Lineare Codes

Codes

Ein Code ist eine eindeutige Zuordnung von Zeichen aus einem gegebenen Zeichenvorrat zu einem Codewort aus einem anderen Zeichenvorrat.

Codes

Ein Code ist eine eindeutige Zuordnung von Zeichen aus einem gegebenen Zeichenvorrat zu einem Codewort aus einem anderen Zeichenvorrat.

- Matrikelnummern codieren Studenten.

Codes

Ein Code ist eine eindeutige Zuordnung von Zeichen aus einem gegebenen Zeichenvorrat zu einem Codewort aus einem anderen Zeichenvorrat.

- Matrikelnummern codieren Studenten.
- ASCII-Zeichen codieren das lateinische Alphabet (und etliche weitere Zeichen).

Codes

Ein Code ist eine eindeutige Zuordnung von Zeichen aus einem gegebenen Zeichenvorrat zu einem Codewort aus einem anderen Zeichenvorrat.

- Matrikelnummern codieren Studenten.
- ASCII-Zeichen codieren das lateinische Alphabet (und etliche weitere Zeichen).
- Bitfolgen codieren im Rechner natürliche Zahlen

Codes

Ein Code ist eine eindeutige Zuordnung von Zeichen aus einem gegebenen Zeichenvorrat zu einem Codewort aus einem anderen Zeichenvorrat.

- Matrikelnummern codieren Studenten.
- ASCII-Zeichen codieren das lateinische Alphabet (und etliche weitere Zeichen).
- Bitfolgen codieren im Rechner natürliche Zahlen und mit etwas Geschick auch einige rationale Zahlen.

Achtung

Code $\neq$ Chiffre

Binäre Codes

Wir konzentrieren uns nun auf die Codierung durch endliche Bitfolgen fester Länge k:

$$\underbrace{0,1,0,1,0,0,1,0}_{k}$$

Binäre Codes

Wir konzentrieren uns nun auf die Codierung durch endliche Bitfolgen fester Länge k:

$$\underbrace{0, 1, 0, 1, 0, 0, 1, 0}_{k}$$

In diesem Fall sprechen wir von einem binären Code.

Binäre Codes

Ein Alphabet $\mathcal{A}$ habe m Zeichen.

- Codierung von $\mathcal{A}$ durch (mindestens)

$$k = \lceil \log_2(m) \rceil$$

  Bits.
- Speicheraufwand: $m \cdot k$ Bits ($m \gg k$).

Binäre Arithmetik

Bitweise Verknüpfungen...

AND	0	1
0	0	0
1	0	1

XOR	0	1
0	0	1
1	1	0

Binäre Arithmetik

Bitweise Verknüpfungen...

AND	0	1
0	0	0
1	0	1

XOR	0	1
0	0	1
1	1	0

...entsprechen der Arithmetik im Körper $\mathbb{F}_2 = \mathbb{Z}/2\mathbb{Z}$:

Binäre Arithmetik

Bitweise Verknüpfungen...

AND	0	1
0	0	0
1	0	1

XOR	0	1
0	0	1
1	1	0

...entsprechen der Arithmetik im Körper $\mathbb{F}_2 = \mathbb{Z}/2\mathbb{Z}$:

$\cdot$	0	1
0	0	0
1	0	1

$+$	0	1
0	0	1
1	1	0

Binäre Codes

Notationswechsel:

Binäre Codes

Notationswechsel:

$$0\,1\,0\,1\,0\,1\,1\,1\,0\,1\,1\,0\,1\,1\,0 \quad \rightsquigarrow \quad c = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} \in \mathbb{F}_2^k$$

Lineare Codes

Ein linearer Code $\mathcal{C}$ ist ein Untervektorraum von $\mathbb{F}_2^n$.

Lineare Codes

Ein linearer Code $\mathcal{C}$ ist ein Untervektorraum von $\mathbb{F}_2^n$.

- $\mathcal{C}$ enthält 2^k Codeworte für $\dim \mathcal{C} = k \leq n$.

Lineare Codes

Ein linearer Code $\mathcal{C}$ ist ein Untervektorraum von $\mathbb{F}_2^n$.

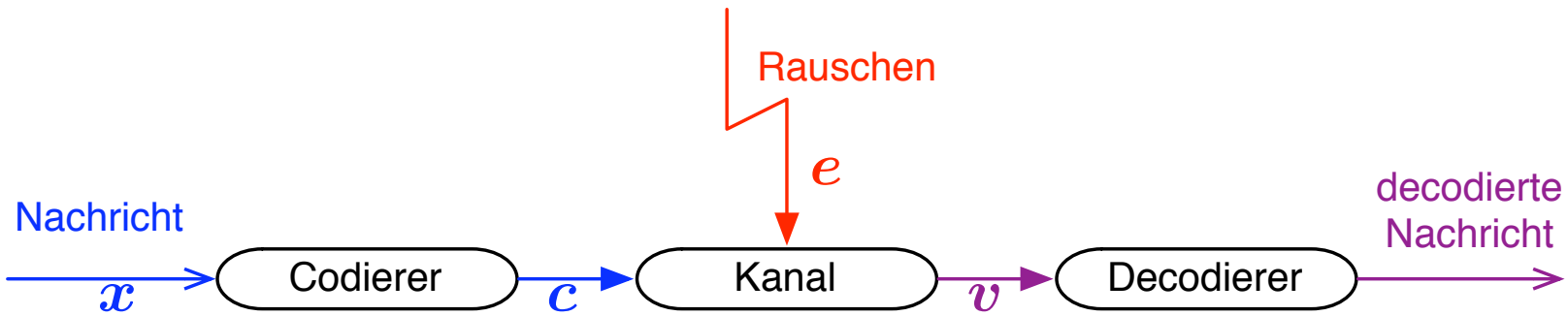
- $\mathcal{C}$ enthält 2^k Codeworte für $\dim \mathcal{C} = k \leq n$.
- Zur Codierung von $\mathcal{A}$ werden nur $n \cdot k$ Bits benötigt:

Lineare Codes

Ein linearer Code $\mathcal{C}$ ist ein Untervektorraum von $\mathbb{F}_2^n$.

- $\mathcal{C}$ enthält 2^k Codeworte für $\dim \mathcal{C} = k \leq n$.
- Zur Codierung von $\mathcal{A}$ werden nur $n \cdot k$ Bits benötigt:
- Die übrigen Codeworte entstehen aus Linearkombinationen der Basis von $\mathcal{C}$.

Nachrichtenübertragung



Effizienz vs. Fehlerkorrektur

Bei der Konstruktion von Codes hat man zwei Ziele im Auge:

Effizienz vs. Fehlerkorrektur

Bei der Konstruktion von Codes hat man zwei Ziele im Auge:

1. Effizienz der Darstellung.
2. Möglichkeiten zur Fehlerkorrektur.

Effizienz vs. Fehlerkorrektur

Bei der Konstruktion von Codes hat man zwei Ziele im Auge:

1. Effizienz der Darstellung.
2. Möglichkeiten zur Fehlerkorrektur.

Dies sind zwei gegensätzliche Ziele:

- Korrekturfähigkeit wird verbessert, wenn man zusätzliche Bits zur Darstellung hinzufügt (also die Redundanz erhöht).
- Dadurch wird jedoch mehr Speicher benötigt.

Fehlerkorrektur

Wir betrachten die Fehlerkorrektur.

Fehlerkorrektur

Wir betrachten die Fehlerkorrektur.

- Hinzufügen redundanter Bits geschieht durch Einbetten von $\mathbb{F}_2^k$ in $\mathbb{F}_2^n$ mit $n > k$.

$$\mathbb{F}_2^k \hookrightarrow \mathcal{C} \subset \mathbb{F}_2^n$$

Fehlerkorrektur

- Der Hamming-Abstand zweier Codeworte $c = (c_1 \ \ldots \ c_n), c' = (c'_1 \ \ldots \ c'_n)$ ist die Anzahl der Stellen, an denen sich die beiden unterscheiden:

$$\mathrm{dist}(c, c') = \#\{j \mid c_j \neq c'_j\}.$$

- Die Minimaldistanz d ist der minimale Abstand zwischen zwei Codeworten aus $\mathcal{C}$,

$$d = \min\{\mathrm{dist}(c, c') \mid c, c' \in \mathcal{C}\}.$$

Fehlerkorrektur

- Der Hamming-Abstand zweier Codeworte $c = (c_1 \ \ldots \ c_n), c' = (c'_1 \ \ldots \ c'_n)$ ist die Anzahl der Stellen, an denen sich die beiden unterscheiden:

$$\mathrm{dist}(c, c') = \#\{j \mid c_j \neq c'_j\}.$$

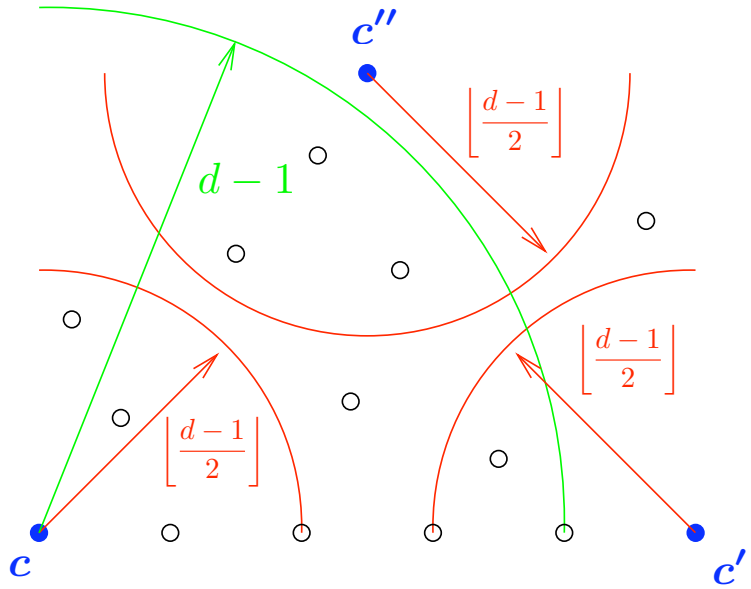
- Die Minimaldistanz d ist der minimale Abstand zwischen zwei Codeworten aus $\mathcal{C}$,

$$d = \min\{\mathrm{dist}(c, c') \mid c, c' \in \mathcal{C}\}.$$

Sie ist entscheidend für die Fehlerkorrekturfähigkeit des Codes.

Fehlerkorrektur

Die Minimaldistanz d bestimmt,
wieviele Fehler erkannt bzw. korrigiert werden können.

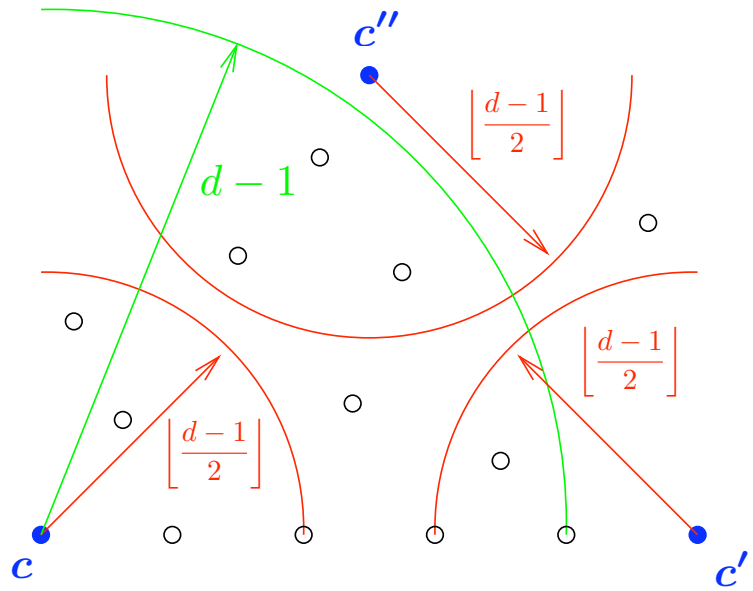


$\mathrm{dist}(\boldsymbol{c}, \boldsymbol{c'}) = 5$

$\circ \notin C$

Fehlerkorrektur

Die Minimaldistanz d bestimmt,
wieviele Fehler erkannt bzw. korrigiert werden können.



$\mathrm{dist}(\boldsymbol{c}, \boldsymbol{c'}) = 5$
$\circ \notin C$

Es können bis zu $d-1$ Fehler erkannt
oder $\lfloor \frac{d-1}{2} \rfloor$ Fehler korrigiert werden
(hängt vom Übertragungskanal ab).

Lineare Codes vs. beliebige Codes

Nutze die Vektorraumstruktur des linearen Codes:

Lineare Codes vs. beliebige Codes

Nutze die Vektorraumstruktur des linearen Codes:

- Ein beliebiger Code $\mathcal{B}$ ist eine Teilmenge von $\mathbb{F}_2^n$.
- Ein linearer Code $\mathcal{C}$ ist ein Untervektorraum von $\mathbb{F}_2^n$.

Lineare Codes vs. beliebige Codes

Nutze die Vektorraumstruktur des linearen Codes:

- Ein beliebiger Code $\mathcal{B}$ ist eine Teilmenge von $\mathbb{F}_2^n$.
- Ein linearer Code $\mathcal{C}$ ist ein Untervektorraum von $\mathbb{F}_2^n$.
- Ein beliebiger Code $\mathcal{B}$ hat keine Struktur, es müssen alle Codeworte separat gespeichert werden.
  $\rightsquigarrow$ linearer Speicheraufwand.

Lineare Codes vs. beliebige Codes

Nutze die Vektorraumstruktur des linearen Codes:

- Ein beliebiger Code $\mathcal{B}$ ist eine Teilmenge von $\mathbb{F}_2^n$.
- Ein linearer Code $\mathcal{C}$ ist ein Untervektorraum von $\mathbb{F}_2^n$.
- Ein beliebiger Code $\mathcal{B}$ hat keine Struktur, es müssen alle Codeworte separat gespeichert werden.
  $\rightsquigarrow$ linearer Speicheraufwand.
- Ist $\mathcal{C}$ ein k-dimensionaler Unterraum, so reichen k Basisvektoren, um alle 2^k Codeworte darzustellen.
  $\rightsquigarrow$ logarithmischer Speicheraufwand.

Codierungsabbildung

Nutze die Vektorraumstruktur des linearen Codes:

Codierungsabbildung

Nutze die Vektorraumstruktur des linearen Codes:

- Wir haben 2^k Informationsworte $x \in \mathbb{F}_2^k$,
  die wir durch Bitfolgen c der Länge $n > k$ codieren wollen.

Codierungsabbildung

Nutze die Vektorraumstruktur des linearen Codes:

- Wir haben 2^k Informationsworte $x \in \mathbb{F}_2^k$,
  die wir durch Bitfolgen c der Länge $n > k$ codieren wollen.
- Codierung erfolgt durch die lineare Codierungsabbildung

$$\Phi : \mathbb{F}_2^k \to \mathbb{F}_2^n, \quad x \mapsto c = G \cdot x.$$

Codierungsabbildung

Nutze die Vektorraumstruktur des linearen Codes:

- Wir haben 2^k Informationsworte $x \in \mathbb{F}_2^k$,
  die wir durch Bitfolgen c der Länge $n > k$ codieren wollen.
- Codierung erfolgt durch die lineare Codierungsabbildung

$$\Phi : \mathbb{F}_2^k \to \mathbb{F}_2^n, \quad x \mapsto c = G \cdot x.$$

Sie ist durch die Erzeugermatrix G definiert,
deren Spalten eine Basis unseres Codes $\mathcal{C}$ sind.

Codierungsabbildung

Nutze die Vektorraumstruktur des linearen Codes:

- Wir haben 2^k Informationsworte $x \in \mathbb{F}_2^k$,
  die wir durch Bitfolgen c der Länge $n > k$ codieren wollen.
- Codierung erfolgt durch die lineare Codierungsabbildung

$$\Phi : \mathbb{F}_2^k \to \mathbb{F}_2^n, \quad x \mapsto c = G \cdot x.$$

  Sie ist durch die Erzeugermatrix G definiert,
  deren Spalten eine Basis unseres Codes $\mathcal{C}$ sind.
- Es ist also

$$\mathcal{C} = \text{Bild } \Phi \subsetneq \mathbb{F}_2^n.$$

Fehlererkennung

Nutze die Vektorraumstruktur des linearen Codes:

Fehlererkennung

Nutze die Vektorraumstruktur des linearen Codes:

- Problem:
  Sender schickt ein Codewort $c \in \mathbb{F}_2^n$,
  Empfänger erhält $v \in \mathbb{F}_2^n$.
  Ein Fehler liegt vor, wenn $v \neq c$ gilt.

Fehlererkennung

Nutze die Vektorraumstruktur des linearen Codes:

- Problem:
  Sender schickt ein Codewort $c \in \mathbb{F}_2^n$,
  Empfänger erhält $v \in \mathbb{F}_2^n$.
  Ein Fehler liegt vor, wenn $v \neq c$ gilt.
- Wie kann man Fehler erkennen?

Fehlererkennung

Nutze die Vektorraumstruktur des linearen Codes:

- Problem:
  Sender schickt ein Codewort $c \in \mathbb{F}_2^n$,
  Empfänger erhält $v \in \mathbb{F}_2^n$.
  Ein Fehler liegt vor, wenn $v \neq c$ gilt.
- Wie kann man Fehler erkennen?
- Prüfe, ob v ein Codewort aus $\mathcal{C}$ ist:
  Löse das LGS

$$v = G \cdot x$$

  nach $x \in \mathbb{F}_2^k$.

Fehlererkennung

- Der Code $\mathcal{C}$ ist als Untervektorraum die Lösungsmenge eines homogenen LGS.

Fehlererkennung

- Der Code $\mathcal{C}$ ist als Untervektorraum die Lösungsmenge eines homogenen LGS.
- Es gibt also eine Prüfmatrix P mit

$$P \cdot G = O.$$

Fehlererkennung

- Der Code $\mathcal{C}$ ist als Untervektorraum die Lösungsmenge eines homogenen LGS.
- Es gibt also eine Prüfmatrix P mit

$$P \cdot G = O.$$

Mit P kann man ein empfangenes v auf Fehler prüfen:

$$v \in \mathcal{C} \quad \Leftrightarrow \quad P \cdot v = 0.$$

Fehlerkorrektur

- Das Fehlersyndrom s von v ist

$$s = P \cdot v \in \mathbb{F}_2^{n-k}.$$

Fehlerkorrektur

- Das Fehlersyndrom s von v ist

$$s = P \cdot v \in \mathbb{F}_2^{n-k}.$$

Es hängt nur von einem additiven Fehler e ab, nicht von c:

$$v = c + e \quad \Rightarrow \quad P \cdot v = \underbrace{P \cdot c}_{=0} + P \cdot e = P \cdot e = s.$$

Fehlerkorrektur

- Das Fehlersyndrom s von v ist

$$s = P \cdot v \in \mathbb{F}_2^{n-k}.$$

Es hängt nur von einem additiven Fehler e ab, nicht von c:

$$v = c + e \quad \Rightarrow \quad P \cdot v = \underbrace{P \cdot c}_{=0} + P \cdot e = P \cdot e = s.$$

- Aus v kann man also das ursprüngliche Codewort c rekonstruieren, wenn man eine Lösung e des LGS

$$P \cdot e = s$$

findet.

Fehlerkorrektur

- Das Fehlersyndrom s von v ist

$$s = P \cdot v \in \mathbb{F}_2^{n-k}.$$

Es hängt nur von einem additiven Fehler e ab, nicht von c:

$$v = c + e \quad \Rightarrow \quad P \cdot v = \underbrace{P \cdot c}_{=0} + P \cdot e = P \cdot e = s.$$

- Aus v kann man also das ursprüngliche Codewort c rekonstruieren, wenn man eine Lösung e des LGS

$$P \cdot e = s$$

findet.

- Zur Fehlerkorrektur finde Vektor e mit möglichst kleinem Hamming-Gewicht. Dieses Problem ist NP-schwer.

Weitere Eigenschaften und Problemstellungen

- Systematische Codierung: Durch geeigneten Basiswechsel S die Erzeugermatrix G auf eine einfachere Form $G \cdot S$ bringen.

Weitere Eigenschaften und Problemstellungen

- Systematische Codierung: Durch geeigneten Basiswechsel S die Erzeugermatrix G auf eine einfachere Form $G \cdot S$ bringen.
- Zyklische Codes: Periodische Struktur der Erzeugermatrix liefert Abschätzungen für Minimaldistanz d.

Weitere Eigenschaften und Problemstellungen

- Systematische Codierung: Durch geeigneten Basiswechsel S die Erzeugermatrix G auf eine einfachere Form $G \cdot S$ bringen.
- Zyklische Codes: Periodische Struktur der Erzeugermatrix liefert Abschätzungen für Minimaldistanz d.
- Bestimmung der Minimaldistanz d bei gegebener Erzeugermatrix G.

Weitere Eigenschaften und Problemstellungen

- Systematische Codierung: Durch geeigneten Basiswechsel S die Erzeugermatrix G auf eine einfachere Form $G \cdot S$ bringen.
- Zyklische Codes: Periodische Struktur der Erzeugermatrix liefert Abschätzungen für Minimaldistanz d.
- Bestimmung der Minimaldistanz d bei gegebener Erzeugermatrix G.
- Bestimmung der maximalen Anzahl der Codeworte bei gegebener Minimaldistanz.

Weitere Eigenschaften und Problemstellungen

- Systematische Codierung: Durch geeigneten Basiswechsel S die Erzeugermatrix G auf eine einfachere Form $G \cdot S$ bringen.
- Zyklische Codes: Periodische Struktur der Erzeugermatrix liefert Abschätzungen für Minimaldistanz d.
- Bestimmung der Minimaldistanz d bei gegebener Erzeugermatrix G.
- Bestimmung der maximalen Anzahl der Codeworte bei gegebener Minimaldistanz.
- Aufzählen aller Codeworte.