Covering Relations in the Poset of Combinatorial Neural Codes

R. Amzi Jeffs AMZI.JEFFS@PNNL.GOV
Pacific Northwest National Laboratory, Richland, Washington.

Trong-Thuc Trang TTRANG2019@FAU.EDU
Florida Atlantic University, Boca Raton, Florida.

Editors: List of editors' names

Abstract

A combinatorial neural code is a subset of the power set $2^{[n]}$ on $[n] = \{1, \dots, n\}$, in which each $1 \leq i \leq n$ represents a neuron and each element (codeword) represents the co-firing event of some neurons. Consider a space $X \subseteq \mathbb{R}^d$, simulating an animal's environment, and a collection $\mathcal{U} = \{U_1, \dots, U_n\}$ of open subsets of X. Each $U_i \subseteq X$ simulates a place field which is a specific region where a place cell i is active. Then, the code of $\mathcal{U}$ in X is defined as $\text{code}(\mathcal{U}, X) = \left\{ \sigma \subseteq [n] \middle| \bigcap_{i \in \sigma} U_i \setminus \bigcup_{j \notin \sigma} U_j \neq \varnothing \right\}$. If a neural code $\mathcal{C} = \text{code}(\mathcal{U}, X)$ for some X and $\mathcal{U}$, we say $\mathcal{C}$ has a realization of open subsets of some space X. Although every combinatorial neural code obviously has a realization by some open subsets, determining whether it has a realization by some open convex subsets remains unsolved. Many studies attempted to tackle this decision problem, but only partial results were achieved. In fact, a previous study showed that the decision problem of convex neural codes is NP-hard. Furthermore, the authors of this study conjectured that every convex neural code can be realized as a minor of a neural code arising from a representable oriented matroid, which can lead to an equivalence between convex and polytope convex neural codes. Even though this conjecture has been confirmed in dimension two, its validity in higher dimensions is still unknown. To advance the investigation of this conjecture, we provide a complete characterization of the covering relations within the poset $\mathbf{P_{Code}}$ of neural codes.

Keywords: intrinsic geometric interpretation, neural codes, convexity, combinatorial topology, combinatorial geometry.

1. Introduction

In 1971, John O'Keefe and Jonathan Dostrovsky discovered individual hippocampal neurons in rats that were active when the rats were in specific locations (O'Keefe and Dostrovsky, 1971). These neurons are known as *place cells*, and the regions in the environment where these neurons fire are called *place fields*. For long, these neurons have been thought of as encoding a cognitive map of a mammal's environment. Many mathematical tools such as algebraic topology have been used to decode any meaningful representation of neural signals coming from place cells. Curto and Itskov applied topological data analysis to place cell data in (Curto and Itskov, 2008) and found that many topological features of the stimulus space can be extracted from cell groups when place fields were assumed to be convex and even when a small portion of them was multi-peaked. Every neural code obtained from cell groups also turns out to carry one geometric information of place fields, their convexity (Curto et al., 2013), (Curto, 2017). This encoded information is intrinsic because it can be

inferred from the neural signals of place cells alone. To have a clearer picture, we provide the following definitions and some conventions made for this work.

Place fields that correspond to active place cells in a mammalian hippocampus can be topologically simulated by open subsets of some ambient topological space representing the environment that the animal explores. Then, a code can be naturally derived from such realization of open subsets as follows. Note that, throughout this work, any ambient space is restricted to some Euclidean space $\mathbb{R}^d$.

Definition 1 (Codes of covers) *Given a cover $\mathcal{U} = \{U_1, \ldots, U_n\}$ of open subsets of some Euclidean space $\mathbb{R}^d$. The **code of $\mathcal{U}$ in** $\mathbb{R}^d$ is defined as*

$$\text{code}(\mathcal{U}, \mathbb{R}^d) = \left\{ \sigma \subseteq [n] \middle| \bigcap_{i \in \sigma} U_i \setminus \bigcup_{j \notin \sigma} U_j \neq \varnothing \right\}.$$

Conventionally, if $\sigma = \varnothing$, then the intersection of U_i over all $i \in \sigma$ is $\mathbb{R}^d$. The cover $\mathcal{U}$ is called a *realization* of $\text{code}(\mathcal{U}, \mathbb{R}^d)$. The definition of combinatorial neural codes is given below. Some past works define this mathematical structure using binary patterns, but they are equivalent.

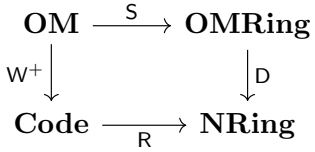
Definition 2 (Combinatorial neural codes) *A **(combinatorial) neural code** $\mathcal{C}$ is a collection of some subsets of $[n] = \{1, \ldots, n\}$ for some natural number n. Elements of $[n]$ represent **neurons** and elements of $\mathcal{C}$ called **codewords** represent the co-firing events of some neurons.*

In this work, every combinatorial neural code must contain the $\varnothing$-codeword[1] denoted by $\emptyset$. Without any ambiguity, instead of writing a codeword as a subset of $[n]$ (e.g., $\{1, 2\}$) we usually write it compactly as a string of numbers in ascending numerical order (e.g., 12). Hereafter, the term *neural code* or, shortly, *code* will be used interchangeably to replace the full term *combinatorial neural code*.

Obviously, every neural code $\mathcal{C} = \text{code}(\mathcal{U}, \mathbb{R}^d)$ for some $\mathbb{R}^d$ and $\mathcal{U}$, and we say that $\mathcal{C}$ has a realization. Furthermore, if there is a realization $\mathcal{U}$ consisting of open convex subsets of $\mathbb{R}^d$, we say that $\mathcal{C}$ is an *open convex neural code*. Many partial attempts have been made to answer the question "When is a neural code convex?". The list includes, but is not limited to (Curto et al., 2013), (Giusti and Itskov, 2014), (Curto and Youngs, 2015.), (Curto et al., 2017), (Cruz et al., 2016), (Gross et al., 2016), (Lienkaemper et al., 2017), (Mulas and Tran, 2020), (Jeffs et al., 2018), (Gambacini et al., 2019), and (Chen et al., 2019). To advance the study of open convex neural codes, Jeffs introduced in (Jeffs, 2020) the poset of combinatorial neural codes $\mathbf{P_{Code}}$ and connected the category of neural codes **Code** with the category of neural rings **NRing** studied in previous research. Continuing on this, Kunin, Lienkaemper, and Rosen built connections of these two categories with the categories of oriented matroids **OM** and oriented matroid rings **OMRing** in (Kunin et al.,

1. In reality, this means that some certain region of the environment is not covered by any place field.

2020).



These connections gave them a nice way of showing that the decision problem of open convex neural codes is in fact NP-hard and that open convex polytope neural codes (a subclass of open convex neural codes) are essentially those that lie below some representable oriented matroids in $\mathbf{P_{Code}}$. They also conjectured that open convex neural codes are open convex polytope.

Conjecture 3 (Kunin et al., 2020) *A neural code $\mathcal{C}$ is convex if and only if $\mathcal{C}$ lies below some representable oriented matroid in $\mathbf{P_{Code}}$.*

The above conjecture is true in the plane $\mathbb{R}^2$ (Bukh and Jeffs, 2023), but has not been proven in general. If this conjecture holds, we can avoid the decision problem of open convex neural codes and instead check the representability of all oriented matroid codes lying above a given code. While being $\exists\mathbb{R}$-hard[2], the representability of oriented matroids is still an active area of research and can be solved for small ranks. However, to be certain of this replacement, we need to validate or refute Conjecture 3. One way to do this is to construct a method to travel up the poset $\mathbf{P_{Code}}$ towards oriented matroid codes above it. Although, for the time being, we have not been able to fully construct a method for this idea, we know how one can "climb up" from one code to any code that covers it, which will be presented in this paper. Our paper is structured as follows: some background of the poset $\mathbf{P_{Code}}$ is given in Section 2, and the reader can find our main contribution which is the method of traveling upward $\mathbf{P_{Code}}$ in Section 3. All supporting results can be found in Appendix A and Appendix B.

2. The Poset of Combinatorial Neural Codes

In the following, we provide some background of the poset $\mathbf{P_{Code}}$ of combinatorial neural codes. The reader may also read Appendix A for necessary information.

Definition 4 (Trunks in neural codes) *A **trunk** in a neural code $\mathcal{C}$ is the set $\mathrm{Tk}_{\mathcal{C}}(\sigma)$ (possibly empty) defined by*

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

for some $\sigma \subseteq [n]$.

Definition 5 (Simple trunks and proper trunks) *A trunk in a neural code $\mathcal{C}$ of a single neuron i, $\mathrm{Tk}_{\mathcal{C}}(\{i\})$, is called a **simple trunk** and simply denoted by $\mathrm{Tk}_{\mathcal{C}}(i)$. A trunk in $\mathcal{C}$ is **proper** if it is nonempty and not the trunk of $\emptyset$ (i.e., not $\mathcal{C}$).*

2. The complexity class $\exists\mathbb{R}$ is read as the *existential theory of the reals*.

An obvious property of trunks is that the intersection between any two trunks is again a trunk (Jeffs, 2020, Proposition 2.2). The purpose of the trunks in a neural code is to encode important combinatorial properties of certain groups of codewords, and the mappings which preserve these important combinatorial relations among the codewords in a neural code are called morphisms or neural codes. Intuitively, the notion of trunks in neural codes is analogous to that of open stars in simplicial complexes. Morphisms of neural codes is then in analogy to that of continuous functions between topological spaces: they are "continuous" with respect to trunks.

Definition 6 (Morphisms of neural codes) *A function $f : \mathcal{C} \to \mathcal{D}$ is a **morphism** of neural codes if the preimage $f^{-1}(T)$ of every proper trunk T in $\mathcal{D}$ is a proper trunk in $\mathcal{C}$.*

This definition uses proper trunks to define code morphisms rather than just any trunks as in (Jeffs, 2020). As for this adjustment mentioned in (Jeffs, 2021, Remark 3.1.2.), it ensures several nice properties of neural codes under morphisms. The following, which implies that the image of every neural code in the domain is a subcode of the neural code in the codomain, is an example.

Proposition 7 *If f is a morphism of neural codes, then $f(\emptyset) = \emptyset$.*

Proof See (Jeffs, 2021, Proposition 3.1.3). ■

Definition 8 *Given some proper trunks $T_1, \ldots, T_m$ in $\mathcal{C}$. Then, the function $f : \mathcal{C} \to 2^{[m]}$ given by $c \mapsto \{j \in [m] \mid c \in T_j\}$ is called the **morphism determined by** $T_1, \ldots, T_m$.*

Proposition 9 *The function described above in Definition 8 is a morphism. Moreover, every morphism arises in this way: if $\mathcal{D} \subseteq 2^{[m]}$ and $f : \mathcal{C} \to \mathcal{D}$ is any morphism, then f can be obtained by restricting to $\mathcal{D}$ the codomain of the morphism g determined by the trunks $T_j = f^{-1}(\mathrm{Tk}_{\mathcal{D}}(j))$ in $\mathcal{C}$.*

Proof See (Jeffs, 2020, Propositions 2.11 and 2.12) . ■

Remark 10 *Sometimes, it will be convenient to talk about the morphism determined by a set of trunks without indexing the trunks. We may do this when we only care about the isomorphism class of a neural code because a choice of indexing simply corresponds to a permutation of neurons in the codomain, which is an isomorphism.*

In the following, we present the poset $\mathbf{P_{Code}}$ that was first introduced in (Jeffs, 2020) and then slightly modified in (Jeffs, 2021). The formulation below has the advantage that we no longer need to consider "replacement by a trunk" as a possible covering relation, since every nonempty trunk in $\mathcal{C}$, up to adding the $\varnothing$-codeword $\emptyset$, is a minor of $\mathcal{C}$ via a surjective morphism (see Proposition 12). The formulation below allows us to maintain the previously desirable properties of $\mathbf{P_{Code}}$ while accounting for the usual neural code convention that $\emptyset$ is always a codeword.

Definition 11 (The poset $\mathbf{P_{Code}}$) *Let $\mathcal{C}$ and $\mathcal{D}$ be neural codes. We say that $\mathcal{D}$ **is a minor of** $\mathcal{C}$ (written $\mathcal{D} \leq \mathcal{C}$) if there is a surjective morphism $f : \mathcal{C} \to \mathcal{D}$. As a consequence, this relation defines a partial order on the set of distinct isomorphism classes of neural codes, denoted by $\mathbf{P_{Code}}$.*

Note that the above definition no longer allows "replace $\mathcal{C}$ by a trunk" as a possible covering relation as in (Jeffs, 2020) since we require the $\varnothing$-codeword to be in every code. However, the following proposition shows that with a small adjustment we do not lose too much when doing this.

Proposition 12 *Let T be a (possibly empty) trunk in a neural code $\mathcal{C}$. Then, $\{\emptyset\} \cup T \leq \mathcal{C}$.*

Proof Consider the function sending $T \subset \mathcal{C}$ to itself and all other codewords in $\mathcal{C}$ to $\emptyset$. This function from $\mathcal{C}$ to $\{\emptyset\} \cup T$ is obviously surjective and a morphism of neural codes because the preimage of any proper trunk $\mathrm{Tk}_T(\sigma)$ in T is equal to $\mathrm{Tk}_{\mathcal{C}}(\sigma) \cap T$ which is a proper trunk in $\mathcal{C}$. ■

3. Upward Covering Relations in $\mathbf{P_{Code}}$

In this section, we provide a method to describe all codes that cover a given neural code $\mathcal{D}$ in $\mathbf{P_{Code}}$ as opposed to Appendix A. To fully understand the development of how we construct covering codes, the reader may read Appendix B before this section. In Appendix B, we instead described $\mathcal{C}$ in terms of $\mathcal{D}$ when they both are intersection-complete. In the following, we extend the idea in Appendix B to all neural codes. We start by showing that every covering relation $\mathcal{C}$ and $\mathcal{D}$ in $\mathbf{P_{Code}}$ extends in a natural way to a covering relation between their intersection-completions. The hat notation (e.g. $\widehat{\mathcal{C}}$) will be used to denote the intersection-completion of any code (e.g. $\mathcal{C}$). This extension will be a key tool in characterizing all upward covering relations in $\mathbf{P_{Code}}$.

Lemma 13 *Let $f : \mathcal{C} \to \mathcal{D}$ be a morphism. There exists a unique morphism $g : \widehat{\mathcal{C}} \to \widehat{\mathcal{D}}$ such that the diagram*

$$\begin{array}{ccc} \mathcal{C} & \hookrightarrow & \widehat{\mathcal{C}} \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle g} \\ \mathcal{D} & \hookrightarrow & \widehat{\mathcal{D}} \end{array}$$

commutes. Moreover, if f is surjective, then so is g.

Proof First, let us prove that the morphism g exists. We know that f is determined by some collection of trunks $\{T_1, \ldots, T_m\}$ in $\mathcal{C}$. Each T_j can be naturally associated with the smallest trunk (in terms of inclusion) S_j in $\widehat{\mathcal{C}}$ that contains it by setting S_j to be the intersection of all trunks containing T_j in $\widehat{\mathcal{C}}$. The collection $\{S_1, \ldots, S_m\}$ in $\widehat{\mathcal{C}}$ determines a morphism $h : \widehat{\mathcal{C}} \to 2^{[m]}$. Observe that if $c \in \mathcal{C}$, then $f(c) = h(c)$, so h extends the morphism f to $\widehat{\mathcal{C}}$ while also extending the codomain to $2^{[m]}$. By Proposition 38 (Appendix B), the image of $\widehat{\mathcal{C}}$ under h is a subset of $\widehat{\mathcal{D}}$. Thus, we can restrict h to a morphism $g : \widehat{\mathcal{C}} \to \widehat{\mathcal{D}}$.

Moreover, Proposition 38 (Appendix B) implies that the action of g on a codeword in $\widehat{\mathcal{C}}$ is uniquely determined by its action on $\mathcal{C}$, and so the choice of g is unique.

For the surjectivity statement, suppose that f is surjective and consider an arbitrary codeword in $\widehat{\mathcal{D}}$. Since f is surjective, this codeword is an intersection of various $f(c)$. This implies that it is the intersection of various $g(c)$, and by Proposition 38 (Appendix B), this means that it is the image under g of some codeword in $\widehat{\mathcal{C}}$, so g is surjective as desired. ■

Such a diagram can help us recognize when codes and their intersection completions cover one another, as described in the lemma below.

Lemma 14 *Let $f : \mathcal{C} \to \mathcal{D}$ be a surjective morphism, and $g : \widehat{\mathcal{C}} \to \widehat{\mathcal{D}}$ be the unique surjective morphism so that the diagram*

$$\begin{array}{ccc} \mathcal{C} & \hookrightarrow & \widehat{\mathcal{C}} \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle g} \\ \mathcal{D} & \hookrightarrow & \widehat{\mathcal{D}} \end{array}$$

commutes. Then $\widehat{\mathcal{C}}$ covers $\widehat{\mathcal{D}}$ if and only if $\mathcal{C}$ covers $\mathcal{D}$.

Proof This follows from an application of Proposition 33 (Appendix B) and Corollary 31 (Appendix A). Observe that nonempty trunks in $\mathcal{C}$ are in bijection with elements of $\widehat{\mathcal{C}}$ via the map which sends a nonempty trunk to the intersection of all its elements. Thus the following statements are equivalent:

(i) $\widehat{\mathcal{C}}$ covers $\widehat{\mathcal{D}}$,

(ii) $\widehat{\mathcal{C}}$ has one more nonempty trunk than $\widehat{\mathcal{D}}$,

(iii) $\widehat{\mathcal{C}}$ has one more element than $\widehat{\mathcal{D}}$,

(iv) $\mathcal{C}$ has one more nonempty trunk than $\mathcal{D}$, and

(v) $\mathcal{C}$ covers $\mathcal{D}$.

■

Theorem 15 *Let $\mathcal{C}$ and $\mathcal{D}$ be codes and suppose that $\mathcal{C}$ covers $\mathcal{D}$. Then $\widehat{\mathcal{C}}$ is isomorphic to $\widehat{\mathcal{D}}_{[\mathcal{I}]}$ (see Definition 32, Appendix B) for some choice of isolated subset $\mathcal{I} \subseteq \widehat{\mathcal{D}}$.*

Proof If $\mathcal{C}$ covers $\mathcal{D}$, we may construct the diagram in Lemma 14, and conclude that $\widehat{\mathcal{C}}$ covers $\widehat{\mathcal{D}}$. The rest of the result follows from Theorem 40 (Appendix B). ■

Corollary 16 *Suppose that $\mathcal{C}$ covers $\mathcal{D}$. Then $\widehat{\mathcal{C}}$ covers $\widehat{\mathcal{D}}$.*

The following definitions provide an important building block of constructing covering codes and a list of all codes that can cover a given code $\mathcal{D}$, respectively. We will prove the latter is true in Theorem 21.

Definition 17 *Let $\mathcal{C}$ be an intersection-complete code, $\mathcal{I} \subseteq \mathcal{C}$ its nonempty intersection-complete subset. Then, $\mathcal{I}$ is **isolated** if no codeword in $\mathcal{C} \setminus \mathcal{I}$ contains any non-minimal codeword in $\mathcal{I}$; i.e., $\sigma \not\supset \tau$ for every $\sigma \in \mathcal{C} \setminus \mathcal{I}$ and every $\tau \in \mathcal{I} \setminus \{\mu\}$ where μ is the minimal element of $\mathcal{I}$.*

Note that any singleton set is an isolated subset, and so is any nonempty trunk in intersection-complete codes.

Definition 18 *Let $\mathcal{D}$ be any neural code, and $\mathcal{I} \subseteq \widehat{\mathcal{D}}$ an isolated subset with minimal element μ. We define four types of covering code for $\mathcal{D}$, subject to certain conditions on $\mathcal{I}$. We summarize these conditions and the resulting codes in Table 1. The notation $(\cdot)_\alpha$ denotes the action of adding a new neuron α to every codeword in the argument (e.g. $(S)_\alpha = \{c \cup \alpha \,|\, c \in S\}$).*

Type	Conditions	Construction
1	$\mu \in \mathcal{D}$	$\mathcal{D}_{[\mathcal{I}]} = (\mathcal{D} \cap \mathcal{I})_\alpha \cup \mathcal{D} \setminus \mathcal{I} \cup \{\mu\}$
2	$\mu \in \mathcal{D}$ and $\mathrm{Tk}_{\mathcal{D}}(\mu) \setminus \mathcal{I} \neq \varnothing$	$\mathcal{D}_{(\mathcal{I})} = (\mathcal{D} \cap \mathcal{I})_\alpha \cup \mathcal{D} \setminus \mathcal{I}$
3	$\mu \in \mathcal{D}$ and $\mu = \bigcap_{\sigma \in \mathcal{D} \cap \mathcal{I} \setminus \{\mu\}} \sigma$	$\mathcal{D}_{[\mathcal{I})} = (\mathcal{D} \cap \mathcal{I} \setminus \{\mu\})_\alpha \cup \mathcal{D} \setminus \mathcal{I} \cup \{\mu\}$
4	$\mu \notin \mathcal{D}$, $\mu = \bigcap_{\sigma \in \mathcal{D} \cap \mathcal{I} \setminus \{\mu\}} \sigma$, and $\mathrm{Tk}_{\mathcal{D}}(\mu) \setminus \mathcal{I} \neq \varnothing$	$\mathcal{D}_{(\mathcal{I})} = (\mathcal{D} \cap \mathcal{I})_\alpha \cup \mathcal{D} \setminus \mathcal{I}$

Table 1: Constructions of covering codes, subject to the conditions on an isolated subset.

Note that the conditions in Table 1 are not mutually exclusive. Thus for a given choice of $\mathcal{I}$ we may be able to form multiple covering codes. Besides, when $\mathcal{C}$ is intersection-complete, the notation above is compatible with Definition 32 (Appendix B). We give several results below, which eventually show that the codes above are exactly those covering $\mathcal{D}$ in $\mathbf{P_{Code}}$. Our first result below says that the constructions in Definition 18 do not differ too much: when we apply the construction and compute the intersection-completion of the resulting code, we will always obtain the covering code $\widehat{\mathcal{D}}_{[\mathcal{I}]}$ of Definition 32 (Appendix B).

Theorem 19 *Let $\mathcal{D}$ be a code, $\mathcal{I} \subseteq \widehat{\mathcal{D}}$ an isolated subset with the minimal codeword μ, and $\mathcal{C}$ one of the covering codes described in Definition 18. Then $\widehat{\mathcal{C}} = \widehat{\mathcal{D}}_{[\mathcal{I}]}$.*

Proof In every type of covering code in Definition 18, we see that the new neuron α is only added to elements of $\mathcal{D} \cap \mathcal{I}$ (possibly, except for μ). Comparing this to Definition 32 (Appendix B), we see that $\mathcal{C} \subseteq \widehat{\mathcal{D}}_{[\mathcal{I}]}$. Since $\widehat{\mathcal{D}}_{[\mathcal{I}]}$ is intersection-complete, this immediately implies that $\widehat{\mathcal{C}} \subseteq \widehat{\mathcal{D}}_{[\mathcal{I}]}$.

It remains to prove the reverse inclusion. Let $c \in \widehat{\mathcal{D}}_{[\mathcal{I}]}$. We must show that c is an element or an intersection of elements of $\mathcal{C}$. First, suppose that c contains α and is not $\mu \cup \alpha$. Then $c \setminus \alpha$ is an element of $\mathcal{I}$ and equal to an intersection of some codewords in $\mathcal{D}$. If any of the codewords whose intersection is $c \setminus \alpha$ is not an element of $\mathcal{I}$, then Definition 17 implies that $c \setminus \alpha = \mu$, a contradiction. Therefore, $c \setminus \alpha$ is an intersection of codewords in $(\mathcal{D} \cap \mathcal{I}) \setminus \{\mu\}$. In every type of covering code, we add α to all such codewords, and hence c is an intersection of some codewords in $\mathcal{C}$ as desired. Next, suppose that c does not contain

α and is not μ. Then c is an intersection of some codewords in $\mathcal{D}$, at least one of which is not an element of $\mathcal{D} \cap \mathcal{I}$. When we form $\mathcal{C}$, we add α to all codewords in $\mathcal{D} \cap \mathcal{I}$ (possibly, except μ). Since at least one of the codewords whose intersection is c is not in $\mathcal{D} \cap \mathcal{I}$, the intersection of these codewords in $\mathcal{C}$ will still be c. This case is concluded. This leaves the following two cases: $c = \mu$ and $c = \mu \cup \alpha$. We will show that both μ and $\mu \cup \alpha$ are codewords in $\widehat{\mathcal{C}}$, regardless of covering type $\mathcal{C}$ is. Each case is considered as follows.

Type 1: Here the definition stipulates that μ stays in $\mathcal{C}$. Since we add α to all elements of $\mathcal{D} \cap \mathcal{I}$, we also obtain $\mu \cup \alpha$ as a codeword in $\mathcal{C}$.

Type 2: Since we assume $\mu \in \mathcal{D}$, this implies $\mu \in \mathcal{D} \cap \mathcal{I}$, so $\mu \cup \alpha$ is a codeword in $\mathcal{C}$. Since $\mathrm{Tk}_{\mathcal{D}}(\mu) \setminus \mathcal{I}$ is nonempty, there is a codeword $c' \in \mathcal{C}$ which contains μ but not α. This leads to the fact that $c' \cap (\mu \cup \alpha) = \mu$ is a codeword in $\widehat{\mathcal{C}}$.

Type 3: The definition stipulates that we include μ as a codeword in $\mathcal{C}$. We also assume that μ is the intersection of all elements in $(\mathcal{D} \cap \mathcal{I}) \setminus \{\mu\}$, which are exactly the codewords added with α. Thus, $\mu \cup \alpha$ is the intersection of all codewords in $\mathrm{Tk}_{\mathcal{C}}(\alpha)$ and lies in $\widehat{\mathcal{C}}$.

Type 4: Here neither μ nor $\mu \cup \alpha$ are codewords in $\mathcal{C}$. However, similarly to Type 2, the condition $\mathrm{Tk}_{\mathcal{D}}(\mu) \setminus \mathcal{I}$ is nonempty will lead to the existence of μ in $\widehat{\mathcal{C}}$. Besides, similarly to Type 3, the assumption that μ is the intersection of all elements in $(\mathcal{D} \cap \mathcal{I}) \setminus \{\mu\}$ will lead to the existence of $\mu \cup \alpha$ in $\widehat{\mathcal{C}}$.

We have shown in all cases that every $c \in \widehat{\mathcal{D}}_{[\mathcal{I}]}$ is a codeword in $\widehat{\mathcal{C}}$. The result then follows. ■

Corollary 20 *Every covering code $\mathcal{C}$ described in Definition 18 covers $\mathcal{D}$ in $\mathbf{P}_{\mathbf{Code}}$.*

Proof Let $\mathcal{C}$ be one of the covering codes of Definition 18. Note that there is a natural surjective morphism $f : \mathcal{C} \to \mathcal{D}$ given by deleting the neuron α. By Lemma 13 we obtain a unique surjective morphism $g : \widehat{\mathcal{C}} \to \widehat{\mathcal{D}}$, which extends f. By Lemma 14, $\mathcal{C}$ covers $\mathcal{D}$ if and only if $\widehat{\mathcal{C}}$ covers $\widehat{\mathcal{D}}$. Theorem 19 tells us that $\widehat{\mathcal{C}}$ is equal to $\widehat{\mathcal{D}}_{[\mathcal{I}]}$, which covers $\widehat{\mathcal{D}}$ by Theorem 40 (Appendix B). Thus, $\mathcal{C}$ covers $\mathcal{D}$ as desired. ■

To end this, we provide the following theorem which shows that if one uses constructions described in Definition 18, they can "climb up" the poset $\mathbf{P}_{\mathbf{Code}}$ from a given code to all its covering codes.

Theorem 21 *Let $\mathcal{C}$ and $\mathcal{D}$ be codes. The following are equivalent:*

(i) $\mathcal{C}$ covers $\mathcal{D}$, and

(ii) $\mathcal{C}$ is isomorphic to one of the covering codes for $\mathcal{D}$ described in Definition 18.

Proof Corollary 20 tells us that (ii) implies (i), and so it remains to prove the converse. Suppose that $\mathcal{C}$ covers $\mathcal{D}$ and let $f : \mathcal{C} \to \mathcal{D}$ be a surjective morphism with unique surjective extension $g : \widehat{\mathcal{C}} \to \widehat{\mathcal{D}}$ as guaranteed by Lemma 13. Lemma 14 tells us that $\widehat{\mathcal{C}}$ covers $\widehat{\mathcal{D}}$,

and Theorem 40 (Appendix B) implies we may choose an isolated subset $\mathcal{I} \subseteq \widehat{\mathcal{D}}$ so that $\widehat{\mathcal{C}}$ is isomorphic to $\widehat{\mathcal{D}}_{[\mathcal{I}]}$. Since $\mathbf{P_{Code}}$ describes covering relations between isomorphism classes of codes and the diagram is unchanged by an isomorphism on $\widehat{\mathcal{C}}$ (which induces an isomorphism on $\mathcal{C}$), we may replace $\widehat{\mathcal{C}}$ by $\widehat{\mathcal{D}}_{[\mathcal{I}]}$. Thus we can assume without loss of generality that $\widehat{\mathcal{D}}_{[\mathcal{I}]}$ is the intersection-completion of $\mathcal{C}$. We thus have the following diagram:

$$\begin{array}{ccc} \mathcal{C} & \hookrightarrow & \widehat{\mathcal{C}} = \widehat{\mathcal{D}}_{[\mathcal{I}]} \\ {\scriptstyle f}\downarrow & & \downarrow{\scriptstyle g} \\ \mathcal{D} & \hookrightarrow & \widehat{\mathcal{D}} \end{array}$$

Let μ be the minimal element of $\mathcal{I}$. Notice that the surjective map $\widehat{\mathcal{D}}_{[\mathcal{I}]} \to \widehat{\mathcal{D}}$ is bijective except for the fact that it identifies the codewords μ and $\mu \cup \alpha$. Thus for the diagram to commute, we see that $\mathcal{C}$ contains at least the codewords of $\mathcal{D}$ that do not have a new neuron added to them when forming $\widehat{\mathcal{D}}_{[\mathcal{I}]}$, as well as the codewords obtained by adding α to codewords of $(\mathcal{D} \cap \mathcal{I}) \setminus \{\mu\}$. Note that all of these are present in the covering codes described in Definition 18.

This leaves four cases to consider, based on whether $\mathcal{C}$ contains μ and/or $\mu \cup \alpha$.

- **Case 1:** Suppose $\mathcal{C}$ contains both μ and $\mu \cup \alpha$. Then we see that $\mathcal{C}$ is equal to the construction in Type 1, and indeed $\mu \in \mathcal{D}$ which is the condition for Type 1.
- **Case 2:** Suppose $\mathcal{C}$ contains $\mu \cup \alpha$ but not μ. In this case $\mathcal{C}$ is equal to the construction in Type 2. Again we must have $\mu \in \mathcal{D}$ since this is the image of $\mu \cup \alpha$ under the surjective map $\mathcal{C} \to \mathcal{D}$ that deletes α. However, for $\widehat{\mathcal{C}}$ to be $\widehat{\mathcal{D}}_{[\mathcal{I}]}$, there must exist some codewords in $\mathcal{C}$ whose intersection is μ. Since $\mu \cup \alpha$ is a codeword of $\mathcal{C}$ and μ is not, this is equivalent to the statement that there is a codeword in $\mathcal{C}$ that properly contains μ and not α. Such a codeword in $\mathcal{C}$ must come from a codeword in $\mathrm{Tk}_{\mathcal{D}}(\mu) \setminus \mathcal{I}$, and so $\mathcal{I}$ satisfies the conditions stipulated for Type 2.
- **Case 3:** Suppose $\mathcal{C}$ contains μ but not $\mu \cup \alpha$. Again $\mu \in \mathcal{D}$, and we see that $\mathcal{C}$ is equal to the construction in Type 3. We know that $\mu \cup \alpha$ lies in $\widehat{\mathcal{C}}$, and so there must exist some codewords in $\mathrm{Tk}_{\mathcal{C}}(\alpha)$ whose intersection is $\mu \cup \alpha$. Codewords in $\mathrm{Tk}_{\mathcal{C}}(\alpha)$ are obtained by adding α to elements of $\mathcal{D} \cap \mathcal{I}$ except μ, and so the intersection of all codewords in $(\mathcal{D} \cap \mathcal{I}) \setminus \{\mu\}$ must be μ, as stipulated for Type 3.
- **Case 4:** Lastly, suppose that $\mathcal{C}$ contains neither μ nor $\mu \cup \alpha$. Then $\mathcal{C}$ is equal to the construction in Type 4, and $\mathcal{D}$ cannot contain μ because the only codewords that could map to it under the surjective map $\mathcal{C} \to \mathcal{D}$ are μ and $\mu \cup \alpha$. However, we must be able to recover both μ and $\mu \cup \alpha$ as intersections of codewords in $\mathcal{C}$. Using similar arguments to cases 2 and 3, we see that this implies μ is the intersection of codewords in $(\mathcal{D} \cap \mathcal{I}) \setminus \{\mu\}$, and also that $\mathcal{D}$ must have a codeword properly containing μ but not in $\mathcal{I}$.

Thus if $\mathcal{C}$ covers $\mathcal{D}$, then $\mathcal{C}$ is indeed isomorphic to one of the codes from Definition 18. This proves the result. ■

Acknowledgments

We are grateful to the anonymous referees for their thoughts and feedback on our paper. The authors gratefully acknowledge Zvi Rosen for providing valuable feedback during the early stages of this work. We also thank Alex Kunin for elucidating a compelling connection between morphisms of neural codes and Boolean matrix factorizations, as well as for highlighting a motivating application in connectomics. Finally, we greatly appreciate the insightful discussions with Carina Curto, which have inspired further development of this research.

References

Anders Björner, Michel Las Vergnas, Bernd Sturmfels, Neil White, and Gunter M. Ziegler. *Oriented Matroids*. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 2nd edition, 1999.

Boris Bukh and R. Amzi Jeffs. Planar convex codes are decidable. *SIAM Journal on Discrete Mathematics*, 37(2):951–963, 2023.

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(1):44–66, 2019.

Joshua Cruz, Chad Giusti, Vladimir Itskov, and Bill Kronholm. On open and closed convex codes. *Discrete & Computational Geometry*, 61:247–270, 2016.

Carina Curto. What can topology tell us about the neural code? *Bulletin of the American Mathematical Society*, 54:63–78, 2017.

Carina Curto and Vladimir Itskov. Cell groups reveal structure of stimulus space. *PLoS computational biology*, 4:e1000205, 2008.

Carina Curto and Ramón Vera. The Leray Dimension of a Convex Code. *arXiv e-prints*, art. arXiv:1612.07797, 1612.07797. 2016.

Carina Curto and Nora Youngs. Neural ring homomorphisms and maps between neural codes. 2015.

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(9):1571–1611, 2013.

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(1):222–238, 2017.

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(9):3919–3940, 2019.

Brianna Gambacini, Sam Macdonald, and Anne Shiu. Open and Closed Convexity of Sparse Neural Codes. *arXiv e-prints*, art. arXiv:1912.00963, Dec 2019.

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(4):553–571, 2018.

Chad Giusti and Vladimir Itskov. A no-go theorem for one-layer feedforward networks. *Neural Computation*, 26(11):2527–2540, 2014.

Sarah Ayman Goldrup and Kaitlyn Phillipson. Classification of open and closed convex codes on five neurons. *arXiv e-prints*, 1909.09004. 2019.

Elizabeth Gross, Kazi Obatake Nida, and Nora Youngs. Neural ideals and stimulus space visualization. *Advances in Applied Mathematics*, 95:65–95, 2016.

Sema Gunturkun, Jack Jeffries, and Jeffrey Sun. Polarization of neural rings. *Journal of Algebra and Its Applications*, 2019.

Vladimir Itskov, Alex Kunin, and Zvi Rosen. Hyperplane Neural Codes and the Polar Complex. *arXiv e-prints*, art. 1801.02304, 1801.02304. 2018.

R. Amzi Jeffs. Sunflowers of convex open sets. *Advances in Applied Mathematics*, 111: 101935, 2019.

R. Amzi Jeffs. Morphisms of neural codes. *SIAM Journal on Applied Algebra and Geometry*, 4:99–122, 2020.

R. Amzi Jeffs. Morphisms, minors, and minimal obstructions to convexity of neural codes. 2021. URL https://www.math.cmu.edu/~amzij/pdf/Amzi_Jeffs_Thesis.pdf.

R. Amzi Jeffs. Embedding dimension phenomena in intersection complete codes. *Selecta Mathematica*, 28, 2021.

R. Amzi Jeffs and Isabella Novik. Convex union representability and convex codes. *International Mathematics Research Notices*, 2021(9):7132–7158, 04 2019.

R. Amzi Jeffs, Mohamed Omar, Natchanon Suaysom, Aleina Wachtel, and Nora Youngs. Sparse neural codes and convexity. *Involve, a Journal of Mathematics*, 12(5):737–754, 2015.

R. Amzi Jeffs, Mohamed Omar, and Nora Youngs. Neural ideal preserving homomorphisms. *Journal of Pure and Applied Algebra*, 222:3470–3482, 2018.

Alex Kunin, Caitlin Lienkaemper, and Zvi Rosen. Oriented matroids and combinatorial neural codes. *Combinatorial Theory*, 3, 2020.

Caitlin Lienkaemper, Anne Shiu, and Zev Woodstock. Obstructions to convexity in neural codes. *Advances in Applied Mathematics*, 85:31–59, 2017. ISSN 0196-8858.

Raffaella Mulas and Ngoc M Tran. Minimal embedding dimensions of connected neural codes. *Algebraic Statistics*, 11(1):99–106, 2020.

John O'Keefe and Jonathan Dostrovsky. The hippocampus as a spatial map. preliminary evidence from unit activity in the freely-moving rat. *Brain Research*, pages 171–175, 1971.

Ethan Petersen, Nora Youngs, Ryan Kruse, Dane Miyata, Rebecca Garcia, and Luis David Garcia Puente. Neural Ideals in SageMath. *arXiv e-prints*, art. arXiv:1609.09602, 2016.

Zvi Rosen and Yan X. Zhang. Convex Neural Codes in Dimension 1. *arXiv e-prints*, art. arXiv:1702.06907, 2017.

Alexander Ruys de Perez, Laura Felicia Matusevich, and Anne Shiu. Neural codes and the factor complex. *arXiv e-prints*, art. arXiv:1904.03235, 1904.03235. 2019.

Martin Tancer. d-representability of simplicial complexes of fixed dimension. *Journal of Computational Geometry*, 2(1):183–188, 2011. doi: 10.20382/jocg.v2i1a9.

Appendix A. Downward covering relations in $\mathbf{P_{Code}}$: a revision

This section is a slightly adjusted version of Section 3 in (Jeffs, 2019) to account for our assumption that the $\varnothing$-codeword exists in any neural code. We will characterize the downward covering relation in $\mathbf{P_{Code}}$ in the sense that we combinatorially describe all codes covered by a given neural code $\mathcal{C}$ in $\mathbf{P_{Code}}$.

Definition 22 (Trivial and redundant neurons) *Let $\mathcal{C} \subseteq 2^{[n]}$ be a neural code. A neuron $i \in [n]$ is called **trivial** if $\mathrm{Tk}_{\mathcal{C}}(i) = \varnothing$. A neuron i is called **redundant** if there exists $\sigma \subseteq [n] \setminus \{i\}$ such that $\mathrm{Tk}_{\mathcal{C}}(i) = \mathrm{Tk}_{\mathcal{C}}(\sigma)$.*

Definition 23 (Covered neural codes) *Let $\mathcal{C} \subseteq 2^{[n]}$ and for $j \in [n]$ let $T_j = \mathrm{Tk}_{\mathcal{C}}(j)$. If $i \in [n]$ is a nontrivial neuron, then the **ith covered code** of $\mathcal{C}$ is the image of $\mathcal{C}$ under the morphism determined by the following collection of trunks: $\{T_j \mid T_j \neq T_i\} \cup \{T_j \cap T_i \mid T_j \cap T_i \neq T_i\}$. This code is denoted $\mathcal{C}^{(i)}$.*

Example 1 *Let $\mathcal{C} = \{\emptyset, 1, 12, 23, 13, 123\}$. Then the 1st covered code*

$$\mathcal{C}^{(1)} = \{\emptyset, ac, ab, bd, abcd\}$$

is the image of $\mathcal{C}$ under the morphism defined by the following relabeled trunks

$$T_a = \mathrm{Tk}_{\mathcal{C}}(2), T_b = \mathrm{Tk}_{\mathcal{C}}(3), T_c = \mathrm{Tk}_{\mathcal{C}}(12), T_d = \mathrm{Tk}_{\mathcal{C}}(13)$$

as in Definition 8. To verify, $\mathcal{C}$ has 8 trunks, namely

$$\mathrm{Tk}_{\mathcal{C}}(\emptyset), \mathrm{Tk}_{\mathcal{C}}(1), \mathrm{Tk}_{\mathcal{C}}(2), \mathrm{Tk}_{\mathcal{C}}(3), \mathrm{Tk}_{\mathcal{C}}(12), \mathrm{Tk}_{\mathcal{C}}(13), \mathrm{Tk}_{\mathcal{C}}(23), \text{ and } \mathrm{Tk}_{\mathcal{C}}(123),$$

and $\mathcal{C}^{(1)}$ has 7 trunks, namely

$$\mathrm{Tk}_{\mathcal{C}^{(1)}}(\emptyset), \mathrm{Tk}_{\mathcal{C}^{(1)}}(a), \mathrm{Tk}_{\mathcal{C}^{(1)}}(b), \mathrm{Tk}_{\mathcal{C}^{(1)}}(c), \mathrm{Tk}_{\mathcal{C}^{(1)}}(d), \mathrm{Tk}_{\mathcal{C}^{(1)}}(ab), \mathrm{Tk}_{\mathcal{C}^{(1)}}(ad)$$

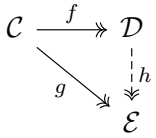
where $\mathrm{Tk}_{\mathcal{C}^{(1)}}(c) = \mathrm{Tk}_{\mathcal{C}^{(1)}}(ac)$, $\mathrm{Tk}_{\mathcal{C}^{(1)}}(d) = \mathrm{Tk}_{\mathcal{C}^{(1)}}(bd)$, and all other $\mathrm{Tk}_{\mathcal{C}^{(1)}}(\sigma)$ are equal when $\sigma \in \{ad, bc, cd, abc, abd, bcd, abcd\}$

We will prove that all codes defined in Definition 23 are exactly the ones that $\mathcal{C}$ covers, as long as the neuron i is non-trivial and non-redundant.

Definition 24 (Trunk generation) *Let $\mathcal{C}$ be a neural code and $\{T_1, \ldots, T_m\}$ a collection of trunks in $\mathcal{C}$. We say that a trunk T in $\mathcal{C}$ is **generated** by $\{T_1, \ldots, T_m\}$ if there exists $\sigma \subseteq [m]$ such that $T = \bigcap_{i \in \sigma} T_i$.*

By convention, we say that the empty intersection is all of $\mathcal{C}$. Note that every nonempty trunk is generated by the set of simple trunks. Furthermore, a neuron i is redundant if and only if $\mathrm{Tk}_{\mathcal{C}}(i)$ is generated by $\{\mathrm{Tk}_{\mathcal{C}}(j) \mid j \neq i\}$ by Definition 22 and Definition 24.

Lemma 25 *Let $\mathcal{C}, \mathcal{D}$, and $\mathcal{E}$ be neural codes, and let $f : \mathcal{C} \to \mathcal{D}$ and $g : \mathcal{C} \to \mathcal{E}$ be surjective morphisms determined by collections of trunks A and B, respectively. There exists a surjective morphism $h : \mathcal{D} \to \mathcal{E}$ such that the diagram below commutes (i.e. $g = h \circ f$) if and only if every trunk in B is generated by A.*



Proof See (Jeffs, 2019, Lemma 3.13). ■

Corollary 26 *Let $f : \mathcal{C} \to \mathcal{D}$ be a surjective morphism. Then f is an isomorphism if and only if $\mathcal{C}$ and $\mathcal{D}$ have the same number of trunks.*

Proof If f is an isomorphism, then the inverse f^{-1} yields a bijection on trunks in $\mathcal{C}$ and trunks in $\mathcal{D}$. The converse is exactly (Jeffs, 2019, Proposition 3.16). ■

Corollary 27 *Let $\mathcal{C} \subseteq 2^{[n]}$ be a code, and suppose that $i \in [n]$ is a redundant neuron. Let $\mathcal{D} = \{c \setminus \{i\} \mid c \in \mathcal{C}\}$. The map $f : \mathcal{C} \to \mathcal{D}$ given by $f(c) = c \setminus \{i\}$ is an isomorphism.*

Proof The map described is a morphism since the preimage of $\mathrm{Tk}_{\mathcal{D}}(\sigma)$ is simply $\mathrm{Tk}_{\mathcal{C}}(\sigma)$. The codes $\mathcal{C}$ and $\mathcal{D}$ have the same number of trunks since every trunk in $\mathcal{C}$ can be expressed as an intersection of simple trunks other than $\mathrm{Tk}_{\mathcal{C}}(i)$, so f is an isomorphism. ■

Proposition 28 *Let $f : \mathcal{C} \to \mathcal{D}$ be a morphism. If S and T are trunks in $\mathcal{D}$, then $f^{-1}(S \cap T) = f^{-1}(S) \cap f^{-1}(T)$.*

Proof This is true for any function $f : \mathcal{C} \to \mathcal{D}$. ■

Lemma 29 *Let $\mathcal{C} \subseteq [n]$ and let $f : \mathcal{C} \to \mathcal{D}$ be a surjective morphism, and suppose that f is not an isomorphism. Then there exists a neuron $i \in [n]$ so that $\mathcal{D} \leq \mathcal{C}^{(i)}$.*

Proof By (Jeffs, 2019, Proposition 3.15), the map f^{-1} is an injection from trunks in $\mathcal{D}$ to those in $\mathcal{C}$. Corollary 26 implies that $\mathcal{C}$ has more trunks than $\mathcal{D}$, and so there must be some trunk in $\mathcal{C}$ that is not the preimage of a trunk in $\mathcal{D}$. In fact there must be a simple trunk in $\mathcal{C}$ that is not the preimage of a trunk in $\mathcal{D}$ (Proposition 28 implies that the set of trunks in $\mathcal{C}$ which are preimages of trunks in $\mathcal{D}$ is closed under intersection, and every nonempty trunk is an intersection of simple trunks in $\mathcal{C}$). Let $i \in [n]$ be a neuron so that $\mathrm{Tk}_{\mathcal{C}}(i)$ is not the preimage of a trunk in $\mathcal{D}$.

Recall that the surjective morphism $\mathcal{C} \to \mathcal{C}^{(i)}$ is determined by a set of trunks which generate all nonempty trunks in $\mathcal{C}$ except for possibly $\mathrm{Tk}_{\mathcal{C}}(i)$. Thus by Lemma 25 there exists a surjective map $g : \mathcal{C}^{(i)} \to \mathcal{D}$ whose composition with $\mathcal{C} \to \mathcal{C}^{(i)}$ is equal to f. This shows that $\mathcal{D} \leq \mathcal{C}^{(i)}$ as desired. ■

Theorem 30 *Let $\mathcal{C} \subseteq 2^{[n]}$ and $\mathcal{D} \subseteq 2^{[m]}$ be codes. Then $\mathcal{C}$ covers $\mathcal{D}$ in $\mathbf{P_{Code}}$ if and only if $\mathcal{D} \cong \mathcal{C}^{(i)}$ for some non-redundant, nontrivial neuron $i \in [n]$.*

Proof By Corollary 27 we can reduce to the case where $\mathcal{C}$ has no redundant neurons. Then suppose that $\mathcal{C}$ covers $\mathcal{D}$, so there exists a surjective morphism $f : \mathcal{C} \to \mathcal{D}$ that is not an isomorphism. Lemma 29 implies that for some $i \in [n]$ we have $\mathcal{D} \leq \mathcal{C}^{(i)}$. But $\mathcal{C}^{(i)} < \mathcal{C}$ since i is not redundant, and since $\mathcal{C}$ covers $\mathcal{D}$ we must have that $\mathcal{D} = \mathcal{C}^{(i)}$ in $\mathbf{P_{Code}}$ so they are isomorphic.

For the converse, we must argue that $\mathcal{C}$ covers $\mathcal{C}^{(i)}$. The two codes are not isomorphic since $\mathcal{C}^{(i)}$ has exactly one less trunk than $\mathcal{C}$. Any code $\mathcal{D}$ such that $\mathcal{C}^{(i)} \leq \mathcal{D} \leq \mathcal{C}$, either has the same number of trunks as $\mathcal{C}$ or as $\mathcal{C}^{(i)}$, and is isomorphic to one of them by Corollary 26, so $\mathcal{C}$ covers $\mathcal{C}^{(i)}$ as desired. ■

Corollary 31 (Covering criterion) *Let $f : \mathcal{C} \to \mathcal{D}$ be a surjective morphism. Then $\mathcal{C}$ covers $\mathcal{D}$ if and only if $\mathcal{C}$ has exactly one more trunk than $\mathcal{D}$.*

Proof Suppose that $\mathcal{C}$ covers $\mathcal{D}$. By Theorem 30, $\mathcal{D} \cong \mathcal{C}^{(i)}$ for some non-redundant $i \in [n]$. But $\mathcal{C}^{(i)}$ has exactly one less trunk than $\mathcal{C}$, since the surjective morphism $\mathcal{C} \to \mathcal{C}^{(i)}$ is determined by a set of trunks which generates all trunks in $\mathcal{C}$ except for $\mathrm{Tk}_{\mathcal{C}}(i)$.

For the converse, suppose that $\mathcal{C}$ has exactly one more trunk than $\mathcal{D}$. For any code $\mathcal{E}$ with $\mathcal{C} \geq \mathcal{E} \geq \mathcal{D}$, we see that $\mathcal{E}$ has as many trunks as either $\mathcal{C}$ or $\mathcal{D}$ depending on which inequality is strict, and Corollary 26 implies that $\mathcal{E}$ is isomorphic to one of $\mathcal{C}$ or $\mathcal{D}$. Thus $\mathcal{C}$ covers $\mathcal{D}$ as desired. ■

Appendix B. Intersection-complete neural codes

This section is supplementary to Section 3. We will focus only on intersection-complete codes. It turns out that describing all codes that cover a given neural code $\mathcal{C}$ in $\mathbf{P_{Code}}$ is a simpler task if we assume that $\mathcal{C}$ is intersection-complete and restrict our attention to the intersection-complete codes that cover it. We will use these results in Section 3 to describe the general case. Isolated subsets defined in Definition 17 are the key objects that we will use to describe the covering relation in $\mathbf{P_{Code}}$ for intersection-complete codes. Our main construction is as follows.

Definition 32 *Let $\mathcal{C}$ be any intersection-complete code and let $\mathcal{I} \subseteq \mathcal{C}$ be an isolated subset with minimal element μ. Define*

$$\mathcal{C}_{[\mathcal{I}]} = \{\mu\} \cup \mathcal{C} \setminus \mathcal{I} \cup (\mathcal{I})_\alpha \tag{1}$$

where $(\cdot)_\alpha$ is defined the same as in Definition 18. In other words, we obtain $\mathcal{C}_{[\mathcal{I}]}$ by adding a new neuron α to every codeword in $\mathcal{I}$ while also maintaining μ as a codeword.

When $\mathcal{I} = \{\mu\}$, the code $\mathcal{C}_{[\mathcal{I}]}$ is the result of adding $(\mu)_\alpha$ as a new codeword. Furthermore, we can see that $\mathcal{C} \leq \mathcal{C}_{[\mathcal{I}]}$, since there is a surjective morphism from $\mathcal{C}_{[\mathcal{I}]}$ to $\mathcal{C}$ given by the deletion of the neuron α. Our goal is to prove that the construction in Definition 32 is the only way to construct intersection-complete covering codes of a given intersection-complete code. We start with some supporting propositions and lemmas.

Proposition 33 *A code $\mathcal{C}$ is intersection-complete if and only if the assignment $\sigma \mapsto \mathrm{Tk}_{\mathcal{C}}(\sigma)$ defines a bijection between $\mathcal{C}$ and its nonempty trunks.*

Proof Assume the map $\sigma \mapsto \mathrm{Tk}_{\mathcal{C}}(\sigma)$ is a bijection between $\mathcal{C}$ and its nonempty trunks. Let $\sigma_1, \sigma_2 \in \mathcal{C}$, and T an inclusion-minimal trunk that contains both codewords. Then there is σ such that $T = \mathrm{Tk}_{\mathcal{C}}(\sigma)$. Clearly, $c \subseteq \sigma_1 \cap \sigma_2$ since σ is contained in both σ_1 and σ_2. If σ were a proper subset of $\sigma_1 \cap \sigma_2$, then $\mathrm{Tk}_{\mathcal{C}}(\sigma_1 \cap \sigma_2)$ would be properly contained in T, which is a contradiction to the choice of T. Thus, $\sigma = \sigma_1 \cap \sigma_2$.

Conversely, suppose $\mathcal{C}$ is intersection-complete. The assignment $\sigma \mapsto \mathrm{Tk}_{\mathcal{C}}(\sigma)$ clearly defines an injection because σ is the unique minimal element of its trunk in $\mathcal{C}$. The map is a surjection because if T is a nonempty trunk then the intersection of all elements of T must be a codeword σ such that $T = \mathrm{Tk}_{\mathcal{C}}(\sigma)$. ■

This proposition gives us a quick way to compare the numbers of trunks between any two intersection-complete codes based on the numbers of their codewords.

Proposition 34 *Let $\mathcal{C}$ be an intersection-complete code and $\mathcal{I} \subseteq \mathcal{C}$ an isolated subset of $\mathcal{C}$ with minimal element μ. Then, $\{\mu\} \cup (\mathcal{C} \setminus \mathcal{I})$ is intersection-complete.*

Proof Definition 17 implies that the intersection of codewords in $\mathcal{C} \setminus \mathcal{I}$ is an element of $\mathcal{C} \setminus \mathcal{I}$ or equal to μ. Intersecting an element of $\mathcal{C} \setminus \mathcal{I}$ with μ itself yields either μ or some codeword in $\mathcal{C}$ that is below μ which is definitely not in $\mathcal{I}$. Thus $\{\mu\} \cup (\mathcal{C} \setminus \mathcal{I})$ is intersection-complete. ■

Proposition 35 *Let $\mathcal{C}$ be an intersection-complete code and $\mathcal{I} \subseteq \mathcal{C}$ an isolated subset. The code $\mathcal{C}_{[\mathcal{I}]}$ of Definition 32 is intersection-complete.*

Proof Let μ be the minimal element of $\mathcal{I}$. The codewords in $\mathcal{C}_{[\mathcal{I}]}$ come in two types: those in $\{\mu\} \cup (\mathcal{C} \setminus \mathcal{I})$ and those of the form $c \cup \alpha$ where $c \in \mathcal{I}$. For any two codewords $c_1, c_2 \in \mathcal{C}_{[\mathcal{I}]}$, there are 3 cases to consider.

Case 1: If both c_1 and c_2 lie in $\{\mu\} \cup (\mathcal{C} \setminus \mathcal{I})$, then so does $c_1 \cap c_2$ by Proposition 34. This guarantees $c_1 \cap c_2 \in \mathcal{C}_{[\mathcal{I}]}$ in this case.

Case 2: If $c_1 = a_1 \cup \alpha$ and $c_2 = a_2 \cup \alpha$ for some $a_1, a_2 \in \mathcal{I}$, then $c_1 \cap c_2 = (a_1 \cap a_2) \cup \{\alpha\}$. Since $\mathcal{I}$ is intersection-complete, $a_1 \cap a_2 \in \mathcal{I}$. This means $c_1 \cap c_2$ is in $(\mathcal{I})_\alpha$ and a codeword of $\mathcal{C}_{[\mathcal{I}]}$.

Case 3: Without loss of generality, suppose that c_1 contains α and c_2 does not. In this case $c_1 = a_1 \cup \alpha$ for some $a_1 \in \mathcal{I}$, and $c_2 \in \mathcal{C} \setminus \mathcal{I}$. We can immediately see that $c_1 \cap c_2 = a_1 \cap c_2$. By Definition 17, we know that $a_1 \cap c_2$ is in $\{\mu\} \cup (\mathcal{C} \setminus \mathcal{I})$ and hence a codeword of $\mathcal{C}_{[\mathcal{I}]}$.

In all cases $c_1 \cap c_2$ lies in $\mathcal{C}_{[\mathcal{I}]}$, and thus this code is intersection-complete as desired. ■

Proposition 36 *Let $\mathcal{C}$ be an intersection-complete code and $\mathcal{I} \subseteq \mathcal{C}$ an isolated subset. Then $\mathcal{C}_{[\mathcal{I}]}$ covers $\mathcal{C}$ in $\mathbf{P}_{\mathbf{Code}}$.*

Proof There is a natural surjective morphism $\mathcal{C}_{[\mathcal{I}]} \to \mathcal{C}$ given by deleting the neuron α. By Corollary 31, it suffices to prove that $\mathcal{C}_{[\mathcal{I}]}$ has exactly one more nonempty trunk than $\mathcal{C}$. Since $\mathcal{C}$ is intersection-complete, $\mathcal{C}_{[\mathcal{I}]}$ is intersection-complete by Proposition 35. Note that nonempty trunks in an intersection-complete code are in bijection with its codewords. Definition 32 shows that $\mathcal{C}_{[\mathcal{I}]}$ has one more codeword than $\mathcal{C}$, so the result follows. ■

Lemma 37 *Let $\mathcal{C} \subseteq 2^{[n]}$ be an intersection-complete code, and $i \in [n]$ a non-redundant, nontrivial neuron. Let μ be the minimal element of $\mathrm{Tk}_{\mathcal{C}}(i)$, and let $f : \mathcal{C} \to \mathcal{C}^{(i)}$ be the natural surjective morphism. Then the following are true:*

(i) $\mu \setminus \{i\}$ is also a codeword in $\mathcal{C}$, and

(ii) $f(\mu) = f(\mu \setminus \{i\})$.

Proof For (i), suppose for contradiction that $\mu \setminus \{i\}$ was not a codeword of $\mathcal{C}$. Then the intersection of all codewords in $\mathrm{Tk}_{\mathcal{C}}(\mu \setminus \{i\})$ must properly contain $\mu \setminus \{i\}$. If this intersection is not μ, then intersecting it with μ will yield $\mu \setminus \{i\}$ as a codeword because $\mathcal{C}$ is intersection-complete. Otherwise, the minimal element of $\mathrm{Tk}_{\mathcal{C}}(\mu \setminus \{i\})$ is μ, which implies that $\mathrm{Tk}_{\mathcal{C}}(i) = \mathrm{Tk}_{\mathcal{C}}(\mu \setminus \{i\})$. This also contradicts the assumption that i is not a redundant neuron.

For (ii), recall from Definition 23 that f is determined by the collection of trunks $\{T_j \mid T_j \neq T_i\} \cup \{T_j \cap T_i \mid T_j \cap T_i \neq T_i\}$ where $T_j = \mathrm{Tk}_{\mathcal{C}}(j)$. Since μ is the unique minimal element of T_i, trunks of the form $T_j \cap T_i$ above do not contain it (otherwise they would be the same as T_i). Thus $f(\mu)$ records which T_j contains μ, except for T_i. In other words, $f(\mu)$ records the neurons in μ other than neuron i. These are exactly the neurons that are in $\mu \setminus \{i\}$, so $f(\mu) = f(\mu \setminus \{i\})$. ■

Proposition 38 *Let $f : \mathcal{C} \to \mathcal{D}$ be a morphism, and let $c_1, c_2 \in \mathcal{C}$ be such that $c_1 \cap c_2 \in \mathcal{C}$. Then $f(c_1 \cap c_2) = f(c_1) \cap f(c_2)$.*

Proof We know that f is determined by some collection of trunks $\{T_1, \ldots, T_m\}$ in $\mathcal{C}$. But this implies that

$$f(c_1 \cap c_2) = \{j \in [m] \mid c_1 \cap c_2 \in T_j\} = \{j \in [m] \mid c_1 \in T_j \text{ and } c_2 \in T_j\} = f(c_1) \cap f(c_2).$$

■

Lemma 39 *Let $\mathcal{C} \subseteq 2^{[n]}$ be an intersection-complete code, and $i \in [n]$ be a non-redundant, nontrivial neuron. Let $f : \mathcal{C} \to \mathcal{C}^{(i)}$ be the natural surjective morphism. Then $\mathcal{C}^{(i)}$ is intersection-complete, and $\mathcal{I} = f(\mathrm{Tk}_{\mathcal{C}}(i))$ is an isolated subset in $\mathcal{C}^{(i)}$.*

Proof It is known that the image of an intersection-complete code is again intersection-complete (see (Jeffs, 2020, Theorem 1.5)), and so $\mathcal{C}^{(i)}$ is intersection-complete. The set $\mathrm{Tk}_{\mathcal{C}}(i)$ is certainly intersection-complete, and since $\mathcal{I}$ is the image of this set under f, it follows that $\mathcal{I}$ is intersection-complete. To see that $\mathcal{I}$ is isolated in $\mathcal{C}^{(i)}$, it remains to show that it satisfies Definition 17.

Let $c_1 \in \mathcal{C}^{(i)} \setminus \mathcal{I}$ and $c_2 \in \mathcal{I} \setminus \{\mu'\}$ where μ' is the minimal element of $\mathcal{I}$. Note that $\mu' = f(\mu) = f(\mu \setminus \{i\})$ where μ is the minimal element of $\mathrm{Tk}_{\mathcal{C}}(i)$ by part (ii) of Lemma 37. Suppose for contradiction that $c_1 \supset c_2$, which implies $c_1 \cap c_2 \in \mathcal{I} \setminus \{\mu'\}$. Let $a_1, a_2 \in \mathcal{C}$ such that $f(a_1) = c_1$ and $f(a_2) = c_2$. By Proposition 38, $f(a_1 \cap a_2) = c_1 \cap c_2$. Because $c_1 \notin \mathcal{I}$, a_1 is not in $\mathrm{Tk}_{\mathcal{C}}(i)$, and neither is $a_1 \cap a_2$. Again, part (ii) of Lemma 37 implies that f is bijective except that it identifies μ and $\mu \setminus \{i\}$. Since $a_1 \cap a_2$ is not in of $\mathrm{Tk}_{\mathcal{C}}(i)$, this implies that if $c_1 \cap c_2 = f(a_1 \cap a_2)$ is an element of $\mathcal{I} = f(\mathrm{Tk}_{\mathcal{C}}(i))$, then the only case that can happen is $a_1 \cap a_2 = \mu \setminus \{i\}$. Then $c_1 \cap c_2$ must be μ', which is a contradiction. ■

We are now ready to prove the following theorem.

Theorem 40 *Let $\mathcal{C} \subseteq 2^{[n]}$ and $\mathcal{D} \subseteq 2^{[m]}$ be intersection-complete codes. The following are equivalent:*

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

(ii) $\mathcal{C} \cong \mathcal{D}_{[\mathcal{I}]}$ where $\mathcal{I} \subseteq \mathcal{D}$ is some isolated subset.

Proof The fact that (ii) implies (i) is proven in Proposition 36. For the converse, suppose that $\mathcal{C}$ covers $\mathcal{D}$. From Theorem 30, we know that $\mathcal{D} \cong \mathcal{C}^{(i)}$ for some non-redundant, nontrivial neuron i. Since the choice of isolated subset $\mathcal{I} \subseteq \mathcal{D}$ is invariant up to isomorphism, it suffices to prove the result when $\mathcal{D} = \mathcal{C}^{(i)} \subseteq 2^{[m]}$.

Let $f : \mathcal{C} \to \mathcal{C}^{(i)}$ be the natural surjective morphism, $T_j = \mathrm{Tk}_{\mathcal{C}}(j)$ for $j \in [n]$, and $\mathcal{I} = f(T_i)$. By Lemma 39, $\mathcal{I}$ is an isolated subset of $\mathcal{C}^{(i)}$ and so we may define $\mathcal{E} = (\mathcal{C}^{(i)})_{[\mathcal{I}]} \subseteq 2^{[m+1]}$ with the new neuron $\alpha = m+1$. Let $g : \mathcal{E} \to \mathcal{C}^{(i)}$ be the natural surjective morphism defined by deleting the new neuron $m+1$. Lastly, let μ' be the minimal element of $\mathcal{I}$ in $\mathcal{C}^{(i)}$, and μ the minimal element of T_i in $\mathcal{C}$. We will prove that $\mathcal{E} \cong \mathcal{C}$ by constructing $h : \mathcal{C} \to \mathcal{E}$ as follows:

$$h(c) = \begin{cases} f(c) & c \notin T_i, \\ f(c) \cup \{m+1\} & c \in T_i. \end{cases}$$

Observe that $g(h(c)) = f(c)$ for all c. Moreover, observe that h is bijective since $\mathcal{E}$ is constructed from $\mathcal{C}^{(i)}$ by adding $m+1$ to all codewords in $f(T_i)$ while maintaining μ' as a codeword, and $\mu' = f(\mu \setminus \{i\})$ by (ii) of Lemma 37.

$$\begin{array}{ccc} \mathcal{C} & \xrightarrow{h} & \mathcal{E} = (\mathcal{C}^{(i)})_{[\mathcal{I}]} \\ & \searrow^{f} & \downarrow g \\ & & \mathcal{C}^{(i)} \end{array}$$

To prove that h is a morphism, observe that it is the morphism determined by the collection of trunks $\{T_j \mid T_j \neq T_i\} \cup \{T_j \cap T_i \mid T_j \cap T_i \neq T_i\}$ together with a new trunk $\mathrm{Tk}_{\mathcal{C}}(i)$ corresponding to the new neuron $m+1$. Since h is a bijective morphism between two intersection-complete codes $\mathcal{C}$ and $\mathcal{E}$, they must have the same number of trunks and hence be isomorphic. ■