Morphisms, Minors, and Minimal Obstructions to Convexity of Neural Codes

Robert Amzi Jeffs

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2021

Reading Committee:

Isabella Novik, Chair

Steven Klee

Rekha Thomas

Program Authorized to Offer Degree:
Mathematics

University of Washington

Abstract

Morphisms, Minors, and Minimal Obstructions to Convexity of Neural Codes

Robert Amzi Jeffs

Chair of the Supervisory Committee:
Robert R. & Elaine F. Phelps Professor Isabella Novik
Mathematics

We study open and closed convex codes from a geometric and combinatorial point of view. We prove constructive geometric results that establish new upper bounds on the open and closed embedding dimensions of intersection complete codes. We introduce a combinatorial framework of morphisms and minors for the study of convex codes, and show that open and closed embedding dimension are monotone invariants when codes are partially ordered by minors (in particular, open or closed convex codes form a minor-closed family). We establish new discrete geometry theorems and use them to exhibit infinite families of minimally non-convex codes, including new local obstructions to convexity. We also describe families of codes with novel embedding dimension behavior: arbitrary disparity between open and closed embedding dimension, open embedding dimensions that are exponential in the number of neurons in a code, and large increases in closed embedding dimension when adding a new non-maximal codeword. We conclude with an extensive discussion of open questions.

TABLE OF CONTENTS

Page

LIST OF FIGURES

SYMBOLS AND NOTATION

Symbols

Δ, Γ	Simplicial complexes
$\mathcal{A}, \mathcal{B}, \mathcal{C}, \ldots, \mathcal{S}, \mathcal{T}$	Codes
$\mathcal{U}, \mathcal{V}, \mathcal{W}$	Open realizations
$\mathcal{X}, \mathcal{Y}, \mathcal{Z}$	Closed realizations

Codes and Combinatorics

$[n]$	The set of neurons $\{1, 2, \ldots, n\}$
$[\overline{n}]$	The set of overlined neurons $\{\overline{1}, \overline{2}, \ldots, \overline{n}\}$
$\Delta(\mathcal{C})$	The simplicial complex of $\mathcal{C}$
$\mathrm{Tk}_{\mathcal{C}}(\sigma)$	The trunk of σ in $\mathcal{C}$
$\mathrm{nerve}(\mathcal{U})$	The nerve of $\mathcal{U}$
$\mathrm{code}(\mathcal{U})$	The code of $\mathcal{U}$
U_σ	The intersection $\bigcap_{i\in\sigma} U_i$
$\mathcal{U}^\sigma$	The atom of σ in $\mathcal{U}$
$\mathrm{odim}(\mathcal{C})$	The open embedding dimension of $\mathcal{C}$
$\mathrm{cdim}(\mathcal{C})$	The closed embedding dimension of $\mathcal{C}$
Code	The category of codes together with morphisms
$[\mathcal{C}]$	The isomorphism class of $\mathcal{C}$
$\mathcal{D} \leq \mathcal{C}$	$\mathcal{D}$ is a minor of $\mathcal{C}$
$\mathbf{P}_{\mathbf{Code}}$	The poset of code minors
$\mathcal{C}^{(i)}$	The i-th covered code of $\mathcal{C}$

Geometry and Topology

$\overline{pq}$	The line segment between points p and q
$\operatorname{conv} A$	The convex hull of A
$\operatorname{int} A$	The topological interior of A
$\operatorname{cl} A$	The topological closure of A
∂A	The topological boundary of A
$B_\varepsilon(p)$	The closed ball of radius ε centered at p
$H^>$ (resp. $H^<$)	The positive (resp. negative) side of an oriented hyperplane H
$\langle \sigma_1, \ldots, \sigma_k \rangle$	The simplicial complex generated by the faces $\sigma_1, \ldots, \sigma_k$
$\dim(\Delta)$	The dimension of a simplicial complex Δ
$\mathrm{Lk}_\Delta(\sigma)$	The link of σ in Δ
$\mathrm{St}_\Delta(\sigma)$	The closed star of σ in Δ

ACKNOWLEDGMENTS

I am deeply grateful to my advisor Isabella Novik for her support, advice, and mentorship during my graduate studies. She provided important concrete mathematical guidance, particularly when I was first exploring the framework of morphisms and minors. Our conversations and collaboration during the last few years have been formative (and transformative) to my development as a researcher. Very importantly, she also connected me with others in the discrete geometry community who in turn helped me further refine my ideas and pointed me to rich new avenues of study.

Mohamed Omar and Nora Youngs first introduced me to convex neural codes. Their enthusiasm and feedback during my undergraduate studies and the years following was a vital source of inspiration and encouragement. Regular exchanges with Caitlin Lienkaemper and Alex Kunin— fellow graduate students working on convex codes—sparked new ideas and excitement, and helped ground my decisions about which problems to pursue.

Florian Frick asked a number of incisive questions about my work during a visit to Carnegie Mellon University, prompting me to discover several of the key results in Chapter 5 of this text. Carina Curto likewise pointed me towards rich and fruitful approaches to my research at several times during my graduate studies.

Anne Shiu deserves special thanks for her detailed and energetic feedback on my preprints (and my research generally), and the work she has done to bring new students into the convex neural codes community. Both made my research feel all the more worthwhile.

Countless others shaped the mathematical content of this work through conversations, email exchanges, and questions at my talks. Among these, I am very grateful to Bruno Benedetti, Olivia Borghi, Federico Castillo, Vladimir Itskov, Zvi Rosen, Alex Ruys de Perez, Jose Samper, and Tuomas Tajakka.

I am very grateful to Steve Klee and Rekha Thomas for agreeing to be on my reading committee, and to Anna Karlin for serving as the graduate student representative on my committee.

In addition to her vital support and encouragement during the more typical years of my graduate work, I appreciate my partner Renata Kalnin for her patience with the past year of my research, which has been conducted almost exclusively from the slightly cramped bedroom of our apartment.

Finally, my family's enthusiasm for my mathematical endeavors continues to make me grateful and excited to continue my work.

Financially, my work was supported by the National Science Foundation through the grants DMS-1664865 and DGE-176112.

DEDICATION

For my fellow union members in UAW Local 4121, whose fighting spirit has made it possible to survive graduate school emotionally, whose hard-won gains in wages and healthcare have made it possible to survive graduate school financially, and whose organized power and solidarity have provided some of the most concrete and practical lessons of my graduate studies.

Chapter 1

INTRODUCTION

How can convex sets be arranged in d-dimensional Euclidean space? This general question captures a variety of more specific topics in discrete geometry, from Helly-type theorems, to hyperplane arrangements, to simplicial complexes, to polytopes. This work deals with the study of convex codes, a more recent topic under this umbrella, in which we wish to classify all of the combinatorial intersection and covering patterns that may arise from a finite collection of (usually open, sometimes closed) convex sets in $\mathbb{R}^d$.

The study of convex codes was initiated in 2013 [CIVCY13], and strictly generalizes the classical topic of d-representability of simplicial complexes. This relatively young area of research has already generated a sizeable body of work, including many striking families of examples, and results of possible general interest. An elegant classification of all convex codes seems intractable for the moment, but progress on the problem will require the development of novel and interesting tools in geometry and combinatorics. An *efficient* classification of convex codes is unfortunately out of the question: recent work in [KLR20] shows that recognizing convex codes is NP-hard (in fact, $\exists\mathbb{R}$-hard).

In this work we approach the study of convex codes from a geometric and combinatorial point of view. Our results come in three main categories. First, we provide constructive geometric results which establish new bounds on the "embedding dimensions" of certain families of codes. Second, we introduce and study a notion of "minors" for codes, providing a general framework in which to study convex codes. Third, we prove new discrete geometry results and apply them to generate many different families of codes which exhibit novel geometric behavior.

To set the stage, we begin by recalling some motivation for the study of convex codes, from both neuroscientific and mathematical perspectives (see Sections 1.1 and 1.2 respectively). In Section 1.3 we provide the basic definitions of convex codes, embedding dimensions, and other relevant terms. Section 1.4 gives a (necessarily incomplete) overview of the existing literature on convex codes. Finally, we outline the structure of subsequent chapters and summarize our main results in Section 1.5.

In this work we will assume familiarity with the fundamentals of Euclidean topology, discrete geometry, and combinatorics: homotopy type, partially ordered sets, simplicial complexes, hyperplanes, convex sets, polytopes, and so on. In many cases we provide a brief reminder or reference for a concept that we require (such as cyclic polytopes in Chapter 2). For any terms or results that we do not recall explicitly, two good references are [Mat02] and [Zie95].

1.1 Neuroscientific Motivation: Hippocampal Place Cells

In 1971 O'Keefe and Dostrovsky [OD71] made a groundbreaking observation: certain neurons in the hippocampus of rats were active primarily when the rat occupied a specific, approximately convex, region of its environment. We call the region in which a neuron is active its *receptive field*. These neurons can be thought of as encoding a "cognitive map" of the rat's environment, and were thus dubbed *place cells*. This phenomenon has been observed in other sensory perception tasks such as flight paths of bats [WYM18], sense of direction [KKM95], "grid cells" [MRM15], and visual recognition of angles [BYBOS95, WB74]. Non-visual stimuli such as smell also play a role in the behavior of place cells [ZMV15].

In each of these cases a stimulus space is covered by a collection of convex sets, and each set is "recognized" by a unique neuron in the brain. A key question in this context is the following: "how much topological and geometric structure of the stimulus space can place cells accurately recognize or encode?" As we will see in Section 1.3, convex codes provide a correspondence between the geometry of the environment and neural activity, and thus allow us to study this question mathematically.

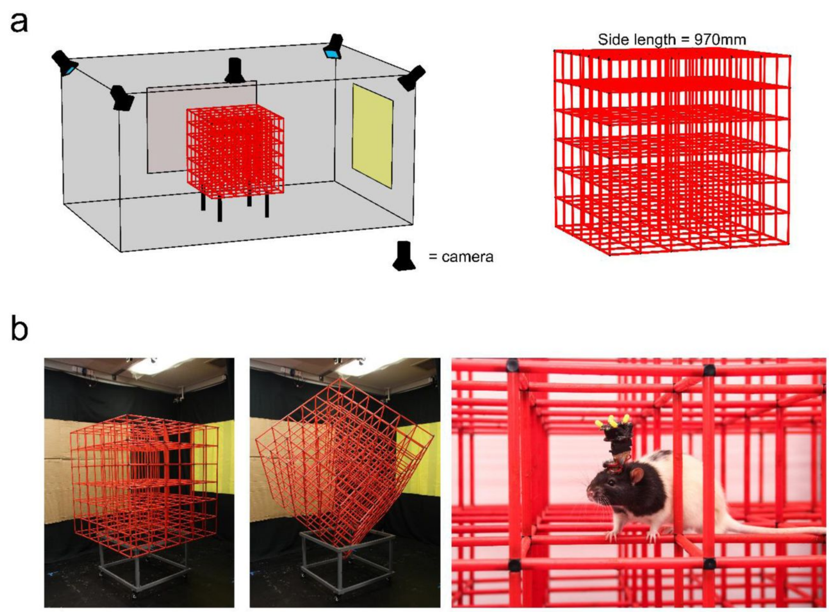


Figure 1.1: This figure appears in [GJAM+20], in which the authors study the activity of place cells in a rat as it moves through a 3-dimensional environment.

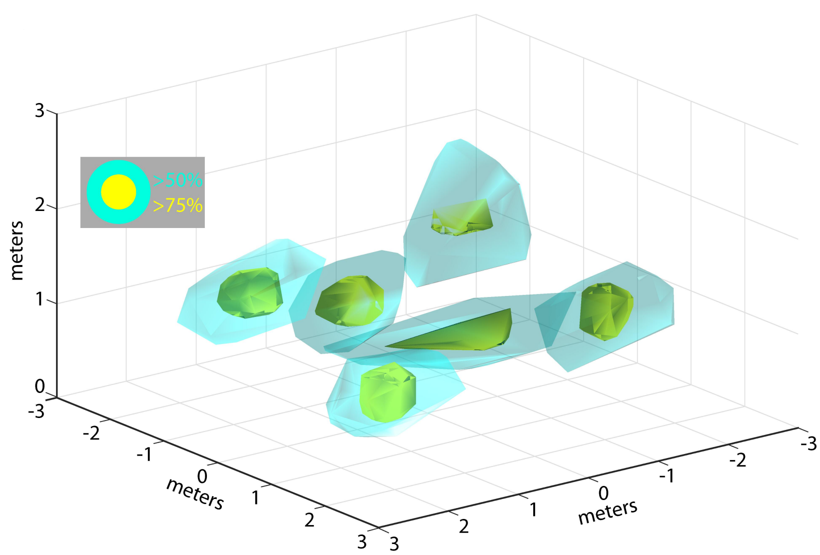


Figure 1.2: A figure from [WYM18] of 3-dimensional place fields for place cells in flying bats.

1.2 Mathematical Motivation: Nerve Complexes, d-Representability, and Helly-Type Theorems

The study of convex codes is also well-motivated from the perspective of discrete geometry. In this section we will highlight several classical areas of work in discrete geometry that are relevant to the study of convex codes, beginning with nerve complexes. Due to the neuroscientific motivation for studying convex codes, it is customary in the literature to refer to positive integers as "neurons" (corresponding to labeling a finite set of neurons in an experiment with integers 1 through n).

Definition 1.2.1. We will let $[n]$ denote the set $\{1, 2, \ldots, n\}$ consisting of the first n positive integers. We will often refer to elements of $[n]$ as *neurons*.

Despite the linguistic similarity, the term "nerve complex" is not a reference to place cells—indeed, the term significantly predates the discovery of place cells.

Definition 1.2.2. Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be a collection of open sets. The *nerve* of $\mathcal{U}$ is the simplicial complex

$$\text{nerve}(\mathcal{U}) := \left\{ \sigma \subseteq [n] \,\middle|\, \bigcap_{i\in\sigma} U_i \neq \emptyset \right\}.$$

One may view the nerve of $\mathcal{U}$ as a combinatorial object recording the "intersection information" of the collection $\mathcal{U}$. However, by virtue of being a simplicial complex, we may also regard nerve($\mathcal{U}$) from a topological and geometric perspective. To obtain useful information on this front, it often convenient to restrict our attention to collections $\mathcal{U}$ that are "topologically well-behaved." More formally, we are interested in good covers, defined below. Since we will often need to refer to regions of the form $\bigcap_{i\in\sigma} U_i$, we use the more concise notation $U_\sigma := \bigcap_{i\in\sigma} U_i$ from here onwards. We adopt the convention that the empty intersection $U_\emptyset$ is equal to the "ambient space" that $\mathcal{U}$ sits inside of (usually $\mathbb{R}^d$).

Definition 1.2.3. Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be a collection of open sets. We say that $\mathcal{U}$ is a *good cover* if U_σ is either contractible or empty for all nonempty $\sigma \subseteq [n]$.

Note that any collection of convex open sets forms a good cover. When we have a good cover $\mathcal{U}$, the nerve of $\mathcal{U}$ captures the homotopy type of the space that $\mathcal{U}$ covers. There are a variety of versions of this result, and consequently a variety of names for it, including "the nerve lemma" and "the nerve theorem." To our knowledge this type of result was first proved in [Bor48], and we are interested in the version below.

Theorem 1.2.4 (Borsuk's nerve lemma)**.** *Let* $\mathcal{U} = \{U_1, \ldots, U_n\}$ *be a good cover. Then* $\text{nerve}(\mathcal{U})$ *is homotopy equivalent to* $\bigcup_{i\in[n]} U_i$*.*

Example 1.2.5. Figure 1.3 shows a collection $\mathcal{U} = \{U_1, U_2, U_3, U_4\}$ of convex open sets in $\mathbb{R}^2$, and their nerve $\langle 123, 14, 24, 34\rangle$. One may confirm that the nerve is homotopy equivalent to the union of the sets in $\mathcal{U}$, as implied by the nerve lemma.

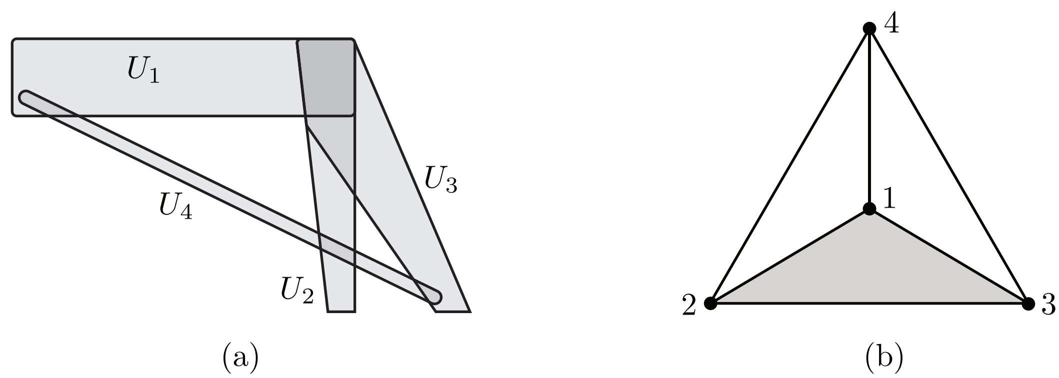


Figure 1.3: (a) A collection $\mathcal{U} = \{U_1, U_2, U_3, U_4\}$ of convex open sets in $\mathbb{R}^2$. (b) The nerve $\langle 123, 14, 24, 34\rangle$ of $\mathcal{U}$.

One may also study good covers consisting of closed sets, and obtain the corresponding closed version of the nerve lemma. Beyond homotopy type, nerves of good covers can also capture other geometric and topological features of an object, such as the dimension of the ambient space it lies in. This motivates the study of d-representable complexes.

Definition 1.2.6. Let $\Delta \subseteq 2^{[n]}$ be a simplicial complex. We say that Δ is *d-representable* if there exists a collection $\mathcal{U} = \{U_1, \ldots, U_n\}$ of convex (not necessarily open or closed) sets in $\mathbb{R}^d$ such that $\Delta = \text{nerve}(\mathcal{U})$.

Above, we allow for possibility that some U_i are the empty set. As a result, some of the "vertices" in $[n]$ may not actually be faces of the nerve complex. This differs from the convention that every simplicial complex must contain its vertices as faces, which is sometimes adopted in other works. However, this will not cause any ambiguities for us.

For a recent survey of d-representability (and nerves of collections of convex sets generally) see [Tan13]. One key result in this area of work is that every simplicial complex is d-representable for a large enough value of d. In particular, one can find a representation in a dimension no larger than a linear function of the dimension of the complex.

Theorem 1.2.7 ([Weg67], [Pm85])**.** *Let $\Delta \subseteq 2^{[n]}$ be a simplicial complex, and let $d = \dim(\Delta)$. Then Δ is $(2d+1)$-representable.*

The result above provides an upper bound on the dimension in which a simplicial complex is representable. Other results, such as Helly's theorem below, provide lower bounds. Lower bounds are particularly interesting when we are trying to infer information about a space that is covered by convex sets. One way to read Helly's theorem is that certain nerves cannot arise until we are in a high enough dimension. If we are given $\text{nerve}(\mathcal{U})$ without being told $\mathcal{U}$ itself, Helly's theorem can allow us to infer a lower bound on the dimension of the space in which $\mathcal{U}$ lies.

Theorem 1.2.8 (Helly's theorem)**.** *Suppose that $\Delta \subseteq 2^{[n]}$ is d-representable and $n \geq d+2$. If Δ contains all possible faces of dimension d, then $\Delta = 2^{[n]}$.*

Example 1.2.9. One may use Helly's theorem to verify that the complex $\langle 123, 14, 24, 34 \rangle$ from Example 1.2.5 is *not* 1-representable. Indeed, this complex contains all possible 1-dimensional faces (edges), but does not contain $[4]$. Thus $\mathbb{R}^2$ is the smallest dimension in which this complex has a representation.

Helly's theorem has many generalizations—"fractional" versions, "colorful" versions, and others (see [ADLS17] for a recent overview). Helly-type theorems can generally be interpreted as saying that certain nerves cannot arise in certain dimensions, i.e. certain complexes are not d-representable for certain values of d. However, these theorems are not enough to fully characterize d-representable complexes for a fixed value of d. In fact, even 2-representable complexes do not have an elegant characterization, and the computational problem of recognizing them is NP-hard (see [Tan13, Section 4.1]).

Despite the apparent intractability of fully characterizing d-representable complexes, their study has been fruitful and has motivated novel work in discrete and combinatorial geometry. The study of convex codes is a strictly more general task than the study of d-representable complexes. Where nerves capture only the "intersection information" of a collection $\mathcal{U}$, codes capture the "intersection and covering information" of $\mathcal{U}$. We will make this informal statement precise in the following section.

1.3 Definitions: Codes, Realizations, and Embedding Dimensions

So far we have explained how simplicial complexes are used in existing literature to capture intersection information from a collection $\mathcal{U} = \{U_1, \dots, U_n\}$. As mentioned above, we are interested in generalizing this framework to capture more geometric and topological information from $\mathcal{U}$. Correspondingly, we must work with more general combinatorial objects than simplicial complexes. Our combinatorial objects of interest will be codes.

Definition 1.3.1. A *code* (sometimes called a *combinatorial code* or *neural code*) is a subset of the power set $2^{[n]}$. Elements of a code are called *codewords*. The *weight* of a codeword is the number of neurons it contains (i.e. its cardinality). We will often partially order codewords by containment, and refer to *maximal codewords*, which are the codewords that are not properly contained in any other codeword.

We will adopt the convention that **every code contains the empty set as a codeword**. This convention is not universal in the convex neural code literature. In fact, many of the results that we will prove in this work first appeared in papers where we did not adopt this convention. For our work here, we believe this convention allows for cleaner and more accurate statements.

One small consequence of this convention is that when we speak of a "subcode of $\mathcal{C}$" we do not simply mean a subset of $\mathcal{C}$, we mean a subset of $\mathcal{C}$ that contains the empty set as an element.

Codes are extremely general combinatorial objects, and may seem an unwieldy tool at first. It is often helpful to return to the more familiar terrain of simplicial complexes, and to this end we associate every code to a unique simplicial complex.

Definition 1.3.2. Let $\mathcal{C} \subseteq 2^{[n]}$ be a code. The *simplicial complex* of $\mathcal{C}$ is

$$\Delta(\mathcal{C}) := \{\sigma \subseteq [n] \mid \sigma \text{ is contained in some codeword } c \text{ of } \mathcal{C}\}.$$

Equivalently, $\Delta(\mathcal{C})$ is the smallest simplicial complex containing $\mathcal{C}$.

To further grapple with the combinatorial features of codes, we often refer to "trunks" in a code. These generalize open stars in a simplicial complex, and will play a prominent role in Chapters 3 and 4.

Definition 1.3.3. Let $\mathcal{C} \subseteq 2^{[n]}$ be a code and let $\sigma \subseteq [n]$. The *trunk* of σ in $\mathcal{C}$ is the set

$$\mathrm{Tk}_{\mathcal{C}}(\sigma) := \{c \in \mathcal{C} \mid \sigma \subseteq c\}.$$

A subset of $\mathcal{C}$ is called a *trunk in* $\mathcal{C}$ if it is empty, or equal to $\mathrm{Tk}_{\mathcal{C}}(\sigma)$ for some nonempty $\sigma \subseteq [n]$. When the code $\mathcal{C}$ is clear from context, we will simply write $\mathrm{Tk}(\sigma)$.

Definition 1.3.4. Trunks of the form $\mathrm{Tk}(\{i\})$ will be called *simple* trunks, and denoted $\mathrm{Tk}(i)$.

Example 1.3.5. Consider the code

$$\mathcal{C} = \{\mathbf{123}, \mathbf{14}, \mathbf{24}, \mathbf{34}, 23, 1, 2, 3, 4, \emptyset\}.$$

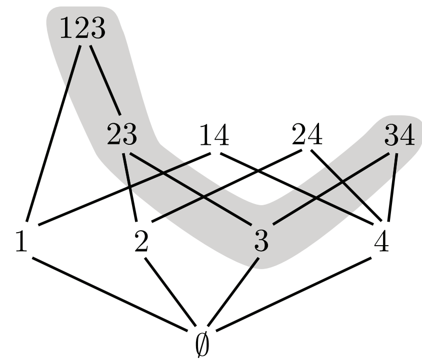


Figure 1.4: The code $\mathcal{C} = \{\mathbf{123}, \mathbf{14}, \mathbf{24}, \mathbf{34}, 23, 1, 2, 3, 4, \emptyset\}$ as a partially ordered set, with $\mathrm{Tk}_\mathcal{C}(3)$ outlined in grey.

The code $\mathcal{C}$ has four neurons and ten codewords. Figure 1.4 shows the Hasse diagram of $\mathcal{C}$, with the simple trunk $\mathrm{Tk}_\mathcal{C}(3)$ highlighted in grey. Above, we have removed brackets and commas from codewords to simplify our notation. For example, 23 refers to the codeword $\{2, 3\}$. We have also bolded all maximal codewords. We will use both of these conventions throughout our work.

Just as we used simplicial complexes to record intersection information about a collection of sets via nerve complexes, we will use codes to capture more general information about $\mathcal{U}$.

Definition 1.3.6. Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be a collection of (not necessarily open, closed, or convex) subsets of a set X. The *code* of $\mathcal{U}$ is the code

$$\mathrm{code}(\mathcal{U}) := \left\{ \sigma \subseteq [n] \,\middle|\, U_\sigma \setminus \bigcup_{j \in [n]\setminus\sigma} U_j \neq \emptyset \right\}$$

on n neurons. Equivalently,

$$\mathrm{code}(\mathcal{U}) := \big\{ \sigma \subseteq [n] \,\big|\, \text{There exists } p \in X \text{ with } p \in U_i \text{ if and only if } i \in \sigma \big\}.$$

We say that $\mathcal{U}$ is a *realization* of $\mathrm{code}(\mathcal{U})$.

Note that $\text{nerve}(\mathcal{U}) = \Delta(\text{code}(\mathcal{U}))$, and so codes generalize nerve complexes. Moreover, observe that $\sigma \notin \text{code}(\mathcal{U})$ if and only if the region U_σ is covered by $\{U_j \mid j \in [n] \setminus \sigma\}$. This is the sense in which codes capture "covering information" about a collection $\mathcal{U}$.

We are primarily interested in the study of open convex realizations $\mathcal{U}$ sitting inside the space $X = \mathbb{R}^d$. From here onwards, the word "realization" will mean "convex realization." When we speak of realizations that are not necessarily convex, we will specify this explicitly.

There are some ambiguities in Definition 1.3.6 that we should briefly resolve. First, observe that when speaking of $\text{code}(\mathcal{U})$ we are not regarding $\mathcal{U}$ as a multiset, but as a list. Although reordering $\mathcal{U}$ does not have a meaningful impact on the resulting code (we are simply permuting the neurons) the ordering is important for bookkeeping purposes, especially in some of our more involved proofs. Although it clashes with combinatorial standards, we use the word "collection" when referring to realizations to emphasize that they are ordered lists of objects sitting inside of an ambient space.

A second ambiguity in Definition 1.3.6 is that we are assuming every code contains $\emptyset$ as a codeword, but $\text{code}(\mathcal{U})$ contains $\emptyset$ if and only if $\mathcal{U}$ does not fully cover the space X. Thus we will only ever work with collections $\mathcal{U}$ that do *not* cover the ambient space X. In the study of convex codes this is not a problem: we have $X = \mathbb{R}^d$, and we may intersect our realization $\mathcal{U}$ with a large ball without changing the resulting code (except by possibly adding the empty codeword). In other words, we may without loss of generality restrict our attention to bounded realizations (and we will often do so).

One way to view codes is the following. A collection $\mathcal{U}$ carves the space X into a number of regions, and $\text{code}(\mathcal{U})$ records these regions (though not their exact locations). Formally, we call these regions "atoms" of the realization $\mathcal{U}$.

Definition 1.3.7. Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be a collection of (not necessarily open, closed, or convex) subsets of a set X, and let $\sigma \subseteq [n]$. The *atom* of σ in $\mathcal{U}$ is the set

$$\mathcal{U}^\sigma := U_\sigma \setminus \bigcup_{j \in [n] \setminus \sigma} U_j.$$

Equivalently,

$$\mathcal{U}^\sigma := \{p \in X \mid p \in U_i \text{ if and only if } i \in \sigma\}.$$

Note that with this notation we have $\text{code}(\mathcal{U}) = \{\sigma \subseteq [n] \mid \mathcal{U}^\sigma \neq \emptyset\}$. In other words, $\text{code}(\mathcal{U})$ tells us precisely which atoms in $\mathcal{U}$ are nonempty.

In the existing convex codes literature, the notation for atoms is typically different than what we have presented above. For example, [CGIK16] uses the notation $A_\sigma^{\mathcal{U}}$. We prefer $\mathcal{U}^\sigma$ to existing conventions for its compactness and small number of symbols.

Example 1.3.8. Figure 1.5 shows an open convex realization $\mathcal{U} = \{U_1, U_2, U_3, U_4\}$ of the code $\mathcal{C} = \{\mathbf{123}, \mathbf{14}, \mathbf{24}, \mathbf{34}, 23, 1, 2, 3, 4, \emptyset\}$ in $\mathbb{R}^2$, and one of the atoms in this realization.

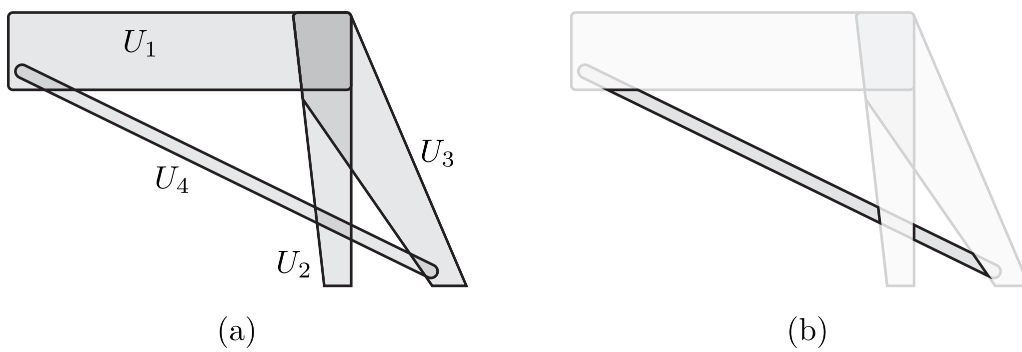


Figure 1.5: (a) An open convex realization $\mathcal{U} = \{U_1, U_2, U_3, U_4\}$ of the code $\mathcal{C} = \{\mathbf{123}, \mathbf{14}, \mathbf{24}, \mathbf{34}, 23, 1, 2, 3, 4, \emptyset\}$. (b) The atom $\mathcal{U}^4$ in this realization.

Our primary question of interest is to classify which codes arise from collections of convex (open or closed) sets in $\mathbb{R}^d$. Note that this generalizes the study of d-representability: A complex Δ is d-representable if and only if there exists a code $\mathcal{C}$ such that $\mathcal{C}$ has a realization in $\mathbb{R}^d$ and $\Delta = \Delta(\mathcal{C})$.

From the perspective of neuroscience, we have the following situation. Given a collection of n place cells with receptive fields $\mathcal{U} = \{U_1, \ldots, U_n\}$ in $\mathbb{R}^d$, $\text{code}(\mathcal{U})$ records which neurons fire concurrently as an animal moves through space. Thus code($\mathcal{U}$) records exactly the neural activity of an animal as it moves through the receptive fields in $\mathcal{U}$. Classifying the spaces in which we can realize code($\mathcal{U}$) thus amounts to answering the question we posed in Section 1.1: "how much topological and geometric structure of the stimulus space can place cells accurately recognize?"

Note that if $\mathcal{C}$ has a realization in $\mathbb{R}^d$, then $\mathcal{C}$ has a realization in any dimension larger than d by taking cylinders over our sets. Thus we are most interested in the minimal dimension in which a code has a realization.

Definition 1.3.9. The *open embedding dimension* of a code $\mathcal{C}$, denoted odim($\mathcal{C}$), is the smallest dimension in which $\mathcal{C}$ has an open convex realization (or ∞ if no such realization exists). That is,

$$\operatorname{odim}(\mathcal{C}) := \min\left(\{d \mid \mathcal{C} \text{ has an open convex realization in } \mathbb{R}^d\} \cup \{\infty\}\right).$$

The *closed embedding dimension* of $\mathcal{C}$, denoted cdim($\mathcal{C}$), is defined similarly using closed convex realizations. That is,

$$\operatorname{cdim}(\mathcal{C}) := \min\left(\{d \mid \mathcal{C} \text{ has a closed convex realization in } \mathbb{R}^d\} \cup \{\infty\}\right).$$

Definition 1.3.10. Let $\mathcal{C}$ be a code. If $\operatorname{odim}(\mathcal{C}) < \infty$ we say that $\mathcal{C}$ is an *open convex* code. Likewise if $\operatorname{cdim}(\mathcal{C}) < \infty$ we say that $\mathcal{C}$ is a *closed convex* code.

When we introduced d-representability we did not specify that our sets were open or closed. Indeed, the study of d-representability does not change when we restrict from arbitrary convex sets to open convex sets, or closed convex sets. There is no a priori reason to believe that the same should not be true in the study of codes. However, there are many subtle and interesting differences between open convex codes and closed convex codes. As one example, Theorem 5.2.2 and Proposition 5.2.3 will show that there may be an arbitrarily large gap between the open and closed embedding dimensions of a code.

Example 1.3.11. Consider the code $\mathcal{C} = \{\mathbf{123}, 13, 23, 1, 3, \emptyset\}$. This code has an open realization in $\mathbb{R}^2$, as shown in Figure 1.6(a). However, this realization can be flattened to realize the same code in $\mathbb{R}^1$. Thus $\text{odim}(\mathcal{C}) = 1$ in this case (note that $\text{odim}(\mathcal{C}) \neq 0$ since $\mathcal{C}$ has more than one codeword). In fact, all of the realizations shown in Figure 1.6 could be regarded as closed realizations, so we have $\text{cdim}(\mathcal{C}) = 1$ as well.

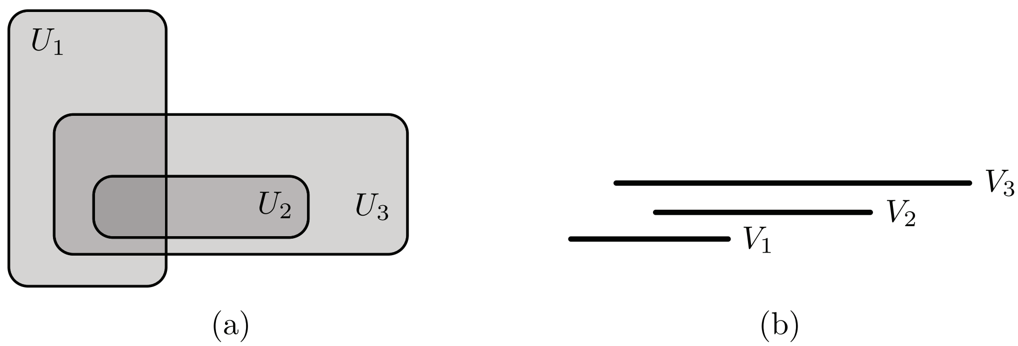


Figure 1.6: (a) An open realization $\mathcal{U} = \{U_1, U_2, U_3\}$ of $\mathcal{C} = \{\mathbf{123}, 13, 23, 1, 3, \emptyset\}$ in the plane. (b) The flattened realization $\mathcal{V} = \{V_1, V_2, V_3\}$ of $\mathcal{C}$ in $\mathbb{R}^1$ (intervals shown with horizontal separation for clarity).

From the perspective of neuroscience, our main interest lies in the study of open convex codes, since receptive fields are full-dimensional and somewhat noisy. Closed convex codes are a natural class to study mathematically, as they have connections to polytopal complexes, hyperplane arrangements, and other discrete geometric notions. In the next section we will provide a brief overview of existing results in the study of open and closed convex codes, on which our own work is built.

1.4 Past Results

While every simplicial complex is d-representable for a large enough value of d, not every code is (open or closed) convex, even if we search for realizations in arbitrarily large dimension. Thus the task of classifying open convex codes (i.e. codes with finite open embedding dimension) is a nontrivial task. A first step in this direction was the study of "local obstructions," defined below. These obstructions were originally observed in [GI14], but we adopt the notation and terminology of later the later work [CGJ$^+$17].

In the following definition we reference collapsibility, a combinatorial notion that is slightly more restrictive than contractibility. See Definition 6.1.3 and Example 6.1.4 in Chapter 6 for more on collapsibility.

Definition 1.4.1 ([CGJ$^+$17], [CFS19]). Let $\mathcal{C} \subseteq 2^{[n]}$ be a code, and let $\sigma \in \Delta(\mathcal{C}) \setminus \mathcal{C}$. If $\mathrm{Lk}_{\Delta(\mathcal{C})}(\sigma)$ is non-contractible, we say that $\mathcal{C}$ has a *local obstruction* at σ. If $\mathrm{Lk}_{\Delta(\mathcal{C})}(\sigma)$ is non-collapsible, we say that $\mathcal{C}$ has a *local obstruction of the second kind* at σ. If $\mathcal{C}$ has no local obstructions we say that $\mathcal{C}$ is *locally good*, and if $\mathcal{C}$ has no local obstructions of the second kind we say that $\mathcal{C}$ is *locally great.*

The word "local" here refers to the fact that these obstructions are defined using links in $\Delta(\mathcal{C})$, and links capture the local geometry of a complex near a face. As one would hope, local obstructions help us recognize when a code is *not* open or closed convex. The following theorems capture this.

Theorem 1.4.2 ([CGJ$^+$17]). *Let $\mathcal{C}$ be a code, and suppose that* $\mathrm{odim}(\mathcal{C}) < \infty$ *or* $\mathrm{cdim}(\mathcal{C}) < \infty$. *Then $\mathcal{C}$ is locally good.*

Theorem 1.4.3 ([CFS19]). *Let $\mathcal{C}$ be a code, and suppose that* $\mathrm{odim}(\mathcal{C}) < \infty$ *or* $\mathrm{cdim}(\mathcal{C}) < \infty$. *Then $\mathcal{C}$ is locally great.*

Since every collapsible complex is also contractible, every local obstruction is also a local obstruction of the second kind. Thus Theorem 1.4.3 implies Theorem 1.4.2. It turns out that local obstructions can be characterized topologically, as formalized in the theorem below. Currently, there is not an analogous "geometric characterization" of locally great codes.

Theorem 1.4.4 ([CFS19]). *A code is locally good if and only if it can be realized using a good cover.*

Good covers are more general than convex covers, and so we should expect that local obstructions are not sufficient to characterize either open or closed convex codes. Indeed, there is a rich and interesting theory of non-local obstructions to convexity, which we will build on. The first non-local obstruction to open convexity was published in [LSW17].

Theorem 1.4.5 ([LSW17])**.** *The code*

$$\mathcal{C} = \{\mathbf{2345}, \mathbf{123}, \mathbf{134}, \mathbf{145}, 13, 14, 23, 34, 45, 3, 4, \emptyset\}$$

has $\operatorname{cdim}(\mathcal{C}) = 2$ *and* $\operatorname{odim}(\mathcal{C}) = \infty$*. In particular, this code is locally great, but not open convex.*

Theorem 1.4.6 ([CGIK16])**.** *The code*

$$\mathcal{D} = \{\mathbf{123}, \mathbf{126}, \mathbf{156}, \mathbf{234}, \mathbf{345}, \mathbf{456}, 12, 16, 23, 34, 45, 56, \emptyset\}$$

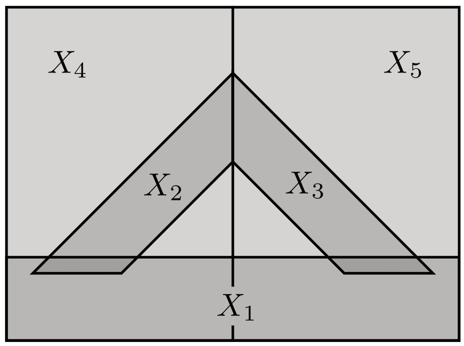
has $\operatorname{odim}(\mathcal{D}) = 2$ *and* $\operatorname{cdim}(\mathcal{D}) = \infty$*. In particular, this code is locally great, but not closed convex.*

The "problem" with the codes $\mathcal{C}$ and $\mathcal{D}$ in Theorems 1.4.5 and 1.4.6 can be seen in Figure 1.7. The closed realization of $\mathcal{C}$ forces the atom of 2345 to have empty interior, so we may not replace our sets with their interiors without changing the resulting code. Similarly, the open realization of $\mathcal{D}$ forces disjoint sets to share boundary points, so that replacing sets with their closures would yield a different code. These pathologies motivated [CGIK16] to define "non-degenerate" realizations.

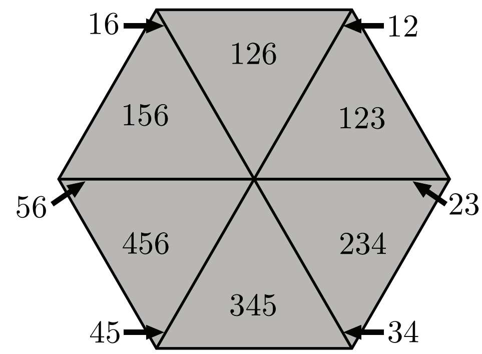
Definition 1.4.7 ([CGIK16])**.** A collection $\mathcal{U} = \{U_1, \dots, U_n\}$ of convex (but not necessarily open) sets in $\mathbb{R}^d$ is called *non-degenerate* if the following two conditions hold:

(i) For all $c \in \operatorname{code}(\mathcal{U})$, the atom $\mathcal{U}^c$ is top-dimensional (i.e. its intersection with any open set is either empty, or has nonempty interior).

(ii) For all nonempty $\sigma \subseteq [n]$, we have $\bigcap_{i\in\sigma} \partial U_i \subseteq \partial U_\sigma$.



(a)



(b)

Figure 1.7: (a) A closed realization $\mathcal{X} = \{X_1, X_2, X_3, X_4, X_5\}$ in $\mathbb{R}^2$ of the code $\mathcal{C}$ from Theorem 1.4.5. (b) An open realization in $\mathbb{R}^2$ of the code $\mathcal{D}$ from Theorem 1.4.6. Rather than label each set, we have labeled the atoms in the realization. For example, U_1 is the top half of the hexagon, and subsequent U_i are rotations of U_1 by multiples of 60 degrees about the center of the hexagon. See also [CGIK16, Figures 2.1 and 2.2].

Such realizations have the property that if we simultaneously replace all sets by their interiors or closures, we do not change the resulting code. In fact, recent work in [CJL$^+$20] shows that non-degenerate realizations are *exactly* the realizations with this property.

All the results we have discussed so far are "negative," in the sense that they describe certain obstructions to finding convex realizations of codes. Constructive or "positive" results have been harder to come by, but some progress has been made for special classes of codes. One such class is max-intersection complete codes.

Definition 1.4.8. Let $\mathcal{C}$ be a code. We say that $\mathcal{C}$ is *intersection complete* if the intersection of any two codewords in $\mathcal{C}$ is again a codeword in $\mathcal{C}$. We say that $\mathcal{C}$ is *max-intersection complete* if the intersection of any number of maximal codewords in $\mathcal{C}$ is a codeword in $\mathcal{C}$. The *(max-)intersection completion* of a code is the smallest (max-)intersection complete code that contains it. The intersection completion of $\mathcal{C}$ is denoted $\widehat{\mathcal{C}}$.

Max-intersection complete codes are especially well-behaved from the perspective of convexity. Not only are they both open and closed convex, but one may construct non-degenerate realizations in a sufficiently large dimension.

Theorem 1.4.9 ([CGIK16])**.** *Let $\mathcal{C}$ be a max-intersection complete code with m maximal codewords, and let $d = \max\{2, m-1\}$. Then $\mathcal{C}$ has an (open or closed) non-degenerate convex realization in $\mathbb{R}^d$. In particular,* $\operatorname{odim}(\mathcal{C}) \leq d$ *and* $\operatorname{cdim}(\mathcal{C}) \leq d$*.*

The upper bound $\max\{2, m-1\}$ is strikingly different from the upper bound $2d+1$ for representability of simplicial complexes (recall Theorem 1.2.7). While $2d+1$ depends linearly on the dimension of Δ (and hence linearly on the number of vertices, or neurons, in Δ), the number of maximal codewords may be very large compared to the number of neurons. One might then expect that the bound $\max\{2, m-1\}$ can be improved. We will see later that it cannot, at least for open embedding dimension.

Another important positive result is that adding a new non-maximal codeword to a code preserves open convexity. In fact, doing so does not increase the open embedding dimension by more than one.

Theorem 1.4.10 ("Monotonicity of Open Convexity", [CGIK16])**.** *Let $\mathcal{C} \subseteq \mathcal{D}$ be codes with the same maximal codewords. Then* $\operatorname{odim}(\mathcal{D}) \leq \operatorname{odim}(\mathcal{C}) + 1$*.*

Monotonicity is a useful tool when building open realizations of a code. Rather than build an open realiation of $\mathcal{C}$, one can (at the cost of a dimension) get away with building a realization of a code contained in $\mathcal{C}$ which has the same maximal codewords. Surprisingly, the theorem above does not hold when open embedding dimension is replaced by closed embedding dimension (this is the topic of Section 5.5).

Remark 1.4.11. In this section we have only scratched the surface of the convex codes literature. There is an entirely algebraic approach to understanding convex codes, which was introduced in [CIVCY13] and further developed in works such as [GJS19, IKR20, RMS20, GGPK$^+$18, GNY16]. There are also many interesting geometric, topological, and combinatorial works that we have not mentioned. For example, [Dav18] uses polytopes to explain the combinatorial and algebraic structure of certain families of codes, [RZ17] characterizes all (open or closed) convex codes in $\mathbb{R}^1$, and [MT20] characterizes codes that can be realized by connected sets.

1.5 Overview of This Work

Our work builds on the existing neural code literature in several directions. In Chapter 2 we provide constructive results that expand the study of intersection complete codes. In particular, we show that open and closed embedding dimensions are equal for simplicial complexes (Theorem 2.1.1), that $\operatorname{cdim}(\mathcal{C}) \leq \operatorname{odim}(\mathcal{C})$ whenever $\mathcal{C}$ is intersection complete (Theorem 2.2.7), and finally that if $\mathcal{C} \subseteq 2^{[n]}$ is intersection complete then

$$\operatorname{cdim}(\mathcal{C}) \leq \min\{2d+1, n-1\}$$

where $d = \dim(\Delta(\mathcal{C}))$ (Theorem 2.3.7). The latter result is a generalization of [CGIK16, Lemma 5.9], which establishes the $n-1$ term in the bound.

Chapter 3 introduces and studies a combinatorial notion of morphism for codes. We provide a combinatorial characterization of all morphisms (Theorem 3.2.3), and show that every code is isomorphic to a unique "reduced" code, up to relabeling neurons (Theorem 3.3.13). We connect the combinatorial study of morphisms to the algebraic theory of neural rings (Theorem 3.5.4), and we also show that the category of codes with morphisms is finitely bicomplete (Theorem 3.6.7).

Chapter 4 shows that morphisms provide a useful framework in which to study open and closed embedding dimensions. We define $\mathcal{D}$ to be a "minor" of $\mathcal{C}$ if there exists a surjective morphism $\mathcal{C} \to \mathcal{D}$, and we show that if $\mathcal{D}$ is a minor of $\mathcal{C}$ then one may use a (closed or open) realization of $\mathcal{C}$ to build a (closed or open) realization of $\mathcal{D}$ (Theorem 4.2.2). As a consequence, (closed or open) convex codes form a minor-closed family, and we may study minimal obstructions to convexity in the form of "minimally non-convex" codes. We exhibit an infinite family of minimally non-convex codes in Theorem 4.4.3, and a further family later in Section 5.6. We study $\mathbf{P_{Code}}$, the set of isomorphism classes of codes partially ordered by minors. We characterize the covering relation in $\mathbf{P_{Code}}$ (Theorem 4.5.10), and we show that $\mathbf{P_{Code}}$ is a graded poset with rank function given by the number of proper nonempty trunks in a code (Corollary 4.5.13).

In Chapter 5 we return to a slightly more concrete setting. We prove a general discrete geometry theorem (Theorem 5.1.13), and apply this theorem to explain novel open and closed embedding dimension bounds for over half a dozen families of codes. Among our families of codes, some novel "firsts" are worth highlighting:

- The codes $\mathcal{S}_n$ are the first family in which open and closed embedding dimension differ by an arbitrarily large finite amount,

- The codes $\mathcal{E}_n$ are the first family in which open embedding dimension grows exponentially as a function of the number of neurons (in fact they are the first family in which odim grows at anything larger than a linear rate), and

- The codes $\mathcal{C}_n$ exhibit the first infinite family of fundamentally distinct non-local obstructions to open convexity.

The families of codes that appear in our work and the phenomena that they exhibit are fully summarized in Figure 1.8 at the end of this chapter. Two families do not appear in this table: the codes $\mathcal{S}_{\mathcal{C}/\mathcal{D}}$ and $\mathcal{S}_{\mathcal{C}/\min}$ of Definition 5.8.1. These families generalize the families $\mathcal{S}_\Delta$ and $\mathcal{S}_n$ respectively to a more complicated setting, which is investigated in Section 5.8.

Chapter 6 introduces the study of "convex union representable" complexes—simplicial complexes which arise as the nerve of a collection of open convex sets whose union is convex. Such complexes were studied to understand local obstructions to convexity in [CGJ$^+$17] and [CFS19]. We improve on past results by providing more refined combinatorial criteria that such complexes must satisfy, thus allowing us to recognize new families of non-convex codes (Corollary 6.8.3).

At the beginning of each chapter we explain which papers the results of that chapter appear in. In general, we have streamlined and tweaked the presentation of our results compared to their initial appearances. We hope that this work provides a more complete, consistent, and elegant presentation of our various frameworks and results.

A myriad of open questions arise from our work. We have collected all of these in Chapter 7 for ease of reference.

Name	Parametrized by	Neurons	Features or Results	See
$\mathcal{C}_\Delta$	Simplicial complexes $\Delta \subseteq 2^{[n]}$	$[n+1]$	Minimally non-convex if Δ is not convex union representable (e.g. non-collapsible, non-contractible)	Theorem 4.4.3
$\mathcal{S}_n$	$n \geq 1$	$[n+1]$	$\text{odim}(\mathcal{S}_n) = n$ and $\text{cdim}(\mathcal{S}_n) \leq 2$	Definition 5.2.1
$\mathcal{S}_\Delta$	Simplicial complexes $\Delta \subseteq 2^{[n]}$ with m facets	$[n+1]$	$\text{odim}(\mathcal{S}_\Delta) = m$ and $\text{cdim}(\mathcal{S}_n) \leq n-1$	Definition 5.3.1
$\mathcal{E}_n$	$n \geq 2$	$[n+1]$	$\text{odim}(\mathcal{E}_n) = \binom{n-1}{\lfloor (n-1)/2 \rfloor}$ and $\text{cdim}(\mathcal{E}_n) \leq n-1$	Corollary 5.3.7
$\mathcal{P}_n$	$n \geq 1$	$[n+1] \cup [\overline{n+2}]$	Monotonicity of convexity is strict in dimension n	Definition 5.4.1
$\mathcal{A}_0$		$[3] \cup [\overline{5}]$	Monotonicity of convexity fails for closed realizations	Theorem 5.5.2
$\mathcal{A}_n$	$n \geq 2$	$[n+1] \cup [\overline{n}]$	Monotonicity of closed convexity fails with arbitrary finite gaps	Definition 5.5.5
$\mathcal{C}_n$	$n \geq 2$	$[n+1] \cup [\overline{n+1}]$	Locally perfect and minimally non-convex	Definition 5.6.1
$\mathcal{T}_n$	$n \geq 1$	$[n] \cup [\overline{n}]$	$\lceil n/2 \rceil \leq \text{odim}(\mathcal{T}_n) \leq n$ and $\text{cdim}(\mathcal{T}_n) \leq 2$	Definition 5.7.1
$\mathcal{M}_k$	$k \geq 1$	$[k]$	Maximum minor among codes with k codewords	Proposition 4.3.7

Figure 1.8: A table of the families of codes that appear in this work.

Chapter 2

EMBEDDING DIMENSIONS OF INTERSECTION COMPLETE CODES

In this chapter we provide several bounds on the open and closed embedding dimensions of intersection complete codes. All of these results first appeared in [Jef19a], from which the content of this chapter is adapted. These bounds can be viewed as "positive" results: they constructively guarantee that realizations exist under certain conditions. This contrasts the results given in Chapters 5 and 6, which are "negative" in the sense that they describe new obstructions to open and closed convexity of codes in various dimensions.

2.1 Embedding Dimensions of Simplicial Complexes

Simplicial complexes are a special case of intersection complete codes. For simplicial complexes, open and closed embedding dimensions turn out to be equal. To our knowledge, this result has been previously observed by researchers in the neural codes community, but we provided the first written proof in [Jef19a, Theorem 1.4]. We duplicate this proof with slightly more detail below.

Theorem 2.1.1. *Let* $\mathcal{C} \subseteq 2^{[n]}$ *be a simplicial complex. Then* $\operatorname{cdim}(\mathcal{C}) = \operatorname{odim}(\mathcal{C})$*.*

Proof. In Theorem 2.2.7 we will show that $\operatorname{cdim}(\mathcal{C}) \le \operatorname{odim}(\mathcal{C})$ for all intersection complete codes, not just simplicial complexes. Thus we need only prove that $\operatorname{odim}(\mathcal{C}) \le \operatorname{cdim}(\mathcal{C})$. Let $\mathcal{X} = \{X_1, \ldots, X_n\}$ be a compact realization of $\mathcal{C}$ in $\mathbb{R}^{\operatorname{cdim}(\mathcal{C})}$. For each nonempty codeword $c \in \mathcal{C}$, choose a point $p_c \in \mathcal{X}^c$. By compactness, each p_c is a positive distance from any X_i not containing it. Likewise, any disjoint X_σ and X_τ are separated by a positive distance.

Choose $\varepsilon > 0$ so that it is less than half the minimum among all these distances, and let $\mathcal{U} = \{U_1, \ldots, U_n\}$ where each U_i is the Minkowski sum of X_i with an open ball of radius ε.

Note that all U_i are open and convex. By choice of ε, if $p_c \notin X_i$ then $p_c \notin U_i$. Thus $p_c \in \mathcal{U}^c$ for all nonempty $c \in \mathcal{C}$, and $\mathcal{C} \subseteq \text{code}(\mathcal{U})$. Moreover, our choice of ε guarantees that if X_σ and X_τ are disjoint, then so are U_σ and U_τ. This implies that if X_σ is empty then so is U_σ, and so $\Delta(\text{code}(\mathcal{U})) \subseteq \Delta(\mathcal{C})$ (recall that $\Delta(\text{code}(\mathcal{U}))$ consists of all σ with $U_\sigma \neq \emptyset$).

Putting these containments together, we have

$$\mathcal{C} \subseteq \text{code}(\mathcal{U}) \subseteq \Delta(\text{code}(\mathcal{U})) \subseteq \Delta(\mathcal{C}) = \mathcal{C}.$$

Thus $\text{code}(\mathcal{U}) = \mathcal{C}$. We have constructed an open realization of $\mathcal{C}$ from a closed realization, proving the result. $\square$

Example 2.1.2. Consider the simplicial complex $\mathcal{C} = \{\mathbf{123}, \mathbf{14}, \mathbf{34}, 12, 13, 23, 1, 2, 3, 4, \emptyset\}$. Here $\text{cdim}(\mathcal{C}) = \text{odim}(\mathcal{C}) = 2$, and Figure 2.1 demonstrates the construction used in Theorem 2.1.1. Part (a) of the figure shows a closed realization $\mathcal{X} = \{X_1, X_2, X_3, X_4\}$ of $\mathcal{C}$ in $\mathbb{R}^2$ along with a choice of points p_c for each nonempty codeword c. Part (b) shows the resulting open realization $\mathcal{U} = \{U_1, U_2, U_3, U_4\}$ of $\mathcal{C}$, obtained by adding a small open ball to each X_i.

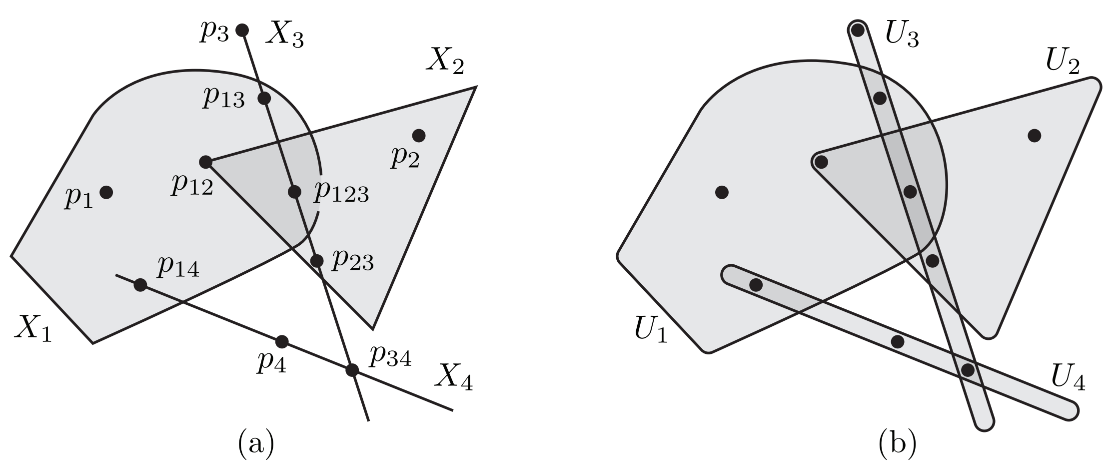


Figure 2.1: An illustration of the construction used in Theorem 2.1.1.

2.2 Closed Embedding Dimension is Bounded by Open Embedding Dimension

In this section our main result is Theorem 2.2.7, which states that $\text{cdim}(\mathcal{C}) \leq \text{odim}(\mathcal{C})$ for all intersection complete codes $\mathcal{C}$. We will prove this result constructively. In particular, Lemma 2.2.4 states that any open realization of an intersection complete code may be modified to obtain a non-degenerate open realization. Results of [CGIK16] then allow us to replace the sets in our realization by their closures without changing the realized code.

We first recall a useful characterization of intersection complete codes in terms of their realizations. This fact has been observed before in various forms, for example [CGJ$^+$19, Theorem 1.9]. We provide our own proof for completeness.

Proposition 2.2.1. *A code $\mathcal{C} \subseteq 2^{[n]}$ is intersection complete if and only if the following holds: for all $\sigma \in \Delta(\mathcal{C}) \setminus \mathcal{C}$ and all (not necessarily open, closed, or convex) realizations $\mathcal{U} = \{U_1, \dots, U_n\}$ of $\mathcal{C}$ there is some $i \in [n] \setminus \sigma$ with $U_\sigma \subseteq U_i$.*

Proof. First suppose that $\mathcal{C}$ is intersection complete, and has a realization $\mathcal{U} = \{U_1, \dots, U_n\}$. Let $\sigma \in \Delta(\mathcal{C}) \setminus \mathcal{C}$ and define $c_0 = \bigcap_{c \in \text{Tk}_\mathcal{C}(\sigma)} c$. The trunk $\text{Tk}_\mathcal{C}(\sigma)$ is nonempty since $\sigma \in \Delta(\mathcal{C})$, and $c_0 \in \mathcal{C}$ since $\mathcal{C}$ is intersection complete. Moreover, σ is a proper subset of c_0 since $\sigma \notin \mathcal{C}$. Thus we may choose $i \in c_0 \setminus \sigma$. We claim that $U_\sigma \subseteq U_i$. Indeed, since c_0 is the unique minimal element of $\text{Tk}_\mathcal{C}(\sigma)$, every codeword containing σ also contains i. This implies that $U_\sigma \subseteq U_i$.

For the converse, we prove the contrapositive. Suppose that $\mathcal{C}$ is not intersection complete, so there exist codewords c_1 and c_2 such that $c_1 \cap c_2 \notin \mathcal{C}$. Define $\sigma = c_1 \cap c_2$ and note that $\sigma \in \Delta(\mathcal{C}) \setminus \mathcal{C}$. Then choose any (not necessarily open, closed, or convex) realization $\mathcal{U} = \{U_1, \dots, U_n\}$ of $\mathcal{C}$, and let $i \in [n] \setminus \sigma$. Observe that i is contained in at most one of c_1 and c_2, and so either U_{c_1} or U_{c_2} is not contained in U_i. Since U_{c_1} and U_{c_2} are subsets of U_σ, we conclude that U_σ is not contained in U_i. This holds for all $i \in [n] \setminus \sigma$, proving the result. □

In Section 2.1 we converted closed realizations into open realizations by adding a small open ball to every set in the realization. To convert open realizations to closed realizations we will take the opposite approach, and uniformly shrink sets in a realization by a small amount. To shrink sets we use the following "trimming" operation, which also appears in [JOS+15].

Definition 2.2.2. Let $U \subseteq \mathbb{R}^d$ be any set and $\varepsilon > 0$. The *trim* of U by ε is the set

$$\operatorname{trim}(U, \varepsilon) := \{p \in U \mid B_\varepsilon(p) \subseteq U\},$$

where $B_\varepsilon(p)$ is the closed ball of radius ε centered at p.

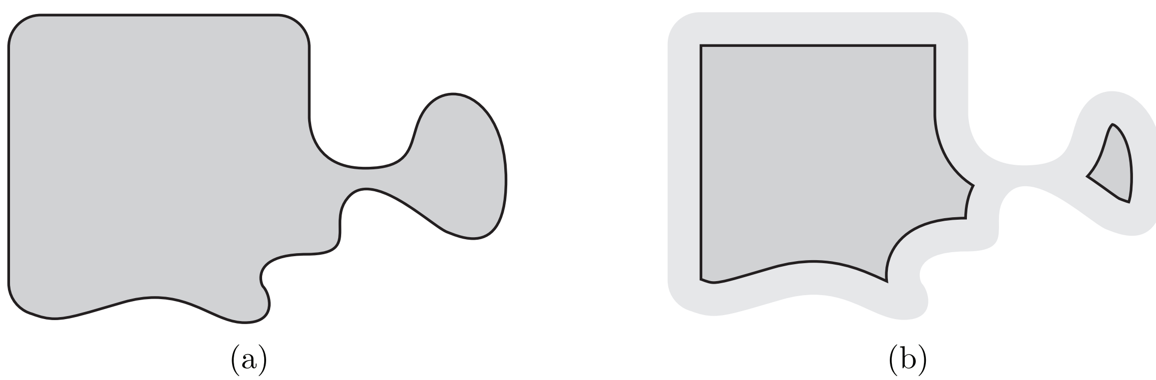


Figure 2.2: (a) An open set $U \subseteq \mathbb{R}^2$. (b) $\operatorname{trim}(U, \varepsilon)$ for a specific choice of ε.

Figure 2.2 shows an open set $U \subseteq \mathbb{R}^2$, and $\operatorname{trim}(U, \varepsilon)$ for one choice of $\varepsilon > 0$. Observe that trimming may create cusps on the boundary of the resulting set, and that it may cause a set to become disconnected. However, trimming has a number of useful properties outlined in Proposition 2.2.3 below, which we will subsequently make use of.

Proposition 2.2.3. *Let $U, V \subseteq \mathbb{R}^d$ be any sets and $\varepsilon > 0$. The following hold:*

(i) If U is open, then $\operatorname{trim}(U, \varepsilon)$ *is open.*

(ii) If U is convex, then $\operatorname{trim}(U, \varepsilon)$ *is convex.*

(iii) $\mathrm{cl}(\mathrm{trim}(U, \varepsilon)) \subseteq U$*.*

(iv) If $U \subseteq V$*, then* $\mathrm{trim}(U, \varepsilon) \subseteq \mathrm{trim}(V, \varepsilon)$*.*

(v) $\mathrm{trim}(U \cap V, \varepsilon) = \mathrm{trim}(U, \varepsilon) \cap \mathrm{trim}(V, \varepsilon)$*.*

Proof. We first prove statement (i). Let $p \in \mathrm{trim}(U, \varepsilon)$. Since $B_\varepsilon(p)$ is a closed subset of U and U is open, there exists $\delta > 0$ such that $B_{\varepsilon+\delta}(p) \subseteq U$. This implies that the open ball of radius δ centered at p is contained in $\mathrm{trim}(U, \varepsilon)$. Thus p is an interior point of $\mathrm{trim}(U, \varepsilon)$, so $\mathrm{trim}(U, \varepsilon)$ is open.

For statement (ii), let p and q be points in $\mathrm{trim}(U, \varepsilon)$. By convexity of U, the Minkowski sum $C = \overline{pq} + B_\varepsilon(0)$ is contained in U. For any r on $\overline{pq}$ we have $B_\varepsilon(r) \subseteq C \subseteq U$. Thus r lies in $\mathrm{trim}(U, \varepsilon)$, proving that $\mathrm{trim}(U, \varepsilon)$ is convex.

To prove statement (iii), observe that $\mathrm{cl}(\mathrm{trim}(U, \varepsilon)) \subseteq \mathrm{trim}(U, \varepsilon/2) \subseteq U$. Statement (iv) is immediate from the definition of trimming, and the final statement follows from the fact that $B_\varepsilon(p)$ is contained in both U and V if and only if it is contained in their intersection. □

Lemma 2.2.4. *Let* $\mathcal{C} \subseteq 2^{[n]}$ *be an intersection complete code, and let* $\mathcal{U} = \{U_1, \ldots, U_n\}$ *be an open realization of* $\mathcal{C}$*. Then there exists* $\varepsilon > 0$ *such that the collection* $\mathcal{V} = \{V_i := \mathrm{trim}(U_i, \varepsilon) \mid i \in [n]\}$ *is a non-degenerate open realization of* $\mathcal{C}$*.*

Proof. For each nonempty codeword c, choose a point $p_c \in \mathcal{U}^c$. Observe that we may choose ε small enough that $B_\varepsilon(p_c) \subseteq U_c$ for all nonempty codewords in $\mathcal{C}$. We claim that this suffices. Note by parts (i) and (ii) of Proposition 2.2.3 that all V_i are open and convex.

By choice of ε, $p_c \in V_c$ for all nonempty codewords c. In fact $p_c \in \mathcal{V}^c$ for all nonempty codewords c, and so $\mathcal{C} \subseteq \mathrm{code}(\mathcal{V})$. Since $V_i \subseteq U_i$ for all $i \in [n]$, we see that $\mathrm{code}(\mathcal{V})$ does not contain any maximal codewords that were not already present in $\mathcal{C}$. To prove that $\mathrm{code}(\mathcal{V}) = \mathcal{C}$, it remains to show that $\sigma \notin \mathrm{code}(\mathcal{V})$ for every $\sigma \in \Delta(\mathcal{C}) \setminus \mathcal{C}$.

Proposition 2.2.1 implies that if $\sigma \in \Delta(\mathcal{C}) \setminus \mathcal{C}$, then there exists $i \in [n] \setminus \sigma$ with $U_\sigma \subseteq U_i$. Parts (iv) and (v) of Proposition 2.2.3 tell us that

$$V_\sigma = \mathrm{trim}(U_\sigma, \varepsilon) \subseteq \mathrm{trim}(U_i, \varepsilon) = V_i.$$

Thus V_σ is covered by V_i, and σ is not a codeword of code$(\mathcal{V})$. We conclude that code$(\mathcal{V}) = \mathcal{C}$.

To see that the V_i form a non-degenerate realization, we must check condition (ii) from the definition of non-degeneracy (see Definition 1.4.7). For any nonempty $\sigma \subseteq [n]$, let p be a point in $\bigcap_{i\in\sigma} \partial V_i$. By part (iii) of Proposition 2.2.3, the closure of any V_i is contained in U_i, and so p lies in U_σ. In particular, U_σ is nonempty, and therefore so is V_σ. Thus we may choose a point $q \in V_\sigma$, and consider the line segment $\overline{pq}$. This situation is illustrated in Figure 2.3. Since p is a boundary point of all V_i with $i \in \sigma$, the line segment $\overline{pq}$ is contained in V_i except for the point p. But this implies that all points on the line segment except p lie in V_σ. Thus p is a boundary point of V_σ, and condition (ii) of Definition 1.4.7 is satisfied. $\square$

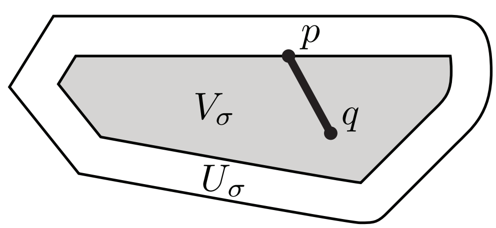


Figure 2.3: The objects used to prove non-degeneracy in Lemma 2.2.4.

Remark 2.2.5. The proof of Lemma 2.2.4 suggests a slightly more general result. Given an open realization $\mathcal{U} = \{U_1, \ldots, U_n\}$ of a (possibly not intersection complete) code $\mathcal{C}$, we could trim sets in the realization to obtain a non-degenerate open realization $\mathcal{V} = \{V_1, \ldots, V_n\}$ of a new code $\mathcal{D}$. The arguments given above imply that $\mathcal{C} \subseteq \mathcal{D} \subseteq \Delta(\mathcal{C})$, and it would be interesting to investigate the extent to which these containments are strict.

Example 2.2.6. Lemma 2.2.4 may fail when a code is not intersection complete. Consider the open realization $\mathcal{U} = \{U_1, U_2, U_3\}$ of the code $\mathcal{C} = \{\mathbf{123}, 12, 13, \emptyset\}$ in $\mathbb{R}^2$ illustrated in Figure 2.4(a). As part (b) of the figure illustrates, trimming this realization by any small $\varepsilon > 0$ will introduce the additional codeword 1 to the realized code. Although one can find realizations of $\mathcal{C}$ where trimming succeeds, some codes have the property that all of their open realizations are degenerate (see [CGIK16, Section 2.3]), and for these codes trimming will always fail.

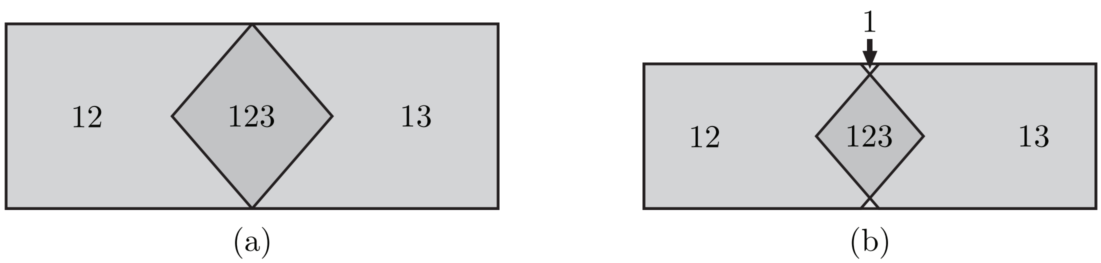


Figure 2.4: An illustration of how Lemma 2.2.4 may fail when $\mathcal{C}$ is not intersection complete.

Theorem 2.2.7. *Let* $\mathcal{C} \subseteq 2^{[n]}$ *be an intersection complete code. Then* $\operatorname{cdim}(\mathcal{C}) \leq \operatorname{odim}(\mathcal{C})$*.*

Proof. Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be an open realization of $\mathcal{C}$ in $\mathbb{R}^{\operatorname{odim}(\mathcal{C})}$. By Lemma 2.2.4, we may trim the sets in $\mathcal{U}$ to obtain a non-degenerate open realization. By [CGIK16, Theorem 2.10], the closures of the trimmed sets form a closed realization of $\mathcal{C}$. Thus $\operatorname{cdim}(\mathcal{C}) \leq \operatorname{odim}(\mathcal{C})$. □

Example 2.2.8. Figure 2.5 provides an illustration of Theorem 2.2.7 for the intersection complete code $\mathcal{C} = \{\mathbf{123}, \mathbf{14}, \mathbf{24}, \mathbf{34}, 23, 1, 2, 3, 4, \emptyset\}$. We begin with a degenerate open realization $\mathcal{U}$, and trim it slightly to obtain a non-degenerate open realization $\mathcal{V}$. Taking closures of the V_i, we can obtain a non-degenerate closed realization of $\mathcal{C}$.

2.3 A General Bound for Closed Embedding Dimension

In this section we provide a recipe for trying to build a closed realization of a (not necessarily intersection complete) code $\mathcal{C} \subseteq 2^{[n]}$ with $d = \dim(\Delta(\mathcal{C}))$. As we will prove in Lemma 2.3.6, this construction will succeed if and only if $\mathcal{C}$ is intersection complete. The dimension of our realization is $\min\{2d+1, n-1\}$, which is noteworthy because it is linear in both the number

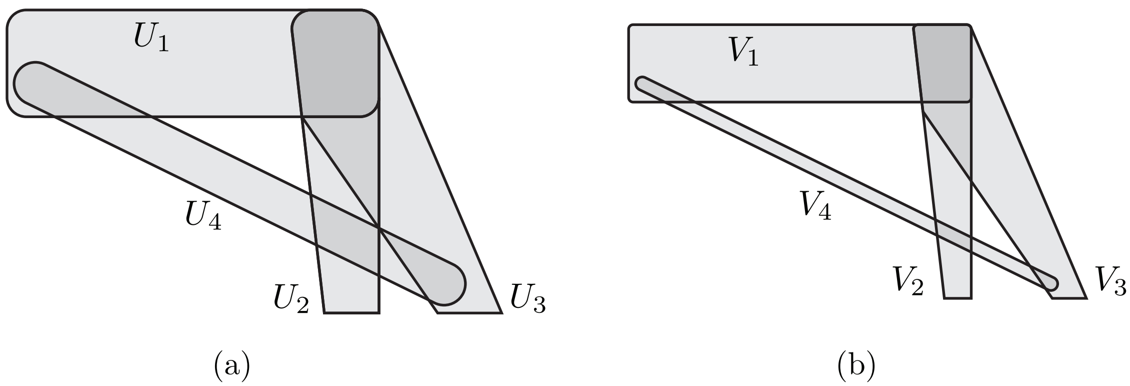


Figure 2.5: (a) A degenerate open realization $\mathcal{U} = \{U_1, U_2, U_3, U_4\}$ of an intersection complete code. (b) The trimmed non-degenerate open realization $\mathcal{V} = \{V_1, V_2, V_3, V_4\}$ of Lemma 2.2.4.

of neurons of $\mathcal{C}$ and linear in the dimension of the simplicial complex $\Delta(\mathcal{C})$. The $n-1$ portion of the bound follows from past work (see [CGIK16, Lemma 5.9]), and so our contribution is the $2d+1$ term. Our approach is based on the construction described in [Tan13, Theorem 3.1]. We describe our recipe in a series of steps, with proofs interspersed to justify steps.

Step 1: *Fix a code* $\mathcal{C} \subseteq 2^{[n]}$*, let* $d = \dim(\Delta(\mathcal{C}))$*, and let* $m = \min\{2d+1, n-1\}$*.*

Step 2: *Fix a pure, full-dimensional polytopal complex* $\mathcal{X}$ *with facets* $\{X_1, \ldots, X_n\}$ *in* $\mathbb{R}^m$ *such that any* $d+1$ *facets of* $\mathcal{X}$ *meet in a unique nonempty face.*

We pause to justify the existence of such a complex.

Lemma 2.3.1. *Let* $m = \min\{2d+1, n-1\}$*. There exists a pure, full-dimensional polytopal complex* $\mathcal{X}$ *in* $\mathbb{R}^m$ *with facets* $\{X_1, \ldots, X_n\}$ *such that any* $d+1$ *facets of* $\mathcal{X}$ *meet in a unique nonempty face of* $\mathcal{X}$*. In particular,* $\operatorname{code}(\mathcal{X})$ *contains all* $\sigma \subseteq [n]$ *with* $|\sigma| \leq d+1$*.*

Proof. First, recall that there exists a $(d+1)$-neighborly polytope in $\mathbb{R}^{m+1}$ with $n+1$ vertices. When $m = 2d+1$, one example is the cyclic polytope (see [Zie95, Corollary 0.8]), and when $m = n-1$ the n-simplex suffices. Let $P \subseteq \mathbb{R}^{m+1}$ be a polytope dual to a $(d+1)$-neighborly polytope with $n+1$ vertices. Let $F_1, \ldots, F_n, F_{n+1}$ be the facets of P, and observe that any $d+1$ facets of P meet in a unique face of P. Consider the Schlegel diagram of P in $\mathbb{R}^m$ based at the facet F_{n+1}. For $1 \leq i \leq n$, define X_i to be the image of F_i in the Schlegel diagram. We claim that the complex $\mathcal{X}$ with facets $\{X_1, \ldots, X_n\}$ is the desired polytopal complex.

Each X_i is full-dimensional since each F_i has dimension m. Furthermore, if $\sigma \subseteq [n]$ and $|\sigma| \leq d+1$, then (by $(d+1)$-neighborliness of the dual of P) the facets $\{F_i \mid i \in \sigma\}$ of P meet at a unique face of P, which implies that the facets $\{X_i \mid i \in \sigma\}$ of $\mathcal{X}$ meet in a unique face of $\mathcal{X}$. A point in the relative interior of this face will not lie in any X_j with $j \notin \sigma$, and so $\sigma \in \text{code}(\mathcal{X})$. This proves the result. □

With the complex $\mathcal{X}$ in hand, we are ready to construct our attempted realization of $\mathcal{C}$ in the final steps below.

Step 3: *For every $\sigma \subseteq [n]$ with $|\sigma| \leq d+1$, choose a point p_σ in the relative interior of the face X_σ of $\mathcal{X}$.*

Step 4: *Define $\mathcal{Y} = \{Y_1, \ldots, Y_n\}$ where $Y_i := \text{conv}\{p_c \mid c \in \text{Tk}_{\mathcal{C}}(i)\}$.*

We will prove in Lemma 2.3.6 that $\text{code}(\mathcal{Y}) = \widehat{\mathcal{C}}$. To arrive at this result we require a series of technical lemmas regarding the geometric structure of the various Y_i and their relationship to the various X_i.

Lemma 2.3.2. *Let $\sigma \subseteq [n]$ with $|\sigma| \geq 2$, and let $i \in \sigma$. Let H be a supporting hyperplane for the proper face X_σ of X_i. Then $Y_i \cap H = \text{conv}\{p_c \mid c \in \text{Tk}_{\mathcal{C}}(\sigma)\}$.*

Proof. Consider the points $\{p_c \mid c \in \text{Tk}_{\mathcal{C}}(i)\}$, the convex hull of which is equal to Y_i by definition. Since $Y_i \subseteq X_i$, we see that $Y_i \subseteq H^{\geq}$. Thus $Y_i \cap H$ is the convex hull of all points in $\{p_c \mid c \in \text{Tk}_{\mathcal{C}}(i)\}$ which lie in H. If $c \in \text{Tk}_{\mathcal{C}}(i)$ but $\sigma \not\subseteq c$, then we may choose $j \in \sigma \setminus c$, noting that $p_c \notin X_j$. In particular, $p_c \in X_i$ but $p_c \notin X_\sigma$. Thus p_c lies in $H^{>}$ when $\sigma \not\subseteq c$. On the other hand, if $\sigma \subseteq p_c$ then $p_c \in X_\sigma \subseteq H$. Thus $Y_i \cap H$ is the convex hull of $\{p_c \mid c \in \text{Tk}_{\mathcal{C}}(\sigma)\}$ as desired. □

Lemma 2.3.3. *Let $\sigma \subseteq [n]$ be nonempty. Then $Y_\sigma = \text{conv}\{p_c \mid c \in \text{Tk}_{\mathcal{C}}(\sigma)\}$.*

Proof. Let $C = \text{conv}\{p_c \mid c \in \text{Tk}_{\mathcal{C}}(\sigma)\}$. Then $C \subseteq Y_\sigma$ since each p_c with $c \in \text{Tk}_{\mathcal{C}}(\sigma)$ lies in Y_i for all $i \in \sigma$. For the reverse inclusion, we consider two cases. If $\sigma = \{i\}$ then $C = Y_i$ and the result is immediate. Otherwise, $|\sigma| \geq 2$ and we may choose $i \in \sigma$ and H a supporting hyperplane for the face X_σ of X_i. Observe that $Y_\sigma \subseteq Y_i \cap X_\sigma \subseteq Y_i \cap H$, and by Lemma 2.3.2 $Y_i \cap H = C$, proving the result. □

Lemma 2.3.4. *Let σ and τ be nonempty subsets of $[n]$. Then Y_σ is a face of Y_τ if and only if $\text{Tk}_{\mathcal{C}}(\sigma) \subseteq \text{Tk}_{\mathcal{C}}(\tau)$.*

Proof. First suppose that $\text{Tk}_{\mathcal{C}}(\sigma) \subseteq \text{Tk}_{\mathcal{C}}(\tau)$. This implies that every codeword that contains σ also contains τ, and so $\text{Tk}_{\mathcal{C}}(\sigma) = \text{Tk}_{\mathcal{C}}(\sigma \cup \tau)$. Lemma 2.3.3 then implies that $Y_\sigma = Y_{\sigma\cup\tau}$, and so it suffices to prove that $Y_{\sigma\cup\tau}$ is a face of Y_τ. Equivalently, we may reduce to the case in which $\tau \subseteq \sigma$. It will suffice to prove that Y_σ is a face of all Y_i with $i \in \tau$. If $\sigma = \{i\}$ then $\tau = \{i\}$ and the result is immediate. Otherwise, $|\sigma| \geq 2$, and for any $i \in \tau$ we may choose a hyperplane H supporting the face X_σ of X_i. Lemma 2.3.2 implies that $H \cap Y_i = \text{conv}\{p_c \mid c \in \text{Tk}_{\mathcal{C}}(\sigma)\}$, and Lemma 2.3.3 implies that this is Y_σ. Thus $Y_i \cap H = Y_\sigma$ and Y_σ is a face of Y_i for all $i \in \tau$ as desired.

For the converse, we argue by contrapositive. If $\text{Tk}_{\mathcal{C}}(\sigma) \not\subseteq \text{Tk}_{\mathcal{C}}(\tau)$ then there exists $c \in \mathcal{C}$ with $\sigma \subseteq c$ but $\tau \not\subseteq c$. Consider the point p_c. Since $\tau \not\subseteq c$, there exists $i \in \tau \setminus c$, and we see that $p_c \notin X_i$. But $Y_\tau \subseteq Y_i \subseteq X_i$, so $p_c \notin X_\tau$. On the other hand, $p_c \in X_\sigma$, so X_σ is not contained in X_τ, proving the result. □

Lemma 2.3.5. *Let $\sigma \subseteq [n]$ be nonempty. Then $\sigma \in \widehat{\mathcal{C}}$ if and only if the following holds: $\text{Tk}_{\mathcal{C}}(\sigma)$ is nonempty and properly contains $\text{Tk}_{\mathcal{C}}(\sigma \cup \{i\})$ for all $i \in [n] \setminus \sigma$.*

Proof. If σ is an intersection of codewords in $\mathcal{C}$, then there must be a codeword containing σ, and thus $\text{Tk}_{\mathcal{C}}(\sigma)$ is nonempty. If there exists $i \in [n] \setminus \sigma$ such that $\text{Tk}_{\mathcal{C}}(\sigma) = \text{Tk}_{\mathcal{C}}(\sigma \cup \{i\})$, then every codeword of $\mathcal{C}$ containing σ also contains i. This is a contradiction, since σ is the intersection of all codewords in $\mathcal{C}$ that contain it.

For the converse we consider two cases. If $\sigma = [n]$ and $\text{Tk}_{\mathcal{C}}(\sigma)$ is nonempty then $[n] \in \mathcal{C}$ and the result follows. Otherwise σ is a proper subset of $[n]$. Since $\text{Tk}_{\mathcal{C}}(\sigma)$ is nonempty and properly contains $\mathrm{Tk}_{\mathcal{C}}(\sigma \cup \{i\})$ for all $i \in [n] \setminus \sigma$, for every $i \in [n] \setminus \sigma$ we may choose a codeword c_i with $\sigma \subseteq c_i$ and $i \notin c_i$. The intersection of all such c_i is σ, proving the result. □

Lemma 2.3.6. *The collection $\mathcal{Y} = \{Y_1, \ldots, Y_n\}$ is a closed realization of $\widehat{\mathcal{C}}$. In particular, $\mathcal{Y}$ is a realization of $\mathcal{C}$ if and only if $\mathcal{C}$ is intersection complete.*

Proof. We argue for each nonempty $\sigma \subseteq [n]$ that $\sigma \in \widehat{\mathcal{C}}$ if and only if $\sigma \in \mathrm{code}(\mathcal{Y})$. By Lemma 2.3.5 it suffices to argue that $\sigma \in \mathrm{code}(\mathcal{Y})$ if and only if $\mathrm{Tk}_{\mathcal{C}}(\sigma)$ is nonempty and $\mathrm{Tk}_{\mathcal{C}}(\sigma \cup \{i\})$ is a proper subset of $\mathrm{Tk}_{\mathcal{C}}(\sigma)$ for all $i \in [n] \setminus \sigma$. By Lemma 2.3.4, this condition is equivalent to the requirement that Y_σ is nonempty, and $Y_{\sigma \cup \{i\}}$ is a proper face of Y_σ for all $i \in [n] \setminus \sigma$. This is in turn equivalent to the statement that Y_σ is nonempty and not covered by $\{Y_i \mid i \in [n] \setminus \sigma\}$, which happens if and only if $\sigma \in \mathrm{code}(\mathcal{Y})$, proving the result. □

Theorem 2.3.7. *Let $\mathcal{C} \subseteq 2^{[n]}$ be an intersection complete code, and let $d = \dim(\Delta(\mathcal{C}))$. Then $\mathrm{cdim}(\mathcal{C}) \leq \min\{2d + 1, n - 1\}$.*

Proof. In this section we have chosen a polytopal complex $\mathcal{X}$ in $\mathbb{R}^{\min\{2d+1,n-1\}}$, and used it to construct a collection $\mathcal{Y} = \{Y_1, \ldots, Y_n\}$ of closed convex sets. Lemma 2.3.6 says that $\mathcal{Y}$ realizes $\mathcal{C}$ if and only if $\mathcal{C}$ is intersection complete. This proves the result. □

Example 2.3.8. Consider the intersection complete code $\mathcal{C} = \{\mathbf{123}, \mathbf{14}, \mathbf{24}, \mathbf{34}, 23, 1, 2, 3, 4, \emptyset\}$ from Example 2.2.8. We have $d = 2$ and $n = 4$, and so $\min\{2d + 1, n - 1\} = \min\{5, 3\} = 3$. The polytopal complex $\mathcal{X} = \{X_1, X_2, X_3, X_4\}$ of Step 2 will be the Schlegel diagram of a 4-simplex, consisting of four tetrahedra subdividing a larger tetrahedron. Figure 2.6 shows this complex, along with its 1-skeleton. The figure also shows a choice of points p_c for every nonempty codeword c, as in Step 3, and the resulting closed realization $\mathcal{Y} = \{Y_1, Y_2, Y_3, Y_4\}$ from Step 4. In this realization Y_1 is a triangle, while Y_2, Y_3, and Y_4 are tetrahedra. Our construction is not minimal in dimension, since Example 2.2.8 implies $\mathrm{cdim}(\mathcal{C}) = \mathrm{odim}(\mathcal{C}) = 2$.

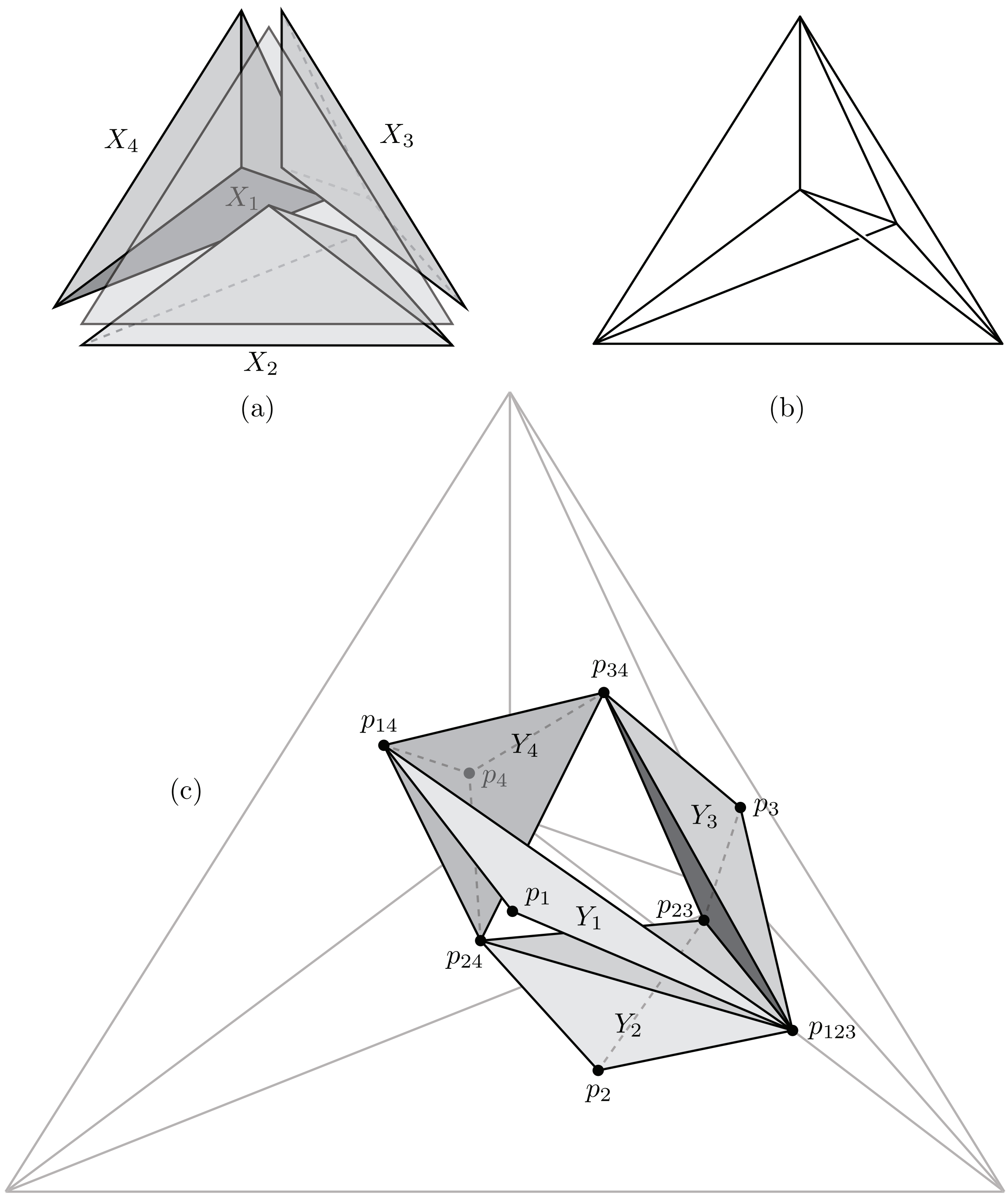


Figure 2.6: (a) The polytopal complex $\mathcal{X}$ with facets $\{X_1, X_2, X_3, X_4\}$ in $\mathbb{R}^3$, with X_1 transparent and facets split apart slightly to show structure. (b) The 1-skeleton of $\mathcal{X}$. (c) The closed realization $\mathcal{Y} = \{Y_1, Y_2, Y_3, Y_4\}$ of $\mathcal{C}$.

Chapter 3
MORPHISMS OF CODES

In order to more easily study the properties of codes and their realizations, it is important that we are able to compare codes to one another. This motivated us to define a notion of morphism for codes in [Jef20], and this chapter lays out the basic combinatorial theory of these morphisms. Importantly, we work with slightly different conventions and definitions than those in [Jef20]. We have inserted remarks to this effect where relevant, and we hope that the definitions we present here are an improvement on the original work.

The sections in this chapter proceed sequentially from concrete basics into more abstract material. The first four sections provide important background for the remaining chapters, while Section 3.5 and Section 3.6 delve into category theory related to morphisms, which we will not make signifcant reference to in further chapters.

3.1 Morphisms and Their Basic Properties

Morphisms behave similarly to continuous functions. While continuous functions capture topological information by requiring the preimage of an open set to be open, morphisms capture combinatorial (and as we will see later, geometric and algebraic) information by requiring the preimage of a trunk to be a trunk.

Definition 3.1.1. Let $\mathcal{C}$ and $\mathcal{D}$ be codes. A function $f : \mathcal{C} \to \mathcal{D}$ is a *morphism* if for every proper trunk $T \subseteq \mathcal{D}$ the preimage $f^{-1}(T)$ is a proper trunk in $\mathcal{C}$. A morphism is an *isomorphism* if it has an inverse function which is also a morphism.

One can observe immediately that the identity function on a code is a morphism, and that the composition of two morphisms is again a morphism. Thus the class of codes together with morphisms forms a category, which we investigate in later sections. It is also worth noting that because $f^{-1}(\mathcal{D}) = \mathcal{C}$, the preimage of any (possibly not proper) trunk under a morphism is again a trunk.

Remark 3.1.2. When we introduced morphisms in [Jef20], we allowed the preimage of a proper trunk to be a non-proper trunk. This was convenient in the context of that paper, in which we did not assume that the empty codeword was present in every code. We have added the word "proper" to Definition 3.1.1 because it allows for cleaner statements of certain theorems and ensures several nice properties of morphisms. For example, Proposition 3.1.3 implies that the image of any morphism (as a function) is a subcode of the codomain. It also endows the category of codes with an initial object (see Section 3.6 for details).

Proposition 3.1.3. *Let $f : \mathcal{C} \to \mathcal{D}$ be a morphism. Then $f(\emptyset) = \emptyset$.*

Proof. Suppose that $f(\emptyset) = d \neq \emptyset$ and consider the proper trunk $\mathrm{Tk}_{\mathcal{D}}(d)$ in $\mathcal{D}$. Note that $f^{-1}(\mathrm{Tk}_{\mathcal{D}}(d))$ is a trunk in $\mathcal{C}$ that contains the empty codeword. The only such trunk is $\mathcal{C}$, contradicting the fact that f is a morphism. □

Although the definition of morphisms draws paralells with the definition of continuous functions, they are not the same as continuous functions with respect to the topology generated by trunks. The following example describes a morphism that is a homeomorphism with respect to the topology generated by trunks, but is not an isomorphism.

Example 3.1.4. Let $\mathcal{C} = \{\mathbf{12}, \mathbf{13}, \emptyset\}$ and $\mathcal{D} = \{\mathbf{1}, \mathbf{2}, \emptyset\}$. The Hasse diagrams of these two codes are shown in Figure 3.1, and the proper trunks in each code are highlighted in grey. Note that $\mathcal{C}$ has three nonempty trunks, while $\mathcal{D}$ has only two.

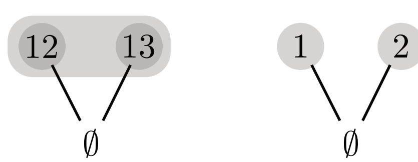


Figure 3.1: The codes $\mathcal{C}$ and $\mathcal{D}$, with proper trunks highlighted in grey.

Observe that there is a bijection $f : \mathcal{C} \to \mathcal{D}$ given by sending $12 \mapsto 1$, $13 \mapsto 2$, $\emptyset \mapsto \emptyset$. One can check that f is a morphism by hand. However, the inverse function $f^{-1} : \mathcal{D} \to \mathcal{C}$ is not a morphism. The preimage of the proper trunk $\mathrm{Tk}_{\mathcal{C}}(1) = \{12, 13\}$ under this inverse function is $\{1, 2\}$, which is not a proper trunk in $\mathcal{D}$.

In the remainder of this section we establish some basic properties of trunks and morphisms. Many of these results (such as Propositions 3.1.5, 3.1.6, and 3.1.7) will be used repeatedly and sometimes implicitly throughout the remainder of the text.

Proposition 3.1.5. *The intersection of two trunks is a trunk.*

Proof. Let $\mathcal{C} \subseteq 2^{[n]}$ be a code, and let T_1 and T_2 be trunks in $\mathcal{C}$. If either T_1 or T_2 is empty, then $T_1 \cap T_2 = \emptyset$, which is by definition a trunk in $\mathcal{C}$. Otherwise choose index sets σ_1 and σ_2 such that $T_1 = \mathrm{Tk}_{\mathcal{C}}(\sigma_1)$ and $T_2 = \mathrm{Tk}_{\mathcal{C}}(\sigma_2)$. One may verify that $T_1 \cap T_2 = \mathrm{Tk}_{\mathcal{C}}(\sigma_1 \cup \sigma_2)$, which is a trunk. □

Proposition 3.1.6 (Simple trunk criterion for morphisms)**.** *Let $\mathcal{C} \subseteq 2^{[n]}$ and $\mathcal{D} \subseteq 2^{[m]}$ be codes. A function $f : \mathcal{C} \to \mathcal{D}$ is a morphism if and only if $f^{-1}(\mathrm{Tk}_{\mathcal{D}}(i))$ is a proper trunk in $\mathcal{C}$ for every $i \in [m]$.*

Proof. The forward implication follows from the definition of morphisms. For the reverse implication, note that every proper trunk in $\mathcal{D}$ can be written as $\mathrm{Tk}_{\mathcal{D}}(\tau)$ for some nonempty $\tau \subseteq [m]$. Then observe that

$$f^{-1}(\mathrm{Tk}_{\mathcal{D}}(\tau)) = f^{-1}\left(\bigcap_{i \in \tau} \mathrm{Tk}_{\mathcal{D}}(i)\right) = \bigcap_{i \in \tau} f^{-1}(\mathrm{Tk}_{\mathcal{D}}(i)).$$

The righthand term is a finite intersection of proper trunks, which by Proposition 3.1.5 is a proper trunk in $\mathcal{C}$. Thus f is a morphism. □

Proposition 3.1.7 (Morphisms are montone)**.** *Let $f : \mathcal{C} \to \mathcal{D}$ be a morphism. If $c_1, c_2 \in \mathcal{C}$ are such that $c_1 \subseteq c_2$, then $f(c_1) \subseteq f(c_2)$.*

Proof. Consider the trunk $f^{-1}(\mathrm{Tk}_{\mathcal{D}}(f(c_1)))$ in $\mathcal{C}$. This trunk contains c_1 by construction, and since $c_1 \subseteq c_2$ we conclude that c_2 also lies in this trunk. Hence $f(c_2)$ lies in $\mathrm{Tk}_{\mathcal{D}}(f(c_1))$. By definition, this implies that $f(c_1) \subseteq f(c_2)$. □

Remark 3.1.8. Note that Example 3.1.4 implies that not every monotone map between codes is a morphism. In particular, the inverse function of the morphism described in this example is monotone, but is not a morphism.

Proposition 3.1.9. *Let $\mathcal{C} \subseteq 2^{[n]}$ be a code and let $\mathcal{D} \subseteq \mathcal{C}$ be a subcode. The inclusion function $\iota : \mathcal{D} \to \mathcal{C}$ is a morphism.*

Proof. One may verify that $\iota^{-1}(\text{Tk}_{\mathcal{C}}(\sigma)) = \text{Tk}_{\mathcal{D}}(\sigma)$ for every $\sigma \subseteq [n]$, noting that $\text{Tk}_{\mathcal{D}}(\sigma)$ is a proper trunk in $\mathcal{D}$ whenever $\text{Tk}_{\mathcal{C}}(\sigma)$ is a proper trunk in $\mathcal{C}$ (i.e. whenever σ is nonempty). This proves the result. □

Proposition 3.1.10 (Restricting the domain)**.** *Let $f : \mathcal{C} \to \mathcal{D}$ be a morphism and let $\mathcal{E} \subseteq \mathcal{C}$ be any subcode. The restricted function $f|_{\mathcal{E}} : \mathcal{E} \to \mathcal{D}$ is a morphism.*

Proof. The map $f|_{\mathcal{E}}$ is the composition of the inclusion morphism $\mathcal{E} \hookrightarrow \mathcal{C}$ with f, which is a morphism. □

Proposition 3.1.11 (Restricting the codomain)**.** *Let $f : \mathcal{C} \to \mathcal{D}$ be a morphism and let $\mathcal{E} \subseteq \mathcal{D}$ be any subcode containing $f(\mathcal{C})$. The restricted function $f : \mathcal{C} \to \mathcal{E}$ is a morphism.*

Proof. Since $f(\mathcal{C}) \subseteq \mathcal{E}$, any proper trunk $\text{Tk}_{\mathcal{E}}(\sigma)$ has the property that $f^{-1}(\text{Tk}_{\mathcal{E}}(\sigma)) = f^{-1}(\text{Tk}_{\mathcal{D}}(\sigma))$, which is a proper trunk in $\mathcal{C}$, proving the result. □

Definition 3.1.12. Let $\mathcal{C} \subseteq 2^{[n]}$ and let $w : [n] \to [n]$ be a permutation. Define a map $p_w : \mathcal{C} \to 2^{[n]}$ by $p_w(c) = \{w(i) \mid i \in c\}$. The map p_w is called a *permutation morphism.*

Proposition 3.1.13 (Permuting neurons is an isomorphism)**.** *Let $\mathcal{C} \subseteq 2^{[n]}$ and let $w : [n] \to [n]$ be a permutation. The permutation morphism $p_w : \mathcal{C} \to 2^{[n]}$ of Definition 3.1.12 is a morphism. It is an isomorphism onto its image, with inverse given by the restriction of $p_{w^{-1}}$ to this image.*

Proof. To prove that p_w is a morphism it suffices by Proposition 3.1.6 to show that $p_w^{-1}(\text{Tk}_{2^{[n]}}(i))$ is a proper trunk in $\mathcal{C}$ for all $i \in [n]$. One may compute that $p_w^{-1}(\text{Tk}_{2^{[n]}}(i)) = \text{Tk}_{\mathcal{C}}(w^{-1}(i))$, which is indeed a proper trunk in $\mathcal{C}$. Let $\mathcal{D} \subseteq 2^{[n]}$ be the image of $\mathcal{C}$ under p_w, and observe that the morphism $p_{w^{-1}} : 2^{[n]} \to 2^{[n]}$ restricts to an inverse of $p_w : \mathcal{C} \to \mathcal{D}$. Thus p_w is an isomorphism onto its image as desired. □

Definition 3.1.14. Let $\mathcal{C} \subseteq 2^{[n]}$ be a code, and let $\sigma \subseteq [n]$. Define a function $\pi_\sigma : \mathcal{C} \to 2^\sigma$ by $\pi_\sigma(c) = c \cap \sigma$. This is called the *restriction morphism* defined by σ. We will refer to the image of $\mathcal{C}$ under π_σ as $\mathcal{C}$ *restricted to* σ, and denote it $\mathcal{C}|_\sigma := \pi_\sigma(\mathcal{C})$.

Proposition 3.1.15. *The restriction morphism π_σ described in Definition 3.1.14 is a morphism.*

Proof. To prove that π_σ is a morphism it suffices by Proposition 3.1.6 to show that $\pi_\sigma^{-1}(\text{Tk}_{2^\sigma}(i))$ is a proper trunk in $\mathcal{C}$ for all $i \in \sigma$. One may verify that $\pi_\sigma^{-1}(\text{Tk}_{2^\sigma}(i)) = \text{Tk}_\mathcal{C}(i)$, proving the result. □

Example 3.1.16. Recall the codes $\mathcal{C} = \{\mathbf{12}, \mathbf{13}, \emptyset\}$ and $\mathcal{D} = \{\mathbf{1}, \mathbf{2}, \emptyset\}$ from Example 3.1.4. Let $\sigma = \{2, 3\}$, and observe that $\mathcal{C}|_\sigma = \{\mathbf{2}, \mathbf{3}, \emptyset\}$. The code $\mathcal{C}|_\sigma$ is isomorphic to $\mathcal{D}$ via a permutation morphism that sends $2 \mapsto 1$ and $3 \mapsto 2$.

3.2 A Combinatorial Characterization of Morphisms

Proving that a function is a morphism is generally not too arduous, particularly with the aid of Proposition 3.1.6. However, constructing a morphism from scratch is not an obviously straightforward task. The following definition and proposition provide a general recipe for constructing morphisms. In fact, Theorem 3.2.3 tells us that every morphism arises from this recipe. This also provides a new way to show that a function is a morphism: one needs only show that the function arises from the recipe below.

Definition 3.2.1. Let $\mathcal{C} \subseteq 2^{[n]}$ be a code, and let $\{T_1, \ldots, T_m\}$ be a collection of proper trunks in $\mathcal{C}$. Define a function $f : \mathcal{C} \to 2^{[m]}$ by

$$f(c) = \{j \in [m] \mid c \in T_j\}.$$

The function f is called the *morphism determined by the trunks* $\{T_1, \ldots, T_m\}$.

The word “collection” above indicates that we are regarding $\{T_1, \ldots, T_m\}$ as an ordered multiset, similar to how we regard a realization as an ordered multiset. Note that the function f records the indices of the trunks that a codeword c lies in. In other words, the function f records how the trunks $\{T_1, \ldots, T_m\}$ intersect and cover one another inside $\mathcal{C}$, similar to how the association between points in a realization and codewords tells us how sets in the realization intersect and cover one another.

We allow the case $m = 0$ in this definition, in which case the collection of trunks in question is empty, and the morphism they determine is the map $c \mapsto \emptyset$. The map $c \mapsto \emptyset$ also arises from the case when the collection of trunks is nonempty, but every trunk in the collection is itself empty.

Proposition 3.2.2. *The function described in Definition 3.2.1 is a morphism.*

Proof. By Proposition 3.1.6 we need only check that $f^{-1}(\mathrm{Tk}_{2^{[m]}}(j))$ is a proper trunk in $\mathcal{C}$ for all $j \in [m]$. By construction $f(c) \in \mathrm{Tk}_{2^{[m]}}(j)$ if and only if $c \in T_j$. Thus $f^{-1}(\mathrm{Tk}_{2^{[m]}}(j)) = T_j$ for all $j \in [m]$, and so f is a morphism. □

In fact, up to restricting the codomain appropriately, every morphism arises in this way. Moreover, the following theorem tells us exactly which trunks determined a given morphism $f : \mathcal{C} \to \mathcal{D}$. This result will be particularly useful to a number of our proofs, since it reduces questions about the morphism f to questions about how trunks intersect and cover one another in $\mathcal{C}$.

Theorem 3.2.3 (Every morphism is determined by trunks)**.** *Let $\mathcal{C} \subseteq 2^{[n]}$ and $\mathcal{D} \subseteq 2^{[m]}$ be codes and let $f : \mathcal{C} \to \mathcal{D}$ be a morphism. Then f is the morphism determined by the trunks $\{T_1, \ldots, T_m\}$ where $T_j = f^{-1}(\mathrm{Tk}_{\mathcal{D}}(j))$, and we restrict the codomain of f from $2^{[m]}$ to $\mathcal{D}$.*

Proof. We must show that $f(c) = \{j \in [m] \mid c \in T_j\}$ for all $c \in \mathcal{C}$. Equivalently, we must show that $f(c) \in \mathrm{Tk}_{\mathcal{D}}(j)$ if and only if $c \in T_j$. For the forward implication, observe that $f(c) \in \mathrm{Tk}_{\mathcal{D}}(j)$ implies that $c \in f^{-1}(\mathrm{Tk}_{\mathcal{D}}(j)) = T_j$. The converse follows from the fact that if $c \in T_j$ then $f(c) \in f(T_j) \subseteq \mathrm{Tk}_{\mathcal{D}}(j)$. This proves the result. □

Example 3.2.4. Recall the codes $\mathcal{C} = \{\mathbf{12}, \mathbf{13}, \emptyset\}$ and $\mathcal{D} = \{\mathbf{1}, \mathbf{2}, \emptyset\}$ from Example 3.1.4. Also recall the bijective morphism $f : \mathcal{C} \to \mathcal{D}$ given by $12 \mapsto 1$, $13 \mapsto 2$, $\emptyset \mapsto \emptyset$. One can compute that f is the morphism determined by the collection of trunks $\{T_1, T_2\}$, where $T_1 = \mathrm{Tk}_{\mathcal{C}}(2) = \{12\}$ and $T_2 = \mathrm{Tk}_{\mathcal{C}}(3) = \{13\}$.

3.3 Isomorphism Classes and Reduced Codes

Our main result in this section is that every isomorphism class of codes has a "nice" representative, which is unique up to permutation of neurons (see Theorem 3.3.13). By "nice," we mean that the representative does not contain any unnecessary or redundant information. The following three definitions make this precise.

Definition 3.3.1. A neuron $i \in [n]$ is *trivial* in a code $\mathcal{C} \subseteq 2^{[n]}$ if $\mathrm{Tk}_{\mathcal{C}}(i) = \emptyset$. Equivalently, i is trivial in $\mathcal{C}$ if and only if it does not appear in any codeword of $\mathcal{C}$.

Definition 3.3.2. Let $\mathcal{C} \subseteq 2^{[n]}$ be a code, let $i \in [n]$ be a nontrivial neuron in $\mathcal{C}$, and let $\sigma \subseteq [n]$ be such that $i \notin \sigma$. Then i is *redundant* to σ if $\mathrm{Tk}_{\mathcal{C}}(i) = \mathrm{Tk}_{\mathcal{C}}(\sigma)$. For any $i \in [n]$ we call i simply *redundant* if there exists σ so that i is redundant to σ.

Definition 3.3.3. A code is called *reduced* if it does not have any trivial or redundant neurons.

Example 3.3.4. Our two running example codes $\mathcal{C} = \{\mathbf{12}, \mathbf{13}, \emptyset\}$ and $\mathcal{D} = \{\mathbf{1}, \mathbf{2}, \emptyset\}$ are both reduced. However, the code $\mathcal{E} = \{\mathbf{124}, 1, 2, \emptyset\} \subseteq 2^{[4]}$ is not reduced: the neuron 3 is trivial, and the neuron 4 is redundant to $\sigma = \{1, 2\}$. If we restrict $\mathcal{E}$ to $\{1, 2\}$, we obtain the reduced code $\mathcal{E}|_{\{1,2\}} = \{\mathbf{12}, 1, 2, \emptyset\}$.

Reduced codes are the main topic of this section. To begin investigating them, we require one further definition: irreducible trunks. Irreducible trunks will help us connect the neurons in a code with the actual combinatorics of trunks in the code, as Proposition 3.3.6 and Theorem 3.3.7 begin to establish.

Definition 3.3.5. Let $\mathcal{C} \subseteq 2^{[n]}$ be a code. A nonempty proper trunk $T \subseteq \mathcal{C}$ is called *irreducible* if it is not the intersection of two trunks that properly contain it.

Proposition 3.3.6 (Irreducible trunks are simple)**.** *Let $\mathcal{C} \subseteq 2^{[n]}$ be a code and let T be an irreducible trunk in $\mathcal{C}$. Then $T = \mathrm{Tk}_{\mathcal{C}}(i)$ for some $i \in [n]$.*

Proof. Write $T = \mathrm{Tk}_{\mathcal{C}}(\sigma)$ for some nonempty σ, noting that we can do so because T is proper and nonempty. We have that $T = \bigcap_{i\in\sigma} \mathrm{Tk}_{\mathcal{C}}(i)$. Since T is irreducible, at least one of the terms in this intersection must be equal to T itself. Thus $T = \mathrm{Tk}_{\mathcal{C}}(i)$ as desired. □

Theorem 3.3.7. *Let $\mathcal{C} \subseteq 2^{[n]}$ be a code. Then $\mathcal{C}$ is reduced if and only if the map $i \mapsto \mathrm{Tk}_{\mathcal{C}}(i)$ is a bijection between neurons and the irreducible trunks in $\mathcal{C}$.*

Proof. First suppose that $\mathcal{C}$ is reduced. We argue that $\mathrm{Tk}(i)$ is irreducible for all $i \in [n]$. Note that $\mathrm{Tk}(i)$ is nonempty since $\mathcal{C}$ has no trivial neurons. To prove that $\mathrm{Tk}(i)$ is irreducible, we just have to show it is not the intersection of two trunks properly containing it. Suppose for contradiction that $\mathrm{Tk}(i) = \mathrm{Tk}(\sigma) \cap \mathrm{Tk}(\tau)$ where $\mathrm{Tk}(\sigma)$ and $\mathrm{Tk}(\tau)$ properly contain $\mathrm{Tk}(i)$. Since the containment is proper, we have that $i \notin \sigma \cup \tau$. But $\mathrm{Tk}(\sigma) \cap \mathrm{Tk}(\tau) = \mathrm{Tk}(\sigma \cup \tau)$, so i is redundant to $\sigma \cup \tau$. Since $\mathcal{C}$ is reduced this is a contradiction.

Combined with Proposition 3.3.6, we conclude that the set of simple trunks is exactly the set of irreducible trunks. To prove that $i \mapsto \mathrm{Tk}_{\mathcal{C}}(i)$ is a bijection we must show that no two neurons map to the same simple trunk. Suppose for contradiction that $\mathrm{Tk}(i) = \mathrm{Tk}(j)$ for some $i \neq j$. Then i is redundant to $\{j\}$, which is a contradiction since $\mathcal{C}$ is reduced. This proves the forward implication.

For the converse, suppose that $i \mapsto \mathrm{Tk}(i)$ is a bijection between neurons and irreducible trunks, and let $i \in [n]$ be arbitrary. Since $\mathrm{Tk}(i)$ is irreducible, it is nonempty, and i is not trivial. Suppose for contradiction that i were redundant to some $\sigma \subseteq [n]$. Note that σ must be nonempty, and so we may write $\mathrm{Tk}(i) = \bigcap_{j\in\sigma} \mathrm{Tk}(j)$. Since the map $i \mapsto \mathrm{Tk}(i)$ is injective, $\mathrm{Tk}(i) \neq \mathrm{Tk}(j)$ for all $j \in \sigma$, so in particular $\mathrm{Tk}(i)$ is properly contained in all $\mathrm{Tk}(j)$ in the intersection. But then we can group the terms in the intersection appropriately so that $\mathrm{Tk}(i)$ is the intersection of two trunks that properly contain it, contradicting its irreducibility. This proves the result. □

Classifying reduced codes and isomorphism classes also requires us to better understand surjective morphisms, which will make a reappearance in Chapter 4. As Theorem 3.3.9 below shows, there are a number of ways to recognize when a surjective morphism is in fact an isomorphism.

Proposition 3.3.8. *Let $f : \mathcal{C} \to \mathcal{D}$ be a surjective morphism. Then the map $T \mapsto f^{-1}(T)$ is an injective map from the set of trunks in $\mathcal{D}$ to the set of trunks in $\mathcal{C}$. In particular, $\mathcal{C}$ has at least as many trunks as $\mathcal{D}$.*

Proof. Suppose that T and S are trunks in $\mathcal{D}$ such that $f^{-1}(T) = f^{-1}(S)$. Surjectivity of f then implies that $T = f(f^{-1}(T)) = f(f^{-1}(S)) = S$, proving the result. □

Theorem 3.3.9. *Let $f : \mathcal{C} \to \mathcal{D}$ be a surjective morphism. Then the following are equivalent:*

(i) f is an isomorphism,

(ii) The map $T \mapsto f^{-1}(T)$ is a bijection between trunks in $\mathcal{D}$ and trunks in $\mathcal{C}$,

(iii) The map $T \mapsto f^{-1}(T)$ is a surjection between trunks in $\mathcal{D}$ and trunks in $\mathcal{C}$, and

(iv) $\mathcal{C}$ and $\mathcal{D}$ have the same number of trunks.

Proof. First note that (i) implies (ii) because the inverse morphism $f^{-1} : \mathcal{D} \to \mathcal{C}$ is bijective. To prove that (ii) implies (i) we must first argue that f is bijective as a function. By hypothesis f is surjective, so we need only show injectivity. Let c_1 and c_2 be codewords in $\mathcal{C}$ be such that $f(c_1) = f(c_2)$. Since f induces a bijection on trunks, there exist trunks T and S in $\mathcal{D}$ such that $f^{-1}(T) = \mathrm{Tk}_{\mathcal{C}}(c_1)$ and $f^{-1}(S) = \mathrm{Tk}_{\mathcal{C}}(c_2)$. Since $f(c_1) = f(c_2)$ and f is surjective, we have the following:

$$T = f(f^{-1}(T)) = f(\mathrm{Tk}_{\mathcal{C}}(c_1)) = f(\mathrm{Tk}_{\mathcal{C}}(c_2)) = f(f^{-1}(S)) = S.$$

Since $T = S$ and f induces a bijection on trunks we conclude that $\mathrm{Tk}_{\mathcal{C}}(c_1) = \mathrm{Tk}_{\mathcal{C}}(c_2)$, which implies $c_1 = c_2$. Thus f is a bijection. One may then verify that the inverse function $f^{-1} : \mathcal{D} \to \mathcal{C}$ is a morphism, again using surjectivity of f. This implies that f is an isomorphism as desired.

The equivalence of items (ii)-(iv) is more straightforward. Clearly (ii) implies (iii), and (iii) implies (iv) because by Proposition 3.3.8 the map $T \mapsto f^{-1}(T)$ is injective. Finally, the fact that (iv) implies (ii) again follows from Proposition 3.3.8. □

This result also confirms the useful and intuitive fact that isomorphisms induce a bijection on trunks. As a first application of this theorem, we see that isomorphic reduced codes are unique up to permutation of neurons.

Corollary 3.3.10. *Let $\mathcal{C} \subseteq 2^{[n]}$ and $\mathcal{D} \subseteq 2^{[m]}$ be codes, and let $f : \mathcal{C} \to \mathcal{D}$ be an isomorphism. If both $\mathcal{C}$ and $\mathcal{D}$ are reduced, then f is a permutation isomorphism.*

Proof. By Theorem 3.3.9, the isomorphism f induces a bijection between trunks in $\mathcal{C}$ and trunks in $\mathcal{D}$. This bijection preserves containments and intersections of trunks, and so it restricts to a bijection between the irreducible trunks in $\mathcal{C}$ and irreducible trunks in $\mathcal{D}$. Theorem 3.3.7 implies that the sets of irreducible trunks in $\mathcal{C}$ and $\mathcal{D}$ are in bijection with $[n]$ and $[m]$ respectively. Thus f induces a bijection $w : [n] \to [m]$ where $w(i) = j$ whenever $f(\mathrm{Tk}_{\mathcal{C}}(i)) = \mathrm{Tk}_{\mathcal{D}}(j)$. One may then verify that $f(c) = \{w(i) \mid i \in c\}$, so f is the permutation isomorphism induced by w, proving the result. □

Our final goal in this section is to prove Theorem 3.3.13 below, which tells us that every code is isomorphic to a reduced code, and that reduced codes use the "fewest" neurons possible to represent the information in a code. We first require a small supporting lemma.

Lemma 3.3.11. *Let $\mathcal{C} \subseteq 2^{[n]}$ be a code, and suppose that n is a redundant neuron. Then the restriction morphism $\pi : \mathcal{C} \to \mathcal{C}|_{[n-1]}$ given by $c \mapsto c \cap [n-1]$ is an isomorphism.*

Proof. Since π is surjective, it suffices by Theorem 3.3.9 to show that the map $T \mapsto \pi^{-1}(T)$ is a surjection on trunks. Since n is a redundant neuron, we may choose $\sigma \subseteq [n-1]$ such that $\mathrm{Tk}_{\mathcal{C}}(n) = \mathrm{Tk}_{\mathcal{C}}(\sigma)$. Let $T \subseteq \mathcal{C}$ be any trunk. If T is empty, then it is the preimage of the empty trunk in $\mathcal{C}|_{[n-1]}$. Otherwise, we may write $T = \mathrm{Tk}_{\mathcal{C}}(\tau)$ for some $\tau \subseteq [n]$. In fact, we may assume that $\tau \subseteq [n-1]$: if $n \in \tau$, then we may replace τ by $(\tau \setminus \{n\}) \cup \sigma$. Then observe that $\mathrm{Tk}_{\mathcal{C}}(\tau) = \pi^{-1}(\mathrm{Tk}_{\mathcal{C}|_{[n-1]}}(\tau))$. Thus $T \mapsto \pi^{-1}(T)$ is a surjection on trunks, and π is an isomorphism. □

Definition 3.3.12. Let $\mathcal{C} \subseteq 2^{[n]}$ be a code. The *minimum neuron number* of $\mathcal{C}$ is the smallest m such that $\mathcal{C}$ is isomorphic to a subcode of $2^{[m]}$.

Theorem 3.3.13. *Every code $\mathcal{C}$ is isomorphic to a reduced code. This reduced code is unique up to permutation of neurons, and it is a subcode of $2^{[m]}$ where m is the minimum neuron number of $\mathcal{C}$.*

Proof. By Lemma 3.3.11, we may repeatedly permute and then delete redundant neurons from $\mathcal{C}$ to obtain an isomorphic code with no redundant neurons. We may likewise permute and delete all trivial neurons, obtaining a reduced code that is isomorphic to $\mathcal{C}$. By Corollary 3.3.10 this reduced code is unique up to permutation isomorphism.

Let $\mathcal{D} \subseteq 2^{[n]}$ be a reduced code and let m be the minimum neuron number of $\mathcal{D}$, noting that $m \leq n$. Since $\mathcal{D}$ is reduced, Theorem 3.3.7 implies that n is the number of irreducible trunks in $\mathcal{D}$. Proposition 3.3.6 then implies that $m \geq n$. Thus $m = n$ and the result follows. □

Working with a reduced codes is generally more convenient than working with an arbitrary code—for example, trying to draw a realization of a code is generally more straightforward when working with fewer neurons. We will see in Section 4.2 that when two codes are isomorphic, realizations of one can be built from the other, which makes this result somewhat practical. Note that the proof above provides a rough algorithm for computing a reduced code that is isomorphic to a given code: simply search for redundant and trivial neurons and delete them repeatedly until none are left. This result also yields a combinatorial characterization of the minimum neuron number.

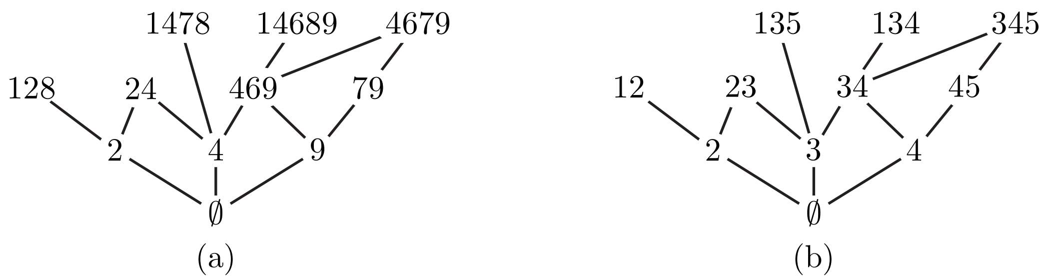


Figure 3.2: (a) A non-reduced code $\mathcal{C}$. (b) An isomorphic reduced code $\mathcal{D}$.

Corollary 3.3.14. *Let $\mathcal{C} \subseteq 2^{[n]}$ be a code. The minimum neuron number of $\mathcal{C}$ is equal to the number of irreducible trunks in $\mathcal{C}$.*

Proof. Let $\mathcal{D} \subseteq 2^{[m]}$ be a reduced code isomorphic to $\mathcal{C}$. Note that the number of irreducible trunks in $\mathcal{C}$ is an isomorphism invariant, so $\mathcal{D}$ has the same number of irreducible trunks. Theorem 3.3.13 says that m is the minimum neuron number of $\mathcal{C}$, while Theorem 3.3.7 tells us that $[m]$ is in bijection with the irreducible trunks in $\mathcal{D}$. This proves the result. □

Example 3.3.15. Here is a slightly more complex example. Consider the code

$$\mathcal{C} = \{\mathbf{14689}, \mathbf{1478}, \mathbf{4679}, \mathbf{128}, \mathbf{24}, 469, 79, 2, 4, 9, \emptyset\} \subseteq 2^{[9]}.$$

This code is not reduced for the following reasons:

- The neurons 3 and 5 are trivial.
- The neuron 1 is redundant to $\{8\}$.
- The neuron 8 is redundant to $\{1\}$.
- The neuron 6 is redundant to $\{4, 9\}$.

This code is isomorphic to the reduced code $\mathcal{D} = \{\mathbf{134}, \mathbf{135}, \mathbf{345}, \mathbf{12}, \mathbf{23}, 34, 45, 2, 3, 4, \emptyset\}$. Figure 3.2 shows that Hasse diagrams of these two codes, and the bijection between the codes implicit in the figure yields the desired isomorphism.

3.4 Morphisms and Intersection Complete Codes

In Chapter 2 we saw that working with intersection complete codes allowed us to leverage a number of useful geometric constructions. Trunks and morphisms likewise admit additional results and structure when we restrict our scope to intersection complete codes. We start with a characterization of intersection complete codes in terms of their trunks.

Lemma 3.4.1. *Let $\mathcal{C}$ be a code. The following are equivalent:*

(i) $\mathcal{C}$ is intersection complete,

(ii) the map $c \mapsto \operatorname{Tk}_{\mathcal{C}}(c)$ is a bijection from $\mathcal{C}$ to the set of nonempty trunks in $\mathcal{C}$, and

(iii) every nonempty trunk in $\mathcal{C}$ has a unique minimal element.

Proof. We first show that item (i) implies item (ii). The map $c \mapsto \operatorname{Tk}_{\mathcal{C}}(c)$ is injective whether or not $\mathcal{C}$ is intersection complete, since c is the unique minimal element of $\operatorname{Tk}_{\mathcal{C}}(c)$. To see that the map is surjective when $\mathcal{C}$ is intersection complete, let T be any nonempty trunk. Then $T = \operatorname{Tk}_{\mathcal{C}}(c)$ where c is the intersection of all codewords in T.

The fact that (ii) implies (iii) follows from the observation made above that c is the unique minimal element of $\operatorname{Tk}_{\mathcal{C}}(c)$, so it remains to show that item (iii) implies item (i). For this, let c_1 and c_2 be codewords in $\mathcal{C}$ and let $\sigma = c_1 \cap c_2$. Then $\operatorname{Tk}_{\mathcal{C}}(\sigma)$ contains a unique minimal codeword c_3. The codeword c_3 contains σ by definition. On the other hand, it is contained in both c_1 and c_2 since it is minimal in $\operatorname{Tk}_{\mathcal{C}}(\sigma)$. Hence $c_3 \subseteq c_1 \cap c_2 = \sigma$. We conclude that $c_3 = \sigma$ and so $\mathcal{C}$ is intersection complete. □

The following theorem explains the relevance of morphisms to intersection complete codes: intersection completeness is preserved in the image of a morphism, and likewise for max-intersection completeness. Perhaps more remarkable is Corollary 3.4.3, which tells us that intersection complete codes are exactly the images of simplicial complexes. Given an intersection complete code $\mathcal{C}$, little is understood about the set of simplicial complexes Δ that admit a surjective morphism $\Delta \to \mathcal{C}$. This may be an interesting area of further study.

Theorem 3.4.2. *The image of an intersection complete code under a morphism is intersection complete. The image of a max-intersection complete code is max-intersection complete.*

Proof. Let $f : \mathcal{C} \to \mathcal{D}$ be a surjective morphism of codes and suppose that $\mathcal{C}$ is intersection complete. By Lemma 3.4.1 every nonempty trunk in $\mathcal{C}$ has a unique minimal element, and it will suffice to prove the same is true of $\mathcal{D}$. Let $T \subseteq \mathcal{D}$ be a nonempty trunk. Then $f^{-1}(T)$ has a unique minimal element. Since morphisms are monotone by Proposition 3.1.7, the same must be true of $f(f^{-1}(T))$. But $f(f^{-1}(T)) = T$, so T has a unique minimal element and $\mathcal{D}$ is intersection complete.

To prove the result for max-intersection complete codes, let $\mathcal{E} \subseteq \mathcal{C}$ be the subcode of $\mathcal{C}$ consisting of maximal codewords in $\mathcal{C}$ and all their intersections. The subcode $f(\mathcal{E}) \subseteq \mathcal{D}$ is intersection complete by the first part of the theorem, and so it suffices to argue that every maximal codeword in $\mathcal{D}$ is contained in $f(\mathcal{E})$. If d is a maximal codeword in $\mathcal{D}$, then $f^{-1}(\mathrm{Tk}_{\mathcal{D}}(d)) = f^{-1}(\{d\})$. This is a trunk in $\mathcal{C}$, and any maximal element of this trunk is a maximal codeword in $\mathcal{C}$ that maps to d. This proves the result. □

Corollary 3.4.3. *A code $\mathcal{C}$ is intersection complete if and only if it is the image of a simplicial complex.*

Proof. Let $\mathcal{C} \subseteq 2^{[n]}$ be a code. If $\mathcal{C}$ is the image of a simplicial complex, then it is intersection complete by Theorem 3.4.2. For the converse, suppose that $\mathcal{C}$ is intersection complete and let $\{c_1, \ldots, c_m\}$ be the codewords of $\mathcal{C}$ that are irreducible with respect to intersection (i.e., no c_i is the intersection of two other codewords not equal to c_i). Observe that every codeword in $\mathcal{C}$ can be written as an intersection of the various c_i. Then let $\Delta = 2^{[m]} \setminus [m]$, and consider the map $f : \Delta \to \mathcal{C}$ defined by $f(\sigma) = \bigcap_{i \in [m] \setminus \sigma} c_i$.

Observe that since $[m] \notin \Delta$, the intersection $\bigcap_{i \in [m] \setminus \sigma} c_i$ defining $f(\sigma)$ is never indexed over the empty set, and so f is a well-defined function from Δ to $\mathcal{C}$. Moreover, f is clearly surjective since every codeword in $\mathcal{C}$ is an intersection of various c_i. We further claim that f is a morphism. To see this, for $j \in [n]$ define $\tau_j = \{i \in [m] \mid j \notin c_i\}$. We claim that $f^{-1}(\mathrm{Tk}_{\mathcal{C}}(j)) = \mathrm{Tk}_{\Delta}(\tau_j)$. Indeed, for any face $\sigma \in \Delta$ we see that $f(\sigma)$ contains j if and only if all c_i with $i \notin \sigma$ have $j \in c_i$, which is equivalent to $\tau_j \subseteq \sigma$. By Proposition 3.1.6 this proves that f is a morphism. □

Example 3.4.4. Consider the intersection complete code $\mathcal{C} = \{\mathbf{123}, 1, 2, 3, \emptyset\}$. The codewords in $\mathcal{C}$ that are irreducible with respect to intersection are $c_1 = 1, c_2 = 2, c_3 = 3$ and $c_4 = 123$. Thus the construction used in Corollary 3.4.3 implies that $\mathcal{C}$ admits a surjective morphism from the simplicial complex $2^{[4]} \setminus \{1234\}$. However, $\mathcal{C}$ admits a surjective morphism from a smaller simplicial complex, namely $2^{[3]}$. The morphism $2^{[3]} \to \mathcal{C}$ is given by sending $123 \mapsto 123, 12 \mapsto 1, 13 \mapsto 2, 23 \mapsto 3$ and all other codewords mapping to the empty codeword.

The Hasse diagrams of the simplicial complexes $2^{[4]} \setminus \{1234\}$ and $2^{[3]}$ are shown in Figure 3.3 along with the Hasse diagram of $\mathcal{C}$. The trunks that determine the surjective morphisms from these simplicial complexes to $\mathcal{C}$ are highlighted in grey.

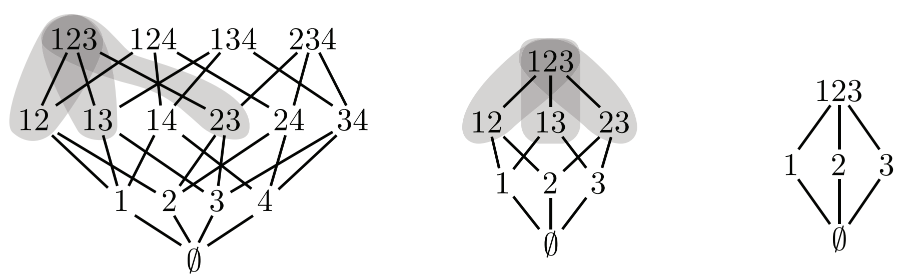


Figure 3.3: Two simplicial complexes, with collections of trunks determining surjective morphisms to the code $\mathcal{C} = \{\mathbf{123}, 1, 2, 3, \emptyset\}$.

3.5 Morphisms and Neural Rings

As mentioned previously, the class of codes together with morphisms forms a category. In this section we will use this category to explain how codes and morphisms can be interpreted algebraically in the context of neural rings. We will not make use of any advanced category theory: we require only the notions of (contravariant) functors, and equivalence of categories.

For some related work studying morphisms and neural rings, see [CK20] in which the authors provide a characterization of neural rings and the monomial maps between them.

Definition 3.5.1. We will let **Code** denote the category of codes together with morphisms.

Our main result is Theorem 3.5.4, which states that when we equip neural rings with a certain type of ring homomorphism, the category they form is equivalent to **Code**. To begin, we review some of the basic definitions related to neural rings.

Let $\mathbb{F}_2$ be the two element field. Recall that any polynomial $p(x_1, \dots, x_n) \in \mathbb{F}_2[x_1, \dots, x_n]$ defines a function $p : 2^{[n]} \to \mathbb{F}_2$, where evaluation of p at a codeword c is given by replacing x_i by 1 if $i \in c$, and by 0 otherwise.

Definition 3.5.2 ([CIVCY13])**.** Let $\mathcal{C} \subseteq 2^{[n]}$ be a code. The *vanishing ideal* of $\mathcal{C}$ is

$$I_{\mathcal{C}} := \{p \in \mathbb{F}_2[x_1, \dots, x_n] \mid p(c) = 0 \text{ for all } c \in \mathcal{C}\} \subseteq \mathbb{F}_2[x_1, \dots, x_n].$$

The *neural ring* of $\mathcal{C}$ is the quotient ring $R_{\mathcal{C}} := \mathbb{F}_2[x_1, \dots, x_n]/I_{\mathcal{C}}$, together with the coordinate functions $x_i \in R_{\mathcal{C}}$.

Every neural ring can be regarded simply as a finite Boolean algebra, but the additional information of which elements correspond to coordinate functions allows us to recover information about the associated code. For example, we can recover the number of neurons, which is simply the number of coordinate functions. More strongly, [CIVCY13] proves that the neural ring uniquely determines its associated code, and vice versa.

A useful fact about the neural ring of a code $\mathcal{C}$ is that it is isomorphic to the ring of functions from $\mathcal{C}$ to $\mathbb{F}_2$. Thus to prove that two elements of a neural ring are equal, it suffices to show that they are the same when regarded as functions from $\mathcal{C}$ to $\mathbb{F}_2$.

Before presenting our main result, we require a few more definitions. For any $\sigma \subseteq [n]$, the monomial $\prod_{i\in\sigma} x_i$ will be denoted x_σ. For any $\sigma \subseteq [n]$, we define the *indicator function* of σ as

$$\rho_\sigma := \prod_{i\in\sigma} x_i \prod_{j\notin\sigma}(1 - x_j) \in \mathbb{F}_2[x_1, \dots, x_n].$$

Note that the function ρ_σ has the property that it evaluates to 1 only at σ. Finally, we must equip the class of neural rings with a class of morphisms. We will use monomial maps, defined below.

Definition 3.5.3. Let $R_\mathcal{C}$ and $R_\mathcal{D}$ be neural rings with coordinate functions $\{x_1, \ldots, x_n\}$ and $\{y_1, \ldots, y_m\}$ respectively. A *monomial map* from $R_\mathcal{D}$ to $R_\mathcal{C}$ is a ring homomorphism $\phi : R_\mathcal{D} \to R_\mathcal{C}$ with the property that for every nonempty $\tau \subseteq [m]$, either $\phi(y_\tau) = 0$, or there exists nonempty $\sigma \subseteq [n]$ such that $\phi(y_\tau) = x_\sigma$.

Theorem 3.5.4. *Let* **NRing** *be the category whose objects are neural rings, and whose morphisms are monomials maps. There is a contravariant equivalence of categories* R : **Code** $\to$ **NRing** *given by associating a code to its neural ring, and associating a morphism* $f : \mathcal{C} \to \mathcal{D}$ *to the ring homomorphism* $R(f) : R_\mathcal{D} \to R_\mathcal{C}$ *given by precomposition with* f.

Proof. We will let $f^* : R_\mathcal{D} \to R_\mathcal{C}$ denote $R(f)$ for any morphism $f : \mathcal{C} \to \mathcal{D}$. We start by showing that R gives us a well defined function from morphisms $\mathcal{C} \to \mathcal{D}$ to monomial maps $R_\mathcal{D} \to R_\mathcal{C}$. We must show that if $f : \mathcal{C} \to \mathcal{D}$ is a morphism of codes, then $f^* : R_\mathcal{D} \to R_\mathcal{C}$ is a monomial map. It will suffice to show for all $j \in [m]$ that $f^*(y_j) = 0$ or there exists nonempty $\sigma \subseteq [n]$ with $f^*(y_j) = x_\sigma$.

To this end, suppose that y_j is such that $f^*(y_j) \neq 0$. Then observe that the codewords $c \in \mathcal{C}$ where $f^*(y_j)$ evaluates to 1 are exactly those in $f^{-1}(\mathrm{Tk}_\mathcal{D}(j))$. Indeed, we have the following chain of equivalences:

$$f^*(y_j)(c) = 1 \quad \Leftrightarrow \quad (y_j \circ f)(c) = 1 \quad \Leftrightarrow \quad y_j(f(c)) = 1 \quad \Leftrightarrow \quad j \in f(c) \quad \Leftrightarrow \quad c \in f^{-1}(\mathrm{Tk}_\mathcal{D}(j)).$$

If the trunk $f^{-1}(\mathrm{Tk}_\mathcal{D}(j))$ is empty, then $f^*(y_j) = 0$. Otherwise, there exists nonempty $\sigma \subseteq [n]$ such that $f^{-1}(\mathrm{Tk}_\mathcal{D}(j)) = \mathrm{Tk}_\mathcal{C}(\sigma)$. In this case, $f^*(y_j) = x_\sigma$ as functions, since $f^*(y_j)$ is equal to 1 exactly on those codewords that contain σ. Thus f^* is a monomial map.

So far we have shown that R is a functor. To show that it is an equivalence of categories we must show that it is faithful and full. To prove that R is faithful, suppose f and g are two distinct morphisms from a code $\mathcal{C}$ to a code $\mathcal{D}$. We must show that f^* and g^* are distinct ring homomorphisms from $R_{\mathcal{D}}$ to $R_{\mathcal{C}}$. To this end let $c \in \mathcal{C}$ be such that $f(c) \neq g(c)$. Then consider the indicator function $\rho_{f(c)} : \mathcal{D} \to \mathbb{F}_2^n$, recalling that this function evaluates to 1 on a codeword if and only if that codeword is equal to $f(c)$. Then consider $f^*(\rho_c)$ and $g^*(\rho_c)$. The function $f^*(\rho_c)$ takes c to 1, while $g^*(\rho_c)$ takes it to 0. This proves that f^* and g^* are distinct ring homomorphisms, and so the map from $\mathrm{Hom}_{\mathbf{Code}}(\mathcal{C}, \mathcal{D})$ to $\mathrm{Hom}_{\mathbf{NRing}}(R_{\mathcal{D}}, R_{\mathcal{C}})$ induced by R is injective as desired.

It remains to show that R is full. Let $\phi : R_{\mathcal{D}} \to R_{\mathcal{C}}$ be a monomial map. We must show $\phi = f^*$ for some morphism $f : \mathcal{C} \to \mathcal{D}$. We construct the appropriate morphism f by specifying a set of proper trunks that determine it, as in Definition 3.2.1. Every y_j maps to either zero, or some monomial x_{σ_j} where $\sigma_j \subseteq [n]$ is nonempty. Let $f : \mathcal{C} \to 2^{[m]}$ be the morphism determined by the trunks $\{T_1, \ldots, T_m\}$ where

$$T_j = \begin{cases} \emptyset & \text{if } \phi(y_j) = 0, \\ \mathrm{Tk}_{\mathcal{C}}(\sigma_j) & \text{if } \phi(y_j) = x_{\sigma_j}. \end{cases}$$

Observe that all of these trunks are proper since all σ_j are nonempty. We claim that $f(\mathcal{C}) \subseteq \mathcal{D}$, so we can regard f as a morphism $\mathcal{C} \to \mathcal{D}$. Let c be a codeword in $\mathcal{C}$, and consider the indicator function $\rho_{f(c)} \in \mathbb{F}_2[x_1, \ldots, x_m]$, which is 1 on $f(c)$ and zero everywhere else. We can then consider $\rho_{f(c)}$ as an element of $R_{\mathcal{D}} = \mathbb{F}_2[x_1, \ldots, x_m]/I_{\mathcal{D}}$. Note that

$$\phi(\rho_{f(c)}) = \phi\Bigg(\prod_{i \in f(c)} y_i \prod_{j \notin f(c)} (1 - y_j) \Bigg) = \prod_{i \in f(c)} x_{\sigma_i} \prod_{j \notin f(c)} (1 - x_{\sigma_j}).$$

Now, $\phi(\rho_{f(c)})$ will yield 1 when evaluated at c since $x_{\sigma_i}(c) = 1$ if and only if $c \in T_i$, which happens if and only if $i \in f(c)$. We conclude that $\rho_{f(c)}$ is nonzero in $R_{\mathcal{D}}$ and so $f(c) \in \mathcal{D}$. Thus we can restrict f to a morphism from $\mathcal{C}$ to $\mathcal{D}$.

Finally, we claim that $f^* : R_{\mathcal{D}} \to R_{\mathcal{C}}$ is the same monomial map as ϕ. It suffices to argue that $f^*(y_j) = \phi(y_j)$ for all $j \in [m]$. Observe that $f^*(y_j) = 0$ if and only if T_j is empty, which implies that $\phi(y_j) = 0$. This leaves the case that $f^*(y_j) \neq 0$, or equivalently $T_j \neq \emptyset$. In this case, we need only argue that $f^*(y_j)$ is equal to 1 when evaluated at some $c \in \mathcal{C}$ if and only if x_{σ_j} is 1 when evaluated at c. But the latter condition is equivalent to saying that

$c \in T_j$, which is equivalent to the statement that $f^*(y_j)(c) = 1$ since $f^*(y_j)(c) = y_j(f(c))$. Therefore $f^* = \phi$, and the functor R is full as desired. We conclude that R is a contravariant equivalence of categories. □

3.6 Morphisms and Category Theory

In this section we turn to more abstract properties of the category **Code**. Our main goal is to prove Theorem 3.6.7, which states that **Code** is finitely bicomplete. In other words, all finite diagrams in this category have both limits and colimits. This result is meant to provide some evidence that our framework of morphisms is relatively sound, at least from a theoretical perspective. For the reader who is less experienced with category theory, we recommend [AHS06], which is available online and includes definitions for the concepts that we reference in this section.

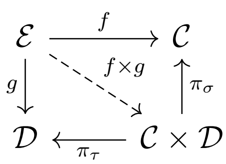
To prove that **Code** is finitely bicomplete it suffices to exhibit terminal and initial objects, together with four constructions: pairwise products, pairwise coproducts, equalizers, and coequalizers. We provide all of these in sequence below. Except for products, none of these will appear in later sections or chapters. Products make a reappearance in Proposition 4.3.8, since they have a constructive geometric interpretation.

Definition 3.6.1. Let $\mathcal{C} \subseteq 2^\sigma$ and $\mathcal{D} \subseteq 2^\tau$ be codes, and assume without loss of generality that σ and τ are disjoint. The *product* of $\mathcal{C}$ and $\mathcal{D}$ is the code

$$\mathcal{C} \times \mathcal{D} := \{c \cup d \mid c \in \mathcal{C}, d \in \mathcal{D}\} \subseteq 2^{\sigma \cup \tau}.$$

Proposition 3.6.2 (Existence of Products)**.** *The product defined in Definition 3.6.1, together with the restriction morphisms $\pi_\sigma : \mathcal{C} \times \mathcal{D} \to \mathcal{C}$ and $\pi_\tau : \mathcal{C} \times \mathcal{D} \to \mathcal{D}$, is the product of $\mathcal{C}$ and $\mathcal{D}$ in the category* **Code***.*

Proof. Let $\mathcal{E}$ be any code and let $f : \mathcal{E} \to \mathcal{C}$ and $g : \mathcal{E} \to \mathcal{D}$ be morphisms. Define a function $f \times g : \mathcal{E} \to \mathcal{C} \times \mathcal{D}$ by $(f \times g)(e) = f(e) \cup g(e)$. Note that this is the unique function such that the following diagram commutes:



It remains to show that $f \times g$ is a morphism. Let T be a proper trunk in $\mathcal{C} \times \mathcal{D}$. If $T = \emptyset$, then its preimage under $f \times g$ will be empty, and hence a trunk in $\mathcal{E}$. If T is nonempty, we may write it as $\mathrm{Tk}_{\mathcal{C}\times\mathcal{D}}(\sigma' \cup \tau')$ where $\sigma' \subseteq \sigma$, $\tau' \subseteq \tau$, and at least one of σ' and τ' is nonempty. One may then compute that

$$(f \times g)^{-1}(T) = f^{-1}(\mathrm{Tk}_{\mathcal{C}}(\sigma')) \cap g^{-1}(\mathrm{Tk}_{\mathcal{D}}(\tau')).$$

Since at least one of σ' and τ' is nonempty, at least one of the terms in the intersection above will be a proper trunk, and so the preimage of T under $f \times g$ is a proper trunk in $\mathcal{E}$ as desired. This proves the result. $\square$

Definition 3.6.3. Let $\mathcal{C} \subseteq 2^{\sigma}$ and $\mathcal{D} \subseteq 2^{\tau}$ be codes, and assume without loss of generality that σ and τ are disjoint. For every pair (S, T) of proper trunks with $S \subseteq \mathcal{C}$ and $T \subseteq \mathcal{D}$, let $i_{(S,T)}$ be a new neuron. The *coproduct* of $\mathcal{C}$ and $\mathcal{D}$ is the code

$$\mathcal{C} \sqcup \mathcal{D} := \{c \cup \{i_{(S,T)} \mid c \in S\} \mid c \in \mathcal{C}\} \cup \{d \cup \{i_{(S,T)} \mid d \in T\} \mid d \in \mathcal{D}\}$$

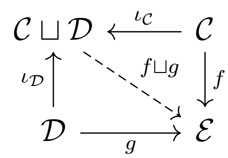
on the set of neurons $\sigma \cup \tau \cup \{i_{(S,T)} \mid S \subseteq \mathcal{C}, T \subseteq \mathcal{D}$ proper trunks.$\}$.

Proposition 3.6.4 (Existence of Coproducts)**.** *The coproduct defined in Definition 3.6.3, together with the morphisms $\iota_{\mathcal{C}} : \mathcal{C} \to \mathcal{C} \sqcup \mathcal{D}$ and $\iota_{\mathcal{D}} : \mathcal{D} \to \mathcal{C} \sqcup \mathcal{D}$ defined by $\iota_{\mathcal{C}}(c) = c \cup \{i_{(S,T)} \mid c \in S\}$ and $\iota_{\mathcal{D}}(d) = d \cup \{i_{(S,T)} \mid d \in T\}$, is the coproduct of $\mathcal{C}$ and $\mathcal{D}$ in the category* **Code**.

Proof. Let $\mathcal{E}$ be any code and let $f : \mathcal{C} \to \mathcal{E}$ and $g : \mathcal{D} \to \mathcal{E}$ be morphisms. Define a function $f \sqcup g : \mathcal{C} \sqcup \mathcal{D} \to \mathcal{E}$ by

$$(f \sqcup g)(a) = \begin{cases} f(c) & \text{if } a = c \cup \{i_{(S,T)} \mid c \in S\}, \\ g(d) & \text{if } a = d \cup \{i_{(S,T)} \mid d \in T\}. \end{cases}$$

Note that this function is well defined because the only case in which the two conditions defining $f \sqcup g$ overlap is $c = d = \emptyset$, and under both conditions we have $(f \sqcup g)(\emptyset) = \emptyset$. Moreover, observe that $f \sqcup g$ is the unique function such that the following diagram commutes:



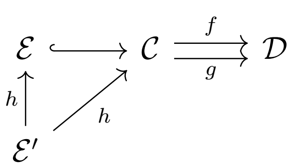
It remains to show that $f \sqcup g$ is a morphism. Let Q be a proper trunk in $\mathcal{E}$, let $T' = f^{-1}(Q)$, and let $S' = g^{-1}(Q)$. One may then compute that

$$(f \sqcup g)^{-1}(Q) = \{\iota_{\mathcal{C}}(c) \mid c \in S'\} \cup \{\iota_{\mathcal{D}}(d) \mid d \in T'\}.$$

This is exactly $\mathrm{Tk}_{\mathcal{C} \sqcup \mathcal{D}}(i_{(S',T')})$, which is a proper trunk in $\mathcal{C} \sqcup \mathcal{D}$. Thus $f \sqcup g$ is a morphism and the result follows. □

Proposition 3.6.5 (Existence of Equalizers)**.** *Let $f, g : \mathcal{C} \to \mathcal{D}$ be morphisms. Let $\mathcal{E} = \{c \in \mathcal{C} \mid f(c) = g(c)\}$. The code $\mathcal{E}$ together with the inclusion $\mathcal{E} \hookrightarrow \mathcal{C}$ is the equalizer of f and g in the category* **Code**.

Proof. Let $\mathcal{E}'$ be any code and let $h : \mathcal{E}' \to \mathcal{C}$ be such that $f \circ h = g \circ h$. Observe that $h(\mathcal{E}') \subseteq \mathcal{E}$, and that the restricted morphism $h : \mathcal{E}' \to \mathcal{E}$ is the unique morphism making the following diagram commute:



This proves the result. □

Proposition 3.6.6 (Existence of Coequalizers)**.** *Let $f, g : \mathcal{C} \to \mathcal{D}$ be morphisms. Let $\{T_1, \ldots, T_m\}$ be the collection of all proper nonempty trunks in $\mathcal{D}$ such that $f(c) \in T_j$ if and only if $g(c) \in T_j$ for all $c \in \mathcal{C}$ and $j \in [m]$. Let $h : \mathcal{D} \to 2^{[m]}$ be the morphism determined by $\{T_1, \ldots, T_m\}$, and let $\mathcal{E} \subseteq 2^{[m]}$ be the image of h. The code $\mathcal{E}$ together with the morphism $h : \mathcal{D} \to \mathcal{E}$ is a coequalizer of f and g in the category* **Code**.

Proof. Let $\mathcal{E}' \subseteq 2^{[n]}$ be any code and let $h' : \mathcal{D} \to \mathcal{E}'$ be such that $h' \circ f = h' \circ g$. Let $\{S_1, \ldots, S_n\}$ be the set of proper trunks that determine h', and observe that each S_i has the property that $f(c) \in S_i$ if and only if $g(c) \in S_i$. In other words, every S_i is equal to some T_j. Up to permutation of $[m]$ we may assume that $n \leq m$ and $S_i = T_i$ for all $i \in [n]$.

Then consider the restriction morphism $\pi_{[n]} : \mathcal{E} \to 2^{[n]}$. One may compute that $\pi_{[n]}(h(d)) = h'(d)$ for any codeword d in $\mathcal{D}$. Thus $\pi_{[n]}$ restricts to a morphism $\mathcal{E} \to \mathcal{E}'$, making the following diagram commute:

$$\begin{array}{ccccc} \mathcal{C} & \overset{f}{\underset{g}{\rightrightarrows}} & \mathcal{D} & \xrightarrow{h} & \mathcal{E} \\ & & & \searrow^{h'} & \downarrow \pi_{[n]} \\ & & & & \mathcal{E}' \end{array}$$

In fact $\pi_{[n]}$ is the unique function making this diagram commute. This proves the result. □

Theorem 3.6.7. *The category* **Code** *is finitely bicomplete.*

Proof. We have seen in the propositions above that **Code** has products, coproducts, equalizers, and coequalizers. To show that **Code** is finitely bicomplete, we need only exhibit an initial and terminal object. We claim that the code $\mathcal{C} = \{\emptyset\}$ suffices for both (i.e. $\mathcal{C}$ is a zero object). Indeed, if $\mathcal{D}$ is any code then the only morphism from $\mathcal{D}$ to $\mathcal{C}$ is the constant map, and the only morphism $\mathcal{C} \to \mathcal{D}$ is the map sending $\emptyset \mapsto \emptyset$. □

Chapter 4
MINORS OF CODES

In this chapter we bring morphisms to bear on our main problem of interest: classifying the open and closed embedding dimensions of codes. In particular, we use morphisms to define a totally combinatorial notion of minors for codes, analogous to minors of graphs or matroids. Our notion of minor does not generalize these traditional minors, but it does provide a framework in which to investigate convexity. As we will see in Theorem 4.3.4, essentially every "geometric realizability" property of codes is closed under taking minors.

4.1 Definitions: Minors of Codes

We have already seen in results like Theorem 3.3.9 and Theorem 3.4.2 that surjective morphisms preserve important structure in codes. Minors are defined with this in mind. Although the definition below is completely combinatorial, we will see in Section 4.2 that it has important geometric consequences.

Definition 4.1.1. Let $\mathcal{C}$ and $\mathcal{D}$ be codes. We say that $\mathcal{D}$ is a *minor* of $\mathcal{C}$, and write $\mathcal{D} \leq \mathcal{C}$, if there exists a surjective morphism $f : \mathcal{C} \to \mathcal{D}$. A *proper minor* of $\mathcal{C}$ is a minor that is not isomorphic to $\mathcal{C}$.

Remark 4.1.2. When we defined minors in [Jef20], we also defined any trunk in a code to be a minor of the code. Our conventions in this work allow us to omit this, significantly simplifying the definition of minors. The following proposition tells us that this omission does not actually impact the definition.

Proposition 4.1.3. *Let $\mathcal{C}$ be a code and let T be a trunk in $\mathcal{C}$. Then the code $T \cup \{\emptyset\}$ is a minor of $\mathcal{C}$.*

Proof. Let $\mathcal{D} = T \cup \{\emptyset\}$, and consider the function $f : \mathcal{C} \to \mathcal{D}$ given by

$$f(c) = \begin{cases} c & \text{if } c \in T, \\ \emptyset & \text{otherwise.} \end{cases}$$

Clearly this function is surjective, and for every $i \in [n]$ one may compute that $f^{-1}(\mathrm{Tk}_{\mathcal{D}}(i)) = T \cap \mathrm{Tk}_{\mathcal{C}}(i)$, which is a proper trunk in $\mathcal{C}$. By Proposition 3.1.6 f is a morphism, and we conclude that $\mathcal{D}$ is a minor of $\mathcal{C}$ as desired. □

Example 4.1.4. Consider the code $\mathcal{C} = \{\mathbf{12}, \mathbf{13}, \emptyset\}$. We saw in Example 3.1.4 that there is a surjective morphism from $\mathcal{C}$ to the code $\mathcal{D} = \{\mathbf{1}, \mathbf{2}, \emptyset\}$. Thus $\mathcal{D}$ is a minor of $\mathcal{C}$. In fact, we claim that up to isomorphism $\mathcal{C}$ has five total minors, which are:

$$\{\mathbf{12}, \mathbf{13}, \emptyset\}, \qquad \{\mathbf{1}, \mathbf{2}, \emptyset\}, \qquad \{\mathbf{12}, \mathbf{2}, \emptyset\}, \qquad \{\mathbf{1}, \emptyset\}, \qquad \{\emptyset\}.$$

Note that any minor of $\mathcal{C}$ has no more than three codewords, and the above set of codes consists of all codes with no more than three codewords (up to isomorphism). Thus we only have to exhibit surjective morphisms from $\mathcal{C}$ to any of the above codes. Appropriate morphisms can be found by hand: for example, the morphism $\mathcal{C} \to \{\mathbf{12}, \mathbf{2}, \emptyset\}$ is given by deleting neuron 3.

Interestingly, $\mathcal{C}$ is the unique code (up to isomorphism) with the property that every code with no more than 3 codewords is a minor of $\mathcal{C}$. We will generalize this phenomenon in Proposition 4.3.7.

4.2 Minors and Realizations

Minors allow us to replace codes with new, modified codes. The following definition allows us to replace realizations with new, modified realizations. Theorem 4.2.2 tells us that these two modification processes are actually analogous. In other words, derived realizations provide a geometric interpretation of code minors.

Definition 4.2.1. Let $\mathcal{U} = \{U_1, \dots, U_n\}$ be a (not necessarily open, closed, or convex) realization of a code $\mathcal{C} \subseteq 2^{[n]}$. A *derived realization* of $\mathcal{U}$ is a collection $\mathcal{V} = \{V_j := U_{\sigma_j} \mid j \in [m]\}$ for some choice of nonempty index sets $\sigma_1, \dots, \sigma_m$.

Theorem 4.2.2 (Fundamental Theorem of Minors)**.** *Let $\mathcal{U} = \{U_1, \dots, U_n\}$ be a (not necessarily convex, open, or closed) realization of a code $\mathcal{C} \subseteq 2^{[n]}$. Let $\{\mathrm{Tk}_{\mathcal{C}}(\sigma_1), \dots, \mathrm{Tk}_{\mathcal{C}}(\sigma_m)\}$ be a collection of proper trunks in $\mathcal{C}$, let $f : \mathcal{C} \to 2^{[m]}$ be the morphism determined by these trunks, and let $\mathcal{D}$ be the minor $f(\mathcal{C})$ of $\mathcal{C}$. Then the derived realization $\mathcal{V} = \{V_j := U_{\sigma_j} \mid j \in [m]\}$ is a realization of $\mathcal{D}$.*

Proof. We must show for all nonempty $\tau \subseteq [m]$ that $\mathcal{V}^\tau \neq \emptyset$ if and only if $\tau = f(c)$ for some codeword c in $\mathcal{C}$. Fix a nonempty index set $\tau \subseteq [m]$. Then $\mathcal{V}^\tau$ is nonempty if and only if there exists a point p in the stimulus space that lies in exactly the V_j with $j \in \tau$. Such a point p would belong to an atom in the realization $\mathcal{U}$, and by construction this must be the atom of a codeword c in $\mathcal{C}$ with the property that $\sigma_j \subseteq c$ exactly when $j \in \tau$. Such codewords are exactly those with the property that $f(c) = \tau$. Thus $\tau \in f(\mathcal{C}) = \mathcal{D}$ if and only if $\tau \in \mathrm{code}(\mathcal{V})$. This proves the result. □

Example 4.2.3. Consider the simplicial complex code $\mathcal{C} = \{\mathbf{123}, \mathbf{14}, \mathbf{34}, 12, 13, 23, 1, 2, 3, 4, \emptyset\}$. Let us choose $\sigma_1 = \{1,2\}, \sigma_2 = \{1,3\}, \sigma_3 = \{2,3\}, \sigma_4 = \{3\}$, and $\sigma_5 = \{4\}$, and for $i \in [5]$ define $T_i = \mathrm{Tk}_{\mathcal{C}}(\sigma_i)$. We see that

$$T_1 = \{123, 12\} \qquad T_2 = \{123, 13\} \qquad T_3 = \{123, 23\}$$
$$T_5 = \{14, 34, 4\} \qquad T_4 = \{123, 34, 13, 23, 3\}$$

Let $f : \mathcal{C} \to 2^{[5]}$ be the morphism determined by the collection of trunks $\{T_1, T_2, T_3, T_4, T_5\}$. We can compute that

$$f(\mathcal{C}) = \{\mathbf{1234}, \mathbf{45}, 24, 34, 1, 4, 5, \emptyset\}.$$

We saw in Example 2.1.2 that $\mathcal{C}$ has an open realization $\mathcal{U} = \{U_1, U_2, U_3, U_4\}$ in $\mathbb{R}^2$. Figure 4.1 shows this realization, and the derived realization $\mathcal{V} = \{V_j := U_{\sigma_j} \mid j \in [5]\}$ of $f(\mathcal{C})$ as guaranteed by Theorem 4.2.2.

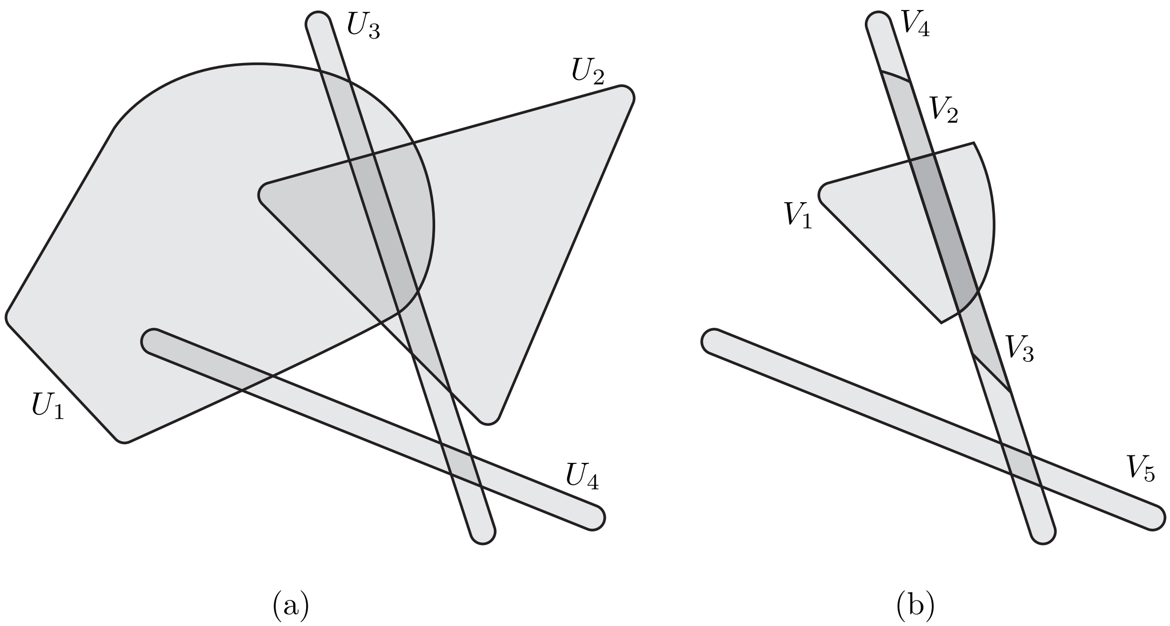


Figure 4.1: (a) A realization of $\mathcal{C}$. (b) The derived realization of $f(\mathcal{C})$.

4.3 $\mathbf{P_{Code}}$*, the Poset of Code Minors*

Note that the relation "is a minor of" is not a partial order on the class of all codes. The only property that does not hold is antisymmetry: isomorphic codes are minors of one another, but need not be equal. To create an appropriate partially ordered set of code minors, we should thus work with isomorphism classes of codes.

Definition 4.3.1. For a code $\mathcal{C}$, we let $[\mathcal{C}]$ denote the isomorphism class of $\mathcal{C}$ in **Code**. If $[\mathcal{C}]$ and $[\mathcal{D}]$ are isomorphism classes of codes, we write $[\mathcal{D}] \leq [\mathcal{C}]$ and say that $[\mathcal{D}]$ is a *minor* of $[\mathcal{C}]$ whenever $\mathcal{D}$ is a minor of $\mathcal{C}$.

Definition 4.3.2. The *poset of code minors* is the set of isomorphism classes of codes, together with the relation $\leq$, and is denoted $\mathbf{P_{Code}}$.

Proposition 4.3.3. $\mathbf{P_{Code}}$ *is a partially ordered set.*

Proof. Note that since isomorphisms are surjective, the relation $\leq$ is well-defined on isomorphism classes of codes. Since the identity function is a morphism, the relation $\leq$ is also reflexive. To prove antisymmetry, let $\mathcal{C}$ and $\mathcal{D}$ be codes with $[\mathcal{C}] \leq [\mathcal{D}]$ and $[\mathcal{D}] \leq [\mathcal{C}]$. Then there exist surjective morphisms $f : \mathcal{C} \to \mathcal{D}$ and $g : \mathcal{C} \to \mathcal{D}$. Proposition 3.3.8 implies that $\mathcal{C}$ and $\mathcal{D}$ have the same number of trunks, and Theorem 3.3.9 implies that f is an isomorphism. Thus $[\mathcal{C}] = [\mathcal{D}]$ as desired.

For transitivity, suppose that $[\mathcal{E}] \leq [\mathcal{D}] \leq [\mathcal{C}]$. Then there exist surjective morphisms $f : \mathcal{C} \to \mathcal{D}$ and $g : \mathcal{D} \to \mathcal{E}$. The composition $g \circ f : \mathcal{C} \to \mathcal{E}$ is a surjective morphism, and so $[\mathcal{E}] \leq [\mathcal{C}]$. Thus the relation $\leq$ is a partial order. □

We will often abuse notation and refer to codes, rather than isomorphism classes, inside $\mathbf{P}_{\mathbf{Code}}$. When we are speaking of isomorphism-invariant properties of codes this does not cause any ambiguity. The following theorems fully explain the relevance of minors to the study of convexity and other properties of codes. Specific families of examples and further content in this direction will primarily appear in Sections 4.4, 5.6, and 5.8.

Theorem 4.3.4. *Fix $d \in \mathbb{N} \cup \{\infty\}$. The following classes of codes are minor-closed. In other words, each of these classes of codes forms a downset in $\mathbf{P}_{\mathbf{Code}}$.*

(i) Codes with $\operatorname{odim}(\mathcal{C}) < d$.

(ii) Codes with $\operatorname{cdim}(\mathcal{C}) < d$.

(iii) Codes with an (open or closed) non-degenerate realization in dimension less than d.

(iv) Codes with an (open or closed) good cover realization in dimension less than d.

(v) Intersection complete codes.

(vi) Max-intersection complete codes.

Proof. For item (i), let $\mathcal{U}$ be an open convex realization of a code $\mathcal{C}$ in dimension less than d. Note that any derived realization of $\mathcal{U}$ is also open and convex. Theorem 4.2.2 implies that every minor of $\mathcal{C}$ has such a derived realization, proving the result. The same holds for (ii), (iii), and (iv): one needs only observe that derived realizations preserve convexity, closedness, non-degeneracy, and good covers.

For items (v) and (vi), Theorem 3.4.2 proves the result. □

Very roughly, Theorem 4.3.4 tells us that we can think of $\mathbf{P}_{\mathbf{Code}}$ as being stratified into different "layers" consisting of codes with different embedding dimensions. In the simplest case, we could look at just two layers: open convex codes, and codes that are not open convex (with the former lying "below" the latter). Analogously, one could imagine codes with finite closed embedding dimension lying "below" codes with closed embedding dimension equal to infinity. Figure 4.2 provides a very informal illustration of this situation. The figure is misleading in a number of ways (in particular, there are not finitely many codes which are both open and closed convex), but it can provide a useful intuitive guide. For a more detailed and technically correct illustration of $\mathbf{P}_{\mathbf{Code}}$, see Figure 4.4 at the end of this chapter.

Corollary 4.3.5. *The following code invariants are isomorphism invariants, and are monotone on $\mathbf{P}_{\mathbf{Code}}$. In other words, if $\mathcal{D}$ is a minor of $\mathcal{C}$, then the value of each invariant on $\mathcal{C}$ is at least as large as its value on $\mathcal{D}$.*

(i) Open embedding dimension.

(ii) Closed embedding dimension.

(iii) Non-degenerate embedding dimension.

(iv) Good cover embedding dimension.

Proof. If any of these invariants was not monotone, we would obtain codes $\mathcal{C}$ and $\mathcal{D}$ with $[\mathcal{D}] \le [\mathcal{C}]$ and the invariant taking a larger value on $\mathcal{D}$. This would contradict the corresponding item in Theorem 4.3.4. □

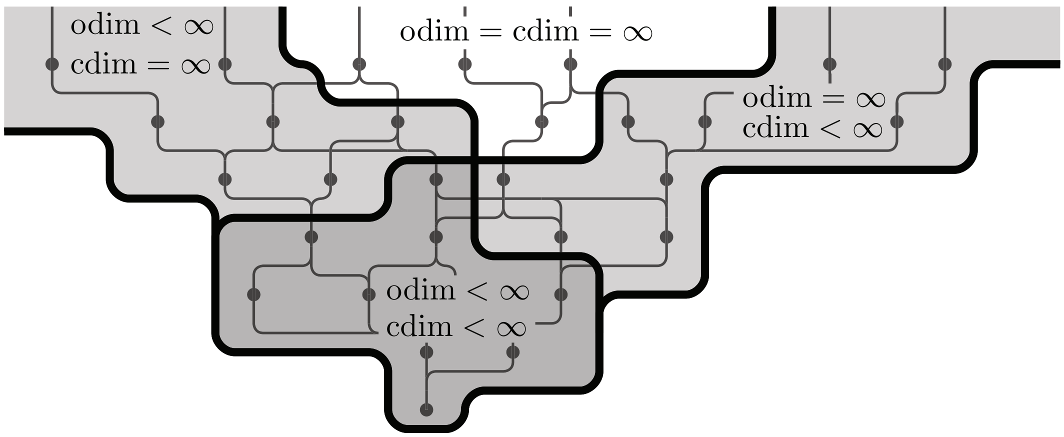


Figure 4.2: An informal illustration of $\mathbf{P}_{\mathbf{Code}}$ stratified by open and closed convexity. Each dot represents a code, light grey lines represent relations in the poset, and thick black lines represent the "boundaries" between different classes of codes.

Remark 4.3.6. Notice that Theorem 4.2.2 implies that there are many more minor-closed "realizability" properties. Essentially, "realizability in dimension less than d by a family of sets $\mathcal{F}$" is a minor-closed property as long as the family $\mathcal{F}$ is closed under intersections. The following are potentially interesting candidates for the family $\mathcal{F}$, and could be a subject of future work:

- (Open or closed) axis-parallel boxes,
- Polytopes whose normal vectors come from a fixed set of vectors,
- Polytopes generally, and
- Affine subspaces.

Finally, it is worth noting that codes with no more than k maximal codewords also form a minor-closed family, since every maximal codeword has a maximal preimage under a surjective morphism. This family may warrant further investigation for small values of k, given recent work in [JSS20] that classifies the convexity of codes with no more than three maximal codewords.

Below we prove a related and somewhat surprising fact: the minor-closed family of codes with no more than k total codewords has a unique maximal element $\mathcal{M}_k$ in $\mathbf{P_{Code}}$. Note that this generalizes Example 4.1.4, which treated the case $k = 3$. One can think of the code $\mathcal{M}_k$ as analogous to a discrete topological space on k points.

Also note that an analogous result does *not* hold for the family of codes with no more than k maximal codewords: both $\{\mathbf{12}, \mathbf{13}, \emptyset\}$ and $\{\mathbf{12}, \mathbf{13}, 1, \emptyset\}$ have two maximal codewords, but neither is a minor of the other.

Proposition 4.3.7. *For any natural number k, define a code*

$$\mathcal{M}_k := \{[k] \setminus \{i\} \mid i \in [k-1]\} \cup \{\emptyset\} \subseteq 2^{[k]}.$$

If $\mathcal{C}$ is any code with no more than k codewords, then $\mathcal{C}$ is a minor of $\mathcal{M}_k$. In other words, $\mathcal{M}_k$ is the unique maximal code among all codes with no more than k codewords.

Proof. For $i \in [k-1]$, define $m_i = [k] \setminus \{i\}$. Note that for any $\sigma \subseteq [k-1]$, the collection of codewords $\{m_i \mid i \in \sigma\}$ is a proper trunk in $\mathcal{M}_k$, namely it is the trunk of $[k] \setminus \sigma$. Now let $\mathcal{C}$ be any code with no more than k codewords. Label the nonempty codewords of $\mathcal{C}$ by $c_1, \ldots, c_l$, where $l \leq k-1$. Define a function $f : \mathcal{M}_k \to \mathcal{C}$ by $f(m_i) = c_i$ for all $i \in [l]$, and $f(m) = \emptyset$ for all other $m \in \mathcal{M}_k$. Observe that f is surjective, and the preimage of any proper trunk in $\mathcal{C}$ is a collection of various m_i with $i \in [l]$. Since any subset of the m_i is a proper trunk in $\mathcal{M}_k$, the function f is a morphism. This proves that $\mathcal{C}$ is a minor of $\mathcal{M}_k$, as desired. □

To conclude this section, we use minors to prove a result regarding products of codes (see Definition 3.6.1).

Proposition 4.3.8. *Let $\mathcal{C} \subseteq 2^{\sigma}$ and $\mathcal{D} \subseteq 2^{\tau}$ be codes, where σ and τ are disjoint. Then*

$$\max\{\mathrm{odim}(\mathcal{C}), \mathrm{odim}(\mathcal{D})\} \leq \mathrm{odim}(\mathcal{C} \times \mathcal{D}) \leq \mathrm{odim}(\mathcal{C}) + \mathrm{odim}(\mathcal{D}),$$

and the analogous inequalities hold for closed embedding dimension.

Proof. To see that $\max\{\mathrm{odim}(\mathcal{C}), \mathrm{odim}(\mathcal{D})\} \leq \mathrm{odim}(\mathcal{C} \times \mathcal{D})$, recall that both $\mathcal{C}$ and $\mathcal{D}$ are minors of $\mathcal{C} \times \mathcal{D}$. It remains to prove that $\mathrm{odim}(\mathcal{C} \times \mathcal{D}) \leq \mathrm{odim}(\mathcal{C}) + \mathrm{odim}(\mathcal{D})$. If either $\mathrm{odim}(\mathcal{C})$ or $\mathrm{odim}(\mathcal{D})$ is infinite, this is trivial. Otherwise, let $d_1 = \mathrm{odim}(\mathcal{C})$ and $d_2 = \mathrm{odim}(\mathcal{D})$. Then let $\mathcal{U} = \{U_i \mid i \in \sigma\}$ be an open realization of $\mathcal{C}$ in $\mathbb{R}^{d_1}$, and let $\mathcal{V} = \{V_j \mid j \in \tau\}$ be an open realization of $\mathcal{D}$ in $\mathbb{R}^{d_2}$. For $i \in \sigma$ define $W_i = U_i \times \mathbb{R}^{d_2}$, and for $j \in \tau$ define $W_j = \mathbb{R}^{d_1} \times V_j$ (here we are identifying $\mathbb{R}^{d_1}$ with the first d_1 coordinates in $\mathbb{R}^{d_1+d_2}$, and $\mathbb{R}^{d_2}$ with the last d_2 coordinates). We claim that $\mathcal{W} = \{W_k \mid k \in \sigma \cup \tau\}$ is an open realization of $\mathcal{C} \times \mathcal{D}$.

To see this, let p be any point in $\mathbb{R}^{d_1+d_2}$, let $p_1 \in \mathbb{R}^{d_1}$ be the first d_1 coordinates of p, and let $p_2 \in \mathbb{R}^{d_2}$ be the remaining coordinates. For $i \in \sigma$ observe that $p \in W_i$ if and only if $p_1 \in U_i$. Symmetrically, we see for $j \in \tau$ that $p \in W_j$ if and only if $p_2 \in V_j$. We conclude that codewords in $\mathrm{code}(\mathcal{W})$ are exactly those of the form $c \cup d$ where c is a codeword in $\mathcal{C}$ and d is a codeword in $\mathcal{D}$. Thus $\mathrm{code}(\mathcal{W}) = \mathcal{C} \times \mathcal{D}$. This proves that $\mathrm{odim}(\mathcal{C} \times \mathcal{D}) \leq \mathrm{odim}(\mathcal{C}) + \mathrm{odim}(\mathcal{D})$ as desired. □

4.4 Minimally Non-Convex Codes

Minor-closed properties of graphs, such as planarity, can be classified by a finite list of forbidden minors. Likewise, some minor-closed families of matroids—such as regular or graphic matroids—admit a characterization in terms of finitely many forbidden minors. To understand convexity of codes, we wish to investigate the forbidden minors of convexity, or minimally non-convex codes, which are defined below. Unlike the case of graphs, we will see in Theorem 4.4.3 and Corollary 4.4.5 that there are not finitely many forbidden minors for convexity. It is a wide open question whether the set of all minimally non-convex codes can be partitioned into a finitely many well-understood families.

Definition 4.4.1. Let $\mathcal{C}$ be a code. We will say that $\mathcal{C}$ is *minimally non-convex* if $\mathcal{C}$ is not an open convex code, but every proper minor of $\mathcal{C}$ is open convex.

Remark 4.4.2. Note that Definition 4.4.1 refers only to *open* convex codes. One could likewise define forbidden minors for closed convex codes, but our primary direction of investigation will be the open case. Where we do need to mention forbidden minors of closed convexity, we refer to them as *minimally non-closed-convex codes.* In a few cases we also make reference to codes that are *minimally non-d-convex*, i.e. they are not open convex in $\mathbb{R}^d$, but all of their proper minors are.

To begin with, we exhibit a relatively straightforward family of minimally non-convex codes. We will see a further family in Section 5.6. Note that the proof of Theorem 4.4.3 relies on Lemma 4.5.4 which appears in the following section. Since we do not make further use of Theorem 4.4.3 beyond its proof, this does not cause any issues.

Theorem 4.4.3. *Let* $\Delta \subseteq 2^{[n]}$ *be a simplicial complex, and let*

$$\mathcal{C}_\Delta := \{\sigma \cup \{n+1\} \mid \sigma \text{ is a nonempty face of } \Delta\} \cup \{\emptyset\} \subseteq 2^{[n+1]}.$$

If Δ *is not collapsible, then* $\mathcal{C}_\Delta$ *is minimally non-convex.*

Proof. Observe that $\mathcal{C}_\Delta$ has a local obstruction of the second kind at $\sigma = \{n+1\}$ since $n+1$ is not a codeword, and $\text{Lk}_{\Delta(\mathcal{C}_\Delta)}(n+1) = \Delta$. Thus $\mathcal{C}_\Delta$ is not open convex, and it will suffice to show that every proper minor of $\mathcal{C}_\Delta$ is convex. In fact, we will show a stronger result: every proper minor of $\mathcal{C}_\Delta$ is intersection complete. Let $\mathcal{D}_\Delta = \mathcal{C}_\Delta \cup \{\{n+1\}\}$, and observe that $\mathcal{D}_\Delta$ is an intersection complete code.

Now let $\mathcal{D} \subseteq 2^{[m]}$ be a proper minor of $\mathcal{C}_\Delta$, and let $f : \mathcal{C}_\Delta \to \mathcal{D}$ be a surjective non-isomorphism determined by a collection $\{T_1, \ldots, T_m\}$ of proper trunks in $\mathcal{C}$. For each $j \in [m]$, let $\sigma_j \subseteq [m]$ be a nonempty index set such that $T_j = \text{Tk}_{\mathcal{C}}(\sigma_j)$. Finally, let $\overline{f} : \mathcal{D}_\Delta \to 2^{[m]}$ be the morphism determined by the collection of proper trunks $\{\overline{T}_1, \ldots, \overline{T}_m\}$ in $\mathcal{D}_\Delta$, where $\overline{T}_j = \text{Tk}_{\mathcal{D}_\Delta}(\sigma_j)$. We will show that $\mathcal{D} = \overline{f}(\mathcal{D}_\Delta)$. Since $\mathcal{D}_\Delta$ differs from $\mathcal{C}_\Delta$ only by the codeword $\{n+1\}$, it suffices to show that $\overline{f}(\{n+1\}) = f(c)$ for some codeword c in $\mathcal{C}_\Delta$.

By Lemma 4.5.4 there exists a neuron $i \in [n+1]$ such that $\text{Tk}_{\mathcal{C}_\Delta}(i)$ is equal to an intersection of the varios T_j. We consider two cases. If $i = n+1$, then no T_j is equal to

$\mathrm{Tk}_{\mathcal{C}_\Delta}(n+1)$, and so no $\overline{T}_j$ contains $\{n+1\}$. We then have $\overline{f}(\{n+1\}) = \emptyset = f(\emptyset)$. Otherwise $i \in [n]$, and every T_j either properly contains $\mathrm{Tk}_{\mathcal{C}_\Delta}(i)$ (and hence is equal to $\mathrm{Tk}_{\mathcal{C}_\Delta}(n+1)$), or does not contain the codeword $\{i, n+1\}$. From this we see that $f(\{i, n+1\}) = \{j \in [m] \mid \sigma_j = \{n+1\}\} = \overline{f}(\{n+1\})$.

We have shown that $\mathcal{D} = \overline{f}(\mathcal{D}_\Delta)$. Since $\mathcal{D}_\Delta$ is intersection complete, so is $\mathcal{D}$, and we conclude that all proper minors of $\mathcal{C}_\Delta$ are convex. This proves the result. □

Remark 4.4.4. In Chapter 6 we will see that we can replace the phrase "not collapsible" in Theorem 4.4.3 by "not convex union representable," a more general condition.

Corollary 4.4.5. *The set of minimally non-convex codes is infinite. As a consequence,* $\mathbf{P_{Code}}$ *contains infinite antichains.*

Proof. There are infinitely many non-isomorphic codes of the form $\mathcal{C}_\Delta$ since there are infinitely many non-collapsible simplicial complexes with a different number of faces. Isomorphism classes of minimally non-convex codes form an antichain in $\mathbf{P_{Code}}$, proving the result. □

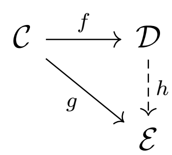
4.5 Covering Relations in $\mathbf{P_{Code}}$

We have just seen in Corollary 4.4.5 that $\mathbf{P_{Code}}$ contains infinite antichains. In this section we further investigate the basic structure of $\mathbf{P_{Code}}$. Our most important result is Theorem 4.5.10, which describes the covering relation in $\mathbf{P_{Code}}$. This is a useful tool for proving that certain codes are minimally non-convex: rather than prove that all proper minors of a non-convex code $\mathcal{C}$ are open convex, one needs only prove this for the (much smaller) set of minors covered by $\mathcal{C}$.

The statement of Theorem 4.5.10 is relatively straightforward, and readers who wish to apply it without delving into its proof are encouraged to skip to Definition 4.5.7. Proving the result will require a significant amount of abstract machinery, and we begin by introducing some terminology and technical lemmas.

Definition 4.5.1. Let $\mathcal{C} \subseteq 2^{[n]}$ be a code, and let $\{T_1, \ldots, T_m\}$ be a collection of proper trunks in $\mathcal{C}$. A trunk T in $\mathcal{C}$ is *generated* by $\{T_1, \ldots, T_m\}$ if there exists nonempty $\tau \subseteq [m]$ such that $T = \bigcap_{j\in\tau} T_j$.

Lemma 4.5.2. *Let $\mathcal{C} \subseteq 2^{[n]}$, $\mathcal{D} \subseteq 2^{[m]}$, and $\mathcal{E} \subseteq 2^{[l]}$ be codes. Let $f : \mathcal{C} \to \mathcal{D}$ be a surjective morphism and let $g : \mathcal{C} \to \mathcal{E}$ be a morphism. Let $\{T_1, \ldots, T_m\}$ and $\{S_1, \ldots, S_l\}$ be the collections of proper trunks in $\mathcal{C}$ that determine f and g respectively. Then there exists a morphism $h : \mathcal{D} \to \mathcal{E}$ such that $g = h \circ f$ if and only if every for every $k \in [l]$ the trunk S_k is either empty, or generated by $\{T_1, \ldots, T_m\}$. In other words, there exists a morphism h making the following diagram commute if and only if the nonempty trunks determining g are generated by the trunks determining f:*



Proof. We prove the forward direction first. Let $k \in [l]$, and let $\{R_1, \ldots, R_l\}$ be the proper trunks in $\mathcal{D}$ that determine the morphism $h : \mathcal{D} \to \mathcal{E}$. Recall that $S_k = g^{-1}(\mathrm{Tk}_{\mathcal{E}}(k))$. Since $g = h \circ f$, we may compute

$$S_k = f^{-1}\left(h^{-1}(\mathrm{Tk}_{\mathcal{E}}(k))\right) = f^{-1}(R_k).$$

If S_k is nonempty, then so is R_k, so there exists nonempty $\tau \subseteq [m]$ with $R_k = \mathrm{Tk}_{\mathcal{D}}(\tau)$. We continue our computation:

$$f^{-1}(R_k) = f^{-1}(\mathrm{Tk}_{\mathcal{D}}(\tau)) = f^{-1}\left(\bigcap_{j\in\tau} \mathrm{Tk}_{\mathcal{D}}(j)\right) = \bigcap_{j\in\tau} f^{-1}(\mathrm{Tk}_{\mathcal{D}}(j)) = \bigcap_{j\in\tau} T_j.$$

Thus S_k is generated by $\{T_1, \ldots, T_m\}$ as desired.

We now prove the converse. Suppose that for every nonempty S_k there exists nonempty $\tau_k \subseteq [m]$ with $S_k = \bigcap_{j\in\tau_k} T_j$. Consider the morphism $h : \mathcal{D} \to 2^{[l]}$ determined by the proper trunks $\{R_1, \ldots, R_l\}$ in $\mathcal{D}$ where $R_k = \mathrm{Tk}_{\mathcal{D}}(\tau_k)$. We first claim that $h(\mathcal{D}) \subseteq \mathcal{E}$. To see this, let d be a codeword in $\mathcal{D}$. By surjectivity of f, there exists a codeword c in $\mathcal{C}$ such that

$f(c) = d$. Observe that by construction $f^{-1}(R_k) = S_k$, and so $c \in S_k$ if and only if $d \in R_k$. We now compute:

$$\begin{aligned} h(d) &= h(f(c)) \\ &= \{k \in [l] \mid f(c) \in R_k\} \\ &= \{k \in [l] \mid c \in f^{-1}(R_k)\} \\ &= \{k \in [l] \mid c \in S_k\} \\ &= g(c). \end{aligned}$$

This computation proves that $h(\mathcal{D}) \subseteq \mathcal{E}$, and moreover that $h \circ f = g$ as desired. □

As a first consequence of this lemma, we provide a corollary that helps us recognize when two minors of a given code are isomorphic.

Corollary 4.5.3. *Let $\mathcal{C} \subseteq 2^{[n]}$, $\mathcal{D} \subseteq 2^{[m]}$, and $\mathcal{E} \subseteq 2^{[l]}$ be codes. Let $f : \mathcal{C} \to \mathcal{D}$ and $g : \mathcal{C} \to \mathcal{E}$ be surjective morphisms. Let $\{T_1, \ldots, T_m\}$ and $\{S_1, \ldots, S_l\}$ be the collections of proper trunks in $\mathcal{C}$ that determine f and g respectively. If these collections generate the same trunks in $\mathcal{C}$, then $\mathcal{D}$ and $\mathcal{E}$ are isomorphic.*

Proof. By Lemma 4.5.2 there exist morphisms $h_1 : \mathcal{D} \to \mathcal{E}$ and $h_2 : \mathcal{E} \to \mathcal{D}$ such that the following diagrams commute:

$$\begin{array}{ccc} \mathcal{C} & \xrightarrow{f} & \mathcal{D} \\ & {\scriptstyle g}\searrow & \downarrow{\scriptstyle h_1} \\ & & \mathcal{E} \end{array} \qquad\qquad \begin{array}{ccc} \mathcal{C} & \xrightarrow{f} & \mathcal{D} \\ & {\scriptstyle g}\searrow & \uparrow{\scriptstyle h_2} \\ & & \mathcal{E} \end{array}$$

One may compute that h_1 and h_2 are mutual inverses, so $\mathcal{D}$ and $\mathcal{E}$ are isomorphic. □

Note that the converse of the above corollary is false: for any code $\mathcal{C} \neq \{\emptyset\}$, there are a variety of morphisms $\mathcal{C} \to \{\emptyset\}$ whose determining trunks are distinct. As a further application of Lemma 4.5.2, we see that any surjective morphism that is determined by a collection of trunks that generates all simple trunks must in fact be an isomorphism. Stated differently, a collection of trunks that determines a surjective non-isomorphism cannot generate all simple trunks. The following lemma captures this observation formally.

Lemma 4.5.4. *Let $\mathcal{C} \subseteq 2^{[n]}$ and $\mathcal{D} \subseteq 2^{[m]}$ be codes. Let $f : \mathcal{C} \to \mathcal{D}$ be a surjective morphism determined by a collection $\{T_1, \ldots, T_m\}$ of proper trunks in $\mathcal{C}$. Then f is not an isomorphism if and only if there exists a neuron $i \in [n]$ such that $\mathrm{Tk}_{\mathcal{C}}(i)$ is not generated by $\{T_1, \ldots, T_m\}$.*

Proof. Let $\mathrm{id} : \mathcal{C} \to \mathcal{C}$ denote the identity function, noting that id is the morphism determined by the collection of simple trunks in $\mathcal{C}$. Consider the following commutative diagram:

$$\begin{array}{ccc} \mathcal{C} & \xrightarrow{f} & \mathcal{D} \\ & {\scriptstyle \mathrm{id}} \searrow & \downarrow {\scriptstyle f^{-1}} \\ & & \mathcal{C} \end{array}$$

By Lemma 4.5.2, the existence of an inverse morphism $f^{-1} : \mathcal{D} \to \mathcal{C}$ is equivalent to the statement that all simple trunks are generated by $\{T_1, \ldots, T_m\}$. This proves the result. □

With these technical tools, we have a finer control over the existence and behavior of morphisms. Our final result will also rely on the following notion of irreducibility for neurons. Note that being redundant and not being irreducible are distinct notions for neurons, although they have similarities. Up to deleting neurons which have equal trunks (i.e. identical behavior), one can view irreducible neurons as those which are necessary to generate (in the sense of intersecting trunks) the activity of all other neurons.

Definition 4.5.5. Let $\mathcal{C} \subseteq 2^{[n]}$ be a code. A neuron $i \in [n]$ is called *irreducible* in $\mathcal{C}$ if the simple trunk $\mathrm{Tk}_{\mathcal{C}}(i)$ is irreducible in the sense of Definition 3.3.5.

Proposition 4.5.6. *Let $\mathcal{C} \subseteq 2^{[n]}$ be a code. The collection of trunks*

$$\{\mathrm{Tk}_{\mathcal{C}}(i) \mid i \in [n] \text{ is irreducible in } \mathcal{C}\}$$

generates all proper nonempty trunks in $\mathcal{C}$.

Proof. Let T be a proper nonempty trunk in $\mathcal{C}$. Then $T = \bigcap_{i \in \sigma} \mathrm{Tk}_{\mathcal{C}}(i)$ for some nonempty $\sigma \subseteq [n]$. If any $i \in \sigma$ is not irreducible, then we may replace $\mathrm{Tk}_{\mathcal{C}}(i)$ by an intersection of two trunks that properly contain it. These trunks will be generated by simple trunks not equal to $\mathrm{Tk}_{\mathcal{C}}(i)$, and by repeating such replacements we may reduce to the case where σ consists of only irreducible neurons, as desired. □

We are now ready to state our main definition. Below, we use neurons decorated by an overline for notational convenience. In particular, i and $\bar{i}$ are distinct neurons.

Definition 4.5.7. Let $\mathcal{C} \subseteq 2^{[n]}$ be a code. Fix a neuron $i \in [n]$, and let $f_i : \mathcal{C} \to 2^{[n]\cup[\overline{n}]}$ be the morphism determined by the collection of proper trunks $\{T_1, \ldots, T_n, T_{\overline{1}}, \ldots, T_{\overline{n}}\}$ where

$$T_j = \begin{cases} \mathrm{Tk}_{\mathcal{C}}(j) & \text{if } \mathrm{Tk}_{\mathcal{C}}(j) \neq \mathrm{Tk}_{\mathcal{C}}(i) \\ \emptyset & \text{otherwise} \end{cases} \quad \text{and} \quad T_{\overline{j}} = \begin{cases} \mathrm{Tk}_{\mathcal{C}}(\{i,j\}) & \text{if } \mathrm{Tk}_{\mathcal{C}}(\{i,j\}) \neq \mathrm{Tk}_{\mathcal{C}}(i) \\ \emptyset & \text{otherwise} \end{cases}$$

for all $j \in [n]$. The *i-th covered code* of $\mathcal{C}$ is the code $\mathcal{C}^{(i)} := f_i(\mathcal{C})$. The morphism $f_i : \mathcal{C} \to \mathcal{C}^{(i)}$ is called the *i-th covering morphism* of $\mathcal{C}$.

Example 4.5.8. Consider the code $\mathcal{C} = \{\mathbf{123}, \mathbf{14}, \mathbf{24}, \mathbf{34}, 1, 2, 3, 4, \emptyset\}$. We will examine the covered codes $\mathcal{C}^{(i)}$ for each neuron i. Since $\mathcal{C}$ is invariant under permutation of the neurons $1, 2$, and 3, we have $\mathcal{C}^{(1)} \cong \mathcal{C}^{(2)} \cong \mathcal{C}^{(3)}$. For $i = 1$, we have $T_1 = T_{\overline{1}} = \emptyset$, and

$$T_2 = \mathrm{Tk}_{\mathcal{C}}(2) = \{123, 24, 2\}, \quad T_3 = \mathrm{Tk}_{\mathcal{C}}(3) = \{123, 34, 3\}, \quad T_4 = \mathrm{Tk}_{\mathcal{C}}(4) = \{14, 24, 34, 4\},$$
$$T_{\overline{2}} = \mathrm{Tk}_{\mathcal{C}}(\{1,2\}) = \{123\}, \quad T_{\overline{3}} = \mathrm{Tk}_{\mathcal{C}}(\{1,3\}) = \{123\}, \quad T_{\overline{4}} = \mathrm{Tk}_{\mathcal{C}}(\{1,4\}) = \{14\}.$$

We can then compute that

$$\mathcal{C}^{(1)} = \{\mathbf{23\overline{2}\overline{3}}, \mathbf{4\overline{4}}, \mathbf{24}, \mathbf{34}, 2, 3, 4, \emptyset\}.$$

For $i = 4$, we have $T_4 = T_{\overline{4}} = \emptyset$, and

$$T_1 = \mathrm{Tk}_{\mathcal{C}}(1) = \{123, 14, 1\}, \quad T_2 = \mathrm{Tk}_{\mathcal{C}}(2) = \{123, 24, 2\}, \quad T_3 = \mathrm{Tk}_{\mathcal{C}}(3) = \{123, 34, 3\},$$
$$T_{\overline{1}} = \mathrm{Tk}_{\mathcal{C}}(\{1,4\}) = \{14\}, \quad T_{\overline{2}} = \mathrm{Tk}_{\mathcal{C}}(\{2,4\}) = \{24\}, \quad T_{\overline{3}} = \mathrm{Tk}_{\mathcal{C}}(\{3,4\}) = \{34\}.$$

This allows us to compute that

$$\mathcal{C}^{(4)} = \{\mathbf{123}, \mathbf{1\overline{1}}, \mathbf{2\overline{2}}, \mathbf{3\overline{3}}, 1, 2, 3, \emptyset\}.$$

The Hasse diagrams of $\mathcal{C}$, $\mathcal{C}^{(1)}$, and $\mathcal{C}^{(4)}$ are shown in Figure 4.3.

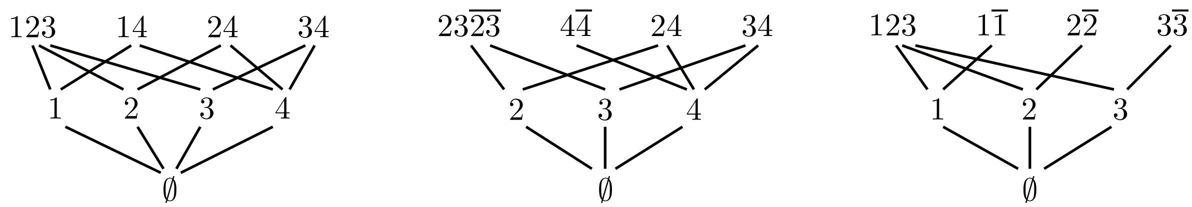


Figure 4.3: The Hasse diagrams of $\mathcal{C}$ and the covered codes $\mathcal{C}^{(1)}$ and $\mathcal{C}^{(4)}$.

As we will see in the following lemma, the collection of trunks that determine the i-th covering morphism is constructed so that when i is irreducible it generates every trunk except for $\text{Tk}_{\mathcal{C}}(i)$. Intuitively, the image of $\mathcal{C}$ under the morphism f_i should thus capture almost (but not all) of the structure of $\mathcal{C}$. In other words, $\mathcal{C}$ should cover $\mathcal{C}^{(i)}$, which we will confirm in Theorem 4.5.10.

Lemma 4.5.9. *Let $\mathcal{C} \subseteq 2^{[n]}$ be a code, let $i \in [n]$ be a neuron, and let $\{T_1, \ldots, T_n, T_{\overline{1}}, \ldots, T_{\overline{n}}\}$ be the collection of proper trunks described in Definition 4.5.7. If i is irreducible, this collection of trunks generates all proper nonempty trunks in $\mathcal{C}$ except for* $\text{Tk}_{\mathcal{C}}(i)$. *If i is not irreducible, then this collection of trunks generates all proper nonempty trunks in $\mathcal{C}$.*

Proof. Let T be a proper nonempty trunk in $\mathcal{C}$ that is not equal to $\text{Tk}_{\mathcal{C}}(i)$. We will show that T is generated by the collection of trunks described in Definition 4.5.7, regardless of whether i is irreducible. We start by writing $T = \bigcap_{j\in\sigma} \text{Tk}_{\mathcal{C}}(j)$ where $\sigma \subseteq [n]$ is nonempty, and $\sigma \neq \{i\}$. We may then consider two cases. If $T \subseteq \text{Tk}_{\mathcal{C}}(i)$ then we may assume that $i \in \sigma$, and that $\text{Tk}_{\mathcal{C}}(j) \neq \text{Tk}_{\mathcal{C}}(i)$ for every $j \in \sigma \setminus \{i\}$. In this case, fix some $j \in \sigma \setminus \{i\}$, define $\tau = \sigma \setminus \{i, j\}$, and observe that $T = T_{\overline{j}} \cap \bigcap_{k\in\tau} T_k$. In the second case that $T \not\subseteq \text{Tk}_{\mathcal{C}}(i)$, we automatically have $\text{Tk}_{\mathcal{C}}(j) \neq \text{Tk}_{\mathcal{C}}(i)$ for all $j \in \sigma$. Then $T = \bigcap_{j\in\sigma} T_j$. In both cases, T is generated by the appropriate set of trunks.

It remains to show that, when $\text{Tk}_{\mathcal{C}}(i)$ is nonempty, the collection of trunks from Definition 4.5.7 generates $\text{Tk}_{\mathcal{C}}(i)$ if and only if i is not irreducible. Since every $T_{\overline{j}}$ is either empty or a proper subset of $\mathrm{Tk}_{\mathcal{C}}(i)$, the collection $\{T_1, \ldots, T_n, T_{\overline{1}}, \ldots, T_{\overline{n}}\}$ generates $\mathrm{Tk}_{\mathcal{C}}(i)$ if and only if the collection $\{T_1, \ldots, T_n\}$ does. But every T_j is either empty or not equal to $\mathrm{Tk}_{\mathcal{C}}(i)$, so this is equivalent to the statement that i is not irreducible. □

We can now prove our main result. This result provides two useful methods for recognizing when a code $\mathcal{C}$ covers a code $\mathcal{D}$. One may show that $\mathcal{D}$ is ismorphic to one of the covered codes $\mathcal{C}^{(i)}$ defined above, or one may exhibit a surjective morphism $\mathcal{C} \to \mathcal{D}$ and simply count the trunks in $\mathcal{C}$ and $\mathcal{D}$. Moreover, this result allows us to directly construct all the codes that a given code covers.

Theorem 4.5.10. *Let $\mathcal{C} \subseteq 2^{[n]}$ be a code and let $\mathcal{D} \subseteq 2^{[m]}$ be a minor of $\mathcal{C}$. Then the following are equivalent:*

(i) $[\mathcal{C}]$ covers $[\mathcal{D}]$ in $\mathbf{P}_{\mathbf{Code}}$,

(ii) $\mathcal{D}$ is isomorphic to $\mathcal{C}^{(i)}$ for some irreducible neuron $i \in [n]$, and

(iii) $\mathcal{C}$ has exactly one more trunk than $\mathcal{D}$.

Proof. We will start by proving that item (i) implies item (ii). If $[\mathcal{C}]$ covers $[\mathcal{D}]$ then there exists a surjective non-isomorphism $g : \mathcal{C} \to \mathcal{D}$, determined by a set of proper trunks $\{S_1, \ldots, S_m\}$ in $\mathcal{C}$. Lemma 4.5.4 implies that there exists some simple trunk $\mathrm{Tk}_{\mathcal{C}}(i)$ that is not generated by $\{S_1, \ldots, S_m\}$, and by Proposition 4.5.6 we may assume that i is irreducible.

Now, let $f_i : \mathcal{C} \to \mathcal{C}^{(i)}$ be i-th covering morphism. The neuron i is irreducible and so by Lemma 4.5.9 the trunks determining f_i generate every nonempty proper trunk in $\mathcal{C}$ except for $\mathrm{Tk}_{\mathcal{C}}(i)$. In particular, Lemma 4.5.4 tells us that f_i is not an isomorphism, while Lemma 4.5.2 tells us that there exists a morphism $h : \mathcal{C}^{(i)} \to \mathcal{D}$ such that the following diagram commutes:

$$\begin{array}{ccc} \mathcal{C} & \xrightarrow{f_i} & \mathcal{C}^{(i)} \\ & \searrow^{g} & \downarrow h \\ & & \mathcal{D} \end{array}$$

Observe that h must be surjective, and so we have $\mathcal{D} \leq \mathcal{C}^{(i)} < \mathcal{C}$. Since $\mathcal{C}$ covers $\mathcal{D}$, we conclude that $\mathcal{D}$ is isomorphic to $\mathcal{C}^{(i)}$ as desired.

Next we prove that item (ii) implies item (iii). Let $i \in [n]$ be an irreducible neuron, and let $f_i : \mathcal{C} \to \mathcal{C}^{(i)}$ be the i-th covering morphism of $\mathcal{C}$. We saw above that when i is irreducible, $f_i : \mathcal{C} \to \mathcal{C}^{(i)}$ is not an isomorphism, and so $\mathcal{C}$ has strictly more trunks than $\mathcal{C}^{(i)}$.

Moreover, Proposition 3.3.8 tells us that the map $T \mapsto f_i^{-1}(T)$ is an injective map from trunks in $\mathcal{C}^{(i)}$ to trunks in $\mathcal{C}$. This map preserves intersections, and has the property that $\mathrm{Tk}_{\mathcal{C}^{(i)}}(j) \mapsto T_j$ and $\mathrm{Tk}_{\mathcal{C}^{(i)}}(\overline{j}) \mapsto T_{\overline{j}}$ for all $j \in [n]$. In particular, the image of this map includes all trunks generated by the collection $\{T_1, \ldots, T_n, T_{\overline{1}}, \ldots, T_{\overline{n}}\}$. Lemma 4.5.9 tells us that this includes all proper nonempty trunks except for $\mathrm{Tk}_{\mathcal{C}}(i)$. The empty trunk and the trunk $\mathcal{C}$ are also clearly in the image of this map. Thus the map $T \mapsto f_i^{-1}(T)$ is an injective map from trunks in $\mathcal{C}^{(i)}$ to all trunks in $\mathcal{C}$ except for $\mathrm{Tk}_{\mathcal{C}}(i)$. We conclude that $\mathcal{C}$ has exactly one more trunk than $\mathcal{C}^{(i)}$.

Finally, we prove that item (iii) implies item (i). Since $\mathcal{D}$ is a minor of $\mathcal{C}$ with strictly fewer trunks, Theorem 3.3.9 implies that $\mathcal{D}$ is not isomorphic to $\mathcal{C}$. Suppose for contradiction that $[\mathcal{C}]$ does not cover $[\mathcal{D}]$, so there exists a code $\mathcal{E}$ with $\mathcal{D} < \mathcal{E} < \mathcal{C}$. Either $\mathcal{E}$ has the same number of trunks as $\mathcal{D}$, or $\mathcal{E}$ has the same number of trunks as $\mathcal{C}$. Theorem 3.3.9 implies that $\mathcal{E}$ is isomorphic to $\mathcal{D}$ in the former case, and that $\mathcal{E}$ is isomorphic to $\mathcal{C}$ in the latter case. In either case, we obtain a contradiction, proving the result. $\square$

We will apply this theorem extensively in Section 5.6. For now, we highlight two immediate consequences.

Corollary 4.5.11. *Let $\mathcal{C} \subseteq 2^{[n]}$ be a code. Then $\mathcal{C}$ covers no more than n codes in $\mathbf{P_{Code}}$.*

Example 4.5.12. Although there are n distinct covered codes for a given code $\mathcal{C} \subseteq 2^{[n]}$, many of these may be isomorphic to one another, particularly when $\mathcal{C}$ has a great deal of symmetry. As an extreme example, let $\mathcal{C} \subseteq 2^{[n]}$ be the compete k-skeleton of the $(n-1)$-simplex. Then $\mathcal{C}$ is invariant under permutation of neurons, and so all $\mathcal{C}^{(i)}$ are isomorphic. Thus $\mathcal{C}$ covers only one code in $\mathbf{P_{Code}}$.

Corollary 4.5.13. $\mathbf{P_{Code}}$ *is a graded poset, with rank function given by the number of nonempty proper trunks in a code.*

Proof. Proposition 3.3.8 ensures that this rank function is compatible with the partial order in $\mathbf{P_{Code}}$ generally, and Theorem 4.5.10 implies that it decreases by exactly one in any covering relation. Furthermore, note that the unique minimal element $\{\emptyset\}$ in $\mathbf{P_{Code}}$ has rank zero. □

Example 4.5.14. Figure 4.4 shows the Hasse diagram of the first five ranks of $\mathbf{P_{Code}}$. A few words are in order about the notation in this figure. The various ranks are highlighted in grey boxes, and codes that are not intersection complete are shown with a dashed border. A valid covering relation in the figure corresponds to a path between two codes that never turns at an acute angle. Equivalently, a covering relation arises from any path that proceeds monotonically downwards.

It is also worth explaining how we computed these codes and the covering relations between them. We began by computing all intersection complete codes with no more than four nonempty proper trunks. Lemma 3.4.1 tells us that nonempty trunks in an intersection complete code are in bijection with codewords, so we needed only compute the set of intersection complete codes with no more than five codewords. Up to isomorphism, intersection complete codes are in bijection with meet semilattices, and so we used an existing list[1] of small semilattices to find all 24 intersection complete codes with no more than five codewords.

To find the remaining non-intersection complete codes, we used a general fact: nonempty trunks in a code $\mathcal{C}$ are always in bijection with elements of $\widehat{\mathcal{C}}$. Thus every code with no more than four nonempty proper trunks can be obtained from an intersection complete code with no more than five codewords by removing some nonempty codewords. Among the relevant intersection complete codes, there are very few codewords that can be deleted

[1]In fact we used the following list of lattices, and the fact that semilattice can be obtained from a lattice by removing the top element: `http://math.chapman.edu/~jipsen/posets/lattices77.html`.

without changing the intersection completion of the resulting code. We computed the six possibilities by hand.

All that remained was to compute the covering relations between all of these codes. We used Theorem 4.5.10 to do so, yielding the poset in Figure 4.4.

This figure highlights several noteworthy structural features of $\mathbf{P_{Code}}$, and raises some questions:

- **$\mathbf{P_{Code}}$ is not a lattice:** Note that the codes $\{\mathbf{12}, 1, \emptyset\}$ and $\{\mathbf{1}, \mathbf{2}, \emptyset\}$ do not have a least upper bound. In particular, both these codes are covered by $\{\mathbf{12}, 1, 3, \emptyset\}$ and $\{\mathbf{12}, \mathbf{13}, 1, \emptyset\}$.

- **Most small codes contain symmetry:** Any code $\mathcal{C} \subseteq 2^{[n]}$ covers up to n codes of the form $\mathcal{C}^{(i)}$. It will cover fewer than n whenever we have $\mathcal{C}^{(i)}$ and $\mathcal{C}^{(j)}$ isomorphic for some $i \neq j$. Among the codes in the figure, this happens for all but four codes: $\{\mathbf{12}, \mathbf{23}, 1, 2, \emptyset\}$, $\{\mathbf{12}, \mathbf{23}, 1, \emptyset\}$, $\{\mathbf{123}, \mathbf{14}, 12, \emptyset\}$, and (trivially) $\{\mathbf{1}, \emptyset\}$. In general, it would be interesting to know the proportion of codes of a given rank for which all $\mathcal{C}^{(i)}$ are distinct. More generally, one could ask the following probabilistic question: given a random code $\mathcal{C} \subseteq 2^{[n]}$, how many codes do we expect it to cover in $\mathbf{P_{Code}}$?

- **Are intervals in $\mathbf{P_{Code}}$ unimodal?** One could examine the intervals of minors in $\mathbf{P_{Code}}$, in particular counting the number of codes at each rank in a given interval. It is natural to wonder whether the resulting list of integers is unimodal. This is true for every interval in the figure, but since these intervals are very small this does not provide much evidence for the general case.

Remark 4.5.15. There is one respect in which Theorem 4.5.10 is lacking: it only characterizes the covering relation in $\mathbf{P_{Code}}$ "from above." That is, given a code $\mathcal{C}$, this theorem does not provide a recipe to nicely enumerate the codes that cover $\mathcal{C}$. We have come close to achieving such a result in unpublished work, though we have not found a way to make the classification elegant beyond the limited case of intersection complete codes. We welcome any readers who are interested in such a result to get in touch (email is best) for discussion and collaboration.

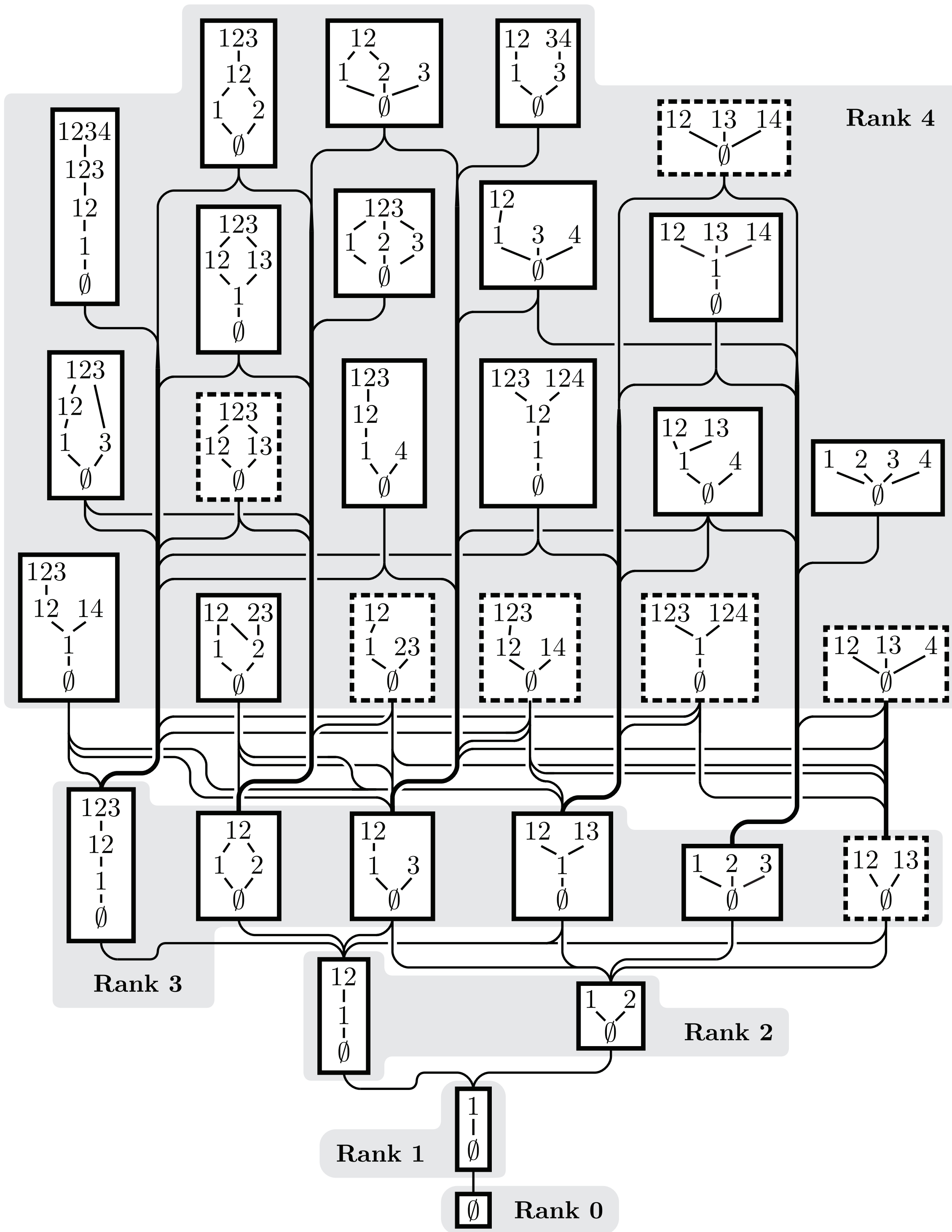


Figure 4.4: The first five ranks of $\mathbf{P}_{\mathbf{Code}}$.

Chapter 5

k-FLEXIBLE SUNFLOWERS OF OPEN CONVEX SETS AND APPLICATIONS

In this chapter we study arrangements of convex sets called “sunflowers.” All of the sets in a sunflower have a nonempty common intersection, but not too many meet outside of this common intersection (see Definition 5.1.1 below). Note that sunflowers of finite sets have been studied combinatorially in the context of the “sunflower conjecture”, see for example [ALWZ20, ER60]. Our work is purely geometric, and only treats sunflowers of convex sets. These arrangements provide a wide range of novel results regarding the discrepancies between open and closed embedding dimensions of codes.

We begin in Section 5.1 by establishing a general discrete geometry theorem (Theorem 5.1.13) that relates the geometric structure of an open convex sunflower to the dimension of the ambient space it sits in. Importantly, this theorem does *not* apply to sunflowers of closed convex sets. In subsequent sections we highlight five distinct applications of this theorem, each consisting of a family of codes whose embedding dimensions are understood via this theorem. In Section 5.7 we present a family of codes whose embedding dimensions can be partially characterized by this theorem. Finally, we generalize some of our families of codes in Section 5.8.

The content of this chapter comes from several papers. Most of the content (Sections 5.1, 5.2, 5.3, 5.7, 5.8) comes from [Jef19a]. Section 5.6 contains the chronologically first general application of sunflowers to the study of convex codes, which appeared in [Jef19b]. The results in Section 5.4 come from joint work with Caitlin Lienkaemper and Nora Youngs in [JLY20], and the results in Section 5.5 come from joint work with Brianna Gambacini, Sam Macdonald, and Anne Shiu in [GJMS20].

5.1 A Geometric Theorem Regarding k-Flexible Sunflowers of Open Convex Sets

We begin by formally defining sunflowers, as well as more general arrangements called k-flexible sunflowers.

Definition 5.1.1. Let $\mathcal{U} = \{U_1, \dots, U_n\}$ be a collection of convex sets. We say that $\mathcal{U}$ is a *sunflower* if $U_i \cap U_j = U_{[n]} \neq \emptyset$ for all $1 \leq i < j \leq n$. Equivalently, $\mathcal{U}$ is a sunflower if $\text{code}(\mathcal{U})$ contains $[n]$ and all other codewords have weight at most one.

Definition 5.1.2. Let $\mathcal{U} = \{U_1, \dots, U_n\}$ be a collection of convex sets. We say that $\mathcal{U}$ is a k*-flexible sunflower* if $U_\sigma = U_{[n]} \neq \emptyset$ for every $\sigma \subseteq [n]$ with $|\sigma| \geq k+1$. Equivalently, $\mathcal{U}$ is a k-flexible sunflower if $\text{code}(\mathcal{U})$ contains $[n]$ and all other codewords have weight at most k.

Definition 5.1.3. If $\mathcal{U} = \{U_1, \dots, U_n\}$ is a k-flexible sunflower, we say that the U_i are *petals*, and $U_{[n]}$ is the *center* of $\mathcal{U}$.

Note that a sunflower is simply a 1-flexible sunflower. As indicated in the definition above, we will only consider sunflowers of convex sets. Moreover, we almost always work with sunflowers of *open* convex sets.

Remark 5.1.4. In terms of neuroscientific motivation, flexible sunflowers are natural to investigate. Allowing codewords beyond singletons, but of a fixed weight, accounts for some tolerance to error in data gathering and captures a wider range of codes. We hope that flexible sunflowers and our subsequent results may aid in the analysis of experimental data.

Example 5.1.5. Figure 5.1 shows four different open k-flexible sunflowers. In part (c), the center of the sunflower is an open cube, and three of the petals are Minkowski sums of the cube with line segments in the three coordinate directions. The fourth petal is the Minkowski sum of the cube with a line segement in the direction $(-1, -1, -1)$. Part (d) of the figure is constructed similarly using an open triangular prism for the center, and letting the five petals be Minkowski sums of the prism with line segments in the facet normal directions.

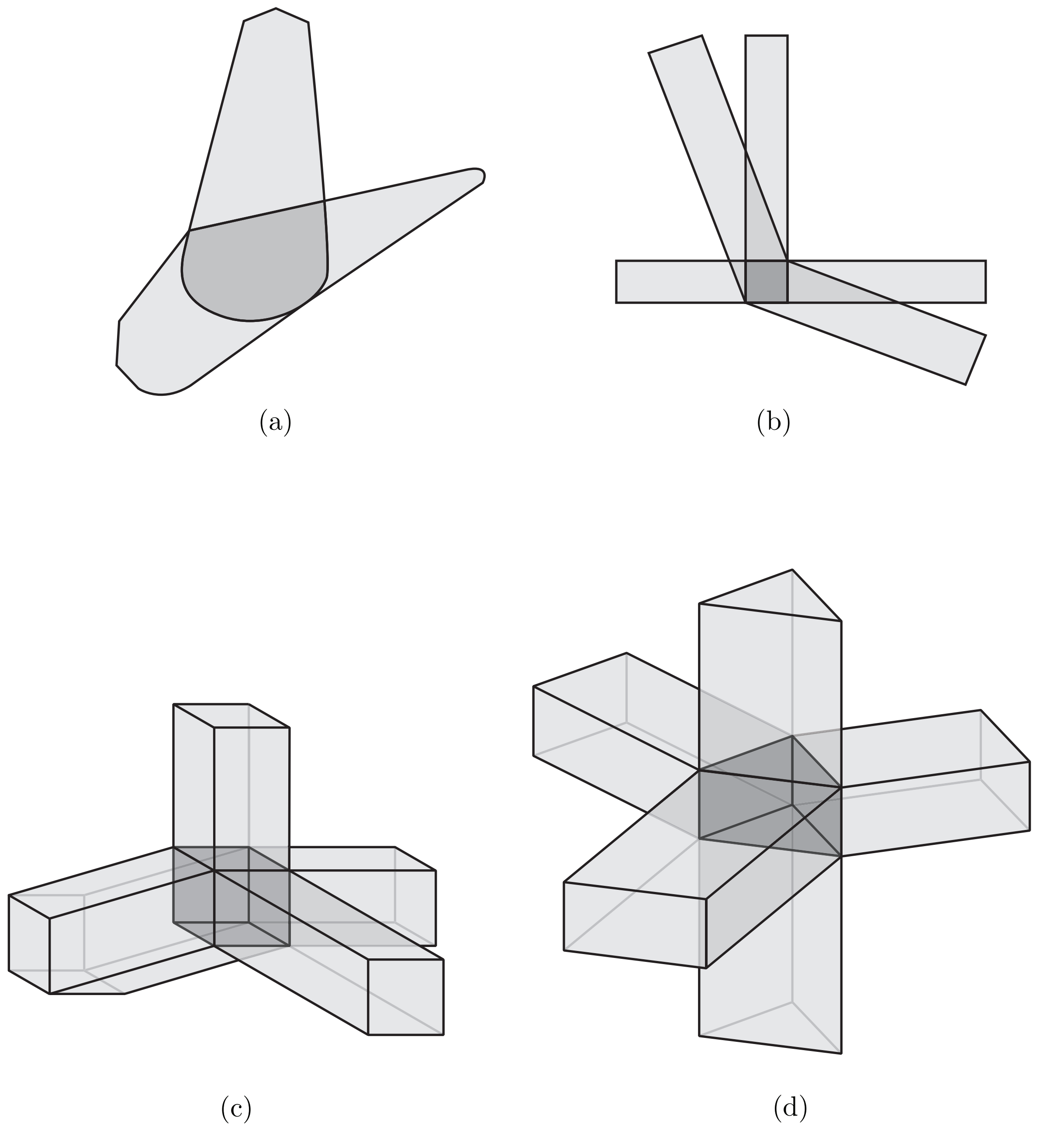


Figure 5.1: Open k-flexible sunflowers with n petals in $\mathbb{R}^d$ for various choices of k, n, and d. (a) $k = 1, n = 3, d = 2$. (b) $k = 2, n = 5, d = 2$. (c) $k = 1, n = 4$, $d = 3$. (d) $k = 1$, $n = 5$, $d = 3$.

To build up to our main theorem, we require one more definition and several technical lemmas.

Definition 5.1.6. Let $\mathcal{U} = \{U_1, \dots, U_n\}$ be an open k-flexible sunflower in $\mathbb{R}^d$ with center U. A point $b \in \partial U$ is called *well supported* if b does not lie in $\mathrm{cl}(\partial U \cap U_i) \setminus U_i$ for any $i \in [n]$.

Remark 5.1.7. In [Jef19a] we used a slightly different (but equivalent) definition of well supported points. We prefer the definition above because it allows us to avoid difficulties that arise from examining boundaries in a subspace topology.

Example 5.1.8. Figure 5.2 shows the open sunflower from Figure 5.1(c), highlighting in black the points in the boundary of the center that are not well-supported.

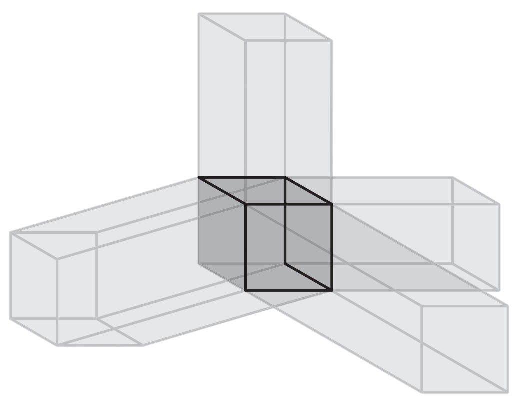


Figure 5.2: Points that are not well-supported in a sunflower in $\mathbb{R}^3$.

Lemma 5.1.9. *Let $\mathcal{U} = \{U_1, \dots, U_n\}$ be an open k-flexible sunflower in $\mathbb{R}^d$ with center U. Then well supported points form a dense subset of ∂U.*

Proof. For each $i \in [n]$, let $X_i = \mathrm{cl}(\partial U \cap U_i) \setminus U_i$. Well supported points are exactly those in ∂U that do not lie in any X_i. Thus it will suffice to show that each X_i has a dense complement in ∂U.

Let p be any point in ∂U. To show that $\partial U \setminus X_i$ is dense in ∂U, we must show that p lies in the closure of $\partial U \setminus X_i$. It suffices to examine the case where $p \in X_i$. In this case, p lies in the closure of $\partial U \cap U_i$, but not in U_i. In particular, p lies arbitrarily close to points in $\partial U \cap U_i$. But $\partial U \cap U_i$ is a subset of $\partial U \setminus X_i$. Thus p lies arbitrarily close to points in $\partial U \setminus X_i$, and hence lies in the closure of this set. This proves the result. □

In the remainder of this section we adopt the convention that supporting hyperplanes for a convex set are always oriented so that the set lies on the positive side of the hyperplane.

Lemma 5.1.10. *Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be an open k-flexible sunflower in $\mathbb{R}^d$ with center U, and let $b \in \partial U$ be well supported. Let H_b be a supporting halfspace for U at b, and let $\sigma = \{i \in [n] \mid b \notin U_i\}$. Then $U_i \subseteq H_b^>$ for all $i \in \sigma$.*

Proof. Suppose not, so that there exists $i \in \sigma$ for which U_i is not contained in $H_b^>$. Since U_i is open, we may assume that there exists a point $p \in U_i$ strictly on the negative side of H_b. Then choose any point $q \in U$, and consider the line segment $\overline{qb}$. All points on this line segment other than b lie in U. For each $r \in \overline{qb}$ with $r \neq b$, note that the line segment $\overline{pr}$ is contained in U_i and intersects ∂U since it begins in the interior of U and ends outside of U.

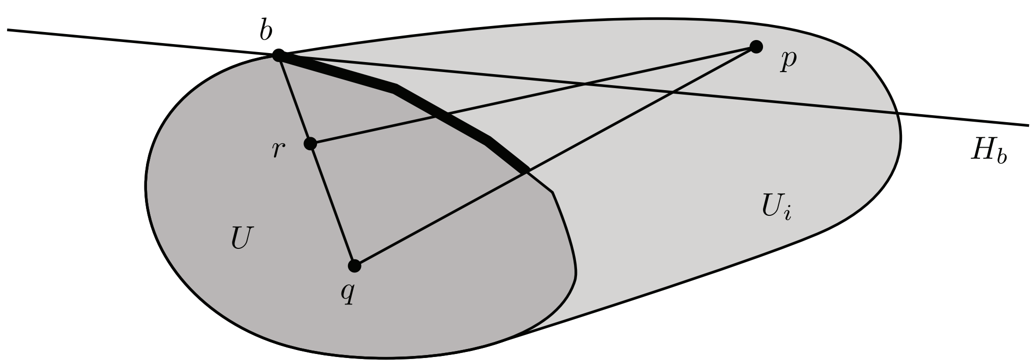


Figure 5.3: An example in $\mathbb{R}^2$ of the objects used to prove Lemma 5.1.10. The bolded segment of ∂U is contained in U_i, and has b as a limit point.

The set of these intersection points is a subset of $U_i \cap \partial U$ whose closure contains b (see Figure 5.3 for an illustration in $\mathbb{R}^2$). In particular, b lies in $\mathrm{cl}(\partial U \cap U_i)$. But since $b \notin U_i$, we have that $b \in \mathrm{cl}(\partial U \cap U_i) \setminus U_i$. This contradicts the fact that b is well supported, proving the result. □

Lemma 5.1.11. *Let $U \subseteq \mathbb{R}^d$ be a convex open set. Let B be a dense subset of the boundary of U, and for each $b \in B$ let H_b be a supporting hyperplane to U at b. Then $\bigcap_{b \in B} H_b^>$ is contained in* $\mathrm{cl}(U)$.

Proof. Consider any point $p \notin \mathrm{cl}(U)$. It will suffice to show that $p \notin \bigcap_{b \in B} H_b^>$. In particular, we will show that there exists $b \in B$ such that $p \notin H_b^>$.

Since p lies a positive distance away from U, the intersection of $\mathrm{int}(\mathrm{conv}(\{p\} \cup U))$ with ∂U is a relatively open subset of ∂U, and thus contains some $b \in B$. Since $\mathrm{int}(\mathrm{conv}(\{p\} \cup U))$ is open, the line segment $\overline{pb}$ can be extended so that it ends at a point $q \in U$, as shown in Figure 5.4.

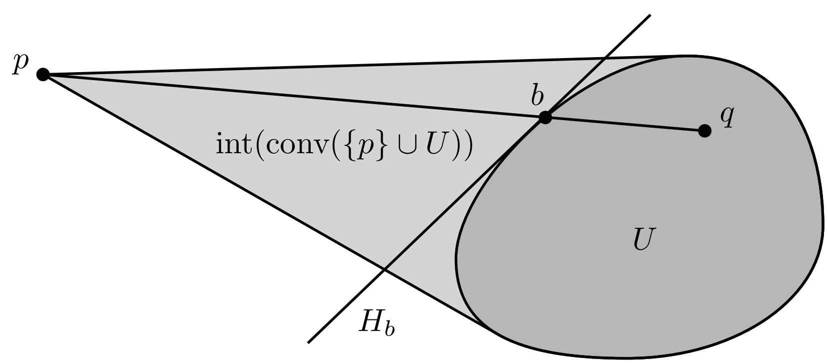


Figure 5.4: An example in $\mathbb{R}^2$ of the objects used to prove Lemma 5.1.11.

Now, consider the supporting hyperplane H_b. We have $U \subseteq H_b^>$. In particular, $H_b^>$ contains q but not b. Since b lies between q and p, we see that $H_b^>$ does not contain p. Thus $p \notin \bigcap_{b \in B} H_b^>$ and the lemma follows. □

Finally, we recall Tverberg's theorem. After stating this theorem, we are ready to prove Theorem 5.1.13, our main result.

Theorem 5.1.12 (Tverberg's theorem)**.** *Let $d \geq 1$, $r \geq 2$, and $n = (d+1)(r-1)+1$. For any set of points $P = \{p_1, \ldots, p_n\}$ in $\mathbb{R}^d$, there is a partition of P into r parts $P_1, \ldots, P_r$ such that $\bigcap_{i=1}^{r} \operatorname{conv} P_i \neq \emptyset$.*

Theorem 5.1.13 (Flexible Sunflower Theorem)**.** *Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be an open k-flexible sunflower in $\mathbb{R}^d$. Suppose that $n \geq dk+1$, and for each $i \in [n]$ let $p_i \in U_i$. Then $\operatorname{conv}\{p_1, \ldots, p_n\}$ contains a point in the center of $\mathcal{U}$.*

Proof. It suffices to prove the first statement for $n = dk+1$. Let U denote the center of $\mathcal{U}$. Suppose for contradiction that the theorem does not hold, so that $\operatorname{conv}\{p_1, \ldots, p_n\}$ does not contain a point in U. Since the U_i are open, we may move each p_i a fixed distance ε away from a chosen point $p \in U$, and choose a separating hyperplane H between $\operatorname{conv}\{p_1, \ldots, p_n\}$ and U such that H does not contain any boundary point of U. Moreover, we can replace each p_i by the intersection of the line segment $\overline{p_i p}$ with H, so that all p_i lie inside H.

Now, H has dimension $d-1$, and $n = ((d-1)+1)k+1$. Thus we may apply Tverberg's theorem with parameter $r = k+1$ to our points $\{p_1, \ldots, p_n\}$. We obtain a partition $P_1, \ldots, P_{k+1}$ such that $\bigcap_{i=1}^{k+1} \operatorname{conv} P_i \neq \emptyset$. Choose any point p' lying in this intersection, and observe that $p' \in H$. Note that the point p' does not lie in U, and so p' lies in at most k petals of $\mathcal{U}$.

Let B be the set of well supported points in ∂U, and choose supporting halfspaces $\{H_b \mid b \in B\}$ as per Lemma 5.1.10. By Lemma 5.1.10, each $H_b^>$ contains all p_j except for at most k. In particular, for every $b \in B$ there must be some P_i such that $H_b^>$ contains all points in P_i, and hence also their convex hull. Thus $p' \in H_b^>$ for all $b \in B$. But by Lemma 5.1.11, this implies that $p' \in \overline{U}$. Since $p' \in H$ and H was constructed not to contain U or any of its boundary points, this is a contradiction. □

Below we state an immediate corollary in the case $k = 1$. This version was proved in

[Jef19b], before the general case. For most of the applications in this chapter, we will only need this $k = 1$ version of the theorem.

Note that when $k = 1$ the application of Tverberg's theorem in the proof above reduces to an application of Radon's theorem. This proof technique was first suggested to us in 2018 by Zvi Rosen for the $k = 1$ case. At the time we instead used a more convoluted proof.

Corollary 5.1.14 (Sunflower Theorem)**.** *Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be an open sunflower in $\mathbb{R}^d$. Suppose that $n \geq d + 1$, and for each $i \in [n]$ let $p_i \in U_i$. Then $\operatorname{conv}\{p_1, \ldots, p_n\}$ contains a point in the center of $\mathcal{U}$.*

Example 5.1.15. Figure 5.5 shows Theorem 5.1.13 in action for the open k-flexible sunflowers illustrated earlier in Figure 5.1. In all cases $n \geq dk + 1$, and so the theorem applies.

Remark 5.1.16. Corollary 5.1.14 and Theorem 5.1.13 have equivalent formulations in terms of "slicing" petals with affine subspaces of certain dimensions. For example, see [Jef19b, Corollary 2.2].

Example 5.1.17. Theorem 5.1.13 fails dramatically for closed k-flexible sunflowers. Indeed, for any n we may construct a closed sunflower $\mathcal{X} = \{X_1, \ldots, X_n\}$ in $\mathbb{R}^2$ consisting of line segments that meet at a common point, and choose points $p_i \in X_i$ for all $i \in [n]$ such that $\operatorname{conv}\{p_1, \ldots, p_n\}$ does not intersect the center of $\mathcal{X}$. This is shown in Figure 5.6.

The inequality $n \geq dk+1$ in Theorem 5.1.13 is tight in the following sense: for every $d \geq 2$ and every $k \geq 1$ there exists an open k-flexible sunflower in $\mathbb{R}^d$ with $n = dk$ petals for which the conclusion of the theorem fails. The following proposition formalizes this observation.

Proposition 5.1.18. *Let $d \geq 2$ and $k \geq 1$. Then there exists an open k-flexible sunflower $\mathcal{U} = \{U_1, \ldots, U_n\}$ in $\mathbb{R}^d$ with $n = dk$, and points $p_1 \in U_1, \ldots, p_n \in U_n$ such that $\operatorname{conv}\{p_1, \ldots, p_n\}$ does not contain a point in the center of $\mathcal{U}$.*

Proof. For $k = 1$, we begin with an open unit hypercube in $\mathbb{R}^d$ centered at the origin, and let U_i be the Minkowski sum of this hypercube with a line segment from the origin to a large

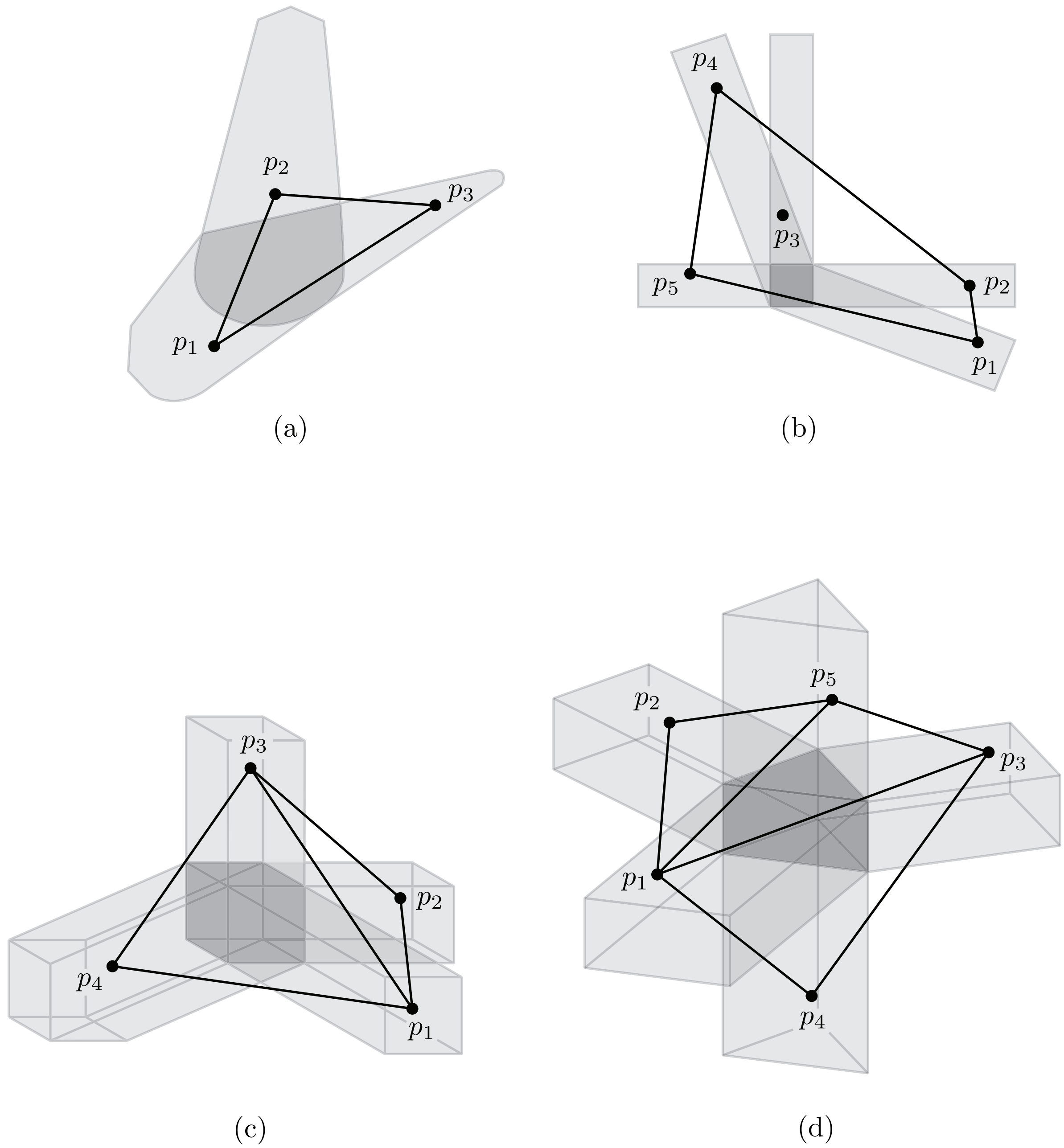


Figure 5.5: Theorem 5.1.13 applied to the open k-flexible sunflowers of Figure 5.1. In each case $\operatorname{conv}\{p_1, \ldots, p_n\}$ is shown in black, and has nonempty intersection with the center of the k-flexible sunflower in question.

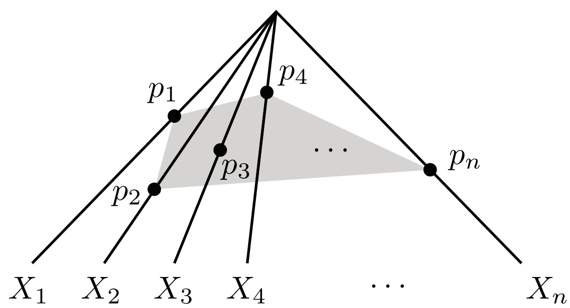


Figure 5.6: A construction showing that Theorem 5.1.13 fails for closed sunflowers.

positive multiple of e_i. We can see that the U_i form an open convex sunflower with d petals, and our desired p_i are just the large multiples of e_i.

For $k \geq 2$, we can take the sunflower described above and duplicate each of the d petals k times. This creates a k-flexible sunflower, and the same choice of p_i (with each duplicated k times) proves the result. □

Remark 5.1.19. One might argue that the construction used to prove Proposition 5.1.18 is unsatisfying, since many of the petals involved are equal. One could address this as follows. Start with the usual coordinate-direction sunflower whose center is a unit hypercube, as described in the proof. If $k = 1$ we are done. Otherwise, choose a cyclic permutation w of $[d]$, for example $i \mapsto i + 1 \mod d$. Then, we can duplicate each petal in our coordinate-direction sunflower k times, but when duplicating the i-th petal we "skew" it slightly in the direction of $-e_{w(i)}$. If each duplicated petal is skewed a different amount, our petals are distinct and diverge from one another. As long as we skew an appropriately small amount, this yields the desired k-flexible sunflower. Figure 5.7 illustrates this modified construction.

We conclude with a corollary which examines the extremal case in which we have an open k-flexible sunflower $\mathcal{U} = \{U_1, \ldots, U_n\}$ with $n = dk$ petals for which Theorem 5.1.13 fails. In this case Theorem 5.1.13 implies code($\mathcal{U}$) must contain at least one codeword of weight k, but we can actually say something slightly stronger:

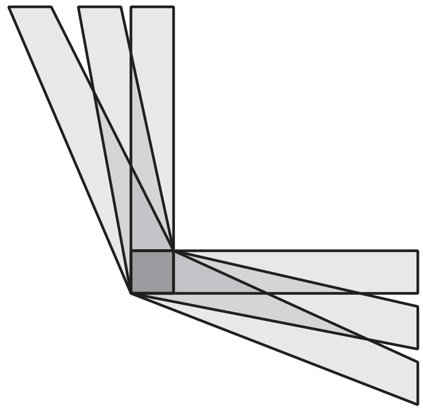


Figure 5.7: A construction proving Proposition 5.1.18 when $d = 2$ and $k = 3$.

Corollary 5.1.20. *Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be an open k-flexible sunflower in $\mathbb{R}^d$. Suppose that $n = dk$, and there exist points $p_1, \ldots, p_n$ such that $p_i \in U_i$ and* $\text{conv}\{p_1, \ldots, p_n\}$ *does not contain a point in the center of $\mathcal{U}$. Then* $\text{code}(\mathcal{U})$ *contains at least d distinct codewords of weight k.*

Proof. We work by induction on k. When $k = 1$ the result is clear since if there are fewer than d codewords of weight k in $\text{code}(\mathcal{U})$ then some U_i is equal to the center of $\mathcal{U}$, and so some p_i lies in the center of $\mathcal{U}$, a contradiction. For $k \geq 2$, suppose for contradiction that $\text{code}(\mathcal{U})$ contains fewer than d codewords of weight k. For each of these codewords c, select some petal U_i with $i \in c$. Deleting these U_i yields a $(k-1)$-flexible sunflower, and since we have deleted fewer than d petals our new $(k-1)$-flexible sunflower has more than $d(k-1)$ petals. But the same choice of p_i yields a collection of points whose convex hull does not contain a point in the center of this $(k-1)$-flexible sunflower, contradicting Theorem 5.1.13. $\square$

5.2 Application 1: Arbitrary Disparity Between Open and Closed Embedding Dimensions

Our first application of the sunflower theorem is to characterize the open and closed embedding dimensions of a family of intersection complete codes. Interestingly, the open and closed embedding dimensions diverge from one another in this family of codes. Previous tools in the literature for bounding open embedding dimension—for example Leray dimension or Helly's theorem [CV16]—are not able to exactly characterize the open embedding dimension of the codes below. This demonstrates that our sunflower theorem has some nontrivial immediate value.

Definition 5.2.1. For $n \geq 1$, define $\mathcal{S}_n \subseteq 2^{[n+1]}$ to be the code consisting of the following codewords: $[n]$, all singleton sets, all pairs $\{i, n+1\}$ for $1 \leq i \leq n$, and the empty set.

Note that $\mathcal{S}_n$ is an intersection complete code. The sunflower theorem can be restated as follows.

Theorem 5.2.2 (Sunflower Theorem, Code Version)**.** *For all* $n \geq 1$*, we have* $\operatorname{odim}(\mathcal{S}_n) = n$*.*

Proof. When $n = 1$, we have $\mathcal{S}_n = \{\mathbf{12}, 1, 2, \emptyset\}$, which can be realized by two overlapping open intervals in $\mathbb{R}^1$. For $n \geq 2$, $\mathcal{S}_n$ has $n+1$ maximal codewords, and so by [CGIK16, Theorem 1.2] $\mathcal{S}_n$ has an open realization in $\mathbb{R}^n$. We will show that this is the smallest dimension possible using open convex sets.

Suppose for contradiction that there exists an open realization $\mathcal{U} = \{U_1, \ldots, U_{n+1}\}$ of $\mathcal{S}_n$ in $\mathbb{R}^{n-1}$. Observe that $\{U_1, \ldots, U_n\}$ forms a sunflower, and U_{n+1} intersects U_i for all $i \in [n]$, but $U_{n+1} \cap U_{[n]} = \emptyset$ since $[n+1]$ is not a codeword in $\mathcal{S}_n$. Thus for each $i \in [n]$ we may choose $p_i \in U_i \cap U_{n+1}$. The convex hull $\operatorname{conv}\{p_1, \ldots, p_n\}$ is contained in U_{n+1} and so does not meet $U_{[n]}$. But Corollary 5.1.14 implies that this convex hull must meet $U_{[n]}$, a contradiction. □

Proposition 5.2.3. *For all* $n \geq 2$ *we have* $\operatorname{cdim}(\mathcal{S}_n) = 2$*. Furthermore,* $\operatorname{cdim}(\mathcal{S}_1) = 1$*.*

Proof. Observe that $\mathcal{S}_1$ can be realized by two overlapping closed line segements in $\mathbb{R}^1$. To see that $\operatorname{cdim}(\mathcal{S}_n) = 2$ for all $n \geq 2$, first observe that the code $\{\mathbf{12}, \mathbf{13}, \mathbf{23}, 1, 2, 3, \emptyset\}$ is a minor of $\mathcal{S}_n$ and has closed embedding dimension equal to two. Thus $\text{cdim}(\mathcal{S}_n) \geq 2$. To prove that $\text{cdim}(\mathcal{S}_n) \leq 2$, we construct a closed realization $\mathcal{X} = \{X_1, \ldots, X_{n+1}\}$ in $\mathbb{R}^2$ as follows. Let $\{X_1, \ldots, X_n\}$ be line segments meeting at a common point as in Figure 5.6. Then let X_{n+1} be a line segment meeting all other X_i that does not pass through their common meeting point. This yields the desired realization, as illustrated in Figure 5.8. □

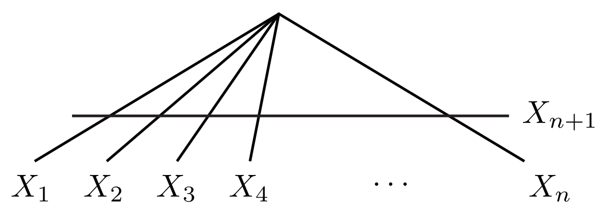


Figure 5.8: A closed realization of $\mathcal{S}_n$ in $\mathbb{R}^2$.

Example 5.2.4. Let us examine $\mathcal{S}_n$ for small values of n. We have

$$\mathcal{S}_1 = \{\mathbf{12}, 1, 2, \emptyset\},$$
$$\mathcal{S}_2 = \{\mathbf{12}, \mathbf{13}, \mathbf{23}, 1, 2, 3, \emptyset\},$$
$$\mathcal{S}_3 = \{\mathbf{123}, \mathbf{14}, \mathbf{24}, \mathbf{34}, 1, 2, 3, 4, \emptyset\}.$$

Figure 5.9 illustrates open realizations of these codes in $\mathbb{R}^1$, $\mathbb{R}^2$, and $\mathbb{R}^3$ respectively. Theorem 5.2.2 says that these realizations are minimal in dimension.

Taken together, these results show that the gap between open and closed embedding dimension of a code may be arbitrarily large, even when we are working with intersection complete codes. Note that this contrasts d-representability of simplicial complexes, a framework in which closed and open sets are interchangeable.

5.3 Application 2: Exponential Open Embedding Dimension

In the previous section we saw that open and closed embedding dimensions may differ from one another. In this section we go a step further, and show that the open embedding

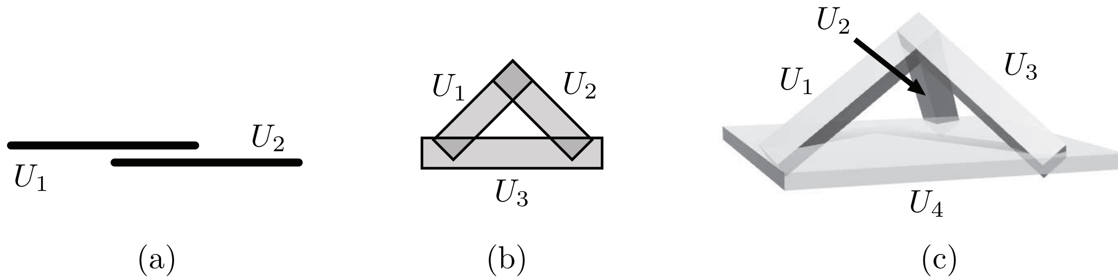


Figure 5.9: Open realizations (a), (b), and (c) of $\mathcal{S}_1$, $\mathcal{S}_2$, and $\mathcal{S}_3$ in dimensions 1, 2, and 3 respectively.

dimension of a code $\mathcal{C} \subseteq 2^{[n]}$ can grow as an exponential function of n, even if closed embedding dimension remains linear in n.

In particular, will associate to every simplicial complex $\Delta \subseteq 2^{[n]}$ an intersection complete code $\mathcal{S}_\Delta \subseteq 2^{[n+1]}$. As long as Δ has at least two facets, the open embedding dimension of $\mathcal{S}_\Delta$ is exactly the number of facets in Δ, which can be large as a function of n.

Definition 5.3.1. Let $\Delta \subseteq 2^{[n]}$ be a simplicial complex. Define $\mathcal{S}_\Delta \subseteq 2^{[n+1]}$ to be the code

$$\mathcal{S}_\Delta := (\Delta * (n+1)) \cup \{[n]\},$$

where $\Delta * (n+1)$ denotes the cone over Δ with apex $n+1$.

Note that the code $\mathcal{S}_n$ of Definition 5.2.1 is equal to $\mathcal{S}_\Delta$ where $\Delta = \{\mathbf{1}, \ldots, \mathbf{n}, \emptyset\}$. We start with some straightforward structural observations about the code $\mathcal{S}_\Delta$.

Proposition 5.3.2. *Let $\Delta \subsetneq 2^{[n]}$ be a simplicial complex with m facets. Then $\mathcal{S}_\Delta$ is intersection complete and has $m+1$ maximal codewords. In particular,* $\operatorname{odim}(\mathcal{S}_\Delta) \leq \max\{2, m\}$.

Proof. First note that $\mathcal{S}_\Delta \setminus \{[n]\}$ is a simplicial complex. Adding a single codeword to a simplicial complex always yields an intersection complete code, so $\mathcal{S}_\Delta$ is intersection complete.

Let $F_1, \ldots, F_m$ be the facets of Δ. Observe that the maximal codewords of $\mathcal{S}_\Delta$ are either facets of $\Delta * (n+1)$, or equal to $[n]$. The facets of $\Delta * (n+1)$ are just $F_i \cup \{n+1\}$ for $i \in [m]$.

Since $\Delta \subsetneq 2^{[n]}$, $[n]$ is also a maximal codeword of $\mathcal{S}_\Delta$, so $\mathcal{S}_\Delta$ has $m+1$ maximal codewords in total. The bound $\operatorname{odim}(\mathcal{S}_\Delta) \le \max\{2, m\}$ then follows immediately from [CGIK16, Theorem 1.2]. □

In fact, the bound $\max\{2, m\}$ in the above proposition is tight, as we will show below. One consequence of this result is that the construction used to prove [CGIK16, Theorem 1.2] is best possible in terms of dimension, at least when we are working with general intersection complete codes.

Proposition 5.3.3. *Let $\Delta \subsetneq 2^{[n]}$ be a simplicial complex with m facets. Then $\mathcal{S}_m$ is a minor of $\mathcal{S}_\Delta$.*

Proof. Label the facets of Δ as $\{\sigma_1, \ldots, \sigma_m\}$, and define $\sigma_{m+1} = \{n+1\}$. For $i \in [m+1]$ define $T_i = \operatorname{Tk}_{\mathcal{S}_\Delta}(\sigma_i)$, and let $f : \mathcal{S}_\Delta \to 2^{[m+1]}$ be the morphism determined by the collection of trunks $\{T_1, \ldots, T_{m+1}\}$. We claim that $f(\mathcal{S}_\Delta) = \mathcal{S}_m$. For any codeword $c \in \mathcal{S}_\Delta$, we can compute that

$$f(c) = \begin{cases} [m] & \text{if } c = [n], \\ \{i, m+1\} & \text{if } c = \sigma_i \cup \{n+1\} \text{ for some } i \in [m], \\ \{i\} & \text{if } c = \sigma_i \text{ for some } i \in [m], \\ \{m+1\} & \text{if } c = \sigma \cup \{n+1\} \text{ where } \sigma \text{ is a proper face in } \Delta, \\ \emptyset & \text{if } c = \sigma \text{ where } \sigma \text{ is a proper face in } \Delta. \end{cases}$$

These possibilities exactly correspond to the codewords of $\mathcal{S}_m$. Thus $f(\mathcal{S}_\Delta) = \mathcal{S}_m$ and the result follows. □

Theorem 5.3.4. *Let $\Delta \subsetneq 2^{[n]}$ be a simplicial complex with $m \ge 2$ facets. Then $\operatorname{odim}(\mathcal{S}_\Delta) = m$.*

Proof. By Proposition 5.3.2 we know that $\operatorname{odim}(\mathcal{S}_\Delta) \le m$. Corollary 4.3.5, Proposition 5.3.3, and Theorem 5.2.2 together tell us that $\operatorname{odim}(\mathcal{S}_\Delta) \ge \operatorname{odim}(\mathcal{S}_m) = m$, proving the result. □

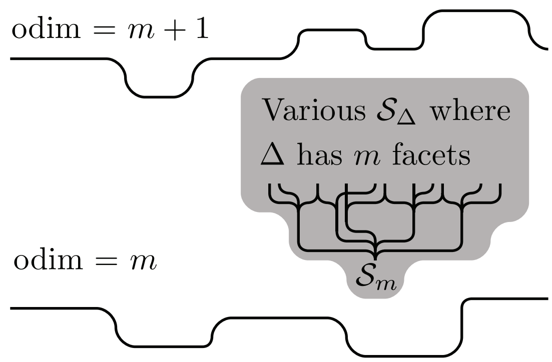


Figure 5.10: A visualization of the codes $\mathcal{S}_\Delta$ in $\mathbf{P}_{\mathbf{Code}}$.

Remark 5.3.5. One way to think of Proposition 5.3.3 is as follows. The set

$$\{\mathcal{S}_\Delta \mid \Delta \text{ is a simplicial complex with } m \text{ facets}\}$$

inherits a partial order from $\mathbf{P}_{\mathbf{Code}}$, and with this inherited order $\mathcal{S}_m$ is the unique minimal element of the set. Theorem 5.3.4 says that for $m \geq 2$ all of these live in the "layer" of codes with open embedding dimension m. An informal visualization of this situation is shown in Figure 5.10.

We highlight two immediate and striking consequences of Theorem 5.3.4 below.

Corollary 5.3.6. *For any $n \geq 2$ and $1 \leq m \leq \binom{n-1}{\lfloor (n-1)/2 \rfloor}$, there exists an intersection complete code on n neurons with $m+1$ maximal codewords, and open embedding dimension equal to m.*

Proof. For $m = 1$, the code $\{\mathbf{1}, \emptyset\}$ suffices. For $m \geq 2$ we apply Theorem 5.3.4. Among all $\binom{n-1}{\lfloor (n-1)/2 \rfloor}$ subsets of $[n-1]$ with size $\lfloor (n-1)/2 \rfloor$, we may select m. Letting Δ be the simplicial complex with these subsets as its facets, we see that $\mathcal{S}_\Delta$ is the desired code. $\square$

Corollary 5.3.7. *For any $n \geq 2$, let $\mathcal{E}_n := \mathcal{S}_\Delta \subseteq 2^{[n]}$ where Δ is the $(\lfloor (n-1)/2 \rfloor - 1)$-skeleton of the simplex $2^{[n-1]}$. Then $\operatorname{odim}(\mathcal{E}_n) = \binom{n-1}{\lfloor (n-1)/2 \rfloor}$, and $\operatorname{cdim}(\mathcal{E}_n) \leq n-1$. In particular, $\operatorname{odim}(\mathcal{E}_n)$ grows exponentially as a function of n, while $\operatorname{cdim}(\mathcal{E}_n)$ is no more than linear.*

Proof. Note that the $(\lfloor (n-1)/2\rfloor - 1)$-skeleton of the simplex $2^{[n-1]}$ has $m = \binom{n-1}{\lfloor (n-1)/2\rfloor}$ facets, which grows exponentially as a function of n. The fact that $\operatorname{cdim}(\mathcal{E}_n) \le n-1$ follows from Theorem 2.3.7 since $\mathcal{E}_n$ is intersection complete. □

Remark 5.3.8. From the perspective of the neuroscience which motivates the study of convex codes, Corollary 5.3.7 has the following interpretation: theoretically, n neurons may "recognize" dimensions that are exponentially large in n. Whether such a phenomenon ever occurs in experimental data could be an interesting avenue of investigation.

These results are qualititatively surprising from a mathematical perspective. The codes $\mathcal{S}_\Delta$ are "almost" simplicial complexes (we have added the single codeword $[n]$ to a simplicial complex), but their open embedding dimensions grow exponentially faster than that of any simplicial complex. These codes also provide the first example of codes whose embedding dimension (open or closed) is larger than $n-1$.

Example 5.3.9. The smallest value of n for which $\binom{n-1}{\lfloor (n-1)/2\rfloor} > n-1$ is $n = 5$. In this case $6 = \binom{n-1}{\lfloor (n-1)/2\rfloor} > n-1 = 4$, and

$$\begin{aligned}\mathcal{E}_5 = \{&\mathbf{1234}, \mathbf{125}, \mathbf{135}, \mathbf{145}, \mathbf{235}, \mathbf{245}, \mathbf{345},\\ &12, 13, 14, 15, 23, 24, 25, 34, 35, 45,\\ &1, 2, 3, 4, 5, \emptyset\}.\end{aligned}$$

Corollary 5.3.7 implies that $\operatorname{odim}(\mathcal{E}_5) = 6$. We do not know whether it is possible for embedding dimension to exceed one less than the number of neurons with fewer than six neurons, i.e. whether there exists a code $\mathcal{C} \subseteq 2^{[5]}$ such that $4 < \operatorname{odim}(\mathcal{C}) < \infty$. Recent work in [GP20] characterizes which codes on five neurons are open convex, but not their exact embedding dimensions.

5.4 Application 3: Monotonicity of Convexity is Strict in Every Dimension

In Section 1.4 we recalled the "monotonicity of open convexity" result from [CGIK16, Theorem 1.3]. This result states that if $\mathcal{C} \subseteq \mathcal{D}$ are codes with the same maximal codewords, then $\operatorname{odim}(\mathcal{D}) \leq \operatorname{odim}(\mathcal{C}) + 1$. In general, this bound need not be tight. For example, if $\mathcal{C} = \{\mathbf{123}, 12, 2, 3, \emptyset\}$ and $\mathcal{D} = \{\mathbf{123}, 12, 13, 2, 3, \emptyset\}$, we have $\operatorname{odim}(\mathcal{C}) = \operatorname{odim}(\mathcal{D}) = 1$.

It is natural to ask whether we can improve $\operatorname{odim}(\mathcal{C}) + 1$ to a smaller quantity, such as $\operatorname{odim}(\mathcal{C})$. In this section we will see that we cannot improve this bound, no matter what value $\operatorname{odim}(\mathcal{C})$ takes. In other words, no matter what the open embedding dimension of a code is, it is possible that the open embedding dimension will strictly increase when we add a new non-maximal codeword. This result was first obtained in joint work with Caitlin Lienkaemper and Nora Youngs (see [JLY20]) and we duplicate our proof below with slight modification of notation and one additional illustration.

We will prove our result by examining a particular family of codes, defined below. As we did in Section 4.5 (in particular Definition 4.5.7), we make use of neurons decorated by an overline to simplify our notation. When $\sigma \subseteq [n]$, we let $\overline{\sigma} = \{\bar{i} \mid i \in \sigma\}$, and the neurons in $\overline{\sigma}$ are distinct from those in σ.

Definition 5.4.1. Let $n \geq 1$. The n-th *prism code*, denoted $\mathcal{P}_n$, is the code on neurons $\{1, 2, \dots, n+1\} \cup \{\overline{1}, \overline{2}, \dots, \overline{n+2}\}$ which has the following codewords:

(i) All subsets of $([n+1] \setminus \{i\}) \cup \{\bar{i}\}$ for all $i \in [n+1]$, and

(ii) $[\overline{n+2}]$.

Observe that $\mathcal{P}_n$ is intersection complete and has $n+2$ maximal codewords. We start by characterizing the open embedding dimension of $\mathcal{P}_n$.

Proposition 5.4.2. *For all $n \geq 1$, we have* $\operatorname{odim}(\mathcal{P}_n) = n$*.*

Proof. Note that every n-subset of $[n+1]$ appears in some codeword of $\mathcal{P}_n$ but $[n+1]$ does not appear in any codeword. This implies that the receptive fields of the neurons $1, 2, \dots, n+1$ in any realization of $\mathcal{P}_n$ will have a nerve that is the boundary of a n-simplex. This can only occur in dimension n or higher (see [Tan13, Section 1.2] or [CV16] for further details), and so $\operatorname{odim}(\mathcal{P}_n) \geq n$. To prove that $\operatorname{odim}(\mathcal{P}_n) \leq n$, we must exhibit an open realization of $\mathcal{P}_n$ in $\mathbb{R}^n$.

For $n = 1$, we have $\mathcal{P}_1 = \{\mathbf{1\overline{2}}, \mathbf{\overline{1}2}, \mathbf{\overline{123}}, 1, 2, \overline{1}, \overline{2}, \emptyset\}$. In this case an open realization is given by $U_1 = (-3,-1), U_2 = (2,4), U_{\overline{1}} = (0,3), U_{\overline{2}} = (-2,1), U_{\overline{3}} = (0,1)$. From here onwards we will assume $n \geq 2$.

First choose points $p_1, \ldots, p_n$ where $p_i = e_i$ in $\mathbb{R}^n$. Also choose $p_{n+1} = -\mathbf{1}$, the vector whose entries are all -1. For $i \in [n+1]$ define F_i to be the facet $\text{conv}\{p_j \mid j \neq i\}$ of the n-simplex $\text{conv}\{p_1, \ldots, p_{n+1}\}$, and let U_i to be the Minkowski sum of F_i with a small ball of radius ε. Choose a small n-simplex with center of mass at the origin and facet normal vectors equal to the various p_i, and let U denote its interior. For $i \in [n+1]$ define $U_{\overline{i}}$ to be Minkowski sum of U with a ray in the direction of p_i. Lastly, define $U_{\overline{n+2}}$ to be equal to U.

Observe that we may choose U small enough that its closure is contained in the interior of $\text{conv}\{p_1, \ldots, p_{n+1}\}$. We may then choose ε small enough that the various U_i do not intersect U. Let $\mathcal{D} \subseteq 2^{[n+1]\cup\overline{[n+2]}}$ denote the code of the realization $\mathcal{U} = \{U_1, \ldots, U_{n+1}, U_{\overline{1}}, \ldots, U_{\overline{n+2}}\}$. We claim that $\mathcal{D}$ has the same maximal codewords as $\mathcal{P}_n$, and that $\mathcal{D} \subseteq \mathcal{P}_n$.

Let us first determine the maximal codewords that arise in $\mathcal{D}$. One maximal codeword is $[\overline{n+2}]$, which arises only inside U. The codeword$([n+1] \setminus \{i\}) \cup \{\overline{i}\}$ arises in a small neighborhood of the point p_i, and it is maximal since this neighborhood can be separated from U_i and all $U_{\overline{j}}$ with $j \neq i$ by a hyperplane with normal vector equal to p_i. This shows that the maximal codewords of $\mathcal{P}_n$ arise as maximal codewords in $\mathcal{D}$.

We must argue that no other maximal codewords arise. Clearly the only maximal codeword containing $\overline{n+2}$ is $[\overline{n+2}]$ since $U = U_{\overline{n+2}}$ is disjoint from all U_i. The other possibilities are a maximal codeword that contains $[n+1]$ or a maximal codeword that contains $\{i, \overline{i}\}$ for some $i \in [n+1]$. The former is impossible since we have chosen ε small enough that various U_i do not intersect U and thus do not all meet at a single point. The latter is impossible because $U_{\overline{i}}$ and U_i are separated by a hyperplane parallel to F_i. Thus the maximal codewords arising in $\mathcal{D}$ are exactly those in $\mathcal{P}_n$.

We next show that the non-maximal codewords in $\mathcal{D}$ are codewords in $\mathcal{P}_n$. First let us consider the codewords of $\mathcal{D}$ that do not contain any $i \in [n+1]$. By construction the various $U_{\overline{i}}$ only overlap inside U, and so the only codewords of this type in $\mathcal{D}$ are the singleton codewords $\{\bar{i}\}$ for $i \in [n+1]$, which arise near the face of $\mathrm{cl}(U)$ with normal vector p_i (and $U_{\overline{n+2}} = U$, so $\{\overline{n+2}\}$ does not arise as a singleton codeword). Any other codeword of $\mathcal{D}$ contains a neuron $i \in [n+1]$ and is thus contained in some maximal codeword $([n+1] \setminus \{i\}) \cup \{\bar{i}\}$. But $\mathcal{P}_n$ contains all subsets of $([n+1] \setminus \{i\}) \cup \{\bar{i}\}$ and so we conclude that $\mathcal{D} \subseteq \mathcal{P}_n$.

We now modify our realization $\mathcal{U} = \{U_1, \ldots, U_{n+1}, U_{\bar{1}}, \ldots, U_{\overline{n+2}}\}$ to obtain a realization of $\mathcal{P}_n$ by applying techniques of [CGIK16, Section 5.4]. Since we are working in $\mathbb{R}^n$ we may choose an open ball B whose boundary contains every p_i. Let us replace every set in our realization by its intersection with B. The resulting code is still contained in $\mathcal{P}_n$, and since B contains the interior of $\mathrm{conv}\{p_1, \ldots, p_{n+1}\}$ it still has the same maximal codewords. Moreover, the atoms of the maximal codewords $([n+1] \setminus \{i\}) \cup \{\bar{i}\}$ now have closures which intersect the boundary ∂B of B in a relatively open subset. The proof technique of [CGIK16, Lemma 5.7] together with the fact that $n \geq 2$ implies that we may repeatedly "shave off" pieces of the sets in our realization near this region to add the desired non-maximal codewords contained in $\{1, 2, \ldots, n+1, \bar{i}\} \setminus \{i\}$, obtaining a realization of $\mathcal{P}_n$ in $\mathbb{R}^n$. □

Example 5.4.3. Figure 5.11 shows an open realization of

$$\mathcal{P}_2 = \{\mathbf{12\overline{3}}, \mathbf{1\overline{2}3}, \mathbf{\overline{1}23}, \mathbf{\overline{1234}}, 12, 13, 23, 1\overline{2}, 1\overline{3}, 2\overline{1}, 2\overline{3}, 3\overline{1}, 3\overline{2}, 1, 2, 3, \overline{1}, \overline{2}, \overline{3}, \emptyset\}$$

in $\mathbb{R}^2$ as given by the proof of Proposition 5.4.2. The "shaved off" regions arise at the rounded corners, realizing the codewords 12, 13, and 23. Observe that we could not modify this realization to add $\overline{4}$ as a codeword since any extension of $U_{\overline{4}}$ outside the central triangle would overlap $U_{\overline{1}}$, $U_{\overline{2}}$, or $U_{\overline{3}}$. We will formalize this observation in Theorem 5.4.4.

Theorem 5.4.4. *Let* $n \geq 1$. *Then* $\mathrm{odim}(\mathcal{P}_n \cup \{\{\overline{n+2}\}\}) = n+1$. *That is, adding a non-maximal codeword to* $\mathcal{P}_n$ *may increase its open embedding dimension.*

Proof. Let $\mathcal{C} = \mathcal{P}_n \cup \{\{\overline{n+2}\}\}$. By monotonicity of convexity and Proposition 5.4.2 we know that $\mathrm{odim}(\mathcal{C}) \leq n+1$. It remains to show that there is no realization of $\mathcal{C}$ in $\mathbb{R}^n$.

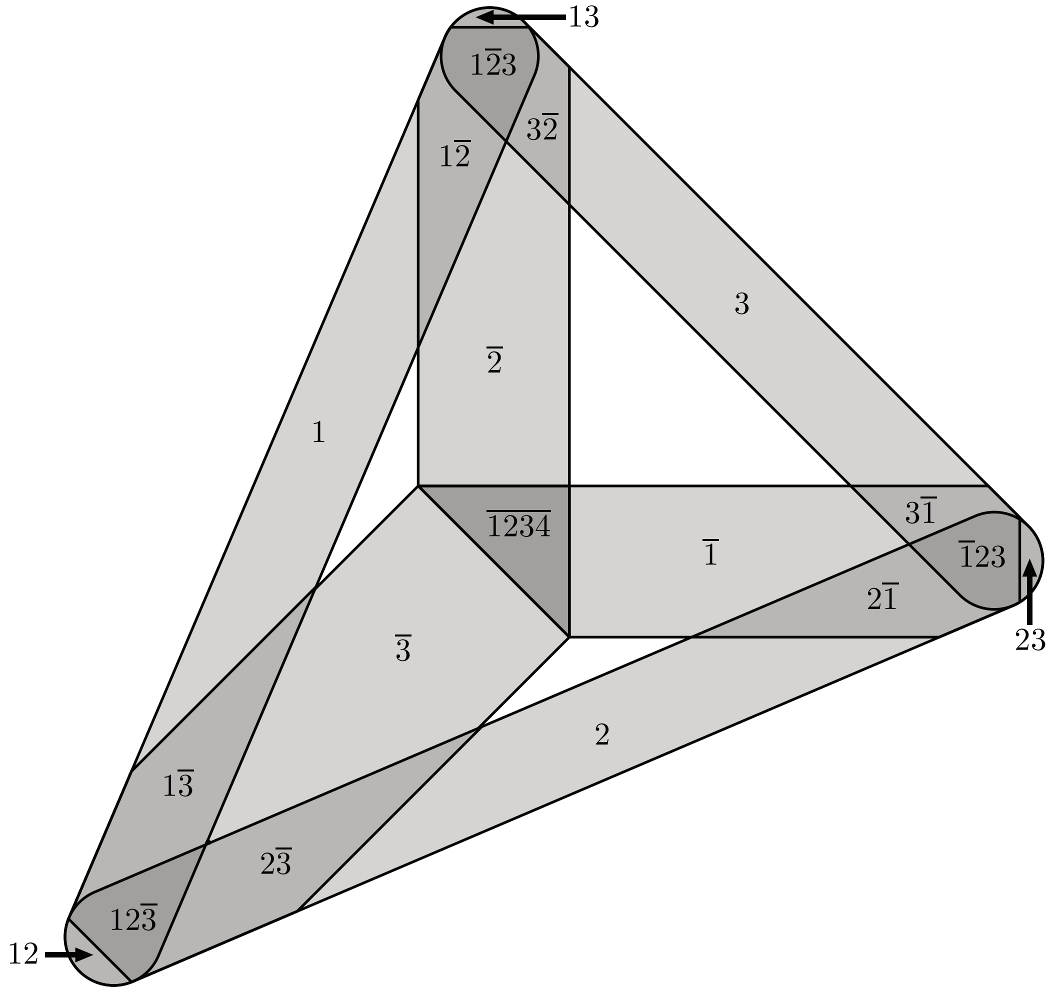


Figure 5.11: An open realization of $\mathcal{P}_2$ in $\mathbb{R}^2$.

Suppose for contradiction that there exists an open realization $\mathcal{U} = \{U_1, \ldots, U_{n+1}, U_{\bar{1}}, \ldots, U_{\overline{n+2}}\}$ of $\mathcal{C}$ in $\mathbb{R}^n$. Since the only codeword in $\mathcal{C}$ containing $\bar{i}$ and $\bar{j}$ for $i \neq j$ is $[\overline{n+2}]$, the sets $U_{\bar{1}}, \ldots, U_{\overline{n+2}}$ form a sunflower in $\mathbb{R}^n$. Call its center U, and for each i between 1 and $n+1$ choose a point p_i in the intersection $U_{\bar{i}} \cap \bigcap_{j \neq i} U_j$. Since the region $U_{\bar{i}} \cap \bigcap_{j \neq i} U_j$ is open, we may assume that the p_i are in general position, so that their convex hull is an n-simplex.

Corollary 5.1.14 tells us that $\operatorname{conv}\{p_1, \ldots, p_{n+1}\}$ contains a point in U. In fact, we claim that this convex hull contains the entirety of U. If not, then U would have to cross one of the facets of the simplex $\text{conv}\{p_1, \ldots, p_{n+1}\}$ (since we are working in $\mathbb{R}^n$). By choice of p_i each of these facets is contained in some U_j. Since U is disjoint from all U_j, it cannot cross any of these facets. Thus U is contained in $\text{conv}\{p_1, \ldots, p_{n+1}\}$.

Since $\{\overline{n+2}\}$ is a codeword in $\mathcal{C}$, we may choose a point $p \in U_{\overline{n+2}} \setminus U$, and examine a generic line segment L from p to a point in U. The line L crosses ∂U at a well-supported point b (recall Definition 5.1.6). Let H_b be a supporting hyperplane for U at b. By Lemma 5.1.10 all $U_{\bar{i}}$ with $i \in [n+1]$ lie on the same side of H_b as U. In particular, H_b separates p from p_i for all $i \in [n+1]$, and so p does not lie in $\text{conv}\{p_1, \ldots, p_{n+1}\}$.

Since U is contained in $\text{conv}\{p_1, \ldots, p_{n+1}\}$, the line L crosses the boundary of this simplex. Since L is contained in $U_{\overline{n+2}}$ this implies that $U_{\overline{n+2}}$ intersects some U_i. But the only codewords of $\mathcal{C}$ containing $\overline{n+2}$ are $\{\overline{n+2}\}$ and $[\overline{n+2}]$, so this is a contradiction. Thus $\mathcal{C}$ is not convex in $\mathbb{R}^n$ and the result follows. □

These results help us better understand monotonicity of open convexity for codes. In the following section, we turn our attention to the question of whether the same results hold for closed embedding dimension. As we will see, the answer to this question is "no" in a variety of ways.

5.5 Application 4: Monotonicity of Convexity Fails With Arbitrarily Large Gap for Closed Convexity

The monotonicity of convexity theorem of [CGIK16] states that if $\mathcal{C} \subseteq \mathcal{D}$ are codes with the same maximal codewords, then $\text{odim}(\mathcal{D}) \leq \text{odim}(\mathcal{C}) + 1$. The authors in [CGIK16] indicated that they did not know whether the same result held for closed embedding dimension. In this section we show that this result is not true for closed embedding dimension. More strongly, we will show that adding a new non-maximal codeword to a code $\mathcal{C}$ can increase its closed embedding dimension by *any* amount (including increasing it to infinity).

These results were first obtained in joint work with Brianna Gambacini, Sam MacDonald, and Anne Shiu [GJMS20]. Here we present these results with slightly abridged proofs, and slightly different notation for consistency with the rest of this work. Our first main result is Theorem 5.5.2, which provides an example of a code whose closed embedding dimension increases from 2 to infinity with the addition of a single non-maximal codeword. After this, we prove Theorem 5.5.6, which shows that closed embedding dimension may increase from 2 to n for any finite choice of $n \geq 3$.

We require a lemma regarding a code $\mathcal{C}_2$, which we will use to prove our first theorem. Our use of the name $\mathcal{C}_2$ for this code will become clear in Section 5.6, where we will define a family of codes $\{\mathcal{C}_n \mid n \geq 2\}$ and investigate their open realizations. For now we require only a supplemental lemma regarding closed realizations of this code. Recall that a convex set $Y \subset \mathbb{R}^d$ is *full-dimensional* if its affine hull is $\mathbb{R}^d$. Note that a convex set is full-dimensional if and only if it has nonempty interior.

Lemma 5.5.1. *The code*

$$\mathcal{C}_2 = \{\mathbf{123\overline{3}}, \mathbf{13\overline{2}}, \mathbf{23\overline{1}}, \mathbf{\overline{123}}, 13, 23, \overline{1}, \overline{2}, \overline{3}, \emptyset\}$$

has closed embedding dimension equal to two, and every closed realization $\mathcal{X} = \{X_1, X_2, X_3, X_{\overline{1}}, X_{\overline{2}}, X_{\overline{3}}\}$ *of* $\mathcal{C}_2$ *in* $\mathbb{R}^2$ *is such that* $X_{\overline{123}}$ *is not full-dimensional.*

Proof. A closed realization of $\mathcal{C}_2$ in $\mathbb{R}^2$ is shown in Figure 5.12. One cannot form a closed realization in $\mathbb{R}^1$ because $\Delta(\mathcal{C}_2)$ has a nontrivial first homology group.

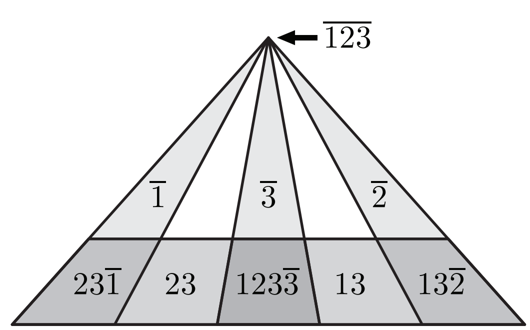


Figure 5.12: A closed realization of $\mathcal{C}_2$ in $\mathbb{R}^2$.

To prove the rest of the lemma, let $\mathcal{X} = \{X_1, X_2, X_3, X_{\overline{1}}, X_{\overline{2}}, X_{\overline{3}}\}$ be a closed realization of $\mathcal{C}_2$ in $\mathbb{R}^2$. We will show that $X_{\overline{123}}$ is not full-dimensional. Below, we let $U_{\overline{i}}$ denote the interior of $X_{\overline{i}}$ for $i \in [3]$.

Suppose for contradiction that $X_{\overline{123}}$ is full-dimensional. Then $X_{\overline{1}}$, $X_{\overline{2}}$, and $X_{\overline{3}}$ are full-dimensional, and we see that $\{U_{\overline{1}}, U_{\overline{2}}, U_{\overline{3}}\}$ is an open sunflower.

Next, X_3 is disjoint from $X_{\overline{123}}$ and so there exists a line L properly separating the two sets. Since each of $\{X_{\overline{1}}, X_{\overline{2}}, X_{\overline{3}}\}$ intersects both X_3 and $X_{\overline{123}}$, the line L passes through $U_{\overline{i}}$ for each $i \in [3]$. But now $\{U_{\overline{1}}, U_{\overline{2}}, U_{\overline{3}}\}$ is an open sunflower in $\mathbb{R}^2$, and the line L passes through all three petals, but not the center of this sunflower. This contradicts Corollary 5.1.14, and so $X_{\overline{123}}$ is not full-dimensional. □

With this lemma, we are ready to prove our first main result.

Theorem 5.5.2 (Closed convexity is non-monotone)**.** *Consider the code*

$$\mathcal{A}_0 = \{\mathbf{123\overline{3}}, \mathbf{13\overline{25}}, \mathbf{23\overline{14}}, \mathbf{\overline{12345}}, 13, 23, \overline{14}, \overline{25}, \overline{3}, \emptyset\}.$$

This code has closed embedding dimension equal to 2, but $\mathcal{A}_0 \cup \{\overline{345}\}$ *has closed embedding dimension equal to* ∞*.*

Proof. Notice that $\mathcal{A}_0$ is isomorphic to the code $\mathcal{C}_2$ from Lemma 5.5.1; we have simply added neurons $\overline{4}$ and $\overline{5}$ which duplicate $\overline{1}$ and $\overline{2}$ respectively. Thus $\operatorname{cdim}(\mathcal{A}_0) = \operatorname{cdim}(\mathcal{C}_2) = 2$.

Now let $\mathcal{A}_0^* = \mathcal{A}_0 \cup \{\overline{345}\}$, and suppose for contradiction that there is a closed realization

$$\mathcal{X} = \{X_1, X_2, X_3, X_{\overline{1}}, X_{\overline{2}}, X_{\overline{3}}, X_{\overline{4}}, X_{\overline{5}}\}$$

of $\mathcal{A}_0^*$ in $\mathbb{R}^d$. We see that $d = 1$ is impossible since $\Delta(\mathcal{A}_0^*)$ has nontrivial first homology group. Thus we may assume that $d \geq 2$.

Let $p_1 \in X_{23\overline{14}}$, $p_2 \in X_{13\overline{25}}$, and $p_3 \in X_{\overline{345}} \setminus X_{\overline{12345}}$ (so, $p_i \in X_{\overline{i}}$ and the three points are distinct). Let A be a 2-dimensional affine subspace of $\mathbb{R}^d$ containing p_1, p_2, and p_3. Define $Y_i = X_i \cap A$ and $Y_{\overline{i}} = X_{\overline{i}} \cap A$ for all i and $\overline{i}$. We claim that $\mathcal{Y} = \{Y_1, Y_2, Y_3, Y_{\overline{1}}, Y_{\overline{2}}, Y_{\overline{3}}, Y_{\overline{4}}, Y_{\overline{5}}\}$

is a realization of $\mathcal{A}_0^*$ in A (i.e. in $\mathbb{R}^2$), and moreover that $Y_{\overline{123}}$ is full-dimensional in this realization.

Clearly the code of $\mathcal{Y}$ is contained in $\mathcal{A}_0^*$ since $\mathcal{X}$ realizes $\mathcal{A}_0^*$. So, we must show that every codeword from $\mathcal{A}_0^*$ arises inside A. By choice of p_1, p_2, and p_3, A automatically contains points that realize the codewords $23\overline{14}$, $13\overline{25}$, and $\overline{345}$.

Consider the line segment L_1 from p_3 to p_2. This line segment is contained entirely in $X_{\overline{5}}$, and so the codewords which appear along it must come from the set $\{\overline{345}, \overline{12345}, \overline{25}, 13\overline{25}\}$. In fact, each of these codewords must appear, and in exactly this order, since the code along the line segment must be a 1-dimensional code (see the arguments in [RZ17]). A symmetric argument shows that the line segment L_2 from p_3 to p_1 has the codewords $\{\overline{345}, \overline{12345}, \overline{14}, 23\overline{14}\}$ along it in that order. Finally, a similar argument shows that the line segment L_3 from p_1 to p_2 has along it the codewords $\{23\overline{14}, 23, 123\overline{3}, 13, 13\overline{25}\}$ in that order. This is shown in Figure 5.13(a).

It remains to show that the codewords $\overline{3}$ and $\emptyset$ arise in A. The codeword $\overline{3}$ can be recovered by examining a line segment from p_3 to a point in $Y_{123\overline{3}}$, and $\emptyset$ can be obtained by assuming that $\mathcal{Y}$ is bounded.

To see that $Y_{\overline{123}}$ is full-dimensional in A, we again consider the line segments L_1, L_2, and L_3. The points p_1, p_2, and p_3 must be in general position: the codeword $23\overline{14}$ that p_1 gives rise to does not appear on the line segment L_1 between p_2 and p_3, the codeword $13\overline{25}$ corresponding to p_2 does not appear on L_2, and the codeword $\overline{345}$ corresponding to p_3 does not appear on L_3. Thus p_1, p_2, p_3 define a triangle in $\mathbb{R}^2$ with edges L_1, L_2, L_3, as shown in Figure 5.13.

Next, L_1 and L_2 both pass through $Y_{\overline{12345}}$ and intersect only at p_3, so we may choose distinct points q_1 and q_2 in $L_1 \cap Y_{\overline{12345}}$ and $L_2 \cap Y_{\overline{12345}}$, respectively. Now consider the triangles T_1 and T_2 with respective vertex sets $\{q_1, q_2, p_1\}$ and $\{q_1, q_2, p_2\}$ (see the figure). The vertices of T_1 are contained in $Y_{\overline{1}}$, so $T_1 \subset Y_{\overline{1}}$. Similarly, $T_2 \subset Y_{\overline{2}}$. Hence, $T_1 \cap T_2 \subset Y_{\overline{1}} \cap Y_{\overline{2}}$. The intersection $T_1 \cap T_2$ is full-dimensional (the doubly shaded region in part (b) of Figure 5.13), and therefore so is $Y_{\overline{1}} \cap Y_{\overline{2}}$.

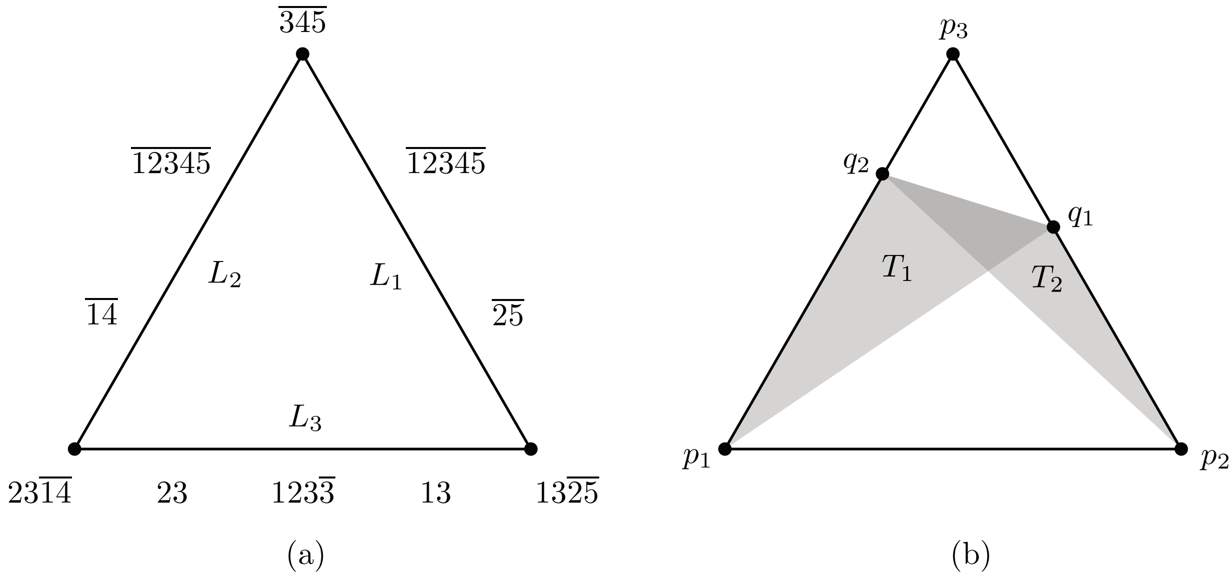


Figure 5.13: (a) The line segments L_1, L_2, and L_3 and the codewords that appear along them. (b) The triangles T_1 and T_2.

However, $Y_{\overline{1}} \cap Y_{\overline{2}} = Y_{\overline{123}}$ (because only the codeword $\overline{12345}$ contains both neurons $\overline{1}$ and $\overline{2}$). By deleting the sets $Y_{\overline{4}}$ and $Y_{\overline{5}}$, we obtain a closed realization $\mathcal{Y}' = \{Y_1, Y_2, Y_3, Y_{\overline{1}}, Y_{\overline{2}}, Y_{\overline{3}}\}$ of the code $\mathcal{C}_2$ in $A \cong \mathbb{R}^2$ with $Y_{\overline{123}}$ full-dimensional. This contradicts Lemma 5.5.1, proving the result. □

Remark 5.5.3. Previous works such as [CGIK16, Lemma 2.9] and [GP20, Theorem 4.1] have used minimum-distance arguments to prove that certain codes are not closed convex. Our proof of Theorem 5.5.2 took a different approach, effectively reducing the argument to the case of open sets. In the future, it would be useful to develop a general set of criteria that preclude closed convexity, and which prove, as special cases, that the code $\mathcal{A}_0 \cup \{\overline{345}\}$ of Theorem 5.5.2 and the relevant codes in [CGIK16, GP20] are not closed convex. Some general criteria regarding closed convexity have been developed recently in [CJL$^+$20], though these do not yet prove that $\mathcal{A}_0 \cup \{\overline{345}\}$ is not closed convex (see [CJL$^+$20, Proposition 5.6]).

Theorem 5.5.2 shows that when we add a non-maximal codeword to a code, its closed embedding dimension may increase to infinity. However, one might hope that if the increase is finite, then it cannot be too large. This is not the case. As mentioned earlier, adding a non-maximal codeword may increase the closed embedding dimension by any finite amount, as we show in our next theorem. To prove this result we first require a lemma similar to Lemma 5.5.1.

Lemma 5.5.4. *Let $n \geq 2$, and let $\mathcal{X} = \{X_1, X_2, \dots, X_{n+1}\}$ be a closed realization of $\mathcal{S}_n$ in $\mathbb{R}^d$. If $d < n$, then the region $X_{[n]}$ is not full-dimensional.*

Proof. For $i \in [n]$ let U_i denote the interior of X_i. If $X_{[n]}$ is full dimensional, then $\mathcal{U} = \{U_1, \dots, U_n\}$ is an open sunflower. But then any hyperplane H separating X_{n+1} from the center of $\mathcal{U}$ has the property that $H \cap U_i$ is nonempty for all $i \in [n]$, and H does not intersect the center of $\mathcal{U}$. When $d < n$ this contradicts Corollary 5.1.14. □

We are now ready to prove our second main result, which deals with the following family of codes.

Definition 5.5.5. For $n \geq 2$, let $\mathcal{A}_n \subseteq 2^{[n+1] \cup [\overline{n}]}$ be the code which consists of the following $2n + 3$ codewords:

(i) The following three codewords: $[n] \cup [\overline{n}]$, $\{n + 1\}$, and the empty set,

(ii) The codeword $\{i, \bar{i}, n + 1\}$ for all $i \in [n]$, and

(iii) The codeword $\{i, \bar{i}\}$ for all $i \in [n]$.

For $i \in [n]$, note that the neurons i and $\bar{i}$ have identical behavior in $\mathcal{A}_n$. Thus $\mathcal{A}_n$ is isomorphic to $\mathcal{A}_n|_{[n+1]}$, which is equal to $\mathcal{S}_n$. The utility of the redundant neurons in $[\overline{n}]$ is illustrated in the following theorem.

Theorem 5.5.6 (Large increase in closed embedding dimension)**.** *For $n \geq 2$, the code $\mathcal{A}_n$ has closed embedding dimension equal to two, and the code $\mathcal{A}_n \cup \{[\overline{n}]\}$ has closed embedding dimension equal to n.*

Proof. The code $\mathcal{A}_n$ is isomorphic to $\mathcal{S}_n$, and we have seen in Proposition 5.2.3 than $\operatorname{cdim}(\mathcal{S}_n) = 2$ for all $n \geq 2$. Two tasks remain: to construct a closed realization of $\mathcal{A}_n \cup \{[\overline{n}]\}$ in $\mathbb{R}^n$, and to prove that we cannot construct a closed realization in any smaller dimension. Below we let $\mathcal{A}_n^* := \mathcal{A}_n \cup \{[\overline{n}]\}$.

We start by constructing a non-degenerate closed realization of $\mathcal{S}_n$ in $\mathbb{R}^n$ as follows. For $i \in [n]$ let X_i be the Minkowski sum of a closed unit cube centered at the origin with a ray in the direction of e_i. Then let X_{n+1} be the thickened hyperplane in which the sum of all coordinates is between $2n$ and $2n+1$. From this realization we may form a closed realization $\mathcal{X} = \{X_1, \ldots, X_{n+1}, X_{\overline{1}}, \ldots, X_{\overline{n}}\}$ of $\mathcal{A}_n$ by taking $X_{\overline{i}} = X_i$. All that remains is to adjust $\mathcal{X}$ so that we obtain the codeword $[\overline{n}]$. This can be accomplished by replacing X_i with $X_i \cap H^{\geq}$ where $H^{\geq}$ is the closed halfspace in which the sum of all coordinates is nonnegative, yielding a closed non-degenerate realization of $\mathcal{A}_n^*$. This construction is shown for $n = 2$ in Figure 5.14.

Now it remains only to show that there is no closed convex realization of $\mathcal{A}_n^*$ in $\mathbb{R}^{n-1}$. Suppose for contradiction that we have such a realization $\mathcal{X} = \{X_1, \ldots, X_{n+1}, X_{\overline{1}}, \ldots, X_{\overline{n}}\}$. Choose a point p^* in the atom of the codeword $[\overline{n}]$, and for $i \in [n]$ choose a point p_i in $X_i \cap X_{n+1}$ (i.e., p_i lies in the atom of the codeword $\{i, \overline{i}, n+1\}$).

For $i \in [n]$, let L_i denote the line segment from p^* to p_i, and observe that $L_i \subset X_{\overline{i}}$. Moreover, the codewords $[\overline{n}]$, $[n] \cup [\overline{n}]$, $\{i, \overline{i}\}$, and $\{i, \overline{i}, n+1\}$ appear along L_i in precisely that order. Also, the codeword $\{n+1\}$ must arise along any line segment between distinct p_i. Thus, all codewords of $\mathcal{A}_n^*$ arise inside the affine span A of $\{p^*, p_1, \ldots, p_n\}$.

It follows that by replacing the sets in $\mathcal{X}$ by their intersections with A, we obtain a closed convex realization of $\mathcal{A}_n^*$ inside $A \cong \mathbb{R}^d$, for some $d \leq n-1$, such that the convex hull of the points $\{p^*,\ p_1, p_2, \ldots, p_n\}$ is full-dimensional (by construction). Observe that $d \geq 2$, as $\mathcal{A}_n^*$ is not convex in $\mathbb{R}^1$.

The code $\mathcal{A}_n^*$ is invariant under permutations of $[n]$ provided we simultaneously permute $[\overline{n}]$ accordingly. Thus we may assume without loss of generality that $\{p^*, p_1, p_2, \ldots, p_d\}$ form the vertices of a d-simplex Δ in A.

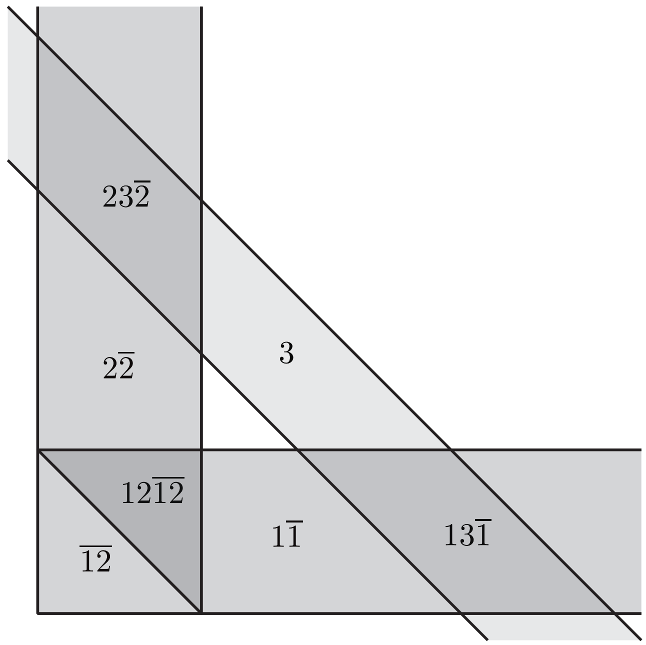


Figure 5.14: A closed realization of $\mathcal{A}_2 \cup \{\overline{12}\} = \{\mathbf{12\overline{12}}, \mathbf{13\overline{1}}, \mathbf{23\overline{2}}, 1\overline{1}, 2\overline{2}, \overline{12}, 3, \emptyset\}$ in $\mathbb{R}^2$. Note that the realization is non-degenerate.

For $i \in [d]$, each L_i is a distinct edge of Δ. Let q_i be a point on L_i in the atom of the codeword $[n] \cup [\overline{n}]$. In particular, $p_i \neq q_i \in X_{[n]}$. Since the q_i lie on distinct edges of Δ, the affine hull H of $\{q_1, q_2, \ldots, q_d\}$ has dimension $d-1$ and so is a hyperplane $H \subset A$. We orient H so that its negative side contains p^* and hence its positive side contains $\{p_1, p_2, \ldots, p_d\}$.

For $i \in [d]$, let Δ_i be the d-simplex with vertices $\{q_1, q_2, \ldots, q_d,\ p_i\}$ and observe that $\Delta_i \subset X_i$. Since all Δ_i lie on the nonnegative side of H and share the common face whose vertices are $\{q_1, q_2, \ldots, q_d\}$, we may choose a point q^* that lies in the interior of all Δ_i, and hence in $X_{[d]}$. Since $d \geq 2$ and $\{X_1, \ldots, X_n\}$ is a sunflower, $X_{[d]} = X_{[n]}$. Thus, q^* lies in $X_{[n]}$. The point q^* lies strictly on the positive side of H, and so the convex hull of $\{q^*,\ q_1, q_2, \ldots, q_d\}$ is a d-simplex contained in $X_{[n]}$. Therefore, $X_{[n]}$ is full-dimensional in A. Since $\{X_1, \ldots, X_{n+1}\}$ is a realization of $\mathcal{S}_n$ and $d < n$, this contradicts Lemma 5.5.4. $\square$

Monotonicity of convexity is a useful tool in the study of open convex codes, and we have just seen that this result fails in a wide variety of ways for closed convex codes. Open convexity is a natural framework from the perspective of neuroscience, as discussed in Chapter 1, and the results of this section show that open convexity can also be more robust than closed convexity from a mathematical point of view.

5.6 Application 5: An Infinite Family of Locally Good Minimally Non-Open-Convex Codes

So far, we have used sunflowers to understand the open embedding dimensions of intersection complete codes, and deepen our understanding of monotonicity of convexity in both the open and closed cases. We now return to the topic of minimally non-convex codes (recall Definition 4.4.1). We saw in Theorem 4.4.3 that there are infinitely many minimally non-convex codes. However, the codes described in Theorem 4.4.3 all have local obstructions. In this section we will use sunflowers to provide an infinite family $\{\mathcal{C}_n \mid n \geq 2\}$ of codes that are both locally good (in fact, locally perfect) and minimally non-convex.

Definition 5.6.1. Let $n \geq 2$. Define $\mathcal{C}_n \subseteq 2^{[n+1] \cup \overline{[n+1]}}$ to be the code that consists of the following codewords:

(i) The empty set,

(ii) $\sigma \cup \{n+1\}$ for every proper nonempty subset σ of $[n]$,

(iii) $\{\bar{i}\}$ for all $i \in [n+1]$,

(iv) $([n+1] \setminus \{i\}) \cup \{\bar{i}\}$ for all $i \in [n]$,

(v) The codeword $[n+1] \cup \{\overline{n+1}\}$, and

(vi) The codeword $\overline{[n+1]}$.

Remark 5.6.2. This family of codes first appeared in [Jef19b] with slightly different notation. Above, we have used overlined vertices to simplify our presentation of these codes.

Remark 5.6.3. In this section we will show that the codes $\{\mathcal{C}_n \mid n \geq 2\}$ are minimally non-convex, and so in particular $\text{odim}(\mathcal{C}_n) = \infty$ for all $n \geq 2$. Recent work in [CJL$^+$20, Section 6] studies the closed embedding dimensions of this family of codes, in particular showing that $\text{cdim}(\mathcal{C}_n) \leq 3$ for all $n \geq 2$.

Note that the maximal codewords in $\mathcal{C}_n$ are those of types (iv), (v), and (vi). Any realization of $\mathcal{C}_n$ essentially comes in two pieces: a sunflower consisting of $n+1$ petals, and a collection of sets whose nerve is the n-simplex. The petals of the sunflower are incident to the unique facet region of the n-simplex, as well as regions corresponding to all codimension-1 faces. Slightly more formally, if $\mathcal{U} = \{U_1, \ldots, U_{n+1}, U_{\overline{1}}, \ldots, U_{\overline{n+1}}\}$ is a realization of $\mathcal{C}_n$ then we have the following:

- The collection $\{U_{\overline{1}}, \ldots, U_{\overline{n+1}}\}$ is a sunflower whose center is disjoint from U_{n+1}. This follows from codewords of types (iii) and (vi).
- U_{n+1} is equal to $\bigcup_{i\in[n]} U_i$. This follows from the fact that $n+1$ is present in all codewords that contain any $i \in [n]$, together with the fact that $\{n+1\}$ is not a codeword in $\mathcal{C}_n$.
- The collection $\{U_1, \ldots, U_n\}$ has nerve equal to $2^{[n]}$. This follows from the codeword of type (v), which has $[n]$ as a subset.
- For all $i \in [n]$ the petal $U_{\overline{i}}$ intersects U_{n+1} inside the region corresponding to the face $[n] \setminus \{i\}$ of $2^{[n]}$. This is described by codewords of type (iv).
- The petal $U_{\overline{n+1}}$ covers the region $U_{[n]}$ in U_{n+1}. This region corresponds to the unique maximal face $[n]$ of $2^{[n]}$. This is a result of the codeword of type (v), together with the fact that this is the only codeword in $\mathcal{C}_n$ containing $[n]$ as a subset.

The code $\mathcal{C}_n$ is not open convex, as we will argue in Theorem 5.6.4 below. However, one may informally visualize a realization of $\mathcal{C}_n$ as pictured in Figure 5.15.

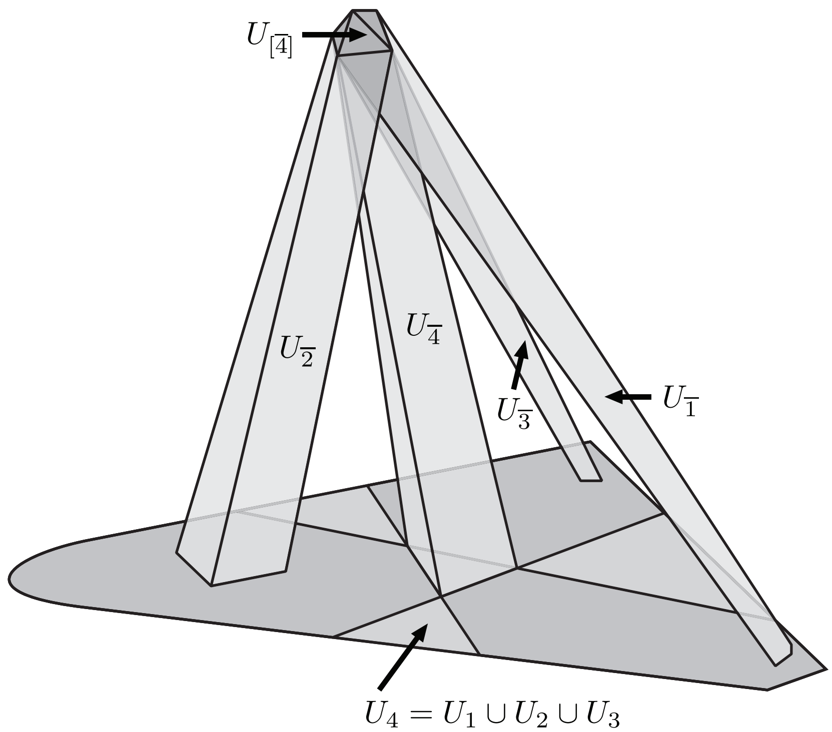


Figure 5.15: An informal illustration of a hypothetical open realization of $\mathcal{C}_3$ in $\mathbb{R}^3$.

Theorem 5.6.4. *The code $\mathcal{C}_n$ is not open convex.*

Proof. Suppose for contradiction that $\mathcal{C}_n$ has an open realization $\mathcal{U} = \{U_1, \ldots, U_{n+1}, U_{\overline{1}}, \ldots, U_{\overline{n+1}}\}$ in $\mathbb{R}^d$. For $i \in [n]$, let p_i be a point in the atom of the codeword $([n+1] \setminus \{i\}) \cup \{\bar{i}\}$. Let $C = \operatorname{conv}\{p_1, \ldots, p_n\}$. We claim that C has nonempty intersection with the atom of the codeword $[n+1] \cup \{\overline{n+1}\}$.

Observe that $C \subseteq U_{n+1}$. Since $\{U_1, \ldots, U_n\}$ covers U_{n+1}, the collection $\{U_1 \cap C, \ldots, U_n \cap C\}$ covers C. Since $p_i \in U_{[n]\setminus\{i\}}$ for all $i \in [n]$, the nerve $\Delta \subseteq 2^{[n]}$ of this cover contains $[n] \setminus \{i\}$ for all $i \in [n]$. Moreover, the nerve lemma (see [Bjö95, Theorem 10.6]) implies that Δ is contractible. The only contractible subcomplex of $2^{[n]}$ that contains $[n] \setminus \{i\}$ for all $i \in [n]$ is $2^{[n]}$ itself. Thus $\Delta = 2^{[n]}$, and there must exist a point $p_{n+1} \in U_{[n]} \cap C$. But the only codeword of $\mathcal{C}_n$ that contains $[n]$ is $[n+1] \cup \{\overline{n+1}\}$, and so p_{n+1} lies in the atom of this codeword. In particular, p_i lies in $U_{\bar{i}}$ for all $i \in [n+1]$.

Choose a point $p \in U_{\overline{[n+1]}}$, and let H be the affine span of $\{p, p_1, \ldots, p_n\}$. Then H is a subspace in $\mathbb{R}^d$ of dimension no more than n, and H additionally contains C and hence p_{n+1}. Note that the collection $\{U_{\bar{1}} \cap H, \ldots, U_{\overline{n+1}} \cap H\}$ is an open sunflower in H, and p_i lies in the $\bar{i}$-th petal of this sunflower. By Corollary 5.1.14, $\operatorname{conv}\{p_1, \ldots, p_{n+1}\}$ contains a point in $U_{\overline{[n+1]}}$. But all p_i lie in U_{n+1}, so this implies that U_{n+1} has nonempty intersection with $U_{\overline{[n+1]}}$. There are no codewords in $\mathcal{C}_n$ that contain $\{n+1\} \cup \overline{[n+1]}$, and so we arrive at a contradiction. Thus $\mathcal{C}_n$ is not open convex. □

Our goal in the remainder of this section is to prove that $\mathcal{C}_n$ is locally good, and that all proper minors of $\mathcal{C}_n$ are open convex. The following lemma will be a useful tool for both of these tasks.

Lemma 5.6.5. *The code $\mathcal{C}_n$ is not max-intersection complete. If we add the codeword $\{n+1\}$ to $\mathcal{C}_n$, then the resulting code is max-intersection complete. In other words, $\widehat{M(\mathcal{C}_n)} \setminus \mathcal{C}_n = \{\{n+1\}\}$.*

Proof. The maximal codewords of $\mathcal{C}_n$ are exactly those of types (iv)-(vi) in Definition 5.6.1, i.e.,

(1) $([n+1] \setminus \{i\}) \cup \{\bar{i}\}$ for all $i \in [n]$,

(2) The codeword $[n+1] \cup \{\overline{n+1}\}$, and

(3) The codeword $\overline{[n+1]}$.

We need only examine intersections of these codewords that consist of more than one term. Any intersection involving codeword (3) is either empty, or equal to $\{\bar{i}\}$ for some $i \in [n+1]$. Definition 5.6.1 states that both possibilities are codewords in $\mathcal{C}_n$. Intersections consisting of codewords of type (1) and (2) are of the form $[n+1] \setminus \sigma$ for some nonempty $\sigma \subseteq [n]$. When $\sigma \neq [n]$ this is a codeword of type (ii) from Definition 5.6.1. When $\sigma = [n]$ the resulting intersection is $\{n+1\}$, which is not a codeword in $\mathcal{C}_n$. Thus $\{n+1\}$ is the only element of $\widehat{M(\mathcal{C}_n)} \setminus \mathcal{C}_n$, as desired. □

Recall that local obstructions in a code can only occur at intersections of maximal codewords (see [CGJ+17, Lemma 1.4]). Lemma 5.6.5 therefore helps us check that $\mathcal{C}_n$ is locally good, which we do below.

Theorem 5.6.6. *The code $\mathcal{C}_n$ is locally good.*

Proof. We must check that $\text{Lk}_{\Delta(\mathcal{C}_n)}(\sigma)$ is contractible for every $\sigma \in \widehat{M(\mathcal{C}_n)} \setminus \mathcal{C}_n$. Lemma 5.6.5 tells us that the only case to check is $\sigma = \{n+1\}$. We may compute that $\text{Lk}_{\Delta(\mathcal{C}_n)}(n+1)$ is the simplex Δ with vertex set $[n] \cup \{\overline{n+1}\}$ together with simplices Δ_i on vertex sets $([n] \setminus \{i\}) \cup \{\bar{i}\}$ glued to it along the faces $[n] \setminus \{i\}$. This complex is contractible since we may simultaneously retract the simplices Δ_i onto Δ, which is then contractible. This link is shown in Figure 5.16 in the case $n = 3$. □

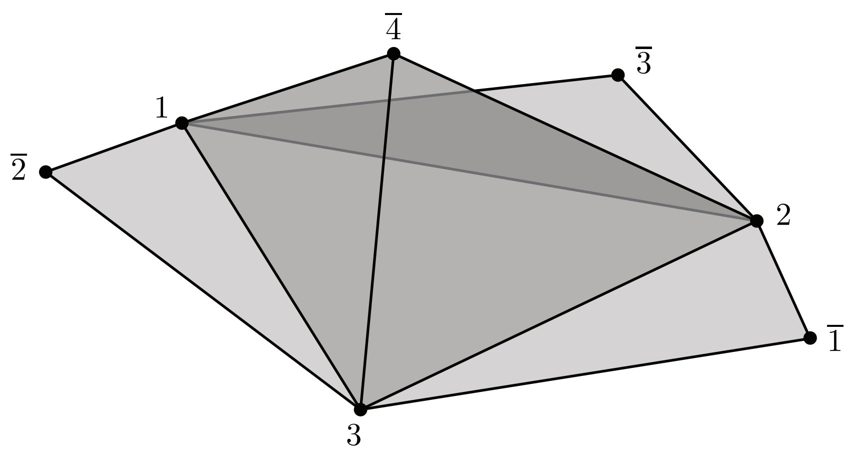


Figure 5.16: The link of $n+1$ in $\Delta(\mathcal{C}_n)$ when $n = 3$.

Remark 5.6.7. Work in [CFS19] generalizes the notion of local obstructions to "local obstructions of the second kind," and correspondingly specializes the class of locally good codes to a smaller class of "locally great" codes. Our joint work with Isabella Novik in [JN19] takes a further step by defining "nerve obstructions" and "locally perfect" codes. This is the subject of Chapter 6. We will see in Theorem 6.8.5 that the codes $\mathcal{C}_n$ are in fact locally perfect.

We now argue that $\mathcal{C}_n$ is minimally non-convex for all n. We have seen that $\mathcal{C}_n$ is not open convex, so it remains to argue that every proper minor of $\mathcal{C}_n$ is open convex. It suffices to show that every covered code of $\mathcal{C}_n$ is open convex. However, we must first address a small conflict of notation: the code $\mathcal{C}_n$ is defined using neurons decorated by overlines, which we also use when defining covered codes in Definition 4.5.7. It turns out that every covered code of $\mathcal{C}_n$ is either max-intersection complete, or can be expressed without introducing any new overlined neurons—to see this, it suffices to argue for all choices of distinct a and b in $[n+1] \cup \overline{[n+1]}$, that if $\operatorname{Tk}_{\mathcal{C}_n}(\{a,b\})$ is nonempty and not equal to $\operatorname{Tk}_{\mathcal{C}_n}(a)$, then it is generated by simple trunks in $\mathcal{C}_n$ that are not equal to $\operatorname{Tk}_{\mathcal{C}_n}(a)$.

We will argue that the covered codes of $\mathcal{C}_n$ are open convex in a series of lemmas below which treat three separate cases, examining the covered codes combinatorially and then explaining why they are open convex. For Lemmas 5.6.9 and 5.6.11, recall Corollary 4.5.3, which states the following: if f is a morphism determined by a collection of trunks $\{T_1, \ldots, T_m\}$, then the image of f is determined (up to isomorphism) by which trunks the collection $\{T_1, \ldots, T_m\}$ generates.

Lemma 5.6.8. *For each $i \in [n+1]$, the i-th covered code of $\mathcal{C}_n$ is max-intersection complete, and hence open convex.*

Proof. Recall from Lemma 5.6.5 that the code $\mathcal{D} = \mathcal{C}_n \cup \{\{n+1\}\}$ is max-intersection complete. Since the image of a max-intersection complete code is max-intersection complete, it will suffice to exhibit a surjective morphism from $\mathcal{D}$ to the i-th covered code $\mathcal{C}_n^{(i)}$. To do this, we extend the i-th covering morphism $f_i : \mathcal{C}_n \to \mathcal{C}_n^{(i)}$ to a function $\overline{f}_i : \mathcal{D} \to \mathcal{C}_n^{(i)}$ by defining $\overline{f}_i(\{n+1\}) = f_i(\{i, n+1\})$ for $i \in [n]$, and defining $\overline{f}_{n+1}(\{n+1\}) = \emptyset$.

Clearly all $\overline{f}_i : \mathcal{D} \to \mathcal{C}_n^{(i)}$ are surjective functions, so it remains to show that they are morphisms. Every trunk T in $\mathcal{C}_n$ is a subset of $\mathcal{D}$, and may be associated to the smallest trunk in $\mathcal{D}$ that contains it. One may check that under this association $\overline{f}_i$ is exactly the morphism determined by the same collection of trunks that determines f_i. □

Lemma 5.6.9. *For each $i \in [n]$, the $\bar{i}$-th covered code of $\mathcal{C}_n$ is isomorphic to the code obtained from $\mathcal{C}_n$ by deleting $\bar{i}$ from all codewords except for $([n+1] \setminus \{i\}) \cup \{\bar{i}\}$.*

Proof. Consider the collection of trunks $\{T_1, \ldots, T_{n+1}, T_{\bar{1}}, \ldots, T_{\overline{n+1}}\}$ where $T_j = \mathrm{Tk}_{\mathcal{C}_n}(j)$ for all $j \in [n+1]$ and

$$T_{\bar{j}} = \begin{cases} \mathrm{Tk}_{\mathcal{C}_n}(\bar{j}) & \text{if } j \neq i, \\ \{([n+1] \setminus \{i\}) \cup \{\bar{i}\}\} & \text{if } j = i. \end{cases}$$

Observe that the image of $\mathcal{C}_n$ under the morphism determined by this collection of trunks is the result of deleting $\bar{i}$ from all codewords except for $([n+1] \setminus \{i\}) \cup \{\bar{i}\}$. To prove that the resulting code is isomorphic to $\mathcal{C}_n^{(\bar{i})}$, it will suffice to show that the above collection of trunks generates every nonempty trunk in $\mathcal{C}_n$ except for $\mathrm{Tk}_{\mathcal{C}_n}(\bar{i})$ (recall from Lemma 4.5.9 that these are exactly the trunks generated by the collection that determines the $\bar{i}$-th covering morphism).

First note that the collection $\{T_1, \ldots, T_{n+1}, T_{\bar{1}}, \ldots, T_{\overline{n+1}}\}$ does not generate $\mathrm{Tk}_{\mathcal{C}_n}(\bar{i})$ since every trunk in the collection either does not contain it, or is properly contained in it. Then let T be a nonempty trunk in $\mathcal{C}_n$ that is not equal to $\mathrm{Tk}_{\mathcal{C}_n}(\bar{i})$. If T is not contained in $\mathrm{Tk}_{\mathcal{C}_n}(\bar{i})$ then $T = \mathrm{Tk}_{\mathcal{C}_n}(\sigma)$ for some $\sigma \subseteq ([n+1] \cup \overline{[n+1]}) \setminus \{\bar{i}\}$, and we have $T = \bigcap_{a\in\sigma} T_a$. Otherwise, T is a proper subset of $\mathrm{Tk}_{\mathcal{C}_n}(\bar{i})$. The only such trunks are the singleton sets $\big\{([n+1] \setminus \{i\}) \cup \{\bar{i}\}\big\}$ and $\big\{\overline{[n+1]}\big\}$. The former is equal to $T_{\bar{i}}$, and the latter is equal to $T_{\bar{j}} \cap T_{\bar{k}}$ for a choice of distinct $j, k \in [n+1] \setminus \{i\}$ (and such a choice is possible since $n \geq 2$). In all cases T is generated by the above collection of trunks, proving the result. □

Lemma 5.6.10. *For each $i \in [n]$, the $\bar{i}$-th covered code of $\mathcal{C}_n$ is open convex.*

Proof. Let c_i be the codeword $([n+1] \setminus \{i\}) \cup \{\bar{i}\}$ in $\mathcal{C}_n$. Lemma 5.6.9 tells us that $\mathcal{C}_n^{(\bar{i})}$ is isomorphic to the code $\mathcal{D}$ obtained from $\mathcal{C}_n$ by deleting $\bar{i}$ from all codewords except for c_i. Observe that if we remove c_i from $\mathcal{D}$, then it is max-intersection complete (indeed, the missing intersection $\{n+1\}$ observed in Lemma 5.6.5 is no longer an intersection of maximal codewords when we remove c_i). Thus we may form a non-degenerate open realization of the code $\mathcal{D}\setminus\{c_i\}$. To form a realization of $\mathcal{D}$ we need only let $U_{\bar{i}}$ be a small ball contained in the interior of the atom of the codeword $[n+1]\setminus\{i\}$, noting that this atom has nonempty interior because we are working with a non-degenerate realization. This proves the result. □

This leaves one final covered code to check, namely $\mathcal{C}_n^{(\overline{n+1})}$. This case is the most complicated, and involves directly constructing a realization using inequalities. We will use monotonicity of convexity to make this task slightly less arduous.

Lemma 5.6.11. *The $(\overline{n+1})$-th covered code of $\mathcal{C}_n$ is isomorphic to the code obtained from $\mathcal{C}_n$ by deleting $\overline{n+1}$.*

Proof. Consider the collection of trunks $\{T_1,\ldots,T_{n+1},T_{\bar{1}},\ldots,T_{\bar{n}}\}$, where $T_i = \mathrm{Tk}_{\mathcal{C}_n}(i)$ and $T_{\bar{i}} = \mathrm{Tk}_{\mathcal{C}_n}(\bar{i})$ for all i. This collection of trunks determines the morphism that deletes the neuron $\overline{n+1}$, and so it will suffice to show that this collection generates all nonempty trunks in $\mathcal{C}_n$ except for $\mathrm{Tk}_{\mathcal{C}_n}(\overline{n+1})$.

First note that this collection does not generate $\mathrm{Tk}_{\mathcal{C}_n}(\overline{n+1})$ since no trunk in the collection contains it. Then let T be a nonempty trunk in $\mathcal{C}_n$ that is not equal to $\mathrm{Tk}_{\mathcal{C}_n}(\overline{n+1})$. If T is not contained in $\mathrm{Tk}_{\mathcal{C}_n}(\overline{n+1})$ then we may write $T=\mathrm{Tk}_{\mathcal{C}_n}(\sigma)$ for some $\sigma\subseteq[n+1]\cup[\overline{n}]$. Then $T=\bigcap_{a\in\sigma}T_a$. Otherwise, T is contained in $\mathrm{Tk}_{\mathcal{C}_n}(\overline{n+1})$, but the only such trunks are the singleton sets $\{[n+1]\cup\{\overline{n+1}\}\}$ and $\{[\overline{n+1}]\}$. The former is the trunk of $[n+1]$ in $\mathcal{C}_n$ (hence equal to $\bigcap_{i\in[n+1]}T_i$) and the latter is equal to $T_{\bar{1}}\cap T_{\bar{2}}$. Thus our collection of trunks generates all trunks except for $\mathrm{Tk}_{\mathcal{C}_n}(\overline{n+1})$, proving the result. □

Lemma 5.6.12. *The $(\overline{n+1})$-th covered code of $\mathcal{C}_n$ is open convex.*

Proof. Lemma 5.6.11 tells us that $\mathcal{C}_n^{(\overline{n+1})}$ is isomorphic to the restriction of $\mathcal{C}_n$ to the set of neurons $[n+1]\cup[\overline{n}]$. Throughout our proof we will identify $\mathcal{C}_n^{(\overline{n+1})}$ with this restriction.

Note that $\mathcal{C}_n^{(\overline{n+1})}$ is *not* max-intersection complete, since $\{n+1\}$ is an intersection of maximal codewords, but is not present in the code. Thus we must argue the open convexity of $\mathcal{C}_n^{(\overline{n+1})}$ more directly. By monotonicity of convexity, it will suffice to construct a realization of a code $\mathcal{D} \subseteq 2^{[n+1]\cup[\overline{n}]}$ with $\mathcal{D} \subseteq \mathcal{C}_n^{(\overline{n+1})} \subseteq \Delta(\mathcal{D})$. We proceed in this manner below.

For $i \in [n]$, define $U_{\bar{i}} \subseteq \mathbb{R}^n$ to be the product of open intervals $(0,1) \times (0,1) \times \cdots \times \mathbb{R} \times \cdots \times (0,1)$ where the factor $\mathbb{R}$ appears in the i-th index. That is, $U_{\bar{i}}$ is the convex open set in $\mathbb{R}^n$ defined by the inequalities $0 < x_j < 1$ for $j \neq i$. Observe that $\{U_{\bar{1}}, \ldots, U_{\bar{n}}\}$ is an open sunflower whose center is an open unit hypercube in the positive orthant with a vertex at the origin.

Next, for any interval $(a,b) \subseteq \mathbb{R}$ define a convex open set

$$H_{(a,b)} = \{(x_1, \ldots, x_n) \mid a < x_1 + x_2 + \cdots + x_n < b\}.$$

Thus, the set $H_{(a,b)}$ is just a union of translates of the hyperplane given by the equation $x_1 + \cdots + x_n = 0$. Now, define U_{n+1} to be the intersection of $H_{(2n-1,2n)}$ with the (open) positive orthant. Observe that U_{n+1} intersects $U_{\bar{i}}$ for all $i \in [n]$ (in particular, this intersection contains the point $(\frac{1}{2}, \frac{1}{2}, \ldots, \frac{3n}{2}, \ldots, \frac{1}{2})$ where the term $\frac{3n}{2}$ appears in the i-th coordinate). However, U_{n+1} does not intersect the center of the sunflower $\{U_{\bar{1}}, \ldots, U_{\bar{n}}\}$ since the sum of coordinates of any point in the center is always strictly less than n.

Finally, for all $i \in [n]$, let H_i^- be the open halfspace defined by $\sum_{j\in[n]\setminus\{i\}} x_j > n-1$, and define $U_i = U_{n+1} \cap H_i^-$. We now have an open realization $\{U_1, \ldots, U_{n+1}, U_{\bar{1}}, \ldots, U_{\bar{n}}\}$ in $\mathbb{R}^n$. Figure 5.17 illustrates this realization in the case $n = 2$.

It remains to show that this realization relates to the code $\mathcal{C}_n^{(\overline{n+1})}$. Let $\mathcal{D} = \text{code}(\mathcal{U}) \subseteq 2^{[n+1]\cup[\overline{n}]}$. We claim that $\mathcal{D} \subseteq \mathcal{C}_n^{(\overline{n+1})} \subseteq \Delta(\mathcal{D})$. First, let us describe explicitly the codewords of $\mathcal{C}_n^{(\overline{n+1})}$. These are:

(i) The empty set,

(ii) $\sigma \cup \{n+1\}$ for every proper nonempty subset σ of $[n]$,

(iii) $\{\bar{i}\}$ for $i \in [n]$,

(iv) $([n+1] \setminus \{i\}) \cup \{\bar{i}\}$ for all $i \in [n]$,

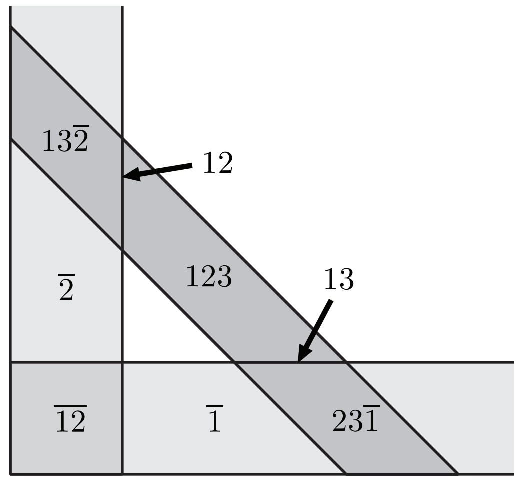


Figure 5.17: The construction used to partially realize the code $\mathcal{C}_2^{(\overline{3})}$ in $\mathbb{R}^2$. Note that this realization is degenerate, as the atoms of 12 and 13 are not full-dimensional.

(v) the codeword $[n+1]$, and

(vi) the codeword $[\overline{n}]$.

Observe that the maximal codewords are those of types (iv), (v), and (vi).

To see that $\mathcal{D} \subseteq \mathcal{C}_n^{(\overline{n+1})} \subseteq \Delta(\mathcal{D})$ we determine the codewords of $\mathcal{D}$ explicitly. We start by examining the codewords that do not contain $n+1$. We claim that such codewords are precisely those of types (i), (iii), and (vi) above: the empty set arises from any point with all coordinates sufficiently large and negative, codewords of type (iii) arise from points in $U_{\overline{i}}$ with large j-th coordinate, and the codeword of type (vi) arises from any point in the center of the sunflower $\{U_{\overline{1}}, \ldots, U_{\overline{n}}\}$. No other codewords that do not contain $n+1$ can arise since $U_i \subseteq U_{n+1}$ for all $i \in [n]$, and the sets $U_{\overline{i}}$ form a sunflower.

Next we turn to the codewords that contain the neuron $n+1$, or equivalently, we examine the atoms of our realization that are contained in U_{n+1}. We make three claims:

(1) For all $i \in [n]$, the set $U_{\overline{i}}$ intersects U_{n+1} only at points contained in $U_{[n]\setminus\{i\}} \setminus U_i$,

(2) The sets U_i for $i \in [n]$ cover U_{n+1}, and

(3) $U_{[n+1]}$ is nonempty.

For (1), let $x = (x_1, \ldots, x_n)$ be a point in $U_{n+1} \cap U_{\bar{i}}$, noting that all coordinates of x are positive. The definition of $U_{\bar{i}}$ implies that $x_j < 1$ for every $j \in [n] \setminus \{i\}$. Thus $\sum_{j \in [n] \setminus \{i\}} x_j < n - 1$, and $x \notin U_i$. Additionally, the definition of U_{n+1} implies that

$$2n - 1 < \sum_{j \in [n]} x_j < x_i + n - 1$$

which implies that $x_i > n$. This shows that our point is in U_j for $j \in [n] \setminus \{i\}$, proving the first claim.

Claim (2) can be argued as follows: any point $x = (x_1, \ldots, x_n)$ in U_{n+1} by definition has

$$2n - 1 < \sum_{i \in [n]} x_i < 2n$$

and all x_i positive. This implies that there is $i \in [n]$ with $x_i < 2$. Deleting this x_i from the sum, this yields $\sum_{j \in [n] \setminus \{i\}} x_j > 2n - 3$. But since $n \geq 2$, we have $2n - 3 \geq n - 1$. Therefore $\sum_{j \in [n] \setminus \{i\}} x_j > n - 1$, and x lies in U_i. This proves the second claim.

Finally, we prove claim (3). Consider the point x whose coordinates are all equal to $2 - \frac{1}{2n}$. This lies in U_{n+1} since the sum of its coordinates is $2n - \frac{1}{2}$. It also lies in U_i for all $i \in [n]$, since the sum of any $n - 1$ of its coordinates is equal to $2n + \frac{1}{2n} - \frac{5}{2}$ which is larger than $n - 1$ since $n \geq 2$. Thus $U_{[n+1]}$ is nonempty.

Now, (1), (2), and (3) imply that any codeword in $\mathcal{D}$ that contains $n + 1$ is one of the codewords of type (ii), (iv), or (v) in $\mathcal{C}_n^{\overline{(n+1)}}$. From this and our discussion of codewords in $\mathcal{D}$ that do not contain $n + 1$, we conclude that $\mathcal{D} \subseteq \mathcal{C}_n^{\overline{(n+1)}}$. Moreover, we saw that codewords of types (iv), (v), and (vi) all arise in $\mathcal{D}$. Since these are the maximal codewords of $\mathcal{C}_n^{\overline{(n+1)}}$ we conclude that $\mathcal{C}_n^{\overline{(n+1)}} \subseteq \Delta(\mathcal{D})$. Monotonicity of convexity then implies that $\mathcal{C}_n^{\overline{(n+1)}}$is a convex code, proving the result. □

Theorem 5.6.13. *For all $n \geq 2$, the code $\mathcal{C}_n$ is minimally non-convex.*

Proof. We argued in Theorem 5.6.4 that $\mathcal{C}_n$ is not open convex. In Lemmas 5.6.8, 5.6.10, and 5.6.12 we argued that every covered code (and hence every proper minor) of $\mathcal{C}_n$ is open convex. Thus $\mathcal{C}_n$ is minimally non-convex. □

The codes $\mathcal{C}_n$ form an infinite family of minimal obstructions to open convexity. Importantly, these codes fail to be open convex for geometric reasons based on sunflowers (rather than topological reasons such as having a local obstruction). Thus Corollary 5.1.14 not only deepens our understanding of codes with finite open embedding dimension, it also provides fundamentally new examples of codes that fail to be open convex in any dimension.

5.7 Tangled Sunflowers

So far, we have only examined codes whose realizations contain a single open sunflower. In this section we examine a family of codes, each of which describes two sunflowers whose petals touch, thus "tangling" the two sunflowers together. We prove some bounds on the embedding dimensions of these codes (in particular, they have finite but arbitrarily large open embedding dimension). However, unlike the codes $\mathcal{S}_n$ of Section 5.2, we are not able to determine the exact open embedding dimensions of these codes beyond the first few.

Definition 5.7.1. Let $n \geq 1$. The *n-th tangled sunflower code* is the code $\mathcal{T}_n \subseteq 2^{[n]\cup[\overline{n}]}$ that consists of the following codewords:

(i) $\{i, \bar{i}\}$ for all $i \in [n]$,

(ii) $[n]$ and $[\overline{n}]$,

(iii) all singletons, and

(iv) the empty set.

Additionally, we define $t_n := \mathrm{odim}(\mathcal{T}_n)$.

Observe that codewords of type (i) and (ii) are the maximal codewords in $\mathcal{T}_n$ for $n \geq 2$; in particular $\mathcal{T}_n$ has $n+2$ maximal codewords. Furthermore observe that $\mathcal{T}_n$ is intersection complete, and hence open convex. Thus t_n is finite for all n.

Moreover, note that if $\mathcal{U} = \{U_1, \dots, U_n, U_{\overline{1}}, \dots, U_{\overline{n}}\}$ is a realization of $\mathcal{T}_n$ then $\{U_1, \dots, U_n\}$ and $\{U_{\overline{1}}, \dots, U_{\overline{n}}\}$ are both sunflowers. Codewords of type (i) imply that U_i intersects $U_{\overline{i}}$ for all $i \in [n]$, but no further overlap between these two sunflowers can occur.

Note that, similar to the codes $\mathcal{S}_n$, the closed embedding dimensions of the codes $\mathcal{T}_n$ are not of much interest. This is formalized below. Example 5.7.3 then gives a sense of how the open realizations of $\mathcal{T}_n$ behave for small n.

Proposition 5.7.2. $\operatorname{cdim}(\mathcal{T}_1) = 1$, *and* $\operatorname{cdim}(\mathcal{T}_n) = 2$ *for all* $n \geq 2$.

Proof. The open realization of $\mathcal{T}_1$ shown in Figure 5.19 may be converted to a closed realization by replacing the sets with their closures. Observe that $\operatorname{cdim}(\mathcal{T}_2) \geq 2$ since $\Delta(\mathcal{T}_2)$ has a nonzero first homology group. By Proposition 5.7.4 we then have $\operatorname{cdim}(\mathcal{T}_n) \geq 2$ for all $n \geq 2$. Finally, one may form a closed realization of $\mathcal{T}_n$ in $\mathbb{R}^2$ by choosing distinct points p_1 and p_2 in $\mathbb{R}^2$, then choosing distinct points $\{q_1, \dots, q_n\}$ on a line separating p_1 and p_2, and defining $X_i = \overline{p_1 q_i}$ and $X_{\overline{i}} = \overline{p_2 q_i}$ for all $i \in [n]$. This construction is shown in Figure 5.18.

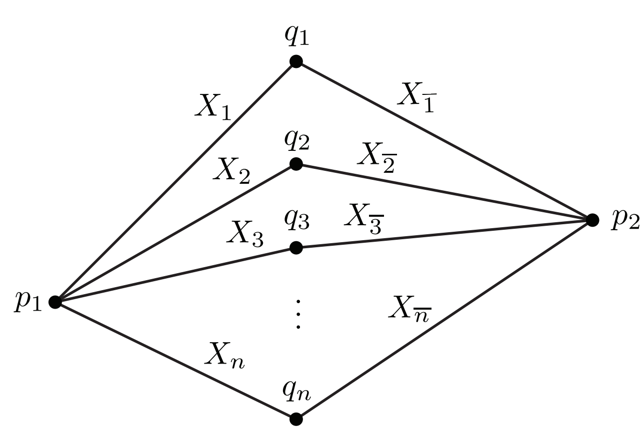


Figure 5.18: A closed realization of $\mathcal{T}_n$ in $\mathbb{R}^2$.

□

Example 5.7.3. The first four $\mathcal{T}_n$ are given below:

$$\mathcal{T}_1 = \{\mathbf{1\overline{1}}, 1, \overline{1}, \emptyset\},$$
$$\mathcal{T}_2 = \{\mathbf{12}, \overline{\mathbf{12}}, \mathbf{1\overline{1}}, \mathbf{2\overline{2}}, 1, 2, \overline{1}, \overline{2}, \emptyset\},$$
$$\mathcal{T}_3 = \{\mathbf{123}, \overline{\mathbf{123}}, \mathbf{1\overline{1}}, \mathbf{2\overline{2}}, \mathbf{3\overline{3}}, 1, 2, 3, \overline{1}, \overline{2}, \overline{3}, \emptyset\},$$
$$\mathcal{T}_4 = \{\mathbf{1234}, \overline{\mathbf{1234}}, \mathbf{1\overline{1}}, \mathbf{2\overline{2}}, \mathbf{3\overline{3}}, \mathbf{4\overline{4}}, 1, 2, 3, 4, \overline{1}, \overline{2}, \overline{3}, \overline{4}, \emptyset\}.$$

These have open realizations in $\mathbb{R}^1, \mathbb{R}^2$, $\mathbb{R}^3$ and $\mathbb{R}^3$ respectively, shown in Figure 5.19.

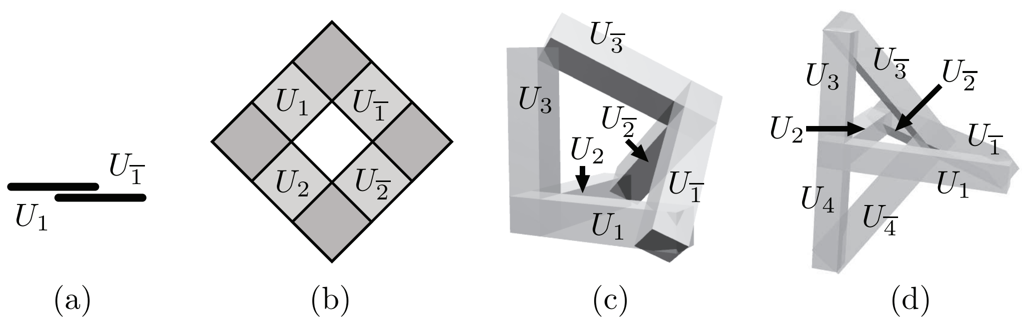


Figure 5.19: Open realizations (a), (b), (c), and (d) of $\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3$, and $\mathcal{T}_4$ in dimensions 1, 2, 3, and 3 respectively.

We will see that in fact each of the realizations in Figure 5.19 is minimal with respect to dimension. That is, we will prove that $t_1 = 1, t_2 = 2$, and $t_3 = t_4 = 3$. To build towards this result, we first prove some general results about the sequence $\{t_n \mid n \geq 1\}$.

Proposition 5.7.4. *For any $n \geq 1$, the code $\mathcal{T}_n$ is a minor of $\mathcal{T}_{n+1}$. In particular, the codes $\{\mathcal{T}_n \mid n \geq 1\}$ form a chain in $\mathbf{P}_{\mathbf{Code}}$.*

Proof. The code $\mathcal{T}_n$ is the restriction of $\mathcal{T}_{n+1}$ to the set of neurons $[n] \cup [\overline{n}]$. □

Proposition 5.7.5. *Let $n \geq 1$. Then $t_n \leq t_{n+1} \leq t_n + 1$. That is, the sequence $\{t_n \mid n \geq 1\}$ is weakly increasing and changes by at most 1 at each step.*

Proof. The inequality $t_n \leq t_{n+1}$ follows immediately from Proposition 5.7.4. To prove the inequality $t_{n+1} \leq t_n + 1$, let $\mathcal{U} = \{U_1, \dots, U_n, U_{\overline{1}}, \dots, U_{\overline{n}}\}$ be an open realization of $\mathcal{T}_n$ in $\mathbb{R}^d$. We will use $\mathcal{U}$ to construct an open realization of $\mathcal{T}_{n+1}$ in $\mathbb{R}^{d+1}$.

By Lemma 2.2.4 we may assume that $\mathcal{U}$ is a non-degenerate realization. In particular, we may assume that the distance between $U_{[n]}$ and $U_{[\overline{n}]}$ is positive. Now, identify $\mathbb{R}^d$ with the subspace of $\mathbb{R}^{d+1}$ in which $x_{d+1} = 0$, and define $U = U_{[n]}$ and $U' = U_{[\overline{n}]}$. Up to translation, we may assume that the origin lies in U, and choose a nonzero vector $u \in U'$. Finally, for a small $\varepsilon > 0$ we define a collection of convex open sets $\mathcal{V} = \{V_1, \dots, V_{n+1}, V_{\overline{1}}, \dots, V_{\overline{n+1}}\}$ in $\mathbb{R}^{d+1}$ as follows:

$$V_i = \begin{cases} U_i + (0,\varepsilon)e_{d+1} & \text{if } i \in [n], \\ U + (0,\infty)e_{d+1} & \text{if } i = n+1, \end{cases} \quad \text{and} \quad V_{\overline{i}} = \begin{cases} U_{\overline{i}} + (0,\varepsilon)(e_{d+1} - u) & \text{if } i \in [n], \\ U' + (0,\infty)(e_{d+1} - u) & \text{if } i = n+1, \end{cases}$$

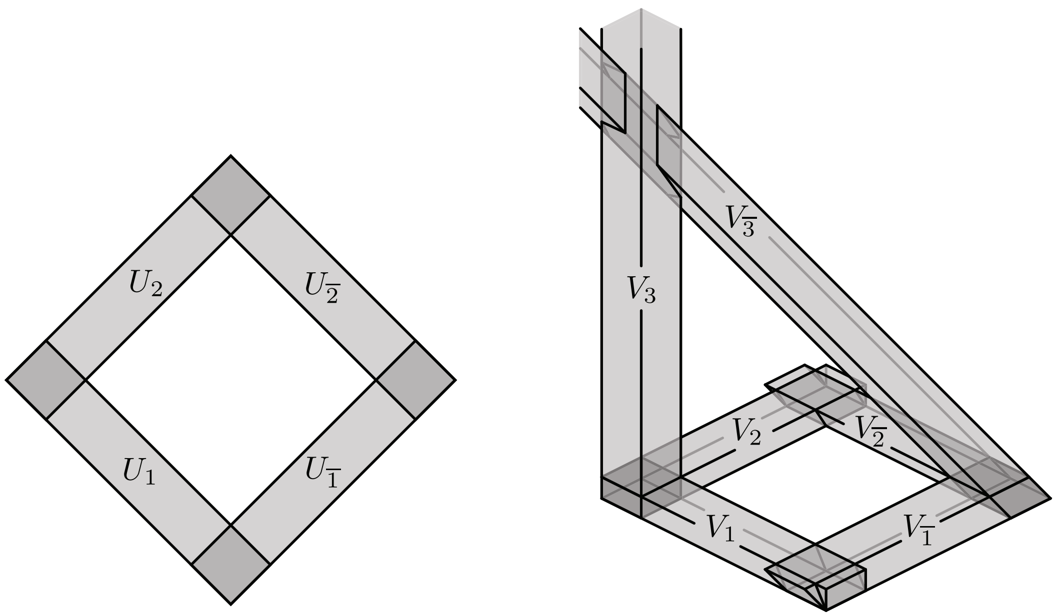
where $(0,\varepsilon)e_{d+1}$ denotes the set of multiples of e_{d+1} between 0 and ε (and similarly in the other three cases). In words, V_i is a small open cylinder over U_i for $i \in [n]$, and V_{n+1} is an infinite open cylinder over U, while $V_{\overline{i}}$ is a small skewed open cylinder over $U_{\overline{i}}$ and $V_{\overline{n+1}}$ is an infinite skewed open cylinder over U'. An example of this construction is shown in Figure 5.20 in the case $n = d = 2$.

We claim that $\mathcal{V}$ is a realization of $\mathcal{T}_{n+1}$. It suffices to check that the collections

$$\{V_1, \dots, V_{n+1}\} \quad \text{and} \quad \{V_{\overline{1}}, \dots, V_{\overline{n+1}}\}$$

are both sunflowers, and that $V_i \cap V_{\overline{j}}$ is nonempty if and only if $i = j$.

Note that $\{V_1, \dots, V_n\}$ is a sunflower since $\{U_1, \dots, U_n\}$ is a sunflower and the V_i with $i \in [n]$ are uniform cylinders over the U_i. Adding V_{n+1} to this collection still yields a sunflower since V_{n+1} is a cylinder over the center of this sunflower. Similar logic holds for the $V_{\overline{i}}$: the collection $\{V_{\overline{1}}, \dots, V_{\overline{n}}\}$ is a sunflower since we have uniformly extended cylinders over the



(a) (b)

Figure 5.20: (a) An open realization $\mathcal{U} = \{U_1, U_2, U_{\overline{1}}, U_{\overline{2}}\}$ of $\mathcal{T}_2$ in $\mathbb{R}^2$. (b) The resulting open realization $\mathcal{V} = \{V_1, V_2, V_3, V_{\overline{1}}, V_{\overline{2}}, V_{\overline{3}}\}$ of $\mathcal{T}_3$ in $\mathbb{R}^3$.

sunflower $\{U_{\overline{1}}, \ldots, U_{\overline{n}}\}$, and the additional petal $V_{\overline{n+1}}$ only overlaps any other $V_{\bar{i}}$ inside the region $U' + (0, \varepsilon)(e_{d+1} - u)$, which is the center of this sunflower.

To see that the petals of our two sunflowers overlap appropriately, first note that $V_i \cap V_{\bar{i}}$ is nonempty for all $i \in [n]$ since the same holds for $U_i \cap U_{\bar{i}}$. Moreover, e_{d+1} is an element of $V_{n+1} \cap V_{\overline{n+1}}$, and so this intersection is nonempty. It remains to show that no further overlaps occur between our two sunflowers.

Formally, we must show that for every choice of distinct i and j in $[n+1]$, the intersection $V_i \cap V_{\bar{j}}$ is empty. We consider two cases. If $i \in [n]$, then by non-degeneracy of the realization $\mathcal{U}$, the set U_i has positive distance to $V_{\bar{j}}$. Since V_i is a small cylinder of height ε over U_i, we may assume ε is small enough that V_i does not intersect $V_{\bar{j}}$. If $i = n+1$, then $j \in [n]$ and a similar logic applies. Namely, $U_{\bar{j}}$ has positive distance to V_i by non-degeneracy of $\mathcal{U}$, and we may choose ε small enough that the skewed cylinder $V_{\bar{j}}$ over $U_{\bar{j}}$ does not meet V_i. We conclude that $\mathcal{V}$ is an open realization of $\mathcal{T}_{n+1}$ in $\mathbb{R}^{d+1}$, proving the result. □

Propositions 5.7.4 and 5.7.5 imply that the sequence $\{t_n \mid n \geq 1\}$ weakly increases, and at a rate that is no more than linear. So far, we have not used the fact that sunflowers appear in realizations of $\mathcal{T}_n$. Below we leverage this fact to prove that the sequence $\{t_n \mid n \geq 1\}$ takes on arbitrarily large values, and in fact takes on all positive integer values.

Theorem 5.7.6. *Let $n \geq 1$. Then $\lceil n/2 \rceil \leq t_n \leq n$. In particular, the sequence $\{t_n \mid n \geq 1\}$ is unbounded.*

Proof. The fact that $t_n \leq n$ follows from the fact that $t_1 = 1$ and Proposition 5.7.5. To prove that $\lceil n/2 \rceil \leq t_n$, let $d = \lceil n/2 \rceil - 1$. We must show that $\mathcal{T}_n$ does not have an open realization in $\mathbb{R}^d$. Suppose for contradiction that such a realization $\mathcal{U} = \{U_1, \ldots, U_n, U_{\bar{1}}, \ldots, U_{\bar{n}}\}$ exists, and let H be a hyperplane separating $U_{[n]}$ and $U_{[\bar{n}]}$.

Choose points $p_1 \in U_{[n]}$ and $p_2 \in U_{[\bar{n}]}$, and for every $i \in [n]$ choose a point $q_i \in U_i \cap U_{\bar{i}}$. For all $i \in [n]$ consider the line segments $L_i = \overline{p_1 q_i}$ and $M_i = \overline{q_i p_2}$. The union $L_i \cup M_i$ forms a path that begins on one side of H and ends on the other, so for all $i \in [n]$ either L_i or M_i contains a point in H. By choice of d and the pigeonhole principle, either at least $d+1$ of the line segments $\{L_i \mid i \in [n]\}$ contain a point in H, or at least $d+1$ of the line segments $\{M_i \mid i \in [n]\}$ contain a point in H.

Without loss of generality, we may assume that at least $d+1$ of the L_i contain a point p_i^* in H. The convex hull of these p_i^* lies in H, and therefore does not intersect the center $U_{[n]}$ of the sunflower $\{U_1, \ldots, U_n\}$. But $L_i \subseteq U_i$, so each p_i^* lies in the petal U_i. Since there are at least $d+1$ of the p_i^*, Corollary 5.1.14 implies that their convex hull must intersect $U_{[n]}$, a contradiction. □

Corollary 5.7.7. *The sequence $\{t_n \mid n \geq 1\}$ takes on all positive integer values.*

Proof. We know that $t_1 = 1$. Theorem 5.7.6 implies that the sequence is unbounded, and Proposition 5.7.5 tells us that it increases by at most one at each step. Thus it must achieve every positive integer value. □

In the remainder of this section, we determine t_n for all $n \leq 5$. The arguments used below are concrete, but seem difficult to generalize. Determining even t_6 would be a potentially interesting next step.

Proposition 5.7.8. *The code $\mathcal{T}_3$ does not have an open realization in $\mathbb{R}^2$, but does have an open realization in $\mathbb{R}^3$. In other words, $t_3 = 3$.*

Proof. An open realization of $\mathcal{T}_3$ in $\mathbb{R}^3$ is given in Figure 5.19. Thus we just have to argue that $\mathcal{T}_3$ does not have a convex realization in $\mathbb{R}^2$. Suppose for contradiction that we have an open realization $\mathcal{U} = \{U_1, U_2, U_3, U_{\overline{1}}, U_{\overline{2}}, U_{\overline{3}}\}$ of $\mathcal{T}_3$ in $\mathbb{R}^2$. Choose points $q_1 \in U_1 \cap U_{\overline{1}}$, $q_2 \in U_2 \cap U_{\overline{2}}$, and $q_3 \in U_3 \cap U_{\overline{3}}$. Since we are working with open sets we may assume that $\{q_1, q_2, q_3\}$ are in general position.

Note that $\{U_1, U_2, U_3\}$ and $\{U_{\overline{1}}, U_{\overline{2}}, U_{\overline{3}}\}$ are both open sunflowers and that the points $\{q_1, q_2, q_3\}$ each lie in distinct petals of these sunflowers. By Corollary 5.1.14 the triangle $\text{conv}\{q_1, q_2, q_3\}$ contains points $p_1 \in U_{[3]}$ and $p_2 \in U_{\overline{[3]}}$. Again since we are working with open sets we may assume that p_1 and p_2 lie in the interior of this triangle. The set of points $\{p_1, q_1, q_2, q_3\}$ can be visualized as shown part (a) of Figure 5.21.

Now, p_2 falls in one of the three triangular regions surrounding p_1. Suppose that p_2 lies in $\text{conv}\{p_1, q_1, q_2\}$. Then consider the line segment $L = \overline{p_2 q_3}$, observing that L is contained in $U_{\overline{3}}$. The line segment L must cross either the line segment $\overline{p_1 q_1} \subseteq U_1$ or $\overline{p_1 q_2} \subseteq U_2$. In the former case we see that $U_1 \cap U_{\overline{3}} \neq \emptyset$, and in the latter $U_2 \cap U_{\overline{3}} \neq \emptyset$. The latter case is shown in part (b) Figure 5.21. But there is no codeword in $\mathcal{T}_3$ containing $1\overline{3}$ or $2\overline{3}$, so both of these situations lead to a contradiction. Thus $\mathcal{T}_3$ is not open convex in $\mathbb{R}^2$. □

The technical lemma below will allow us to prove that $t_5 \geq 4$ by showing that if $\mathcal{T}_5$ has a realization in $\mathbb{R}^3$, then $\mathcal{T}_3$ has a realization in $\mathbb{R}^2$, contradicting Proposition 5.7.8.

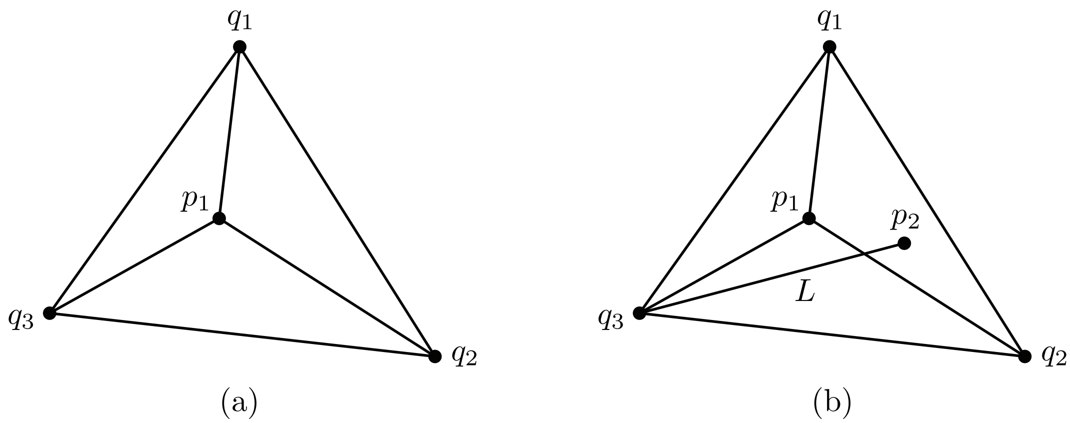


Figure 5.21: (a) The arrangement of points used to prove Proposition 5.7.8. (b) A contradictory line segment $L = \overline{p_2 q_3} \subseteq U_{\bar{3}}$ that crosses $\overline{p_1 q_2} \subseteq U_2$.

Lemma 5.7.9. *Given five points in $\mathbb{R}^3$ in general position, there exists a plane H containing three of the points and with the remaining two points on opposite sides of H.*

Proof. Up to affine transformation we may assume that our set of points is $\{0, e_1, e_2, e_3, p\}$ where p is a point none of whose coordinates are zero. We consider two cases. First suppose that one of the coordinates of p is negative. By permuting our coordinates we can assume this is the last coordinate. Then choose $H = \text{span}\{e_1, e_2\}$. This contains the three points $0, e_1$, and e_2. Moreover since e_3 has positive last coordinate and p has negative last coordinate, they lie on opposite sides of H and the lemma follows.

Otherwise every coordinate of p is positive. In this case, write $p = (x, y, z)$ and choose $H = \text{span}\{e_3, p\}$. Observe that H contains the three points $0, e_3$, and p, and that $v = (y, -x, 0)$ is a normal vector to H. We see that $v \cdot e_1 > 0$ and $v \cdot e_2 < 0$, so the remaining two points e_1 and e_2 lie on opposite sides of H. This proves the result. □

Proposition 5.7.10. *The code $\mathcal{T}_5$ does not have an open realization in $\mathbb{R}^3$.*

Proof. Suppose for contradiction that $\mathcal{U} = \{U_1, \ldots, U_5, U_{\bar{1}}, \ldots, U_{\bar{5}}\}$ is an open realization of $\mathcal{T}_5$ in $\mathbb{R}^3$. For $i \in [5]$, choose a point p_i in the open set $U_i \cap U_{\bar{i}}$, such that all p_i are in general position. Applying Lemma 5.7.9 to these five points, we obtain a hyperplane H with contains three of them, and with the remaining two on opposite sides. By permuting the neurons in $\mathcal{T}_5$, we may assume that p_1, p_2, and p_3 all lie in H.

Now, consider the two tetrahedra $\Delta_1 = \text{conv}\{p_1, p_2, p_3, p_4\}$ and $\Delta_2 = \text{conv}\{p_1, p_2, p_3, p_5\}$. The vertices of these tetrahedra belong to distinct petals of the sunflowers $\{U_1, \ldots, U_5\}$ and $\{U_{\bar{1}}, \ldots, U_{\bar{5}}\}$, so by Corollary 5.1.14 each of these tetrahedra contains a point in the center of both of these sunflowers. Since the tetrahedra lie on opposite sides of H, each of the centers of these two sunflowers contains a point on each side of H. But the center of a sunflower is convex, and so H itself must contain a point in the center of each of the two sunflowers.

With this observation, consider the collection $\mathcal{V} = \{V_1, V_2, V_3, V_{\bar{1}}, V_{\bar{2}}, V_{\bar{3}}\}$ where $V_i = U_i \cap H$ and $V_{\bar{i}} = U_{\bar{i}} \cap H$ for all $i \in [3]$. Since $H \cong \mathbb{R}^2$, we can regard $\mathcal{V}$ as an open realization of a code in $\mathbb{R}^2$. We claim that in fact $\text{code}(\mathcal{V}) = \mathcal{T}_3$. To verify this, it suffices to show that (i) $\{V_1, V_2, V_3\}$ and $\{V_{\bar{1}}, V_{\bar{2}}, V_{\bar{3}}\}$ are both sunflowers, (ii) that $V_i \cap V_{\bar{i}}$ is nonempty for all $i \in [3]$, and (iii) that no other petals overlap between these two sunflowers.

Condition (i) follows from the fact that the various V_i and $V_{\bar{i}}$ are subsets of U_i and $U_{\bar{i}}$ respectively, and that the sunflowers making up the realization of $\mathcal{T}_5$ both have centers that intersect H. Condition (ii) follows by considering the points p_1, p_2, and p_3, which lie in $V_1 \cap V_{\bar{1}}$, $V_2 \cap V_{\bar{2}}$, and $V_3 \cap V_{\bar{3}}$ respectively. Finally, condition (iii) is a consequence of the fact that the petals of our two sunflowers in $\mathcal{U}$ overlap appropriately.

However, this is a contradiction: $\mathcal{T}_3$ is not open convex in $\mathbb{R}^2$ by Proposition 5.7.8. Thus $\mathcal{T}_5$ cannot be open convex in $\mathbb{R}^3$. □

Corollary 5.7.11. *The sequence t_n begins as follows:*

n	*1*	*2*	*3*	*4*	*5*
t_n	*1*	*2*	*3*	*3*	*4*

Proof. Clearly $t_1 = 1$ since $\mathcal{T}_1$ is convex in $\mathbb{R}^1$ but has more than one codeword, so is not convex in $\mathbb{R}^0$. The code $\mathcal{T}_2$ has a realization in $\mathbb{R}^2$ as shown in Figure 5.19, but has no realization in $\mathbb{R}^1$ since $\Delta(\mathcal{T}_2)$ has nonzero first homology. Thus $t_2 = 2$.

Note that $t_3 \leq 3$ and $t_4 \leq 3$ as shown in Figure 5.19. These bounds are tight by Propositions 5.7.8 and 5.7.5. By Proposition 5.7.10 we know that $t_5 \geq 4$, and simultaneously Proposition 5.7.5 implies that $t_5 \leq t_4 + 1 = 4$. This proves the result. □

The codes $\mathcal{T}_n$ hint at many further applications of sunflowers to the study of open embedding dimension. One could imagine families of codes in which more than two sunflowers are "tangled" together, or the tangling is more complicated than the 1-to-1 intersection of matched petals in the $\mathcal{T}_n$ codes. One could also define codes which tangle together flexible sunflowers. Finding an elegant formulation for these various generalizations could be a fruitful subject of future work.

5.8 Flexible Sunflower Codes Generalizing $\mathcal{S}_n$ and $\mathcal{S}_\Delta$

All of the families of codes that we have worked with so far have only had realizations that involve 1-flexible open sunflowers, and hence only require Corollary 5.1.14 to analyze. In this section we provide more general families of examples, for which Theorem 5.1.13 is essential to bounding open embedding dimension. While the families $\mathcal{S}_n$ and $\mathcal{S}_\Delta$ were parametrized by integers and simplicial complexes respectively, our family of interest in this section is parametrized by pairs of intersection complete codes, one of which contains the other.

Definition 5.8.1. Let $\mathcal{D} \subseteq \mathcal{C} \subseteq 2^{[n]}$ be intersection complete codes. We define

$$\mathcal{S}_{\mathcal{C}/\mathcal{D}} := \mathcal{C} \cup \{[n]\} \cup \{d \cup \{n+1\} \mid d \in \mathcal{D}\} \subseteq 2^{[n+1]}.$$

Note that choosing $\mathcal{D} = \{\text{minimal nonempty codewords in } \mathcal{C}\} \cup \{\emptyset\}$ always satisfies the above conditions. In this case, we will let $\mathcal{S}_{\mathcal{C}/\min}$ denote $\mathcal{S}_{\mathcal{C}/\mathcal{D}}$.

Qualitatively, $\mathcal{S}_{\mathcal{C}/\mathcal{D}}$ is the result of forming a flexible sunflower using the codewords in $\mathcal{C}$, and then "gluing" the petals of that sunflower to a new set U_{n+1} along codewords in $\mathcal{D}$. Observe that $\mathcal{S}_\Delta$ is equal to $\mathcal{S}_{\Delta/\Delta}$ in this notation (see Definition 5.3.1). Also, if $\mathcal{C} = \{\mathbf{1}, \ldots, \mathbf{n}, \emptyset\}$, then $\mathcal{S}_n$ is equal to $\mathcal{S}_{\mathcal{C}/\min}$ (recall Definition 5.2.1). The following proposition may be viewed as a direct generalization of Proposition 5.3.2.

Proposition 5.8.2. *Let $\mathcal{D} \subseteq \mathcal{C} \subseteq 2^{[n]}$ be intersection complete codes. The code $\mathcal{S}_{\mathcal{C}/\mathcal{D}}$ is intersection complete. If $\mathcal{D}$ has m maximal codewords and does not contain $[n]$, then $\mathcal{S}_{\mathcal{C}/\mathcal{D}}$ has $m+1$ maximal codewords. In particular,* $\operatorname{odim}(\mathcal{S}_{\mathcal{C}/\mathcal{D}}) \le \max\{2, m\}$.

Proof. Codewords in $\mathcal{S}_{\mathcal{C}/\mathcal{D}}$ come in three types: codewords from $\mathcal{C}$, the codeword $[n]$, and those of the form $d \cup \{n+1\}$ where d is a codeword in $\mathcal{D}$. Since $\mathcal{C}$ and $\mathcal{D}$ are intersection complete, the intersection of two codewords of the same type always yields another codeword of that type (and hence lying in $\mathcal{S}_{\mathcal{C}/\mathcal{D}}$). This leaves the intersections of codewords of different types. The intersection of a codeword in $\mathcal{C}$ with $[n]$ is simply the same codeword in $\mathcal{C}$. The intersection of $d \cup \{n+1\}$ with $[n]$ is just d, which lies in $\mathcal{S}_{\mathcal{C}/\mathcal{D}}$ since $\mathcal{D} \subseteq \mathcal{C}$. Finally, the intersection of $c \in \mathcal{C}$ with $d \cup \{n+1\}$ is $c \cap d$, which lies in $\mathcal{C}$ since $\mathcal{C}$ is intersection complete.

For the second part of the statement, note that if d is a maximal codeword of $\mathcal{D}$, then $d \cup \{n+1\}$ is a maximal codeword of $\mathcal{S}_{\mathcal{C}/\mathcal{D}}$. Since $[n] \notin \mathcal{D}$, the codeword $[n]$ is also a maximal codeword of $\mathcal{S}_{\mathcal{C}/\mathcal{D}}$, yielding $m+1$ total maximal codewords. The bound on $\operatorname{odim}(\mathcal{S}_{\mathcal{C}/\mathcal{D}})$ follows immediately from [CGIK16, Theorem 1.2]. □

Continuing in this vein, we generalize Theorem 5.2.2 (which applied to $\mathcal{S}_n$) to the family of codes of the form $\mathcal{S}_{\mathcal{C}/\min}$.

Proposition 5.8.3. *Let $\mathcal{C} \subseteq 2^{[n]}$ be an intersection complete code which contains every singleton set. Then*

$$\operatorname{odim}(\mathcal{S}_{\mathcal{C}/min}) \ge \left\lceil \frac{n}{\dim(\Delta(\mathcal{C}))+1} \right\rceil .$$

Proof. We start with a degenerate case: if $n = 1$, then $\mathcal{C} = \{\mathbf{1}, \emptyset\}$ and $\mathcal{S}_{\mathcal{C}/\min} = \{\mathbf{12}, 1, 2, \emptyset\}$. In this case $\operatorname{odim}(\mathcal{S}_{\mathcal{C}/\min}) = 1$, while $n = 1$ and $\dim(\Delta(\mathcal{C})) + 1 = 1$. We see that the bound given above is satisfied as desired.

Otherwise, $n \ge 2$. In this case, let $\mathcal{U} = \{U_1, \ldots, U_{n+1}\}$ be an open convex realization of $\mathcal{S}_{\mathcal{C}/\min}$ in $\mathbb{R}^d$. Since the minimal nonempty codewords of $\mathcal{C}$ are all singletons, the code $\mathcal{S}_{\mathcal{C}/\min}$ consists of codewords from $\mathcal{C}$, the codeword $[n]$, codewords of the form $\{i, n+1\}$ for all $i \in [n]$, and lastly the codeword $\{n+1\}$. Since $[n]$ is a codeword, the sets $\{U_1, \ldots, U_n\}$

have a nonempty common intersection. In particular, $\{U_1, \dots, U_n\}$ is a k-flexible sunflower, where k is the largest weight of a codeword in $\mathcal{C}$ other than possibly $[n]$. In particular $k \le \dim(\Delta(\mathcal{C})) + 1$, with equality if $[n]$ is not a codeword in $\mathcal{C}$.

Now consider the set U_{n+1}. This set does not meet $U_{[n]}$ since $[n+1]$ is not a codeword of $\mathcal{S}_{\mathcal{C}/\min}$. However, it does meet U_i for all $i \in [n]$ since $\{i, n+1\}$ is a codeword. If we choose $p_i \in U_i \cap U_{n+1}$ for all $i \in [n]$, then $\operatorname{conv}\{p_1, \dots, p_n\}$ is contained in U_{n+1} and therefore does not contain a point in $U_{[n]}$. By Theorem 5.1.13, such a sampling of points is impossible if $n \ge dk+1$. Therefore we must have $n \le dk$. Rearranging, this implies $d \ge \lceil n/k \rceil$. Applying the inequality $k \le \dim(\Delta(\mathcal{C})) + 1$ yields the desired result. □

The added assumption in Proposition 5.8.3 that $\mathcal{C}$ contains all singletons is not too restrictive. Indeed, it ensures that the set of minimal nonempty codewords interacts non-trivially with all neurons, and adding singleton sets to $\mathcal{C}$ preserves intersection completeness. Our final result in this section provides a simultaneous generalization of Theorem 5.3.4 and Proposition 5.3.3.

Proposition 5.8.4. *Let $\mathcal{D} \subseteq \mathcal{C} \subseteq 2^{[n]}$ be intersection complete codes. Let m be the number of maximal codewords in $\mathcal{D}$, assume $m \ge 2$, and let k be the largest number of maximal codewords in $\mathcal{D}$ whose union lies in $\Delta(\mathcal{C})$. Then there exists an intersection complete code $\mathcal{E} \subseteq 2^{[m]}$ containing all singleton sets such that*

(i) $k = \dim(\Delta(\mathcal{E})) + 1$, *and*

(ii) $\mathcal{S}_{\mathcal{E}/min}$ *is a minor of* $\mathcal{S}_{\mathcal{C}/\mathcal{D}}$.

As a consequence, $\lceil \frac{m}{k} \rceil \le \operatorname{odim}(\mathcal{S}_{\mathcal{C}/\mathcal{D}}) \le m$.

Proof. Let $\sigma_1, \dots, \sigma_m$ be the maximal codewords of $\mathcal{D}$. For $i \in [m]$ define $T_i = \operatorname{Tk}_{\mathcal{S}_{\mathcal{C}/\mathcal{D}}}(\sigma_i)$, and define $T_{m+1} = \operatorname{Tk}_{\mathcal{S}_{\mathcal{C}/\mathcal{D}}}(n+1)$. Let $f : \mathcal{S}_{\mathcal{C}/\mathcal{D}} \to 2^{[m]}$ be the morphism determined by the collection of trunks $\{T_1, \dots, T_{m+1}\}$. Recall from Definition 5.8.1 that the codewords of $\mathcal{S}_{\mathcal{C}/\mathcal{D}}$ come in the following types:

- c where c is a codeword in $\mathcal{C}$,

- $d \cup \{n+1\}$ for where d is a codeword in $\mathcal{D}$, and

- $[n]$.

The respective images of these codewords under f are as follows:

- $f(c)$ is equal to $\{i \in [m] \mid c \text{ contains } \sigma_i\}$,

- $f(d \cup \{n+1\})$ is equal to $\{m+1\}$ if d is not equal to some σ_i, and is equal to $\{i, m+1\}$ if $d = \sigma_i$ for some $i \in [m]$, and

- $f([n]) = [m]$ since $[n]$ contains all maximal codewords in $\mathcal{D}$, but not $n+1$.

Let $\mathcal{E} = f(\mathcal{C}) \subseteq 2^{[m]}$. Since the image of an intersection complete code is again intersection complete, we see that $\mathcal{E}$ is intersection complete. Moreover, $\mathcal{E}$ contains every singleton set since $f(\sigma_i) = \{i\}$.

Observe that $f(\mathcal{S}_{\mathcal{C}/\mathcal{D}})$ contains $\mathcal{E}$, as well as codewords of the form $\{i, m+1\}$ for all $i \in [m]$, the codeword $\{m+1\}$, and the codeword $[m]$. These are exactly the codewords of $\mathcal{S}_{\mathcal{E}/\min}$. Thus $\mathcal{S}_{\mathcal{E}/\min} = f(\mathcal{S}_{\mathcal{C}/\mathcal{D}})$.

To prove the result, it remains to show that $k = \dim(\Delta(\mathcal{E})) + 1$. The codewords in $\mathcal{E}$ are of the form $f(c) = \{i \in [m] \mid \sigma_i \subseteq c\}$. Thus a codeword in $\mathcal{E}$ corresponds to a collection of maximal codewords in $\mathcal{D}$ all of which are contained in some $c \in \mathcal{C}$. A codeword in $\mathcal{E}$ with largest weight thus corresponds to a largest possible collection of maximal codewords in $\mathcal{D}$ whose union is contained in $\Delta(\mathcal{C})$. The largest such collection has size k by definition, so any largest codeword in $\mathcal{E}$ has weight k, proving the result. $\square$

Remark 5.8.5. Recall from Remark 5.3.5 that $\mathcal{S}_m$ is the unique minimal minor among all $\mathcal{S}_\Delta$ codes where Δ has m facets. Generalizing this, we see that among all codes of the form $\mathcal{S}_{\mathcal{C}/\mathcal{D}}$ with parameters m and k as described in Proposition 5.8.4, the minimal elements (with respect to the partial order inhereted from $\mathbf{P_{Code}}$) are always of the form $\mathcal{S}_{\mathcal{E}/\min}$ where $\mathcal{E} \subseteq 2^{[m]}$ contains all singletons, and $k = \dim(\Delta(\mathcal{E})) + 1$. Figure 5.22 provides an informal picture of this situation.

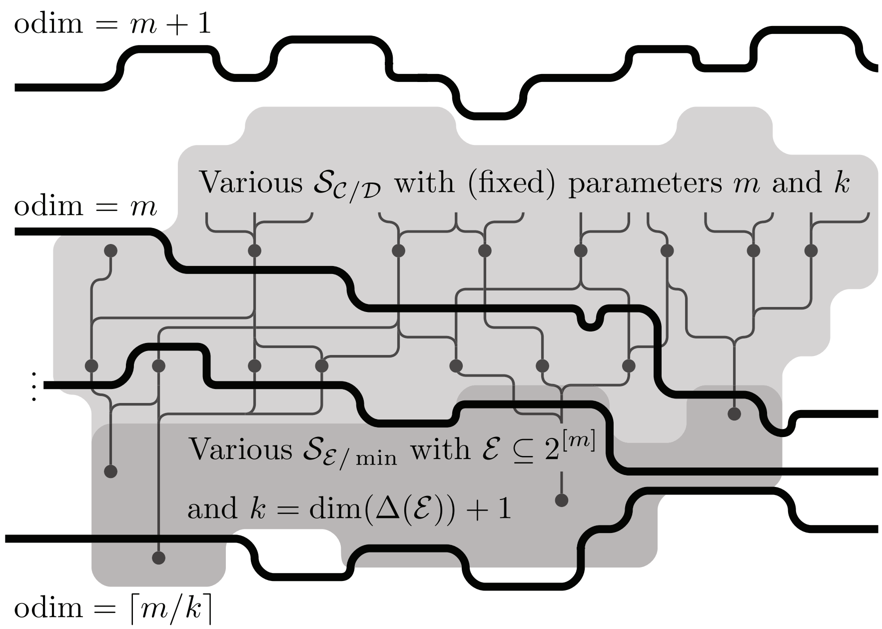


Figure 5.22: The codes $\mathcal{S}_{\mathcal{C}/\mathcal{D}}$ and $\mathcal{S}_{\mathcal{E}/\min}$ for a fixed choice of m and k stratified by open embedding dimension in $\mathbf{P_{Code}}$. Compare with Figure 5.10 which treats the $k = 1$ case.

These results use Theorem 5.1.13 to provide a more complete picture of the open embedding dimensions of intersection complete codes. There is still much to be done, however. For example, the bound $m \geq \text{odim}(\mathcal{S}_{\mathcal{C}/\mathcal{D}}) \geq \left\lceil \frac{m}{k} \right\rceil$ of Proposition 5.8.4 leaves quite a large gap for $k \geq 2$. Sharpening this bound based on the combinatorial structure of $\mathcal{C}$ and $\mathcal{D}$ would be a natural task of interest.

Chapter 6

CONVEX UNION REPRESENTABLE COMPLEXES AND NEW LOCAL OBSTRUCTIONS TO CONVEXITY

In this chapter we establish a strong relationship between open convex covers of a convex set, and the combinatorics of the nerve of such a cover. These results provide new insights in the study of local obstructions to open and closed convexity in codes (recall Definition 1.4.1, and see [CFS19, CGJ$^+$17]). In particular, we expand the infinite family of minimally non-convex codes $\{\mathcal{C}_\Delta \mid \Delta \text{ is not collapsible}\}$ from Theorem 4.4.3 to an infinitely larger family of minimally non-convex codes. More generally we initiate the study of "convex union representable" complexes (see Definition 6.1.1), a family of simplicial complexes that may be of interest in their own right.

These results are joint work with Isabella Novik, and first appeared in [JN19]. With only a few small additions and notational changes, we present these results as they appear in the original work.

6.1 Collapsibility of Convex Union Representable Complexes

Recall that if $\mathcal{C} \subseteq 2^{[n]}$ is a convex code with an open realization $\mathcal{U} = \{U_1, \ldots, U_n\}$, then $\sigma \in \Delta(\mathcal{C}) \setminus \mathcal{C}$ if and only if U_σ is nonempty and covered by the collection $\{U_i \cap U_\sigma \mid i \in [n] \setminus \sigma\}$. We are interested in studying the nerves of covers that arise in this way. Equivalently, we are interested in studying the nerves of collections of convex open sets whose union is convex. These are exactly "convex union representable complexes," defined formally below.

Definition 6.1.1. Let $\Delta \subseteq 2^{[n]}$ be a simplicial complex. We say that Δ is *d-convex union representable* if there is a collection of convex open sets $\mathcal{U} = \{U_1, \ldots, U_n\}$ in $\mathbb{R}^d$ such that

(i) $\bigcup_{i \in [n]} U_i$ is a convex open set, and

(ii) $\Delta = \text{nerve}(\mathcal{U})$.

We say that Δ is *convex union representable* if there exists some d such that Δ is d-convex union representable. The collection $\mathcal{U}$ is called a *d-convex union representation* of Δ.

Example 6.1.2. The complex $\langle 123, 124, 234 \rangle$ is 2-convex union representable, as shown in Figure 6.1.

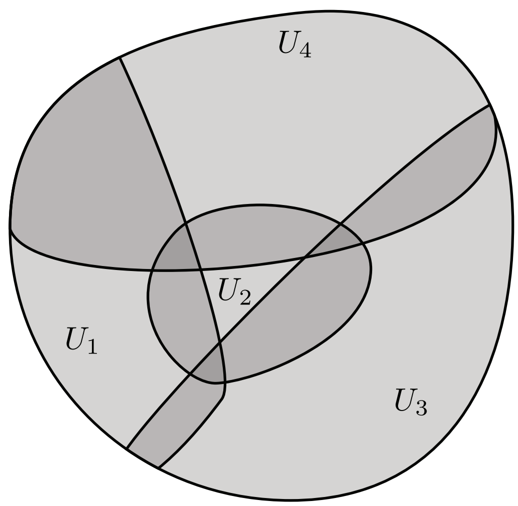


Figure 6.1: A 2-convex union representation $\mathcal{U} = \{U_1, U_2, U_3, U_4\}$ of $\langle 123, 124, 234 \rangle$.

One of the main results of [CFS19] is that convex union representable complexes are collapsible (though [CFS19] does not use the term convex union representable). The authors in [CFS19] asked whether the converse also holds: is every collapsible complex convex union representable? We will answer this question in the negative (see Corollary 6.2.1) using the results that we establish in this section. Before proceeding, we briefly recall the definition of collapsibility.

Definition 6.1.3. Let $\Delta \subseteq 2^{[n]}$ be a simplicial complex. A non-facet $\sigma \in \Delta$ is called a *free face* if σ is contained in a unique facet. The operation of deleting a free face σ

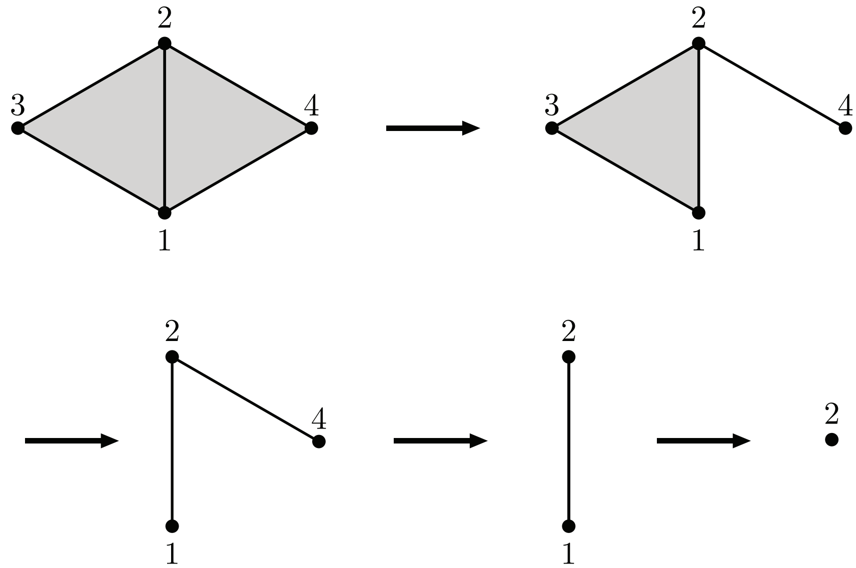


Figure 6.2: Collapsing the complex $\langle 123, 124\rangle$ to a point.

and all faces that contain it is called a *collapse* of Δ, and denoted $\Delta \to \Delta \setminus \sigma$ (where $\Delta \setminus \sigma := \{\tau \in \Delta \mid \sigma \not\subseteq \tau\}$). We say that Δ is *collapsible* if there is a sequence of collapses $\Delta \to \Delta_1 \to \cdots \to \Delta_k = \emptyset$.

For a nonempty complex, collapsibility is equivalent to the statement that a complex collapses to a point. Collapses preserve homotopy type, and thus collapsibility provides a combinatorial analog of contractibility. Importantly, not every contractible complex is collapsible. For example, the dunce hat is contractible, but no triangulation of it is collapsible.

Example 6.1.4. Consider the complex $\Delta = \langle 123, 124\rangle$. Figure 6.2 shows a collapsing sequence from Δ to one of its vertices.

A key result of our work is Theorem 6.1.7, which shows that a collapsing order for a convex union representable complex can be obtained by sweeping a hyperplane across a convex union representation. This approach is due to [Weg75], and was also applied in [CFS19]. To establish this result in our context we require a technical lemma that allows us to “shrink” a given convex union representation by a small amount without changing its nerve. Below, recall that the *Hausdorff distance* $\mathrm{dist}_H(X,Y)$ between bounded subsets X and Y of $\mathbb{R}^d$ is defined as

$$\mathrm{dist}_H(X,Y) := \inf\{\varepsilon \geq 0 \mid X \subseteq Y + B_\varepsilon \text{ and } Y \subseteq X + B_\varepsilon\}.$$

Also recall that the Hausdorff distance makes the space of compact subsets of $\mathbb{R}^d$ into a metric space.

Lemma 6.1.5. *Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be a convex union representation of a simplicial complex $\Delta \subseteq 2^{[n]}$, where all U_i are bounded. Then for all $\varepsilon > 0$ there exists a convex union representation $\mathcal{V} = \{V_1, \ldots, V_n\}$ of Δ with the following properties:*

(i) $\mathrm{cl}(V_i)$ *is a polytope contained in U_i for all $i \in [n]$,*

(ii) The union $\bigcup_{i=1}^n V_i$ is the interior of a polytope, and

(iii) The Hausdorff distance between $\mathrm{cl}(V_\sigma)$ *and* $\mathrm{cl}(U_\sigma)$ *is less than ε for all $\emptyset \neq \sigma \in \Delta$.*

Proof. The result holds if all U_i are empty sets: simply take all V_i to be empty. Thus, assume Δ has at least one vertex, and for each nonempty face $\sigma \in \Delta$, fix a d-dimensional polytope $P_\sigma \subseteq U_\sigma$ with the property that $\mathrm{dist}_H\big(\mathrm{cl}(U_\sigma), P_\sigma\big) < \varepsilon/3$. Let C be the convex hull of the union of all P_σ. Then C is a d-dimensional polytope (hence compact) covered by $\{U_1, \ldots, U_n\}$, so we may choose a Lebesgue number $\delta > 0$ for this cover.

Consider the lattice $(\delta\mathbb{Z})^d$. For every point p in this lattice, let W_p be the closed d-cube with side length 2δ centered at p. Then $W_p \cap C$ is a polytope for all $p \in (\delta\mathbb{Z})^d$. Let S denote the collection of vertices of all nonempty polytopes of this form. Note that S is finite and that S might not be a subset of $(\delta\mathcal{Z})^d$ since some of the cells of the lattice may only partially intersect C.

By shrinking δ, we may assume that the following conditions hold:

(1) Every nonempty set of the form $U_\sigma \cap C$ contains some W_p,

(2) for every $p \in (\delta\mathcal{Z})^d$ with $W_p \cap C \neq \emptyset$, there exists $i \in [n]$ such that $W_p \subseteq U_i$, and

(3) the diameter of W_p is less than $\varepsilon/3$.

For $i \in [n]$, define $V_i = \operatorname{int}(\operatorname{conv}(S \cap U_i))$. Note that condition (1) above guarantees that $\operatorname{conv}(S \cap U_i)$ is full-dimensional, and so $\operatorname{cl}(V_i) = \operatorname{conv}(S \cap U_i)$ is a polytope contained in U_i. We will show that $\mathcal{V} = \{V_1, \dots, V_n\}$ is the desired convex union representation of Δ.

By choice of C, $U_\sigma \cap C \neq \emptyset$ for every nonempty face $\sigma \in \Delta$. Condition (1) then implies that $U_\sigma \cap C$ contains W_p for some $p \in (\delta\mathcal{Z})^d$, which, in turn, implies that $p \in V_i$ for all $i \in \sigma$. Thus $\sigma \in \operatorname{nerve}(\mathcal{V})$. Since $V_i \subseteq U_i$ for all i, we conclude that $\operatorname{nerve}(\mathcal{V}) = \Delta$.

To verify property (ii), we show that $\bigcup_{i=1}^n V_i = \operatorname{int} C$. Indeed, the interior of C is the union of all sets in the collection $\{\operatorname{int} W_p \cap \operatorname{int} C \neq \emptyset \mid p \in (\delta\mathcal{Z})^d\}$. Furthermore, by condition (2), each polytope $W_p \cap C$ is contained in U_i for some i, and so its vertices are in $U_i \cap S$. By definition of V_i this implies that $\operatorname{int}(W_p \cap C) \subseteq V_i$. The assertion follows since $\operatorname{int}(W_p \cap C) = \operatorname{int} W_p \cap \operatorname{int} C$.

For property (iii), note that $P_\sigma \subseteq \operatorname{cl}(U_\sigma) \cap C \subseteq \operatorname{cl}(U_\sigma)$, and so by choice of P_σ,

$$\operatorname{dist}_H\left(\operatorname{cl}(U_\sigma), \operatorname{cl}(U_\sigma) \cap C\right) \leq \operatorname{dist}_H\left(\operatorname{cl}(U_\sigma), P_\sigma\right) < \varepsilon/3.$$

Also, $\operatorname{dist}_H\left(\operatorname{cl}(U_\sigma) \cap C, \operatorname{cl}(V_\sigma)\right) < \varepsilon/3$ by definition of the sets V_i and by condition (3). Property (iii) follows since the Hausdorff distance is a metric on the space of compact subsets of $\mathbb{R}^d$. $\square$

With a second technical lemma, we will be ready to prove our main result. Below, $(\Delta, \Gamma) := \Delta \setminus \Gamma$ denotes the *relative complex* of Γ in Δ.

Lemma 6.1.6. *Let $\Gamma \subsetneq \Delta$ be acyclic simplicial complexes. Then (Δ, Γ) contains at least two faces.*

Proof. Since $\{\emptyset\}$ is not acyclic, (Δ, Γ) contains at least one nonempty face. Suppose for contradiction that (Δ, Γ) contains a unique face σ, and let $k = \dim \sigma \geq 0$. Observe that

$\langle\sigma\rangle \cap \Gamma = \partial\langle\sigma\rangle$, and that $\Delta = \Gamma \cup \langle\sigma\rangle$. We obtain a Mayer-Vietoris sequence

$$\cdots \to \tilde{H}_k(\Delta) \to \tilde{H}_{k-1}(\partial\langle\sigma\rangle) \to \tilde{H}_{k-1}(\Gamma) \oplus \tilde{H}_{k-1}(\langle\sigma\rangle) \to \cdots .$$

Since Δ, Γ, and $\langle\sigma\rangle$ are all acyclic this part of the sequence becomes $0 \to \tilde{H}_{k-1}(\partial\langle\sigma\rangle) \to 0$. This is a contradiction to the fact that the boundary of $\langle\sigma\rangle$ has nonzero $(k-1)$-homology. □

Throughout other chapters we have always used $\mathcal{X} = \{X_1, \ldots, X_n\}$ to denote a collection of closed sets. In the proof below we violate this convention. In particular, the proof requires us to keep track of four distinct collections of convex open sets, which we denote $\mathcal{U}, \mathcal{V}, \mathcal{W}$, and $\mathcal{X}$ respectively.

Theorem 6.1.7. *Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be a d-convex union representation of a simplicial complex $\Delta \subseteq 2^{[n]}$, and let $H \subseteq \mathbb{R}^d$ be a closed or open halfspace. Then Δ collapses to* nerve$(\{U_i \cap H \mid i \in [n]\})$.

Proof. Observe that it is enough to prove the result in the case that all U_i are bounded. It also suffices to consider the case that H is an open halfspace, since for a closed halfspace the nerve is unaffected by replacing the halfspace with its interior. Thus throughout the proof we assume that all U_i are bounded and that H is open.

We work by induction on the number of faces in Δ. There is nothing to prove if $\Delta = \emptyset$. If Δ is a single vertex, then every convex union representation consists of a single convex open set, and nerve$(\{U_1 \cap H\})$ is either Δ or the void complex, depending on whether $U_1 \cap H$ is empty. In either case Δ collapses to nerve$(\{U_1 \cap H\})$ and the result follows.

Otherwise Δ has more than two faces. Again if nerve$(\{U_i \cap H \mid i \in [n]\}) = \Delta$ we are done. If not, let A be the hyperplane defining the halfspace H, oriented so that $H = A^+$. Choose ε so that every nonempty $U_\sigma \cap H$ contains an ε-ball, and apply Lemma 6.1.5 to obtain a new convex union representation $\mathcal{V} = \{V_1, \ldots, V_n\}$ of Δ. Observe that by choice of ε and property (iii) of Lemma 6.1.5, nerve$(\{U_i \cap H \mid i \in [n]\}) =$ nerve$(\{V_i \cap H \mid i \in [n]\})$. Property (i) and the boundedness of the U_i imply that if $V_\sigma \cap H = \emptyset$, then V_σ and H have positive distance to one another. Thus we may perturb the position and angle of H slightly without changing the nerve $\operatorname{nerve}(\{V_i \cap H \mid i \in [n]\})$.

Perform a perturbation of H so that it is in general position relative to $\mathcal{V}$ in the following sense: no hyperplane parallel to A simultaneously supports two disjoint nonempty V_σ and V_τ. For all facets σ in the relative complex $\big(\Delta, \operatorname{nerve}(\{V_i \cap H \mid i \in [n]\})\big)$, let d_σ be the distance from V_σ to H. There is at least one such facet σ since we are assuming that $\operatorname{nerve}(\{V_i \cap H \mid i \in [n]\})$ is a proper subcomplex of Δ. The distances d_σ are finite since each V_σ is bounded, and they are distinct by genericity of A. Let V_{σ_0} be the region whose distance to H is largest. Let A_0 be the hyperplane separating V_{σ_0} from H and supporting V_{σ_0}, oriented so that H lies on its positive side. Finally, let $\Gamma = \operatorname{nerve}(\{V_i \cap A_0^> \mid i \in [n]\})$. Then Γ is a proper acyclic subcomplex of Δ containing $\operatorname{nerve}(\{V_i \cap H \mid i \in [n]\})$ and σ_0 is the unique facet of (Δ, Γ). Applying Lemma 6.1.6 we conclude that there is a minimal face $\tau_0 \in \Delta \setminus \Gamma$ with $\tau_0 \subsetneq \sigma_0$. Uniqueness of σ_0 implies that τ_0 is a free face of Δ.

We modify our representation one last time. By property (i) of Lemma 6.1.5, disjoint V_σ and V_τ have nonzero distance between them, and furthermore any V_σ with $V_\sigma \cap H = \emptyset$ has nonzero distance to H. Let $\delta > 0$ be smaller than one half the minimum of all these distances, and let B_δ be the open ball with radius δ centered at the origin. For $i \in [n]$, define

$$W_i = \begin{cases} V_i & i \in \tau_0 \\ (V_i + B_\delta) \cap \bigcup_{i=1}^n V_i & i \notin \tau_0. \end{cases}$$

By choice of δ the nerve of $\mathcal{W} = \{W_1, \ldots, W_n\}$ is equal to Δ. Moreover the union of the W_i is the same as the union of the V_i. Since $W_i = V_i$ for $i \in \tau_0$, it follows that $V_{\tau_0} = W_{\tau_0}$ and that W_{σ_0} is supported by A_0. Finally, by choice of δ, $\operatorname{nerve}(\{W_i \cap H \mid i \in [n]\}) = \operatorname{nerve}(\{V_i \cap H \mid i \in [n]\})$.

Now for $i \in [n]$ define $X_i = W_i \cap A_0^>$, and define $\mathcal{X} = \{X_1, \ldots, X_n\}$. Then $\operatorname{nerve}(\{X_i \cap H \mid i \in [n]\}) = \operatorname{nerve}(\{W_i \cap H \mid i \in [n]\})$ and $X_{\sigma_0} = W_{\sigma_0} \cap A_0^> = \emptyset$. By inductive hypothesis, the nerve $\operatorname{nerve}(\mathcal{X})$ collapses to $\operatorname{nerve}(\{X_i \cap H \mid i \in [n]\})$.

We claim that $\operatorname{nerve}(\mathcal{X})$ is equal to $\Delta \setminus \tau_0$. It suffices to show that $X_\gamma = W_\gamma \cap A_0^>$ is

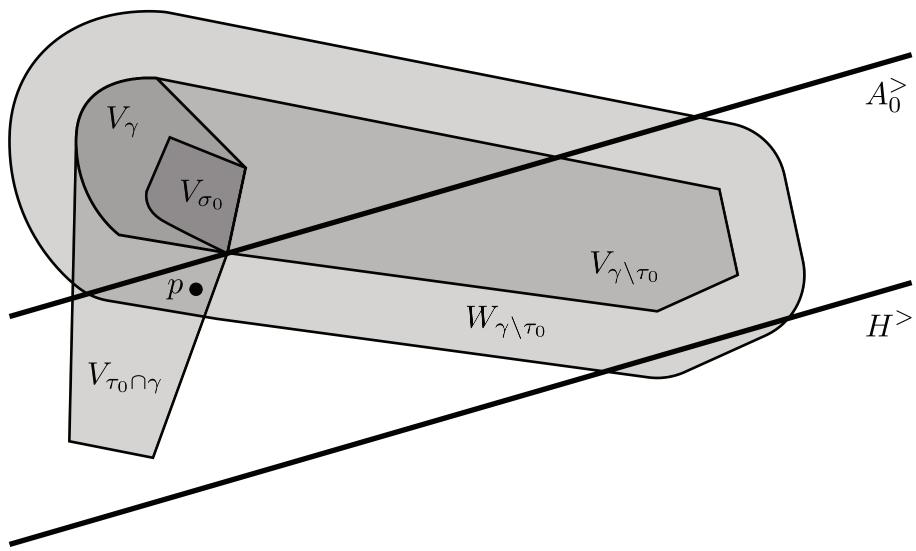


Figure 6.3: Objects used to prove Theorem 6.1.7. The point p lies in $X_\gamma = W_\gamma \cap A_0^>$.

nonempty for every $\gamma \in (\Delta, \Gamma)$ with $\tau_0 \not\subseteq \gamma$. Note that for such a γ, $\tau_0 \cap \gamma \subsetneq \tau_0$, and so by minimality of τ_0, $W_{\tau_0\cap\gamma} \cap A_0^> = V_{\tau_0\cap\gamma} \cap A_0^> \neq \emptyset$. Since $V_{\sigma_0} \subseteq W_{\sigma_0} \subseteq W_{\tau_0\cap\gamma}$ and V_{σ_0} is supported by A_0, we may choose a point $p \in W_{\tau_0\cap\gamma} \cap A_0^>$ which is arbitrarily close to V_{σ_0}. By construction of W_i the region $W_{\gamma\setminus\tau_0}$ contains the Minkowski sum $V_{\sigma_0} + B_\delta$. But then it must contain p, so $W_\gamma = W_{\gamma\cap\tau_0} \cap W_{\gamma\setminus\tau_0}$ contains p. In particular, W_γ has nonempty intersection with $A_0^>$. This situation is illustrated in Figure 6.3.

Since $W_\gamma \cap A_0^> \neq \emptyset$ for all $\gamma \in \Delta \setminus \Gamma$ with $\tau_0 \not\subseteq \gamma$, we conclude that $\text{nerve}(\mathcal{X}) = \Delta \setminus \tau_0$. Furthermore, since $\Delta \to \Delta \setminus \tau_0$ is a collapse, we conclude that Δ collapses to $\text{nerve}(\{X_i \cap H \mid i \in [n]\}) = \text{nerve}(\{U_i \cap H \mid i \in [n]\})$, proving the result. □

This result has several immediate geometric and combinatorial corollaries, which we highlight below.

Corollary 6.1.8. *Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be a d-convex union representation of a simplicial complex Δ, and let $C \subseteq \mathbb{R}^d$ be a convex set. Then Δ collapses onto* $\text{nerve}(\{U_i \cap C \mid i \in [n]\})$.

Proof. For all U_σ such that $U_\sigma \cap C \neq \emptyset$, choose a point $p_\sigma \in U_\sigma \cap C$, and let C' be the convex hull of these points. Observe that C' is a polytope contained in C such that $\text{nerve}(\{U_i \cap C \mid i \in [n]\}) = \text{nerve}(\{U_i \cap C' \mid i \in [n]\})$. Since C' is the intersection of finitely many closed halfspaces, we can repeatedly apply Theorem 6.1.7 to obtain that Δ collapses to $\text{nerve}(\{U_i \cap C' \mid i \in [n]\})$, proving the result. □

Corollary 6.1.9. *Let Δ be a convex union representable complex, and let $\sigma \in \Delta$ be an arbitrary face. Then Δ collapses onto the star of σ. In particular, if $\sigma \neq \emptyset$, then Δ collapses onto $\langle \sigma \rangle$.*

Proof. Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be a d-convex union representation of Δ, and let $C = U_\sigma$. Then C is a nonempty convex subset of $\mathbb{R}^d$. Therefore, by Corollary 6.1.8, Δ collapses onto $\text{nerve}\big(\{U_i \cap C \mid i \in [n]\}\big)$. The result follows since $\text{nerve}(\{U_i \cap C \mid i \in [n]\}) = \text{St}_\Delta(\sigma)$. Indeed, if $\tau \subseteq [n]$, then $\bigcap_{j \in \tau}(U_j \cap U_\sigma) = U_{\tau \cup \sigma}$. Thus, $\tau \in \text{nerve}(\{U_i \cap C \mid i \in [n]\})$ if and only if $\tau \cup \sigma \in \text{nerve}(\mathcal{U}) = \Delta$, which happens if and only if $\tau \in \text{St}_\Delta(\sigma)$. □

Corollary 6.1.10. *Let Δ be a convex union representable complex. Then the free faces of Δ cannot all share a common vertex.*

Proof. Suppose the free faces of Δ share a common vertex v. Then no collapse of Δ other than Δ itself would contain $\text{St}_\Delta(v)$. Since Δ collapses to $\text{St}_\Delta(v)$ by Corollary 6.1.9, this implies $\Delta = \text{St}_\Delta(v)$. Thus Δ is a cone over v. But then any facet of $\Delta \setminus v$ is a free face of Δ. Such a free face does not contain v, a contradiction. □

We will study the consequences of these corollaries in greater depth throughout subsequent sections, and we will also establish further results on convex union representable complexes. In particular, Corollary 6.1.10 provides the basis for our result that not every collapsible complex is convex union representable, answering the previously discussed question posed by [CFS19].

6.2 Collapsible Complexes That Are Not Convex Union Representable

In [ABL17, Theorem 2.3] the authors construct examples for all $d \geq 2$ of a d-dimensional collapsible simplicial complex Σ_d with only one free face. According to Corollary 6.1.10, these provide an example of collapsible complexes that are not convex union representable. The authors also give examples for all $d \geq 2$ of a d-dimensional simplicial complex E_d which is pure and non-evasive, has only two free faces, and, furthermore, these two free faces share a common ridge (see [ABL17, Theorem 2.5]). By Corollary 6.1.10 these complexes are not convex union representable either. We formalize these observations in the following corollary.

Corollary 6.2.1. *The simplicial complexes Σ_d of [ABL17] are pure, collapsible, and shellable, but not convex union representable. Similarly, the simplicial complexes E_d of [ABL17] are pure and non-evasive, but not convex union representable.*

Example 6.2.2. The complex Σ_2 has only 7 vertices. Its facets are

$$125, 134, 136, 137, 145, 167, 234, 236, 237, 247, 256, 456, 467,$$

and 12 is its unique free face. Figure 6.4 shows Σ_2 (up to indentification on the appropriate exterior edges). See also [ABL17, Figure 2].

The complexes Σ_d and E_d allow us to generalize the infinite family of minimally non-convex codes from Theorem 4.4.3. We will do this in Corollary 6.8.3. First we study the properties of convex union representable complexes in more detail.

6.3 Equivalence of Open and Closed Convex Union Representability

In the other chapters of this work we have seen surprising differences between collections of open convex sets and collections of closed convex sets—for example Corollary 5.3.7 exhibited a family of codes where the open and closed embedding dimensions diverge exponentially from one another, and Theorem 5.5.2 showed that monotonicity of convexity holds for open realizations of codes but not closed realizations. Below, we show that we need not worry

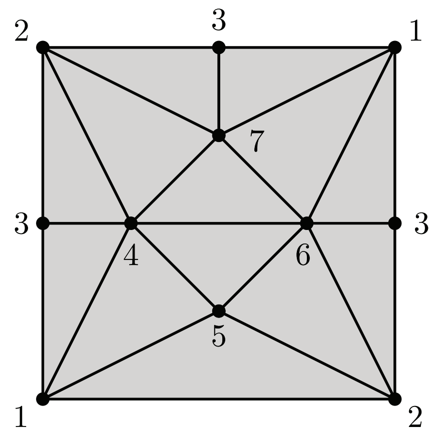


Figure 6.4: The collapsible complex Σ_2 from [ABL17], which has the unique free face 12, and is not convex union representable.

about such distinctions when studying convex union representable complexes: if Δ is d-convex union representable, then Δ also has a convex union representation in $\mathbb{R}^d$ consisting of closed convex sets. In fact, we may obtain such representations constructively from one another.

Proposition 6.3.1. *For a simplicial complex $\Delta \subseteq 2^{[n]}$, the following are equivalent:*

(i) Δ is d-convex union representable,

(ii) There exists a d-convex union representation $\mathcal{V} = \{V_1, \ldots, V_n\}$ of Δ such that (1) the collection of closures of the V_i has nerve Δ, (2) each $\mathrm{cl}(V_i)$ is a polytope, and (3) the union of all V_i is the interior of a polytope,

(iii) Δ is the nerve of a collection $\mathcal{X} = \{X_1, \ldots, X_n\}$ of d-dimensional polytopes whose union is a polytope in $\mathbb{R}^d$, and

(iv) Δ is the nerve of a collection $\mathcal{X} = \{X_1, \ldots, X_n\}$ of closed convex sets whose union is a closed convex set in $\mathbb{R}^d$.

Proof. We first show that (i) implies (ii). Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be a convex union representation of Δ. By intersecting with an open ball of a sufficiently large radius, we can assume that $\mathcal{U}$ is bounded. Choose a representation $\mathcal{V}$ as guaranteed by Lemma 6.1.5. Properties (2) and (3) of (ii) follow from the statement of Lemma 6.1.5, so we just need to check that the nerve of $\mathcal{X} = \{X_i := \mathrm{cl}(V_i) \mid i \in [n]\}$ is Δ. This is immediate from the fact that $V_i \subseteq \mathrm{cl}(V_i) \subseteq U_i$ for all $i \in n$.

The implications from *(ii)* to *(iii)* and *(iii)* to *(iv)* are straightforward: the former by taking closures of the V_i, and the latter since polytopes are closed and convex.

To prove that (iv) implies (i), assume that all X_i are compact by intersecting with a closed ball of sufficiently large radius. Then if X_σ and X_τ are disjoint, they are a positive distance apart, and so we can take the Minkowski sum of all X_i with an open ball B_ε of sufficiently small radius while preserving the nerve. The resulting collection $\mathcal{W} = \{W_i := X_i + B_\varepsilon \mid i \in [n]\}$ is a convex union representation of Δ in $\mathbb{R}^d$: the sets $X_i + B_\varepsilon$ are open convex sets, and so is their union $\bigcup_{i=1}^n \left(X_i + B_\varepsilon\right) = \left(\bigcup_{i=1}^n X_i\right) + B_\varepsilon$. □

6.4 Constructible-Like Behavior

As we saw in Corollary 6.2.1, not all collapsible complexes are convex union representable. Thus one of our goals is to establish additional necessary conditions for convex union representability. The following theorem provides a step in this direction, and shows that convex union representable complexes are similar in spirit to constructible complexes—a notion introduced in [Zee63].

Theorem 6.4.1. *Let Δ be a d-convex union representable simplicial complex, and let $\tau_1, \tau_2 \in \Delta$ be such that $\tau_1 \cup \tau_2 \notin \Delta$. Then there exist simplicial complexes $\Delta_1 \subseteq \Delta \setminus \tau_1$ and $\Delta_2 \subseteq \Delta \setminus \tau_2$ satisfying*

(i) $\Delta = \Delta_1 \cup \Delta_2$,

(ii) Δ *collapses to* Δ_i *(for* $i = 1, 2$*),*

(iii) Δ_i *collapses to* $\Delta_1 \cap \Delta_2$ *(for* $i = 1, 2$*),*

(iv) Δ_1 *and* Δ_2 *are d-convex union representable, and*

(v) $\Delta_1 \cap \Delta_2$ *is* $(d-1)$*-convex union representable.*

Proof. Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be a convex union representation of Δ. The condition that $\tau_1 \cup \tau_2 \notin \Delta$ implies that U_{τ_1} and U_{τ_2} are disjoint. Thus we may choose a hyperplane H separating U_{τ_1} and U_{τ_2}, oriented such that U_{τ_2} lies on the open positive side $H^>$ of H. Apply Lemma 6.1.5 to obtain a new representation $\mathcal{V} = \{V_1, \ldots, V_n\}$ of Δ. This representation has the property that if $\sigma \in \Delta$ and $V_\sigma \cap H = \emptyset$, then there is a positive distance between V_σ and H. In particular, there is a small ε so that the Minkowski sum of H with an ε-ball induces the same nerve as H when intersected with the various V_i.

Now, let Δ_1 be the nerve of $\{V_i \cap H^> \mid i \in [n]\}$ and let Δ_2 be the nerve of $\{V_i \cap H^< \mid i \in [n]\}$. We claim that Δ_1 and Δ_2 satisfy the conditions stated above.

First let us argue that $\Delta_1 \subseteq \Delta \setminus \tau_1$ and $\Delta_2 \subseteq \Delta \setminus \tau_2$. Note that $\Delta_1 \subseteq \Delta$ since the sets representing Δ_1 are subsets of the sets representing Δ. Moreover $\tau_1 \notin \Delta_1$ since $V_{\tau_1} \cap H^> = \emptyset$, and thus $\Delta_1 \subseteq \Delta \setminus \tau_1$. A symmetric argument shows that $\Delta_2 \subseteq \Delta \setminus \tau_2$.

For (i), let $\sigma \in \Delta$. Then $V_\sigma \neq \emptyset$, and since V_σ is open it has nonempty intersection with $H^>$ or with $H^<$. In the former case $\sigma \in \Delta_1$ and in the latter $\sigma \in \Delta_2$. Thus $\Delta = \Delta_1 \cup \Delta_2$. For (ii) we can apply Theorem 6.1.7 with the open halfspaces $H^>$ and $H^<$.

To prove (iii), we first claim that $\Delta_1 \cap \Delta_2 = \text{nerve}(\{V_i \cap H \mid i \in [n]\})$. If $\sigma \in \Delta_1 \cap \Delta_2$, then V_σ contains points on both sides of H, and by convexity it contains points in H. Thus $\sigma \in \text{nerve}(\{V_i \cap H \mid i \in [n]\})$. Conversely, if $\sigma \in \text{nerve}(\{V_i \cap H \mid i \in [n]\})$, then V_σ contains points in H, and by openness it contains points in both $H^>$ and $H^<$.

To see that Δ_i collapses to $\text{nerve}(\{V_i \cap H \mid i \in [n]\})$, let C be the Minkowski sum of H with a small ε-ball, so that $\text{nerve}(\{V_i \cap H \mid i \in [n]\}) = \text{nerve}(\{V_i \cap C \mid i \in [n]\})$. Then observe that $C \cap H^>$ induces the nerve $\Delta_1 \cap \Delta_2$ when intersected with the convex union representation $\{V_i \cap H^> \mid i \in [n]\}$ of Δ_1. By Corollary 6.1.8 this implies that Δ_1 collapses to $\Delta_1 \cap \Delta_2$. A symmetric argument shows that Δ_2 collapses to $\Delta_1 \cap \Delta_2$.

For (iv) simply observe that $\{V_i \cap H^> \mid i \in [n]\}$ and $\{V_i \cap H^< \mid i \in [n]\}$ are d-convex union representations for Δ_1 and Δ_2 respectively. To prove (v) recall that $\Delta_1 \cap \Delta_2 = \text{nerve}(\{V_i \cap H \mid i \in [n]\})$. Since $H \cong \mathbb{R}^{d-1}$, this yields a $(d-1)$-convex union representation of $\Delta_1 \cap \Delta_2$. □

If Δ is a simplicial complex, we let $\Sigma\Delta$ denote the *suspension* of Δ, which is the complex obtained by joining Δ with a complex consisting of two disjoint vertices. The following result uses Theorem 6.4.1 to show that suspending a complex strictly increases the smallest dimension in which it is convex union representable.

Corollary 6.4.2. *Let Δ be a simplicial complex that is not d-convex union representable. Then the suspension $\Sigma\Delta$ of Δ is not $(d+1)$-convex union representable. In particular, if Δ is not convex union representable, then neither is $\Sigma\Delta$.*

Proof. Assume $\Sigma\Delta$ is $(d+1)$-convex union representable, and let u and v be the suspension vertices. Observe that $\{u, v\}$ is not a face of $\Sigma\Delta$, so by Theorem 6.4.1 there exist complexes $\Delta_1 \subseteq \Sigma\Delta \setminus u = v * \Delta$ and $\Delta_2 \subseteq \Sigma\Delta \setminus v = u * \Delta$ satisfying (i)-(v) in the theorem statement. But since $\Delta_1 \cup \Delta_2 = \Sigma\Delta = (v * \Delta) \cup (u * \Delta)$, it must be the case that $\Delta_1 = v * \Delta$ and $\Delta_2 = u * \Delta$. Then $\Delta_1 \cap \Delta_2 = \Delta$, and by (v) we conclude that Δ is d-convex union representable, a contradiction. □

Note that Corollary 6.4.2 together with Corollary 6.2.1 provides us with additional examples of collapsible complexes that are not convex union representable. In some situations, such as Corolary 6.7.4 below, it also allows us to establish lower bounds on the minimum dimension of a convex union representation.

6.5 Alexander Duality

Recall that if $\Delta \subseteq 2^{[n]}$ is a simplicial complex, then the *Alexander dual* of Δ is

$$\Delta^* := \{\sigma \subseteq [n] \mid [n] \setminus \sigma \notin \Delta\}.$$

In other words, the Alexander dual is the set of complements of non-faces of Δ. The goal of this section is to show that if Δ is convex union representable and $n \geq 1$, then the Alexander dual of Δ is collapsible.

For our result we require the following standard lemma, which is proven in [KSS84].

Lemma 6.5.1. *Let $\Delta \subseteq \Gamma$ be simplicial complexes. Then Γ collapses onto Δ if and only if Δ^* collapses onto Γ^*.*

Corollary 6.5.2. *Let Δ be a simplicial complex with vertex set $[n]$. Then Δ^* is collapsible if and only if $2^{[n]}$ collapses onto Δ.*

Proof. Take $\Gamma = 2^{[n]}$ and use Lemma 6.5.1, noting that $\Gamma^* = \emptyset$. □

With Corollary 6.5.2 in hand, we are ready to prove the main result of this section.

Theorem 6.5.3. *Let $n \geq 1$ and let $\Delta \subseteq 2^{[n]}$ be a convex union representable complex. Then Δ^* is collapsible.*

Proof. Let $\mathcal{U} = \{U_1, \dots, U_n\}$ be a convex union representation of Δ in $\mathbb{R}^d$ such that the collection of closures of the U_i has nerve equal to Δ, as guaranteed by Proposition 6.3.1. Embed $\mathcal{U}$ into $\mathbb{R}^{d+1}$ by identifying $\mathbb{R}^d$ with the hyperplane defined by $x_{d+1} = 0$. Let $U = \bigcup_{i \in [n]} U_i$, and let V be the shifted copy of U contained in the hyperplane defined by $x_{d+1} = 1$. Then for $i \in [n]$ define $V_i = \operatorname{int}(\operatorname{conv}(U_i \cup V))$. This construction is illustrated for $d = 2$ and $n = 4$ in Figure 6.5.

Observe that the nerve of $\mathcal{V} = \{V_1, \dots, V_n\}$ is $2^{[n]}$. Since the U_i were chosen such that their closures have the same nerve, it follows that for a sufficiently small $\varepsilon > 0$, the halfspace H_ε defined by $x_{d+1} < \varepsilon$ has the property that

$$\operatorname{nerve}(\mathcal{U}) = \operatorname{nerve}(\{V_i \cap H_\varepsilon \mid i \in [n]\}).$$

But by Theorem 6.1.7, this implies that $2^{[n]}$ collapses to $\operatorname{nerve}(\mathcal{U}) = \Delta$. Corollary 6.5.2 then yields that Δ^* is collapsible. □

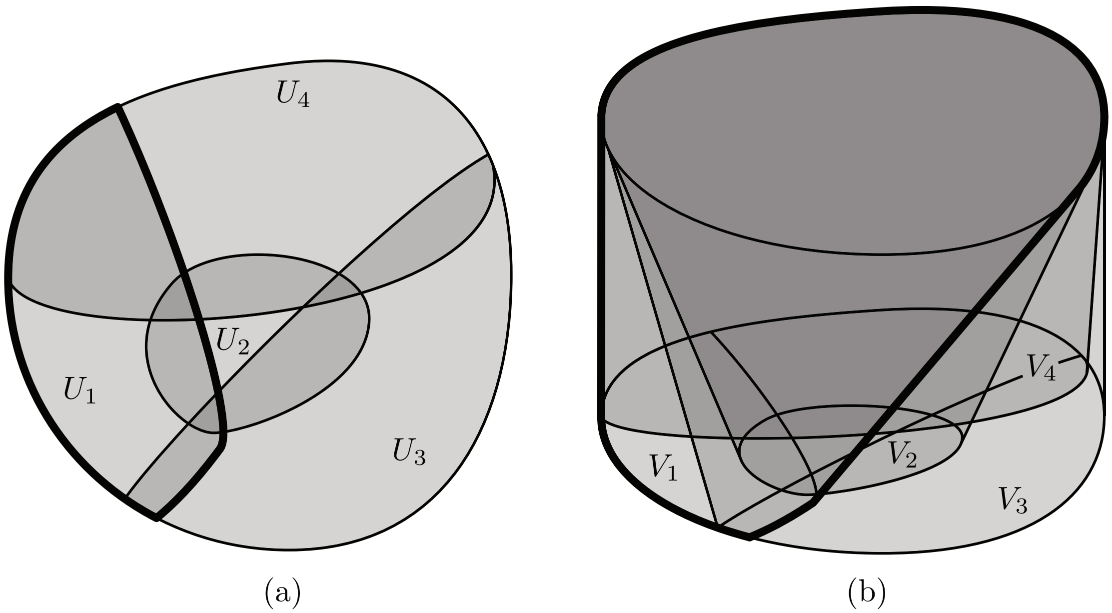


Figure 6.5: (a) A 2-convex union representation $\mathcal{U} = \{U_1, U_2, U_3, U_4\}$ of $\langle 123, 124, 234 \rangle$, with U_1 outlined in bold. (b) The resulting 3-convex union representation $\mathcal{V} = \{V_1, V_2, V_3, V_4\}$ of $2^{[4]}$, with V_1 outlined in bold.

There exist collapsible complexes whose Alexander dual is not collapsible (see, for instance [Wel99, Example 3.3])—such complexes thus provide additional examples of collapsible complexes that are not convex union representable. On the other hand, a complex is non-evasive if and only if its Alexander dual is non-evasive (see [KSS84] and [Wel99, Lemma 2.5]). This suggests that convex union representability may imply non-evasiveness.

6.6 Convex Union Representable Complexes With a Few Free Faces

Below we bound the minimum dimension of a representation of a convex union representable complex Δ by the number of free faces of Δ. More specifically, we establish the following result.

Theorem 6.6.1. *If Δ is a convex union representable complex with k facets that contain all free faces of Δ, then Δ is $(k-1)$-convex union representable.*

Proof. Let $\mathcal{U} = \{U_1, \dots U_n\}$ be a convex union representation of Δ, and let $\sigma_1, \dots, \sigma_k$ be the facets of Δ containing all free faces. Choose points $p_i \in U_{\sigma_i}$ for all $i \in [k]$, and let $C = \text{conv}(\{p_1, \dots, p_k\})$. By Corollary 6.1.8, Δ collapses to $\text{nerve}(\{U_i \cap C \mid i \in [n]\})$, but by choice of the p_i this nerve contains all free faces of Δ. Thus $\Delta = \text{nerve}(\{U_i \cap C \mid i \in [n]\})$. Since C is the convex hull of k points, it is contained in an affine subspace of dimension no larger than $k-1$. Taking relative interiors in this affine subspace yields a convex union representation of Δ in dimension no larger than $k-1$, proving the result. □

Theorem 6.6.1 has two immediate consequences, highlighted below. In Corollary 6.6.3, recall that the *stellar subdivision* of Δ at a facet σ is a (homeomorphic) complex with an additional vertex v, in which we have deleted σ and added faces of the form $\tau \cup \{v\}$ for every τ that is a proper face of σ.

Corollary 6.6.2. *If Δ is a convex union representable complex with k facets that contain all free faces of Δ, then Δ is $(k-1)$-representable. In particular it is $(k-1)$-Leray.*

Corollary 6.6.3. *Let Δ be a d-dimensional collapsible complex. Suppose that Δ has only d or fewer free faces. Let Δ' be a stellar subdivision of Δ at one of its d-dimensional faces. Then Δ' is collapsible but not convex union representable.*

Proof. Note that a stellar subdivision at a facet does not affect collapsibility nor the collection of free faces. Let $\sigma \in \Delta$ be the d-face at which the subdivision occurs. Then Δ' will have the boundary of $\langle \sigma \rangle$ as an induced subcomplex. This boundary has nonvanishing $(d-1)$-homology, and so Δ' is not $(d-1)$-Leray. On the other hand, Δ' has only d or fewer free faces. Thus by Corollary 6.6.2, Δ' is not convex union representable. □

The following corollary of Theorem 6.6.1 characterizes convex union representable complexes that have at most two free faces. In particular, it provides a different proof of Corollary 6.2.1.

Corollary 6.6.4. *Let Δ be a convex union representable complex. Then Δ has at most two free faces if and only if Δ is a path.*

Proof. One direction is clear: paths are 1-convex union representable complexes that have at most two free faces. For the other direction, assume that Δ is a convex union representable complex with at most two free faces. Then by Corollary 6.6.2, Δ is 1-representable. Thus to prove the result, it suffices to show that if Δ is a collapsible, 1-representable complex with at most two free faces, then Δ is a path.

The class of 1-representable complexes, also known as clique complexes of interval graphs, is well understood. In particular, [FG65, Theorem 7.1] asserts that Γ is 1-representable if and only if Γ satisfies the following property (*): the facets of Γ can be numbered $F_1, \ldots, F_m$ in such a way that for all pairs (i, k) with $1 \leq i < k \leq m$ and for any vertex $v \in \Gamma$, if $v \in F_i \cap F_k$, then v belongs to *all* F_j with $i < j < k$.

We use induction on m to show that any collapsible Γ that satisfies (*) and has at most two free faces must be a path. In the base case of $m = 1$, Γ is a simplex $\langle F_1 \rangle$, in which case the result is immediate: indeed, 0- and 1-dimensional simplices form a path, while a simplex of dimension $d \geq 2$ has more than two free faces.

For the case of $m > 1$, let v be a vertex that belongs to F_m, but not to F_{m-1}, noting that v exists since F_m is a facet, and hence it is not a subset of F_{m-1}. Similarly, let $w \in F_1 \setminus F_2$. Since Γ satisfies (*), F_m is the only facet that contains v. Consequently, $v \neq w$, and any proper subset of F_m that contains v is a free face of Γ. By a symmetric argument, w is a free face of Γ. Thus, for Γ to have at most two free faces, F_m must be 1-dimensional. We write $F_m = \{v, u\}$ and consider $\Gamma' = \Gamma \setminus v = \bigcup_{i=1}^{m-1} \langle F_i \rangle$. Then Γ' is collapsible and satisfies property (*). Can it happen that Γ' has more than two free faces? If yes, then at least two of these free faces are not $\{u\}$; hence they are also free faces of Γ, which together with the face $\{v\}$ already accounts for three free faces of Γ, contradicting our assumption. Thus Γ' has at most two free faces, and hence Γ' is a path by inductive hypothesis. We conclude that $\Gamma = \Gamma' \cup \langle \{u, v\} \rangle$ is a 1-dimensional collapsible complex, so a tree. The result follows since every tree that is not a path has at least three leaves, and each leaf is a free face. $\square$

6.7 Constructions

So far we have established a variety of necessary conditions for a complex to be convex union representable. In this section we discuss some sufficient conditions, beginning with cones and joins.

Proposition 6.7.1. *Every cone is convex union representable.*

Proof. Consider a cone $\Delta * (n+1)$ where $\Delta \subseteq 2^{[n]}$. Recall that we may choose an open d-representation $\mathcal{U} = \{U_1, \ldots, U_n\}$ of Δ. Adding $U_{n+1} = \mathbb{R}^d$ to this d-representation yields a d-convex union representation of $\Delta * (n+1)$. $\square$

Remark 6.7.2. Proposition 6.7.1 and the fact that all convex union representable complexes are collapsible and hence $\mathbb{Q}$-acyclic leads to the following set of (strict) implications:

$$\text{cone} \implies \text{convex union representable} \implies \mathbb{Q}\text{-acyclic.}$$

Since the set of f-vectors of the class of simplicial complexes that are cones coincides with set of f-vectors of the class of $\mathbb{Q}$-acyclic simplicial complexes [Kal85], it follows that it also coincides with the set of f-vectors of the class of convex-union representable complexes.

Proposition 6.7.3. *Let Δ be a d_1-convex union representable complex, and let Γ be a d_2-representable simplicial complex. Then $\Delta * \Gamma$ is $(d_1 + d_2)$-convex union representable.*

Proof. Let σ and τ be the (without loss of generality, disjoint) vertex sets of Δ and Γ respectively. Let $\mathcal{U} = \{U_i \mid i \in \sigma\}$ be a d_1-convex union representation of Δ, and let $U = \bigcup_{i\in\sigma} U_i$. Let $\mathcal{V} = \{V_j \mid j \in \tau\}$ be a d_2-representation of Γ consisting of convex open sets. Define $V = \operatorname{conv}\left(\bigcup_{j\in\tau} V_j\right)$. Then for $k \in \sigma \cup \tau$ define

$$W_k = \begin{cases} U_k \times V & k \in \sigma, \\ U \times V_k & k \in \tau. \end{cases}$$

The union of all W_k is the convex open set $U \times V$ since U is the union of all U_i. Moreover the nonempty regions of intersection among the W_k are of the form $U_{\sigma'} \times V_{\tau'}$ for $\sigma' \in \Delta$ and $\tau' \in \Gamma$, and so $\text{nerve}(\{W_k \mid k \in \sigma \cup \tau\}) = \Delta * \Gamma$. Thus $\Delta * \Gamma$ is $(d_1 + d_2)$-convex union representable as desired. □

In some instances, Proposition 6.7.3 along with Corollary 6.4.2 allows us to compute the minimum dimension of a convex union representation of a given complex. Below is one such example.

Corollary 6.7.4. *Let Δ be a point, and let $\Sigma^k\Delta$ denote the k-fold suspension of Δ. Then for all $k \geq 0$, the complex $\Sigma^k\Delta$ is k-convex union representable but not $(k-1)$-convex union representable.*

Proof. We work by induction on k. If $k = 0$ then $\Sigma^k\Delta = \Delta$ and the result holds. Suppose $k \geq 1$ and let Γ be the complex consisting of two points. Observe that Γ is 1-representable. Further observe that $\Sigma^k\Delta = (\Sigma^{k-1}\Delta) * \Gamma$, and so Proposition 6.7.3 implies that $\Sigma^k\Delta$ is k-convex union representable. Since $\Sigma^{k-1}\Delta$ is not $(k-2)$ convex union representable, Corollary 6.4.2 implies that $\Sigma^k\Delta$ is not $(k-1)$-convex union representable. The result follows. □

The following proposition shows that building cones over certain subcomplexes of convex union representable complexes preserves convex union representability.

Proposition 6.7.5. *Let $\Delta \subseteq 2^{[n]}$ be a d-convex union representable simplicial complex. Choose a face $\sigma \in \Delta$ and a set ω with $\sigma \subseteq \omega \subseteq [n]$. Then the complex*

$$\Delta' := \Delta \cup \big((n+1) * \text{St}_\Delta(\sigma)|_\omega\big)$$

is $(d+1)$-convex union representable.

Proof. Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be a bounded convex union representation of Δ in $\mathbb{R}^d$, and let $U = \bigcup_{i\in[n]} U_i$. Choose an open convex set $V \subseteq U_\sigma$ such that $\overline{V} \subseteq U_\sigma$, and $V \cap U_\tau \neq \emptyset$ for all $\tau \supset \sigma$. Identify $\mathbb{R}^d$ with the hyperplane defined by $x_{d+1} = 0$ in $\mathbb{R}^{d+1}$, and let $\widetilde{V}$ be the shifted copy of V contained in the hyperplane defined by $x_{d+1} = 1$. Define $W = \operatorname{int}\operatorname{conv}\left(U \cup \widetilde{V}\right)$, and for $i \in [n]$ define

$$W_i = (U_i \times (0,1)) \cap W.$$

Observe that $\mathcal{W} = \{W_1, \dots, W_n\}$ is a $(d+1)$-convex union representation of Δ and that the union of all W_i is equal to W.

For $\varepsilon > 0$, consider the hyperplane H_ε defined by $x_{d+1} = 1 - \varepsilon$, oriented so that $\widetilde{V}$ lies on the positive side. We claim that for small enough ε, $H_\varepsilon^> \cap W \subseteq W_\sigma$. Suppose not, so that there exists some $W_\tau \not\subseteq W_\sigma$ with $W_\tau \setminus W_\sigma$ containing points arbitrarily close to the hyperplane defined by $x_{d+1} = 1$. Since the only such points in W are those approaching $V \times (0,1)$, we see that W_τ must contain points arbitrarily close to $V \times (0,1)$ but not in W_σ. But then the projection of these points onto $\mathbb{R}^d$ yields a series of points in U approaching V, but not contained in U_σ. This contradicts the condition that $\overline{V} \subseteq U_\sigma$. We conclude that for some $\varepsilon > 0$, $H_\varepsilon^> \cap W \subseteq W_\sigma$.

Now, for $i \notin \omega$, replace W_i with $W_i \cap H_\varepsilon^<$. By choice of ε this does not affect the nerve or union of $\mathcal{W}$. Then define $W_{n+1} = W \cap H_\varepsilon^>$. We claim that $\mathcal{W}' = \{W_1, \dots, W_n, W_{n+1}\}$ is a $(d+1)$-convex union representation of $\Delta \cup \left((n+1) * \operatorname{St}_\Delta(\sigma)|_\omega\right)$. To see this, observe that the regions of maximal intersection in $W \cap H_\varepsilon^>$ are the inclusion-maximal faces γ of Δ satisfying $\sigma \subseteq \gamma \subseteq \omega$. These are exactly the facets of $\operatorname{St}_\Delta(\sigma)|_\omega$. Thus the new faces introduced by W_{n+1} are of the form $\{n+1\} \cup \gamma$ for $\gamma \in \operatorname{St}_\Delta(\sigma)|_\omega$, and the result follows. □

One immediate corollary of Proposition 6.7.5 is that trees are convex union representable.

Corollary 6.7.6. *A 1-dimensional complex Δ is convex union representable if and only if it is collapsible (in particular, if and only if it is a tree).*

Proof. If $\Delta \subseteq 2^{[n]}$ is a tree, we can build a convex union representation inductively using Proposition 6.7.5. In the base case that Δ is a single vertex, a realization is given in $\mathbb{R}^0$ by $U_1 = \{0\}$. For Δ containing at least one edge, label the vertices so that $n+1$ is a leaf, and let i be the vertex that $n+1$ is adjacent to. Then $\Delta|_{[n]}$ is convex union representable by inductive hypothesis. Choosing $\sigma = \omega = \{i\}$ and applying Proposition 6.7.5 we obtain that the complex

$$\Delta|_{[n]} \cup \langle\{i, n+1\}\rangle$$

is convex union representable. But this is just Δ, so the result follows. □

Remark 6.7.7. In fact, it is not hard to show by induction that all trees are 2-convex union representable. This expands a result of [JOS$^+$15], which showed that trees (and in fact all planar graphs) are 2-representable.

Remark 6.7.8. Essentially the same argument as in Corollary 6.7.6 shows that if Δ is *strong collapsible* and flag, then Δ is convex union representable. Strong collapsible complexes were introduced in [BM12, Section 2]. The condition of being a flag complex is only needed to guarantee that all vertex links are induced subcomplexes. In particular, since barycentric subdivisions are always flag and since the barycentric subdivision of any strong collapsible complex is strong collapsible [BM12, Theorem 4.15], it follows that barycentric subdivisions of strong collapsible complexes are convex union representable. For example, the barycentric subdivision of an arbitrary cone complex is convex union representable.

A simplicial complex Δ is a *simplicial ball* (or a *simplicial sphere*) if the geometric realization of Δ is homeomorphic to a ball (or a sphere, respectively). Another immediate application of Proposition 6.7.5 is that all stacked balls are convex union representable. (Stacked balls are defined recursively: a d-dimensional simplex is a stacked ball with $d+1$ vertices; a d-dimensional stacked ball Δ on $n+1 \geq d+2$ vertices is obtained from a d-dimensional stacked ball Γ on n vertices by choosing a free ridge σ of Γ and building a cone on it: $\Delta = \Gamma \cup ((n+1) * \overline{\sigma})$.) We close this section by showing that all simplicial balls in a certain larger class are convex union representable. Recall that if P is a simplicial polytope of dimension d and v is a vertex of P, then ∂P and $\partial P \setminus v$ are simplicial complexes of dimension $d-1$: the former is a simplicial sphere while the latter is a simplicial ball.

Proposition 6.7.9. *Let P be a d-dimensional simplicial polytope, and let v be a vertex of P. Then $\partial P \setminus v$ is a $(d-1)$-convex union representable simplicial complex.*

Proof. Label the vertices of P as $\{v_1, \ldots, v_n\}$ so that $v = v_n$, and translate P if necessary so that the origin is in the interior of P. Let P^* be the polar polytope of P, and let $F_n = \hat{v}_n$ be the facet of P^* corresponding to the vertex v. Consider the Schlegel diagram $\mathcal{S}(F_n)$ of P^* based at F_n. This is a polytopal complex; in particular the facets of $\mathcal{S}(F_n)$ are polytopes.

Furthermore, $\mathcal{S}(F_n)$ satisfies the following properties (see Chapters 2, 5, and 8 of [Zie95] for basics on polar polytopes, Schlegel diagrams, and polytopal complexes): (i) the set of facets of $\mathcal{S}(F_n)$ is in bijection with the set of facets of P^* other than F, which in turn is in bijection with the vertex set $\{v_1, \ldots, v_{n-1}\}$ of $\partial P \setminus v$, that is, we can index the facets of $\mathcal{S}(F_n)$ as $\{G_1, \ldots, G_{n-1}\}$; (ii) for any $\sigma \subseteq [n-1]$, the facets $\{G_i \mid i \in \sigma\}$ have a nonempty intersection if and only if σ is a face of $\partial P \setminus v$; (iii) the union of all facets $\{G_i \mid i \in [n-1]\}$ of $\mathcal{S}(F_n)$ is F_n. These properties imply that $\{G_i \mid i \in [n-1]\}$ is a collection of closed convex subsets of $\mathrm{Aff}(F_n) \cong \mathbb{R}^{d-1}$ whose nerve is $\partial P \setminus v$, and whose union is convex. This along with Proposition 6.3.1 yields the result. □

Example 6.7.10. Let $P \subseteq \mathbb{R}^3$ be the regular octahedron. Up to isomorphism, deleting any vertex from ∂P yields the complex

$$\partial P \setminus v = \langle 135, 145, 235, 245 \rangle$$

on five vertices. The proof of Proposition 6.7.9 tells us that we may obtained a closed 2-convex union representation of this complex from the Schlegel diagram of the polar polytope P^*. The polar of P is the cube, and its Schlegel diagram (hence a closed 2-convex union representation of $\partial P \setminus v$) is shown in Figure 6.6.

Corollary 6.7.11. *Let Δ be an arbitrary 2-dimensional simplicial ball. Then Δ is 2-convex union representable.*

Proof. Let v be a vertex not in Δ, and let $\partial\Delta$ denote the boundary of Δ, that is, $\partial\Delta$ is the 1-dimensional subcomplex of Δ whose facets are precisely the free edges of Δ. Let $\Lambda := \Delta \cup (v * \partial\Delta)$. Then Λ is a 2-dimensional simplicial sphere, and so by Steinitz' theorem

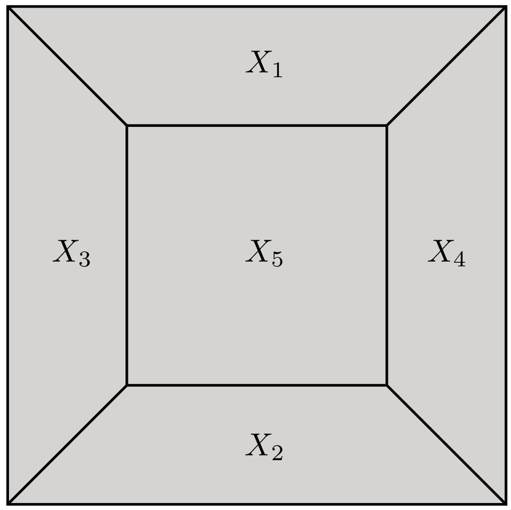


Figure 6.6: A 2-convex union representation $\mathcal{X} = \{X_1, X_2, X_3, X_4, X_5\}$ of $\partial P \setminus v$, where $P \subseteq \mathbb{R}^3$ is the regular octahedron.

(see [Zie95, Chapter 4]), Λ can be realized as the boundary complex of a simplicial polytope. Since $\Delta = \Lambda \setminus v$, the previous proposition implies the result. □

The situation with higher-dimensional balls is much more complicated. For instance, there exist 3-dimensional simplicial balls that are not even collapsible. (See [BL13] for an explicit non-collapsible example with only 15 vertices.) It would be interesting to understand which collapsible triangulations of balls are convex union representable.

6.8 New Local Obstructions to Open and Closed Convexity

Having established various results regarding convex union representable complexes, we now explicitly connect them to the study of convex codes.

Definition 6.8.1. Let $\mathcal{C} \subseteq 2^{[n]}$ and $\sigma \in \Delta(\mathcal{C}) \setminus \mathcal{C}$. We say that $\mathcal{C}$ has a *d-dimensional nerve obstruction* at σ if $\mathrm{Lk}_{\Delta(\mathcal{C})}(\sigma)$ is not d-convex union representable. We say that $\mathcal{C}$ has a *nerve obstruction* at σ if $\mathrm{Lk}_{\Delta(\mathcal{C})}(\sigma)$ is not convex union representable. If $\mathcal{C}$ has no nerve obstructions, then $\mathcal{C}$ is called *locally perfect.*

Nerve obstructions generalize local obstructions and local obstructions of the second kind. Indeed, if $\mathcal{C}$ has a local obstruction of the first or second kind at σ, then $\mathcal{C}$ has a nerve obstruction at σ. A significant advantage of nerve obstructions is that they also capture bounds on the dimension of a code, where local obstructions only capture whether or not a code is convex. The following proposition formalizes these observations.

Proposition 6.8.2. *Let $\mathcal{C} \subseteq 2^{[n]}$ be a code, let $d \geq 0$, and suppose that* $\mathrm{odim}(\mathcal{C}) \leq d$*. Let $\mathcal{U} = \{U_1, \dots, U_n\}$ be an open realization of $\mathcal{C}$ in $\mathbb{R}^d$. Then for any $\sigma \in \Delta(\mathcal{C})\setminus\mathcal{C}$, the collection $\{U_i \cap U_\sigma \mid i \in [n] \setminus \sigma\}$ is a d-convex union representation of* $\mathrm{Lk}_{\Delta(\mathcal{C})}(\sigma)$*. In particular, $\mathcal{C}$ does not have any d-dimensional nerve obstructions, and $\mathcal{C}$ is locally perfect.*

Proof. It has been previously observed in the literature that if $\sigma \in \Delta(\mathcal{C}) \setminus \mathcal{C}$ then $\{U_i \cap U_\sigma \mid i \in [n] \setminus \sigma\}$ is a cover of $U_\sigma \neq \emptyset$, and the nerve of this collection is exactly $\mathrm{Lk}_\Delta(\sigma)$ (see for example [CGJ+17, Section 3.3]). This proves the result. □

Since open convex union representations are interchangeable with closed convex union representations (recall Proposition 6.3.1), we could replace odim in Proposition 6.8.2 with cdim. Thus d-dimensional nerve obstructions provide bounds on both open and closed embedding dimensions of codes. They also allow us to generalize the family of minimally non-convex codes that we exhibited in Theorem 4.4.3.

Corollary 6.8.3. *Let $\Delta \subseteq 2^{[n]}$ be a simplicial complex, and let*

$$\mathcal{C}_\Delta := \{\sigma \cup \{n+1\} \mid \sigma \textit{ is a nonempty face of } \Delta\} \cup \{\emptyset\} \subseteq 2^{[n+1]}.$$

If Δ is not convex union representable (for example, if Δ is one of the complexes Σ_d or E_d of [ABL17]), then $\mathcal{C}_\Delta$ is minimally non-convex.

Proof. Observe that if Δ is not convex union representable then $\mathcal{C}_\Delta$ has a nerve obstruction at $\sigma = \{n+1\}$, and so $\mathcal{C}_\Delta$ is not open convex. We saw in the proof of Theorem 4.4.3 that every proper minor of $\mathcal{C}_\Delta$ is intersection complete, and hence open convex. This proves the result. □

In [CGJ+17] and [CFS19] the authors show that local obstructions and local obstructions of the second kind only occur at intersections of maximal codewords, thus making it easier to search for such obstructions. Below we establish the analogous result for nerve obstructions.

Proposition 6.8.4. *Let $\mathcal{C} \subseteq 2^{[n]}$ be a neural code with a nerve obstruction at $\sigma \subseteq [n]$. Then σ is an intersection of maximal codewords of $\mathcal{C}$.*

Proof. We argue the contrapositive. Suppose that σ is not an intersection of maximal codewords. Then $\text{Lk}_{\Delta(\mathcal{C})}(\sigma)$ is a cone, which is convex union representable by Proposition 6.7.1. Thus $\mathcal{C}$ does not have a nerve obstruction at σ. □

Finally, we provide a result to show that nerve obstructions do not fully characterize convex codes. In fact, one example already exists in the literature, in [LSW17, Theorem 3.1]. The non-convex code on 5 neurons described in this theorem has no local obstructions, and in fact it has no nerve obstructions. (The latter follows from a simple fact that all contractible complexes with at most 4 vertices are convex union representable.) Below, we provide an infinite family of locally perfect non-convex codes.

Theorem 6.8.5. *The minimally non-convex codes $\mathcal{C}_n$ of Definition 5.6.1 are locally perfect.*

Proof. Recall from the proof of Theorem 5.6.6 that the only missing max intersection in $\mathcal{C}_n$ is $\{n+1\}$, which has a link with facets $[n] \cup \{\overline{n+1}\}$ and $([n] \setminus \{i\}) \cup \{\bar{i}\}$ for all $i \in [n]$. The simplex $2^{[n]\cup\{\overline{n+1}\}}$ is certainly convex union representable, and for each $i \in [n]$ we may apply Proposition 6.7.5 with $\sigma = \omega = [n] \setminus \{i\}$ to add the remaining simplices in the link while preserving convex union representability. This proves that $\mathcal{C}_n$ is locally perfect. □

Chapter 7

DISCUSSION AND OPEN PROBLEMS

We have made progress on several fronts in the study of convex codes and their embedding dimensions. We proved new upper bounds for intersection complete codes in Chapter 2, we introduced and developed the framework of morphisms and minors in Chapters 3 and 4, and we developed several discrete geometric results and used them to prove new lower bounds on embedding dimension in Chapters 5 and 6. All of this progress reveals new and enticing questions, many of which are quite accessible.

Some of these questions lie in a more abstract or theoretical direction, which can feel separate from the original neuroscientific motivation for studying convex codes. However, answers to these questions may highlight the limitations or strengths of convex codes as a model of place cells, and could potentially motivate experimental work. For example, results such as those in Chapter 5 motivate us to ask whether or not k-flexible sunflowers appear regularly in receptive fields of place cells. If not, this would indicate that convex codes may be too general as a model of place cell activity. If so, our results may help extract information from this place cell activity.

Results regarding the combinatorial sunflower lemma and sunflower conjecture imply that combinatorial sunflowers will appear in large enough codes (see [ALWZ20, ER60]). Sunflowers in a code do not mean sunflowers in a realization of the code (the intersection of sets in a realization corresponds to a union of codewords), but one could take a Ramsey-theoretic approach to investigating embedding dimension. For example, are sufficiently large codes guaranteed to have certain types of minors? If so, this would help bound the embedding dimensions of large codes.

Developing the theory of convex codes is also useful mathematically. The theory itself is quite rich, and more importantly, the tools we develop to study it may find more general applications. Sunflowers and convex union representable complexes may appear in other contexts, and the combinatorics of morphisms and minors could be applied to study other families of codes. It would be natural to investigate applications of convex code results to other areas where intersection patterns of sets in Euclidean space play a role, such as topological data analysis.

In the following sections we highlight some of the natural questions and next steps that arise from our work.

7.1 Open Problems on Constructive Geometric Results

The results in Chapter 2 apply only to intersection complete codes, whereas [CGIK16] was able to prove their results for max-intersection complete codes. Since max-intersection complete codes are both open and closed convex, it makes sense to ask whether our results can be extended to them.

Question 7.1.1. *Can Theorem 2.2.7 be extended to max-intersection complete codes? That is, if $\mathcal{C}$ is max-intersection complete is it always true that $\operatorname{cdim}(\mathcal{C}) \leq \operatorname{odim}(\mathcal{C})$?*

Question 7.1.2. *Can Theorem 2.3.7 be extended to max-intersection complete codes? That is, if $\mathcal{C} \subseteq 2^{[n]}$ is max-intersection complete is it always true that $\operatorname{cdim}(\mathcal{C}) \leq \min\{2d+1, n-1\}$ where $d = \dim(\Delta(\mathcal{C}))$?*

Theorems 2.2.7 and 2.3.7 provided upper bounds on closed embedding dimension, and relied on the combinatorial structure of intersection complete codes. Analogous results are not known for open embedding dimension, but it is entirely possible that there are families of codes where the same results hold with $\operatorname{odim}(\mathcal{C})$ and $\operatorname{cdim}(\mathcal{C})$ swapped.

Question 7.1.3. *Is there a natural family of codes (beyond simplicial complexes) with the property that $\operatorname{odim}(\mathcal{C}) \leq \operatorname{cdim}(\mathcal{C})$? Likewise, is there a family in which $\operatorname{odim}(\mathcal{C}) \leq \min\{2d+1, n-1\}$?*

A key ingredient in Theorem 2.2.7 was the fact that we may trim an open realization of an intersection complete code by a small positive amount to obtain a non-degenerate realization, without changing the realized code. Trimming an open set by ε is equivalent to adding a closed ε-ball to the complement of the set. Thus we can view "trimming" as dual to the operation "adding a closed ε-ball."

It is then natural to ask which families of codes have closed realizations in which adding a small closed ball yields a non-degenerate closed realization of the same code. Such a family would have the property that $\text{odim}(\mathcal{C}) \leq \text{cdim}(\mathcal{C})$, yielding progress on Question 7.1.3, and moreover such a family may be (in some sense) "dual to" intersection complete codes.

Question 7.1.4. *Is there a natural family of codes (beyond simplicial complexes) whose closed realizations can be augmented by a Minowski sum with a small closed ball without changing the realized code?*

Results such as Corollary 5.3.7 tell us that a family answering Questions 7.1.3 and 7.1.4 cannot include all intersection complete codes (since the inequality $\text{cdim}(\mathcal{C}) \leq \text{odim}(\mathcal{C})$ is strict for many intersection complete codes).

7.2 Open Problems on Morphisms and Minors

Given that we introduced morphisms and minors only a few years ago, there are many basic but unanswered questions in this framework. To get a sense for the lay of the land, one natural first step would be to improve our understanding of how existing families of codes side inside $\mathbf{P_{Code}}$. Corollary 3.4.3 provided a nice characterization of intersection complete codes: they are exactly the minors of simplicial complexes. Does such a characterization exist for max-intersection complete codes (which also form a minor-closed family in $\mathbf{P_{Code}}$)?

Question 7.2.1. *Is there a natural family of codes whose minors are exactly max-intersection complete codes?*

If $\mathcal{C}$ is an intersection complete code, we know it is a minor of some simplicial complex. However, it is very unclear exactly which simplicial complexes it is a minor of, or whether there is a unique minimal such complex. Investigating this question would provide a better picture of $\mathbf{P}_{\mathbf{Code}}$, and could be done computationally for small examples to start off.

Question 7.2.2. *Let $\mathcal{C}$ be an intersection complete code. Among all simplicial complexes that $\mathcal{C}$ is a minor of, is there a unique minimal one (with respect to minors)?*

One reason to investigate Question 7.2.2 is that the set of simplicial complexes lying above $\mathcal{C}$ in $\mathbf{P}_{\mathbf{Code}}$ may capture important combinatorial or geometric information about $\mathcal{C}$. In the best case, such complexes may provide new (or even exact) bounds on the embedding dimension of $\mathcal{C}$. An affirmative answer to the question below would reduce the study of embedding dimensions of intersection complete codes to the study of embedding dimensions for simplicial complexes.

Question 7.2.3. *Let $\mathcal{C}$ be an intersection complete code. Among all simplicial complexes that $\mathcal{C}$ is a minor of, is there one with open embedding dimension equal to* $\operatorname{odim}(\mathcal{C})$*?*

We saw in Theorems 4.4.3 and 5.6.13 that there are an infinite number of minimally non-convex codes. Thus open convexity does not admit a characterization by finitely many forbidden minors in our framework. However, other families of codes may admit such a characterization, or a characterization by finitely many "nice" families of forbidden minors.

Question 7.2.4. *Are there any interesting minor-closed families of codes that admit a characterization by finitely many forbidden minors? (For example, do axis-parallel box codes admit such a characterization? What about codes with no more than k maximal codewords?)*

One of the results of [CFS19] is that locally good codes are exactly those which admit a good cover realization. Thus Theorem 4.3.4 implies that locally good codes form a minor-closed family. We do not know whether the same is true for locally great codes or locally perfect codes. The locally perfect case may be easier to investigate since convex union representability is more concretely geometric than collapsibility. On the other hand, perhaps the combinatorial nature of collapsibility plays well with morphisms and minors.

Question 7.2.5. *Do locally great codes form a minor-closed family? How about locally perfect codes?*

The authors in [KLR20] connect the study of convex codes to the study of oriented matroids via $\mathbf{P_{Code}}$. In particular, they associate every oriented matroid to a code in such a way that the resulting code is open convex when the matroid is representable. The authors raise the following question, which would amount to "axiomatizing convexity" for neural codes, in the same sense that matroids axiomatize independence.

Question 7.2.6 (Question 6.4 of [KLR20])**.** *Can the minors of oriented matroid codes (in* $\mathbf{P_{Code}}$*) be characterized by a set of combinatorial axioms?*

Several computational questions remain open in the study of morphisms and minors. We do not currently have the ability to recgonize whether one code is a minor of another, other than by trial and error or brute force. An efficient algorithm for this would greatly improve our ability to contextualize existing and future results via minors.

Question 7.2.7. *Is it possible to efficiently recognize whether one code is a minor of another?*

As mentioned previously, we also do not yet have an elegant description of the covering relation in $\mathbf{P_{Code}}$ "from below." We have made some unpublished progress on this question, and invite the interested reader to get in touch via email for discussion and collaboration.

Question 7.2.8. *Given a code* $\mathcal{C}$*, can one efficiently or elegantly describe the codes that cover* $\mathcal{C}$ *in* $\mathbf{P_{Code}}$*?*

Understanding the structure of $\mathbf{P_{Code}}$ as a poset from a macroscopic or theoretical perspective would also be useful. Doing so may provide some intuition for the strengths and weaknesses of using minors to analyze convex codes. We can think of two immediate questions in this vein.

Question 7.2.9. *Among all codes of rank* k*, on average how many codes of rank* $k-1$ *does each cover? In other words, what is the "average downward degree" of a code of rank* k *in* $\mathbf{P_{Code}}$*?*

Question 7.2.10. *Are intervals in* $\mathbf{P_{Code}}$ *unimodal? That is, do the number of codes of each rank in an interval of* $\mathbf{P_{Code}}$ *always form a unimodal sequence?*

7.3 Open Problems on Sunflowers of Open Convex Sets

In Chapter 5 we used Theorem 5.1.13 to exhibit families of codes with new embedding dimension behavior, such as an arbitrarily large finite gap between open and closed embedding dimension. However, for every example we have seen (and indeed, every known example in the literature) we have that $\operatorname{cdim}(\mathcal{C}) \leq \operatorname{odim}(\mathcal{C})$ whenever both quantities are finite. It is natural to ask whether there exists a code $\mathcal{C}$ with $\operatorname{odim}(\mathcal{C}) < \operatorname{cdim}(\mathcal{C}) < \infty$. Thanks to recent work in [CJL$^+$20] that introduces "rigid structures" in codes, we have been able to answer this question affirmatively (and much more generally).

Result in Preparation 7.3.1 ([Jef21])**.** *For all natural numbers* a, b, c, d *with*

$$2 \leq a \leq \min(b, c) \quad \text{and} \quad \max(b, c) \leq d$$

there exists a code $\mathcal{C}_{(a,b,c,d)}$ *with "good cover embedding dimension" equal to* a*,* $\operatorname{odim}(\mathcal{C}) = b$*,* $\operatorname{cdim}(\mathcal{C}) = c$*, and "non-degenerate embedding dimension" equal to* d*.*

It would be interesting to augment this result by explaining the relationship between the embedding dimensions mentioned above, and other invariants such as the PL embedding dimension or topological embedding dimension of $\Delta(\mathcal{C})$. One example of work in this spirit is [Tan11], which develops a relationship between d-representability of a simplicial complex and Van Kampen obstructions. Such connections could also shed light on computational aspects of characterizing open and closed convex codes (see for example [MSTW18], which shows that one can decide algorithmically whether or not a 2-dimensional complex admits an embedding into $\mathbb{R}^3$).

Result in Preparation 7.3.2 ([Jef21])**.** *Let* $\mathcal{C}$ *be a code. The following are equivalent:*

- $\mathcal{C}$ *has "good cover embedding dimension" equal to 1,*

- odim($\mathcal{C}$) $= 1$,
- cdim($\mathcal{C}$) $= 1$, *and*
- $\mathcal{C}$ *has "non-degenerate embedding dimension" equal to* 1.

Our forthcoming work mentioned above does not yet provide a "closed version" of Corollary 5.3.7. We leave this as an open problem for now.

Question 7.3.3. *Does there exist a family of codes whose closed embedding dimension grows exponentially as a function of the number of neurons in the codes, mirroring Corollary 5.3.7? As a start, does there exist a code* $\mathcal{C} \subseteq 2^{[n]}$ *with* $\operatorname{cdim}(\mathcal{C}) > n - 1$*?*

The codes $\mathcal{T}_n$ of Section 5.7 remain somewhat mysterious. Although they are a very restricted example, characterizing their embedding dimensions more precisely could lead to general insights or interesting techniques.

Question 7.3.4. *What is the precise value of the embedding dimension* $t_n = \operatorname{odim}(\mathcal{T}_n)$ *from Definition 5.7.1? As a start, does* $t_6 = 4$ *or does* $t_6 = 5$*?*

The bounds provided in Section 5.8 also leave some potential room for improvement. Understanding the extreme cases in these bounds (or improving them) could be a productive task.

Question 7.3.5. *Do there exist codes achieving the extremes of the bound* $\lceil \frac{m}{k} \rceil \leq \operatorname{odim}(\mathcal{S}_{\mathcal{C}/\mathcal{D}}) \leq m$ *from Proposition 5.8.4 for all possible choices of* m *and* k*?*

Question 7.3.6. *Do there exist special cases in which we can tighten the bound* $\lceil \frac{m}{k} \rceil \leq \operatorname{odim}(\mathcal{S}_{\mathcal{C}/\mathcal{D}}) \leq m$ *from Proposition 5.8.4?*

Finally, Proposition 5.8.4 generalizes Proposition 5.3.3 but this generalization is slightly incomplete. In particular, Proposition 5.3.3 implies that $\mathcal{S}_n$ is the *unique* minimal minor of all $\mathcal{S}_\Delta$ codes, but Proposition 5.8.4 only tells us that every minimal $\mathcal{S}_{\mathcal{C}/\mathcal{D}}$ code takes the form $\mathcal{S}_{\mathcal{C}/\min}$. It is unclear whether the converse holds (i.e. whether or not $\mathcal{S}_{\mathcal{C}/\min}$ is always minimal among $\mathcal{S}_{\mathcal{C}/\mathcal{D}}$ codes).

Question 7.3.7. *Among the collection of codes $\mathcal{S}_{\mathcal{C}/\mathcal{D}}$ with fixed parameters m and k from Definition 5.8.1, are the codes $\mathcal{S}_{\mathcal{C}/\min}$ the* exact *set of minimal minors? In other words, do the various non-isomorphic $\mathcal{S}_{\mathcal{C}/\min}$ with parameters m and k form an antichain in $\mathbf{P}_{\mathbf{Code}}$?*

7.4 Open Problems on Convex Union Representable Complexes

Since a complex Δ is d-representable if and only if $\Delta * v$ is d-convex union representable, recognizing d-convex union representable complexes is NP-hard for $d \geq 2$ (see [Tan13, Section 4.1]). However, we do not know whether the problem of recognizing d-convex union representable complexes is even decidable.

Question 7.4.1. *Is the computational problem of recognizing d-convex union representable complexes decidable?*

Non-evasiveness is a natural combinatorial specialization of collapsibility for simplicial complexes (see [Wel99, KSS84]). We do not know whether convex union representability implies non-evasiveness. We would also like to know more about how convex union representability relates to other important classes of simplicial complexes, such as shellable and constructible complexes.

Question 7.4.2. *Is every convex union representable complex non-evasive?*

Question 7.4.3. *Is the Alexander dual of a convex union representable complex also convex union representable?*

Question 7.4.4. *Is every shellable (or constructible) simplicial ball a convex union representable complex?*

Question 7.4.5. *Can one characterize the class of topological spaces that possess convex union representable triangulations?*

Recall from Theorem 1.2.7 that when Δ is d-representable, we may assume that d is no larger than $2\dim(\Delta)+1$. When Δ is d-convex union representable, does a similar statement hold? Or is it possible that convex union representability captures sufficiently complex geometric structure to force a large embedding dimension with a low-dimensional simplicial complex? Generally, the relationship between d-representations and d-convex union representations would be interesting to understand.

Question 7.4.6. *For a fixed $d \geq 2$, do there exist d-dimensional complexes that are only convex union representable in an arbitrarily high dimension? As a start, do there exist d-dimensional complexes that are only convex union representable in dimension larger than $2d+1$?*

Question 7.4.7. *Does there exist a complex that is d-representable, $(d+1)$-convex union representable, and not d-convex union representable?*

Note that an affirmative answer to Question 7.4.6 would imply an affirmative answer to Question 7.4.7.

Finally, it would be useful to understand how combinatorial operations such as barycentric subdivision interact with d-convex union representability. We remark that even though convex union representable complexes are collapsible, collapsing does not preserve convex union representability: if Δ is collapsible but not convex union representable, then $\Delta * v$ is a convex union representable complex that collapses to Δ.

Question 7.4.8. *Does every collapsible complex become convex union representable after sufficiently many subdivisions? As a start, do subdivisions preserve convex union representability?*

Remark 7.4.9. Perhaps surprisingly, subdivisions do not preserve d-convex union representability. Indeed, the simplex $\Delta = 2^{[n+1]}$ is 1-convex union representable, but its first barycentric subdivision is not $(n-1)$-representable since it has an induced subcomplex homeomorphic to the boundary of Δ, which has nonzero $(n-1)$-st homology. This indicates that Question 7.4.8 may have a negative answer.

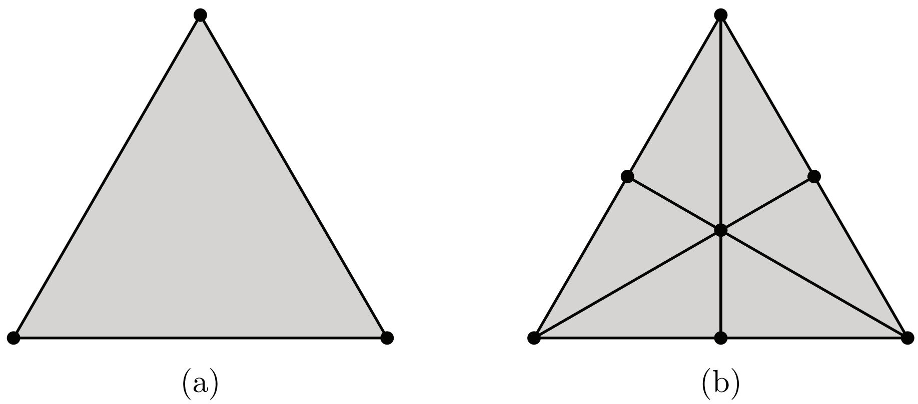


Figure 7.1: (a) The simplicial complex $2^{[3]}$, which is 1-convex union representable (b) The barycentric subdivision of $2^{[3]}$, which is not 1-convex union representable.

7.5 Other Open Problems

Finally, we highlight two problems that do not fit into the categories above. First, a key ingredient in Section 5.5 was the guarantee that certain regions in every closed realization of a code were sufficiently large in dimension. Our arguments for this were ad hoc, and a general framework for such arguments would be a useful tool in generating new families of codes with interesting embedding dimensions or realizations.

Question 7.5.1. *Can one provide general combinatorial criteria on a code $\mathcal{C}$ guaranteeing lower bounds on the dimension of regions X_σ in every closed realization $\mathcal{X}$ of $\mathcal{C}$?*

Sections 5.4 and 5.5 investigated monotonicity of of convexity. The current formulation of monotonicity of convexity guarantees that open embedding dimension cannot increase too much when adding a new non-maximal codeword to a code. Very little is understood about how much the embedding dimension can decrease, however. Answers to this question may be of interest in an experimental context, in which one has partial sample of a code and wishes to know which new non-maximal codewords should be added to obtain a code with lower embedding dimension.

Question 7.5.2. *Let* $\mathcal{C}$ *be a code and let* $\sigma \in \Delta(\mathcal{C}) \setminus \mathcal{C}$*. How much smaller can* $\operatorname{odim}(\mathcal{C} \cup \{\sigma\})$ *be than* $\operatorname{odim}(\mathcal{C})$*?*

BIBLIOGRAPHY

[ABL17] Karim A. Adiprasito, Bruno Benedetti, and Frank H. Lutz, *Extremal examples of collapsible complexes and random discrete Morse theory*, Discrete & Computational Geometry. An International Journal of Mathematics and Computer Science **57** (2017), no. 4, 824–853. MR 3639606

[ADLS17] Nina Amenta, Jesús A. De Loera, and Pablo Soberón, *Helly's theorem: new variations and applications*, Algebraic and geometric methods in discrete mathematics, Contemp. Math., vol. 685, Amer. Math. Soc., Providence, RI, 2017, pp. 55–95. MR 3625571

[AHS06] Jiří Adámek, Horst Herrlich, and George E. Strecker, *Abstract and concrete categories*, TAC Reprints, no. 17, Reprints in Theory and Applications of Categories, 2006.

[ALWZ20] Ryan Alweiss, Shachar Lovett, Kewen Wu, and Jiapeng Zhang, *Improved bounds for the sunflower lemma*, Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing (New York, NY, USA), Association for Computing Machinery, 2020, p. 624–630.

[Bjö95] Anders Björner, *Topological methods*, Handbook of combinatorics, Vol. 1, 2, Elsevier Sci. B. V., Amsterdam, 1995, pp. 1819–1872. MR 1373690

[BL13] Bruno Benedetti and Frank H. Lutz, *Knots in collapsible and non-collapsible balls*, Electronic Journal of Combinatorics **20** (2013), no. 3. MR 3104529

[BM12] Jonathan Ariel Barmak and Elias Gabriel Minian, *Strong homotopy types, nerves and collapses*, Discrete & Computational Geometry **47** (2012), no. 2, 301–328. MR 2872540

[Bor48] K. Borsuk, *On the imbedding of systems of compacta in simplicial complexes*, Fundamenta Mathematicae **35** (1948), 217–234.

[BYBOS95] Rani Ben-Yishai, Raphael Bar-Or, and Haim Sompolinsky, *Theory of orientation tuning in visual cortex*, Proceedings of the National Academy of Sciences of the United States of America **92** (1995), 3844–3848.

[CFS19] Aaron Chen, Florian Frick, and Anne Shiu, *Neural codes, decidability, and a new local obstruction to convexity*, SIAM Journal on Applied Algebra and Geometry **3** (2019), no. 1, 44–66.

[CGIK16] Joshua Cruz, Chad Giusti, Vladimir Itskov, and Bill Kronholm, *On open and closed convex codes*, Discrete & Computational Geometry **61** (2016), 247–270.

[CGJ+17] Carina Curto, Elizabeth Gross, Jack Jeffries, Katherine Morrison, Mohamed Omar, Zvi Rosen, Anne Shiu, and Nora Youngs, *What makes a neural code convex?*, SIAM Journal on Applied Algebra and Geometry **1** (2017), no. 1, 222–238. MR 3633775

[CGJ+19] Carina Curto, Elizabeth Gross, Jack Jeffries, Katherine Morrison, Zvi Rosen, Anne Shiu, and Nora Youngs, *Algebraic signatures of convex and non-convex codes*, Journal of Pure and Applied Algebra **223** (2019), no. 9, 3919–3940.

[CIVCY13] Carina Curto, Vladimir Itskov, Alan Veliz-Cuba, and Nora Youngs, *The neural ring: an algebraic tool for analyzing the intrinsic structure of neural codes*, Bulletin of Mathematical Biology **75** (2013), no. 9, 1571–1611. MR 3105524

[CJL+20] Patrick Chan, Katherine Johnston, Joseph Lent, Alexander Ruys de Perez, and Anne Shiu, *Nondegenerate neural codes and obstructions to closed-convexity*, arXiv e-prints: 2011.04565 (2020).

[CK20] Katie Christensen and Hamid Kulosman, *Some remarks about trunks and morphisms of neural codes*, arXiv e-prints: 1904.04470 (2020).

[CV16] Carina Curto and Ramón Vera, *The Leray dimension of a convex code*, arXiv e-prints: 1612.07797 (2016).

[Dav18] Robert Davis, *State polytopes related to two classes of combinatorial neural codes*, arXiv e-prints: 1808.06721 (2018).

[ER60] Paul Erdös and Richard Rado, *Intersection theorems for systems of sets*, Journal of the London Mathematical Society **s1-35** (1960), no. 1, 85–90.

[FG65] D. R. Fulkerson and O. A. Gross, *Incidence matrices and interval graphs*, Pacific Journal of Mathematics **15** (1965), 835–855. MR 0186421

[GGPK+18] Rebecca Garcia, Luis Garcia-Puente, Ryan Kruse, Jessica Liu, Dane Miyata, Ethan Petersen, Kaitlyn Phillipson, and Anne Shiu, *Gröbner bases of neural ideals*, International Journal of Algebra and Computation **28** (2018), no. 4, 553–571.

[GI14] Chad Giusti and Vladimir Itskov, *A no-go theorem for one-layer feedforward networks*, Neural Computation **26** (2014), no. 11, 2527–2540. MR 3243436

[GJAM+20] Roddy M. Grieves, Selim Jedidi-Ayoub, Karyna Mishchanchuk, Anyi Liu, Sophie Renaudineau, and Kate J. Jeffery, *The place-cell representation of volumetric space in rats*, Nature Communications **11** (2020).

[GJMS20] Brianna Gambacini, R. Amzi Jeffs, Sam Macdonald, and Anne Shiu, *Non-monotonicity of closed convexity in neural codes*, arXiv e-prints: 1912.00963 (2020).

[GJS19] Sema Gunturkun, Jack Jeffries, and Jeffrey Sun, *Polarization of neural rings*, Journal of Algebra and Its Applications **19** (2019), no. 8.

[GNY16] Elizabeth Gross, Kazi Obatake Nida, and Nora Youngs, *Neural ideals and stimulus space visualization*, Advances in Applied Mathematics **95** (2016), 65–95.

[GP20] Sarah Ayman Goldrup and Kaitlyn Phillipson, *Classification of open and closed convex codes on five neurons*, Advances in Applied Mathematics **112** (2020), 101948.

[IKR20] Vladimir Itskov, Alexander Kunin, and Zvi Rosen, *Hyperplane neural codes and the polar complex*, Topological Data Analysis - The Abel Symposium, 2018 (Nils A. Baas, Gereon Quick, Markus Szymik, Marius Thaule, and Gunnar E. Carlsson, eds.), Abel Symposia, Springer, 2020, pp. 343–369.

[Jef19a] R. Amzi Jeffs, *Embedding dimension phenomena in intersection complete codes*, arXiv e-prints: 1909.13406 (2019).

[Jef19b] R. Amzi Jeffs, *Sunflowers of convex open sets*, Advances in Applied Mathematics **111** (2019), 101935.

[Jef20] R. Amzi Jeffs, *Morphisms of neural codes*, SIAM Journal on Applied Algebra and Geometry **4** (2020), 99–122.

[Jef21] R. Amzi Jeffs, *Open, closed, and non-degenerate embedding dimensions of neural codes*, In preparation (2021).

[JLY20] R. Amzi Jeffs, Caitlin Lienkaemper, and Nora Youngs, *Order-forcing in neural codes*, arXiv e-prints: 2008.13192 (2020).

[JN19] R. Amzi Jeffs and Isabella Novik, *Convex union representability and convex codes*, International Mathematics Research Notices (2019).

[JOS+15] R. Amzi Jeffs, Mohamed Omar, Natchanon Suaysom, Aleina Wachtel, and Nora Youngs, *Sparse neural codes and convexity*, Involve, a Journal of Mathematics **12** (2015), no. 5, 737–754.

[JSS20] Katherine Johnston, Anne Shiu, and Clare Spinner, *Neural codes with three maximal codewords: convexity and minimal embedding dimension*, arXiv e-prints: 2008.13192 (2020).

[Kal85] Gil Kalai, *f-vectors of acyclic complexes*, Discrete Mathematics **55** (1985), no. 1, 97–99. MR 793634

[KKM95] James Knierim, Hemant Kudrimoti, and Bruce McNaughton, *Place cells, head direction cells, and the learning of landmark stability*, The Journal of Neuroscience **15** (1995), 1648–1659.

[KLR20] Alexander Kunin, Caitlin Lienkaemper, and Zvi Rosen, *Oriented matroids and combinatorial neural codes*, arXiv e-prints: 2002.03542 (2020), arXiv:2002.03542.

[KSS84] Jeff Kahn, Michael Saks, and Dean Sturtevant, *A topological approach to evasiveness*, Combinatorica **4** (1984), no. 4, 297–306.

[LSW17] Caitlin Lienkaemper, Anne Shiu, and Zev Woodstock, *Obstructions to convexity in neural codes*, Advances in Applied Mathematics **85** (2017), 31–59. MR 3595298

[Mat02] Jiří Matoušek, *Lectures on discrete geometry*, Graduate Texts in Mathematics, vol. 212, Springer-Verlag, New York, 2002. MR 1899299

[MRM15] May-Britt Moser, David Rowland, and Edvard Moser, *Place cells, grid cells, and memory*, Cold Spring Harbor Perspectives in Medicine **5** (2015).

[MSTW18] Jiří Matoušek, Eric Sedgwick, Martin Tancer, and Uli Wagner, *Embeddability in the 3-sphere is decidable*, J. ACM **65** (2018), no. 1.

[MT20] Raffaella Mulas and Ngoc M Tran, *Minimal embedding dimensions of connected neural codes*, Algebraic Statistics **11** (2020), no. 1, 99–106.

[OD71] John O'Keefe and Jonathan Dostrovsky, *The hippocampus as a spatial map. preliminary evidence from unit activity in the freely-moving rat*, Brain Research (1971), 171–175.

[Pm85] G. Ya. Perel' man, *Realization of abstract k-skeletons as k-skeletons of intersections of convex polyhedra in* $\mathbf{R}^{2k-1}$, Geometric questions in the theory of functions and sets, Kalinin. Gos. Univ., Kalinin, 1985, pp. 129–131. MR 829936

[RMS20] Alexander Ruys de Perez, Laura Felicia Matusevich, and Anne Shiu, *Neural codes and the factor complex*, Advances in Applied Mathematics **114** (2020), 101977.

[RZ17] Zvi Rosen and Yan X. Zhang, *Convex neural codes in dimension 1*, arXiv e-prints (2017), arXiv:1702.06907.

[Tan11] Martin Tancer, *d-representability of simplicial complexes of fixed dimension*, Journal of Computational Geometry **2** (2011), no. 1, 183–188.

[Tan13] ______, *Intersection patterns of convex sets via simplicial complexes: a survey*, Thirty essays on geometric graph theory, Springer, New York, 2013, pp. 521–540. MR 3205172

[WB74] David Watkins and Mark Berkley, *The orientation selectivity of single neurons in cat striate cortex*, Experimental Brain Research **19** (1974), 433–446.

[Weg67] Gerd Wegner, *Eigenschaften der nerven homologisch-einfacher familien im* $\mathbb{R}^n$, Ph.D. thesis, Universität Göttingen, 1967.

[Weg75] ______, *d-collapsing and nerves of families of convex sets*, Archiv der Mathematik **26** (1975), 317–321. MR 0375333

[Wel99] Volkmar Welker, *Constructions preserving evasiveness and collapsibility*, Discrete Mathematics **207** (1999), no. 1-3, 243–255. MR 1710494

[WYM18] Melville J. Wohlgemuth, Chao Yu, and Cynthia F. Moss, *3D hippocampal place field dynamics in free-flying echolocating bats*, Frontiers in Cellular Neuroscience **12** (2018), 270.

[Zee63] E. Christopher Zeeman, *Seminar on combinatorial topology, fascicule 1 (exposés I à V inclus)*, Institut des Hautes Etudes Scientifiques, 1963.

[Zie95] Günter M. Ziegler, *Lectures on polytopes*, Graduate Texts in Mathematics, vol. 152, Springer-Verlag, New York, 1995. MR 1311028

[ZMV15] Sijie Zhang and Denise Manahan-Vaughan, *Spatial olfactory learning contributes to place field formation in the hippocampus*, Cerebral Cortex **25** (2015), 423–432.