Algebraic Transformations of Convex Codes

Amzi Jeffs

Advisor: Mohamed Omar

Feb 2, 2016
**Neuroscience:** Place cells and neural codes.
Outline

- **Neuroscience:** Place cells and neural codes.
- **Algebra Background:** Neural ideals and the canonical form.
Outline

- **Neuroscience:** Place cells and neural codes.

- **Algebra Background:** Neural ideals and the canonical form.

- **My Thesis Work:** Understanding how codes relate to one another algebraically.
**Biological Motivation**

**Place cells:** Neurons which are active in a particular region of an animal’s environment.

https://upload.wikimedia.org/wikipedia/commons/5/5e/Place_Cell_Spiking_Activity_Example.png
Biological Motivation

How is data on place cells collected?
Biological Motivation

How is data on place cells collected?

[Diagram showing activity of place cells over time]
Biological Motivation

How is data on place cells collected?
Biological Motivation

How is data on place cells collected?

\[ C = \{000, 100, 001, 011, 110, 111\} \]
Mathematical Formulation

Neural codes capture an animal’s response to a stimulus.

We assume that the receptive fields for place cells are open convex sets in Euclidean space.
We associate collections of convex sets to binary codes, and attempt to classify these codes.

**Definition**

Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be a collection of convex open sets. The *code* of $\mathcal{U}$ is

$$C(\mathcal{U}) := \left\{ v \in \{0, 1\}^n \mid \bigcap_{i=1}^{n} U_i \setminus \bigcup_{j=0}^{n} U_j \neq \emptyset \right\}$$

$\mathcal{U} = \{U_1, U_2, U_3\}$

$C(\mathcal{U}) = \{000, 100, 010, 001, 110, 011\}$
Mathematical Formulation

**Definition**

Let $C \subseteq \{0, 1\}^n$ be a code. If there exists a collection of convex open sets $\mathcal{U}$ so that $C = C(\mathcal{U})$ we say that $C$ is *convex*. We call $\mathcal{U}$ a *convex realization* of $C$.

**From last time:** Not all codes are convex!
Mathematical Formulation

Definition
Let \( C \subseteq \{0, 1\}^n \) be a code. If there exists a collection of convex open sets \( \mathcal{U} \) so that \( C = C(\mathcal{U}) \) we say that \( C \) is convex. We call \( \mathcal{U} \) a convex realization of \( C \).

From last time: Not all codes are convex!

Question
How can we detect whether a code \( C \) is convex?
Question: Can we find meaningful criteria that guarantee a code is convex?

Answer: Yes! Simplicial complex codes, intersection complete codes, codes with 11⋯1 in them, and many more!
Classifying Convex Codes

**Question**

*Can we find meaningful criteria that guarantee a code is convex?*

**Answer:** Yes! Simplicial complex codes, intersection complete codes, codes with 11⋯1 in them, and many more!

**A Constructive Approach:** Take a realization $\mathcal{U}$, modify it, and see how that affects $C(\mathcal{U})$. 

![Diagram](image-url)
Restricting a Convex Realization

\[ c(U) = \left\{ 0000, 1000, 0100, 0010, 0001, \\
1100, 0110, 0101, 0111, 1101 \right\} \]

\[ c(U') = \{000, 010, 110, 011\} \]
An Algebraic Approach

We will work in the polynomial ring $\mathbb{F}_2[x_1, \ldots, x_n]$. 

Definition (CIVCY2013)

A pseudomonomial is a polynomial of the form $f^M_i \sigma x_i^M_j \tau^1 \hat{x}_j$ where $\sigma, \tau \sim 1, 2, \ldots, n$ are disjoint.

Example: $x_1 x_2 \hat{1}$ and $\hat{1} x_1 \hat{1} x_5 \hat{1}$ are both pseudomonomials.

$x_1 \hat{1} x_1 \hat{1} x_2$ and $x_3 \hat{2}$ are NOT pseudomonomials.
An Algebraic Approach

We will work in the polynomial ring $\mathbb{F}_2[x_1, \ldots, x_n]$.

**Definition (CIVCY2013)**

A *pseudomonomial* is a polynomial of the form

$$f = \prod_{i \in \sigma} x_i \prod_{j \in \tau} (1 - x_j)$$

where $\sigma, \tau \subseteq \{1, 2, \ldots, n\}$ are disjoint.
An Algebraic Approach

We will work in the polynomial ring $\mathbb{F}_2[x_1, \ldots, x_n]$.

**Definition (CIVCY2013)**

A *pseudomonomial* is a polynomial of the form

$$f = \prod_{i \in \sigma} x_i \prod_{j \in \tau} (1 - x_j)$$

where $\sigma, \tau \subseteq \{1, 2, \ldots, n\}$ are disjoint.

**Example:** $x_1 x_2 (1 - x_3)$ and $(1 - x_1)(1 - x_5)$ are both pseudomonomials.

$x_1 (1 - x_1) x_2$ and $x_2^3$ are NOT pseudomonomials.
For any $f \in \mathbb{F}_2[x_1, \ldots, x_n]$ and $\nu \in \{0, 1\}^n$ we define $f(\nu)$ to be the result of replacing $x_i$ by $\nu_i$, the $i$-th bit of $\nu$.

**Example:** Let $f = x_1x_2(1 - x_3)$ and $\nu = 110$. Then

$$f(\nu) = 1 \times 1 \times (1 - 0) = 1.$$
**Definition (CIVCY2013)**

Let \( v \in \{0, 1\}^n \). Then *indicator pseudomonomial* for \( v \) is

\[
\rho_v := \prod_{v_i=1} x_i \prod_{v_j=0} (1 - x_j).
\]

**Example:** \( \rho_{110} = x_1x_2(1 - x_3) \)
Definition (CIVCY2013)

Let \( v \in \{0, 1\}^n \). Then *indicator pseudomonomial* for \( v \) is

\[
\rho_v := \prod_{v_i=1} x_i \prod_{v_j=0} (1 - x_j).
\]

**Example:** \( \rho_{110} = x_1 x_2 (1 - x_3) \)

Note that \( \rho_v \) is always a pseudomonomial of degree \( n \).
An Algebraic Approach

Definition (CIVCY2013)
Let $v \in \{0, 1\}^n$. Then indicator pseudomonomial for $v$ is

$$\rho_v := \prod_{v_i=1} x_i \prod_{v_j=0} (1 - x_j).$$

Example: $\rho_{110} = x_1 x_2 (1 - x_3)$

Note that $\rho_v$ is always a pseudomonomial of degree $n$.

Proposition
Let $u, v \in \{0, 1\}^n$. Then $\rho_v(u) = 1$ if and only if $u = v$. That is, $\rho_v$ vanishes everywhere in $\{0, 1\}^n$ except for at $v$. 
The Neural Ideal of a Code

**Definition (CIVCY2013)**

Let $\mathcal{C}$ be a code. The *neural ideal* of $\mathcal{C}$ is

$$J_\mathcal{C} := \langle \rho_v \mid v \notin \mathcal{C} \rangle.$$
The Neural Ideal of a Code

**Definition (CIVCY2013)**

Let $\mathcal{C}$ be a code. The *neural ideal* of $\mathcal{C}$ is

$$J_\mathcal{C} := \langle \rho_v \mid v \notin \mathcal{C} \rangle.$$ 

**Proposition**

Let $f \in J_\mathcal{C}$ and $c \in \mathcal{C}$. Then $f(c) = 0$.

**Proof Idea:** $J_\mathcal{C}$ is generated by polynomials which vanish on all of $\mathcal{C}$. 
Theorem

Neural ideals are precisely the ideals generated by pseudomonomials.
Presenting the Neural Ideal

**Theorem**

*Neural ideals are precisely the ideals generated by pseudomonomials.*

**Definition (CIVCY2013)**

Let $J_C$ be a neural ideal. The *canonical form* of $J_C$ is the set of minimal pseudomonomials in $J_C$ with respect to division. Equivalently:

$$CF(J_C) := \{ f \in J_C \mid f \text{ is a PM and no proper divisor of } f \text{ is in } J_C \}.$$
A Concrete Example

\[ x_1 x_2 x_3 \in CF(J_C) \]
A Concrete Example

\[ x_1 x_2 x_3 \in CF(J_C) \]

First Piece of Information:
- This polynomial vanishes on all of \( C \)
A Concrete Example

\[ x_1x_2x_3 \in CF(J_C) \]

First Piece of Information:

- This polynomial vanishes on all of \( C \)
- So 111 is NOT in \( C \)
A Concrete Example

\[ x_1 x_2 x_3 \in CF(J_{\mathcal{C}}) \]

First Piece of Information:
- This polynomial vanishes on all of \( \mathcal{C} \)
- So 111 is NOT in \( \mathcal{C} \)
- \( So\ U_1 \cap U_2 \cap U_3 \) is empty!
A Concrete Example

\[ x_1 x_2 x_3 \in CF(J_C) \]

First Piece of Information:
- This polynomial vanishes on all of \( C \)
- So 111 is NOT in \( C \)
- So \( U_1 \cap U_2 \cap U_3 \) is empty!

Second Piece of Information:
- No divisor of \( x_1 x_2 x_3 \) is in \( J_C \).
A Concrete Example

\[ x_1x_2x_3 \in CF(J_C) \]

First Piece of Information:
- This polynomial vanishes on all of \( C \)
- So 111 is NOT in \( C \)
- So \( U_1 \cap U_2 \cap U_3 \) is empty!

Second Piece of Information:
- No divisor of \( x_1x_2x_3 \) is in \( J_C \).
- So \( x_1x_2 \) is NOT in \( J_C \).
A Concrete Example

\[ x_1 x_2 x_3 \in CF(J_C) \]

First Piece of Information:
- This polynomial vanishes on all of \( C \)
- So 111 is NOT in \( C \)
- So \( U_1 \cap U_2 \cap U_3 \) is empty!

Second Piece of Information:
- No divisor of \( x_1 x_2 x_3 \) is in \( J_C \).
- So \( x_1 x_2 \) is NOT in \( J_C \).
- Must have 110 or 111 in \( C \).
A Concrete Example

\[ x_1 x_2 x_3 \in CF(J_C) \]

First Piece of Information:
- This polynomial vanishes on all of \( C \)
- So 111 is NOT in \( C \)
- So \( U_1 \cap U_2 \cap U_3 \) is empty!

Second Piece of Information:
- No divisor of \( x_1 x_2 x_3 \) is in \( J_C \).
- So \( x_1 x_2 \) is NOT in \( J_C \).
- Must have 110 or 111 in \( C \).
A Concrete Example

\[ x_1x_2x_3 \in CF(J_C) \]

First Piece of Information:
- This polynomial vanishes on all of \( C \)
- So 111 is NOT in \( C \)
- So \( U_1 \cap U_2 \cap U_3 \) is empty!

Second Piece of Information:
- No divisor of \( x_1x_2x_3 \) is in \( J_C \).
- So \( x_1x_2 \) is NOT in \( J_C \).
- Must have 110 or 111 in \( C \).
- So \( U_1 \cap U_2 \) is nonempty! (Likewise for \( U_1 \cap U_3 \) and \( U_2 \cap U_3 \))
We associate codes to neural ideals, and use the canonical form to compactly present the neural ideal and encode information about the code and its realizations.
We associate codes to neural ideals, and use the canonical form to compactly present the neural ideal and encode information about the code and its realizations.

We hope to understand convex codes by examining neural ideals and their canonical forms.
Operations on Convex Codes and the Canonical Form

\[ CF(J_C) \xrightarrow{x_4 \mapsto 1} CF(J_{C'}) \]

Intersecting with \( U_4 \)
An Interesting Homomorphism

The map \( \phi : \mathbb{F}_2^n \to \mathbb{F}_2^{n-1} \) given by

\[
f(x_1, \ldots, x_{n-1}, x_n) \mapsto f(x_1, \ldots, x_{n-1}, 1)
\]

is a homomorphism. Furthermore, it maps neural ideals to neural ideals!

**Most Importantly:** The action of \( \phi \) is exactly that “restricting” to the set \( U_n \) in any realization of \( C \). We have described a geometric operation purely algebraically!

The map \( \phi \) also sends convex neural ideals to convex neural ideals!
Homomorphisms Respecting Neural Ideals

**Definition**

We say a homomorphism $\phi : \mathbb{F}_2[n] \to \mathbb{F}_2[m]$ respects neural ideals if for every $C \subseteq \{0,1\}^n$ there exists $D \subseteq \{0,1\}^n$ so that

$$\phi(J_C) = J_D.$$ 

That is, if $\phi$ maps neural ideals to neural ideals.

Can we classify all such homomorphisms? Do they have geometric meaning?
**Homomorphisms Respecting Neural Ideals**

**Restriction:** Mapping $x_i \mapsto 1$ or $x_i \mapsto 0$ for some $i$.

- $x_i \mapsto 1$ corresponds with replacing each $U_j$ by $U_j \cap U_i$.
- $x_i \mapsto 0$ corresponds with replacing each $U_j$ by $U_j \setminus U_i$. 
**Homomorphisms Respecting Neural Ideals**

**Restriction:** Mapping $x_i \mapsto 1$ or $x_i \mapsto 0$ for some $i$.
- $x_i \mapsto 1$ corresponds with replacing each $U_j$ by $U_j \cap U_i$.
- $x_i \mapsto 0$ corresponds with replacing each $U_j$ by $U_j \setminus U_i$.

**Bit Flipping:** Mapping $x_i \mapsto 1 - x_i$ for some $i$.
- Corresponds to taking the complement of $U_i$. 
Homomorphisms Respecting Neural Ideals

**Restriction:** Mapping $x_i \mapsto 1$ or $x_i \mapsto 0$ for some $i$.
- $x_i \mapsto 1$ corresponds with replacing each $U_j$ by $U_j \cap U_i$.
- $x_i \mapsto 0$ corresponds with replacing each $U_j$ by $U_j \setminus U_i$.

**Bit Flipping:** Mapping $x_i \mapsto 1 - x_i$ for some $i$.
- Corresponds to taking the complement of $U_i$.

**Permutation:** Permuting labels on the variables in $\mathbb{F}_2[n]$.
- Corresponds to permuting labels on the sets in a realization.
Theorem

Let $\phi : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^m$ be a homomorphism respecting neural ideals. Then $\phi$ is the composition of the three types of maps previously described:

- Permutation
- Restriction
- Bit flipping
Conclusion

In This Talk:

- We associated polynomial ideals to codes.
- We used these ideals to understand codes and their realizations.
- We described a class of homomorphisms which play nicely with these ideals. These homomorphisms can be used to understand convex codes, and also computationally.

What's Next?

- How do maps respecting neural ideals affect canonical forms?
- What other algebraic techniques can be leveraged?
- What can we do to understand convex codes without the algebraic approach?
Conclusion

In This Talk:

- We associated polynomial ideals to codes.
- We used these ideals to understand codes and their realizations.
- We described a class of homomorphisms which play nicely with these ideals. These homomorphisms can be used to understand convex codes, and also computationally.

What’s Next?

- How do maps respecting neural ideals affect canonical forms?
- What other algebraic techniques can be leveraged?
- What can we do to understand convex codes without the algebraic approach?