

Lecture 9: Error-Correcting Codes

Information Theory

Willard Þór Rafnsson

Reykjavík University, Iceland

24. February, 2009

Preliminaries
 Abstract Algebra
 Field \mathbb{F}
 Finite Field \mathbb{F}_p
 Polynomials
 Finite Field \mathbb{F}_p
 Linear Algebra
 Vector Space
 Terminology
 Linear Block Codes
 Definition
 Repetition Code
 Parity Check Code
 Hamming Code
 Gilbert-Vašanek
 Bound
 Minimum
 Distance
 Sphere-Packing
 Theorem
 Example

Recap

We have seen all the main results in this field.

Lecture 9:
E -C -ectig
C des

P e i i a i e s

Abst act A geb a

Fie d \mathbb{F}

Fi ite Fie d \mathbb{F}_p

P y i a s

Fi ite Fie d \mathbb{F}_p

Li ea A geb a

Vect S ace

Te i gy

Li ea B ck C des

Defi iti

Re etiti C de

Pa ity Check C de

Ha i g C de

Gi be t-Va sha v

B u d

Mi i u

Distace

S he e-Packi g

The e

E i gue

Recap

We have seen all the main results in this field.

Source Coding: We **can** get $L(\mathcal{C})$ for \mathcal{C} arbitrarily close to $H(\mathcal{S})$ by **applying source extension**.

Lecture 9:

Encoding Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gibbs-Vaishnav

Budd

Middle

Distance

Shannon-Packing

Theorem

Example

Recap

We have seen all the main results in this field.

Source Coding: We **can** get $L(\mathcal{C})$ for \mathcal{C} arbitrarily close to $H(\mathcal{S})$ by **applying source extension**.

Channel Coding: We **can** get R for \mathcal{C} to approach C with $\Pr_E \rightarrow 0$. There is no better capacity $C' > C$ that satisfies this.

Lecture 9:
Error-Correcting
Codes

Prerequisites
Abstract Algebra
Field \mathbb{F}
Finite Field \mathbb{F}_p
Polynomials
Finite Field \mathbb{F}_p
Linear Algebra
Vector Space
Terminology
Linear Block Codes
Definition
Redundancy Code
Parity Check Code
Hamming Code
Gilbert-Varslavsky
Bound
Minimum
Distance
Shannon-Packing
Theorem
Example

Recap

We have seen all the main results in this field.

Source Coding: We **can** get $L(\mathcal{C})$ for \mathcal{C} arbitrarily close to $H(\mathcal{S})$ by **applying source extension**.

Channel Coding: We **can** get R for \mathcal{C} to approach C with $\Pr_E \rightarrow 0$. There is no better capacity $C' > C$ that satisfies this.

Combined Source-Channel Coding: (theoretically) just as effective as separating them.

Lecture 9:
Error-Correcting
Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Codes

Hamming Codes

Cyclic Codes

Burrows

Minimum

Distance

Shannon's Packing

Theorem

Example

Recap

We have seen all the main results in this field.

Source Coding: We **can** get $L(\mathcal{C})$ for \mathcal{C} arbitrarily close to $H(\mathcal{S})$ by **applying source extension**.

Channel Coding: We **can** get R for \mathcal{C} to approach C with $\Pr_E \rightarrow 0$. There is no better capacity $C' > C$ that satisfies this.

Combined Source-Channel Coding: (theoretically) just as effective as separating them.

There is a **Hole** here:

***How** do we **construct** \mathcal{C} with (R, \Pr_E) arbitrarily close $(C, 0)$?*

This is what researchers in **Coding Theory** have been working on for the last 50 years.

Lecture 9:
Error-Correcting
Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gibbs-Vaishnav

Build

Minimum

Distance

Shannon-Packing

Theorem

Example

Recap

We have seen all the main results in this field.

Source Coding: We **can** get $L(\mathcal{C})$ for \mathcal{C} arbitrarily close to $H(\mathcal{S})$ by **applying source extension**.

Channel Coding: We **can** get R for \mathcal{C} to approach C with $\Pr_E \rightarrow 0$. There is no better capacity $C' > C$ that satisfies this.

Combined Source-Channel Coding: (theoretically) just as effective as separating them.

There is a **Hole** here:

***How** do we **construct** \mathcal{C} with (R, \Pr_E) arbitrarily close $(C, 0)$?*

This is what researchers in **Coding Theory** have been working on for the last 50 years. *The problem remains unsolved!* (area of active research).

Lecture 9:
Error-Correcting
Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gibbs-Van Schaik

Build

Minimum

Distance

Shannon-Packing

Theorem

Example

Recap

We have seen all the main results in this field.

Source Coding: We **can** get $L(\mathcal{C})$ for \mathcal{C} arbitrarily close to $H(S)$ by **applying source extension**.

Channel Coding: We **can** get R for \mathcal{C} to approach C with $\Pr_E \rightarrow 0$. There is no better capacity $C' > C$ that satisfies this.

Combined Source-Channel Coding: (theoretically) just as effective as separating them.

There is a **Hole** here:

How do we *construct* \mathcal{C} with (R, \Pr_E) arbitrarily close $(C, 0)$?

This is what researchers in **Coding Theory** have been working on for the last 50 years. *The problem remains unsolved!* (area of active research). But mathematicians have done quite well. We will, in the next few lectures, look into the underlying basics of the *algebraic aspects* of error-correcting codes, to get a good impression of how error-correction works.

Outline

1 Preliminaries

- Abstract Algebra
 - Field \mathbb{F}
 - Finite Field \mathbb{F}_p
 - Polynomials
 - Finite Field \mathbb{F}_{p^m}
- Linear Algebra
 - Vector Space
 - Terminology

2 Linear Block Codes

- Definition
- Repetition Code
- Parity Check Code
- Hamming Code

3 Gilbert-Varshamov Bound

- Minimum Distance
- Sphere-Packing
- Theorem

4 Epilogue

Lecture 9:
Error-Correcting
Codes

Preliminaries

Abstract Algebra
Field \mathbb{F}
Finite Field \mathbb{F}_p
Polynomials
Finite Field \mathbb{F}_{p^m}
Linear Algebra
Vector Space
Terminology

Linear Block Codes
Definition
Repetition Code
Parity Check Code
Hamming Code

Gilbert-Varshamov
Bound
Minimum
Distance
Sphere-Packing
Theorem

Epilogue

Reasoning About Γ

Our communicating systems are well-defined.

Definition (Discrete Memoryless Channel)

Lecture 9:
E -C ecti g
C des

P e i i a i e s

Abst act A geb a

Fie d \mathbb{F}

Fi ite Fie d \mathbb{F}_p

P y i a s

Fi ite Fie d \mathbb{F}_p

Li ea A geb a

Vect S ace

Te i gy

Li ea B ck C des

Defi iti

Re etiti C de

Pa ity Check C de

Ha i g C de

Gi be t-Va sha v

B u d

Mi i u

Distace

S he e-Packi g

The e

E i gue

Reasoning About Γ

Our communicating systems are well-defined.

Definition (Discrete Memoryless Channel)

A discrete memoryless channel Γ is a triple

$$\Gamma = (\mathcal{X}, p(b | a), \mathcal{Y}),$$

where X is the input variable, Y the output variable, and where Y depends only on X with the conditional probability $p(b | a) = \Pr(Y = b | X = a)$.

Assumption: $\mathcal{X} = \mathcal{Y}$. This is not a problem; we could extend \mathcal{X} to $\mathcal{X} \cup \mathcal{Y}$ with some symbols never sent, for instance.

There has been a *long tradition* in mathematics to study abstract, simple algebraic structures. We now show how our framework fits into those. This lets us *reason about our framework, in a well-understood framework*.

Lecture 9:
E -C ection
C des

P e i i a i e s

Abstr act A geb a

Fie d \mathbb{F}

Fi ite Fie d \mathbb{F}_p

P y i a s

Fi ite Fie d \mathbb{F}_p

Li ea A geb a

Vect S ace

Te i gy

Li ea B ck C des

Defi iti

Re etiti C de

Pa ity Check C de

Ha i g C de

Gi be t-Va sha v

B u d

Mi i u

Distace

S he e-Packi g

The e

E i gue

And now...

It's

Definition Time

Lecture 9:

E -C ecti g
C des

P e i i a i e s

Abst act A geb a

Fie d \mathbb{F}

Fi ite Fie d \mathbb{F}_p

P y i a s

Fi ite Fie d \mathbb{F}_p

Li ea A geb a

Vect S ace

Te i gy

Li ea B ck C des

Defi iti

Re etiti C de

Pa ity Check C de

Ha i g C de

Gi be t-Va sha v

B u d

Mi i u

Distace

S he e-Packi g

The e

E i gue

Outline

1 Preliminaries

- Abstract Algebra
 - Field \mathbb{F}
 - Finite Field \mathbb{F}_p
 - Polynomials
 - Finite Field \mathbb{F}_{p^m}
- Linear Algebra
 - Vector Space
 - Terminology

2 Linear Block Codes

- Definition
- Repetition Code
- Parity Check Code
- Hamming Code

3 Gilbert-Varshamov Bound

- Minimum Distance
- Sphere-Packing
- Theorem

4 Epilogue

Lecture 9: Error-Correcting Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}
 Finite Field \mathbb{F}_p
 Polynomials
 Finite Field \mathbb{F}_{p^m}
 Linear Algebra
 Vector Space
 Terminology

Linear Block Codes

Definition
 Repetition Code
 Parity Check Code
 Hamming Code

Gilbert-Varshamov Bound

Minimum
 Distance
 Sphere-Packing
 Theorem

Epilogue

Field, Definition

A Field is a set \mathbb{F} closed under $+$, $-$, \cdot , \div , and satisfying typical algebraic properties wrt. these operators. Example: \mathbb{R} , \mathbb{Q} , \mathbb{Z} , even \mathbb{C} . Formally:

Lecture 9:
E -C e c t i g
C d e s

P e i i a i e s
A b s t a c t A g e b a
F i e d \mathbb{F}
F i i t e F i e d \mathbb{F}_p
P y i a s
F i i t e F i e d \mathbb{F}_p
L i e a A g e b a
V e c t S a c e
T e i g y
L i e a B c k C d e s
D e f i t i
R e e t i t i C d e
P a i t y C h e c k C d e
H a i g C d e
G i b e t - V a s h a v
B u d
M i i u
D i s t a c e
S h e e - P a c k i g
T h e e
E i g u e

Field, Definition

A Field is a set \mathbb{F} closed under $+$, $-$, \cdot , \div , and satisfying typical algebraic properties wrt. these operators. Example: \mathbb{R} , \mathbb{Q} , \mathbb{Z} , even \mathbb{C} . Formally:

Definition (Field)

A field is a triple $(\mathbb{F}, +, \cdot)$, where \mathbb{F} is a nonempty set containing 0 and 1, and $+$, \cdot are binary operators defined on \mathbb{F} , satisfying these axioms^a:

$\forall x \in \mathbb{F}. x + 0 = x$	(add-neutral)
$\forall x, y \in \mathbb{F}. x + y = y + x$	(add-comm)
$\forall x, y, z \in \mathbb{F}. (x + y) + z = x + (y + z)$	(add-assoc)
$\forall x \in \mathbb{F} \exists (-x) \in \mathbb{F}. x + (-x) = 0$	(add-inv)
$\forall x \in \mathbb{F}. x \cdot 1 = x$	(mult-neutral)
$\forall x, y \in \mathbb{F}. x \cdot y = y \cdot x$	(mult-comm)
$\forall x, y, z \in \mathbb{F}. x \cdot (y \cdot z) = (x \cdot y) \cdot z$	(mult-assoc)
$\forall x, y, z \in \mathbb{F}. x \cdot (y + z) = x \cdot y + x \cdot z$	(distr)
$\forall x \in \mathbb{F}; x \neq 0 \exists x^{-1} \in \mathbb{F}. x \cdot x^{-1} = 1.$	(mult-inv)

^aNote that \cdot has precedence over $+$.

Finite Field

Usually, $+$ and \cdot are obvious, so we omit them, and refer to $(\mathbb{F}, +, \cdot)$ as \mathbb{F} .

Definition (Finite Field)

A finite field \mathbb{F} is a field with finitely many elements.

Finite Field

Usually, $+$ and \cdot are obvious, so we omit them, and refer to $(\mathbb{F}, +, \cdot)$ as \mathbb{F} .

Definition (Finite Field)

A finite field \mathbb{F} is a field with finitely many elements.

Q: Beautiful idea! But, do these things even exist? For fields using $+$ and \cdot , like \mathbb{Z} , we have that for any $x \in \mathbb{Z}$, for instance, 7, you can always get “the next number”, $7 + 1 \in \mathbb{Z} \dots$

Finite Field

Usually, $+$ and \cdot are obvious, so we omit them, and refer to $(\mathbb{F}, +, \cdot)$ as \mathbb{F} .

Definition (Finite Field)

A finite field \mathbb{F} is a field with finitely many elements.

Q: Beautiful idea! But, do these things even exist? For fields using $+$ and \cdot , like \mathbb{Z} , we have that for any $x \in \mathbb{Z}$, for instance, 7, you can always get “the next number”, $7 + 1 \in \mathbb{Z} \dots$

A: Who said $+$ and \cdot needs to be $+$ and \cdot ?

Example (Finite Field $\mathbb{F} = \mathbb{F}_2 = \mathbb{Z}_2 = \{0, 1\}$)

Let $\mathbb{F} = \{0, 1\}$, let $x \hat{+} y = x + y \bmod 2$, and $x \hat{\cdot} y = x \cdot y \bmod 2$. Then $(\mathbb{F}, \hat{+}, \hat{\cdot})$ is a field!

$\hat{+}$	0	1
0	0	1
1	1	0

$\hat{\cdot}$	0	1
0	0	0
1	0	1

$1 \hat{+} 0 \hat{+} 0 \hat{+} 1 \hat{+} 1 \hat{+} 0 \hat{+} 1 \hat{+} 0 \hat{+} 1 \hat{+} 0 \hat{+} 0 = 1$, for instance.

More Finite Fields?

Q: Awesome! So, *when* $A = \{0, 1\}$, then A is a finite field! But, what if $A = \{0, 1, 2\}$? Are there other finite fields?

Lecture 9:

E - C - C C - C - C C - C - C

Peirce's

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomial

Finite Field \mathbb{F}_p

Linear Algebra

Vector Space

Topology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gibbs-Vassiliou

Bund

Distance

Distance

Shannon's

Theorem

Example

More Finite Fields?

Q: Awesome! So, *when* $A = \{0, 1\}$, then A is a finite field! But, what if $A = \{0, 1, 2\}$? Are there other finite fields?

A: Yes! There are *many*!

Theorem (All Finite Fields are Known (proof omitted))

- i) *There is a finite field of order q iff $q = p^m$ for some prime p and integer $m \geq 1$.*
- ii) *Any two finite fields of the same order are isomorphic.*

More Finite Fields?

Q: Awesome! So, when $A = \{0, 1\}$, then A is a finite field! But, what if $A = \{0, 1, 2\}$? Are there other finite fields?

A: Yes! There are *many*!

Theorem (All Finite Fields are Known (proof omitted))

- i) *There is a finite field of order q iff $q = p^m$ for some prime p and integer $m \geq 1$.*
- ii) *Any two finite fields of the same order are isomorphic.*

We denote a “finite field of order q ” by \mathbb{F}_q . Since for a given q , all \mathbb{F}_q are isomorphic, i.e., “the same”, we speak of “the” finite field \mathbb{F}_q . \mathbb{F}_q is also called the *Galois Field* of order q ; $\mathbb{F}_q = \text{GF}(q)$.

More Finite Fields?

Q: Awesome! So, *when* $A = \{0, 1\}$, then A is a finite field! But, what if $A = \{0, 1, 2\}$? Are there other finite fields?

A: Yes! There are *many*!

Theorem (All Finite Fields are Known (proof omitted))

- i) *There is a finite field of order q iff $q = p^m$ for some prime p and integer $m \geq 1$.*
- ii) *Any two finite fields of the same order are isomorphic.*

We denote a “finite field of order q ” by \mathbb{F}_q . Since for a given q , all \mathbb{F}_q are isomorphic, i.e., “the same”, we speak of “the” finite field \mathbb{F}_q . \mathbb{F}_q is also called the *Galois Field* of order q ; $\mathbb{F}_q = \text{GF}(q)$.

Definition (Prime Element)

Let \mathbb{F}_q be a finite field consisting of q elements. $\alpha \in \mathbb{F}_q$ is a *prime element* of \mathbb{F}_q if $\mathbb{F}_q \setminus \{0\} = \{\alpha^i : i = 0, \dots, q-2\}$.

So, α “generates” \mathbb{F}_q . In \mathbb{Z}_7 , $\alpha = 5$.

A Problem

If $m = 1$, that is, $\mathbb{F}_q = \mathbb{F}_p$, we usually take $\mathbb{F}_p = \mathbb{Z}_p = \{0, 1, \dots, p-1\}$ ¹.

Q: Ha! Now I am going to ruin your day. You cannot do this when $m = 2$.

Lecture 9: Error-Correcting Codes

Peirce's
Abstract Algebra
Field \mathbb{F}
Finite Field \mathbb{F}_p
Polynomials
Finite Field \mathbb{F}_p
Linear Algebra
Vectorspace
Terminology
Linear Block Codes
Definition
Redundancy Code
Parity Check Code
Hamming Code
Gibbert-Vanderbilt
Burrows
Minimum
Distance
Shannon-Packing
Theorem
Exercise

¹From isomorphism, for **any** $F \neq \mathbb{F}_p$ with $|F| = |\mathbb{F}_p|$, we can construct a $+$ and \cdot that makes $(F, +, \cdot)$ a finite field (bijective mapping).

²Computation modulus q

³Because fields are “domains”.

A Problem

If $m = 1$, that is, $\mathbb{F}_q = \mathbb{F}_p$, we usually take $\mathbb{F}_p = \mathbb{Z}_p = \{0, 1, \dots, p-1\}$ ¹.

Q: Ha! Now I am going to ruin your day. You cannot do this when $m = 2$. For instance, for $p = 2$ and $m = 2$, we get $\mathbb{F}_4 = \mathbb{F}_{2^2} = \{0, 1, 2, 3\}$, and² $p^m = pp = 0$ in this field.

¹From isomorphism, for **any** $F \neq \mathbb{F}_p$ with $|F| = |\mathbb{F}_p|$, we can construct a $+$ and \cdot that makes $(F, +, \cdot)$ a finite field (bijective mapping).

²Computation modulus q

³Because fields are “domains”.

A Problem

If $m = 1$, that is, $\mathbb{F}_q = \mathbb{F}_p$, we usually take $\mathbb{F}_p = \mathbb{Z}_p = \{0, 1, \dots, p-1\}$ ¹.

Q: Ha! Now I am going to ruin your day. You cannot do this when $m = 2$. For instance, for $p = 2$ and $m = 2$, we get $\mathbb{F}_4 = \mathbb{F}_{2^2} = \{0, 1, 2, 3\}$, and² $p^m = pp = 0$ in this field. This makes p a “zero divisor”. And $p \neq 0$.

¹From isomorphism, for **any** $F \neq \mathbb{F}_p$ with $|F| = |\mathbb{F}_p|$, we can construct a $+$ and \cdot that makes $(F, +, \cdot)$ a finite field (bijective mapping).

²Computation modulus q

³Because fields are “domains”.

A Problem

If $m = 1$, that is, $\mathbb{F}_q = \mathbb{F}_p$, we usually take $\mathbb{F}_p = \mathbb{Z}_p = \{0, 1, \dots, p-1\}$ ¹.

Q: Ha! Now I am going to ruin your day. You cannot do this when $m = 2$. For instance, for $p = 2$ and $m = 2$, we get $\mathbb{F}_4 = \mathbb{F}_{2^2} = \{0, 1, 2, 3\}$, and² $p^m = pp = 0$ in this field. This makes p a “zero divisor”. And $p \neq 0$. In a field, only 0 may be a zero divisor³. So \mathbb{F}_{2^2} *cannot be a field*! Even when your theorem said it should be! Haha!

¹From isomorphism, for **any** $F \neq \mathbb{F}_p$ with $|F| = |\mathbb{F}_p|$, we can construct a $+$ and \cdot that makes $(F, +, \cdot)$ a finite field (bijective mapping).

²Computation modulus q

³Because fields are “domains”.

A Problem

If $m = 1$, that is, $\mathbb{F}_q = \mathbb{F}_p$, we usually take $\mathbb{F}_p = \mathbb{Z}_p = \{0, 1, \dots, p-1\}$ ¹.

Q: Ha! Now I am going to ruin your day. You cannot do this when $m = 2$. For instance, for $p = 2$ and $m = 2$, we get $\mathbb{F}_4 = \mathbb{F}_{2^2} = \{0, 1, 2, 3\}$, and² $p^m = pp = 0$ in this field. This makes p a “zero divisor”. And $p \neq 0$. In a field, only 0 may be a zero divisor³. So \mathbb{F}_{2^2} *cannot be a field*! Even when your theorem said it should be! Haha!

A: Partially correct; \mathbb{Z}_{p^m} for $m > 1$, as you exemplified with \mathbb{Z}_4 , is *not a field*. **But** that does **not** mean \mathbb{F}_{p^m} for $m > 1$ does not exist.

¹From isomorphism, for **any** $F \neq \mathbb{F}_p$ with $|F| = |\mathbb{F}_p|$, we can construct a $+$ and \cdot that makes $(F, +, \cdot)$ a finite field (bijective mapping).

²Computation modulus q

³Because fields are “domains”.

A Problem

If $m = 1$, that is, $\mathbb{F}_q = \mathbb{F}_p$, we usually take $\mathbb{F}_p = \mathbb{Z}_p = \{0, 1, \dots, p-1\}$ ¹.

Q: Ha! Now I am going to ruin your day. You cannot do this when $m = 2$. For instance, for $p = 2$ and $m = 2$, we get $\mathbb{F}_4 = \mathbb{F}_{2^2} = \{0, 1, 2, 3\}$, and² $p^m = pp = 0$ in this field. This makes p a “zero divisor”. And $p \neq 0$. In a field, only 0 may be a zero divisor³. So \mathbb{F}_{2^2} *cannot be a field*! Even when your theorem said it should be! Haha!

A: Partially correct; \mathbb{Z}_{p^m} for $m > 1$, as you exemplified with \mathbb{Z}_4 , is *not a field*. **But** that does **not** mean \mathbb{F}_{p^m} for $m > 1$ does not exist.

What do we do? How do we find \mathbb{F}_{p^m} ? By using *polynomials*.

¹From isomorphism, for **any** $F \neq \mathbb{F}_p$ with $|F| = |\mathbb{F}_p|$, we can construct a $+$ and \cdot that makes $(F, +, \cdot)$ a finite field (bijective mapping).

²Computation modulus q

³Because fields are “domains”.

Polynomials 101

There is nothing scary about polynomials f over \mathbb{F} .

Lecture 9:

E -C -ectig
 C des

Peirce's

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gibert-Vashkov

Bund

Minimum

Distance

Shannon-Packing

Theorem

Erasure

Polynomials 101

There is nothing scary about polynomials f over \mathbb{F} . They are just:

$$f : \mathbb{F} \rightarrow \mathbb{F}$$

where we only allow positive powers.

Lecture 9:

E -C e c t i g
 C d e s

P e i i a i e s

A b s t a c t A g e b a

F i e d \mathbb{F}

F i i t e F i e d \mathbb{F}_p

P y i a s

F i i t e F i e d \mathbb{F}_p

L i e a A g e b a

V e c t S a c e

T e i g y

L i e a B c k C d e s

D e f i t i

R e e t i t i C d e

P a i t y C h e c k C d e

H a i g C d e

G i b e t - V a s h a v

B u d

M i i u

D i s t a c e

S h e e - P a c k i g

T h e e

E i g u e

Polynomials 101

There is nothing scary about polynomials f over \mathbb{F} . They are just:

$$f : \mathbb{F} \rightarrow \mathbb{F}$$

where we only allow positive powers. So

$$f = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$$

or, if you feel like specifying the parameter to f explicitly (I won't),

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0.$$

Lecture 9:

E - C e c t i g
C d e s

P e i i a i e s

A b s t a c t A l g e b r a

F i e d F

F i n i t e F i e d F_p

P o l y n o m i a l s

F i n i t e F i e d F_p

L i n e a r A l g e b r a

V e c t o r S p a c e

T e r m i n o l o g y

L i n e a r B a c k C o d e s

D e f i n i t i o n

R e c o m m e n d e d

P a r i t y C h e c k C o d e

H a m m i n g C o d e

G i b e t - V a s h a v

B u d

M i n i m u m

D i s t a n c e

S h e e - P a c k i n g

T h e e

E i g u e

Polynomials 101

There is nothing scary about polynomials f over \mathbb{F} . They are just:

$$f : \mathbb{F} \rightarrow \mathbb{F}$$

where we only allow positive powers. So

$$f = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$$

or, if you feel like specifying the parameter to f explicitly (I won't),

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0.$$

Here, n is the **degree** of f , denoted $\deg(f)$.

Polynomials 101

There is nothing scary about polynomials f over \mathbb{F} . They are just:

$$f : \mathbb{F} \rightarrow \mathbb{F}$$

where we only allow positive powers. So

$$f = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$$

or, if you feel like specifying the parameter to f explicitly (I won't),

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0.$$

Here, n is the **degree** of f , denoted $\deg(f)$.

Example

For instance, a polynomial f over \mathbb{Z} ,

$$f : \mathbb{Z} \rightarrow \mathbb{Z}$$

could be $f(x) = 4x^2 + 3x - 27$. So, for instance, $f(2) = 49 \in \mathbb{Z}$. Here, $\deg(f) = 2$.

Polynomials, Basic Results

The set of all polynomials over \mathbb{F} is defined by

$$\mathbb{F}[x] = \{a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0 \mid n \in \mathbb{N}, a_i \in \mathbb{F}\}.$$

The set of polynomials up to (not including) degree m :

$$\mathbb{P}_q(m) = \{a_{m-1} x^{m-1} + \cdots + a_1 x + a_0 \mid a_i \in \mathbb{F}_q\} \subseteq \mathbb{F}_q[x]$$

Lecture 9:

E -C e c t i g
C d e s

P e i i a i e s

A b s t a c t A g e b a

F i e d \mathbb{F}

F i n i t e F i e d \mathbb{F}_p

P y i a s

F i n i t e F i e d \mathbb{F}_p

L i n e a r A g e b a

V e c t o r S p a c e

T e r m i n o l o g y

L i n e a r B a c k C o d e s

D e f i n i t i o n s

R e c o n s t r u c t i o n C o d e

P a r i t y C h e c k C o d e

H a m m i n g C o d e

G i b e r t - V a n d e r V e e d

B u d g e t

M i n i m u m

D i s t a n c e

S h e e r - P a c k i n g

T h e e

E i g e n

Polynomials, Basic Results

The set of all polynomials over \mathbb{F} is defined by

$$\mathbb{F}[x] = \{a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0 \mid n \in \mathbb{N}, a_i \in \mathbb{F}\}.$$

The set of polynomials up to (not including) degree m :

$$\mathbb{P}_q(m) = \{a_{m-1} x^{m-1} + \cdots + a_1 x + a_0 \mid a_i \in \mathbb{F}_q\} \subseteq \mathbb{F}_q[x]$$

Polynomials behave like numbers. There are even “prime polynomials”.

Lecture 9:

E - C o m p l e t i n g
C o d e s

P e i i a i e s

A b s t a c t A l g e b r a

F i e d \mathbb{F}

F i n i t e F i e d \mathbb{F}_p

P o l y n o m i a l s

F i n i t e F i e d \mathbb{F}_p

L i n e a r A l g e b r a

V e c t o r S p a c e

T e r m i n o l o g y

L i n e a r B a s i s C o d e s

D e f i n i t i o n

R e c o n s t r u c t i o n C o d e

P a r i t y C h e c k C o d e

H a m m i n g C o d e

G i b e t - V a s h a n o v

B u d

M i n i m u m

D i s t a n c e

S h e e - P a c k i n g

T h e e

E i g e n

Polynomials, Basic Results

The set of all polynomials over \mathbb{F} is defined by

$$\mathbb{F}[x] = \{a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0 \mid n \in \mathbb{N}, a_i \in \mathbb{F}\}.$$

The set of polynomials up to (not including) degree m :

$$\mathbb{P}_q(m) = \{a_{m-1} x^{m-1} + \cdots + a_1 x + a_0 \mid a_i \in \mathbb{F}_q\} \subseteq \mathbb{F}_q[x]$$

Polynomials behave like numbers. There are even “prime polynomials”.

Division w/remainder: For $d \in \mathbb{F}[x]$ nonzero with an invertible leading coefficient, then

$$\forall f \in \mathbb{F}[x]. \exists q, r \in \mathbb{F}[x]. f = qd + r.$$

Lecture 9:

E - C e c t i g
C d e s

P e i i a i e s

A b s t a c t A l g e b r a

F i e d F

F i t e F i e d F_p

P o l y n o m i a l s

F i t e F i e d F_p

L i n e a r A l g e b r a

V e c t o r S p a c e

T e n o r

L i n e a r B a c k C o d e s

D e f i n i t i o n

R e c o n s t r u c t i o n C o d e

P a r i t y C h e c k C o d e

H a m m i n g C o d e

G i b e r t - V a n d e r V e e r

B u d g e t

M i n i m u m

D i s t a n c e

S h e e r - P a c k i n g

T h e o r e m

E i g e n

Polynomials, Basic Results

The set of all polynomials over \mathbb{F} is defined by

$$\mathbb{F}[x] = \{a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0 \mid n \in \mathbb{N}, a_i \in \mathbb{F}\}.$$

The set of polynomials up to (not including) degree m :

$$\mathbb{P}_q(m) = \{a_{m-1} x^{m-1} + \cdots + a_1 x + a_0 \mid a_i \in \mathbb{F}_q\} \subseteq \mathbb{F}_q[x]$$

Polynomials behave like numbers. There are even “prime polynomials”.

Division w/remainder: For $d \in \mathbb{F}[x]$ nonzero with an invertible leading coefficient, then

$$\forall f \in \mathbb{F}[x]. \exists q, r \in \mathbb{F}[x]. f = qd + r.$$

Irreducible Polynomial: The polynomial $f \in \mathbb{F}[x]$ is *irreducible*, if

$$\forall a, b \in \mathbb{F}[x]. f = ab \implies \deg(a) = 0 \vee \deg(b) = 0.$$

Lecture 9:

E - C e c t i g
C d e s

P e i i a i e s

A b s t a c t A l g e b r a

F i e d \mathbb{F}

F i t e F i e d \mathbb{F}_p

P o l y n o m i a l s

F i t e F i e d \mathbb{F}_p

L i n e a r A l g e b r a

V e c t o r S p a c e

T e n o r

L i n e a r B a c k C o d e s

D e f i n i t i o n

R e c o n s t r u c t i o n C o d e

P a r i t y C h e c k C o d e

H a m m i n g C o d e

G i b b e t - V a n d e r b i l t

B u d g e t

M i n i m u m

D i s t a n c e

S h e e r - P a c k i n g

T h e e

E i g u e

Polynomials, Basic Results

The set of all polynomials over \mathbb{F} is defined by

$$\mathbb{F}[x] = \{a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0 \mid n \in \mathbb{N}, a_i \in \mathbb{F}\}.$$

The set of polynomials up to (not including) degree m :

$$\mathbb{P}_q(m) = \{a_{m-1} x^{m-1} + \cdots + a_1 x + a_0 \mid a_i \in \mathbb{F}_q\} \subseteq \mathbb{F}_q[x]$$

Polynomials behave like numbers. There are even “prime polynomials”.

Division w/remainder: For $d \in \mathbb{F}[x]$ nonzero with an invertible leading coefficient, then

$$\forall f \in \mathbb{F}[x]. \exists q, r \in \mathbb{F}[x]. f = qd + r.$$

Irreducible Polynomial: The polynomial $f \in \mathbb{F}[x]$ is *irreducible*, if

$$\forall a, b \in \mathbb{F}[x]. f = ab \implies \deg(a) = 0 \vee \deg(b) = 0.$$

Irreducible Polynomials \equiv Primes: If $f \in \mathbb{F}[x]$ is irreducible, then

$$\forall a, b \in \mathbb{F}[x]. f|ab \implies f|a \vee f|b.$$

Lecture 9:
E - C e c t i g
C d e s

P e i i a i e s

A b s t a c t A l g e b r a

F i e d \mathbb{F}

F i n i t e F i e d \mathbb{F}_p

P o l y n o m i a l s

F i n i t e F i e d \mathbb{F}_p

L i n e a r A l g e b r a

V e c t o r S p a c e

T e r m i n o l o g y

L i n e a r B a c k C o d e s

D e f i n i t i o n

R e c o n s t r u c t i o n C o d e

P a r i t y C h e c k C o d e

H a m m i n g C o d e

G i b b e t - V a n d e r b i l t

B u d

M i n i m u m

D i s t a n c e

S h e e r - P a c k i n g

T h e e

E i g u e

Polynomials, Basic Results

The set of all polynomials over \mathbb{F} is defined by

$$\mathbb{F}[x] = \{a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0 \mid n \in \mathbb{N}, a_i \in \mathbb{F}\}.$$

The set of polynomials up to (not including) degree m :

$$\mathbb{P}_q(m) = \{a_{m-1} x^{m-1} + \cdots + a_1 x + a_0 \mid a_i \in \mathbb{F}_q\} \subseteq \mathbb{F}_q[x]$$

Polynomials behave like numbers. There are even “prime polynomials”.

Division w/remainder: For $d \in \mathbb{F}[x]$ nonzero with an invertible leading coefficient, then

$$\forall f \in \mathbb{F}[x]. \exists q, r \in \mathbb{F}[x]. f = qd + r.$$

Irreducible Polynomial: The polynomial $f \in \mathbb{F}[x]$ is *irreducible*, if

$$\forall a, b \in \mathbb{F}[x]. f = ab \implies \deg(a) = 0 \vee \deg(b) = 0.$$

Irreducible Polynomials \equiv Primes: If $f \in \mathbb{F}[x]$ is irreducible, then

$$\forall a, b \in \mathbb{F}[x]. f|ab \implies f|a \vee f|b.$$

Root: $r \in \mathbb{F}$ is the root of $f \in \mathbb{F}[x]$ if $f(r) = 0$.

Examples

Example

Let $\mathbb{F} = \mathbb{Z}$, and let $f \in \mathbb{F}[x]$; $f = 4x^2 + 2x + 1$.

Lecture 9:

Euclidean Rings

Polynomials

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vector Space

Topology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gibbs-Vaughan

Bund

Minimum

Distance

Shannon-Packing

Theorem

Exercise

Examples

Example

Let $\mathbb{F} = \mathbb{Z}$, and let $f \in \mathbb{F}[x]$; $f = 4x^2 + 2x + 1$. Then, if $d = 4x + 2$, $q = x$, and $r = 1$, then

$$f = qd + r = (x)(4x + 2) + (1) = 4x^2 + 2x + 1.$$

Lecture 9:

E -C e c t i g
C d e s

P e i i a i e s

A b s t a c t A g e b a

F i e d \mathbb{F}

F i t e F i e d \mathbb{F}_p

P y i a s

F i t e F i e d \mathbb{F}_p

L i e a A g e b a

V e c t S a c e

T e i g y

L i e a B c k C d e s

D e f i t i

R e e t i t i C d e

P a i t y C h e c k C d e

H a i g C d e

G i b e t - V a s h a v

B u d

M i i u

D i s t a c e

S h e e - P a c k i g

T h e e

E i g u e

Examples

Example

Let $\mathbb{F} = \mathbb{Z}$, and let $f \in \mathbb{F}[x]$; $f = 4x^2 + 2x + 1$. Then, if $d = 4x + 2$, $q = x$, and $r = 1$, then

$$f = qd + r = (x)(4x + 2) + (1) = 4x^2 + 2x + 1.$$

In $\mathbb{Z}_2 = \{0, 1\}$, $f = x^2 + 1$ is *irreducible*. No matter what you try, no $x \in \mathbb{Z}_2$ will make $f(x) = 0$.

Lecture 9:

E - C e c t i g
C d e s

P e i i a i e s

A b s t a c t A l g e b r a

F i e d \mathbb{F}

F i t e F i e d \mathbb{F}_p

P o l y n o m i a s

F i t e F i e d \mathbb{F}_p

L i n e a r A l g e b r a

V e c t o r S p a c e

T e r m i n o l o g y

L i n e a r B a s i s C o d e s

D e f i n i t i o n s

R e c o n s t r u c t i o n C o d e

P a r i t y C h e c k C o d e

H a m m i n g C o d e

G i b e r t - V a n d e r V e e r

B u d g e t

M i n i m u m

D i s t a n c e

S h e e - P a c k i n g

T h e e

E i g e n

Examples

Example

Let $\mathbb{F} = \mathbb{Z}$, and let $f \in \mathbb{F}[x]$; $f = 4x^2 + 2x + 1$. Then, if $d = 4x + 2$, $q = x$, and $r = 1$, then

$$f = qd + r = (x)(4x + 2) + (1) = 4x^2 + 2x + 1.$$

In $\mathbb{Z}_2 = \{0, 1\}$, $f = x^2 + 1$ is *irreducible*. No matter what you try, no $x \in \mathbb{Z}_2$ will make $f(x) = 0$.

Because f is irreducible, f cannot be written as a *product*. Which makes f a *prime element* in $\mathbb{F}_2[x]$.

Lecture 9:

E -C e c t i g
C d e s

P e i i a i e s

A b s t a c t A l g e b r a

F i e d F

F i t e F i e d \mathbb{Z}_p

P r y i a s

F i t e F i e d \mathbb{Z}_p

L i e a A l g e b r a

V e c t o r S p a c e

T e i g y

L i e a B a c k C d e s

D e f i n i t i o n

R e c o n s t r u c t i o n C d e

P a r i t y C h e c k C d e

H a r d i g C d e

G i b e t - V a s h a v

B u d

M i i u

D i s t a n c e

S h e e - P a c k i g

T h e e

E i g u e

Finding \mathbb{F}_p

So, how do we use this to find \mathbb{F}_p ?

Lecture 9:

E - Constructing
 Codes

Peirce's

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gibert-Vashkov

Bud

Minimum

Distance

Shannon-Packing

Theorem

Exercises

Finding \mathbb{F}_{p^m}

So, how do we use this to find \mathbb{F}_{p^m} ?

Hack: Notice that $f = x^2 + 1$ is also irreducible in \mathbb{R} .

Lecture 9:

Euclidean Cosets

Peirce's

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_{p^m}

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gibbert-Vander

Burrows

Minimum

Distance

Shannon-Packet

Theorem

Exercise

Finding \mathbb{F}_{p^m}

So, how do we use this to find \mathbb{F}_{p^m} ?

Hack: Notice that $f = x^2 + 1$ is also irreducible in \mathbb{R} . f has a root, $i = \sqrt{-1}$. But $i \notin \mathbb{R}$.

Lecture 9:

Euclidean Cosets

Peirce's

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_{p^m}

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gibbert-Vander

Burrows

Minimum

Distance

Shannon-Packet

Theorem

Example

Finding \mathbb{F}_{p^m}

So, how do we use this to find \mathbb{F}_{p^m} ?

Hack: Notice that $f = x^2 + 1$ is also irreducible in \mathbb{R} . f has a root, $i = \sqrt{-1}$. But $i \notin \mathbb{R}$. If we “insert” i into \mathbb{R} , then f is *not* irreducible in the resulting set, \mathbb{C} .

Lecture 9:

E -C -ectig
 C des

Peirce's

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_{p^m}

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gibert-Vaughan

Bund

Minim

Distance

Shannon-Packing

Theorem

Exercise

Finding \mathbb{F}_{p^m}

So, how do we use this to find \mathbb{F}_{p^m} ?

Hack: Notice that $f = x^2 + 1$ is also irreducible in \mathbb{R} . f has a root, $i = \sqrt{-1}$. But $i \notin \mathbb{R}$. If we “insert” i into \mathbb{R} , then f is *not* irreducible in the resulting set, \mathbb{C} . So, from an irreducible polynomial for $\mathbb{R}[x]$, we obtain a $\alpha = i \in \mathbb{C}$.

Lecture 9:

Extending Fields

Peirce's

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_{p^m}

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Redundancy Codes

Parity Check Codes

Hamming Codes

Gibert-Vaughan

Bund

Minimality

Distance

Shannon-Packing

Theorem

Example

Finding \mathbb{F}_{p^m}

So, how do we use this to find \mathbb{F}_{p^m} ?

Hack: Notice that $f = x^2 + 1$ is also irreducible in \mathbb{R} . f has a root, $i = \sqrt{-1}$. But $i \notin \mathbb{R}$. If we “insert” i into \mathbb{R} , then f is *not* irreducible in the resulting set, \mathbb{C} . So, from an irreducible polynomial for $\mathbb{R}[x]$, we obtain a $\alpha = i \in \mathbb{C}$.

We do the same thing in \mathbb{F}_q ; Let $m > 1$. Recall then that $\mathbb{F}_q \neq \mathbb{Z}_q$. How to find \mathbb{F}_q ?

Lecture 9:
 Extending
 Codes

Peirce's
 Abstract Algebra
 Finite Field \mathbb{F}_p
 Polynomials
 Finite Field \mathbb{F}_{p^m}
 Linear Algebra
 Vector Space
 Terminology
 Linear Block Codes
 Definition
 Redundancy Codes
 Parity Check Codes
 Hamming Codes
 Gilbert-Varslavsky
 Bound
 Minimum
 Distance
 Single-Packet
 Theorem
 Example

Finding \mathbb{F}_{p^m}

So, how do we use this to find \mathbb{F}_{p^m} ?

Hack: Notice that $f = x^2 + 1$ is also irreducible in \mathbb{R} . f has a root, $i = \sqrt{-1}$. But $i \notin \mathbb{R}$. If we “insert” i into \mathbb{R} , then f is *not* irreducible in the resulting set, \mathbb{C} . So, from an irreducible polynomial for $\mathbb{R}[x]$, we obtain a $\alpha = i \in \mathbb{C}$.

We do the same thing in \mathbb{F}_q ; Let $m > 1$. Recall then that $\mathbb{F}_q \neq \mathbb{Z}_q$. How to find \mathbb{F}_q ? If f is irreducible in $\mathbb{F}_q[x]$, then we let α be the “that which is the root of f ” (we may not know what it is).

Finding \mathbb{F}_{p^m}

So, how do we use this to find \mathbb{F}_{p^m} ?

Hack: Notice that $f = x^2 + 1$ is also irreducible in \mathbb{R} . f has a root, $i = \sqrt{-1}$. But $i \notin \mathbb{R}$. If we “insert” i into \mathbb{R} , then f is *not* irreducible in the resulting set, \mathbb{C} . So, from an irreducible polynomial for $\mathbb{R}[x]$, we obtain a $\alpha = i \in \mathbb{C}$.

We do the same thing in \mathbb{F}_q ; Let $m > 1$. Recall then that $\mathbb{F}_q \neq \mathbb{Z}_q$. How to find \mathbb{F}_q ? If f is irreducible in $\mathbb{F}_q[x]$, then we let α be the “that which is the root of f ” (we may not know what it is). Insert α into \mathbb{Z}_q , and you get \mathbb{F}_q .

Lecture 9:

E -C e c t i g
C d e s

P e i i a i e s

A b s t a c t A g e b a

F i e d \mathbb{F}

F i n i t e F i e d \mathbb{F}_p

P o l y n o m i a l s

F i n i t e F i e d \mathbb{F}_{p^m}

L i n e a r A g e b a

V e c t o r S p a c e

T e r m i n o l o g y

L i n e a r B a c k C o d e s

D e f i n i t i o n s

R e c o n s t r u c t i o n C o d e s

P a r i t y C h e c k C o d e s

H a m m i n g C o d e

G i b e t - V a s h a n v

B u d

M i n i m u m

D i s t a n c e

S h e e - P a c k i n g

T h e e

E i g u e

Finding \mathbb{F}_{p^m}

So, how do we use this to find \mathbb{F}_{p^m} ?

Hack: Notice that $f = x^2 + 1$ is also irreducible in \mathbb{R} . f has a root, $i = \sqrt{-1}$. But $i \notin \mathbb{R}$. If we “insert” i into \mathbb{R} , then f is *not* irreducible in the resulting set, \mathbb{C} . So, from an irreducible polynomial for $\mathbb{R}[x]$, we obtain a $\alpha = i \in \mathbb{C}$.

We do the same thing in \mathbb{F}_q ; Let $m > 1$. Recall then that $\mathbb{F}_q \neq \mathbb{Z}_q$. How to find \mathbb{F}_q ? If f is irreducible in $\mathbb{F}_q[x]$, then we let α be the “that which is the root of f ” (we may not know what it is). Insert α into \mathbb{Z}_q , and you get \mathbb{F}_q .

Another Way of finding \mathbb{F}_q for $m > 1$: Each element of $\mathbb{P}_q(m)$ can be *uniquely* represented by an m -tuple over \mathbb{F} .

The Field \mathbb{F}_{2^m}

Q: Why are we so obsessed with \mathbb{F}_{p^m} for $m > 1$? Isn't $m = 1$ enough?

Lecture 9:

Error-Correcting Codes

Peirce's

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_{p^m}

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gilbert-Vandewalle

Bound

Minimum

Distance

Single-Packet

Theorem

Example

The Field \mathbb{F}_{2^m}

- Q:** Why are we so obsessed with \mathbb{F}_{p^m} for $m > 1$? Isn't $m = 1$ enough?
A: While there are infinitely many primes, there are not many primes within a given interval. But *more importantly*, we want to *have access to* \mathbb{F}_q *with* $|\mathbb{F}_q| = 2^m$. These are \mathbb{F}_{2^m} !

Lecture 9:

Encoding Codes

Peirce's

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_{p^m}

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gilbert-Vaughan

Bound

Minimum

Distance

Single-Packet

Theorem

Example

The Field \mathbb{F}_{2^m}

Q: Why are we so obsessed with \mathbb{F}_{p^m} for $m > 1$? Isn't $m = 1$ enough?

A: While there are infinitely many primes, there are not many primes within a given interval. But *more importantly*, we want to *have access to \mathbb{F}_q with $|\mathbb{F}_q| = 2^m$* . These are \mathbb{F}_{2^m} !

Definition (\mathbb{F}_{2^m})

Let $f \in \mathbb{P}_2(m)$ be irreducible in \mathbb{Z}_{2^m} , and $\deg(f) = m$. The finite field \mathbb{F}_{2^m} is a triple $(\mathbb{F}_{2^m}, +, \cdot)$, **where \mathbb{F}_{2^m} can both be interpreted as $\mathbb{P}_2(m)$ and \mathbb{F}_2^m** , $+$ denotes vectoraddition over \mathbb{F}_2^m , and \cdot denotes multiplication over polynomials in $\mathbb{P}_2(m)$ modulo the irreducible element f .

Lecture 9:

E - C e c t i g
C d e s

P e i i a i e s

A b s t a c t A l g e b r a

F i e l d \mathbb{F}

F i n i t e F i e l d \mathbb{F}_p

P o l y n o m i a l s

F i n i t e F i e l d \mathbb{F}_{p^m}

L i n e a r A l g e b r a

V e c t o r S p a c e

T e r m i n o l o g y

L i n e a r B a c k C o d e s

D e f i n i t i o n s

R e c o n s t r u c t i o n C o d e

P a r i t y C h e c k C o d e

H a m m i n g C o d e

G i b e t - V a s h a v

B u d

M i n i m u m

D i s t a n c e

S h e e - P a c k i n g

T h e e

E i g u e

Example

Example

Consider $\mathbb{Z}_2[x]$, which has the irreducible element $f(x) = x^4 + x + 1$. Let $a = (1, 1, 0, 1)$ and $b = (0, 1, 0, 0)$, where $a, b \in \mathbb{F}_2^4$, with corresponding polynomials $a(x) = x^3 + x + 1$ and $b(x) = x$, with $a(x), b(x) \in \mathbb{F}_2[x]$.

$$a + b = (1, 1, 0, 1) + (0, 1, 0, 0) = (1, 0, 0, 1) = c,$$

where $c(x) = x^3 + 1$, and

$$\begin{aligned} a(x) \cdot b(x) &= [(x^3 + x + 1)x]_{x^4+x+1} = x^4 + x^2 + x + 1 - (x^4 + x + 1) \\ &= x^2 - 1 = x^2 + 1 = d(x), \end{aligned}$$

where $d = (1, 0, 1, 0)$.

Outline

1 Preliminaries

- Abstract Algebra
 - Field \mathbb{F}
 - Finite Field \mathbb{F}_p
 - Polynomials
 - Finite Field \mathbb{F}_{p^m}
- Linear Algebra
 - Vector Space
 - Terminology

2 Linear Block Codes

- Definition
- Repetition Code
- Parity Check Code
- Hamming Code

3 Gilbert-Varshamov Bound

- Minimum Distance
- Sphere-Packing
- Theorem

4 Epilogue

Lecture 9:

Encoding Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_{p^m}

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gilbert-Varshamov

Bound

Minimum

Distance

Sphere-Packing

Theorem

Epilogue

E -C ecti g C des

Vect Space

Grabe t-va sha v
B u d

Definition

A vector space is a “set of lists” with $+$ defined as a component-wise sum, and \cdot as a constant scale to each element in the list.

Definition (Vector Space)

A vector space is a quadruple $(V, \mathbb{F}, +, \cdot)$ satisfying

$$\exists 0 \in V. \forall \mathbf{u} \in V. \mathbf{u} + 0 = \mathbf{u} \quad (\text{add-neutral})$$

$$\forall \mathbf{u}, \mathbf{v} \in V. \mathbf{u} + \mathbf{v} = \mathbf{v} + \mathbf{u} \quad (\text{add-comm})$$

$$\forall \mathbf{u}, \mathbf{v}, \mathbf{w} \in V. (\mathbf{u} + \mathbf{v}) + \mathbf{w} = \mathbf{u} + (\mathbf{v} + \mathbf{w}) \quad (\text{add-assoc})$$

$$\forall \mathbf{u} \in V \exists \mathbf{v} \in V. \mathbf{u} + \mathbf{v} = 0 \quad (\text{add-inv})$$

$$\forall \mathbf{u} \in V. \mathbf{u} \cdot 1 = \mathbf{u} \quad (\text{mult-neutral})$$

$$\forall a, b, c \in \mathbb{F}. a(bc) = (ab)c \quad (\text{mult-assoc})$$

$$\forall a \in \mathbb{F}. \forall \mathbf{u}, \mathbf{v} \in V. a(\mathbf{u} + \mathbf{v}) = a\mathbf{u} + a\mathbf{v} \quad (\text{distr1})$$

$$\forall a, b \in \mathbb{F}. \forall \mathbf{u} \in V. (a + b)\mathbf{u} = a\mathbf{u} + b\mathbf{u} \quad (\text{distr2})$$

\mathbb{F} is a *field*, and usually, $V = \mathbb{F}^n$ for some n . When $+$, \cdot , \mathbb{F} are obvious, we denote $(V, \mathbb{F}, +, \cdot)$ by V .

Terminology

Recall:

Base: Every vector space V can be written as a **linear combination** of its **basis**, which is a **linearly independent** list of vectors in V that **spans** V .

Lecture 9:

E -C e c t i g
C d e s

P e i i a i e s

A b s t a c t A g e b a

F i e d \mathbb{F}

F i i t e F i e d \mathbb{F}_p

P y i a s

F i i t e F i e d \mathbb{F}_p

L i e a A g e b a

V e c t S a c e

T e i g y

L i e a B c k C d e s

D e f i t i

R e e t i t i C d e

P a i t y C h e c k C d e

H a i i g C d e

G i b e t - V a s h a v

B u d

M i i u

D i s t a c e

S h e e - P a c k i g

T h e e

E i g u e

Terminology

Recall:

Base: Every vector space V can be written as a **linear combination** of its **basis**, which is a **linearly independent** list of vectors in V that **spans** V .

Dimension: V 's **dimension** is the nr. of elements in any of its bases.

Lecture 9:
E -C ection g
C des

P e i i a i e s
Abst act A g e b a
F i e d \mathbb{F}
F i i t e F i e d \mathbb{F}_p
P y i a s
F i i t e F i e d \mathbb{F}_p
L i e a A g e b a
V e c t S a c e
T e i g y
L i e a B c k C d e s
D e f i t i
R e e t i t i C d e
P a i t y C h e c k C d e
H a i g C d e
G i b e t - V a s h a v
B u d
M i i u
D i s t a c e
S h e e - P a c k i g
T h e e
E i g u e

Terminology

Recall:

Base: Every vector space V can be written as a **linear combination** of its **basis**, which is a **linearly independent** list of vectors in V that **spans** V .

Dimension: V 's **dimension** is the nr. of elements in any of its bases.

Subspace: A **subspace** of V is a subset of V still closed under V 's $+$ and \cdot .

Lecture 9:
E -C ection g
C des

P e i i a i e s
Abst act A g e b a
F i e d \mathbb{F}
F i i t e F i e d \mathbb{F}_p
P y i a s
F i i t e F i e d \mathbb{F}_p
L i e a A g e b a
V e c t S a c e
T e i g y
L i e a B c k C d e s
D e f i t i
R e e t i t i C d e
P a i t y C h e c k C d e
H a i g C d e
G i b e t - V a s h a v
B u d
M i i u
D i s t a c e
S h e e - P a c k i g
T h e e
E i g u e

Terminology

Recall:

Base: Every vector space V can be written as a **linear combination** of its **basis**, which is a **linearly independent** list of vectors in V that **spans** V .

Dimension: V 's **dimension** is the nr. of elements in any of its bases.

Subspace: A **subspace** of V is a subset of V still closed under V 's $+$ and \cdot .

Inner Product: The **inner product** of $\mathbf{u}, \mathbf{v} \in V$ results in a **scalar** in \mathbb{F} :

$$\mathbf{u} \cdot \mathbf{v} = u_0 v_0 + \cdots + u_{n-1} v_{n-1}.$$

Lecture 9:

E -C ection g
C des

P e i i a i e s

Abst act A g e b a

F i e d \mathbb{F}

F i i t e F i e d \mathbb{Z}_p

P y i a s

F i i t e F i e d \mathbb{Z}_p

L i e a A g e b a

V e c t S a c e

T e i g y

L i e a B c k C des

D e f i t i

R e e t i t i C d e

P a i t y C h e c k C d e

H a i g C d e

G i b e t - V a s h a v

B u d

M i i u

D i s t a c e

S h e e - P a c k i g

T h e e

E i g u e

Terminology

Recall:

Base: Every vector space V can be written as a **linear combination** of its **basis**, which is a **linearly independent** list of vectors in V that **spans** V .

Dimension: V 's **dimension** is the nr. of elements in any of its bases.

Subspace: A **subspace** of V is a subset of V still closed under V 's $+$ and \cdot .

Inner Product: The **inner product** of $\mathbf{u}, \mathbf{v} \in V$ results in a **scalar** in \mathbb{F} :

$$\mathbf{u} \cdot \mathbf{v} = u_0 v_0 + \cdots + u_{n-1} v_{n-1}.$$

Orthogonality: \mathbf{u}, \mathbf{v} are **orthogonal** if $\mathbf{u} \cdot \mathbf{v} = 0$.

Lecture 9:

E -C ection
C des

P e i i a i e s

Abst act A g e b a

F i e d \mathbb{F}

F i t e F i e d \mathbb{Z}_p

P y i a s

F i t e F i e d \mathbb{Z}_p

L i e a A g e b a

V e c t S a c e

T e i g y

L i e a B c k C des

D e f i t i

R e e t i t i C d e

P a i t y C h e c k C d e

H a i g C d e

G i b e t - V a s h a v

B u d

M i i u

D i s t a c e

S h e e - P a c k i g

T h e e

E i g u e

Outline

1 Preliminaries

- Abstract Algebra
 - Field \mathbb{F}
 - Finite Field \mathbb{F}_p
 - Polynomials
 - Finite Field \mathbb{F}_{p^m}
- Linear Algebra
 - Vector Space
 - Terminology

2 Linear Block Codes

- Definition
- Repetition Code
- Parity Check Code
- Hamming Code

3 Gilbert-Varshamov Bound

- Minimum Distance
- Sphere-Packing
- Theorem

4 Epilogue

Lecture 9:
Error-Correcting
Codes

Preliminaries

Abstract Algebra

Field \mathbb{F} Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_{p^m}

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gilbert-Varshamov
Bound

Minimum

Distance

Sphere-Packing

Theorem

Epilogue

Limitations

The codes we will work with are all *block codes*.

Lecture 9:

E -C ecti g
C des

P e i i a i e s

Abst act A geb a

Fie d \mathbb{F}

Fi ite Fie d \mathbb{F}_p

P y i a s

Fi ite Fie d \mathbb{F}_p

Li ea A geb a

Vect S ace

Te i gy

Li ea B ck C des

Defi iti

Re etiti C de

Pa ity Check C de

Ha i g C de

Gi be t-Va sha v

B u d

Mi i u

Dista ce

S he e-Packi g

The e

E i gue

Limitations

The codes we will work with are all *block codes*.

- This is okay; if we wanted to use *variable-length codes* \mathcal{C}_{var} , we would use it first, then cut output into snippets of equal length using $\mathcal{C}_{\text{block}}$. That is, $\mathcal{C}(s) = \mathcal{C}_{\text{block}}(\mathcal{C}_{\text{var}}(s))$.

Lecture 9:

Encoding Codes

Prefix Codes

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gibert-Vashkov

Bound

Minimum

Distance

Shannon-Packing

Theorem

Example

Limitations

The codes we will work with are all *block codes*.

- This is okay; if we wanted to use *variable-length codes* \mathcal{C}_{var} , we would use it first, then cut output into snippets of equal length using $\mathcal{C}_{\text{block}}$. That is, $\mathcal{C}(s) = \mathcal{C}_{\text{block}}(\mathcal{C}_{\text{var}}(s))$.

Our codes will also be *linear*. (A normal limitation)

Lecture 9:

Error-Correcting Codes

Prerequisites

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gibert-Vaishnav

Bund

Minimum

Distance

Shannon-Packing

Theorem

Example

Limitations

The codes we will work with are all *block codes*.

- This is okay; if we wanted to use *variable-length codes* \mathcal{C}_{var} , we would use it first, then cut output into snippets of equal length using $\mathcal{C}_{\text{block}}$. That is, $\mathcal{C}(s) = \mathcal{C}_{\text{block}}(\mathcal{C}_{\text{var}}(s))$.

Our codes will also be *linear*. (A normal limitation)

- That is, \mathcal{C} is a *subspace* of some vector space V . Note: $0 \in \mathcal{C}$ always.

Lecture 9:

Error-Correcting Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gilbert-Vaishyanov

Bound

Minimum

Distance

Shannon-Packing

Theorem

Example

Limitations

The codes we will work with are all *block codes*.

- This is okay; if we wanted to use *variable-length codes* \mathcal{C}_{var} , we would use it first, then cut output into snippets of equal length using $\mathcal{C}_{\text{block}}$. That is, $\mathcal{C}(s) = \mathcal{C}_{\text{block}}(\mathcal{C}_{\text{var}}(s))$.

Our codes will also be *linear*. (A normal limitation)

- That is, \mathcal{C} is a *subspace* of some vector space V . Note: $0 \in \mathcal{C}$ always.

At last, we assume code words to be *equiprobable*, and that we use *nearest-neighbour decoding*.

- Asymptotically, this is good. And usually, we cannot tell what is to be sent before-hand anyways.

Lecture 9:
Error-Correcting
Codes

Peel it apart

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomial $\mathbb{F}[x]$

Finite Field $\mathbb{F}_p[x]$

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Codes

Hamming Codes

Gilbert-Variance

Bound

Minimum

Distance

Shannon-Packing

Theorem

Example

Outline

1 Preliminaries

- Abstract Algebra
 - Field \mathbb{F}
 - Finite Field \mathbb{F}_p
 - Polynomials
 - Finite Field \mathbb{F}_{p^m}
- Linear Algebra
 - Vector Space
 - Terminology

2 Linear Block Codes

- Definition
- Repetition Code
- Parity Check Code
- Hamming Code

3 Gilbert-Varshamov Bound

- Minimum Distance
- Sphere-Packing
- Theorem

4 Epilogue

Lecture 9:

Encoding Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_{p^m}

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gilbert-Varshamov Bound

Bound

Minimum

Distance

Sphere-Packing

Theorem

Epilogue

Definition

Strategy:

Pick \mathcal{C} such that all $\mathbf{u} \in \mathcal{C}$ are as far apart from each other as possible (distance measured by metric d_H).

Lecture 9:

E -C eeti g
C des

P e i i a i e s

Abst act A geb a

Fi e d \mathbb{F}

Fi ite Fi e d \mathbb{F}_p

P y i a s

Fi ite Fi e d \mathbb{F}_p

Li ea A geb a

Vect S ace

Te i gy

Li ea B ck C des

Defi iti

Re etiti C de

Pa ity Check C de

Ha i g C de

Gi be t-Va sha v

B u d

Mi i u

Distace

S he e-Packi g

The e

E i gue

Definition

Strategy:

Pick \mathcal{C} such that all $\mathbf{u} \in \mathcal{C}$ are as far apart from each other as possible (distance measured by metric d_H).

Definition (Linear Block Code)

A linear (n, k) block code is a k -dimensional subspace of the n -dimensional space V .

Note: $k < n$, since \mathcal{C} is a subspace of \mathbb{F}^n . Also, $|\mathcal{C}| = M = q^k \leq q^n |\mathbb{F}^n|$, with $|\mathbb{F}| = q$.

Lecture 9:

Error-Correcting Codes

Prerequisites

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gilbert-Varslavsky Bound

Minimum Distance

Minimum Distance

Shannon's Packing Theorem

Theorem

Theorem

Theorem

Theorem

Definition

Strategy:

Pick \mathcal{C} such that all $\mathbf{u} \in \mathcal{C}$ are as far apart from each other as possible (distance measured by metric d_H).

Definition (Linear Block Code)

A linear (n, k) block code is a k -dimensional subspace of the n -dimensional space V .

Note: $k < n$, since \mathcal{C} is a subspace of \mathbb{F}^n . Also, $|\mathcal{C}| = M = q^k \leq q^n |\mathbb{F}^n|$, with $|\mathbb{F}| = q$. At last,

$$R = \frac{\log_q M}{n} = \frac{\log_q q^k}{n} = \frac{k}{n},$$

since \mathbb{F} is the input alphabet per-symbol, before we put symbols together into “blocks”.

Lecture 9:

Error-Correcting Codes

Prerequisites

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_q

Polynomials

Finite Field \mathbb{F}_q

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gilbert-Vaishagan

Bound

Minimum

Distance

Sphere-Packing

Theorem

Example

Outline

1 Preliminaries

- Abstract Algebra
 - Field \mathbb{F}
 - Finite Field \mathbb{F}_p
 - Polynomials
 - Finite Field \mathbb{F}_{p^m}
- Linear Algebra
 - Vector Space
 - Terminology

2 Linear Block Codes

- Definition
- Repetition Code
- Parity Check Code
- Hamming Code

3 Gilbert-Varshamov Bound

- Minimum Distance
- Sphere-Packing
- Theorem

4 Epilogue

Lecture 9: Repetition Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_{p^m}

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gilbert-Varshamov

Bound

Minimum

Distance

Sphere-Packing

Theorem

Epilogue

Repetition Codes (draw d_H graph for $n = 3$)

You have already seen linear block codes. For instance, the *repetition codes*.

Input Alphabet: \mathbb{F}_q .

Code: $\mathcal{R}_n = \{00 \dots 0, 11 \dots 1\}$ is the $k = 1$ -dimensional subspace of \mathbb{F}_q^n .

Error-Correction: When n is odd, \mathcal{R}_n corrects $\frac{n-1}{2}$ errors. When n is even, it can *detect* 1 more error.

Error-Detection: When n is even it can *detect* 1 more error than it can correct.

Rate: $R = \frac{1}{n}$. So, $R \rightarrow 0$ as $n \rightarrow \infty$.

Lecture 9:

E -C ecti g
C des

P e i i a i e s

Abst act A geb a

Fie d \mathbb{F}

Fi ite Fie d \mathbb{F}_p

P y i a s

Fi ite Fie d \mathbb{F}_p

Li ea A geb a

Vect S ace

Te i gy

Li ea B ck C des

Defi iti

Re etiti C de

Pa ity Check C de

Ha i g C de

Gi be t-Va sha v

B u d

Mi i u

Dista ce

S he e-Packi g

The e

E i gue

Repetition Codes (draw d_H graph for $n = 3$)

You have already seen linear block codes. For instance, the *repetition codes*.

Input Alphabet: \mathbb{F}_q .

Code: $\mathcal{R}_n = \{00 \cdots 0, 11 \cdots 1\}$ is the $k = 1$ -dimensional subspace of \mathbb{F}_q^n .

Error-Correction: When n is odd, \mathcal{R}_n corrects $\frac{n-1}{2}$ errors. When n is even, it can *detect* 1 more error.

Error-Detection: When n is even it can *detect* 1 more error than it can correct.

Rate: $R = \frac{1}{n}$. So, $R \rightarrow 0$ as $n \rightarrow \infty$.

Example (Repetition Code)

Let $q = 2$ and $n = 3$. Then

$$\mathbb{F}_q = \mathbb{F}_2 = \{0, 1\}$$

and

$$\mathcal{R}_n = \mathcal{R}_3 = \{000, 111\}.$$

So \mathcal{R}_3 corrects 1 error.

Parity Check Codes

If we expect few errors, it might be sufficient to be able to *detect* them.

Usage: Many low-level architectures (microprocessors, RAID), an telecommunication standards like TCP/IP.

Lecture 9:

E -C ecti g C des

P e i i a i es

Abst act A geb a

Fie d \mathbb{F}

Fi ite Fie d \mathbb{F}_p

P y i a s

Fi ite Fie d \mathbb{F}_p

Li ea A geb a

Vect S ace

Te i gy

Li ea B ck C des

Defi iti

Re etiti C de

Pa ity Check C de

Ha i g C de

Gi be t-Va sha v

B u d

Mi i u

Distace

S he e-Packi g

The e

E i gue

Parity Check Codes

If we expect few errors, it might be sufficient to be able to *detect* them.

Usage: Many low-level architectures (microprocessors, RAID), an telecommunication standards like TCP/IP.

Input Alphabet: \mathbb{F}_q .

Code: $\mathcal{P}_n = \{u_1 \cdots u_n \in \mathbb{F}_q^n \mid \sum_i u_i = 0\}$. This is “half of \mathbb{F}_q^n ”.
So $k = n - 1$ (remove a digit from $u_1 \cdots u_n \in \mathbb{F}_q^n$).

Error-Correction: 0.

Error-Detection: 1.

Rate: $R = \frac{k}{n} = \frac{n-1}{n} = 1$. So, $R \rightarrow 1$ as $n \rightarrow \infty$.

Lecture 9:

E -C ecti g
C des

P e i i a i e s

Abst act A geb a

Fie d \mathbb{F}

Fi ite Fie d \mathbb{F}_p

P y i a s

Fi ite Fie d \mathbb{F}_p

Li ea A geb a

Vect S ace

Te i g y

Li ea B ck C des

Defi iti

Re etiti C de

Pa ity Check C de

Ha i g C de

Gi be t-Va sha v

B u d

Mi i u

Dista ce

S he e-Packi g

The e

E i gue

Parity Check Codes

If we expect few errors, it might be sufficient to be able to *detect* them.

Usage: Many low-level architectures (microprocessors, RAID), an telecommunication standards like TCP/IP.

Input Alphabet: \mathbb{F}_q .

Code: $\mathcal{P}_n = \{u_1 \cdots u_n \in \mathbb{F}_q^n \mid \sum_i u_i = 0\}$. This is “half of \mathbb{F}_q^n ”.
So $k = n - 1$ (remove a digit from $u_1 \cdots u_n \in \mathbb{F}_q^n$).

Error-Correction: 0.

Error-Detection: 1.

Rate: $R = \frac{k}{n} = \frac{n-1}{n} = 1$. So, $R \rightarrow 1$ as $n \rightarrow \infty$.

But it corrects no error, and detects only 1. No improvement for larger n , so the bigger n gets, if \bar{P} stays the same, then the bigger the chance of more than 1 digit flipping. If 2 digits flip, we get a *different code word*, and are none the wiser.

Lecture 9:

Error-Correcting Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gibbs-Vaughan

Bound

Minimum

Distance

Shannon-Packing

Theorem

Example

Parity Check Codes

If we expect few errors, it might be sufficient to be able to *detect* them.

Usage: Many low-level architectures (microprocessors, RAID), an telecommunication standards like TCP/IP.

Input Alphabet: \mathbb{F}_q .

Code: $\mathcal{P}_n = \{u_1 \cdots u_n \in \mathbb{F}_q^n \mid \sum_i u_i = 0\}$. This is “half of \mathbb{F}_q^n ”.
So $k = n - 1$ (remove a digit from $u_1 \cdots u_n \in \mathbb{F}_q^n$).

Error-Correction: 0.

Error-Detection: 1.

Rate: $R = \frac{k}{n} = \frac{n-1}{n} = 1$. So, $R \rightarrow 1$ as $n \rightarrow \infty$.

But it corrects no error, and detects only 1. No improvement for larger n , so the bigger n gets, if \bar{P} stays the same, then the bigger the chance of more than 1 digit flipping. If 2 digits flip, we get a *different code word*, and are none the wiser.

Example (Parity Check Code)

Let $q = 2$ and $n = 3$. Then $\mathbb{F}_q = \mathbb{F}_2 = \{0, 1\}$ and

$$\mathcal{P}_n = \{000, 011, 101, 110\}$$

Distance between words is 2.

Outline

1 Preliminaries

- Abstract Algebra
 - Field \mathbb{F}
 - Finite Field \mathbb{F}_p
 - Polynomials
 - Finite Field \mathbb{F}_{p^m}
- Linear Algebra
 - Vector Space
 - Terminology

2 Linear Block Codes

- Definition
- Repetition Code
- Parity Check Code
- Hamming Code

3 Gilbert-Varshamov Bound

- Minimum Distance
- Sphere-Packing
- Theorem

4 Epilogue

Lecture 9:

Encoding Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_{p^m}

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gilbert-Varshamov

Bound

Minimum

Distance

Sphere-Packing

Theorem

Epilogue

Hamming Codes

Developed by Hamming in frustration at Bell Labs in 1947 to compensate for system crashes while doing weekend-work. **See page 102**, fig. 6.3.

Usage: RAM. $\mathcal{H}_7 : \mathbb{F}_2^4 \rightarrow \mathbb{F}_2^7$. (recall: $\mathbb{F}_2^4 \simeq \mathbb{F}_{2^4}$)

Construction: Map $\alpha = \alpha_1\alpha_2\alpha_4\alpha_4$ to $\mathbf{u} = u_1u_2u_3u_4u_5u_6u_7$ as follows:

u_1 : *Pick s.t. equation $0 = u_1 + u_3 + u_5 + u_7$ is satisfied.*

u_2 : *Pick s.t. equation $0 = u_2 + u_3 + u_6 + u_7$ is satisfied.*

u_3 : α_1 .

u_4 : *Pick s.t. equation $0 = u_4 + u_5 + u_6 + u_7$ is satisfied.*

u_5 : α_2 .

u_6 : α_3 .

u_7 : α_4 .

The Idea: 3 parity bits, arranged s.t. *each* bit in α has 2 parity bits. And in a sense, the parity has parity as well. This arrangement allows *minimum-distance decoding* to not just detect, but *correct*, 1 error.

Lecture 9:
Error-Correcting
Codes

Peeliiaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Cyclic Codes

Bur

Minimum

Distance

Shannon's Packing

Theorem

Erasure

Improvement?

Decoding: On receiving $\mathbf{v} = v_1 \cdots v_7$, check parity:

$s_1 := v_4 + v_5 + v_6 + v_7$. *Compare to v_4 .*

$s_2 := v_2 + v_3 + v_6 + v_7$. *Compare to v_2 .*

$s_3 := v_1 + v_3 + v_5 + v_7$. *Compare to v_1 .*

If mismatch, then an error has occurred. Where: in digit $s_1 s_2 s_3$ *base 10*.
All goes well if no more than 1 error.

Example

Encode 1101, and introduce an error in the 6th position.

Q: How is this better than \mathcal{R}_n ?

A: Well,

- \mathcal{R}_3 has $R = \frac{1}{3}$. \mathcal{H}_7 has $R = \frac{4}{7}$.
- $|\mathcal{R}_3| = 2$ while $|\mathcal{H}_7| = 16$. That's *good*.

This idea can be *generalised* to bigger n . Then $R \rightarrow 1$ as $n \rightarrow \infty$.

Outline

1 Preliminaries

- Abstract Algebra
 - Field \mathbb{F}
 - Finite Field \mathbb{F}_p
 - Polynomials
 - Finite Field \mathbb{F}_{p^m}
- Linear Algebra
 - Vector Space
 - Terminology

2 Linear Block Codes

- Definition
- Repetition Code
- Parity Check Code
- Hamming Code

3 Gilbert-Varshamov Bound

- Minimum Distance
- Sphere-Packing
- Theorem

4 Epilogue

Lecture 9: Error-Correcting Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_{p^m}

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gilbert-Vašha v Bound

Minimum

Distance

Sphere-Packing

Theorem

Epilogue

Take a Step Back

Recall our goal:

Pick \mathcal{C} such that all $\mathbf{u} \in \mathcal{C}$ are as far apart from each other as possible (distance measured by metric d_H).

Lecture 9:

Encoding Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gibert-Vasanth Bud

Minimality

Distance

Shannon's Packing

Theorem

Example

Take a Step Back

Recall our goal:

Pick \mathcal{C} such that all $\mathbf{u} \in \mathcal{C}$ are as far apart from each other as possible (distance measured by metric d_H).

Herein there is a **conflict of interest**. Consider \mathbb{F}_q as a *space*.

- We want many elements of \mathbb{F}_q in \mathcal{C} .
- We want the elements we pick into \mathcal{C} to be far away from each other.

Lecture 9:

Error-Correcting Codes

Prerequisites

Abstract Algebra

Field \mathbb{F} Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gibert-Vasanth Bud

Minimum

Distance

Sphere-Packing

Theorem

Erasure

Take a Step Back

Recall our goal:

Pick \mathcal{C} such that all $\mathbf{u} \in \mathcal{C}$ are as far apart from each other as possible (distance measured by metric d_H).

Herein there is a **conflict of interest**. Consider \mathbb{F}_q as a *space*.

- We want many elements of \mathbb{F}_q in \mathcal{C} .
- We want the elements we pick into \mathcal{C} to be far away from each other.

If we add an element from \mathbb{F}_q into \mathcal{C} , then \mathcal{C} 's portion of \mathbb{F}_q *increases*. So the words in \mathcal{C} will be *closer* in \mathbb{F}_q . Let's study this relationship.

Lecture 9:

E -Constructing Codes

Preliminaries

Abstract Algebra

Field \mathbb{F} Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gibert-Vasanth Bud

Minimality

Distance

Sphere-Packing

Theorem

Exercises

Outline

- 1 Preliminaries
 - Abstract Algebra
 - Field \mathbb{F}
 - Finite Field \mathbb{F}_p
 - Polynomials
 - Finite Field \mathbb{F}_{p^m}
 - Linear Algebra
 - Vector Space
 - Terminology
- 2 Linear Block Codes
 - Definition
 - Repetition Code
 - Parity Check Code
 - Hamming Code
- 3 Gilbert-Varshamov Bound
 - Minimum Distance
 - Sphere-Packing
 - Theorem
- 4 Epilogue

Lecture 9: Error-Correcting Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_{p^m}

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gilbert-Varesham Bound

Bound

Minimum

Distance

Sphere-Packing

Theorem

Epilogue

Maximise Distance to Improve Error Correctness

To ensure that elements in \mathcal{C} are all *far away* from each other, we introduce the notion of *minimum distance* between words.

Maximise Distance to Improve Error Correctness

To ensure that elements in \mathcal{C} are all *far away* from each other, we introduce the notion of *minimum distance* between words.

Definition (Minimum Distance)

The *minimum distance*, d , between codewords in a code \mathcal{C} is defined by

$$d = d(\mathcal{C}) = \min\{d_H(\mathbf{u}, \mathbf{u}') \mid \mathbf{u}, \mathbf{u}' \in \mathcal{C}, \mathbf{u} \neq \mathbf{u}'\}$$

Lecture 9:
Error-Correcting
Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gibert-Vashev Bud

Minimum

Distance

Shannon-Packet

Theorem

Example

Exercise

Maximise Distance to Improve Error Correctness

To ensure that elements in \mathcal{C} are all *far away* from each other, we introduce the notion of *minimum distance* between words.

Definition (Minimum Distance)

The *minimum distance*, d , between codewords in a code \mathcal{C} is defined by

$$d = d(\mathcal{C}) = \min \{d_H(\mathbf{u}, \mathbf{u}') \mid \mathbf{u}, \mathbf{u}' \in \mathcal{C}, \mathbf{u} \neq \mathbf{u}'\}$$

We sometimes refer to \mathcal{C} as a (n, k, d) -code.

Example

\mathcal{R}_3 has $d = 3$. So does \mathcal{H}_7 .

Lecture 9:
Error-Correcting
Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gibet-Vasha v

Bud

Mi i u

Distance

Shannon-Packing

Theorem

Example

Maximise Distance to Improve Error Correctness

To ensure that elements in \mathcal{C} are all *far away* from each other, we introduce the notion of *minimum distance* between words.

Definition (Minimum Distance)

The *minimum distance*, d , between codewords in a code \mathcal{C} is defined by

$$d = d(\mathcal{C}) = \min \{d_H(\mathbf{u}, \mathbf{u}') \mid \mathbf{u}, \mathbf{u}' \in \mathcal{C}, \mathbf{u} \neq \mathbf{u}'\}$$

We sometimes refer to \mathcal{C} as a (n, k, d) -code.

Example

\mathcal{R}_3 has $d = 3$. So does \mathcal{H}_7 .

Q: Computing d is going to be *living hell*! Do you really expect us to make $\binom{M}{2}$ comparisons?

A: Fortunately, no. Since our \mathcal{C} are *linear*, we can do something much simpler.

Lectu e 9:
E -C ecti g
C des

P e i i a i e s

Abst act A geb a

Fie d F

Fi ite Fie d \mathbb{F}_p

P y i a s

Fi ite Fie d \mathbb{F}_p

Li ea A geb a

Vect S ace

Te i gy

Li ea B ck C des

Defi iti

Re etiti C de

Pa ity Check C de

Ha i g C de

Gi b e t-Va sha v

B u d

Mi i u

Dista ce

S he e-Packi g

The e

E i gue

Computing d

Let $\text{wt}(\mathbf{v}) = d_H(\mathbf{v}, 0)$. Note that $d_H(\mathbf{u}, \mathbf{u}') = \text{wt}(\mathbf{u} - \mathbf{u}')$.

Lemma

If \mathcal{C} is linear, then d is given by

$$d = \min \{\text{wt}(\mathbf{v}) \mid \mathbf{v} \in \mathcal{C}, \mathbf{v} \neq 0\}.$$

Proof.

Since \mathcal{C} is a linear subspace of some V , then $\mathbf{v} = \mathbf{u} - \mathbf{u}'$ for different $\mathbf{u}, \mathbf{u}' \in \mathcal{C}$ is also in \mathcal{C} . We have $d_H(\mathbf{u}, \mathbf{u}') = \text{wt}(\mathbf{v})$. \mathbf{v} ranges over all non-zero elements of \mathcal{C} . Thus $d(\mathcal{C}) = \min \{\text{wt}(\mathbf{v}) \mid \mathbf{v} \in \mathcal{C}, \mathbf{v} \neq 0\}$. □

Now all we have to do is compute $\{\text{wt}(\mathbf{v}) \mid \mathbf{v} \in \mathcal{C}, \mathbf{v} \neq 0\}$, and pick the least (linear time). Contains $|\mathcal{C}|$ elements.

Lecture 9:
E -C ectig
C des

Pe i i aies

Abst act A geb a

Fie d \mathbb{F}

Fi ite Fie d \mathbb{F}_p

P y ias

Fi ite Fie d \mathbb{F}_p

Li ea A geb a

Vect S ace

Te i gy

Li ea B ck C des

Def iti

Re etiti C de

Pa ity Check C de

Ha i g C de

Gibert-Vašara v

B u d

Mi i u

Distace

S he e-Packig

The e

E i gue

t-error Correcting

We have seen that d is intimately connected with *how many* errors \mathcal{C} can correct.

Lecture 9:

Error-Correcting Codes

Prerequisites

Abstract Algebra

Field \mathbb{F} Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gibert-Vasiliu

Project

Mini-Project

Distance

Shannon's Packing

Theorem

Exercises

t-error Correcting

We have seen that d is intimately connected with *how many* errors \mathcal{C} can correct.

Definition (t-error-correcting (compare to graph))

A code \mathcal{C} is t-error-correcting if for any two codewords \mathbf{u}, \mathbf{v} , you *cannot* flip up to t symbols in \mathbf{u} and \mathbf{v} , yielding \mathbf{u}', \mathbf{v}' , such that $\mathbf{u}' = \mathbf{v}' \in \mathcal{C}$.

Lecture 9:

Error-Correcting Codes

Prerequisites

Abstract Algebra

Field \mathbb{F} Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gibets-Vashayevs Budd

Minut

Distance

Shannon's Packing

Theorem

Example

t-error Correcting

We have seen that d is intimately connected with *how many* errors \mathcal{C} can correct.

Definition (t-error-correcting (compare to graph))

A code \mathcal{C} is t-error-correcting if for any two codewords \mathbf{u}, \mathbf{v} , you *cannot* flip up to t symbols in \mathbf{u} and \mathbf{v} , yielding \mathbf{u}', \mathbf{v}' , such that $\mathbf{u}' = \mathbf{v}' \in \mathcal{C}$.

We represent the “flipped bits” that brought \mathbf{u} to \mathbf{u}' as the *error pattern* \mathbf{e} . So $\mathbf{u} + \mathbf{e} = \mathbf{u}'$. Then \mathbf{e} is allowed to have $\text{wt}(\mathbf{e}) \leq t$.

Lecture 9:
Error-Correcting
Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gibet-Va sha v

Bud

Mi i u

Distance

Shannon-Packing

Theorem

Example

t-error Correcting

We have seen that d is intimately connected with *how many* errors \mathcal{C} can correct.

Definition (t-error-correcting (compare to graph))

A code \mathcal{C} is t-error-correcting if for any two codewords \mathbf{u}, \mathbf{v} , you *cannot* flip up to t symbols in \mathbf{u} and \mathbf{v} , yielding \mathbf{u}', \mathbf{v}' , such that $\mathbf{u}' = \mathbf{v}' \in \mathcal{C}$.

We represent the “flipped bits” that brought \mathbf{u} to \mathbf{u}' as the *error pattern* \mathbf{e} . So $\mathbf{u} + \mathbf{e} = \mathbf{u}'$. Then \mathbf{e} is allowed to have $\text{wt}(\mathbf{e}) \leq t$.

Q: We know \mathcal{H}_7 is 1-error-correcting. How can we easily check that it is *not* 2-error-correcting?

Lecture 9:

E -C ecti g
C des

P e i i a i es

Abst act A geb a

Fie d \mathbb{F} Fi ite Fie d \mathbb{F}_p

P y i a s

Fi ite Fie d \mathbb{F}_p

Li ea A geb a

Vect S ace

Te i gy

Li ea B ck C des

Defi iti

Re etiti C de

Pa ity Check C de

Ha i g C de

Gi be t-Va sha v

B u d

Mi i u

Dista ce

S he e-Packi g

The e

E i gue

t-error Correcting

We have seen that d is intimately connected with *how many* errors \mathcal{C} can correct.

Definition (t-error-correcting (compare to graph))

A code \mathcal{C} is t-error-correcting if for any two codewords \mathbf{u}, \mathbf{v} , you *cannot* flip up to t symbols in \mathbf{u} and \mathbf{v} , yielding \mathbf{u}', \mathbf{v}' , such that $\mathbf{u}' = \mathbf{v}' \in \mathcal{C}$.

We represent the “flipped bits” that brought \mathbf{u} to \mathbf{u}' as the *error pattern* \mathbf{e} . So $\mathbf{u} + \mathbf{e} = \mathbf{u}'$. Then \mathbf{e} is allowed to have $\text{wt}(\mathbf{e}) \leq t$.

Q: We know \mathcal{H}_7 is 1-error-correcting. How can we easily check that it is *not* 2-error-correcting?

A: Behold:

Lecture 9:
Error-Correcting
Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gibet-Va sha v

B u d

M i i u

Dista ce

Shannon-Packing

Theorem

Example

Error-Correction of Linear Codes

Theorem

\mathcal{C} corrects t errors iff $d \geq 2t + 1$ (equivalently, \mathcal{C} corrects up to $\lfloor \frac{d-1}{2} \rfloor$ errors).

Proof

Each direction of iff.

Lecture 9: Error-Correcting Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gibert-Vašara v

Bud

Mini-Distance

Shannon-Packing

Theorem

Example

Exercise

Error-Correction of Linear Codes

Theorem

\mathcal{C} corrects t errors iff $d \geq 2t + 1$ (equivalently, \mathcal{C} corrects up to $\lfloor \frac{d-1}{2} \rfloor$ errors).

Proof

Each direction of iff.

\Leftarrow : Let $d \geq 2t + 1$. Assume $\mathbf{u} \in \mathcal{C}$ is sent and $\mathbf{v} = \mathbf{u} + \mathbf{e} \in V$ received.

Lecture 9:
Error-Correcting
Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gibert-Vašara v Bud

Bud

Mini-Distance

Shannon's Packing

Theorem

Example

Example

Error-Correction of Linear Codes

Theorem

\mathcal{C} corrects t errors iff $d \geq 2t + 1$ (equivalently, \mathcal{C} corrects up to $\lfloor \frac{d-1}{2} \rfloor$ errors).

Proof

Each direction of iff.

\Leftarrow : Let $d \geq 2t + 1$. Assume $\mathbf{u} \in \mathcal{C}$ is sent and $\mathbf{v} = \mathbf{u} + \mathbf{e} \in V$ received.
For all $\mathbf{u}' \in \mathcal{C}$; $\mathbf{u}' \neq \mathbf{u}$,

$$d_H(\mathbf{u}, \mathbf{u}') \geq d \geq 2t + 1.$$

Lecture 9:
Error-Correcting
Codes

Prerequisites

Abstract Algebra

Field \mathbb{F} Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gibert-Vašara v Bud

Mini u

Distance

Shannon-Packing

Theorem

Error-Correction

Error-Correction

Error-Correction

Error-Correction of Linear Codes

Theorem

\mathcal{C} corrects t errors iff $d \geq 2t + 1$ (equivalently, \mathcal{C} corrects up to $\lfloor \frac{d-1}{2} \rfloor$ errors).

Proof

Each direction of iff.

\Leftarrow : Let $d \geq 2t + 1$. Assume $\mathbf{u} \in \mathcal{C}$ is sent and $\mathbf{v} = \mathbf{u} + \mathbf{e} \in V$ received.
For all $\mathbf{u}' \in \mathcal{C}$; $\mathbf{u}' \neq \mathbf{u}$,

$$d_H(\mathbf{u}, \mathbf{u}') \geq d \geq 2t + 1.$$

By *triangle inequality* (d_H is a metric):

$$d_H(\mathbf{u}, \mathbf{u}') \leq d_H(\mathbf{u}, \mathbf{v}) + d_H(\mathbf{v}, \mathbf{u}').$$

Prerequisites

Abstract Algebra

Field F Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Codes

Hamming Codes

Gibert-Vašara v

Bud

Mit

Distance

Shannon-Packing

Theorem

Exercise

Error-Correction of Linear Codes

Theorem

\mathcal{C} corrects t errors iff $d \geq 2t + 1$ (equivalently, \mathcal{C} corrects up to $\lfloor \frac{d-1}{2} \rfloor$ errors).

Proof

Each direction of iff.

\Leftarrow : Let $d \geq 2t + 1$. Assume $\mathbf{u} \in \mathcal{C}$ is sent and $\mathbf{v} = \mathbf{u} + \mathbf{e} \in V$ received.
For all $\mathbf{u}' \in \mathcal{C}$; $\mathbf{u}' \neq \mathbf{u}$,

$$d_H(\mathbf{u}, \mathbf{u}') \geq d \geq 2t + 1.$$

By *triangle inequality* (d_H is a metric):

$$d_H(\mathbf{u}, \mathbf{u}') \leq d_H(\mathbf{u}, \mathbf{v}) + d_H(\mathbf{v}, \mathbf{u}').$$

Together, we get that $\Delta(\mathbf{v}) = \mathbf{u}$, since

$$d_H(\mathbf{v}, \mathbf{u}') \geq d_H(\mathbf{u}, \mathbf{u}') - d_H(\mathbf{u}, \mathbf{v}) \geq (2t + 1) - t = t + 1 > d_H(\mathbf{u}, \mathbf{v}).$$

Cont.

Lecture 9:
Error-Correcting
Codes

Preliminaries

Abstract Algebra

Field F

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gibert-Vašara v Bud

Bud

Mitu Distance

Shannon-Packing

Theorem

Example

Example

continued.

\Rightarrow : By contradiction: Assume $d < 2t + 1$, that is, $d \leq 2t$ (that we can correct more than $\lfloor \frac{d-1}{2} \rfloor$ errors).

Lecture 9:

E -C -ectig
C des

P e i i a i e s

Abst act A geb a

Fie d \mathbb{F}

Fi ite Fie d \mathbb{F}_p

P y i a s

Fi ite Fie d \mathbb{F}_p

Li ea A geb a

Vect S ace

Te i gy

Li ea B ck C des

Defi iti

Re etiti C de

Pa ity Check C de

Ha i g C de

Gibert-Vasilev

Burd

Mini

Distance

Shee-Packig

The e

E i gue

continued.

\Rightarrow : By contradiction: Assume $d < 2t + 1$, that is, $d \leq 2t$ (that we can correct more than $\lfloor \frac{d-1}{2} \rfloor$ errors). Choose $\mathbf{u}, \mathbf{u}' \in \mathcal{C}$ such that $d_H(\mathbf{u}, \mathbf{u}') = d$.

Lecture 9:

Error-Correcting Codes

Prerequisites

Abstract Algebra

Field \mathbb{F} Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gibert-Vašar Bud

Bud

Mit

Distance

Shannon's Packing

Theorem

Example

continued.

\implies : By contradiction: Assume $d < 2t + 1$, that is, $d \leq 2t$ (that we can correct more than $\lfloor \frac{d-1}{2} \rfloor$ errors). Choose $\mathbf{u}, \mathbf{u}' \in \mathcal{C}$ such that $d_H(\mathbf{u}, \mathbf{u}') = d$. Since $d \leq 2t$, there is a vector $\mathbf{v} \in V$ with

$$d_H(\mathbf{u}, \mathbf{v}) \leq t \quad \text{and} \quad d_H(\mathbf{u}', \mathbf{v}) \leq t.$$

Lecture 9:

Error-Correcting Codes

Prerequisites

Abstract Algebra

Field \mathbb{F} Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gibert-Vašara v

Bud

Mi i u

Distance

Shannon-Packing

Theorem

Exercises

continued.

\Rightarrow : By contradiction: Assume $d < 2t + 1$, that is, $d \leq 2t$ (that we can correct more than $\lfloor \frac{d-1}{2} \rfloor$ errors). Choose $\mathbf{u}, \mathbf{u}' \in \mathcal{C}$ such that $d_H(\mathbf{u}, \mathbf{u}') = d$. Since $d \leq 2t$, there is a vector $\mathbf{v} \in V$ with

$$d_H(\mathbf{u}, \mathbf{v}) \leq t \quad \text{and} \quad d_H(\mathbf{u}', \mathbf{v}) \leq t.$$

However, $\Delta(\mathbf{v})$ could now decode to *either* \mathbf{u} or \mathbf{u}' , a contradiction. (spheres overlap. \mathbf{v} in overlap)



Lecture 9:

Error-Correcting Codes

Prerequisites

Abstract Algebra

Field \mathbb{F} Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gibert-Vashayev

Bud

Mitut

Distance

Sphere-Packing

Theorem

Exercise

Outline

1 Preliminaries

- Abstract Algebra
 - Field \mathbb{F}
 - Finite Field \mathbb{F}_p
 - Polynomials
 - Finite Field \mathbb{F}_{p^m}
- Linear Algebra
 - Vector Space
 - Terminology

2 Linear Block Codes

- Definition
- Repetition Code
- Parity Check Code
- Hamming Code

3 Gilbert-Varshamov Bound

- Minimum Distance
- Sphere-Packing
- Theorem

4 Epilogue

Lecture 9:
Error-Correcting
Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_{p^m}

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gilbert-Varshamov
Bound

Minimum

Distance

Sphere-Packing

Theorem

Epilogue

Motivation

So, the greatest t is the maximal **radius** of spheres around $\mathbf{u} \in \mathcal{C}$ such that spheres do not overlap.

Lecture 9:

Encoding
Codes

Prerequisites

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gibert-Vashev Bud

Bud

Minimum

Distance

Shee-Packig

Theorem

Example

Motivation

So, the greatest t is the maximal **radius** of spheres around $\mathbf{u} \in \mathcal{C}$ such that spheres do not overlap. Remember the conflict:

- We want many elements of \mathbb{F}_q in \mathcal{C} .
- We want the elements we pick into \mathcal{C} to be far away from each other.

Lecture 9:
E -C ectig
C des

Pe i i aies

Abst act A geb a

Fie d \mathbb{F}

Fi ite Fie d \mathbb{F}_p

P y i a s

Fi ite Fie d \mathbb{F}_p

Li ea A geb a

Vect S ace

Te i gy

Li ea B ck C des

Def i ti

Re etiti C de

Pa ity Check C de

Ha i g C de

Gibert-Vashev Bud

B u d

Mi i u

Dist ace

Shee-Packig

The e

E i gue

Motivation

So, the greatest t is the maximal **radius** of spheres around $\mathbf{u} \in \mathcal{C}$ such that spheres do not overlap. Remember the conflict:

- We want many elements of \mathbb{F}_q in \mathcal{C} .
- We want the elements we pick into \mathcal{C} to be far away from each other.

How exactly do these conflict? For q, n fixed, we have a “box” of limited size, and then we have a large collection of **cubes** (which we call *spheres*)

$$S_t(\mathbf{u}) = \{\mathbf{v} \in V \mid d_H(\mathbf{u}, \mathbf{v}) \leq t\} \quad ; \quad \mathbf{u} \in \mathcal{C}$$

of different sizes.

Lecture 9:
Error-Correcting
Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Codes

Hamming Codes

Gibert-Vasanth Bud

Bud

Minimality

Distance

Shee-Packig

Theorem

Example

Motivation

So, the greatest t is the maximal **radius** of spheres around $\mathbf{u} \in \mathcal{C}$ such that spheres do not overlap. Remember the conflict:

- We want many elements of \mathbb{F}_q in \mathcal{C} .
- We want the elements we pick into \mathcal{C} to be far away from each other.

How exactly do these conflict? For q, n fixed, we have a “box” of limited size, and then we have a large collection of **cubes** (which we call *spheres*)

$$S_t(\mathbf{u}) = \{\mathbf{v} \in V \mid d_H(\mathbf{u}, \mathbf{v}) \leq t\} \quad ; \quad \mathbf{u} \in \mathcal{C}$$

of different sizes. We can *at best* fill the box completely. This gives an *upper bound* to *sphere packing*.

Lecture 9:
Error-Correcting
Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gibert-Vasanth Bund

Bound

Minimum

Distance

Sphere-Packing

Theorem

Exercise

Theorem (Hamming's Sphere-Packing Bound)

Let \mathcal{C} be a q -ary t -error-correcting code of length n , with M code-words. Then

$$M \left(1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \dots + \binom{n}{t}(q-1)^t \right) \leq q^n.$$

Terms after M : First 1 is \mathbf{u} . $\binom{n}{1}(q-1)$ are all \mathbf{v} distance 1 from \mathbf{u} and so on.

Theorem (Hamming's Sphere-Packing Bound)

Let \mathcal{C} be a q -ary t -error-correcting code of length n , with M code-words. Then

$$M \left(1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \cdots + \binom{n}{t}(q-1)^t \right) \leq q^n.$$

Terms after M : First 1 is \mathbf{u} . $\binom{n}{1}(q-1)$ are all \mathbf{v} distance 1 from \mathbf{u} and so on.

Proof.

There are M spheres; one for each $\mathbf{u} \in \mathcal{C}$.

Lecture 9:
E -C ecti g
C des

P e i i a i e s

Abst act A geb a

Fie d \mathbb{F}

Fi ite Fie d \mathbb{F}_p

P y i a s

Fi ite Fie d \mathbb{F}_p

Li ea A geb a

Vect S ace

Te i gy

Li ea B ck C des

Def iti

Re etiti C de

Pa ity Check C de

Ha i g C de

Gi be t-Va sha v

B u d

Mi i u

Distace

S he e-Packi g

The e

E i gue

Theorem (Hamming's Sphere-Packing Bound)

Let \mathcal{C} be a q -ary t -error-correcting code of length n , with M code-words. Then

$$M \left(1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \cdots + \binom{n}{t}(q-1)^t \right) \leq q^n.$$

Terms after M : First 1 is \mathbf{u} . $\binom{n}{1}(q-1)$ are all \mathbf{v} distance 1 from \mathbf{u} and so on.

Proof.

There are M spheres; one for each $\mathbf{u} \in \mathcal{C}$. There are $\binom{n}{i}(q-1)^i$ vectors \mathbf{v} with $d_H(\mathbf{u}, \mathbf{v}) = i$.

Theorem (Hamming's Sphere-Packing Bound)

Let \mathcal{C} be a q -ary t -error-correcting code of length n , with M code-words. Then

$$M \left(1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \dots + \binom{n}{t}(q-1)^t \right) \leq q^n.$$

Terms after M : First 1 is \mathbf{u} . $\binom{n}{1}(q-1)$ are all \mathbf{v} distance 1 from \mathbf{u} and so on.

Proof.

There are M spheres; one for each $\mathbf{u} \in \mathcal{C}$. There are $\binom{n}{i}(q-1)^i$ vectors \mathbf{v} with $d_H(\mathbf{u}, \mathbf{v}) = i$. Such a \mathbf{v} differs from \mathbf{u} in *exactly* i of its n coordinates;

Theorem (Hamming's Sphere-Packing Bound)

Let \mathcal{C} be a q -ary t -error-correcting code of length n , with M code-words. Then

$$M \left(1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \dots + \binom{n}{t}(q-1)^t \right) \leq q^n.$$

Terms after M : First 1 is \mathbf{u} . $\binom{n}{1}(q-1)$ are all \mathbf{v} distance 1 from \mathbf{u} and so on.

Proof.

There are M spheres; one for each $\mathbf{u} \in \mathcal{C}$. There are $\binom{n}{i}(q-1)^i$ vectors \mathbf{v} with $d_H(\mathbf{u}, \mathbf{v}) = i$. Such a \mathbf{v} differs from \mathbf{u} in *exactly* i of its n coordinates; these i coordinates can be selected in $\binom{n}{i}$ different ways.

Theorem (Hamming's Sphere-Packing Bound)

Let \mathcal{C} be a q -ary t -error-correcting code of length n , with M code-words. Then

$$M \left(1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \dots + \binom{n}{t}(q-1)^t \right) \leq q^n.$$

Terms after M : First 1 is \mathbf{u} . $\binom{n}{1}(q-1)$ are all \mathbf{v} distance 1 from \mathbf{u} and so on.

Proof.

There are M spheres; one for each $\mathbf{u} \in \mathcal{C}$. There are $\binom{n}{i}(q-1)^i$ vectors \mathbf{v} with $d_H(\mathbf{u}, \mathbf{v}) = i$. Such a \mathbf{v} differs from \mathbf{u} in *exactly* i of its n coordinates; these i coordinates can be selected in $\binom{n}{i}$ different ways. In each coordinate j , there are $q-1$ choices of values different from that in j .

Theorem (Hamming's Sphere-Packing Bound)

Let \mathcal{C} be a q -ary t -error-correcting code of length n , with M code-words. Then

$$M \left(1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \cdots + \binom{n}{t}(q-1)^t \right) \leq q^n.$$

Terms after M : First 1 is \mathbf{u} . $\binom{n}{1}(q-1)$ are all \mathbf{v} distance 1 from \mathbf{u} and so on.

Proof.

There are M spheres; one for each $\mathbf{u} \in \mathcal{C}$. There are $\binom{n}{i}(q-1)^i$ vectors \mathbf{v} with $d_H(\mathbf{u}, \mathbf{v}) = i$. Such a \mathbf{v} differs from \mathbf{u} in *exactly* i of its n coordinates; these i coordinates can be selected in $\binom{n}{i}$ different ways. In each coordinate j , there are $q-1$ choices of values different from that in j . Summing this gives

$$|S_t(\mathbf{u})| = 1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \cdots + \binom{n}{t}(q-1)^t.$$

Lecture 9:
Error-Correcting
Codes

Prerequisites

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_q

Polynomials

Finite Field \mathbb{F}_q

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gibet-Va sha v
B u d

Miniature

Distance

Sphere-Packing

Theorem

Exercise

Theorem (Hamming's Sphere-Packing Bound)

Let \mathcal{C} be a q -ary t -error-correcting code of length n , with M code-words. Then

$$M \left(1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \cdots + \binom{n}{t}(q-1)^t \right) \leq q^n.$$

Terms after M : First 1 is \mathbf{u} . $\binom{n}{1}(q-1)$ are all \mathbf{v} distance 1 from \mathbf{u} and so on.

Proof.

There are M spheres; one for each $\mathbf{u} \in \mathcal{C}$. There are $\binom{n}{i}(q-1)^i$ vectors \mathbf{v} with $d_H(\mathbf{u}, \mathbf{v}) = i$. Such a \mathbf{v} differs from \mathbf{u} in *exactly* i of its n coordinates; these i coordinates can be selected in $\binom{n}{i}$ different ways. In each coordinate j , there are $q-1$ choices of values different from that in j . Summing this gives

$$|S_t(\mathbf{u})| = 1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \cdots + \binom{n}{t}(q-1)^t.$$

These M spheres are disjoint since $2t < d$.

Peiliaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_q

Polynomials

Finite Field \mathbb{F}_q

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gibet-Va sha v

B u d

Mini u

Distance

S he e-Packi g

The e

E i gue

Theorem (Hamming's Sphere-Packing Bound)

Lecture 9:
 Error-Correcting
 Codes

Prerequisites

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gibert-Vašara v
 Bud

Minimum

Distance

Sphere-Packing

Theorem

Exercise

Theorem (Hamming's Sphere-Packing Bound)

Let \mathcal{C} be a q -ary t -error-correcting code of length n , with M code-words. Then

$$M \left(1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \cdots + \binom{n}{t}(q-1)^t \right) \leq q^n.$$

Terms after M : First 1 is \mathbf{u} . $\binom{n}{1}(q-1)$ are all \mathbf{v} distance 1 from \mathbf{u} and so on.

Proof.

There are M spheres; one for each $\mathbf{u} \in \mathcal{C}$. There are $\binom{n}{i}(q-1)^i$ vectors \mathbf{v} with $d_H(\mathbf{u}, \mathbf{v}) = i$. Such a \mathbf{v} differs from \mathbf{u} in *exactly* i of its n coordinates; these i coordinates can be selected in $\binom{n}{i}$ different ways. In each coordinate j , there are $q-1$ choices of values different from that in j . Summing this gives

$$|S_t(\mathbf{u})| = 1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \cdots + \binom{n}{t}(q-1)^t.$$

These M spheres are disjoint since $2t < d$. All these are in \mathbb{F}_q^n with $|\mathbb{F}_q^n| = q^n$. So $M|S_t(\mathbf{u})| \leq q^n$. □

Outline

1 Preliminaries

- Abstract Algebra
 - Field \mathbb{F}
 - Finite Field \mathbb{F}_p
 - Polynomials
 - Finite Field \mathbb{F}_{p^m}
- Linear Algebra
 - Vector Space
 - Terminology

2 Linear Block Codes

- Definition
- Repetition Code
- Parity Check Code
- Hamming Code

3 Gilbert-Varshamov Bound

- Minimum Distance
- Sphere-Packing
- Theorem

4 Epilogue

Lecture 9: Error-Correcting Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_{p^m}

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gilbert-Vaishya v

Bud

Minimum

Distance

Sphere-Packing

Theorem

Epilogue

Gilbert-Varshamov Bound

How about a *lower bound* to how well we can pack a *given box* full of *cubes*?

Lecture 9:

Error-Correcting Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gilbert-Varshamov

Bound

Minimum

Distance

Sphere-Packing

Theorem

Example

Gilbert-Varshamov Bound

How about a *lower bound* to how well we can pack a *given box* full of *cubes*?

Theorem

If $q \geq 2$ and $n \geq d \geq 1$, then

$$A_q(n, d) \left(1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \cdots + \binom{n}{d-1}(q-1)^{d-1} \right) \geq q^n.$$

Lecture 9:

Encoding Codes

Prerequisites

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Codes

Hamming Codes

Gilbert-Varshamov

Bound

Minimum

Distance

Sphere-Packing

Theorem

Example

Gilbert-Varshamov Bound

How about a *lower bound* to how well we can pack a *given box* full of *cubes*?

Theorem

If $q \geq 2$ and $n \geq d \geq 1$, then

$$A_q(n, d) \left(1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \cdots + \binom{n}{d-1}(q-1)^{d-1} \right) \geq q^n.$$

Proof.

Among \mathcal{C} with fixed q, n, d , let \mathcal{C} have the maximum number of code words.

Lecture 9:
Error-Correcting
Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gilbert-Varshamov
Bound

Minimum

Distance

Sphere-Packing

Theorem

Example

Gilbert-Varshamov Bound

How about a *lower bound* to how well we can pack a *given box* full of *cubes*?

Theorem

If $q \geq 2$ and $n \geq d \geq 1$, then

$$A_q(n, d) \left(1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \cdots + \binom{n}{d-1}(q-1)^{d-1} \right) \geq q^n.$$

Proof.

Among \mathcal{C} with fixed q, n, d , let \mathcal{C} have the maximum number of code words. That is, $M = |\mathcal{C}| = A_q(n, d)$.

Lecture 9:

Error-Correcting Codes

Prerequisites

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gilbert-Varshamov Bound

Minimum

Distance

Sphere-Packing

Theorem

Example

Gilbert-Varshamov Bound

How about a *lower bound* to how well we can pack a *given box* full of *cubes*?

Theorem

If $q \geq 2$ and $n \geq d \geq 1$, then

$$A_q(n, d) \left(1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \cdots + \binom{n}{d-1}(q-1)^{d-1} \right) \geq q^n.$$

Proof.

Among \mathcal{C} with fixed q, n, d , let \mathcal{C} have the maximum number of code words. That is, $M = |\mathcal{C}| = A_q(n, d)$. The spheres

$$S_{d-1}(\mathbf{u}) = \{\mathbf{v} \in V \mid d_H(\mathbf{u}, \mathbf{v}) \leq d-1\}$$

must cover \mathbb{F}_q^n (with lots of overlap):

Lecture 9:

Error-Correcting Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_p

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gilbert-Varshamov Bound

Minimum

Distance

Sphere-Packing

Theorem

Example

Gilbert-Varshamov Bound

How about a *lower bound* to how well we can pack a *given box* full of *cubes*?

Theorem

If $q \geq 2$ and $n \geq d \geq 1$, then

$$A_q(n, d) \left(1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \cdots + \binom{n}{d-1}(q-1)^{d-1} \right) \geq q^n.$$

Proof.

Among \mathcal{C} with fixed q, n, d , let \mathcal{C} have the maximum number of code words. That is, $M = |\mathcal{C}| = A_q(n, d)$. The spheres

$$S_{d-1}(\mathbf{u}) = \{\mathbf{v} \in V \mid d_H(\mathbf{u}, \mathbf{v}) \leq d-1\}$$

must cover \mathbb{F}_q^n (with lots of overlap): Assume $\mathbf{v} \in \mathbb{F}_q^n$ is not in any $S_{d-1}(\mathbf{u})$.

Lecture 9:

Encoding Codes

Prerequisites

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_q

Polynomials

Finite Field \mathbb{F}_q

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gilbert-Varshamov Bound

Minimum

Distance

Sphere-Packing

Theorem

Example

Gilbert-Varshamov Bound

How about a *lower bound* to how well we can pack a *given box* full of *cubes*?

Theorem

If $q \geq 2$ and $n \geq d \geq 1$, then

$$A_q(n, d) \left(1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \cdots + \binom{n}{d-1}(q-1)^{d-1} \right) \geq q^n.$$

Proof.

Among \mathcal{C} with fixed q, n, d , let \mathcal{C} have the maximum number of code words. That is, $M = |\mathcal{C}| = A_q(n, d)$. The spheres

$$S_{d-1}(\mathbf{u}) = \{\mathbf{v} \in V \mid d_H(\mathbf{u}, \mathbf{v}) \leq d-1\}$$

must cover \mathbb{F}_q^n (with lots of overlap): Assume $\mathbf{v} \in \mathbb{F}_q^n$ is not in any $S_{d-1}(\mathbf{u})$. Then for each \mathbf{u} , $d_H(\mathbf{u}, \mathbf{v}) \geq d$.

Lecture 9:

Error-Correcting Codes

Prerequisites

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_q

Polynomials

Finite Field \mathbb{F}_q

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gilbert-Varshamov Bound

Minimum

Distance

Sphere-Packing

Theorem

Example

Gilbert-Varshamov Bound

How about a *lower bound* to how well we can pack a *given box* full of *cubes*?

Theorem

If $q \geq 2$ and $n \geq d \geq 1$, then

$$A_q(n, d) \left(1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \cdots + \binom{n}{d-1}(q-1)^{d-1} \right) \geq q^n.$$

Proof.

Among \mathcal{C} with fixed q, n, d , let \mathcal{C} have the maximum number of code words. That is, $M = |\mathcal{C}| = A_q(n, d)$. The spheres

$$S_{d-1}(\mathbf{u}) = \{\mathbf{v} \in V \mid d_H(\mathbf{u}, \mathbf{v}) \leq d-1\}$$

must cover \mathbb{F}_q^n (with lots of overlap): Assume $\mathbf{v} \in \mathbb{F}_q^n$ is not in any $S_{d-1}(\mathbf{u})$. Then for each \mathbf{u} , $d_H(\mathbf{u}, \mathbf{v}) \geq d$. So we could add \mathbf{v} into \mathcal{C} , without changing q, n, d , and obtain a greater M .

Lecture 9:

Error-Correcting Codes

Prerequisites

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_q

Polynomials

Finite Field \mathbb{F}_q

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gilbert-Varshamov Bound

Minimum

Distance

Sphere-Packing

Theorem

Example

Gilbert-Varshamov Bound

How about a *lower bound* to how well we can pack a *given box* full of *cubes*?

Theorem

If $q \geq 2$ and $n \geq d \geq 1$, then

$$A_q(n, d) \left(1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \cdots + \binom{n}{d-1}(q-1)^{d-1} \right) \geq q^n.$$

Proof.

Among \mathcal{C} with fixed q, n, d , let \mathcal{C} have the maximum number of code words. That is, $M = |\mathcal{C}| = A_q(n, d)$. The spheres

$$S_{d-1}(\mathbf{u}) = \{\mathbf{v} \in V \mid d_H(\mathbf{u}, \mathbf{v}) \leq d-1\}$$

must cover \mathbb{F}_q^n (with lots of overlap): Assume $\mathbf{v} \in \mathbb{F}_q^n$ is not in any $S_{d-1}(\mathbf{u})$. Then for each \mathbf{u} , $d_H(\mathbf{u}, \mathbf{v}) \geq d$. So we could add \mathbf{v} into \mathcal{C} , without changing q, n, d , and obtain a greater M . This contradicts choice of \mathcal{C} . □

Lecture 9:

Encoding Codes

Prerequisites

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_q

Polynomials

Finite Field \mathbb{F}_q

Linear Algebra

Vectorspace

Terminology

Linear Block Codes

Definition

Redundancy Code

Parity Check Code

Hamming Code

Gilbert-Varshamov Bound

Minimum

Distance

Sphere-Packing

Theorem

Example

Outline

1 Preliminaries

- Abstract Algebra
 - Field \mathbb{F}
 - Finite Field \mathbb{F}_p
 - Polynomials
 - Finite Field \mathbb{F}_{p^m}
- Linear Algebra
 - Vector Space
 - Terminology

2 Linear Block Codes

- Definition
- Repetition Code
- Parity Check Code
- Hamming Code

3 Gilbert-Varshamov Bound

- Minimum Distance
- Sphere-Packing
- Theorem

4 Epilogue

Lecture 9: Error-Correcting Codes

Preliminaries

Abstract Algebra

Field \mathbb{F}

Finite Field \mathbb{F}_p

Polynomials

Finite Field \mathbb{F}_{p^m}

Linear Algebra

Vector Space

Terminology

Linear Block Codes

Definition

Repetition Code

Parity Check Code

Hamming Code

Gilbert-Varshamov

Bound

Minimum

Distance

Sphere-Packing

Theorem

Epilogue

The Punchline

We obtain⁴ a sort of *special case* of **Shannon's Fundamental Theorem**. Since **binary** codes have $R = \frac{1}{n} \log_2 M$, then *if* $d \leq \lfloor \frac{n}{2} \rfloor$, *then* there exists a code with length n that has the wooping *rate*

$$1 - H_2\left(\frac{d-1}{n}\right) \leq R \leq 1 - H_2\left(\frac{t}{n}\right).$$

Note: $t = \lfloor \frac{d-1}{2} \rfloor$. We do **not** know how to construct them, but at least we have a framework within which to try.

Next lecture: More codes, and *meta-results* about the *class of linear codes*, and *syndrome decoding*. Then you have a basis to work your workshop paper on.

⁴See the book for details