Kanalcodierung

Backward Error Correction (BEC)

Die Redundanz erlaubt lediglich, Fehler zu erkennen und eine Neuübertragung der Daten anzufordern. (Blockcodes, CRC)

Forward Error Correction (FEC)

Die von der Kanalcodierung hinzugefügte Redundanz reicht, um beim Empfänger Fehler zu korrigieren. (Blockcodes, Minimum-Distance-Decoding, Faltungscodes)

Repetitionscode

(linear, zyklisch, systematisch)

z.B. $R^5 = (\underbrace{11111}_{5}), (\underbrace{00000}_{5})$ $N = 5$, $K = 1$, $\partial_{min} = 5$

erkennbare Fehler $= 4$ $(5-1)$, korrigierbare Fehler $= 2 \left(\left\lfloor \frac{5-1}{2} \right\rfloor \right)$

Bitfehlerwahrscheinlichkeit ε (BER - Bit Error Ratio)

Anzahl fehlerhafte Bits im Verhältnis zur Gesamtzahl der Bits.

Alle Bits falsch : BER = 1

Keine Bits falsch : BER = 0

1 von 2 Bits falsch : BER = 0.5

Mit der BER kann die Wahrscheinlichkeit $P_{0,N}$ ausgerechnet werden, mit der eine Sequenz von N Datenbits korrekt (0 Bitfehler) übertragen wird.

Erfolgswahrscheinlichkeit : $P_{0,N} = (1-\varepsilon)^N$

Fehlerwahrscheinlichkeit : $1 - P_{0,N} = 1 - (1-\varepsilon)^N$

Mehr-Bit-Fehlerwahrscheinlichkeit

Die Wahrscheinlichkeit $P_{F,N}$, dass in einer Sequenz von N Datenbits genau F Bitfehler auftreten: $P_{F,N} = \binom{N}{F} \cdot \varepsilon^F \cdot (1-\varepsilon)^{N-F}$

$\binom{N}{F} = \frac{N!}{(N-F)! \cdot F!}$: Anzahl der Möglichkeiten F Fehler in N Bits anzuordnen. N! : Fakultät, z.B. $3! = 3 \cdot 2 \cdot 1$

Taschenrechner Zahl, PRB, !

ε^F : Wahrscheinlichkeit für einen F-fachen Bit-Fehler

$(1-\varepsilon)^{N-F}$: Wahrscheinlichkeit, dass die restlichen Bits (N-F) alle keinen Fehler haben.

Wahrscheinlichkeit, dass maximal F Fehler bei einer Übertragung von N Bits auftreten : $P_{\leq F,N} = \sum_{t=0}^{F} \binom{N}{t} \cdot \varepsilon^t \cdot (1-\varepsilon)^{N-t}$

↳ Oder Wahrscheinlichkeit einer korrekten Übertragung mit F = Anzahl möglicher Fehlerkorrekturen

Weitere Berechnungen: $(P_{\leq F,N})^{\text{Anzahl Codewörter}}$, $(P_{\leq F,N})^{\frac{\text{Anzahl Nutzbits}}{K}}$

Wahrscheinlichkeit, dass mehr als F Fehler bei einer Übertragung von N Bits auftreten : $P_{>F,N} = \sum_{t=F+1}^{N} \binom{N}{t} \cdot \varepsilon^t \cdot (1-\varepsilon)^{N-t}$

Fakultät 0 = 1

Kanalcodierungstheorem

Beschreibt unter welcher Bedingung sich die Wahrscheinlichkeit von Fehlern beliebig reduzieren lässt.

Möchte man die Restfehlerwahrscheinlichkeit eines Fehlerschutzcodes beliebig klein machen so muss $R < C$ sein.

C: Kanalkapazität in bit/bit (Nutzbare Bits pro Kanalbenutzung) $C_{BSC}(\varepsilon) = 1 - H_b(\varepsilon)$

$C_{BSC}(\varepsilon) = 1 - \left\{ \varepsilon \cdot \log_2 \cdot \frac{1}{\varepsilon} + (1-\varepsilon) \cdot \log_2 \cdot \frac{1}{1-\varepsilon} \right\}$

R: Coderate in bit/bit (Infobits pro Codebit) $R = \frac{K}{N}$

R muss kleiner als C sein, damit alle Information in den nutzbaren Bits Platz hat und zuverlässig übertragen werden kann.

Hamming-Distanz

Anzahl wechselnden Bits von einem gültigen Code zum nächsten gültigen Code.

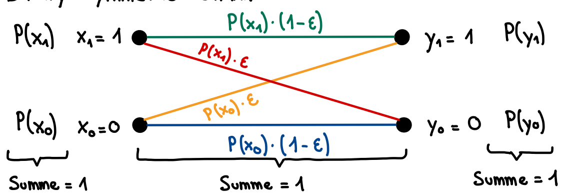
Minimale Hamming-Distanz $\partial_{min}(C)$: Für sichere Fehlererkennung eines Codes (C) relevant. Kleinstes Hamming-Gewicht (w_H) über alle Codewörter hinweg ist ∂_{min}.

Hamming-Gewicht $w_H(c_j)$: XOR-Differenz zweier unterschiedlichen Codewörtern bilden und Anzahl Einsen des erhaltenen Codeworts bestimmen. z.B $\partial_H(110, 011) = w_H(110 \oplus 011) = w_H(101) = 2$

Anzahl erkennbarer Fehler e : $\partial_{min} - 1$ Fehler sind sicher erkennbar

Wahrscheinlichkeit eines BSC (Ein-/Ausgang)

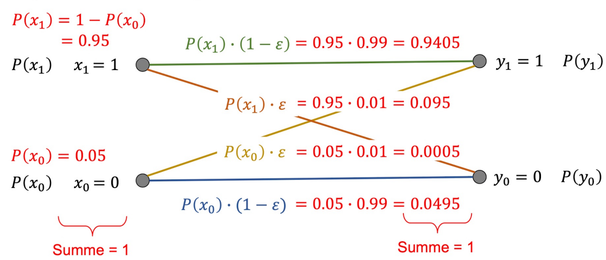
Binary Symmetric Channel



$P(y_1) = P(x_1) \cdot (1-\varepsilon) + P(x_0) \cdot \varepsilon$

$P(y_0) = P(x_0) \cdot (1-\varepsilon) + P(x_1) \cdot \varepsilon$

Beispiel: Quelle mit $P(x_0) = 0.05$ und BSC mit $\varepsilon = 0.01$



$P(y_1) = P(x_1) \cdot (1-\varepsilon) + P(x_0) \cdot \varepsilon = 0.9405 + 0.0005 = 0.9410$

$P(y_0) = P(x_0) \cdot (1-\varepsilon) + P(x_1) \cdot \varepsilon = 0.0495 + 0.0095 = 0.0590$

Summe = 1

Systematischer Code

Die K Informationsbits erscheinen im Codewort an einem Stück.

ⓘ $K = \log_2(\text{Anzahl Codeworte})$

← N Codebits →

N-K Fehlerschutzbits	K Informationsbits

oder

← N Codebits →

K Informationsbits	N-K Fehlerschutzbits

z.B. $\underset{N\ \ K}{(3,2)}$ Block-Code $C = \{[000], [110], [101], [011]\}$

Parity-Bit : 0 XOR 0 = 0
1 XOR 1 = 0
1 XOR 0 = 1
0 XOR 1 = 1

Zyklischer Code

Zyklische Verschiebung eines Codewortes ergibt wieder ein gültiges Codewort.

z.B. $C = (000), (110), (101), (110)$

$(000) \rightarrow (000)$, $(110) \rightarrow (011) \rightarrow (101) \rightarrow (110)$

C_0	C_1	C_2	..	C_{N-1}

ⓘ Null-Codewort separat betrachten, es ist immer zyklisch

Perfekter Code

Wenn jedes Codewort eine d_{min} zu genau einem (nicht mehrere) Codewort aufweist. z.B. $\{[00000], [11111]\}$ ✓

nicht perfekt: $\{[000], [110], [101], [011]\}$ ⊗

Linearer Code

Bitweise XOR-Verknüpfung von 2 beliebigen Codewörtern (inklusive sich selbst) ergibt wieder ein gültiges Codewort.

z.B. $C = (000), (110), (101), (110)$ ⓘ Muss zwingend ein Null-Codewort enthalten

Beliebiges Codewort XOR mit sich selber: $\underline{c}_j \oplus \underline{c}_j = (000)$

Beliebiges Codewort XOR mit (000) : $\underline{c}_j \oplus (000) = \underline{c}_j$

Restliche Fälle : $(110) \oplus (011) = (101)$, $(110) \oplus (101) = (011)$, $(011) \oplus (101) = (110)$ ✓