La Matematica
https://doi.org/10.1007/s44007-024-00105-1

ORIGINAL RESEARCH ARTICLE

Canonical Forms of Neural Ideals

Hugh Geller[1] · **Rebecca R.G.**[2]

Received: 23 May 2023 / Revised: 28 November 2023 / Accepted: 5 April 2024

Abstract

Neural ideals, originally defined in Curto et al., give a way of translating information about the firing pattern of a set of neurons into a pseudomonomial ideal in a polynomial ring. We give a simple criterion for determining whether a neural ideal is in canonical form, along with an improved algorithm for computing the canonical form of a neural ideal.

Keywords Neural ideals · Stanley–Reisner rings · Primary decomposition · Receptive field structure

1 Introduction

Neural rings were first introduced in [4] as an algebraic tool to study *receptive field codes* (RF codes). In certain regions of the brain, such as the hippocampus, neurons have receptive fields, which are the locations where they fire in response to stimuli. These receptive fields are typically convex subsets of the stimulus space, and the collection of receptive fields covers the stimulus space. The receptive field code consists of a binary codeword for each distinct intersection of receptive fields that appears in the stimulus space. Typically, we consider stimulus spaces as subsets of a subspace X of $\mathbb{R}^d$. One major question in the field is how to identify whether a given binary code can arise from the intersection patterns of some collection of convex sets in some Euclidean space [1–3, 5, 8–10].

The neural ring serves as a bridge allowing for algebraic methods to be applied to neuroscience and coding theory. Given n neurons, the first step of the construction

✉ Rebecca R.G.
rrebhuhn@gmu.edu

Hugh Geller
hugh.geller@mail.wvu.edu

1 School of Mathematical and Data Sciences, West Virginia University, Morgantown, WV 26505, USA

2 Department of Mathematical Sciences, George Mason University, Fairfax, VA 22030, USA

Published online: 08 May 2024

translates neural firing patterns to a binary code $\mathcal{C} \subseteq \{0, 1\}^n$ where each possible combination of neuron firing is translated into a codeword of length n. For a given codeword $c = (c_1, \ldots, c_n)$, the entry $c_i = 1$ means the ith neuron is firing whereas $c_i = 0$ means the ith neuron is not firing. Given a neural code $\mathcal{C}$, the neural ring is $R/I_{\mathcal{C}}$, where $R = \mathbb{F}_2[x_1, \ldots, x_n]$ and $I_{\mathcal{C}}$ is the vanishing ideal of $\mathcal{C}$. Using a vanishing ideal enables us to apply tools from algebraic geometry to neural codes. However, this ideal includes the trivial Boolean relations $\{x_i(1 - x_i) : 1 \leq i \leq n\}$; these hold for every neural code and so are not useful in distinguishing neural codes from each other. Instead, [4] defines the *neural ideal* $J_{\mathcal{C}} \subseteq I_{\mathcal{C}} \subseteq R$ formed from the non-trivial relations of the neural code. The two ideals relate through the equation

$$I_{\mathcal{C}} = J_{\mathcal{C}} + \langle x_i(1 - x_i) : 1 \leq i \leq n \rangle$$

(cf. [4, Lemma 3.2]). As such, one can study the neural ideal in place of the vanishing ideal.

The information encoded in the neural ideal is made more accessible by presenting the ideal in *canonical form* (cf. [4, Sect. 4.3]). The canonical form of a neural ideal can be used to identify various obstructions to convexity [3]. As such it is useful to know whether or not a neural ideal is being presented in canonical form.

One downside to the neural ideal from an algebraic perspective is that it is generated by *pseudomonomials*, products $\Pi_{i\in\sigma} x_i \Pi_{i\in\tau}(1-x_i)$, which are neither graded nor local and as a result are not as well studied in commutative algebra. To deal with this issue, Güntürkün, Jeffries, and Sun give a process for *polarizing* a neural ideal, turning it into a squarefree monomial ideal in the extension ring $\mathbb{F}_2[x_1, \ldots, x_n, y_1, \ldots, y_n]$ [7]. In our work, we utilize the polarization technique from [7] to apply monomial ideal results to classify when certain (polarized) neural ideals are presented in canonical form. As a result, our main theorem below establishes sufficient conditions on the generators of a polarized neural ideal to determine when it is in canonical form. Here, saying that two monomials g and h "share an index" means that there is some $1 \leq i \leq n$ such that $x_i \mid g$ and $y_i \mid h$ or vice versa.

Theorem 1.1 (Theorem 5.5) *Let* $\mathfrak{a} = (g_1, \ldots, g_k)$ *be a polarized neural ideal such that* $g_{j_1} \nmid g_{j_2}$ *for any* $1 \leq j_1 \neq j_2 \leq k$*, and* $x_i y_i \nmid g_j$ *for any* $1 \leq i \leq n$ *and* $1 \leq j \leq k$*. If for some pair* g_{j_1}, g_{j_2} *of generators of* $\mathfrak{a}$*,* g_{j_1} *and* g_{j_2} *share exactly 1 index i and no other generator of* $\mathfrak{a}$ *divides* $\frac{lcm\{g_{j_1}, g_{j_2}\}}{x_i y_i}$*, then* $\mathfrak{a}$ *is not in canonical form. Otherwise,* $\mathfrak{a}$ *is in canonical form.*

This result will be useful to researchers trying to prove results about all neural ideals, as they can restrict to the set of canonical forms described by this theorem.

As a consequence of this theorem, we are able to give a shortened algorithm for computing the canonical form of a neural ideal (Algorithm 5.10). This algorithm removes the need to compute the primary decomposition of a neural ideal, which simplifies the process significantly, and appears to speed up the computation in initial tests. It is distinct from the algorithm given in [12], which starts with a binary code and produces a neural ideal in canonical form, whereas ours starts with a neural ideal. As a result, we expect our algorithm to be more useful to mathematicians proving results about families of neural ideals, for example classification results, that can then be applied to neural codes.

In Sect. 6, we prove that it is possible to compute the almost canonical form of a family of neural ideals all at once, by passing to a *generic canonical form*. We anticipate that algebraic results about the generic canonical form should carry nicely to every member of the corresponding family. This will make generic canonical forms a useful tool for classifying neural ideals by their algebraic properties.

The structure of our paper is as follows. Section 2 covers the necessary background information from [4] and [7]. We also recall some useful observations about squarefree monomial ideals.

Section 3 is dedicated to translating the algorithm from [4] for computing the canonical form of a neural ideal into an algorithm that can be applied to polarized neural ideals (Algorithm 3.2). In Sect. 4, we identify common patterns in the algorithm based off of the generators of our ideal, giving us a number of shortcuts through Algorithm 3.2.

Along with containing our main theorem, Sect. 5 includes a complete classification of the canonical forms of all two-generated neural ideals and a simplified algorithm for computing the canonical form of a neural ideal (Algorithm 5.10). In Sect. 6, we discuss a *generic* canonical form, a tool that allows us to use the canonical form of a single neural ideal to compute the canonical forms of a number of related neural ideals.

2 Background on Neural Ideals and Polarization

Throughout the paper, we make use of the following notation.

Notation 2.1 (1) R will denote the ring $\mathbb{F}_2[x_1, \ldots, x_n]$.
(2) A *pseudomonomial* is a product $\prod_{i\in\sigma} x_i \prod_{i\in\tau}(1 - x_i)$ in R where $\sigma, \tau \subseteq \{1, 2, \ldots, n\}$ and $\sigma \cap \tau = \varnothing$.

Definition 2.2 To get the neural ideal, we work with a space $X \subseteq \mathbb{R}^d$ and open sets $U_1, \ldots, U_n \subseteq X$ where neuron i fires at the points in U_i. The *neural ideal*, as introduced in [4], is an ideal of R that captures the *RF-structure* of a neural code, specifically the relations

$$\bigcap_{i\in\sigma} U_i \subseteq \bigcup_{i\in\tau} U_i, \tag{1}$$

where $\sigma, \tau \subseteq \{1, \ldots, n\}$ and $\sigma \cap \tau = \varnothing$. More precisely, the neural ideal is generated by the pseudomonomials $\prod_{i\in\sigma} x_i \prod_{i\in\tau}(1 - x_i)$ corresponding to the relations above [4, Sect. 4.2].

Note that the Boolean relations $U_i \subseteq U_i$ always hold, but they do not distinguish between distinct neural codes, so they are not included in the neural ideal. Hence we assume that no generator of a neural ideal is divisible by $x_i(1 - x_i)$ for any $1 \le i \le n$.

Definition 2.3 The *canonical form* of the neural ideal captures the minimal relations in the RF-structure of the neural code, in the sense that in Eq. 1 from Definition 2.2

the intersection on the left and the union on the right are irredundant (i.e., if we removed any U_i from either side of the inclusion, the inclusion would no longer hold). The canonical form consists of the pseudomonomials corresponding to the minimal relations and is a generating set for the neural ideal [4, Sect. 4.3].

One downside to these ideals is that they are generated by pseudomonomials, and as a result are neither graded nor local. This is an issue when applying techniques from commutative and homological algebra as most of our tools require rings to be local or graded, for example computing a *minimal free resolution* and corresponding *Betti numbers*. In order to resolve this issue, Güntürkün, Jeffries, and Sun [7] developed a method to pass from pseudomonomial ideals in R to squarefree monomial ideals in $S = \mathbb{F}_2[x_1, \ldots, x_n, y_1, \ldots, y_n]$ by *polarization*.

Definition 2.4 ([7])

(1) S will denote the extension ring $\mathbb{F}_2[x_1, \ldots, x_n, y_1, \ldots, y_n]$ of R.
(2) For a monomial m we denote its largest square free divisor as $[m]$. For example, $[x^3y^2] = xy$. One can quickly check that if m_1 and m_2 are squarefree monomials, then $[m_1m_2] = \operatorname{lcm}\{m_1, m_2\}$.
(3) We work with neural ideals, which are generated by pseudomonomials.
(4) We will refer to a *polarization* function $\mathcal{P}$ (not a homomorphism) sending pseudomonomials in R to squarefree monomials in S via

$$\prod_{i\in\sigma} x_i \prod_{i\in\tau}(1-x_i) \mapsto \prod_{i\in\sigma} x_i \prod_{i\in\tau} y_i.$$

(5) If $\mathcal{A}$ is an ideal of R, then $\mathcal{P}(\mathcal{A})$ is the ideal of S generated by $\mathcal{P}(f)$ for all pseudomonomials $f \in \mathcal{A}$.
(6) We will refer to a *depolarization* map $d : S \to R$, which is the ring homomorphism sending $x_i \mapsto x_i$, $y_i \mapsto 1 - x_i$. The map d induces an isomorphism of S-modules $R \cong S/\left(\sum_{i=1}^{n}(x_i + y_i - 1)\right)$.

We will often work with the polarization of a neural ideal, while explaining how this translates to the depolarized version.

Theorem 2.5 ([7, Theorem 3.2]) *Let $\mathcal{A} = (g_1, \ldots, g_k) \subseteq R$ be a neural ideal in canonical form. Then $\mathcal{P}(\mathcal{A}) = (\mathcal{P}(g_1), \ldots, \mathcal{P}(g_k))$.*

However, this frequently does not hold when the ideal is not in canonical form.

Example 2.6 For example, $(x_1, x_2(1-x_1))$ contains x_2, so $\mathcal{P}(x_1, x_2(1-x_1))$ contains x_2. However,

$$x_2 \notin (\mathcal{P}(x_1), \mathcal{P}(x_2(1-x_1))) = (x_1, x_2y_1).$$

The canonical form of $(x_1, x_2(1-x_1))$ turns out to be (x_1, x_2).

Definition 2.7 ([4]) A *pseudomonomial prime* of R is an ideal p generated by a subset of $\{x_1, \ldots, x_n, 1-x_1, \ldots, 1-x_n\}$ such that for each $1 \leq i \leq n$, p does not contain both x_i and $1-x_i$.

Theorem 2.8 ([4, Theorem 5.4]) *Let $\mathcal{A}$ be a pseudomonomial ideal in $\mathbb{F}_2[x_1, \ldots, x_n]$. Then the primary decomposition of $\mathcal{A}$ consists of pseudomonomial primes, and so $\mathcal{A}$ has no embedded primes. Hence its primary decomposition is unique.*

Theorem 2.9 ([11, Sect. 4.5]) *Let $\mathfrak{a}$ be a squarefree monomial ideal in a polynomial ring $k[x_1, \ldots, x_n]$. The primary decomposition of $\mathfrak{a}$ consists of primes generated by a subset of $\{x_1, \ldots, x_n\}$ (monomial primes), and so $\mathfrak{a}$ has no embedded primes. Hence its primary decomposition is unique.*

Remark 2.10 Let $g_1, \ldots, g_k$ be squarefree monomials in a polynomial ring $k[x_1, \ldots, x_n]$. If $g_j = hh'$ for any squarefree monomials h, h' in the ring, then

$$(g_1, \ldots, g_j, \ldots, g_k) = (g_1, \ldots, h, \ldots, g_k) \cap (g_1, \ldots, h', \ldots, g_k).$$

In order to compute the primary decomposition of $(g_1, \ldots, g_k)$, we may use this rule repeatedly until we get an intersection of monomial primes.

3 Computing the Canonical Form of a Polarized Neural Ideal

In this section we give a polarized version of the algorithm introduced in [4] for computing the canonical form of a neural ideal and prove that it agrees with the original algorithm in [4, Sect. 4.5].

We recall the original algorithm for computing the canonical form of a neural ideal below.

Algorithm 3.1 ([4, Sect. 4.5])

(1) Start with a neural ideal $\mathcal{A} = (g_1, \ldots, g_k)$ in R.
(2) Compute the primary decomposition of $\mathcal{A}$. By Theorem 2.8, the ideals $p_1, \ldots, p_s$ in the primary decomposition will all be generated by a subset of $\{x_1, \ldots, x_n, 1 - x_1, \ldots, 1 - x_n\}$.
(3) Compute the set of products $h_1 \cdots h_s$ where h_ℓ is a generator of p_ℓ.
(4) Set $x_i(1 - x_i) = 0$ for each $1 \le i \le n$, and as a result remove any product divisible by $x_i(1 - x_i)$ and replace any power x_i^t by x_i and $(1 - x_i)^t$ by $1 - x_i$. All remaining elements are now pseudomonomials.
(5) Remove any product that is a multiple of a product of lower degree.

The remaining products give the canonical form of $\mathcal{A}$ in R.

We give our polarized version of the algorithm below.

Algorithm 3.2 (1) Start with a squarefree monomial ideal $\mathfrak{a} = (g_1, \ldots, g_k) \in S$.
(2) Compute the primary decomposition of $\mathfrak{a}$. By Theorem 2.9, the ideals $p_1, \ldots, p_s$ in the primary decomposition will all be generated by a subset of $\{x_1, \ldots, x_n, y_1, \ldots, y_n\}$.
(3) Set $x_i + y_i = 1$ for $1 \le i \le n$, and as a result remove any ideal in the primary decomposition containing both x_i and y_i for any $1 \le i \le n$.

(4) Compute the intersection of the remaining ideals. Since the p_ℓ are squarefree monomial ideals, this is equivalent to computing $[h_1 \cdots h_s]$ for every set of choices of h_ℓ a generator of p_ℓ.
(5) Impose the relations $x_i y_i = 0$ for $1 \leq i \leq n$, and as a result remove any generator divisible by $x_i y_i$ for some i. (These come from imposing the relations $x_i(1 - x_i) = 0$ in the depolarized ideal.)
(6) Remove any generator that is a multiple of another generator of lower degree.

The remaining generators give the canonical form of $\mathfrak{a}$ in S.

Lemma 3.3 (1) *Let $f = \prod_{i\in\sigma} x_i \prod_{i\in\tau}(1 - x_i)$ be a pseudomonomial in R. Then $d(\mathcal{P}(f)) = f$.*
(2) *Let $g = \prod_{i\in\sigma} x_i \prod_{i\in\tau} y_i$ be a monomial in S. Then $\mathcal{P}(d(g)) = g$.*
(3) *Let $\mathfrak{a} = (g_1, \ldots, g_k)$ be a squarefree monomial ideal in S. Then $d(\mathfrak{a}) = (d(g_1), \ldots, d(g_k))R$, though the generators may not be minimal. If $x_i, y_i \in \mathfrak{a}$ for some $1 \leq i \leq n$, then $d(\mathfrak{a}) = (1)$.*
(4) *Let p be a pseudomonomial prime of R. Then p is in canonical form.*
(5) *Let p be a pseudomonomial prime of R. Then $\mathcal{P}(p)$ is a monomial prime of S.*
(6) *Let $\mathfrak{a}$ be a squarefree monomial ideal of S. If for all $1 \leq i \leq n$ $x_i y_i$ does not divide any generator, then $d(\mathfrak{a})$ is an ideal of R generated by pseudomonomials. In particular, if p is a monomial prime of S such that for all $1 \leq i \leq n$, $\{x_i, y_i\}$ is not a subset of p, then $d(p)$ is a pseudomonomial prime of R.*
(7) *Let $\mathcal{A}$ be a pseudomonomial ideal of R. Then $\mathcal{A} \subseteq d(\mathcal{P}(\mathcal{A}))$. If $\mathcal{A}$ is a pseudomonomial prime, this is an equality.*
(8) *Let p be a monomial prime of S such that for $1 \leq i \leq n$, at most one of x_i, y_i is in p. Then $\mathcal{P}(d(p)) = p$.*

Proof Parts (1) and (2) are clear. Part (3) follows from observing that d is an S-module homomorphism.

For (4), we note that p is in canonical form since it is preserved by Algorithm 3.1. Then (5) follows by Theorem 2.5.

Statement (6) follows from (3) and the fact that an ideal generated by some x_i and some $1-x_j$ where there is no index for which we have both x_i and $1-x_i$ as generators is prime.

To prove (7), first let $f \in \mathcal{A}$ be a pseudomonomial. Then by part (1), $f = d(\mathcal{P}(f)) \in d(\mathcal{P}(\mathcal{A}))$. If $\mathcal{A}$ is a pseudomonomial prime, then without loss of generality it has the form $(x_1, \ldots, x_k, (1-x_{k+1}), \ldots, (1-x_{k+t}))$. Since this ideal is in canonical form by (4), it follows from Theorem 2.5 that $\mathcal{P}(\mathcal{A}) = (x_1, \ldots, x_k, y_{k+1}, \ldots, y_{k+t})$. Then $d(\mathcal{P}(\mathcal{A})) = \mathcal{A}$.

For (8), assume without loss of generality that

$$p = (x_1, \ldots, x_k, y_{k+1}, \ldots, y_{k+t}).$$

Then $d(p) = (x_1, \ldots, x_k, (1 - x_{k+1}), \ldots, (1 - x_{k+t}))$ is in canonical form by (4). The result now follows from Theorem 2.5. □

Example 3.4 Consider the ideal $\mathfrak{a} = (x_1, x_2 y_1)$ of S. The depolarized ideal is $d(\mathfrak{a}) = (x_1, x_2(1 - x_1)) = (x_1, x_2)$, so we see that there is a “more minimal" way to write $d(\mathfrak{a})$, as mentioned in part (3) of Lemma 3.3.

Lemma 3.5 *Let $\mathcal{A} \subseteq R$ be a neural ideal, and let $\mathfrak{a} \subseteq S$ be its polarization. The set of associated primes of $\mathcal{A}$ is equal to the set of primes obtained by depolarizing the set of associated primes of $\mathfrak{a}$, after removing primes containing both x_i and y_i for some $1 \leq i \leq n$.*

Proof We need to prove two claims:

(1) If p is a minimal prime of $\mathcal{A}$, then $\mathcal{P}(p)$ is a minimal prime of $\mathfrak{a}$.
(2) If p is a minimal prime of $\mathfrak{a}$ and there is no $1 \leq i \leq n$ such that p contains both x_i and y_i, then $d(p)$ is a minimal prime of $\mathcal{A}$.

Note that by Theorem 2.9, the irredundant primary decomposition of $\mathfrak{a}$ is unique. Similarly, by Theorem 2.8, the irredundant primary decomposition of $\mathcal{A}$ is also unique. □

Proof of Claim 1 Let p be a minimal prime of $\mathcal{A}$. By Theorem 2.8, p is generated by pseudomonomials x_i and $1 - x_i$ (where for each i, at most one of $x_i, 1 - x_i$ is a generator). By Lemma 3.3, $\mathcal{P}(p)$ is prime, and $\mathfrak{a} \subseteq \mathcal{P}(p)$. Suppose $\mathcal{P}(p)$ is not minimal, i.e. there is a prime q with $\mathfrak{a} \subseteq q \subsetneq \mathcal{P}(p)$. Since $\mathfrak{a}$ is a squarefree monomial ideal, q is generated by a subset of the variables $x_1, \ldots, x_n, y_1, \ldots, y_n$. Since $q \subseteq \mathcal{P}(p)$, which does not contain both x_i and y_i for any i, Lemma 3.3 implies $d(q)$ is prime in R. We have

$$\mathcal{A} \subseteq d(\mathcal{P}(\mathcal{A})) = d(\mathfrak{a}) \subseteq d(q) \subseteq d(\mathcal{P}(p)) = p.$$

Since p was a minimal prime of $\mathcal{A}$ by assumption, $d(q) = p$. Suppose that $q \subsetneq \mathcal{P}(p)$. Then there must be some $z \in \{x_1, \ldots, x_n, y_1, \ldots, y_n\}$ such that $z \in \mathcal{P}(p)$, but $z \notin q$. Then $d(z) \in d(\mathcal{P}(p)) = p = d(q)$. Applying Lemma 3.3, $z = \mathcal{P}(d(z)) \in \mathcal{P}(d(q)) = q$, since q does not contain x_i and y_i for any $1 \leq i \leq n$. So $q = \mathcal{P}(p)$, which implies that $\mathcal{P}(p)$ is a minimal prime of $\mathfrak{a}$. □

Proof of Claim 2 Let p be a minimal prime of $\mathfrak{a}$ such that for $1 \leq i \leq n$, p does not contain both x_i and y_i. Then by Lemma 3.3, $d(p)$ is a pseudomonomial prime and

$$\mathcal{A} \subseteq d(\mathcal{P}(\mathcal{A})) = d(\mathfrak{a}) \subseteq d(p).$$

Suppose there is some pseudomonomial prime q of R such that $\mathcal{A} \subseteq q \subsetneq d(p)$. Then $\mathcal{P}(q)$ is prime and $\mathfrak{a} \subseteq \mathcal{P}(q) \subseteq \mathcal{P}(d(p))$, which is equal to p by Lemma 3.3. Since p is a minimal prime of $\mathfrak{a}$, $\mathcal{P}(q) = p$. Then by Lemma 3.3 and since d is a homomorphism, $q = d(\mathcal{P}(q)) = d(p)$. So $d(p)$ is a minimal prime of $\mathcal{A}$, as desired.

Since neither $\mathcal{A}$ nor $\mathfrak{a}$ has any embedded primes, this proves the result. □

Proposition 3.6 *Let $\mathcal{A} \subseteq \mathbb{F}_2[x_1, \ldots, x_n]$ be a neural ideal, and let $\mathfrak{a}$ be its polarization. The canonical form of $\mathcal{A}$ as computed in Algorithm 3.1 is equal to the depolarization of the canonical form of $\mathfrak{a}$ as computed in Algorithm 3.2.*

Proof First, by Lemma 3.5, at the end of Step 3 of Algorithm 3.2, the remaining set of prime ideals, when depolarized, is equal to the set of prime ideals occurring in the primary decomposition of $\mathcal{A}$.

Next we prove that the set of products left at the end of Step 5 of Algorithm 3.2, depolarized, agree with the products left at the end of Step 4 of Algorithm 3.1. We need to show that if z and w are pseudomonomials in R, if we apply the relations $x_i(1-x_i) = 0$ to zw, the result is the same as if we compute $[\mathcal{P}(z)\mathcal{P}(w)]$, impose the relation $x_i y_i = 0$, and then depolarize. Suppose

$$z = \prod_{i\in\sigma_1} x_i \prod_{i\in\tau_1}(1-x_i), \quad \text{and} \quad w = \prod_{i\in\sigma_2} x_i \prod_{i\in\tau_2}(1-x_i).$$

Then the result of applying the first process to zw is

$$\begin{cases} \prod_{i\in(\sigma_1\cup\sigma_2)} x_i \prod_{i\in(\tau_1\cup\tau_2)}(1-x_i) & (\sigma_1\cup\sigma_2)\cap(\tau_1\cup\tau_2) = \varnothing \\ 0 & \text{else.} \end{cases}$$

The result of applying the second process to $[\mathcal{P}(z)\mathcal{P}(w)]$ is

$$\begin{cases} \prod_{i\in(\sigma_1\cup\sigma_2)} x_i \prod_{i\in(\tau_1\cup\tau_2)} y_i & (\sigma_1\cup\sigma_2)\cap(\tau_1\cup\tau_2) = \varnothing \\ 0 & \text{else.} \end{cases}$$

Depolarizing, this agrees with the result of the first process.

Finally, by [7, Lemma 3.1], if z and w are two pseudomonomials in R, $z \mid w$ if and only if $\mathcal{P}(z) \mid \mathcal{P}(w)$. Hence the depolarization of the result of Step 6 of Algorithm 3.2 agrees with the result of Step 5 of Algorithm 3.1. □

4 Deconstructing the Canonical Form Algorithm

In this section we prove a number of lemmas describing in more detail what happens to the generators of a polarized neural ideal under the steps of Algorithm 3.2 and giving us shortcuts through the algorithm.

Definition 4.1 We say two monomials m_1 and m_2 *share an index* i if x_i divides one of them and y_i divides the other. If a single monomial is divisible by $x_i y_i$, we do not count this as sharing an index with itself (see Remark 4.8). Moreover, we say that the generators of a squarefree monomial ideal $\mathfrak{a}$ *share the index* i if there are two generators of the ideal sharing the index i.

Example 4.2 For example, x_1y_2 and x_3y_1 share the index 1. However, x_1y_2 and x_1y_3 share no index, even though x_1 divides both.

Remark 4.3 We state the results in this section for squarefree monomial ideals rather than for polarized neural ideals so that we can apply them to ideals appearing as intermediate stages in Algorithm 3.2, which may have generators divisible by $x_i y_i$ for some $1 \le i \le n$.

Definition 4.4 Let $\mathfrak{a}$ be a squarefree monomial ideal in S. We say that $\mathfrak{a}$ is *recomposed* if it is the result of applying Steps 1–4 of Algorithm 3.2 to some squarefree monomial ideal, and call the process of applying Steps 1–4 to a squarefree monomial ideal *recomposing* it. We say that $\mathfrak{a}$ is in *almost canonical form* if it is the result of applying Steps 1–5 of Algorithm 3.2 to some squarefree monomial ideal.

Lemma 4.5 *Let $\mathfrak{a}$ be a squarefree monomial ideal in S. For some $0 \leq t \leq n$ and $i_1, \ldots, i_t \in \{1, \ldots, n\}$, we can write $\mathfrak{a} = \mathfrak{a}_0 \cap \mathfrak{a}_1 \cap \ldots \cap \mathfrak{a}_t$, where*

(1) *$\mathfrak{a}_0$ is a (possibly empty) intersection of monomial primes, none of which contain both x_i and y_i for any $1 \leq i \leq n$, and*
(2) *for $1 \leq j \leq t$, $\mathfrak{a}_j$ is an intersection of monomial prime ideals containing both x_{i_j} and y_{i_j}.*

In particular, $\mathfrak{a}_j$ is either the unit ideal or a squarefree monomial ideal for each $0 \leq j \leq n$. For any such decomposition, recomposing $\mathfrak{a}$ returns $\mathfrak{a}_0$, and so $\mathfrak{a}$ and $\mathfrak{a}_0$ have the same canonical form. In fact for any $s \leq t$, recomposing $\mathfrak{a}' = \mathfrak{a}_0 \cap \mathfrak{a}_s \cap \ldots \cap \mathfrak{a}_t$ yields $\mathfrak{a}_0$, and so $\mathfrak{a}'$ has the same canonical form as $\mathfrak{a}_0$.

Consequently, if $\mathfrak{a} = \mathfrak{b} \cap \mathfrak{c}$ with $\mathfrak{b}$ and $\mathfrak{c}$ squarefree monomial ideals and $(x_i, y_i) \subseteq \mathfrak{b}$ for some $1 \leq i \leq n$, recomposing $\mathfrak{a}$ returns the same result as recomposing $\mathfrak{c}$.

Remark 4.6 If $\mathfrak{a}_0$ is empty, then the canonical form of $\mathfrak{a}$ has no generators. This corresponds to the code containing all code words.

Proof We prove this by induction on t, the order of the set

$$I = \{i : 1 \leq i \leq n, (x_i, y_i) \subseteq p \text{ for some } p \text{ in a primary decomposition of } \mathfrak{a}\}.$$

If $t = 0$, $\mathfrak{a} = \mathfrak{a}_0$ so the result is immediate. Suppose $t > 0$. Without loss of generality, assume that $I = \{1, \ldots, t\}$. For $1 \leq i \leq t$, set $\mathfrak{a}_i$ to be the intersection of the prime ideals p appearing in a primary decomposition of $\mathfrak{a}$ that contain x_i and y_i. Set $\mathfrak{a}_0$ to be the intersection of the primes p appearing in the primary decomposition of $\mathfrak{a}$ that do not contain x_i, y_i for any $1 \leq i \leq n$.

When we perform Step 3 of Algorithm 3.2, we will remove exactly the primes p that contain a pair x_i, y_i for some $1 \leq i \leq t$. This exactly corresponds to removing $\mathfrak{a}_1, \ldots, \mathfrak{a}_t$, leaving us with $\mathfrak{a}_0$ when we recompose $\mathfrak{a}$ in Step 4.

The same argument holds for $\mathfrak{a}'$.

For the last statement, if $\mathfrak{a} = \mathfrak{b} \cap \mathfrak{c}$ with $\mathfrak{b}$ and $\mathfrak{c}$ squarefree monomial ideals, where $(x_i, y_i) \subseteq \mathfrak{b}$ for some $1 \leq i \leq n$, every minimal prime of $\mathfrak{b}$ will be part of $\mathfrak{a}_s$ for some $1 \leq s \leq t$, so they will all be removed in recomposing $\mathfrak{a}$. Thus it suffices to recompose $\mathfrak{c}$. □

Definition 4.7 We call applying Steps 1–4 of Algorithm 3.2 to a squarefree monomial ideal $\mathfrak{a} \subseteq S$ but only removing primes containing (x_i, y_i) for a particular i *recomposing $\mathfrak{a}$ with respect to the index i*. We may also refer to recomposing $\mathfrak{a}$ with respect to a subset of $\{1, \ldots, n\}$, by which we mean we recompose with respect to each index in the set. If no set is specified, we are recomposing with respect to $\{1, \ldots, n\}$.

Remark 4.8 We observe that a minimal prime of $\mathfrak{a}$ contains (x_i, y_i) if and only if at least two generators of $\mathfrak{a}$ share the index i. The backward direction follows from the process of decomposing a squarefree monomial ideal. For example, $(x_1, x_2y_1) = (x_1, x_2) \cap (x_1, y_1)$. For the forward direction, if some minimal prime of $\mathfrak{a}$ contains (x_i, y_i), then $\mathfrak{a}$ must contain at least one multiple of x_i and at least one (distinct) multiple of y_i.

As a result, the indices $i_1, \ldots, i_t$ from Lemma 4.5 are the indices shared by the generators of $\mathfrak{a}$. Hence it suffices to recompose an ideal with respect to the indices shared by its generators, as recomposing with respect to a non-shared index removes no minimal primes and thus preserves the ideal.

This also explains why, in Definition 4.1, we do not count i as a shared index among the generators of an ideal $\mathfrak{a}$ if x_iy_i divides a single generator, but neither x_i nor y_i divides any other generator of $\mathfrak{a}$ (see Definition 4.1): such a generator does not give us a minimal prime containing (x_i, y_i). For example, $(x_1y_1, x_2) = (x_1, x_2) \cap (y_1, x_2)$.

Remark 4.9 In computing canonical forms, we may remove generators that are a multiple of another generator of lower degree (or duplicate generators) at any stage of the algorithm, rather than waiting until Step 6. Removing such generators preserves the ideal, and hence the primary decomposition as well as the intersection that occurs in Step 4 of Algorithm 3.2.

Lemma 4.10 *Let $t \geq 2$. Let*

$$\mathfrak{a} = \bigcap_{j=1}^{t} \mathfrak{a}_j,$$

where $\mathfrak{a}$, $\mathfrak{a}_j$ are squarefree monomial ideals of S. Recomposing $\mathfrak{a}$ with respect to an index i is equivalent to recomposing each of the $\mathfrak{a}_j$ with respect to the index i and then intersecting the results.

Proof We prove the case $t = 2$. The rest follows by induction. We show that the set of primes $\{p_i\}$ appearing in a primary decomposition of $\mathfrak{a}$ is equal to the subset of the primes $\{q_i^{(1)}, q_j^{(2)}\}$ appearing in the primary decompositions of $\mathfrak{a}_1$ and $\mathfrak{a}_2$ obtained by removing duplicates and primes that are now non-minimal: we first prove that any prime minimal over $\mathfrak{a}$ is minimal over $\mathfrak{a}_1$ or $\mathfrak{a}_2$. Suppose $\mathfrak{p}$ is minimal over $\mathfrak{a}$ but $\mathfrak{a}_1 \not\subseteq \mathfrak{p}$; we will show $\mathfrak{p}$ is minimal over $\mathfrak{a}_2$. Since $\mathfrak{a}_1 \not\subseteq \mathfrak{p}$, there exists $\alpha \in \mathfrak{a}_1 \backslash \mathfrak{p}$. We observe that $\alpha\mathfrak{a}_2 \subseteq \mathfrak{a}_1 \cap \mathfrak{a}_2 = \mathfrak{a} \subseteq \mathfrak{p}$. Since $\mathfrak{p}$ is prime and $\alpha \notin \mathfrak{p}$, we must have $\mathfrak{a}_2 \subseteq \mathfrak{p}$.

To see that $\mathfrak{p}$ is minimal over $\mathfrak{a}_2$, we consider another prime $\mathfrak{p}'$ such that $\mathfrak{a}_2 \subseteq \mathfrak{p}' \subseteq \mathfrak{p}$. Since $\mathfrak{a} \subseteq \mathfrak{a}_2$, we have $\mathfrak{a} \subseteq \mathfrak{p}' \subseteq \mathfrak{p}$. Since $\mathfrak{p}'$ is prime and $\mathfrak{p}$ is a minimal prime over $\mathfrak{a}$, we must have $\mathfrak{p}' = \mathfrak{p}$.

Next, we consider a prime $\mathfrak{p}$ that is minimal over $\mathfrak{a}_1$ but not over $\mathfrak{a}$. In that case, there is another prime $\mathfrak{p}'$ minimal over $\mathfrak{a}$ such that $\mathfrak{a} \subseteq \mathfrak{p}' \subsetneq \mathfrak{p}$. By the minimality of $\mathfrak{p}$ over $\mathfrak{a}_1$, we know that $\mathfrak{a}_1 \not\subseteq \mathfrak{p}'$. Since $\mathfrak{p}'$ is minimal over $\mathfrak{a}$ but not over $\mathfrak{a}_1$, we find that $\mathfrak{p}'$ is minimal over $\mathfrak{a}_2$. Consequently, if we take all the primes minimal over $\mathfrak{a}_1$ or $\mathfrak{a}_2$ and remove any primes that are not minimal over $\mathfrak{a}$ or are duplicates, we obtain all the primes minimal over $\mathfrak{a}$.

This remains true after removing any primes containing (x_i, y_i) for some $1 \leq i \leq n$ from both sets. Computing the intersection of the remaining p_i is equivalent to separately intersecting the remaining $q_i^{(1)}$ and $q_i^{(2)}$, and then intersecting the two resulting ideals. □

Example 4.11 We use Lemmas 4.5 and 4.10 to compute the canonical form of $\mathfrak{a} = (x_1x_2x_3, x_4y_1y_2)$. We begin decomposing $\mathfrak{a}$ as follows:

$$\begin{aligned}(x_1x_2x_3, x_4y_1y_2) &= (x_1, x_4y_1y_2) \cap (x_2, x_4y_1y_2) \cap (x_3, x_4y_1y_2)\\ &= (x_1, y_1) \cap (x_1, x_4y_2) \cap (x_2, y_2) \cap (x_2, x_4y_1) \cap (x_3, x_4y_1y_2).\end{aligned}$$

We remove the 1st and 3rd pieces. Notice that it is impossible for any of the other 3 pieces to decompose into primes containing a pair x_i, y_i. By Lemma 4.5, these pieces are recomposed, and by Lemma 4.10, their intersection is also recomposed, so we can jump straight to Steps 4–6, performed together:

$$(x_1, x_4y_2) \cap (x_2, x_4y_1) \cap (x_3, x_4y_1y_2) \to (x_1x_2x_3, x_4y_1y_2).$$

Notation 4.12 We refer to the generators of an ideal as $g_1, \ldots, g_k$, frequently pulling out x_i and y_i when i is a shared index between the g_j. For example, we write (x_1g_1, y_1g_2). The g_j will always be squarefree monomials.

For ease of notation, we write $x_1, x_2, \ldots$ and $y_1, y_2, \ldots$ instead of $x_{i_1}, x_{i_2}, \ldots$ and $y_{i_1}, y_{i_2}, \ldots$. However, all of our results hold if we permute the indices $1, \ldots, n$ on the variables.

Lemma 4.13 *Let $\mathfrak{a} = (g_1, \ldots, g_k, h_1, \ldots, h_\ell)$ be a squarefree monomial ideal in S such that for every $1 \leq j_1 \leq k$ and $1 \leq j_2 \leq \ell$, g_{j_1} and h_{j_2} do not share any indices (resp. the index i). Then recomposing $\mathfrak{a}$ (resp. with respect to the index i) gives the same result as recomposing $(g_1, \ldots, g_k)$ and $(h_1, \ldots, h_\ell)$ (resp. with respect to the index i) and taking the union of their sets of generators.*

More generally, recomposing the neural ideal

$$\mathfrak{a} = (g_{1,1}, \ldots, g_{1,k_1}, g_{2,1}, \ldots, g_{2,k_2}, \ldots, g_{m,k_m})$$

(resp. with respect to the index i), where g_{j_1,t_1} and g_{j_2,t_2} do not share any indices (resp. the index i) for any $j_1 \neq j_2$, agrees with recomposing each of the $(g_{j,1}, \ldots, g_{j,k_j})$ (resp. with respect to the index i) and taking the union of their sets of generators.

Consequently, the canonical form (resp. almost canonical form) of $\mathfrak{a}$ is the union of the sets of generators of the canonical forms (resp. almost canonical form) of the $(g_{j,1}, \ldots, g_{j,t_j})$, up to removing any generator divisible by another.

Proof We prove the result in the first paragraph; the second paragraph follows by induction. We first prove the case where the g_{j_1} and h_{j_2} do not share the index i. Suppose the generators of $\mathfrak{a}$ share the index i. Without loss of generality, assume i is a shared index for $(g_1, \ldots, g_k)$. Then by our hypotheses, x_i and y_i do not divide any h_j. Hence no minimal prime of $(h_1, \ldots, h_\ell)$ contains either x_i or y_i.

Consequently, we can write $\mathfrak{a} = (G' \cap G_i) + H = (G' + H) \cap (G_i + H)$, where $H = (h_1, \ldots, h_\ell)$ and G', G_i are the components of $G = (g_1, \ldots, g_k)$ from the statement of Lemma 4.5, where G_i is the squarefree monomial ideal that is the intersection of all minimal primes of G containing (x_i, y_i), and G' is the squarefree monomial ideal that is the intersection of the minimal primes of G not containing (x_i, y_i). In particular, G' is equal to the result of recomposing G with respect to the index i. The minimal primes of $G' + H$ are all of the form $\mathfrak{p}' + \mathfrak{p}$ where $\mathfrak{p}'$ is minimal over G' and $\mathfrak{p}$ is minimal over H [11, Theorem 7.3.4]. Since no minimal prime of H contains (x_i) or (y_i), $G' + H$ is equal to the intersection of the minimal primes of $\mathfrak{a}$ not containing (x_i, y_i), and $G_i + H$ is equal to the intersection of the minimal primes of $\mathfrak{a}$ containing (x_i, y_i).

By Lemma 4.5 applied to $\mathfrak{a}$, recomposing $\mathfrak{a}$ is equal to recomposing $G' + H$ (with respect to any index or set of indices). Further, $G' + H$ has no minimal primes containing (x_i, y_i) so it is already recomposed with respect to the index i. Finally, by Lemma 4.5, recomposing G' agrees with recomposing $(g_1, \ldots, g_k)$. As a result, if instead we recomposed G and H separately with respect to the index i and add the results, we will get $G' + H$. This proves the result for a single index.

The multiple index result follows by recomposing $\mathfrak{a}$ one index at a time as in Lemma 4.5; at each index i, no g_{j_1} and h_{j_2} share the index i. The final paragraph of the result follows by computing the almost canonical form or canonical form as appropriate. □

Example 4.14 Let $\mathfrak{a} = (x_1x_2, x_2y_1, x_3x_4y_5, x_2x_5y_4)$. We can write $\mathfrak{a} = \mathfrak{b} + \mathfrak{c}$, where $\mathfrak{b} = (x_1x_2, x_2y_1)$ and $\mathfrak{c} = (x_3x_4y_5, x_2x_5y_4)$, since these pieces share no indices with each other. Note that x_2 appears in both, but since y_2 never appears, 2 is not a shared index. The canonical form of $\mathfrak{b}$ is (x_2) and of $\mathfrak{c}$ is $(x_3x_4y_5, x_2x_5y_4)$, so the almost canonical form of $\mathfrak{a}$ is $(x_2, x_3x_4y_5, x_2x_5y_4)$. Removing $x_2x_5y_4$, which is a multiple of x_2, we are left with $(x_2, x_3x_4y_5)$.

Lemma 4.15 *Let $\mathfrak{a} = (g_1, \ldots, g_k)$ be a squarefree monomial ideal of S. If for some $0 \le \ell < k$ and every $\ell+1 \le j \le k$, g_j shares no indices (resp. does not share the index i) with any other generator, then $g_{\ell+1}, \ldots, g_k$ are generators of the recomposition of $\mathfrak{a}$ with respect to every $1 \le i \le n$ (resp. the index i). If in addition $x_iy_i \nmid g_j$ for any $\ell + 1 \le j \le k$, they are generators of the almost canonical form of $\mathfrak{a}$.*

In particular, if $\mathfrak{a}$ does not have any shared indices, then the ideal is recomposed. If in addition no generator is divisible by any x_iy_i and no generator is a multiple of another, the ideal is in canonical form.

Proof This follows from Lemma 4.13. □

Example 4.16 By Lemma 4.15, the neural ideal (x_1y_2, x_3y_2, x_1x_4) is in canonical form.

Lemma 4.17 *Let $\mathfrak{a} = (x_1g_{11}, \ldots, x_1g_{1k_1}, y_1g_{21}, \ldots, y_1g_{2k_2}, g_{31}, \ldots, g_{3k_3})$, be a squarefree monomial ideal in S, where $k_1, k_2 \ge 0$ and the only factors of x_1, y_1 appearing among the generators of $\mathfrak{a}$ are the ones shown. Then recomposing $\mathfrak{a}$ with respect to the index 1 returns*

$$\mathfrak{a}' = (x_1g_{11}, \ldots, x_1g_{1k_1}, y_1g_{21}, \ldots, y_1g_{2k_2}, [g_{1j_1}g_{2j_2}]_{1\le j_1\le k_1, 1\le j_2\le k_2}, g_{31}, \ldots, g_{3k_3}).$$

If the only indices shared by the generators of $\mathfrak{a}$ *are the* x_1, y_1 *shown,* $\mathfrak{a}'$ *is the almost canonical form of* $\mathfrak{a}$*. In this case the canonical form of* $\mathfrak{a}$ *is found by removing generators of* $\mathfrak{a}'$ *that are a multiple of another generator.*

For any pair g_{1j_1}, g_{2j_2} *that share an index,* $[g_{1j_1}g_{2j_2}]$ *will be removed in Step 5 of Algorithm 3.2.*

Proof By Lemma 4.15, we may assume $k_3 = 0$. We prove this by strong induction on $k_1 + k_2$. If $k_1 + k_2 = 0$ or 1, then $k_1 = 0$ or $k_2 = 0$. In this case, 1 is not a shared index for the generators of $\mathfrak{a}$ and $\mathfrak{a}' = \mathfrak{a}$. The result follows from Lemmas 4.5 and 4.15. We do the base case of $k_1 = k_2 = 1$ so that $k_1 + k_2 = 2$. In this case, we have $\mathfrak{a} = (x_1g_1, y_1g_2)$. We write $\mathfrak{a} = (x_1, y_1) \cap (x_1, g_2) \cap (g_1, y_1g_2)$.

Neither piece 2 nor 3 have a minimal prime containing (x_1, y_1), so by Lemma 4.5, we remove the first component and intersect components 2 and 3, leaving us with $(x_1g_1, x_1y_1g_2, [g_1g_2], y_1g_2)$. Since the second generator is divisible by the last, we remove it, leaving us with $(x_1g_1, y_1g_2, [g_1g_2])$.

For the inductive step, suppose that $k_1 + k_2 > 2$, $k_1, k_2 \geq 1$, and without loss of generality, that $k_1 > 1$. We have

$$\mathfrak{a} = (x_1, y_1g_{21}, \ldots, y_1g_{2k_2}) \cap (g_{11}, x_1g_{12}, \ldots, x_1g_{1k_1}, y_1g_{21}, \ldots, y_1g_{2k_2}).$$

By Lemma 4.10, it suffices to recompose these ideals with respect to the index 1 and intersect the results. By the induction hypothesis, this gives us:

$$(x_1, g_{21}, \ldots, g_{2k_2}) \cap (g_{11}, x_1g_{12}, \ldots, x_1g_{1k_1}, y_1g_{21}, \ldots, y_1g_{2k_2}, \\ [g_{1j_1}g_{2j_2}]_{1\leq j_1\leq k_1, 1\leq j_2\leq k_2}).$$

Intersecting these, we get the desired result. □

Remark 4.18 We can interpret this lemma in terms of the RF-structure of the neural code as well. For each pair of generators x_1g_1, y_1g_2, say $g_1 = \prod_{i\in\sigma_1} x_i \prod_{i\in\tau_1} y_i$ and $g_2 = \prod_{i\in\sigma_2} x_i \prod_{i\in\tau_2} y_i$, where $1 \notin \sigma_1 \cup \sigma_2 \cup \tau_1 \cup \tau_2$, we have relations

$$U_1 \bigcap \left(\bigcap_{i\in\sigma_1} U_i \right) \subseteq \bigcup_{i\in\tau_1} U_i \quad \text{and} \quad \bigcap_{i\in\sigma_2} U_i \subseteq U_1 \bigcup \left(\bigcup_{i\in\tau_2} U_i \right).$$

We claim that these two relations together imply the relation

$$\left(\bigcap_{i\in\sigma_1} U_i \right) \bigcap \left(\bigcap_{i\in\sigma_2} U_i \right) \subseteq \left(\bigcup_{i\in\tau_1} U_i \right) \bigcup \left(\bigcup_{i\in\tau_2} U_i \right).$$

To see this, we set $X = U_1$, $A = \bigcap_{i\in\sigma_1} U_i$, $B = \bigcup_{i\in\tau_1} U_i$, $C = \bigcap_{i\in\sigma_2} U_i$, and $D = \bigcup_{i\in\tau_2} U_i$. Then $X \cap A \subseteq B$ and $C \subseteq X \cup D$. Since $A \cap C \subseteq C$, it is also contained in $X \cup D$. Then

$$A \cap C \subseteq A \cap (X \cup D) = (A \cap X) \cup (A \cap D) \subseteq B \cup (A \cap D) \subseteq B \cup D,$$

proving our claim.

As a result, by [4, Sect. 4.3] the neural ideal contains a monomial

$$[g_1 g_2] = \left[\prod_{i \in \sigma_1 \cup \sigma_2} x_i \prod_{i \in \tau_1 \cup \tau_2} y_i \right].$$

If g_1 and g_2 share no indices, then either this monomial or a monomial dividing it must be in the canonical form. If g_1 and g_2 share an index, this monomial will be removed in Step 5 of Algorithm 3.2.

Proposition 4.19 *Let $\mathfrak{a} = (g_1, \ldots, g_k) \subseteq S$ be a squarefree monomial ideal. Then recomposing $\mathfrak{a}$ (with respect to any shared index or to all shared indices) returns an ideal whose generators include $g_1, \ldots, g_k$.*

Consequently, for any j such that $x_i y_i \nmid g_j$ for any $1 \le i \le n$, the generators of the almost canonical form of $\mathfrak{a}$ include g_j. Any removal of such a g_j happens in Step 6. Hence for each such j, the canonical form of $\mathfrak{a}$ has a generator dividing g_j.

In addition, if $\mathfrak{a}$ has a generator g_j such that $x_i y_i \mid g_j$ for some $1 \le i \le n$, g_j may be removed at any stage of Algorithm 3.2.

Proof To prove that recomposing $\mathfrak{a}$ does not remove g_j from the ideal, the main thing we need to prove is that removing primes containing an x_i, y_i pair from the primary decomposition of $\mathfrak{a}$ does not remove g_j. We prove the result for recomposing with respect to a single index. The result for recomposing with respect to multiple indices follows by induction.

Without loss of generality, suppose that 1 is a shared index for the generators of $\mathfrak{a}$. Rename the generators of $\mathfrak{a}$ as

$$\mathfrak{a} = (x_1 y_1 g_{11}, \ldots, x_1 y_1 g_{1k_1}, x_1 g_{21}, \ldots, x_1 g_{2k_2}, y_1 g_{31}, \ldots, y_1 g_{3k_3}, g_{41}, \ldots, g_{4k_4}),$$

where $k_j \ge 0$ for $i = 1, 2, 3, 4$, and $x_1, y_1 \nmid g_{ij}$ for any i, j. We work by induction on k_1.

If $k_1 = 0$, $\mathfrak{a} = (x_1 g_{21}, \ldots, x_1 g_{2k_2}, y_1 g_{31}, \ldots, y_1 g_{3k_3}, g_{41}, \ldots, g_{4k_4})$. By Lemma 4.5, recomposing $\mathfrak{a}$ agrees with recomposing the ideal $\mathfrak{a}'$ we get by removing all primes containing both x_1 and y_1 from $\mathfrak{a}$. By Lemmas 4.17 and 4.15,

$$\begin{aligned}\mathfrak{a}' =& (x_1 g_{21}, \ldots, x_1 g_{2k_2}, y_1 g_{31}, \ldots, y_1 g_{3k_3}, \\ & [g_{2j_2} g_{3j_3}]_{1 \le j_2 \le k_2, 1 \le j_3 \le k_3}, g_{41}, \ldots, g_{4k_4}).\end{aligned}$$

This contains all of the original generators of the ideal. Thus when $k_1 = 0$, the original generators of $\mathfrak{a}$ are generators of the recomposed $\mathfrak{a}'$, and hence of the recomposed $\mathfrak{a}$.

Our induction hypothesis is that for $\ell_1 = k_1 - 1$, $\ell_2, \ell_3, \ell_4 \ge 0$, and any choices of h_{ij} not divisible by x_1 or y_1, recomposing the ideal

$$\mathfrak{a} = (x_1 y_1 h_{11}, \ldots, x_1 y_1 h_{1\ell}, x_1 h_{21}, \ldots, x_1 h_{2\ell_2}, y_1 h_{31}, \ldots, y_1 h_{3\ell_3}, h_{41}, \ldots, h_{4\ell_4})$$

with respect to the index 1 returns

$$\mathfrak{a}' = (x_1y_1h_{11}, \ldots, x_1y_1h_{1\ell}, x_1h_{21}, \ldots, x_1h_{2\ell_2}, y_1h_{31}, \ldots, y_1h_{3\ell_3},\\ [h_{2j_2}g_{3j_3}]_{1\le j_2\le k_2, 1\le j_3\le k_3}, h_{41}, \ldots, h_{4\ell_4}).$$

If $k_1 > 0$, we have:

$$\mathfrak{a} =(x_1, y_1g_{31}, \ldots, y_1g_{3k_3}, g_{41}, \ldots) \cap (y_1, x_1g_{21}, \ldots, x_1g_{2k_2}, g_{41}, \ldots)\\ \cap (g_{11}, x_1y_1g_{12}, \ldots, x_1y_1g_{1k_1}, x_1g_{21}, \ldots, x_1g_{2k_2}, y_1g_{31}, \ldots, y_1g_{3k_3}, g_{41}, \ldots).$$

By Lemma 4.10, recomposing $\mathfrak{a}$ with respect to the index 1 agrees with recomposing (with respect to the index 1) and then intersecting the 3 components. Lemma 4.17 along with the induction hypothesis gives us their recomposed forms:

$$(x_1, g_{31}, \ldots, g_{3k_3}, g_{41}, \ldots) \cap (y_1, g_{21}, \ldots, g_{2k_2}, g_{41}, \ldots) \cap (g_{11}, x_1y_1g_{12}, \ldots, x_1y_1g_{1k_1},\\ x_1g_{21}, \ldots, x_1g_{2k_2}, y_1g_{31}, \ldots, y_1g_{3k_3}, [g_{2j_2}g_{3j_3}]_{1\le j_2\le k_2, 1\le j_3\le k_3}, g_{41}, \ldots, g_{4k_4}).$$

Note that in each of the first two pieces, we have already removed a number of pieces divisible by x_1 (resp. y_1), following Remark 4.9.

Intersecting the first two ideals (while removing monomials divisible by another monomial) gives

$$(x_1y_1, x_1g_{21}, \ldots, x_1g_{2k_2}, y_1g_{31}, \ldots, y_1g_{3k_3}, [g_{2j_2}g_{3j_3}]_{1\le j_2\le k_2, 1\le j_3\le k_3}, g_{41}, \ldots, g_{4k_4}).$$

Intersecting this with the 3rd piece then gives

$$(x_1y_1g_{11}, \ldots, x_1y_1g_{1k_1}, x_1g_{21}, \ldots, x_1g_{2k_2}, y_1g_{31}, [g_{21}g_{31}], \ldots, [g_{2k_2}g_{31}],\\ \ldots, y_1g_{3k_3}, [g_{21}g_{3k_3}], \ldots, [g_{2k_2}g_{3k_3}], g_{41}, \ldots, g_{4k_4}),$$

as desired. The original generators of $\mathfrak{a}$ are still generators of this ideal, proving the first statement.

For the second statement, we see by the first statement that every generator g_j of $\mathfrak{a}$ appears in the recomposed $\mathfrak{a}$, and hence if $x_iy_i \nmid g_j$ for any $1 \le i \le n$, g_j appears in the almost canonical form of $\mathfrak{a}$. As a result, performing Step 6 of Algorithm 3.2 leaves us with a generator that divides g_j for each j such that $x_iy_i \nmid g_j$ for any $1 \le i \le n$.

We return to viewing $\mathfrak{a} = (g_1, \ldots, g_k)$. Now suppose $x_iy_i \mid g_j$ for some $1 \le j \le k$ and $1 \le i \le n$. We see from the proof of the first statement that recomposing $\mathfrak{a}$ with respect to the index i or any other index i' such that $x_{i'}y_{i'} \mid g_j$ preserves g_j, but g_j does not contribute to any of the new generators produced by this recomposition. When we perform Step 5 of Algorithm 3.2, g_j will be removed.

We have 3 cases for what happens to g_j when we recompose $\mathfrak{a}$ with respect to an index i' such that $x_{i'}y_{i'} \nmid g_j$: first, if $x_{i'}, y_{i'} \nmid g_j$, preserves g_j, and g_j does not contribute to any of the new generators produced by this recomposition. When we perform Step 5 of Algorithm 3.2, g_j will be removed since it is divisible by x_iy_i.

Second, if $x_{i'} \mid g_j$, but $y_{i'} \nmid g_j$, the recomposition produces g_j and some number of $\left[\frac{g_j}{x_{i'}}h\right]$. All of these are removed in Step 5 of Algorithm 3.2 as they are divisible by $x_i y_i$. The third case, where $y_{i'} \mid g_j$ and $x_{i'} \nmid g_j$, is similar.

Since any generator built from g_j will be removed in Step 5 of Algorithm 3.2, removing g_j before recomposing $\mathfrak{a}$ with respect to any index or set of indices does not change the result of Algorithm 3.2. □

Remark 4.20 By Remark 4.9 and Proposition 4.19, we may remove generators divisible by $x_i y_i$ for some $1 \leq i \leq n$ or by another generator of lower degree at any point of Algorithm 3.2 without altering the result. We use this to shorten computations throughout Sects. 5 and 6. Consequently, we refer to an ideal that only requires Steps 5 and 6 (resp. Step 6) of Algorithm 3.2 to find the canonical form as recomposed (resp. in almost canonical form), even though these ideals may not agree with the results of Step 4 (resp. Step 5) of the algorithm.

5 Classification of Canonical Neural Ideals

In this section, we apply the results of Sect. 4 to determine which polarized neural ideals are in canonical form. Our main result, Theorem 5.5, gives a simple criterion for determining whether a neural ideal is in canonical form based only on the shared indices in its generators. This allows us to describe explicitly the set of squarefree monomial ideals in a particular ring $\mathbb{F}_2[x_1, \ldots, x_n, y_1, \ldots, y_n]$ that are neural ideals in canonical form.

As an application of this result, we give a simpler algorithm for computing the canonical form of a neural ideal in Algorithm 5.10. This algorithm starts with a neural ideal, making it useful to mathematicians studying properties of neural ideals with an eye towards applications to neural codes.

Notably, our results do not depend on the total number of neurons, only the indices involved in the generators of the ideal.

Remark 5.1 Any neural ideal with a single generator is in canonical form. This is because the components of its primary decomposition are all principal, and cannot contain (x_i, y_i) for any $1 \leq i \leq n$. Alternatively, with only one generator, there is no pair of generators that can share an index.

Theorem 5.2 *The canonical forms of polarized neural ideals with 2 generators, neither dividing the other, are as follows:*

(1) *Any polarized neural ideal of the form (g_1, g_2) where g_1 and g_2 share no indices or at least two indices, is in canonical form.*
(2) *Any polarized neural ideal of the form $(x_1 g_1, y_1 g_2)$, where g_1 and g_2 share no indices, is not in canonical form. Recomposing the ideal yields $(x_1 g_1, y_1 g_2, [g_1 g_2])$, and so the canonical form of this neural ideal is:*
   (a) $(x_1 g_1, y_1 g_2, [g_1 g_2])$ *if* $[g_1 g_2] \neq g_1, g_2$,
   (b) $(x_1 g_1, g_2)$ *if* $[g_1 g_2] = g_2 \neq g_1$,

(c) $(g_1, y_1 g_2)$ *if* $[g_1 g_2] = g_1 \neq g_2$
(d) *and* (g_1) *if* $g_1 = g_2$.

Proof (1) The case where g_1 and g_2 share no indices follows from Lemma 4.15.

In the case where g_1 and g_2 share two or more indices, we can rewrite the ideal as either

$$(x_1 \cdots x_t g_1, y_1 \cdots y_t g_2),$$

where $t \geq 2$ and g_1 and g_2 share no indices, or

$$(x_1 \cdots x_t y_{t+1} \cdots y_{t+s} g_1, x_{t+1} \cdots x_{t+s} y_1 \cdots y_t g_2),$$

where $t, s > 0$ and g_1 and g_2 share no indices.

We prove the first case by induction on k. First assume $k = 2$. We use Lemma 4.5 to simplify the computation. We have

$$(x_1 x_2 g_1, y_1 y_2 g_2) = (x_1, y_1) \cap (x_2 g_1, y_1) \cap (x_1, y_2 g_2) \cap (x_2 g_1, y_2 g_2).$$

The first piece is the only piece containing (x_1, y_1), so we remove it. Next we look for components containing (x_2, y_2):

$$\begin{aligned}&(x_2 g_1, y_1) \cap (x_1, y_2 g_2) \cap (x_2 g_1, y_2 g_2)\\ &\qquad = (x_2 g_1, y_1) \cap (x_1, y_2 g_2) \cap (x_2, y_2) \cap (x_2, g_2) \cap (g_1, y_2) \cap (g_1, g_2).\end{aligned}$$

Removing (x_2, y_2) and completing Algorithm 3.2 yields $(x_1 x_2 g_1, y_1 y_2 g_2)$, as desired. Now suppose $t > 2$. We have

$$(x_1 \cdots x_t g_1, y_1 \cdots y_t g_2) = (x_1, y_1) \cap (x_1, y_2 \cdots y_t g_2) \cap (x_2 \cdots x_t g_1, y_1 \cdots y_t g_2).$$

By Lemma 4.5, the canonical form of the original ideal is equal to the canonical form of the intersection of pieces 2 and 3. By (1) and the induction hypothesis, both pieces are in canonical form. Hence by Lemma 4.10, we may apply Steps 5 and 6 of Algorithm 3.2 to their intersection to get the canonical form of the original ideal. Intersecting the ideals

$$(x_1, y_2 \cdots y_t g_1) \cap (x_2 \cdots x_t g_1, y_1 \cdots y_t g_2)$$

and completing Steps 5 and 6 of Algorithm 3.2, we get the desired result.

We prove the second case by induction on $t+s$, starting with the base case $t = s = 1$. In this case, we have

$$(x_1 y_2 g_1, x_2 y_1 g_2) = (x_1, y_1) \cap (x_1, x_2 g_2) \cap (y_2, x_2) \cap (g_1, x_2) \cap (y_2 g_1, y_1 g_2).$$

Removing the pieces (x_1, y_1) and (y_2, x_2) as in Lemma 4.5, we intersect the rest of the ideals following Algorithm 3.2 to get the ideal $(x_1 y_2 g_1, x_2 y_1 g_2)$.

Now assume that $t+s>1$, and without loss of generality that $t>1$. We have

$$(x_1\cdots x_t y_{t+1}\cdots y_{t+s}g_1, x_{t+1}\cdots x_{t+s}y_1\cdots y_t g_2)$$
$$= (x_1, x_{t+1}\cdots x_{t+s}y_1\cdots y_t g_2)\cap(x_2\cdots x_t y_{t+1}\cdots y_{t+s}g_1, x_{t+1}\cdots x_{t+s}y_1\cdots y_t g_2).$$

By the induction hypothesis, the second piece is in canonical form (1 is no longer a shared index). By part (4), the canonical form of the first piece is $(x_1, x_{t+1}\cdots x_{t+s}y_2\cdots y_t g_2)$. Combining these using Lemma 4.10, we get the desired result.

(2) We have $(x_1g_1, y_1g_2) = (x_1, y_1)\cap(x_1, g_2)\cap(g_1, y_1g_2)$. By Lemma 4.5, the canonical form of the original ideal is equal to the canonical form of the intersection of the second and third ideals. Removing the first piece and combining the other two via Algorithm 3.2, we get the almost canonical form $(x_1g_1, y_1g_2, [g_1g_2])$. The result follows by removing generators that are a multiple of another generator. If $[g_1g_2]\neq g_1, g_2$, we remove no generators. If $[g_1g_2] = g_1 \neq g_2$, we remove x_1g_1. If $[g_1g_2] = g_2 \neq g_1$, we remove y_1g_2. If $[g_1g_2] = g_1 = g_2$, we remove x_1g_1 and y_1g_2. □

Example 5.3 We give examples of each case in Theorem 5.2.

(1) The neural ideal (x_1y_2, x_3y_2) is in canonical form.
(2) The neural ideal $(x_1x_2y_3, x_4y_1y_2)$ is in canonical form.
(3) The neural ideal (x_1y_2, x_2y_1) is in canonical form.
(4) The neural ideal (x_1y_2, x_3y_1) has canonical form (x_1y_2, x_3y_1, x_3y_2). In contrast, the neural ideal (x_1, x_3y_1) has canonical form (x_1, x_3).

Remark 5.4 Following the lead of Remark 4.18, in the case where the two generators share a single index and either $g_1 = 1$ or $g_2 = 1$, we can see that the ideal is not in canonical form more directly from the generators and the conditions for the canonical form given in [4, Theorem 4.3]:

First, we assume that $g_2 = 1$, so the ideal is (x_1g_1, y_1). Suppose that $g_1 = x_2\cdots x_t y_{t+1}\cdots y_{t+s}$. Then the generators tell us that

$$U_1\cap U_2\cap\ldots\cap U_t\subseteq U_{t+1}\cup\ldots\cup U_{t+s},\ X\subseteq U_1. \tag{2}$$

But then $U_1\cap\ldots\cap U_t = U_2\cap\ldots\cap U_t$, so the left hand side in Eq. 2 is non-minimal. Hence (x_1g_1, y_1) is not in canonical form. Further, removing x_1 from the first generator fixes the issue, which agrees with the result of Theorem 5.2.

Now suppose that $g_1 = 1$, so that the ideal becomes (x_1, y_1g_2). Suppose that $g_2 = x_2\cdots x_t y_{t+1}\cdots y_{t+s}$. Then the generators tell us that

$$U_1 = \varnothing,\ U_2\cap\ldots\cap U_t\subseteq U_1\cup U_{t+1}\cup\ldots\cup U_{t+s}. \tag{3}$$

Since $U_1 = \varnothing$, the right hand side in Eq. 3 is non-minimal. Hence (x_1, y_1g_2) is not in canonical form. In this case, removing y_1 from the second generator fixes the issue, which is consistent with the result of Theorem 5.2.

Theorem 5.5 *Let $\mathfrak{a} = (g_1, \ldots, g_k)$ be a polarized neural ideal such that $g_{j_1} \nmid g_{j_2}$ for any $1 \leq j_1 \neq j_2 \leq k$, and $x_i y_i \nmid g_j$ for any $1 \leq i \leq n$ and $1 \leq j \leq k$. If for some pair g_{j_1}, g_{j_2} of generators of $\mathfrak{a}$, g_{j_1} and g_{j_2} share exactly 1 index i and no other generator of $\mathfrak{a}$ divides $\frac{[g_{j_1} g_{j_2}]}{x_i y_i}$, then $\mathfrak{a}$ is not in canonical form. Otherwise, $\mathfrak{a}$ is in canonical form.*

Proof First assume $\mathfrak{a}$ has at least one pair of generators with exactly one shared index, say the index 1. To distinguish the generators with a factor of x_1 or y_1, we rewrite $\mathfrak{a} = (x_1 g_{11}, \ldots, x_1 g_{1k_1}, y_1 g_{21}, \ldots, y_1 g_{2k_2}, g_{31}, \ldots, g_{3k_3})$, where $x_1, y_1 \nmid g_{j\ell}$ for any j, ℓ. By Lemma 4.17, recomposing $\mathfrak{a}$ with respect to the index 1 returns

$$\mathfrak{a}' = (x_1 g_{11}, \ldots, x_1 g_{1k_1}, y_1 g_{21}, \ldots, y_1 g_{2k_2}, [g_{1j_1} g_{2j_2}]_{1 \leq j_1 \leq k_1, 1 \leq j_2 \leq k_2}, g_{31}, \ldots, g_{3k_3}),$$

which by Lemma 4.5 has the same canonical form of $\mathfrak{a}$.

By our hypothesis, some pair g_{1j_1}, g_{2j_2} has no shared indices, so by Proposition 4.19, the canonical form of $\mathfrak{a}$ either has the generator $[g_{1j_1} g_{2j_2}] = \frac{[x_1 g_{1j_1} y_1 g_{2j_2}]}{x_1 y_1}$ or has another generator g that is a proper divisor of $[g_{1j_1} g_{2j_2}]$.

If no original generator of $\mathfrak{a}$ divides $[g_{1j_1} g_{2j_2}]$, then g is not an original generator of $\mathfrak{a}$, so $\mathfrak{a}$ is not in canonical form.

If, however, for every index i and every pair of generators g_{j_1} and g_{j_2} of $\mathfrak{a}$ that share only the index i, some other generator of $\mathfrak{a}$ divides $\frac{[g_{j_1} g_{j_2}]}{x_i y_i}$, then every term of the form $\frac{[g_{j_1} g_{j_2}]}{x_i y_i}$ is removed by Step 6 of Algorithm 3.2, and by Remark 4.9 may be removed now. Hence recomposing $\mathfrak{a}$ returns $\mathfrak{a}$. By the hypotheses that no $x_i y_i$ divides any g_j and no g_j divides another, $\mathfrak{a}$ is in canonical form.

Next assume that no generators of $\mathfrak{a}$ share any indices. Then $\mathfrak{a}$ is in canonical form by Lemma 4.15.

Finally, assume that no generators of $\mathfrak{a}$ share exactly one index, and that at least one pair of generators shares an index. Without loss of generality, assume 1 is a shared index among the generators of $\mathfrak{a}$. Rewrite

$$\mathfrak{a} = (x_1 g_{11}, \ldots, x_1 g_{1k_1}, y_1 g_{21}, \ldots, y_1 g_{2k_2}, g_{31}, \ldots, g_{3k_3}),$$

where $x_1, y_1 \nmid g_{j\ell}$ for any j, ℓ.

By Lemma 4.17, recomposing $\mathfrak{a}$ with respect to the index 1 returns

$$\mathfrak{a}' = (x_1 g_{11}, \ldots, x_1 g_{1k_1}, y_1 g_{21}, \ldots, y_1 g_{2k_2}, [g_{1j_1} g_{2j_2}]_{1 \leq j_1 \leq k_1, 1 \leq j_2 \leq k_2}, g_{31}, \ldots, g_{3k_3}),$$

which by Lemma 4.5 has the same canonical form as $\mathfrak{a}$.

However, since any two generators that share an index share at least 2 indices, each $[g_{1j_1} g_{2j_2}]$ is divisible by $x_i y_i$ for some $2 \leq i \leq n$. So they will all vanish when Step 5 of Algorithm 3.2 is applied. By Proposition 4.19, we can remove them now, leaving us with $\mathfrak{a}$ again. We repeat this process for each additional shared index, getting $\mathfrak{a}$ again each time. Then $\mathfrak{a}$ recomposed is equal to $\mathfrak{a}$. By assumption, $x_i y_i$ does not divide any generator of $\mathfrak{a}$, so $\mathfrak{a}$ is then in almost canonical form. Since we assumed no generator of $\mathfrak{a}$ divides another, $\mathfrak{a}$ is in canonical form. □

Note that the divisibility condition in Theorem 5.5 requires the ideal to have at least 3 generators, hence why it is not a part of Theorem 5.2.

Example 5.6 We include an example to illustrate the case where two generators g_{j_1} and g_{j_2} share exactly one index i, but another generator divides $\frac{[g_{j_1} g_{j_2}]}{x_i y_i}$. Let $\mathfrak{a} = (x_1x_2, x_3x_4y_1, x_2x_3)$. The first two generators share only the index 1, but this ideal is nevertheless in canonical form. By Theorem 5.2 along with Lemma 4.15, the almost canonical form of this ideal is

$$(x_1x_2, x_3x_4y_1, x_2x_3x_4, x_2x_3).$$

We remove $x_2x_3x_4$ since it is divisible by x_2x_3, returning the original ideal. Hence $\mathfrak{a}$ is in canonical form.

Remark 5.7 We note that any squarefree monomial ideal in $\mathbb{F}_2[x_1, \dots, x_n]$ can be realized as a neural ideal in canonical form by passing to the extension ring $\mathbb{F}_2[x_1, \dots, x_n, y_1, \dots, y_n]$. For example, $(x_1x_2, x_2x_3, x_3x_4, x_4x_5, x_5x_1)$ is the canonical form of a neural ideal in $\mathbb{F}_2[x_1, \dots, x_5, y_1, \dots, y_5]$.

However, remaining inside of a particular $\mathbb{F}_2[x_1, \dots, x_n, y_1, \dots, y_n]$, Theorem 5.5 describes the set of squarefree monomial ideals that are canonical forms of neural ideals.

Corollary 5.8 *Let $\mathfrak{a}$ be a polarized neural ideal such that any pair of generators that share the index i share at least 2 indices. Then recomposing $\mathfrak{a}$ with respect to the index i returns $\mathfrak{a}$.*

If each pair of generators that share any index share at least 2 indices, then recomposing $\mathfrak{a}$ with respect to any set of indices returns $\mathfrak{a}$.

Proof This follows from the proof of Theorem 5.5. □

Corollary 5.9 *Let $\mathfrak{a}$ be a polarized neural ideal. In order to compute the canonical form of $\mathfrak{a}$, it suffices to recompose $\mathfrak{a}$ with respect to the indices $i_1, \dots, i_t$ such that for each $1 \leq j \leq t$, some pair of generators of $\mathfrak{a}$ shares only the index i_j.*

Proof By Lemma 4.17, recomposing

$$\mathfrak{a} = (x_i g_{11}, \dots, x_i g_{1k_1}, y_i g_{21}, \dots, y_i g_{2k_2}, g_{31}, \dots, g_{3k_3})$$

with respect to the index i returns

$$\mathfrak{a}' = (x_i g_{11}, \dots, x_i g_{1k_1}, y_i g_{21}, \dots, y_i g_{2k_2}, [g_{1j_1} g_{2j_2}]_{1 \leq j_1 \leq k_2, 1 \leq j_2 \leq k_2}, g_{31}, \dots, g_{3k_3}).$$

As in the proof of Theorem 5.5, if each pair $x_i g_{1j_1}, y_i g_{2j_2}$ sharing the index i also shares an additional index, every $[g_{1j_1} g_{2j_2}]$ can be removed by Proposition 4.19, and we get $\mathfrak{a}$ back. □

We can now give a new algorithm for computing the canonical form, which avoids computing a primary decomposition. Since this algorithm starts from a neural ideal, this algorithm will be more useful to mathematicians studying neural ideals with an eye towards future results classifying neural codes. Those beginning from a neural code will likely prefer the iterative algorithm of [12].

Algorithm 5.10 Begin with a list of generators $L = \{g_1, \ldots, g_k\}$ for a polarized neural ideal $\mathfrak{a} \subseteq \mathbb{F}_2[x_1, \ldots, x_n, y_1, \ldots, y_n]$.

(1) Set $i = 1$.
(2) For each pair of generators g_{j_1}, g_{j_2} that shares only the index i, add a generator $\frac{[g_{j_1} g_{j_2}]}{x_{i_1} y_{i_1}}$ to a new list M.
(3) Join the lists L and M.
(4) Increment i and repeat steps 2 and 3 until $i = n + 1$.
(5) Remove any remaining generators divisible by $x_i y_i$ for any $1 \leq i \leq n$ and any generators divisible by another generator of smaller degree.

Remark 5.11 Note that in Step 4, it is only necessary to remove generators divisible by $x_i y_i$ if $g_1, \ldots, g_k$ included such a generator. This algorithm will not otherwise produce generators divisible by $x_i y_i$ as if g_{j_1} and g_{j_2} share only the index i_1, $\frac{[g_{j_1} g_{j_2}]}{x_{i_1} y_{i_1}}$ will not be divisible by $x_i y_i$ for any i.

Remark 5.12 The generators $\frac{[g_{j_1} g_{j_2}]}{x_{i_1} y_{i_1}}$ that we add relate directly to the RF structure of the corresponding neural code: in Remark 4.18 we saw the relations corresponding to the generators g_{j_1} and g_{j_2} imply the relation corresponding to the generator $\frac{[g_{j_1} g_{j_2}]}{x_{i_1} y_{i_1}}$ must also hold.

Proof By Corollary 5.9, it suffices to recompose $\mathfrak{a}$ with respect to the indices i such that some pair of generators g_{j_1} and g_{j_2} of $\mathfrak{a}$ share only the index i. This is the list made in Step 1. Performing Step 2 recomposes $\mathfrak{a}$ with respect to the index i by Lemma 4.17, with one change: any new generator coming from a pair of generators g_{j_1}, g_{j_2} that share more than one index is skipped, as by the proof of Theorem 5.5 they would be removed in Step 5 of Algorithm 3.2 and by Proposition 4.19 they may be removed at any time. Hence after Step 3, $\mathfrak{a}$ is recomposed. Step 4 removes any remaining generators divisible by $x_i y_i$ or by another generator that have not been removed already, completing Steps 5 and 6 of Algorithm 3.2. □

Remark 5.13 We implemented Algorithms 5.10, 3.2, and a polarized version the iterative algorithm from [12] in Macaulay2 [6] to compare how long they took on examples. We include data from a few initial examples here, and plan to include more extensive speed testing data and complexity analysis in an upcoming Macaulay2 package and associated paper.

All times are listed in seconds. If we interrupted a computation before it completed, we list $> t$, where t is the number of seconds at which we interrupted the code.

List of ideals:

(1) $I_1 = (x_1 x_2, x_3 y_1)$

(2) $I_2 = (x_1x_3, x_2y_1, x_2x_3)$
(3) $I_3 = (x_1x_4, x_2x_3y_1, x_4y_2)$
(4) $I_4 = (x_1y_2, x_2y_3, x_4y_3)$
(5) $I_5 = (x_3y_1, x_1y_2, y_2y_4, x_2x_4y_3, x_3x_4y_1y_2, x_2x_3x_4y_1, x_1x_3x_4y_2)$
(6) $I_6 = (x_1x_2x_4, x_3y_1y_2, x_4y_5, x_4x_5x_6x_7)$
(7) $I_7 = (x_1x_2, x_3y_4, x_6y_7, x_2x_4x_7, x_6y_1y_2, x_2x_3x_5x_6y_7)$

Ideal	Number of neurons	3.2	5.10
I_1	3	0.0648534	0.00192403
I_2	3	0.061073	0.00157261
I_3	4	0.143508	0.00402159
I_4	4	0.0800547	0.00117581
I_5	4	0.367402	0.0249538
I_6	7	>173.618	0.00847768
I_7	7	94.8539	0.0130173

Based on these initial results, we are optimistic that Algorithm 5.10 outperforms Algorithm 3.2 in general, making it a better choice for computing the canonical form of a neural ideal. We also find Algorithm 5.10 significantly easier to use when computing small examples by hand than Algorithm 3.2.

List of codes:

(1) $C_1 = \{100, 101, 001, 000, 110\}$
(2) $C_2 = \{1001, 0110, 0000, 0111, 0010\}$
(3) $C_3 = \{0000000, 1000000, 0110000, 0100000, 0000011, 0000101\}$
(4) $C_4 = \{0000000, 1000000, 0110000, 0100000, 0000011, 0000101, 1000001, 1100010\}$

Code	Number of neurons	3.2	5.10	Iterative
C_1	3	0.0539101	0.00636247	0.0323265
C_2	4	0.372083	0.0387233	0.0830163
C_3	7	38.0215	> 157.579	0.107555
C_4	7	>368.461	>189.821	0.097687

However, when going from a neural code on more than 3–4 neurons straight to its canonical form, the iterative algorithm from [12] appears significantly faster than computing the neural ideal as in [4] and then performing Algorithm 5.10 (or Algorithm 3.2, as seen in [12]) to get the canonical form. On two 7-neuron examples we tried, we interrupted our algorithm after some time, but the iterative algorithm finished in under .1 s.

6 Generic Canonical Forms

The results of this section give us a way to compute a single almost canonical form for a set of neural ideals that have identical patterns of shared indices among their generators (e.g. (x_1x_2, x_3y_1) and $(x_1x_2x_3, x_3y_1)$). We do this by passing to an extension ring, computing a “generic" canonical form, and then returning to our original ring. These generic canonical forms create families of canonical forms.

Notation 6.1 Let $g_1, \dots, g_k \in S$ be squarefree monomials sharing no indices. We define an extension ring $P = S[z_1, \dots, z_k]$ and a map $\pi : P \to S$ sending $z_j \mapsto g_j$. Then $S \cong P/(z_j - g_j)$.

We compute an analogue of the canonical form for squarefree monomial ideals in P by modifying Algorithm 3.2 as follows:

Algorithm 6.2 Let $P = S[z_1, \dots, z_k]$.

(1) Start with a squarefree monomial ideal $\mathfrak{a} = (f_1, \dots, f_m)$ in P, where $f_t = \prod_{i\in\sigma} x_i \prod_{i\in\tau} y_i \prod_{i\in\upsilon} z_i$.
(2) Compute the primary decomposition of $\mathfrak{a}$. The ideals $p_1, \dots, p_s$ in the primary decomposition will all be generated by a subset of $(x_1, \dots, x_n, y_1, \dots, y_n, z_1, \dots, z_k)$.
(3) Impose the relation $x_i + y_i = 1$, and as a result remove any ideal in the primary decomposition containing both x_i and y_i for any $1 \le i \le n$.
(4) Compute $[h_1 \cdots h_s]$ for every set of choices of h_ℓ a generator of p_ℓ. (That is, intersect the remaining prime ideals.)
(5) Impose the relations $x_i y_i = 0$, and remove any monomials with a factor of $x_i y_i$ for some $1 \le i \le n$.
(6) Remove any remaining products that are multiples of a product of lower degree.

Definition 6.3 We say that an ideal that is the result of Algorithm 6.2 is in *generic canonical form*. As we did in S, we will refer to an ideal in P that has had Steps 1–4 of Algorithm 6.2 applied to it as *generically recomposed*, and an ideal that has had Steps 1–5 of Algorithm 6.2 as being in *generic almost canonical form*.

As with their non-generic version (see Remark 4.20), we refer to an ideal that only requires Steps 5 and 6 (resp. Step 6) of Algorithm 6.2 to find the generic canonical form as generically recomposed (resp. in generic almost canonical form), even though these ideals may not agree with the results of Step 4 (resp. Step 5) of Algorithm 6.2.

Lemma 6.4 *Let $b_1, \dots, b_k, g_1, \dots, g_k \in S$ be squarefree monomials such that the g_j share no indices with the $b_j g_j$. In recomposing $\mathfrak{a} = (b_1g_1, \dots, b_kg_k)$, it is not necessary to split the g_j into their factors.*

Proof We proceed by induction on k. We noted in Remark 5.1 that neural ideals with one generator are in canonical form, so we begin with $k = 2$. In this case we have $(b_1g_1, b_2g_2) = (b_1, b_2) \cap (b_1, g_2) \cap (g_1, b_2) \cap (g_1, g_2)$. By Lemma 4.15, since no g_j shares any index with any $b_j g_j$, the second, third, and fourth pieces are already recomposed, and by Lemma 4.10, recomposed $\mathfrak{a}$ is equal to the intersection of the recomposed components on the right hand side. This proves the case $k = 2$.

Now assume $k > 2$. We have

$$(b_1g_1, \ldots, b_kg_k) = (b_1, b_2g_2, \ldots, b_kg_k) \cap (g_1, b_2g_2, \ldots, b_kg_k).$$

By Lemma 4.15, the recomposed second piece is equal to g_1 plus the recomposed $(b_2g_2, \ldots, b_kg_k)$. By the induction hypothesis, we do not need to split the g_j into their factors to compute this. We continue to split the first piece:

$$(b_1, b_2g_2, \ldots, b_kg_k) = (b_1, b_2, b_3g_3, \ldots, b_kg_k) \cap (b_1, g_2, b_3g_3, \ldots, b_kg_k).$$

As before, we do not need to split the g_j into their factors to recompose the second piece. Continuing in this way, we eventually get

$$(b_1, b_2, \ldots, b_{k-1}, b_kg_k) = (b_1, \ldots, b_{k-1}, b_k) \cap (b_1, \ldots, b_{k-1}, g_k).$$

The first piece has no g_k, and as before, we do not need to split g_k into its factors to recompose the second piece. □

Notation 6.5 Let $(b_1z_1, \ldots, b_kz_k)$ be a squarefree monomial ideal of P. When referring to the generically recomposed, almost canonical, or canonical form of $(b_1z_1, \ldots, b_kz_k)$, we will denote it

$$(f(z_1, \ldots, z_k)) := (f_1(z_1, \ldots, z_k), \ldots f_\ell(z_1, \ldots, z_k))$$

for some ℓ where each generator $f_j(z_1, \ldots, z_k) \in P$ is a monomial. This allows us to view the generically recomposed (resp. almost canonical or canonical) form as a function of the z_j that is determined by the b_j.

Proposition 6.6 *Let $b_1, \ldots, b_k, g_1, \ldots, g_k \in S$ be squarefree monomials such that the g_j share no indices with the b_jg_j. Let $P = S[z_1, \ldots, z_k]$ and $\pi : P \to S$ be the homomorphism sending S to itself and $z_j \mapsto g_j$. If the generically recomposed form of $(b_1z_1, \ldots, b_kz_k)$ in P is given by $(f(z_1, \ldots, z_k))$, then the recomposed form of $(b_1g_1, \ldots, b_kg_k)$ in S is given by*

$$[\pi(f(z_1, \ldots, z_k))] := ([f_1(g_1, \ldots, g_k)], \ldots, [f_\ell(g_1, \ldots, g_k)]).$$

Further, if x_iy_i divides $[f_j(g_1, \ldots, g_k)]$ for some $1 \le i \le n$ and $1 \le j \le \ell$, then x_iy_i divides $f_j(z_1, \ldots, z_k)$. Hence the canonical form of $(b_1g_1, \ldots, b_kg_k)$ in S may be computed by computing the generic almost canonical form of $(b_1z_1, \ldots, b_kz_k)$, applying π and replacing any x_i^c for $c > 1$ with x_i, and then applying Step 6 of Algorithm 3.2.

In particular, if we know the canonical form of $(b_1g_1, \ldots, b_kg_k)$ (where the g_j still share no indices with the b_jg_j), and $g \mid g_j$, we can find the canonical form of $(b_1g_1, \ldots, b_{j-1}g_{j-1}, b_jg, b_{j+1}g_{j+1}, \ldots, b_kg_k)$ by computing the canonical form of $(b_1g_1, \ldots, b_kg_k)$, replacing g_j by g, and then applying Step 6 of Algorithm 3.2 again.

Proof Say that the generically recomposed $(b_1z_1, \dots, b_kz_k)$ in P is given by

$$(f(z_1, \dots, z_k)) := (f_1(z_1, \dots, z_k), \dots, f_\ell(z_1, \dots, z_k)),$$

where each $f_s(z_1, \dots, z_k) \in P$. Applying π, we get

$$(f_1(g_1, \dots, g_k), \dots, f_\ell(g_1, \dots, g_k)).$$

We claim that after replacing products with LCMs this is equal to the recomposed $(b_1g_1, \dots, b_kg_k)$. By Lemma 6.4, since the g_j contain no x_i or y_i with shared indices, we can leave the g_j alone when decomposing $(b_1g_1, \dots, b_kg_k)$. Hence π applied to the primary decomposition of $(b_1z_1, \dots, b_kz_k)$ gives us a sufficient decomposition of $(b_1g_1, \dots, b_kg_k)$ to recompose the latter. Since $\pi(x_i) = x_i$ and $\pi(y_i) = y_i$, the components we remove in Step 3 of Algorithm 3.2 in S are equal to π applied to the components we remove in Step 3 of Algorithm 6.2. The only potential difference is in Step 4, and this is resolved by replacing products with LCM's.

Since the g_j contain no indices shared with the b_jg_j (i.e., if $x_i \mid g_\ell$ for any ℓ, then $y_i \nmid \Pi_j b_jg_j$, and the same with x_i and y_i reversed), this will hold after recomposing $(b_1g_1, \dots, b_kg_k)$ as well. As a result, any generator of the recomposed $(b_1g_1, \dots, b_kg_k)$ divisible by x_iy_i for some $1 \le i \le n$ must be equal to π of a generator of the recomposed $(b_1z_1, \dots, b_kz_k)$ divisible by the same x_iy_i, up to replacing products with LCMs. Applying Step 6 of Algorithm 3.2 to $(f_1(g_1, \dots, g_k), \dots, f_\ell(g_1, \dots, g_k))$ now gives us the canonical form of $(b_1g_1, \dots, b_kg_k)$, as desired.

For the final statement, note that

$$\mathfrak{a} = (b_1g_1, \dots, b_kg_k) \text{ and } \mathfrak{a}' = (b_1g_1, \dots, b_jg, \dots, b_kg_k)$$

have the same generic canonical form, $(f(z_1, \dots, z_k))$. Applying π_1 sending $z_j \mapsto g_j$ and π_2 sending $z_j \mapsto g$, we have effectively replaced g_j by g everywhere it appears. The only possible difference is in Step 6 of Algorithm 3.2. □

Example 6.7 Consider $\mathfrak{a} = (x_1x_2x_4, x_3x_4y_1)$. Set $P = S[z_1, z_2]$, with $\pi(z_1) = x_2x_4$ and $\pi(z_2) = x_3x_4$. Write $\tilde{\mathfrak{a}} = (x_1z_1, y_1z_2) \subseteq P$. Using the proof of Theorem 5.2, this ideal has canonical form (x_1z_1, y_1z_2, z_1z_2) in P. Passing back to S, we get $(x_1x_2x_4, x_3x_4y_1, x_2x_3x_4)$. In this case, no generator is a multiple of another, so we are done. This computation agrees with the result of Theorem 5.2.

Further, by the last statement of the Theorem, the canonical form of $\mathfrak{b} = (x_1x_2, x_3x_4y_1)$ is $(x_1x_2, x_3x_4y_1, x_2x_3x_4)$ and the canonical form of $\mathfrak{c} = (x_1, x_3x_4y_1)$ is (x_1, x_3x_4). All of these agree with the result of Theorem 5.2.

Theorem 6.8 *Suppose that the generic almost canonical form of the ideal* $(b_1z_1, \dots, b_kz_k)$ *in* $P = S[z_1, \dots, z_k]$ *is*

$$(f(z_1, \dots, z_k)) := (f_1(z_1, \dots, z_k), \dots, f_\ell(z_1, \dots, z_k)).$$

Let $\mathfrak{a} = (b_1g_{11}, \dots, b_1g_{1t_1}, b_2g_{21}, \dots, b_2g_{2t_2}, \dots, b_kg_{k1}, \dots, b_kg_{kt_k})$, *where for any index* i *shared by the generators of* $\mathfrak{a}$, $x_i, y_i \nmid \prod g_{j_1j_2}$. *Then the almost canonical form* *of* $\mathfrak{a}$ *is*

$$
\begin{aligned}
&([g_{1j_1}g_{2j_2}\cdots g_{kj_k}f(1,\ldots,1)],\\
&[g_{2j_2}\cdots g_{kj_k}f(g_{1t_1},1,\ldots,1)],\ldots,[g_{1j_1}\cdots g_{(k-1)j_{k-1}}f(1,\ldots,1,g_{kt_k})],\\
&\vdots\\
&[g_{kj_k}f(g_{1t_1},\ldots,g_{(k-1)t_{k-1}},1)],\ldots,[g_{1j_1}f(1,g_{2t_2},\ldots,g_{kt_k})],\\
&[f(g_{1t_1},g_{2t_2},\ldots,g_{kt_k})]),
\end{aligned}
$$

where j_i *ranges over the integers* $1, 2, \ldots, t_i - 1$ *and all products have been replaced with LCM's.*

Remark 6.9 The statement of Theorem 6.8 is asymmetrical in the g_{ij_i}, with only the g_{it_i} acting as input to the f's. This is a result of the particular decomposition used in the proof, and the result works equally well if we fix for each $1 \le i \le k$ a choice of $1 \le s_i \le t_i$, replace every g_{it_i} with g_{is_i}, and let j_i range over the integers $1, 2, \ldots, \widehat{s_i}, \ldots, t_i$.

We give an example before proving Theorem 6.8 in order to illustrate the notation.

Example 6.10 We compute the almost canonical form of

$$(x_1g_{11}, x_1g_{12}, y_1g_{21}, y_1g_{22}, y_1g_{23}),$$

where the $g_{j_1j_2}$ contain no shared indices. Continuing from Example 6.7, the generic canonical form of (x_1g_1, y_1g_2) is (x_1z_1, y_1z_2, z_1z_2). Here

$$f_1(z_1, z_2) = x_1z_1, \qquad f_2(z_1, z_2) = y_1z_2, \qquad f_3(z_1, z_2) = z_1z_2.$$

In this example, $j_1 = 1$ and $j_2 \in \{1, 2\}$. Applying Theorem 6.8, we have

$$g_{1j_1}g_{2j_2}f_1(1,1) = g_{1j_1}g_{2j_2}x_1, \qquad g_{1j_1}g_{2j_2}f_2(1,1) = g_{1j_1}g_{2j_2}y_1,$$
$$g_{1j_1}g_{2j_2}f_3(1,1) = g_{1j_1}g_{2j_2},$$

so we will only need generators of the third type, which are $[g_{11}g_{21}]$ and $[g_{11}g_{22}]$. Similarly,

$$g_{2j_2}f_1(g_{12},1) = g_{2j_2}x_1g_{12} \quad g_{2j_2}f_2(g_{12},1) = g_{2j_2}y_1 \quad g_{2j_2}f_3(g_{12},1) = g_{2j_2}g_{12},$$

from which we get generators $[g_{12}g_{21}]$, $[g_{12}g_{22}]$, y_1g_{21}, and y_1g_{22},

$$g_{1j_1}f_1(1,g_{23}) = g_{1j_1}x_1 \quad g_{1j_1}f_2(1,g_{23}) = g_{1j_1}y_1g_{23} \quad g_{1j_1}f_3(1,g_{23}) = g_{1j_1}g_{23},$$

from which we get generators x_1g_{11} and $[g_{11}g_{23}]$, and

$$f_1(g_{12},g_{23}) = x_1g_{12} \qquad f_2(g_{12},g_{23}) = y_1g_{23} \qquad f_3(g_{12},g_{23}) = g_{12}g_{23},$$

from which we get generators x_1g_{12}, y_1g_{23}, and $[g_{12}g_{23}]$. Hence by Theorem 6.8, the almost canonical form of $\mathfrak{a} = (x_1g_{11}, x_1g_{12}, y_1g_{21}, y_1g_{22}, y_1g_{23})$ is

$$([g_{11}g_{21}], [g_{11}g_{22}], [g_{11}g_{23}], [g_{12}g_{21}], [g_{12}g_{22}], [g_{11}g_{23}],\\ x_1g_{11}, x_1g_{12}, y_1g_{21}, y_1g_{22}, y_1g_{23}).$$

This matches the result of Lemma 4.17. In this situation of only 1 shared index, using Lemma 4.17 might be easier, but as the number of shared indices increases, Theorem 6.8 becomes more useful.

Proof We prove this by induction on $\sum_{i=1}^{k} t_i$. If $t_1 = \ldots = t_k = 1$, we are done by Proposition 6.6. Now suppose $t_i > 1$ for at least one $1 \leq i \leq k$. Without loss of generality, assume $t_1 > 1$. We have

$$(b_1g_{11},\ldots, b_1g_{1t_1}, b_2g_{21},\ldots) = (b_1, b_2g_{21},\ldots) \cap (g_{11}, b_1g_{12},\ldots, b_1g_{1t_1}, b_2g_{21},\ldots).$$

By Lemma 4.10 and Proposition 4.19, the almost canonical form of $\mathfrak{a}$ is equal to the result of applying Step 5 of Algorithm 3.2 to the intersection of the almost canonical forms of the two components. By the induction hypothesis and Proposition 6.6, the almost canonical form of the first piece is

$$\begin{aligned}&([g_{2j_2}\cdots g_{kj_k} f(1,\ldots,1)],\\ &[g_{3j_3}\cdots g_{kj_k} f(1, g_{2t_2}, 1,\ldots,1)],\ldots,[g_{2j_2}\cdots g_{(k-1)j_{k-1}} f(1,\ldots,1, g_{kt_k})],\\ &\vdots\\ &[g_{kj_k} f(1, g_{2t_2},\ldots, g_{(k-1)t_{k-1}}, 1)],\ldots,[g_{2j_2} f(1, 1, g_{3t_3},\ldots, g_{kt_k})],\\ &[f(1, g_{2t_2},\ldots, g_{kt_k})]),\end{aligned}$$

where j_i ranges over the integers $1, 2, \ldots, t_i$ for $2 \leq i \leq k$. By the induction hypothesis and Lemma 4.15, the almost canonical form of the second piece is

$$\begin{aligned}&(g_{11}, [g_{1j_1}g_{2j_2}\cdots g_{kj_k} f(1,\ldots,1)],\\ &[g_{2j_2}\cdots g_{kj_k} f(g_{1t_1}, 1,\ldots,1)],\ldots,[g_{1j_1}\cdots g_{(k-1)j_{k-1}} f(1,\ldots,1, g_{kt_k})],\\ &\vdots\\ &[g_{kj_k} f(g_{1t_1},\ldots, g_{(k-1)t_{k-1}}, 1)],\ldots,[g_{1j_1} f(1, g_{2t_2},\ldots, g_{kt_k})],\\ &[f(g_{1t_1}, g_{2t_2},\ldots, g_{kt_k})]),\end{aligned}$$

where j_i ranges over the integers $1, 2, \ldots, t_i - 1$ for $2 \leq i \leq k$ and j_1 ranges over the integers $2, \ldots, t_1 - 1$.

We claim that

$$\begin{aligned}&[[f(h_1,\ldots,h_{i-1}, 1, h_{i+1},\ldots,h_k)][f(h_1,\ldots,h_{i-1}, h_i, h_{i+1},\ldots,h_k)]]\\ &\quad = [f(h_1,\ldots,h_{i-1}, h_i, h_{i+1},\ldots,h_k)].\end{aligned}$$

To see this, note that for any s such that $f_s(z_1, \ldots, z_k)$ has no factor of z_i, the two components agree, and for any s such that $f_s(z_1, \ldots, z_k)$ has a factor of z_i, say $f_s(z_1, \ldots, z_k) = z_i h$, the first component gives h and the second component gives $h_i h$, so their lcm is $h_i h = f_s(h_1, \ldots, h_{i-1}, h_i, h_{i+1}, \ldots, h_k)$.

Using this claim to intersect the two pieces, the proof is complete. □

Remark 6.11 In computing the canonical form of $\mathfrak{a}$, if $f_s(z_1, \ldots, z_k)$ contains factors of, without loss of generality, $z_1, \ldots, z_p$ but not $z_{p+1}, \ldots, z_k$, then

$$\begin{aligned} f_s(g_{1t_1}, \ldots, g_{pt_p}, 1, \ldots, 1) &= f_s(g_{1t_1}, \ldots, g_{pt_p}, g_{(p+1)t_{p+1}}, 1, \ldots, 1) \\ &= \ldots = f_s(g_{1t_1}, \ldots, g_{kt_k}). \end{aligned}$$

(See the proof of Theorem 6.8 for justification of this fact.) So any term with a factor of $f_s(g_{1t_1}, \ldots, g_{pg_p}, 1, \ldots, 1)$ or any of the intermediate lines is a multiple of $f_s(g_{1t_1}, \ldots, g_{kt_k})$ and will be removed in the canonical form.

We end by computing the canonical forms of a class of polarized neural ideals to illustrate the benefit of the generic approach.

Theorem 6.12 *The polarized neural ideal*

$$\mathfrak{a} = (x_1 g_1, x_2 y_1 g_2, x_3 y_2 g_3, \ldots, x_{k-1} y_{k-2} g_{k-1}, y_{k-1} g_k),$$

where $k > 2$ and the only shared indices are the x_i and y_i shown, is not in canonical form. The almost canonical form of $\mathfrak{a}$ is

$$\begin{aligned} (&x_1 g_1, x_2[g_1 g_2], x_3[g_1 g_2 g_3], \ldots, x_{k-1}[g_1 \cdots g_{k-1}], [g_1 \cdots g_k], \\ &x_2 y_1 g_2, x_3 y_1[g_2 g_3], \ldots, x_{k-1} y_1[g_2 \cdots g_{k-1}], y_1[g_2 \cdots g_k], \\ &x_3 y_2 g_3, \ldots, x_{k-1} y_2[g_3 \cdots g_{k-1}], y_2[g_3 \cdots g_k], \\ &\vdots \\ &x_{k-1} y_{k-2} g_{k-1}, y_{k-2}[g_{k-1} g_k], \\ &y_{k-1} g_k). \end{aligned}$$

If for any $1 \leq t \leq t' < k$, $[g_t \cdots g_k] \neq [g_t \cdots g_{t'}]$, and for any $1 < s' \leq s \leq k$, $[g_1 \cdots g_s] \neq [g_{s'} \cdots g_s]$, then this is the canonical form of the ideal.

If for some $1 \leq s' \leq s \leq t \leq t' \leq k$, $[g_1 \cdots g_s] = [g_{s'} \cdots g_s]$ and $[g_t \cdots g_k] = [g_t \cdots g_{t'}]$, where s is minimal, s' is maximal, t is maximal, and t' is minimal with respect to these properties, then the canonical form of $\mathfrak{a}$ is

$$\begin{aligned} (&x_1 g_1, x_2[g_1 g_2], \ldots, x_{s-1}[g_1 \cdots g_{s-1}], x_s[g_{s'} \cdots g_s], \ldots, x_{t'-1} \\ &[g_{s'} \cdots g_{t'-1}], [g_{s'} \cdots g_{t'}], \\ &x_2 y_1 g_2, x_3 y_1[g_2 g_3], \ldots, x_{s-1} y_1[g_2 \cdots g_{s-1}], \\ &\vdots \end{aligned}$$

$$
\begin{aligned}
&x_{s'}y_{s'-1}g_{s'}, \dots, x_{s-1}y_{s'-1}[g_{s'}\cdots g_{s-1}],\\
&x_{s'+1}y_{s'}g_{s'+1}, x_{s'+2}y_{s'}[g_{s'+1}g_{s'+2}], \dots, x_{t'-1}y_{s'}[g_{s'+1}\cdots g_{t'-1}], y_{s'}[g_{s'+1}\cdots g_{t'}],\\
&\vdots\\
&x_t y_{t-1}g_t, \dots, x_{t'-1}y_{t-1}[g_t\cdots g_{t'-1}], y_{t-1}[g_t\cdots g_{t'}],\\
&x_{t+1}y_t g_{t+1}, \dots, x_{k-1}y_t[g_{t+1}\cdots g_{k-1}], y_t[g_{t+1}\cdots g_k],\\
&\vdots\\
&x_{k-1}y_{k-2}g_{k-1}, y_{k-2}[g_{k-1}g_k],\\
&y_{k-1}g_k).
\end{aligned}
$$

Remark 6.13 If $k = 2$, we are in the case of Theorem 5.2, so we skip that case here. If $k = 1$, there are no shared indices and so the ideal is in canonical form by Lemma 4.15.

Notice that the canonical form of $\mathfrak{a}$ usually has more generators than $\mathfrak{a}$, often many more. The leftmost column of the canonical form above contains the original generators of the ideal, and all other generators are additional.

Proof We prove this by induction, beginning with the case $k = 3$. We decompose the ideal $(x_1g_1, x_2y_1g_2, y_2g_3)$, removing any component that contains a pair (x_i, y_i) as generators as we go by Lemma 4.5:

$$
\begin{aligned}
&(x_1g_1, x_2y_1g_2, y_2g_3) = (x_1, x_2y_1g_2, y_2g_3) \cap (g_1, x_2y_1g_2, y_2g_3)\\
&= (x_1, y_1, y_2g_3) \cap (x_1, x_2g_2, y_2g_3) \cap (g_1, x_2, y_2g_3) \cap (g_1, y_1g_2, y_2g_3)\\
&\quad\to (x_1, x_2, y_2g_3) \cap (x_1, g_2, y_2g_3) \cap (g_1, x_2, y_2) \cap (g_1, x_2, g_3) \cap (g_1, y_1g_2, y_2g_3)\\
&\quad\to (x_1, x_2, y_2) \cap (x_1, x_2, g_3) \cap (x_1, g_2, y_2g_3) \cap (g_1, x_2, g_3) \cap (g_1, y_1g_2, y_2g_3)\\
&\quad\to (x_1, x_2, g_3) \cap (x_1, g_2, y_2g_3) \cap (g_1, x_2, g_3) \cap (g_1, y_1g_2, y_2g_3).
\end{aligned}
$$

By Lemma 4.15, since none of the remaining pieces have any shared indices, each piece is recomposed. By Lemma 4.10, to get the recomposed $\mathfrak{a}$, it suffices to intersect these ideals. Intersecting these ideals and removing components divisible by x_iy_i or by another component as in Proposition 4.19, we are left with the almost canonical form$(x_1g_1, x_2g_1g_2, g_1g_2g_3, x_2y_1g_2, y_1g_2g_3, y_2g_3)$.

Now assume $k > 3$. We have

$$
\begin{aligned}
&(x_1g_1, x_2y_1g_2, \dots, x_{k-1}y_{k-2}g_{k-1}, y_{k-1}g_k)\\
&\quad= (x_1, y_1, x_3y_2g_3, \dots, x_{k-1}y_{k-2}g_{k-1}, y_{k-1}g_k)\\
&\qquad\cap (x_1, x_2g_2, x_3y_2g_3, \dots, y_{k-1}g_k) \cap (g_1, x_2y_1g_2, x_3y_2g_3, \dots, y_{k-1}g_k).
\end{aligned}
$$

We remove the first piece, which contains the pair (x_1, y_1). By Lemma 4.10, the recomposed $\mathfrak{a}$ is equal to the intersection of the recomposed second piece with the recomposed third piece. By the induction hypothesis, the generically recomposed second piece is

$$
\begin{aligned}
(&x_1, x_2g_2, x_3g_2g_3, \ldots, x_{k-1}g_2\cdots g_{k-1}, g_2\cdots g_k,\\
&x_3y_2g_3, \ldots, x_{k-1}y_2g_3\cdots g_{k-1}, y_2g_3\cdots g_k,\\
&\vdots\\
&x_{k-1}y_{k-2}g_{k-1}, y_{k-2}g_{k-1}g_k,\\
&y_{k-1}g_k).
\end{aligned}
$$

Viewing y_1g_2 as a piece sharing no indices with other generators, the recomposed 3rd piece is

$$
\begin{aligned}
(&g_1, x_2y_1g_2, x_3y_1g_2g_3, \ldots, x_{k-1}y_1g_2\cdots g_{k-1}, y_1g_2\cdots g_k,\\
&x_3y_2g_3, \ldots, x_{k-1}y_2g_3\cdots g_{k-1}, y_2g_3\cdots g_k,\\
&\vdots\\
&x_{k-1}y_{k-2}g_{k-1}, y_{k-2}g_{k-1}g_k,\\
&y_{k-1}g_k).
\end{aligned}
$$

Intersecting the two pieces and removing generators that are divisible by x_iy_i, we get

$$
\begin{aligned}
(&x_1g_1, x_2g_1g_2, \ldots, x_{k-1}g_1\cdots g_{k-1}, g_1\cdots g_k,\\
&x_2y_1g_2, x_3y_1g_2g_3, \ldots, x_{k-1}y_1g_2\cdots g_{k-1}, y_1g_2\cdots g_k,\\
&x_3y_2g_3, \ldots, x_{k-1}y_2g_3\cdots g_{k-1}g, y_2g_3\cdots g_k,\\
&\vdots\\
&x_{k-1}y_{k-2}g_{k-1}, y_{k-2}g_{k-1}g_k,\\
&y_{k-1}g_k).
\end{aligned}
$$

Since by assumption, the g_j share no indices, this is the almost canonical form of $\mathfrak{a}$.

If $[g_1\cdots g_s] \neq [g_{s'}\cdots g_s]$ for any $1 < s' \leq s \leq k$ and $[g_t\cdots g_k] \neq [g_t\cdots g_{t'}]$ for any $1 \leq t \leq t' < k$, it is not possible for any generator to be a multiple of another generator, so we have found the canonical form. Otherwise, we remove generators that are a multiple of another generator to get the result.

Regardless of the value of $[g_1\cdots g_k]$, it will always be a generator of the ideal. Since $[g_1\cdots g_k]$ has no factors of x_i or y_i for $1 \leq i \leq k-1$, whereas every original generator of $\mathfrak{a}$ has a factor of x_i or of y_i, the canonical form is distinct from the original list of generators. $\square$

Remark 6.14 In the case of Theorem 6.12, we note that the canonical form of $\mathfrak{a}$ is never equal to the original ideal. This is best illustrated by noting that $[g_1\cdots g_k]$ is always added as a generator and is not divisible by any of the original generators since it isn't divisible by any of $x_1, \ldots, x_{k-1}, y_1, \ldots, y_{k-1}$.

Example 6.15 Consider the polarized neural ideal $\mathfrak{a} = (x_1g_1, x_2y_1g_2, y_2g_3)$, where the g_j contain no shared indices. By Theorem 6.12, the almost canonical form of this ideal is $(x_1g_1, x_2[g_1g_2], [g_1g_2g_3], x_2y_1g_2, y_1[g_2g_3], y_2g_3)$. Consider $g_1 = x_3$, $g_2 = x_4$, and $g_3 = x_3x_4$, so that $[g_1g_2g_3] = g_3$. Substituting our values of g_1, g_2, and g_3 and removing generators that are a multiple of another generator, we are left with $(x_1x_3, x_2y_1x_4, x_3x_4)$. This is a degenerate case like in parts of Theorem 5.2, (4)–instead of adding generators to get the canonical form we change a generator.

If instead we set $g_1 = g_2 = g_3 = x_3$, we find every generator is divisible by x_3 thereby making the almost canonical form (x_3). This is similar to the last case of Theorem 5.2.

Acknowledgements We thank Jack Jeffries and Nora Youngs for helpful conversations. In particular, Jack Jeffries inspired Remark 4.18. In addition, Ray Karpman gave extremely helpful combinatorial advice on an earlier version of this paper. We also thank the anonymous referee for comments that improved the exposition of this paper.

Data availability Not applicable.

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no Conflict of interest.

References

1. Chen, A., Frick, F., Shiu, A.: Neural codes, decidability, and a new local obstruction to convexity. SIAM J. Appl. Algebra Geom. **3**(1), 44–66 (2019)
2. Curto, C., Gross, E., Jeffries, J., Morrison, K., Omar, M., Rosen, Z., Shiu, A., Youngs, N.: What makes a neural code convex? SIAM J. Appl. Algebra Geom. **1**(1), 222–238 (2017)
3. Curto, C., Gross, E., Jeffries, J., Morrison, K., Rosen, Z., Shiu, A., Youngs, N.: Algebraic signatures of convex and non-convex codes. J. Pure Appl. Algebra **223**(9), 3919–3940 (2019)
4. Curto, C., Itskov, V., Veliz-Cuba, A., Youngs, N.: The neural ring: an algebraic tool for analyzing the intrinsic structure of neural codes. Bull. Math. Biol. **75**(9), 1571–1611 (2013)
5. de Perez, A.R., Matusevich, L.F., Shiu, A.: Neural codes and the factor complex. Adv. Appl. Math. **114**, 101977, 16 (2020)
6. Grayson, D.R., Stillman, M.E.: Macaulay2, a software system for research in algebraic geometry. Available at http://www2.macaulay2.com
7. Güntürkün, S., Jeffries, J., Sun, J.: Polarization of neural rings. J. Algebra Appl. **19**(8), 2050146, 17 (2020)
8. Jeffs, R.A.: Sunflowers of convex open sets. Adv. Appl. Math. **111**, 101935, 21 (2019)
9. Jeffs, R.A., Omar, M., Suaysom, N., Wachtel, A., Youngs, N.: Sparse neural codes and convexity. Involve **12**(5), 737–754 (2019)
10. Lienkaemper, C., Shiu, A., Woodstock, Z.: Obstructions to convexity in neural codes. Adv. Appl. Math. **85**, 31–59 (2017)
11. Moore, W.F., Rogers, M., Sather-Wagstaff, K.: Monomial ideals and their decompositions. Universitext. Springer, Cham (2015). https://ssather.people.clemson.edu/DOCS/monomial.pdf

12. Petersen, E., Youngs, N., Kruse, R., Miyata, D., Garcia, R., Puente, L.D.G.: Neural ideals in Sagemath. In: Davenport, J.H., Kauers, M., Labahn, G., Urban, J. (eds.) Mathematical Software—ICMS 2018, pp. 182–190. Springer, Cham (2018)