Wykład nr 3: 17-01-2005
Temat: Kody korygujące błędy

1 Motywacje

1. Ograniczenia nośnika powodują, że czasami ilość błędów jest zbyt duża, aby można ją było tolerować.
2. Czasami nie możemy sobie pozwolić na retransmisję (komunikacja jest jednostronna, retransmisja na duże odległości bardzo zwalnia przesyłanie informacji).
3. Kody są używane także przy zapisie na nośniku; nośnik może ulegać z czasem degradacji.

2 Zastosowania kodów

1. Dyski kompaktowe, DVD.
2. Komunikacja z sondami kosmicznymi.
3. Transmisja z dużymi szybkościami.

3 Podstawowe definicje i oznaczenia

- Przy omawianiu kodów stosuje się pojęcie *odległości Hamminga* kodów. Niech $\vec{x} = (x_1, \ldots, x_n)$ i $\vec{y} = (y_1, \ldots, y_n)$. Odległość Hamminga tych dwóch wektorów $\Delta(\vec{x}, \vec{y})$ to liczba pozycji, na których $\vec{x}$ różni się od $\vec{y}$.
- Minimalna odległość między słowami kodowymi w zbiorze C jest określona jako

$$\Delta(C) = \min_{\vec{x} \neq \vec{y} \in C} \Delta(\vec{x}, \vec{y}).$$

- Niech Σ będzie alfabetem używanych symboli. Kod systematyczny $(n, k, d)_q$ to podzbiór $C \subseteq \Sigma^n$, gdzie
  - n — długość bloku/pakietu danych,
  - $k = \log_q |C|$ — ilość informacji umieszczona w zakodowanym pakiecie,
  - d — minimalna odległość między słowami kodowymi w zbiorze C,

– $q = |\Sigma|$ — rozmiar alfabetu.

Kody systematyczne mają tę własność, że najpierw występuje k symboli właściwych danych, po czym $n - k$ symboli dodatkowych. Kody systematyczne zostały wprowadzone przez Hamminga [Ham50].

- Podobnie kod jest liniowy $[n, k, d]_q$, gdy C jest podprzestrzenią liniową zbioru $GF(q)^n$ (ciała skończonego o charakterystyce q podniesionego do n-tej potęgi).

4 Przykłady kodów

Wiele przykładów kodów liniowych dostajemy przez włożenie zbioru komunikatów w zbiór kodów.

1. *Identyczność* daje kod $[n, n, 1]_q$.

2. Przekształcenie
$$(x_1, \ldots, x_{n-1}) \longmapsto (x_1, \ldots, x_{n-1}, \Sigma_{i=1}^{n-1} x_i)$$
daje kod $[n, n-1, 2]_q$ — tu można już próbować odzyskiwać stracone dane; *bit parzystości*.

3. Kody *Reeda-Solomona* (będące kodami $[n, k, n-k+1]_q$) otrzymujemy za pomocą następującego przepisu [RS60]:

   - Ustalamy n różnych elementów $GF(q)$: $a_1, \ldots, a_n$.
   - Dla danego komunikatu $m = (m_0, \ldots, m_{k-1})$ tworzymy wielomian
   $$m(x) = \sum_{i=0}^{k-1} m_i x^i,$$
   (który można traktować jak wektor o k współrzędnych).
   - Kodujemy stosując przekształcenie:
   $$(m_0, \ldots, m_{k-1}) \longmapsto (m(a_1), \ldots, m(a_n)).$$
   - Odtwarzanie: przy odtwarzaniu komunikatu m wybieramy wielomian stopnia $k-1$ najbliższy w sensie odległości Hamminga otrzymanemu wielomianowi stopnia $n-1$.
   - W ogólności dla dowolnych kodów ten problem jest NP-zupełny [BMT78]. Ostatnio pokazano również, że jest on NP-zupełny także dla kodów Reeda-Solomona [GV04].

- Istnieją algorytmy obliczające powyższe w czasie $O(k^3)$ (a nawet w $O(k^{2,3\cdots})$ — w granicach znanych ograniczeń na mnożenie macierzy) przy założeniu, że liczba błędów e nie przekracza $(n-k+1)/2$ [WB83].
- Kody Reeda-Solomona oznacza się często przez $RS(n,k)$. Koder bierze tutaj k symboli danych i dodaje do nich symbole korekcyjne tak, aby w sumie powstało słowo kodowe złożone z n symboli.

4. *Kody Reeda-Müllera*

   - Zamiast wielomianów jednej zmiennej, jak w kodach Reeda-Solomona, używamy wielomianów wielu zmiennych.
   - W wyniku tego nie musimy zakładać, że $q \geq n$. Jest tak dlatego, że l różnych zmiennych pozwala na kodowanie l różnych współrzędnych o rozmiarach q każda.
   - Kody te zostały wymyślone przez D.E. Müllera [Mul54], zaś I.S. Reed podał algorytm dla ich dekodowania [Ree54].

5 Przykładowe zastosowanie

CIRC — Cross-Interleaved Reed-Solomon Coding (opracowane na podstawie [Han]).

- Kody Reeda-Solomona korygują do $\frac{n-k}{2}$ błędów.
- Na przykład kod Reeda-Solomona $[255, 223, \bullet]_\bullet$ dla bajtów koryguje do 16×8 błędów bitowych.
- CIRC

  1. Najpierw bierzemy 24 ośmiobitowe słowa.
  2. Kodujemy je kodem $RS(28,24)$ — 4 nadmiarowe bity pozwalają na korekcję 2 bitów.
  3. Rozpraszamy wynik pośród 109 ramek.
  4. Kodujemy wynik kodem Reeda-Solomona $RS(32,28)$ — korekcja 2 bitów.
  5. Rozpraszamy wynik pośród 109 ramek.

- 109 ramek koryguje 436 błędów; przy zastosowaniu rozpraszania można w ten sposób poprawić 13625 ramek.
- $RS(28,24)$ jest kodowaniem poziomu $C2$, koryguje ono błędy nagrania i fizycznego stanu dysku.
- $RS(32,28)$ jest kodowaniem poziomu $C1$, koryguje ono błędy odcisków palców i zarysowań.

6 Podstawowe ograniczenia na własności kodów

- Podstawowe ograniczenie teorioinformacyjne na kody $(n, k, d)_q$ zwane ograniczeniem Hamminga [Ham50]

$$q^k \cdot \mathrm{Vol}_q\Big(\lfloor\frac{d-1}{2}\rfloor, n\Big) \leq q^n,$$

  gdzie

  - q^k — liczba kul w sensie odległości Hamminga (kodów),
  - $\mathrm{Vol}_q(r, s)$ — objętość kuli o promieniu r w przestrzeni s wymiarowej,
  - q^n — rozmiar przestrzeni wynikowych słów kodowych.

- Ograniczenie Singletona (Singleton to nazwisko) [Sin64]. Dla kodów $(n, k, d)_q$ zachodzi

$$d \leq n - k + 1.$$

- Ograniczenie Plotkina [Plo60]

  1. Jeśli kod $(n, k, d)_2$ spełnia $d \geq \frac{n}{2}$, to $k \leq \log_2(2n)$.
  2. Każdy kod $(n, k, d)_2$ spełnia

$$k \leq n - 2d + \log_2(4d).$$

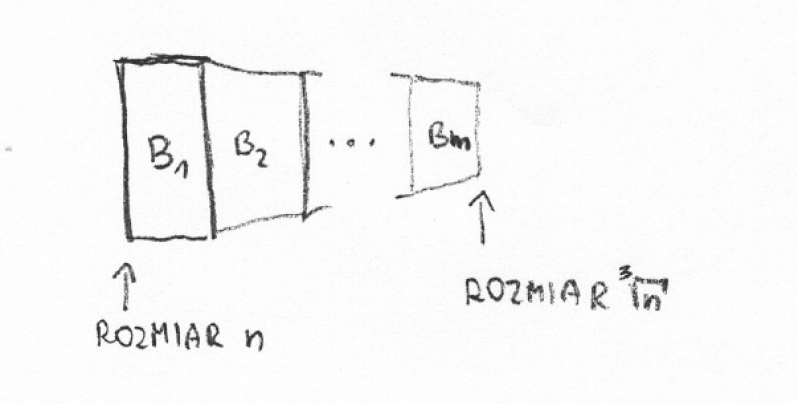
- Ograniczenie Gilberta-Warshamowa[Gil52, Var57]: Dla każdego δ takiego, że $0 < \delta < \frac{1}{2}$ i dla każdego $\epsilon > 0$ istnieje N takie, że dla każdego $n > N$ istnieje kod

$$[n, n(1 - H(\delta) - \epsilon), \delta n]_q$$

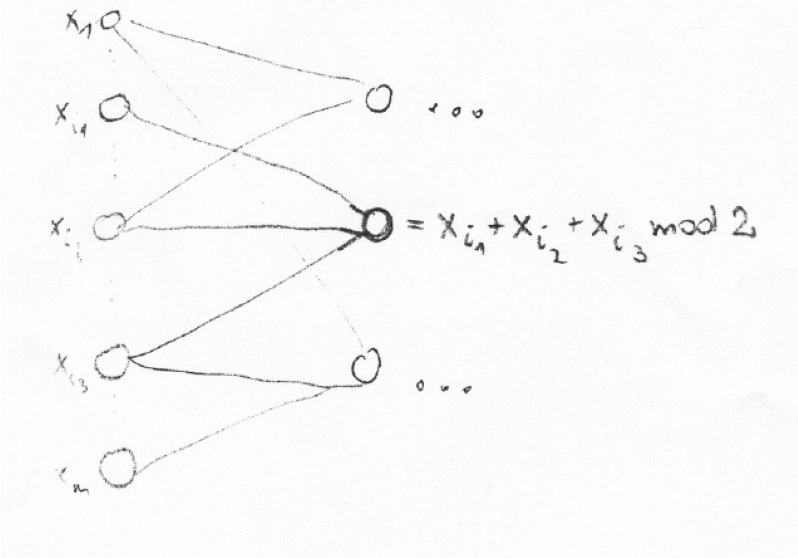
  (**$H(\delta) = -(\delta \log_2 \delta + (1 - \delta) \log_2(1 - \delta))$**)

7 Kody liniowo dekodowalne

Główna idea [SS96] Łączymy ze sobą bloki:



Każdy blok B_i dla $i = 1, \ldots, m$ to graf dwudzielny (L_i, R_i, E_i). Utożsamione są wierzchołki R_i z wierzchołkami L_{i+1}. Podstawowa idea związana z pojedynczym blokiem B_i:



Kodowanie

1. Dla każdego kolejnego bloku B_i spośród bloków $B_1, \ldots, B_m$ wyprowadź z wartości w wierzchołkach L_i wartości w wierzchołkach R_i przez dodawanie $\mathrm{mod}2$.

2. Dla R_m zastosuj tradycyjną technikę, np. kodowanie Reeda-Solomona.

Odkodowywanie Procedura postępuje z prawa na lewo. Dopóki istnieje x_l tze. ma więcej niespełnionych prawych więzów niż spełnionych, to należy odwrócić wartość takiego x_l.

Główny problem Jaką strukturę powinny mieć bloki B_i?

Bloki deterministyczne

Definicja 1 Graf dwudzielny $B = (L, R, E)$ jest (d_L, d_R)*-ograniczony,* jeśli wierzchołki w L mają stopień ograniczony przez d_L, zaś wierzchołki w R mają stopień ograniczony przez d_R.

Definicja 2 Graf j.w. jest (d_L, d_R)*-regularny,* jeśli wierzchołki w L mają stopień dokładnie d_L, zaś wierzchołki w R mają stopień dokładnie d_R.

Definicja 3 Graf dwudzielny $B = (L, R, E)$ jest (α, δ)-ekspanderem, jeśli dla wszystkich $S \subset L$ ($|L| = n$) takich, że $|S| < \delta n$ zachodzi $|\partial(S)| \geq \alpha|S|$, gdzie $\partial(S) = \{r \in R | \exists s \in S, (s, r) \in E\}$.

Twierdzenie 1 Jeśli graf $G = (L, R, E)$ jest (d_L, d_R) ograniczony oraz jest (α, δ)-ekspanderem, to kod uzyskany z G przez konkatenację wartości bitów z L i R mają odległość

$$\frac{2\alpha\delta}{d_L},$$

o ile $2\alpha > d_L$.

Uwaga: (d_L, d_R)-ograniczony ekspander ma $\alpha \leq d_L$.

Twierdzenie 2 Jeśli graf $G = (L, R, E)$ jest (d_L, d_R)-ograniczony oraz jest (α, δ)-ekspanderem i $\alpha > \frac{3}{4}d_L$, to wspomniany kod koryguje do

$$\Big(\frac{2\alpha - d_L}{d_L} \cdot \delta\Big)n,$$

błędów w czasie liniowym.

Złe wieści: nie ma jawnych konstrukcji (d_L, d_R)-ograniczonych ekspanderów dla $\alpha > \frac{3}{4}d_L$.

Dobre wieści: losowe (d_L, d_R) regularne grafy mają z dużym prawdopodobieństwem $\alpha > \frac{3}{4}d_L$.

Bloki losowe Dla grafów losowych można określić wektory

$$(\lambda_1, \ldots, \lambda_{d_L}) \qquad (\rho_1, \ldots, \rho_{d_R}),$$

gdzie

- λ_i — prawdopodobieństwo, że wierzchołek po lewej ma stopień i,
- ρ_i — prawdopodobieństwo, że wierzchołek po prawej ma stopień i.

Optymalne rozkłady:
$\lambda_i \propto \frac{1}{i^2}$
$\rho_i \propto \frac{\alpha^i}{i!}$

Literatura

[BMT78] E. R. Berlekamp, R. J. McEliece, and H.C.A.van Tilborg. On the inherent intractability of certain coding problems. *IEEE Transactions on Information Theory*, 24:384–386, May 1978.

[Gil52] E.N. Gilbert. A comparison of signaling alphabets. *Bell System Technical Journal*, 31:504–522, 1952.

[GV04] Venkatesan Guruswami and Alexander Vardy. Maximum-likelihood decodong of reed-solomon codes is np-hard. *Electronic Colloquium on Computational Complexity*, (40), 2004.

[Ham50] Richard W. Hamming. Error detecting and error correcting codes. *Bell System Technical Journal*, 29:147–160, April 1950.

[Han] Stan Hanley. Reed-solomon codes and cd encoding. http://web.usna.navy.mil/˜wdj/reed-sol.htm. Notes to a course of Professor David Joyner.

[Mul54] D.E. Muller. Application of boolean algebra to switching circuit design and to error detection. *IEEE Transactions on Computers*, 3:6–12, 1954.

[Plo60] M. Plotkin. Binary codes with specified minimum distance. *IRE Transactions on Information Theory*, 6:445–450, 1960.

[Ree54] Irving S. Reed. A class of multiple-error-correcting codes and the decoding scheme. *IEEE Transactions on Information Theory*, 4:38–49, 1954.

[RS60] Irving S. Reed and Gustav Solomon. Polynomial codes over certain finite fields. *J. SIAM*, 8:300–304, 1960.

[Sin64] Richard C. Singleton. Maximum distance q-nary codes. *IEEE Transactions on Information Theory*, 10:116–118, April 1964.

[SS96] M. Spiser and D. Spielman. Expander codes. *IEEE Transactions on Information Theory*, 49(6):1710–1722, 1996.

[Var57] R.R. Varshamov. Estimate of the number of signals in error correcting codes. *Doklady Akademii Nauk SSSR*, 117:739–741, 1957. Russian.

[WB83] L. Welch and E.R. Berlekamp. Error correction for algebraic block codes. U.S. Patent 4 633 470, September 1983.