Kurs Information und Codierung
Informationstheorie

Autoren:
Prof. Dr. Marcel Rupf & Kurt Hauser

5

Kanalcodierung (Teil I)

Fehlererkennung und –korrektur
Kanalcodierungstheorem
Hammingdistanz
Block-Code

Information und Codierung (INCO)

Inhaltsverzeichnis

5

5 Kanalcodierung

5.1 Einführung

Um die Bedeutung der Kanalcodierung zu verdeutlichen, ist nochmals das generische Blockdiagramm aus der Einführung 1 (unten nun Abb. 5-1) abgebildet.

Quellencodierung, Chiffrierung und Kanalcodierung sind wichtige ‚Elemente' in jedem digitalen Kommunikationssystem.

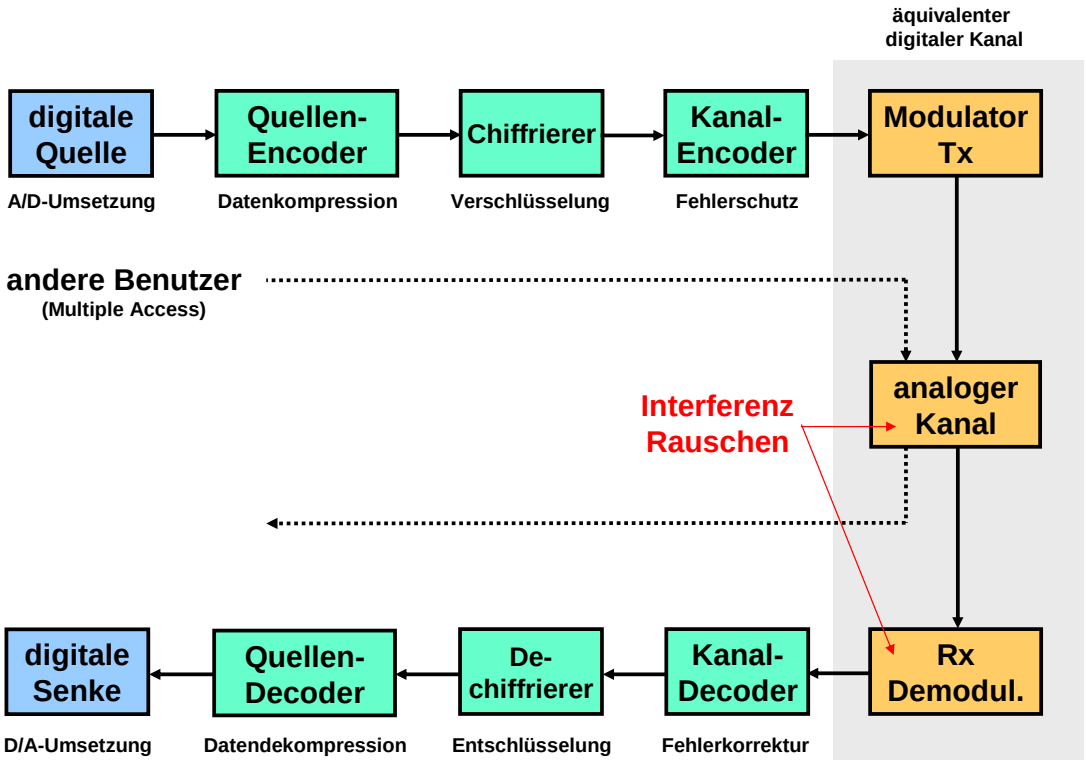

Abb. 5.1: Generisches Blockdiagramm eines digitalen Kommunikationssystems

Bei der Quellencodierung ist die Hauptaufgabe die Datenkompression, mit dem Ziel, Speicherplatz und Bandbreite bei der Speicherung und Übertragung von Daten einzusparen.

Bei der Kanalcodierung ist das Hauptziel, die Daten zuverlässig von der Quelle bis zur Senke zu übertragen. Dies kann oftmals nur durch zusätzliche Fehlerschutzbits, die in den Bitstrom eingefügt werden, erreicht werden.

Die für die nachfolgenden Betrachtungen relevanten Blöcke sind in der Abbildung 5-2 nochmals dargestellt:

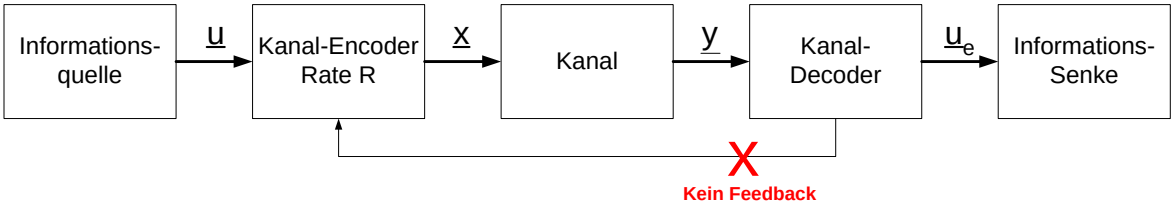

Abb. 5.2: Blockdiagramm mit Fokus auf die Kanalcodierung

Die Tatsache, dass kein Feedback vom Kanaldecoder zurück zum Encoder vorhanden ist, bedeutet, dass der Decoder selbstständig entscheiden können muss, wie die richtige, also die ursprünglich gesendete Bitfolge lautet.

Das Endziel besteht darin, Daten erfolgreich und fehlerfrei von der Quelle zur Senke zu transportieren. Dabei sind zwei grundsätzliche Vorgehensweisen denkbar:

- Erste Variante: Es wird über einen bestimmten Datensatz, zum Beispiel ein Datenrahmen, eine Prüfsumme gebildet. Stimmen die Prüfsummen beim Sender und Empfänger überein, kann von einer fast 100 %-ig fehlerlosen Übertragung ausgegangen werden. Der Sender erhält ein „Acknowledgement“ (ACK). Stimmen die Prüfsummen nicht überein, so bleibt der ACK aus. Nach einer bestimmten Zeit des Ausbleibens des ACK wird der Datensatz nochmals gesendet (Retransmission), bis die Prüfsummen schliesslich übereinstimmen (Beispiel TCP, Transmission Control Protocol).
- Zweite Variante: Es ist eine „Forward Error Correction“ angewandt. In den Sendebitstrom werden zusätzliche Bits eingeschleust, um damit beim Empfänger einfache oder mehrfache Bitfehler korrigieren zu können. Ein Wiedersenden (Retransmission) ist nicht vorgesehen. Der Sendebitstrom mit den eingeschleusten zusätzlichen Bits bietet die Grundlage, damit der Empfänger bis zu einem gewissen Grad Fehler selbst zu korrigieren vermag.

Im Folgenden ist nur noch die zweite Variante, also die „Forward Error Correction“, das weiter zu verfolgende Thema. Diese zweite Variante der Fehlerkorrektur ist sehr schnell, im Gegensatz dazu geht bei der ersten Variante viel Zeit verloren durch das Warten sowie das allfällige „Wiedersenden“ von Datenrahmen.

Es geht bei der „Forward Error Correction“ darum, die zu sendenden Daten so aufzubereiten, dass eine bestimmte Anzahl von Bitfehlern vom Empfänger selbst korrigiert werden können.

5.2 Der Kommunikationskanal

Der Kanal, über den kommuniziert werden muss, kann in einigen Fällen beeinflusst werden:

Im Fall einer Kabelverbindung kann durch geeignete Wahl des Mediums auch die Kanalcharakteristik weitgehend bestimmt werden. Ohne im Detail darauf einzugehen sei daran erinnert, dass zum Beispiel eine Monomodeglasfaser die besseren Übertragungseigenschaften aufweist als eine Multimodeglasfaser. Bei Kupferaderkabeln ist unter anderem die Schirmung ein wesentliches Merkmal, das die Störempfindlichkeit des Gesamtsystems ‚Kabel’ beeinflusst.

Handelt es sich um Freiraumübertragung, so ist der Kommunikationskanal nur noch ganz geringfügig beeinflussbar. So kann etwa eine Richtstrahlverbindung geografisch günstig gelegt werden. In den meisten Fällen ist jedoch der Kanal **nicht** beeinflussbar. Seine nichtidealen Eigenschaften bieten äusserst grosse Herausforderungen.

Die Beziehung zwischen der Ausgangsgrösse y und der Eingangsgrösse x ist stochastisch, weil Rauschen und Mehrwegausbreitung vorhanden sind. Ausserdem beeinflusst der Kanal die Signalform, weil der Kanal eine Filtercharakteristik aufweist.

Weil der Kanal kaum beeinflussbar ist, müssen andere Grössen an den Kanal angepasst werden. Hauptthema in diesem Kapitel ist die Anwendung geeigneter Fehlerschutzmassnahmen in der Codierung der Daten, damit die digitalen Nachrichtensignale in der gewünschten Weise fehlerfrei rekonstruiert werden können.

Definition: Diskreter Kanal (vgl. [1], Seite 628)

Wenn in einem zeitdiskreten Kanal die Werte, welche Eingangs- und Ausgangsvariabeln einnehmen können, endlich oder zählbar unendlich sind, bezeichnet man den Kanal als diskreten Kanal.

5.2.1 Das Kanalcodierungstheorem nach Shannon

Die Kanalkapazität eines diskreten gedächtnisfreien Kanals ist gegeben durch die Beziehung

$$C = \max_{P(x)} [H(Y) - H(Y|X)] \qquad (5.1)$$

Wobei für die Einheit von C gilt: [C] = Bit / Kanalbenützung.

Erläuterungen zur Beziehung (5.1):

Es geht um die Berechnung der „Kanalkapazität“. Sie ist zwischen Apostrophs gesetzt, weil hier die Einheit wie folgt ist: [C] = Bit/Kanalbenützung. (Anm: Die übliche Einheit der Kanalkapazität ist Bit/s; vgl. Formel (1.1) in der Einführung 1).

Zur Berechnung der „Kanalkapazität“ in Bit/Kanalbenützung ist die Entropie am Kanalausgang und die bedingte Entropie am Kanalausgang, wenn der Eingang gegeben ist, verwendet. Die bedingte Entropie H(Y|X) hat mit den Fehlern zu tun, die bei der Übertragung auf dem Kanal entstehen. Falls keine Übertragungsfehler auftreten, so ist die bedingte Entropie H(Y|X) = 0. Als „Kanalkapazität“ ist das erzielbare Maximum der Differenz definiert, das auftritt, wenn die Wahrscheinlichkeitsverteilung der Eingangssymbole variiert. Pro memoria: Falls Nullen genau gleich häufig wie Einsen auftreten, ist H(Y) maximal, weil dann maximale Ungewissheit besteht.

Es gilt: Falls die Übertragungsrate R<C beträgt, existiert ein Code mit ausreichend grosser Blocklänge N, mit dem die Fehlerwahrscheinlichkeit beliebig klein gemacht werden kann.

Umgekehrt gilt: Falls R>C beträgt, so ist die Fehlerwahrscheinlichkeit von jedem Code immer >0, sei N noch so lang.

5.2.2 Kanaleigenschaften

- Rauschen im Kanal beschränkt nicht die Zuverlässigkeit der Übertragung, sondern hauptsächlich die Übertragungsrate.
- Verschiedene Kanäle lassen sich auf diese Weise mit je einer Zahl vergleichen.
- Je grösser die Blocklänge N des verwendeten Block-Codes ist, desto komplexer wird der Dekoder.
- Das Kanalcodierungstheorem gibt Bedingungen für die Existenz von Kanalcodes an. Das Theorem liefert jedoch keine Algorithmen.

5.2.3 Binäre Block-Codes

Ein binärer Block-Code ist ein Code, bei dem der Encoder die Informationssequenz in Blöcke aufteilt. Jeder solche Block weist die Länge von K Informationsbits auf. Ein Block wird als Informationswort u bezeichnet. Total sind theoretisch 2^K verschiedene Informationsworte derselben Länge K möglich.

Der Encoder für den Block-Code wandelt ein Informationswort u in ein Codewort x um. Jedes Codewort x weist die Länge N auf, wobei N>K beträgt. Aufgrund dieser Werte N und K wird der Block-Code als **(N,K) Block-Code** bezeichnet. Weil 2^K Informationsworte in ebenso viele Codeworte gewandelt werden, resultieren 2^K Codeworte. Die Menge der 2^K Codeworte der Länge N, die aus den 2^K Informationsworte gebildet werden, stellt also den (N,K) Block-Code dar.

Das Verhältnis K/N ist die Coderate. Der Encoder eines (N,K) Block-Codes ist gedächtnislos. Der Encoder kann zum Beispiel mittels einer kombinatorischen Logikschaltung realisiert werden.

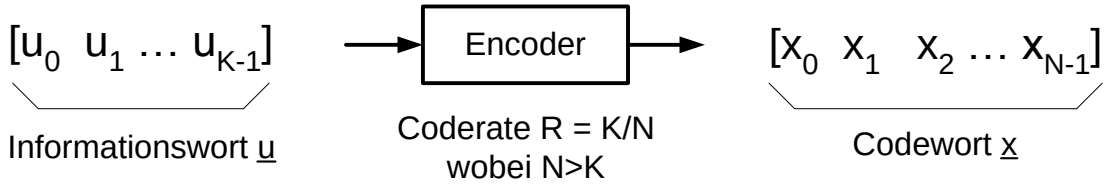

Abb. 5.3: Encoder für den (N,K) Block-Code

Begriff ‚systematischer (N,K) Block-Code C'

Falls die K Informationsbits, also das Informationswort u ‚en bloc' auch im Codewort x erscheint, ist der Block-Code als **systematischer (N,K) Block-Code** bezeichnet. Dadurch gestaltet sich die Rücktransformation, das heisst die Umwandlung des Codeworts x in das Informationswort u, relativ einfach. Die zusätzlichen (N-K) Bits, die den Informationsbits beigefügt werden, sind als ‚Parity Check Bits' bezeichnet. Der Begriff ‚Parity' hat nicht die Bedeutung des Repräsentierens einer geraden oder ungeraden Zahl, sondern der Begriff hat hier die allgemeinere obige Bedeutung.

Darstellung der Informationsbits und Parity-Check Bits:

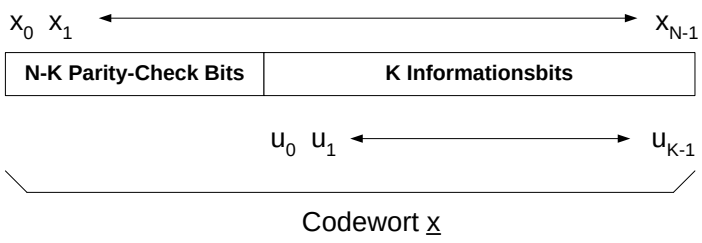

Abb. 5.4: Bildung eines Codeworts

Begriff ‚linearer (N,K) Block-Code C'

Falls die modulo-2 Summe zweier Codewörter wieder ein Codewort ergibt, dann ist der Block Code linear.

Begriff ‚linearer, zyklischer (N,K) Block-Code C'

Falls die zyklische Verschiebung eines Codeworts wieder ein Codewort ergibt, ist der Code ausserdem zyklisch. Aufgrund dieser Eigenschaft sind die verschiedenen Codeworte sehr einfach mit Hilfe eines LFSR (Linear Feedback Shift Register) realisierbar.

Menge der Codeworte des (N,K) Block-Codes C

Die Menge der Codeworte beträgt 2^K.

Allgemein: CW $\underline{x}$ = $[x_0\ x_1\ \dots\ x_{N-1}]$

Erstes CW: CW $\underline{x}_0$ = $[x_{00}\ x_{01}\ \dots\ x_{0\,(N-1)}]$

Zweites CW: CW $\underline{x}_1$ = $[x_{10}\ x_{11}\ \dots\ x_{1\,(N-1)}]$

u.s.w.

Das letzte Codewort heisst CW $\underline{x}_{2K-1}$, also: Index von $\underline{x}$ ist $2^K - 1$, weil es insgesamt 2^K Codeworte gibt und bei null mit Zählen begonnen wird.

Einige Begriffe im Zusammenhang mit der Fehlerkorrektur

Hamming-Gewicht $w_H(\underline{x})$

Das Hamming-Gewicht $w_H(\underline{x})$ entspricht der Anzahl „1" im Codewort $\underline{x}$.

Hamming-Distanz $d_H(\underline{x}_i, \underline{x}_j)$

Die Hamming-Distanz $d_H(\underline{x}_i, \underline{x}_j)$ entspricht der Anzahl unterschiedlicher Positionen in $\underline{x}_i$ und $\underline{x}_j$.

Beispiel**:** Gegeben seien zwei Codeworte $\underline{x}_j$ und $\underline{x}_k$. Die Anzahl Bits, in denen $\underline{x}_j$ und $\underline{x}_k$ verschieden sind, ist die Hammingdistanz. Dazu ein Beispiel:

Sei $\underline{x}_j$ = 0 0 1 0 1 1

Sei $\underline{x}_k$ = 0 1 1 0 1 0

Verschieden in: x x ➔ in **zwei** Bits verschieden.

Die Hammingdistanz dieser zwei Codeworte beträgt **zwei**.

Minimaldistanz d_{min} des linearen (N,K) Block-Codes C

$$d_{min} = \min_{ij} d_H(\underline{x}_i, \underline{x}_j) = \min_{ij} w_H(\underline{x}_i + \underline{x}_j) = \min_k w_H(\underline{x}_k) = w_{min} \quad \text{wobei } (i \neq j) \qquad (5.2)$$

Beispiel eines (3,2) Block-Codes

Der binäre Block-Code C = {[0 0 0], [1 1 0], [1 0 1], [0 1 1]} soll analysiert werden.

Encoder: K=2 Input-Bits. Daraus werden $2^K = 4$ Codewörter der Länge N=3 gebildet.

Coderate R = K/N = 2/3.

Um von einem Codewort zu einem anderen zu gelangen, müssen zwei Bits geändert werden. Damit beträgt die minimale Hamming-Distanz $d_{min} = 2$.

Räumliche Darstellung des (3,2) Block-Codes C

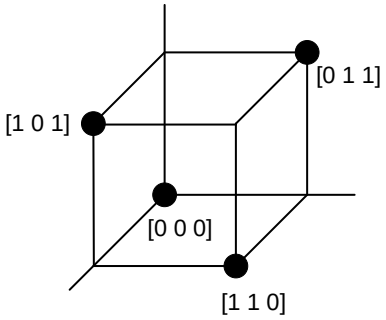

Abb. 5.5: Der (3,2) Block-Code C

Alle Bitmuster mit 1 Fehler sind detektierbar.

$[b_0\ b_1\ b_2]$	
[0 0 0]	Codewort
[0 0 1]	Muster mit 1 Fehler: Entweder b_0 oder b_1 oder b_2 falsch
[0 1 0]	Muster mit 1 Fehler: Entweder b_0 oder b_1 oder b_2 falsch
[0 1 1]	Codewort
[1 0 0]	Muster mit 1 Fehler: Entweder b_0 oder b_1 oder b_2 falsch
[1 0 1]	Codewort
[1 1 0]	Codewort
[1 1 1]	Muster mit 1 Fehler: Entweder b_0 oder b_1 oder b_2 falsch

Es ist aber **nicht möglich**, einen Einbitfehler zu lokalisieren und damit zu korrigieren.

Drei Fehlermuster mit zwei Fehlern sind **nicht** detektierbar.

Beispiel: Vom Sender wurde [0 0 0] gesendet.

Die drei Fehlermuster mit jeweils genau zwei Bitfehlern sind: [0 1 1], [1 1 0], [1 0 1]. Das sind allesamt gültige Codewörter (CW). Der Empfänger kann daher die zwei Fehler nicht erkennen.

Anderes Beispiel: Vom Sender wurde [1 0 1] gesendet.

Die drei Fehlermuster mit jeweils genau zwei Bitfehlern sind: [1 1 0], [0 1 1], [0 0 0].

Diese drei Muster sind jedoch ebenfalls allesamt gültige CWs.

Wie steht es mit der Detektierbarkeit von drei Bitfehlern?

Gültiges gesendetes Codewort	Mit drei Bitfehlern versehen	Andere Interpretation	Schlussfolgerung
[0 0 0]	[1 1 1]	Muster mit 1 Fehler	Entweder 1 oder 3 Bitfehler
[0 1 1]	[1 0 0]	Muster mit 1 Fehler	Entweder 1 oder 3 Bitfehler
[1 0 1]	[0 1 0]	Muster mit 1 Fehler	Entweder 1 oder 3 Bitfehler
[1 1 0]	[0 0 1]	Muster mit 1 Fehler	Entweder 1 oder 3 Bitfehler

Zusammengefasst:

- Alle Muster mit 1 Bitfehler sind detektierbar; auch alle Muster mit 3 Bitfehlern. Es kann aber keine Aussage gemacht werden, ob jeweils *ein* Bitfehler vorliegt oder deren *drei*.
- Drei Fehlermuster mit 2 Bitfehlern sind nicht detektierbar
- **Kein** Bitfehler ist korrigierbar
- **Für die Fehlerkorrektur ist dieser Code nicht einsetzbar**

Es interessiert nun die Frage, ob der Block-Code C systematisch, linear und zyklisch ist.

Ist der Block-Code systematisch?

Weil K=2 Informationsbits vorliegen, sind vier Kombinationen möglich: 00, 01, 10 und 11.

Die Untersuchung ergibt, dass die K=2 Informationsbits bei allen vier Codewörtern ‚en bloc' erscheinen (kursiv gedruckt): C = {[0 *0 0*], [1 *1 0*], [1 *0 1*], [0 *1 1*]}. Der vorliegende (3,2) Block-Code ist also systematisch.

Ist der Block-Code linear?

Falls die modulo-2 Summe zweier CW wieder ein CW ergibt, ist der Block Code linear.

CW i	CW j	Modulo-2 Summe	Codewort?
[0 0 0]	[0 0 0]	[0 0 0]	ja
[0 0 0]	[1 1 0]	[1 1 0]	ja
[0 0 0]	[1 0 1]	[1 0 1]	ja
[0 0 0]	[0 1 1]	[0 1 1]	ja
[1 1 0]	[1 1 0]	[0 0 0]	ja
[1 1 0]	[1 0 1]	[0 1 1]	ja
[1 1 0]	[0 1 1]	[1 0 1]	ja
[1 0 1]	[1 0 1]	[0 0 0]	ja
[1 0 1]	[0 1 1]	[1 1 0]	ja
[0 1 1]	[0 1 1]	[0 0 0]	ja

Der vorliegende (3,2) Block-Code ist also linear. Weitere Frage: Ist der Block-Code zyklisch?

Falls die zyklische Verschiebung eines CW wieder zu einem CW führt, ist der Code zyklisch.

CW i	Zyklisch um ein Bit verschoben	Codewort?
[0 0 0]	[0 0 0]	Ja
[1 1 0]	[0 1 1]	Ja
[1 0 1]	[1 1 0]	Ja
[0 1 1]	[1 0 1]	Ja

Der vorliegende (3,2) Block-Code ist also auch zyklisch.

Nochmals die Begriffe im Zusammenhang mit der Fehlerkorrektur:

Hamming-Gewicht $w_H(\underline{x})$

Das Hamming-Gewicht $w_H(\underline{x})$ entspricht der Anzahl „1" im Codewort $\underline{x}$.

Hamming-Distanz $d_H(\underline{x}_i, \underline{x}_j)$

Die Hamming-Distanz $d_H(\underline{x}_i, \underline{x}_j)$ entspricht der Anzahl unterschiedlicher Positionen in $\underline{x}_i$ und $\underline{x}_j$.

Minimaldistanz d_{min} des linearen (N,K) Block-Codes C

$$d_{min} = \min_{ij} d_H(\underline{x}_i, \underline{x}_j) = \min_{ij} w_H(\underline{x}_i + \underline{x}_j) = \min_k w_H(\underline{x}_k) = w_{min} \quad \text{wobei } (i \neq j) \qquad (5.3)$$

Wenn dies auf unser Beispiel angewandt wird:

$d_{min} = 2$. Wie gross ist aber $w_H(\underline{x}_i + \underline{x}_j)$?

Kombinationen von CWs, die Modulo-2 verknüpft werden	Resultat; Anzahl „1“
[0 0 0] + [0 1 1]	[0 1 1]; ➔ $w_H = 2$
[0 0 0] + [1 0 1]	[1 0 1]; ➔ $w_H = 2$
[0 0 0] + [1 1 0]	[1 1 0]; ➔ $w_H = 2$
[0 1 1] + [1 0 1]	[1 1 0]; ➔ $w_H = 2$
[0 1 1] + [1 1 0]	[1 0 1]; ➔ $w_H = 2$
[1 0 1] + [1 1 0]	[0 1 1]; ➔ $w_H = 2$

Damit ist (5.6) bestätigt: Der Wert für die Hammingdistanz d_{min} beträgt = 2 für alle Kombinationen. Ausserdem ist es auch der Wert für das Hamming-Gewicht der Modulo-2 Summe ($\underline{x}_i + \underline{x}_j$) für jede mögliche Kombination zweier gültiger Codeworte.

Fehlerdetektion

Gegeben sei ein Block-Code C mit der Hammingdistanz d_{min}. Dann ergibt sich:

Alle Muster mit $\leq (d_{min} - 1)$ Fehlern sind detektierbar. Total sind lediglich $2^K - 1$ Fehlermuster $\underline{e}$ undetektierbar, nämlich falls $\underline{e} = \underline{x}_j$, $\underline{y} = \underline{x}_i + \underline{e} = \underline{x}_k$.

In Worten: Falls der Fehlervektor $\underline{e}$ gerade einem gültigen Codewort $\underline{x}_j$ entspricht, dann ergibt die Modulo-2 Addition eines gesendeten gültigen Codeworts mit dem Fehlervektor ein anderes gültiges Codewort. Damit kann ein solcher Fehler nicht detektiert werden.

Wiederum an unserem Beispiel des (3,2) Block-Codes nachvollzogen:

Falls [0 0 0] gesendet wurde, und [0 1 1] beim Empfangsdetektor resultiert, ist der Fehlervektor e = [0 1 1].

[0 0 0] + [0 1 1] = [0 1 1] = $\underline{x}_k$, eben wieder ein gültiges Codewort.

Ein anderer Fall: Falls [0 1 1] gesendet wurde und der Fehlervektor $\underline{e}$ = [0 1 1] beträgt, resultiert:

[0 1 1] + [0 1 1] = [0 0 0], also ebenfalls ein gültiges Codewort, wie wir wissen.

Fehlerkorrektur

Das Ziel besteht darin, Fehler nicht nur zu erkennen, sondern auch zu korrigieren.

Gegeben sei wiederum ein Block-Code C mit der Hammingdistanz d_{min}.

Dann gilt:

Alle Muster mit $t \leq \lfloor (d_{\min} - 1)/2 \rfloor$ (5.4) Fehlern sind korrigierbar

Bei unserem Beispiel mit den Codewörtern [0 0 0], [1 1 0], [1 0 1, [0 1 1] beträgt $d_{min} = 2$.

Damit ist $(d_{min} - 1)/2 = 0.5$. Weil das ‚Floor'-Zeichen bedeutet, dass der Wert auf die nächste ganze Zahl abgerundet werden muss, wird der Wert 0.5 auf null abgerundet.

Also sind in diesem Beispiel „Muster mit null Fehlern korrigierbar“, also sind null Fehler korrigierbar.

Begriff „t-Fehlerkorrektur“: Der Begriff „t-Fehlerkorrektur“ bedeutet, dass alle Muster, die die Anzahl t oder weniger Fehler aufweisen, innerhalb der gegebenen Wahrscheinlichkeiten eines Binary Symmetric Channel korrigierbar sind. Eine Sicherheit für die Fehlerkorrektur gibt es jedoch nicht. Stets sind Wahrscheinlichkeiten die Richtgrössen.

Für die t-Fehlerkorrektur mit einem (N,K,t)-Block-Code auf einem Binary Symmetric Channel (BSC) gilt die Beziehung (5.5). Mit anderen Worten: Einen (N,K) Block-Code, der t Fehler korrigieren kann, bezeichnet man auch als (N,K,t) Block-Code. **Die Wahrscheinlichkeit, dass auf einem BSC mit einem (N,K,t) Block-Code ein Codewort korrekt übertragen wird, ist:**

t-Fehlerkorrektur mit einem (N,K,t)-Block-Code auf einem BSC:

$$P(\underline{u} = \underline{u}_e) = \sum_{i=0}^{t} \binom{N}{i} \cdot \varepsilon^i \cdot (1-\varepsilon)^{N-i} \qquad (5.5)$$

Darin bedeutet $\underline{u} = \underline{u}_e$, dass ein Fehlervektor $\underline{u}_e$ mit maximal definierter Anzahl Fehler auftreten kann.

Der Blockcode selbst kann aber mit der Formel nicht erhalten werden. Die Formel gibt nur an, wie gross die WSK ist, dass null Fehler pro CW resultieren plus die WSK, dass 1 Fehler pro CW resultieren plus die WSK, dass zwei Fehler pro CW resultieren, usw: bis t Fehler. Falls ein (N,K,t) Block Code vorliegt, wird die Aufsummierung nach der Berechnung der WSK, dass t Fehler resultieren gestoppt, **denn soviel kann ja korrigiert werden**. (Anmerkung: Zur Vertiefung des Verfahrens dient eine Übung).

Diese Formel wird nun erarbeitet. Dazu ist nochmals der BSC mit seinen Eigenschaften zu betrachten:

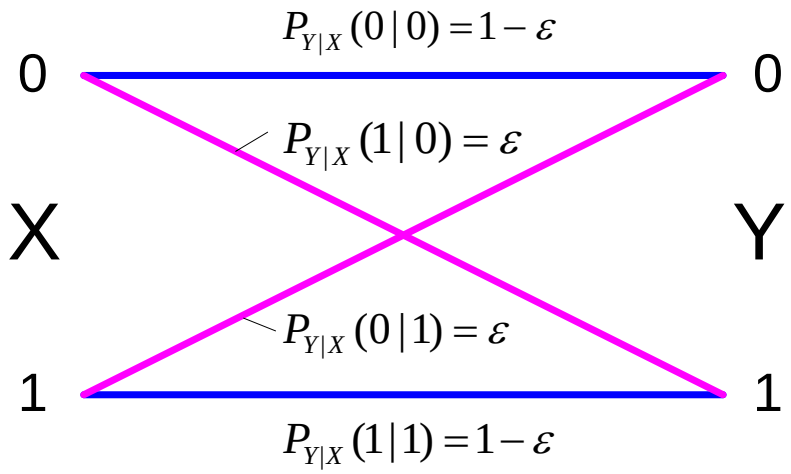

Abb. 5.6: Binary Symmetric Channel (BSC)

Im Folgenden werden die Wahrscheinlichkeiten von Ereignissen angegeben, die in der Kombination zu (5.5) führen.

$P_{\text{kein Fehler wenn 1 Bit gesendet}} = 1-\varepsilon$

$P_{\text{kein Fehler wenn N Bit gesendet}} = (1-\varepsilon)^N$

Dies ist eine Verkettung von Ereignissen, die hintereinander geschehen. Wenn ein Gesamtereignis als eine Aufeinanderfolge von Einzelereignissen auftritt, so müssen die Teil-Wahrscheinlichkeiten multipliziert werden, um die Gesamt-WSK zu berechnen.

$P_{\text{1 Fehler wenn 1 Bit gesendet}} = \varepsilon$

$P_{\text{1 Fehler wenn 2 Bit gesendet}} = \varepsilon\cdot(1-\varepsilon) + (1-\varepsilon)\cdot\varepsilon = 2\cdot\varepsilon\cdot(1-\varepsilon) = 2\cdot\varepsilon\cdot(1-\varepsilon)^1$

Wie ist das zu erklären: Wenn ein Ereignis auf verschiedene Weise eintreten kann, müssen die Einzelwahrscheinlichkeiten der verschiedenen Möglichkeiten addiert werden. Der Term $\varepsilon\cdot(1-\varepsilon)$ bedeutet: Der Fehler tritt beim ersten Bit auf; das zweite Bit wird richtig übertragen.

Der zweite Term $(1-\varepsilon)\cdot\varepsilon$ bedeutet: Der Fehler tritt beim zweiten Bit auf; das erste Bit wurde richtig übertragen.

Die genau gleiche Überlegung gilt, wenn ein Fehler auftritt bei drei gesendeten Bits: Der Fehler ist entweder beim ersten Bit (was dem Term $\varepsilon\cdot(1-\varepsilon)\cdot(1-\varepsilon)$ in der nachfolgenden Summe entspricht) oder im zweiten Bit (vgl. Term $(1-\varepsilon)\cdot\varepsilon\cdot(1-\varepsilon)$) oder eben im dritten Bit (vgl. Term $(1-\varepsilon)\cdot(1-\varepsilon)\cdot\varepsilon$). Weil es sich um verschiedene Weisen, wie das Ereignis „Ein Fehler bei drei gesendeten Bits" auftreten kann, handelt, müssen die Einzel-WSKs addiert werden:

$$P_{\text{1 Fehler wenn 3 Bit gesendet}} = \varepsilon\cdot(1-\varepsilon)\cdot(1-\varepsilon) + (1-\varepsilon)\cdot\varepsilon\cdot(1-\varepsilon) + (1-\varepsilon)\cdot(1-\varepsilon)\cdot\varepsilon$$
$$= 3\cdot\varepsilon(1-\varepsilon)^2$$

$$P_{\text{2 Fehler wenn 2 Bit gesendet}} = \varepsilon\cdot\varepsilon = \varepsilon^2$$

$$P_{\text{q Fehler wenn q Bit gesendet}} = \varepsilon^q$$

Das sind wiederum zwei Beispiele, wo es sich um eine Verkettung von Ereignissen handelt.

Aus den angestellten Überlegungen folgt für nun den Binary Symmetric Channel:

$$P_{\text{1 Fehler wenn N Bit gesendet}} = N\cdot\varepsilon^1\cdot(1-\varepsilon)^{N-1}$$

Daraus folgt:

$$P_{\text{t Fehler wenn N Bits gesendet}} = \binom{N}{t}\cdot\varepsilon^t\cdot(1-\varepsilon)^{N-t}$$

Die Zahl „N tief t" berechnet sich wie folgt (Definition: N tief 0 =1):

Der Wert von (N tief t) berechnet sich wie folgt:

$$\binom{N}{t} = \frac{N!}{t!\cdot(N-t)!}$$

Falls nun alle Möglichkeiten aufsummiert werden, also

$P_{\text{1 Fehler}} + P_{\text{2 Fehler}} + \ldots\ldots + P_{\text{t Fehler}}$ berechnet werden soll, so wird die Variable t durch die Variable i ersetzt und die Summenformel gebildet:

$$P_{\text{0 Fehler oder 1 Fehler oder 2 Fehler oder … oder t Fehler}} = P(\underline{u} = \underline{u}_e)$$

Und damit resultiert wiederum die Beziehung (5.5)

t-Fehlerkorrektur mit einem (N,K,t)-Block-Code auf einem BSC:

$$P(\underline{u} = \underline{u}_e) = \sum_{i=0}^{t}\binom{N}{i}\cdot\varepsilon^i\cdot(1-\varepsilon)^{N-i} \qquad (5.5)$$

Minimum Distance Decoding

Bei diesem Verfahren wird die folgende Annahme getroffen:

Wenige Fehler sind wahrscheinlicher als viele Fehler.

Aufgrund dieser Annahme wird ein empfangenes Codewort, das mit einem oder evtl. mehreren Bitfehlern behaftet ist, zum „nächstgelegenen" Codewort korrigiert.

Repetition: Bei einem Block-Code C mit der Hammingdistanz d_{min} gilt gemäss (5.4) die Aussage:

Alle Muster mit $t \leq \lfloor (d_{min} - 1)/2 \rfloor$ (5.4) Fehlern sind korrigierbar

Das „Floor"-Zeichen bedeutet, dass auf die nächste ganze Zahl abgerundet werden muss.

Beispiel mit minimaler Hammingdistanz von fünf:

Das gesendete Codewort sei $\underline{x}_i$ und das empfangene (fehlerbehaftete) Codewort sei $\underline{y}$. Andere gültige Codeworte weisen mindestens 5 Bits auf, die vom Codewort $\underline{x}_i$ verschieden sein müssen. Dies gilt für alle Codeworte des so vereinbarten Codealphabets: Alle gesendeten Codeworte unterscheiden sich mindestens in fünf Bits. Zwei Codeworte, ein Codewort $\underline{x}_i$ und ein Codewort $\underline{x}_j$ haben also eine minimale Hammingdistanz von 5.

Das empfangene fehlerbehaftete CW $\underline{y}$ wird also im Empfänger korrigiert, indem die Annahme getroffen wird, das CW $\underline{x}_i$ sei das ursprünglich gesendete CW. Es wird zum „nächstgelegenen" CW korrigiert.

Das folgende Bild soll dies veranschaulichen:

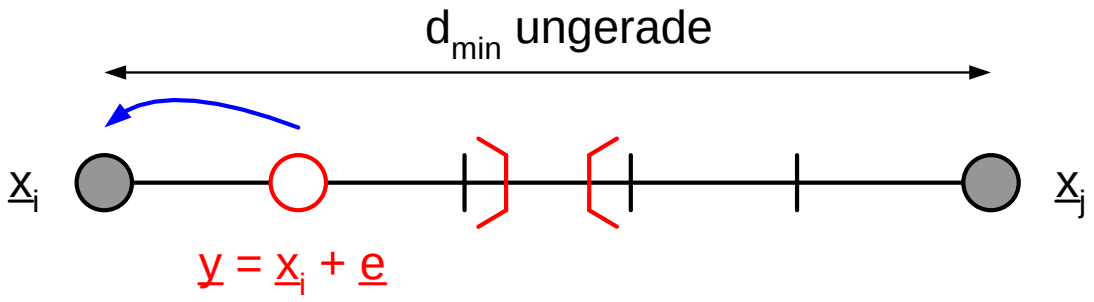

Abb. 5.7: Minimum Distance Decoding mit ungeradem d_{min}

Die eingezeichneten Schranken geben an, in welche Richtung korrigiert wird.

Für den Fall, dass d_{min} eine gerade Zahl ist, kann dies wie folgt veranschaulicht werden:

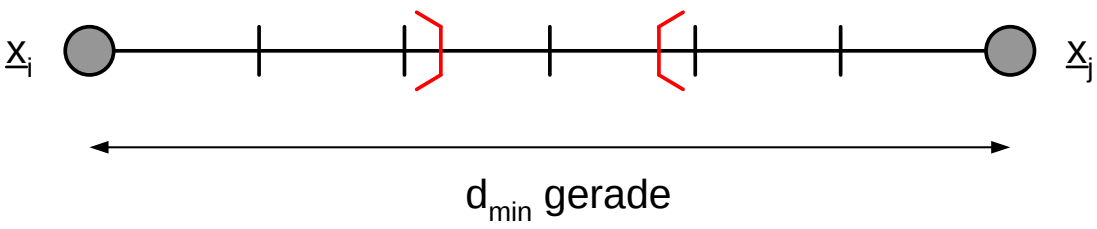

Abb. 5.8: Minimum Distance Decoding mit geradem d_{min}

Generator-Matrix

Für jeden linearen (N,K) Code existiert eine K x N Generator-Matrix G. Dabei ist N gleich die Summe aus Anzahl Informationsbits plus Anzahl Parity Check Bits. Die Zahl K ist die Anzahl der Informationsbits:

$$[x_0, \ldots, x_{N-1}] = [u_0, \ldots, u_{K-1}] \cdot G \quad \text{bzw.} \quad \underline{x} = \underline{u} \cdot G \qquad (5.6)$$

Beispiel mit einem linearen (7,4) Hamming Code

$x_0 \ldots\ldots\ldots\ldots x_6$ $\quad u_0 \ldots u_3$

$$[0\,0\,0\,1\,1\,0\,1] = [1\,1\,0\,1] \begin{bmatrix} 1\,1\,0\,\mathit{1}\,0\,0\,0 \\ 0\,1\,1\,0\,\mathit{1}\,0\,0 \\ 1\,1\,1\,0\,0\,\mathit{1}\,0 \\ 1\,0\,1\,0\,0\,0\,\mathit{1} \end{bmatrix}$$

$\underline{x} = [x_0, \ldots, x_6]$ $\quad \underline{u} = [u_0, \ldots, u_3]$ $\quad$ G

$x_6 = u_3$
$x_5 = u_2$
$x_4 = u_1$
$x_3 = u_0$
} systematisch

$x_2 = u_1 + u_2 + u_3$
$x_1 = u_0 + u_1 + u_2$
$x_0 = u_0 + u_2 + u_3$

Die Werte unterliegen einer Modulo 2 Addition.

Beispiel: $x_2 = (u_1 + u_2 + u_3) \bmod 2 = (1 + 0 + 1) \bmod 2 = 2 \bmod 2 = 0$. Dabei ist zu beachten:

0 mod 2 = 0

1 mod 2 = 1

2 mod 2 = 0

3 mod 2 = 1

4 mod 2 = 0

usw.

Weil die K=4 Informationsbits $\underline{u}$ auch im Codewort $\underline{x}$ erscheinen, ist der Code systematisch.

Linearer systematischer Block-Code

Bei einem solchen Code besitzt die Generator-Matrix die Form $G = [P\ I_K]$. I_K ist einen K x K – Einheitsmatrix. Beachte: Keine Multiplikation von P und I_K. Es soll lediglich ausgedrückt werden, dass die Einheitsmatrix I_K rechts einer Matrix P angefügt ist.

Die Matrix P ist: $P = \begin{bmatrix} 110 \\ 011 \\ 111 \\ 101 \end{bmatrix}$ Die Matrix I_K ist: $I_K = \begin{bmatrix} 1000 \\ 0100 \\ 0010 \\ 0001 \end{bmatrix}$

Damit resultiert: Die Parity-Check Bits x_0, x_1 und x_2 sind lineare Modulo 2 Summen der Informationsbits u_0, u_1, u_2 und u_3.

Parity-Check-Matrix

Jeder lineare (N,K) Code hat eine (N-K) x N Parity-Check-Matrix H, so dass die folgende Beziehung gilt:

$$[x_0, \ldots, x_{N-1}] \cdot H^T = [0, \ldots, 0] \quad \text{bzw.} \quad \underline{x} \cdot H^T = \underline{0} \qquad (5.7)$$

Falls G in systematischer Form vorliegt, also $G = [P\ I_K]$, dann gilt:

$$H = [I_{N-K}\ P^T]$$

Die transponierte Matrix P^T kann aus der Matrix P durch folgende Massnahme bestimmt werden:

Eine Spalte der Matrix P wird zu einer Zeile: Die linke Spalte von P wird zur ersten Zeile von P^T, usw.

Beachte: Keine Multiplikationen innerhalb der eckigen Klammern, wie das Beispiel veranschaulicht:

Wie benutzen wiederum unsere bekannte Generatormatrix G aus dem vorangehenden Beispiel:

$$G = \begin{bmatrix} 1101000 \\ 0110100 \\ 1110010 \\ 1010001 \end{bmatrix}; \qquad I_{N-K} = I_3 = \begin{bmatrix} 100 \\ 010 \\ 001 \end{bmatrix}; \qquad P^T = \begin{bmatrix} 1011 \\ 1110 \\ 0111 \end{bmatrix}$$

Demzufolge wird die Parity-Check-Matrix H: $H = \begin{bmatrix} 1001011 \\ 0101110 \\ 0010111 \end{bmatrix}$

Es existiert im weitern ein Syndrom s, das nur vom Fehlervektor e abhängt.

Die Themen:

Parity Check Matrix
Syndrom- / Table-Lookup-Decoding
Zyklische Codes: Generator
Zyklische Codes: Encoder
Zyklische Codes: Syndrom
BCH-Codes (Übersicht)

sind im Foliensatz beschrieben.

5.3 Anhang

5.3.1 Tabelle: Binäre BCH Codes bis Länge N=255

N = Anzahl Codebits

K = Anzahl Informationsbits; t = Anzahl korrigierbare Fehler

Falls die Theorie der zyklischen Codes behandelt wird (je nach Zeitbudget):

Generatorpolynom: Beispiel: Oktal 721 ➔ dann sind die Koeffizienten des Generatorpolynoms g(D) des zyklischen Codes wie folgt:

1 1 1 0 1 0 0 0 1 ➔ $D^8 + D^7 + D^6 + D^4 + 1 = g(D)$

N	K	t	Oktal	Generator Polynom: Koeffizienten
7	4	1	13	001 011
15	11	1	23	010 011
15	7	2	721	111 010 001
15	5	3	2467	010 100 110 111
31	26	1	45	100 101
31	21	2	3551	011 101 101 001
31	16	3	…..	…..
31	11	5		
31	6	7		
63	57	1		
63	51	2		
63	45	3		
63	39	4		
63	36	5		
63	30	6		
63	24	7		
63	18	10		
63	16	11		
63	10	13		
63	7	15		
127	120	1		
127	113	2		
127	106	3		
127	99	4		
127	92	5		
127	85	6		
127	78	7		
127	71	9		

127	64	10		
127	57	11		
127	50	13		
127	43	14		
127	36	15		
127	29	21		
127	22	23		
127	15	27		
127	8	31		
255	247	1		
255	239	2		
255	231	3		
255	223	4		
255	215	5		
255	207	6		
255	199	7		
255	191	8		
255	187	9		
255	179	10		
255	171	11		
255	163	12		
255	155	13		
255	147	14		
255	139	15		

Quelle: ‚Error Control Coding', pp. 1231 & 1232; Shu Lin & Daniel J. Costello, Jr; PEARSON ; 2004

5.3.2 Abkürzungen

A/D	Analog/Digital
AWGN	Additive White Gaussian Noise (additives weisses Gauss'sches Rauschen)
BCH	Bose, Chaudhuri & Hocquenchem
BER	Bit Error Ratio
BSC	Binary Symmetric Channel
CW	Codewort
LFSR	Linear Feedback Shift Register
Rx	Receive x
Tx	Transmit x
WSK	Wahrscheinlichkeit

5.3.3 Literatur

[1] John G. Proakis & Masoud Salehi; “Grundlagen der Kommunikationstechnik”, Pearson Verlag; 2004

[2] Shu Lin & Daniel J. Costello, “Error Control Coding”, Prentice Hall; ISBN 0-13-017973-B