Einführung Internet Computing Teil 2

Medien und Information

Kapitel 1: Digitalisierung und Kodierung

Michael Granitzer

Harald Kosch

Universität Passau

Kapitel EIC Teil 2- Medien und Information:I

I. EIC Teil 2- Medien und Information

- ❑ Digitalisierung
- ❑ Kodierung
- ❑ Informationstheorie
- ❑ Kompression

Lernziele

- Digitalisierung und Abtasttheorem - wie entstehen digitale Medien?
- Kodierung von Medien - wie werden digitale Medien repräsentiert?
- Informationstheorie - wie kann man den Gehalt von Information messen und was sind gute Repräsentationsformen von Medien?
- Kompression - wie kann ich Repräsentationsformen automatisiert optimieren, d.h. kürzere Kodierungen für die gleiche Information finden?

Einleitung

Digitale Medien

Definition 1 (Digitale Medien)

Digitale Medien sind Medien, welche in digitaler Form gespeichert, verarbeitet oder manipuliert werden.

Typische Beispiele digitaler Medien und Computer-gestützter Funktionen:

- **Datenträger/Darstellungstechniken**: eBooks, digitales Fernsehen, digitale Bilder
- **Neue Medien und Interaktionsparadigmen**: World Wide Web, Hypertext, Ubiquitous Computing
- **Erstellung/Manipulation von Inhalten**: Rechner-basierte Authoring Tools und Programme zur Medienbearbeitung

Wie können Medien digitalisiert und kodiert werden?

Kapitel EIC Teil 2- Medien und Information:I

I. EIC Teil 2- Medien und Information

- Digitalisierung
- Kodierung
- Informationstheorie
- Kompression

Digitalisierung - Analog und Digital

Digitalisierung - Analog und Digital

Analoge Signale

Definition 2 (Analoges Signal)

Ein analoges Signal ist die deterministische Änderung einer physikalischen Größe (entlang Raum und/oder Zeit) entsprechend einem Messwert der zu übertragenen Information. Das analoge Signal kann als wertkontinuierliche Funktion $f(x_1, x_2, \ldots)$ von ein bis mehrerer unabhängigen kontinuierlichen Variablen (z.B. Raum, Zeit) aufgefasst werden.

Eigenschaften von Signalen

- Kontinuierlich bedeutet, dass für jeden beliebigen Wert der unabhängigen Variablen, das Signal einen Wert aus einem beliebig genauen Wertebereich annehmen kann
- In analogen Signalen sind prinzipiell „beliebig“ genaue Beobachtung möglich (bei idealen Messgeräten)
- Analoge Signale sind störanfällig bei Übertragung oder Vervielfältigung

Digitalisierung - Analog und Digital

Beispiele für analoge Signale

- Helligkeit einer Lichtquelle in Candela
- Farbton einer Lichtquelle (Wellenlänge)
- Luftdruck Schwankungen als Schall
- Drehzahl eines rotierenden Objekts
- Mechanische Kraft
- Elektrische Spannung
- Elektrischer Widerstand, elektrische Kapazität

Digitalisierung - Analog und Digital

Digitales Signal

Definition 3 (Digitales Signal)

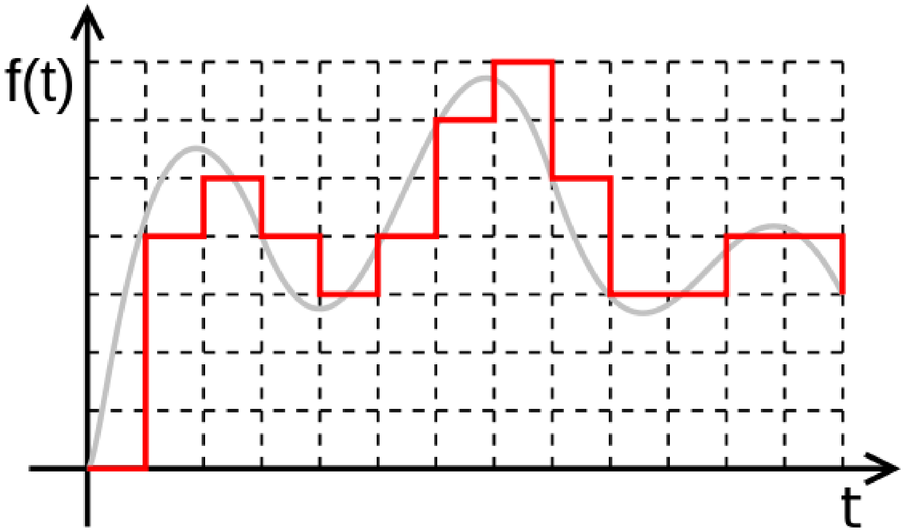
Ein digitales Signal ist die Repräsentation eines analogen Signals durch eine abzählbare, diskrete Anzahl von Werten zu einer diskreten, abzählbaren Anzahl von Zuständen. Das digitale Signal kann als Funktion $f(x_1, x_2, \ldots)$ mit abzählbaren, diskreten Wertebereiche von ein bis mehrerer unabhängigen, diskreten Variablen (z.B. Raum, Zeit) aufgefasst werden

Eigenschaften Digitales Signal

- Raum- und Zeitwerte eines Signals werden durch einen endlichen (meist numerischen) Zeichenvorrat repräsentiert
- Digitale Signale sind Annäherungen an das Originalsignal
- Digitale Signale sind eine **Reihe von binär repräsentierten Zahlen**, welche verlustfrei vervielfältigt werden können
- Entstehung digitaler Signale:
  - **Synthese**: Digitales Signal wird aus anderen digitalen Signale errechnet
  - **Abtastung**: Ein gegebenes analoges Signal wird diskretisiert.
    - Bemerkung: Ein „analoges Signal“ kann auch als ideale mathematische Funktion (z.B. 3D Modell für Film) vorliegen

Digitalisierung - Analog und Digital

Digitales Signal - Beispiel



Bildquelle Wikipedia

Digitales Signal:

f(t)	0	4	5	4	3	4	6	7	...
t	1	2	3	4	5	6	7	8	...

Unterscheidung: **Diskretisierung** vs. **Quantisierung**

Digitalisierung - Abtastung

Digitalisierung - Abtastung

Digitales Signal - Diskretisierung/Quantisierung

Die Umwandlung eines analogen in ein digitales Signal, die **Digitalisierung**, entsteht durch Abtastung entlang des **Signalverlaufs (Diskretisierung)** und der **Signalstärke (Quantisierung)**

Definition 4 (Diskretisierung (Sampling))

Bei der (Abtastachsen-)Diskretisierung wird ein festes Raster von Messpunkten gleichen Abstands Δt auf der Achse festgelegt, über die sich das analoge Signal verändert. Der aktuelle Wert des Signals zu den Messpunkten dieses Rasters bezeichnet man als Sample (Messwert).

- Die Dichte der Messwerte wird als **Abtastrate (Sampling Rate)** bezeichnet.
- Das feste Raster können Zeitpunkte sein (Beispiel Audiosignal) oder aber auch Raumdimensionen (Beispiel Pixel)
- Die Höhe der Abtastrate spezifiziert wie genau das Originalsignal wieder rekonstruiert werden kann

Digitalisierung - Abtastung

Digitales Signal - Diskretisierung/Quantisierung

Definition 5 (Quantisierung)

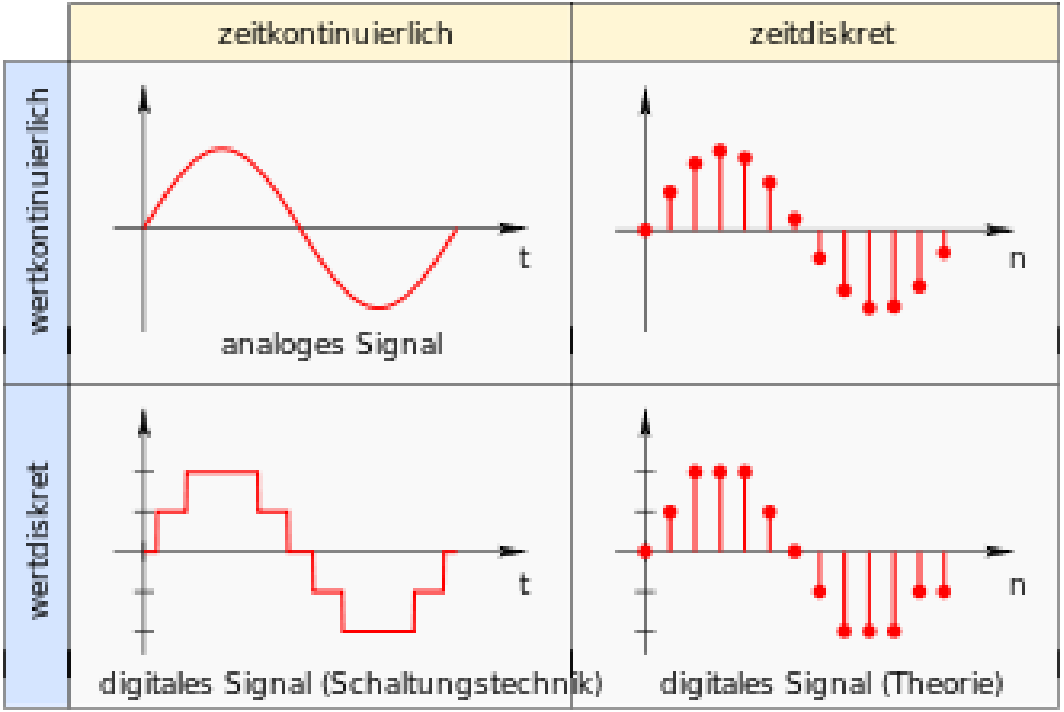
Die Quantisierung entspricht der Überführung der bei der Diskretisierung erhaltenen Messwerte auf einen diskreten, abzählbaren Wertebereich (meist im Binärsystem).

Eigenschaften:

- Jedem Messwert wird einer Zahl in einem abzählbaren Wertebereich zugeschrieben
- Die Kodierung erfolgt im Allgemeinen im binären Zahlensystem
- **Auflösung**: Bits pro Sample (bit resolution)
- Die Genauigkeit abhängig von der Anzahl der Bits pro Messwert

Digitalisierung - Abtastung

Kategorisierung von Signalen



Bildquelle Wikipedia

Digitalisierung - Abtastung

Kategorisierung von Digitalisierungsverfahren

Das Verfahren wird auch als **Pulse Code Modulation (PCM)** oder**waveform encoding** bezeichnet. In der technischen Realisierung der PCM können folgende Verfahren unterschieden werden:

Eigenschaften:

- **Lineare PCM:** Die Quantisierung findet in fixen Intervallen gleicher Größe statt.
- **Dynamische PCM:** Die Quantisierung findet in fixen Intervallen unterschiedlicher Größe statt (z.B. logarithmische Skala bei Audio Signalen zur Optimierung f. d. menschliche Wahrnehmung).
- **Differenz PCM:** Es werden nur die Differenzen zweier Abtastwerte kodiert. Bei sich langsam verändernden Signalen kann so die benötigte Kodierungsgröße reduziert werden.

Digitalisierung - Abtastung

Digitalisierung unterschiedlicher Medienarten

Verschiedene (Repräsentations-)Medien haben **verschiedene Arten von Bezugs- und Wertachsen** für die Signale (abweichende Terminologie)

Audio

- x-Achse = Zeit, y-Achse = Amplitude
- Genauigkeit der Diskretisierung = Abtastrate (sampling rate) (Hz)
- Genauigkeit der Quantisierung = Auflösung (resolution) (Bit)

Bild

- Zwei räumliche Achsen (x,y), z-Achse = Helligkeit/Farbwert
- Genauigkeit der Diskretisierung = räumliche Auflösung (Dichte der Bildelemente, z.B. dots per inch)
- Genauigkeit der Quantisierung = Farb- bzw. Grauwertauflösung (color resolution in bit, z.B. 16 Bit Farbtiefe)

Digitalisierung - Abtastung

Generalisierung der Darstellungsdimension

Ein Medium kann bis zu drei räumliche Dimensionen und eine zeitlich Dimension enthalten

- Text: Eine räumliche (oder zeitliche) Dimension
- Bild: Zwei räumliche Dimensionen
- Video: Zwei räumliche Dimensionen, eine zeitliche Dimension
- Raumklang und 3D-Video: Drei räumliche Dimensionen, eine zeitliche Dimension

Man spricht von raumabhängigen und zeitabhängigen Medien

Informationserhaltende Transformationen möglich

- z.B. Scrollen (Raum $\Rightarrow$ Zeit)
- z.B. Notenschrift (Zeit $\Rightarrow$ Raum)

Digitalisierung - Abtastung

Vorteile digitaler Signale

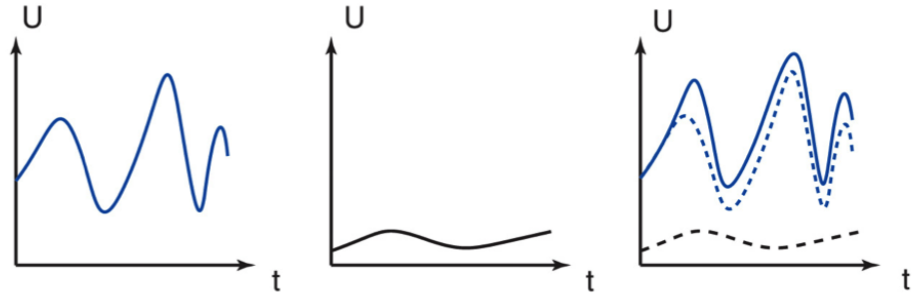


Abbildung 2.2: Analoges Signal und Störsignal: links Originalsignal, Mitte Störsignal, rechts resultierendes analoges Signal

Bildquelle [Malaka et.al. 2009]

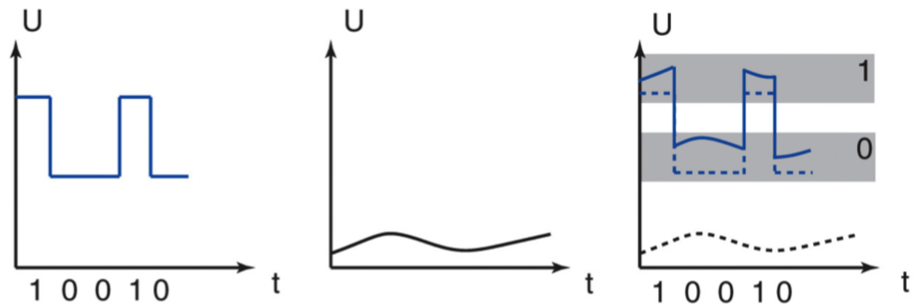


Abbildung 2.3: Digitales Signal und Störsignal: links Originalsignal, Mitte Störsignal, rechts resultierendes analoges Signal

Bildquelle [Malaka et.al. 2009]

Digitalisierung - Abtastung

Vor- und Nachteile

Vorteile

- Unempfindlichkeit gegen Störungen des Übertragungs- bzw. Speichermediums (e.g. magnetische Instabilität)
- Fehlererkennung/-korrektur durch entsprechende Codierung möglich („error correcting codes“)
- Verlustfrei kopierbar
- Viele Signale entstehen bereits in digitaler Form

Nachteile

- Informationsverlust
- Hoher Speicheraufwand/große Kanalkapazität
- Konvertierung mit Rechenaufwand verbunden

Digitalisierung - Abtasttheorem

Digitalisierung - Abtasttheorem

Abtasttheorem

Fragen:

- Wie gut kann das Originalsignal repräsentiert werden?
- Was sind gute Diskretiesierungsintervalle?

Lehrziel:

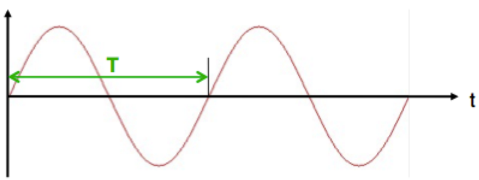
- Einführung einer präziseren Definition von Signalen
- Erklärung Abtasttheorem: $f_{abtastung} > 2 * f_{max}$
- Effekte unzureichender Abtastung

Digitalisierung - Abtasttheorem

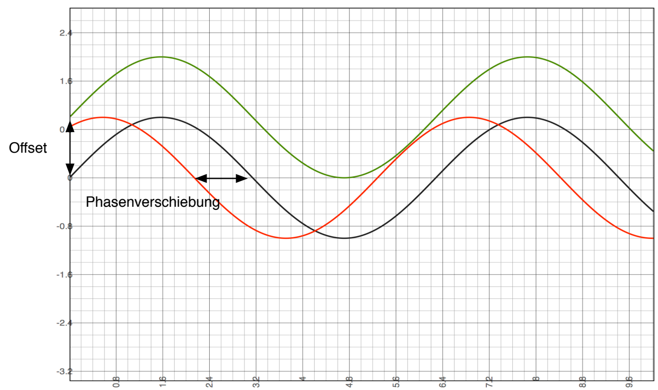
Eigenschaften Analoger Signale

Analoge Signale können als Summe von potentiell unendlich vielen periodische Sinusoiden-Signalen aufgefasst werden.

- Periodische bedeutet, dass sich der Signalverlauf regelmäßig wiederholt
- **Amplitude A**: Auslenkung des Signals
- **Periodenlänge T**: Dauer bis zum Beginn der nächsten Wiederholung
- **Frequenz f**: Anzahl der Wiederholungen pro Sekunde
- Zusammenhang: $T = \frac{1}{f}$
- **Phasenverschiebung** Φ: Verschiebung des Startzeitpunktes des Signals
- **Offset/Gleichanteil** DC: Verschiebung des Signals um einen konstanten Faktor



Quelle Prof. Butz, LMU

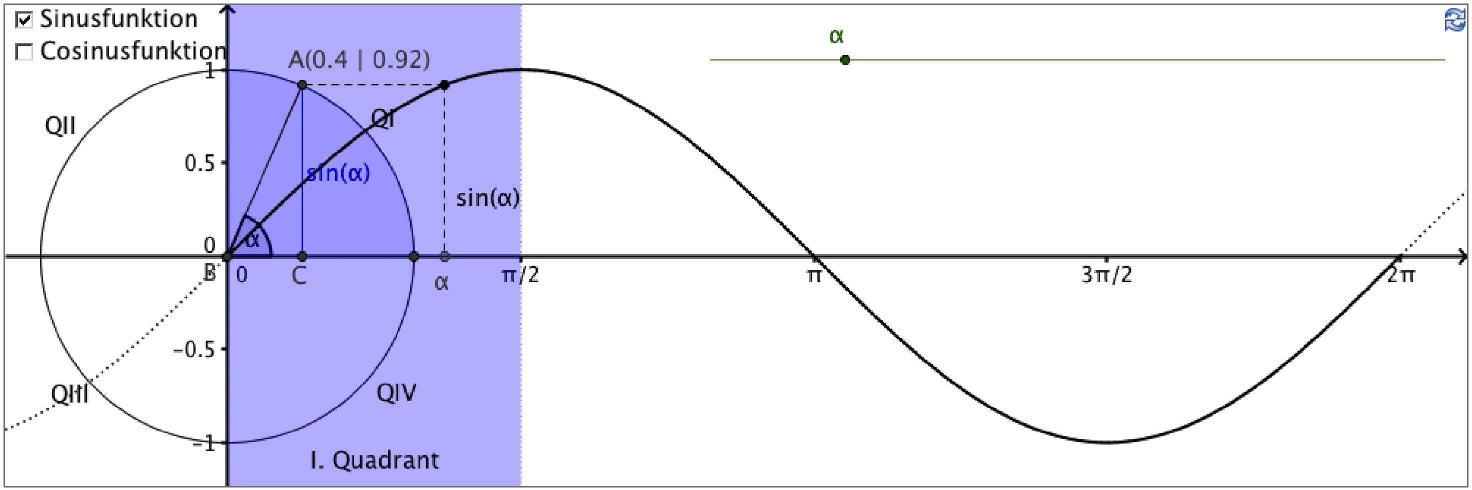


Digitalisierung - Abtasttheorem

Modell zur Entstehung sinusoider Signale

Enstehung über einen sich am Einheitskreis mit der Zeit drehenden Zeiger.

- **Bogenmaß**: Angabe des Winkels als Teil des Umfangs eines Einheitskreises (2π) z.B. (90° = $\pi/2$, 360° = 2π)
- **Kreisfrequenz** ω: Pro Sekunde von einem drehenden Zeiger
  $\omega = 2\pi \cdot f = 2\pi/T$ (Anmerkung: Winkelangabe in Radian, d.h. $2\pi = 360$)



Sinusoide Signale: Sinus plus variable Phasenverschiebung.

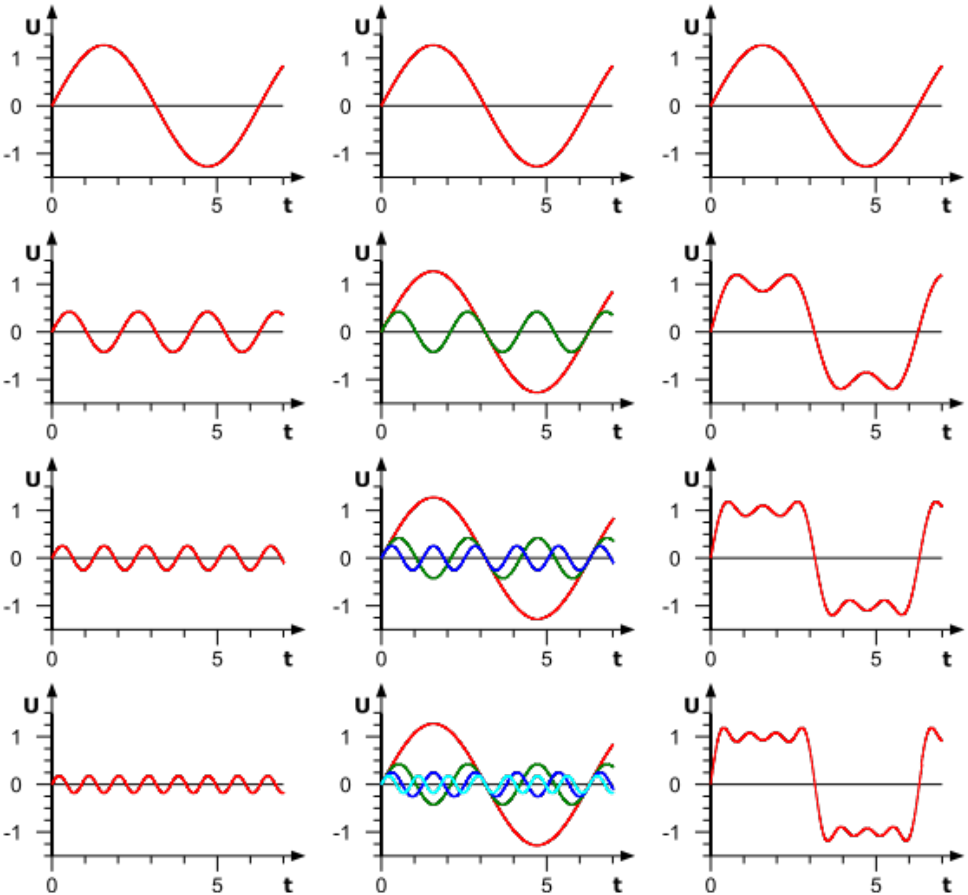
(Demo unter

http://www.wi.uni-muenster.de/qm/organisation/terveer/analysis_brueckenkurs/DGS_5/Sources/abb_5_7_sinusfunktion_genese.html)

Digitalisierung - Abtasttheorem

Überlagerung von sinusoider Signale

Wie sieht dies nun für komplexere Signale aus?

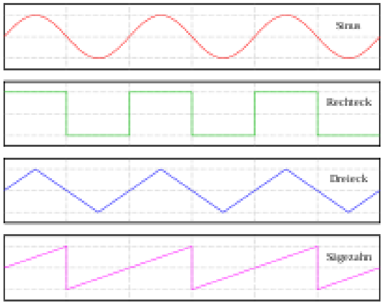


Bildquelle Wikipedia

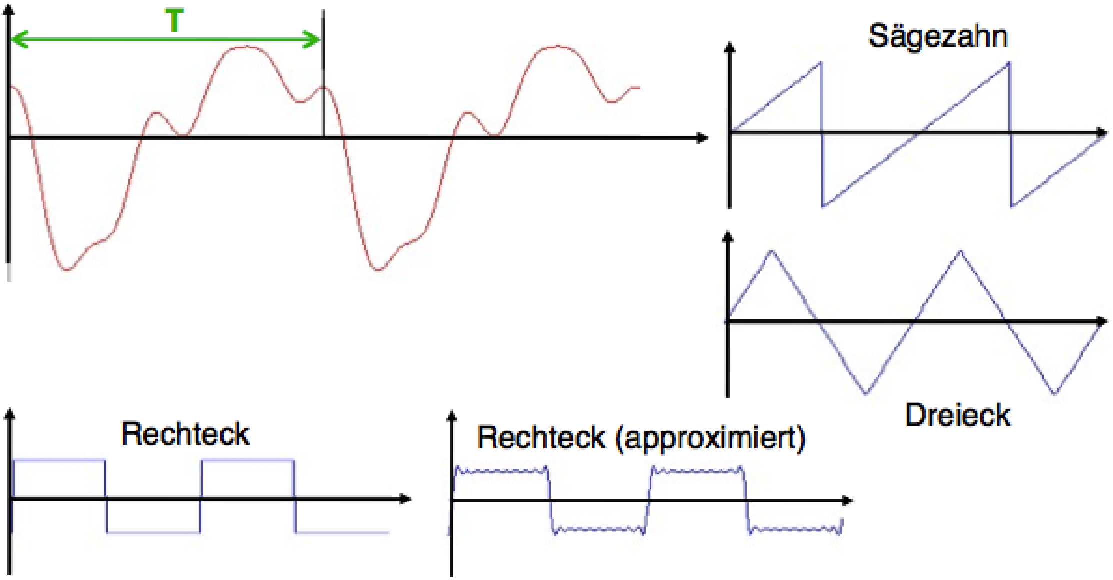
Fourier Theorem: Jedes periodische Signal kann durch eine potentiell unendliche Anzahl von Cosinus u. Sinus konstruiert werden. D.h. wir können uns bei der Analyse analoger Signale auf die Analyse sinusoider Funktionen beschränken.

Digitalisierung - Abtasttheorem

Beispiele periodischer Signale



Bildquelle Wikipedia



Quelle Prof. Butz, LMU

- Können die Originalsignale nach Digitalisierung rekonstruiert werden?
- Welche Effekte erzeugt eine zu niedrige Abtastrate?

Digitalisierung - Abtasttheorem

Was passiert bei zu geringer Abtastrate?

Beispiel eines komplexeren periodischen Beispielsignals

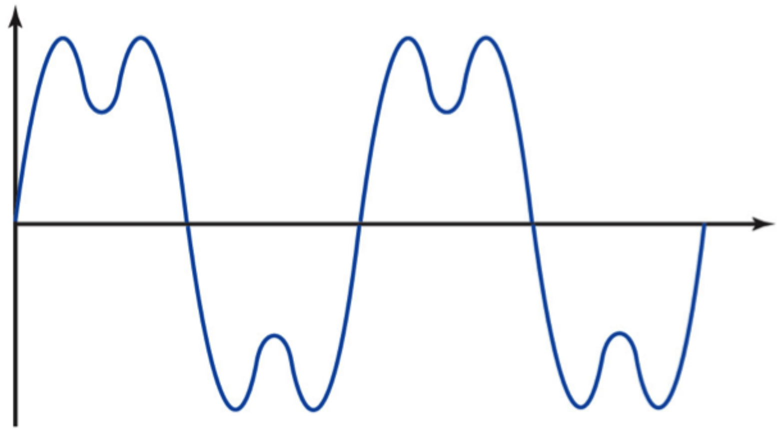


Abbildung 2.4: Ein periodisches Beispielsignal

Digitalisierung - Abtasttheorem

Was passiert bei zu geringer Abtastrate?

Zu niedrige Abtastrate

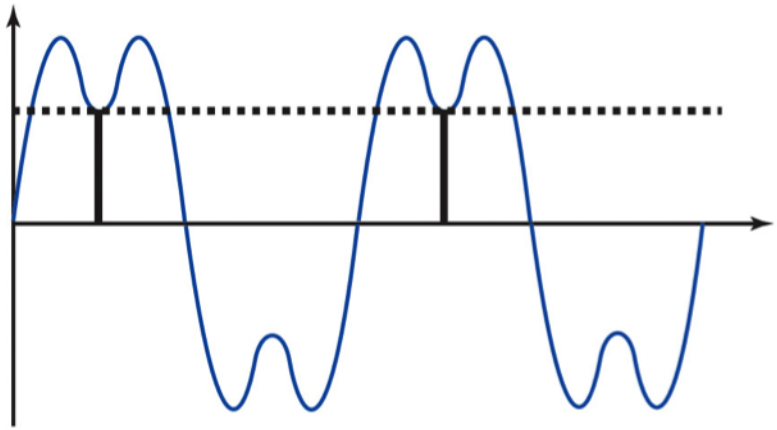


Abbildung 2.5: Abtastung mit zu niedriger Abtastrate I

Das Abastsignal ist als idealisierters periodisches Impulssignal (in schwarz) dargestellt und definiert die Diskretisierungseitpunkte.

Digitalisierung - Abtasttheorem

Was passiert bei zu geringer Abtastrate?

Zu niedrige Abtastrate

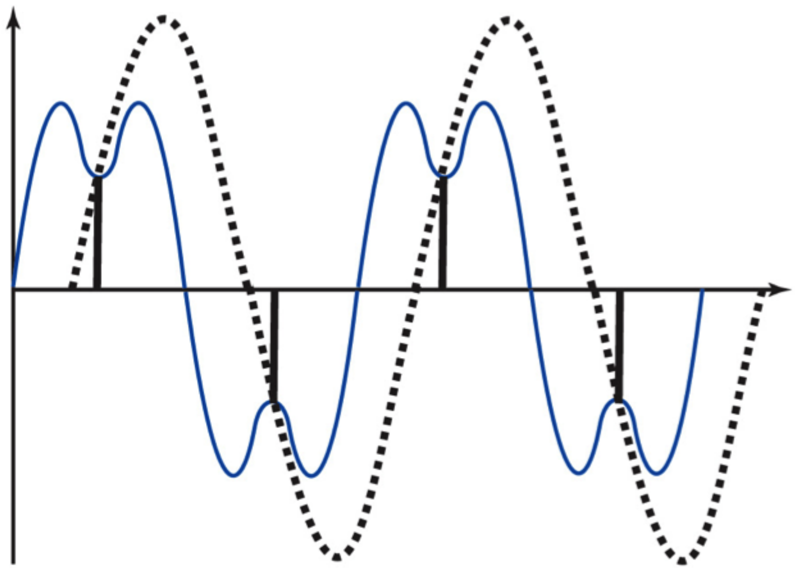
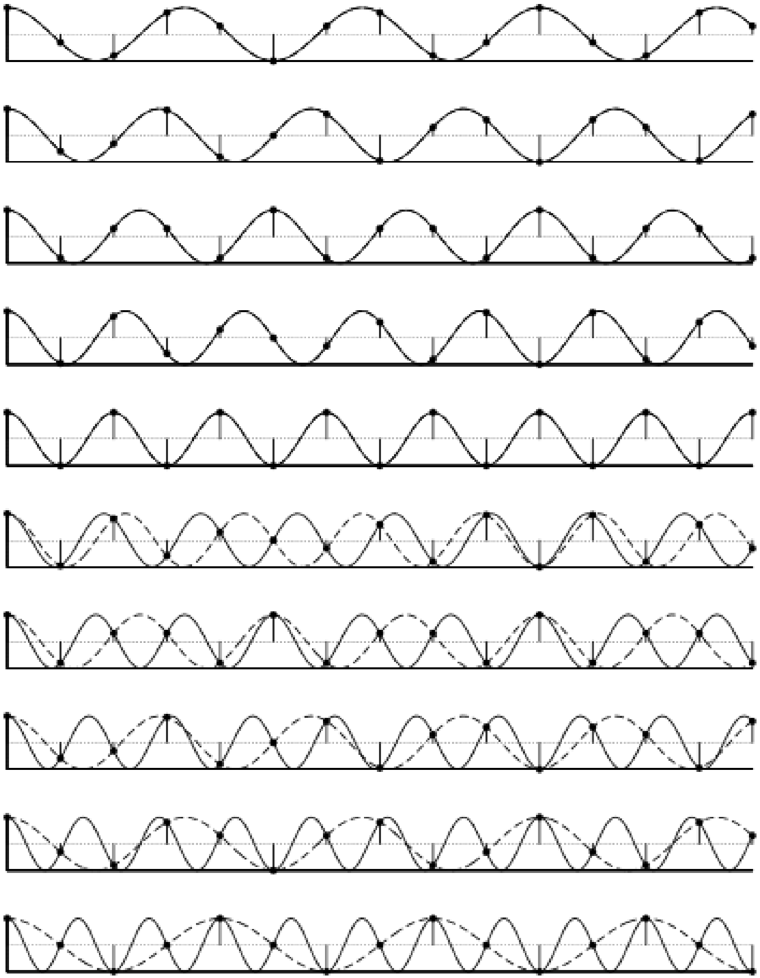


Abbildung 2.6: Abtastung mit zu niedriger Abtastrate II

Digitalisierung - Abtasttheorem

Was passiert bei zu geringer Abtastrate?



Bildquelle Wikipedia

- Fixe Abtastrate (Punkte)
- Steigende Signalfrequenz (von oben nach unten)
- Liegt die höchste Signalfrequenz über der halben Abtastfrequenz kann das Signal nicht mehr korrekt rekonstruiert werden (siehe gestrichelte Linie)

Digitalisierung - Abtasttheorem

Abtasttheorem (Nyquist-Shannon-Abtasttheorem)

Satz 1 (Abtasttheorem (Nyquist-Shannon-Abtasttheorem))
Wenn eine Funktion (Signal) mit höchster vorkommender Frequenz f_g (Bandbegrenzung) mit einer Abtastrate f_S abgetastet wird, so dass

$$f_S > 2 * f_g,$$

dann kann diese Funktion eindeutig aus den Abtastwerten rekonstruiert werden.

Entwickelt von Harry Nyquist (1928) und bewiesen durch Claude Shannon

Beispiele:

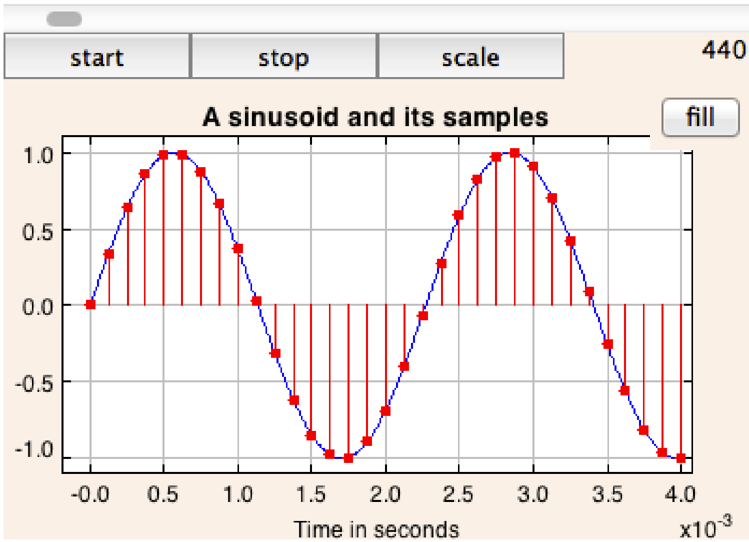
- Abtastrate für Audio CDs ist 44,1 kHz bei maximaler Höhrfrequenz des Menschen mit 20 kHz
- Telefonsprachverbindung (ISDN): 8 kHz (ausreichend für menschliche Sprache)

Digitalisierung - Abtasttheorem

Aliasing

Wird ein Signal mit zu geringer Frequenz abgetastet, spricht man vom „Aliasing“ (Alias ist der Stellvertreter)

Beispiel Audio: liegt die Frequenz des Signals über der halben Abtastfrequenz, sinkt die Frequenz des Digitalen Signals

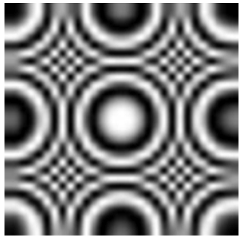


Applet unter http://www.aw-bc.com/lee_varaiya/protected/week13/aliasing.html

Digitalisierung - Abtasttheorem

Aliasing

Klassische Aliasing Effekte bei Bilddarstellung



Morie Effect: Ziegelwand mit Morie-Effekt entstanden durch nicht ausreichende Abtastfrequenz bei wellenförmiger Helligkeitsverläufe

Ohne Morie-Effekt

mit Morie-Effekt

Vermeidung von Aliasing durch Filterung aller Frequenzen im analog Signal mit höheren Anteilen als die Abtastrate $(\geq 0.5 f_s)$

Digitalisierung - Abtasttheorem

Speicherbedarf für Digitale Medien

Aus den Abtastraten lässt sich der Speicherbedarf von Digitalen Medien einfach ermitteln.

- Audio: 44 kHz pro Sekunde, 16 Bit pro Sample (=2^{16} unterschiedliche Töne) $\Rightarrow$ 1,4 MBit/s bzw. 635 MByte pro Stunde
- Bilder: 8 Millionen Pixel, 3 Farbkanäle, 8 bit pro Kanal $\Rightarrow$ 24 MByte pro Bild

Relativ hoher Speicherbedarf im Rohformat. Notwendigkeit für effiziente Kodierung und Kompression (siehe nächste Abschnitte)

Digitalisierung -Zusammenfassung

Zusammenfassung

Digitalisierung und Abtasttheorem

Gegeben Analoges Signal als veränderliche physikalische Größe:

- Diskretisierung der Ausbreitungsrichtung (räumlich und/oder zeitlich) eines Signals
- Quantisierung der Signalstärke

$\Rightarrow$ Digitales Signal

Zu berücksichtigen:

- Signale können als Überlagerung von potentiell unendlich vielen Sinus/Cosnius Signalen aufgefast werden (Fourier Theorem)
- Abtasttheorem/Nyquist-Frequenz: Mehr als doppelte Abtastfrequenz der höchsten im Signal vorkommenden Frequenz garantiert vollständige Rekonstruktion des Signals (unter der Annahme periodischer Signale und genügender Quantisierung)
- Aliasing entsteht, wenn das Abtasttheorem verletzt wird.

Kapitel EIC Teil 2- Medien und Information:I

I. EIC Teil 2- Medien und Information

- ❑ Digitalisierung
- ❑ Kodierung
- ❑ Informationstheorie
- ❑ Kompression

Kodierung - Repräsentation von Information

Kodierung - Repräsentation von Information

Was ist Information?

Information ist ein nichtstoffliches Phänomen, welches durch die Interpretation Zeichen entsteht, die man die Repräsentation der Information nennt. In der physikalischen Welt erfolgt diese Repräsentation durch analoge Signale.

In der Informatik repräsentiert man Information im Allgemeinen durch folgende Komponenten:

- **Zeichenvorrat:** Mengen von endlichen Zeichen zur Repräsentation von Information (z.B. Buchstaben bei Sprache, Zahlen bei Bild)
- **Nachricht:** Eine Sequenz von Zeichen aus einem Zeichenvorrat A bezeichnet man als Nachri cht.
- Die Menge aller Nachrichten wird mit A^* bezeichnet

Beispiele für Nachrichten

- Nachricht$=< 01010101 >$; Zeichenvorrat$= \{0, 1\}$
- Nachricht 1 $=< Medien >$, Nachricht 2 $=< nedi >$; Zeichenvorrat = $\{M, e, d, i, e, n\}$

Kodierung - Repräsentation von Information

Kodierung von Information

Definition 6 (Kodierung)

Seien A und B Zeichenvorräte. Eine Kodierung c ist eine Abbildung von Nachrichten in A auf Nachrichten in B

$$c : A \rightarrow B*$$

Beispiele:

- $ababa \rightarrow 0001000101001010101001010101$
- $127, 168, 24 \rightarrow \ Michael$
-  $\rightarrow 10$

Fragen:

- Welche Arten von Kodierungen kann man Unterscheiden?
- Was sind nun effiziente Kodierungen für verschiedene Informationen?
- Wieviel Information steckt in einer Nachricht?

Kodierung - Binärcodes und das binäre Zahlensystem

Kodierung - Binärcodes und das binäre Zahlensystem

Motivation

Digitale Geräte basieren auf digitalen Schaltungen welche zwei Zustände unterscheiden können:

- Strom Ein (oder 1)
- Strom Aus (oder 0)

Auf digitalen Geräten wird Information daher als Binärcode repräsentiert. Die Digitalisierung kann somit auch als Kodierung eines analogen Signales in einen Binärcode verstanden werden.

Kodierung - Binärcodes und das binäre Zahlensystem

Formale Definition

Definition 7 (Binärcode)

Ein Binärcode c ist eine Kodierung, bei dem der Ziel-Zeichenvorrat B (die Bildmenge) aus genau zwei Symbolen besteht. Diese werden als Bit (binary digit) bezeichnet.

- Das duales Zahlensystem bestehend aus dem Zeichenvorrat $A = \{0, 1\}$ ermöglicht arithmethische Operationen im Binärcode
- Die Boolesche Algebra (Aussagenlogik), bestehend aus dem Zeichenvorrag $A = \{Wahr, Falsch\}$, ermöglicht logische Operationen
- Alle weiterführenden Kodierungen, wie z.B. Zeichensätze, Audio, Bilder, Videos, Programme etc. werden auf binäre Codes zurückgeführt.

Kodierung - Binärcodes und das binäre Zahlensystem

Duales Zahlensystem

Fragestellung: Wie sieht ein Binärcode zur Repräsentation eines analogen Signals (z.B. Schalldruck) aus, sodass die arithmetische Operationen wie Addition, Subtraktion, Multiplikation erhalten bleiben?

Ein solcher Code ermöglicht beispielsweise die Addition zweier Signale oder die mathematische Analyse mittels digitalen Geräten (Bild- oder Videobearbeitung, digitale Signalanalyse, Musikkomposition etc.).

Kodierung - Binärcodes und das binäre Zahlensystem

Duales Zahlensystem - Definition

Beobachtung: Unser dezimales Zahlensystem ist ein Positionssystem mit der **Basis 10**, in dem eine Zahl in 10er Potenzen zerlegt wird.

Definition 8 (Positionszahlensystem)

Ein Positionszahlensystem mit der Basis B zerlegt ein Zahl $n =< b_0, b_2, \ldots, b_{N-1} >$ in Ziffern $< b_0, b_2, \ldots, b_{N-1} >$ mit N Stellen aufsteigend geordnet nach Potenzen von B. Der Wert der Zahl n kann als folgende Summe berechnet werden:

$$\sum_{i=0}^{N-1} b_i \cdot B^i$$

wobei gilt

- $B \in \mathbb{N}$ uynd $B \geq 2$
- $b_i \in \mathbb{N}_0$, $0 \geq b_i < B$

Kodierung - Binärcodes und das binäre Zahlensystem

Duales Zahlensystem - Beispiele

Mit Änderung der Basis ändert sich auch das Zahlensystem und die notwendige Anzahl der Ziffernzeichen:

- Dezimalsystem: $B = 10$, Ziffern $0 - 9$
- Binär- bzw. Dualsystem: $B = 2$, Ziffern $\{0, 1\}$
- Octalsystem: $B = 8$, Ziffern $0 - 7$
- Hexadezimalsystem: $B = 16$, Ziffern $\{0 - 9, A, B, C, D, E, F\}$

Beispiele:

- $n_2 = 1101$
  Zahlenwert $n_{10} = 1 * 2^3 + 1 * 2^2 + 0 * 2^1 + 1 * 2^0 = 1 * 8 + 1 * 4 + 0 * 2 + 1 * 1 = 13$
- $n_{16} = BAB$
  Zahlenwert $n_{10} = 11 * 16^2 + 10 * 16^1 + 11 * 16^0 = 11 * 256 + 10 * 16 + 11 * 1 = 2987$
- $n_8 = 3110$
  Zahlenwert $n_{10} = 3 * 8^3 + 1 * 8^2 + 1 * 8^1 + 0 * 8^0 = 1608$

Kodierung - Binärcodes und das binäre Zahlensystem

Duales Zahlensystem - Hex, Octal und Binär

Das Hexadezimal- und Oktalsystem stellen eine kürzere, einfach ermittelbare Schreibweise zum Binärsystem dar, da ihre Basen Potenzen von 2 darstellen. Eine Umrechnung ist einfach möglich, da Hexadezimalzahlen durch 4 Bit und Oktalzahlen durch 3 Bit repräsentiert werden können:

n_{10}	n_2	n_8	n_{16}
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

Kodierung - Binärcodes und das binäre Zahlensystem

Duales Zahlensystem - Umwandlung aus dem Dezimalsystem

Für die Umwandlung einer Dezimalzahl x in ein Zahlensystem mit der Basis B kann folgender Algorithmus verwendet werden:

1. Division durch Basis: $\frac{x}{B} = y \; Rest \; z$
2. Mache y zum neuen x
3. Solange x ungleich 0 gehe zu Schritt 1
4. Die ermittelten Reste z von unten nach oben nebeneinander (von links nach rechts) geschrieben ergeben die entsprechende Zahl zur Basis B

Warum?

Laut Hornerschema kann ein Zahl zur Basis B wie folgt zerlegt werden:

$$\sum_{i=0}^{N-1} b_i \cdot B^i = (\ldots(((b_N * B_N + b_{N-1}) * B + b_{N-2}) * B + \ldots + b_1) * B + b_0$$

Eine Division der obigen Summe durch B ergibt:

$y = (\ldots(((b_N * B_N + b_{N-1}) * B + b_{N-2}) * B + \ldots + b_1)$ und $z = b_0$

Kodierung - Binärcodes und das binäre Zahlensystem

Duales Zahlensystem - Umwandlung aus dem Dezimalsystem

Beispiel: $n_{10} = 68$ im Dualsystem?

x	y	z
68	34	0
34	17	0
17	8	1
8	4	0
4	2	0
2	2	0
1	2	1
0	-	-

$n_{10} = 68 \rightarrow n_2 = 1000100$

Kodierung - Rechnen im binären Zahlensystem

Kodierung - Rechnen im binären Zahlensystem

Arithmetische Operationen

Das binäre Zahlensystem erlaubt alle arithmetischen Operationen und ist somit äquivalent zum Dezimalsystem. Es gelten die gleichen Rechenregeln:

x	y	$x + y$	$x - y$	$x * y$	x/y
0	0	0	0	0	not defined (n.d.)
0	1	1	-1 bzw. 11[1]	0	0
1	0	1	1	0	not defined (n.d.)
1	1	10	0	1	1

Bei der Subtraktion entspricht 11 der B-Komplementdarstellung, die auf den nächsten Folien erklärt wird.

Kodierung - Rechnen im binären Zahlensystem

Arithmetische Operationen

Beispiel Addition mehrstelliger Zahlen:

x	10011011	(155)
y	00111110	(62)
Übertrag	01111100	
Ergebnis	11011001	(217)

Wichtig: der Übertrag erfolgt immer wenn die Summe größer 1 wird

Kodierung - Rechnen im binären Zahlensystem

Arithmetische Operationen

Zur Subtraktion müssen wir noch überlegen, wie negative Zahlen dargestellt werden können. Im Normalfall werden negative Zahlen durch ihren Betrag mit vorangestellten Minus dargestellt, worauf ein Bit verwendet werden könnte. Um jedoch die Subtraktion rechnerisch mittels Addition abzubilden (und somit die Rechnerarchitektur zu vereinfachen), wird auf die so genannten **Zweierkomplementdarstellung** zurückgegriffen.

In der Zweierkomplementdarstellung (oder auch B-Komplement darstellung wobei B die Basis, im dual System also 2, angibt) werden negative Zahlen durch ein auf 1 gesetztes führendes Bit dargestellt. Die positiven Zahlen inklusive 0 werden durch ein auf 0 gesetztes führendes Bit dargestellt.

Kodierung - Rechnen im binären Zahlensystem

Arithmetische Operationen

Zweierkomplement: Beispiel mit drei Bit:

Dezimal	B-Komplement	mit Vorzeichen Bit
-4	100	NA
-3	101	111
-2	110	110
-1	111	101
0	000	000 od. 100
1	001	001
2	010	010
3	011	011

Wie erkennbar “verschwendet” das Vorzeichenbit zwei Binäerzahlen zur Kodierung der 0.

Kodierung - Rechnen im binären Zahlensystem

Arithmetische Operationen

Das Zweierkomplement wird wie folgt erzeugt:

1. Konvertiere die Zahl ohne Vorzeichen in das Binärsystem
2. Invertiere die Bits der konvertierten Zahl
3. Addiere 1 zur invertierten Zahl.

Beispiel für -5:

Umwandeln	0101
Invertieren	1010
+1	1011

Kodierung - Rechnen im binären Zahlensystem

Arithmetische Operationen

Beispiel Subtraktion mehrstelliger Zahlen: 155-62.

1. Wir rechnen 155 + (-62).
2. Zweierkomplementdarstellung von 62:

Umwandeln	000111110
Invertieren	111000001
+1	111000010

3. Durchführen der Addition:

x	010011011	(155)
y	111000010	(-62)
Übertrag	100000100	Das führende Übertragsbit wird verworfen
Ergebnis	001011101	(93)

Kodierung - ASCII Code

Kodierung - ASCII Code

ASCII Code

Eine ander Art der Kodierung sind Zeichensatzkodierungen, wie der **American Standard Code for Information Interchange (kurz ASCII Code).**

Der Zeichensatz kodiert mit 7-Bit das lateinische Alphabet in Groß- und Kleinschreibung, die arabischen Ziffern sowie Satzzeichen und Sonderzeichen. Er beinhaltet alle wesentlichen Zeichen einer Tastatur.

Jedem Zeichen des Zeichensatzes wird dabei ein Binärcode per Definition zugewiesen. D.h. es gibt 128 verschiedene Bitmuster.

Kodierung - ASCII Code

ASCII Code

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

Source: www.LookupTables.com

Bildquelle `http://www.asciitable.com/`

Kodierung -Zusammenfassung

Zusammenfassung

Kodierung

- Repräsentation von Information mittels Zeichen und Nachrichten
- Kodierung als Vorschrift zur Überführung Nachrichten aus einem Zeichensatz in den anderen.
- Binärcodes bestehen aus genau zwei Zeichen, welche unterschiedliche interpretiert werden können:
  - Als Zahl im dualen Zahlensystem
    - Positionszahlensystem zur Basis 2
    - Arithmetische Operationen
    - Subtraktion als Addition mittels Zweierkomplement
  - Als Zeichen einer Tastatur (ASCII Code)
  - Als Wahrheitswerte (Wahr/Falsch) im Bereich der Aussagenlogik

Bibliographie

- **Malaka, Butz, Hussmann (2009)** - Medieninformatik: Eine Einführung (Pearson Studium - IT), Kapitel 2.4
- **Herold, Lurz, Wohlrab** - Grundlagen der Informatik, Pearson Studium, Kapitel 3

Kapitel EIC Teil 2- Medien und Information:I

I. EIC Teil 2- Medien und Information

- ❑ Digitalisierung
- ❑ Kodierung
- ❑ Informationstheorie
- ❑ Kompression

Informationstheorie - Ein einführendes Beispiel

Informationstheorie - Ein einführendes Beispiel

Problemstellung

Wir haben unterschiedliche Kodierungen mit verschiedenen Eigenschaften kennen gelernt.

Doch wann ist eine Kodierung optimial, d.h. existiert eine Kodierung die mit weniger Bit die gleiche Information übermitteln kann?

Wie kann man die Menge der in einer Nachricht erhaltenen Information quantifizieren?

Wie kann man den Informationsgehalt einer Nachrichtenquellen quantifizieren?

Informationstheorie - Ein einführendes Beispiel

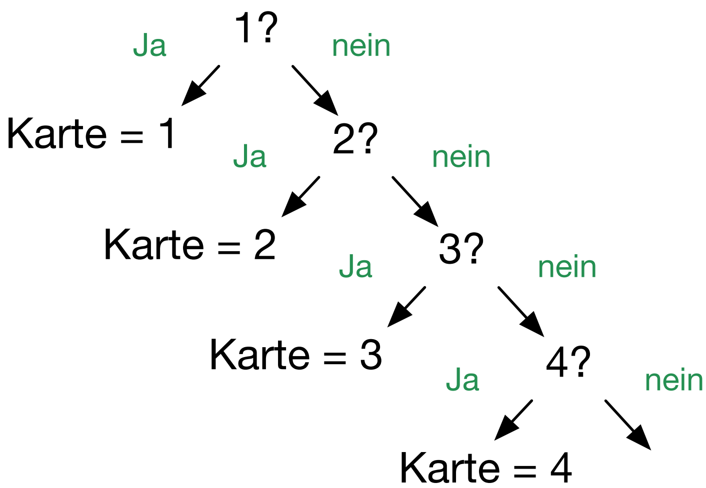
Beispiel Kartenspiel

Nehmen wir an sie haben ein Kartenspiel bestehend aus 4 Karten, $K = \{1, 2, 3, 4\}$. Person A zieht eine Karte. Person B muss über möglichst wenige "Ja/Nein" Antworten herausfinden, um welche Karte es sich handelt.

Informationstheorie - Ein einführendes Beispiel

Beispiel Kartenspiel

Nehmen wir an sie haben ein Kartenspiel bestehend aus 4 Karten, $K = \{1, 2, 3, 4\}$. Person A zieht eine Karte. Person B muss über möglichst wenige “Ja/Nein” Antworten herausfinden, um welche Karte es sich handelt.

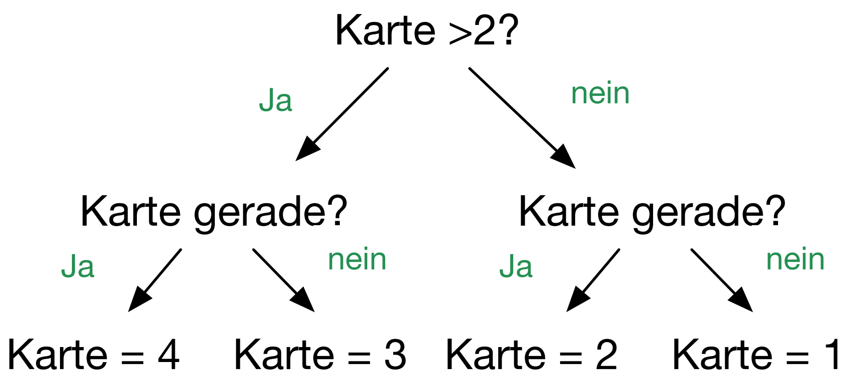


Beispiel Fragebaum A

Informationstheorie - Ein einführendes Beispiel

Beispiel Kartenspiel

Nehmen wir an sie haben ein Kartenspiel bestehend aus 4 Karten, $K = \{1, 2, 3, 4\}$. Person A zieht eine Karte. Person B muss über möglichst wenige “Ja/Nein” Antworten herausfinden, um welche Karte es sich handelt.



Beispiel Fragebaum B

Informationstheorie - Ein einführendes Beispiel

Beispiel Kartenspiel

Jede “Ja/Nein” Antwort kann als Bit (“0/1”) verstanden werden, wodurch sich folgender Binärcode ergibt:

Karte	**Code A**	**Code B**	**Informationsgehalt**
1	1	00	2
2	10	01	2
3	100	10	2
4	1000	11	2

Beobachtungen:

- Code B scheint effizienter, da er im Schnitt weniger Fragen (Bits) benötigt.
- Code B scheint optimal zu sein, da wir für 4 Zustände 2 Bits benötigen ($N = 2^{\#bits}$ bzw. $\#bits = \log_2(N)$ mit N als Anzahl der Karten)
- Alternativ: In Code B reduziert jede Frage die Unsicherheit um 50 % d.h. die Anzahl der verbleibenden Möglichkeiten wird um die Hälfte reduziert
- Der Informationsgehalt einer Karte kann als minimale Anzahl von Ja/Nein Fragen aufgefasst werden.

Informationstheorie - Ein einführendes Beispiel

Beispiel Kartenspiel

Ein neues Spiel hat folgende 4 Karten: $K = \{1, 1, 2, 3\}$, d.h. Karte 1 kommt 2x vor.
Wie ändert sich der Code?

Informationstheorie - Ein einführendes Beispiel

Beispiel Kartenspiel

Ein neues Spiel hat folgende 4 Karten: $K = \{1, 1, 2, 3\}$, d.h. Karte 1 kommt 2x vor.

Karte	Code B	Code C	Informationsgehalt
1	00	0	1
1	01	0	1
3	10	01	2
4	11	11	2

Fragen:

- **Bit 0**: Ist es nicht 1?
- **Bit 1**: Ist es 4?

Der Code ist optimal, da beide Fragen die Unsicherheit um 50% reduzieren, d.h. die Anzahl der verbleibenden Möglichkeiten auf die Hälfte reduzieren.

Informationstheorie - Ein einführendes Beispiel

Beispiel Kartenspiel

Ein neues Spiel hat folgende 4 Karten: $K = \{1, 1, 2, 3\}$, d.h. Karte 1 kommt 2x vor.

Karte	Code B	Code C	Informationsgehalt
1	00	0	1
1	01	0	1
3	10	01	2
4	11	11	2

Beobachtungen:

- Code C scheint effizienter, da er im Schnitt weniger Fragen (Bits) benötigt.
- 2 Karten haben dieselbe Bedeutung.
  - Da es wahrscheinlicher ist, Karte 1 zu ziehen, ist der Informationsgehalt von Karte 1 jedoch geringer.
  - Alternativ: Da Karte 1 öfters gezogen wird, wollen wir der Karte einen kürzeren Code geben.

Informationstheorie - Ein einführendes Beispiel

Beispiel Kartenspiel

Ein neues Spiel hat folgende 4 Karten: $K = \{1, 1, 2, 3\}$, d.h. Karte 1 kommt 2x vor.

Karte	Code B	Code C	Informationsgehalt
1	00	0	1
1	01	0	1
3	10	01	2
4	11	11	2

Wie hoch ist der Informationsgehalt von Karte 1?

- Für 4 Karten benotigen wir 2 bit ($\log_2(4)$).
- Karte 1 würde dabei 2 Codes erhalten, die jedoch die gleiche Bedeutung haben und somit redundant sind. D.h. 1 Frage/Bit ($\log_2(2)$) ist redundant.
- $IG(Karte_1) = \log_2(4) - \log_2(2) = \log_2(\frac{4}{2}) = \log_2(\frac{1}{p(Karte_1)})$ wobei $p(Karte_1)$ die Wahrscheinlichkeit darstellt, dass $Karte_1$ gezogen wird.
- Analog f. Karte 3: $IG(Karte_3) = \log_2(4) - \log_2(1) = \log_2(\frac{4}{1}) = \log_2(\frac{1}{p(Karte_3)})$

Informationstheorie - Formale Definition

Informationstheorie - Formale Definition

Überblick

- Die Informationstheorie nach Shannon wurde 1948 von **Claude Shannon** entwickelt.
- Sie stellt ein **mathematisches Modell** für Kommunikationssysteme dar.
- Dabei werden Nachrichten als **stochastische Ereignisse**, die von einer Nachrichtenquelle, dem Sender, ausgesendet und von einem Empfänger empfangen werden.
  - Stochastische Prozesse sind zeitlich geordnete, zufällige Prozesse die mit Hilfe der Statistik und Wahrscheinlichkeitsrechnung beschrieben werden
  - Nachrichten bestehen somit aus einer endlichen Anzahl diskreter Zeichen, welche mit einer gewissen Wahrscheinlichkeit auftreten.

Informationstheorie - Formale Definition

Überblick

Kernelemente der Informationstheorie:

- Definition einer **Nachrichtenquellen** und deren Eigenschaften
- Definition des **Informationsgehaltes bzw. der Information eines Zeichens,** $I(x)$
- Definition der **Entropie** $H(X)$, d.h. der Informationsdichte bzw. des durchschnittlichen Informationsgehaltes einer Nachrichtenquelle
- Definition von **optimalen Kodierungen** und darauf aufbauender **Redundanz**

Informationstheorie - Formale Definition

Nachrichtenquelle

Definition 9 (Diskrete, gedächtnislose Nachrichtenquelle)

Eine diskrete, gedächtnislose Nachrichtenquelle setzt in jedem Zeittakt ein Zeichen $a_i \in A$ aus einem Zeichenvorrat A mit der Wahrscheinlicheit $P(a_i) = p_i$ ab. Die Auswahl der Zeichen ist unabhängig von bereits emitierten Zeichen in der Vergangenheit.

Beispiel mit 3 Quellen, Zeichenvorrat $\{A, B, C, D\}$:

Zeichen a	A	B	C	D
Wahrscheinlichkeit p_a in Quelle 1	1.0	0.0	0.0	0.0
Wahrscheinlichkeit p_a in Quelle 2	0.25	0.25	0.25	0.25
Wahrscheinlichkeit p_a in Quelle 3	0.5	0.25	0.125	0.125

- Welche Quelle überträgt die meiste Information in einer Nachricht?
- Wieviel Bit benötigen wir zur Kodierung eines Zeichens einer Quelle?
- Wie hoch ist die Information (bzw. der Informationsgehalt) eines Zeichens?

Informationstheorie - Formale Definition

Informationsgehalt eines Zeichen

Axiomatische Definition[2] des Informationsgehalts über 3 Axiome (grundsätzliche Eigenschaften):

(I) Der Informationsgehalt eines Zeichens $a \in A$ mit der Wahrscheinlicheit p_i ist ein nicht-negatives Maß, welches nur von der Wahrscheinlichkeit des Zeichens abhängt: $I(p_i) \geq 0$

(II) Der Informationsgehalt zweier voneinander unabhängiger Zeichen a_i und a_j mit den Wahrscheinlichkeiten p_i und p_j addiert sich: $I(p_i, p_j) = I(p_i) + I(p_j)$

(III) Der Informationsgehalt ist eine stetige Funktion der Wahrscheinlichkeiten der Zeichen.

[2] Axiome sind die Grundsätze bzw. fixen Annahmen einer Theorie. In einer Axiomatisch Definition werden dies a-priori festgelegt. Aus den Axiomen werden anschließend deduktiv weitere Eigenschaften abgeleitet.

Informationstheorie - Formale Definition

Informationsgehalt eines Zeichen

Satz 2 (Informationsgehalt/Entscheidungsgehalt eines Zeichens)

Sei p_i die Wahrscheinlichkeit für das Auftreten von Zeichen $a_i \in A$. Der Entscheidungsgehalt x_i für das Zeichen $a_i \in A$ ist definiert als

$$2^{x_a} = 1/p_a \Rightarrow x_a = \log_2(1/p_a)$$

Die Einheit ist bit (basic indissoluble[3] information unit) pro Zeichen. n bit entsprechen dabei mindestens n Bit (Binary Digits)

Zeichen a	A	B	C	D
p_a in Quelle 1	1.0	0.0	0.0	0.0
x_a (in bit) in Quelle 1	0	undef.	undef.	undef.
p_a in Quelle 2	0.25	0.25	0.25	0.25
x_a (in bit)in Quelle 2	2	2	2	2
p_a in Quelle 3	0.5	0.25	0.125	0.125
x_a (in bit)in Quelle	1	2	3	3

[3] unauflöslich

Informationstheorie - Formale Definition

Informationsgehalt eines Zeichen

Satz 3 (Informationsgehalt/Entscheidungsgehalt eines Zeichens)

Sei p_i die Wahrscheinlichkeit für das Auftreten von Zeichen $a_i \in A$. Der Entscheidungsgehalt x_i für das Zeichen $a_i \in A$ ist definiert als

$$2^{x_a} = 1/p_a \Rightarrow x_a = \log_2(1/p_a)$$

Die Einheit ist bit (basic indissoluble[4] information unit) pro Zeichen. n bit entsprechen dabei mindestens n Bit (Binary Digits)

Eigenschaften:

- Ein sicheres Ereignis ($p_i = 1.0$) besitzt keine Information.
- Unwahrscheinliche Ereignisse besitzen viel Information.
- Der Informationsgehalt spiegelt die Axiome wider, wobei der Logarithmus für die Additivität (Axiom II) sorgt
- Die Basis des Logarithmus definiert die Einheit. Bei einem $\log_{10}$ wüerden Zeichen in einem Dezimalcode kodiert werden.
- Ein bit definiert die kleinste Einheit und kann als “Ja/Nein” Entscheidung verstanden werden.

[4] unauflöslich

Informationstheorie - Formale Definition

Durchschnittliche Informationsgehalt einer Quelle

Definition 10 (Entropie (durchschnittlichen Entscheidungsgehalt))
Sei A ein Zeichenvorrat einer diskreten, gedächtnislosen Quelle und sei p_i die Wahrscheinlichkeit für das Auftreten von Zeichen $a_i \in A$. Dann definiert die Entropie H den *durchschnittlichen Informationsgehalt der Quelle* als:

$$H = \sum_{a \in A} p_a * \log_2(\frac{1}{p_a})$$

Bemerkungen

- $p_a * \log_2(\frac{1}{p_a}) = p_a * x_a$ bedeutet das der Entscheidungsgehalt eines Zeichens noch mit dessen Auftrittswahrscheinlichkeit gewichtet wird
  - Seltene Zeichen besitzen hohe Information, werden jedoch nicht oft gesendet.
  - Häufige Zeichen besitzen niedrige Information, werden jedoch häufig gesendet.
- Die Entropie gibt die minimale Anzahl von bit's an die notwendig sind, um Nachrichten einer diskreten, gedächtnislosen Quelle zu kodieren. Es gibt keine Kodierung mit weniger bit's.
- Entropie kann im physikalischen Sinn der Thermodynamik auch als Unordnung eines Systems verstanden werden.
- Alternative Darstellung: $-p_a * \log_2 \mathbf{p_a}$

Frage: Wann ist die Entropie maximal?

Informationstheorie - Formale Definition

Informationsgehalt einer Quelle

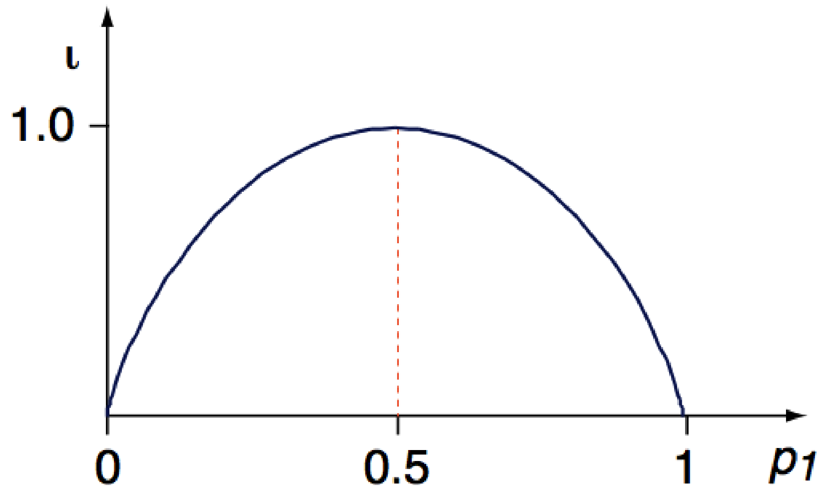
Satz 4 (Maximalität der Entropie)

Sei A ein Zeichenvorrat einer Quelle und p_a die Wahrscheinlichkeit für das Auftreten von Zeichen $a \in A$. Die **Entropie-Funktion ist maximal** wenn gilt

$$\forall_{i,j \in A} | p_i = p_j$$

d.h. die **Auftrittswahrscheinlichkeit für alle Zeichen gleich groß ist.**

Darstellung für den Fall von zwei Zeichen A, B ($p_a = 1 - p_b$):



Informationstheorie - Formale Definition

Beispiel Informationsgehalt einer Quelle

Zeichen a	A	B	C	D
p_a in Quelle 1	1.0	0.0	0.0	0.0
x_a (in bit) in Quelle 1	0	undef.	undef.	undef.
p_a in Quelle 2	0.25	0.25	0.25	0.25
x_a (in bit)in Quelle 2	2	2	2	2
p_a in Quelle 3	0.5	0.25	0.125	0.125
x_a (in bit)in Quelle 3	1	2	3	3

Entropie:

- $H_{Quelle1} = 0$
- $H_{Quelle2} = 2$
- $H_{Quelle3} = 1.75$

Redundanz

Redundanz

Redundanz einer Kodierung

Was bedeutet Redundanz?

- Redundanz bedeutet, daß es überflüssige Nachrichtenteile gibt
- Redundante Kodierungen sind nicht optimal. Dies kann gewollt sein (z.B. zur Fehlerkorrektur) oder ungewollt, wodurch mehr Speicher als nötig verwendet wird.

Wie hoch ist die Redundanz einer gegebenen Kodierung c für eine Nachrichtenquelle?

Intuitive Vorgehensweiße:

- Ermittle die durchschnittliche (Wort)Länge einer Kodierung in Bits (L)
- Vergleiche diese mit dem Informationsgehalt (Entropie) einer Quelle (H)

Redundanz

Wortlänge

Was ist die durchschnittliche Länge einer Kodierung in Bits?

Definition 11 (Wortlänge (Nachrichtenlänge))

Sie A^* die Menge aller endlichen Sequenzen/Worte (Nachrichten) aus einem Zeichenvorrat A.
Für ein Wort (Nachricht) $w \in A^*$ ist die Länge die Anzahl der in dem Wort (der Nachricht) enthaltenen Zeichen, abgekürzt durch $|w|$.
Wenn eine Kodierung c einem Zeichen $a \in A$ ein Wort $c(a) \in B^*$ zuweist, dann ist $|c(a)|$ die **Wortlänge der Kodierung** des Zeichens a

Beispiele

- $w_1 =< 010101 >\Rightarrow |w| = 6$
- $c_1(w_1) = c_1(010101) =< ab >\Rightarrow |w| = 2$
- $c_2(w_1) = c_2(010101) =< Medien >\Rightarrow |w| = 6$

Die Einheit der Wortlänge eines Binärcodes ist Bit.

Redundanz

Wortlänge

Definition 12 (Durchschnittliche Wortlänge)

Bei einer Kodierung c einer Nachrichtenqulle ist die **durchschnittliche Wortlänge** L die nach Auftrittswahrscheinlichkeit gewichtete Summe der Wortlängen aller Kodierungen der Einzelzeichen, als Formel:

$$L = \sum_{a \in A} p_a * |c(a)|$$

Zusammenhang zur Entropie/Informationsgehalt:

- $|c(a)|$ ist in Bit angegeben (wenn wir von einem Binärcode ausgehen)
- $\log_2(\frac{1}{p_a})$ ist der Informationsgehalt eines Zeichens, ebenfalls in Bit angegeben.
- Der Informationsgehalt stellt die Bits eines idealen Code dar, während $|c(a)|$ die Bitlänge eines echten Codes darstellt

Redundanz

Beispiel Redundanz

Beispiel für zwei Kodierungen c_1 und c_2

Zeichen a	**A**	**B**	**C**	**D**
Kodierung c_1	00	01	10	11
Kodierung c_2	0	10	110	111

Beispiel für eine redundante Kodierungen c_1

Zeichen a	**A**	**B**	**C**	**D**
Wahrscheinlichkeit p_a in Quelle 3	0.5	0.25	0.125	0.125
Kodierung c_1	00	01	10	11
Wortlänge	2	2	2	2

Durchschnittliche Wortlänge $L_{c_1} = 0.5 * 2 + 0.25 * 2 + 0.125 * 2 + 0.125 * 2 = 2$

Informationsgehalt: $H_3 = 0.5 * 1 + 0.25 * 2 + 0.125 * 3 + 0.125 * 3 = 1.75$

Redundanz

Definition Redundanz

Definition 13 (Redundanz R)

Die **Redundanz** R einer binären Kodierung c für eine Nachrichtenquelle ist die Differenz der mittleren Wortlänge und der Entropie:

$$R = L - H$$

Optimale Kodierung: Eine Kodierung heißt optimal, wenn die Redundanz gleich Null ist ($R = 0$)

Zeichen a	**A**	**B**	**C**	**D**
Wahrscheinlichkeit p_a in Quelle 3	0.5	0.25	0.125	0.125
Kodierung c_2	0	10	110	111
Wortlänge	1	2	3	3

Durchschnittliche Wortlänge $L_{c_2} = 0.5 * 1 + 0.25 * 2 + 0.125 * 3 + 0.125 * 3 = 1.75$
Informationsgehalt: $H_3 = 0.5 * 1 + 0.25 * 2 + 0.125 * 3 + 0.125 * 3 = 1.75$

$\Rightarrow$ Eine gute Kodierung berücksichtigt immer die Verteilung der Zeichen in der Nachrichtenquelle

$\Rightarrow$ Mit Wissen über Medieneigenschaften und wie sie wahrgenommen werden, können gute Kodierungen erreicht werden (JPEG, GIF, MP3 etc.)

Zusammenfassung

Informationstheorie

- Wir wissen nun was Information ist (Zeichenvorrat, Nachricht, Kodierung)
- Wir können den Informationsgehalt einer Nachricht bestimmen
- Wir können feststelle ob Nachrichten redundante Information enthalten

Gibt es eine Möglichkeit, optimale Codes zu erzeugen und beliebige Kodierungen zu „Komprimieren“?

Bibliographie

- **Malaka, Butz, Hussmann (2009)** - Medieninformatik: Eine Einführung (Pearson Studium - IT), Kapitel 2.4
- **Daugmann, John (2012)**- Lecture Notes on Information Theory and Coding, chapter 3
  `http://www.cl.cam.ac.uk/teaching/0809/InfoTheory/InfoTheoryLectures.pdf`

Kapitel EIC Teil 2- Medien und Information:I

I. EIC Teil 2- Medien und Information

- Digitalisierung
- Kodierung
- Informationstheorie
- Kompression

Kompression

Lernziel

Unterthemen

- Kompressionsarten und genereller Prozess
- Lauflängenkodierung
- Huffman Kodierung - Entropie-basiert
- Arithmetische Kodierung
- LZW Kodierung - Wörterbuch-basiert

Fragestellungen

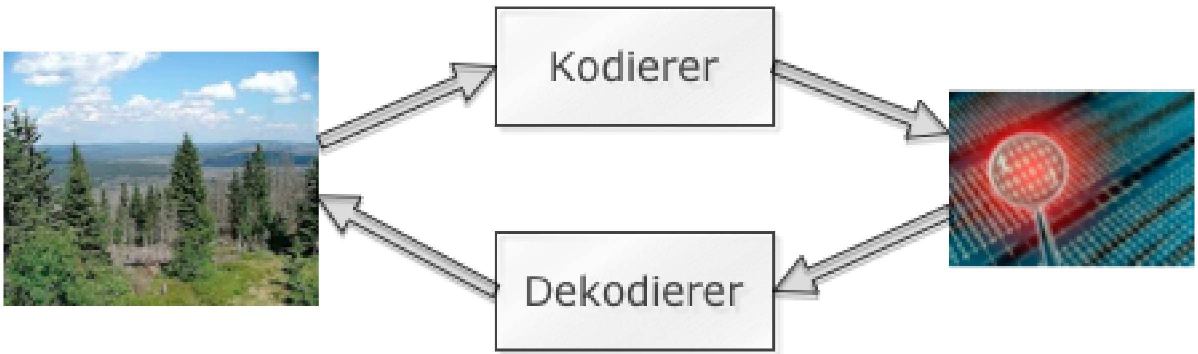
- Wie kann Information komprimiert werden?
- Wie nutzt man Entropie zur Komprimierung?
- Welche Kompressionsverfahren gibt es?

Kompressionsverfahren

Kompressionsverfahren

Prozess der Kompression/Dekompression

- **Kompression:** Kodierung der Ausgangsdaten in neuen, kleineren Code
- **Dekompression:** Dekodierung des kleineren Code in die Ausgangsdaten
- Kompressionscode verwendet i.a. variable lange Codewörter, um unterschiedlichen Frequenzen der Zeichen im Originalcode zu berücksichtigen



Kompressionsverfahren

Definition und Klassifikation

Definition 14 (Kompression)

Unter Kompression versteht man die Kodierung eines Datenstroms mit einem neuem, i.a. zu errechnenden Code geringerer Redundanz, sodass die Originaldaten wieder herstellbar sind. Eine optimale Kompression erzeugt eine optimale Kodierung.

Klassifikation nach **Anwendungsdomäne**

- **Universelle Kompressionsverfahren:** Treffen keine Annahmen über Daten oder Domäne und sind daher universell auf jeden Datenstrom anwendbar
- **Speziell Kompressionsverfahren:** Treffen Annahme über Daten bzw. Domäne. Sind daher nur beschränkt einsetzbar, arbeiten i.A. aber auf dieser Domäne effizienter

Klassifikation nach **Erhaltung der Information**

- **Verlustfreie Kompression:** Der Originaldatenstrom kann wieder rekonstruiert werden
- **Verlustbehaftete Kompression:** Der Originaldatenstrom kann nicht mehr rekonstruiert werden. Information geht verloren, dafür erreicht man i.A. bessere Kompressionsraten

Verlustbehaftete Verfahren werden bei den entsprechenden Medien behandelt

Kompressionsverfahren

Überblick verlustfreie Kompressionsverfahren

Statistische bzw. Entropie-basierte Verfahren

- Huffman-Kodierung - Optimale Kodierung im Shannon'schen Sinn
- Arithmetische Kodierung - Erweiterung von Huffman

Zeichenorientierte Verfahren

- Lauflängen Kodierung (Run Length Encoding)
- LZW-Kodierung (Wörterbuch-basierter Ansatz)

	Universelle Verfahren	**Spezielle Verfahren**
Verlustfreie Verfahren	z.B. Huffman, LZW	z.B. PNG, AIFF
Verlustbehaftete Verfahren	(nicht sinnvoll)	z.B. JPEG, MP3

Kompressionsverfahren

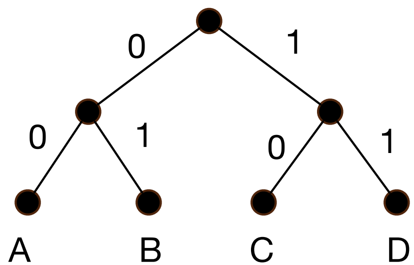
Code-Bäume und Vergleich Codes variabler/nicht-variabler Länge

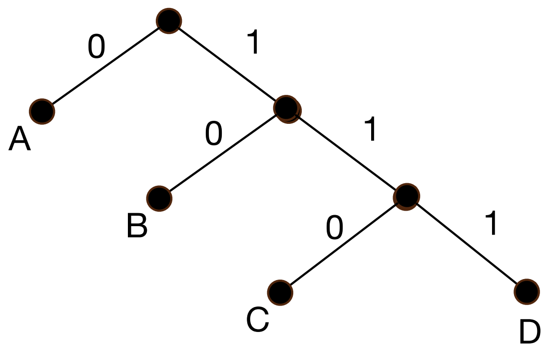
Arten v. Codes

- Codes fixer Länge: Jedes Zeichen wird mit der gleichen Anzahl an Bits kodiert (i.a. hohe Redundanz)
- Codes variabler Länge: Jedes Zeichen wird mit unterschiedlichen Anzahl an Bits kodiert

Zeichen a	**A**	**B**	**C**	**D**
Kodierung c_1	00	01	10	11
Kodierung c_2	0	10	110	111

Code-Baum: Darstellung der Kodierung als Baumstruktur





c_1 links und c_2 rechts

Kompressionsverfahren

Codes mit variabler Codelänge

Wie trennt man jedoch Zeichen in Codes variabler Bitlänge?

Beispiel: Morse-Code Codebaum

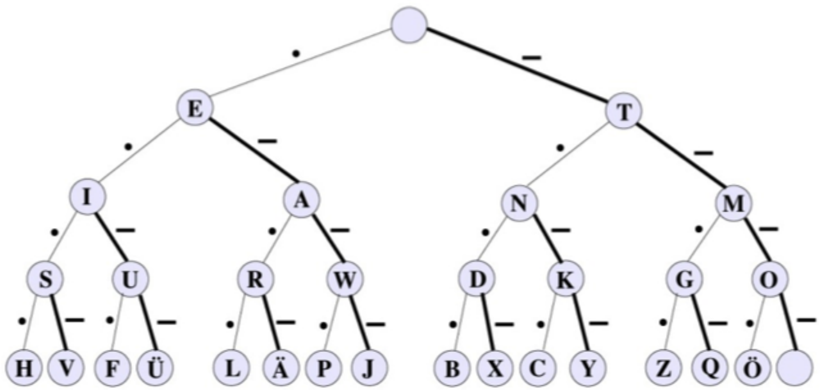


Abbildung 21.1: Baumstruktur des Morsealphabets mit deutschen Umlauten

Bildquelle [**?**]

Problem der Dekodierung: Morsecode benötigt ein Trennzeichen (Pause).
Beispiel: „`. . . . . - .`“ kodiert `seen` und `eier`

Kompressionsverfahren

Codes mit variabler Codelänge

Definition 15 (Fano-Bedingung)

Ein Code erfüllt die Fano-Bedingung, wenn kein Wort aus dem Code der Anfang eines anderen Wortes desselben Codes ist.

- Präfix ist die Teilfolge von Zeichen die am Beginn eines Wortes stehen
- z.B. `C`, `Co`, `Cod` und `Code` wären Präfixe des Wortes `Code`
- Codes die der Fano-Bedingung genügen nennt man **präfixfreie Codes**
- Präfixfreie Codes können ohne Trennzeichen dekodiert werden

Beispiel:

Zeichen	Code
a	0
b	10
c	11

```
10 11 10 11 0 0 10 11
b  c  b  c  a a b  c
```

Huffmann Kodierung

Huffmann Kodierung

Fragestellung

Gegeben sei folgende Quellen:

Zeichen a	**A**	**B**	**C**	**D**
Quelle 1: Wahrscheinlichkeit p_a	0.5	0.25	0.125	0.125
Quelle 2: Wahrscheinlichkeit p_a	0.35	0.3	0.20	0.15

Wie kann der pro Quelle dazugehörige optimale Code konstruiert werden und wie sieht dieser optimale Binärcode aus?

Huffmann Kodierung

Grundidee Huffman-Kodierung

Entwickelt von David Huffman als PhD Student 1952 am MIT

- Zeichen größerer Häufigkeit werden durch kürzere Codes repräsentiert
- Dies führt zu Codes mit variabler Länge
- Für eine effiziente Kodierung/Dekodierung muss die Fano Bedingung erfüllt sein
- Beobachtung: Wenn der **Code optimal** ist, müssen die beiden Symbole der **niedrigsten Häufigkeit mit gleicher Länge** kodiert sein.
- Beweis Skizze:
  - Wären die **Längen verschieden**, könnte man das längere Wort bei **der Länge des kürzeren abschneiden**
    - Dann sind die beiden **entsehenden Codes verschieden**, da **sonst die Fano-Bedingung vorher verletzt** gewesen wäre
    - **Kein anderes Codewort kann länger sein** (da Zeichen niedrigster Häufigkeit und Code optimal ist), also kann die Kürzung selbst nicht die Fano-Bedingung verletzen, das ein anderes Wort mit der Kürzung kodiert wird

    $\Rightarrow$ **Die Fano-Bedingung bleibt nach der Kürzung erhalten**
  - Dann hätten wir einen **neuen Code mit kleinerer durchschnittlicher Wortlänge** und der erste Code wäre nicht optimal gewesen.

Huffmann Kodierung

Vorgehensweise Huffman-Kodierung

Gegeben: Zeichenvorrat und Häufigkeitsverteilung

Gesucht: Kodierung (ist optimal, wenn alle Häufigkeiten Kehrwerte von Zweierpotenzen sind)

- *Schritt 1*: Ersetze die beiden Einträge niedriger Häufigkeit durch einen Code-Teilbaum mit zwei Ästen „0“ und „1“
- *Schritt 2*: Trage die Summe der Häufigkeiten als Häufigkeit für den neuen Code-Teilbaum ein
- Wiederhole 1 und 2 bis alle Zeichen im Codebaum enthalten sind

Huffmann Kodierung

Vorgehensweise Huffman-Kodierung - Beispiel

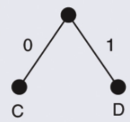
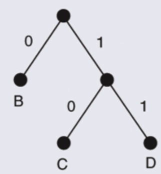
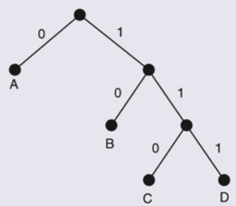
Erster Durchlauf				
Zeichen	**A**	**B**	**C**	**D**
Wahrscheinlichkeit	0.5	0.25	0.125	0.125
Zweiter Durchlauf				
Zeichen/Baum	**A**	**B**		
Wahrscheinlichkeit	0.5	0.25	0.25	
Dritter Durchlauf				
Zeichen/Baum	**A**			
Wahrscheinlichkeit	0.5	0.5		
Vierter Durchlauf				
Zeichen/Baum				
Wahrscheinlichkeit	1.0			

Tabelle 2.7: Ablauf des Huffman-Algorithmus

Bildquelle [?]

Huffmann Kodierung

Experiment Bilddaten

- Graubild, 256x256 Pixel, 8 Bit (i.e. 256 Graustufen)
- Unkopmrimiert: 256*256 = 65 536 Bytes
- Mit Huffman: 40.543 Bytes (ca. 38% Reduktion)
- Einfacher Zusatztrick:
  - Umwandlung der Kodierung vor Kompression: Differenzkodierung
    - Differenz zwischen benachbarten Pixeln
  - Huffman Kodierung auf dieser Differenzkodierung erreicht 33 880 Bytes (50% Reduktion)
  - Differenzkodierung verwendet Domänenwissen $\Rightarrow$ keine universelle Kompression mehr (Details siehe Bild-Kapitel)

Weitere Anwendungsgebiete:

- JPEG, MP3, PNG als Teil der Kompressionsschritte
- bzip bzw. bzip2

Huffmann Kodierung

Eigenschaften

- Code-Baum muss für Dekodierung übergeben werden
- Häufigkeiten in einer Domäne können vorgegeben werden
  - z.B. Zeichen der deutschen Sprache
- Optimaler Code (im Sinne der Entropie) wenn Wahrscheinlichkeiten von der Form $1/2^n$ sind (e.g. 0.5, 0.25, 0.125)
  - 1-Bit Auflösung pro Häufigkeit moeglich
  - In der Praxis kaum gegeben, d.h. digitale Repräsentation ist nicht optimal
  - Ein Zeichen welches häufiger wie 50% auftritt, kann kein Code < 1-Bit gegeben werden
  - Für eine optimale Repräsentation müss das Alphabet verändert werden (z.B. Betrachtung von Teilworten anstatt Zeichen)
- Anzahl der Codewörter steigt exponentiell wenn man Teilworte als Einzelzeichen verwendet (e.g Einzelzeichen 01 od. 10), d.h. der Code Baum wächst exponentiell.

$\Rightarrow$ Arithmetische Codierung

Arithmetische Codierung

Arithmetische Codierung

Überblick

Idee

- Verwende **unendliche genaue reele Zahlen** in einem definierten Intervall und unterteilt dieses Intervall auf Basis der Daten
- Die durch diese Teilung entstandene Zahl entspricht dem komprimierten Code

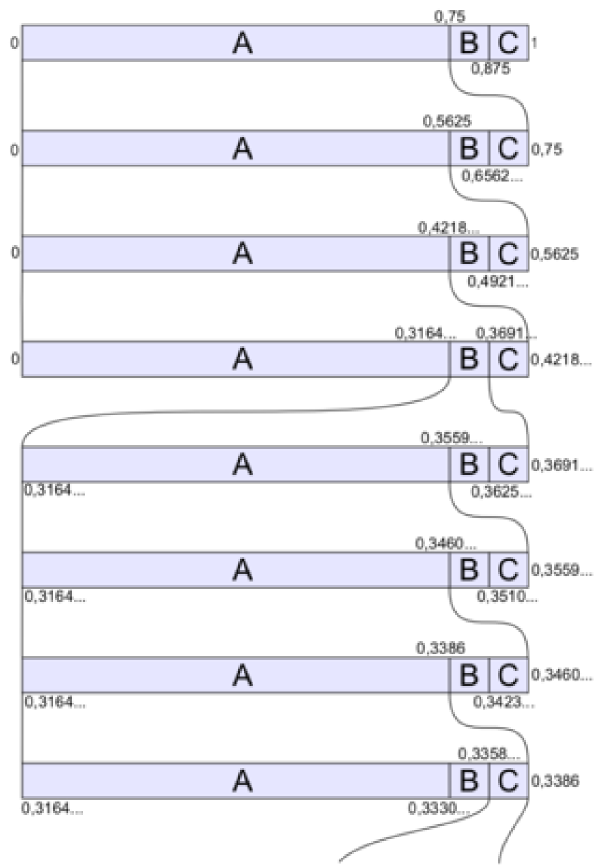
Verfahren

1. Starte mit vereinbarten Intervall (e..g $[0,1)$)
2. Zerlege das aktuelle Intervall in Subintervalle, sodass jedes Zeichen ein Subintervall hat. Die Grösse der Subintervall entspricht der Auftrittswahrscheinlichkeit des Zeichens. Die Zeichenreihenfolge (und damit die Intervallreihenfolge) ist vereinbart
3. Find das Subintervall für das aktuell zu kodierende Zeichen z.
4. Mache dieses Subintervall zum neuen Intervall. Wenn es noch weitere Zeichen gibt gehe zu 2.
5. Ausgabe einer beliebigen Zahl x aus dem aktuellen Intervall plus die Anzahl der kodierten Zeichen, sodaß die Zahl möglichst wenige Nachkommastellen hat.

Arithmetische Codierung

Kodierung - Beispiel

Arithmetische Kodierung der Zeichenkette „AAABAAAC“



Bildquelle Wikipedia

Arithmetische Codierung

Dekodierung

1. Starte mit vereinbarten Intervall (e.g. $[0, 1)$)
2. Unterteile das Intervall in Subintervalle deren Größe von der Häufigkeit der Zeichen abhängt. Die Zeichenreihenfolge ist mit dem Kodierer vereinbart.
3. Finde das Subintervall der übertragenen Zahl und gib das Zeichen aus
4. Wiederhole 2 solange noch Zeichen auszugeben sind

Arithmetische Codierung

Eigenschaften

- Erzeugt einen theoretische optimalen Code, der jedoch durch Rundungseffekte der reelen Zahl i.a. nicht optimal abgebildet werden kann.
  - Bei der Implementierung muss das Intervall wieder vergrößert werden, d.h. ändert sich eine Stelle sicher nicht mehr, dann wird diese Stelle ausgegeben und das Intervall neu skaliert
  - Die Teilung des Intervalls sowie das Intervall selbst muss dem Kodierer und Dekodierer bekannt sein.
- Implementierung meist aufwendig und langsam, da Divsion ausgeführt werden muss oder mehrmals kodiert werden muss

Lauflängenkodierung

Lauflängenkodierung

Überblick

- Einfache From der Datenkomprimierung
- Zeichenkette-basiertes Verfahren ohne statistischen Hintergrund

Verfahren

- Für gleiche, aufeinander folgende Zeichen wird die Anzahl der Zeichen und das Zeichen gespeichert

Beispiel

- `AAAABBCD`
- RLE Kodierung: `4A2B1C1D`

http://www.daniel-lemire.com/blog/archives/2009/11/27/run-length-encoding-part-2/

Lauflängenkodierung

Unerschiedliche Formate

- Zähler wird nur verwendet wenn ein Zeichen zweimal vorkommt, um unötige Zähler zu vermeiden (`AAABBBBBZWWK` $\Rightarrow$ `AA1BB3ZWWK`)
- Benutzung eines Bits pro Zeichen zur Spezifikation ob ein Zähler verwendet werden soll od. nicht
- Es wird nicht der Zähler, sondern die Position gespeichert. Dies erlaubt randomisierten Zugriff in $O(log(n))$, verhindert aber Zählervermeidungsstrategien (`AAABBBBBZWWK` $\Rightarrow$ `1A4B9Z10W11K`)

Lauflängenkodierung

Eigenschaften

- RLE komprimierte Daten können in einem Lauf gelesen und geschrieben werden (keine Statistik)
- Auf einen in RLE repräsentierte Vektor V können Skalarfunktionen in $O(|V|)$ angewendet werden (lineare Laufzeit)
- Lineare Laufzeit auch für Addition zweier Vektoren
- Langsamer randomisierter Zugriff
- Gute Komprimierung benötigt lange Sequenz gleicher Zeichen
- Burrows-Wheeler Transformation ermöglicht die Erzeugung langer Sequenzen
- Anwendung der Burrows-Wheeler Transformation in bzip2

LZW Kompression

LZW Kompression

Überblick

Geschichte:

- Entwickelt 1978 von Abraham Lempel und Jacob Ziv (LZ78)
- Einige Verbesserungen von Terry A. Welch (LZW)
- Angewendet in Bildformat GIF

Verfahren

- Wörterbuch-basiertes Verfahren bei dem das Wörterbuch automatisch aufgebaut wird
- Während dem Einlesen des Datenstroms, wir für jede noch nicht bekannte Sequenz ein Wörterbucheintrag angelegt
- Gibt es schon einen Wörterbucheintrag, wird diesr Code raus geschrieben.
- Das Wörterbuch muss nicht mitübertragen werden

LZW Kompression

Kodierung

1. Präfix ist Leer und Wörterbuch ist mit allen vorkommenden Zeichen initialisiert
2. lies das nächste Zeichen z vom Eingabestrom
3. Ist „Präfix+z" im Wörterbuch?

   Ja: Präfix = Präfix +z

   Nein:

      Gib Code für Präfix aus

      trage **Präfix+z im Wörterbuch** ein

      Präfix = z

4. Ist das Ende des Eingabestroms erreicht?

   Nein: Gehe zu Schritt 2

   Ja: Ist Präfix nicht leer, gib den korrepsondierenden Code aus

LZW Kompression

Beispiel

Kodiere ABBABABAC (Z steht für Zeichen, P für Präfix)

			P+z im Wörterbuch?			
			Ja	Nein		
z	P	P+z	Neues P	Ausgabe	Wörterbuch 1:A,2:B,3:C	Neues P
A		A	A			
B	A	AB		1	4:AB	B
B	B	BB		2	5:BB	B
A	B	BA		2	6:BA	A
B	A	AB	AB			
A	AB	ABA		4	7:ABA	A
B	A	AB	AB			
A	AB	ABA	ABA			
C	ABA	ABAC		7	8:ABAC	
-	C			3		

LZW Kompression

Dekodierung

Dekodierer benötigt den Code plus das **initiale** Wörterbuch. Das vollständige Wörterbuch kann während der Dekodierung generiert werden.

Beobachtung: Wann gibt ein Kodierer den Code aus?

- Wenn die Zeichenkette "Präfix+z“ **nicht** im Wörterbuch ist, wird der Code für „Präfix“ geschrieben und „z“ wird das neue Präfix
- Lesen wir also den Code, so wissen wir, daß "Präfix+ ? “ (? steht für ein beliebiges Zeichen) zu dieser Kodierung geführt hat.
  - Präfix ist bekannt und steht im WB
  - ? kann ermittelt werden, da es das **erste Zeichen vom nächsten Code** ist

$\Rightarrow$ wir müssen für jeden Code den wir ausgeben, das erste Zeichen des Code plus die Zeichen des vorhergehenden Code (der im WB ist) ins Wörterbuch eintragen

$\Rightarrow$ Das Wörterbuch des Dekodierers ist immer um einen Schritt hinter dem des Kodierers hinterher, da wir ja den Wörterbucheintrag über den **aktuell gelesenen** Code und den **zuvor** gelesenen Code erstellen

LZW Kompression

Dekodierung

Wann ist nun ein Code nicht im Wörterbuch des Dekodierers?

- Annahme: $z\Omega z$ wird derzeit kodiert und $z\Omega$ ist im Wörterbuch
  Ω ist eine beliebige Zeichenkette und z ein einzelnes Zeichen
- Der Kodierer schreibt nun den Code für $z\Omega$ in die Ausgabe und fügt $z\Omega z$ zum Wörterbuch hinzu. z wird neues Präfix.
- Liest der Kodierer gleich darauf nochmals die Sequenz Ωz ein, was mit dem Präfix z die Sequenz $z\Omega z$ ergibt, gibt er den Code für das zuvor hinzugefügt $z\Omega z$ aus.
- Da der Dekodierer immer um einen Wörterbucheintrag hinterher hinkt, kann er den Code nicht kennen.
- Da diese Situation **nur genau dann Auftritt, wenn** $z\Omega z$ zuvor zum Wörterbuch hinzugefügt wurde, kann der Dekodierer den neuen Wörterbucheintrag aus dem alten Code $z\Omega$ vollständig rekonstruieren

LZW Kompression

Dekodierung

1. Wörterbuch mit allen vorkommenden Zeichen initialisiert
2. Code = erster Code aus dem Eingabstrom (immer ein Zeichen)
3. Ausgabe des Eintrags für Code und
4. Speicher Code in altCode
5. Code = nächster Code im Eingabestrom
6. Ist Code im Wörterbuch?

   Ja:

   - Gib Zeichen von „Code " aus
   - ‘Präfix = Zeichen von „altCode"
   - Zeichen = erstes Zeichen von „Code "
   - Trage „‘Präfix+Zeichen" ins Wörterbuch ein vgl. die Beobachtungvon vorher

   Nein (bei Zeichenfolgen $z\Omega z\Omega z$)

   - Präfix = Zeichen von „altCode" (=$z\Omega$)
   - Zeichen = erstes Zeichen **von „altCode**" (=z)
   - Trage „‘Präfix+Zeichen" (=$z\Omega z$) in Wörterbuch ein **UND gib es aus**

7. Solange noch Zeichen existieren, lese neuen Code und gehe zu 4.

LZW Kompression

Beispiel

Code: 122473

C	Ausgabe	WB?	Präfix =code(altCode)	altCode	Wörterbuch 1:A,2:B,3:C	Bemerkung
1	A	J				Erstes Zeichen muss im WB sein: Ausgabe
2	B	J	A	1	4:AB	Präfix + neues Zeichen ins WB
2	B	J	B	2	5:BB	neuer Präfix ist alter Code (hier 2)
4	AB	J	B	2	6:BA	Neuer WB Eintrag Präfix+Zeichen
7	ABA	N	AB	4	7:ABA	neuer Code **nicht** im Wörterbuch
3	C	J	ABA	7	8:ABAC	
-						

Zusammenfassung

Kompression

Verschiedene Kompressionsarten

- Verlustfrei/Verlustbehaftet
- Unviersell/Speziell

Verlusfreie Kompression

- Entropy-basierte, optimale Kodierungen Huffman-Code, Artihmetische Kodierung
  - Aufbau von Code-Bäumen mit variabler Code Lönge
  - Rechenintensive
- Zeichenketten-basierte Kodierung
  - Lauflängenkodierung: Wiederholende Zeichen als Anzahl-Zeichen Paar
  - LZW Kodierung: Automatische Erzeugung eines Wörterbuchs

Zusammenfassung

Bibliographie

- **Malaka, Butz, Hussmann (2009)** - Medieninformatik: Eine Einführung (Pearson Studium - IT), Kapitel 2.4
- **Herold, Lurz, Wohlrab (2006)** - Grundlagen der Informatik, Pearson Studium