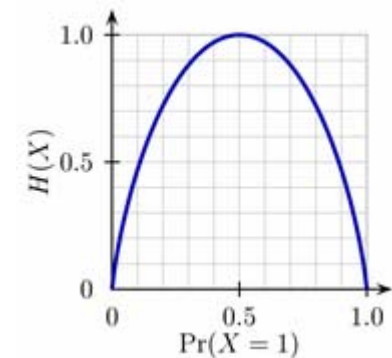


Information und Kommunikation

Hartmut Klauck
Universität Frankfurt
SS 07
21.5.



Schranken für Codes

- Wir wollen nun allgemeine Schranken für die Parameter von Codes angeben
- Intuitiv bedeutet ein großer Code eine kleine Distanz etc.
- Hamming Codes sind $(n, n - \log n, 3)_2$ Codes, Hadamard Codes sind $(n, \log n, n/2)_2$ Codes
- Wir suchen Codes mit guter Distanz und großer Rate
- Ist es notwendig, $\log n$ check-Bits zu verwenden, um 1 Fehler zu korrigieren?

Singleton Bound

- **Theorem 11.1**
 - Für jeden (n,k,d) Code mit Alphabet A gilt:
 - $n \geq k+d-1$
- **Beweis**
 - Wir wollen $d \leq n-(k-1)$ zeigen
 - Wir projizieren alle Codeworte auf die ersten $k-1$ Zeichen (d.h. werfen den Rest weg)
 - Es gibt q^k Codeworte
 - Daher gibt es 2 Codeworte, die auf den ersten $k-1$ Zeichen übereinstimmen
 - Die zwei Codeworte können damit auf höchstens $n-(k-1)$ Zeichen übereinstimmen
 - $d \leq (n-(k-1))$

Singleton Bound

- Codes, die die Schranke erreichen heissen auch MDS Codes („maximum distance separable“)
- Parity check codes sind MDS
 - Distanz 2, $k=n-1$
- Repetition Codes sind MDS
 - n -fache Wiederholung: $k=1$, Distanz n
- Hamming Codes sind nicht MDS

Hamming Schranke

- In der Singleton Schranke spielt q keine Rolle
- Wir betrachten ein einfaches "Volumen"-argument
- Die Hamming "Bälle" mit Radius t um Codeworte x und y dürfen sich nicht schneiden, wenn der Code t Fehler korrigiert

Hamming Schranke

- Binärer Fall
- Sei K die Grösse des Codes, $k = \log_2 K$ die Anzahl der kodierten Bits
- $B_2(x, d, n)$ sei die Menge der strings mit n Bits, die in Hamming Distanz $\leq d$ von x liegen
- $\text{Vol}_2(t, n) = |\{x \in \{0, 1\}^n : |x| \leq t\}|$ sei die Grösse eines Hamming Balls; $|x|$ die Anzahl von Einsen in x
- Dann gilt $K \text{Vol}(t, n) \leq 2^n$ wenn t Fehler korrigiert werden
 - Denn Bälle $B(x, t, n)$, $B(y, t, n)$ müssen disjunkt sein
- Für $t=1$ also: $2^k(n+1) \leq 2^n$, oder $k + \log(n+1) \leq n$
 - D.h. $\log(n+1)$ zusätzliche Bits, um 1 Fehler zu korrigieren

Hamming Schranke

- Hamming Codes treffen die Hamming Schranke, solche Codes heissen *perfekt*
- D.h. Jeder string in $\{0,1\}^n$ für $n=2^l-1$ ist entweder ein Hamming Codewort, oder liegt in Distanz 1 von genau einem Hamming Codewort

Hamming Schranke

- Wir erweitern die Schranke nun in Abhängigkeit von q
- $B_q(x,d,n)$ sei die Menge der Strings der Länge n über einem Alphabet mit q Zeichen, die in Hamming Distanz $\leq d$ von x liegen
- $\text{Vol}_q(d,n) = |B_q(0^n, d, n)| = \sum_{i=0, \dots, d} \binom{n}{i} (q-1)^i$
- **Theorem 11.2**
 - Sei K ein $(n,k,d)_q$ Code.
 - Dann gilt: $q^k \text{Vol}_q(\lfloor (d-1)/2 \rfloor, n) \leq q^n$
- Beweis: Die entsprechenden Bälle um die q^k Codeworte müssen disjunkt sein.
- **Definition 11.3**
 - Wenn Gleichheit in 11.2 gilt, heißt ein Code perfekt.

Hamming Schranke

- Wir können die Hamming Schranke leicht „umkehren“
- Man kann einen Code iterativ konstruieren, indem man ein beliebiges Codewort x in den (anfangs leeren) Code einfügt, dann $B_q(x, d-1, n)$ aus der Menge der noch möglichen Codeworte entfernt, und dies wiederholt solange möglich.
- **Theorem 11.4**
 - Wenn $q^k \text{Vol}_q(d-1, n) \geq q^n$, dann existiert ein $(n, k, d)_q$ Code
- Beweis:
 - Es ist klar das der oben konstruierte Code Distanz d hat. Es werden mindestens $K = q^n / \text{Vol}_q(d-1, n)$ Codeworte gewählt. Für jedes k mit $K \geq q^k$ gibt es also einen $(n, k, d)_q$ Code
- Die Schranke heißt auch Gilbert-Varshamov Schranke

Hamming Schranke

- Obige Konstruktion von Codes ist natürlich nicht praktikabel, da die Codes keine Struktur haben, welche einfache Dekodierung erlaubt

Reed Solomon Codes

- Wir betrachten nun einen Code, der auf Polynomen basiert
- Die wesentliche Eigenschaft von Polynomen hier ist:
- **Fakt 11.5**
 - Ein Polynom $P(x)$ vom Grad k über einem Körper K hat höchstens k Nullstellen.
- Damit folgt unmittelbar, dass zwei Polynome $P(x), Q(x)$ vom Grad k nur auf k Elementen von K übereinstimmen können, da $P-Q$ Grad k hat

Reed Solomon Codes

- Das bedeutet, wir können P rekonstruieren, sobald wir $k+1$ Werte $P(x)$ kennen
- Sei q eine Primzahl, $n \leq q$. Z_q der Körper mit Elementen $\{0, \dots, q-1\}$ (die Konstruktion funktioniert auch mit Primzahlpotenzen q)
- Wir betrachten Nachrichten, die aus k Elementen von Z_q bestehen, c_0, \dots, c_{k-1}
- Diese Nachrichten fassen wir als Polynome auf: $C = \sum_{i=0, \dots, k-1} c_i x^i$
- Der Reed Solomon Code $RS_{q,n,k}$ wird wie folgt konstruiert:
 - Wähle n verschiedene (beliebige) Elemente a_1, \dots, a_n von Z_q
 - Die Kodierungsfunktion bildet c_0, \dots, c_{k-1} auf $C(a_1), \dots, C(a_n)$ ab

Reed Solomon Codes

- Theorem 11.6
 - Reed Solomon Codes sind lineare Codes
- Beweis:
 - Angenommen $C(a_1), \dots, C(a_n)$ und $D(a_1), \dots, D(a_n)$ sind Codeworte, $b \in \mathbb{Z}_q$
 - Wir müssen zeigen, dass $C(a_1)+D(a_1), \dots, C(a_n)+D(a_n)$, sowie $b C(a_1), \dots, b C(a_n)$ Codeworte sind
 - Ersteres ist eine Auswertung von $C+D$
 - Letzteres von bC
 - Beide Polynome haben Grad $k-1$
 - Daher sind dies ebenfalls Codeworte

Reed Solomon Codes

- Offensichtlich haben die Codes Blocklänge n und Nachrichtenlänge k , Alphabetsgrösse q
- Was ist die Distanz?
- **Theorem 11.7**
 - $RS_{q,n,k}$ ist ein $(n,k,n-k+1)$ -Code
- **Beweis**
 - Gemäss der Singleton Schranke gilt $d \leq n-k+1$

Beweis

- Wir müssen also zeigen, dass $d \geq n-k+1$
- Wir betrachten 2 Codeworte
 $C(a_1), \dots, C(a_n); D(a_1), \dots, D(a_n)$
- C und D haben Grad $k-1$ und stimmen daher an höchstens $k-1$ Stellen überein.
- Daher sind $C(a_i) \neq D(a_i)$ für mindestens $n-k+1$ Werte von i

Reed Solomon Codes

- Wir haben als für jede Primzahl(potenz) q einen $(n,k,n-k+1)_q$ Code konstruiert, also einen MDS Code.
 - Wenn $q \geq n$
- Wie nützlich ist das?
 - Das Alphabet ist gross, wächst mit n ($\log n$ Bits Kodierung für Zeichen)
- RS Codes für CD, DSL, WiMAX

Ein Beispiel

- Kodierung von Daten auf CD
- Die Daten sind ein extrem langer Bitstring
- Wird in Blöcke unterteilt (z.B. 240 Bytes).
Jedes Byte als ein Element eines Körpers mit 256 Elementen
- Verwende $n=q=256$ und somit $RS_{256,256,240}$,
Distanz ist also 17
- Speichern als binären Code mit $256*8$ Bits
Block und $240*8$ Nachrichtenlänge
- Distanz des binären Codes ist auch 17

Ein Beispiel

- **Theorem 11.8**
 - Für alle $k \leq n \leq q$ gibt es binäre $(n \log q, k \log q, n-k+1)_2$ Codes
- **Wie gut ist dies?**
 - Wir erhalten im wesentlichen $(K+(1+o(1))d \log K, K, d)_2$ Codes
 - Hamming Schranke sagt, dass die Blocklänge mindestens $K+(1-o(1))d/2 \cdot \log K$ sein muss
 - Also ein Faktor 2 vom Optimum
- **Es gibt "bessere" Codes**
- **Warum verwendet man Reed Solomon?**
 - Burst error: Kratzer auf CD
 - Wenn die ersten 8 Bytes falsch sind, kann immer noch korrigiert werden

Ein Beispiel

- Tatsächlich wird ein Cross-Interleaved Reed Solomon Coding (CIRC) verwendet
- Schwacher innerer RS Code, der 2 Bytes pro 32 Byte Block korrigiert
- Blöcke mit mehr als 2 Bytes werden als Erasures gekennzeichnet
- Dann Korrektur durch einen äusseren RS Code
 - Ein erasure Block wird auf 28 äussere Codeblöcke verteilt, die 4 erasures korrigieren können
- Damit kann der Code bursts von bis zu 4000 Bits korrigieren
 - 2.5 mm auf der CD