

# RS - Übung 5

Arne Beer (MN 6489196),  
Rafael Epplee (MN 6269560),  
Julian Polatynski (MN 6424884)

November 22, 2012

## 5.1

Zunächst wird jeweils der vordere Teil von b hinzugefügt, indem das entsprechende a zu b addiert wird und man b so weit logisch nach links verschiebt, dass nur noch die relevanten bits von a im 8-bit-Bereich von b sind.

Dann werden nur die jeweils relevanten vorderen bits des nächsten a durch logisches rechts-shiften isoliert und zu b addiert.

```
int b1 = a1
b1 << 2
b1 += (a2 >>>4)
```

```
int b2 = a2
b2 << 4
b2 += (a3 >>>2)
```

```
int b3 = a3
b3 << 6
b3 += a4
```

Wichtig zu beachten ist hierbei, dass auch über den 8-bit-Bereich hinaus noch bits gesetzt sind. Diese könnte man (falls gefordert) mit dem folgenden Code auf null setzen (wobei bn eine beliebige Ergebnisvariable ist):

```
bn = bn & (~0 >>> 26)
```

## 5.2

a)

Der Code hat 24 Codewörter. Dies lässt sich errechnen, indem man die möglichen  $360^\circ$  durch die gewünschte Auflösung teilt:  $\frac{360}{15} = 24$ .

**b)**

Wir beginnen mit zwei Wörtern: 1 und 0. Der Code  $A_1$  mit der Bitlänge 1 ist:

$$\{0, 1\}$$

Dann setzen wir vor jedes Wort von  $A_1$  eine 0, drehen die Wörter von  $A_1$  um und setzen eine 1 davor. Es entsteht der neue Code  $A_2$  mit der Bitlänge 2:

$$\{00, 01, 11, 10\}$$

Dann wenden wir das gleiche Verfahren auf  $A_2$  an. Es entsteht  $A_3$  mit drei Bit länge.

$$\{000, 001, 011, 010, 110, 111, 101, 100\}$$

Gleiches Verfahren wieder,  $A_4$  hat 16 Wörter:

$$\{0000, 0001, 0011, 0010, 0110, 0111, 0101, 0100, 1100, 1101, 1111, 1110, 1010, 1011, 1001, 1000\}$$

$A_5$  hat 32 Wörter.

$$\begin{aligned} &\{00000, 00001, 00011, 00010, 00110, 00111, 00101, 00100, \\ &01100, 01101, 01111, 01110, 01010, 01011, 01001, 01000, \\ &11000, 11001, 11011, 11010, 11110, 11111, 11101, 11100, \\ &10100, 10101, 10111, 10110, 10010, 10011, 10001, 10000\} \end{aligned}$$

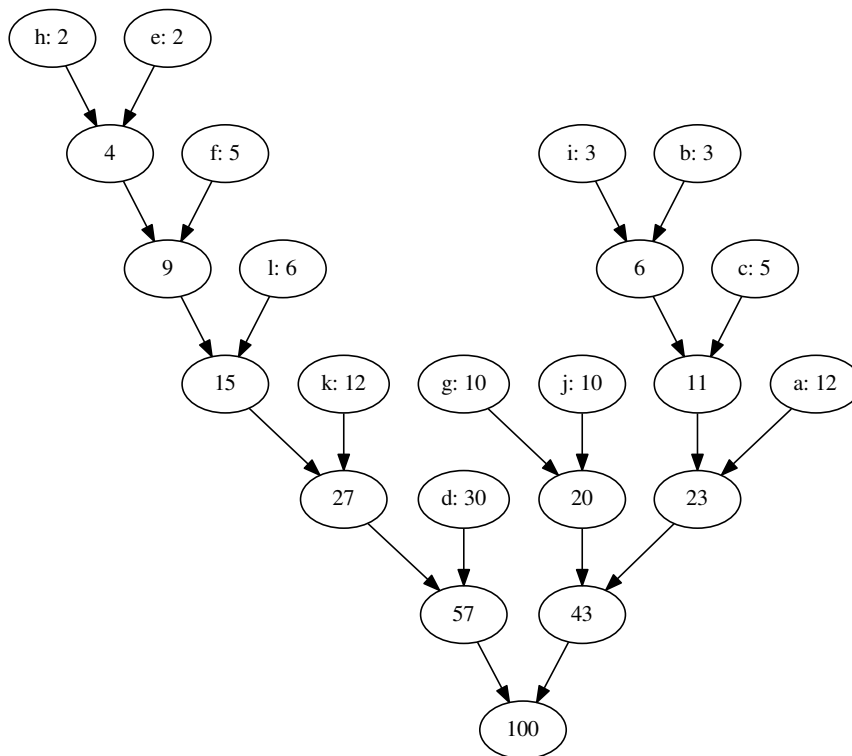
Mit dieser 5-Bit Code erhält man eine Auflösung von  $11,25^\circ$  oder auch 32 Codewörter. Da wir jedoch 24 Codewörter benötigen werden die vorderen und hinteren 4 Codewörter weggelassen. So erhalten wir einen einschrittig-zyklischen Code mit einer Auflösung von  $15^\circ$ :

$$\begin{aligned} &\{00110, 00111, 00101, 00100, 01100, 01101, 01111, 01110, 01010, 01011, 01001, 01000, \\ &11000, 11001, 11011, 11010, 11110, 11111, 11101, 11100, 10100, 10101, 10111, 10110\} \end{aligned}$$

## 5.3

**a)**

Nach dem Verfahren aus der Vorlesung ergibt sich folgender Baum:



Wenn man die linke Seite jeweils mit einer 0 und die rechte Seite mit einer 1 versieht, erhält man den zugehörigen Code:

a	111
b	11001
c	1101
d	01
e	000001
f	00001
g	100
h	000000
i	11000
j	101
k	001
l	0001

b)

Die Entropie berechnet sich nach der Formel aus der Vorlesung (Summe aller Produkte der Wahrscheinlichkeit eines Wortes mit seinem Informationsgehalt):

$$\begin{aligned}
& -(0,12 \cdot \log_2(0,12) + 0,03 \cdot \log_2(0,03) + 0,05 \cdot \log_2(0,05) + 0,3 \cdot \log_2(0,3) + 0,02 \cdot \log_2(0,02) + \\
& 0,05 \cdot \log_2(0,05) + 0,06 \cdot \log_2(0,06) + 0,12 \cdot \log_2(0,12) + 0,1 \cdot \log_2(0,1) + 0,03 \cdot \log_2(0,03) + 0,02 \cdot \\
& \log_2(0,02) + 0,1 \cdot \log_2(0,1)) \\
& \approx 3,12
\end{aligned}$$

## 5.4

a)

Mit einem 4-bit-wort lassen sich 16 verschiedene Codewörter bilden. Sei  $p_i$  die Wahrscheinlichkeit eines Wortes mit dem Index  $i$  und  $l_i$  seine Länge (hier immer 4); Dann ist der maximale Informationsgehalt  $H_0$ :

$$\begin{aligned}
H_0 &= \sum_{i=1}^{16} p_i \cdot l_i \\
&= 16 \cdot \frac{1}{16} \cdot 4 = 4
\end{aligned}$$

Wenn wir nur 10 Zeichen (alle mit der gleichen Wahrscheinlichkeit) belegen, ergibt sich für den tatsächlichen durchschnittlichen Informationsgehalt (Entropie  $H$ ):

$$H = \sum_{i=1}^{10} p_i \cdot \log_2\left(\frac{1}{p_i}\right)$$

wobei  $p_i = \frac{1}{10} = 0,1$ :

$$\begin{aligned}
H &= 10 \cdot 0,1 \cdot \log_2\left(\frac{1}{0,1}\right) \\
&= \log_2(10) \approx 3,22
\end{aligned}$$

Die Redundanz:

$$R = H_0 - H \approx 0,68$$

b)

Anstatt eine Dezimalziffer mit 2 Stellen durch 2 Codewörter dazustellen, wird hier eine neue Codierung erschaffen. Die Codierung in Aufgabe 5.4 a) bräuchte zur Darstellung einer Dezimalzahl mit 2 Stellen 8 bits. Definieren wir die Zahlen 00-99 jedoch in einem neuen Code benötigen wir lediglich 7 bits, da sich mit einem 7-bit langen Code 128 Codewörter bilden lassen.

$H_0$  errechnet sich auf die gleiche Art und Weise wie in a). Da die Summe aller Wahrscheinlichkeiten wieder = 1 ist, ist  $H_0$ :

$$H_0 = 1 \cdot l_i = 7$$

Entropie  $H$ , wie in a):

$$H = 100 \cdot 0,01 \cdot \log_2\left(\frac{1}{0,01}\right) \\ = \log_2(100) \approx 6,64$$

Redundanz  $R$ :

$$R = H_0 - H \approx 0,36$$

**c)**

Um die Zahlen 000-999 darzustellen werden 10 bits benötigt.

$$H_0 = 10 \\ H = 1000 \cdot 0,001 \cdot \log_2\left(\frac{1}{0,001}\right) = 9,97 \\ R = H_0 - H \approx 0,034$$

Um die Zahlen 0000-9999 darzustellen, werden 14 bits benötigt.

$$H_0 = 14 \\ H = 10000 \cdot 0,0001 \cdot \log_2\left(\frac{1}{0,0001}\right) = 13,288 \\ R = H_0 - H \approx 0,712$$

Die Unterschiede ergeben sich durch die Differenz zwischen dem möglichen Informationsgehalt und dem momentan genutzten. So werden beim Code mit 10 bits 1000 Codewörter der maximal 1024 Codewörter ausgenutzt. Es liegt also beinahe ein Minimalcode vor. Das Verhältnis ist  $\frac{1000}{1024} \approx 0,977$ . Beim Code mit 14 bits werden lediglich 10000 der 16384 möglichen Codewörter benutzt. Es liegt also ein Verhältnis von  $\frac{10000}{16385} \approx 0,61$  vor.

Die Unterschiede ergeben sich also aus der unterschiedlich ausgenutzen möglichen Codewortmenge.

**d**

Berechnung nach Fano:

Wir sortieren die Menge aller zu codierenden Wörter nach ihrer Wahrscheinlichkeit (hier überflüssig) und teilen sie in zwei Gruppen mit möglichst gleicher Gesamtwahrscheinlichkeit auf. Die eine Gruppe bekommt als MSB eine 1, die andere eine 0:

Wörter	Code-Präfix	Wahrscheinlichkeit
0-4	0	0,5
5-9	1	0,5

Diesen Schritt wiederholen wir:

{0, 1, 2}	00	0,3
{3, 4}	01	0,2
{5, 6, 7}	10	0,3
{8, 9}	11	0,2

Und noch einmal:

{0, 1}	000	0,2
2	001	0,1
3	010	0,1
4	011	0,1
{5, 6}	100	0,2
7	101	0,1
8	110	0,1
9	111	0,1

Im letzten Schritt werden noch die letzten beiden Zweiergruppen aufgeteilt:

0	0000	0,1
1	0001	0,1
2	001	0,1
3	010	0,1
4	011	0,1
5	1000	0,1
6	1001	0,1
7	101	0,1
8	110	0,1
9	111	0,1

Der mögliche Informationsgehalt ist, gemäß obiger Formel:

$$H_0 = 4 \cdot 0,4 + 6 \cdot 0,6 = 5,2$$

Entropie H:

$$\begin{aligned}
 H &= \sum_{i=1}^{10} p_i \cdot \log_2\left(\frac{1}{p_i}\right) \\
 &= 4 \cdot 0,4 \cdot \log_2\left(\frac{1}{0,4}\right) + 6 \cdot 0,6 \cdot \log_2\left(\frac{1}{0,6}\right) \\
 &\quad - 1,6 \cdot \log_2(0,4) - 3,6 \cdot \log_2(0,6) \\
 &\approx 4,77
 \end{aligned}$$

Redundanz R:

$$R = H_0 - H = 5,2 - 4,77 \approx 0,432$$