\{c_i + \mu_i\}_{i=1}^n$. During the recovery process, one obtains

$$\widehat{v} = \sum_{i=1}^{n} (c_i + \mu_i) v_i = \sum_{i=1}^{n} c_i v_i + \sum_{i=1}^{n} \mu_i v_i = v + \epsilon.$$

Instead of a basis suppose that one uses a spanning set, that is, a set that spans $\mathcal{V}$ but is linearly dependent. Such a set could be $\{v_1, \ldots, v_n, v_{n+1}, \ldots, v_m\} = \{v_i\}_{i=1}^m$,

$m > n$, obtained from B by adding some additional vectors of $\mathcal{V}$. Then v can be recovered by calculating

$$\widehat{v} = \sum_{i=1}^{m} c_i v_i + \sum_{i=1}^{m} \mu_i v_i$$

where $c_i = 0$, $n < i \leq m$. Since $\{v_i\}_{i=1}^{m}$ is a linearly dependent set, there is a possibility that the second summation $\sum_{i=1}^{m} \mu_i v_i$ will become zero, thereby canceling the noise, something that is never possible when using a basis. Intuitively, this shows how the effect of noise can be reduced by using a redundant set.

As another instance of the benefit of having a redundant spanning set, let us consider $v = (1, 1) \in \mathbb{R}^2$.[1] Using the standard orthonormal basis of $\mathbb{R}^2$, $\{e_1 = (1, 0), e_2 = (0, 1)\}$, v can be written as

$$v = 1 \cdot e_1 + 1 \cdot e_2.$$

If one of the coefficients, say the second coefficient is lost, then at the reconstruction stage one gets

$$\widehat{v} = 1 \cdot e_1 + 0 \cdot e_2 = (1, 0)$$

and the error in the reconstruction is

$$\|v - \widehat{v}\|_2 = 1$$

where $\|.\|_2$ is the Euclidean distance in $\mathbb{R}^2$. Instead of an ONB, let us now use the redundant set

$$\{f_k = (\cos(2k\pi/6), \sin(2k\pi/6))\}_{k=0}^{5}.$$

See Figure 1(a). In terms of this set, the vector $(1, 1)$ can be written as

$$v = 0.333f_0 + 0.455f_1 + 0.122f_2 - 0.333f_3 - 0.455f_4 - 0.122f_5.$$

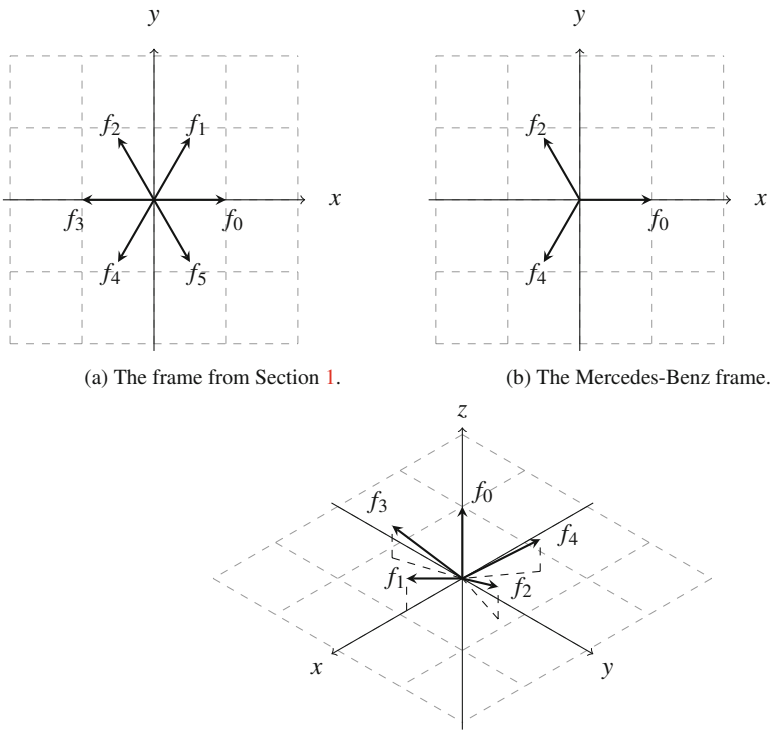
If the coefficient corresponding to f_0 is lost then one obtains

$$\widehat{v} = 0.455f_1 + 0.122f_2 - 0.333f_3 - 0.455f_4 - 0.122f_5$$

and the reconstruction error is

$$\|v - \widehat{v}\|_2 = 1/3$$

[1] The authors would like to thank Andy Kebo for sharing this example.



(a) The frame from Section 1.

(b) The Mercedes-Benz frame.

(c) The $(5,3)$ Grassmannian frame.

Fig. 1 Examples of frames.

which is less than the error when the first coefficient was lost while using the standard ONB of $\mathbb{R}^2$. Losing a single coefficient with an ONB in $\mathbb{R}^2$ can be thought of as losing 50% of the coefficients. In the case of the redundant set under consideration suppose that 50%, that is, three coefficients are lost. If they are the first three, then the reconstructed vector is

$$\widehat{\widehat{v}} = -0.333f_3 - 0.455f_4 - 0.122f_5$$

and the reconstruction error is

$$\|v - \widehat{\widehat{v}}\|_2 = 1/\sqrt{2}$$

which is still less than when using the standard ONB of $\mathbb{R}^2$.

In the above, what we have sacrificed by using a linearly dependent spanning set instead of a basis is that we no longer have the unique representation of Theorem 1. The question is whether a unique representation is necessary for our purpose and the answer is no; as long as we are able to represent every vector in terms of the set, it

does not matter. This notion of adding redundancy is what is incorporated in a *frame* for a finite dimensional vector space. The set $\{f_k = (\cos(2k\pi/6), \sin(2k\pi/6))\}_{k=0}^{5}$ used in the discussion above is not a basis but is a frame for $\mathbb{R}^2$, and we have shown how redundancy is better when we have transmission losses. The next section gives the basics of finite frame theory. The reader is urged to read [6,15] for more details. An excellent overview of the ideas underlying frames can be found in [18].

2 Frames in Finite Dimensional Spaces

To help us think of frames as generalizations of bases, let us look at Parseval's Formula in Proposition 1. Relaxing the condition in (2) gives the following definition.

Definition 5. Let $\mathcal{V}$ be an inner product space and let $\{f_i\}_{i\in\mathcal{I}}$ be a subset of $\mathcal{V}$ indexed by some countable set $\mathcal{I}$.

(i) The set $\{f_i\}_{i\in\mathcal{I}}$ is a *frame* if there exist constants $0 < A \leq B < \infty$ such that for every $v \in \mathcal{V}$,

$$A\|v\|^2 \leq \sum_{i\in\mathcal{I}} |\langle v, f_i\rangle|^2 \leq B\|v\|^2. \tag{4}$$

(ii) The constants A and B are called the *lower* and *upper frame bound*, respectively.
(iii) If $A = B$, the frame is called a *tight frame*.
(iv) If for each $i \in \mathcal{I}$, $\|f_i\| = 1$, the frame is called a *unit normed frame*.

Due to Parseval's Formula, an orthonormal basis is a unit normed tight frame with frame bound equal to 1. In a finite d-dimensional vector space $\mathcal{V}$, a finite set $\{f_i\}_{i=1}^{n}$, $n \geq d$, is a frame if and only if $\{f_i\}_{i=1}^{n}$ is a spanning set of $\mathcal{V}$ [6].

Let $\{f_i\}_{i=1}^{n}$ be a frame for a finite dimensional inner product space $\mathcal{V}$. The *Bessel map* $F : \mathcal{V} \to \mathbb{F}^n$ is defined by

$$F(v) = \{\langle v, f_i\rangle\}_{i=1}^{n}, \quad v \in \mathcal{V}. \tag{5}$$

The adjoint of F is given by

$$F^* : \mathbb{F}^n \to \mathcal{V}, \quad F^*(\{c_i\}_{i=1}^{n}) = \sum_{i=1}^{n} c_i f_i. \tag{6}$$

The mapping F is often referred to as the *analysis operator*, while F^* is referred to as the *synthesis operator*. In a finite dimensional space like $\mathbb{R}^d$ or $\mathbb{C}^d$, the synthesis operator F^* can be written as a $d \times n$ matrix whose columns are the frame vectors. The analysis operator F is then an $n \times d$ matrix whose i^{th} row is f_i^*. In other words, F^* is just the conjugate transpose of F in this setting.

Lemma 1. *The analysis operator F given by (5) is one to one.*

Proof. To show that F is one to one, it is enough to show that the null space of F consists of only the zero vector. Let $Fv = 0$ for some $v \in \mathscr{V}$. Then for $i = 1, \ldots, n$, $\langle v, f_i \rangle = 0$. Since $\{f_i\}_{i=1}^n$ spans $\mathscr{V}$, v must equal zero. Thus F is one to one. □

Lemma 2. *If $\{e_i\}_{i=1}^n$ is the standard orthonormal basis [2] of $\mathbb{F}^n$, then for $i = 1, \ldots, n$,*

$$F^*(e_i) = f_i.$$

Proof. Let v be a vector in $\mathscr{V}$. Then for $i = 1, \ldots, n$,

$$\begin{aligned} \langle v, F^* e_i \rangle &= \langle Fv, e_i \rangle \\ &= \langle v, f_i \rangle. \end{aligned}$$

Thus $F^* e_i = f_i$. □

Lemma 3. *The synthesis operator given by (6) maps $\mathbb{F}^n$ onto $\mathscr{V}$.*

Proof. Let v be a vector in $\mathscr{V}$. Since $\{f_i\}_{i=1}^n$ spans $\mathscr{V}$, there exist constants $\alpha_1, \ldots, \alpha_n$ such that

$$\begin{aligned} v &= \alpha_1 f_1 + \ldots + \alpha_n f_n \\ &= \alpha_1 F^*(e_1) + \ldots + \alpha_n F^*(e_n) \\ &= F^*(\alpha_1 e_1 + \ldots + \alpha_n e_n). \end{aligned}$$

where the penultimate step follows from Lemma 2. Thus the vector $\alpha_1 e_1 + \ldots + \alpha_n e_n \in \mathbb{F}^n$ gets mapped to v showing that F^* is onto. □

The composition of F^* with F gives the *frame operator*

$$S : \mathscr{V} \to \mathscr{V}, \quad S(v) = F^* F(v) = \sum_{i=1}^{n} \langle v, f_i \rangle f_i. \tag{7}$$

If we restrict ourselves to either $\mathbb{R}^d$ or $\mathbb{C}^d$, then the frame operator S is the $d \times d$ matrix F^*F where F is the matrix corresponding to the analysis operator. Note that in terms of the frame operator, for any $v \in \mathscr{V}$,

$$\sum_{i=1}^{n} |\langle v, f_i \rangle|^2 = \langle Sv, v \rangle,$$

[2] e_i is the vector whose i^{th} coordinate is equal to 1 and the rest are zero.

and (4) can be rewritten as

$$A\|v\|^2 \leq \langle Sv, v\rangle \leq B\|v\|^2. \tag{8}$$

For a tight frame, when $A = B$, (8) implies that

$$S = AI,$$

and so

$$\sum_{i=1}^{n} |\langle v, f_i\rangle|^2 = A\|v\|^2$$

$$\text{or,} \quad \|v\|^2 = \sum_{i=1}^{n} \frac{1}{A} |\langle v, f_i\rangle|^2. \tag{9}$$

Note that (9) resembles Parseval's formula that is satisfied by an ONB. In this regard, unit normed tight frames are the redundant counterparts of orthonormal bases. Finally, since $S = AI$ for a tight frame, (7) implies a representation of any $v \in \mathcal{V}$ in terms of the vectors of a tight frame:

$$v = \frac{1}{A} \sum_{i=1}^{n} \langle v, f_i\rangle f_i. \tag{10}$$

If we define a new set of vectors as

$$g_i = \frac{1}{A} f_i,$$

then (10) can be written as

$$v = \sum_{i=1}^{n} \langle v, g_i\rangle f_i = \sum_{i=1}^{n} \langle v, f_i\rangle g_i. \tag{11}$$

As can be seen from (11), for a tight frame, the coefficients in the expression for v can be obtained simply by calculating the inner products with the g_is or the f_is, in a manner similar to the case of an ONB. In the case of a general frame that is not necessarily tight, a frame representation as in (10) or (11) is still possible and is now given.

Theorem 3 ([6]). *Let $\{f_i\}_{i=1}^n$ be a frame for a finite dimensional inner product space $\mathcal{V}$, and let S be the frame operator. Then*

(i) S is invertible and self-adjoint, i.e., $S = S^*$.

(ii) Every $v \in \mathscr{V}$ can be represented as

$$v = \sum_{i=1}^{n} \langle v, S^{-1} f_i \rangle f_i = \sum_{i=1}^{n} \langle v, f_i \rangle S^{-1} f_i. \tag{12}$$

Proof.

(i) We will first show that the frame operator S is one to one. By (8), $Sv = 0$ forces $v = 0$. Therefore the null space of S contains only the zero vector and S is one-to-one. Since S maps $\mathscr{V}$ to $\mathscr{V}$, S is also onto, and hence invertible.

By properties of the adjoint operator:

$$S^* = (F^*F)^* = F^*(F^*)^* = F^*F = S.$$

Thus S is self-adjoint.

(ii) By (i), S^{-1} exists, and so for each $v \in \mathscr{V}$ we have

$$v = S^{-1}Sv = S^{-1} \sum_{i=1}^{n} \langle v, f_i \rangle f_i, \quad \text{by (7)},$$

$$= \sum_{i=1}^{n} \langle v, f_i \rangle S^{-1} f_i.$$

Note that since S is self-adjoint, so is S^{-1}. Thus, again by (7),

$$v = SS^{-1}v = \sum_{i=1}^{n} \langle S^{-1}v, f_i \rangle f_i$$

$$= \sum_{i=1}^{n} \langle v, S^{-1} f_i \rangle f_i.$$

□

The set $\{S^{-1}f_i\}_{i=1}^n$ is also a frame for $\mathscr{V}$ and is called the *canonical dual* of $\{f_i\}_{i=1}^n$. The numbers $\langle v, S^{-1}f_i \rangle$ are called the *frame coefficients*.

If $\{f_i\}_{i=1}^n$ is a spanning set that is not a basis of $\mathscr{V}$, there must exist constants $\{\gamma_i\}_{i=1}^n$, not all zero, such that $\sum_{i=1}^n \gamma_i f_i = 0$. Thus, by adding zero to the first part of (12),

$$v = \sum_{i=1}^{n} \langle v, S^{-1} f_i \rangle f_i + \sum_{i=1}^{n} \gamma_i f_i$$

$$= \sum_{i=1}^{n} (\langle v, S^{-1} f_i \rangle + \gamma_i) f_i,$$

and so there are infinitely many representations of v in terms of the frame vectors. The representation of v in (12) is called the *canonical form*. With a frame, we therefore have much more flexibility when choosing a representation for v compared to a basis representation. However, the special feature of the frame coefficients $\{\langle v, S^{-1}f_i\rangle\}$ is that they have the minimal ℓ^2-norm among all coefficients $\{c_i\}_{i=1}^n$ such that $v = \sum_{i=1}^n c_i f_i$. This can be stated as follows.

Theorem 4 ([6]). *Let $\{f_i\}_{i=1}^n$ be a frame for a finite dimensional inner product space $\mathcal{V}$ with frame operator S. If $v \in \mathcal{V}$ has the representation $v = \sum_{i=1}^n c_i f_i$, then*

$$\sum_{i=1}^n |c_i|^2 = \sum_{i=1}^n |\langle v, S^{-1}f_i\rangle|^2 + \sum_{i=1}^n |c_i - \langle v, S^{-1}f_i\rangle|^2.$$

Example 2 (Computing a Frame Expansion). Consider the frame in $\mathbb{R}^2$ given by the rows of

$$F = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix}$$

and let

$$v = \begin{bmatrix} 1 \\ -2 \end{bmatrix}.$$

The frame operator corresponding to this frame is given by

$$S = F^T F = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix},$$

and the frame coefficients for v are

$$\langle v, S^{-1}f_1\rangle = \frac{4}{3}$$
$$\langle v, S^{-1}f_2\rangle = -\frac{5}{3}$$
$$\langle v, S^{-1}f_3\rangle = -\frac{1}{3}.$$

One can then verify that

$$v = \frac{4}{3}f_1 - \frac{5}{3}f_2 - \frac{1}{3}f_3,$$

which is the canonical form of v. The ℓ^2-norm of the coefficients in the canonical form is $\sqrt{\frac{14}{3}}$. It is not too difficult to see that v can also be written as $v = f_1 - 2f_2$. The ℓ^2-norm of the coefficients in this expansion is given by $\sqrt{5}$. As expected, this is greater than the ℓ^2-norm of the coefficients in the canonical form.

Note that computing the frame coefficients involves inverting the frame operator S which can be numerically unstable. In the case of a tight frame, this step is drastically simplified due to the fact that $S^{-1} = \frac{1}{A}I$, a constant multiple of the identity. This feature has made tight frames highly desirable. They emulate ONBs and at the same time provide the benefits of redundancy that come from using frames.

The *frame potential* [2] of the frame $\{f_i\}_{i=1}^n$ is the number $FP(\{f_i\}_{i=1}^n)$ defined by

$$FP(\{f_i\}_{i=1}^n) = \sum_{i=1}^{n}\sum_{j=1}^{n} |\langle f_i, f_j\rangle|^2.$$

The frame potential is used to give an important characterization of unit normed tight frames and ONBs.

Theorem 5 (Theorem 6.2, [2]). *Let $d, n \in \mathbb{N}$ with $d \leq n$, and let $\{f_i\}_{i=1}^n$ be a unit normed frame in $\mathbb{R}^d$ or $\mathbb{C}^d$. Then $FP(\{f_i\}_{i=1}^n) \geq \frac{n^2}{d}$ with equality if and only if $\{f_i\}_{i=1}^n$ is a unit normed tight frame (or an ONB in the case $n = d$).*

The optimal lower bound of a frame is the supremum over all constants A that satisfy the left inequality in (4). Similarly, the optimal upper bound is the infimum over all constants B that satisfy the right side of inequality (4). In the finite dimensional setting, the optimal lower and upper frame bounds are given by the minimum and maximum eigenvalues of the matrix of S. The following theorem gives some useful results involving the eigenvalues of the frame operator. The proof, given in [6], is included here.

Theorem 6 ([6]). *Let $\{f_i\}_{i=1}^n$ be a frame for a d-dimensional space $\mathscr{V}$. Then the following hold.*

(i) The optimal lower frame bound is the smallest eigenvalue of S, and the optimal upper frame bound is its largest eigenvalue.
(ii) Let $\{\lambda_i\}_{i=1}^d$ denote the eigenvalues of S. Then

$$\sum_{i=1}^{d} \lambda_i = \sum_{i=1}^{n} \|f_i\|^2.$$

(iii) If $\{f_i\}_{i=1}^n$ is a tight frame and for all i, $\|f_i\| = 1$, then the frame bound is $A = \frac{n}{d}$.

Proof.

(i) Since the frame operator $S : \mathcal{V} \to \mathcal{V}$ is self-adjoint, there is an orthonormal basis of $\mathcal{V}$ consisting of eigenvectors of S. Denote this eigenvector basis by $\{e_i\}_{i=1}^d$ and the corresponding eigenvalues by $\{\lambda_i\}_{i=1}^d$. Every $v \in \mathcal{V}$ can be written as

$$v = \sum_{i=1}^{d} \langle v, e_i \rangle e_i.$$

Then

$$Sv = \sum_{i=1}^{d} \langle v, e_i \rangle Se_i = \sum_{i=1}^{d} \lambda_i \langle v, e_i \rangle e_i,$$

and

$$\sum_{i=1}^{n} |\langle v, f_i \rangle|^2 = \langle Sv, v \rangle = \sum_{i=1}^{d} \lambda_i |\langle v, e_i \rangle|^2.$$

Therefore,

$$\lambda_{\min} \|v\|^2 \leq \sum_{i=1}^{n} |\langle v, f_i \rangle|^2 \leq \lambda_{\max} \|v\|^2.$$

This shows that $\lambda_{\min}$ is a lower frame bound and $\lambda_{\max}$ is an upper frame bound. Taking v to be an eigenvector corresponding to $\lambda_{\min}$, respectively, $\lambda_{\max}$, proves that it is the optimal lower bound, respectively, upper bound.

(ii) We have

$$\begin{aligned} \sum_{i=1}^{d} \lambda_i = \sum_{i=1}^{d} \lambda_i \|e_i\|^2 &= \sum_{i=1}^{d} \langle Se_i, e_i \rangle \\ &= \sum_{i=1}^{d} \sum_{j=1}^{n} |\langle e_i, f_j \rangle|^2. \end{aligned}$$

Interchanging the order of summation and using the fact that $\{e_i\}_{i=1}^d$ is an ONB for $\mathcal{V}$ gives the desired result.

(iii) By the assumptions, $S = AI$ and so S has one eigenvalue equal to A with multiplicity d. By part (ii), this means that

$$dA = n$$

and this gives $A = \frac{n}{d}$. □

The $n \times n$ matrix FF^* is the *Gram matrix* of the set $\{f_i\}_{i=1}^n$. The Gram matrix has rank d, and its non-zero eigenvalues are the same as the eigenvalues of the frame operator S. The $(i,j)^{\text{th}}$ entry of the Gram matrix is the inner product $\langle f_j, f_i \rangle$.

Example 3 (Types of Frames).

(a) The frame $\{f_k = (\cos(2k\pi/6), \sin(2k\pi/6))\}_{k=0}^5$ given in Section 1 is a unit normed tight frame of six vectors in $\mathbb{R}^2$. See Figure 1(a). The analysis operator is given by

$$F = \begin{bmatrix} 1 & 0 \\ 1/2 & \sqrt{3}/2 \\ -1/2 & \sqrt{3}/2 \\ -1 & 0 \\ -1/2 & -\sqrt{3}/2 \\ 1/2 & -\sqrt{3}/2 \end{bmatrix}$$

and the frame operator is

$$S = \begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix}.$$

We thus have a tight frame with frame bound equal to 3.

(b) The vectors $\{f_0, f_2, f_4\}$ from the frame given in (a) form a special unit normed tight frame known as the *Mercedes-Benz frame*. These vectors are the first, third and fifth rows of the matrix F. The corresponding frame operator is the matrix

$$\begin{bmatrix} \frac{3}{2} & 0 \\ 0 & \frac{3}{2} \end{bmatrix}.$$

The frame bound for this frame is therefore equal to $\frac{3}{2}$. See Figure 1(b). This is an example of an *equiangular tight frame*, see Definition 6. Note that the absolute value of the inner product of any two distinct vectors in this set is equal to $\frac{1}{2}$.

(c) Let $\mu = \frac{1}{\sqrt{5}}$. The set $\{f_k\}_{k=0}^4 \subset \mathbb{R}^3$ given by the rows of

$$\begin{bmatrix} 0 & 0 & 1 \\ \sqrt{1-\mu^2} & 0 & \mu \\ \mu\sqrt{\frac{1-\mu}{1+\mu}} & \sqrt{\frac{(1+2\mu)(1-\mu)}{1+\mu}} & \mu \\ \mu\sqrt{\frac{1-\mu}{1+\mu}} & -\sqrt{\frac{(1+2\mu)(1-\mu)}{1+\mu}} & \mu \\ -\mu\sqrt{\frac{1+\mu}{1-\mu}} & \sqrt{\frac{(1-2\mu)(1+\mu)}{1-\mu}} & \mu \end{bmatrix}$$

forms a unit normed frame, but not a tight frame, of five vectors in $\mathbb{R}^3$. This is an example of a *Grassmannian frame* [3], see Definition 7. The eigenvalues of the corresponding frame operator are 1 and 2. By Theorem 6, the optimal lower frame bound must be 1 and the optimal upper frame bound must be 2. See Figure 1(c).

(d) Given $n \in \mathbb{N}$, let $\omega = e^{-\frac{2\pi i}{n}}$. The $n \times n$ discrete Fourier transform (DFT) matrix is given by

$$F = \frac{1}{\sqrt{n}} \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \omega & \omega^2 & \dots & \omega^{n-1} \\ 1 & \omega^2 & \omega^4 & \dots & \omega^{2(n-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{n-1} & \omega^{2(n-1)} & \dots & \omega^{(n-1)(n-1)} \end{bmatrix}.$$

The rows form an orthonormal basis of $\mathbb{C}^n$. Let $\widehat{F}$ denote the $n \times d$ matrix formed by selecting d columns from F. The set $\{f_k\}_{k=0}^{n-1}$ given by the rows of $\widehat{F}$ forms a tight frame in $\mathbb{C}^d$, but not necessarily a unit normed frame.

3 Equiangular Tight Frames

In a communications system, when there are several signals trying to access the same channel, it is desired that there is minimum interference between the signals. The cross correlation between two signals, given by the absolute value of their inner product, can be interpreted as a measure of their interference. This is minimized when the two vectors (signals) are mutually orthogonal i.e. their inner product is zero. However, in a finite dimensional space, the maximum number of mutually orthogonal vectors is the same as the dimension. When the number of vectors exceeds the dimension of the underlying space, we desire that the maximum cross correlation between any two vectors is minimized. Welch [24] gave the following lower bound for the maximum cross correlation among unit normed vectors $\{f_i\}_{i=1}^n$ in $\mathbb{F}^d$:

$$\max_{i \neq j} |\langle f_i, f_j \rangle| \geq \sqrt{\frac{n-d}{d(n-1)}}, \quad n \geq d. \tag{13}$$

In order to minimize the maximum cross correlation among vectors, one seeks sets of vectors that meet the lower bound of (13). Equiangular tight frames (ETFs) are a class of frames that meet the lower bound. ETFs are highly desirable and are the closest one can come to an ONB while having the redundancy of a frame, in the sense of minimizing the cross correlation while maintaining tightness. The bound $\sqrt{\frac{n-d}{d(n-1)}}$ in (13) is called the Welch bound and is denoted by α. The formal definition of an ETF is as follows.

Definition 6 ([22]). An equiangular tight frame (ETF) is a set $\{f_i\}_{i=1}^n$ in a d-dimensional space $\mathcal{V}$ satisfying:

(i) $F^*F = \frac{n}{d}I$, i.e., the set is a tight frame.
(ii) $\|f_i\| = 1$, for $i = 1, \ldots, n$, i.e., the set is unit normed.
(iii) $|\langle f_i, f_j\rangle| = \alpha$, $1 \leq i < j \leq n$, where α is the Welch bound.

For a given dimension d and frame size n, an ETF of n vectors in $\mathbb{F}^d$ may not exist [22]. Even when they do exist, ETFs are hard to construct. However, ETFs of $d + 1$ vectors for dimension d always exist and can be viewed as the vertices of a regular simplex centered at the origin [21,22]. Examples of such ETFs can be found in [13], and an explicit construction is also given in [7]. Note that for such an ETF the Welch bound is given by $\alpha = \frac{1}{d}$. An example of an ETF of three vectors in $\mathbb{R}^2$ was given earlier in Example 3(b).

Example 4 (An ETF of Four Vectors in $\mathbb{R}^3$). Let $d = 3$. Consider the four vectors given by

$$f_1 = \begin{bmatrix} -\frac{2}{\sqrt{6}} \\ -\frac{\sqrt{2}}{3} \\ -\frac{1}{3} \end{bmatrix}, \quad f_2 = \begin{bmatrix} \frac{2}{\sqrt{6}} \\ -\frac{\sqrt{2}}{3} \\ -\frac{1}{3} \end{bmatrix}, \quad f_3 = \begin{bmatrix} 0 \\ \frac{2\sqrt{2}}{3} \\ -\frac{1}{3} \end{bmatrix}, \quad f_4 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

In this case, for $i \neq j$, $\langle f_i, f_j\rangle = -1/3$.

When ETFs cannot exist, the Welch bound cannot be attained by any set of n vectors in $\mathbb{F}^d$. However, the maximum cross correlation between frame vectors can still be minimized even if the minimum does not coincide with the Welch bound. Such sets are called *Grassmannian frames* [3].

Definition 7. A frame of n vectors in $\mathbb{F}^d$ is called a Grassmannian frame if it is a solution to

$$\min\{\max_{i \neq j} |\langle f_i, f_j\rangle|\}$$

where the minimum is taken over all unit normed frames $\{f_i\}_{i=1}^n$ in $\mathbb{F}^d$.

An example of a Grassmannian frame of five vectors in $\mathbb{R}^3$ is shown in Figure 1(c). Figure 2 demonstrates the relationships present between different families of frames.

3.1 k-Angle Tight Frames

As mentioned previously, equiangular tight frames are useful in applications because they minimize the maximum cross correlation among pairs of unit vectors,

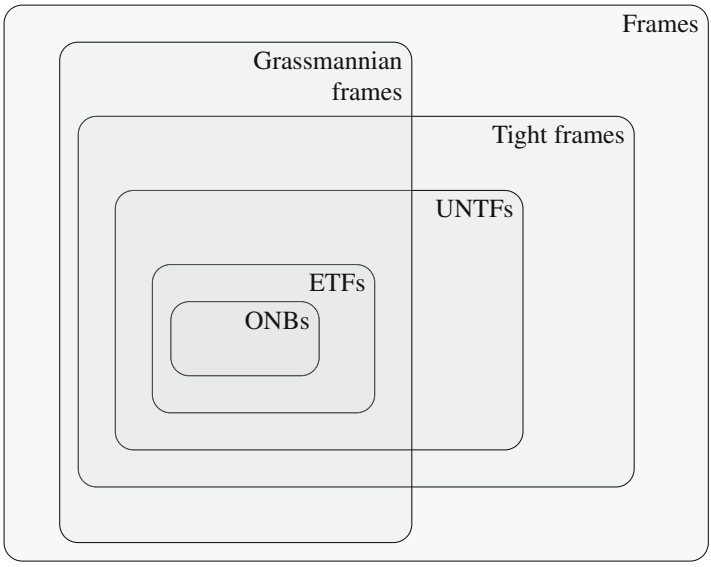


Fig. 2 Families of frames.

but they are also rare. Therefore, it is desirable to construct sets that mimic ETFs in some way. If $\{f_i\}_{i=1}^n$ is an ETF in $\mathbb{R}^d$ or $\mathbb{C}^d$, then when $i \neq j$, $|\langle f_i, f_j\rangle|$ must be equal to the Welch bound α. Relaxing this condition, one obtains a larger class of frames that contains the equiangular tight frames. In particular, a *k-angle tight frame* is a set $\{f_i\}_{i=1}^n$ in a d-dimensional space $\mathscr{V}$ satisfying [7]:

(i) $F^*F = \frac{n}{d}I$, i.e., the set is a tight frame.
(ii) $\|f_i\| = 1$ for $i = 1, \ldots, n$, i.e., the set is unit normed.
(iii) $|\langle f_i, f_j\rangle| \in \{\alpha_m\}_{m=1}^k$ for $1 \leq i < j \leq n$, where $\{\alpha_m\}_{m=1}^k \subset [0, 1]$.

Example 5 (Mutually Unbiased Bases). Two orthonormal bases $\{f_i\}_{i=1}^d$ and $\{g_i\}_{i=1}^d$ in a d-dimensional space $\mathscr{V}$ form a pair of *mutually unbiased bases* if $|\langle f_i, g_j\rangle| = \frac{1}{\sqrt{d}}$ for $1 \leq i, j \leq d$. Let $\{\widetilde{f}_i\}_{i=1}^{2d}$ denote the union $\{f_i\}_{i=1}^d \cup \{g_i\}_{i=1}^d$. Then $\{\widetilde{f}_i\}_{i=1}^{2d}$ is a 2-angle tight frame, since $|\langle \widetilde{f}_i, \widetilde{f}_j\rangle| \in \{0, \frac{1}{\sqrt{d}}\}$ for $1 \leq i < j \leq 2d$.

A way to construct k-angle tight frames uses ETFs as starting points.

Theorem 7. *[7] Let $d, k \in \mathbb{N}$ with $k < d + 1$, and set $d' = \binom{d+1}{k}$. Denote the collection of all subsets of $\{1, \ldots, d+1\}$ of size k by $\{\Lambda_i\}_{i=1}^{d'}$. Let $\{f_i\}_{i=1}^{d+1} \subseteq \mathbb{R}^d$ denote the ETF with $\langle f_i, f_j\rangle = -\frac{1}{d}$ for $i \neq j$. Define a new collection $\{g_i\}_{i=1}^{d'}$ as follows:*

$$g_i := \frac{\sum_{j\in\Lambda_i} f_j}{\|\sum_{j\in\Lambda_i} f_j\|}.$$

Then $\{g_i\}_{i=1}^{d'}$ *forms a* $\hat{k}$*-angle tight frame of* d' *vectors in* $\mathbb{R}^d$*, where* $\hat{k} \leq k$*.*

The proof of Theorem 7 can be found in [7]. The construction of the starting ETF with $\langle f_i, f_j\rangle = -\frac{1}{d}$, $i \neq j$, can be done based on an algorithm in [7], and an example of such an ETF for $d = 3$ has been provided earlier in this section in Example 4. The main idea behind the proof of the tightness part is to compute the frame potential of $\{g_i\}_{i=1}^{d'}$ and then use Theorem 5.

Example 6 (A 2-Angle Tight Frame in $\mathbb{R}^2$*).* Consider the Mercedes-Benz (MB) frame discussed in Example 3(b), and shown below in Figure 3. It consists of the three vectors in $\mathbb{R}^2$ given by

$$f_0 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, f_1 = \begin{bmatrix} -1/2 \\ \sqrt{3}/2 \end{bmatrix}, f_2 = \begin{bmatrix} -1/2 \\ -\sqrt{3}/2 \end{bmatrix}.$$

Note that for $i \neq j$, $\langle f_i, f_j\rangle = -1/2$. Let $k = 2$. Follow the construction given in Theorem 7, i.e., for every distinct pair f_i, f_j, $i < j$, in the MB frame, calculate $\frac{f_i+f_j}{\|f_i+f_j\|}$ to get three more vectors. Now add these three vectors to the MB frame. Doing so produces a unit normed tight frame of six vectors in $\mathbb{R}^2$, that was shown in Figure 1(a), and is reproduced in Figure 4. $\{f_i\}_{i=0}^{5}$ is a 2-angle tight frame: for $0 \leq i < j \leq 5$, $\langle f_i, f_j\rangle \in \{-\frac{1}{3}, \frac{1}{3}, -1\}$ which means that $|\langle f_i, f_j\rangle| \in \{\frac{1}{3}, 1\}$.

3.2 Tight Frames and Graphs

Equiangular tight frames have important connections to graph theory. Perhaps the most well-known connection is that between ETFs and graph theoretic objects known as *regular two-graphs* [16, 21]. In particular, [16] gives a one-to-one correspondence between equivalence classes of real ETFs and regular two-graphs.

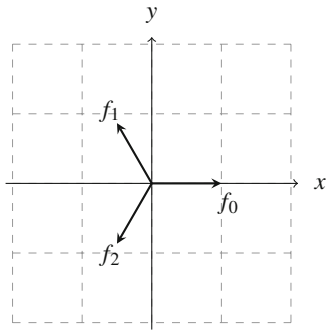


Fig. 3 The Mercedes-Benz frame.

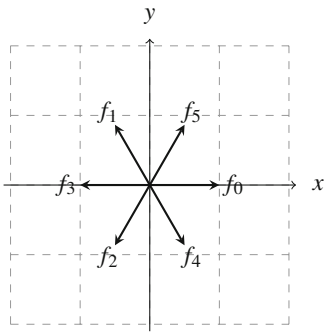


Fig. 4 A 2-angle tight frame in $\mathbb{R}^2$.

Another important connection exists between ETFs and strongly regular graphs as described below. A graph for which every vertex has the same number of neighbors is a *regular graph*. A regular graph $\mathscr{G}$ is called a *strongly regular graph* if any two adjacent vertices in $\mathscr{G}$ have λ common neighbors, and any two non-adjacent vertices in $\mathscr{G}$ have μ common neighbors. In [23], it is shown that a real ETF $\{f_i\}_{i=1}^n$ in $\mathbb{R}^d$ exists if and only if a particular strongly regular graph exists. Graph theory has also proven useful in obtaining error estimates for signal reconstruction when using ETFs in the presence of transmission losses [4].

These connections go beyond characterizing ETFs. A unit normed tight frame $\{f_i\}_{i=1}^n \subset \mathbb{R}^d$ is called a *two-distance tight frame*[3] if $\langle f_i, f_j \rangle \in \{\alpha_1, \alpha_2\} \subset [-1, 1]$ for $\alpha_1 \neq \alpha_2$ and $1 \leq i < j \leq n$. If G is the Gram matrix of a two-distance tight frame, then $G = I + \alpha_1 Q_1 + \alpha_2 Q_2$, where Q_1 and Q_2 are symmetric binary matrices with zeros on the diagonal. Recall that a graph $\mathscr{G}$ has adjacency matrix $Q = [q_{ij}]$ defined by

$$q_{ij} = \begin{cases} 1 & \text{if vertex } i \text{ is adjacent to vertex } j, \\ 0 & \text{otherwise.} \end{cases}$$

The matrices Q_1 and Q_2 in the decomposition of the Gram matrix G above can then be viewed as adjacency matrices of graphs. In [1], the authors prove the following result.

Theorem 8 (Proposition 3.2, [1]). *Let $\{f_i\}_{i=1}^n$ be a two-distance tight frame with Gram matrix $G = I + \alpha_1 Q_1 + \alpha_2 Q_2$, where $\alpha_1 \neq \pm\alpha_2$. Then Q_1 and Q_2 are both adjacency matrices of strongly regular graphs.*

[3]Note that two-distance tight frames as defined here and in [1] are either ETFs or 2-angle tight frames from the beginning of Section 3.1.

4 Possible Research Topics

4.1 Construction of Grassmannian Frames

Equiangular tight frames are important in many applications, but as discussed previously they do not always exist. However, ETFs are themselves a specific example of a larger class of frames called Grassmannian frames (see Figure 2), and for any choice of d, n, a Grassmannian frame always exists [3]. Grassmannian frames are important in frame theory for many of the same reasons that ETFs are, since Grassmannian frames minimize the maximum cross correlation among sets of unit vectors.

Despite their importance, many questions remain unanswered about the construction of Grassmannian frames for given values of d, n, in either $\mathbb{R}^d$ or $\mathbb{C}^d$. Constructions in $\mathbb{R}^2$ and $\mathbb{R}^3$ have been given in [3].

Research Project 1. Some interesting questions that can be explored in the context of Grassmannian frames are the following.

- How can one verify whether or not a given frame is a Grassmannian frame?
- When can one construct Grassmannian frames from previously existing "structured" objects such as ETFs?
- Improvements on the Welch bound exist in certain situations, such as the following bound given in [25]:

$$\max_{i\neq j} |\langle f_i, f_j\rangle| \geq \max\left\{\sqrt{\frac{n-d}{d(n-1)}}, 1-2n^{-\frac{1}{d-1}}\right\}.$$

  How sharp are these bounds for various values of d and n? Using computer experiments to find a wide variety of unit normed frames and comparing their maximum cross correlation with these bounds could give insight into how close to the lower bound one can get and what the minimizers of the maximum cross correlation could be when ETFs do not exist.

4.2 k-Angle Tight Frames and Regular Graphs

As mentioned in Section 3.2, there exists a correspondence between certain 2-angle tight frames and strongly regular graphs. A natural course of action would be to extend this result to k-angle tight frames where $k \geq 3$, and some progress has already

been made in the case $k = 3$ [19]. Let G be the Gram matrix of a k-angle tight frame $\{f_i\}$, and suppose that $|\langle f_{i_1}, f_{j_1}\rangle| = |\langle f_{i_2}, f_{j_2}\rangle|$ if and only if $\langle f_{i_1}, f_{j_1}\rangle = \langle f_{i_2}, f_{j_2}\rangle$. In other words, if $\alpha \in \{\langle f_i, f_j\rangle\}$, then $-\alpha \notin \{\langle f_i, f_j\rangle\}$. Let $\{\alpha_i\}_{i=1}^k = \{\langle f_i, f_j\rangle\}$, which are the distinct off-diagonal entries of G. Then

$$G = I + \alpha_1 Q_1 + \cdots + \alpha_k Q_k, \tag{14}$$

where the matrices Q_i are symmetric binary matrices with 0s along the diagonal for $1 \leq i \leq k$. Each matrix Q_i is the adjacency matrix of a graph. The question then is as follows:

Research Project 2.
What can be said about the structure of the graphs determined by the matrices $\{Q_i\}_{i=1}^k$ in (14)?

To illustrate this question, consider the matrix G given by

$$G = \begin{bmatrix}
1 & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & -\frac{2}{3} & \frac{1}{6} & \frac{1}{6} & -\frac{2}{3} & -\frac{2}{3} \\
\frac{1}{6} & 1 & \frac{1}{6} & \frac{1}{6} & -\frac{2}{3} & \frac{1}{6} & \frac{1}{6} & -\frac{2}{3} & \frac{1}{6} & -\frac{2}{3} \\
\frac{1}{6} & \frac{1}{6} & 1 & -\frac{2}{3} & \frac{1}{6} & \frac{1}{6} & -\frac{2}{3} & \frac{1}{6} & \frac{1}{6} & -\frac{2}{3} \\
\frac{1}{6} & \frac{1}{6} & -\frac{2}{3} & 1 & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & -\frac{2}{3} & -\frac{2}{3} & \frac{1}{6} \\
\frac{1}{6} & -\frac{2}{3} & \frac{1}{6} & \frac{1}{6} & 1 & \frac{1}{6} & -\frac{2}{3} & \frac{1}{6} & -\frac{2}{3} & \frac{1}{6} \\
-\frac{2}{3} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & 1 & -\frac{2}{3} & -\frac{2}{3} & \frac{1}{6} & \frac{1}{6} \\
\frac{1}{6} & \frac{1}{6} & -\frac{2}{3} & \frac{1}{6} & -\frac{2}{3} & -\frac{2}{3} & 1 & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} \\
\frac{1}{6} & -\frac{2}{3} & \frac{1}{6} & -\frac{2}{3} & \frac{1}{6} & -\frac{2}{3} & \frac{1}{6} & 1 & \frac{1}{6} & \frac{1}{6} \\
-\frac{2}{3} & \frac{1}{6} & \frac{1}{6} & -\frac{2}{3} & -\frac{2}{3} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & 1 & \frac{1}{6} \\
-\frac{2}{3} & -\frac{2}{3} & -\frac{2}{3} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & 1
\end{bmatrix}$$

G is the Gram matrix of a 2-angle tight frame in $\mathbb{R}^4$. If Q_1 and Q_2 are defined as

$$Q_1 = \begin{bmatrix} 0&0&0&0&0&1&0&0&1&1 \\ 0&0&0&0&1&0&0&1&0&1 \\ 0&0&0&1&0&0&1&0&0&1 \\ 0&0&1&0&0&0&0&1&1&0 \\ 0&1&0&0&0&0&1&0&1&0 \\ 1&0&0&0&0&0&1&1&0&0 \\ 0&0&1&0&1&1&0&0&0&0 \\ 0&1&0&1&0&1&0&0&0&0 \\ 1&0&0&1&1&0&0&0&0&0 \\ 1&1&1&0&0&0&0&0&0&0 \end{bmatrix} \quad \text{and} \quad Q_2 = \begin{bmatrix} 0&1&1&1&1&0&1&1&0&0 \\ 1&0&1&1&0&1&1&0&1&0 \\ 1&1&0&0&1&1&0&1&1&0 \\ 1&1&0&0&1&1&1&0&0&1 \\ 1&0&1&1&0&1&0&1&0&1 \\ 0&1&1&1&1&0&0&0&1&1 \\ 1&1&0&1&0&0&0&1&1&1 \\ 1&0&1&0&1&0&1&0&1&1 \\ 0&1&1&0&0&1&1&1&0&1 \\ 0&0&0&1&1&1&1&1&1&0 \end{bmatrix},$$

then

$$G = I - \frac{2}{3}Q_1 + \frac{1}{6}Q_2.$$

Since each row of Q_1 contains three 1s, this means that every vertex of the graph whose adjacency matrix is Q_1 has three adjacent vertices, hence this graph is regular. A similar statement is true for Q_2 and the graph associated with it. See Figure 5. In fact, the graphs associated to Q_1 and Q_2 are strongly regular as well by Theorem 8, but the purpose of this example is only to illustrate how one obtains graphs from k-angle tight frames and studies their structure.

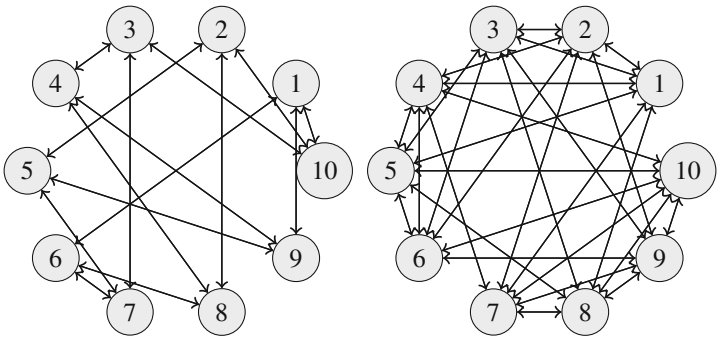


Fig. 5 The graphs corresponding to Q_1 and Q_2.

4.3 Frame Design Issues

Designing frames with good properties is important. This chapter has discussed an important class of frames called equiangular tight frames. When designing frames, it is important to know the kinds of transformations under which a given frame does not lose the properties it already possesses. For example, if U is a unitary transformation and if $\{f_i\}$ is a unit normed tight frame, then the set $\{Uf_i\}$ is also a unit normed tight frame. If $\{f_i\}$ is an ETF, then $\{Uf_i\}$ is also an ETF. Recall that for a tight frame the frame operator S has a nice structure, being a constant multiple of the identity. In other words, S is a diagonal matrix with all diagonal entries equal to A, the frame bound. Having such a structure is useful for computational reasons. In particular, it is convenient to invert S in the expansion formula (12), and the process is numerically stable. One can ask the following.

Challenge Problem 1. What are the operators M for which $\{Mf_i\}$ is a frame whose frame operator S is a diagonal matrix?

This is useful since diagonal matrices are also easy to invert and the frame $\{Mf_i\}$ can therefore offer advantages similar to that of a tight frame. Of course, there are other variations of the above question that can be considered.

In the context of erasures or losses, as discussed in Section 1, suppose that e number of frame coefficients are lost during transmission. If the index set of erasures is denoted by E, then to the receiver it is as if the signal is to be recovered using the frame $\{f_i\}_{i\notin E}$, where $|E| = e$, assuming that $\{f_i\}_{i\notin E}$ is a frame. In the presence of random noise, it has been shown [13] that when starting with a unit normed frame the mean-squared error is minimized if the remaining vectors $\{f_i\}_{i\notin E}$ form a tight frame. Unfortunately, as given in Theorem 4.3 in [13], it is not possible for *every* $\{f_i\}_{i\notin E}$ with $|E| = e$ to be tight. In this context, it might be interesting to investigate the following.

Research Project 3. Fix the number of erasures e such that $e \leq n-d$, where n represents the size of a frame and d is the dimension. Starting with a unit normed frame of n vectors in $\mathbb{F}^d$, remove e vectors and see for how many of the $\binom{n}{e}$ cases the set $\{f_i\}_{i\notin E}$ is a tight frame. Try this for different unit normed frames using a computer. Is it possible to characterize the starting frames that maximize the number of cases when $\{f_i\}_{i\notin E}$ is a tight frame?

4.4 Frame Algorithms

In order to reconstruct a signal v from the frame coefficients $\{\langle v, f_i\rangle\}_{i=1}^n$ one can use (12). However, that entails inverting the frame operator which can be complicated if the dimension is large. To avoid inverting S one can find successive approximations to f. This can be done using a well-known algorithm known as the *frame algorithm* [6]. This is given below in Lemma 4.

Lemma 4. *[6] Let $\{f_k\}_{k=1}^n$ be a frame for $\mathcal{V}$ with frame bounds A, B. Given $v \in \mathcal{V}$, define the sequence $\{g_k\}_{k=0}^\infty$ in $\mathcal{V}$ by*

$$g_0 = 0,\ g_k = g_{k-1} + \frac{2}{A+B} S(f - g_{k-1}), k \geq 1. \tag{15}$$

Then

$$\|f - g_k\| \leq \left(\frac{B-A}{B+A}\right)^k \|f\|.$$

It should be noted in Lemma 4 above that if B is much larger than A then the convergence is slow. There are acceleration algorithms based on the Chebyshev method and the conjugate gradient method [14] that improve the speed of convergence in (15). Other algorithms on approximating the inverse frame operator include work in [5, 20]. An interesting project might be the following.

Research Project 4. Compare the various existing algorithms for inverting a frame operator and study other techniques to improve existing methods.

4.5 Reconstruction in Presence of Random Noise

Assume that during transmission the frame coefficients get corrupted by some random noise so that what is received are the corrupted coefficients $\{\langle v, f_i\rangle + \eta_i\}_{i=1}^n$ where each η_i has mean zero and variance σ^2. Further, for $i \neq j$, η_i and η_j are uncorrelated. The reconstruction of the signal x from the noisy coefficients is done as follows:

$$\widehat{x} = \sum_{i=1}^n [\langle v, f_i\rangle + \eta_i] S^{-1} f_i = x + \sum_{i=1}^n \eta_i S^{-1} f_i.$$

Due to the assumptions,

$$E[\widehat{x}] = x.$$

Assuming an unbiased estimator, the mean-squared error (MSE) is the trace of the covariance matrix of $\widehat{x}$. It has been shown in [13] that for a single erasure (or lost coefficient), among all unit normed frames, tight frames minimize the MSE. It is also shown in [13] that under a single erasure from a unit normed frame, the MSE averaged over all erasures is minimized if and only if the starting frame is tight. In [13], a uniform distribution is taken for the erasure model, i.e., each coefficient is equally likely to be lost. This leads to the following.

Research Project 5. Study the effect of a general distribution for the erasure model, and investigate the properties of the starting frame that would minimize the MSE. As one example, how well do k-angle tight frames, as described in Section 3.1, perform in the face of erasures?

For more than two erasures, it is known that all tight frames do not minimize the two-erasure MSE. It has been shown in [16] that under two erasures the optimal frames are ones that are equiangular. So far, it has been assumed that when a frame vector has been erased the remaining vectors still form a frame. In this case, assuming that the location of the loss is known, the signal recovery is done using the frame operator of the new frame. However, it may be the case that eliminating just a single vector leaves a set that is no longer a frame. This consideration can lead to other areas of research.

It might also be interesting to investigate the effect of random noise or other random phenomena on equiangularity of an ETF.

Research Project 6. Starting from an ETF, estimate the deviations of the frame from being equiangular or being tight by adding random perturbations to the frame vectors.

Acknowledgements The authors would like to thank the anonymous reviewer for in-depth comments and helpful suggestions. The authors were partially supported by the NSF under Award No. CCF-1422252.

References

1. Barg, A., Glazyrin, A., Okoudjou, K., Yu, W.-H.: Finite two-distance tight frames. Linear Algebra Appl. **475**, 163–175 (2015)
2. Benedetto, J.J., Fickus, M.: Finite normalized tight frames. Adv. Comput. Math. **18**, 357–385 (2003)
3. Benedetto, J.J., Kolesar, J.D.: Geometric properties of Grassmannian frames for $\mathbb{R}^2$ and $\mathbb{R}^3$. EURASIP J. Adv. Signal Process. **2006**(1), 1–17 (2006)
4. Bodmann, B.G., Paulsen, V.I.: Frames, graphs and erasures. Linear Algebra Appl. **404**(1–3), 118–146 (2005)
5. Casazza, P.G., Christensen, O.: Approximation of the inverse frame operator and applications to Gabor frames. J. Approx. Theory **103**(2), 338–356 (2000)
6. Christensen, O.: An Introduction to Frames and Riesz Bases. Birkhäuser, Boston (2003)
7. Datta, S., Oldroyd, J.: Construction of *k*-angle tight frames. Numer. Funct. Anal. Optim. **37**(8), 975–989 (2016)
8. Datta, S., Oldroyd, J.: Low coherence unit norm tight frames. Linear Multilinear Algebra (to appear)
9. Daubechies, I.: Ten Lectures on Wavelets. SIAM, Philadelphia (1992)
10. Daubechies, I., Grossmann, A., Meyer, Y.: Painless nonorthogonal expansions. J. Math. Phys. **27**(5), 1271–1283 (1986)
11. Duffin, R.J., Schaeffer, A.C.: A class of nonharmonic Fourier series. Trans. Am. Math. Soc. **72**(2), 341–366 (1952)
12. Friedberg, S.H., Insel, A.J., Spence, L.E.: Linear Algebra, 4th edn. Prentice Hall, Inc. Upper Saddle River, NJ (2003)
13. Goyal, V.K., Kovačević, J., Kelner, J.A.: Communicated Henrique, Malvar, S.: Quantized frame expansions with erasures. Appl. Comput. Harmon. Anal. **10**, 203–233 (2001)
14. Grochenig, K.: Acceleration of the frame algorithm. IEEE Trans. Signal Process. **41**(12), 3331–3340 (1993)
15. Han, D., Kornelson, K., Larson, D., Weber, E.: Student Mathematical Library. Frames for Undergraduates, vol. 40. American Mathematical Society, Providence, RI (2007)
16. Holmes, R.B., Paulsen, V.I.: Optimal frames for erasures. Linear Algebra Appl. **377**, 31–51 (2004)
17. Horn, R.A., Johnson, C.R.: Matrix Analysis. Cambridge University Press, Cambridge (1990)
18. Kovačević, J., Chebira, A.: Life beyond bases: the advent of frames (Part I). IEEE Signal Process. Mag. **24**(4), 86–104 (2007)
19. Oldroyd, J.: Regular graphs and *k*-angle tight frames. (In preparation)
20. Song, G., Gelb, A.: Approximating the inverse frame operator from localized frames. Appl. Comput. Harmon. Anal. **35**(1), 94–110 (2013)
21. Strohmer, T., Heath, R.W. Jr.: Grassmannian frames with applications to coding and communication. Appl. Comput. Harmon. Anal. **14**(3), 257–275 (2003)
22. Sustik, M.A., Tropp, J.A., Dhillon, I.S., Heath, R.W. Jr.: On the existence of equiangular tight frames. Linear Algebra Appl. **426**(2–3), 619–635 (2007)
23. Waldron, S.: On the construction of equiangular tight frames from graphs. Linear Algebra Appl. **431**(11), 2228–2242 (2009)
24. Welch, L.R.: Lower bounds on the maximum cross correlation of signals. IEEE Trans. Inf. Theory **20**(3), 397–399 (1974)
25. Xia, P., Zhou, S., Giannakis, G.B.: Achieving the Welch bound with difference sets. IEEE Trans. Inf. Theory **51**(5), 1900–1907 (2005)
26. Young, R.M.: An Introduction to Nonharmonic Fourier Series. Pure and Applied Mathematics, vol. 93. Academic [Harcourt Brace Jovanovich Publishers], New York (1980)

Mathematical Decision-Making with Linear and Convex Programming

Jakob Kotas

Suggested Prerequisites. *Differential calculus, Linear algebra.*

1 Introduction

The Institute for Operations Research and the Management Sciences (INFORMS) defines operations research (OR) as "a discipline that deals with the application of advanced analytical methods to help make better decisions." It uses tools and techniques from mathematics, including modeling, statistics, data analysis, and optimization, to find a maximum (profit, yield) or minimum (cost, risk) solution to a problem, typically in the presence of one or more system constraints. OR as a discipline began during World War II with the study of military planning and resource allocation problems, and as such, has had an applied focus from its inception. Today, principles of OR are widely used in business practice, and OR is a well-established research field.

OR's application-driven focus lends itself excellently to a variety of practical research problems of interest to industry, from scheduling to allocation to management. At the same time, OR stands on rigorous mathematical footing, and students with a more theoretical bent will have no shortage of problems involving structural properties, uniqueness of solutions, and algorithms.

There are many classes of optimization problems that are of interest in OR applications. A non-exhaustive list follows:

- **Linear programming:** The problem of optimizing a linear objective function subject to linear equality and inequality constraints. The subject gained popular-

J. Kotas (✉)
University of Portland, 5000 N Willamette Blvd, Portland, OR 97203, USA
e-mail: kotas@up.edu

A. Wootton et al. (eds.), *A Primer for Undergraduate Research*, Foundations for Undergraduate Research in Mathematics, DOI 10.1007/978-3-319-66065-3_8

ity due to the development of the Simplex method, one of the first reliable and efficient optimization algorithms. (It should be noted that the word "program," in the OR context, is used to mean an optimization problem, and its use predates that of computer programs.)

- **Convex programming:** A generalization of linear programming, convex programming is the problem of optimizing a convex objective function subject to convex constraints. Many practical optimization problems can be formulated (or re-formulated) as convex problems. Again, efficient algorithms for the solution of convex problems have been developed, especially for problems of certain standard forms; thus, formulation of a problem as convex often leads to computational tractability.
- **Integer programming:** The problem of optimizing a linear objective function subject to linear equality and inequality constraints, in which some or all variables are constrained to be integer-valued. Integer programs are non-convex and are generally much more difficult to solve than linear programs.
- **Stochastic programming:** The problem of optimizing a system in which some or all constraints or parameters depend on random variables.
- **Dynamic programming:** A technique for solving an optimization problem by separating it into a collection of simpler sub-problems. Among other applications, they are useful for sequential decision-making, where the problem of reaching some optimal state in the future is broken down into a series of finite or countable individual problems at each time step.
- **Combinatorial/network optimization:** The problem of determining a discrete optimal object from a finite set of objects on a graph, a set of vertices with arcs connecting pairs of vertices. The famous Traveling Salesman problem of finding a minimum-distance route connecting a known set of points is one such example.

In applications, oftentimes combinations of these problem classes are used. However, in this short introduction, we focus on just the first two classes of problems: linear programs and convex programs.

In these problems, there are two different sorts of quantities of interest: parameters and decision variables. **Parameters** are typically known and are treated as constants for the purposes of solving the problem. **Decision variables** are the values that we seek. For example, in determining a shortest-distance route between two points on a map, the parameters would be the (fixed) distances of each road segment, which are known; the decision variable would be the sequence of segments that we decide to travel on. After the problem has been solved and we have a value for the decision variables, we are often interested in how our solution depends on the parameter values. In the example, how would our route change if any particular segment on the map had been shorter, or longer, or removed altogether? This procedure is called **sensitivity analysis.**

In this paper, we begin with linear programming, then move to the more general convex programming, explaining the theory, providing examples, and describing possible research ideas for both. We conclude with pointers to further reading as well as software tools for solving these problems.

2 Linear Programming

A linear program (LP) is a technique for optimization (minimization/maximization) of a linear objective function subject to linear equality and inequality constraints. Software packages exist for efficient solution of LPs, even in high dimensions with many variables and constraints. Thus, formulating a problem as a LP is often computationally advantageous. LPs have been used in many applications, including shift scheduling, network design, and manufacturing. We begin this section with an example, the diet problem. We then discuss a general formulation of LPs, as well as algorithms for their solution. Finally, we end with a discussion of duality.

2.1 Diet Problem

Let us illustrate the basic idea through an example. We wish to construct a daily diet with the minimum possible cost. The diet is selected from certain candidate foods and must satisfy certain nutritional requirements. To begin, we consider only two candidate foods: milk and bread; and four nutritional requirements: protein, carbs, vitamins, and sugar. The parameters are given in the following chart.

	Unit	Milk	Bread	Min Req.	Max Req.
Protein	gram	6	2	6	none
Carbs	gram	5	15	15	none
Vitamins	gram	1	1	2	none
Sugar	gram	0.6	1	none	3
Cost	$	0.3	0.2	none	none

Let m be the number of units of milk to consume, and b the number of units of bread; these are our decision variables. We seek m and b to minimize the total cost:

$$\min 0.3m + 0.2b \tag{1}$$

subject to the constraints:

$$\begin{cases} 6m + 2b \geq 6 \\ 5m + 15b \geq 15 \\ m + b \geq 2 \\ 0.6m + b \leq 3 \\ m \geq 0 \\ b \geq 0. \end{cases} \tag{2}$$

Here, the first, second, and third constraints correspond to the minimum requirements for protein, carbs, and vitamins, respectively. The fourth constraint corresponds to the maximum requirement for sugar. The fifth and sixth constraints are non-negativity constraints that ensure a physically meaningful result.

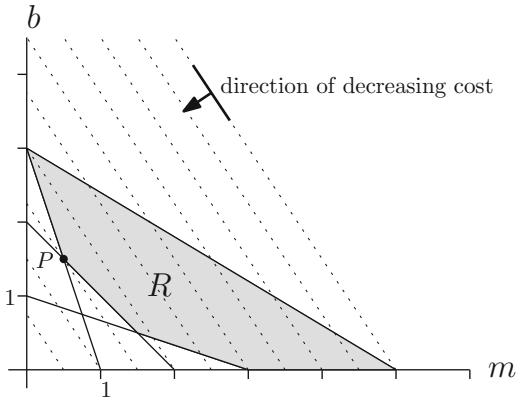


Fig. 1 Constraints (solid lines), feasible region (R), and lines of constant cost (dotted lines) for the diet problem.

Since we have only two variables, we can easily plot the constraints on a pair of axes. The region where all constraints are satisfied is known as the feasible region or feasible set, and points in the feasible region are called feasible points. On Figure 1, the feasible region is labeled R. On the same plot, we show dotted lines of constant cost: where $0.3m + 0.2b$ is a constant. By moving in a direction of decreasing cost, we can visually see that the lowest-cost point within the feasible region occurs at point P, at $(m, b) = (0.5, 1.5)$, which is also a vertex of R. The cost at point P is the value of the objective function there: $0.3(0.5) + 0.2(1.5) = 0.45$.

Upon viewing Figure 1, we make several observations. First, the feasible region will always be a convex polygon. (It may be bounded or unbounded.) In 2D, we can think of a convex polygon as one in which all interior angles are $\leq 180°$; a more general definition of convexity is discussed in section 3.1. In higher dimensions, this generalizes to a convex polytope.

Second, in this problem, the unique solution was a vertex. This will become important later when we discuss solution algorithms.

2.2 Standard Forms

We now consider a general minimization LP. To avoid trivialities, assume that the feasible region R is non-empty and that the objective function is bounded below within R. Then there exists an optimal solution which is a vertex of the feasible region. (A proof of this statement is given in section 2.6 of Bertsimas and Tsitsiklis [2]). If there are multiple, non-unique solutions, they occur at an entire edge (or face, in higher dimensions,) of R. Going back to the diet problem for a moment, this would have happened happen if the lines of constant cost were parallel with one of the three minimum requirements of Figure 1. Even in this case, there still exists a solution at some vertex (in 2 dimensions, the vertices on either end of a polygon edge). Since in a typical decision-making framework we only need one solution to implement, the issue of uniqueness is less important, and we may restrict our search for the solution to only vertices of the feasible region. This is a key motivation for the Simplex method, which will be discussed shortly.

Let $x \in \mathbb{R}^n$ be a vector containing the n decision variables and $c \in \mathbb{R}^n$ be a vector containing the parameters representing coefficients of each corresponding variable in the objective function. Assume the ith inequality constraint out of m total is of the form $\sum_{j=1}^{n} a_{ij}x_j \geq b_i$; then, all such constraints can be succinctly written in matrix-vector notation as $Ax \geq b$, where $A \in \mathbb{R}^{m \times n}$ is a matrix containing the coefficients of the constraint functions (including non-negativity constraints), $b \in \mathbb{R}^m$ is a vector of the constraint right-hand side values, and "$\geq$" is meant in the component-wise sense. (We will shortly show that this assumption is not restrictive.) If this is the case, then one general form of an LP is:

$$\begin{aligned} &\min c'x \\ &\text{subject to } Ax \geq b. \end{aligned} \tag{3}$$

We have denoted transpose by $'$; thus, $c'x = \sum_{i=1}^{n} c_i x_i$. Other constraints can be put into the form $\sum_{j=1}^{n} a_{ij}x_j \geq b_i$ as well: "$\leq$" inequality constraints of the form $a_{ij}x_j \leq b_i$ can be changed to greater-than through multiplication by -1, and "=" constraints of the form $a_{ij}x_j = b_i$ can be written as the pair of inequality constraints $a_{ij}x_j \geq b_i$ and $-a_{ij}x_j \geq -b_i$. Finally, any linear maximization problem $\max c'x$ is equivalent to $\min -c'x$.

The feasible region $R \subseteq \mathbb{R}^n$ is the set of points satisfying the constraints:

$$R = \left\{x \middle| Ax \geq b\right\}. \tag{4}$$

Geometrically when R exists it is a convex polytope.

While the general form (3) is geometrically intuitive, from an algorithmic standpoint, it is often more convenient to write the constraints in a slightly different way. Namely, we can equivalently define R of (4) as

$$R = \left\{x \middle| Ax = b, x \geq 0\right\}. \tag{5}$$

for a different A, x, and b. Namely, equality constraints of the form $a_{ij}x_j = b_i$ are left alone, while inequality constraints are converted to equality constraints via the introduction of so-called "slack" or "surplus" variables. We proceed by example. Consider the "$\leq$" constraint:

$$3x_1 + 4x_2 + 5x_3 \leq 10. \tag{6}$$

We introduce a new nonnegative variable x_4, called the slack variable. Equation (6) can then be written equivalently as a pair of equality and non-negativity constraints:

$$3x_1 + 4x_2 + 5x_3 + x_4 = 10, \quad x_4 \geq 0 \tag{7}$$

A "$\geq$" constraint can be handled in a similar way by introducing a so-called surplus variable. The final conversion is to eliminate free variables, that is, variables that are not restricted to be non-negative or non-positive. If x_6 is a free variable, we eliminate it and introduce two new non-negative variables x_7, x_8. The free variable x_6 can then be replaced with

$$x_6 = x_7 - x_8, \;\; x_7 \geq 0, \;\; x_8 \geq 0. \tag{8}$$

The LP with constraints written in the aforementioned form:

$$\begin{gathered} \min c'x \\ \text{subject to } Ax = b, \\ x \geq 0 \end{gathered} \tag{9}$$

is referred to as a "standard-form" LP.

Research Project 1. Develop a more realistic diet problem using nutritional data and formulate it as an LP. For example, Bosch compared several fast food chains to determine which, if any, offered a combination of entrees could meet federal dietary guidelines at lowest cost [3]. Bosch's formulation was an integer program, where one or more variables is restricted to be integer-valued. However, by relaxing this assumption, an analogous LP can be developed.

Research Project 2. Scheduling problems are a classic area of interest in OR. Develop a minimum-cost schedule for a company in which employees work on various tasks by formulating the problem as a LP. Constraints could include, but are not limited to, working hours for every individual employee; total working hours spent on each task; and penalizing shift changes to avoid frequent off-and-on times for each employee.

2.3 Solution of LPs

Many algorithms for solving LPs operate by first finding a feasible point and then iteratively moving in a direction of improving objective function value. Because the objective function is linear, any local optimum is also a global optimum; thus, a greedy algorithm will find an optimum eventually. Once a feasible solution has been found that moving in any direction would result in a worse objective function value, the algorithm terminates. Algorithms for LPs thus often have an easily understandable geometric interpretation. The main difficulty in solving LPs is that most problems of interest lie in a high-dimensional space, oftentimes with thousands (or more) of variables and constraints.

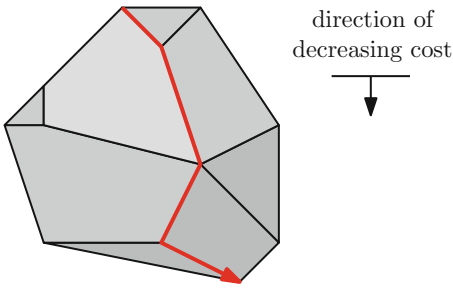


Fig. 2 A toy problem in 3 variables: the feasible region is a 3-dimensional convex polyhedron. The Simplex method, depicted with a red arrow, follows edges from vertex to vertex in a direction of improving (decreasing, for a minimization problem) objective function value.

In optimization, algorithmic computational complexity is often described in terms of strongly polynomial, weakly polynomial, or non-polynomial time. We assume basic arithmetic operations of addition, subtraction, multiplication, division, and comparison take one time step to perform. The number of operations in a weakly polynomial time algorithm is bounded by a polynomial in the number of constraints and variables. For an algorithm to be strongly polynomial time, additionally the memory used by the algorithm must be bounded by a polynomial in the number of constraints and variables. Certain algorithms including interior-point methods are weakly polynomial; however, no strongly polynomial method has yet been found for LPs. In fact, the existence of such an algorithm was listed on Stephen Smale's 18 open problems of mathematics for the 21st century [13]. Nevertheless, there is a fortunate disconnect between theoretical and practical notions of run time, in that algorithms exist for solving in LPs that are highly efficient for many practical problems of interest, despite the fact that they are not strongly (or in many cases, even weakly) polynomial in theory.

The first efficient algorithm for solving LPs was the Simplex method, developed by George Dantzig in the 1940s. The Simplex method has an intuitive geometrical interpretation and was widely used for several decades; even today, it lies at the heart of many LP solvers.

2.3.1 The Simplex Method

Without loss of generality, we consider a minimization problem. The Simplex method begins with a subroutine to find a vertex of the feasible region, if one exists. It then chooses the edge of the polytope with the fastest decrease (steepest descent) in objective function. It then moves the solution along that edge until hitting a vertex; then repeats. There are certain caveats to take into consideration at so-called degenerate points, where several constraints coincide at the same point, and when a tie for the steepest descent direction occurs, but the essential geometry is shown in Figure 2.

The run time is, in theory, exponential in the number of variables. Indeed, a worst-case scenario for an n-variable problem was found by Klee and Minty where the Simplex method visits every vertex of a perturbed n-dimensional hypercube, of which there are 2^n, before finding the optimal solution [8]. Nevertheless, the average behavior of the Simplex method on problems of interest seems to be significantly

better. Of course, we have not provided a rigorous definition of "average behavior"; indeed, the problem of defining such a concept and using it to define the run time of Simplex method remains somewhat open [1, 12].

Research Project 3. Define an appropriate notion of average-case behavior for the Simplex method. Computational experiments can be performed by generating random problems according to some probabilistic setup, and finding the run time of the Simplex method on each. Then, using regression, find the relationship between the mean and variance of the observed run times as a function of the problem size (number of variables + constraints).

2.3.2 Interior Point Methods

In contrast to the Simplex method, where the algorithm visits vertices of the feasible polytope by traveling along edges of the region, interior point methods travel through the interior of the feasible region. The basic idea in interior point methods is to choose a path that optimizes a combination of a reduction in objective value function and the distance from the edge of the feasible region. This is often done through the introduction of a barrier function: a function that is inversely proportional to the distance from the nearest edge of the polytope and thus carries low values in the center and approaches $+\infty$ on the edge itself. Let us introduce a scalar $\alpha \geq 0$ that indicates the relative importance of the barrier function as compared with the change in the objective value function $c'x$. By minimizing an appropriate sum of the objective function value and α times the barrier function, we find a path through the interior of the feasible region that depends on α. This path is counterintuitive, as the barrier function avoids the boundary of the feasible region, when it is known that all solutions lie on the boundary. However, by letting $\alpha \to 0$, we can then recover the optimum of the original objective function.

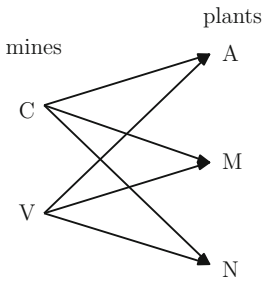
Interior point methods have been developed which are weakly polynomial-time in the number of variables. In practice they are often competitive with the Simplex method and similar edge-tracing algorithms, and for some sparse problems are significantly faster.

2.4 Transportation Problem

The following example illustrates another application that can be modeled as a LP. In order to produce a certain product, raw material must be transported from the mine to a plant where it is refined. Say a company has control over 2 mines in Colorado and Virginia, and 3 plants in Alabama, Minnesota, and New Mexico. We denote the mines C and V, and plants A, M, and N, respectively. Below is a table of the parameters representing estimated shipping costs between each plant and mine, in thousands of dollars per ton of material (Table 1).

Table 1 Cost of shipping from each mine to each plant, in thousands of $ per ton of ore.

	Plant A	Plant M	Plant N
Mine C	22	18	7
Mine V	14	20	24



For a certain week, the Colorado mine is expected to output 150 tons of ore, and the Virginia mine 130. To fulfill demand in the same week, the Alabama plant requires at least 88 tons of ore, the Minnesota plant 125 tons, and the New Mexico plant 55 tons.

Exercise 1.

(a) Formulate the problem as a linear program. Denote x_{CA} as the decision variable for the amount of ore shipped (in tons) from Colorado to Alabama, etc.
(b) A colleague suggests that it would be easiest if the Colorado mine exclusively supplied New Mexico and Alabama, and the Virginia mine exclusively supplied Minnesota (only enough to meet demand.) Check that this option is feasible. Find the total shipping cost (in thousands of dollars) in this case.
(c) Solve for all x values using an off-the-shelf linear program solver. What is the total shipping cost (in thousands of dollars) in this case? How much did we save as compared to part (b)?

Solution 1.

(a)

$$\min \quad 22x_{CA} + 18x_{CM} + 7x_{CN} + 14x_{VA} + 20x_{VM} + 24x_{VN}$$

$$\begin{aligned} \text{subject to} \qquad x_{CA} + x_{CM} + x_{CN} &\leq 150 \\ x_{VA} + x_{VM} + x_{VN} &\leq 130 \\ x_{CA} + x_{VA} &\geq 88 \\ x_{CM} + x_{VM} &\geq 125 \\ x_{CN} + x_{VN} &\geq 55 \\ x_{CA}, x_{CM}, x_{CN}, x_{VA}, x_{VM}, x_{VN} &\geq 0 \end{aligned}$$

(b)

$$x = \left[x_{CA}, x_{CM}, x_{CN}, x_{VA}, x_{VM}, x_{VN}\right]' = \left[88, 0, 55, 0, 125, 0\right]'$$

$$c'x = \left[22, 18, 7, 14, 20, 24\right]\left[88, 0, 55, 0, 125, 0\right]' = 4831$$

(c)

$$x = \left[x_{CA}, x_{CM}, x_{CN}, x_{VA}, x_{VM}, x_{VN}\right]' = \left[0, 95, 55, 88, 30, 0\right]'$$

$$c'x = \left[22, 18, 7, 14, 20, 24\right]\left[0, 95, 55, 88, 30, 0\right]' = 3927$$

This solution represents a savings of $4831 - 3927 = 904$ dollars.

Research Project 4. Formulate the problem of delivering electricity to consumers as a LP. Electricity must be delivered to meet demand and can come from a variety of sources, such as coal, natural gas, wind, and solar. Investigate the effect of a carbon tax on the solution by penalizing electricity coming from nonrenewable resources. Consider how future growth in a certain geographic area will affect the solution.

2.5 Duality

In calculus, students are typically taught the Lagrange multiplier method for optimizing functions subject to equality constraints. The key idea there is to introduce a new scalar parameter for each equality constraint (the multiplier), and reformulate the hard constraints as soft constraints additively combined with the original objective function (and scaled by the appropriate multiplier)– this quantity is the Lagrangian L. The new problem is to optimize L with no constraints. Assuming differentiability of L, this can be solved by equating the partial derivatives of L with zero. For the proper choice of the multipliers, the presence or absence of each hard constraint does not affect the optimal value of the objective function; thus, the optimal solution to the original constrained problem and the new unconstrained problem are the same.

Duality theory is a generalization of the Lagrange multiplier method that also accounts for inequality constraints. We again associate a multiplier with each constraint and seek a set of values for the multipliers such that the specific value of the constraint does not affect the optimal objective function value. We consider the standard-form LP of (9), which we call the "primal" problem. Let x^* be an optimal value of x, and assume x^* exists. We relax the standard-form problem by changing the hard constraint $Ax - b = 0$ to a soft one with the introduction of a length-m vector of multipliers λ. We then have the problem:

$$\begin{aligned} &\min\ c'x + \lambda'(b - Ax) \\ &\text{subject to } x \geq 0 \end{aligned} \tag{10}$$

Let g be the optimal value of this relaxed problem. Since g depends on λ, let us consider it to be a function: $g(\lambda)$. The relaxed problem is broader than the original problem in that if $Ax = b$, we recover the original problem, but there are also feasible points in which $Ax \neq b$. In other words, the feasible region of the relaxed problem R' contains the feasible region of the original problem R: $R \subseteq R'$. For this reason, the minimum value of the objective function in R' must be no larger than the minimum value in R.

$$g(\lambda) = \min_{x \geq 0} \Big(c'x + \lambda'(b - Ax) \Big) \leq c'x^* + \lambda'(b - Ax^*) = c'x^*. \tag{11}$$

Here, the inequality arises because x^* is a member of the set $x \geq 0$ and thus is a feasible solution to the primal problem, and the final equality is due to the fact that x^* satisfies $Ax^* = b$ because it again is a feasible solution to the primal problem. Thus, $g(\lambda)$ is a lower bound for the optimal cost $c'x^*$.

Now consider the unconstrained problem:

$$\max\ g(\lambda) \tag{12}$$

This problem gives us the tightest possible lower bound for the optimal cost $c'x^*$. This problem is known as the dual problem. Strong duality proves that the optimal cost of the dual problem is equal to the optimal cost $c'x^*$ of the primal problem (see Theorem 4.4 of Bertsimas and Tsitsiklis [2]). Continuing a bit further, we have

$$g(\lambda) = \min_{x \geq 0} \Big(c'x + \lambda'(b - Ax) \Big) = \lambda'b + \min_{x \geq 0} \Big((c' - \lambda'A)x \Big). \tag{13}$$

Noting that the quantity $(c' - \lambda'A)x$ can be made arbitrarily small unless $c' - \lambda'A \geq 0$, we restrict our search in the dual problem to $c' - \lambda'A \geq 0$. The dual problem then becomes, in its final form:

$$\begin{aligned} &\max\ \lambda'b \\ &\text{subject to } \lambda'A \leq c' \end{aligned} \tag{14}$$

One important feature of duality is that the dual of the dual of a primal problem is itself (see Theorem 4.1 of Bertsimas and Tsitsiklis [2]). Because of the correspondence of the solutions of the primal and dual problems, solvers can make use of both the primal and dual problems in searching for a solution. For example, a dual Simplex method can be developed that pivots on the vertices of the dual problem's feasible region.

3 Convex Programming

In this section we investigate the problem of minimizing convex problems, which are defined as minimizing a convex function subject to convex constraints. Formally, the problem is:

$$\begin{aligned} &\min \quad f_0(x) \\ &\text{subject to} \quad f_i(x) \leq b_i, \quad i = 1, \ldots, m \end{aligned} \tag{15}$$

where $x \in \mathbb{R}^n$ are the variables, $f_0 : \mathbb{R}^n \to \mathbb{R}$ is the objective function, and $f_i : \mathbb{R}^n \to \mathbb{R}, \ \ i = 1, \ldots, m$ are the constraint functions, and the objective and constraint functions are convex. We define convexity in the sections to follow.

In general, convex optimization problems do not have analytical solutions, but there do exist efficient algorithms for their solution. Convex programs are more general than linear programs and a large number of problems can be formulated as convex problems, though there are some tricks and such a formulation can often feel more of an art than a science.

In this section, we begin by defining convex sets and functions. We then give several well-known examples of classes of problems that are convex, before concluding with a mention of some software packages for solving convex problems.

3.1 Convex Sets

Whereas linear programs have feasible regions that are convex polytopes, convex programs have feasible regions that are general convex sets. A set $S \subseteq \mathbb{R}^n$ is convex if

$$\theta x + (1 - \theta)y \in S, \ \ \forall \ 0 \leq \theta \leq 1, \ \ x, y \in S. \tag{16}$$

In other words, for any two points in S, the line segment connecting them lies entirely within S. Some examples of convex sets are given here.

- Convex hull: The convex hull of a set S is the intersection of all convex sets containing S. Intuitively, the convex hull "fills in" the non-convex sections of the set, or could be thought of as stretching an elastic band around S. If $S \subseteq \mathbb{R}^n$ is a set of finite, discrete points $x_1, \ldots, x_k \in \mathbb{R}^n$, and $\theta_1, \ldots, \theta_k$ are constants such that $\sum_{i=1}^{k} \theta_i = 1$ and $\theta_i \geq 0 \ \ \forall i = 1, \ldots, k$, then the convex hull R is the set of points:

$$R = \left\{ x \in \mathbb{R}^n \middle| x = \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_k x_k \right\} \tag{17}$$

  which is also known as the convex combination of $x_1, x_2, \ldots, x_k$ (Figure 3).
- Ellipsoid: A Euclidean ellipsoid $R \subset \mathbb{R}^n$ centered at $x_c \in \mathbb{R}^n$ can be written as:

$$R = \left\{ x_c + Au \middle| ||u||_2 \leq 1 \right\} \tag{18}$$

  with $u \in \mathbb{R}^n$ and $A \in \mathbb{R}^{n \times n}$ and non-singular.

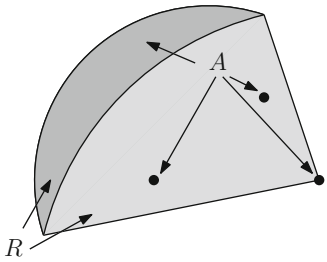


Fig. 3 A is a set consisting of the dark crescent along with three isolated points. R consists of the dark and light regions and is the convex hull of A.

- Hyperplanes and halfspaces: Take $x \in \mathbb{R}^n$, $a \in \mathbb{R}^n$ and $b \in \mathbb{R}$. A hyperplane $R \subset \mathbb{R}^n$ can be defined as the set of points

$$R = \left\{ x \middle| a'x = b, \;\; a \neq 0 \right\} \tag{19}$$

  for some a and b. Similarly, a halfspace can be written as the set of points

$$R = \left\{ x \middle| a'x \leq b, \;\; a \neq 0 \right\}. \tag{20}$$

  for some a and b.
- Convex polytope: Take $x \in \mathbb{R}^n$, $b \in \mathbb{R}^m$, and $A \in \mathbb{R}^{m \times n}$. As discussed in the linear programming section, a convex polytope $R \subset \mathbb{R}^n$ is the set of points:

$$R = \left\{ x \middle| Ax \geq b \right\}. \tag{21}$$

Convex sets have many important properties, but perhaps the most important is that the intersection of (even countably many) convex sets is convex. This fact ensures that adding more convex constraints will still result in a convex program.

3.2 Convex Functions

A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if its domain is a convex set and:

$$f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta) f(y), \quad 0 \leq \theta \leq 1 \tag{22}$$

Further, f is strictly convex if its domain is a convex set and:

$$f(\theta x + (1-\theta)y) < \theta f(x) + (1-\theta) f(y), \quad 0 \leq \theta \leq 1 \tag{23}$$

Clearly, all strictly convex function are convex, but the reverse is not true. The graphical interpretation follows directly from the definition and is illustrated through Figures 4 and 5.

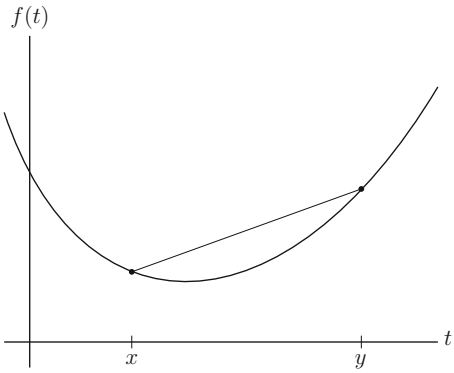


Fig. 4 $f(t)$ is a strictly convex function. For any x,y in the domain of $f(t)$, the line segment connecting $f(x)$ and $f(y)$ is strictly above the function $f(t)$ on $x < t < y$.

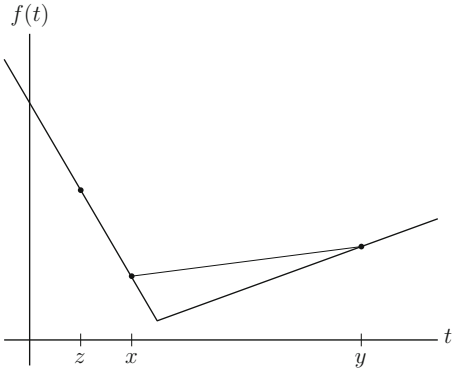


Fig. 5 $f(t)$ is a convex, but not strictly convex function. For any x,y the line segment connecting $f(x)$ and $f(y)$ is strictly above the function $f(t)$ on $x < t < y$; however this is not the case for x and z, where the line segment connecting $f(x)$ and $f(z)$ lies directly on the function $f(t)$.

Some examples of convex functions follow.

- Linear-affine: $f(x) = ax + b$ on $\mathbb{R}$ for any $a, b \in \mathbb{R}$
- Exponential: e^{ax} on $\mathbb{R}$ for any $a \in \mathbb{R}$
- Power: x^a on $x > 0$ for $a \geq 1$ or $a \leq 0$
- Negative logarithm: $-\log(ax)$ on $x > 0$ for $a > 0$

If a function f is twice differentiable with open domain D, then it is convex if and only if

$$\nabla^2 f(x) \geq 0 \ \ \forall x \in D. \tag{24}$$

Here $x \in \mathbb{R}^n$ and $\nabla^2 f(x)$ denotes the Hessian

$$\nabla^2 f(x)_{ij} = \frac{\partial^2 f(x)}{\partial x_i \partial x_j}, \ \ i,j = 1, \ldots, n \tag{25}$$

and "$\geq$" is meant in the sense of a generalized inequality; in this case, that $\nabla^2 f(x)$ is positive semi-definite (psd) (Figure 6).

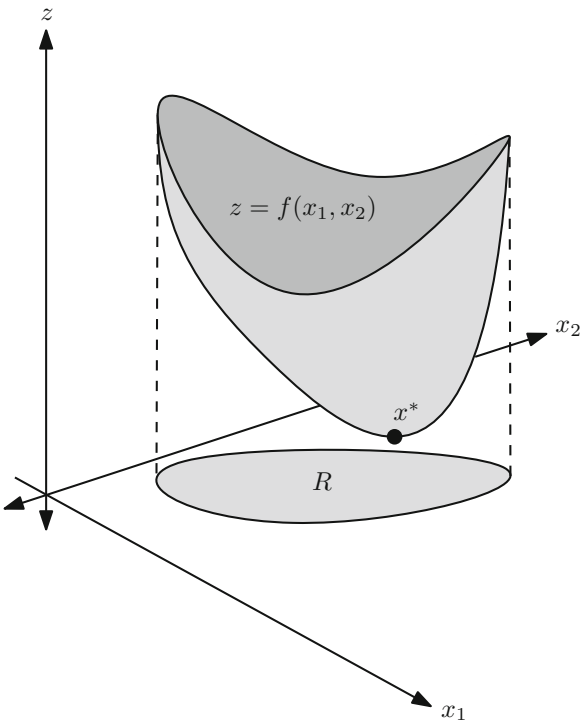


Fig. 6 A convex function $f : \mathbb{R}^2 \to \mathbb{R}$. The feasible region is R; the minimum of f over R is x^*.

We can also show that f is convex if it can be obtained from simple convex functions by convexity-preserving operations, such as

- Nonnegative weighted sum of convex functions
- Pointwise maximum of convex functions
- Composition of convex functions

3.3 Classes of Convex Programs

Many solvers take advantage of certain structural properties of the convex program in question. Here we list some important classes of convex programs seen in applications.

- Linear program (LP): A linear objective function subject to linear equality and inequality constraints; the feasible region is a convex polytope. Discussed previously.
- Quadratic program (QP): A quadratic objective function subject to linear equality and inequality constraints; the feasible region is therefore still a convex polytope. The general form of a QP is:

$$\begin{aligned} &\min \ \tfrac{1}{2}x'Px + q'x + r \\ &\text{subject to } Gx \leq h, \ \ Ax = b \end{aligned} \tag{26}$$

where the decision variable is $x \in \mathbb{R}^n$, P is a psd matrix: $P \in \mathbb{S}^n_+$, $q \in \mathbb{R}^n$, and $r \in \mathbb{R}^n$. Here we have explicitly separated the inequality constraints and equality constraints; the m inequality constraints are contained within $Gx \leq h$ where $G \in \mathbb{R}^{m \times n}$ and $h \in \mathbb{R}^m$. The p equality constraints are contained within $Ax = b$ where $A \in \mathbb{R}^{p \times n}$ and $b \in \mathbb{R}^p$. P is required to be psd in order for the objective function to be convex, as $\nabla^2(x'Px + q'x + r) = P$.

Exercise 2. Least-squares optimization

Formulate the linear least-squares approximation problem in n variables with $m > n$ data points as a QP.

Solution 2. If A is an $m \times n$ matrix containing the input data points and b is a $m \times 1$ vector containing the output data points, then the problem is:

$$\min ||Ax - b||_2^2 = x'A'Ax - 2b'Ax + b'b. \tag{27}$$

The objective function here is quadratic in x, so we must show that $A'A$ is psd. A square matrix $M \in \mathbb{R}^{p \times p}$ is psd if $x'Mx \geq 0$ for all $x \in \mathbb{R}^p$. For any x, we have

$$x'(A'A)x = (Ax)'(Ax) = ||Ax||^2 \geq 0. \tag{28}$$

Thus, A is psd and so the objective function is convex. Furthermore, the constraints are trivially linear as there are none. Therefore, this is a QP. We can thus also consider the constrained least-squares problem, in which values of x must lie in an interval $l \leq x \leq u$; it is also a QP:

$$\begin{aligned} &\min \ ||Ax - b||_2^2 = x'A'Ax - 2b'Ax + b'b \\ &\text{subject to } l_i \leq x_i \leq u_i, \ \ i = 1, \ldots, n \end{aligned} \tag{29}$$

Exercise 3. Portfolio optimization

This example is from section 4.7.6 of Boyd and Vandenberghe [4]. Consider the problem of optimizing two criteria in assembling a financial portfolio: maximize the mean return and minimize the variance of the return. Let x be a vector containing the fractions of each of four possible assets. Each asset's mean and standard deviation of return is given as:

Asset	Mean Return	Std.Dev. of Return
1	12%	20%
2	10%	10%
3	7%	5%
4	3%	0%

Furthermore, the correlation coefficients between assets are: $\rho_{12} = 30\%$ and $\rho_{13} = -40\%$; other pairs zero. Thus, the correlation matrix Σ is:

$$\Sigma = \begin{bmatrix} 0.2^2 & 0.3 \times 0.2 \times 0.1 & -0.4 \times 0.2 \times 0.05 & 0 \\ 0.3 \times 0.2 \times 0.1 & 0.1^2 & 0 & 0 \\ -0.4 \times 0.2 \times 0.05 & 0 & 0.05^2 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

We define the mean return vector to be p: $p = \begin{bmatrix} 0.12, 0.1, 0.07, 0.3 \end{bmatrix}'$, and we define a scaling factor μ that accounts for the relative importance of minimizing the variance of the return against maximizing the mean return. Formulate the problem as a quadratic program and solve.

Solution 3. The problem is formulated as:

$$\begin{aligned} &\min \ -p'x + \mu x' \Sigma x \\ &\text{subject to } 1'x = 1, \quad x \geq 0. \end{aligned} \tag{30}$$

Here 1 represents a vector the same length as x with all entries being 1. Due to the form of the objective function, this is a quadratic program with linear constraints. The solution for various values of μ is graphed below, showing explicitly the tradeoff between the standard deviation of return versus the mean return (Figure 7).

Research Project 5. Chapter 16 of Nocedal and Wright [11] delves into algorithms for QPs. Similar to the Simplex method, looking deeper into the run time of these algorithms for sample problems could provide an avenue to more theoretical projects.

- Quadratically-constrained quadratic program (QCQP): A quadratic objective function subject to convex quadratic constraints. The general form of a QCQP is:

$$\begin{aligned} &\min \ \tfrac{1}{2}x'P_0x + q_0'x + r_0 \\ &\text{subject to } \tfrac{1}{2}x'P_ix + q_i'x + r_i \leq 0, \ \ i = 1, \ldots, m \\ &\qquad Ax = b \end{aligned} \tag{31}$$

with $P_i \in \mathbb{S}_+^n \ \ \forall \ \ i = 0, \ldots, m$.

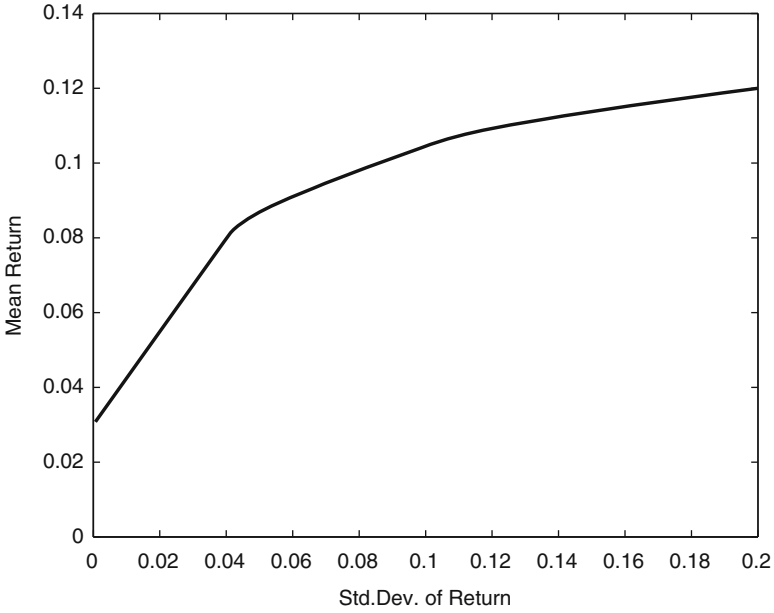


Fig. 7 Tradeoff between standard deviation and mean of the return.

- Second-order cone program (SOCP): A linear objective function subject to so-called second-order cone constraints. The general form of a SOCP is:

$$\begin{aligned} &\min f'x \\ &\text{subject to } ||A_i x + b_i||_2 \leq c_i' x + d_i, \quad i = 1, \ldots, m \\ &\qquad\qquad Fx = G \end{aligned} \tag{32}$$

with $A_i \in \mathbb{R}^{n_1 \times n}$ and $F \in \mathbb{R}^{p \times n}$. SOCPs are a generalization of QCQPs and LPs. QCQPs can be turned into SOCPs by reformulating the objective function as a constraint.

Research Project 6. One problem from the area of robotics and optimal control is grasping a rigid body with robot fingers. To do so, we must determine the amount of force each finger shall exert. Lobo et al. [9] describe a formulation of this problem as a SOCP which takes into account friction and equilibrium constraints and limits on contact forces. We may simply be interested in whether the object can be grasped at all: this amounts to a feasibility problem, where we simply determine whether or not the feasible region is non-empty. If a solution does exist, we can investigate a variety

(continued)

of problems, such as finding the gentlest grip in some sense, or finding a grip which has the smallest difference in forces on each finger. Formulate the problem with a given number of fingers and a given object. This problem could also incorporate data-gathering, namely the physical properties of the object to be grasped.

Research Project 7. Section 8.8 of Boyd and Vandenberghe [4] describes a floor-planning problem, where we seek the minimum perimeter fence to bound a set of rectangular objects of known dimension. Variants of the problem, including having a minimum spacing between the objects and allowing the dimensions (but not area) of the objects to vary, are considered. These can be formulated as SOCPs (where the $||\cdot||_2$ constraint corresponds to a Euclidean distance) or in some cases even LPs depending on the variant. Using this as a starting point, optimal packing problems can be considered for boxes or other shapes. This application has uses in shipping and transport problems.

- Semi-definite program (SDP): One general form of an SDP is:

$$\begin{aligned} &\min\ c'x \\ &\text{subject to } x_1F_1 + x_2F_2 + \ldots + x_nF_n + G \geq 0 \\ &\qquad Ax = b \end{aligned} \tag{33}$$

with $F_i, G \in \mathbb{S}^k, i = 1, 2, \ldots, n$. The first constraint is known as a linear matrix inequality (LMI) constraint; since the left hand side is a $k \times k$ matrix, the "$\geq$" here is a generalized inequality meaning that $x_1F_1 + x_2F_2 + \ldots + x_nF_n + G$ is psd. If $F_1, \ldots, F_n, G$ are all diagonal, then this formulation reduces to a linear program. SDPs are a generalization of SOCPs, as the SOCP constraints can be written as LMIs.

Research Project 8. Weinberger and Saul have formulated learning algorithms with applications to image processing as SDPs. Such algorithms can recognize characters in handwritten text, or identify whether faces from different image are the same person's, even from different angles. Projects could be developed to identify a range of objects in images [14, 15].

Research Project 9. An abundance of SDP applications can be found in the "Handbook of Semidefinite Programming" by Wolkowicz, Saigal, and Vandenberghe [16]. Projects could involve extending these applications and/or formulating new problems as SDPs.

3.4 Solvers

Many software packages exist to solve convex problems. One convenient solver for solving convex problems of moderate size (including LPs) is CVX, [5] which can be downloaded and used as a Matlab® package. Other solvers include Gurobi [6] and CPLEX, [7] all of which are free for academic use, as well as Matlab's Optimization Toolbox™ [10].

4 Concluding Remarks

Optimization is a powerful tool for solving many applied problems of interest to operations research. In this brief chapter we discussed linear programming, followed by the more general convex programming and specific forms therein. Many of these classes of problems have efficient algorithms for their solution, even in high dimensions; thus, formulation of an optimization problem in one of these forms often results in greatly improved computational tractability. For the undergraduate student, there are many open problems that are application-based. In addition, delving into the inner workings of algorithms for generic problems could provide an avenue to interesting projects.

4.1 Further Reading

Parts of this chapter were adapted from the textbooks "Introduction to Linear Optimization" by Bertsimas and Tsitsiklis, [2] and "Convex Optimization" by Boyd and Vandenberghe [4]. Another comprehensive text is Nocedal and Wright's "Numerical Optimization." [11].

References

1. Adler, I., Megiddo, N., Todd, M.J.: New results on the average behavior of simplex algorithms. Bull. Am. Math. Soc. **11**(2), 378–382 (1984)
2. Bertsimas, D., Tsitsiklis, J.N.: Introduction to Linear Optimization. Athena Scientific, Belmont (1997)
3. Bosch, R.A.: The battle of the burger chains: which is best, burger king, mcdonald's, or wendy's? Socio Econ. Plan. Sci. **30**(3), 157–162 (1996)

4. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, Cambridge (2004)
5. Grant, M., Boyd, S.: CVX: Matlab software for disciplined convex programming, version 2.1. http://cvxr.com/cvx, Mar (2014)
6. Gurobi optimization. http://www.gurobi.com/products/gurobi-optimizer
7. IBM ILOG cplex optimization studio community edition. https://www-01.ibm.com/software/websphere/products/optimization/cplex-studio-community-edition
8. Klee, V., Minty, G.J.: How Good is the Simplex Algorithm? Defense Technical Information Center (1970)
9. Lobo, M.S., Vandenberghe, L., Boyd, S., Lebret, H.: Applications of second-order cone programming. Linear Algebra Appl. **284**, 193–228 (1998)
10. MathWorks.: Matlab optimization toolbox. http://www.mathworks.com/products/optimization
11. Nocedal, J., Wright, S.J.: Numerical Optimization, 2nd edn. Springer, Berlin (2006)
12. Shamir, R.: The efficiency of the simplex method: a survey. Manag. Sci. **33**(3), 301–34 (1987)
13. Smale, S.: Mathematical problems for the next century. Math. Intell. **20**(2), 7–15 (1998)
14. Weinberger, K.Q., Saul, L.K.: Unsupervised learning of image manifolds by semidefinite programming. Proc. 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (2004)
15. Weinberger, K.Q., Saul, L.K.: Distance metric learning for large margin nearest neighbor classification. Machine Lear. Res. **10**, 207–244 (2009)
16. Wolkowicz, H., Saigal, R., Vandenberghe, L. (eds.): Handbook of Semidefinite Programming. Kluwer Academic Publishers, Boston (2000)

Computing Weight Multiplicities

Pamela E. Harris

Suggested Prerequisites. *Group theory and introductory linear algebra.*

1 Introduction

Representation theory is a subject at the intersection of abstract and linear algebra. It uses tools from both fields to understand how elements of algebraic structures, such as groups or algebras, behave as linear transformations between vector spaces. The challenge in working in this field is the daunting amount of background and definitions necessary to understand the questions of interest. However, it is often the case that these questions can be boiled down and presented in the language of combinatorics. This is the approach we take to present the weight multiplicity computations involved in the representation theory of Lie algebras.

Computing weight multiplicities, which is equivalent to determining the dimension of a vector subspace, is central to the study of the representation theory of Lie algebras. In the language of combinatorics, the weight multiplicity computation is determined by finding the number ways we can express a finite set of vectors (called weights) as nonnegative integral sums of a fixed set of vectors (called positive roots). This is analogous to finding the number of ways to express a positive integer as a sum of positive integers. For example, the integer 5 can be written as a sum of positive integers in the following seven ways:

$$5,\ 4+1,\ 3+2,\ 3+1+1,\ 2+2+1,\ 2+1+1+1,\ 1+1+1+1+1.$$

P.E. Harris (✉)
Department of Mathematics and Statistics, Williams College, Williamstown, MA 01267, USA
e-mail: pamela.e.harris@williams.edu

A. Wootton et al. (eds.), *A Primer for Undergraduate Research*, Foundations for Undergraduate Research in Mathematics, DOI 10.1007/978-3-319-66065-3_9

Each of the above expressions are referred to as partitions of 5. In the Lie algebra setting, the set of positive roots plays the role of the positive integers and a weight plays the role of the integer we are partitioning. For example, in the Lie algebra $\mathfrak{sl}_3(\mathbb{C})$ (defined in Section 2.2) the set of positive roots is denoted by

$$\Phi^+ = \{\alpha_1, \alpha_2, \alpha_1 + \alpha_2\}$$

and the question of interest is: *Given* $n, m \in \mathbb{N} := \{0, 1, 2, 3, \ldots\}$ *in how many ways can we express* $n\alpha_1 + m\alpha_2$ *as a nonnegative integral sum of positive roots?* The answer is $\min(n, m)+1$, as the number of distinct ways to express $n\alpha_1+m\alpha_2$ as a sum of positive roots depends entirely on the number of times we use the positive root $\alpha_1 + \alpha_2$. Since we can use the positive root $\alpha_1 + \alpha_2$ from 0 to $\min(n, m)$ times we reach the desired result.

The connection between the above combinatorial question and computing weight multiplicities comes from the use of Kostant's weight multiplicity formula [21]:

$$m(\lambda, \mu) = \sum_{\sigma \in W} (-1)^{\ell(\sigma)} \wp(\sigma(\lambda + \rho) - (\mu + \rho)) \tag{1}$$

where the function $\wp$ counts the number of ways to express $\sigma(\lambda + \rho) - (\mu + \rho)$ as a nonnegative integral sum of positive roots. As it stands, Equation (1) is involved and the main roadblock is understanding all of the moving pieces. We now provide working definitions of the objects in this formula and present the technical definitions in Section 2.

We begin by noting that $m(\lambda, \mu)$ is read as the multiplicity of the weight μ in the finite-dimensional complex irreducible representation of a simple Lie algebra with dominant integral highest weight λ. This multiplicity represents a dimension of a very special vector subspace associated to μ called a weight space, which can be thought of as a generalized eigenspace. Although, the weights μ and λ are linear functionals (functions from a vector space to its field of scalars) we treat them as vectors that we express in two important bases: the simple root basis and the fundamental weight basis (more on this in Section 2). The importance of λ is that there is a bijection between dominant integral weights λ, which are nonnegative integral sums of fundamental weights, and finite-dimensional irreducible representations of a simple Lie algebra. This result is known as the theorem of the highest weight.

We now describe the finite group W indexing the sum in Equation (1), which is called the Weyl group. In the Lie algebra $\mathfrak{sl}_{r+1}(\mathbb{C})$, which we study in this paper, the Weyl group is isomorphic to $\mathfrak{S}_{r+1}$ the symmetric group on $r + 1$ letters. More generally, the Weyl group is defined as a group generated by reflections about hyperplanes that lie perpendicular to the simple roots $\alpha_1, \ \alpha_2, \ \ldots, \ \alpha_r$, which are identified as vectors in $\mathbb{R}^{r+1}$. When we specialize $r = 2$ and consider the Lie algebra $\mathfrak{sl}_3(\mathbb{C})$, we can visualize the roots and the generators of the Weyl group, denoted s_1 and s_2, as presented in Figure 1. Then the elements of the Weyl group

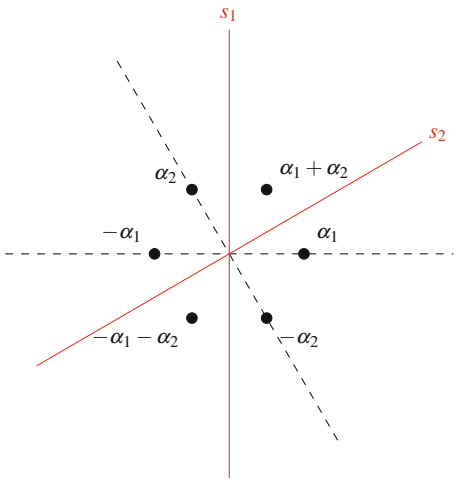


Fig. 1 Roots of $\mathfrak{sl}_3(\mathbb{C})$ along with the hyperplanes perpendicular to the simple roots.

are concatenations of the reflections s_1 and s_2, that is, words in these letters. Given that $W \cong \mathfrak{S}_3$ and since the adjacent transpositions $(1\ 2)$, $(2\ 3)$ generate $\mathfrak{S}_3$, we can identify s_1 with $(1\ 2)$ and s_2 with $(2\ 3)$ to generate all of the elements of the Weyl group: $1,\ s_1,\ s_2,\ s_1s_2,\ s_2s_1,\ s_1s_2s_1$, where 1 denotes the empty word, which is the identity of W. Given $\sigma \in W$ we let $\ell(\sigma)$ represent the smallest number k so that σ is a product of k reflections. For example, $\ell(s_1) = 1$ as s_1 is a minimal expression for this Weyl group element, but $\ell(s_1s_1) = 0$ as $s_1s_1 = 1$, which has length zero.

Lastly, we need to know how to compute $\sigma(\lambda+\rho)-(\mu+\rho)$ for $\sigma \in W$, where ρ is defined as the half sum of the positive roots. Since the reflections perpendicular to the simple roots generate all of the elements of W it is enough to know explicitly how the generators act on $\lambda+\rho$. In the Lie algebra $\mathfrak{sl}_{r+1}(\mathbb{C})$, if $s_1,\ s_2,\ s_3,\ \ldots,\ s_r$ denote the reflections perpendicular to the simple roots $\alpha_1,\ \alpha_2,\ \alpha_2,\ \ldots,\ \alpha_r$, respectively, then

$$s_i(\alpha_j) = \begin{cases} \alpha_j & \text{if } |i-j| > 1 \\ -\alpha_j & \text{if } i = j \\ \alpha_i + \alpha_j & \text{if } |i-j| = 1. \end{cases}$$

The action of any other Weyl group element uses the fact that the generators act linearly. That is, for any $r \in \mathbb{N}$ and constants $c_1, c_2, \ldots, c_r$ all elements $\sigma \in W$ satisfy

$$\sigma(c_1\alpha_1 + c_2\alpha_2 + \cdots + c_r\alpha_r) = c_1\sigma(\alpha_1) + c_2\sigma(\alpha_2) + \cdots + c_r\sigma(\alpha_r).$$

For example, in the Lie algebra $\mathfrak{sl}_3(\mathbb{C})$ we know $\rho = \frac{1}{2}\sum_{\alpha\in\Phi^+}\alpha = \alpha_1+\alpha_2$, and if $\sigma = s_1s_2$, $\lambda = 2\alpha_1+\alpha_2$, and $\mu = 0$, then we can compute

$$\begin{aligned} s_1s_2(\lambda+\rho)-(\mu+\rho) &= s_1s_2(3\alpha_1+2\alpha_2)-(\alpha_1+\alpha_2) \\ &= 3s_1s_2(\alpha_1)+2s_1s_2(\alpha_2)-(\alpha_1+\alpha_2) \\ &= 3s_1(\alpha_1+\alpha_2)+2s_1(-\alpha_2)-(\alpha_1+\alpha_2) \\ &= 3[s_1(\alpha_1)+s_1(\alpha_2)]-2s_1(\alpha_2)-(\alpha_1+\alpha_2) \\ &= 3[(-\alpha_1)+(\alpha_1+\alpha_2)]-2(\alpha_1+\alpha_2)-(\alpha_1+\alpha_2) \\ &= -3\alpha_1, \end{aligned}$$

which cannot be written as a nonnegative integral sum of positive roots, hence $\wp(s_1s_2(\lambda+\rho)-\rho)=0$.

Exercise 1. In the Lie algebra $\mathfrak{sl}_3(\mathbb{C})$ verify that if $n, m \in \mathbb{N}$, then

$$s_1s_2s_1(n\alpha_1+m\alpha_2) = s_2s_1s_2(n\alpha_1+m\alpha_2).$$

Having fully described the moving pieces of Equation (1) we now consider the types of complications that arise in weight multiplicity computations. First, the number of terms in the computation is determined by the order of the Weyl group. At first glance this may not seem like a problem, as the Weyl group is a finite group. However, we have remarked that the Weyl group of the Lie algebra $\mathfrak{sl}_{r+1}(\mathbb{C})$ is isomorphic to $\mathfrak{S}_{r+1}$ and, hence, the number of terms in Equation (1) is $(r+1)!$, which grows extremely fast. Second, we must compute the value of Kostant's partition function, a function for which no general closed form is known. Thus, we have reached a point in which not only do we have a factorial number of terms in the sum, but we also do not have a closed formula for computing the value of each term.

These complications may dishearten even the most enthusiastic representation theorist. Luckily for us, there are ways in which we can discover new results in this area by focusing on two types of problems. The first is motivated by the observation that in practice most terms appearing in the weight multiplicity formula actually contribute a value of zero. This means that for many $\sigma \in W$ it is not possible to express $\sigma(\lambda+\rho)-(\mu+\rho)$ as a nonnegative integral sum of positive roots. Hence, one can work on describing and enumerating the elements of the Weyl group that contribute a nonzero value to $m(\lambda,\mu)$ for fixed weights λ and μ. This leads to numerous open problems. Second, once we can reduce the sum to only the terms that contribute nontrivially, i.e. not a value of zero, we can focus on finding closed formulas for the partition function. In light of this, the objective of this paper is to present techniques to approach the computation of weight multiplicities from this point of view.

This work is organized as follows: Section 2 provides the technical definitions and the necessary background to begin our study. This background section assumes the reader has a working knowledge of linear algebra, but no familiarity with Lie algebras or their representations. Hence, this section is quite technical and some

readers may decide to begin their study in Section 3, which is dedicated to learning through examples, and return to the background section as definitions and concepts are needed. The detailed exposition of Section 3 illustrates techniques one may use to begin investigating the open problems provided in Section 4.

2 Background

This section provides a short historical account of the representation theory of simple Lie algebras along with the technical background and definitions needed to study this field. The concepts presented in this section are technical and may require multiple reads to digest all that is presented. However, these concepts are presented here so that this article is self contained. The reader is encouraged to skip ahead to Section 3 (Learning from examples) which limits the study to the Lie algebra $\mathfrak{sl}_{r+1}(\mathbb{C})$, and return to this section as needed for the technical definitions and concepts.

2.1 History

The representation theory of Lie algebras has been understood for quite some time due to the work of Élie Cartan [6], Hermann Weyl [24], and Harish-Chandra who "...develop[ed] a general algebraic theory of the irreducible representations of (a Lie algebra) $\mathfrak{g}$" [12, 23]. Their collective work showed that the representations of a semisimple Lie algebra $\mathfrak{g}$ can be analyzed by choosing a Cartan subalgebra $\mathfrak{h}$. Then the structure of a finite-dimensional irreducible representation $\tau : \mathfrak{g} \to \mathfrak{gl}(V)$ of a Lie algebra $\mathfrak{g}$ on the vector space V can be studied by decomposing the vector space V into the direct sum of subspaces:

$$V = \oplus V_\alpha. \tag{2}$$

This direct sum is indexed via a finite set of eigenvalues α called weights, and the corresponding eigenspace V_α is called a weight space. The multiplicity of a weight appearing in (2) is defined to be the dimension of the weight space V_α. This theory reduces the analysis of representations to determining which weights occur in (2) and with what multiplicity.

To compute this multiplicity, we use the theorem of the highest weight, which states that every irreducible representation of a semisimple Lie algebra $\mathfrak{g}$ arises as a highest weight representation with highest dominant integral weight λ, which we denote $L(\lambda)$. The converse holds true: All highest weight representations are irreducible representations of $\mathfrak{g}$. This theorem provides a one-to-one correspondence between the finite-dimensional complex irreducible representations of a semisimple Lie algebra and the set of dominant integral weights of $\mathfrak{g}$.

We now use Kostant's weight multiplicity formula to compute the multiplicity of the weight μ in the representation $L(\lambda)$, which we denote by $m(\lambda, \mu)$ [21]:

$$m(\lambda, \mu) = \sum_{\sigma \in W} (-1)^{\ell(\sigma)} \wp(\sigma(\lambda + \rho) - (\mu + \rho)), \tag{1}$$

where $\rho = \frac{1}{2}\sum_{\alpha\in\Phi^+}\alpha$, W is the Weyl group, $\ell(\sigma)$ is the length of σ, and $\wp$ denotes Kostant's partition function. The value of the partition function $\wp(\xi)$ is computed by counting the number of ways the weight ξ can be written as a nonnegative integral combination of positive roots. Combinatorics will play a role in finding the value of this partition function.

We remark that even though (1) exists to compute weight multiplicities, its implementation can be extremely difficult. This is due to the fact that the order of the Weyl group grows factorially as the rank of the Lie algebra increases. Also there is a lack of general closed formulas for the value of the partition function involved. Motivated by these complications, the author's previous work has focused on finding concrete descriptions of *Weyl alternation sets*, which are subsets of the Weyl group that contribute nonzero terms to the sum in (1), see [13–15, 17]. Additionally, the author, with collaborators Insko and Omar, computed closed formulas for the value of the partition function on the highest root [16].

Other work in the literature connects the value of the partition function to volume computations for polytopes and includes vector partition formulas for multiplicities of $\mathfrak{sl}_r(\mathbb{C})$ and generating functions for weight multiplicities in the rank 2 Lie algebras [1, 2, 4, 22]. This growing literature continues to provide new insights and results in this classical subject.

We remark that although this research seems very technical, the main question of interest stems from determining when an element of the Weyl group is in the Weyl alternation set and, hence, contributes a nonzero term to the multiplicity formula. Using this information we turn to the question of counting the number of ways one can write a weight as a sum of positive roots. Our approach to these problems involves the use of combinatorial techniques accessible to undergraduate students. Using these techniques students can contribute original research to the representation theory of Lie algebras, a topic rarely seen at the undergraduate level.

2.2 Technical Background and Definitions

Throughout this work we assume that the reader has a working knowledge of linear algebra, but little background in Lie algebras and their representations is assumed. Following the notation of [10] we begin with the needed definitions to make our approach precise. For the reader interested in more detailed background on Lie algebras and their representations we recommend [9, 10, 18].

Definition 1. A vector space $\mathfrak{g}$ over a field $\mathbb{F}$ together with a bilinear map $[\cdot,\cdot] : \mathfrak{g}\times\mathfrak{g}\to\mathfrak{g}$ is said to be a *Lie algebra* if the map satisfies:

1. $[X,Y] = -[Y,X]$ for all $X, Y \in \mathfrak{g}$ (skew symmetry).
2. $[X,[Y,Z]] + [Y,[Z,X]] + [Z,[X,Y]] = 0$ for all $X, Y, Z \in \mathfrak{g}$ (Jacobi identity).

The bilinear map is often referred to as the *Lie bracket* and the *dimension of the Lie algebra* $\mathfrak{g}$ is its dimension as a vector space over $\mathbb{F}$.

Calculus students have studied Lie algebras in vector calculus, but usually not under this name. Which leads us to another exercise.

Exercise 2. Confirm that $\mathbb{R}^3$ is a Lie algebra under the cross product of vectors.

Finite-dimensional (classical) Lie algebras are classified into families in the following way. For any $r \geq 2$, let $M_k(\mathbb{C})$ denote the set of $k \times k$ matrices with complex valued entries, then the classical Lie algebras of type A, B, C, and D are defined as:

- Type A_r $(r \geq 1)$: $\mathfrak{sl}_{r+1}(\mathbb{C}) = \{X \in M_{r+1}(\mathbb{C}) : Trace(X) = 0\}$.
- Type B_r $(r \geq 2)$: $\mathfrak{so}_{2r+1}(\mathbb{C}) = \{X \in M_{2r+1}(\mathbb{C}) : X^t = -X\}$.
- Type C_r $(r \geq 3)$: $\mathfrak{sp}_{2r}(\mathbb{C}) = \{X \in M_{2r}(\mathbb{C}) : X^t J = -JX\}$, where $J = \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix}$ with I the $r \times r$ identity matrix.
- Type D_r $(r \geq 4)$: $\mathfrak{so}_{2r}(\mathbb{C}) = \{X \in M_{2r}(\mathbb{C}) : X^t = -X\}$.

In these cases, the Lie bracket is the commutator bracket. That is, if $X, Y \in \mathfrak{g}$, then $[X, Y] = XY - YX$. In addition to these families, there are five other finite-dimensional simple Lie algebras, which do not belong to the Lie algebra families mentioned above, and are called the exceptional Lie algebras. These Lie algebras are identified as G_2, F_4, E_6, E_7, and E_8. As we specialize to the Lie algebra $\mathfrak{sl}_{r+1}(\mathbb{C})$ we omit their definition, but point the interested reader to [20] for further information on the exceptional Lie algebras. Having defined a Lie algebra we now shift our attention to their representations.

Definition 2. A *representation* of a Lie algebra $\mathfrak{g}$ on a vector space V is a Lie algebra homomorphism $\tau : \mathfrak{g} \to \mathfrak{gl}(V)$, where $\mathfrak{gl}(V)$ denotes the set of all linear maps from the vector space V to itself.

A Lie algebra homomorphism is a linear map between two Lie algebras that is compatible with the Lie bracket. Namely, the map $f : \mathfrak{g} \to \mathfrak{g}'$ is said to be a Lie algebra homomorphism if it satisfies

$$f([X, Y]) = [f(X), f(Y)] \quad \text{for all } X, Y \in \mathfrak{g}. \tag{3}$$

The bracket on the left hand side of (3) is the bracket defining the Lie algebra $\mathfrak{g}$, whereas the bracket on the right hand side of (3) is the bracket defining the Lie algebra $\mathfrak{g}'$.

Definition 3. The Lie algebra representation $\tau : \mathfrak{g} \to \mathfrak{gl}(V)$ is said to be *irreducible* if the only subspaces $W \subseteq V$ satisfying $\tau(X)(W) \subset W$ for all $X \in \mathfrak{g}$ are (0) and V.

One example of a Lie algebra representation is the trivial representation, which is defined by letting every element of the Lie algebra act as the identity map on the

vector space V. An important example of an irreducible representation is the adjoint representation of a Lie algebra. This representation uses the fact that the Lie algebra $\mathfrak{g}$ is itself a vector space. Thus, we can set $V = \mathfrak{g}$ and use the Lie bracket defining the Lie algebra $\mathfrak{g}$ as the map by which each Lie algebra element acts on $V = \mathfrak{g}$. More precisely, the adjoint representation is defined as

$$\text{ad} : \mathfrak{g} \to \mathfrak{gl}(\mathfrak{g}),$$

where each $X \in \mathfrak{g}$ defines a map from $\mathfrak{g}$ to itself. That is, $X \mapsto \text{ad}_X$, where $\text{ad}_X(Y) = [X, Y]$ for all $Y \in \mathfrak{g}$. Much is known about the adjoint representation and it will be the object of detailed study in our examples. Later, we even show a connection between this representation and the Fibonacci numbers, but before we do so we need to know about Cartan subalgebras.

Definition 4. We say $\mathfrak{h}$ is a *Cartan subalgebra* of $\mathfrak{g}$ if it satisfies the following:

- $\mathfrak{h}$ is a subalgebra of $\mathfrak{g}$, i.e. $\mathfrak{h} \subset \mathfrak{g}$ that is closed under the Lie bracket.
- $\mathfrak{h}$ is nilpotent, meaning the lower central series

$$\mathfrak{h} > [\mathfrak{h}, \mathfrak{h}] > [[\mathfrak{h}, \mathfrak{h}], \mathfrak{h}] > [[[\mathfrak{h}, \mathfrak{h}], \mathfrak{h}], \mathfrak{h}] > \cdots$$

  eventually terminates.
- $\mathfrak{h}$ is self-normalizing, namely if $[X, Y] \in \mathfrak{h}$ for all $X \in \mathfrak{h}$, then $Y \in \mathfrak{h}$.

For any Lie algebra $\mathfrak{g}$, there are many choices for a Cartan subalgebra. Following the convention set in [10] for $\mathfrak{sl}_{r+1}(\mathbb{C})$ we choose the Cartan subalgebra $\mathfrak{h}$ as the set of traceless diagonal matrices in $\mathfrak{sl}_{r+1}(\mathbb{C})$. That is,

$$\mathfrak{h} = \left\{ diag[a_1, a_2, \ldots, a_{r+1}] : a_1, a_2, \ldots, a_{r+1} \in \mathbb{C} \text{ and } \sum_{i=1}^{r+1} a_i = 0 \right\}.$$

Challenge Problem 1. Verify that $\mathfrak{h}$, as defined above, satisfies Definition 4.

In general, we let $\mathfrak{h}^*$ denote the dual of the (chosen) Cartan subalgebra, where the dual of a vector space consists of all linear functionals on the vector space. For us this means that an element $\xi \in \mathfrak{h}^*$ is a linear map $\xi : \mathfrak{h} \to \mathbb{C}$.

Definition 5. If $\mathfrak{h}^*$ denotes the dual of the Cartan subalgebra $\mathfrak{h}$, then the *weights of the adjoint representation* for the Lie algebra $\mathfrak{g}$ are the linear functionals $\xi \in \mathfrak{h}^*$, i.e. maps $\xi : \mathfrak{h} \to \mathbb{C}$. For the weight $\alpha \in \mathfrak{h}^*$, define

$$\mathfrak{g}_\alpha = \{X \in \mathfrak{g} : [H, X] = \alpha(H)X, \text{ for all } H \in \mathfrak{h}\}. \tag{4}$$

We now summarize some Lie algebra information we will need in order to compute weight multiplicities. First, note that (4) implies that the weights of the

adjoint representation for a Lie algebra are a generalization of an eigenvalue and $\mathfrak{g}_\alpha$ is a generalization of an eigenspace. If $\alpha \neq 0$, and $\mathfrak{g}_\alpha \neq (0)$, then α is called a root and $\mathfrak{g}_\alpha$ is called a root space. Whenever α is a root the $\dim \mathfrak{g}_\alpha = 1$. The set Φ is called the set of roots for $(\mathfrak{g}, \mathfrak{h})$ the root system. Then the Lie algebra $\mathfrak{g}$ has the following root space (or Cartan) decomposition

$$\mathfrak{g} = \mathfrak{h} \oplus \sum_{\alpha \in \Phi} \mathfrak{g}_\alpha.$$

One can decompose[1] the root system $\Phi = \Phi^+ \cup -\Phi^+$, where Φ^+ denotes the set of positive roots and $-\Phi^+$ denotes the set of negative roots. A subset of the positive roots $\Delta = \{\alpha_1,\ \alpha_2,\ \ldots, \alpha_r\} \subset \Phi^+$ is a set of simple roots if every $\gamma \in \Phi^+$ can be written uniquely as $\gamma = n_1\alpha_1 + n_2\alpha_2 + \cdots + n_r\alpha_r$ where $n_1,\ \ldots,\ n_r \in \mathbb{N}$. It is known that the set of simple roots is unique. This uniqueness, together with the fact that Φ^+ spans $\mathfrak{h}^*$, implies that Δ is a basis for $\mathfrak{h}^*$.

Combinatorially, we note that if $\beta = n_1\alpha_1 + \cdots + n_r\alpha_r$ with $n_1, \ldots, n_r \in \mathbb{N}$ is a root, then we define the height of β, relative to Δ, as $ht(\beta) = n_1 + \cdots + n_r$. The positive roots of a Lie algebra are thus described as the roots β for which $ht(\beta) > 0$. Moreover, a root β is called the highest root of Φ (or of the Lie algebra), relative to the simple roots Δ, if $ht(\beta) > ht(\gamma)$ for all roots $\gamma \neq \beta$.

For each $\alpha \in \Phi^+$ there exist $e_\alpha \in \mathfrak{g}_\alpha$ and $f_\alpha \in \mathfrak{g}_{-\alpha}$ such that the element $h_\alpha = [e_\alpha, f_\alpha] \in \mathfrak{h}$ satisfies $\alpha(h_\alpha) = 2$ and we call h_α, also denoted by $\check{\alpha}$, the coroot of α. For $X, Y \in \mathfrak{g}$, define the symmetric bilinear form $(X, Y) = tr(XY)$ on $\mathfrak{g}$. Let $\mathfrak{h}_\mathbb{R}$ be the real span of the coroots, then the real dual space, $\mathfrak{h}^*_\mathbb{R}$, is the real linear span of the roots. Since the trace form is positive definite on $\mathfrak{h}_\mathbb{R}$, we can use it to identify $\mathfrak{h}_\mathbb{R}$ with $\mathfrak{h}^*_\mathbb{R}$ to obtain a positive definite inner product, $(,)$, on $\mathfrak{h}^*_\mathbb{R}$. We define the simple root reflection $s_\alpha : \mathfrak{h}^* \to \mathfrak{h}^*$ by

$$s_\alpha(\beta) = \beta - \frac{2(\beta, \alpha)}{(\alpha, \alpha)}\alpha. \tag{5}$$

The Weyl group is a finite group that plays an important role in the representation theory of Lie algebras. The Weyl group is defined as the group $W = W(\mathfrak{g}, \mathfrak{h})$ of orthogonal transformations of $\mathfrak{h}^*_\mathbb{R}$ generated by the simple root reflections, as defined in (5). So each element $\sigma \in W$ can be written as the product of generators $\sigma = s_{\alpha_{i_1}} s_{\alpha_{i_2}} \cdots s_{\alpha_{i_k}}$, where $1 \leq i_j \leq r$ for all $j = 1, \ldots, k$. If $s_{\alpha_{i_1}} s_{\alpha_{i_2}} \cdots s_{\alpha_{i_k}}$ is minimal among all such expressions for σ, then we call k the length of σ and write $\ell(\sigma) = k$. If $k = 0$, then σ is the empty product of simple reflections and hence $\sigma = 1$ is the identity element in W.

Let $\{\varpi_1, \ldots, \varpi_r\}$ be the set of fundamental weights of $\mathfrak{g}$ corresponding to our choice of $\mathfrak{h}$. The fundamental weights are defined by the conditions $\varpi_i(h_{\alpha_j}) = \delta_{ij}$,

[1]The decomposition of the root system into positive and negative roots involves a choice of Borel subgroup. The interested reader can see more details of this choice in [10].

where δ_{ij} denotes the Kronecker delta function, which is 1 when $i = j$ and 0 otherwise. The set of integral and dominant integral weights are defined as

$$P(\mathfrak{g}) = \{a_1\varpi_1 + \cdots + a_r\varpi_r | a_1, \ldots, a_r \in \mathbb{Z}\} \text{ and} \tag{6}$$

$$P_+(\mathfrak{g}) = \{n_1\varpi_1 + \cdots + n_r\varpi_r | n_1, \ldots, n_r \in \mathbb{N}\}, \text{ respectively.} \tag{7}$$

With this background on Lie algebras, let us return to representation theory by restating the theorem of the highest weight. This theorem "asserts that among the weights that occur in the decomposition, there is a unique maximal element, relative to a partial order coming from a choice of positive roots for G. This maximal element, called the highest weight, occurs with multiplicity one and uniquely determines the representation" [10, p. 129]. Conversely, every dominant integral weight of a simple (in fact, semisimple) Lie algebra $\mathfrak{g}$ is the highest weight of an irreducible finite-dimensional representation of $\mathfrak{g}$. Therefore, for any $\lambda \in P_+(\mathfrak{g})$ we let $L(\lambda)$ denote the irreducible highest weight representation of $\mathfrak{g}$ whose highest weight is λ.

The theorem of the highest weight explicitly states a one-to-one correspondence between dominant integral weights and finite-dimensional complex irreducible representations of a Lie algebra. This together with the fact that all representations of a finite-dimensional (semisimple) Lie algebra are completely reducible, i.e. can be decomposed into the direct sum of irreducible representations, we can now focus on studying the structure of the irreducible representations.

Let $\tau : \mathfrak{g} \to \mathfrak{gl}(V)$ be an arbitrary irreducible finite-dimensional representation of the Lie algebra $\mathfrak{g}$ on the vector space V. The structure of this representation can be studied by observing that the vector space V can be decomposed as the direct sum

$$V = \bigoplus V_\alpha, \tag{8}$$

which is indexed by a finite number of linear functionals $\alpha \in \mathfrak{h}^*$. Moreover, the linear functionals $\alpha \in \mathfrak{h}^*$, over which the direct sum in (8) is taken, are eigenvalues as they satisfy

$$H(v) = \alpha(H)v \quad \text{for any } H \in \mathfrak{h} \text{ and } v \in V_\alpha.$$

Definition 6. An eigenvalue $\alpha \in \mathfrak{h}^*$ appearing in (8) is called a *weight* of the representation, and the associated vector subspace V_α is called a *weight space*. The dimension of the weight space V_α is called the *multiplicity* of the weight α.

As one can imagine, our goal is to now determine the weights that actually appear in (8), as well as compute the dimension of the associated weight space. In other words, we aim to compute $m(\lambda, \mu)$ the multiplicity of the weight μ in an irreducible highest weight representation with highest weight λ, which we denote by $L(\lambda)$. If $m(\lambda, \mu) = 0$, then $\dim(V_\mu) = 0$. Hence $V_\mu = (0)$ will appear in the sum (8). However, this term appears trivially and does not provide any further information

on the nontrivial subspaces that appear in the decomposition (8). Hence, we are interested in finding those weight spaces V_μ for which $\dim(V_\mu) > 0$. In this case, if $m(\lambda, \mu) > 0$ then μ does appear in (8) and the dimension of V_μ is exactly $m(\lambda, \mu)$.

One way to compute the multiplicity $m(\lambda, \mu)$ is through the use of Kostant's weight multiplicity formula [21]:

$$m(\lambda, \mu) = \sum_{\sigma \in W} (-1)^{\ell(\sigma)} \wp(\sigma(\lambda + \rho) - (\mu + \rho)), \tag{2}$$

where $\rho = \frac{1}{2} \sum_{\alpha \in \Phi^+} \alpha$, W is the Weyl group, and $\wp$ denotes Kostant's partition function, which counts the number of ways the weight $\sigma(\lambda + \rho) - (\mu + \rho)$ may be written as a nonnegative integral sum of positive roots.

One complication in using (1) to compute weight multiplicities is that closed formulas for the value of Kostant's partition function are often unknown. Also we must contend with the complication that the order of the Weyl group, over which the sum in (1) is taken, grows factorially in terms of the Lie algebra's rank.

In practice, one develops the intuition, as was noted in [8], that most of the terms in Kostant's weight multiplicity formula are zero and hence do not contribute to the overall multiplicity. This means that the partition function involved in (1) allows us to reduce the computation drastically as many of the terms $\sigma(\lambda+\rho)-\rho-\mu$ cannot be expressed as nonnegative integral sums of positive roots. With this in mind, we aim to describe the elements of W that actually contribute nonzero terms to (1) which introduces the following.

Definition 7. For λ, μ integral weights of $\mathfrak{g}$ define the Weyl alternation set to be $\mathscr{A}(\lambda, \mu) = \{\sigma \in W | \ \wp(\sigma(\lambda + \rho) - (\mu + \rho)) > 0\}$.

This definition implies that $\sigma \in \mathscr{A}(\lambda, \mu)$ if and only if $\sigma(\lambda + \rho) - (\mu + \rho)$ can be expressed as a nonnegative integral combination of positive roots. In particular, since all positive roots are nonnegative integral combinations of the simple roots, we deduce that $\sigma \in \mathscr{A}(\lambda, \mu)$ if and only if $\sigma(\lambda + \rho) - (\mu + \rho)$ can be expressed as a nonnegative integral combination of the simple roots. There will be times when specifying the rank of the Lie algebra is important. In these cases, we use the notation $\mathscr{A}_r(\lambda, \mu)$ in place of $\mathscr{A}(\lambda, \mu)$.

We now continue our study with concrete examples, which help us develop the necessary techniques to begin working on some open problems in this area.

3 Learning Through Examples

This section presents an approach to the computation of weight multiplicities and shows that although the background material is extremely technical, the techniques used in computing multiplicities are accessible to students with a linear algebra and abstract algebra background. Working through some examples will help students learn the background material and will lead students to make contributions to the representation theory of Lie algebras, a topic rarely encountered at the undergraduate level.

In this section we limit our study to the Lie algebra $\mathfrak{g} = \mathfrak{sl}_{r+1}(\mathbb{C})$ and more specifically to the case where $r = 2$. Let us record some important information about $\mathfrak{sl}_3(\mathbb{C})$, which will be helpful shortly. The simple roots of $\mathfrak{sl}_3(\mathbb{C})$ are given by

$$\Delta = \{\alpha_1,\ \alpha_2\},$$

the set of roots is

$$\Phi = \{\alpha_1,\ \alpha_2,\ \alpha_1 + \alpha_2,\ -\alpha_1,\ -\alpha_2,\ -\alpha_1 - \alpha_2\},$$

the positive roots are

$$\Phi^+ = \{\alpha_1,\ \alpha_2,\ \alpha_1 + \alpha_2\}$$

and $\rho = \frac{1}{2}\sum_{\alpha\in\Phi^+}\alpha = \alpha_1 + \alpha_2$. Since the highest root is the positive root whose coefficient sum (height) is the largest among all roots, we see that the highest root of $\mathfrak{sl}_3(\mathbb{C})$ is $\alpha_1 + \alpha_2$, which we denote by $\tilde{\alpha} = \alpha_1 + \alpha_2$. It is a coincidence that $\rho = \tilde{\alpha}$ as this only happen in $\mathfrak{sl}_3(\mathbb{C})$.

As we described in Section 1, in $\mathfrak{sl}_3(\mathbb{C})$ the roots can be visualized on a 2-dimensional plane, see Figure 2. In this figure the black dashed lines represent the real span of the simple roots, and the red solid lines represent the reflections perpendicular to the labeled simple root. Namely, s_1 denotes the element of the Weyl group that takes a root and reflects it about the line perpendicular to the simple root α_1, similarly for s_2. As these simple reflections generate the Weyl group, all of the elements will be concatenations (also referred to as words) of s_1 and s_2. We let 1 denote the empty word, which is the identity element in W. Note that the simple reflections are elements of order two. If we reflect any weight (vector) about the same hyperplane (line in this case) twice, it returns us to the original weight (vector). Hence, it acts the same as if we did not reflect the weight at all.

We can now begin our study by working through an in-depth example.

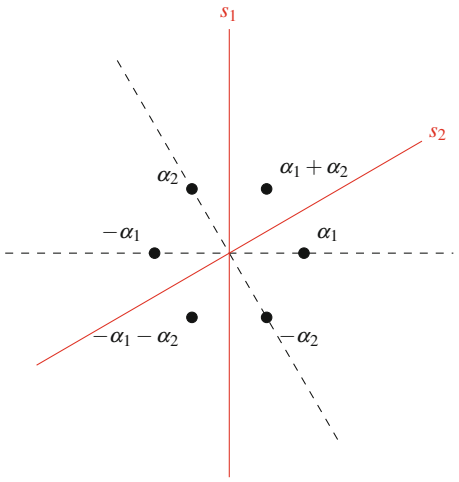


Fig. 2 Roots of $\mathfrak{sl}_3(\mathbb{C})$ along with the hyperplanes perpendicular to the simple roots.

Example 1. Compute the dimension of the zero weight space in the adjoint representation of the Lie algebra $\mathfrak{sl}_3(\mathbb{C})$.

Solution 1. In general, the adjoint representation of a Lie algebra $\mathfrak{g}$ is the representation with highest weight equal to the highest root of $\mathfrak{g}$. Hence, computing the dimension of the zero weight space in the adjoint representation is equivalent to computing

$$m(\tilde{\alpha}, 0) = \sum_{\sigma \in W} (-1)^{\ell(\sigma)} \wp(\sigma(\tilde{\alpha} + \rho) - \rho), \tag{9}$$

where $\tilde{\alpha}$ is the highest root of $\mathfrak{g}$. To compute this multiplicity, we first concretely describe the Weyl alternation set $\mathscr{A}(\tilde{\alpha}, 0)$. To do so we must compute $\sigma(\tilde{\alpha} + \rho) - \rho$ for every element σ of the Weyl group, as the Weyl alternation set consists of Weyl group elements for which the expression $\sigma(\tilde{\alpha}+\rho)-\rho$ can be written as a nonnegative integral sum of the positive roots.

Since we need to compute the value of $\sigma(\tilde{\alpha} + \rho) - \rho$ for every element $\sigma \in W$ we must know explicitly how s_1 and s_2 act on weights. For example, if we want to determine $s_1 s_2(\alpha_1)$, first we find $s_2(\alpha_1)$ and then apply s_1 to the result. Hence, concatenation of simple reflections behave as compositions of functions, and we must compute by working from the inner-most simple reflection and move outward.

Let us compute $s_1 s_2(\alpha_1)$ directly. Using Figure 1, note that reflecting α_1 about s_2 yields $\alpha_1+\alpha_2$. Then reflecting $\alpha_1+\alpha_2$ about s_1 yields α_2. Thus $s_1 s_2(\alpha_1+\alpha_2) = \alpha_2$. Following this process for all possible elements of the Weyl group, which are words in s_1 and s_2, we arrive at the entries in Table 1, which provide the action of the Weyl group elements on the roots of $\mathfrak{sl}_3(\mathbb{C})$.

Now that we understand how the elements of the Weyl group act on the roots of the Lie algebra $\mathfrak{sl}_{r+1}(\mathbb{C})$, we can compute $\sigma(\tilde{\alpha} + \rho) - \rho$ for every $\sigma \in W$. Since $\tilde{\alpha} + \rho = 2\alpha_1 + 2\alpha_2$, we can use Table 1 to compute

$$1(\tilde{\alpha} + \rho) - \rho = \tilde{\alpha} + \rho - \rho = \tilde{\alpha} = \alpha_1 + \alpha_2 \tag{10}$$

$$s_1(\tilde{\alpha} + \rho) - \rho = 2s_1(\alpha_1 + \alpha_2) - \alpha_1 - \alpha_2 = -\alpha_1 + \alpha_2 \tag{11}$$

$$s_2(\tilde{\alpha} + \rho) - \rho = 2s_2(\alpha_1 + a_2) - \alpha_1 - \alpha_1 = \alpha_1 - \alpha_2 \tag{12}$$

Table 1 Weyl group elements and their action on the roots of $\mathfrak{sl}_3(\mathbb{C})$.

Weyl group element σ	$\sigma(\alpha_1)$	$\sigma(\alpha_2)$	$\sigma(\alpha_1 + \alpha_2)$	$\sigma(-\alpha_1)$	$\sigma(-\alpha_2)$	$\sigma(-\alpha_1 - \alpha_2)$
1	α_1	α_2	$\alpha_1 + \alpha_2$	$-\alpha_1$	$-\alpha_2$	$-\alpha_1 - \alpha_2$
s_1	$-\alpha_1$	$\alpha_1 + \alpha_2$	α_2	α_1	$-\alpha_1 - \alpha_2$	$-\alpha_2$
s_2	$\alpha_1 + \alpha_2$	$-\alpha_2$	α_1	$-\alpha_1 - \alpha_2$	α_2	$-\alpha_1$
$s_1 s_2$	α_2	$-\alpha_1 - \alpha_2$	$-\alpha_1$	$-\alpha_2$	$\alpha_1 + \alpha_2$	α_1
$s_2 s_1$	$-\alpha_1 - \alpha_2$	α_1	$-\alpha_2$	$\alpha_1 + \alpha_2$	$-\alpha_1$	α_2
$s_1 s_2 s_1$	$-\alpha_2$	$-\alpha_1$	$-\alpha_1 - \alpha_2$	α_2	α_1	$\alpha_1 + \alpha_2$

$$s_1 s_2(\tilde{\alpha}+\rho)-\rho = 2s_1 s_2(\alpha_1+\alpha_2)-\alpha_1-\alpha_2 = -3\alpha_1-\alpha_2 \tag{13}$$

$$s_2 s_1(\tilde{\alpha}+\rho)-\rho = 2s_2 s_1(\alpha_1+\alpha_2)-\alpha_1-\alpha_2 = -\alpha_1-3\alpha_2 \tag{14}$$

$$s_1 s_2 s_1(\tilde{\alpha}+\rho)-\rho = 2s_1 s_2 s_1(\alpha_1+\alpha_2)-\alpha_1-\alpha_2 = -3\alpha_1-3\alpha_2. \tag{15}$$

For every element of the Weyl group, we have written $\sigma(\tilde{\alpha}+\rho)-\rho$ as a linear combination of simple roots. Now we need to determine which elements of the Weyl group are elements of the Weyl alternation set $\mathscr{A}(\tilde{\alpha},0)$. The question we need to answer is:

For which $\sigma \in W$ can we write $\sigma(\tilde{\alpha}+\rho)-\rho$ as a nonnegative integral sum of the positive roots?

Note that if an expression $c_1\alpha_2 + c_2\alpha_2$ is to be written as a nonnegative sum of positive roots, you must be able to write it as a nonnegative integral sum of simple roots, since, by definition, all positive roots are nonnegative integral sums of simple roots. Hence, we can refer back to Equations (10)-(15) and if we see a negative coefficient on either of the simple roots α_1 or α_2, this immediately implies that the respective expression cannot be written as a nonnegative integral sum of the positive roots. This indicates that the value of the partition function on such expressions is zero, and hence that particular element of the Weyl group is not an element of the Weyl alternation set. Doing this establishes that in the Lie algebra $\mathfrak{sl}_3(\mathbb{C})$, the only element in $\mathscr{A}(\tilde{\alpha},0)$ is 1, the identity element of W.

This reduces the sum in the computation of the multiplicity $m(\tilde{\alpha},0)$ from the 6 terms, i.e. the six elements of the Weyl group, to only the term associated to the identity element. Hence, we can reduce equation (9) to

$$m(\tilde{\alpha},0) = (-1)^{\ell(1)}\wp(\tilde{\alpha}). \tag{16}$$

To finish the computation we restate the following facts: the length of the identity element, denoted $\ell(1)$, is by definition 0, and for any weight ξ, $\wp(\xi)$ counts the number of ways the weight ξ can be written as the nonnegative integral sum of positive roots. Observe that there are only two ways to write $\tilde{\alpha}$ as a sum of positive roots. One way to write $\tilde{\alpha}$ is by using the two simple roots α_1 and α_2. The second is to write $\tilde{\alpha}$ using the positive root $\alpha_1+\alpha_2$. This establishes that $m(\tilde{\alpha},0)$=2.

In the previous example, we saw that by concatenating simple reflections we eventually created all of the group elements of W. It is important to note that sometimes different words in the letters s_1 and s_2 are in fact the same. For example, $s_1 s_2 s_1 = s_2 s_1 s_2$, as the reader who worked out Exercise 1 showed. These types of relations are called braid relations on Coxeter groups. Coxeter groups form a larger family of groups, which include the Weyl groups, and there is a vast literature on their combinatorial properties. We point the interested reader to [5,7,19] for a more detailed study of Coxeter groups.

For our purposes, we now summarize information on the simple roots, fundamental weights, and the action of the simple reflections on the simple roots in the general case of the Weyl group associated to $\mathfrak{sl}_{r+1}(\mathbb{C})$ with $r \geq 1$. This information will be of importance in our next few examples.

For the Lie algebra of type A_r, the simple roots are

$$\Delta = \{\alpha_1, \alpha_2, \ldots, \alpha_r\}.$$

The set of positive roots consists of the simple roots and sums of consecutive simple roots:

$$\Phi^+ = \Delta \cup \{\alpha_i + \alpha_{i+1} + \cdots + \alpha_j : 1 \le i < j \le r\}.$$

From this we observe that the highest root, the one with largest coefficient sum, is given by

$$\tilde{\alpha} = \alpha_1 + \alpha_2 + \cdots + \alpha_r. \tag{17}$$

If[2] $1 \le i \le r$, then

$$\begin{aligned}\varpi_i = \frac{r+1-i}{r+1}(\alpha_1 + 2\alpha_2 + \cdots + (i-1)\alpha_{i-1}) \\ + \frac{i}{r+1}((r-i+1)\alpha_i + (r-i)\alpha_{i+1} + \cdots + \alpha_r).\end{aligned} \tag{18}$$

By definition, $\rho = \frac{1}{2}\sum_{\alpha\in\Phi^+}\alpha$. However there is an additional definition of ρ.

Challenge Problem 2. Prove that

$$\rho = \varpi_1 + \varpi_2 + \cdots + \varpi_r, \tag{19}$$

where $\varpi_1, \varpi_2, \ldots, \varpi_r$ are the fundamental weights of the Lie algebra $\mathfrak{sl}_{r+1}(\mathbb{C})$.

The Weyl group is generated by the simple reflections s_i $(1 \le i \le r)$ associated with the simple roots α_i $(1 \le i \le r)$. The reflections $s_1, s_2, \ldots, s_r$ act on the simple roots $\alpha_1, \alpha_2, \ldots, \alpha_r$ in the following way: If $1 \le i, j \le r$ are nonconsecutive integers, then

$$s_i(\alpha_j) = \alpha_j \quad \text{and} \quad s_j(\alpha_i) = \alpha_i. \tag{20}$$

If $1 \le i \le r-1$, then

$$s_i(\alpha_{i-1}) = \alpha_{i-1} + \alpha_i, \tag{21}$$

$$s_i(\alpha_i) = -\alpha_i, \tag{22}$$

$$s_i(\alpha_{i+1}) = \alpha_i + \alpha_{i+1}. \tag{23}$$

[2] This is [10, Exercise 3.2(a)].

If $i = r$, then

$$s_r(\alpha_{r-1}) = \alpha_{r-1} + \alpha_r \tag{24}$$

$$s_r(\alpha_r) = -\alpha_r. \tag{25}$$

If $1 \leq i,j \leq r$, then the action of the simple reflections on the fundamental weights is defined by

$$s_i(\varpi_j) = \begin{cases} \varpi_j - \alpha_i & \text{if } i = j \\ \varpi_j & \text{if } i \neq j. \end{cases} \tag{26}$$

Another important property of the action of Weyl group elements on weights is that the action is linear. That is, all elements $\sigma \in W$ satisfy

$$\sigma(c_1\xi_1 + c_2\xi_2 + \cdots + c_n\xi_n) = c_1\sigma(\xi_1) + c_2\sigma(\xi_2) + \cdots + c_n\sigma(\xi_n)$$

for any $n \in \mathbb{N}$, constants $c_1, c_2, \ldots, c_n$, and weights $\xi_1, \xi_2, \ldots, \xi_n$.

Example 1 illustrated a well-known result in Lie theory:

The dimension of the zero weight space in the adjoint representation is equal to the rank of the Lie algebra, where the rank is equal to the dimension of any of its Cartan subalgebras.

For the classical Lie algebras of types A_r, B_r, C_r, and D_r, the rank is indicated by the value r. In Example 1, we considered $\mathfrak{sl}_3(\mathbb{C})$, which has rank 2, and we confirmed that $m(\tilde{\alpha}, 0) = 2$. Using techniques similar to those presented in Example 1, we provided a combinatorial proof of the above mentioned result for the Lie algebra $\mathfrak{sl}_{r+1}(\mathbb{C})$ [15].

Theorem 1. *Let $r \geq 2$. If $\tilde{\alpha}$ is the highest root of $\mathfrak{sl}_{r+1}(\mathbb{C})$, then $m(\tilde{\alpha}, 0) = r$.*

In order to prove Theorem 1 we will need the following results, which first appeared in [13]. We summarize the findings below and provide proofs to results that were previously omitted in [13].

Theorem 2. *Let $r \geq 2$ and let $\mathfrak{g} = \mathfrak{sl}_{r+1}(\mathbb{C})$. Then $\sigma \in \mathscr{A}(\tilde{\alpha}, 0)$ if and only if $\sigma = 1$ or $\sigma = s_{i_1}s_{i_2}\cdots s_{i_k}$ for some nonconsecutive integers $i_1, i_2, \ldots, i_k$ between 2 and $r-1$.*

The proof of Theorem 2 presented in [15] uses the fact that the Weyl group of $\mathfrak{sl}_{r+1}(\mathbb{C})$ is isomorphic to the group of permutations on $r+1$ letters. A worthwhile endeavor is to prove this result using the definition of the Weyl group as generated by the reflections about hyperplanes perpendicular to the simple roots, as presented in the current exposition.

Challenge Problem 3. Prove Theorem 2 using the definition of the elements of the Weyl group as being generated by reflections $s_1,\ s_2,\ \ldots,\ s_r$.

We can now state a connection between the adjoint representation of the Lie algebra $\mathfrak{sl}_{r+1}(\mathbb{C})$ and the Fibonacci numbers, which are defined by the recurrence relation

$$F_r = F_{r-1} + F_{r-2} \text{ for } r \geq 3, \text{ and } F_1 = F_2 = 1.$$

Corollary 1. *If $r \geq 2$ and $\mathfrak{g} = \mathfrak{sl}_{r+1}(\mathbb{C})$, then $|\mathscr{A}(\tilde{\alpha}, 0)| = F_r$, where F_r denotes the r^{th} Fibonacci number.*

Proof. From Theorem 2 we know every element in $\mathscr{A}(\tilde{\alpha}, 0)$ can be written as $s_{i_1} s_{i_2} \cdots s_{i_k}$ where $i_1, i_2, \ldots i_k$ are nonconsecutive integers between 2 and $r-1$. Hence, the proof that $|\mathscr{A}(\tilde{\alpha}, 0)| = F_r$ reduces to showing that there are F_r subsets of $\{2, \ldots, r-1\}$ that contain only nonconsecutive integers.

Let $\mathscr{A}_r(\tilde{\alpha}, 0)$ denote the set $\mathscr{A}(\tilde{\alpha}, 0)$ in $\mathfrak{sl}_{r+1}(\mathbb{C})$. We proceed by induction. If $r = 2$, then the empty set is the only subset of $\emptyset$ consisting of nonconsecutive integers. So $|\mathscr{A}_2(\tilde{\alpha}, 0)| = 1 = F_2$. If $r = 3$, then there exist two subsets of $\{2\}$ consisting of nonconsecutive integers: $\emptyset$ and $\{2\}$. Hence, $|\mathscr{A}_3(\tilde{\alpha}, 0)| = 2 = F_3$. If $r = 4$, then there are three subsets of $\{2, 3\}$ consisting of nonconsecutive integers: $\emptyset$, $\{2\}$, and $\{3\}$. Hence, $|\mathscr{A}_4(\tilde{\alpha}, 0)| = 3 = F_4$.

Assume that for $3 \leq k < r$, $|A_k(\tilde{\alpha}, 0)| = F_k$. Let us count the total number of subsets of $\{2, 3, \ldots, r-1\}$ consisting of nonconsecutive integers. First note that all of these subsets of $\{2, 3, \ldots, r-1\}$ will either contain $r-1$ or not. If they do not contain $r-1$, then we are counting the subsets of $\{2, 3, \ldots, r-2\}$ which consist of nonconsecutive integers. By our induction hypothesis, this is given by F_{r-1}. If, on the other hand, $r-1$ is included in the subsets of $\{2, 3, \ldots, r-1\}$ consisting of nonconsecutive integers, then we must count the number of subsets of $\{2, 3, \ldots, r-3\}$ as $r-3$ would be the largest integer in that set which we could include in our subset of nonconsecutive integers that already contain $r-1$. By our induction hypothesis, this is given by F_{r-2}. As any given subset of nonconsecutive integers either includes $r-1$ or not, and it cannot do both, we have shown that the total number of subsets of $\{2, 3, \ldots, r-1\}$ consisting of nonconsecutive integers is given by $F_{r-2} + F_{r-1} = F_{r-1}$. Thus $|\mathscr{A}_r(\tilde{\alpha}, 0)| = F_r$.

By having an explicit description of the elements of the Weyl alternation set $\mathscr{A}(\tilde{\alpha}, 0)$ along with a simplification of $\sigma(\tilde{\alpha} + \rho) - \rho$ for each $\sigma \in \mathscr{A}(\tilde{\alpha}, 0)$ we provide a closed formula for $\wp(\sigma(\tilde{\alpha} + \rho) - \rho)$. This is our next result.

Proposition 1. *Let $r \geq 2$. If $\tilde{\alpha}$ is the highest root of $\mathfrak{sl}_{r+1}(\mathbb{C})$ and $\sigma \in \mathscr{A}(\tilde{\alpha}, 0)$, then $\wp(\sigma(\tilde{\alpha} + \rho) - \rho) = 2^{r-1-2\ell(\sigma)}$.*

To prove Proposition 1 we need the following technical lemmas.

Lemma 1. *Let $r \geq 2$. If $\sigma = s_{i_1} s_{i_2} \cdots s_{i_k}$ with $i_1, i_2, \ldots, i_k$ nonconsecutive integers between 2 and $r-1$, then $\sigma(\tilde{\alpha} + \rho) - \rho = \tilde{\alpha} - \sum_{j=1}^{k} \alpha_{i_j}$.*

Proof. Let $\sigma = s_{i_1}s_{i_2}\cdots s_{i_k}$ with $i_1, i_2, \ldots, i_k$ nonconsecutive integers between 2 and $r-1$. Since $s_is_j = s_js_i$ whenever i and j are nonconsecutive, we can assume that $\sigma = s_{i_1}s_{i_2}\cdots s_{i_k}$ where $1 < i_1 < i_2 < \cdots < i_k < r$ are nonconsecutive integers. Since σ acts linearly, we know

$$\sigma(\tilde{\alpha}+\rho) - \rho = \sigma(\tilde{\alpha}) + \sigma(\rho) - \rho. \tag{27}$$

Again using the linearity of the action of the Weyl group elements and equations (20)–(23) we can observe that for any $1 \leq j \leq k$

$$\begin{aligned}
s_{i_j}(\tilde{\alpha}) &= s_{i_j}(\alpha_1+\alpha_2+\cdots+\alpha_{i_j-2}+\alpha_{i_j-1}+\alpha_{i_j}+\alpha_{i_j+1}+\alpha_{i_j+2}+\cdots+\alpha_r)\\
&= s_{i_j}(\alpha_1)+s_{i_j}(\alpha_2)+\cdots+s_{i_j}(\alpha_{i_j-2})+s_{i_j}(\alpha_{i_j-1})+s_{i_j}(\alpha_{i_j})+s_{i_j}(\alpha_{i_j+1})\\
&\qquad +s_{i_j}(\alpha_{i_j+2})+\cdots+s_{i_j}(\alpha_r)\\
&= \alpha_1+\cdots+\alpha_{i_j-2}+[\alpha_{i_j-1}+\alpha_{i_j}]+[-\alpha_{i_j}]+[\alpha_{i_j}+\alpha_{i_j+1}]+\alpha_{i_j+2}+\cdots+\alpha_r\\
&= \alpha_1+\cdots+\alpha_{i_j-2}+\alpha_{i_j-1}+\alpha_{i_j}+\alpha_{i_j+1}+\alpha_{i_j+2}+\cdots+\alpha_r\\
&= \tilde{\alpha}.
\end{aligned}$$

As $\sigma = s_{i_1}s_{i_2}\cdots s_{i_k}$ where $1 < i_1 < i_2 < \cdots < i_k < r$ are nonconsecutive integers, we have shown that $\sigma(\tilde{\alpha}) = \tilde{\alpha}$.

Recall $\rho = \varpi_1+\varpi_2+\cdots+\varpi_r$. By (26), if $1 \leq j \leq k$, then $s_{i_j}(\varpi_\ell) = \varpi_\ell - \delta_{i_j,\ell}\alpha_{i_j}$, where $\delta_{i_j,\ell} = 1$ if $i_j = \ell$ and 0 otherwise. Hence, for any $1 \leq j \leq k$

$$\begin{aligned}
s_{i_j}(\rho) &= s_{i_j}(\varpi_1+\varpi_2+\cdots+\varpi_r)\\
&= s_{i_j}(\varpi_1)+s_{i_j}(\varpi_2)+\cdots+s_{i_j}(\varpi_{i_j-1})+s_{i_j}(\varpi_{i_j})+s_{i_j}(\varpi_{i_j+1})+\cdots+s_{i_j}(\varpi_r)\\
&= \varpi_1+\varpi_2+\cdots+\varpi_{i_j-1}+[\varpi_{i_j}-\alpha_{i_j}]+\varpi_{i_j+1}+\cdots+\varpi_r\\
&= \rho-\alpha_{i_j}.
\end{aligned}$$

As $\sigma = s_{i_1}s_{i_2}\cdots s_{i_k}$ where $1 < i_1 < i_2 < \cdots < i_k < r$ are nonconsecutive integers, we have shown $\sigma(\rho) = \rho - \sum_{j=1}^k \alpha_{i_j}$. The result follows from substituting $\sigma(\tilde{\alpha}) = \tilde{\alpha}$ and $\sigma(\rho) = \rho - \sum_{j=1}^k \alpha_{i_j}$ into equation (27).

Lemma 2. *If $r \geq 1$ and $\tilde{\alpha}$ is the highest root of $\mathfrak{sl}_{r+1}(\mathbb{C})$, then $\wp(\tilde{\alpha}) = 2^{r-1}$.*

Proof. We proceed by induction on r. If $r = 1$, then $\Phi^+ = \{\alpha_1\}$, so $\wp(\alpha_1) = 1 = 2^0$. If $r = 2$, then $\Phi^+ = \{\alpha_1, \alpha_2, \alpha_1+\alpha_2\}$, and $\tilde{\alpha} = \alpha_1+\alpha_2$ can be written using the simple roots α_1 and α_2, or we can use the positive root $\alpha_1+\alpha_2$. Hence $\wp(\tilde{\alpha}) = 2 = 2^{2-1}$.

Assume that $\wp(\tilde{\alpha}) = 2^{k-1}$ for any $k \leq r$. If $k = r+1$, then let Φ^+_{r+1} be the set of positive roots of $\mathfrak{sl}_{r+2}(\mathbb{C})$ and Φ^+_r be the set of positive roots of $\mathfrak{sl}_{r+1}(\mathbb{C})$. Observe that $\Phi^+_{r+1} \setminus \Phi^+_r = \{\alpha_i+\alpha_{i+1}+\cdots+\alpha_{r+1} : 1 \leq i \leq r+1\}$. For $1 \leq i \leq r$, let $a_i = \alpha_i+\cdots+\alpha_{r+1}$ and let $a_{r+1} = \alpha_{r+1}$.

To find the number of ways to write $\tilde{\alpha} = \alpha_1 + \cdots + \alpha_{r+1}$ as a nonnegative integral sum of the positive roots in Φ^+_{r+1}, we add the number of ways to write $\tilde{\alpha}$ when we use one of the positive roots in $\Phi^+_{r+1} \setminus \Phi^+_r$. If we use a_1, then we can write $\tilde{\alpha} = \alpha_1 + \cdots + \alpha_{r+1}$ in exactly 1 way. Observe that for $2 \leq i \leq r+1$, if we use a_i, then we need to count the number of ways we can write $\tilde{\alpha} - a_i = \alpha_1 + \alpha_2 + \cdots + \alpha_{i-1}$, which by induction hypothesis, we can write in 2^{i-2} ways. Therefore, the total number of ways to write $\tilde{\alpha}$ as a sum of positive roots is given by the number of ways we can write $\tilde{\alpha} - a_i$ (where $1 \leq i \leq r+1$) as a sum of positive roots. Thus

$$\wp(\tilde{\alpha}) = 1 + 1 + 2 + \cdots + 2^{r-3} + 2^{r-2} + 2^{r-1} = 1 + \sum_{i=0}^{r-1} 2^i = 1 + \frac{1-2^r}{1-2} = 2^r.$$

Lemma 3. *Let $r \geq 2$. If $1 \leq i < j \leq r$, then $\wp(\alpha_i + \alpha_{i+1} + \cdots + \alpha_j) = 2^{j-i}$.*

Proof. We proceed by induction on r. If $r = 2$, then $i = 1, j = 2$ and by Lemma 2, $\wp(\alpha_1 + \alpha_2) = 2 = 2^{2-1}$. Assume that for any $k \leq r$, $\wp(\alpha_i + \alpha_{i+1} + \cdots + \alpha_j) = 2^{j-i}$, whenever $i, j \in \mathbb{N}$ satisfy $1 \leq i < j \leq k$. Let $k = r+1$, and let i, j be integers such that $1 \leq i < j \leq r+1$. If $j < r+1$, then by induction hypothesis $\wp(\alpha_i + \cdots + \alpha_j) = 2^{j-i}$. Now assume that $j = r+1$. Let Φ^+_{r+1} be the set of positive roots of $\mathfrak{sl}_{r+2}(\mathbb{C})$ and Φ^+_r be the set of positive roots of $\mathfrak{sl}_{r+1}(\mathbb{C})$. Observe that $\Phi^+_{r+1} \setminus \Phi^+_r = \{\alpha_i + \alpha_{i+1} + \cdots + \alpha_{r+1} : 1 \leq i \leq r+1\}$. For $1 \leq i \leq r$, let $a_i = \alpha_i + \cdots + \alpha_{r+1}$ and let $a_{r+1} = \alpha_{r+1}$.

We want to know the number of ways to write $\alpha_i + \cdots + \alpha_{r+1}$ as a nonnegative integral sum of the positive roots in Φ^+_{r+1}. We can compute this number by adding the number of ways to write $\alpha_i + \cdots + \alpha_{r+1}$ when we use one of the positive roots in $\Phi^+_{r+1} \setminus \Phi^+_r$. Notice that to write $\alpha_i + \cdots + \alpha_{r+1}$ we will not use any a_m for $1 \leq m \leq i-1$. If we use a_i, then we can write $\alpha_i + \cdots + \alpha_{r+1}$ in exactly 1 way. Observe that for $i+1 \leq m \leq r+1$, if we use a_m, then, by induction hypothesis, we can write $\alpha_i + \cdots + \alpha_{r+1}$ in 2^{m-1-i} ways.

Thus the total number of ways to write $\alpha_i + \cdots + \alpha_r + 1$ is given by

$$\wp(\alpha_i + \cdots + \alpha_{n+1}) = 1 + \sum_{i=0}^{r-i} 2^i = 1 + \frac{1-2^{r+1-i}}{1-2} = 2^{r+1-i}.$$

We can now return to the proof of Proposition 1.

Proof (Proposition 1). Assume that $r \geq 1$ and $\sigma \in \mathscr{A}(\tilde{\alpha}, 0)$, with $k = \ell(\sigma)$. We proceed by induction on k. If $\sigma \in \mathscr{A}(\tilde{\alpha}, 0)$ and $k = 0$, then $\sigma = 1$. Hence, by Lemma 2

$$\wp(\sigma(\tilde{\alpha} + \rho) - \rho) = \wp(\tilde{\alpha}) = 2^{r-1}.$$

If $\sigma \in \mathscr{A}(\tilde{\alpha}, 0)$ and $k = 1$, then $\sigma = s_i$ for some $2 \leq i \leq r-1$. Hence, $\sigma(\tilde{\alpha}+\rho)-\rho = \tilde{\alpha} - \alpha_i = \alpha_1 + \cdots + \alpha_{i-1} + \alpha_{i+1} + \cdots + \alpha_r$. By Lemma 3,

$$\wp(\alpha_1 + \cdots + \alpha_{i-1}) = 2^{i-2} \quad \text{and} \quad \wp(\alpha_{i+1} + \cdots + \alpha_r) = 2^{r-i-1}.$$

Since the subset of the positive roots used to write $\alpha_1 + \cdots + \alpha_{i-1}$ is disjoint from the subset of the positive roots used to write $\alpha_{i+1} + \cdots + \alpha_n$, we have

$$\begin{aligned}\wp(\alpha_1 + \cdots + \alpha_{i-1} + \alpha_{i+1} + \cdots + \alpha_r) &= \wp(\alpha_1 + \cdots + \alpha_{i-1}) \cdot \wp(\alpha_{i+1} + \cdots + \alpha_r) \\ &= 2^{i-2} \cdot 2^{r-i-1} \\ &= 2^{r-1-2(1)}.\end{aligned}$$

Assume that for any $\sigma \in \mathscr{A}(\tilde{\alpha}, 0)$ with $\ell(\sigma) \leq k$, $\wp(\sigma(\tilde{\alpha}+\rho)-\rho) = 2^{r-1-2\ell(\sigma)}$. If $\sigma \in \mathscr{A}(\tilde{\alpha}, 0)$ with $\ell(\sigma) = k+1$, then $\sigma = s_{i_1} s_{i_2} \cdots s_{i_k} s_{i_{k+1}}$ for some nonconsecutive integers $i_1, \ldots, i_{k+1}$ satisfying $2 \leq i_1 < i_2 < \cdots < i_k < i_{k+1} \leq r-1$. By Lemma 1,

$$\sigma(\tilde{\alpha}+\rho)-\rho = \sum_{j=1}^{i_1-1} \alpha_j + \sum_{j=i_1+1}^{i_2-1} \alpha_j + \sum_{j=i_2+1}^{i_3-1} \alpha_j + \cdots + \sum_{j=i_k+1}^{i_{k+1}-1} \alpha_j + \sum_{j=i_{k+1}+1}^{r} \alpha_j. \tag{28}$$

We need to determine the number of ways to write equation (28) as a nonnegative integral sum of positive roots. By Lemma 3, we know

$$\begin{aligned}\wp\left(\sum_{j=1}^{i_1-1} \alpha_j\right) &= 2^{i_1-2}, \\ \wp\left(\sum_{j=i_1+1}^{i_2-1} \alpha_j\right) &= 2^{(i_2-1)-(i_1+1)} = 2^{i_2-i_1-2}, \\ \wp\left(\sum_{j=i_2+1}^{i_3-1} \alpha_j\right) &= 2^{(i_3-1)-(i_2+1)} = 2^{i_3-i_2-2}, \\ &\vdots \\ \wp\left(\sum_{j=i_k+1}^{i_{k+1}-1} \alpha_j\right) &= 2^{(i_{k+1}-1)-(i_k+1)} = 2^{i_{k+1}-i_k-2}, \\ \wp\left(\sum_{j=i_{k+1}+1}^{r} \alpha_j\right) &= 2^{r-(i_{k+1}+1)} = 2^{r-i_{k+1}-1}.\end{aligned}$$

Since the subsets of the positive roots of $\mathfrak{sl}_{r+1}(\mathbb{C})$ used to write each of the above sums are pairwise disjoint we have that

$$\begin{aligned}\wp(\sigma(\tilde{\alpha}+\rho)-\rho) &= \wp\left(\sum_{j=1}^{i_1-1}\alpha_j+\sum_{j=i_1+1}^{i_2-1}\alpha_j+\sum_{j=i_2+1}^{i_3-1}\alpha_j+\cdots+\sum_{j=i_k+1}^{i_{k+1}-1}\alpha_j+\sum_{j=i_{k+1}+1}^{r}\alpha_j\right)\\ &= \wp\left(\sum_{j=1}^{i_1-1}\alpha_j\right)\wp\left(\sum_{j=i_1+1}^{i_2-1}\alpha_j\right)\cdots\wp\left(\sum_{j=i_k+1}^{i_{k+1}-1}\alpha_j\right)\wp\left(\sum_{j=i_{k+1}+1}^{r}\alpha_j\right)\\ &= 2^{i_1-2}\cdot 2^{i_2-i_1-2}\cdots 2^{i_{k+1}-i_k-2}\cdot 2^{r-i_{k+1}-1}\\ &= 2^{r-1-2(k+1)}.\end{aligned}$$

Lemma 4. *Let $r\geq 1$ and let $\tilde{\alpha}$ denote the highest root of $\mathfrak{sl}_{r+1}$. Then the cardinality of the set $\{\sigma\in\mathscr{A}(\tilde{\alpha},0)\mid \ell(\sigma)=k\}$ is $\binom{r-1-k}{k}$ and $max\{\ \ell(\sigma)\mid\sigma\in\mathscr{A}(\tilde{\alpha},0)\}=\lfloor\frac{r-1}{2}\rfloor$.*

Proof. The proof that $|\{\sigma\in\mathscr{A}(\tilde{\alpha},0)\}|=\binom{r-1-k}{k}$ follows from showing that the number of ways to select k nonconsecutive integers from the set $\{2,3,\ldots,r-1\}$ is given by $\binom{r-1-k}{k}$. We leave this proof to the reader. Now notice that if r is odd, then we can choose at most $\frac{r-1}{2}$ many nonconsecutive integers from the set $\{2,3,4,\ldots,r-1\}$, namely the even numbers. If r is even, then $r-1$ is odd and we can choose at most $\frac{r-2}{2}$ many nonconsecutive integers from the set $\{2,3,4,\ldots,r-1\}$, either all the even or all the odd numbers. Observe that when r is odd, $\frac{r-1}{2}=\lfloor\frac{r-1}{2}\rfloor$ and when r is even, $\frac{r-2}{2}=\lfloor\frac{r-1}{2}\rfloor$. Thus $max\{\ell(\sigma:\sigma\in\mathscr{A}(\tilde{\alpha},0))\}=\lfloor\frac{r-1}{2}\rfloor$.

We can now prove that $m(\tilde{\alpha},0)=r$.

Proof (Theorem 1). By Theorem 2 we can reduce

$$\begin{aligned}m(\tilde{\alpha},0) &= \sum_{\sigma\in W}(-1)^{\ell(\sigma)}\wp(\sigma(\tilde{\alpha}+\rho)-\rho)\\ &= \sum_{\sigma\in\mathscr{A}(\tilde{\alpha},0)}(-1)^{\ell(\sigma)}\wp(\sigma(\tilde{\alpha}+\rho)-\rho)\end{aligned}\tag{29}$$

where $\sigma\in\mathscr{A}(\tilde{\alpha},0)$ if and only if $\sigma=s_{i_1}s_{i_2}\cdots s_{i_k}$ for some nonconsecutive integers satisfying $2\leq i_1<i_2<\cdots<i_k\leq r-1$. Now by Proposition 1 we know that for any $\sigma\in\mathscr{A}(\tilde{\alpha},0)$, $\wp(\sigma(\tilde{\alpha}+\rho)-\rho)=2^{r-1-2\ell(\sigma)}$. Let $k=\ell(\sigma)$, then by Lemma 4 we know that $\min\{k:\sigma\in\mathscr{A}(\tilde{\alpha},0)\}=0$, $\max\{k:\sigma\in\mathscr{A}(\tilde{\alpha},0)\}=\lfloor\frac{r-1}{2}\rfloor$, and there are $\binom{r-1-k}{k}$ elements in $\mathscr{A}(\tilde{\alpha},0)$ of length k. Hence, (29) reduces to

$$m(\tilde{\alpha},0)=\sum_{k=0}^{\lfloor\frac{r-1}{2}\rfloor}(-1)^k2^{r-1-2k}.\tag{30}$$

A proof by induction verifies that

$$\sum_{k=0}^{\lfloor \frac{r-1}{2} \rfloor} (-1)^k 2^{r-1-2k} = r.$$

Having developed some experience computing Weyl alternation sets and creating formulas for the value of Kostant's partition function for special sets of weights, we now focus on a more geometric style of problem. In Example 1 we focused our attention on a problem where we fixed both the weights λ and μ, but this is only necessary if we are actually concerned with computing weight multiplicities. For the sake of discovery, let us investigate what occurs to the Weyl alternation sets $\mathscr{A}(\lambda, \mu)$ when we fix $\mu = 0$ and let λ vary among the set of integral weights of $\mathfrak{sl}_3(\mathbb{C})$.

Example 2. Compute the sets $\mathscr{A}(\lambda, 0)$ for all integral weights λ of $\mathfrak{sl}_3(\mathbb{C})$.

Solution 2. By (6), λ is an integral weight if $\lambda = c_1\varpi_1 + c_2\varpi_2$ for some $c_1, c_2 \in \mathbb{Z}$. There is a nice relationship between the simple roots α_1 and α_2 and the fundamental weights ϖ_1 and ϖ_2,

$$\varpi_1 = \frac{2}{3}\alpha_1 + \frac{1}{3}\alpha_2 \tag{31}$$

$$\varpi_2 = \frac{1}{3}\alpha_1 + \frac{2}{3}\alpha_2. \tag{32}$$

Figure 3 provides a visualization of the fundamental weights and the simple roots on a 2-dimensional plane. The shaded region represents the dominant Weyl chamber, and the weights lying on the lattice $\varpi_1\mathbb{N} + \varpi_2\mathbb{N}$ index the irreducible finite-dimensional representations of the Lie algebra $\mathfrak{sl}_3(\mathbb{C})$. This was the statement of the theorem of the highest weight.

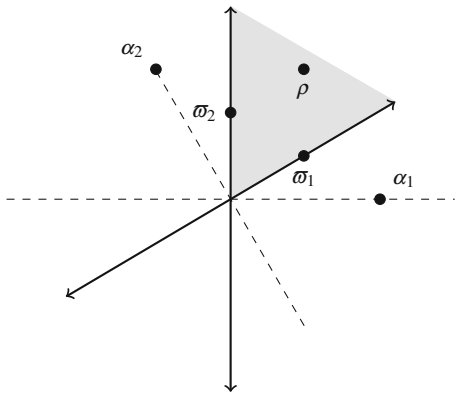


Fig. 3 Fundamental weights in terms of simple roots.

Since we want to describe the sets $\mathscr{A}(c_1\varpi_1 + c_2\varpi_2, 0)$ for all $c_1, c_2 \in \mathbb{Z}$, we first have to compute $\sigma(c_1\varpi_1 + c_2\varpi_2 + \rho) - \rho$ for all $\sigma \in W$. To do this, first write the fundamental weights and ρ in terms of the simple roots, then use Table 1 to determine $\sigma(c_1\varpi_1 + c_2\varpi_2 + \rho) - \rho$ for all $\sigma \in W$. Writing the result in terms of the simple roots, as we will need this information shortly, the reader should complete:

Exercise 3. Verify the following computations

$$1(c_1\varpi_1 + c_2\varpi_2 + \rho) - \rho = \left(\frac{2c_1 + c_2}{3}\right)\alpha_1 + \left(\frac{c_1 + 2c_2}{3}\right)\alpha_2 \tag{33}$$

$$s_1(c_1\varpi_1 + c_2\varpi_2 + \rho) - \rho = \left(\frac{-c_1 + c_2 - 3}{3}\right)\alpha_1 + \left(\frac{c_1 + 2c_2}{3}\right)\alpha_2 \tag{34}$$

$$s_2(c_1\varpi_1 + c_2\varpi_2 + \rho) - \rho = \left(\frac{2c_1 + c_2}{3}\right)\alpha_1 + \left(\frac{c_1 - c_2 - 3}{3}\right)\alpha_2 \tag{35}$$

$$s_1s_2(c_1\varpi_1 + c_2\varpi_2 + \rho) - \rho = \left(\frac{-c_1 - 2c_2 - 6}{3}\right)\alpha_1 + \left(\frac{c_1 - c_2 - 3}{3}\right)\alpha_2 \tag{36}$$

$$s_2s_1(c_1\varpi_1 + c_2\varpi_2 + \rho) - \rho = \left(\frac{-c_1 + c_2 - 3}{3}\right)\alpha_1 + \left(\frac{-2c_1 - c_2 - 6}{3}\right)\alpha_2 \tag{37}$$

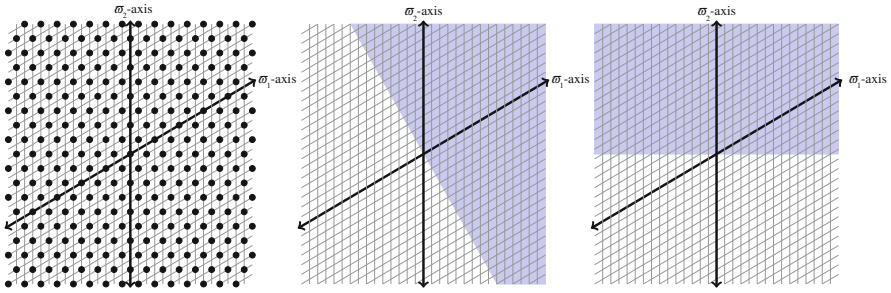
$$s_1s_2s_1(c_1\varpi_1 + c_2\varpi_2 + \rho) - \rho = \left(\frac{-c_1 - 2c_2 - 6}{3}\right)\alpha_1 + \left(\frac{-2c_1 - c_2 - 6}{3}\right)\alpha_2. \tag{38}$$

As we noted in the solution to Example 1, in order to determine when an element of the Weyl group is in the Weyl alternation set $\mathscr{A}(c_1\varpi_1 + c_2\varpi_2, 0)$ we must find when both of the coefficients of α_1 and α_2 in the above equations are nonnegative integers. For example, considering (33), the identity element will be in the Weyl alternation set $\mathscr{A}(c_1\varpi_1 + c_2\varpi_2, 0)$ if and only if

$$\frac{2c_1 + c_2}{3} \quad \text{and} \quad \frac{c_1 + 2c_2}{3}$$

are nonnegative integers. We expand these conditions as follows:

Condition 1: $3|2c_1 + c_2$
Condition 2: $3|c_1 + 2c_2$
Condition 3: $2c_1 + c_2 \geq 0$ and
Condition 4: $c_1 + 2c_2 \geq 0$.



(a) Condition 1 and 2 (b) Condition 3 (c) Condition 4

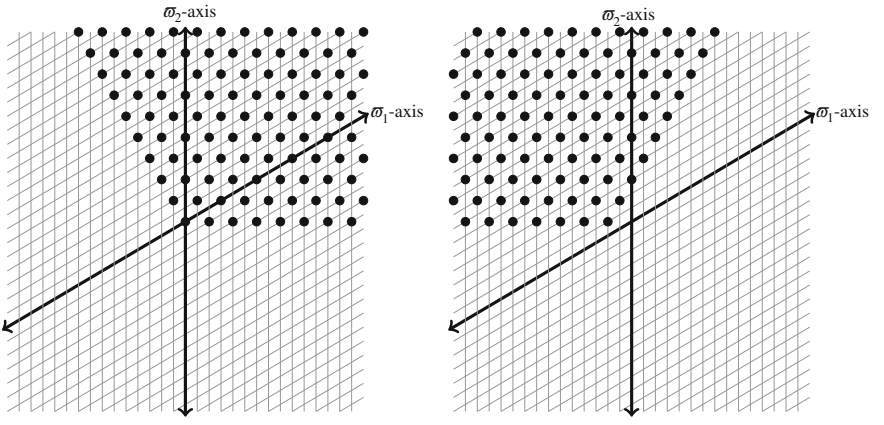
Fig. 4 Illustrating Conditions 1-4, so that $1 \in \mathscr{A}(c_1\varpi_1 + c_2\varpi_2, 0)$.

Observe that Condition 1 and Condition 2 are number theoretic conditions. We can simplify them by observing that since $3|2c_1 + c_2$ and $3|c_1 + 2c_2$, then $3|(2c_1 + c_2) - (c_1 + 2c_2)$, hence $3|c_1 - c_2$. This implies that $c_1 - c_2 = 3x$ for some integer x and so $c_1 = 3x + c_2$. Letting $c_2 = y$, we note that the only way Condition 1 and Condition 2 can be satisfied is if $\lambda = c_1\varpi_1 + c_2\varpi_2$ was actually $\lambda = (3x + y)\varpi_1 + y\varpi_2$ for some $x, y \in \mathbb{Z}$. The points that satisfy this condition are provided in Figure 4a, where the lattice is given by $\mathbb{Z}\varpi_1 + \mathbb{Z}\varpi_2$. Now note that Condition 3 and Condition 4 are linear inequalities whose solutions, as is standard, can be depicted by shading the appropriate region of the plane in Figure 3. Figures 4b and 4c provide the appropriately shaded regions. Since all four conditions must be satisfied, we find that the identity element will be in $\mathscr{A}(c_1\varpi_1 + c_2\varpi_2, 0)$ if and only if $c_1\varpi_1 + c_2\varpi_2$ is one of the integral weights we have highlighted by placing a black circle on the weights location within the plane in Figure 5a.

Repeating this process for each $\sigma \in W$, we determine the weights $c_1\varpi_1 + c_2\varpi_2$, with $c_1, c_2 \in \mathbb{Z}$, for which $\sigma \in \mathscr{A}(c_1\varpi_1 + c_2\varpi_2, 0)$. This is depicted in Figure 5, and each subfigure presented is found in an analogous way as that presented above for Figure 5a. Moreover, each subfigure shows which weights $c_1\varpi_1 + c_2\varpi_2$ would have a given Weyl group element in their Weyl alternation set $\mathscr{A}(c_1\varpi_1 + c_2\varpi_2, 0)$. We leave the verification of these computations to the reader.

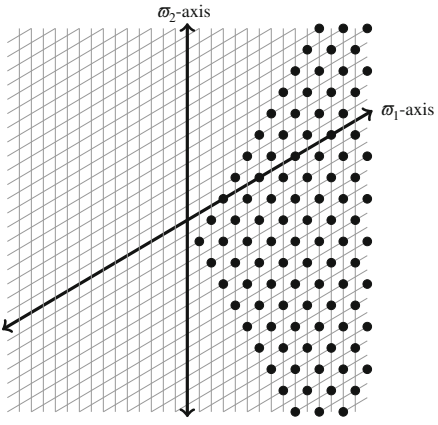
From Figure 5 one can observe that there are weights that have no elements in their Weyl alternation set as they are never highlighted in any of the subfigures, i.e. they have no black circles placed on top of the weights location within the plane. Take for example the dominant integral weight $2\varpi_1$. This weight is not highlighted in any of the subfigures of Figure 5. This implies that $\mathscr{A}(2\varpi_1, 0) = \emptyset$ and hence $m(2\varpi_1, 0) = 0$.

Figure 5 also shows that there are weights that have a few elements of the Weyl group in their Weyl alternation set. Those are weights that are highlighted in more than one of the subfigures. Take for example the dominant integral weight $3\varpi_1$. This weight is highlighted in Figures 5a and 5c, and not highlighted in the rest. This implies that $\mathscr{A}(3\varpi_2, 0) = \{1, s_2\}$.

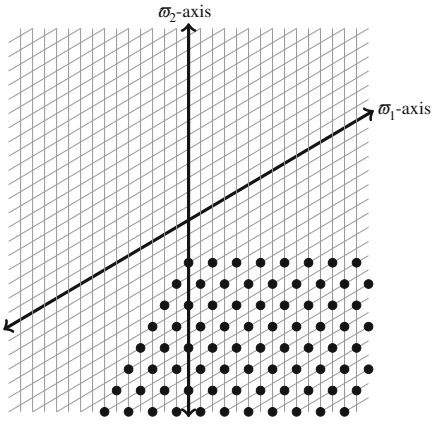


(a) Weights with $1 \in \mathscr{A}(\lambda, 0)$

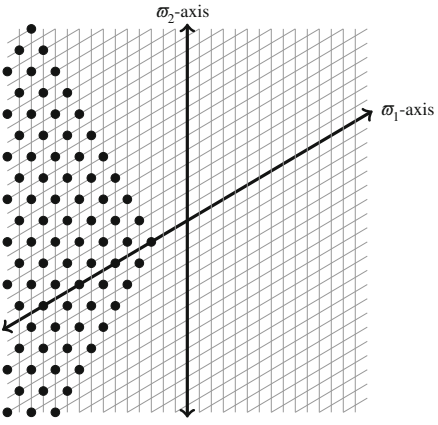
(b) Weights with $s_1 \in \mathscr{A}(\lambda, 0)$



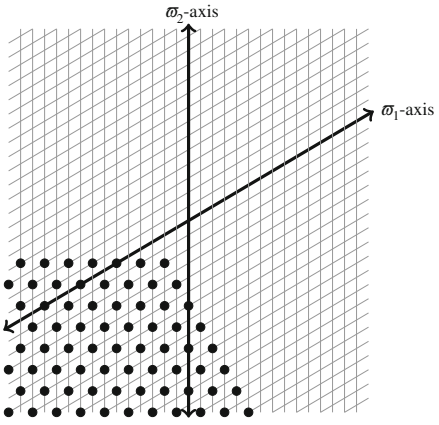
(c) Weights with $s_2 \in \mathscr{A}(\lambda, 0)$



(d) Weights with $s_1 s_2 \in \mathscr{A}(\lambda, 0)$



(e) Weights with $s_2 s_1 \in \mathscr{A}(\lambda, 0)$



(f) Weights with $s_1 s_2 s_1 \in \mathscr{A}(\lambda, 0)$

Fig. 5 Weights for which $\sigma \in \mathscr{A}(\lambda, 0)$ where $\lambda = c_1\varpi_1 + c_2\varpi_2$ and $c_1, c_2 \in \mathbb{Z}$.

These observations motivate the need for a single diagram which provides all of the information regarding the Weyl alternation sets $\mathscr{A}(c_1\varpi_1 + c_2\varpi_2, 0)$, where $c_1, c_2 \in \mathbb{Z}$. This is exactly the definition of the *zero weight Weyl alternation diagram*. This diagram places a circle of the same color on the integral weights λ and β whenever $\mathscr{A}(\lambda, 0) = \mathscr{A}(\beta, 0) \neq \emptyset$. That is weights with the same Weyl alternation sets get colored the same way, and weights whose Weyl alternation set is the empty set do not get colored at all. Following this construction, the zero weight Weyl alternation diagram for the Lie algebra $\mathfrak{sl}_3(\mathbb{C})$ is found in Figure 6. This diagram allows one to quickly reduce a weight's multiplicity computation as it explicitly lists the elements of the Weyl group that provide nonzero terms to the sum in the multiplicity formula.

We remark that the computation of weight multiplicities only make sense in a Lie theoretic way when λ is a dominant integral weight. That is, when $\lambda = c_1\varpi_1 + c_2\varpi_2$ for some $c_1, c_2 \in \mathbb{N}$. Whenever λ is a dominant integral weight, it represents a finite-dimensional irreducible representation of the Lie algebra $\mathfrak{sl}_3(\mathbb{C})$. However, other integral weights can be viewed as "virtual" representations, where we focus solely on the computation and not on representation theory. Moreover, by considering the Weyl alternation sets of all integral weights, we discovered the beautiful symmetry presented in Figure 6. A symmetry we would have missed had we restricted the computation of $\mathscr{A}(\tilde{\alpha}, 0)$ for only dominant integral weights λ of $\mathfrak{sl}_3(\mathbb{C})$.

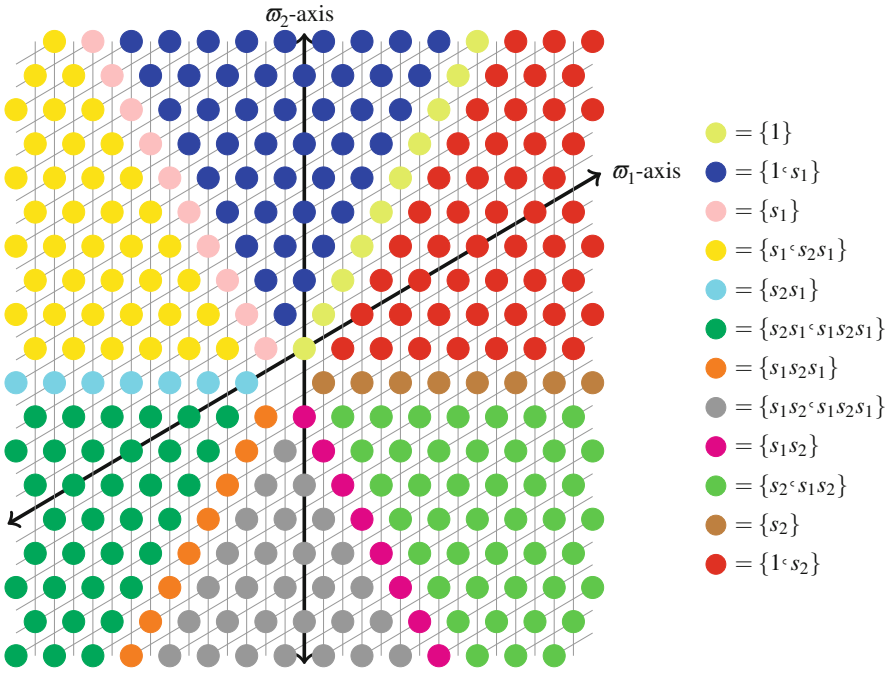


Fig. 6 Weyl alternation diagram for the sets $\mathscr{A}(c_1\varpi_1 + c_2\varpi_2, 0)$, where $c_1, c_2 \in \mathbb{Z}$

4 Open Problems

Having worked through the detailed examples and proofs provided in this paper, students can now focus on the computation of weight multiplicities to study the representation theory of Lie algebras. More concretely, students can use techniques from combinatorics to:

1. Provide descriptions of Weyl alternation sets $\mathscr{A}(\lambda, \mu)$ for integral weights λ and μ in the Lie algebra $\mathfrak{sl}_{r+1}(\mathbb{C})$;
2. Compute Weyl alternation diagrams for some low rank examples; and
3. Find closed formulas for the value of Kostant's partition function, as well as Kostant's weight multiplicity formula for specific weights λ and μ.

Before stating the first (concrete) open problem of this section, we begin by sharing the work of Brett Harder, an undergraduate at Moravian College, whose honors thesis focused on exploring the intricacies involved in computing weight multiplicities for $\mathfrak{sl}_4(\mathbb{C})$ [11]. In his thesis, Harder formulated and proved results on specific families of Weyl alternation sets including:

Theorem 3 (Theorem 5.1 [11]). *If $\tilde{\alpha}$ the highest root of $\mathfrak{sl}_4(\mathbb{C})$, then $\mathscr{A}(n\tilde{\alpha}, 0) = \{1, s_2\}$ for all $n \in \mathbb{N}$.*

Theorem 4 (Theorem 6.4 [11]). *If $\tilde{\alpha}$ the highest root of $\mathfrak{sl}_4(\mathbb{C})$, then*

$$\mathscr{A}(\tilde{\alpha}, n\alpha_i) = \begin{cases} \{1, s_2\} & \text{if } n = 1 \text{ and } i = 1, 3 \\ \{1\} & \text{if } n = 1 \text{ and } i = 2 \\ \emptyset & \text{if } n > 1 \text{ and } 1 \leq i \leq 3. \end{cases}$$

Theorem 5 (Theorem 7.6 [11]). *For the fundamental weights $\varpi_1, \varpi_2, \varpi_3$ of $\mathfrak{sl}_4(\mathbb{C})$,*

1. *$\mathscr{A}(n\varpi_1 + n\varpi_2 + n\varpi_3, 0) = \{1, s_1, s_2, s_3, s_1s_3\}$ for all even $n \in \mathbb{N}$;*
2. *$\mathscr{A}(n\varpi_1 + n\varpi_2 + n\varpi_3, 0) = \{s_1s_2s_3s_2s_1, s_2s_3s_1s_2s_1, s_1s_2s_3s_1s_2, s_1s_2s_3s_2s_1s_2\}$ for all even integers $n < -2$; and*
3. *$\mathscr{A}(n\varpi_1 + n\varpi_2 + n\varpi_3, 0) = \emptyset$ for $n = -2$ and for all odd $n \in \mathbb{Z}$.*

With the above results at hand and some supporting computational evidence, Harder conjectured the following.

Conjecture 1 (Conjecture 5.2 [11]). If $\tilde{\alpha}$ is the highest root of $\mathfrak{sl}_4(\mathbb{C})$, then

$$m(n\tilde{\alpha}, 0) = \frac{(n+1)(n+2)}{2} \quad \text{for all} \quad n \in \mathbb{N}.$$

Conjecture 2 (Conjecture 6.5 [11]). If $\tilde{\alpha}$ is the highest root of $\mathfrak{sl}_4(\mathbb{C})$, then for $1 \leq i \leq 4$,

$$m(\tilde{\alpha}, n\alpha_i) = \begin{cases} 1 & \text{if } n = 1 \\ 0 & \text{for } n > 1. \end{cases}$$

Conjecture 3 (Conjectures 7.3 and 7.5 [11]). For even $n \in \mathbb{N}$,

$$m(n\varpi_1 + n\varpi_2 + n\varpi_3, 0) = \left(\frac{1}{2}n + 1\right)^4 - \left(\frac{1}{2}n\right)^4$$

and for even $n < -2$,

$$m(n\varpi_1 + n\varpi_2 + n\varpi_3, 0) = \left|\left(\frac{1}{2}n + 1\right)^4 - \left(\frac{1}{2}n\right)^4\right|.$$

A starting point for a student who desires to contribute to this body of knowledge is:

Research Project 1. Prove or provide counterexamples to Conjectures 1–3.

For those students more interested in computing Weyl alternation diagrams or those with some programming background, we suggest:

Research Project 2. Consider the Lie algebra $\mathfrak{sl}_4(\mathbb{C})$. Fix $\mu = 0$ and compute all Weyl alternation sets $\mathscr{A}(\lambda, 0)$ where $\lambda = c_1\varpi_1 + c_2\varpi_2 + c_3\varpi_3$ and $c_1, c_2, c_3 \in \mathbb{Z}$. From these sets create the zero weight Weyl alternation diagram for $\mathfrak{sl}_4(\mathbb{C})$.

We note that the above mentioned Weyl alternation diagram will be a three dimensional plot and the use of software is highly encouraged.

Research Project 3. Consider the Lie algebra $\mathfrak{sl}_3(\mathbb{C})$. Create a program whose input is a pair of weights λ, μ and whose output is the associated μ weight Weyl alternation diagram.

Consider the Lie algebra $\mathfrak{sl}_{r+1}(\mathbb{C})$. As we saw in Corollary 1, the cardinality of $\mathscr{A}(\tilde{\alpha}, 0)$ is F_r, where F_r denotes the r^{th} Fibonacci number. This motivates the following.

Research Project 4. Determine all pairs of weights λ, μ for which $\mathscr{A}(\lambda, \mu) = F_r$ as the rank of the Lie algebra $\mathfrak{sl}_{r+1}(\mathbb{C})$ increases.

Whether this is even possible for any other pair of weights aside from the highest root $\tilde{\alpha}$ and the zero weight, is currently unknown.

Research Project 5. Berenshtine and Zelevinskii provide a list of pairs of weights λ, μ for which $m(\lambda, \mu) = 1$, see [3]. An open problem is to describe the sets $\mathscr{A}(\lambda, \mu)$ for the pairs of weights λ and μ in these lists.

These are only a few of the open problem in this field of study. The present work focused on the Lie algebra $\mathfrak{sl}_{r+1}(\mathbb{C})$, however, all of the work and questions we have investigated can be extended to the other simple Lie algebras. This opens the door for numerous other variations of these problems and extends the investigation further than is currently presented.

Acknowledgements The author thanks the reviewer for their insightful comments and suggestions, which greatly improved the exposition of this manuscript. The author also thanks Kevin Chang, Edward Lauber, Haley Lescinsky, Grace Mabie, Gabriel Ngwe, Cielo Perez, Aesha Siddiqui, and Anthony Simpson for reading and providing feedback on an initial draft of this manuscript.

References

1. Baldoni, W., Vergne, M.: Kostant partitions functions and flow polytopes. Transform. Groups **13**(3–4), 447–469 (2008)
2. Baldoni, W., Beck, M., Cochet, C., Vergne, M.: Volume computation for polytopes and partition functions for classical root systems. Discret. Comput. Geom. **35**, 551–595 (2006). Programs available at www.math.polytechnique.fr/cmat/vergne
3. Berenshtein, A.D., Zelevinskii, A.V.: When is the multiplicity of a weight equal to 1? Funct. Anal. Appl. **24**, 259–269 (1991)
4. Billey, S., Guillemin, V., Rassart, E.: A vector partition function for the multiplicities of $\mathfrak{sl}_k\mathbb{C}$. J. Algebra **278**(1), 251–293 (2004)
5. Björner, A., Brenti, F.: Combinatorics of Coxeter groups. Springer, Berlin (2005)
6. Cartan, E.: Sur la structure des groupes de transformations finis et continus. Thèse, Paris, Nony (1894) [Ouvres Completes, Partie I. **1**, 137–287]

7. Carter, R.W.: Finite Groups of Lie Type: Conjugacy Classes and Complex Characters. Wiley Classics Library. Wiley, Chichester (1993)
8. Cochet, C.: Vector partition function and representation theory. In: Conference Proceedings Formal Power Series and Algebraic Combinatorics, Taormina, Sicile (2005), 12 pp.
9. Fulton, W., Harris, J.: Representation Theory - A First Course. Graduate Texts in Mathematics, vol. 129. Springer, Berlin (2004)
10. Goodman, R., Wallach, N.R.: Symmetry, Representations and Invariants. Springer, New York (2009)
11. Harder, B.: An exploration of lie algebras and Kostant's weight multiplicity formula. Moravian College, Undergraduate Honors Thesis (2016)
12. Harish-Chandra: On some applications of the universal enveloping algebra of a semi-simple lie algebra. Trans. Am. Math. Soc. **70**, 28–96 (1951)
13. Harris, P.E.: On the adjoint representation of $\mathfrak{sl}_n$ and the Fibonacci numbers. C. R. Math. Acad. Sci. Paris 935–937 (2011). arXiv:1106.1408
14. Harris, P.E.: Kostant's weight multiplicity formula and the Fibonacci numbers. http://arxiv.org/pdf/1111.6648v1.pdf
15. Harris, P.E.: Combinatorial problems related to Kostant's weight multiplicity formula. Doctoral dissertation. University of Wisconsin-Milwaukee, Milwaukee, WI (2012)
16. Harris, P.E., Insko, E., Omar, M.: The q-analog of Kostant's partition function and the highest root of the classical Lie algebras. Preprint http://arxiv.org/pdf/1508.07934 (Submitted)
17. Harris, P.E., Insko, E., Williams, L.K.: The adjoint representation of a classical Lie algebra and the support of Kostant's weight multiplicity formula . J. Comb. **7**(1), 75–116 (2016)
18. Humphreys, J.E.: Introduction to Lie Algebras and Representation Theory. Springer, New York (1978)
19. Humphreys, J.E.: Reflection Groups and Coxeter Groups. Cambridge University Press, Cambridge (1990)
20. Knapp, A.W.: Lie Groups Beyond an Introduction. Birkhäuser Boston Inc. Boston, MA (2002)
21. Kostant, B.: A formula for the multiplicity of a weight. Proc. Natl. Acad. Sci, U. S. A. **44**, 588–589 (1958)
22. Núñez, J.F., Fuertes, W.G., Perelomov, A.M.: Generating functions and multiplicity formulas: the case of rank two simple Lie algebras. Preprint: http://arxiv.org/pdf/1506.07815v1.pdf
23. Varadarajan, V.S.: Lie Groups, Lie Algebras, and Their Representations. Springer, Berlin (1984)
24. Weyl, H.: Theorie der Darstellung kontinuierlicher halbeinfacher Gruppen durch lineare Transformationen, I, II, III, Nachtrag, Mathematische Zeitschrift **23** (1925), **24** 271–309/328–376/377–395/789–791 (1926)

Vaccination Strategies for Small Worlds

Winfried Just and Hannah Callender Highlander

Suggested Prerequisites. *Knowledge of basic notions of probability theory and prior exposure to theorem proving are essential. Some acquaintance with graph theory, mathematical epidemiology, as well as experience with mathematical modeling and simulations will be helpful, but are not required. Two of the five suggested research projects involve coding in the* NETLOGO *language.*

1 Introduction

The goal of this chapter is to empower students to work on the research projects described in Section 6. Each of these projects concerns optimal vaccination strategies. In plain language this means that we are asking, in a certain mathematically well-defined setting, how one should choose individuals who will receive vaccination that is in short supply so as to achieve maximal overall protection against a disease outbreak for the entire population. Thus the questions of our projects are closely related to important issues in public health, fairly intuitive, and natural. Nevertheless, to the best of our knowledge, none of these projects has been covered to a large extent by existing work in the literature. Each of our proposed projects is open-ended and has a broad scope that leaves room for genuinely novel discoveries by aspiring undergraduate researchers, with relatively basic mathematical background on the one hand, and also for deep mathematical theorems on the other hand.

W. Just (✉)
Department of Mathematics, Ohio University, Athens, OH 45701, USA
e-mail: mathjust@gmail.com

H.C. Highlander
Department of Mathematics, University of Portland, MSC 60, Portland, OR 97203, USA
e-mail: highland@up.edu

A. Wootton et al. (eds.), *A Primer for Undergraduate Research*, Foundations for Undergraduate Research in Mathematics, DOI 10.1007/978-3-319-66065-3_10

While none of our projects require highly specialized mathematical knowledge at a very high level of abstraction, they do require some familiarity with a number of concepts and techniques. Some background knowledge of mathematical epidemiology and graph theory will be needed, but is not assumed here. In Section 1.1 we give a bird's-eye overview of these topics, while in Section 1.2 we give pointers to sources where students can study this material in greater depth. Sections 2–5 introduce four crucial notions from graph theory that are essential for our projects: Erdős-Rényi random graphs, clustering coefficients, the small-world property, and small-world networks. References to more in-depth readings about these concepts will be given throughout these sections. Most of the material in Sections 1–5 has been adapted from the modules that were developed in cooperation with Drew LaMar and Ying Xin and are posted at QUBEShub.org [14]. For ease of browsing, a complete list of the modules are also posted on the first author's website [12]. The excerpts are chosen in such a way that they directly build up towards the research projects. In Section 6 we describe the projects themselves.

Preparation for research requires not only absorbing a number of facts, but also mastering certain skills. Our presentation includes numerous exercises that we recommend be treated as an integral part of the text and attempted right away. On the one hand, they are designed to deepen the understanding of the concepts covered in the text; on the other hand, they will develop the same skills that are needed for success with the research projects. Among these skills is familiarity with using simulations for exploring predictions of models. In order to keep the exposition focused on the concepts, we did not include a separate "how-to" section on simulation studies in general and using our recommended software tool IONTW in particular. Instead, we rely on students consulting the relevant references to such material, and we provide ample opportunity to practice these skills in our exercises and three challenge problems. Hints for selected exercises are provided in the appendix. Solutions to the exercises that were adapted from the modules can be provided upon request by research advisors at [14].

1.1 Modeling Infectious Diseases and Contact Networks

Epidemiology studies the spread of diseases caused by *pathogens,* such as viruses or bacteria, in *populations* of *hosts,* which can be humans, animals, or plants. We focus here on diseases that are transmitted by *direct contact* between two hosts[1]. The goal is to *predict* the time course of an *outbreak* of a given disease in a population and the effect of conceivable *control measures,* such as vaccination or behavior modification, on the severity of the outbreak.

Mathematical models can help answer these questions. Based on implementation of such models in computer code, epidemiologists can *simulate* hypothetical out-

[1]This leaves out vector-born infectious diseases such as malaria and diseases such as cholera that are transmitted through shared environmental resources.

breaks under various assumptions about control that might possibly be implemented and derive predictions about their likely effectiveness. For our simulations we will use the customized software tool IONTW built upon the NETLOGO agent-based modeling platform [28]. The IONTW code was developed by Drew LaMar in consultation with us.

It is important to remember that mathematical models are greatly simplified representations of reality. Thus when making inferences to the spread of infections in populations of actual hosts, one needs to carefully examine how the assumptions of the models influence their predictions. IONTW allows us to study how the simplifying assumptions about the contact pattern that are embodied in the *contact network* influence the outcomes of simulations. It also can be used for exploring the *sensitivity* of these outcomes with respect to the disease transmission parameters of the model.

A standing assumption of our modeling is that we investigate the spread of one given infectious disease within a fixed population of hosts that are numbered from 1 to N or from 0 to $N-1$. Thus we ignore *demographics* (births, deaths from causes that are unrelated to the disease, immigration, and emigration). At any given time t, host i can be in one of the following states:

- S: *Susceptible* to infection.
- I: *Infectious,* that is, able to infect other hosts through direct contact.
- R: *Removed,* that is, not infectious and immune to subsequent infection (through recovery or death from the disease, or through vaccination).

The sets of hosts that are in states S, I, R will be referred to as the ***S**-compartment,* the ***I**-compartment,* and the ***R**-compartment,* respectively. Membership in the compartments changes over time, and one can conceptualize the time course of an outbreak as *movement of hosts between compartments.*

Here we mostly limit ourselves to the study of *SIR-models* that are suitable for *immunizing infections* where recovery from the disease confers permanent immunity. Depending on the specifics of the disease under study, models of other types, such as SIS, may be more appropriate; or a model may need to include an Exposed state, where a host has been exposed to the pathogen but is not yet infectious.

Figure 1 schematically depicts models of type *SIR*. Hosts move from the **S**- to **I**-compartment upon *effective* contact. An effective contact between hosts i and j is a contact that would result in infection of host i *if* host i were susceptible and host j infectious at the time of the contact. We assume that for any given pair (i,j) of hosts the probability of at least one effective contact over a time interval of length Δt remains fixed, and hosts make contacts independently. In discrete-time models this probability is denoted by $b_{i,j}$, and Δt is taken as the physical time that corresponds to one time step in the model. In continuous-time models we assume that hosts i and j make effective contact at a fixed rate $\beta_{i,j}$.

Similarly, in discrete-time models we assume that a host who is infectious at time t will cease to be infectious at time $t+1$ (Δt units of physical time later) with probability a, which is fixed and the same for all hosts. In continuous-time models we assume that hosts will leave the **I**-compartment at a fixed rate α.

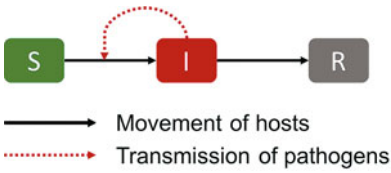


Fig. 1 Schematic representation of *SIR*-models.

When $a = 0$ or $\alpha = 0$, then infectious hosts never recover, and we obtain models of *type SI*. Such models are useful in some explorations of network structure. Discrete-time models with $a = 1$ assume that hosts stay infectious for exactly one time step. Since in these models there is a strict correspondence between the time course of an outbreak and the so-called generations of the infection, we call them *next-generation models.*

This short background review barely scratches the surface of disease modeling. For more detailed information on how individuals transition from one state to another, including a discussion on the differences between discrete-time and continuous-time models, see our online review *Network-based models of transmission of infectious diseases: a brief overview* [15]. For readers who prefer a more slow-paced and much more detailed development of this material we recommend our book chapters [16, 18].

1.1.1 Compartment-Level Models

Many mathematical models of disease transmission are based on the assumptions of *homogeneity of hosts* and *uniform mixing.* In our terminology, the first of these assumptions boils down to assuming that the parameters a, α that govern the transition from the **I**-compartment into the **R**-compartment are identical for each host. We make this assumption in all our models. In contrast, the uniform mixing assumption boils down to assuming that the parameters $b_{i,j}$ or $\beta_{i,j}$ that give the probabilities or rates of making effective contact are identical for each pair (i,j) of hosts with $i \neq j$.

When both assumptions of homogeneity of hosts and uniform mixing are made, hosts lose all individual characteristics, and the state of the model at time t can be conceptualized as the vector of numbers of hosts in the compartments[2]. The distribution of future states of the model is then entirely determined by the current state. Models that operate entirely on the level of these counts or their expected values are usually called *compartment-based models,* but we prefer the phrase *compartment-level models.*

1.1.2 The Basic Reproductive Number

Typically we will consider outbreaks that start at time 0 with exactly one infectious host j^* (the *index case*). When all other hosts are susceptible in the initial state, we speak of *introduction of one index case into an otherwise susceptible population.*

[2]In our modules these numbers are denoted by $(|\mathbf{S}(t)|, |\mathbf{I}(t)|, |\mathbf{R}(t)|)$.

Consider such an initial state. The number of *secondary infections* caused by the index case j^* is a random variable (r.v.). Its mean value is the most important parameter in disease modeling. It has a special name, provided in the definition below.

Definition 1. The *basic reproductive ratio* or *basic reproductive number* R_0 is the mean number of secondary infections caused by an average index case in a large and entirely susceptible population.

Now consider an outbreak that starts with introduction of an index case j^* into an otherwise susceptible population. In an *SIR*-model, the *final size* F of the outbreak, that is, the proportion of hosts who experience infection, must be the proportion of hosts that reside in the **R**-compartment when the outbreak is over. Note that F is also a r.v. In compartment-level models, the expected value, and, more generally, the distribution of F, cannot depend on the particular host j^* that started the outbreak, since all hosts are assumed to have identical properties. Moreover, it is almost entirely determined by R_0 in the following sense: For very large population sizes N, compartment-level models predict that with probability very close to 1:

- When $R_0 \leq 1$, then $F \approx 0$.
- When $R_0 > 1$, there exist $0 < r(\infty), z_\infty < 1$ that depend only on R_0 such that:
  - $F \approx 0$ with probability $\approx z_\infty$.
  - $F \approx r(\infty)$ with probability $\approx 1 - z_\infty$.

In the former case we speak of a *minor outbreak* that affects only a negligible fraction of hosts and in the latter case of a *major outbreak* that affects a significant fraction of hosts. If you run a version of Exercise 22(a) in Section 5.4 without "vaccinating" any hosts and by choosing **infection-prob** := 0.015 instead of the recommended value 0.03 for that exercise, you will get a nice illustration of this distinction.

1.1.3 Models of Contact Networks: Graphs

In this chapter, instead of restricting ourselves to the assumptions of compartment-level models, we study network-based models, where the infection can only be spread if individuals are connected to one another in a given *contact network*. Contact networks can be modeled by mathematical structures called *graphs,* and we will treat the words "graph" and "network" as synonyms here, although in other sources of the literature "networks" comprise a broader class of structures (see, for example, our discussion in Section 9.3.1 in [16]). The terminology used in graph theory is not as consistent as in other areas of mathematics, so let us briefly review the concepts that we will need here.

Technically, a *graph* G is a pair $G = (V(G), E(G))$. The elements of $V(G)$ are called *nodes* or *vertices;* we will use these words interchangeably. We will always assume here that $V(G)$ is a set of N nonnegative integers. For mathematical explorations it is usually most convenient to take $V(G) = \{1, \ldots, N\}$, but NETLOGO will number the nodes from 0 to $N-1$. The set $E(G)$ contains unordered

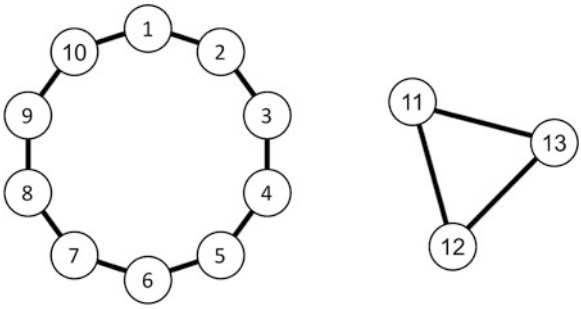


Fig. 2 The graph G_1.

pairs of nodes that represent the *edges* of G. Figure 2 shows an example of a graph that we call G_1.

Here $V(G_1) = \{1, \ldots, 12\}$, and, for example, $\{1, 2\} \in E(G_1)$ while $\{1, 3\} \notin E(G_1)$. We say that nodes 1 and 2 are *adjacent* (in G_1), while nodes 1 and 3 are not adjacent. The total number of nodes j that are adjacent to node i is called the *degree* of i and will be denoted by k_i. In G_1, each node has degree $k_i = 2$. Graphs in which all nodes have the same degree k are called *k-regular graphs.* Thus G_1 is a 2-regular graph.

The *mean degree* will be denoted by $\langle k \rangle$. For a network of size N it is equal to

$$\langle k \rangle = \frac{1}{N} \sum_{i=1}^{N} k_i = \frac{k_1 + k_2 + \cdots + k_N}{N} = \frac{2|E(G)|}{N}, \tag{1}$$

where $|A|$ denotes the size of a finite set A.

If we remove part of the vertex set of a graph G and retain only $V^- \subset V(G)$ together with all edges $\{i, j\} \subset V^-$ that are in $E(G)$, then we obtain a structure that is called *the subgraph of G induced by* V^-.

In Figure 2 two such structures stand out, and they are examples of important classes of graphs that will be extensively considered in later sections.

Consider the subgraph of G_1 that is induced by $V^- = \{11, 12, 13\}$. It looks like a triangle and illustrates why mathematicians speak of "vertices" and "edges" of a graph. More importantly, it contains every *possible* edge between its 3 vertices and will be denoted by K_3. More generally, the *complete graph* K_N is a graph with N vertices that contains all possible edges between them. Our use of the articles "the" and "a" in the preceding sentence may look inconsistent, because we could also have *another* complete graph $\hat{K}_3$ with three vertices, for example $V(\hat{K}_3) = \{1, 2, 3\}$. But since K_3 and $\hat{K}_3$ will differ only in the chosen numbering of the vertices, they will be what mathematicians call *isomorphic* and have the exact same properties. For our purposes here it will be convenient to leave it to our readers how they want to number the vertices and somewhat informally treat isomorphic graphs as identical.

The subgraph of G_1 that is induced by $V^- = \{1, \ldots, 10\}$ is not a complete graph. For example, $\{1, 3\}$ would be a possible edge, but it is not in $E(G_1)$. You can think of the adjacency relation in this induced subgraph as representing people

sitting next to each other around a table. This induced subgraph will be denoted here by $G_{NN}(10,1)$. It is an example of a *one-dimensional nearest-neighbor network* $G^1_{NN}(N,d)$. Such networks, and their two-dimensional counterparts $G^2_{NN}(N,d)$ will play a key role in this chapter. We urge readers to work through our detailed investigation of these networks in [21].

A *path* in a graph G is a sequence of nodes $P = (i_1, i_2, \ldots, i_m)$ such that each of the pairs $\{i_1, i_2\}, \{i_2, i_3\}, \ldots, \{i_{m-1}, i_m\}$ is an edge of G. The *length* of a path $P = (i_1, \ldots, i_m)$ is $\ell = m - 1$, that is, the number of edges that we traverse along the path. The length of the shortest path between two distinct nodes i and j in a graph G is called the *distance* between i and j in G and is denoted by $d(i,j)$. For example, in G_1 both $(11, 12, 13)$ and $(11, 13)$ are paths from node 11 to node 13, with length 2 and 1 respectively. The distance $d(11, 13) = 1$.

In G_1, there is no path from node 1 to node 12 whatsoever. We will consider the distance between nodes 1 and 12 as infinite. The *connected component of a node i in a graph G* is the subgraph whose vertex set comprises all nodes at a finite distance from i. In G_1, the connected component of node 1 is the subgraph induced by $V^- = \{1, \ldots, 10\}$, and the connected component of node 12 is the subgraph induced by $V^- = \{11, 12, 13\}$. A graph G is *connected* if it has exactly one connected component, that is, if the distance between any two nodes is finite. The *diameter* of a graph G, denoted by $diam(G)$, is the largest distance between any two of its nodes. The diameter of G_1 is infinite; only connected graphs have finite diameter. The diameter of the connected component of node 1 in G_1 is 5, while the diameter of the connected component of node 12 in G_1 is 1. If, for example, we were to add the edges $\{1, 6\}$ and $\{3, 8\}$, then the diameter of the connected component of node 1 would decrease to 4.

1.1.4 Network-Based Models

These models still use compartments—each node will belong to one of the **S**-, **I**-, or **R**-compartments at a single point in time—but unlike compartment-level models, they also allow for differences between individual hosts. More precisely, they are based on the assumption that only hosts that are adjacent in a given contact network can make effective contact.

The parameters of a network-based *SIR*-model are a graph G that represents the contact network and *disease-transmission parameters* a, b (for discrete-time models) or α, β (for continuous-time models). Here we retain the assumption of homogeneity of hosts and consider effective contacts along each edge of the contact network equally likely. Thus the transmission parameters $b_{i,j}$ or $\beta_{i,j}$ between pairs of hosts can be written as $b_{i,j} = b$ (or $\beta_{ij} = \beta$) if $\{i,j\} \in E(G)$ and $b_{i,j} = 0$ (or $\beta_{ij} = 0$) otherwise.

Note that the uniform mixing assumption will be satisfied if, and only if, the contact network is the complete graph K_N. Thus in a sense, network-based models include compartment-level models as special cases.

1.2 Further Reading

This introduction provides only a couple of highlights from the vast literature on the basics of epidemiology and in particular on compartment-level models. For students who want to learn more about these models, we recommend [26] as an easily accessible introduction and [2] as a more comprehensive one, at a more mathematically advanced level. More sources can be found in our online review [15].

Network-based models are covered to a limited extent in some of the resources that we listed above. A mathematically more advanced treatment can be found in [3]. We recommend our chapter [16] as a detailed elementary introduction into this type of model.

The spread of infectious diseases is inherently a stochastic process. A course on stochastic processes is not a prerequisite for success with our projects, but the content in this paper relies heavily on probability theory. Our online module *A brief review of basic probability theory* [6] covers the notions that are used here and in our other materials such as [7–21]. While it cannot replace a regular textbook on probability, it can serve as a short refresher course.

In what follows, we assume the reader has worked through our online module *A quick tour of IONTW* [21]. In this module we provide detailed instructions on how to use our software and guide the reader through some of the capabilities of IONTW. Highlights include the types of networks supported, setting up various types of models of disease transmission, observing the resulting dynamics, and collecting statistics on the outcomes. Along the way, the module also illustrates basic notions of graph theory. It contains many exercises, and sample solutions are included at its end.

2 Exploring Erdős-Rényi Random Graphs

In this section we introduce the notion of random graphs and explain how such networks can be used to derive meaningful predictions about disease transmission. We then define the most basic type of random graphs and explore the distribution of the sizes of their connected components. Along the way we introduce the important concept of asymptotic almost sure convergence.

2.1 Random Graphs

In Section 1.1.4 we have tacitly assumed complete knowledge of the contact network. But for transmission of most diseases in large real communities it is usually impossible to construct the actual relevant contact network. Typically we will have only some partial information about it; for example, estimates of some network parameters such as the mean degree $\langle k \rangle$, or some knowledge of the mechanisms by which contacts are established. How can we possibly draw meaningful inferences from network-based models in the face of such uncertainty?

Moreover, the networks we have looked at so far, complete graphs and nearest-neighbor networks, have a well-defined and very rigid structure. This will not be the case for contact networks in almost any real population of hosts, which will be much more messy. How can we possibly draw meaningful inferences from network-based models in the face of such messiness?

Well, how about using the word "randomness" instead of "uncertainty" and "messiness"? Then we see that the difficulties outlined in the preceding two paragraphs are actually two sides of the same coin (aka a *random number generator*): Usually some randomness is inherent in the processes by which contacts are established. This accounts both for the observed lack of structure and for our uncertainty. But if the processes by which contacts are established are random, then we can think of the actual (messy and unknown) contact network as being drawn from a certain probability distribution, about which we usually have some partial knowledge and which perhaps is similar to a simple distribution with nice mathematical properties.

Even when we have a reasonably good idea about the distribution, we won't know the actual contact network. But if we draw several graphs from the given distribution, we might reasonably expect that they form a *representative sample* of actual contact networks. Essentially this is the same trick that we use whenever we run simulations. During the run of multiple simulations, the computer produces in effect a representative sample of possible outbreaks for the given network, and we can use the outcome of simulations to form reasonably reliable hypotheses about what should happen in real outbreaks. The only novelty is that we are now considering two distinct sources of uncertainty: Uncertainty about the actual contact network, and uncertainty about the actual course of the outbreak.

Thus we may base a disease transmission model on a given distribution of *random graphs* and explore these models either by simulations on a representative sample of networks from this distribution, or by deriving theoretical predictions about the expected courses of outbreaks in these models.

2.2 Definition of Erdős-Rényi Random Graphs

There are various constructions of random graphs; several of them are implemented in IONTW. The most basic of these constructions gives *Erdős-Rényi random graphs,* named after the two Hungarian mathematicians who first systematically explored these graphs in the seminal paper [4]. These graphs serve as a benchmark against which all other constructions of random networks can be compared.

To construct such an Erdős-Rényi graph, we first decide on the number N of nodes. Then we list all edges $e_1, \ldots, e_{\frac{N(N-1)}{2}}$ of the complete graph K_N and repeatedly toss a biased coin that comes up heads with probability p. We include e_ℓ as an actual edge of the *random* graph if, and only if, the coin comes up heads in toss number ℓ.

The mean degree $\langle k \rangle$ of the resulting graph will be approximately

$$\langle k \rangle \approx \lambda = p(N-1). \tag{2}$$

Thus by choosing a suitable value of the *connection probability p*, one can assure that the mean degree $\langle k \rangle$ of an Erdős-Rényi random graph will be close to the values that one might have estimated from data on real networks.

It will be more convenient if we think of Erdős-Rényi random graphs in terms of the parameter λ instead of the parameter p. The connection probability p can then be expressed as $p = \frac{\lambda}{N-1}$. The symbol $G_{ER}(N, \lambda)$ will denote *an* Erdős-Rényi random graph that is constructed with parameters N and λ.

2.3 Properties That Hold Asymptotically Almost Surely

Note that we used the indefinite article *an* in the previous sentence. A graph $G_{ER}(N, \lambda)$ is not uniquely determined; in fact, it could be *any* graph G with N vertices. The symbol $G_{ER}(N, \lambda)$ only signifies that the graph is randomly drawn from a specific probability distribution. We will call a particular graph that has been constructed by the method described above *an instance of* $G_{ER}(N, \lambda)$.

Note also that (2) contains the symbol $\approx$. For a *given instance* of an Erdős-Rényi random graph we should not expect exact equality $\langle k \rangle = \lambda$. But $\langle k \rangle$ will be very close to λ.

But wait a minute. How could we possibly write that $\langle k \rangle$ *will* be very close to λ? If $G_{ER}(N, \lambda)$ could be *any* graph with N nodes, then it could also be the graph with no edges whatsoever (with $\langle k \rangle = 0$) or the complete graph K_N (with $\langle k \rangle = N - 1$). So we cannot be *sure* that $\langle k \rangle$ will be very close to λ, and should not have written "will be." What we should have written is something along the lines of "we can be pretty darn sure that $\langle k \rangle \approx \lambda$." This is somewhat cumbersome, and for this reason in the literature on random graphs "will be" is somewhat sloppily used as shorthand for "pretty darn sure," which of course isn't a phrase that belongs in a respectable research paper. In polite company mathematicians use the phrase "the mean degree $\langle k \rangle$ of $G_{ER}(N, \lambda)$ converges to λ *asymptotically almost surely* (abbreviated *a.a.s*)." This means that for any fixed error bound $\varepsilon > 0$, the probability that the mean degree $\langle k \rangle$ of a randomly drawn instance of $G_{ER}(N, \lambda)$ will differ from λ by more than ε will approach 0 as $N \to \infty$.

Note that we cannot write here "almost surely" (which would mean with probability 1), and cannot translate a.a.s. convergence into a property of instances of $G_{ER}(N, \lambda)$ for a fixed N. It is a property of the *class* of all Erdős-Rényi random graphs with parameter λ that contains instances of $G_{ER}(N, \lambda)$ of arbitrarily large size N. For convenience, we will use the symbol $G_{ER}(N, \lambda)$ both for the entire class of Erdős-Rényi random graphs with fixed parameter λ and variable N and for a given instance. In most cases the correct interpretation will be implied by the context; when there is ambiguity we will insert the phrase "an instance of" or "the class" to avoid confusion.

The phrase "asymptotically almost surely" also can be used for properties that are categorical rather than expressed in numerical values. For example, we will show in the next subsection that for any fixed λ a graph $G_{ER}(N, \lambda)$ will a.a.s. be disconnected. This means that as $N \to \infty$ the probability of drawing a connected

Table 1 Parameter settings for Erdős-Rényi exploration.

infection-prob	end-infection-prob	network-type	num-nodes	lambda	auto-set
1	1	Erdos-Renyi	300	1.5	Off

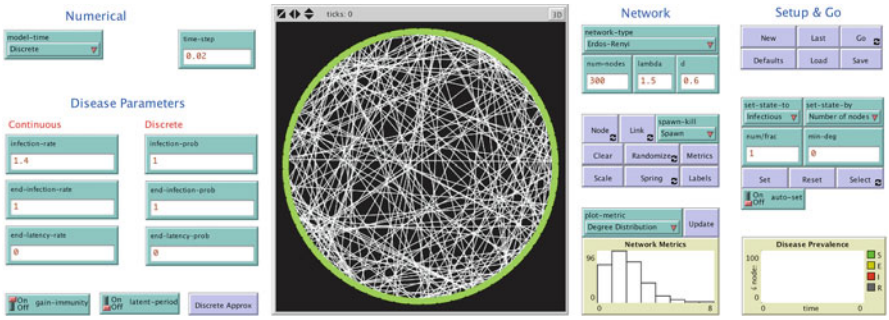


Fig. 3 View of IONTW interface after selecting an Erdős-Rényi contact network with 300 nodes and $\lambda = 1.5$.

instance of $G_{ER}(N, \lambda)$ approaches zero. This does not rule out that even for very large N we could accidentally draw a connected instance. It only implies that for large N these events become very, very unlikely.

When we study properties of random graphs, usually the best possible results we can hope for is that a given property will hold a.a.s.

2.4 Exploring the Connected Components of Erdős-Rényi Random Graphs

Open IONTW, press **Defaults**, set the speed control slider to the extreme right, and use the parameter settings in Table 1.

These parameter settings specify a next-generation *SIR*-model on an Erdős-Rényi network $G_{ER}(300, 1.5)$. Press **New** to look at the network. Your view in IONTW should look like a variation of that in Figure 3.

With this default view it is not possible to visually make out the connected components. (One can view the connected components by pressing the **Spring** button on the interface, but we will determine the same information via another route.) We can use the properties of the disease transmission model to visualize them: Since the probability b of an effective contact until the next time step, controlled by the input field **infection-prob**, is equal to 1, all nodes in the connected component of the index case j^* will eventually experience infection. The input setting **end-infection-prob** = 1 specifies a next-generation *SIR*-model in which all infectious nodes will get removed after exactly one time step and turn grey. If initially there is exactly one index case j^* in an otherwise susceptible population, all nodes outside of the connected component of j^* will remain green, and the connected component of j^* will show up in grey at the end of the simulation.

Table 2 Batch processing settings for testing major/minor outbreaks.

Repetitions	Measure runs using these reporters	Setup commands
100	`count turtles with [removed?]`	`new-network`

To see how this works, press **Set** to introduce one infectious node, and then **Go**. Repeat about 10 times for this network using **Reset** and then **Set**. This will keep the network fixed, but will change the initially infectious node. Record the approximate sizes of the connected component by moving your mouse over the relevant part of the grey curve in the **Disease Prevalence** plot.

Exercise 1. What do you observe? Do you get connected components with a range of different sizes? If the component is large, is it always the same one? How can you tell from the plot?

The results may look puzzling. Repeat 10 more times, but look at a different instance of $G_{ER}(300, 1.5)$ each time by pressing **New** instead of **Reset** before pressing **Set**.

Exercise 2. In what respect are the results similar to the ones of the previous exercise; in what respect are they different?

This is interesting. It appears that there is always one very large component in addition to many small ones.

Let us try to confirm the results we have discovered so far by running a large batch of simulations instead of looking at a few instances.

Switch **auto-set: On**

With the current parameter settings, define and run a batch processing experiment by using the template given in [13]. Define a **New** experiment according to Table 2.

Exercise 3. After the experiment is completed, open and analyze your output files. The column with the header `count turtles with [removed?]` reports the sizes of the connected component of the initially infectious node. Try to detect a distinctive gap between small and large components that were reported. Then record the maximum size of the observed small components and the mean size of the observed large components. Express these numbers as fractions of the total population size. Do your results confirm the preliminary observations that you made in the previous exercises?

Now let us quote a theorem that explains the outcomes that you probably observed in Exercises 1–3. Let Θ denote the proportion of the nodes in the largest connected component, that is, the ratio that is obtained by dividing the size of the largest connected component of a given graph $G_{ER}(N, \lambda)$ by N.

Theorem 1. *Consider the class of Erdős-Rényi random graphs $G_{ER}(N, \lambda)$, where $\lambda > 0$ is fixed and N is arbitrary. Then there exist constants $0 < \varrho = \varrho(\lambda) < 1$ and $c_{small} = c_{small}(\lambda) > 0$ that depend only on λ such that:*

- *When* $\lambda < 1$*, then*
  - $\Theta \to 0$ *a.a.s.*
  - *More precisely, a.a.s. all connected components of the graph* $E(G_{ER}(N, \lambda))$ *will have size* $\leq c_{small} \ln(N)$.
- *When* $\lambda > 1$*, then*
  - $\Theta \to \varrho$ *a.a.s.*
  - *A.a.s., all* other *connected components of the graph* $E(G_{ER}(N, \lambda))$ *will have size* $\leq c_{small} \ln(N)$.

Exercise 4. Before reading on, take a few minutes to think about the theorem. Does it predict the pattern that you observed in your explorations? How would you estimate the value of $\varrho(1.5)$ based on your findings?

Note that the second item of the theorem for $\lambda < 1$ implies the first since

$$\lim_{N \to \infty} \frac{c_{small} \ln(N)}{N} = 0. \tag{3}$$

We will call a connected component of $G_{ER}(N, \lambda)$ *small* if its size does not exceed $c_{small}(\lambda) \ln(N)$. For sufficiently large N, each small component will contain only a negligible fraction of nodes. (Note that by (3) we are allowed to write "will" here.) Since $\varrho(\lambda)$ is always less than 1, the theorem implies that a.a.s. $G_{ER}(N, \lambda)$ will contain many small components, and, in particular, will a.a.s. be disconnected.

In contrast, when $\lambda > 1$ the largest connected component will a.a.s. comprise a fraction of $\approx \varrho(\lambda)$ of all nodes. By the last item of the theorem, it will be a.a.s. the unique connected component with this property. In the literature, it is usually referred to as the *giant component.*

The value $\varrho = \varrho(\lambda)$ is the unique solution of the equation

$$1 - \varrho = e^{-\lambda \varrho} \tag{4}$$

in the interval $(0, 1)$. It can be shown that for $\lambda > 1$ there exists exactly one such solution, while for $\lambda \leq 1$, no solutions of (4) fall in this interval. The function $\varrho(\lambda)$ is strictly increasing, with $\lim_{\lambda \to 1^+} \varrho(\lambda) = 0$ and $\lim_{\lambda \to \infty} \varrho(\lambda) = 1$. For example,

$$\varrho(1.1) = 0.1761, \quad \varrho(1.5) = 0.5828, \quad \varrho(2.0) = 0.7968, \quad \varrho(3.0) = 0.9405. \tag{5}$$

For more information on the connected components of $G_{ER}(N, \lambda)$ as well as the history and extensions of Theorem 1 we recommend the survey article [24].

The alert reader will have noticed the analogy between small *vs.* large components of $G_{ER}(N, \lambda)$ and minor *vs.* major outbreaks of diseases. The itemized list in Section 1.1.1 does look somewhat similar to the items of Theorem 1, and there is in fact a close connection. With some careful wording of the assumptions, the relation between R_0 and minor and major outbreaks can even be phrased in terms of a.a.s. convergence, although this is rarely done in the literature. Having a good understanding of the precise nature of the connection between giant components in

certain random graphs and major outbreaks may be very useful for the theoretical parts of some of our projects. While it would take us too far afield to include the details here, we do recommend that students who did choose to work on one of our projects consult in due course Section 3 of Module [10].

3 Clustering Coefficients

In this section we introduce so-called *clustering coefficients*. A motivating example shows how these characteristics of the contact network may influence the spread of an infectious disease. In later subsections we explore, both with the help of IONTW and theoretically, the behavior of clustering coefficients for various network types.

3.1 A Motivating Example

Random regular graphs $G_{Reg}(N,k)$ are graphs with N nodes where every node has degree k, but where the edges are randomly assigned (see modules [19, 21] for further details). By Exercise 4 of our module *A quick tour of IONTW* [21], the one-dimensional nearest-neighbor networks $G^1_{NN}(N,d)$ are k-regular for $k = 2d$ when $N \geq 2d+1$. Would it be reasonable to expect that diseases would spread on these two types of networks in similar ways? Let us see whether simulations shed any light on this question.

Open IONTW, click **Defaults**, and change the parameter settings in Table 3.

Press **New** to initialize a next-generation *SIR*-model on a network $G_{Reg}(200,4)$ with one index case in an otherwise susceptible population. Similar models were investigated in our modules *Exploring random regular graphs with IONTW* [19]. We would expect to see a significant proportion of major outbreaks in addition to some minor ones. You may want to run a few exploratory simulations to check whether this is what you will see in the **World** window and the **Disease Prevalence** plot.

Now let us confirm the preliminary observations with a lager number of simulations. Using the template that is provided in our online instructions [13], set up a batch processing experiment for the current parameter settings, by defining a **New** experiment with the specifications outlined in Table 4.

Table 3 Parameter settings for the motivating example.

infection-prob	end-infection-prob	network-type	num-nodes	lambda	auto-set
0.5	1	Random Regular	200	4	On

Table 4 Settings for batch processing experiment on $G_{Reg}(200,4)$.

Repetitions	Measure runs using these reporters	Setup commands
100	`count turtles with [removed?]`	`new-network`

Exercise 5.

(a) Run the experiment and analyze your data by sorting the output column (`count turtles with [removed?]`) from lowest to highest. If you see a distinct gap between minor and major outbreaks, report the number of minor outbreaks, as well as the mean and maximum values of the output variable for these outbreaks. Also report the minimum, mean, and maximum for the major outbreaks, as well as the overall mean.

(b) Are the results consistent with your expectations?

Now set **network-type** $\rightarrow$ **Nearest-neighbor 1** and **d** := 2. Press **New** to initialize a next-generation *SIR*-model on a network $G^1_{NN}(200, 2)$ with one index case in an otherwise susceptible population. Remember that this will give degree $k = 2d = 4$ to each node. Press `Clear` on the bar of the **Command Center** (not the **Clear** button within the NETLOGO interface), then press **Metrics** to verify that `<k> = 4`.

You may want to run a few exploratory simulations to check whether you see similar results in the **World** window and the **Disease Prevalence** plot as for the previous network type.

Now let us confirm the preliminary observations by setting up and running a **New** batch processing experiment with 100 runs for the current parameter settings.

Exercise 6.

(a) Run the experiment and analyze your data as in Exercise 5.

(b) Are the results similar to those in the previous experiment? If not, does the structure of $G^1_{NN}(200, 2)$ appear to increase or decrease the severity of outbreaks relative to the corresponding random regular graph?

To illustrate what is going on here, let us consider a state at time $t = 1$ with exactly 4 infectious nodes j_1, j_2, j_3, j_4. Each of these nodes will have one neighbor (the index case) who infected it and is no longer susceptible at time $t = 1$. Let $\mathcal{N}_1(j)$ denote the set of nodes i that are adjacent to j. Each one of these nodes must be adjacent to the index case j^*. In a large random 4-regular graph, with high probability it will be the case that the union of the neighborhoods $\mathcal{N}_1(j_1), \mathcal{N}_1(j_2), \mathcal{N}_1(j_3), \mathcal{N}_1(j_4)$ contain a total of 12 susceptible nodes to whom the pathogen could be transmitted by time step 2.

Now let us see how the situation differs in graphs $G^1_{NN}(N, 2)$. Set **infection-prob** := 1. The pattern you will observe here is the same for all sufficiently large values of N, but for better visualization set **num-nodes** := 20.

Create a **New** network with one index case in an otherwise susceptible population. In the **World** window you will see that $\mathcal{N}_1(j^*)$ contains 4 nodes.

Move the speed slider to a very slow setting; adjust for comfortable viewing as needed. Start a simulation with **Go** and stop it by pressing **Go** again when you see a state with exactly 1 removed and 4 infectious nodes in the **World** window. Count the number of green nodes at the end of red edges that could become infectious at the next time step. It will be less than 12. This effect results from the special structure of the network and explains the discrepancies that you observed.

In your **World** window you will see some white edges with two red endpoints. Look at one of these edges. No effective contact between its endpoints by time $t = 2$ can lead to a new infection, and in some sense this edge decreases the number of potential nodes that can become infectious at the next time step by 2 (one for each endpoint). *Clustering coefficients* allow us to quantify this effect. They indicate whether we should expect many or relatively few such edges. They explain some of the decrease in the number of candidates for infection at the next step from 12 to the one you just found.

Each of the white edges with red endpoints that you see is an edge of a triangle whose third endpoint is the grey node that represents the index case. Clustering coefficients can be defined by counting the number of potential triangles; high clustering coefficients indicate that there are a lot of them; low clustering coefficients indicate few.

Count the number of white edges that connect two red nodes. There should be 3 of them; each one is part of a triangle whose third vertex is the index case and whose other two edges are grey.

Are 3 white edges *a lot?* To make sense of the phrases *a lot* or *few* we need to compare the observed numbers with some benchmark. In the case of clustering coefficients, the benchmark is the complete graph.

Choose **network-type** $\rightarrow$ **Complete Graph** and **num-nodes** := 5.

Create a **New** network. Run a simulation in slow motion for exactly one time step and count the number of white edges that connect red nodes. This number gives us the benchmark; it is the number of edges in a complete graph K_n, where n is the size of $\mathcal{N}_1(j^*)$ in the previous experiment. In our case $n = 4$ and the number of edges in the complete graph is 6.

If we divide the number of white edges that we observed in the previous experiment by 6, we obtain the *node clustering coefficient* of the index case. The formal definition will be given in the next subsection.

3.2 Definitions of Clustering Coefficients

Several subtly different notions of *clustering coefficient* also known as *transitivity* have been studied in the literature. One always needs to carefully read the definition to see what, exactly, these terms mean in the given source. We will work with four such notions.

Consider a node i in a graph G. Recall that $\mathcal{N}_1(i)$ denotes the set of i's neighbors, that is, nodes that are adjacent to i. Let $tr(i)$ denote the number of edges $\{j_1, j_2\} \in E(G)$ such that $j_1, j_2 \in \mathcal{N}_1$. The number $tr(i)$ is exactly the number of triangles that node i forms with two of its neighbors.

Watts and Strogatz [27] define the *node clustering coefficient*[3] $C(i)$ *of* i by dividing $tr(i)$ by its maximum possible value $k_i(k_i - 1)/2$, where k_i is the degree of node i.

[3] Some authors refer to node clustering coefficients as *local* clustering coefficients.

$$C(i) = \frac{2tr(i)}{k_i(k_i - 1)}. \tag{6}$$

If $k_i < 2$, then Equation (6) does not make sense, and we define $C(i)$ as the *edge density,* that is, the probability $\frac{2|E(G)|}{N(N-1)}$ that two randomly chosen nodes are adjacent.

When G represents friendships among people, the clustering coefficient $C(i)$ measures the ratio of the number of friendships between any two of i's friends relative to a situation where all these friends would induce a complete subgraph of G. Mathematicians actually refer to such sets that induce complete subgraphs as *cliques.*

The *network clustering coefficient* C is defined as the mean of the node clustering coefficients $C(i)$:

$$C = \frac{1}{N}\sum_{i=1}^{N} C(i). \tag{7}$$

In Section 3.1 we have already seen one interpretation of clustering coefficients. Here is an alternative interpretation that is often given in the literature. Consider a network G. Suppose we randomly pick a node i, and then we randomly pick two nodes j_1, j_2 that are adjacent to i. Does this procedure make it *more likely* or *less likely* that the pair $\{j_1, j_2\}$ forms an edge in a given graph, relative to a completely random choice of j_1, j_2? Intuitively, one would expect the answer "more likely" for networks of social contacts. Two randomly chosen friends of yours are more likely to be friends of each other than two randomly chosen persons. You and your friends will form a *cluster* in the friendship graph.

To see the connection with clustering coefficients, let us first observe that our procedure requires that $j_1, j_2 \in \mathcal{N}_1(i)$. If $\{j_1, j_2\}$ is an edge, then the subgraph of G that is induced by the set of nodes $\{i, j_1, j_2\}$ will form a triangle. If G contains relatively many triangles, as will be the case in large nearest neighbor graphs with $d > 1$, we might expect that the answer will be "more likely." On the other hand, we should expect the answer to be "less likely" if G contains only relatively few triangles. If G does contain some edges but no triangles at all, as in graphs $G^1_{NN}(N, 1)$ for $N > 3$, then $\{j_1, j_2\}$ simply cannot be an edge and the answer will definitely be "less likely."

But what, exactly, do the phrases "relatively few" and "relatively many" triangles (or cliques of size 3) mean? The network clustering coefficient C does not all by itself tell us whether two randomly chosen nodes are *more* likely, on average, to be adjacent in G if they share a common neighbor. A concrete example of this phenomenon is explored in detail in Section 3 of Module [8]. To remedy this drawback, let us introduce *normalized* clustering coefficients. These are obtained by dividing by the edge density:

$$C_{norm}(i) = C(i)\frac{N(N-1)}{2|E(G)|} \quad \text{and}$$
$$C_{norm} = C\frac{N(N-1)}{2|E(G)|} = C\frac{N-1}{\langle k\rangle} = \frac{1}{N}\sum_{i=1}^{N} C_{norm}(i). \tag{8}$$

In the second line of (8) we used the fact that $2|E(G)| = \langle k\rangle N$.

While $C(i), C$ are numbers between 0 and 1, the normalized clustering coefficients can take any nonnegative rational numbers as values. A value $C_{norm}(i) > 1$ indicates that the nodes in $\mathcal{N}_1(i)$ are *more* likely than average to be adjacent; a value $C_{norm}(i) < 1$ indicates that for average i the nodes in $\mathcal{N}_1(i)$ are *less* likely to be adjacent than randomly chosen nodes. If $C_{norm} > 1$ we will say the graph *exhibits clustering;* if $C_{norm} < 1$ we will say that the graph *avoids clustering.*

Exercise 7. Find the clustering coefficients $C(i), C, C_{norm}(i), C_{norm}$ for (each node i of) each of the following graphs and determine whether the graph exhibits or avoids clustering.
(a) For the graph $G^1_{NN}(9, 2)$.
(b) For the graph $G^2_{NN}(15, 1)$.
(c) For the graph G_1 of Figure 2.

One can think of C_{norm} for a given G as comparing C of G with a benchmark. As the next exercise shows, this benchmark would be an Erdős-Rényi random graph with the same number of nodes and mean degree as G.

Exercise 8. Give an intuitive argument that for large N the normalized network clustering coefficient C_{norm} in $G_{ER}(N, \lambda)$ should be very close to 1.

The values of C_{norm} can be very large; see [8] for some examples of empirically studied networks where they are on the order of several hundreds. In such cases one would like to say that the network exhibits *strong clustering.* But there is no natural fixed threshold value above which clustering would qualify as "strong." A mathematically meaningful definition of this notion will require us to consider a class of graphs that contains representatives of arbitrarily large size N. We can then say that this class of graphs *exhibits strong clustering* if $C_{norm} \to \infty$ a.a.s. (asymptotically almost surely), which means here that for every probability $q < 1$ and fixed C_{target} there exists $N(q, C_{target})$ such that with probability $> q$ a randomly drawn network of size $N > N(q, C_{target})$ in this class will satisfy the inequality $C_{norm} > C_{target}$.

Thus while it makes sense to say that a given network exhibits or avoids clustering, the phrase "strong clustering" does not make sense for an individual network; it applies only to classes of networks. By Exercise 8, for any given λ the class of Erdős-Rényi networks $G_{ER}(N, \lambda)$ does not exhibit strong clustering. In the next subsection you will see examples of classes that do.

Table 5 Parameter settings for investigating clustering coefficients.

network-type	lambda	num-nodes
Erdos-Renyi	8	20, 40, 80, 160, 320

3.3 Exploring Clustering Coefficients of Selected Networks

Open IONTW and press **Defaults**. Work with the parameter settings in Table 5.

For each of the specified network sizes create one network with **New** and then press **Metrics** before creating the next network. When you are done, use the double arrow on the bar **Command Center** to enlarge this window and look at the statistics that you collected.

Exercise 9.
(a) Consider the values of `Edge density`, `Clustering coefficient`, and `Normalized clustering coefficient`. Which limits do these values appear to approach as N increases?
(b) Does the class of networks $G_{ER}(N, 8)$ appear to exhibit strong clustering?
(c) Are these results consistent with what you learned in Section 3.2?

Press `Clear` to clean up the **Command Center** and minimize this window. Change **network-type** $\rightarrow$ **Nearest-neighbor 1** and **d** := 2.

Repeat the steps of the data collection that you did for Erdős-Rényi networks of the sizes specified above. As you proceed, you may want to visualize the distribution of the values of $C(i)$ by choosing **plot-metric** $\rightarrow$ **Normalized Coeffs** and pressing **Update**.

Exercise 10. Answer the analogous questions as in Exercise 9(a),(b).

Save your statistics, and then change **network-type** $\rightarrow$ **Nearest-neighbor 2**. Repeat the exercise for the following values of **num-nodes:** 25, 36, 100, 225.

Repeat the steps that you did for the previous types of networks to collect data on networks $G^2_{NN}(N^2, 2)$ of the specified sizes N^2. Inspect the data.

Exercise 11. Answer the analogous questions as in Exercise 9(a),(b).

Now set **d** :=1. Press **New** and then **Metrics**. The command center will show you that both clustering coefficients C and C_{norm} are 0. This should be expected from the definitions, as the graph in the **World** window contains no triangles whatsoever.

3.4 A General Theorem About Strong Clustering

The following result will be very useful for our work in Section 5.

Theorem 2. *Suppose we are given a class of graphs $G(N)$ that contains representatives of arbitrarily large sizes N. Moreover, assume that the mean degree $\langle k \rangle$ approaches a.a.s. a finite limit as N increases without bound, and that $tr(i) \geq 1$ for each node i in all graphs $G(N)$. Then this class exhibits strong clustering.*

First note that for all fixed $d \geq 1$ and sufficiently large N the graphs $G^1_{NN}(N,d)$ are $2d$-regular and thus satisfy the first assumption of Theorem 2. The graphs $G^2_{NN}(N^2,d)$ are not regular, but one can show that for any fixed value of d they still satisfy this first assumption. Thus in all these classes there exists some finite upper bound k_{max} on the degrees so that $k_i \leq k_{max}$ for all nodes i in all graphs $G(N)$ of the class.

Exercise 12.
(a) Find an upper bound on the degree of any node in $G^2_{NN}(N^2,d)$ that does not depend on N.
(b) Show that the graphs $G^1_{NN}(N,d)$ and $G^2_{NN}(N^2,d)$ with $N > 2$ and $d > 1$ have the property that $tr(i) \geq 1$ for each node i.

Thus Theorem 2 implies that all classes $G^1_{NN}(N,d)$ and $G^2_{NN}(N^2,d)$ with fixed $d > 1$ exhibit strong clustering. You may want to compare this result with your findings in Exercises 10 and 11.

Challenge Problem 1. Prove Theorem 2 under the additional assumption that there exists some finite upper bound k_{max} on the degrees so that $k_i \leq k_{max}$ for all nodes i in all graphs $G(N)$.

4 Exploring Distances with IONTW

This section has two parts. The first part is purely conceptual and invites readers to critically evaluate popular claims based on Stanley Milgram's famous experiment that gave birth to the phrases "small-world property" and "six degrees of separation." In the second part we develop a formal definition of the small-world property. IONTW-based exercises are used in that part to illustrate conceptual subtleties of this notion.

4.1 Milgram's Famous "Six Degrees of Separation" Experiment

Consider the network G_{FN} whose nodes represent humans and whose edges connect any two persons who are acquainted on a first-name basis. Let $d(i,j)$ be the distance between two nodes in this network, that is, the number of edges in the shortest path from i to j in G_{FN}. If i,j are chosen randomly, then $d(i,j)$ becomes a r.v. What can we say about its distribution?

The American social psychologist Stanley Milgram and his collaborators conducted an ingenious experiment to answer this question; the results were reported in [25]. The experimenters recruited 296 volunteers from Nebraska and the Boston area. Each volunteer i was given a letter with some information about a Boston stock broker j and was instructed to send it to a person with whom the volunteer was acquainted on a first-name basis (along an edge of G_{FN}) and whom the volunteer thought to be at a closer distance from j in G_{FN}. Attached to the letter were instructions to continue forwarding it in this manner, until it would be sent to the stock broker j. The experimenters kept track of the number of intermediaries $i_1, \ldots, i_m$ (excluding i and j) that forwarded the letter. Out of the 296 letters, 64 eventually reached their target.

Exercise 13.
(a) Suppose the letter did eventually get sent to j. How is the number m of intermediaries related to the distance $d(i,j)$?
(b) Does the success rate of $\frac{64}{296}$ (approximately 22%) tell us anything about the structure of the network G_{FN}?

For the letters that did arrive, the researchers reported a mean number of 5.2 intermediaries. This result has inspired the popular claim that there are only *six degrees of separation* between any two humans.

Exercise 14. Critically evaluate this claim. What do or don't the results of Milgram tell us about the likely maximum, mean, or median of $d(i,j)$ in G_{FN}?

We strongly recommend that you don't try to look up our solution yet. Instead, write down your own thoughts, and reevaluate your solution after you have worked through the next subsection.

4.2 The IONTW Guide to the Small World (Property)

The claim that you evaluated in Exercise 14 gave birth to the phrase *small-world property*. The literature contains various subtly distinct and not always entirely rigorous definitions of this concept. Here we will construct one that may work best, while using IONTW to build up some intuition about the subtleties involved.

Why would a mean or even maximum distance of 6 (or even 2 or 3) between two randomly chosen people be considered surprisingly small? There would not be anything remarkably small about this number in a village of 100 people or even in the town of 20,000. But on the scale of the whole human population, the number 6 seems surprisingly small. We can see that a rigorous definition of the small-world property will make sense only if we phrase it so that it applies to networks of arbitrarily large sizes N. In other words, it should be a property of a whole class of networks, not of an individual representative of this class. Let's give it a first try:

Table 6 Parameter settings for investigating the small-world property.

network-type	num-nodes	d
Nearest-neighbor 2	100	1

Preliminary Vague Definition. A class of networks has the *small-world property* if the maximum or the average distance between nodes for networks in this class scales at most logarithmically with network size N.

This "Definition" has some problems, doesn't it? First of all, it is vague about whether we should use the maximum or the average distance. And "the" average distance isn't well-defined either, since there are several kinds of averages, most prominently means and medians. You may or may not be already familiar with the precise meaning of the phrase "scales at most logarithmically." And if the class of networks that we are interested in is a class of random graphs, shouldn't there be an "a.a.s." somewhere in the definition?

We will address each of these issues and arrive at a definition of the small-world property that avoids all the road hazards that we will encounter along the way.

Open IONTW, click **Defaults**, and choose the parameter settings in Table 6.

Create a two-dimensional nearest neighbor network $G^2_{NN}(100, 1)$ by pressing **New** and then toggle **Labels** to see how the nodes are numbered.

Exercise 15.

(a) Find $d(13, 21)$ and $d(21, 66)$ in this network.

(b) Find the diameter of this network, that is, find the maximum distance of any pair of nodes. For which pairs of nodes is the maximum attained?

It would be a bit tedious to calculate the mean or median distances for randomly chosen nodes in $G^2_{NN}(100, 1)$. Can you look it up in the **Command Center** after pressing **Metrics**? Well, maybe. The display gives you a line

```
Average path length in largest component = 6.66667
```

The word "Average" is a bit ambiguous; it could refer to a mean, median, or mode. "Path length" is also a bit vague. Since we will use this metric later in some of our explorations, let us divulge that this average reported by IONTW is actually a mean. But we highly recommend that you do Exercise 5 of Module [9] to find out more about the exact meaning of `Average path length`. Exercise 6 of this module provides examples, with illustrations of some beautiful graphs, of how the mean and median distances may differ.

The mean distance and the diameter are properties of a given network. Let us see what they tell us about the spread of diseases. Click **Defaults** and choose the settings in Table 7. This sets up a next-generation *SIR*-model where at each time step every host is guaranteed to make effective contact with all adjacent hosts.

Create a **New** network $G^2_{NN}(100, 1)$, and toggle **Labels** to see the numbering of the nodes. Make node 0 the index case by using the following procedure: Click **Set**. In the dialogue box that appears enter: `[0]`. Click **OK**. Note that you must type the square brackets around `0` for this to work properly.

Table 7 Parameter settings for testing mean distance and diameter.

time-step	**infection-prob**	**end-infection-prob**	**network-type**
1	1	1	Nearest-neighbor 2
num-nodes	**d**	**set-state-by**	
100	1	Vector from input	

Make sure that everything worked as expected so that you have an initial state where node 0 is infectious and all other nodes are susceptible. Set the speed slider to a slow speed; adjust for comfortable viewing as needed. Now click **Go** and watch the movie. If you want to restart it, use **Last.**

Exercise 16.

(a) What is the relationship between node 0 and the nodes that are infectious at time step t?

(b) How is the time at the end of the outbreak, measured in `ticks`, related to the network properties that we discussed earlier?

Now repeat the experiment twice, by using **Reset**: First set node 50 to be the single index case; then repeat the experiment again with node 55 as the single index case. To accomplish this, in the dialogue box that appears after clicking **Set** you will need to first enter `[50]`; after running the experiment for this index case, repeat by entering `[55]` after clicking **Set**.

While previously the duration of the outbreak was $diam(G^2_{NN}(100, 1)) + 1$, now you get shorter outbreaks. But even after subtracting 1, you will get a number that exceeds the mean distance. Let us use the symbol $D(j^*)$ for the number that we get after subtracting 1 from the duration of the outbreak caused by index case j^* in an otherwise susceptible population in a model with the disease transmission parameters listed above. This number can be defined for every network, and it is always somewhere between the mean and the maximum distance of pairs of nodes (see Exercise 8 of Module [9]).

We have observed something very important: Two *network parameters,* the diameter and the mean distance, give us upper and lower bounds on the mean duration of outbreaks of certain types of diseases. Thus one could use these network parameters to make predictions about the duration of outbreaks[4].

Unfortunately, in simulations we are restricted to exploring rather small networks, while real outbreaks happen in populations of thousands or millions of hosts. It is therefore of interest to investigate how the diameter and mean distance scale if we retain the general structure of the network but increase the number of nodes.

[4]The relation between these network parameters and the duration of outbreaks becomes less tight for disease transmission parameters that are different from the ones we considered here, but there will still be a close connection.

Table 8 Settings to test how diameter and mean distance scale with N.

num-nodes	d	network-type
11	1	Nearest-neighbor 2

Let us start with very simple networks. Press `Clear` on the **Command Center** bar and change the settings according to Table 8.

Click **New** and find the diameter of the network by visual inspection. Record it, and then click **Metrics** to verify your results. Repeat with **num-nodes** := 23, 47.

Note that in the case of prime numbers N the graphs $G^2_{NN}(N, 1)$ are not really two-dimensional grids but simple paths.

Enlarge the **Command Center** with the double-arrow icon and look at the data that you have recorded. Would it be fair to say that for graphs $G^2_{NN}(N, 1)$ with N prime the diameter and the mean distance roughly double when you roughly double the number of nodes?

Such a pattern is indicative of *linear scaling*. We say that the value of a quantity $\chi(N)$ that depends on N *scales linearly* if

$$\lim_{N\to\infty} \frac{\chi(N)}{N} = c, \tag{9}$$

where c is a nonzero constant. This does not mean that $\chi(N) = cN$, it only means that for sufficiently large N we can take cN as a fairly good estimate of $\chi(N)$. Since $c(2N) = 2cN$, linear scaling produces the pattern that you observed.

Nonlinear scaling can take many different forms. For example, if D is a fixed exponent, then we say that $\chi(N)$ *scales like* N^D if

$$\lim_{N\to\infty} \frac{\chi(N)}{N^D} = c \neq 0. \tag{10}$$

Of course, (9) is nothing else than the special case of (10) for $D = 1$. In other words, linear scaling is a special kind of *power law scaling*.

Now suppose $\chi(N)$ scales like $N^{0.5} = \sqrt{N}$. Then quadrupling N should have the effect of roughly doubling $\chi(N)$ as $c\sqrt{4N} = \sqrt{4}c\sqrt{N}$.

Let us see whether the class of square grids without diagonals is such an example. Repeat the previous experiment for **num-nodes** := 25, 100, 400.

Does it appear that the diameter and mean distance scale like $\sqrt{N}$ in the class of networks $G^2_{NN}(N, 1)$ with $N = n^2$?

So far, we have considered only networks with a rigid structure. But now let us explore some random graphs. Press `Clear` on the **Command Center** bar. Change the parameter settings according to Table 9.

Click **New** to create an instance of $G_{Reg}(50, 4)$. Use **Metrics** to find the mean distance between nodes.

Since every instance of a random graph is slightly different, we may want to look at several instances drawn from the same distribution. Create 5 instances each of

Table 9 Settings to test scaling in random graphs.

num-nodes	lambda	network-type
50	4	Random Regular

$G_{Reg}(50, 4)$, $G_{Reg}(100, 4)$, $G_{Reg}(200, 4)$ by changing **num-nodes** accordingly and using **New**. For each new instance, click **Metrics**.

Enlarge the **Command Center** with the double-arrow icon and look at the data that you have recorded. Would it be fair to say that the mean distance between nodes roughly increases by a constant number when you double the number of nodes?

This pattern is indicative of *logarithmic scaling* rather than power law scaling. We say that $\chi(N)$ *scales logarithmically* if

$$\lim_{N\to\infty} \frac{\chi(N)}{\ln(N)} = c \neq 0. \tag{11}$$

Since $c\ln(2N) = c(\ln(2) + \ln(N)) = c\ln(N) + c\ln(2)$, if $\chi(N)$ scales logarithmically, then for sufficiently large N we would see a roughly constant increment of $\chi(N)$ by $\approx c\ln(2)$ when we double N.

We need to be very careful though when we want to assert that a numerical characteristic $\chi(N)$ (the mean distance or diameter in our example) of a class of *random* graphs ($G_{Reg}(N, 4)$ in our example) scales (at most) logarithmically. We would like this to mean that for some fixed constant $c > 0$ and all N we have $\chi(N) \leq c\ln N$. But since, for example, $G^1_{NN}(N, 2)$ is also a 4-regular graph of size N, it *could* be drawn as an instance of $G_{Reg}(N, 4)$, although the probability of this event is very, very small. In view of our explorations above, we will not be able to say that for $G_{Reg}(N, 4)$ the average distance or diameter is *always* $\leq c\ln(N)$. But for a suitable choice of c, this inequality will hold asymptotically almost surely, that is, with probability arbitrarily close to 1 as $N \to \infty$. Yes, "a.a.s." should appear in a proper definition of the small-world property.

This clarifies some of the issues we had with our "preliminary vague definition," but doesn't yet resolve them.

We are aiming here at one universal definition that works for all classes of graphs. In particular, it should work both for classes that contain connected graphs and for classes that contain disconnected graphs. Consider, for example, the class of Erdős-Rényi random graphs $G_{ER}(N, \lambda)$ for a fixed $\lambda > 1$. According to Theorem 1 of Section 2.4, the graphs in this class are a.a.s. disconnected, with one giant connected component of size close to $\varrho(\lambda)N$. One can prove that the diameter of the largest connected component a.a.s. scales logarithmically for graphs $G_{ER}(N, \lambda)$ [1].

Exercise 17. How could you use IONTW to obtain a rough empirical confirmation of this theoretical result about the diameter of the largest component by considering instances of $G_{ER}(N, 4)$ for $N = 50, 100, 200$?

These results show that if we randomly pick two nodes in a graph $G_{ER}(N, \lambda)$, then with probability $\approx (\varrho(\lambda))^2$ they will both belong to the giant component and their

distance will be very small. As $(\varrho(\lambda))^2$ does not depend on the network size N, is positive for $\lambda > 1$, and even very close to 1 when λ is sufficiently large (see Section 2.4), we would still consider the class of graphs $G_{ER}(N, \lambda)$ for any fixed $\lambda > 1$ as an example of a class with the small-world property.

But graphs in this class are a.a.s. disconnected, and with positive probability two randomly chosen nodes will belong to different components and thus have an infinite distance. In particular, both the diameter and the mean distance between nodes in $G_{ER}(N, \lambda)$ are a.a.s. infinite. This makes these characteristics unsuitable for our definition of the small-world property. Similarly, the median will work only if λ is sufficiently large so that $(\varrho(\lambda))^2 \geq 0.5$, which would be an artificial cutoff. These considerations lead to the following definition that we will use in the remainder of this chapter.

Definition 2. A class of graphs has the *small-world property* if, and only if, for some fixed $P > 0$ there exists a positive constant c such that the P-th percentile of the distance between randomly chosen nodes a.a.s. satisfies the inequality $\leq c \ln(N)$.

By taking $P < 100(\varrho(\lambda))^2$, from the above discussion we see that the class of Erdős-Rényi graphs $G_{ER}(N, \lambda)$ has this property for each choice of fixed $\lambda > 1$.

Exercise 18. Formally prove that the following classes do not have the small-world property:
(a) The class of Erdős-Rényi random graphs $G_{ER}(N, \lambda)$ with any fixed $\lambda < 1$.
(b) The classes of graphs $G^1_{NN}(N, d)$ and $G^2_{NN}(N^2, d)$ for any choice of fixed d.

5 Small-World Networks

Small-world networks are classes of networks that have both the small-world property and exhibit strong clustering. Here we describe two constructions of such networks that are implemented in IONTW and study, both theoretically and with simulation experiments, the structure of these networks. We also illustrate how this structure influences effectiveness of certain vaccination strategies.

5.1 The Small-World Property and Small-World Networks

In the preceding two sections we studied the notions of strong clustering and of the small-world property separately. Classes of networks that have both the small-world property and exhibit strong clustering will be called *small-world networks.*

While many empirically studied contact networks appear to be small-world networks [27], none of the classes of graphs that we have explored so far qualify. For $\lambda > 1$ the class of Erdős-Rényi random graphs $G_{ER}(N, \lambda)$ has the small-world property but doesn't exhibit strong clustering. On the other hand, as you saw

in Sections 3.4 and 4.2, the classes of nearest neighbor networks $G^1_{NN}(N,d)$ and $G^2_{NN}(N,d)$ with $d > 1$ exhibit strong clustering, but do not have the small-world property.

In the next subsection we will construct specific examples of small-world networks and call them *small-world models.* Readers should be aware that there is no consensus about the usage of the terms "small-world property," "small-world network," and "small-world model." Different sources in the literature use different mappings between these phrases and the three concepts. One always needs to carefully read the definition in each source to figure out the intended meaning.

5.2 Small-World Models

In [16] we considered the monastic order of the Sisters of the Round Table. The sisters spend most of their lives in their individual cells, where they devote themselves to prayer and meditation. The only time they have contact with each other is during meals when they are seated in a fixed order around a giant round table. Within this community, diseases can be transmitted only during mealtime.

The probability of transmission will be largest between sisters who sit next to each other, and then decrease with the distance at the table. It may depend on the particular nature of the disease how far the infectious agents can travel, and for building a contact network we need to decide on a cutoff. Let us assume that there is a significant probability of transmission from sister i to sister j if at most $d-1$ sisters sit in between. Then the relevant contact network for a community of N sisters will be the one-dimensional nearest-neighbor graph $G^1_{NN}(N,d)$.

There is a problem though with our story: Even in a strictly monastic setting, contact networks will rarely have such a rigid structure as $G^1_{NN}(N,d)$. The Sisters will not necessarily head straight to the table from their cells. More likely, along the way they will exchange a few kind words with their next-cell neighbors who may be seated across the table. We can incorporate these more informal contacts into our network model by adding a few randomly chosen edges to a graph $G^1_{NN}(N,d)$. This will result in a class of networks that simultaneously exhibits strong clustering and has the small-world property and that will serve as one of our *small-world models.* Quite literally, the Sisters can have the best of both worlds!

The construction that we are describing here is a modification of the networks that Watts and Strogatz introduced in their seminal paper [27]. The main difference is that the small-world models of [27] were constructed by randomly *rewiring* a small fraction of the edges in networks $G^1_{NN}(N,d)$ rather than randomly adding new edges. The modification we are using here was proposed in [23]. It gives networks that have similar properties to the ones studied by Watts and Strogatz in [27], but are slightly easier to define and a lot easier to analyze mathematically.

Let us give a mathematically rigorous definition of our small-world networks. We construct random graphs $G = G^{dim}_{SW}(N,d,\lambda)$ for $dim = 1,2$, $d \geq 1$, and $\lambda > 0$ as follows:

- $V(G) = \{1, \ldots, N\}$ or $V(G) = \{0, \ldots, N-1\}$ depending on how you prefer to number nodes.
- Let $G_{short} = G_{NN}^{dim}(N, d)$.
- Randomly choose an Erdős-Rényi graph $G_{long} = G_{ER}(N, \lambda)$.
- $E(G) = E(G_{short}) \cup E(G_{long})$.

Notice that for fixed values of dim, d, λ and variable N we have defined another class of random graphs. The symbol $G = G_{SW}^{dim}(N, d, \lambda)$ will denote both the entire class and its instances, and we will rely on the context to point to the intended meaning.

Since we use 4 parameters and need to distinguish between several types of networks, the notation is a bit heavy on symbols here. But we can intuitively think of $G_{SW}^{dim}(N, d, \lambda)$ as having two types of edges: *short edges* that it inherits from $G_{NN}^{dim}(N, d)$ and *long edges* that it inherits from $G_{ER}(N, \lambda)$. In the example of the Sisters of the Round Table the short edges represent the contacts between sisters that are seated close to each other at the table, and the long edges represent the more informal contacts that they make in the hall with sisters from across the table. It is possible that some edges will be simultaneously "short" and "long" in this sense; we will always treat them as long ones. Incidentally, this distinction shows what our small-world models have in common with real contact networks: The short edges can be thought of as connections made in the local neighborhood, where one would expect strong clustering, and the long edges as connections made by long-distance travel, which accounts for the small-world property. In real contact networks there would typically also be a lot of randomness in the short edges, but imposing a rigid structure on this part of a network gives us models that are relatively easy to study.

Let us mention that one can easily generalize the definition given above by starting with higher-dimensional versions of nearest-neighbor networks and/or using different types of random networks (such as the scale-free networks of [11] or [20]) instead of Erdős-Rényi networks $G_{ER}(N, \lambda)$ for the long edges. Exploration of the resulting small-world models is suggested in our research projects of Section 6. As a warm-up, we recommend the following:

Challenge Problem 2. Rigorously define $G = G_{SW}^{dim}(N, d, \lambda)$ for $dim > 2$.

5.3 Exploring Small Worlds

5.3.1 Small World Models in IONTW

Open IONTW, click **Defaults**, and change the parameter settings to match those in Table 10. Click **New**. In the **World** window you will see a network $G_{NN}^{2}(100, 2)$, which is a 10-by-10 rectangular grid with diagonals. Since we have set **lambda** = 0, this isn't really a small-world model; it has no long edges whatsoever. But we want to use it as a baseline for comparison. Click **Metrics** to record its properties for future reference.

Table 10 Settings for exploring Small World models.

num-nodes	d	λ	network-type
100	2	0	Small World 2

Now change **lambda** := 0.2 and press **New** again. This will create an instance of $G^2_{SW}(100, 2, 0.2)$. You will see a few long edges in addition to the short ones. Click **Metrics**, and repeat for **lambda** := 0.4, 0.6. 0.8. Be sure to click **Metrics** after creation of each network.

You will have seen more and more long edges appearing as you increased **lambda**. Let us see what their addition did to the network properties. Enlarge the **Command Center** by clicking on the double-arrow icon and look at the data. Most likely you will observe that as the parameter λ increases in increments of 0.2, the following changes in the characteristic properties of $G^2_{SW}(100, 2, \lambda)$ occur:

- The mean degree $\langle k \rangle$ increases; roughly in increments of 0.2.
- The edge density slightly increases.
- The (normalized) network clustering coefficients decrease.
- The mean distance decreases.
- The diameter may decrease.

Your data may not give a clear-cut picture due to the inherent randomness, but the general pattern should be discernible.

Let us look at a larger network. Set **num-nodes** := 400. Repeat the previous experiment for **lambda** := 0, 0.8.

You may observe that although the mean degree increases by only about 10%, in $G^2_{SW}(400, 2, 0.8)$ the mean distance and the diameter are less than half of what they were in $G^2_{SW}(400, 2, 0)$.

How could adding a few random long edges have such a large effect? This certainly seems puzzling. In Subsection 2.1 of our module *Small-world models* [7] you will find a sequence of explorations with IONTW that illustrate in detail the mechanism behind this phenomenon.

5.3.2 Mathematical Derivations of Some Properties of $G^{dim}_{SW}(N, d, \lambda)$

We will focus on three network properties here: The mean degree $\langle k \rangle$, the normalized network-clustering coefficient C_{norm}, and the median distance between any two nodes. Moreover, in order to sidestep some distracting technicalities, for the two-dimensional case we will mostly focus only on networks $G^2_{SW}(N^2, d, \lambda)$, although the results generalize to small-world models based on rectangular rather than square grids, many of which are implemented in IONTW as $G^2_{SW}(N, d, \lambda)$. Let us begin with a simplified version of Exercise 3 of [7].

Exercise 19. Find approximations to the mean degrees in $G^1_{SW}(N, d, \lambda)$ and in $G^2_{SW}(N^2, 1, \lambda)$ that a.a.s. approach the true mean degrees $\langle k \rangle$ when $N \to \infty$.

We have optimistically labeled the graphs $G_{SW}^{dim}(N,d,\lambda)$ "small-world models." But do they live up to this name? We will need to show that our two classes of networks are small-world networks, that is, exhibit strong clustering and have the small-world property.

Strong clustering requires that normalized network clustering coefficients C_{norm} will grow without bounds when the network size increases without bound while the other two parameters, d and λ, remain fixed. By Exercise 12(b) of Section 3.4, the graphs $G_{NN}^1(N,d)$ and $G_{NN}^2(N^2,d)$ with $d > 1$ have the property that $tr(i) \geq 1$ for each node i, and the same will continue to hold in the corresponding small-world networks $G_{SW}^1(N,d,\lambda)$ and $G_{SW}^2(N^2,d,\lambda)$. Moreover, by Exercise 19 above, for each of these classes the expected mean degree $\langle k \rangle$ will a.a.s. approach some finite number that depends on the particular choice of d, λ. Thus each of the classes $G_{NN}^1(N,d,\lambda)$ and $G_{NN}^2(N^2,d,\lambda)$ with fixed $d > 1$ and λ satisfies the assumptions of Theorem 2 of Section 3.4, and we can conclude that each of these classes exhibits strong clustering.

Exercise 20.

(a) Consider $G_{SW}^{dim}(N,d,\lambda)$, where the parameters dim, N, d are fixed, $d > 1$, and the network size N is very large. Explain why you *usually* will see a decrease in the normalized network clustering coefficient C_{norm} when you increase the value of λ.

(b) What behavior of C_{norm} would you expect in the setup for part (a) when you set $d = 1$? How would you explain the predicted differences from part (a)?

Finally, let us consider the small-world property. We need to show that for some constants P and c, the P-th percentile of the distance between two randomly chosen nodes will a.a.s. be less than $c\ln(N)$.

If $\lambda > 1$, then the graph $G_{long} = G_{ER}(N,\lambda)$ is predicted to a.a.s. contain a giant component of diameter $< c_{diam}\ln(N)$ for some constant c_{diam}. Thus a.a.s., if i,j are nodes in the giant component of $G_{long} = G_{ER}(N,\lambda)$, there will be a path of length $< c_{diam}\ln(N)$ from i to j. Since each such path is also a path in $G_{SW}^{dim}(N,d,\lambda)$, the small-world property of the latter class of networks follows.

However, if $0 < \lambda < 1$, then the class of graphs $G_{ER}(N,\lambda)$ does not have the small-world property. But our small-world models do. Even if λ is very small but positive, we will still be able to travel from any node i to any node j along a path of very small length. This path may need to contain both short and long edges though.

The following exercise gives important insights into the structure of typical small-world networks. A hint is given in the Appendix, before the references, but first try doing the problem without a hint.

Exercise 21. Give an intuitive explanation why for any choice of d and of $\lambda > 0$ the class of networks $G_{SW}^1(N,d,\lambda)$ should have the small-world property.

The following problem will be good practice for the projects of Section 6; we recommend starting with the case $dim = 1$.

Challenge Problem 3. Give a formal proof that for any choice of d, dim and of $\lambda > 0$ the class of networks $G_{SW}^{dim}(N, d, \lambda)$ has the small-world property.

5.4 Vaccination Strategies in Small-World Models

In order to avoid some technical complications, throughout this subsection and Section 6 we will focus only on models of type *SIR*. For these models, *vaccination* can be interpreted as moving a subset of all nodes into the **R**-compartment prior to any outbreak so that these nodes will no longer be susceptible. A *vaccination strategy* is a procedure for choosing the set of hosts to be vaccinated.

Vaccination is an important *control measure*; its goal is to reduce to the extent possible the number of hosts who will experience infection during a possible outbreak. Thus the *effectiveness* of a given vaccination strategy can be quantified either in terms of how much it reduces the probability that a subsequent outbreak caused by a single index case will be a major one, or how much it reduces the expected final size.

Of course, a strategy of vaccinating the entire population would prevent all outbreaks. But such a strategy may not always be feasible: Not enough vaccine may be available, logistical problems may prevent administering the vaccine to all hosts, or some hosts may be unwilling to comply.

A more realistic approach may be to vaccinate xN randomly chosen hosts, where x is a proportion with $0 < x < 1$. These *random* vaccination strategies are in fact the only ones that can be studied when all hosts are assumed to have identical properties and the uniform mixing assumption is made. Recall that in network-based models this assumption is made when the contact network is a complete graph. For models with other given contact networks one can consider *targeted* vaccination strategies, where a particular subset of all hosts is chosen to receive the available doses of vaccine. We will explore one such strategy later in this subsection and ask you to investigate more of them in some of the projects.

An important prediction of compartment-level models is that random vaccination of xN nodes will guarantee that all outbreaks are restricted to minor ones as long as $x \geq HIT$, where the critical proportion HIT is called the *herd immunity threshold* and is given by

$$HIT = \frac{K}{N} = 1 - \frac{1}{R_0}. \tag{12}$$

For example, when $R_0 = 3.6$, then $HIT = 1 - \frac{1}{R_0} = 1 - \frac{1}{3.6} = 0.7222$. For a population of $N = 120$ hosts, we would need to administer vaccine to at least 72.22% or 87 hosts.

We now turn to IONTW to investigate the role of HIT in simulated outbreaks using a few exercises adapted from Project 9.1 of [17] and Section 9.4.3 of [17]. Open IONTW, click **Defaults**, and change the parameter settings according to Table 11.

Table 11 Parameter settings for investigating *HIT* in compartment-level models.

infection-prob	end-infection-prob	num-nodes
0.03	1	120

Note that here we are simulating a next-generation *SIR*-model with $N = 120$ hosts. Press **New** followed by **Metrics** to verify $R_0 = 3.57$, which we will round to $R_0 = 3.6$.

What will happen if we vaccinate $K = 87$ hosts? To vaccinate hosts (aka "turtles" in IONTW), in the command center next to `observer>` type:

```
ask n-of 87 turtles [become-removed]
```

Press **enter**, then type (all in one line):

```
ask n-of 1 turtles with [susceptible?]
[become-infectious]
```

Press **enter** and then **Go** and investigate the severity of the outbreak. Press **Last** and then **Go** to run a few more trials. What do you observe? Is the population avoiding major outbreaks?

We can reach more reliable conclusions by running a batch experiment. Set up a **New** experiment with 100 repetitions and use the following:

Reporters:
`count turtles with [removed?]`
Setup commands:
```
new-network
ask n-of 87 turtles [become-removed]
ask n-of 1 turtles with [susceptible?]
[become-infectious]
```

Exercise 22.

(a) Run the experiment and analyze your data by sorting the output column (`count turtles with [removed?]`) from lowest to highest, as you did in Exercise 5. For each run, calculate the proportion of unvaccinated hosts who are not the index case and who experienced infection. Remember that your output column will include the number of vaccinated hosts and the index case, so you will need to account for this in your calculations.

Do your results indicate that the population of unvaccinated hosts avoided a major outbreak every time? Should you have vaccinated more individuals to be safe? What would you predict would happen if you vaccinated fewer hosts?

(b) Redo part (a) by vaccinating $1.2 * 87$ and $0.8 * 87$ hosts, respectively. (Note: Round up to the nearest integer.) Compare your results to the mean and maximum proportions of unvaccinated hosts who experienced infection in part (a). Also compare the proportions of simulations where only one host (the index case) experiences infection. This means that no *secondary infections* occurred.
(c) Is there an equal effect in increasing the proportion of vaccinated hosts from $0.8 * HIT$ to HIT as there is in increasing the proportion from HIT to $1.2 * HIT$? What does this mean in terms of recommendations for public health policy?

Next we turn to a different type of contact network, where we will consider targeted vaccination strategies. In such networks we might be able to prevent major outbreaks by vaccinating a small number of hosts positioned at key locations in the network. A major challenge lies in determining this optimal subset of hosts.

In IONTW click **Defaults** and change the parameter settings according to Table 12.

Here we are simulating a next-generation *SIR*-model on a $G^1_{NN}(120, 2)$ contact network. Press **New** then **Metrics** to verify $R_0 = 3.6$, as in our previous example on a complete graph. Do you expect we will need to vaccinate the same number of hosts to obtain herd immunity? The next exercise will assist you in answering this question.

Exercise 23. Set up a **New** experiment with 100 repetitions and enter the exact same prompts that preceded Exercise 22. Repeat all steps and answer all questions from part (a) of Exercise 22. What differences do you notice once your contact network has changed, even though your R_0 value is the same?

Suppose now that we have only a limited amount of vaccine so that we are only able to vaccinate 10% of the population. In our population of 120 hosts, this amounts to vaccinating 12 individuals. What would happen in the compartment-level model (i.e., a complete graph contact network) if we only vaccinated 12 hosts? What do you think would happen in the case of the $G^1_{NN}(120, 2)$ contact network? Let's run some more simulations. Select your previous batch processing experiments from Exercises 22 and 23 and **Edit** each one by selecting a different **Experiment name**. For each of these two experiments, edit the second line of **Setup commands** as follows:

```
ask n-of 12 turtles [become-removed]
```

so that you are now only vaccinating 12 hosts instead of 87.

Table 12 Parameter settings for investigating HIT in $G^1_{NN}(N, d)$ models.

infection-prob	end-infection-prob	d	network-type
0.9	1	2	Nearest-neighbor 1

Exercise 24. Repeat the instructions in Exercises 22 and 23. Have your predictions for the compartment-level model changed when the contact network was $G^1_{NN}(120, 2)$?

The results of the preceding exercise suggest that for a given class C of network-based models that has representatives with arbitrarily large N, there might be a proportion HIT_C such that if random vaccination is applied to more than xN hosts, then for any fixed x:

- If $x > HIT_C$, then the expected final size and the probability of a major outbreak will a.a.s. approach 0.
- If $x < HIT_C$, then the expected final size and the probability of a major outbreak will a.a.s. exceed some fixed values $r(x)$ with probability $1 - z_\infty(x) > 0$.

When defining the class C, one needs to specify the network type, the model type, and the relevant network and disease transmission parameters, which may depend on N. The classical result about the herd-immunity threshold says that if the networks in C are complete graphs and the disease transmission parameters depend on N in such a way that R_0 remains fixed for all networks in this class, then HIT_C exists and is given by (12).

Under compartment-level models, we have no choice other than to vaccinate hosts at random. However, perhaps with the $G^1_{NN}(120, 2)$ contact network we can do better. In IONTW, to vaccinate a specific subset of hosts, it is most convenient to create a `.txt` file containing a list of the hosts you wish to vaccinate. For example if you wanted to vaccinated hosts $1, 2, 80$, the only text your file should contain is:

```
[ 1 2 80 ]
```

Create such a file in the same directory where you are running IONTW, name it `vaccinate.txt`. In the command line, next to `observer>` type:

```
ask turtles-from-file "vaccinate.txt" [become-removed]
```

Hit **enter** and then click **Labels** to ensure your file has indeed vaccinated the intended hosts. Once you have verified your file is working properly, select your previous batch processing experiment and **Edit** it by selecting a different **Experiment name**. Then edit the second line of **Setup commands** as follows:

```
ask turtles-from-file "vaccinate.txt" [become-removed]
```

Such an experiment will now vaccinate only turtles labeled 1, 2, and 80.

Exercise 25. Now return to our problem of having only enough vaccine for 12 hosts. What would you consider to be the best vaccination strategy for our $G^1_{NN}(120, 2)$ contact network? Edit your `"vaccinate.txt"` file according to

Table 13 Parameter settings for testing vaccination strategy.

infection-prob	end-infection-prob	network-type	num-nodes	lambda	d
0.9	1	Small World 1	120	0	2

your proposed strategy, and run another batch experiment similar to that of Exercise 23, but now with your 12 selected hosts to vaccinate. Compare your results to previous results in terms of the minimum, maximum, and mean proportions of hosts who experienced infection. Remember to account for the 12 individuals that start in the Removed compartment in calculating the correct proportion. Note: It is recommended that you try several vaccination strategies to determine which is optimal.

Note that the contact network in the above explorations is the limiting case $G^1_{NN}(120, 2)$ of $G^1_{SW}(120, 2, \lambda)$ for $\lambda = 0$. The performance of your vaccination strategy will deteriorate when we increase λ. Before proceeding with an exploration of this effect, to minimize the possibility of error, verify that your vaccination strategy is equivalent to the one we propose in the Appendix hints.

Open IONTW, click **Defaults**, and change the parameter settings according to Table 13. This will set up a next-generation *SIR*-model on a network $G^1_{NN}(120, 2)$. Press **New** followed by **Metrics,** and look up the value of `R0` for this model in the **Command Center**. Convince yourself that it is $R_0 = 3.6$.

Create a plain text file that contains your vector of hosts to be vaccinated as a single line and save it under the name `vaccinate.txt` in the same directory in which you are running IONTW.

To get a set of baseline data, run a batch of 100 simulations for this vaccination strategy using the same set of batch processing commands as in Exercise 25.

Exercise 26. Run the experiment and analyze your output file. Record the minimum, maximum, and mean number of removed nodes at the end. Also record the numbers of runs where you found exactly 30, 48, 66, 84, 102, and 120 hosts who experienced infection.

Now **Edit** your setup for batch processing by setting

```
["lambda" 0.05]
```

and specifying a new suggestive output file name. Run the experiment. Then **Edit** your setup for batch processing again by setting

```
["lambda" 0.1]
```

and specifying another suggestive output file name. Run the experiment.

Exercise 27.
(a) Analyze your output files as you did for Exercise 26.

(b) Verbally summarize your findings. How does the parameter λ appear to influence disease transmission? Does the strategy of vaccinating evenly spaced groups of two adjacent hosts provide as good protection when the contact network is $G^1_{SW}(120, 2, \lambda)$ as it does for contact networks $G^1_{NN}(120, 2)$? Does it provide better protection than randomly vaccinating 12 hosts?
(c) Find an explanation for the high frequencies with which you observed outcomes of 30, 48, 66, 84, 102, or 120 hosts who experienced infection. How is this pattern related to the structure of the network and the disease transmission parameters?

6 Suggested Research Projects

Now that you have learned about disease transmission on small-world models, vaccination strategies, and done a number of exercises, you are ready to attempt some research-level projects. We describe five such projects here. Since these are open research problems, we don't know how difficult it would be to arrive at a complete answer to all parts of a project. However, we know that at least some parts of each of these projects are feasible for aspiring researchers at the undergraduate level. And even substantial partial answers are likely to go beyond what is already known in the literature! In all cases, the best strategy would be to start with simulations to form conjectures, and then to try finding rigorous mathematical proofs for your conjectures. And don't forget to consult the vast and rapidly growing literature on the subject to see whether a solution to your project has been published in the meantime!

We formulated all our projects for small-world networks of arbitrary dimension, but they are of interest even if you restrict yourself just to the case $dim = 1$. This case is the one that you want to always start with. The answers to the questions in our projects will in general depend on whether you explore them for discrete-time or continuous-time models and on the particular choice of disease-transmission parameters. We recommend that you always start with exploring next-generation models. For these models, the simulations run fast, and they are easier to analyze mathematically. The simulations for the first three projects (at least for dimensions one and two) can be done with the current version of IONTW. The simulations for last two projects require some software development, for example, a suitable extension of the current capabilities of our NETLOGO code.

Research Project 1. Explore the herd immunity threshold HIT_C for classes C of models that are based on contact networks $G^{dim}_{SW}(N, \lambda, d)$, where N is arbitrarily large, while dim, λ, d and all disease transmission parameters are fixed.

In particular, for Research Project 1 you want to:

(a) Explore numerically how HIT_C depends on λ and d for $dim = 1, 2$ and representative choices of disease transmission parameters.
(b) Prove mathematically that HIT_C actually exists.
(c) Conjecture an approximate formula for the dependence of HIT_C in terms of the defining parameters of C, and prove it.

Project 1 assumes that individuals to be vaccinated are chosen randomly. The next project aims at comparing the effectiveness of two kinds of targeted strategies.

Research Project 2. Assume you have enough vaccine to vaccinate a fixed proportion x (where $0 < x < 1$) of all hosts. Explore two types of targeted vaccination strategies, $B(x)$ and $H(x)$, as described below.

In particular, for Research Project 2 you want to:

(a) Rigorously formulate a definition of the vaccination strategy $B(x)$ that allows for creating as many evenly spaced barriers in $G_{SW}^{dim}(N, d, \lambda)$ with the available amount of vaccine. Note that this strategy works in a similar way as the strategy provided in the hint for Exercise 27.
(b) Rigorously formulate a definition of the vaccination strategy $H(x)$ that allows for vaccinating the nodes with the highest degrees with the available amount of vaccine.
(c) For these strategies, explore with simulations with various values of dim, d, λ and the disease transmission parameters the expected value of the final size and probabilities of major outbreaks. In particular, explore when $B(x)$ appears to be the better strategy and when $H(x)$ outperforms it.
(d) Formulate mathematical conjectures based on your findings in point (c) and rigorously prove them.

As far as we know, at the time of this writing there is no literature on the effectiveness of $B(x)$. However, some numerical explorations of the effectiveness of strategy $H(x)$ for one-dimensional small-world models have been reported in [30]. For two-dimensional small-world models with $d = 1$ this strategy (along with several additional ones) has also been explored numerically in [5] and for $d = 2$ in [29]. However, the small-world models studied in these papers are based on rewiring a small proportion of nodes rather than adding new random connections.

In Project 2 we assumed that we can in fact precisely target the set of hosts to be vaccinated. In practice this may not be possible, due to logistic difficulties in administering the vaccine or due to the growing problem of unwillingness to comply with a vaccination program. Several of the articles in [22] discuss recent research on this problem.

Note that in the case of strategy $B(x)$ this may create some "leaky barriers," and in the case of strategy $H(x)$ you may fail to vaccinate some highly connected hosts.

Research Project 3. Redo the explorations of Research Project 2 under the additional assumption that only a proportion of y randomly chosen hosts among those you target for vaccination can in fact be vaccinated.

In particular, for Research Project 3 you want to:

(a) Rigorously formulate a definition of the vaccination strategy $B(x, y)$ that allows for creating as many evenly spaced barriers in $G_{SW}^{dim}(N, d, \lambda)$ with the available amount x of vaccine.
(b) Rigorously formulate a definition of the vaccination strategy $H(x, y)$ that allows for vaccinating the nodes with the highest degrees with the available amount x of vaccine.
(c) Explore the strategies $B(x, y)$ and $H(x, y)$ as suggested for $B(x), H(x)$ in the instructions for Research Project 2.

There are other types of small-world models than the ones we introduced here. For example, the random edges can be created by randomly rewiring some connections as in [27], or be modeled as some other random graph, such as a random scale-free network (see [11, 20]).

Research Project 4. Do an analogue of any of Research Projects 1–3 for your favorite alternative type of small-world model and compare the findings with those for the small-world models that are already implemented in IONTW.

In all of the above projects we have assumed that the network connectivity is fixed. But in the real world contact networks change over time. People may break old connections and form new ones. This will happen slowly and independently of any disease outbreak, but the process may be accelerated during an ongoing outbreak, for example, when diseased hosts isolate themselves (thus breaking all their connections), or people respond to the news of overall disease prevalence by traveling less (thus breaking some of their long-distance connections). The study of such responses to an outbreak is the domain of *behavioral epidemiology;* see [22] for an overview of recent research in this field.

Research Project 5. Redo your favorite among Research Projects 1–3 for a model with rewiring.

In particular, for Research Project 5 you want to:

(a) Rigorously formulate a model of how a network gets rewired over time according to your chosen scenario.
(b) Implement a simulation program that would represent the rewiring process you formulated in point (a) when you start with a small-world model. This might be done as an extension of IONTW.
(c) Redo the explorations for your favorite among Research Projects 1–3 for this implementation and compare the findings with those for the underlying small-world models without rewiring.

The projects we have listed here are representative of what can be explored along these lines, but they are far from exhausting all possibilities. If you stumble upon a question in this area that seems worth exploring with simulations and theoretically, by all means, do explore them, and let us know about your findings!

Appendix: Hints for Selected Exercises

Hint for Exercises 1 and 2 In Exercise 1, the large components will always be of the same size; in Exercise 2, their sizes will most likely be similar, but not exactly the same.

Hint for Exercise 3 If you followed instructions on batch processing from [13], choosing the output option **Table output**, the data you will need (total number of removed turtles) should be in the rightmost column of your output file. You may wish to sort your data in this column from lowest to highest. All hosts who are removed by the end of the simulation must be in the same connected component as the index case j^*. In the majority of simulations, these components should be large, as in Exercise 2.

Hint for Exercise 4 You can use the mean size of the components that you classified as "large" for your estimate of $\varrho(1.5)$.

Hint for Exercise 12(a) Choose an N large enough so that you can test several values of d. Find a pattern, dependent on d, for the maximum degree of any node.

Hint for Exercise 15 The results of Milgram's experiment do not tell us anything about the maximum or mean distance. They give a statistically significant estimate of some percentile of the distances, but not necessarily of the median.

Hint for Exercise 21 The mean degrees will a.a.s. approach $\langle k \rangle \approx 2d + \lambda$ in $G^1_{SW}(N, d, \lambda)$ and $\langle k \rangle \approx 4 + \lambda$ in $G^2_{SW}(N^2, 1, \lambda)$.

Hint for Exercise 23(a) Partition the vertex set V of the small-world model into pairwise disjoint sets of consecutively numbered nodes V_i of roughly equal size. Then form a new graph G_I by making each of the *sets* V_i a vertex and drawing an edge $\{V_i, V_j\}$ if, and only if, there is at least one $v^* \in V_i$ and at least one $v^{**} \in V_j$ such that $\{v^*, v^{**}\}$ forms an edge in the original small-world network. For an illustration of this construction see the hint at the very end of Module [7].

It can be shown that (a slight modification of) this construction will give Erdős-Rényi random graphs G_I with sufficiently large mean degrees.

Hint for Exercise 24

(a) You may still see a few outbreaks that look fairly major, which is due to random effects in a relatively small population.

(b) Results will vary, but you may see results similar to the following:

(rounded to 2 decimals)	**K = 70**	**K = 87**	**K = 105**
mean prop. of secondary infections	0.35	0.10	0.04
max. prop. of secondary infections	0.86	0.67	0.47
prop. of runs with no secondary infections	0.17	0.37	0.67

Hint for Exercise 25 Note: Do NOT duplicate your experiment from Exercise 24, as this will change the parameter settings. Instead, create a New experiment. It is highly unlikely to see any major outbreaks. We got: mean prop. of secondary infections = 0.04; max prop. of secondary infections = 0.15; prop. of runs with no secondary infections = 0.31. Your results should be in the same ballpark.

Hint for Exercise 26 If you predicted you will still see major outbreaks, you would be correct. For the complete network, we got: mean prop. of secondary infections = 0.90; max prop. of secondary infections = 0.97; and prop. of runs with no secondary infections = 0.05. For the $G^1_{NN}(120, 2)$ network, we got: mean prop. of secondary infections = 0.65; max prop. of secondary infections = 0.99; and prop. of runs with no secondary infections = 0. Your results may be similar. After sorting the results from the lowest to the highest value of the output column, you will most likely observe a dramatic separation between major and minor outbreaks for the complete network, and a gradual increase without a distinctive gap between minor and major outbreaks for the $G^1_{NN}(120, 2)$ network.

Hint for Exercise 27 The optimal strategy would create barriers that the pathogens cannot cross by vaccinating evenly spaced groups of two adjacent hosts. One vector that implements this type of strategy is

```
[ 1 2 21 22 41 42 61 62 81 82 101 102 ].
```

References

1. Bollobás, B.: Random Graphs, 2nd edn. Cambridge University Press, Cambridge (2001)
2. Diekmann, O., Heesterbeek, H., Britton, T.: Mathematical Tools for Understanding Infectious Disease Dynamics. Princeton University Press, Princeton (2012)
3. Draief, M., Massoulié, L.: Epidemics and Rumors in Complex Networks. London Mathematical Society Lecture Note Series, vol. 369. Cambridge University Press, Cambridge (2010)
4. Erdős, P., Rényi, A.: On the evolution of random graphs. Selected Papers of Alfréd Rényi. **2**, 482–525 (1976)
5. Hartvigsen, G., Dresch, J., Zielinski, A., Macula, A., Leary, C.: Network structure, and vaccination strategy and effort interact to affect the dynamics of influenza epidemics. J. Theor. Biol. **246**(2), 205–213 (2007)
6. Just, W.: A brief review of basic probability theory. https://qubeshub.org/resources/366
7. Just, W., Callender, H.: Small-world models. https://qubeshub.org/resources/404
8. Just, W., Callender, H., LaMar, M.D.: Clustering coefficients. https://qubeshub.org/resources/406
9. Just, W., Callender, H., LaMar, M.D.: Exploring distances with IONTW. https://qubeshub.org/resources/382
10. Just, W., Callender, H., LaMar, M.D.: Exploring Erdős-Rényi random graphs with IONTW. https://qubeshub.org/resources/370
11. Just, W., Callender, H., LaMar, M.D.: Exploring generic scale-free networks. https://qubeshub.org/resources/407
12. Just, W., Callender Highlander, H., LaMar, M.D.: Exploring transmission of infectious diseases on networks with NetLogo. https://people.ohio.edu/just/IONTW/
13. Just, W., Callender, H., LaMar, M.D.: How to use these modules. https://qubeshub.org/groups/iontw/File:InstructionsQ.pdf
14. Just, W., Callender, H., LaMar, M.D.: Modules for exploring transmission of infectious diseases on networks with NetLogo. https://qubeshub.org/groups/iontw/iontwmodules
15. Just, W., Callender, H., LaMar, M.D.: Network-based models of transmission of infectious diseases: a brief overview. https://qubeshub.org/resources/363
16. Just, W., Callender, H., LaMar, M.D.: Disease transmission dynamics on networks: Network structure vs. disease dynamics. In: Robeva, R. (ed.) Algebraic and Discrete Mathematical Methods for Modern Biology, pp. 217–235. Academic, New York (2015)
17. Just, W., Callender, H., LaMar, M.D.: Online appendix: Disease transmission dynamics on networks: Network structure vs. disease dynamics. In: R. Robeva (ed.) Algebraic and Discrete Mathematical Methods for Modern Biology. Academic, New York (2015). http://booksite.elsevier.com/9780128012130/content/Chapter09_Appendix.pdf
18. Just, W., Callender, H., LaMar, M.D., Toporikova, N.: Transmission of infectious diseases: Data, models, and simulations. In: R. Robeva (ed.) Algebraic and Discrete Mathematical Methods for Modern Biology, pp. 193–215. Academic Press (2015)
19. Just, W., Callender, H., LaMar, M.D., Xin, Y.: Exploring random regular graphs with IONTW. https://qubeshub.org/resources/372
20. Just, W., Callender, H., LaMar, M.D., Xin, Y.: The preferential attachment model. https://qubeshub.org/resources/384
21. Just, W., Xin, Y.: A quick tour of IONTW. https://qubeshub.org/resources/201

22. Manfredi, P., D'Onofrio, A.: Modeling the Interplay Between Human Behavior and the Spread of Infectious Diseases. Springer Science & Business Media, New York (2013)
23. Newman, M.E., Watts, D.J.: Renormalization group analysis of the small-world network model. Phys. Lett. A **263**(4), 341–346 (1999)
24. Spencer, J.: The giant component: the golden anniversary. Not. Am Math. Soc. **57**(6), 720–724 (2010)
25. Travers, J., Milgram, S.: An experimental study of the small world problem. Sociometry **32**(4), 425–443 (1969)
26. Vynnycky, E., White, R.: An Introduction to Infectious Disease Modelling. Oxford University Press, Oxford (2010)
27. Watts, D.J., Strogatz, S.H.: Collective dynamics of 'small-world' networks. Nature **393**(6684), 440–442 (1998)
28. Wilensky, U.: Netlogo home page. http://ccl.northwestern.edu/netlogo/
29. Xu, Z., Sui, D.Z.: Effect of small-world networks on epidemic propagation and intervention. Geogr. Anal. **41**(3), 263–282 (2009)
30. Zanette, D.H., Kuperman, M.: Effects of immunization in small-world epidemics. Physica A **309**(3), 445–452 (2002)

Steady and Stable: Numerical Investigations of Nonlinear Partial Differential Equations

R. Corban Harwood

Suggested Prerequisites. *Differential equations, Linear algebra, Some programming experience.*

1 Introduction

Mathematics is a language which can describe patterns in everyday life as well as abstract concepts existing only in our minds. Patterns exist in data, functions, and sets constructed around a common theme, but the most tangible patterns are visual. Visual demonstrations can help undergraduate students connect to abstract concepts in advanced mathematical courses. The study of partial differential equations, in particular, benefits from numerical analysis and simulation.

Applications of mathematical concepts are also rich sources of visual aids to gain perspective and understanding. Differential equations are a natural way to model relationships between different measurable phenomena. Do you wish to predict the future behavior of a phenomenon? Differential equations do just that–after being developed to adequately match the dynamics involved. For instance, say you are interested in how fast different parts of a frying pan heat up on the stove. Derived from simplifying assumptions about density, specific heat, and the conservation of energy, the heat equation will do just that! In Section 2.1 we use the heat equation (6), called a test equation, as a control in our investigations of more complicated partial differential equations (PDEs). To clearly see our predictions of the future behavior, we will utilize numerical methods to encode the dynamics modeled by the PDE into a program which does basic arithmetic to approximate the underlying calculus. To be confident in our predictions, however,

R.C. Harwood (✉)
George Fox University, 414 North Meridian Street, Newberg, OR 97132, USA
e-mail: rharwood@georgefox.edu

A. Wootton et al. (eds.), *A Primer for Undergraduate Research*, Foundations for Undergraduate Research in Mathematics, DOI 10.1007/978-3-319-66065-3_11

we need to make sure our numerical method is developed in a way that keeps errors small for better accuracy. Since the method will compound error upon error at every iteration, the method must manage how the total error grows in a stable fashion. Else, the computed values will "blow up" towards infinity–becoming nonnumerical values once they have exceeded the largest number the computer can store. Such instabilities are adamantly avoided in commercial simulations using *adaptive* methods, such as the Rosenbrock method implemented as `ode23s` in MATLAB [17]. These adaptive methods reduce the step size as needed to ensure stability, but in turn increase the number of steps required for your prediction. Section 2 gives an overview of numerical partial differential equations. Burden [3] and Thomas [22] provide great beginner and intermediate introductions to the topic, respectively. In Section 3 we compare basic and adaptive methods in verifying accuracy, analyzing stability through fundamental definitions and theorems, and finish by tracking oscillations in solutions. Researchers have developed many ways to reduce the effect of numerically induced oscillations which can make solutions appear infeasible [2, 19]. Though much work has been done in studying the nature of numerical oscillations in ordinary differential equations [4, 9], some researchers have applied this investigation to nonlinear evolution PDEs [11, 15]. Recently, others have looked at the stability of steady-state and traveling wave solutions to nonlinear PDEs [10, 16, 18], with more work to be done. We utilize these methods in our parameter analysis in Section 4 and set up several project ideas for further research. Undergraduate students have recently published related work, for example, in steady-state and stability analysis [1, 20] and other numerical investigations of PDEs [13].

2 Numerical Differential Equations

In applying mathematics to real-world problems, a differential equation can encode information about how a quantity changes in time or space relative to itself more easily than forming the function directly by fitting the data. The mass of a bacteria colony is such a quantity. In this example, tracking the intervals over which the population's mass doubles can be related to measurements of the population's mass to find its exponential growth function. Differential equations are formed from such relationships. Finding the pattern of this relationship allows us to solve the differential equation for the function we seek. This pattern may be visible in the algebra of the function, but can be even more clear in graphs of numerical solutions.

2.1 Overview of Differential Equations

An ordinary differential equation (ODE) is an equation involving derivatives of a single variable whose solution is a function which satisfies the given relationship between the function and its derivatives. Because the integration needed to undo each derivative introduces a constant of integration, conditions are added for each

derivative to specify a single function. The order of a differential equation is the highest derivative in the equation. Thus, a first order ODE needs one condition while a third order ODE needs three.

Definition 1. An initial value problem (IVP) with a first order ODE is defined as

$$\begin{aligned} \frac{dx}{dt} &= f(x,t) \\ x(t_0) &= x_0, \end{aligned} \tag{1}$$

where t is the independent variable, $x \equiv x(t)$ is the dependent variable (also called the unknown function) with initial value of $x(t_0) = x_0$, and $f(x,t)$ is the *slope function*.

As relationships between a function and its derivatives, a PDE and an ODE are much alike. Yet PDEs involve multivariable functions and each derivative is a partial derivative in terms of one or more independent variables. Recall that a partial derivative focuses solely on one variable when computing derivatives. For example, $\frac{\partial}{\partial t} e^{-2t}\sin(3x) = -2e^{-2t}\sin(3x)$. Similar to the ways ordinary derivatives are notated, partial derivatives can be written in operator form or abbreviated with subscripts (e.g. $\frac{\partial^2 u}{\partial x^2} = u_{xx}$). *Linear* PDEs are composed of a sum of scalar multiples of the unknown function, its derivatives, as well as functions of the independent variables. A PDE is *nonlinear* when it has term which is not a scalar multiple of an unknown, such as $\rho u(1-u)$ in (3) or an arbitrary function of the unknown. To introduce problems involving PDEs, we begin with the simplest type of boundary conditions, named after mathematician Peter Dirichlet (1805–1859), and a restriction to first order in time (called evolution PDEs). Note that the number of conditions needed for a unique solution to a PDE is the total of the orders in each independent variable [22]. Sufficient number of conditions, however, does not prove uniqueness. The maximum principle and energy method are two ways uniqueness of a solution can be proven [6], but such analysis is beyond the scope of this chapter.

Definition 2. An initial boundary value problem (IBVP) with a first order (in time) evolution PDE with Dirichlet boundary conditions is defined as

$$\begin{aligned} \frac{\partial u}{\partial t} &= f\left(x,t,u,\frac{\partial u}{\partial x},\frac{\partial^2 u}{\partial x^2},\dots\right) \\ u(x,0) &= u_0(x), \\ u(0,t) &= a \\ u(L,t) &= b \end{aligned} \tag{2}$$

where x,t are the independent variables, u is the dependent variable (also called the unknown function) with initial value of $u(x,t) = u_0(x)$ and boundary values $u = a,b$ whenever $x = a,b$ respectively, and f can be any combination of independent variables and any spatial partials of the dependent variable.

Example 1. Let us analyze the components of the following initial boundary value problem:

$$\begin{aligned} u_t &= \delta u_{xx} + \rho u(1-u), \\ u(x,0) &= u_0(x), \\ u(0,t) &= 0 \\ u(10,t) &= 1 \end{aligned} \tag{3}$$

First, the PDE is nonlinear due to the $u(1-u)$ term. Second, the single initial condition matches the 1st order in time (u_t) and the two boundary values match the 2nd order in space (u_{xx}). Thus, this IBVP has sufficient number of conditions needed for a unique solution which supports but does not prove uniquess. Third, parameters δ, ρ and the initial profile function $u_0(x)$ are kept unspecified.

This reaction-diffusion equation is known as the Fisher-KPP equation for the four mathematicians who all provided great analytical insight into it: Ronald Fisher (1890–1962), Andrey Kolmogorov (1903–1987), Ivan Petrovsky (1901–1973), and Nikolaj Piscounov (1908–1977) [7, 14]. Though it is more generally defined as an IVP, in this chapter we study it in its simpler IBVP form. Coefficients δ, ρ represent the diffusion and reaction rates and varying their values lead to many interesting behaviors. The Fisher-KPP equation models how a quantity switches between phases, such as genes switching to advantageous alleles where it was originally studied [7].

The form of the initial condition function, $u_0(x)$, is kept vague due to the breadth of physically meaning and theoretically interesting functions which could initialize our problem. Thus, we will use a polynomial fitting functions `polyfit()` and `polyval()` in example code `PDE_Analysis_Setup.m` to set up a polynomial of any degree which best goes through the boundary points and other provided points. This description of the initial condition allows us to explore functions constrained by their shape within the bounds of the equilibrium point $\bar{u}$ analyzed in Section 4.1.

Exercise 1. Consider the PDE

$$u_t = 4u_{xx}. \tag{4}$$

a) Determine the order in time and space and how many initial and boundary conditions are needed to define a unique solution.
b) Using Definition (2) as a guide, write out the IBVP for an unknown $u(x,t)$ such that it has an initial profile of $\sin(x)$, boundary value of 0 whenever $x = 0$ and $x = \pi$, and is defined for $0 \leq x \leq 1, t \geq 0$.
c) Verify that you have enough initial and boundary conditions as determined previously.
d) Verify that the function,

$$u(x,t) = e^{-4t}\sin(x), \tag{5}$$

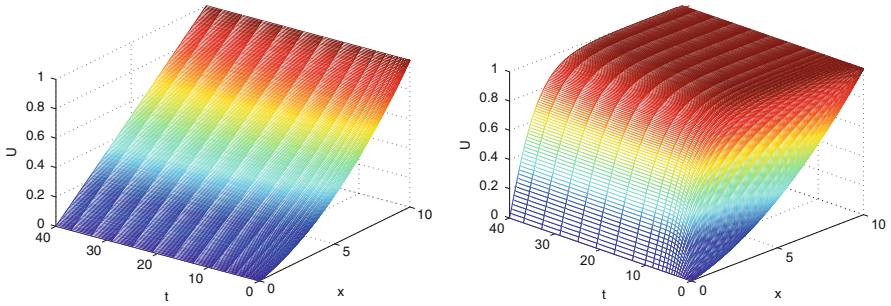


Fig. 1 Comparison of numerical solutions using the adaptive Rosenbrock method (`ode23s` in MATLAB) for the (left) linear Test equation (6) using $\rho = 0$ and (right) Fisher-KPP equation (3) using $\rho = 1$, where all other parameters use the default values of $a = 0, b = 1, L = 10, \delta = 1, \Delta x = 0.05$, degree $= 2, c = \frac{1}{3}$ and initial condition from line 8 in `PDE_Analysis_Setup.m` found in Appendix 5.3.

is a solution to equation (4) by evaluating both sides of the PDE and checking the initial and boundary conditions.

Error is more difficult to analyze for nonlinear PDEs, so it is helpful to have an associated linear version of your equation to analyze first. We will compare our analysis of reaction-diffusion equations to the Test equation,

$$\begin{aligned} u_t &= \delta u_{xx}, \\ u(x,0) &= u_0(x), \\ u(0,t) &= 0 \\ u(10,t) &= 1 \end{aligned} \tag{6}$$

which is the heat equation in one dimension with constant heat forced at the two end points [3]. Note, this is not a direct linearization of the Fisher-KPP equation (3), but it behaves similarly for large values of δ. Figure 1 provides a comparison of the solutions to the linear Test equation (6) and the Fisher-KPP equation (3) for $\delta = 1$. Note how the step size in time for both solutions increases dramatically to have large, even spacing as the solution nears the steady-state solution. Adaptive methods, like MATLAB's ode23s, adjust to a larger step size as the change in the solution diminishes.

2.2 Overview of Numerical Methods

Numerical methods are algorithms which solve problems using arithmetic computations instead of algebraic formulas. They provide quick visualizations and approximations of solutions to problems which are difficult or less helpful to solve exactly.

Numerical methods for differential equations began with methods for approximating integrals: starting with left and right Riemann sums, then progressing to the trapezoidal rule, Simpson's rule and others to increase accuracy more and more efficiently. Unfortunately, the value of the slope function for an ODE is often unknown so such approximations require modifications, such as Taylor series expansions for example, to predict and correct slope estimates. Such methods for ODEs can be directly applied to evolution PDEs (2). Discretizing in space, we create a system of ordinary differential equations with vector $\mathbf{U}(t)$ with components $U_m(t)$ approximating the unknown function $u(x,t)$ at discrete points x_m. The coefficients of linear terms are grouped into matrix $D(t)$ and nonlinear terms are left in vector function $\mathbf{R}(t,\mathbf{U})$. In the following analysis, we will assume that t is not explicit in the matrix D or nonlinear vector $\mathbf{R}(\mathbf{U})$ to obtain the general form of a reaction-diffusion model (2)

$$\frac{d\mathbf{U}}{dt} = D\mathbf{U} + \mathbf{R}(\mathbf{U}) + \mathbf{B}. \tag{7}$$

Example 2. We will discretize the Fisher-KPP equation (3) in space using default parameter values $a = 0, b = 1, L = 10, \delta = 1, \Delta x = 0.05, \rho = 1$ in `PDE_Analysis_ Setup.m` found in Appendix 5.3. See Figure 1 (right) for the graph. Evenly dividing the interval $[0,10]$ with $\Delta x = 0.05 = \frac{1}{20}$ results in 199 spatial points, x_m, where the function is unknown (plus the two end points where it is known: $U_0 = a, U_{200} = b$). Using a centered difference approximation of U_{xx} [3],

$$\begin{aligned}
(U_{xx})_1 &\approx \frac{a - 2U_1 + U_2}{\Delta x^2}, \\
(U_{xx})_m &\approx \frac{U_{m-1} - 2U_m + U_{m+1}}{\Delta x^2},\ 2 \le m \le 198, \\
(U_{xx})_{199} &\approx \frac{U_{198} - 2U_{199} + b}{\Delta x^2},
\end{aligned} \tag{8}$$

the discretization of (3) can be written as

$$\frac{d\mathbf{U}}{dt} = D\mathbf{U} + \mathbf{R}(\mathbf{U}) + \mathbf{B}, \tag{9}$$

$$D = \frac{\delta}{\Delta x^2}\begin{bmatrix} -2 & 1 & \dots & 0 \\ 1 & -2 & \ddots & \dots \\ \dots & \ddots & \ddots & 1 \\ 0 & \dots & 1 & -2 \end{bmatrix}$$

$$\mathbf{R}(\mathbf{U}) = \rho \begin{bmatrix} U_1(1-U_1) \\ \dots \\ U_{199}(1-U_{199}) \end{bmatrix} = \rho \left(I - \text{diag}(\mathbf{U})\right) \mathbf{U}$$

$$\mathbf{B} = \frac{\delta}{\Delta x^2} \begin{bmatrix} a \\ 0 \\ \dots \\ 0 \\ b \end{bmatrix}$$

with a tridiagonal matrix D, a nonlinear vector function $\mathbf{R}$ which can be written as a matrix product using diagonal matrix formed from a vector (diag()), and a sparse constant vector $\mathbf{B}$ which collects the boundary information.

By the Fundamental Theorem of Calculus [3], the exact solution to (7) over a small interval of time Δt is found by integration from t_n to $t_{n+1} = t_n + \Delta t$ as

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \int_{t_n}^{t_{n+1}} \left(D\mathbf{U}(t) + \mathbf{R}(\mathbf{U}(\mathbf{t})) + \mathbf{B}\right) dt, \tag{10}$$

where each component $U_m(t)$ has been discretized in time to create an array of components U_m^n approximating the solution $u(x_m, t_n)$. Note that having the unknown function $\mathbf{U}(t)$ inside the integral (10) makes it impossible to integrate exactly, so we must approximate. Approximating with a left Riemann sum results in the Forward Euler (a.k.a. classic Euler) method [17],

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \Delta t \left(D(\mathbf{U}^n + \mathbf{R}(\mathbf{U}^n) + \mathbf{B}\right), \tag{11}$$

while approximating with a right Riemann sum results in the Backward Euler method [17]

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \Delta t \left(D\mathbf{U}^{n+1} + \mathbf{R}(\mathbf{U}^{n+1}) + \mathbf{B}\right). \tag{12}$$

Although approximating integrals with left and right Riemann sums is a similar task, in solving differential equations, they can be very different. Forward Euler (11) is referred to as an *explicit* method since the unknown $\mathbf{U}^{n+1}$ can be directly computed in terms of known quantities such as the current known approximation $\mathbf{U}^n$, while Backward Euler (12) is referred to as an *implicit* method since the unknown $\mathbf{U}^{n+1}$ is solved in terms of both known $\mathbf{U}^n$ and unknown $\mathbf{U}^{n+1}$ quantities. Explicit methods are simple to set up and compute, while implicit methods may not be solvable at all. If we set $\mathbf{R}(\mathbf{U}) \equiv \mathbf{0}$ to make equation (7) linear, then an implicit method can be easily written in the explicit form, as shown in the Example 4. Otherwise, an unsolvable implicit method can be approximated with a numerical root-finding method such as Newton's method (47) which is discussed in Section 4.2, but nesting numerical methods is much less efficient than implementing

an explicit method as it employs a truncated Taylor series to mathematically approximate the unknown terms. The main reasons to use implicit methods are for stability, addressed in Section 3.3. The following examples demonstrate how to form the two-level matrix form.

Definition 3. A two-level numerical method for an evolution equation (2) is an iteration which can be written in the two-level matrix form

$$\mathbf{U}^{n+1} = M\,\mathbf{U}^n + \mathbf{N}, \tag{13}$$

where M is the combined transformation matrix and $\mathbf{N}$ is the resultant vector. Note, both M and $\mathbf{N}$ may update every iteration, especially when the PDE is nonlinear, but for many basic problems, M and $\mathbf{N}$ will be constant.

Example 3. (Forward Euler) Determine the two-level matrix form for the Forward Euler method for the Fisher-KPP equation (3). Since Forward Euler is already explicit, we simply factor out the $\mathbf{U}^n$ components from equation (11) to form

$$\begin{aligned} \mathbf{U}^{n+1} &= \mathbf{U}^n + \Delta t\left(D(\mathbf{U}^n + \rho\left(I - \operatorname{diag}(\mathbf{U}^n)\right)\mathbf{U}^n + \mathbf{B}\right), \\ &= M\mathbf{U}^n + \mathbf{N}, \\ M &= \left(I + \Delta t D + \Delta t\rho\left(I - \operatorname{diag}(\mathbf{U}^n)\right)\right), \\ \mathbf{N} &= \Delta t\mathbf{B}, \end{aligned} \tag{14}$$

where I is the identity matrix, $\mathbf{N}$ is constant, and M updates with each iteration since it depends on $\mathbf{U}^n$.

Example 4. (Linear Backward Euler) Determine the two-level matrix form for the Backward Euler method for the Test equation (6). In the Backward Euler method (12), the unknown $\mathbf{U}^{n+1}$ terms can be grouped and the coefficient matrix $I - \Delta tD$ inverted to write it explicitly as

$$\begin{aligned} \mathbf{U}^{n+1} &= M\mathbf{U}^n + \mathbf{N}, \\ M &= \left(I - \Delta tD\right)^{-1} \\ N &= \Delta t\left(I - \Delta tD\right)^{-1}\mathbf{B} \end{aligned} \tag{15}$$

where the method matrix M and additional vector $\mathbf{N}$ are constant.

Just as the trapezoid rule takes the average of the left and right Riemann sums, the Crank-Nicolson method (16) averages the Forward and Backward Euler methods [5].

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \frac{\Delta t}{2}D\left(\mathbf{U}^n + \mathbf{U}^{n+1}\right) + \frac{\Delta t}{2}\left(\mathbf{R}(\mathbf{U}^n) + \mathbf{R}(\mathbf{U}^{n+1})\right) + \Delta t\mathbf{B}. \tag{16}$$

One way to truncate an implicit method of a nonlinear equation into an explicit method is called a *semi-implicit* method [3], which treats the nonlinearity as known information (evaluated at current time t_n) leaving the unknown linear terms at t_{n+1}. For example, the semi-implicit Crank-Nicolson method is

$$\mathbf{U}^{n+1} = \left(I - \frac{\Delta t}{2}D\right)^{-1}\left(\mathbf{U}^n + \frac{\Delta t}{2}D\mathbf{U}^n + \Delta t\mathbf{R}(\mathbf{U}^n) + \mathbf{B}\right). \tag{17}$$

Exercise 2. After reviewing Example 3 and Example 4, complete the following for the semi-implicit Crank-Nicolson method (17).

a) Determine the two-level matrix form for the Test equation (6). Note, set $\mathbf{R} = \mathbf{0}$.
b) *Determine the two-level matrix form for the Fisher-KPP equation (3).

*See Section 3.3 for the answer and its analysis.

Taylor series expansions can be used to prove that while the Crank-Nicolson method (16) for the linear Test equation (6) is second order accurate (See Definition 18 in Section 5.1), the semi-implicit Crank-Nicolson method (17) for a nonlinear PDE is only first order accurate in time. To increase truncation error accuracy, unknown terms in an implicit method can be truncated using a more accurate explicit method. For example, blending the Crank-Nicolson method with Forward Euler approximations creates the Improved Euler Crank-Nicolson method, which is second order accurate in time for nonlinear PDEs.

$$\begin{aligned}\mathbf{U}^* &= \mathbf{U}^n + \Delta t\left(D\mathbf{U}^n + \mathbf{R}(\mathbf{U}^n) + \mathbf{B}\right), \\ \mathbf{U}^{n+1} &= \mathbf{U}^n + \frac{\Delta t}{2}D\left(\mathbf{U}^n + \mathbf{U}^*\right) + \frac{\Delta t}{2}\left(\mathbf{R}(\mathbf{U}^n) + \mathbf{R}(\mathbf{U}^*)\right) + \Delta t\mathbf{B}.\end{aligned} \tag{18}$$

This improved Euler Crank-Nicolson method (18) is part of the family of Runge-Kutta methods which embed a sequence of truncated Taylor expansions for implicit terms to create an explicit method of any given order of accuracy [3]. Proofs of the accuracy for the semi-implicit (17) and improved Euler (18) methods are included in Appendix 5.1.

Exercise 3. After reviewing Example 3 and Example 4, complete the following for the Improved Euler Crank-Nicolson method (18).

a) Determine the two-level matrix form for the Test equation (6). Note, set $\mathbf{R} = \mathbf{0}$.
b) Determine the two-level matrix form for the Fisher-KPP equation (3).

2.3 Overview of Software

Several software have been developed to compute numerical methods. Commercially, MATLAB, Mathematica, and Maple are the best for analyzing such methods, though there are other commercial software like COMSOL which can do numerical simulation with much less work on your part. Open-source software capable of the same (or similar) numerical computations, such as Octave, SciLab, FreeFEM, etc. are also available. Once the analysis is complete and methods are fully tested, simulation algorithms are trimmed down and often translated into Fortran or C/C++ for efficiency in reducing compiler time.

We will focus on programming in MATLAB, created by mathematician Cleve Moler (born in 1939), one of the authors of the LINPACK and EISPACK scientific subroutine libraries used in Fortran and C/C++ compilers [17]. Cleve Moler originally created MATLAB to give his students easy access to these subroutines without having to write in Fortran or C themselves. In the same spirit, we will be working with simple demonstration programs, listed in the appendix, to access the core ideas needed for our numerical investigations. Programs `PDE_Solution.m` (Appendix 5.2), `PDE_Analysis_Setup.m` (Appendix 5.3), and `Method_Accuracy_ Verification.m` (Appendix 5.5) are MATLAB scripts, which means they can be run without any direct input and leave all computed variables publicly available to analyze after they are run. Programs `CrankNicolson_SI.m` (Appendix 5.4) and `Newton_System.m` (Appendix 5.6) are MATLAB functions, which means they may require inputs to run, keep all their computations private, and can be effectively embedded in other functions or scripts. All demonstrations programs are run through `PDE_Solution.m`, which is the main program for this group.

Example 5. The demonstration programs can be either downloaded from the publisher or typed into five separate MATLAB files and saved according to the name at the top of the file (e.g. `PDE_Analysis_Setup.m`). To run them, open MATLAB to the folder which contains these five programs. In the command window, type `help PDE_Solution` to view the comments in the header of the main program. Then type `PDE_Solution` to run the default demonstration. This will solve and analyze the Fisher-KPP equation (3) using the default parameters, produce five graph windows, and report three outputs on the command window. The first graph is the numerical solution using MATLAB's built-in implementation of the Rosenbrock method (`ode23s`), which is also demonstrated in Figure 1 (right). The second graph plots the comparable eigenvalues for the semi-implicit Crank-Nicolson method (17) based upon the maximum Δt step used in the chosen method (Rosenbrock by default). The third graph shows a different steady-state solution to the Fisher-KPP equation (3) found using Newton's method (47). The fourth graph shows the rapid reduction of the error of this method as the Newton iterations converge. The fifth graph shows the instability of the Newton steady-state solution by feeding a noisy perturbation of it back into the Fisher-KPP equation (3) as an initial condition. This

noisy perturbation is compared to round-off perturbation in Figure 5 to see how long this Newton steady-state solution can endure. Notice that the solution converges back to the original steady-state solution found in the first graph.
To use the semi-implicit Crank-Nicolson method (17) instead of MATLAB's ode23s, do the following. Now the actual eigenvalues of this method are plotted in the second graph.

Exercise 4. Open `PDE_Solution.m` in the MATLAB Editor. Then comment lines 7–9 (type a % in front of each line) and uncomment lines 12–15 (remove the % in front of each line). Run `PDE_Solution`. Verify that the second graph matches Figure 3.
The encoded semi-implicit Crank-Nicolson method (17) uses a fixed step size Δt, so it is generally not as stable as MATLAB's built-in solver. It would be best to now uncomment lines 7–9 and comment lines 12–15 to return to the default form before proceeding. The main benefit of the ode23s solver is that it is adaptive in choosing the optimal Δt step size and adjusting it for regions where the equation is easier or harder to solve than before. This method is also sensitive to *stiff* problems, where stability conditions are complicated or varying. MATLAB has several other built-in solvers to handle various situations. You can explore these by typing `help ode`.

Once you have run the default settings, open up `PDE_Analysis_Setup` in the editor and tweak the equation parameter values `a,b,L,delta,rho,degree,c` and logistical parameter `dx`. After each tweak, make sure you run the main program `PDE_Solution`. The logistical parameters `tspan,dt` for the numerical method can also be tweaked in `PDE_Solution`, and an inside view of Newton iterations can be seen by uncommenting lines 38–39. Newton's method is covered in Section 4.2. A solution with Newton's method is demonstrated in 2(left), while all of the iterations are graphed in Figure 2(right). Note that Figure 2(right) is very similar to a solution which varies over time, but it is not. The graph of the iterations demonstrates how Newton's method seeks better and better estimates of a fixed steady-state solution discussed in Section 4.1.

Exercise 5. In `PDE_Analysis_Setup`, set parameter values, $a = 0, b = 0, L = 10, \delta = \frac{1}{10}, \Delta x = \frac{1}{20}, \rho = 1, \text{degree} = 2, c = 1$. Then, in `PDE_Solution`, uncomment lines 38–39 and run it. Verify that the third and fourth graphs matches Figure 2.
Notice that the iterations of Newton's method in Figure 2(right) demonstrate oscillatory behavior in the form of waves which diminish in amplitude towards the steady-state solution. These are referred to as stable numerical oscillations similar to the behavior of an underdamped spring [3]. These stable oscillations suggest that the steady-state solution is stable (attracting other initial profiles to it), but due to the negative values in the solution in Figure 2(left), it is actually an unstable steady-state for Fisher-KPP equation (3). You can see this demonstrated in the last graph plotted when you ran `PDE_Solution` where there is a spike up to a value around $-4 \times 10^1 2$. This paradox demonstrates that not all steady-state solutions are stable and that the stability of Newton's method differs from the stability of a steady-state solution to an IBVP.

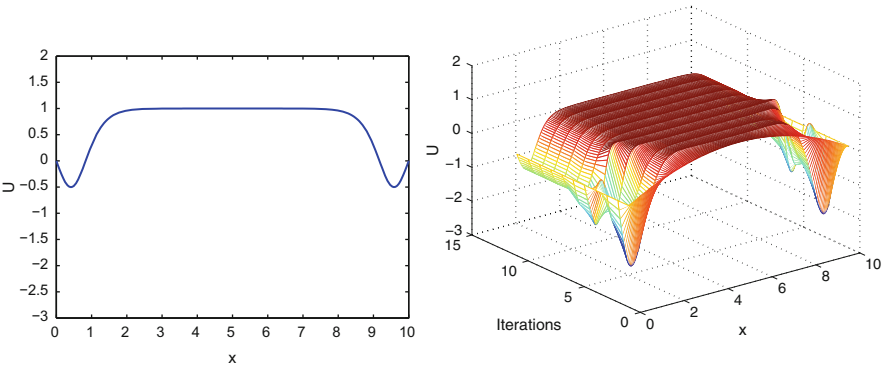


Fig. 2 An example steady-state solution using Newton's method (left) and the iterations to that steady-state (right) using the parameter values $a = 0, b = 0, L = 10, \delta = \frac{1}{10}, \Delta x = \frac{1}{20}, \rho = 1$, degree $= 2, c = 1$ and initial condition from line 8 in `PDE_Analysis_Setup.m` found in Appendix 5.3. Also, uncomment lines 38–39 in `PDE_Solution.m` found in Appendix 5.2.

Some best practices of programming in MATLAB are to clean up before running new computations, preallocate memory, and store calculations which are used more than once. Before new computations are stored in a script file, you can clean up your view of previous results in the command window (`clc`), delete previously held values and structure of all (`clear all`) or selected (`clear` *`name1, name2`*) variables, and close all (`close all`) or selected (`close handle1, handle2`) figures. The workspace for running the main program `PDE_Solution` is actually cleared in line 5 of `PDE_Analysis_Setup` so that this supporting file can be run independently when needed.

When you notice the same calculation being computed more than once in your code, store it as a new variable to trim down the number of calculations done for increased efficiency. Most importantly, preallocate the structure of a vector or matrix that you will fill in values with initial zeros (`zeros(columns,rows)`), so that MATLAB does not create multiple copies of the variable in memory as you fill it in. The code `BCs = zeros(M-2,1);` in line 23 of `PDE_Analysis_Setup.m` is an example of preallocation for a vector. Preallocation is one of most effective ways of speeding up slow code.

Exercise 6. Find the four (total) lines of code in `Newton_System.m`, `Method_Accuracy_Verification.m`, and `CrankNicolson_SI.m` which preallocate a variable.

3 Error Analysis

To encourage confidence in the numerical solution it is important to support the theoretical results with numerical demonstrations. For example, a theoretical condition for stability or oscillation-free behavior can be demonstrated by comparing

solutions before and after the condition within a small neighborhood of it. On the other hand, the order of accuracy can be demonstrated by comparing subsequent solutions over a sequence of step sizes, as we will see in Section 3.1. Demonstrating stability ensures *when*, while accuracy ensures *how rapidly*, the approximation will converge to the true solution. Showing when oscillations begin to occur prevents any confusion over the physical dynamics being simulated, as we will investigate in Section 3.4.

3.1 Verifying Accuracy

Since numerical methods for PDEs use arithmetic to approximate the underlying calculus, we expect some error in our results, including inaccuracy measuring the distance from our target solution as well as some imprecision in the variation of our approximations. We must also balance the mathematical accuracy in setting up the method with the round-off errors caused by computer arithmetic and storage of real numbers in a finite representation. As we use these values in further computations, we must have some assurance that the error is minimal. Thus, we need criteria to describe how confident we are in these results.

Error is defined as the difference between the true value $u(x_m, t_n)$ and approximate value U_m^n, but this value lacks the context given by the magnitude of the solution's value and focuses on the error of individual components of a solution's vector. Thus, the relative error ϵ is more meaningful as it presents the absolute error relative to the true value as long as $u(x_m, t_n) \neq 0$ under a suitable norm such as the max norm $||\cdot||_\infty$.

Definition 4. The *relative error* ϵ for a vector solution $\mathbf{U}^n$ is the difference between the true value $u(x_m, t_n)$ and approximate value U_m^n under a suitable norm $||\cdot||$, relative to the norm of the true value as

$$\epsilon = \frac{||\mathbf{u}(\mathbf{x}, t_n) - \mathbf{U}^n||}{||\mathbf{u}(\mathbf{x}, t_n)||} \times 100\%. \tag{19}$$

The significant figures of a computation are those that can be claimed with confidence. They correspond to a number of confident digits plus one estimated digit, conventionally set to half of the smallest scale division on the measurement device, and specified precisely in Definition 5.

Definition 5. The value U_m^n approximates $u(x_m, t_n)$ to N *significant digits* if N is the largest non-negative integer for which the relative error is bounded by the significant error $\epsilon_s(N)$

$$\epsilon_s(N) = \left(5 \times 10^{-N}\right) \times 100\% \tag{20}$$

To ensure all computed values in an approximation have about N significant figures, definition 5 implies

$$N + 1 > \log_{10}\left(\frac{5}{\epsilon}\right) > N, \tag{21}$$

Although the true value is not often known, the relative error of a previous approximation can be estimated using the best available approximation in place of the true value.

Definition 6. For an iterative method with improved approximations $\mathbf{U}^{(0)}, \mathbf{U}^{(1)}, \ldots, \mathbf{U}^{(k)}, \mathbf{U}^{(k)}$, the *approximate relative error* at position (x_m, t_n) is defined [3] as the difference between current and previous approximations relative to the current approximation, each under a suitable norm $||\cdot||$

$$E^{(k)} = \frac{||\mathbf{U}^{(k+1)} - \mathbf{U}^{(k)}||}{||\mathbf{U}^{(k+1)}||} \times 100\% \tag{22}$$

closely approximates, $\epsilon^{(k)}$, the relative error for the k^{th} iteration assuming that the iterations are converging (that is, as long as $\epsilon^{(k+1)}$ is much less than $\epsilon^{(k)}$).

The following conservative theorem, proven in [21], is helpful in clearly presenting the lower bound on the number of significant figures of our results.

Theorem 1. *Approximation* $\mathbf{U}^n$ *at step n with approximate relative error* $E^{(k)}$ *is correct to at least* $N - 1$ *significant figures if*

$$E^{(k)} < \epsilon_s(N) \tag{23}$$

Theorem 1 is conservatively true for the relative error ϵ, often underestimating the number of significant figures found. The approximate relative error $E^{(k)}$ (22), however, underestimates the relative error ϵ and may predict one significant digit more for low order methods.

Combining theorem 1 with equation (21), the number of significant figures has a lower bound

$$N \geq \left\lfloor \log_{10}\left(\frac{0.5}{E^{(k)}}\right) \right\rfloor. \tag{24}$$

Example 6. Table 1 presents these measures of error to analyze the Crank-Nicolson method (16) for the linear Test equation (6) and the semi-implicit version of the Crank-Nicolson method (17) for the Fisher-KPP equation (3). Column 1 tracks the step size Δt as it is halved for improved approximations $\mathbf{U}^{(k)}$ at $t = 10$. Columns 2 and 5 present the approximate errors in scientific notation for easy readability. Scientific notation helps read off the minimum number of significant figures ensured

Table 1 Verifying Accuracy in Time for Semi-Implicit Crank-Nicolson Method

	Test Equation			Fisher-KPP Equation		
Δt	Approximate Error[a]	Sig. Figs[c]	Order of Accuracy[b]	Approximate Error[a]	Sig. Figs[c]	Order of Accuracy[b]
1	3.8300e-05	4	2 (2.0051)	4.1353e-05	4	-1 (-0.5149)
$\frac{1}{2}$	9.5413e-06	4	2 (2.0009)	5.9091e-05	3	0 (0.4989)
$\frac{1}{4}$	2.3838e-06	5	2 (2.0003)	4.1818e-05	4	1 (0.7773)
$\frac{1}{8}$	5.9584e-07	5	2 (2.0001)	2.4399e-05	4	1 (0.8950)
$\frac{1}{16}$	1.4895e-07	6	2 (2.0000)	1.3120e-05	4	1 (0.9490)
$\frac{1}{32}$	3.7238e-08	7	2 (2.0000)	6.7962e-06	4	1 (0.9749)
$\frac{1}{64}$	9.3096e-09	7	2 (2.0001)	3.4578e-06	5	1 (0.9875)
$\frac{1}{128}$	2.3273e-09	8	2 (1.9999)	1.7439e-06	5	1 (0.9938)

[a] Approximate error under the max norm $||\cdot||_\infty$ for numerical solution $\mathbf{U}^{(k)}$ computed at $t_n = 10$ compared to solution at next iteration $\mathbf{U}^{(k+1)}$ whose time step is cut in half.
[b] Order of accuracy is measured as the power of 2 dividing the error as the step size is divided by 2
[c] Minimum number of significant figures predicted by approximate error bounded by the significant error $\epsilon_s(N)$ as in equation (23)

by Theorem 1, as presented in columns 3 and 6. Notice how the errors in column 2 are divided by about 4 each iteration while those in column 5 are essentially divided by 2. This ratio of approximate relative errors demonstrates the orders of accuracy, p as a power of 2 since the step sizes are divided by 2 each iteration.

$$\frac{\epsilon^{(k+1)}}{\epsilon^{(k)}} = \frac{C}{2^p}, \tag{25}$$

for some positive scalar C which underestimates the integer p when $C > 1$ and overestimates p when $C < 1$. By rounding to mask the magnitude of C, the order p can be computed as

$$p = \text{round}\left(\log_2\left(\frac{\epsilon^{(k)}}{\epsilon^{(k+1)}}\right)\right). \tag{26}$$

Columns 4 and 7 present both rounded and unrounded measures of the order of accuracy for each method and problem. Thus, we have verified that Crank-Nicolson method (16) on a linear problem is second order accurate in time, whereas the semi-implicit version of the Crank-Nicolson method (17) for the nonlinear Fisher-KPP equation (3) is only first order in time.

For comparison, Table 2 presents these same measures for the Rosenbrock method built into MATLAB as ode23s. See example program `Method_Accuracy_Verification.m` in Appendix 5.1 for how to fix a constant step size in such an adaptive solver by setting the initial step and max step to be Δt with a high tolerance to keep the adaptive method from altering the step size.

Table 2 Verifying Accuracy in Time for ode23s Solver in MATLAB

	Test Equation			Fisher-KPP Equation		
Δt	Approximate Error[a]	Sig. Figs[c]	Order of Accuracy[b]	Approximate Error[a]	Sig. Figs[c]	Order of Accuracy[b]
1	1.8829e-05	4	2 (2.00863)	7.4556e-05	3	2 (1.90005)
$\frac{1}{2}$	4.6791e-06	5	2 (2.00402)	1.9976e-05	4	2 (1.98028)
$\frac{1}{4}$	1.1665e-06	5	2 (2.00198)	5.0628e-06	4	2 (1.99722)
$\frac{1}{8}$	2.9123e-07	6	2 (2.00074)	1.2681e-06	5	2 (2.00060)
$\frac{1}{16}$	7.2771e-08	6	2 (2.00054)	3.169e-07	6	2 (2.00025)
$\frac{1}{32}$	1.8186e-08	7	2 (2.00027)	7.9211e-08	6	2 (1.99900)
$\frac{1}{64}$	4.5456e-09	8	2 (1.99801)	1.9817e-08	7	2 (2.00168)
$\frac{1}{128}$	1.138e-09	8	2 (1.99745)	4.9484e-09	8	2 (2.00011)

[a] Approximate error under the max norm $||\cdot||_\infty$ for numerical solution $\mathbf{U}^{(k)}$ computed at $t_n = 10$ compared to solution at next iteration $\mathbf{U}^{(k+1)}$ whose time step is cut in half.
[b] Order of accuracy is measured as the power of 2 dividing the error as the step size is divided by 2
[c] Minimum number of significant figures predicted by approximate error bounded by the significant error $\epsilon_s(N)$ as in equation (23)

Exercise 7. Implement the Improved Euler Crank-Nicolson method (18) and verify that the error in time is $O\left(\Delta t^2\right)$ on both the Test equation (6) and Fisher-KPP equation (3) using a table similar to Table 1.

3.2 Convergence

Numerical methods provide dependable approximations U_m^n of the exact solution $u\left(x_m, t_n\right)$ only if the approximations converge to the exact solution, $U_m^n \to u\left(x_m, t_n\right)$ as the step sizes diminish, $\Delta x, \Delta t \to 0$. Convergence of a numerical method relies on both the consistency of the approximate equation to the original equation as well as the stability of the solution constructed by the algorithm. Since consistency is determined by construction, we need only analyze the stability of consistently constructed schemes to determine their convergence. This convergence through stability is proven generally by the Lax-Richtmeyer Theorem [22], but is more specifically defined for two-level numerical methods (13) in the Lax Equivalence Theorem (2).

Definition 7. A problem is *well-posed* if there exists a unique solution which depends continuously on the conditions.
Discretizing an initial-boundary-value problem (IBVP) into an initial-value problem (IVP) as an ODE system ensures the boundary conditions are well developed for the problem, but the initial conditions must also agree at the boundary for the problem to be well-posed. Further, the slope function of the ODE system needs to be infinitely differential, like the Fisher-KPP equation (3), or at least Lipshitz-continuous, so that Picard's uniqueness and existence theorem via Picard iterations [3] applies to ensure that the problem is well-posed [22]. Theorem 2, proved in [22] ties this altogether to ensure convergence of the numerical solution to the true solution.

Theorem 2 (Lax Equivalence Theorem). *A consistent, two-level difference scheme (13) for a well-posed linear IVP is convergent if and only if it is stable.*

3.3 Stability

The beauty of Theorem 2 (Lax Equivalence Theorem) is that once we have a consistent numerical method, we can explore the bounds on stability to ensure convergence of the numerical solution. We begin with a few definitions and examples to lead us to von Neumann stability analysis, named after mathematician John von Neumann (1903–1957).

Taken from the word *eigenwerte*, meaning one's own values in German, the *eigenvalues* of a matrix define how a matrix operates in a given situation.

Definition 8. For a $k \times k$ matrix M, a scalar λ is an eigenvalue of M with corresponding $k \times 1$ eigenvector $\mathbf{v} \neq \mathbf{0}$ if

$$M\mathbf{v} = \lambda\mathbf{v}. \tag{27}$$

Lines 22–31 of the demonstration code `PDE_Solution.m`, found in Appendix 5.2, compute several measures helpful in assessing stability, including the graph of the eigenvalues of the method matrix for a two-level method (13) on the complex plane. The spectrum, range of eigenvalues, of the default method matrix is demonstrated in Figure 3, while the code also reports the range in the step size ratio, range in the real parts of the eigenvalues, and computes the spectral radius.

Definition 9. The spectral radius of a matrix, $\mu(M)$ is the maximum magnitude of all eigenvalues of M

$$\mu(M) = \max_i |\lambda_i|. \tag{28}$$

The norm of a vector is a well-defined measure of its size in terms of a specified metric, of which the Euclidean distance (notated $||\cdot||_2$), the maximum absolute value ($||\cdot||_\infty$), and the absolute sum ($||\cdot||_1$) are the most popular. See [12] for further details. These measures of a vector's size can be extended to matrices.

Definition 10. For any norm $||\cdot||$, the corresponding matrix norm $|||\cdot|||$ is defined by

$$|||M||| = \max_{\mathbf{x}} \frac{||M\mathbf{x}}{\mathbf{x}}. \tag{29}$$

A useful connection between norms and eigenvalues is the following theorem [12].

Theorem 3. *For any matrix norm* $|||\cdot|||$ *and square matrix M,* $\mu(M) \leq |||M|||$.

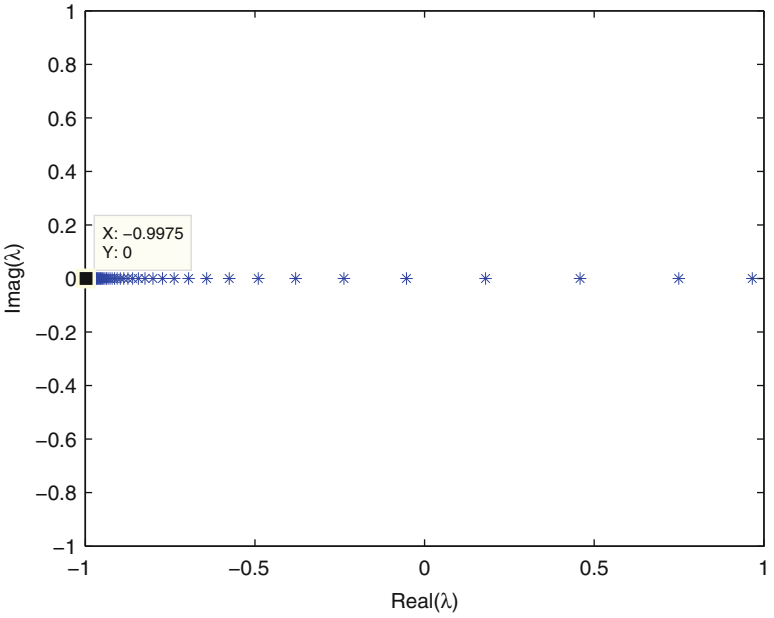


Fig. 3 Plot of real and imaginary components of all eigenvalues of method matrix M for semi-implicit Crank-Nicolson method for Fisher-KPP equation (3) using the default parameter values $a = 0, b = 1, L = 10, \delta = 1, \Delta x = 0.05, \rho = 1, \text{degree} = 2, c = \frac{1}{3}$ and initial condition as given in line 16 of `PDE_Analysis_Setup.m` in Appendix 5.3.

Proof. Consider an eigenvalue λ of matrix M corresponding to eigenvector $\mathbf{x}$ whose magnitude equals the spectral radius, $|\lambda| = \mu(M)$. Form a square matrix X whose columns each equal the eigenvector $\mathbf{x}$. Note that by Definition 8, $MX = \lambda X$ and $|||X||| \neq 0$ since $\mathbf{x} \neq \mathbf{0}$.

$$\begin{aligned} |\lambda|||X||| &= ||\lambda X|| \\ &= ||MX|| \\ &\leq |||M||||||X||| \end{aligned} \tag{30}$$

Therefore, $|\lambda| = \mu(M) \leq |||M|||$.
Theorem 3 can be extended to an equality in Theorem 4 (proven in [12]),

Theorem 4. $\mu(M) = \lim_{k\to\infty} |||M^k|||^{\frac{1}{k}}$.
This offers a useful estimate of the matrix norm by the spectral radius, specifically when the matrix is powered up in solving a numerical method.

Now, we can apply these definitions to the stability of a numerical method. An algorithm is stable if small changes in the initial data produce only small changes

in the final results [3], that is, the errors do not "grow too fast" as quantified in Definition 11.

Definition 11. A two-level difference method (13) is said to be stable with respect to the norm $||\cdot||$ if there exist positive max step sizes Δt_0, *and* Δx_0, and non-negative constants K and β so that

$$||\mathbf{U}^{n+1}|| \leq Ke^{\beta \triangle t}||U^0||,$$

for $0 \leq t,\ 0 < \triangle x \leq \triangle x_0$ *and* $0 < \triangle t \leq \triangle t_0$.

The *von Neumann criterion* for stability (31) allows for stable solution to an exact solution which is not growing (using $C = 0$ for the tight von Neumann criterion) or at most exponentially growing (using some $C > 0$) by bounding the spectral radius of the method matrix,

$$\mu(M) \leq 1 + C\triangle t, \tag{31}$$

for some $C \geq 0$.

Using the properties of norms and an estimation using Theorem 4, we can approximately bound the size of the numerical solution under the von Neumann criterion as

$$\begin{aligned} ||\mathbf{U}^{n+1}|| &= ||M^{n+1}U^0|| \\ &\leq |||M^{n+1}|||\ ||U^0|| \\ &\approx \mu(M)^{n+1}\ ||U^0|| \\ &\leq (1 + C\Delta t)^{n+1}\ \ ||U^0|| \\ &= (1 + (n+1)C\Delta t + \ldots)\ ||U^0|| \\ &\leq e^{(n+1)C\Delta t}\ ||U^0|| \\ &= Ke^{\beta \Delta t}\ ||U^0|| \end{aligned} \tag{32}$$

for $K = 1, \beta = (n+1)C$, which makes the von Neumann criterion sufficient for stability of the solution in approximation for a general method matrix. When the method matrix is symmetric, which occurs for many discretized PDEs including the Test equation with Forward Euler, Backward Euler, and Crank-Nicolson methods, the spectral radius equals the $|||\cdot|||_2$ of the matrix. Then, the von Neumann criterion (31) provides a precise necessary and sufficient condition for stability [22].

If the eigenvalues are easily calculated, they provide a simple means for predicting the stability and behavior of the solution. *Von Neumann analysis*, estimation of the eigenvalues from the PDE itself, provides a way to extract information about the eigenvalues, if not the exact eigenvalues themselves.

For a two-level numerical scheme (13), the eigenvalues of the combined transformation matrix indicate the stability of the solution. Seeking a solution to the linear

difference scheme by separation of variables, as is used for linear PDEs, we can show that the discrete error growth factors are the eigenvalues of the method matrix M. Consider a two-level difference scheme (13) for a linear parabolic PDE so that $\mathbf{R} = 0$, then the eigenvalues can be defined by the constant ratio [22]

$$\frac{U_m^{n+1}}{U_m^n} = \frac{T^{n+1}}{T^n} = \lambda_m, \text{ where } U_m^n = X_m T^n. \tag{33}$$

The error $\epsilon_m^n = u(x_m, t_n) - U_m^n$ satisfies the same equation as the approximate solution U_m^n, so the eigenvalues also define the ratio of errors in time called the error growth (or amplification) factor [22]. Further, the error can be represented in Fourier form as $\epsilon_m^n = \hat{\epsilon} e^{\alpha n \Delta t} e^{im\beta \Delta x}$ where $\hat{\epsilon}$ is a Fourier coefficient, α is the growth/decay constant, $i = \sqrt{-1}$, and β is the wave number. Under this assumptions, the eigenvalues are equivalent to the error growth factors of the numerical method,

$$\lambda_k = \frac{U_m^{n+1}}{U_m^n} = \frac{\epsilon_m^{n+1}}{\epsilon_m^n} = \frac{\hat{\epsilon} e^{\alpha(n+1)\Delta t} e^{im\beta \Delta x}}{\hat{\epsilon} e^{\alpha n \Delta t} e^{im\beta \Delta x}} = e^{\alpha \Delta t}.$$

We can use this equivalency to finding bounds on the eigenvalues of a numerical scheme by plugging the representative growth factor $e^{\alpha \Delta t}$ into the method discretization called von Neumann stability analysis (also known as Fourier stability analysis) [22, 23]. The von Neumann criterion (31) ensures that the matrix stays bounded as it is powered up. If possible, C is set to 0, called the tight von Neumann criterion for simplified bounds on the step sizes. As another consequence of this equivalence, analyzing the spectrum of the transformation matrix also reveals patterns in the orientation, spread, and balance of the growth of errors for various wave modes.

Before we dive into the stability analysis, it is helpful to review some identities for reducing the error growth factors:

$$\begin{gathered} \frac{e^{ix} + e^{-ix}}{2} = \cos(x), \quad \frac{e^{ix} - e^{-ix}}{2} = i \sin(x), \\ \frac{1 - \cos(x)}{2} = \sin^2\left(\frac{x}{2}\right), \quad \frac{1 + \cos(x)}{2} = \cos^2\left(\frac{x}{2}\right). \end{gathered} \tag{34}$$

Example 7. Use the von Neumann stability analysis to determine conditions on $\Delta t, \Delta x$ to ensure stability of the Crank-Nicolson method (16) for the Test equation (6).

For von Neumann stability analysis, we replace each solution term U_m^n in the method with the representative error $\epsilon_m^n = \hat{\epsilon} e^{\alpha n \Delta t} e^{im\beta \Delta x}$,

$$\begin{aligned} U_m^{n+1} &= r U_{m-1}^n + (1 - 2r) U_m^n + r U_{m+1}^n, \\ \hat{\epsilon} e^{\alpha n \Delta t + \alpha \Delta t} e^{im\beta \Delta x} &= r \hat{\epsilon} e^{\alpha n \Delta t} e^{im\beta \Delta x - i\beta \Delta x} + (1 - 2r) \hat{\epsilon} e^{\alpha n \Delta t} e^{im\beta \Delta x} \\ &\quad + r \hat{\epsilon} e^{\alpha n \Delta t} e^{im\beta \Delta x + i\beta \Delta x}, \end{aligned} \tag{35}$$

where $r = \frac{\delta \Delta t}{\Delta x^2}$. Dividing through by the common ϵ_m^n term we can solve for the error growth factor

$$\begin{aligned} e^{\alpha \Delta t} &= 1 - 2r + 2r\left(\frac{e^{i\beta\Delta x} + e^{-i\beta\Delta x}}{2}\right), \\ &= 1 - 4r\left(\frac{1-\cos(\beta\Delta x)}{2}\right), \\ &= 1 - 4r\sin^2\left(\frac{\beta\Delta x}{2}\right), \end{aligned} \tag{36}$$

reduced using the identities in (34). Using a tight ($C = 0$) von Neumann criterion (31), we bound $|e^{\alpha\Delta t}| \leq 1$ with the error growth factor in place of the spectral radius. Since the error growth factor is real, the bound is ensured in the two components, $e^{\alpha\Delta t} \leq 1$ which holds trivially and $e^{\alpha\Delta t} \geq -1$ which is true at the extremum as long as $r \leq \frac{1}{2}$. Thus, as long as $dt \leq \frac{\Delta x^2}{2\delta}$, the Forward Euler method for the Test equation is stable in the sense of the tight von Neumann criterion which ensures diminishing errors at each step.

The balancing of explicit and implicit components in the Crank-Nicolson method create a much less restrictive stability condition.

Exercise 8. Use the von Neumann stability analysis to determine conditions on $\Delta t, \Delta x$ to ensure stability of the Crank-Nicolson method (16) for the Test equation (6).

Hint: verify that

$$e^{\alpha\Delta t} = \frac{1 - 2r\sin^2\left(\frac{\beta\Delta x}{2}\right)}{1 + 2r\sin^2\left(\frac{\beta\Delta x}{2}\right)}, \tag{37}$$

and show that both bounds are trivially true so that Crank-Nicolson method is unconditionally stable.

The default method in `PDE_Solution.m` and demonstrated in Figure 3 is the semi-implicit Crank-Nicolson method (17) for the Fisher-KPP equation (3). If you set $\rho = 0$ in `PDE_Analysis_Setup.m`, however, Crank-Nicolson method for the Test equation (6) is analyzed instead.

The two-level matrix form of the semi-implicit Crank-Nicolson method is

$$\begin{aligned} \mathbf{U}^{n+1} &= M\mathbf{U}^n + \mathbf{N}, \\ M &= \left(I - \frac{\Delta t}{2}D\right)^{-1}\left(I + \frac{\Delta t}{2}D + \rho\Delta t\,(1 - \mathbf{U}^n)\right), \\ \mathbf{N} &= \Delta t\left(I - \frac{\Delta t}{2}D\right)^{-1}\mathbf{B}. \end{aligned} \tag{38}$$

Notice in Figure 3 that all of the eigenvalues are real and they are all bounded between -1 and 1. Such a bound, $|\lambda| < 1$, ensures stability of the method based upon the default choice of $\Delta t, \Delta x$ step sizes.

Example 8. Use the von Neumann stability analysis to determine conditions on $\Delta t, \Delta x$ to ensure stability of the semi-implicit Crank-Nicolson method (17) for the Fisher-KPP equation (3).

Now we replace each solution term U_m^n in the semi-implicit Crank-Nicolson method (17) for the Fisher-KPP equation (3)

$$-\frac{r}{2}U_{m-1}^{n+1}+(1-r)\,U_m^{n+1}-\frac{r}{2}U_{m+1}^{n+1}=\frac{r}{2}U_{m-1}^{n}+(1-r+\rho(1-U_m^n))\,U_m^n+\frac{r}{2}U_{m+1}^{n}$$

with the representative error $\epsilon_m^n = \hat{\epsilon}e^{\alpha n\Delta t}e^{im\beta\Delta x}$ where again $r = \frac{\delta\Delta t}{\Delta x^2}$. Again dividing through by the common ϵ_m^n term we can solve for the error growth factor

$$e^{\alpha\Delta t}=\frac{1-2r\sin^2\left(\frac{\beta\Delta x}{2}\right)+\Delta t\rho(1-\tilde{U})}{1+2r\sin^2\left(\frac{\beta\Delta x}{2}\right)}$$

where constant $\tilde{U}$ represents the extreme values of U_m^n. Due to the equilibrium points $\bar{u} = 0, 1$ to be analyzed in Section 4.1, the bound $0 \le U_m^n \le 1$ holds as long as the initial condition is similarly bounded $0 \le U_m^0 \le 1$. Due to the potentially positive term $\Delta t\rho(1-\tilde{U})$, the tight ($C = 0$) von Neumann criterion fails, but the general von Neumann criterion (31), $|e^{\alpha\Delta t}| \le 1 + C\Delta t$, does hold for $C \ge \rho$. With this assumption, $e^{\alpha\Delta t} \ge -1 - C\Delta t$ is trivially true and $e^{\alpha\Delta t} \le 1 + C\Delta t$ results in

$$\left(\rho(1-\tilde{U})-C\right)\Delta x^2-4\delta\sin^2\left(\frac{\beta\Delta x}{2}\right)\le 2C\delta\Delta t\sin^2\left(\frac{\beta\Delta x}{2}\right)$$

which is satisfied at all extrema as long as $C \ge \rho$ since $\left(\rho(1-\tilde{U})-C\right) \le 0$. Thus, the semi-implicit Crank-Nicolson method is unconditionally stable in the sense of the general von Neumann criterion, which bounds the growth of error less than an exponential. This stability is not as strong as that of the Crank-Nicolson method for the Test equation but it provides useful management of the error.

3.4 Oscillatory Behavior

Oscillatory behavior has been exhaustively studied for ODEs [4, 9] with much numerical focus on researching ways to dampen oscillations in case they emerge [2, 19]. For those wishing to keep their methods oscillation-free, Theorem 5 provides sufficiency of non-oscillatory behavior through the non-negative eigenvalue condition (proven, for example, in [22]).

Theorem 5 (Non-negative Eigenvalue Condition). *A two-level difference scheme (13) is free of numerical oscillations if all the eigenvalues λ_i of the method matrix M are non-negative.*
Following von Neumann stability analysis from Section 3.3, we can use the error growth factors previously computed to determine the non-negative eigenvalue condition for a given method.

Example 9. To find the non-negative eigenvalue condition for the semi-implicit Crank-Nicolson method for the Fisher-KPP equation (3), we start by bounding our previously computed error growth factor as $e^{\alpha \Delta t} \geq 0$ to obtain

$$1 - 2r \sin^2\left(\frac{\beta \Delta x}{2}\right) + \Delta t \rho (1 - \tilde{U}) \geq 0$$

which is satisfied at the extrema, assuming $0 \leq \tilde{U} \leq 1$, by the condition

$$\frac{\delta \Delta t}{\Delta x^2} \leq \frac{1}{2} \tag{39}$$

which happens to be the same non-negative eigenvalue condition for the Crank-Nicolson method applied to the linear Test equation (6).

A numerical approach to track numerical oscillations uses a slight modification of the standard definitions for oscillatory behavior from ODE research to identify numerical oscillations in solutions to PDEs [15].

Definition 12. A continuous function $u(x,t)$ is oscillatory about K if the difference $u(x,t) - K$ has an infinite number of zeros for $a \leq t < \infty$ for any a. Alternately, a function is oscillatory over a finite interval if it has more than two critical points of the same kind (max, min, inflection points) in any finite interval $[a,b]$ [9].
Using the first derivative test, this requires two changes in the sign of the derivative. Using first order finite differences to approximate the derivative results in the following approach to track numerical oscillations.

Definition 13 (Numerical Oscillations). By tracking the sign change of the derivative for each spatial component through sequential steps t_{n-2}, t_{n-1}, t_n in time, oscillations in time can be determined by the logical evaluation

$$(U^{n-2} - U^{n-1})(U^{n-1} - U^n) < 0,$$

which returns true (inequality satisfied) if there is a step where the magnitude oscillates through a critical point. Catching two such critical points will define a numerical oscillation in the solution.

Crank-Nicolson method is known to be unconditionally stable, but damped oscillations have been found for large time steps. The point at which oscillations begin to occur is an open question, but it is known to be bounded from below by the

non-negative eigenvalue condition, which can be rewritten from (39) as $\Delta t \leq \frac{\Delta x^2}{2\delta}$. Breaking the non-negative eigenvalue condition, however, does not always create oscillations.

Challenge Problem 1. Using the example code found in Appendix 5.2, uncomment lines 12–15 (deleting %'s) and comment lines 7–9 (adding %'s) to use the semi-implicit Crank-Nicolson method. Make sure the default values of $\Delta x = 0.05, \rho = 1$ are set in `PDE_Analysis_Setup.m` and choose step size $\Delta t = 2$ in `PDE_Solution.m`. Notice how badly the non-negative eigenvalue condition $\Delta t \leq \frac{\Delta x^2}{2\delta}$ fails and run `PDE_Solution.m` to see stable oscillations in the solution. Run it again with smaller and smaller Δt values until the oscillations are no longer visible. Save this point as $(\Delta x, \Delta t)$. Change $\Delta x = 0.1$ and choose a large enough Δt to see oscillations and repeat the process to identify the lowest time step when oscillations are evident. Repeat this for $\Delta x = 0.5$ and $\Delta x = 1$, then plot all the $(\Delta x, \Delta t)$ points in MATLAB by typing `plot(dx,dt)` where `dx,dt` are vector coordinates of the $(\Delta x, \Delta t)$ points. On the Figure menu, click *Tools*,then *Basic Fitting*, and check *Show equations* and choose a type of plot which best fits the data. Write this as a relationship between Δt and Δx.

Oscillations in linear problems can be difficult to see, so it is best to catalyze any slight oscillations with oscillatory variation in the initial condition, or for a more extreme response, define the initial condition so that it fails to meet one or more boundary condition. Notice that the IBVP will no longer have a unique theoretical solution, but the numerical method will force an approximate solution to the PDE and match the conditions as best as it can. If oscillations are permitted by the method, then they will be clearly evident in this process.

Research Project 1. In `PDE_Analysis_Setup.m`, set `rho=0` on line 12 and multiply line 16 by zero to keep the same number of elements,

```
u0 = 0*polyval(polyfit(...
```

Investigate lowest Δt values when oscillations occur for $\Delta x = 0.05, 0.1, 0.5, 1$ and fit the points with the Basic Fitting used in Challenge Problem 1. Then, investigate a theoretical bound on the error growth factor (37) for the Crank-Nicolson method to the Test equation which approximates the fitting curve. It may be helpful to look for patterns in the computed eigenvalues at those $(\Delta x, \Delta t)$ points.

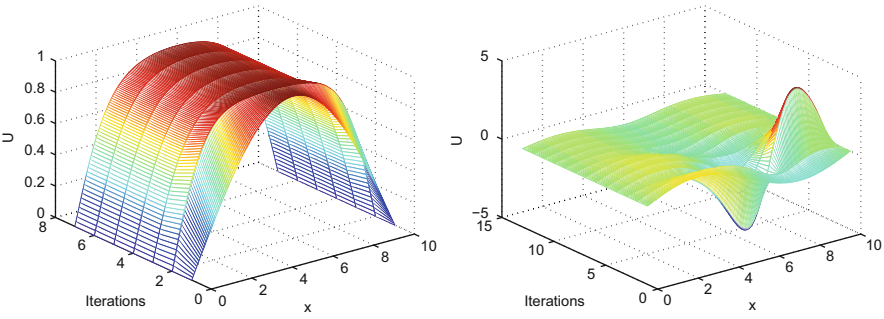


Fig. 4 Example graphs of Newton's method using the parameter values $a = 0, b = 0, L = 10, \delta = 1, \Delta x = \frac{1}{20}, \rho = 1, \text{degree} = 2, c = 1$ and replacing the default initial condition with (left) $\sin\left(\frac{\pi t}{L}\right)$ and (right) $\sin\left(\frac{2\pi t}{L}\right)$ in `PDE_Analysis_Setup.m` found in Appendix 5.3.

4 Parameter Analysis

Though parameters are held fixed when solving a PDE, varying their values can have an interesting impact upon the shape of the solution. We will focus on how changing parameters values and initial conditions affect the end behavior of IBVPs. For example, Figure 4 compares two very different steady-state solutions based upon similar sinusoidal initial conditions. Taking the limit of parameters in a given model can help us see between which limiting functions the steady-state solutions tend.

4.1 Steady-State Solutions

To consider steady-state solutions to the general reaction diffusion model (2), we set the time derivative to zero to create the boundary-value problem (BVP) with updated $u \equiv u(x)$ satisfying

$$\begin{aligned} 0 &= \delta u_{xx} + R(u), \\ u(0) &= a, \\ u(L) &= b. \end{aligned} \tag{40}$$

LeVeque [16], Marangell et. al. [10], and Aron et. al. [1] provide insightful examples and helpful guides for analyzing steady-state and traveling wave solutions of nonlinear PDEs. We will follow their guidelines here in our analysis of steady-states of the Fisher-KPP equation (3). Notice that as $\delta \to \infty$, equation (40) simplifies as $0 = u_{xx} + \frac{1}{\delta} R(u) \to 0 = u_{xx}$. Under this parameter limit, all straight lines are steady-states, but only one,

$$f_\infty(x) = \frac{b-a}{L} x + a, \tag{41}$$

fits the boundary conditions $u(0,t) = a,\ u(L,t) = b$. On the other hand, as $\delta \to 0$, solutions to equation (40) tend towards equilibrium points, $0 = R(\bar{u})$, of the reaction function inside the interval $(0, L)$. Along with the fixed boundary conditions, this creates discontinuous limiting functions

$$f_0^{(\bar{u})}(x) = \begin{cases} a,\ x = 0 \\ \bar{u},\ 0 < x < L, \\ b,\ x = L \end{cases} \tag{42}$$

defined for each equilibrium point $\bar{u}$.

Example 10. The Fisher-KPP equation (3) reduces to the BVP with updated $u \equiv u(x)$ satisfying

$$\begin{aligned} 0 &= \delta u_{xx} + \rho u(1-u), \\ u(0) &= 0, \\ u(10) &= 1, \end{aligned} \tag{43}$$

As parameter δ varies, the steady-state solutions of (43) transform from one of the discontinuous limiting functions

$$f_0^{(\bar{u})}(x) = \begin{cases} 0,\ x = 0 \\ \bar{u},\ 0 < x < 10, \\ 1,\ x = 10 \end{cases} \tag{44}$$

for equilibrium $\bar{u} = 0$ or $\bar{u} = 1$, towards the line

$$f_\infty(x) = \frac{1}{10}x. \tag{45}$$

4.2 Newton's Method

The Taylor series expansion for an analytical function towards a root, that is $0 = f(x_{n+1})$, gives us

$$0 = f(x_n) + \Delta x f'(x_n) + O\left(\Delta x^2\right) \tag{46}$$

Truncating $O\left(\Delta x^2\right)$ terms, and expanding $\Delta x = x_{n+1} - x_n$, an actual root of $f(x)$ can be approximated using Newton's method [3]. Newton's iteration method for a single variable function can be generalized to a vector of functions $\mathbf{G}(u)$ solving a (usually nonlinear) system of equations $0 = \mathbf{G}(u)$ resulting in a sequence of vector approximations $\{\mathbf{U}^{(k)}\}$. Starting near enough to a vector of roots which is stable, the sequence of approximations will converge to it: $\lim_{k\to\infty} \mathbf{U}^{(k)} = \mathbf{U}_s$.

Determining the intervals of convergence for a single variable Newton's method can be a challenge; even more so for this vector version. Note that the limiting vector of roots is itself a discretized version of a solution, $\mathbf{U}_s = u(\mathbf{x})$, to the BVP system (40).

Definition 14 (Vector Form of Newton's Method). For system of equations $0 = \mathbf{G}(u)$,

$$\mathbf{U}^{(k+1)} = \mathbf{U}^{(k)} - J^{-1}(\mathbf{U}^{(k)})\mathbf{G}(\mathbf{U}^{(k)}) \tag{47}$$

where $J(\mathbf{U}^{(k)})$ is the Jacobian matrix, $\frac{\partial \mathbf{G}_i(u)}{\partial U_j}$, which is the derivative of $\mathbf{G}(u)$ in $\mathbb{R}^{M\times M}$ where M is the number of components of $\mathbf{G}(\mathbf{U}^{(k)})$ [16].

Using the standard centered difference, we can discretize the nonlinear BVP (40) to obtain the system of equations $0 = \mathbf{G}\left(\mathbf{U}^n\right)$

$$0 = \delta D\mathbf{U}^n + \rho\mathbf{U}^n\left(1 - \mathbf{U}^n\right) \tag{48}$$

We need an initial guess for Newtons method, so we will use the initial condition u_0 from the IBVP (3). Table 3 shows the change in the solution measured by $||\mathbf{U}^{(k+1)} - \mathbf{U}^{(k)}||_\infty = ||J^{-1}\mathbf{G}||_\infty$ in each iteration. As expected, Newton's method appears to be converging quadratically, that is $\epsilon^{(k+1)} = O\left((\epsilon^{(k)})^2\right)$ according to Definition 15.

Definition 15. Given an iteration which converges, the *order of convergence N* is the power function relationship which bounds subsequent approximation errors as

$$\epsilon^{(k+1)} = O\left((\epsilon^{(k)})^N\right). \tag{49}$$

Note that scientific notation presents an effective display of these solution changes. You can see that the powers are essentially doubling every iteration in column 4 of Table 3 for the Approximate Error^a in the Fisher-KPP equation. This second order convergence, visually demonstrated in Figure 5, can be more carefully measured by computing

$$\text{order}^{(k)} = \text{round}\left(\frac{\log\left(\epsilon^{(k+1)}\right)}{\log\left(\epsilon^{(k)}\right)}\right)$$

where rounding takes into account the variable coefficient C in definition 15. For instance, values in column 4 slightly above 2 demonstrate $C < 1$ and those slightly below 2 demonstrate $C > 1$. Thus it is reasonable to round these to the nearest integer. This is not evident, however, for the Test equation because the error suddenly drops near the machine tolerance and further convergence is stymied by round-off error. If we start with a different initial guess $\mathbf{U}^{(0)}$ (but still close enough to this solution), we would find that the method still converges to this same solution.

Table 3 Verifying Convergence of Newton's Method

	Test Equation		**Fisher-KPP Equation**	
Iteration	Approximate Error[a]	Order of Convergence[b]	Approximate Error[a]	Order of Convergence[b]
1	1.6667e-01	18.2372	1.5421e+00	0 (-0.4712)
2	6.4393e-15	0.9956	8.1537e-01	-1 (-0.5866)
3	7.4385e-15	1.0585	1.1272e+00	-5 (-5.0250)
4	tol.[c] reached		5.4789e-01	2 (2.4533)
5			2.2853e-01	2 (2.2568)
6			3.5750e-02	2 (2.0317)
7			1.1499e-03	2 (2.0341)
8			1.0499e-06	2 (1.9878)
9			1.3032e-12	2 (1.2875)
10			tol.[c] reached	

[a] Approximate error measured as the maximum absolute difference ($||\cdot||_\infty$) between one iteration and the next.
[b] Order of convergence is measured as the power each error is raised to produce the next: $\epsilon_{i+1} = \epsilon_i^p \to p = \log(\epsilon_{i+1})/\log(\epsilon_i)$
[c] Stopping criterion is reached when error is less than tol = $10\epsilon = 2.2204e-15$.

Newton's method can be shown to converge if we start with an initial guess that is sufficiently close to a solution. How close depends on the nature of the problem. For more sensitive problems one might have to start extremely close. In such cases it may be necessary to use a technique such as continuation to find suitable initial data by varying a parameter, for example [16].

The solution found in Figure 6 for $\delta = 1$ is an isolated (or locally unique) solution in the sense that there are no other solutions very nearby. However, it does not follow that this is the unique solution to the BVP (43) as shown by the convergence in Figure 7 to another steady-state solution. In fact, this steady-state is unstable for the Fisher-KPP equation (3), as demonstrated in Figure 7.

Project Idea 3. For $\delta = 1$, use Newton's method in example code in Section 5.2 in the Appendix to investigate steady-state solutions to the bounded Fisher-KPP equation (43). Note the behavior of limiting solutions found in relation to the initial condition used. Also, note the shape of functions for which Newton's method is unstable. One example behavior for a specific quadratic is shown in Figure 6

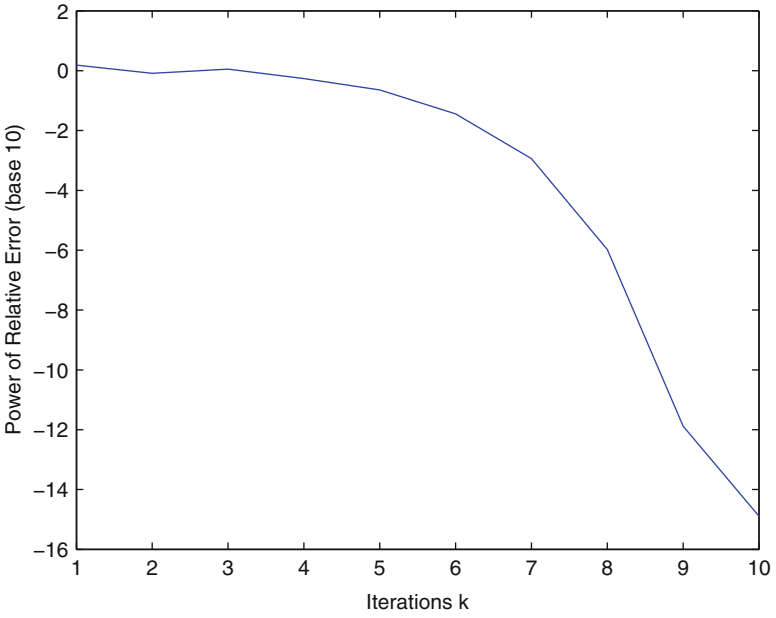


Fig. 5 Log-plot of approximate relative error, $\varepsilon(k)_r = \log_1 0(\max|\mathbf{U}(k+1) - \mathbf{U}(k)|)$ in Newton's method for the Fisher-KPP equation (3) using the default parameter values $a = 0, b = 1, L = 10, \delta = 1, \Delta x = 0.05, \rho = 1, \text{degree} = 2, c = \frac{1}{3}$ and initial condition as given in `PDE_Analysis_Setup.m` found in Appendix 5.3.

Project Idea 4. Find an initial condition which tends to a steady-state different than either $f_0(x)$ or $f_\infty(x)$. Investigate how the shape of the solution changes as $\delta \to 0$ and as $\delta \to \infty$. Does it converge to a limiting function at both ends?

Project Idea 5. Once you find an initial condition which tends to a steady-state different than either $f_0(x)$ or $f_\infty(x)$, run `PDE_Solution.m` in Appendix 5.2 to investigate the time stability of this steady-state using the built-in solver initialized by this steady-state perturbed by some small normally distributed noise.

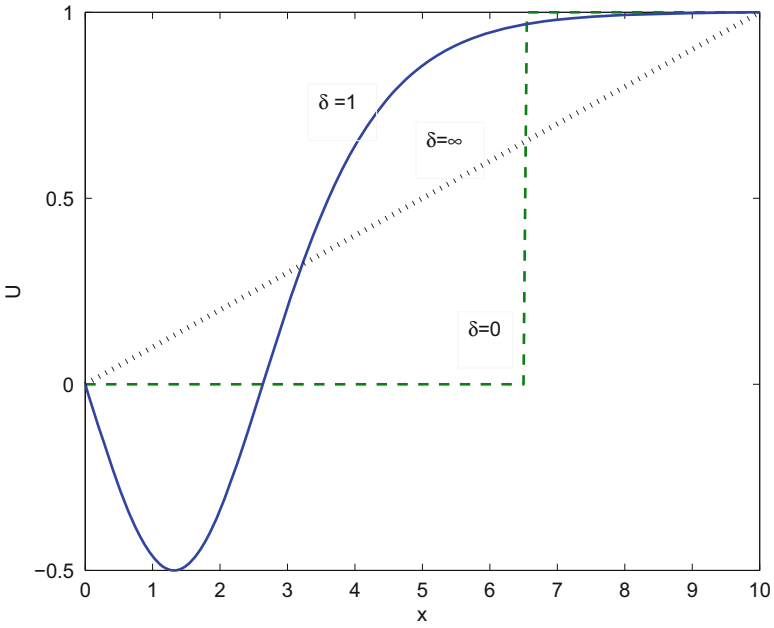


Fig. 6 Graph of three steady-state solutions to BVP (43) by Newton's method for a δ-parameter range of (1) $\delta = 0$, (2) $\delta = 1$, and (3) $\delta \to \infty$ using the other default parameter values $a = 0, b = 1, L = 10, \Delta x = \frac{1}{20}, \rho = 1, \text{degree} = 2, c = \frac{1}{3}$ and initial condition as given in `PDE_Analysis_Setup.m` found in Appendix 5.3.

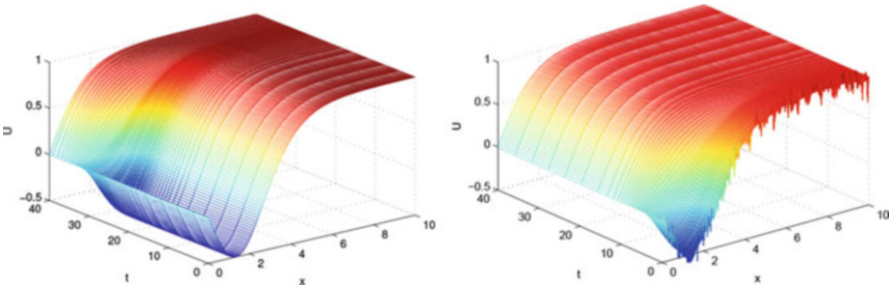


Fig. 7 Demonstration of the instability of the steady-state in Figure 6 for $\delta = 1$ as an initial condition for the Fisher-KPP equation (3) adding (a) no additional noise and (b) some normally distributed noise to slightly perturb the initial condition from the steady-state using the other default parameter values $a = 0, b = 1, L = 10, \Delta x = \frac{1}{20}, \rho = 1, \text{degree} = 2, c = \frac{1}{3}$ and first initial condition as given in `PDE_Analysis_Setup.m` found in Appendix 5.3.

Note, the `randn(i,j)` function in MATLAB is used to create noise in `PDE_Solution.m` using a vector of normally distributed psuedo-random variables with mean 0 and standard deviation 1.

Newton's method is called a local method since it converges to a stable solution only if it is near enough. More precisely, there is an open region around each solution called a basin, from which all initial functions behave the same (either all converging to the solution or all diverging) [3].

Project Idea 6. It is interesting to note that when $b = 0$ in the Fisher-KPP equation, δ-limiting functions coalesce, $f_\infty(x) = f_0(x) \equiv 0$, for $\bar{u} = 0$. Starting with $\delta = 1$, numerically verify that the steady-state behavior as $\delta \to \infty$ remains at zero. Though there are two equilibrium solutions as $\delta \to 0$, $\bar{u} = 0$ and $\bar{u} = 1$, one is part of a continuous limiting function qualitatively similar to $f_\infty(x)$ while the other is distinct from $f_\infty(x)$ and discontinuous. Numerically investigate the steady-state behavior as $\delta \to 0$ starting at $\delta = 1$. Estimate the intervals, called Newton's basins, which converge to each distinct steady-state shape or diverge entirely. Note, it is easiest to distinguish steady-state shapes by number of extremum. For example, smooth functions tending towards $f_0^{(1)}(x)$ have one extrema.

4.3 Traveling Wave Solutions

The previous analysis for finding steady-state solutions can also be used to find asymptotic traveling wave solutions to the initial value problem

$$\begin{aligned} u_t &= \delta u_{xx} + R(u), \\ u(x,0) &= u_0(x), \ -\infty < x < \infty, \ t \geq 0, \end{aligned} \tag{50}$$

by introducing a moving coordinate frame: $z = x - ct$, with speed c where $c > 0$ moves to the right. Note that by the chain rule for $u(z(x,t))$

$$\begin{aligned} u_t &= u_z z_t = -c u_z \\ u_x &= u_z z_x = u_z \\ u_{xx} &= (u_z)_z z_x = u_{zz}. \end{aligned} \tag{51}$$

Under this moving coordinate frame, equation (50) transforms into the boundary value problem

$$\begin{aligned} 0 &= \delta u_{zz} + c u_z + R(u), \\ -\infty &< z < \infty, \end{aligned} \tag{52}$$

Project Idea 7. Modify the example code `PDE_Analysis_Setup` to introduce a positive speed starting with $c = 2$ (updated after the initial condition is defined to avoid a conflict) and adding `+c/dx*spdiags(ones(M-2,1)*[-1 1 ],[-1 0], M-2, M-2)` to the line defining matrix D and update `BCs(1)` accordingly. Once you have implemented this transformation correctly, you can further investigate the effect the wave speed c has on the existence and stability of traveling waves as analyzed in [10]. Then apply this analysis to investigate traveling waves of the FitzHugh-Nagumo equation (53) as steady-states of the transformed equation.

5 Further Investigations

Now that we have some experience numerically investigating the Fisher-KPP equation, we can branch out to other relevant nonlinear PDEs.

Project Idea 8. Use MATLAB's `ode23s` solver to investigate asymptotic behavior of other relevant reaction-diffusion equations, such as the FitzHugh-Nagumo equation (53) which models the phase transition of chemical activation along neuron cells [8]. First investigate steady-state solutions and then transform the IBVP to investigate traveling waves. Which is more applicable to the model? A more complicated example is the Nonlinear Schrödinger equation (54), whose solution is a complex-valued nonlinear wave which models light propagation in fiber optic cable [15].

For the FitzHugh-Nagumo equation (53)

$$\begin{aligned} u_t &= \delta u_{xx} + \rho u(u-\alpha)(1-u),\ 0 < \alpha < 1 \\ u(x,0) &= u_0(x), \\ u(0,t) &= 0 \\ u(10,t) &= 1 \end{aligned} \tag{53}$$

For the Nonlinear Schrödinger (54), the boundary should not interfere or add to the propagating wave. One example boundary condition often used is setting the ends to zero for some large L representing ∞.

$$\begin{aligned} u_t &= i\delta u_{xx} - i\rho|u|^2 u, \\ u(x,0) &= u_0(x), \\ u(-L,t) &= 0 \\ u(L,t) &= 0 \end{aligned} \tag{54}$$

Additionally, once you have analyzed a relevant model PDE, it is helpful to compare end behavior of numerical solutions with feasible bounds on the physical quantities modeled. In modeling gene propagation, for example, with the Fisher-KPP equation (3), the values of u are amounts of saturation of the advantageous gene in a population. As such, it is feasible for $0 < u < 1$ as was used in the stability analysis. The steady-state solution shown in Figure 6, which we showed was unstable in Figure 7, does not stay bounded in the feasible region. This is an example where unstable solutions represent a non-physical solution. While investigating other PDEs, keep in mind which steady-states have feasible shapes and bounds. Also, if you have measurement data to compare to, consider which range of parameter values yield the closest looking behavior. This will help define a feasible region of the parameters.

Appendix

5.1 Proof of Method Accuracy

Numerical methods, such as those covered in Section 2.2, need to replace the underlying calculus of a PDE with basic arithmetic operations in a consistent manner (Definition 17) so that the numerical solution accurately approximates the true solution to the PDE. To determine if a numerical method accurately approximates the original PDE, all components are rewritten in terms of common function values using Taylor series expansions (Definition 16).

Definition 16. Taylor series are power series representations of functions at some point $x = x_0 + \Delta x$ using derivative information about the function at nearby point x_0.

$$f(x) = f(x_0) + f'(x_0)\Delta x + \frac{f''(x_0)}{2!}\Delta x^2 + \ldots + \frac{f^{(n)}(x_0)}{n!}\Delta x^n + O\left(\Delta x^{n+1}\right), \tag{55}$$

where the local truncation error in big-O notation, $\epsilon_L = O\left(\Delta x^{n+1}\right)$, means $||\epsilon_L|| \leq C\left(\Delta x^{n+1}\right)$ for some positive C [3].
For a function over a discrete variable $\mathbf{x} = (x_m)$, the series expansion can be rewritten in terms of indices $U_m \equiv U(x_m)$ as

$$U_{m+1} = U_m + U'_m \Delta x + \frac{U''_m}{2!}\Delta x^2 + \ldots + \frac{U^{(n)}_m}{n!}\Delta x^n + O\left(\Delta x^{n+1}\right), \tag{56}$$

Definition 17. A method is *consistent* if the difference between the PDE and the method goes to zero as all the step sizes, Δx, Δt for example, diminish to zero.
To quantify how accurate a consistent method is, we use the truncated terms from the Taylor series expansion to gauge the order of accuracy [3].

Definition 18. The *order of accuracy* in terms of an independent variable x is the lowest power p of the step size Δx corresponding to the largest term in the truncation error. Specifically,

$$\text{PDE} - \text{Method} = O\left(\Delta x^p\right). \tag{57}$$

Theorem 6. *If a numerical method for a PDE has orders of accuracy greater than or equal to one for all independent variables, then the numerical method is consistent.*

Proof. Given that the orders of accuracy $p_1, \ldots, p_k \geq 1$ for independent variables $x_1, \ldots, x_k$, then the truncation error

$$\text{PDE} - \text{Method} = O\left(\Delta x_1^{p_1}\right) + \cdots + O\left(\Delta x_k^{p_k}\right) \tag{58}$$

goes to zero as $\Delta x_1, \ldots, \Delta x_k \to 0$ since $\Delta x_i^{p_i} \to 0$ as Δx_i if $p_i \geq 1$.

Example 11. (Semi-Implicit Crank-Nicolson Method) The accuracy in space for the Semi-Implicit Crank-Nicolson method is defined by the discretizations of the spatial derivatives. Following example (refex:discretization), a centered difference will construct a second order accuracy in space.

$$\begin{aligned} &\frac{\delta}{\Delta x^2}\left(U^n_{m-1} - 2U^n_m + U^n_{m+1}\right) - \delta(U^n_m)_{xx} \\ &= \frac{\delta}{\Delta x^2}\left(\Delta x^2 (U^n_m)_{xx} + O\left(\Delta x^4\right)\right) - \delta(U^n_m)_{xx} \\ &= O\left(\Delta x^2\right). \end{aligned} \tag{59}$$

Then the discretized system of equations includes a tridiagonal matrix D with entries $D_{i,i-1} = \frac{\delta}{\Delta x^2}, D_{i,i} = -\frac{2\delta}{\Delta x^2}, D_{i,i+1} = \frac{\delta}{\Delta x^2}$ and a resultant vector $\mathbf{R}(\mathbf{U}^n)$ discretized from $\mathbf{R}(u)$. The order of accuracy in time can be found by the difference between the method and the spatially discretized ODE system. For the general reaction-diffusion equation (2),

$$\frac{\mathbf{U}^{n+1} - \mathbf{U}^n}{\Delta t} - \frac{1}{2}\left(D\mathbf{U}^n + D\mathbf{U}^{n+1} + \mathbf{R}(\mathbf{U}^n) + \mathbf{R}(\mathbf{U}^{n+1})\right) \tag{60}$$

$$
\begin{aligned}
&\quad - \left(\mathbf{U}_t^n - D\mathbf{U}^n - \mathbf{R}(\mathbf{U}^n)\right) \\
&= \left(\mathbf{U}_t^n + O(\Delta t)\right) - \frac{1}{2}\left(2D\mathbf{U}^n + O(\Delta t) + 2\mathbf{R}(\mathbf{U}^n) + O(\Delta \mathbf{U})\right) \\
&\quad - \left(\mathbf{U}_t^n - D\mathbf{U}^n - \mathbf{R}(\mathbf{U}^n)\right) \\
&= O(\Delta t)
\end{aligned}
$$

since $O(\Delta \mathbf{U}) = \mathbf{U}^{n+1} - \mathbf{U}^n = \Delta t \mathbf{U}_t^n + O(\Delta t^2)$.

Example 12. (Improved Euler Method) Similar to Example 11, the accuracy in space for the Improved Euler version of the Crank-Nicolson method (18) is defined by the same second order discretizations of the spatial derivatives. The order of accuracy in time can be found by the difference between the method and the spatially discretized ODE system. For the general reaction-diffusion equation (2),

$$
\begin{aligned}
&\frac{\mathbf{U}^{n+1} - \mathbf{U}^n}{\Delta t} - \frac{1}{2}\left(D\mathbf{U}^n + D\mathbf{U}^* + \mathbf{R}(\mathbf{U}^n) + \mathbf{R}(\mathbf{U}^*)\right) \qquad (61) \\
&\qquad - \left(\mathbf{U}_t^n - D\mathbf{U}^n - \mathbf{R}(\mathbf{U}^n)\right) \\
&= \left(\mathbf{U}_t^n + \frac{\Delta t}{2}\mathbf{U}_{tt}^n + O\left(\Delta t^2\right)\right) - \frac{1}{2}\left(2D\mathbf{U}^n + \Delta t\left(D^2\mathbf{U}^n + D\mathbf{U}^n\mathbf{R}(\mathbf{U}^n)\right)\right. \\
&\qquad \left. + O\left(\Delta t^2\right) + 2\mathbf{R}(\mathbf{U}^n) + \left(D\mathbf{U}^n + \mathbf{R}(\mathbf{U}^n)\right)\mathbf{R}_u(\mathbf{U}^n) + O\left(\Delta t^2\right)\right) \\
&\qquad - \left(\mathbf{U}_t^n - D\mathbf{U}^n - \mathbf{R}(\mathbf{U}^n)\right) \\
&= O\left(\Delta t^2\right)
\end{aligned}
$$

since

$$
\begin{aligned}
\mathbf{U}_{tt}^n &= D^2\mathbf{U}^n + D\mathbf{R}(\mathbf{U}^n) + \left(D\mathbf{U}^n + \mathbf{R}(\mathbf{U}^n)\right)\mathbf{R}_u(\mathbf{U}^n), \\
D\mathbf{U}^* &= D\mathbf{U}^n + \Delta t\left(D^2\mathbf{U}^n + D\mathbf{R}(\mathbf{U}^n)\right), \\
\mathbf{R}(\mathbf{U}^*) &= \mathbf{R}(\mathbf{U}^n) + \Delta t\left(D\mathbf{U}^n + \mathbf{R}(\mathbf{U}^n)\right)\mathbf{R}_u(\mathbf{U}^n) + O\left(\Delta t^2\right).
\end{aligned}
$$

5.2 Main Program for Graphing Numerical Solutions

```
%% PDE_Solution.m
% Numerical Solution with Chosen Solver
PDE_Analysis_Setup;                 % call setup script
tspan = [0 40];                     % set up solution time interval

% Comment when calling Crank-Nicolson method
[t,U] = ode23s(f, tspan, u0);       % Adaptive Rosenbrock Method
N = length(t);
dt = max(diff(t));

% Uncomment to call semi-implicit Crank-Nicolson method
% dt = 1;
% t = 0:dt:tspan(end);
% N = length(t);
```

```
% U = CrankNicolson_SI(dt,N,D,R,u0,BCs);

 U = [a*ones(N,1),U,b*ones(N,1)];    % Add boundary values
figure();                                % new figure
mesh(x,t,U);                             % plot surface as a mesh
title('Numerical Solution to PDE');

%% Analyze min/max step ratios and eigenvalues of numerical method matrix
ratio_range = [delta*min(diff(t))/dx^2, delta*max(diff(t))/dx^2],
% Example for semi-implicit CN method for Fisher-KPP and Test equation
Mat=(eye(size(D))-dt/2*D)\(eye(size(D))+dt/2*D+rho*dt*diag(1-U(end,2:end-1)));

eigenvalues = eig(Mat);
eig_real_range = [min(real(eigenvalues)),max(real(eigenvalues))],
spectral_radius = max(abs(eigenvalues)),
figure(); plot(real(eigenvalues),imag(eigenvalues),'*');
title('Eigenvalues');

%% Call Newton's Root finding method for finding steady state solution
[U_SS,err] = Newton_System(G,dG,u0);
U_SS = [a*ones(length(err),1),U_SS',b*ones(length(err),1)]; % Add boundary values
figure(); plot(x,U_SS(end,:));
title('Steady-State Solution by Newton's Method')
% figure(); mesh(x,1:length(err),U_SS);
% title('Convergence of Newton Iterations to Steady-State');
figure(); plot(log10(err)); %plot(-log10(err));
title('Log-Plot of Approximate Error in Newton's Method')
order = zeros(length(err));
for i=1:length(err)-1, order(i) = log10(err(i+1))/log10(err(i)); end

%% Investigate stability of Steady-State Solution by Normal Perturbation
noise = 0.1;
[t,U] = ode23s(f, tspan, U_SS(end,2:end-1) + noise*randn(1,M-2) );
N = length(t);
U = [a*ones(N,1),U,b*ones(N,1)];    % Add boundary values
figure(); mesh(x,t,U);
title('Stability of Steady-State Solution');
```

5.3 Support Program for Setting up Numerical Analysis of PDE

```
%% PDE_Analysis_Setup.m
% Setup Script for Numerical Analysis of Evolution PDEs in MATLAB
%  Example: u_t = delta*u_xx + rho*u(1-u), u(x,0) = u0, u(0,t)=a, u(L,t)=b

close all; clear all;   %close figures, delete variables
a = 0; b = 1;            % Dirichlet boundary conditions
L = 10;                  % length of interval
delta = 1;           % creating parameter for diffusion coefficient
dx = 0.05;               % step size in space
x=(0:dx:L)';             % discretized interval
M = length(x);           % number of nodes in space
rho = 1;                 % scalar for nonlinear reaction term

% Initial condition as polynomial fit through points (0,a),(L/2,c),(L,b)
degree=2; c=1/3;         % degree of Least-Squares fit polynomial
u0 = polyval(polyfit([0 L/2 L],[a c b],degree),x(2:end-1));

% Discretize in space with Finite-Differences: Boundary and Inner Equations
%  (a - 2U_2 + U_3)*(delta/dx^2) = rho*(-U_2 + U_2.^2),
%  (U_{i-1} - 2U_i + U_{i+1})*(delta/dx^2) = rho*(-U_{i} + (U_{i})^2)
%  (U_{M-2} - 2U_{M-1} + b)*(delta/dx^2) = rho*(-U_{M-1} + U_{M-1}.^2)
D = delta/dx^2*spdiags(ones(M-2,1)*[1 -2 1],[-1 0 1], M-2, M-2);
BCs = zeros(M-2,1); BCs(1) = a*delta/dx^2; BCs(end) = b*delta/dx^2;
```

```
% Anonymous functions for evaluting slope function components
f = @(t,u) D*u + BCs + rho*(u - u.^2);  % whole slope function for ode23s
R = @(u) rho*(u - u.^2);                % nonlinear component function
G = @(u) D*u + BCs + rho*(u - u.^2);    % rewritten slope function
dG = @(u) D + rho*diag(ones(M-2,1)-2*u);% Jacobian of slope function
```

5.4 Support Program for Running the Semi-Implicit Crank-Nicolson Method

```
%% CrankNicolson_SI.m
% Semi-Implicit Crank-Nicolson Method
% function [U]=CrankNicolson_SI(dt,N,D,R,u0,BCs)
%  Solving U_t = D*U + R(U) on Dirichlet boundaries vector BCs as
%  (I+dt/2*D)*U^(k+1) = (U^(k)+dt*(1/2*D*U+BCs+R(U^(k))) with time step dt,
%  time iterations N, matrix D, function R, and initial condition vector u0
function [U]=CrankNicolson_SI(dt,N,D,R,u0,BCs)
    sizeD = size(D,1);      % naming computation used twice
    U = zeros(sizeD,N);     % preallocating vector
    U(:,1) = u0(:);         % initializing with u0 as column vector
    for k = 1:N-1
        U(:,k+1) = (eye(sizeD)-dt/2*D)\...
            (U(:,k) + dt*(1/2*D*U(:,k) + BCs + R(U(:,k))));
    end
    U = U';                 % Transpose for plotting
end
```

5.5 Support Program for Verifying Method Accuracy

```
%% Method_Accuracy_Verification.m
% Script for running numerical method for multiple dt step sizes to verify
% order of accuracy
PDE_Analysis_Setup;             % call setup script
T = 10;
dt = 1; N = 1+T/dt;
Nruns = 10;
TestU = zeros(Nruns,1);
Difference = TestU; Change = TestU; SF = TestU;
midpt = ceil((M-1)/2);
%U = CrankNicolson_SI(dt,N,D,R,u0,BCs);         % Comment for ode23s
options = odeset('InitialStep',dt, 'MaxStep', dt,'AbsTol',max(u0));
[t,U] = ode23s(f, [0 T], u0, options);          % Comment for CN
Err = zeros(Nruns,1); SF = Err; Order = Err;    % Preallocate
fprintf(sprintf('App. Error\t Sig. Figs\t Order\n'));
for k = 1:Nruns-1
    dt=dt/2;  N = 1+T/dt;
    Uold = U;
% U = CrankNicolson_SI(dt,N,D,R,u0,BCs);      % Comment for ode23s
    options = odeset('InitialStep',dt, 'MaxStep', dt,'AbsTol',max(u0));
    [t,U] = ode23s(f, [0 T], u0, options);      % Comment for CN
Err(k+1) = norm(U(end,:)-Uold(end,:),Inf)/norm(U(end,:),Inf);
    SF(k+1) = floor(log10(.5/Err(k+1)));
Order(k) = log2(Err(k)/Err(k+1));
    fprintf(sprintf('%0.5g\t %d\t %0.5f\n', Err(k),SF(k),Order(k)));
end
```

5.6 Support Program for Running Newton's Method for ODE Systems

```
%% Newton's Method for Systems of Equations
% function [U,err] = Newton_System(G,dG,u0)
%  Compute vector U satisfying G(U)=0 with Jacobian dG(U) using iterations
%  U^(k+1) = U^(k) - dG(U^(k))\G(U^(k)), starting with initial guess u0,
%  and having approximate errors err
function [U,err] = Newton_System(G,dG,u0)
    tol = 10*eps;                  % tolerance relative to machine epsilon
    cutoff = 500;
    err = zeros(cutoff,1);         % Preallocation
    U = zeros(length(u0),length(err));
    U(:,1)=u0;
    for k=1:cutoff
       U(:,k+1) = U(:,k) - dG(U(:,k))\G(U(:,k));    % Newton Iteration
       err(k) = norm(U(:,k+1)-U(:,k),Inf);  % Estimate current error
       if err(k) < tol,            % stopping criterion
           err = err(1:k); U = U(:,1:k);    %truncate to current
           break;   % break from loop
       end
    end
end
```

References

1. Aron, M., Bowers, P., Byer, N., Decker, R., Demirkaya, A., Ryu, J.H.: Numerical results on existence and stability of steady state solutions for the reaction-diffusion and KleinGordon equations. Involve **7(6)**, 723–742 (2014)
2. Britz, D., Osterby, O., Strutwolf, J.: Damping of Crank–Nicolson error oscillations. Comput. Biol. Chem. **27(3)**, 253–263 (2003)
3. Burden, R.L., Faires, J.: Numerical Analysis, 9th edn. Brooks/Cole, Cengage Learning, Boston (2011)
4. Christodoulou, D.M., Graham-Eagle, J., Katatbeh, Q.D.: A program for predicting the intervals of oscillations in the solutions of ordinary second-order linear homogeneous differential equations. Adv. Differ. Equ. **1**, 1–23 (2016)
5. Crank, J., Nicolson, P.: A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type. Proc. Camb. Philos. Soc. **43(1)**, 50–67 (1947)
6. Evans, L.C.: Partial Differential Equations, 2nd edn. American Math Society, Providence, RI (2010)
7. Fisher, R.A.: The wave of advance of advantageous genes. Ann. Eugen. **7**, 353–369 (1937)
8. FitzHugh, R.: Impulses and physiological states in theoretical models of nerve membrane. Biophys. J. **1(6)** 445 (1961)
9. Gao, J., Song, F.: Oscillation analysis of numerical solutions for nonlinear delay differential equations of hematopoiesis with unimodel production rate. Appl. Math. Comput. **264**, 72–84 (2015)
10. Harley, K., Marangell, R., Pettet, G.J., Wechselberger, M.: Numerical computation of an Evans function in the Fisher/KPP equation. ArXiv: 1312.3685v3 (2015)
11. Harwood, R.C., Edwards, D.B., Manoranjan, V.S.: IEEE:Trans. on Energy Conversion. **26(4)**, 1109–1117 (2011)
12. Horn, R.A., Johnson, C.R.: Matrix Analysis, 2nd edn. Cambridge University Press, New York (2012)
13. Juhnke, S., Ratzkin, J.: A numerical investigation of level sets of extremal Sobolev functions. Involve. **8(5)**, 787–799 (2015)

14. Kolmogorov, A.: Selected Works of A.N. Kolmogorov I: a study of the diffusion equation with increase in the amount of substance, and its application to a biological problem, 2nd edn. Kluwer, Boston (1991)
15. Lakoba, T.I.:Instability of the finite-difference split-step method applied to the nonlinear Schrödinger equation. I. standing soliton. Numer. Methods Partial Differ. Equ. **32(3)**, 1002–1023 (2016)
16. LeVeque, R.J.: Finite Difference Methods for Ordinary and Partial Differential Equations. Society for Industrial Mathematics and Computation, Philadelphia, PN (2007)
17. Moler, C. B.: Numerical Computing with MATLAB: Revised Reprint. Society for Industrial Mathematics and Computation, Philadelphia, PN (2008)
18. Nadin, G., Perthame, B., Tang, M.: Can a traveling wave connect two unstable states? The case of the nonlocal Fisher equation. C.R. Math. **349**(9–10), 553–557 (2011)
19. Osterby, O.: Five ways of reducing the Crank-Nicolson oscillations. BIT Numer. Math. **43**(4), 811–822 (2003)
20. Sarra, S., Meador, C.: Stability properties of a predictor-corrector implementation of an implicit linear multistep method. Involve. **4(1)**, 43–51 (2011)
21. Scarborough, J.B.: Numerical Mathematical Analysis, 6th edn. Johns Hopkins Press, Baltimore (1966)
22. Thomas, J.W.: Numerical Partial Differential Equations, Finite Difference Methods, Texts in Applied Mathematics, vol. 22. Springer, Heidelberg (1995)
23. von Neumann, J., Richtmeyer, R.D.: A method for the numerical calculation of hydrodynamic shocks. J Appl. Phys. **21**, 232–237 (1950)

Index

A. Wootton et al. (eds.), *A Primer for Undergraduate Research*, Foundations for Undergraduate Research in Mathematics, DOI 10.1007/978-3-319-66065-3

D