Kommunikationstechnik II – Wintersemester 08/09

Prof. Dr. Stefan Weinzierl

Musterlösung: 7. Aufgabenblatt

Lösung in der Rechenübung am 22.1.2009

1. Aufgabe: Entropiekodierung

Der Informationsgehalt H_i ist ein Maß für die Unbestimmtheit eines Ereignisses. Je kleiner die Wahrscheinlichkeit eines Symbols (Audio: Amplitude), desto größer ist H_i. Die Entropie H_m ist der mittlere Informationsgehalt einer Quelle (Audio: Signal) und gibt die minimale Anzahl Bits an, mit der die Quelle eindeutig kodiert werden kann. Sie ist maximal, wenn die Auftretenswahrscheinlichkeiten aller Amplituden gleich sind (z.B. weißes Rauschen) und minimal (= 0), wenn es nur eine Amplitude (Gleichspannung) mit der Wahrscheinlichkeit 1 gibt.
Bei der Entropiekodierung werden die Kodewortlängen in Abhängigkeit der Auftretenswahrscheinlichkeit der Quellenzeichen variiert. Die mittlere Kodewortlänge sollte dabei im Idealfall der Entropie entsprechen, sodass die Redundanz minimal ist.

Lesen Sie das Audiofile „test.wav" in Matlab ein.

a) Bei der Requantisierung der Amplituden auf 8 Quantisierungsstufen entsprechend einer Wortbreite von w = 3 bit: Wie ist die Häufigkeitsverteilung der 2^w Kodewörter im Quellkode?

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Musterloesung zum 7. Aufgabenblatt                                      %
%                                                                         %
%-------------------------------------------------------------------------%
% Aufgabe 1: Entropiekodierung

%%%%%%%% a) %%%%%%%%
y = wavread('test');

% Die Haeufigkeitsverteilung der Kodeworte nach einer Quantisierung
% in 8 Stufen kann für jeden Kanal einzeln mit hist bestimmt werden:
steps = 8;

% Zaehle je Kanal die Haeufigkeit der Amplituden je Quantisierungsstufe
hf = hist(y,steps);
hist(y,steps)

% Mittle Anzahl ueber beide Kanaele:
hf_mittel = mean(hf,2);
% Kodewortwahrscheinlichkeiten p_i
p_i = hf_mittel/sum(hf_mittel)

% Kontrolle: (Summe der p_i muss 1 ergeben)
W_ges = sum(p_i)
```

Die 2^3 = 8 Kodeworte und ihre gemessenen Auftretenswahrscheinlichkeiten:

Quellezeichen	x_{-4}	x_{-3}	x_{-2}	x_{-1}	x_1	x_2	x_3	x_4
p_i	0, 004	0, 016	0, 061	0, 418	0, 423	0, 061	0, 014	0, 004

b) Wie groß ist die Quellenentropie in bit/Quellenzeichen?

Mit: $H_i = -\mathrm{ld}(p_i)$

$$H_\mathrm{m} = \sum_{i=1}^{N} p_i \cdot H_i \qquad H_\mathrm{m} = 1{,}78$$

```
%%%%%%%% b) %%%%%%%%

% Bestimme den Informationsgehalt je Zeichen H_i
H_i = -log2(p_i);

% danach die Entropie H_m:
H_m = sum(p_i.*H_i)
```

c) Konstruieren Sie für diese Quelle Optimalkode nach dem Huffman-Verfahren.

Eigenschaften von Huffman-Kodes:
- Je größer die Auftretenswahrscheinlichkeit eines Kodeworts, desto kürzer ist seine Länge l_i.
- Präfixeigenschaft: Es beginnt kein Kodewort mit der Bitfolge eines anderen, kürzeren Kodewortes.
- minimale Redundanz, nur dann optimal (= 0), wenn $p_i = 1/2^n$

$$0{,}423 \quad 0{,}418 \quad 0{,}061 \quad 0{,}061 \quad 0{,}016 \quad 0{,}014 \quad \underbrace{0{,}004 \quad 0{,}004}_{0{,}008}$$

$$0{,}423 \quad 0{,}418 \quad 0{,}061 \quad 0{,}061 \quad 0{,}016 \quad \underbrace{0{,}014 \quad 0{,}008}_{0{,}022}$$

$$0{,}423 \quad 0{,}418 \quad 0{,}061 \quad 0{,}061 \quad \underbrace{0{,}022 \quad 0{,}016}_{0{,}038}$$

$$0{,}423 \quad 0{,}418 \quad 0{,}061 \quad \underbrace{0{,}061 \quad 0{,}038}_{0{,}099}$$

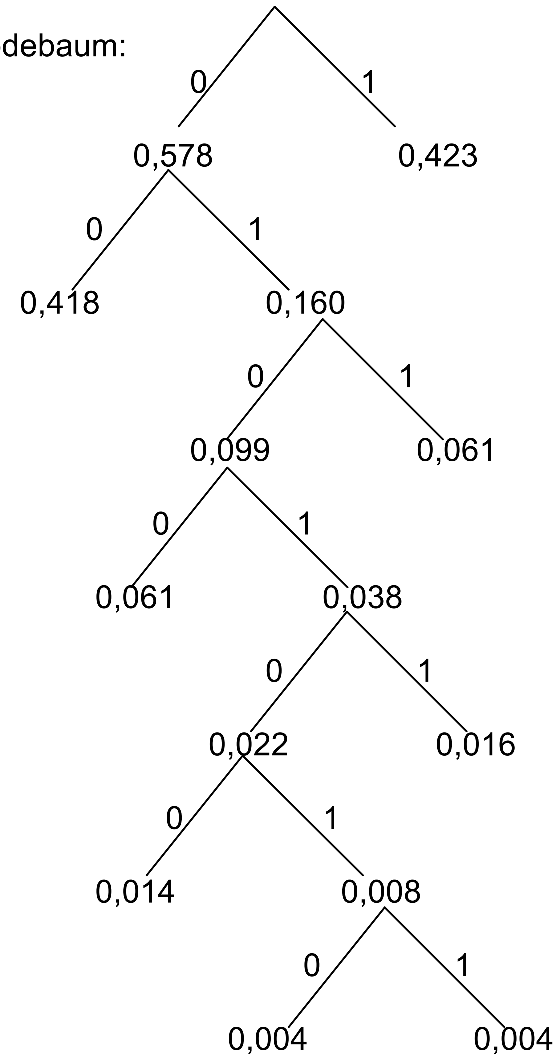
$$0{,}423 \quad 0{,}418 \quad \underbrace{0{,}099 \quad 0{,}061}_{0{,}160}$$

$$0{,}423 \quad \underbrace{0{,}418 \quad 0{,}160}_{0{,}578}$$

$$\underbrace{0{,}578 \quad 0{,}423}_{1}$$

1

Kodebaum:



Quellezeichen	x_{-4}	x_{-3}	x_{-2}	x_{-1}	x_1	x_2	x_3	x_4
p_i	0, 004	0, 016	0, 061	0, 418	0, 423	0, 061	0, 014	0, 004
Kodewort	0101010	01011	0100	00	1	011	010100	0101011

Die Tabelle zeigt eine Möglichkeit eines Huffman-Kodes.

d) Welche mittlere Kodewortlänge ergibt sich aus dem erstellten Kode? Vergleichen Sie die Koderedundanz des Huffman-Kodes mit der eines gleichmäßigen 3-bit-Kodes.

$$l_{\mathrm{m}} = \sum_{i=1}^{N} p_i \cdot l_i \qquad l_{\mathrm{m}} = 1{,}90 \text{ Bit/Zeichen}$$

```
%%%%%%%% d) %%%%%%%%

% mittele Kodewortlänge des Huffman-Kodes:
l_i = [7 5 4 2 1 3 6 7]';

l_m = sum(p_i.*l_i)
```

Koderedundanz $R_{K\_Huffman}$ des Huffman-Kodes:

$R_{K\_Hufmann} = l_m - H_m = 1{,}90 - 1{,}78 = \underline{\underline{0{,}12}}$ Bit/Zeichen
Koderedundanz R_{K_gl} des gleichmäßigen Kodes:

$R_{K\_gl} = l - H_m = 3 - 1{,}78 = \underline{\underline{1{,}22}}$ Bit/Zeichen

e) Wie groß ist der auf diese Weise erzielte „Kompressionsfaktor“, d.h. um welchen Faktor ist die mittlere Kodewortlänge des Huffman-Kodes geringer als die des ursprünglichen Quellkodes?

$$c = \frac{l}{l_m} = \frac{3}{1{,}90} = 1{,}58$$

2. Aufgabe: Kanalkodierung

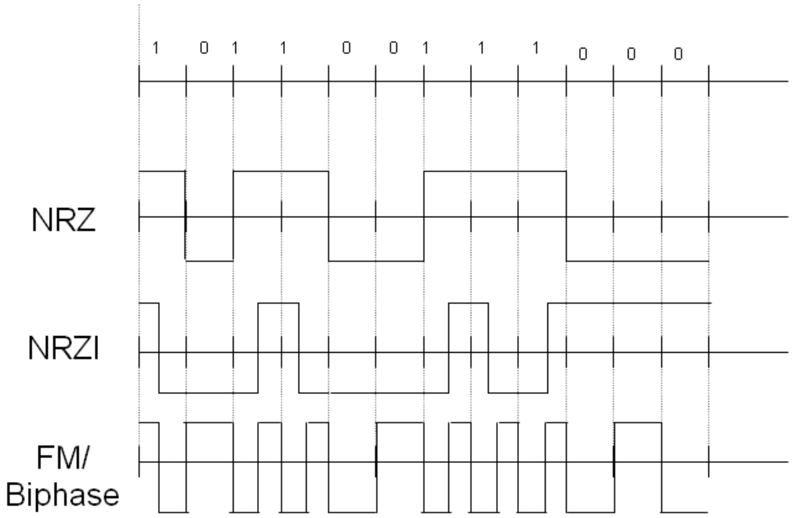
Zweck der Kanalkodierung:

- Abbildung der information bits des Quellkodes auf die channel bits des Kanalkodes. Bei einfachen Kodes wird jedes information bit auf ein channel bit abgebildet, bei Gruppenkodes werden mit Hilfe von Kodetabellen *m* information bits auf *n* channel bits abgebildet.
- anpassen der digitalen Zeichenfolge an realen Übertragungskanal
- Eigenschaften wie Selbsttaktung, Bandbreite, Jitterempfindlichkeit, Gleichspannungsanteil, Fehlererkennung etc. durch Pulsformung und spezielle Kanalkodes beeinflussbar

Gegeben sei eine Bitfolge (information bits) von 101100111000

a) Skizzieren Sie den Spannungsverlauf dieses Signals
i) in NRZ-Kodierung, ii) in NRZI-Kodierung, iii) im Biphase Mark-Kode

i) NRZ (non return to zero): bildet 1 auf hohes, 0 auf niedriges Potential ab
ii) NRZI (NRZ-inverted): Potentialwechsel in der Mitte einer Bitperiode für 1 (in jede Richtung), kein Potentialwechsel für 0
iii) Biphase Mark/FM (Frequency Modulation): Potentialwechsel am Anfang und in der Mitte einer Bitperiode für 1, Potentialwechsel am Anfang einer Bitperiode für 0



b) Konstruieren Sie eine eigene Kodetabelle für einen 3/5-Gruppenkode mit einer (0,2) RLL Lauflängenkodierung (=min/max Anzahl der 0en zwischen zwei 1en).

Der zu erstellende Kode muss Datenworte von m = 3 bit Länge auf speziell ausgewählte Kanalkodeworte der Länge n = 5 abbilden.
Die 2^3 = 8 möglichen Datenworte umfassen das binäre Intervall von [000;111]. Die (0,2)-RLL Kodierung verlangt eine minimale Anzahl von d = 0 und eine maximale Anzahl von k = 2 Nullen zwischen zwei aufeinander folgenden „1“-Symbolen (ist auch bei aufeinander folgenden Kodeworten einzuhalten).

Unter Berücksichtigung dieser Bedingungen lassen sich den Datenworten entsprechende Kanalkodeworte zuweisen. Die Tabelle zeigt für alle Datenworte die ihnen zuzuweisenden Kanalkodeworte. Dabei handelt es sich um eine rein willkürliche (!) Auswahl und Zuweisung. Eine Optimierung unter dem Gesichtspunkt höherer Fehlersicherheit erhielte man z.B. für Kanalkodeworte, deren minimaler Kodewortabstand d_{min} (Hammingabstand = Anzahl der unterschiedlichen Bitstellen je Kodewort, siehe Skript) zueinander maximiert wurde.

Datenwort	Kanalkodewort
000	01110
001	01101
010	01011
011	10011
100	10101
101	10110
110	11010
111	01111

c) Vergleichen Sie die anhand der figure of merit (FoM) die Leistungsfähigkeit des in b) konstruierten, NRZI-kodierten Gruppenkodes mit einer einfachen Biphase Mark-Kodierung.

$FoM_{\mathrm{RLL}}=T_{\mathrm{W}}\cdot DR=(m/n)^2\cdot(1+d) = (3/5)^2\cdot(1+0) = 0{,}36$
$FoM_{\mathrm{Biphase}}= T_{\mathrm{W}}\cdot DR = 0{,}5\cdot 0{,}5 = 0{,}25$ (aus Tabelle, S. 47 Skript)