UNIWERSYTET EKONOMICZNY W POZNANIU

Jak porozumieć się bezbłędnie przez zawodny kanał

Wprowadzenie do teorii informacji

dr inż. Szczepan Górtowski

Po tych zajęciach będziesz umiał

- opisać binarny kanał symetryczny i powiedzieć, co znaczy poziom szumu f
- odróżnić rozwiązanie fizyczne od systemowego — i wskazać, co każde kosztuje
- zdekodować kod powtórzeniowy głosowaniem większościowym i uzasadnić, dlaczego jest optymalne
- zakodować i zdekodować kod Hamminga (7,4) metodą syndromu
- wyliczyć pojemność kanału $C = 1 - H_2(f)$ i powiedzieć, co obiecuje twierdzenie Shannona
- wyjaśnić, dlaczego „mniejszy błąd” nie musi oznaczać „mniejszej szybkości”

Plan zajęć

- ① Problem — kanał, który przekręca bity
  - cztery przykłady, jeden model, jeden wymóg liczbowy
- ② Kody powtórzeniowe — najprostszy pomysł i jego cena
- ③ Kody blokowe — Hamming (7,4) i dekodowanie syndromowe
- ④ Twierdzenie Shannona — gdzie leży granica
- ⑤ Ćwiczenia i co dalej

„Podstawowym problemem komunikacji jest odtworzenie w jednym punkcie — dokładnie albo w przybliżeniu — komunikatu wybranego w innym punkcie.”

— Claude Shannon (1948) [2] — motto rozdziału 1 u MacKaya [1]

01

Problem: kanał, który przekręca bity

Kanały z szumem — cztery przykłady, jeden problem

Kanał	Co przez niego przechodzi	Czym jest szum
analogowa linia telefoniczna	dane cyfrowe między dwoma modemami	przesłuch z innych linii, zniekształcenia sprzętu
łącze radiowe Galileo → Ziemia	telemetria sondy krążącej wokół Jowisza	promieniowanie tła — ziemskie i kosmiczne
dzielące się komórki	DNA komórki macierzystej do komórek potomnych	mutacje i uszkodzenia nici
dysk twardy	bity zapisane jako orientacja namagnesowania	spontaniczne przemagnesowanie, zakłócenia odczytu, interferencja sąsiednich bitów

Dysk pokazuje, że komunikacja nie musi być przesyłaniem w przestrzeni: zapisujemy i czytamy w tym samym miejscu — tylko w innym czasie.

Źródło: [1] MacKay 2003, s. 3 (sekcja 1.1)

Model: binarny kanał symetryczny (BSC)

Wysłano x	Odebrano y	Prawdopodobieństwo	Przy f = 10%
0	0	P(y=0 | x=0) = 1 − f	0,9 — przeszło poprawnie
0	1	P(y=1 | x=0) = f	0,1 — bit przekręcony
1	1	P(y=1 | x=1) = 1 − f	0,9 — przeszło poprawnie
1	0	P(y=0 | x=1) = f	0,1 — bit przekręcony

„Symetryczny”, bo prawdopodobieństwo przekręcenia nie zależy od tego, czy wysłaliśmy zero, czy jedynkę. Cały kanał opisuje jedna liczba: f.

Źródło: [1] MacKay 2003, s. 4, rys. 1.4

Skala wymagania — po co się w to bawić

10%

poziom szumu f w naszym przykładzie: co dziesiąty bit przekręcony

10 lat

tyle ma działać dysk przy 1 GB zapisu i odczytu dziennie

10^{-15}

prawdopodobieństwo błędu bitu, żeby dysk był użyteczny

Dwie drogi do celu

Rozwiązanie fizyczne

Popraw sam kanał, aż przestanie się mylić:

- pewniejsze podzespoły w układach,
- próżnia w obudowie — mniej turbulencji pod głowicą,
- większy obszar magnetyczny na jeden bit,
- mocniejszy sygnał albo chłodzenie (mniej szumu termicznego).

Każdy krok podnosi koszt kanału, a poprawa jest przyrostowa.

Rozwiązanie systemowe

Przyjmij kanał, jaki jest, i dobuduj do niego kodowanie:

- koder dodaje do komunikatu nadmiarowość,
- dekoder wykorzystuje tę znaną nadmiarowość, aby odtworzyć i komunikat, i szum.

Jedyny koszt to obliczenia po obu stronach. Ta droga potrafi zamienić kanał zawodny w kanał niezawodny.

Źródło: [1] MacKay 2003, s. 4–5 (sekcja 1.1)

Podział pracy: co bada teoria informacji, a co teoria kodowania

Teoria informacji

Pyta o granice: jaka jest najlepsza możliwa skuteczność korekcji błędów dla danego kanału?

Odpowiada twierdzeniami o tym, co osiągalne, a co nie.

Teoria kodowania

Pyta o konstrukcje: jak zbudować koder i dekoder, które da się praktycznie policzyć?

Odpowiada kodami — takimi jak Hamming, BCH, LDPC.

Ten wykład

Zaczynamy od dolnego końca: dwa proste kody, policzone do końca.

Kończymy górnym: granicą, której żaden kod nie przeskoczy — i która jest znacznie dalej, niż podpowiada intuicja.

Nadmiarowość nie jest marnowaniem miejsca. Jest walutą, którą płacimy za niezawodność — a twierdzenie Shannona mówi, jaki jest uczciwy kurs.

02

Kody powtórzeniowe: najprostszy pomysł

Kod R3 — powtórz każdy bit trzy razy

Odebrany blok r	Iloraz wiarygodności P(r|s=1) / P(r|s=0)	Decyzja ŝ	Ile głosów
000	γ^{-3}	0	3 × zero
001, 010, 100	γ^{-1}	0	2 × zero, 1 × jeden
011, 101, 110	γ^{1}	1	1 × zero, 2 × jeden
111	γ^{3}	1	3 × jeden

Koder: 0 → 000 i 1 → 111. Dekoder: głosowanie większościowe. Oznaczenie: γ = (1 − f)/f, czyli 9 przy f = 10%.

Źródło: [1] MacKay 2003, s. 5–6, tab. 1.7 i alg. 1.9

Dlaczego głosowanie większościowe jest optymalne

Rachunek

Dekoder optymalny wybiera s najbardziej prawdopodobne po zobaczeniu r. Z twierdzenia Bayesa:

$$P(s \mid r) = P(r \mid s) \cdot P(s) / P(r)$$

Przy założeniu, że 0 i 1 są równie prawdopodobne a priori, maksymalizowanie $P(s|r)$ sprowadza się do maksymalizowania wiarygodności $P(r|s)$.

Kanał działa na każdym bicie niezależnie, więc iloraz wiarygodności jest iloczynem czynników $\gamma = (1-f)/f$ — po jednym na każdy głos.

Wniosek

Skoro $f < ½$, to $\gamma > 1$.

Każdy głos na daną hipotezę mnoży jej wiarygodność przez γ, a każdy głos przeciw — dzieli. Wygrywa hipoteza z większą liczbą głosów.

Głosowanie większościowe nie jest więc zdroworozsądkowym uproszczeniem. Jest dokładnie tym, co wychodzi z rachunku prawdopodobieństwa — pod dwoma jawnymi założeniami: kanał jest BSC, a wiadomości są równie prawdopodobne.

Źródło: [1] MacKay 2003, s. 6–7, wzory (1.18)–(1.23)

R3 — bilans

2,8%

prawdopodobieństwo błędu bitu po dekodowaniu (z 10% przed)

1/3

szybkość kodu: jeden bit informacji na trzy bity kanału

3 ×

tyle dysków (albo tyle rachunków za telefon) trzeba zapłacić

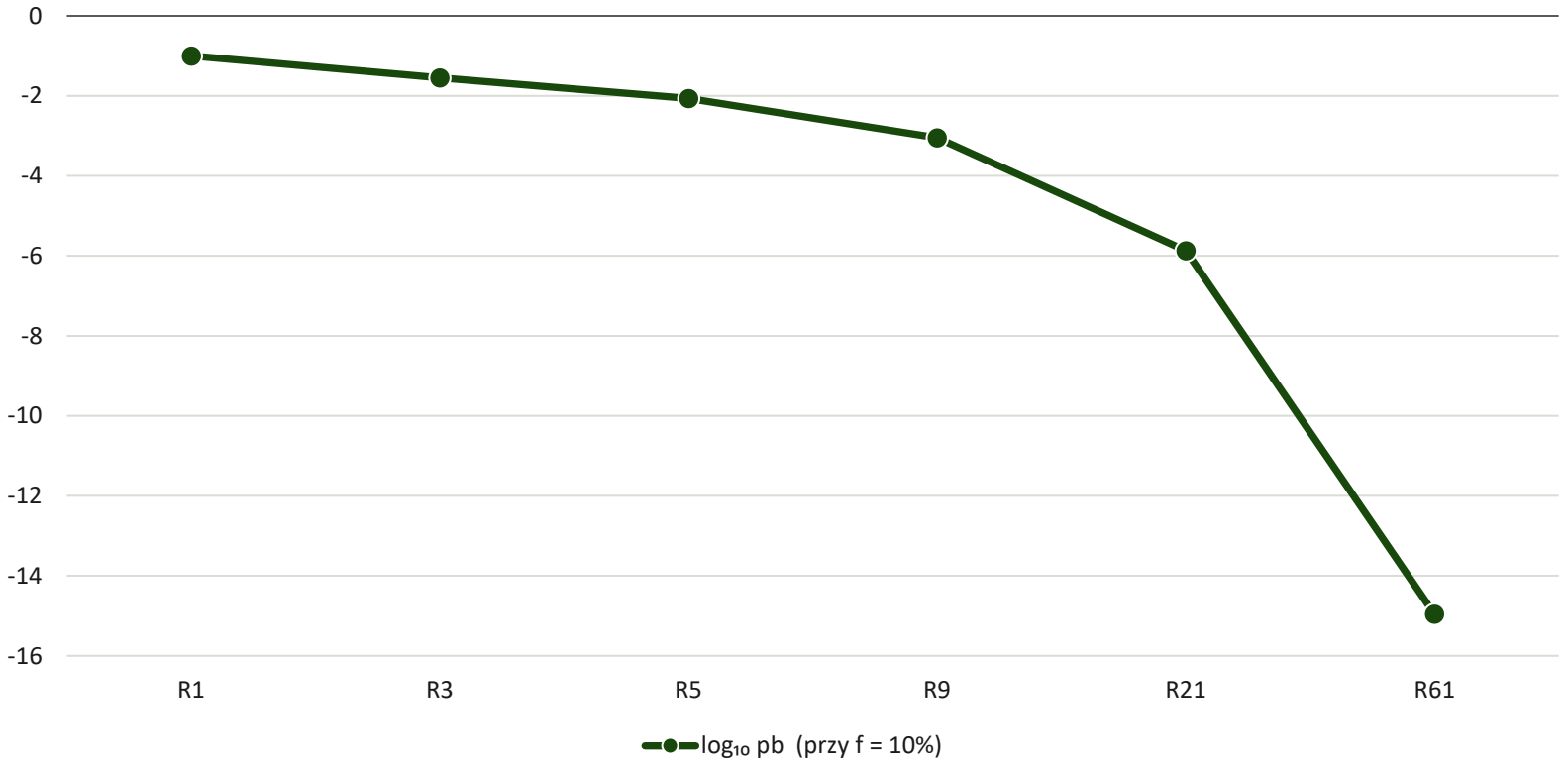
Co dokładnie policzyliśmy dla R3

- Błąd powstaje, gdy szum przekręci dwa albo trzy bity w bloku trzech
  - $p_b = 3f^2(1-f) + f^3 = 3f^2 - 2f^3$ — przy $f = 0{,}1$ daje 0,028, czyli 2,8%
- Dla małych f wyraz wiodący to $3f^2$ — błąd spada kwadratowo z poziomem szumu
- Zysk: błąd zmalał 3,6-krotnie
- Cena: szybkość spadła trzykrotnie — to nie jest transakcja darmowa
  - z trzech zawodnych dysków 1 GB robimy jeden dysk 1 GB o $p_b = 0{,}028$

Źródło: [1] MacKay 2003, zad. 1.2 i rozwiązanie na s. 16; wartości przeliczone niezależnie w kodzie tej prezentacji

Ile powtórzeń trzeba, żeby zejść do 10^{-15}

Oś pionowa to dziesiętny logarytm prawdopodobieństwa błędu bitu. Każdy krok w dół to dziesięciokrotnie mniejszy błąd.



CENA CELU

Kody powtórzeniowe dochodzą do celu — ale skrajnie drogo.

R61 daje pb = $1{,}1 \cdot 10^{-15}$.

Żeby zejść poniżej 10^{-15}, potrzeba 63 powtórzeń: tyle zawodnych dysków na jeden dobry.

Szybkość spada wtedy do 0,016.

Wartości policzone dokładnie ze wzoru pb = Σ C(N,n) $f^n(1-f)$^(N−n) dla n > N/2. MacKay szacuje przybliżeniem Stirlinga „około 60” powtórzeń [1], s. 8 i 16 — rachunek dokładny daje 63.

03

Kody blokowe: Hamming (7,4)

Zmiana pomysłu: nadmiarowość dla całego bloku

- Kod blokowy zamienia K bitów źródła na N bitów kanału, N > K
- W kodzie liniowym te dodatkowe N − K bitów są liniowymi funkcjami bitów źródła — nazywamy je bitami parzystości
- Kod Hamminga (7,4): K = 4 bity źródła → N = 7 bitów kanału, czyli 3 bity parzystości
- Zapis macierzowy: $t = G^T s$ w arytmetyce modulo 2 (1 + 1 = 0)
  - cztery wiersze G to baza czterowymiarowej podprzestrzeni w siedmiowymiarowej przestrzeni binarnej
- Wszystkich słów kodowych jest $2^4 = 16$, a każde dwa różnią się na co najmniej trzech pozycjach

Kodowanie: parzystość w każdym okręgu musi być parzysta

Przykład dla źródła s = 1000. Cztery bity źródła wpisujemy w części wspólne, trzy bity parzystości dopełniamy tak, by w każdym okręgu suma była parzysta.

TRZY KONTROLE PARZYSTOŚCI

- $t_5 = s_1 + s_2 + s_3$

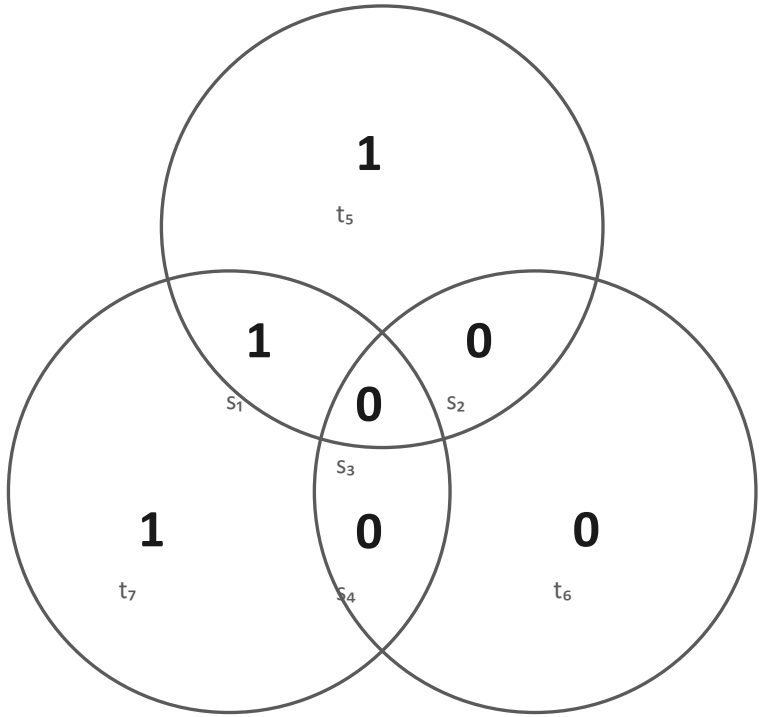
  1+0+0 = 1 → $t_5 = 1$

- $t_6 = s_2 + s_3 + s_4$

  0+0+0 = 0 → $t_6 = 0$

- $t_7 = s_1 + s_3 + s_4$

  1+0+0 = 1 → $t_7 = 1$



WYNIK

s = 1000

t = 1000101

Bity $t_1 \ldots t_4$ to wprost bity źródła.

Bity t_5, t_6, t_7 to trzy kontrole parzystości. Każda pilnuje innej trójki bitów źródła — i to właśnie ta różnica pozwoli później wskazać winowajcę.

Bit s_3 leży w części wspólnej wszystkich trzech okręgów, więc bierze udział w każdej kontroli. Bity parzystości — w dokładnie jednej.

Źródło: [1] MacKay 2003, s. 9, rys. 1.13; [3] Hamming 1950 — diagram wykonany samodzielnie

Dekodowanie: syndrom wskazuje podejrzanego

Wysłano t = 1000101, szum przekręcił drugi bit, odebrano r = 1100101. Sprawdzamy parzystość w każdym okręgu.

TRZY KONTROLE PARZYSTOŚCI

$t_5 = s_1 + s_2 + s_3$

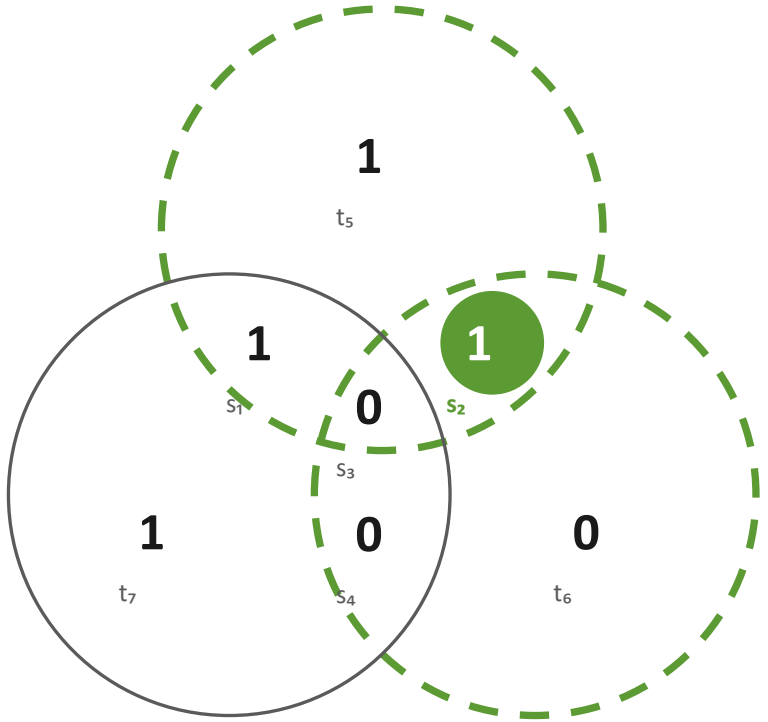
1+1+0+1 = 3 → nieparzyste, kontrola naruszona ($z_1 = 1$)

$t_6 = s_2 + s_3 + s_4$

1+0+0+0 = 1 → nieparzyste, kontrola naruszona ($z_2 = 1$)

$t_7 = s_1 + s_3 + s_4$

1+0+0+1 = 2 → parzyste, kontrola spełniona ($z_3 = 0$)



ROZUMOWANIE

Syndrom z = (1, 1, 0):
dwie pierwsze kontrole naruszone, trzecia w porządku.

Pytanie dekodera: czy istnieje dokładnie jeden bit leżący we wszystkich naruszonych okręgach i poza wszystkimi poprawnymi?

Tak — bit r_2. Żaden inny pojedynczy bit tego nie tłumaczy.

Odwracamy r_2 i odczytujemy $\hat{s}$ = 1000. Błąd naprawiony.

Wzór naruszonych i spełnionych kontroli nazywamy syndromem. Formalnie z = Hr, gdzie H = [P | I_3] — obliczenie syndromu jest operacją liniową.

Źródło: [1] MacKay 2003, s. 10–11, rys. 1.15 i wzór (1.30)

Cały dekoder w jednej tabeli — osiem syndromów

Syndrom z	000	001	010	011	100	101	110	111
Odwróć bit	—	r_7	r_6	r_4	r_5	r_1	r_2	r_3

Trzy kontrole parzystości dają $2^3 = 8$ syndromów: siedem niezerowych — po jednym na każdy jednobitowy wzorzec błędu — i jeden zerowy, gdy szumu nie było.

Każdy odebrany wektor siedmiu bitów jest albo słowem kodowym, albo leży dokładnie jedno odwrócenie od słowa kodowego.

Dlatego dekoder optymalny jest tak tani: nie porównuje r z szesnastoma słowami kodowymi, lecz liczy trzy bity syndromu i sięga do tabeli.

Cena tej prostoty: gdy szum przekręci dwa bity, syndrom wskaże trzeci, niewinny bit — a dekoder go odwróci. Dwa błędy w bloku zamieniają się po dekodowaniu w trzy.

Źródło: [1] MacKay 2003, s. 11–12, alg. 1.16; tabela odtworzona z własnego rachunku syndromów macierzy H

Hamming (7,4) — bilans przy f = 10%

4/7

szybkość kodu — wobec 1/3 dla R3

6,7%

błąd bitu po dekodowaniu (rząd wiodący $9f^2$ dałby 9%)

15%

błąd bloku: dwa lub więcej bitów przekręcone w siódemce

Porównanie: co można kupić za jaką cenę (f = 10%)

Kod	Szybkość R	Błąd bitu pb	Dysków na 1 GB	Co to znaczy
R1 — bez kodowania	1	0,100	1	co dziesiąty bit błędny — dysk bezużyteczny
R3 — powtórz 3×	0,33	0,028	3	błąd 3,6× mniejszy, cena 3× większa
R5 — powtórz 5×	0,20	$8{,}6 \cdot 10^{-3}$	5	im więcej powtórzeń, tym gorszy interes
Hamming (7,4)	0,57	0,067	1,75	ten sam rząd błędu $O(f^2)$ co R3, ale szybkość 4/7 zamiast 1/3
R63 — powtórz 63×	0,016	$3{,}9 \cdot 10^{-16}$	63	cel 10^{-15} osiągnięty — kosztem 63 dysków
granica Shannona	do 0,531	dowolnie małe	2	tyle wystarczy, jeśli kod jest dostatecznie dobry

Kody BCH — rodzina uogólniająca kody Hamminga [4][5] — przesuwają ten kompromis dalej, ale obraz asymptotyczny wciąż wygląda ponuro: aby mocno zmniejszyć błąd, trzeba oddawać szybkość.

Źródło: [1] MacKay 2003, s. 13–15, rys. 1.18–1.19; wszystkie wartości przeliczone niezależnie w kodzie tej prezentacji

04

Twierdzenie Shannona o kanale z szumem

Przekonanie, które okazało się fałszywe

Co sądzono przed 1948

Granica między tym, co osiągalne, a tym, co nie, przechodzi przez początek układu współrzędnych (R, pb) = (0, 0).

Gdyby tak było, dążenie do zerowego błędu wymagałoby dążenia szybkości do zera.

Krótko: „nie ma zysku bez bólu” — dokładnie to, co podpowiadają kody powtórzeniowe.

Co udowodnił Shannon

Granica dotyka osi szybkości w punkcie niezerowym R = C.

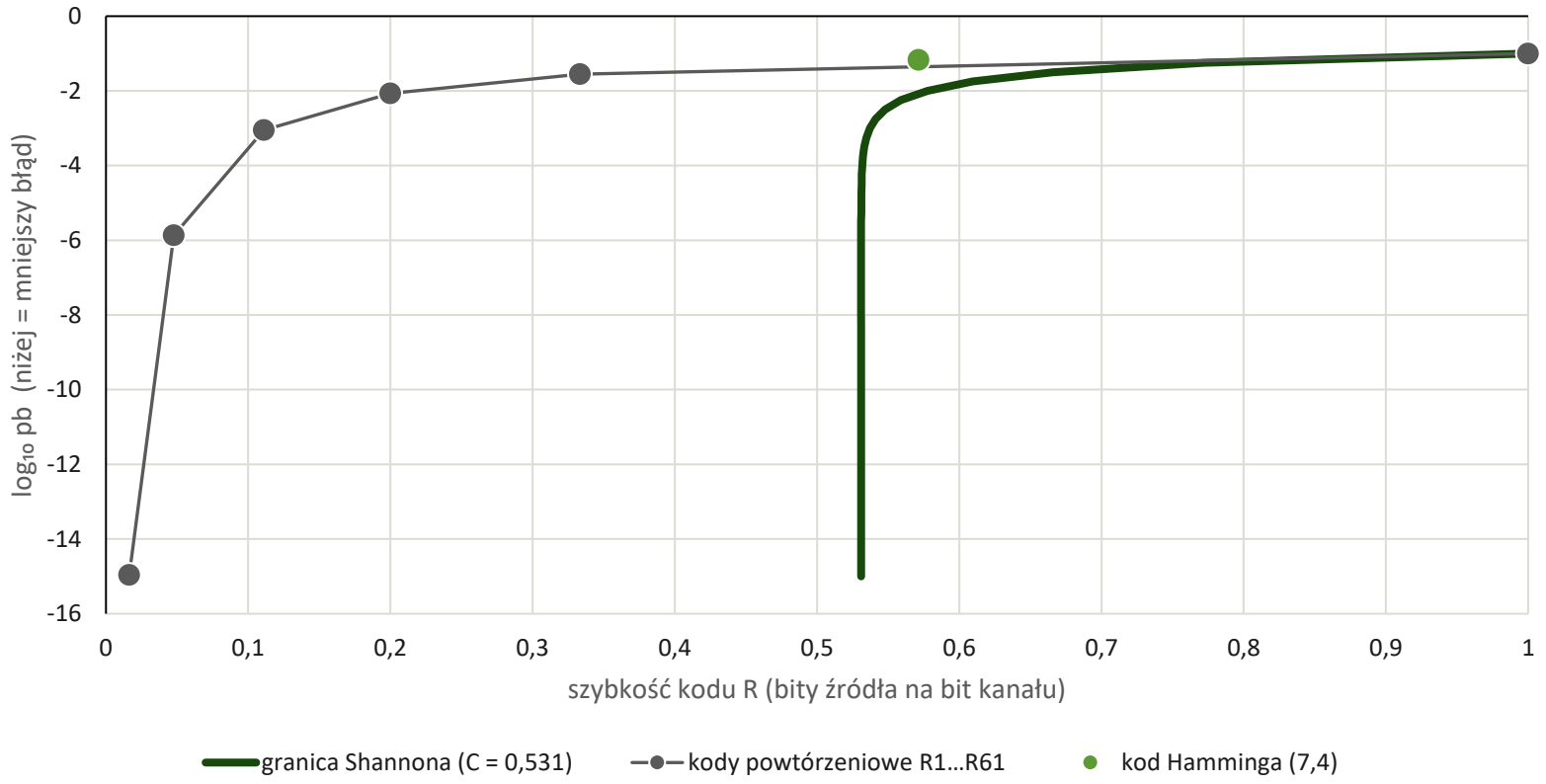
Dla każdego kanału istnieją kody pozwalające komunikować się z dowolnie małym prawdopodobieństwem błędu przy niezerowej szybkości.

To twierdzenie o kanale z szumem. Mówi dwie rzeczy naraz: powyżej pojemności niezawodna komunikacja jest niemożliwa, a poniżej — możliwa.

Źródło: [2] Shannon 1948; [1] MacKay 2003, s. 14–16 (sekcja 1.3) — dowód w rozdziale 10 tej książki

Granica Shannona dla kanału z f = 10%

Krzywa R = C/(1 − H_2(pb)) dzieli płaszczyznę: na lewo od niej wszystko jest osiągalne, na prawo — nie. Punkty pokazują, gdzie leżą kody, które właśnie policzyliśmy.



JAK TO CZYTAĆ

Kody powtórzeniowe schodzą w dół po prawie pionowej ścieżce: kupują mały błąd, oddając szybkość.

Granica jest niemal pionowa dopiero przy R = 0,531.

Cały ten pusty obszar między punktami a krzywą to miejsce na kody, których w 1948 roku nikt nie znał — a które istnieją.

Pojemność kanału — jedna liczba, która zmienia wszystko

$C = 1 - H_2(f)$

wzór na pojemność binarnego kanału symetrycznego

0,531

pojemność naszego kanału z f = 10%, w bitach na użycie kanału

2 dyski

tyle wystarczy zamiast 63, bo 1/2 < 0,53

„Jaką skuteczność chcesz osiągnąć? 10^{-15}? Nie potrzebujesz sześćdziesięciu dysków — wystarczą dwa. A jeśli chcesz 10^{-18} albo 10^{-24}, też dojdziesz tam dwoma.”

Parafraza wniosku z twierdzenia Shannona, nie autentyczny cytat z Shannona

Podsumowanie

- Kod Hamminga (7,4): trzy bity parzystości w bloku siedmiu bitów pozwalają wykryć i naprawić każdy pojedynczy błąd w bloku
- Twierdzenie Shannona: informację można przesyłać przez kanał z szumem z niezerową szybkością i dowolnie małym prawdopodobieństwem błędu
- Granicą jest pojemność C; dla f = 10% wynosi 0,531 bita na użycie kanału
- Twierdzenie jest istnieniowe — mówi, że dobre kody istnieją, nie podaje ich konstrukcji
  - stąd dalsza droga: jak mierzyć zawartość informacji i jak kompresować dane
- Zastrzeżenie samego MacKaya: Shannon dowodził twierdzenia przez ciągi kodów o rosnącej długości bloku — wymagana długość może przekraczać rozmiar naszego dysku

05

Ćwiczenia

Do zrobienia — numeracja zadań jak w książce

- Zad. 1.2 — pokaż, że R3 zmniejsza prawdopodobieństwo błędu; policz je dla dowolnego f
- Zad. 1.3 — wyprowadź pb dla RN, wskaż największy składnik sumy i przybliż go wzorem Stirlinga; ile powtórzeń daje 10^{-15}?
- Zad. 1.5 — zdekoduj: (a) 1101011, (b) 0110110, (c) 0100111, (d) 1111111
- Zad. 1.6 — pokaż, że dla kodu (7,4) $pB \approx 21f^2$, a $pb \approx 9f^2$ w rzędzie wiodącym
- Zad. 1.7 — znajdź wszystkie wektory szumu dające syndrom zerowy; ile ich jest?
- Zad. 1.10 — kod (7,4) naprawia jeden błąd; czy istnieje kod (14,8) naprawiający dwa?

Rozwiązanie zadania 1.5 — do sprawdzenia po zajęciach

Odebrano r	Syndrom z	Odwróć bit	Odczytane ŝ	Komentarz
1101011	011	r4	1100	jeden bit źródła przekręcony — naprawiony
0110110	111	r3	0100	syndrom pełny: podejrzany jest bit środkowy
0100111	001	r7	0100	przekręcony bit parzystości — bity źródła bez zmian
1111111	000	—	1111	to słowo kodowe: syndrom zerowy, dekoder nie rusza nic

Syndromy i wyniki wyliczone w kodzie tej prezentacji z macierzy H, nie przepisane — możesz je odtworzyć na kartce w dwie minuty.

Źródło: [1] MacKay 2003, zad. 1.5, s. 13

Co dalej

- Prawdopodobieństwo, entropia, wnioskowanie: jak w ogóle mierzyć zawartość informacji
- Kompresja danych: druga strona tej samej monety
- Dowód twierdzenia o kanale z szumem
- Kody, które naprawdę dochodzą do granicy Shannona (LDPC, turbo)

Źródła

Poz.	Publikacja	Gdzie w wykładzie
[1]	MacKay, D. J. C. (2003). Information Theory, Inference, and Learning Algorithms. Cambridge University Press. Rozdział 1, s. 3–21 (oraz „About Chapter 1”, s. 1–2). Wersja do czytania na ekranie: www.inference.org.uk/mackay/itila/	podstawa całego wykładu
[2]	Shannon, C. E. (1948). A Mathematical Theory of Communication. Bell System Technical Journal, 27(3), 379–423 i 27(4), 623–656. doi:10.1002/j.1538-7305.1948.tb01338.x	motto · twierdzenie o kanale z szumem
[3]	Hamming, R. W. (1950). Error Detecting and Error Correcting Codes. Bell System Technical Journal, 29(2), 147–160. doi:10.1002/j.1538-7305.1950.tb00463.x	kod (7,4) dekodowanie syndromowe
[4]	Hocquenghem, A. (1959). Codes correcteurs d'erreurs. Chiffres, 2, 147–156.	kody BCH
[5]	Bose, R. C., Ray-Chaudhuri, D. K. (1960). On a class of error correcting binary group codes. Information and Control, 3(1), 68–79. doi:10.1016/S0019-9958(60)90287-4	kody BCH
[6]	Cover, T. M., Thomas, J. A. (2006). Elements of Information Theory (2. wyd.). Wiley. doi:10.1002/047174882X	ujęcie alternatywne (lektura uzupełniająca)
[7]	Stirling, J. (1730). Methodus Differentialis. Londyn. (przybliżenie $x! \approx x^x e^{-x}\sqrt{(2\pi x)}$, używane w zadaniu 1.3 — MacKay, s. 2)	aparat rachunkowy rozdziału

DZIĘKUJĘ

Szczepan Górtowski

Uniwersytet Ekonomiczny w Poznaniu · Katedra Informatyki Ekonomicznej

szczepan.gortowski@ue.poznan.pl