Unterstützende Materialien zur Vorlesung

Verfahren zur Kanalcodierung – Teil 1

Prof. Dr. Bernd Friedrichs
KIT CEL

Inhalt

- Einführung, Modell der Übertragung
- Diskrete Kanäle (DMC, BSC, AWGN), Hard- und Soft-Decision, statistische Beschreibung (Q-Funktion)
- Rechnen mit bedingten Wahrscheinlichkeiten am Beispiel medizinischer Testverfahren (Corona)
- Prinzip der Blockcodierung
- Hammingdistanz
- Maximum-Likelihood-Decodierung allgemein und speziell für BSC und AWGN

In der Shannon'schen Theorie werden drei Arten von Codierung unterschieden:

Quellencodierung (source coding): Die Nachrichten werden so komprimiert, daß zwar keine Informationen verloren gehen und somit eine perfekte Wiedergewinnung der Nachrichten möglich ist, aber dafür wird die Anzahl der zu übertragenden Symbole reduziert. Durch Quellencodierung wird also überflüssige Redundanz eliminiert und das Übertragungssystem entlastet.

Kanalcodierung (error control coding): Diese stellt Methoden und Verfahren zur Verfügung, mit denen Informationen von einer Quelle zur Sinke mit einem Minimum an Fehlern übertragen werden können. Den eigentlichen Informationen wird sendeseitig kontrolliert Redundanz hinzugefügt, so daß bei der Übertragung entstandene Fehler empfangsseitig erkannt und korrigiert werden können. Damit läßt sich eine extrem hohe Zuverlässigkeit der übertragenen Daten erreichen. Ferner sind Störungen kompensierbar, die durch andere Maßnahmen, wie beispielsweise durch eine Erhöhung der Sendeleistung, prinzipiell nicht zu verhindern wären.

Kryptographie: Darunter wird die Codierung zur Verschlüsselung verstanden, um Nachrichten für Unberechtigte unlesbar zu machen bzw. um zu verhindern, daß Nachrichten gefälscht oder vorgetäuscht werden können. Während durch Kanalcodierung Nachrichten auch im Fall von Störungen lesbar bleiben, sollen die verschlüsselten Nachrichten auch bei ungestörter Übertragung ohne Kenntnis des Schlüssels unlesbar sein.

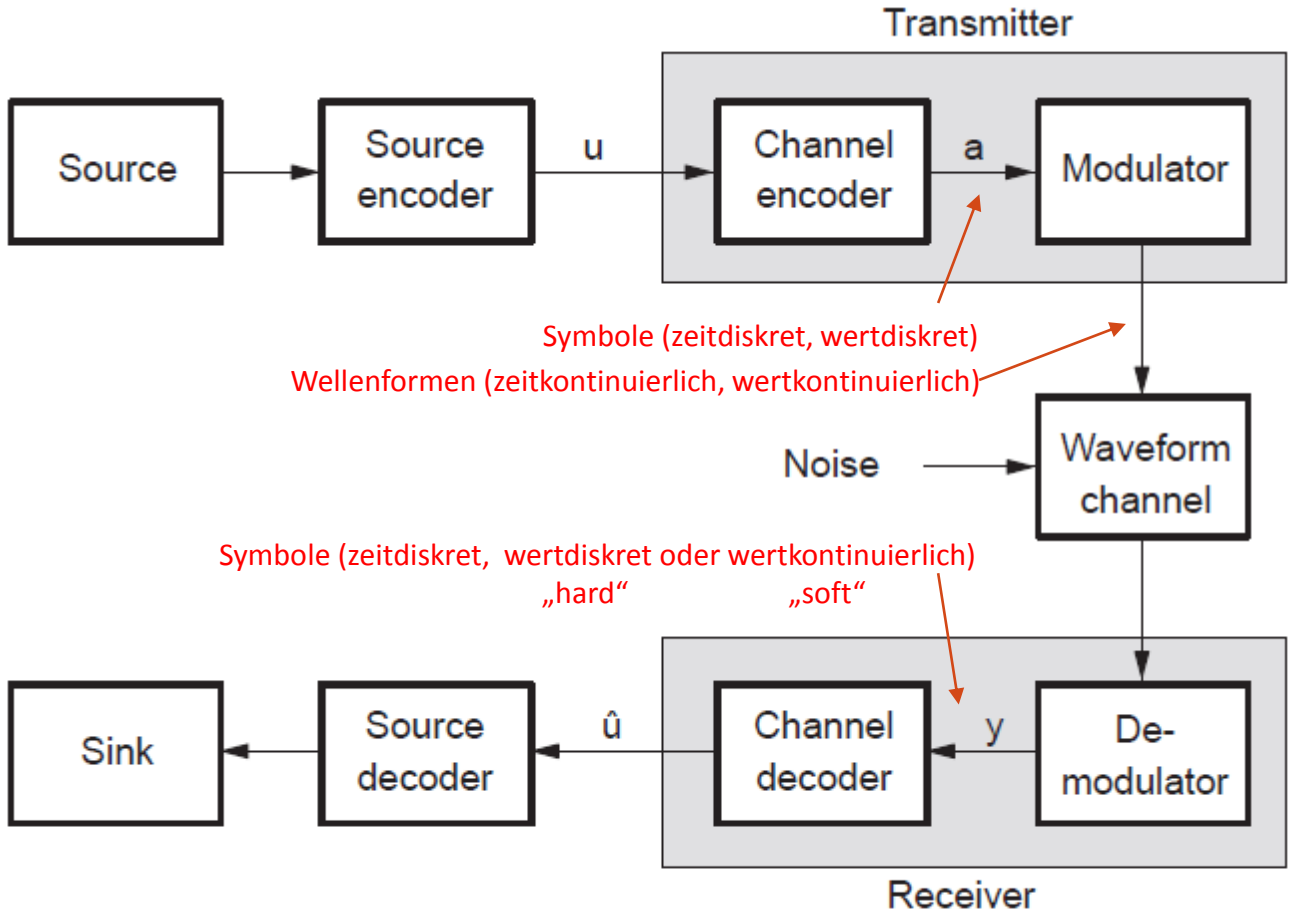
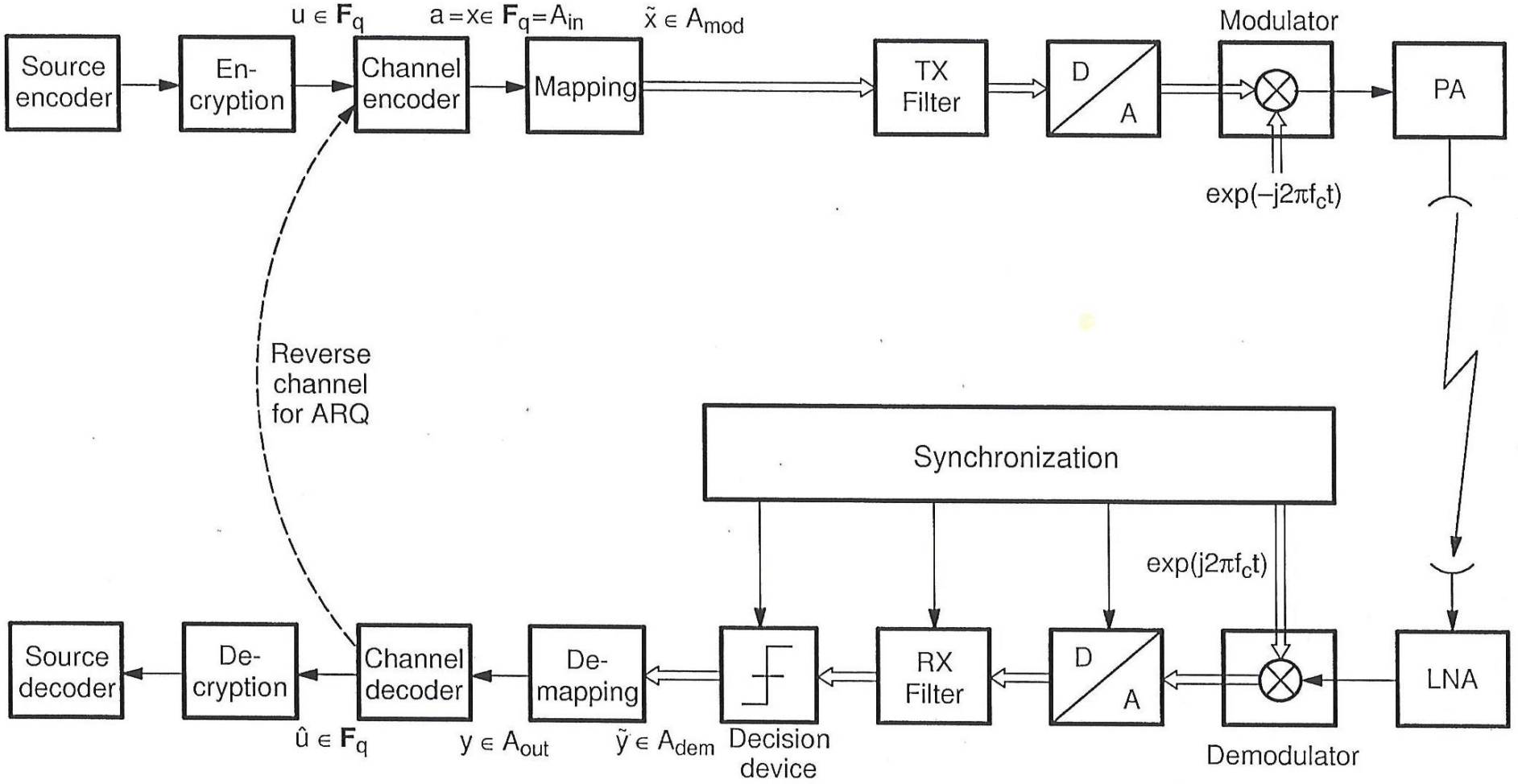


Bild 1.1. Digitale Nachrichtenübertragung mit Kanal- und Quellencodierung

Neben a wird auch die Bezeichnung x verwendet um zu unterscheiden wie folgt:

a = Ausgangswerte des Kanalencoders

x = Eingangswerte des Modulators bzw. Diskreten Kanals



Typisches Blockdiagramm einer drahtlosen Übertragung

Die Daten u werden direkt als Quellendaten angesprochen mit der Bezeichnung *Infobits* im Binärfall bzw. *Infosymbole* im mehrstufigen Fall. Der *Encoder* überführt die Infosymbole bzw. Infobits u in die *Codesymbole* bzw. *Codebits* a. Dabei wird Redundanz hinzugefügt, so daß die Datenrate durch den Encoder erhöht wird.

Mit dem *Modulator* wird der Encoder an den *physikalischen Kanal* (waveform channel, continuous channel, transmission channel) angeschlossen. Der physikalische Kanal kann keine diskreten Symbole übertragen, sondern nur zeitkontinuierliche Signale. Somit ist es die Aufgabe des Modulators, den diskreten Werten a derartige Signale zuzuordnen, die über den physikalischen Kanal übertragbar sind. Darin enthalten ist die Anpassung des modulierten Signals an den Übertragungsbereich bzw. an das Spektrum des physikalischen Kanals, insbesondere also die Verschiebung des Basisbandsignals in die Bandpaßlage. Bei den in Kapitel 10 behandelten Verfahren der trelliscodierten Modulation ist die Aufteilung des Senders in einen Encoder und einen Modulator allerdings nicht mehr eindeutig und in der Form aus Bild 1.1 auch nicht sinnvoll.

Der *physikalische Kanal* ist prinzipiell nicht ideal, d.h. er verändert die Signale bei der Übertragung. Das gilt sowohl bei drahtgebundener Übertragung (z.B. Teilnehmeranschlußleitung, Koaxialkabel, Glasfaserkabel), bei terrestrischen Funkkanälen (z.B. Mobilfunk, Richtfunk, Rundfunk, Kurzwellenfunk), bei Satellitenstrecken, bei Speicherung (z.B. Magnetmedien, elektronische und optische Speicher) wie natürlich auch bei Kombinationen dieser Kanäle. Der physikalische Kanal ist beispielsweise charakterisiert durch nichtideale Amplituden- und Phasengänge, durch Verzerrungen, durch Störungen aufgrund von Rauschen oder Übersprechen oder aufgrund atmosphärischer Effekte oder verschiedener Ausbreitungswege sowie durch absichtliche Störungen.

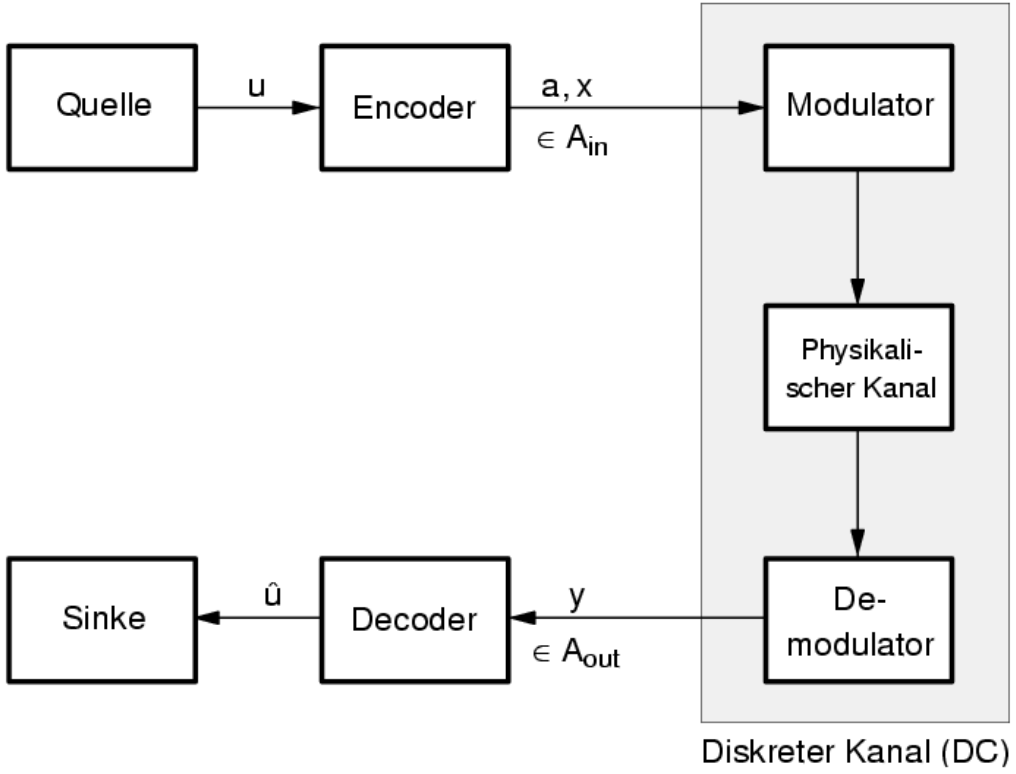
Am *Demodulator* liegen also nicht exakt die zeitkontinuierlichen Sendesignale an, sondern nur eine gestörte und verfälschte Version davon. Dennoch sollte der Empfänger die zeitdiskreten Sendewerte bzw. die Infosymbole möglichst genau rekonstruieren. Dazu wird zunächst im Demodulator aus dem Bandpaßsignal das Basisbandsignal zurückgewonnen, was bei kohärenten Empfängern eine ideale Träger- und Phasensynchronisation einschließt. Daraus wird eine zeitdiskrete Wertefolge hergestellt, so daß jedem Codesymbol a ein *Empfangswert* y entspricht. Bei der sogenannten *Soft-Decision Demodulation* soll der Demodulator derartige Werte y herstellen, die für den Decoder möglichst viel Information enthalten – es muß dann nicht zwangsläufig das Ziel sein, daß y möglichst genau a entspricht.

Der *Decoder* arbeitet zeitdiskret: Aus der im Takt der Codesymbole anliegenden Folge der Empfangswerte y wird eine Schätzung $\hat{u}$ für die Infosymbole u abgeleitet, wobei diese Schätzung i.a. eine zeitliche Verzögerung aufweist. Im idealen Fall arbeitet der Decoder sogar sequenzweise: Erst nach dem Empfang einer ganzen Sequenz von Empfangswerten wird auf einen Schlag die ganze Sequenz der Infosymbole geschätzt.

Für den Modulator sind die Codesymbole a nur Sendewerte ohne Kenntnis der Codierung. Um dies in besonderen Fällen hervorzuheben, werden in Analogie zu den Ausgangswerten y des Demodulators die Eingangswerte des Modulators auch mit x statt a bezeichnet.

diskret = zeitdiskret

(aber eventuell wertkontinuierlich)



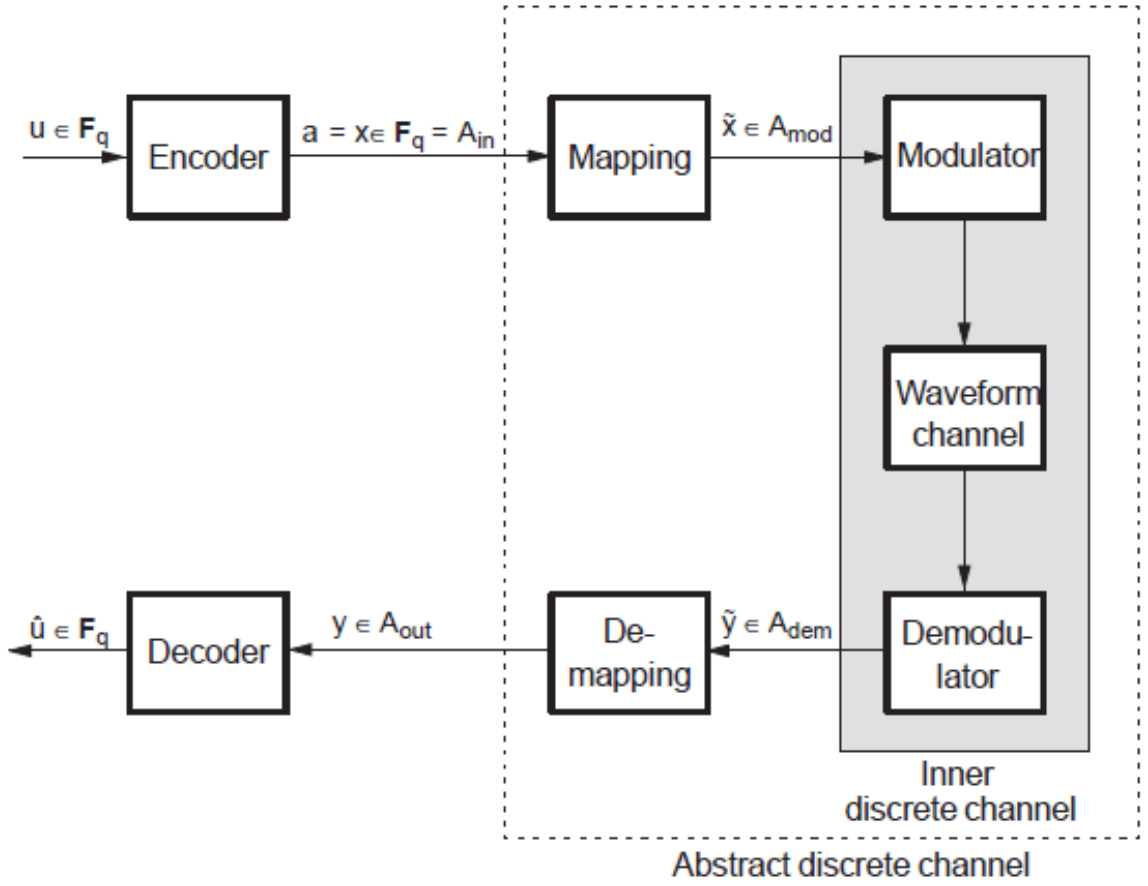


Bild 1.2. Erzeugung des dskreten Kanals durch das Modulatonssystem

Typischerweise wird im Encoder und Decoder mit binären Zahlen {0,1} „gerechnet“ (bei q=2), während der Modulator mit reellen Zahlen {-1,+1} gespeist wird.
Das Mapping im rechten Bild stellt diesen Zusammenhang formal her, meistens ist dieser Formalismus jedoch von untergeordneter Bedeutung.

$A_{in} = F_q = GF(q)$ wird später eingeführt.

In der formalen Beschreibung wird ein diskreter Kanal charakterisiert durch das Tripel $(\mathcal{A}_{\text{in}}, \mathcal{A}_{\text{out}}, P_{y|x})$. Dabei bedeuten:

$\mathcal{A}_{\text{in}} =$ *Eingangsalphabet* mit q Werten. Dies ist der Wertebereich für die Infosymbole u sowie für die Codesymbole a sowie für die geschätzten Infosymbole $\hat{u}$, d.h. $u, a, \hat{u}$ sind jeweils q-stufig. Fallunterscheidungen:

Der einfachste Fall ist $q = 2$ für Binärcodes, wobei die Symbole lediglich Bits sind. Der allgemeine Fall ist $q = p^m$ mit p als Primzahl und m als natürlicher Zahl. Der Normalfall für die meisten Codes ist $q = 2^m$, wobei den Symbole jetzt Bitgruppen (z.B. Bytes bei $m = 8$) entsprechen.

$\mathcal{A}_{\text{out}} =$ *Ausgangsalphabet:* Dies ist der Wertebereich für die Empfangswerte y. Fallunterscheidungen für den Demodulator:

Bei *Hard-Decision* gilt $\mathcal{A}_{\text{out}} = \mathcal{A}_{\text{in}}$, d.h. der Demodulator schätzt direkt die gesendeten Werte a bzw. x. Diese Situation liegt bei einfachen Blockcodes vor. Im binären Fall gilt dann $\mathcal{A}_{\text{in}} = \mathcal{A}_{\text{out}} = \{0, 1\}$.

Bei *Soft-Decision* umfaßt $\mathcal{A}_{\text{out}}$ mehr Werte als $\mathcal{A}_{\text{in}}$ – im Extremfall gilt sogar $\mathcal{A}_{\text{out}} = \mathbb{R}$ für einen wertkontinuierlichen Demodulatorausgang. In diesem Fall kann der Demodulator besonders viel Information über den Kanal (Zustand, Qualität) vermitteln. Beispielsweise teilt der Demodulator dem Decoder mit, mit welcher Sicherheit er seine Entscheidungen getroffen hat (sehr sicher oder gerade an der Grenze). Zwar kann prinzipiell jedes Codierungsverfahren diese Information ausnutzen, praktikabel ist das jedoch meistens nur bei Faltungscodes. Ein typischer Fall bei $\mathcal{A}_{\text{in}} = \{0, 1\}$ ist ein 8-stufiges $\mathcal{A}_{\text{out}}$, d.h. der Empfangswert wird mit 3-Bit quantisiert (siehe dazu auch Bild 1.4 und 1.5).

$P_{y|x} =$ *Übergangswahrscheinlichkeit* (Kanalstatistik; conditional, transition probability): Dabei ist $P_{y|x}(\eta|\xi)$ die bedingte Wahrscheinlichkeit dafür, daß $y = \eta$ empfangen wurde unter der Voraussetzung, daß $x = \xi$ gesendet wurde.

Input x und Output y des Kanals werden hier also als Zufallsgrößen angenommen, deren Werte mit $\xi \in \mathcal{A}_{\text{in}}$ und $\eta \in \mathcal{A}_{\text{out}}$ bezeichnet werden. Vereinfachend wird auch $P(y|x)$ geschrieben, wenn es auf die Unterscheidung zwischen den Zufallsgrößen und ihren Werten nicht ankommt. Für die Übergangswahrscheinlichkeit gilt generell:

$$\sum_{\eta \in \mathcal{A}_{\text{out}}} P_{y|x}(\eta|\xi) = 1 \quad \text{für alle } \xi \in \mathcal{A}_{\text{in}}. \tag{1.3.1}$$

Für den diskreten Kanal sind einige wichtige Fallunterscheidungen zu vermerken: Neben einer Hard-Decision oder Soft-Decision Demodulation können die Übertragungseigenschaften *zeitinvariant* oder auch *zeitvariant* sein. Ferner kann der diskrete Kanal ein *Gedächtnis* haben (d.h. der Empfangswert ist nicht nur vom zuletzt gesendeten Wert abhängig, sondern auch von den vorangehend gesendeten Werten) oder er ist *gedächtnislos* (d.h. der Empfangswert ist nur vom aktuell gesendeten Wert abhängig).

$P_{y|x}(0|1)$ = P(0 empfangen | 1 gesendet) **Übergangswahrscheinlichkeit** = Kanalbeschreibung
das Sendesymbol impliziert eine Wahrscheinlichkeitsverteilung der möglichen Empfangssymbole

$P_{x|y}(0|1)$ = P(0 gesendet | 1 empfangen) **A posteriori Wahrscheinlichkeit** = Ziel der Nachrichtenübertragung
aus dem Empfangssymbol wird auf das Sendesymbol geschlossen

Zusammenhang: $P_{y|x} = P_{x,y} / P_x = P_{x|y} * P_y / P_x$ (siehe auch Aufgabe 1.11)

$p = P(\text{Person krank}) = 0.002$

Testversagen:

$a = P(\text{Test negativ} \mid \text{Person krank}) = 0.01$

$b = P(\text{Test positiv} \mid \text{Person gesund}) = 0.001$

$P(\text{irrtümlich krank}) = P(\text{Person gesund} \mid \text{Test positiv})$

$$= \frac{P(\text{gesund} \wedge \text{positiv})}{P(\text{positiv})} = \frac{P(\text{positiv} \wedge \text{gesund})}{P(\text{positiv} \wedge \text{gesund}) + P(\text{positiv} \wedge \text{krank})}$$

$$= \frac{P(\text{positiv} \mid \text{gesund}) \cdot P(\text{gesund})}{P(\text{positiv} \mid \text{gesund}) \cdot P(\text{gesund}) + P(\text{positiv} \mid \text{krank}) \cdot P(\text{krank})}$$

$$= \frac{b(1-p)}{b(1-p) + (1-a)p} = 0.3351 \quad \approx \begin{cases} b/p & \text{für } b \ll p \\ 1 & \text{für } b \gg p \end{cases}$$

$P(\text{irrtümlich gesund}) = P(\text{Person krank} \mid \text{Test negativ})$

$$= \frac{P(\text{negativ} \wedge \text{krank})}{P(\text{negativ})}$$

$$= \frac{P(\text{negativ} \mid \text{krank}) \cdot P(\text{krank})}{P(\text{negativ} \mid \text{gesund}) P(\text{gesund}) + P(\text{negativ} \mid \text{krank}) \cdot P(\text{krank})}$$

$$= \frac{ap}{(1-b)(1-p) + ap} \approx 0.00002$$

Zahlenbeispiel 1 Mio Personen

	krank	gesund	
negativ	20	997002	997022
positiv	1980	998	2978
	2000	998000	1 000 000

$$P(\text{gesund} \mid \text{positiv}) = \frac{998}{2978} = 0.3351$$

$$P(\text{krank} \mid \text{negativ}) = \frac{20}{997022} = 0.00002$$

Das Zahlenbeispiel auf der vorigen Seite ist schon etliche Jahre alt.

Ein weiteres Beispiel auf dieser Seite reflektiert die Corona-Pandemie mit den im Juni 2020 allgemein vermuteten Parametern.
„krank“ = bis Juni 2020 akkumulierte Nachweise eine Corona-Infektion
(185000 Nachweise bei 83 Millionen ergibt p=0.002, ohne Berücksichtigung einer Dunkelziffer von nicht-erfassten Infektionen)
„positiv“ = Nachweis dieser Infektion durch einen Antikörpertest

$p = P(\text{krank}) = 0.002$ = Prävalenz

Testversagen:

$a = P(\text{negativ} \mid \text{krank}) = 0$ $\quad 1-a = P(\text{positiv} \mid \text{krank})$ = Sensitivität

$b = P(\text{positiv} \mid \text{gesund}) = 0.002 = p$ $\quad 1-b = P(\text{negativ} \mid \text{gesund})$ = Spezifität

Zahlenbeispiel 1 Mio Personen

	krank	gesund	
negativ	0	996.004	996.004
positiv	2000	1.996	3.996
	2000	998.000	1.000.000

$$P(\text{irrtümlich krank}) = P(\text{gesund} \mid \text{positiv}) = \frac{1996}{3996} \approx 0.5$$

$$P(\text{irrtümlich gesund}) = P(\text{krank} \mid \text{negativ}) = 0$$

Ein positiver Antikörpertest hat also nur eine begrenzte Aussagekraft: mit 50% war man einmal infiziert, mit 50% aber auch nicht.

Das überrascht angesichts der scheinbar kleinen Wahrscheinlichkeiten für Testversagen!

Definition 1.1 (DMC). *Als* diskreter gedächtnisloser Kanal *(DMC, Discrete Memoryless Channel) wird ein diskreter Kanal mit endlichen Alphabeten* $\mathcal{A}_{\text{in}}$ *und* $\mathcal{A}_{\text{out}}$ *bezeichnet, der zudem gedächtnislos und zeitinvariant sein soll. Die Gedächtnislosigkeit ist dadurch gekennzeichnet, daß die Übergangswahrscheinlichkeit für Sequenzen in ein Produkt der Übergangswahrscheinlichkeiten für Einzelsymbole übergeht:*

$$P(y_0, \ldots, y_{n-1}|x_0, \ldots, x_{n-1}) = \prod_{i=0}^{n-1} P(y_i|x_i). \tag{1.3.2}$$

Wenn die Übergangswahrscheinlichkeiten bei Hard-Decision gewisse Symmetrien erfüllen, reicht zur Charakterisierung des DMC ein einziger Parameter aus:

Definition 1.2 (Symmetrischer Hard-Decision DMC). *Als ein* q-närer symmetrischer Kanal mit Hard-Decision *wird ein DMC mit* $\mathcal{A}_{\text{in}} = \mathcal{A}_{\text{out}}$ *und der Übergangswahrscheinlichkeit*

$$P(y|x) = \left\{ \begin{array}{ll} 1 - p_e & y = x \\ p_e/(q-1) & y \neq x \end{array} \right\} \tag{1.3.3}$$

bezeichnet. Dieser Kanal ist eindeutig durch die Angabe der Symbol-Fehlerwahrscheinlichkeit p_e *bestimmt. Der binäre Fall mit* $\mathcal{A}_{\text{in}} = \mathcal{A}_{\text{out}} = \{0, 1\}$ *wird als* binärer symmetrischer Kanal *(BSC, Binary Symmetric Channel) bezeichnet.*

Für $p_e = 0$ ist der Kanal fehlerfrei und für $p_e = 1/2$ bei $q = 2$ wird in Kapitel 2 noch gezeigt, daß eine zuverlässige Übertragung prinzipiell unmöglich ist. Bei einem veränderlichen p_e würde ein zeitvarianter Kanal vorliegen. Diese Situation wird noch in Abschnitt 9.7 behandelt. Ausgeschrieben lautet (1.3.3) für den BSC:

$$\begin{aligned} P_{y|x}(0|0) &= P_{y|x}(1|1) = 1 - p_e \\ P_{y|x}(1|0) &= P_{y|x}(0|1) = p_e. \end{aligned} \tag{1.3.4}$$

Mit der Wahrscheinlichkeit p_e wird das Bit bei der Übertragung verfälscht und mit der Wahrscheinlichkeit $1 - p_e$ ist die Übertragung korrekt:

$$\begin{aligned} P(y = x) &= 1 - p_e \\ P(y \neq x) &= p_e. \end{aligned} \tag{1.3.5}$$

Beispiel: Unter der Voraussetzung, daß 110 gesendet wurde, wird 101 mit der Wahrscheinlichkeit $P_{y|x}(101|110) = P_{y|x}(1|1)P_{y|x}(0|1)P_{y|x}(1|0) = (1 - p_e) \cdot p_e \cdot p_e$ empfangen.

Das Prinzip des BSC zeigt Bild 1.3, wobei die Kanten von $x \in \mathcal{A}_{\text{in}}$ nach $y \in \mathcal{A}_{\text{out}}$ mit den Übergangswahrscheinlichkeiten $P(y|x)$ beschriftet sind.

Eine Verallgemeinerung des BSC ist der in Bild 1.3 ebenfalls dargestellte *binäre symmetrische Kanal mit Ausfällen* (BSEC, Binary Symmetric Erasure Channel), bei dem der Output ternär ist: $\mathcal{A}_{\text{out}} = \{0, ?, 1\}$. Hierbei entscheidet der Demodulator auf den "Wert" ?, wenn die Entscheidung auf 0 oder 1 sehr unsicher wäre. Für den Decoder ist es besser, über den Sendewert gar keine Information zu haben als eine Information, die in der Hälfte aller Fälle falsch ist. Der BSEC ist der einfachste Fall eines diskreten Kanals mit Soft-Decision mit

$$P(y|x) = \left\{ \begin{array}{ll} 1 - p_e - q_e & \text{für } y = x \\ q_e & \text{für } y = ? \\ p_e & \text{sonst} \end{array} \right\}. \qquad (1.3.10)$$

Natürlich gilt hierbei $P_{y|x}(0|x) + P_{y|x}(?|x) + P_{y|x}(1|x) = 1$ für $x \in \mathcal{A}_{\text{in}} = \{0, 1\}$. Für $q_e = 0$ wird der BSEC zum BSC und für $p_e = 0$ wird der BSEC zum reinen *Auslöschungskanal* (BEC, Binary Erasure Channel).

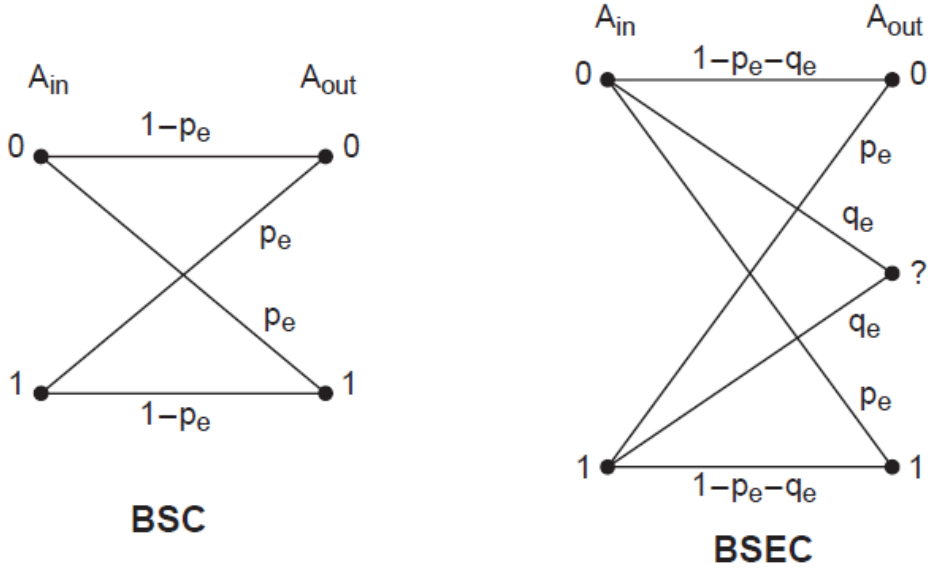


Bild 1.3. Modelle diskreter gedächtnisloser Kanäle (BSC, BSEC)

Für den q-nären symmetrischen Hard-Decision DMC gelten einige wichtige allgemeine Formeln:

(1) Mit P_{ee} (ee = error event) wird die Wahrscheinlichkeit bezeichnet, daß bei der Übertragung einer Sequenz $\boldsymbol{x} = (x_0, x_1, \ldots, x_{n-1})$ der Länge n mindestens ein Fehler auftritt:

$$\begin{aligned} P_{ee} &= P(\boldsymbol{y} \neq \boldsymbol{x}) \\ &= 1 - P(\boldsymbol{y} = \boldsymbol{x}) \\ &= 1 - P(y_0 = x_0, \ldots, y_{n-1} = x_{n-1}) \\ &= 1 - P(y_0 = x_0) \cdots P(y_{n-1} = x_{n-1}) \\ &= 1 - (1 - p_e)^n \qquad (1.3.6) \\ &\approx np_e, \quad \text{bei } np_e \ll 1. \qquad (1.3.7) \end{aligned}$$

(1.3.7) folgt aus der Binomialentwicklung $(1 - p_e)^n = \sum_{i=0}^{n} \binom{n}{i} (-p_e)^i$.

(2) Die Wahrscheinlichkeit dafür, daß eine Sequenz von n Bits in eine andere bestimmte Sequenz verfälscht wird, wobei r Fehler auftreten, beträgt:

$$P(\text{von } n \text{ Bits sind } r \text{ bestimmte Bits falsch}) = p_e^r (1 - p_e)^{n-r}. \qquad (1.3.8)$$

(3) Die Wahrscheinlichkeit für r Fehler in einer Sequenz von n Bits beträgt nach der Binomialverteilung (siehe Anhang A.3):

$$P(\text{von } n \text{ Bits sind } r \text{ Bits falsch}) = \binom{n}{r} p_e^r (1 - p_e)^{n-r}. \qquad (1.3.9)$$

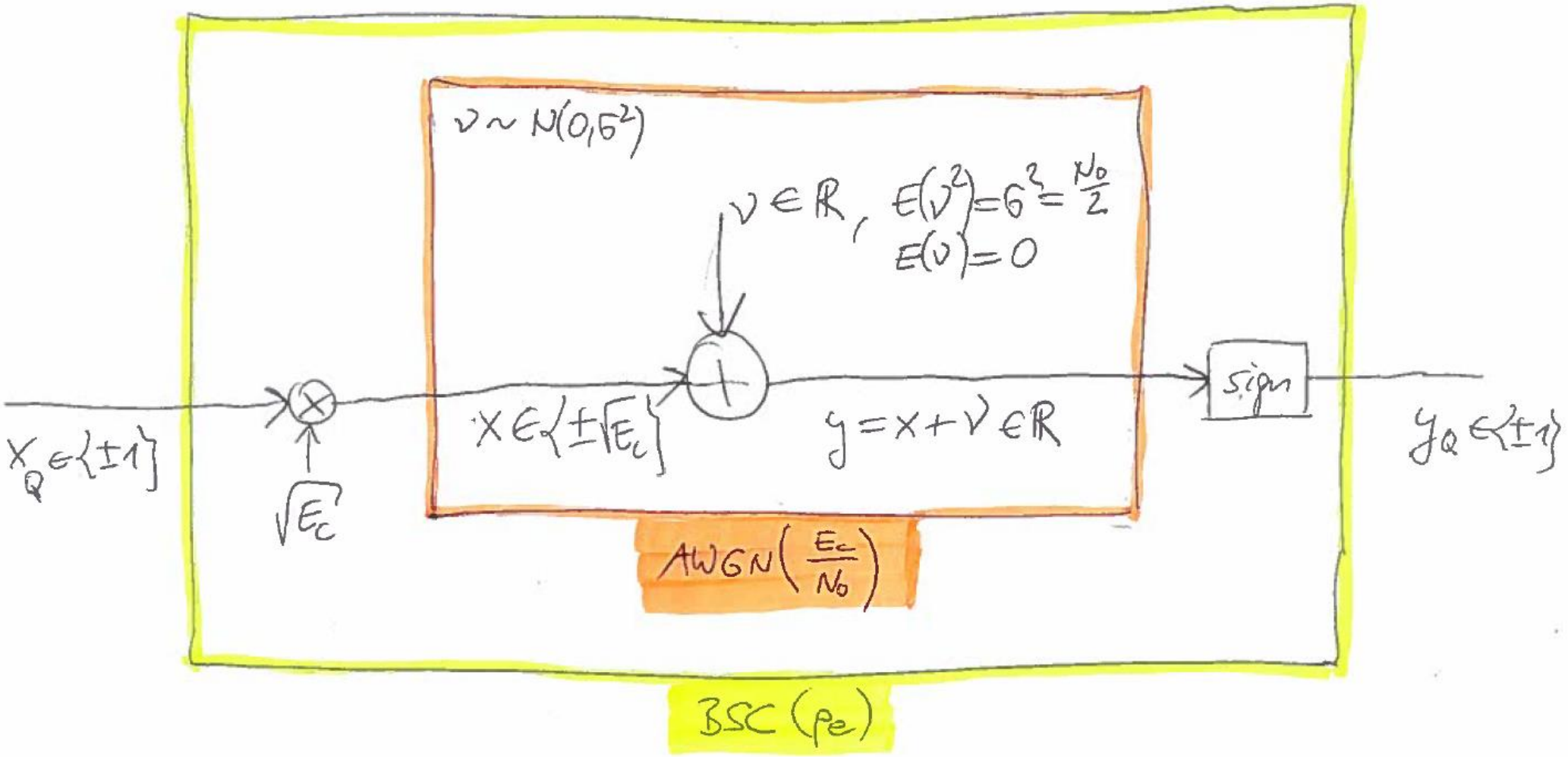
Definition 1.3. *Als AWGN-Kanal (Additive White Gaussian Noise) wird ein Kanal mit binärem Input bezeichnet, bei dem weißes normalverteiltes (Gaußsches) Rauschen ν additiv überlagert wird:*

$$y = x + \nu.$$

Dabei sind x und ν statistisch unabhängig. Mit E_c wird die Energie pro Codebit und mit N_0 wird die einseitige Rauschleistungsdichte bezeichnet. Für die Alphabete gilt $\mathcal{A}_{\text{in}} = \{-\sqrt{E_c}, +\sqrt{E_c}\}$ und $\mathcal{A}_{\text{out}} = \mathbb{R}$ und die Übergangswahrscheinlichkeiten haben die Form von Verteilungsdichten:

$$f_{y|x}(\eta|\xi) = \frac{1}{\sqrt{\pi N_0}} \exp\left(-\frac{(\eta-\xi)^2}{N_0}\right). \tag{1.3.11}$$

Also ist y bei gegebenem x normalverteilt mit dem Erwartungswert $x = \xi$ und der Varianz $\sigma^2 = N_0/2$, die der Varianz des Rauschens entspricht.



Zusammenhang: $p_e = Q\left(\sqrt{\frac{2E_c}{N_0}}\right)$

Wenn der AWGN mit binärer Modulation (ASK, Amplitude Shift Keying) betrieben wird und im Demodulator binär quantisiert wird, so ergibt sich wieder ein BSC mit der Bit-Fehlerwahrscheinlichkeit

$$p_e = P_{y|x}(y < 0 \mid x = +\sqrt{E_c}) = P_{y|x}(y > 0 \mid x = -\sqrt{E_c})$$
$$= \int_0^{+\infty} \frac{1}{\sqrt{\pi N_0}} \exp\left(-\frac{(\eta + \sqrt{E_c})^2}{N_0}\right) d\eta$$
$$= Q\left(\sqrt{\frac{2E_c}{N_0}}\right). \tag{1.3.12}$$

Dabei ist

$$Q(\alpha) = \frac{1}{\sqrt{2\pi}} \int_\alpha^\infty e^{-\eta^2/2}\, d\eta = \frac{1}{2}\mathrm{erfc}\left(\frac{\alpha}{\sqrt{2}}\right) \tag{1.3.13}$$
$$= P(\nu > \alpha\sqrt{N_0/2}) \tag{1.3.14}$$

die *komplementäre Gaußsche Fehlerfunktion* (siehe Anhang A.3). Den numerischen Zusammenhang zwischen p_e und E_c/N_0 zeigt Tabelle 1.1 und der graphische Verlauf ist in den Bildern mit Fehlerwahrscheinlichkeits-Kurven dargestellt (siehe z.B. Bild 1.10, Kurve "uncodiert", $E_c = E_b$).

Tabelle 1.1. BSC-Fehlerwahrscheinlichkeit

p_e	E_c/N_0 [dB]	p_e	E_c/N_0 [dB]
10^{-1}	−0,86	10^{-11}	13,52
10^{-2}	4,33	10^{-12}	13,93
10^{-3}	6,79	10^{-13}	14,31
10^{-4}	8,40	10^{-14}	14,66
10^{-5}	9,59	10^{-15}	14,99
10^{-6}	10,53	10^{-16}	15,29
10^{-7}	11,31	10^{-17}	15,57
10^{-8}	11,97	10^{-18}	15,84
10^{-9}	12,55	10^{-19}	16,09
10^{-10}	13,06	10^{-20}	16,32

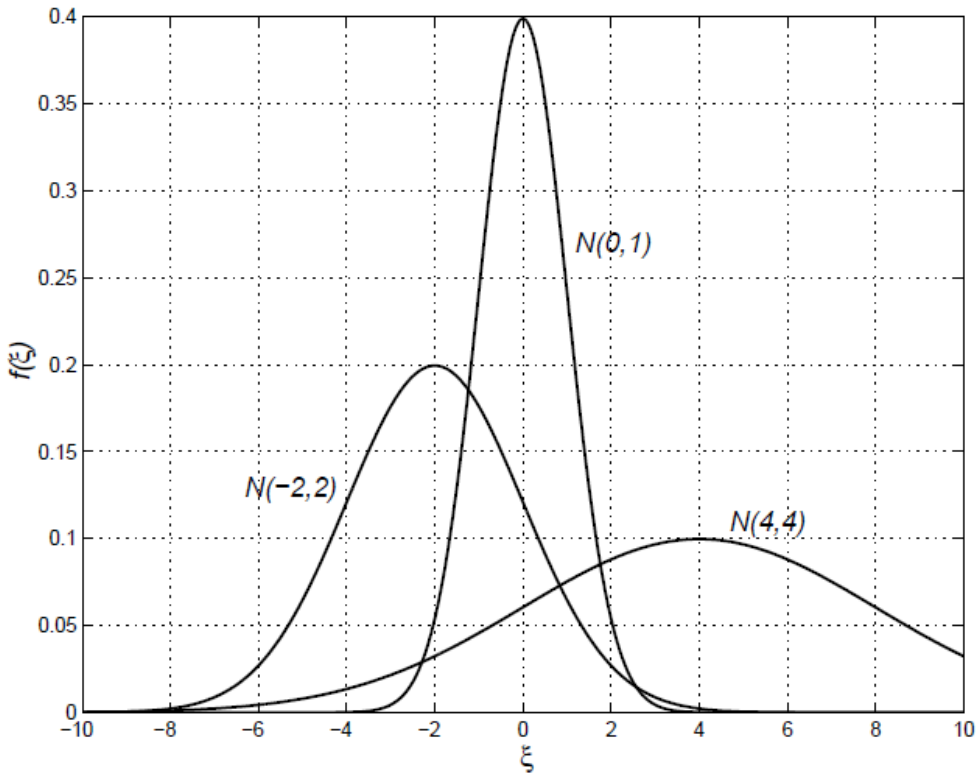


Figure A.3. PDF's of some Gaussian (normal) distributions

Table A.4. The probability that a Gaussian random variable is outside δ standard deviations of the mean

δ	$P(\lvert x-\mu\rvert > \delta\cdot\sigma)$
1	$3.17\cdot 10^{-1}$
2	$4.56\cdot 10^{-2}$
3	$2.70\cdot 10^{-3}$
4	$6.33\cdot 10^{-5}$
5	$5.73\cdot 10^{-7}$
6	$1.97\cdot 10^{-9}$
7	$2.56\cdot 10^{-12}$
8	$1.24\cdot 10^{-15}$

Im Qualitätsmanagement der Industrie ist/war die Six-Sigma-Methode eine populäre Modeerscheinung. Damit ist die Abweichung eines Messwertes vom Mittelwert um das 4.5-fache (nicht etwa das 6-fache) der Streuung gemeint.

Das ist ziemlich albern da es außer Rauschprozessen in RF und Optik kaum reale Vorgänge gibt die in hoher Präzision normalverteilt sind – und schon gar nicht bei industriellen Prozessen die immer durch Rückkopplungen geprägt sind.

Das physikalische Rauschen bei einem AWGN-Kanal entspricht jedoch mit allerhöchster Präzision dem mathematischen Modell der Normalverteilung.

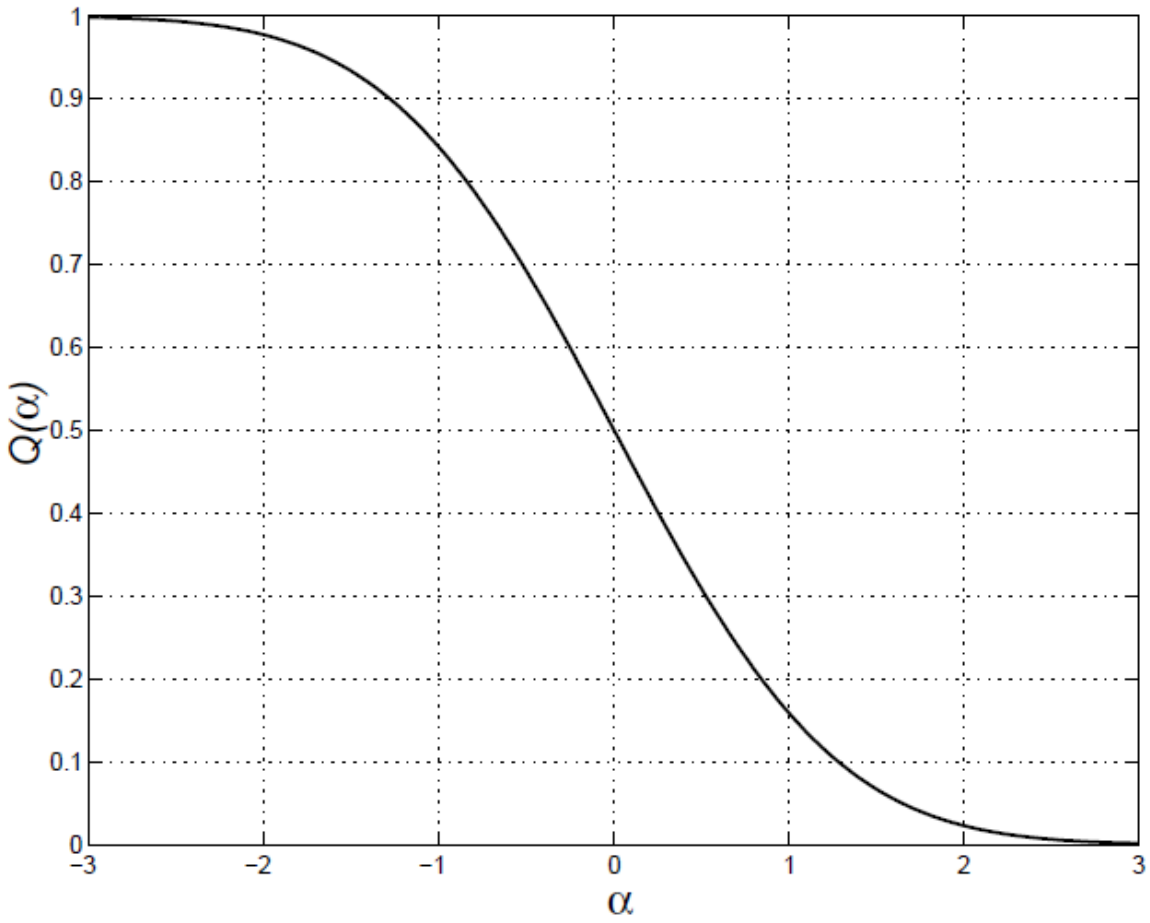


Figure A.4. The complementary Gaussian error function $Q(\alpha)$

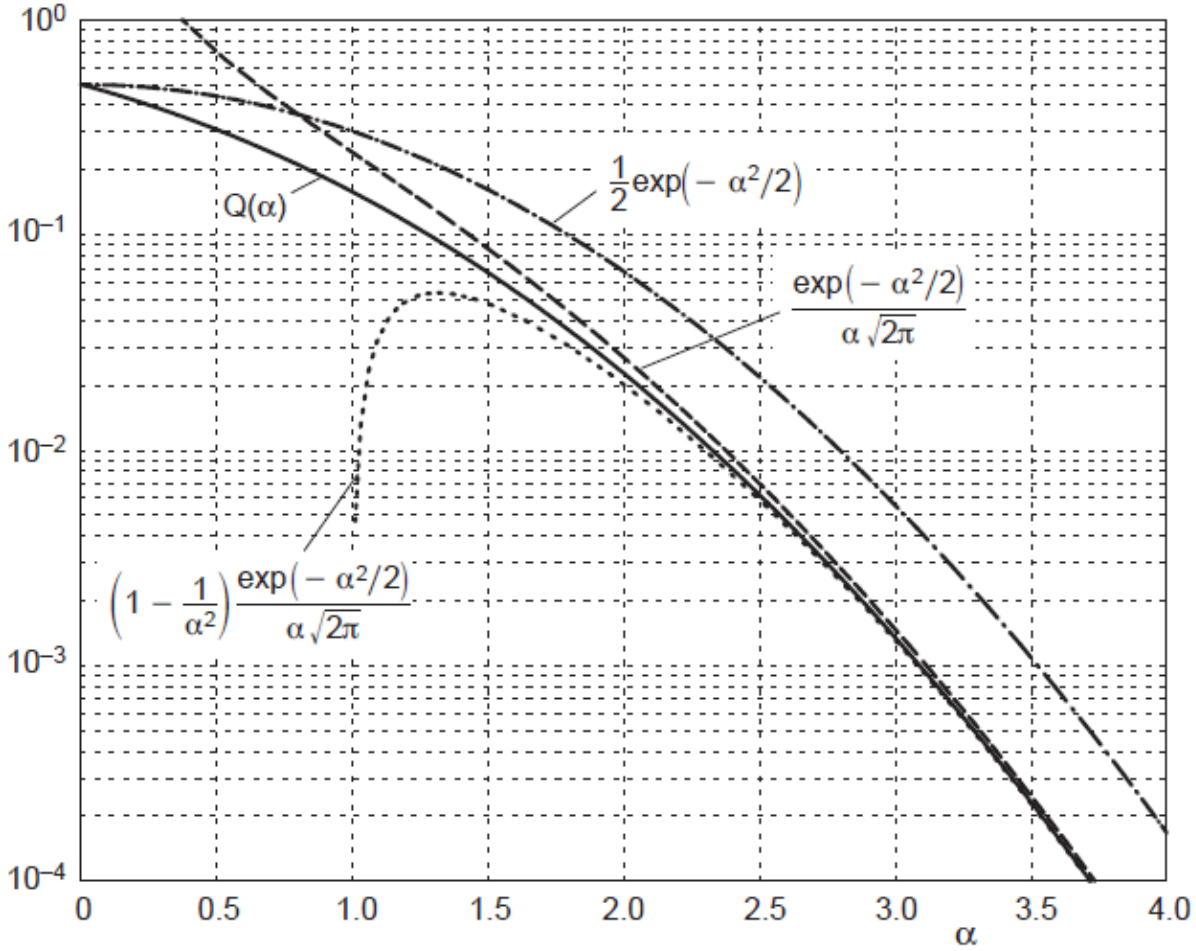


Figure A.5. Bounds for the complementary Gaussian error function $Q(\alpha)$

Wenn beim AWGN im Demodulator nicht binär mit 1 Bit sondern oktal mit 3 Bit quantisiert wird, so ergibt sich ein DMC mit $\mathcal{A}_{\text{in}} = \{-\sqrt{E_c}, +\sqrt{E_c}\}$ und oktalem $\mathcal{A}_{\text{out}} = \{-1, -1', -1'', -1''', +1''', +1'', +1', +1\}$. Von einiger Bedeutung ist dabei die Wahl der 7 Sprungstellen in der *Quantisierungskennlinie*, was in [49] genauer analysiert wird.

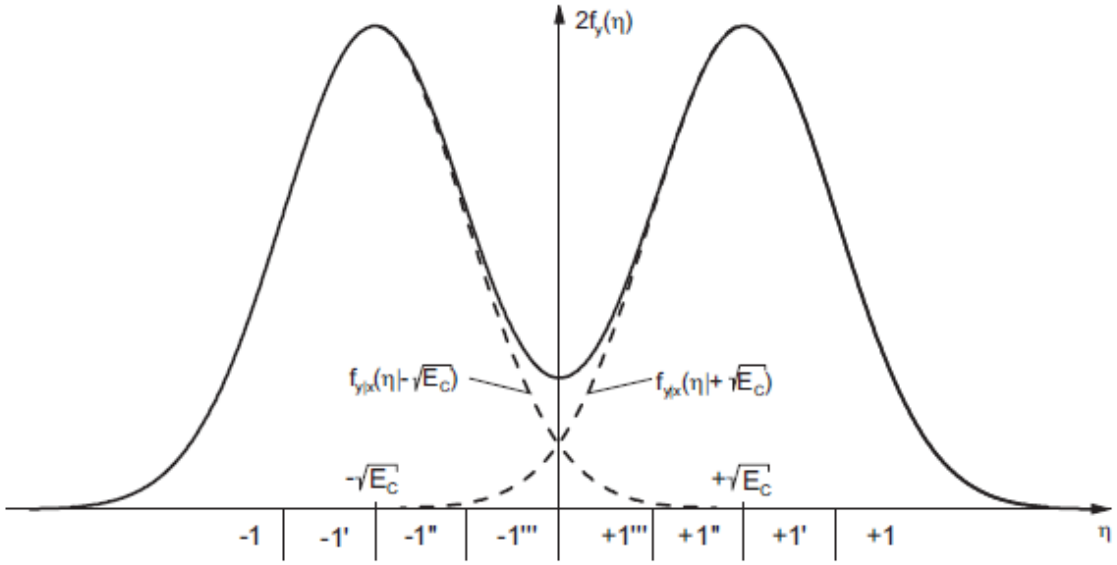


Bild 1.4. Oktale Quantisierung der AWGN-Empfangswerte

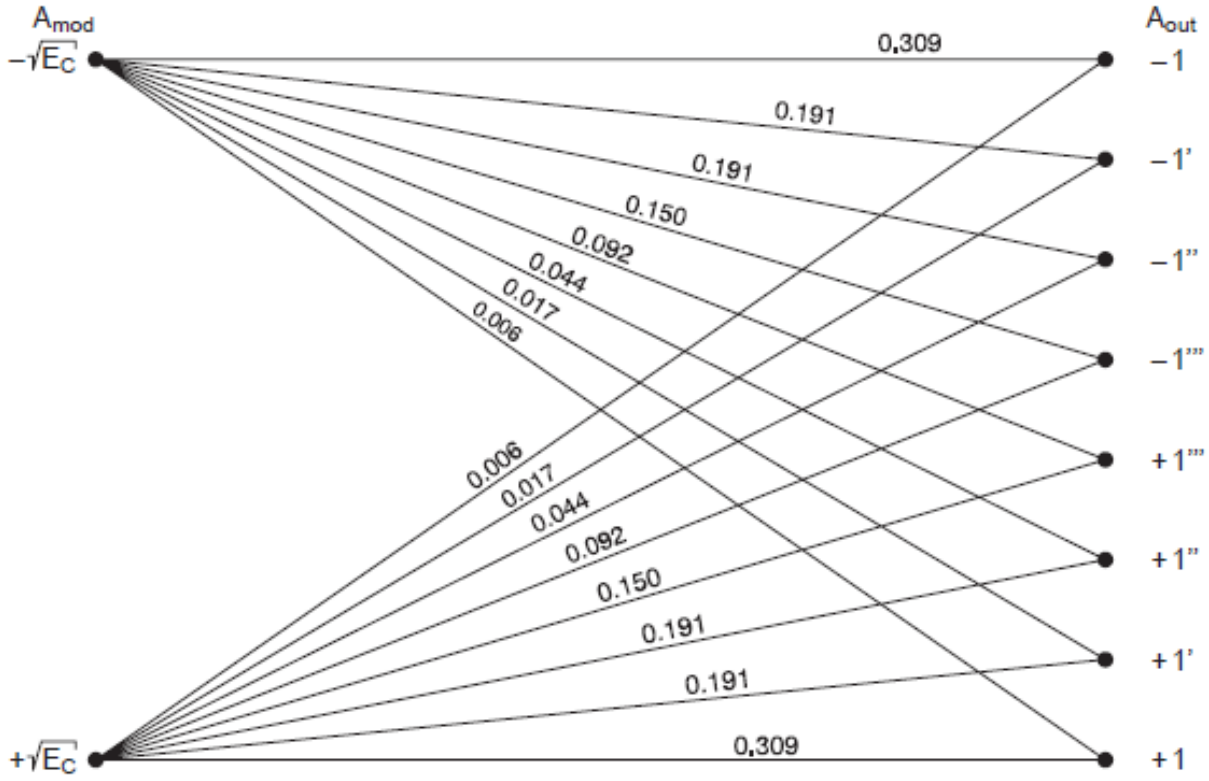


Bild 1.5. Übergangswahrscheinlichkeiten beim oktal quantisierten AWGN

Das Grundprinzip der Blockcodierung zeigt Bild 1.7: Der Datenstrom der Infosymbole bzw. Codesymbole wird unterteilt in Blöcke der Länge k bzw. n, die als *Infowörter* $\boldsymbol{u} = (u_0, \ldots, u_{k-1})$ bzw. *Codewörter* $\boldsymbol{a} = (a_0, \ldots, a_{n-1})$ bezeichnet werden. Dabei gilt $k < n$. Der Encoder ordnet jedem Infowort ein Codewort zu. Am Ausgang des diskreten Kanals entsteht das *Empfangswort* $\boldsymbol{y} = (y_0, \ldots, y_{n-1})$, aus dem der Decoder die Schätzung $\hat{\boldsymbol{u}} = (\hat{u}_0, \ldots, \hat{u}_{k-1})$ für das Infowort gewinnt.

Die Zuordnung der Codewörter zu den Infowörtern im Encoder ist (1) *eindeutig* und *umkehrbar*, indem zwei verschiedenen Infowörtern zwei verschiedene Codewörter zugeordnet werden, so daß zu jedem Codewort genau ein Infowort gehört; (2) *zeitinvariant*, indem die Zuordnungsvorschrift immer gleich bleibt; (3) *gedächtnislos*, indem jedes Infowort nur auf ein Codewort wirkt und jedes Codewort nur durch ein Infowort bestimmt wird.

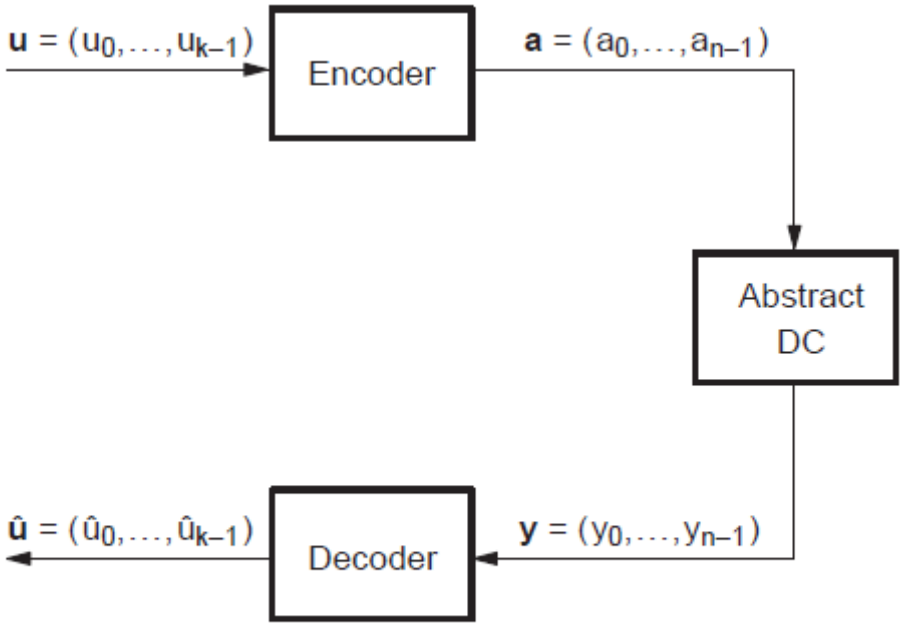


Bild 1.7. Prinzip des (n, k)-Blockcodes

Definition 1.4. *Durch die vorangehend beschriebene Methode wird ein* (n, k)-Blockcode *definiert, der in ausführlicherer Schreibweise auch als* $(n, k, d_{\min})_q$-*Blockcode bezeichnet wird, wobei* q *die Stufenzahl der Symbole* $u_i, a_i, \hat{u}_i$ *und* $d_{\min}$ *die Minimaldistanz (siehe Definition 1.8) bezeichnet. Als* Blocklänge *wird* n *bezeichnet und als* Coderate *wird das Verhältnis von* k *zu* n *bezeichnet:*

$$R = \frac{k}{n} < 1 \qquad \textit{Einheit: Infosymbol/Kanalbenutzung.} \qquad (1.4.1)$$

Als Code $\mathcal{C}$ *wird die Menge aller Codewörter bezeichnet.* Statt C wird auch die Bezeichnung Γ verwendet

Die Coderate $R \leq 1$ ist mit der eigentlich dimensionslosen Einheit Infosymbol/Codesymbol versehen. Da pro Codesymbol der diskrete Kanal genau einmal benutzt wird, ergibt sich die in (1.4.1) angegebene Einheit.

Die Datenraten werden immer auf Bit statt Symbol bezogen, d.h. die *Infobitrate* r_b hat die Einheit Infobit/s und die *Codebitrate* r_c hat die Einheit Codebit/s. Bei q-stufigen Symbolen entspricht ein Symbol genau $\log_2 q$ Bits, so daß $r_b/\log_2 q$ die Infosymbolrate und $r_c/\log_2 q$ die Codesymbolrate bzw. die Kanalbenutzungsrate ist. Pro Sekunde werden also $r_c/(n \cdot \log_2 q)$ Codeblöcke übertragen. Die Codierung bewirkt wegen

$$r_c \;=\; r_b \cdot \frac{1}{R} \;=\; r_b \cdot \frac{n}{k} \tag{1.4.2}$$

eine Erhöhung der Datenrate um den Faktor $1/R = n/k$, der deshalb zuweilen auch als *Bandbreitenexpansionsfaktor* bezeichnet wird. $R = 1$ bedeutet uncodierte Übertragung. Manchmal ist es erforderlich, die Coderate auf Bits statt auf Symbole zu beziehen:

$$R_b \;=\; R \cdot \log_2 q \;=\; \frac{k}{n} \cdot \log_2 q \qquad \text{Einheit: Infobit/Kanalben.} \tag{1.4.3}$$

Im binären Fall gilt natürlich $R_b = R$.

Die Anzahl der Infowörter der Länge k mit q-stufigen Symbolen beträgt offensichtlich q^k und somit gibt es auch $q^k = |\mathcal{C}| = q^{nR} = 2^{nR_b}$ Codewörter. Die Anzahl der möglichen Sendewörter der Länge n beträgt jedoch q^n. Der Code $\mathcal{C}$ ist also eine Untermenge der Mächtigkeit q^k in der Menge aller q^n Wörter.

Die *Güte eines Codes* wird ausschließlich dadurch geprägt, wie geschickt aus den q^n Wörtern die q^k Codewörter ausgewählt werden. Es wird darauf hinauslaufen, daß sich die Codewörter möglichst stark voneinander unterscheiden müssen.

Der Encoder trifft nur eine Zuordnung zwischen den q^k Infowörtern und den q^k Codewörtern. Wie diese Zuordnung über die Forderungen der Eindeutigkeit, Zeitinvarianz und Gedächtnislosigkeit hinaus organisiert ist, bleibt im Prinzip weitgehend belanglos. Insofern ist der Begriff *Güte eines Encoders* sinnlos. Allerdings werden in der Praxis fast ausschließlich systematische Encoder verwendet (siehe Definition 1.5) und zur Vereinfachung der Realisierung fast ausschließlich lineare Codes (siehe Kapitel 3) bzw. zyklische Codes (siehe Kapitel 5).

Definition 1.5. *Bei einem* systematischen Encoder *(auch: systematischer Code) erfolgt die Zuordnung zwischen Infowörtern und Codewörtern derart, daß das Infowort explizit Teil des Codewortes ist. Die restlichen $n-k$ Stellen heißen dann* Prüfstellen *(Prüfbits, parity bits).*

Beispiel 1.1. Die beiden Zuordnungen (Prüfstellen hinten bzw. vorn)

$00 \mapsto 000$	$00 \mapsto 000$
$01 \mapsto 011$	$01 \mapsto 101$
$10 \mapsto 101$	$10 \mapsto 110$
$11 \mapsto 110$	$11 \mapsto 011$

erzeugen den gleichen $(3,2)_2$ Code $\mathcal{C} = \{000, 011, 101, 110\}$. Der Codemenge kann nicht angesehen werden, in welcher Weise encodiert wurde. Beide Encoder sind als gleichwertig anzusehen. ■

Es seien $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$ jeweils Wörter der Länge n mit q-stufigen Werten (beispielsweise Codewörter oder Empfangswörter).

Definition 1.7. *Die Hammingdistanz $d_H(\boldsymbol{x}, \boldsymbol{y})$ ist definiert als die Anzahl der Abweichungen zwischen den Komponenten von $\boldsymbol{x}$ und $\boldsymbol{y}$. Sofern eine Null definiert ist, wird als Hamminggewicht $w_H(\boldsymbol{x})$ die Anzahl der Komponenten von $\boldsymbol{x}$ bezeichnet, die ungleich Null sind.*

Für den Zusammenhang zwischen Abstand und Gewicht gilt:

$$w_H(\boldsymbol{x}) = d_H(\boldsymbol{x}, \boldsymbol{0}) \quad \text{mit} \quad \boldsymbol{0} = (0, \ldots, 0). \tag{1.5.1}$$

Falls im Wertebereich der Symbole eine "Subtraktion" definiert ist, gilt weiter:

$$d_H(\boldsymbol{x}, \boldsymbol{y}) = w_H(\boldsymbol{x} - \boldsymbol{y}). \tag{1.5.2}$$

Satz 1.1. *Die Hammingdistanz ist eine* Metrik *im mathematischen Sinn, d.h. es gelten folgende Eigenschaften:*

$$d_H(\boldsymbol{x}, \boldsymbol{y}) = d_H(\boldsymbol{y}, \boldsymbol{x}) \tag{1.5.3}$$
$$0 \le d_H(\boldsymbol{x}, \boldsymbol{y}) \le n \tag{1.5.4}$$
$$d_H(\boldsymbol{x}, \boldsymbol{y}) = 0 \iff \boldsymbol{x} = \boldsymbol{y} \tag{1.5.5}$$
$$d_H(\boldsymbol{x}, \boldsymbol{y}) \le d_H(\boldsymbol{x}, \boldsymbol{z}) + d_H(\boldsymbol{z}, \boldsymbol{y}). \tag{1.5.6}$$

Falls Addition und Subtraktion im Wertebereich der Symbole so gelten für das Hamminggewicht folgende Eigenschaften:

$$w_H(\boldsymbol{x}) = w_H(-\boldsymbol{x}) \tag{1.5.7}$$
$$0 \le w_H(\boldsymbol{x}) \le n \tag{1.5.8}$$
$$w_H(\boldsymbol{x}) = 0 \iff \boldsymbol{x} = \boldsymbol{0} \tag{1.5.9}$$
$$w_H(\boldsymbol{x} + \boldsymbol{y}) \le w_H(\boldsymbol{x}) + w_H(\boldsymbol{y}). \tag{1.5.10}$$

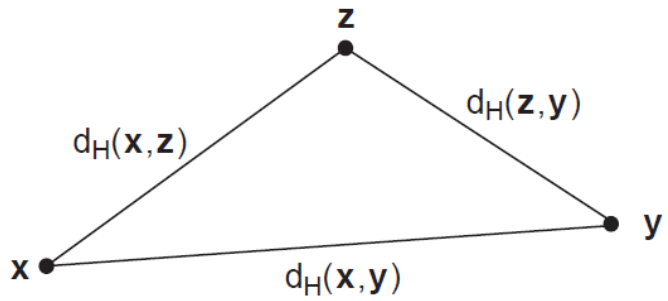


Bild 1.8. Veranschaulichung der Dreiecksungleichung für die Hammingdistanz

Definition 1.8. *Die Minimaldistanz $d_{\min}$ eines $(n, k, d_{\min})$-Blockcodes $\mathcal{C}$ ist definiert als die minimale Hammingdistanz zwischen allen Codewörtern:*

$$d_{\min} = \min\{d_H(\boldsymbol{a}, \boldsymbol{b}) \mid \boldsymbol{a}, \boldsymbol{b} \in \mathcal{C}\ ,\ \boldsymbol{a} \neq \boldsymbol{b}\}. \tag{1.5.11}$$

Die Minimaldistanz ist der wichtigste Parameter, der die Güte eines Codes bestimmt. Die vollständige Charakterisierung eines Codes wird später durch die Gewichtsverteilung gegeben (siehe Definition 3.7). Zur Bestimmung der Minimaldistanz müssen die Abstände aller Codewörterpaare betrachtet werden. Je größer die Minimaldistanz ist, je stärker sich also die Codewörter voneinander unterscheiden, desto besser ist der Code. Bei gegebener Coderate $R = k/n$ und gegebener Blocklänge n sollte derjenige Code gewählt werden, der zu großem $d_{\min}$ führt. Normalerweise wird $d_{\min}$ größer (d.h. besserer Code), wenn die Coderate kleiner wird (d.h. mehr Bandbreite erforderlich) oder wenn die Blocklänge größer wird (d.h. komplexere Verarbeitung erforderlich).

Beispiel 1.2. Der $(7,4)_2$-Hamming-Code besteht aus 16 Codewörtern der Länge 7:

$$\begin{aligned}\mathcal{C} = \{\ & 0000000,\ 1000011,\\ & 0001111,\ 1001100,\\ & 0010110,\ 1010101,\\ & 0011001,\ 1011010,\\ & 0100101,\ 1100110,\\ & 0101010,\ 1101001,\\ & 0110011,\ 1110000,\\ & 0111100,\ 1111111\ \}.\end{aligned}$$

Der Code ist hier durch eine Aufzählung der Codewörter gegeben und nicht durch die (unwesentliche) Art der Encodiervorschrift (systematisch, Infobits vorn). Der Vergleich der ersten beiden Codewörter ergibt $d_{\min} \leq 3$. Bei der Betrachtung aller Paare findet sich kein Paar mit der Hammingdistanz 2, so daß $d_{\min} = 3$ folgt. ■

Klar ist schon jetzt, daß es besserer Methoden zur Beschreibung der Codemenge und zur Berechnung der Minimaldistanz bedarf – dazu wird später eine algebraische Struktur auf der Codemenge eingeführt.

Die optimale Decodiervorschrift ist dadurch definiert, daß die Wort-Fehlerwahrscheinlichkeit $P_w = P(\hat{\boldsymbol{u}} \neq \boldsymbol{u})$ nach dem Decoder minimal wird:

Unterstellt wird also ein stochastischer Kanal (beispielsweise ein DMC), der zu Fehlern im Empfangswort führt, die möglicherweise durch den Decoder nicht korrigiert werden können und damit zu Fehlern im geschätzten Infowort führen. Derartige Fehler sollen während einer Übertragung von vielen Worten möglichst selten auftreten. Nicht berücksichtigt wird dabei, ob in einem falsch geschätzten Infowort nur ein Fehler oder mehrere Fehler enthalten sind. Eine solche Minimierung der Bit-Fehlerwahrscheinlichkeit $P_b = P(\hat{u}_i \neq u_i)$ ist wesentlich schwieriger.

Ziel ist also, daß das geschätzte Infowort möglichst oft exakt mit dem Infowort auf der Sendeseite übereinstimmt. Diese Forderung ist das Kriterium, nach dem der Decoder konstruiert werden soll. Es wird sich gleich zeigen, daß man diese Konstruktionsvorschrift für den Decoder ableiten kann, auch wenn das Kriterium P_w gar nicht explizit berechnet werden kann:

$$P_w = P(\hat{\boldsymbol{u}} \neq \boldsymbol{u}) \quad \longrightarrow \quad \text{Minimum} = P(\hat{\boldsymbol{a}} \neq \boldsymbol{a}). \tag{1.6.1}$$

Wegen der eindeutigen Zuordnung zwischen Infowörtern und Codewörtern kann anstelle des Infowortes auch das Codewort geschätzt werden.

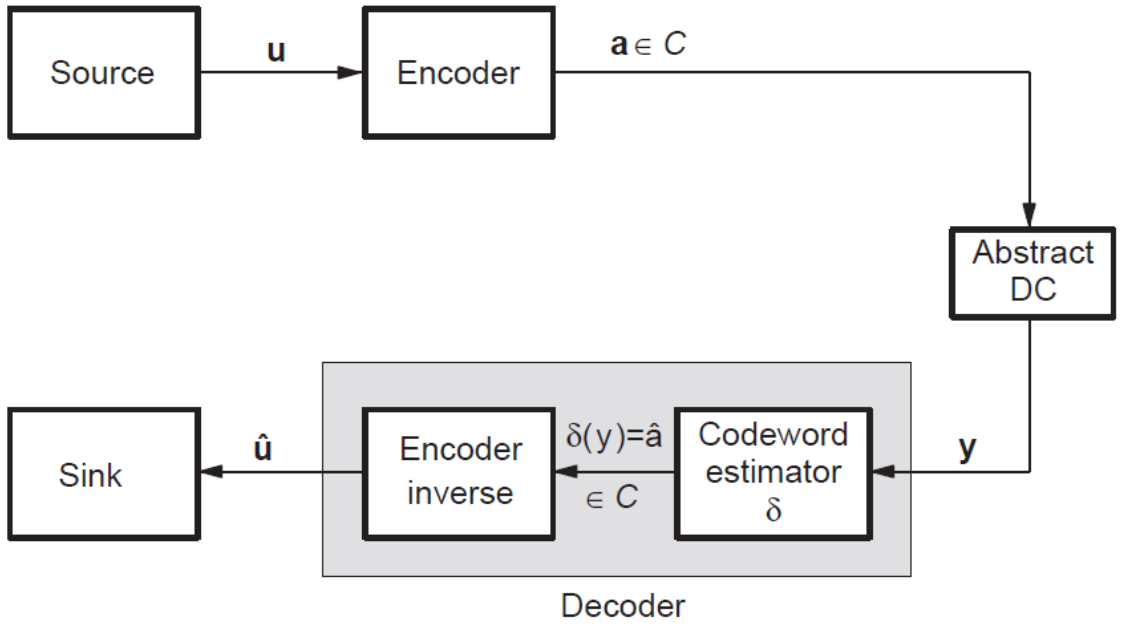


Bild 1.9. Zur Herleitung der Decodiervorschrift

Bild 1.9 zeigt das Prinzip zur Herleitung der Decodiervorschrift. Der Decoder wird wie angegeben zerlegt in einen Codewortschätzer (hier mit δ bezeichnet) und ein Encoder-Inverses. Dieses Encoder-Inverse ist eine direkte Umkehrung des Encoders und hat zu den geschätzten Codewörtern $\hat{\boldsymbol{a}}$ lediglich das zugehörige Infowort $\hat{\boldsymbol{u}}$ zu bestimmen. Das ist eine triviale Operation, beispielsweise sind bei systematischen Encodern lediglich die Prüfstellen auszublenden.

Die gesamte Intelligenz des Decoders steckt im Codewortschätzer, der im Gegensatz zum Encoder-Inversen nicht die Encodiervorschrift, sondern nur die Codemenge kennen muß. Zum Empfangswort $\boldsymbol{y}$ wird also im Codewortschätzer das geschätzte Codewort bestimmt – formal ist das eine Abbildung wie folgt:

$$\delta \;:\; \boldsymbol{y} \mapsto \delta(\boldsymbol{y}) = \hat{\boldsymbol{a}} \in \mathcal{C}. \tag{1.6.2}$$

Die Funktion δ ist so zu konstruieren, daß die Wort-Fehlerwahrscheinlichkeit minimal wird:

$$P_w = P(\delta(\boldsymbol{y}) \neq \boldsymbol{a}) \tag{1.6.3}$$

Bei der Übertragung über einen diskreten Kanal mit Hard-Decision (also mit $\mathcal{A}_{\text{in}} = \mathcal{A}_{\text{out}}$) sind folgende Fälle zu unterscheiden:

$\boldsymbol{y} = \boldsymbol{a}$	Fehlerfreie Übertragung.
$\boldsymbol{y} \in \mathcal{C} \setminus \{\boldsymbol{a}\}$	Verfälschung in ein anderes Codewort – dieser Fall kann niemals erkannt oder korrigiert werden.
$\boldsymbol{y} \notin \mathcal{C}$	Die Verfälschung ist generell erkennbar und eventuell korrigierbar durch den Decoder. Bei $\delta(\boldsymbol{y}) = \boldsymbol{a}$ wird korrekt decodiert und bei $\delta(\boldsymbol{y}) \neq \boldsymbol{a}$ wird falsch decodiert. Der Fall $\delta(\boldsymbol{y}) =$ undefiniert tritt zwar beim idealen Decoder nicht auf, aber bei praktisch realisierten Decodern ist dieser Fall jedoch durchaus sinnvoll (siehe nachfolgende Erklärung).

In der formalen Beschreibung ordnet δ jedem der q^n möglichen Empfangsworte eines der q^k Codewörter zu. Später wird sich noch zeigen, daß man bei der Realisierung des Decoders teilweise darauf verzichtet, jedem möglichen Empfangswort die optimale Schätzung zuzuordnen – stattdessen wird die optimale Decodiervorschrift nur für die häufiger vorkommenden Empfangswörter realisiert. Mit dieser Methode können sich für den Decoder ganz erhebliche Vereinfachungen bei der Realisierung ergeben.

Voraussetzung (gleiche Apriori-Wahrscheinlichkeiten) *zur Herleitung der Decodiervorschrift: Alle q^k Infowörter sollen mit der gleichen Wahrscheinlichkeit q^{-k} von der Quelle abgegeben werden.*

Mit dieser Voraussetzung treten auch alle Codewörter mit der gleichen Wahrscheinlichkeit q^{-k} auf. Die Wahrscheinlichkeit, daß ein Fehler bei der Decodierung auftritt unter der Voraussetzung, daß das Codewort $\boldsymbol{a}$ gesendet wurde, ergibt sich aus der Summation über diejenigen Empfangswörter, die zu einer Schätzung ungleich $\boldsymbol{a}$ führen:

$$\begin{aligned} P(\delta(\boldsymbol{y}) \neq \boldsymbol{a} \mid \boldsymbol{a} \text{ gesendet}) &= \sum_{\substack{\boldsymbol{y} \\ \delta(\boldsymbol{y}) \neq \boldsymbol{a}}} P(\boldsymbol{y} \text{ empfangen} \mid \boldsymbol{a} \text{ gesendet}) \\ &= \sum_{\substack{\boldsymbol{y} \\ \delta(\boldsymbol{y}) \neq \boldsymbol{a}}} P_{y|x}(\boldsymbol{y}|\boldsymbol{a}). \end{aligned} \tag{1.6.4}$$

Ferner gilt bei der Summation über alle Codewörter und alle Empfangswörter:

$$\begin{aligned} \sum_{\boldsymbol{a} \in \mathcal{C}, \boldsymbol{y}} P_{y|x}(\boldsymbol{y}|\boldsymbol{a}) &= \sum_{\boldsymbol{a} \in \mathcal{C}} P_{y|x}(\text{arbitrary } \boldsymbol{y} \text{ empfangen } |\boldsymbol{a}) \\ &= \sum_{\boldsymbol{a} \in \mathcal{C}} 1 = q^k. \end{aligned} \tag{1.6.5}$$

Aus dem Satz von der vollständigen Wahrscheinlichkeit (A.3.1) folgt nun:

$$\begin{aligned} P_w &= P(\delta(\boldsymbol{y}) \neq \boldsymbol{a}) \\ &= \sum_{\boldsymbol{a} \in \mathcal{C}} P(\delta(\boldsymbol{y}) \neq \boldsymbol{a} \mid \boldsymbol{a} \text{ gesendet}) \cdot P(\boldsymbol{a} \text{ gesendet}) \\ &= \sum_{\boldsymbol{a} \in \mathcal{C}} \sum_{\substack{\boldsymbol{y} \\ \delta(\boldsymbol{y}) \neq \boldsymbol{a}}} P_{y|x}(\boldsymbol{y}|\boldsymbol{a}) \cdot q^{-k} \qquad \text{with (1.6.4)} \\ &= q^{-k} \left(\sum_{\boldsymbol{a} \in \mathcal{C}, \boldsymbol{y}} P_{y|x}(\boldsymbol{y}|\boldsymbol{a}) - \sum_{\substack{\boldsymbol{a} \in \mathcal{C}, \boldsymbol{y} \\ \delta(\boldsymbol{y}) = \boldsymbol{a}}} P_{y|x}(\boldsymbol{y}|\boldsymbol{a}) \right) \\ &= 1 - q^{-k} \cdot \sum_{\substack{\boldsymbol{a} \in \mathcal{C}, \boldsymbol{y} \\ \delta(\boldsymbol{y}) = \boldsymbol{a}}} P_{y|x}(\boldsymbol{y}|\boldsymbol{a}) \qquad \text{with (1.6.5)} \\ &= 1 - q^{-k} \cdot \sum_{\boldsymbol{y}} P_{y|x}(\boldsymbol{y}|\delta(\boldsymbol{y})). \end{aligned}$$

Zur Minimierung von P_w sollte für jedes Empfangswort $\boldsymbol{y}$ also $\delta(\boldsymbol{y}) = \hat{\boldsymbol{a}}$ so gewählt werden, daß die Übergangswahrscheinlichkeit $P_{y|x}(\boldsymbol{y}|\hat{\boldsymbol{a}})$ maximal wird.

Satz 1.2 (Maximum-Likelihood-Decoder MLD). *Die Wort-Fehlerwahrscheinlichkeit P_w nach der Decodierung wird minimal, wenn wie folgt decodiert wird: Zum Empfangswort $\boldsymbol{y}$ wird als Schätzung $\hat{\boldsymbol{a}} \in \mathcal{C}$ dasjenige Codewort gewählt, bei dem die Übergangswahrscheinlichkeit maximal wird:*

$$P_{y|x}(\boldsymbol{y}|\hat{\boldsymbol{a}}) \geq P_{y|x}(\boldsymbol{y}|\boldsymbol{b}) \quad \text{für alle } \boldsymbol{b} \in \mathcal{C}. \tag{1.6.6}$$

Die ML-Decodierung kann auch mehrdeutig sein – in diesem Fall wird dann irgendein Codewort mit maximaler Übergangswahrscheinlichkeit gewählt.

Dieses Ergebnis ist auch anschaulich einfach zu verstehen: Bei gegebenem Empfangswort wird dasjenige Sendewort bzw. Codewort gesucht, das am wahrscheinlichsten gesendet wurde. Die Berechnung der Wort-Fehlerwahrscheinlichkeit selbst erfolgt später in Kapitel 3.

Noch anschaulicher werden diese Ergebnisse, wenn der q-näre symmetrische DMC betrachtet wird. Aus (1.3.2) und (1.3.3) folgt dann für $\boldsymbol{y} = (y_0, \ldots, y_{n-1})$ und $\hat{\boldsymbol{a}} = (\hat{a}_0, \ldots, \hat{a}_{n-1})$ mit $d = d_H(\boldsymbol{y}, \hat{\boldsymbol{a}})$:

$$\begin{aligned} P_{y|x}(\boldsymbol{y}|\hat{\boldsymbol{a}}) &= \prod_{i=0}^{n-1} P_{y|x}(y_i|\hat{a}_i) \\ &= \prod_{i=0}^{n-1} \left\{ \begin{array}{ll} 1-p_e & \text{if } y_i = \hat{a}_i \\ p_e/(q-1) & \text{if } y_i \neq \hat{a}_i \end{array} \right\} \\ &= (1-p_e)^{n-d} \cdot \left(\frac{p_e}{q-1}\right)^d \\ &= (1-p_e)^n \cdot \left(\frac{p_e}{(1-p_e)(q-1)}\right)^d . \end{aligned} \tag{1.6.7}$$

Der linke Faktor ist unabhängig von $\hat{\boldsymbol{a}}$ und somit muß nur der rechte Faktor durch geeignete Wahl von $\hat{\boldsymbol{a}}$ maximiert werden. Für $p_e < 0{,}5$ ist der Quotient kleiner als 1 und somit ergibt sich das Maximum, wenn $d = d_H(\boldsymbol{y}, \hat{\boldsymbol{a}})$ minimal wird:

Satz 1.3 (MLD für Hard-Decision-DMC). *Die Wort-Fehlerwahrscheinlichkeit P_w nach der Decodierung wird minimal, wenn wie folgt decodiert wird: Zum Empfangswort $\boldsymbol{y}$ wird als Schätzung $\hat{\boldsymbol{a}} \in \mathcal{C}$ dasjenige Codewort gewählt, das vom Empfangswort den minimalen Hammingabstand hat:*

$$d_H(\boldsymbol{y}, \hat{\boldsymbol{a}}) \leq d_H(\boldsymbol{y}, \boldsymbol{b}) \quad \textit{für alle } \boldsymbol{b} \in \mathcal{C}. \tag{1.6.8}$$

Beispiel 1.3. Es wird der $(5,2)_2$-Code $\mathcal{C} = \{\underbrace{00000}_{\boldsymbol{a}_1}, \underbrace{11100}_{\boldsymbol{a}_2}, \underbrace{00111}_{\boldsymbol{a}_3}, \underbrace{11011}_{\boldsymbol{a}_4}\}$ betrachtet, für den offensichtlich $d_{\min} = 3$ gilt. In der nachfolgenden Tabelle sind für drei als beispielhaft gewählte Empfangswörter die Abstände zu allen Codewörtern angegeben und die daraus resultierende Entscheidung des Codewortschätzers:

$\boldsymbol{y}$	$d_H(\boldsymbol{y}, \boldsymbol{a}_1)$	$d_H(\boldsymbol{y}, \boldsymbol{a}_2)$	$d_H(\boldsymbol{y}, \boldsymbol{a}_3)$	$d_H(\boldsymbol{y}, \boldsymbol{a}_4)$	$\delta(\boldsymbol{y})$
10000	1	2	4	3	$\boldsymbol{a}_1$
11000	2	1	5	2	$\boldsymbol{a}_2$
10001	2	3	3	2	$\boldsymbol{a}_1$ or $\boldsymbol{a}_4$

■

Schon beim $(7,4)$-Hamming-Code aus Beispiel 1.2 wird diese Methode ziemlich umständlich, so daß praktisch anwendbare Codes zwei Forderungen genügen sollten: (1) Die Minimaldistanz $d_{\min}$ soll möglichst groß sein. (2) Die Struktur der Codemenge $\mathcal{C}$ muß so sein, daß sich im Decoder die Suche nach dem minimalen Hammingabstand möglichst einfach organisieren läßt. Beide Forderungen setzen eine algebraische Struktur voraus, denn für (1) ist die Struktur notwendig, um überhaupt gute Codes konstruieren zu können und für (2), um realisierbare Decoder zu ermöglichen.

Satz 1.3 soll nun in entsprechender Form auch für den AWGN abgeleitet werden. Nach (1.3.11) gilt für die Verteilungsdichten:

$$\begin{aligned} f_{y|x}(\boldsymbol{y}|\hat{\boldsymbol{a}}) &= \prod_{i=0}^{n-1} f_{y|x}(y_i|\hat{a}_i) \\ &= \prod_{i=0}^{n-1} \frac{1}{\sqrt{\pi N_0}} \exp\left(-\frac{(y_i-\hat{a}_i)^2}{N_0}\right) \\ &= (\pi N_0)^{-n/2} \exp\left(-\frac{1}{N_0}\sum_{i=0}^{n-1}(y_i-\hat{a}_i)^2\right) \\ &= c \cdot \exp\left(-\frac{1}{N_0}\|\boldsymbol{y}-\hat{\boldsymbol{a}}\|^2\right), \end{aligned} \tag{1.6.9}$$

Dabei ist c eine Konstante und $\|\boldsymbol{y}-\hat{\boldsymbol{a}}\|^2$ die *euklidische Norm* von $\boldsymbol{y}-\hat{\boldsymbol{a}}$ bzw. der *euklidische Abstand* zwischen $\boldsymbol{y}$ und $\hat{\boldsymbol{a}}$. Der rechte Faktor muß durch geeignete Wahl von $\hat{\boldsymbol{a}}$ maximiert werden. Dies wird erreicht durch Minimierung der Norm und somit folgt:

Satz 1.4 (MLD für Soft-Decision-AWGN). *Die Wort-Fehlerwahrscheinlichkeit P_w nach der Decodierung wird minimal, wenn wie folgt decodiert wird: Zum Empfangswort $\boldsymbol{y}$ wird als Schätzung $\hat{\boldsymbol{a}} \in \mathcal{C}$ dasjenige Codewort gewählt, das vom Empfangswort den minimalen euklidischen Abstand hat:*

$$\|\boldsymbol{y}-\hat{\boldsymbol{a}}\| \le \|\boldsymbol{y}-\boldsymbol{b}\| \quad \textit{für alle } \boldsymbol{b} \in \mathcal{C}. \tag{1.6.10}$$

Beispiel $(3,1,3)_2$ – Wiederholungscode

$$\Gamma = \{ \underbrace{(+1,+1,+1)}_{=a_1}, \underbrace{(-1,-1,-1)}_{=a_2} \}$$

Betrachte das Empfangswort $y = (\varepsilon, -1, \varepsilon)$, $\varepsilon > 0$, $\varepsilon \approx 0$

Soft-Decision Entscheider

$$\|y - a_1\| = \sqrt{(1-\varepsilon)^2 + 2^2 + (1-\varepsilon)^2} \approx \sqrt{6}$$
$$\|y - a_2\| = \sqrt{(1+\varepsilon)^2 + 0^2 + (1+\varepsilon)^2} \approx \sqrt{2}$$

$\Rightarrow \hat{a} = a_2$

vernünftig

Hard-Decision Entscheider

$y_a = \mathrm{sign}(y) = (+1, -1, +1)$

$d_H(y_a, a_1) = 1$

$d_H(y_a, a_2) = 2$

$\Rightarrow \hat{a} = a_1$

unsinnig, Quantisierung vernichtet wertvolle Information