TECHNISCHE HOCHSCHULE DEGGENDORF

Spezielle Protokolle des IoT

Redundanz

Einleitung

Redundanz

- Überblick Übertragungsstrecke
- Einige Grundbegriffe der Informationstheorie
- Redundanzreduktion
- Fehlerdetektion und –korrektur (Redundanzerhöhung)

Redundanz

Redundanz im allgemeinen Sprachgebrauch

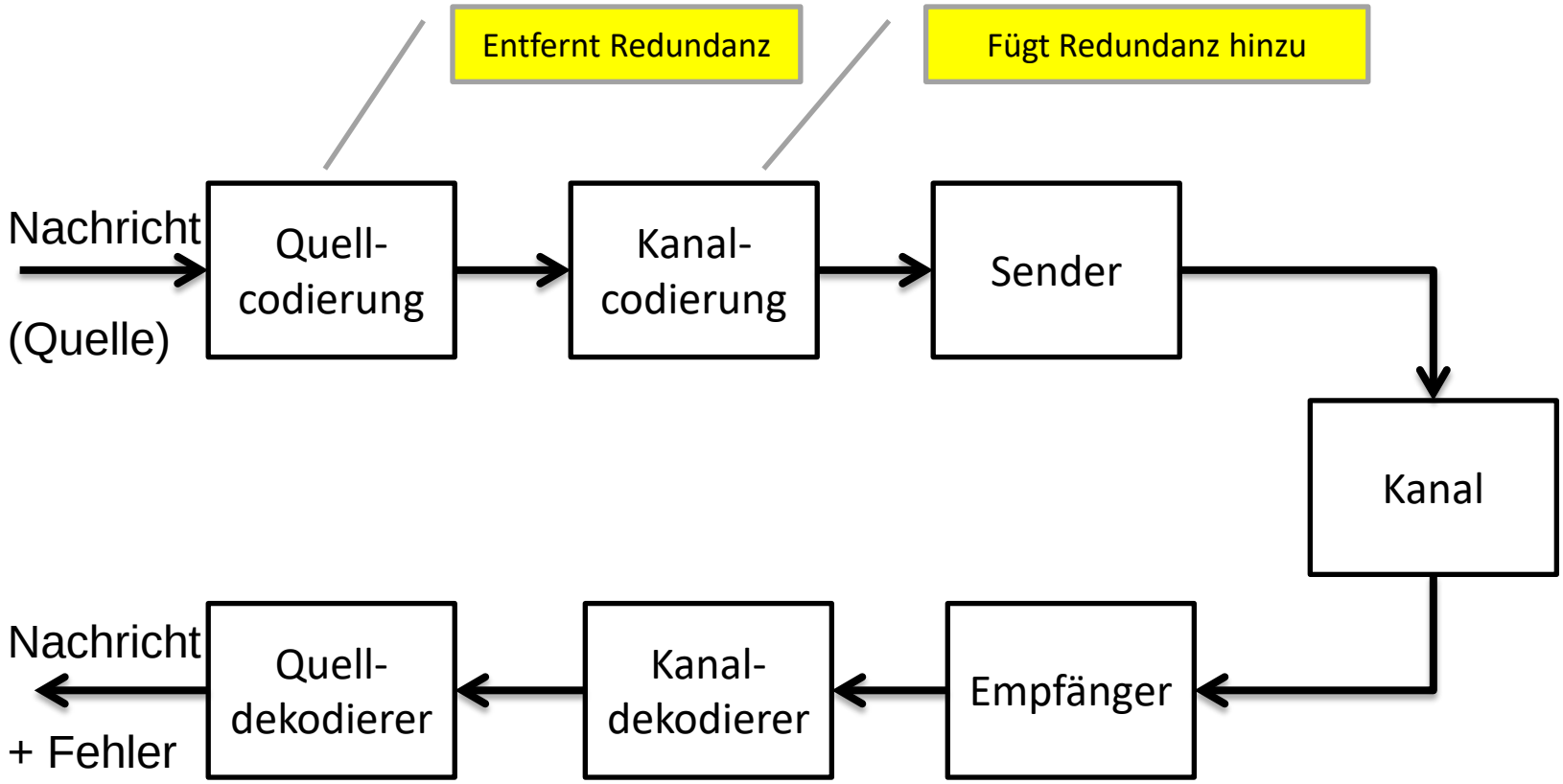
- Welche Situationen oder Beispiele kennen Sie, in denen Redundanz positive oder negative Aspekte hat?

Redundanz

Redundanz im allgemeinen Sprachgebrauch:

- - Redundanz fügt zusätzlichen Aufwand hinzu
- - Redundante Informationen können Langeweile auslösen
- + Redundante Systeme können die (Ausfall-) Sicherheit erhöhen
- + Redundante Informationsspeicherung (Backup)
- ...

Übertragungskette



Quellcodierung

Die gebräuchlichsten Formen der Quellcodierung sind:

- Umwandlung des Signals von einer Repräsentation in eine andere, z.B. 2D Signal durch Zeilenscan zu 1D Signal
- Redundanzreduktion: Verlustlose Kompression des Eingangssignals, z.B. ZIP
- Irrelevanzreduktion: Verlustbehaftete Kompression, insbesondere zum Einhalten einer bestimmten Datenrate, z.B. Audio- und Videocodierung

Maßzahl:

- $Kompressionsrate = \frac{Informationsgehalt\ der\ Originalnachricht\ [bit]}{Informationsgehalt\ der\ komprimierten\ Nachricht\ [bit]}$

Limitierung:

- Das erste Theorem von Shannon legt die Grenze für verlustlose Übertragung fest (Näheres später).

Kanalcodierung

Ziel der Kanalcodierung ist, die Information gegen Übertragungsstörungen zu sichern

Typische Übertragungsstörungen:

- Informationsverlust (Bitverlust oder Paketverlust)
- Informationsveränderungen (fehlerhafter Empfang von Symbolen)

Typische Strategien:

- Fehlerdetektion, z.B. Paritätsinformation
- Fehlerkorrektur, z.B. Forward Error Correction (FEC) oder Automatic Repeat Request (ARQ)

Quellcodierung: Run Length Encoding (RLE)

RLE : Run Length Encoding

- Beispiel: Textseite
- Problem: Sehr viel weiße Fläche (0), sehr wenig schwarze Fläche (1)
- Idee: Anstatt jedes Pixel zu übertragen, werden die Anzahl der Nullen gezählt und danach wird genau eine 1 angefügt.
- Wenn das Quellsignal lautet:
  M=‘00000000100001100000000000000001’
  Überträgt man:
  RLE(M)= ‘8, 4, 0, 17’
  Für die Binärübertragung benötigt man zusätzliche algorithmische Überlegungen, z.B. die maximale Anzahl der aufeinanderfolgenden Nullen
  Beispiel: Anzahld der aufeinanderfolgenden Nullen (run) mit 4 Bit codierung, danach 1 bit für den darauffolgenden Wert (Level) – 0, falls der run länger ist als 16.
- M=‘ 000000001 0000 1 0000000000000000 01’
  RLE(M)=‘ 1000,1, 0100,1, 0000,1, 1111,0, 0001,1’

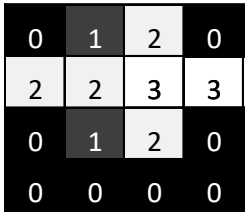
Einige Definitionen

Definition des Begriffs Quelle:

- Bei der Übertragung werden (wenige) **Zeichen** oder **Symbole** eingesetzt
- Das **Alphabet** legt die möglichen **Symbole** fest.
  $A = \{a_1, a_2, a_3, \ldots, a_K\}$.
  Zum Beispiel enthält das Morsezeichen-Alphabet die Symbole “kurz” (a_1), “lang” a_2, und “Pause” a_3
- In einem **Wörterbuch** werden die **Wörter** als erlaubte Kombinationen von Symbolen (und deren Bedeutung) festgehalten:
  $M = \{m_1, m_2, m_3, \ldots, m_N\}$
  Zum Beispiel ist das erste Wort des Morse-Wörterbuchs m_1 = „kurz, lang“, m_2 = „lang, kurz, kurz, kurz“, …
- Eine **Quelle** emittiert schließlich **Nachrichten** aus **Wörtern**.
- In letzter Konsequenz emittiert also eine Quelle Symbole, die im Alphabet festgelegt sind und deren Reihenfolge im Wörterbuch definiert ist.
- Bei den meisten Quellen lässt sich feststellen, dass jedes Wort m_i wird mit einer (mehr oder weniger festen) Wahrscheinlichkeit p_i gesendet wird.
- Anmerkung: Dies ist die einfachste Form einer Quelle (diskret, stationär, gedächtnislos).

Einige Definitionen

Beispiel:



Mögliche Quelle 1:
Alphabet: A={0,1}
Wörterbuch: M={00, 01, 10, 11}
Sendung (raster scan):
00 01 10 00 10 10 11 11 00 …
Formuliert als Wörter:
m1, m2, m3, m1, m3, m3, m4, …

Mögliche Quelle 2:
Alphabet: A={0, 1, 2, 3}
Wörterbuch: M={0, 1, 2, 3}
Sendung:
0 1 2 0 2 2 3 3 0 1 2 0
Formuliert als Wörter:
m1, m2, m3, m1, m3, m3, m4, …

Wahrscheinlichkeiten der Nachrichten (Schätzwerte):
Vorkommen des Wertes 0: 8 Mal → p_1 = 8/16=0.5
p2 = 2/16=0.125, p3=4/16=0.25, p4=2/16=0.125

Mögliche Quelle 3: Codierung von 2 aufeinanderfolgenden Werten
Alphabet: A={0,1, 2, 3}
Wörterbuch: M={00, 01, 02, 03, 10, 11,12,13,20,21,22,23,30,31,32,33}
Sendung (raster scan): 01 20 22 33 01 20 00 00
Formuliert als Wörter: m2, m9, m11, m16, m2, m9, m1, m1
Wahrscheinlichkeit (Schätzung):
Vorkommen von m1 (’00’) : 2 Mal → p1=2/8
p2=2/8, p9=2/8, p11=1/8, p16=1/8
p2=p3=p4=p5=p6=p7=p8=p10=p12=p13=p14=p15=0
→ Ist diese Wahrscheinlichkeitsverteilung realistisch für Bilder?

Entropie

Definitionen:

- Die Entropie liefert den Informationsgehalt einer Quelle
- Der Begriff der „Information“ betitelt hierbei den Kenntnisgewinn nach dem Empfang einer Nachricht
- Beispiel:
  Eine Quelle emittiert 2 Symbole mit den Wahrscheinlichkeiten $p_1=0.99$ und $p_2=0.01$. Die Entropie dieser Quelle ist niedrig, da man im Allgemeinen den Empfang der Nachricht m_1 erwartet und der Empfang der Nachricht m_1 daher wenig Kenntnisgewinn bringt.
- Die Entropie ist eine festgelegte Maßzahl, die für eine Quelle mit N Wörterbucheinträgen mit jeweils der Auftrittswahrscheinlichkeit p_i für jede Nachricht m_i wie folgt berechnet:

$$\mathrm{H} = -\sum_{i=1}^{N} p_i \cdot log_2(p_i)$$

Die Entropie hat die Einheit [bit]

Codierung

Definitionen:

- **Codierung** bedeutet, dass die Wörter eines Wörterbuchs auf die Wörter eines anderen Wörterbuchs eineindeutig abgebildet werden und somit die Symbole einer Quelle in die Symbole einer anderen Quelle überführt werden können.
- Die Codierung assoziiert für jedes Wort m_i ein **Codewort** M_i.
- Beispiel:
  Alphabet1 = {A,B,C,…, Leerzeichen} Wörterbuch1 = {A, B, C, …, Leerzeichen}
  Alphabet2 = {kurz, lang, Pause} Wörterbuch2 = {'Pause, Pause, Pause', 'kurz, lang, Pause', 'lang, kurz, kurz, kurz, Pause', …}
  Codierung: $m_1 \rightarrow M_2$, $m_2 \rightarrow M_3$, … , $m_N \rightarrow M_1$ (Position des Leerzeichens beachten)
- Die Länge des Codeworts wird beschrieben als: $l_i = l(M_i)$
- Eine wichtige Eigenschaft eines Codes ist die mittlere Codewortlänge: $L = \sum_{i=1}^{N} p_i \cdot l_i$

Nachricht	Quelle 1	Quelle 2	Wahrsch.	Codewort	Länge	pi*li
m1	00	0	0,5	M0= 11	2	1
m2	01	1	0,125	M1= 10	2	0,25
m3	10	2	0,25	M2= 01	2	0,5
m4	11	3	0,125	M3= 00	2	0,25
					Mittlere Länge	**2**

Codierung mit variabler Codewortlänge

Codierung mit variabler Codewortlänge:

- Beispiel einer einfachen Quelle mit 4 Nachrichten:
  - m_1 mit der Auftrittswahrscheinlichkeit $Pr(m_1) = 0,5$
  - m_2 mit der Auftrittswahrscheinlichkeit $Pr(m_2) = 0,25$
  - m_3 mit der Auftrittswahrscheinlichkeit $Pr(m_3) = 0,125$
  - m_4 mit der Auftrittswahrscheinlichkeit $Pr(m_4) = 0,125$
- Codierung mit variabler Länge verwendet kurze Codewörter für häufig auftretende Nachrichten und längere für seltene Nachrichten (Anpassung an die Statistik der Quelle):

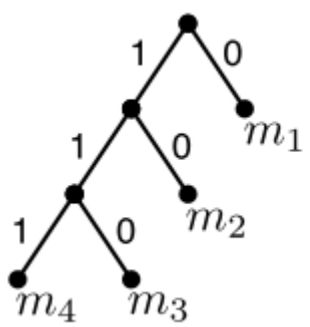
	m1	m2	m3	m4
Codewort mit fester Länge	00	01	10	11
Codewort mit variabler Länge	1	10	110	111

- ➔ Mittlere Länge der Codierung mit fester Codewortlänge: $L_1 = 2$ bits
- ➔ Mittlere Länge der Codierung bei variabler Codewortlänge:
  $L_2 = 1 \times 0,5 + 2 \times 0,25 + 3 \times 0,125 + 3 \times 0,125 = 1,75$ bits
- Der obige Code ist nicht dekodierbar. Warum? Was muss geändert werden?

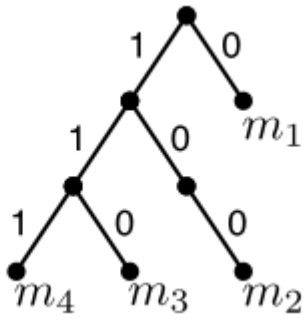
Präfix-Code

Definition Präfix-Code

- Ein Code wird als präfixfreier Code oder kurz Präfix-Code bezeichnet, wenn kein Codewort den Beginn eines anderen Codewortes darstellt
- Jeder Präfix-Code ist eindeutig dekodierbar
- Für jeden Präfix-Code lässt sich ein Baum aufstellen, dessen Blätter die Codewörter sind



$$m_1 \rightarrow 0$$
$$m_2 \rightarrow 10$$
$$m_3 \rightarrow 110$$
$$m_4 \rightarrow 111$$



$$m_1 \rightarrow 0$$
$$m_2 \rightarrow 100$$
$$m_3 \rightarrow 110$$
$$m_4 \rightarrow 111$$

- Ein Code ist nicht reduzierbar (so kurz als möglich), wenn alle Knoten entweder 0 oder 2 Äste haben.

Entropie und Codewortlänge

Erstes Theorem von Shannon:

- Für jede Quelle der Entropie H muss die mittlere Codewortlänge L größer oder gleich der Entropie H sein, damit eine eindeutige Dekodierung erfolgen kann. Kurz:

$$L \geq H$$

- Dies ist die untere Schranke für die durchschnittliche Codewortlänge.
- Für Präfix-Codes gibt es auch eine obere Schranke, diese liegt mit H+1 [bit], so dass immer (mindestens) 1 Präfix Code exististiert, für dessen Länge L gilt:

$$H \leq L \leq H + 1$$

- Die Effizienz E des Codes wird angegeben als: $E = \frac{H}{L}$
- Noch einmal zur Erinnerung, die Entropie wird berechnet als:

$$H = -\sum_{i=1}^{N} p_i \cdot log_2(p_i)$$

Entropiecodierung

Wie erhält man nun einen Code, der möglichst optimal ist?

- Huffman Codierung
- Arithmetische Codierung

Huffman-Codierung

Huffman-Codierung

- Veröffentlichung 1952
- Höchste erreichbare Effizienz E für eine Codierungen mit 1 Symbol pro 1 Nachricht
- Verfahrensbeschreibung am Beispiel:

	$p_i^{(0)}$
m_1	0,40
m_2	0,18
m_3	0,10
m_4	0,10
m_5	0,07
m_6	0,06
m_7	0,05
m_8	0,04

Verfahren: Huffman

Verfahren:

- Zunächst ordnet man die Nachrichten nach absteigender Wahrscheinlichkeit (hier bereits gegeben)
- Dann summiert man die beiden kleinsten Wahrscheinlichkeiten auf und erzeugt so eine neue (kombinierte) Nachricht
- Dann sortiert man die Nachrichten wieder nach absteigender Wahrscheinlichkeit

	$p_i^{(0)}$	$p_i^{(1)}$
m_1	0,40	0,40
m_2	0,18	0,18
m_3	0,10	0,10
m_4	0,10	0,10
m_5	0,07	0,09
m_6	0,06	0,07
m_7	0,05 1	0,06
m_8	0,04 0	

Anmerkung: Die Zuordnung der Nachrichten m_1-m_8 zu den Wahrscheinlichkeiten ist nur für die erste Spalte korrekt, in der 2. Spalte hat sich die kombinierte Nachricht mit der Wahrscheinlichkeit von 0,09 vor die Wahrscheinlichkeiten für die Nachricht m_5 mit 0,07 und m_6 mit 0,06 geschoben.

Verfahren: Huffman

Verfahren:

- Man setzt diese Schritte fort, bis es nur noch 2 Nachrichten gibt
- Für jede Summen-Nachricht, generiert aus 2 Einzelnachrichten wird eine Bitzuweisung von 1 an das obere und 0 an das untere Pfeilende notiert (siehe rote Markierungen)

	$p_i^{(0)}$	$p_i^{(1)}$	$p_i^{(2)}$	$p_i^{(3)}$	$p_i^{(4)}$	$p_i^{(5)}$	$p_i^{(6)}$
m_1	0,40	0,40	0,40	0,40	0,40	0,40	0,60 1
m_2	0,18	0,18	0,18	0,19	0,23	0,37 1	0,40 0
m_3	0,10	0,10	0,13	0,18	0,19 1	0,23 0	
m_4	0,10	0,10	0,10	0,13 1	0,18 0		
m_5	0,07	0,09	0,10 1	0,10 0			
m_6	0,06	0,07 1	0,09 0				
m_7	0,05 1	0,06 0					
m_8	0,04 0						

Verfahren: Huffman

Verfahren:

- Nun kann das Codewort von rechts nach links (!) abgelesen werden, indem man den Pfeilen folgt, hier am Beispiel der Nachricht m_5, für die gilt: $m_5 = 1011$

	$P_i^{(0)}$	$P_i^{(1)}$	$P_i^{(2)}$	$P_i^{(3)}$	$P_i^{(4)}$	$P_i^{(5)}$	$P_i^{(6)}$
m_1	0,40	0,40	0,40	0,40	0,40	0,40	**0,60** (1)
m_2	0,18	0,18	0,18	0,19	**0,23**	0,37 1	0,40 0
m_3	0,10	0,10	**0,13**	0,18	0,19 1	**0,23** (0)	
m_4	0,10	0,10	0,10	**0,13** (1)	0,18 0		
m_5 →	**0,07**	0,09	0,10 1	0,10 0			
m_6	0,06	**0,07** (1)	0,09 0				
m_7	0,05 1	0,06 0					
m_8	0,04 0						

Verfahren: Huffman

Auswertung:

- Die Entropie der Quelle lag bei H=2,553 bit/Codewort
- Die durchschnittliche Länge der Codewörter liegt bei: L = 0,40 x 1 + 0,18 x 3 + 0,10 x 3 + 0,10 x 4 + 0,07 x 4 + 0,06 x 4 + 0,05 x 5 + 0,04 x 5 = 2,61 bits/mots
- Kontrolle: Die durchschnittliche Länge liegt zwischen H und H+1
- Die Effizienz dieses Huffman-Codes: E = 2,553 / 2,61 = 97,8 %
- Zum Vergleich: Bei Codierung mit fester Länge werden bei 8 Nachrichten 3 Bit benötigt, die Effizienz liegt daher bei: E = 2,553 / 3 = 85 %

	$p_i^{(0)}$	Codewort
m_1	0,40	0
m_2	0,18	110
m_3	0,10	100
m_4	0,10	1111
m_5	0,07	1011
m_6	0,06	1010
m_7	0,05	11101
m_8	0,04	11100

Verfahren: Huffman

Limitierung des Huffman-Verfahrens:

- Man kann eine Nachricht nur mit einer ganzzahligen Anzahl an Bits repräsentieren. Gemäß Shannon wäre es optimal, jedes Codewort mit einer Länge zu repräsentieren, die dem Zweier-Logarithmus seiner Auftrittswahrscheinlichkeit entspricht
- Beispiel einer ungünstigen Quelle:

	p_i	mot de code
m_1	0,75	1
m_2	0,25	0

Entropie: H=0.81 bit/Codewort
Mittlere Länge des Huffman-Codes: L=1 bit/Codewort
Effizienz: E= 0.81/1 = 81%

Arithmetische Codierung

Arithmetische Coderung

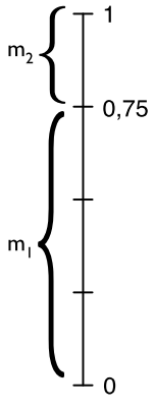
- 1980 entwickelt
- Prinzip: Codierung einer Nachricht in Abhängigkeit von ihrer Auftrittswahrscheinlichkeit mit einer rationalen Anzahl an Bits (im Sinne von: nicht ganzzahlig)
- Technik: Codierung einer Sequenz von Nachrichten durch Wahl eines Repräsentanten in einem Intervall. Dieses Intervall charakterisiert eindeutig genau diese Sequenz von Nachrichten. Es ist ein Teilintervall aus dem Startintervall [0,1[
- Vorteil gegenüber Huffman: Effizienz nähert sich beliebig langen Nachrichtenfolgen beliebig nahe der 100%
- Nachteil gegenüber Huffman: Komplizierter zu verstehen und zu implementieren

Verfahren: Arithmetische Codierung

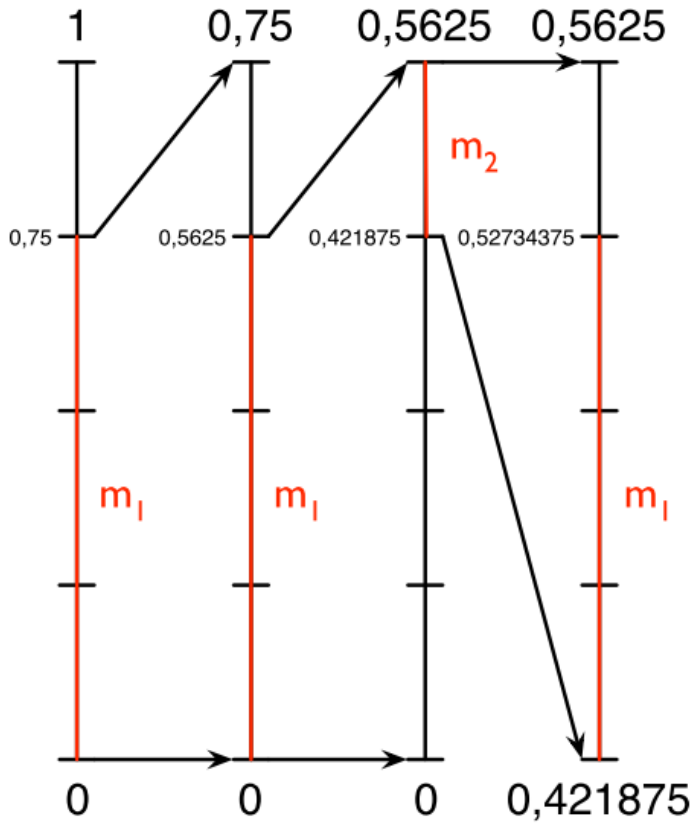
Verfahren:

- Der Algorithmus zerlegt das (Start-)Intervall [0,1[in Teilintervalle
- Für n mögliche Nachrichten werden n Teilintervale angelegt
- Jedes der n Teilintervalle ist so breit, wie die Auftrittswahrscheinlichkeit seiner Nachricht
- Das Teilintervall der tatsächlich gesendeten Nachricht wird im nächsten Schritt auf die gleiche Weise in Teilintervalle zerlegt
- Beispiel:
  Gegeben eine Quelle mit 2 Nachrichten und die Übertragung der Sequenz m_1, m_1, m_2, m_1

	p_i	mot de code
m_1	0,75	1
m_2	0,25	0



Verfahren: Arithmetische Codierung



1. $low_0 = 0$
   $high_0 = 1$

2. $low_1 = low_0 + 0{,}0 \times (high_0 - low_0)$
   $high_1 = low_0 + 0{,}75 \times (high_0 - low_0)$

3. $low_2 = low_1 + 0{,}0 \times (high_1 - low_1)$
   $high_2 = low_1 + 0{,}75 \times (high_1 - low_1)$

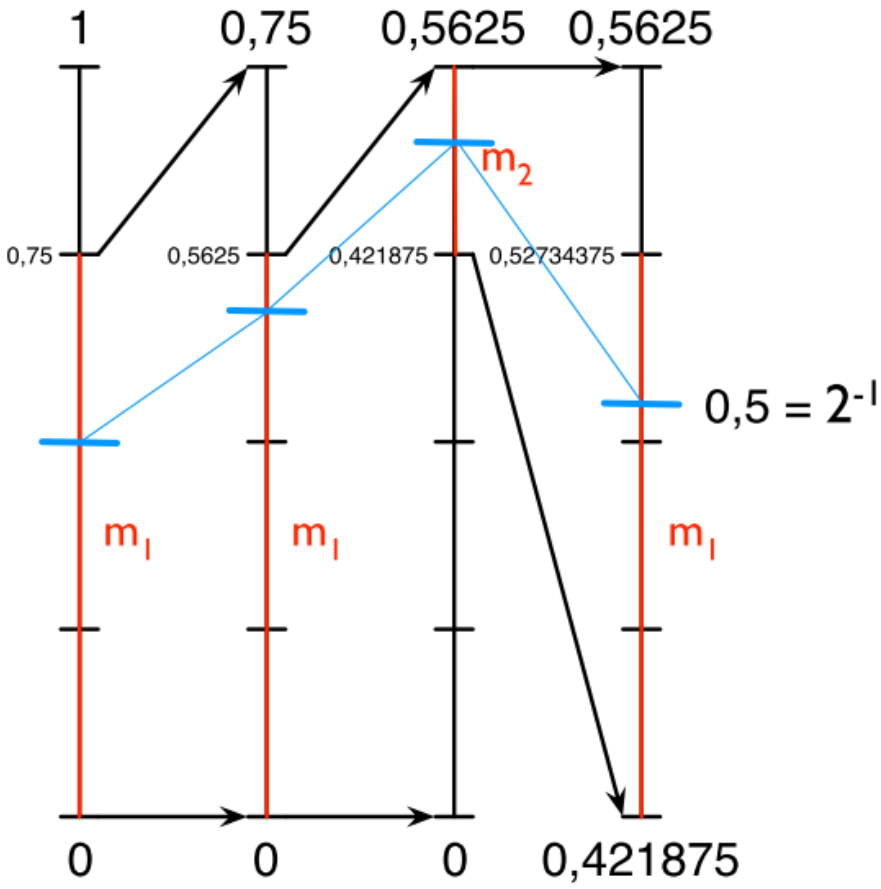
4. $low_3 = low_2 + 0{,}75 \times (high_2 - low_2)$
   $high_3 = low_2 + 1{,}0 \times (high_2 - low_2)$

5. $low_4 = low_3 + 0{,}75 \times (high_3 - low_3)$
   $high_4 = low_3 + 1{,}0 \times (high_3 - low_3)$

Verfahren: Arithmetische Codierung

Verfahren:

- Nach Abschluss der Intervallzerlegung wird ein möglichst kurzer Repräsentant aus dem Intervall gewählt (« kurz » bezieht sich hier auf die Binärrepräsentation der Zahl)
- Hier: 0,5 liegt in dem entgültigen Intervall. 0,5 entspricht in Binärdarstellung für Nachkommastellen: $1 * 2^{-1}$. Es muss also nur ein einziges Bit (« 1 ») übertragen werden.
- Bei der Übertragung von 3 Nachrichten m1 und 1 Nachricht m2 würden im langfristigen Mittel $3*\log2(0,75)+1*\log2(0,25) = 3*0,41+1*2 = 3,2451$ Bit benötigt.
- Mit der Huffman-Codierung würden 4 Bit benötigt



Verfahren: Arithmetische Codierung

Verfahren:

- Decodierung:
  Gegeben den Empfang von « 1 » und die Umrechnung in die Zahl 0,5
  Gegeben die Information, dass 4 Nachrichten zu erwarten sind.
- Wie bei der Codierung: Aufteilung des Intervalls und diesmal Entscheidung für dasjenige Symbol in welchem der Zielwert (hier: 0,5) liegt.

Decodierung mit $code_0 = 0{,}5$

1. $0{,}0 \leq code_0 < 0{,}75 \rightarrow Symbol = m_1$

   $code_1 = (code_0 - 0{,}0) / 0{,}75 = 0{,}667$
2. $0{,}0 \leq code_1 < 0{,}75 \rightarrow Symbol = m_1$

   $code_2 = (code_1 - 0{,}0) / 0{,}75 = 0{,}889$
3. $0{,}75 \leq code_2 < 1{,}0 \rightarrow Symbol = m_2$

   $code_3 = (code_2 - 0{,}75) / 0{,}25 = 0{,}556$
4. $0{,}0 \leq code_3 < 0{,}75 \rightarrow Symbol = m_1$

Verfahren: Arithmetische Codierung

Probleme:

- Problem 1: Wie viele Nachrichten wurden codiert?
- Problem 2: In einem kontinuierlichen Bitstrom – wo hört die rationale Zahl 1 auf und wo fängt die zweite an?

Lösungen für das Problem 1:

- Die Anzahl der Nachrichten ist a priori bekannt, z.B. Anzahl Pixel in einem Bild
- Hinzufügen einer Nachricht: End-of-Transmission im Wörterbuch (mit entsprechend geringer Wahrscheinlichkeit)

Lösung für das Problem 2:

- Definition, dass eine Übertragung dann und nur dann terminiert ist, wenn der Zielwert eineindeutig in dem dekodierten Intervall liegt und keine andere Interpretation mehr mit weiter im Bitstrom folgenden Bits möglich ist (dies erfordert ggf. das Hinzufügen von 1-2 bits am Ende der Übertragung)
  [Anm.: Dies erfordert etwas genaueres Eindenken und die Auseinandersetzung mit dem Q-Coder Verfahren von IBM]

Redundanz

Neben diesen Verfahren zur Ausnutzung der Auftrittswahrscheinlichkeiten gibt es noch weitere Verfahren zur Verringerung der Redundanz, insbesondere:

- Stark korrelierte Signale: Prädiktion und/oder Codebuch
- Signale mit veränderlicher Statistik: Adaptive Verfahren (z.B. Änderung der Wahrscheinlichkeiten in jedem Schritt für arithmetische Codierung)

TECHNISCHE HOCHSCHULE DEGGENDORF

Danke für Ihre Aufmerksamkeit!