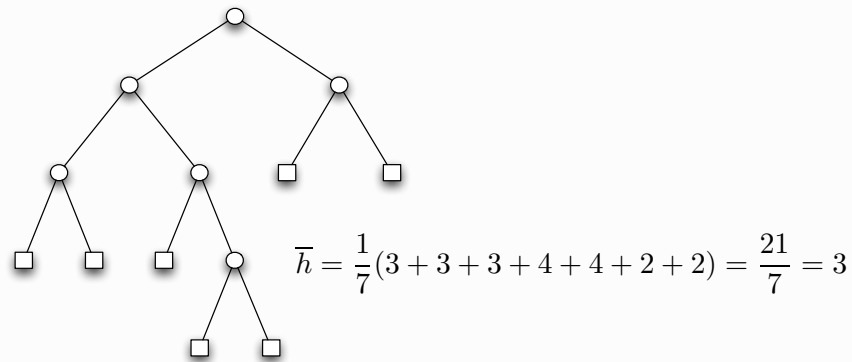


- mittlere Höhe von Binärbäumen
- Entropie
- gewichtete mittlere Höhe von Binärbäumen
- Quellcodierung, Datenkompression
- SHANNONS Theorem
- optimale Quellcodierung (HUFFMAN)

1



3

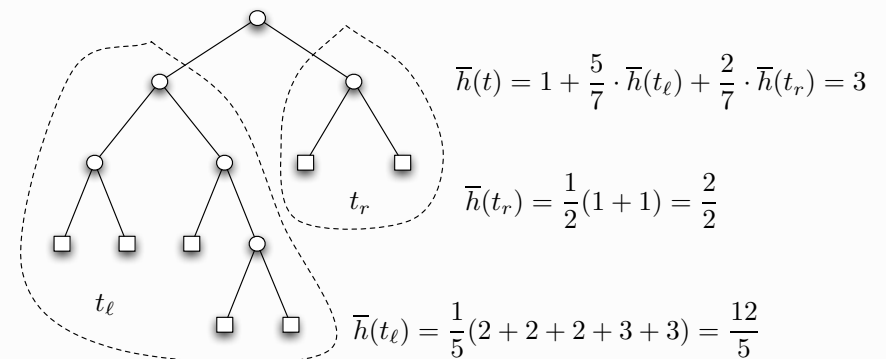
## Zur mittleren Höhe von Binärbäumen

$$\bar{h}(t) = \frac{1}{e(t)} \sum_{b \in E(t)} h(b, t)$$

wobei

- $t$  : Binärbaum
- $E(t)$  : Menge der Blätter (external nodes) von  $t$
- $e(t) = \#E(t)$  : Anzahl der Blätter von  $t$
- $h(b, t)$  : Höhe des Blattes  $b$  in  $t$

2



4

Zusammenhang mit dem rekursiven Aufbau von Binärbäumen

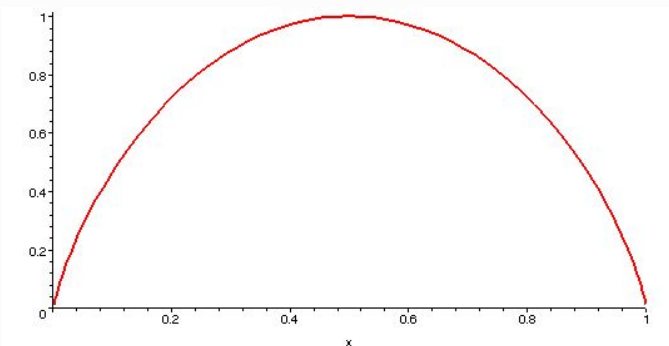
$$\begin{aligned} \bar{h}(\square) &= 0 \\ \bar{h}(\langle \circ, t_\ell, t_r \rangle) &= \frac{1}{e(t)} \sum_{b \in E(t)} h(b, t) \\ &= \frac{1}{e(t)} \left( \sum_{b \in E(t_\ell)} [h(b, t_\ell) + 1] + \sum_{b \in E(t_r)} [h(b, t_r) + 1] \right) \\ &= \frac{e(t_\ell) + e(t_r)}{e(t)} + \frac{e(t_\ell)}{e(t)} \cdot \bar{h}(t_\ell) + \frac{e(t_r)}{e(t)} \cdot \bar{h}(t_r) \\ &= 1 + \frac{e(t_\ell)}{e(t)} \cdot \bar{h}(t_\ell) + \frac{e(t_r)}{e(t)} \cdot \bar{h}(t_r) \end{aligned}$$

5

Dabei ist

$$H(x, 1-x) = -x \log x - (1-x) \log(1-x)$$

die "Entropiefunktion"



7

Fundamentale Ungleichung für Binärbäume

$$\bar{h}(t) \geq \log e(t)$$

folgt per Induktion über der rekursiven Aufbau

$$\begin{aligned} \bar{h}(\square) &= 0 = \log 1 \\ \bar{h}(\langle \circ, t_\ell, t_r \rangle) &= 1 + \frac{e(t_\ell)}{e(t)} \cdot \bar{h}(t_\ell) + \frac{e(t_r)}{e(t)} \cdot \bar{h}(t_r) \\ &\geq 1 + \frac{e(t_\ell)}{e(t)} \cdot \log e(t_\ell) + \frac{e(t_r)}{e(t)} \cdot \log e(t_r) \\ &= 1 + \underbrace{\frac{e(t_\ell)}{e(t)} \cdot \log \frac{e(t_\ell)}{e(t)} + \frac{e(t_r)}{e(t)} \cdot \log \frac{e(t_r)}{e(t)}}_{-H(\frac{e(t_\ell)}{e(t)}, \frac{e(t_r)}{e(t)})} + \log e(t) \\ &= 1 - \underbrace{H(\frac{e(t_\ell)}{e(t)}, \frac{e(t_r)}{e(t)})}_{\geq 0} + \log e(t) \end{aligned}$$

6

SHANNONS Entropie

$X$  : Zufallsvariable, die endlich-viele Werte, z.B.  $1, 2, \dots, n$  annimmt

$p_k = P[X = k]$  : Wahrscheinlichkeit für Eintreten des Ereignisses " $X = k$ "

also

$$p_k \geq 0 \quad (1 \leq k \leq n) \quad \text{mit} \quad \sum_{k=1}^n p_k = 1$$

Bezeichnung:  $\mathbf{p} = \langle p_1, p_2, \dots, p_n \rangle$

"Information(sgehalt)" des Ereignisses " $X = k$ "

$$I[X = k] = -\log p_k$$

"Entropie" = Erwartungswert der Information

$$H(\mathbf{p}) = H(p_1, \dots, p_n) = \mathbf{E}(I, \mathbf{p}) = -\sum_{k=1}^n p_k \log p_k$$

8

## Eigenschaften der Entropiefunktion

1.  $H(p_1, \dots, p_n) \geq 0$
2.  $H(p_1, \dots, p_n) = 0 \Leftrightarrow p_k = \delta_{i,k}$  für ein  $i \in \{1, \dots, n\}$
3.  $H(p_1, \dots, p_n)$  wird maximal für die Gleichverteilung  $p_1 = \dots = p_n = \frac{1}{n}$
4.  $H(p_1, \dots, p_n)$  ist invariant unter Permutation der Komponenten  $p_k$
5.  $H(p_1, \dots, p_n, 0) = H(p_1, \dots, p_n)$
6.  $H(1/n, \dots, 1/n) \leq H(1/(n+1), \dots, 1/(n+1))$
7.  $H(p_1, \dots, p_n)$  ist stetige Funktion der Variablen  $p_k$
8.  $H(1/mn, \dots, 1/mn) = H(1/m, \dots, 1/m) + H(1/n, \dots, 1/n)$
9. für  $p_1, \dots, p_m \geq 0$  und  $q_1, \dots, q_n \geq 0$  mit  $p_1 + \dots + p_m = p$ ,  $q_1 + \dots + q_n = q$  und  $p + q = 1$  gilt

$$H(p_1, \dots, p_m, q_1, \dots, q_n) = H(p, q) + p \cdot H(p_1/p, \dots, p_m/p) + q \cdot H(q_1/q, \dots, q_n/q)$$

9

Zum Beweis der Eigenschaft 3. der Entropiefunktion:

Folgende Aussage bezeichnet man in der Informationstheorie als "key lemma":

Für Wahrscheinlichkeitsverteilungen  $\mathbf{p} = (p_1, \dots, p_n)$  und  $\mathbf{q} = (q_1, \dots, q_n)$  gilt

$$H(p_1, \dots, p_n) \leq - \sum_{k=1}^n p_k \cdot \log q_k$$

und es gilt "=" genau dann, wenn  $\mathbf{p} = \mathbf{q}$ .

Als Konsequenz hat man, indem man für  $\mathbf{q}$  die Gleichverteilung  $q_1 = \dots = q_n = \frac{1}{n}$  wählt:

$$H(p_1, \dots, p_n) \leq \log n$$

11

Bemerkung:

- Alle diese Eigenschaften — ausgenommen vielleicht 3. (s.u.) — lassen sich ausgehend von der Definition leicht nachrechnen.  
Was wichtiger ist: diese Eigenschaften erscheinen als plausible Forderungen, wenn man ausgeht von der intuitiven Vorstellung von der Entropie als dem "Informationsgewinn beim Durchführen eines Experiments"

So beschreibt 8. das Verhalten bei der Kombination unabhängiger und gleichverteilter Experimente zu einem "Produktexperiment" und 9. bei der Ausführung eines Experiments in zwei Stufen.

10

Beweis des "key lemmas" :

Die Logarithmusfunktion ist konvex und es gilt daher

$$\ln x \leq x - 1 \quad (x > 0)$$

mit "=" genau dann, wenn  $x = 1$ . Also ist

$$\ln \frac{q_k}{p_k} \leq \frac{q_k}{p_k} - 1 \quad (1 \leq k \leq n)$$

$$\sum_{1 \leq k \leq n} p_k \ln \frac{q_k}{p_k} \leq \sum_{1 \leq k \leq n} q_k - \sum_{1 \leq k \leq n} p_k = 0$$

$$\sum_{1 \leq k \leq n} p_k \ln q_k \leq \sum_{1 \leq k \leq n} p_k \ln p_k$$

mit "=" genau dann, wenn  $\mathbf{p} = \mathbf{q}$ .

12

Theorem:

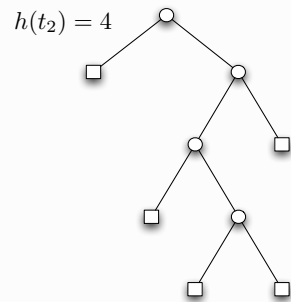
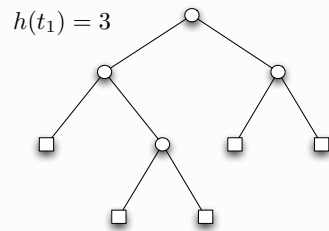
Die Eigenschaften 1.-9. bestimmen die Entropiefunktion eindeutig (bis auf einen konstanten Faktor), d.h.

Eine Funktion mit den Eigenschaften 1.-9. ist notwendig von der Form

$$H(p_1, \dots, p_n) = -c \sum_{k=1}^n p_k \log p_k$$

mit einer Konstanten  $c > 0$ .

13



$$\bar{h}(t_1) = \frac{1}{5}(2 + 3 + 3 + 2 + 2) = \frac{12}{5} \quad \bar{h}(t_2) = \frac{1}{5}(1 + 3 + 4 + 4 + 2) = \frac{14}{5}$$

15

Binäre Bäume mit Gewichten (auf den Blättern)

- betrachten  $\langle t, \mathbf{p} \rangle$  wobei
  - $t$  : binärer Baum
  - $\mathbf{p} = (p_b)_{b \in E(t)}$  Wahrscheinlichkeitsverteilung auf den Blättern von  $t$
- untersuchen Erwartungswert der Höhe der Blätter  
 ("gewichtete mittlere Höhe", "gewichtete externe Pfadlänge")

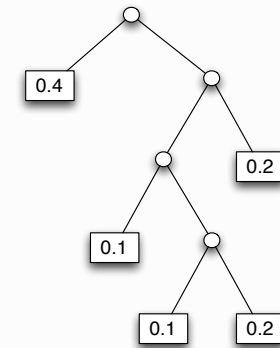
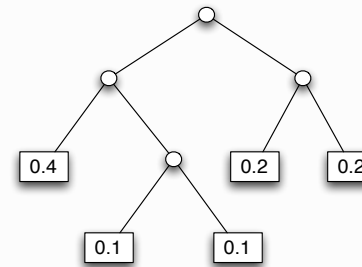
$$\bar{h}(t, \mathbf{p}) = \sum_{b \in E(t)} p_b \cdot h(b, t)$$

- für den Fall der Gleichverteilung auf  $E(t)$  ist das gerade  $\bar{h}(t)$
- Erinnerung:

$$\bar{h}(t) \geq \log e(t)$$

("information theory bound")

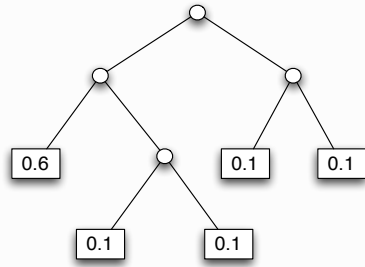
14



$$h(t_1, \mathbf{p}) = 0.4 \cdot 2 + 0.1 \cdot 3 + 0.1 \cdot 3 + 0.2 \cdot 2 + 0.2 \cdot 2 = 2.2$$

$$h(t_2, \mathbf{p}) = 0.4 \cdot 1 + 0.1 \cdot 3 + 0.1 \cdot 4 + 0.2 \cdot 4 + 0.2 \cdot 2 = 2.3$$

16



$$h(t_1, q) = 0.6 \cdot 2 + 0.1 \cdot 3 + 0.1 \cdot 3 + 0.1 \cdot 2 + 0.1 \cdot 2 = 2.2$$

$$h(t_2, q) = 0.6 \cdot 1 + 0.1 \cdot 3 + 0.1 \cdot 4 + 0.1 \cdot 4 + 0.1 \cdot 2 = 1.9$$

17

Der Beweis für den ersten Teil (untere Schranke für  $\bar{h}(t, \mathbf{p})$ ) geht genauso wie der Beweis für  $\log e(t) \leq \bar{h}(t)$  im Fall der Gleichverteilung.

Sei  $t = \langle \bigcirc, t_\ell, t_r \rangle$  Binärbaum mit Knotengewichten

-  $(p_1, \dots, p_m)$  auf  $E(t_\ell) = (b_1, \dots, b_m)$

-  $(q_1, \dots, q_n)$  auf  $E(t_r) = (c_1, \dots, c_n)$

Dabei sind

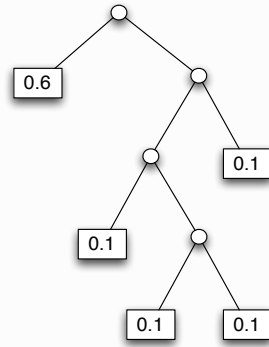
-  $\mathbf{p} = (p_1, \dots, p_m, q_1, \dots, q_n)$

-  $\mathbf{p}_\ell = (p_1/p, \dots, p_m/p)$  mit  $p = p_1 + \dots + p_m$

-  $\mathbf{p}_r = (q_1/q, \dots, q_n/q)$  mit  $q = q_1 + \dots + q_n$

Wahrscheinlichkeitsverteilungen auf  $E(t)$  bzw.  $E(t_\ell)$  bzw.  $E(t_r)$ .

19



Fundamentales Problem der Informationstheorie

- wie klein kann  $\bar{h}(t, \mathbf{p})$  bei gegebenem  $\mathbf{p}$  gemacht werden, indem man den Binärbaum  $t$  geeignet wählt?

Antwort: SHANNONS Quellcodierungstheorem

- untere Schranke: für jeden gewichteten Binärbaum  $\langle t, \mathbf{p} \rangle$  gilt

$$H(\mathbf{p}) \leq \bar{h}(t, \mathbf{p})$$

- obere Schranke: zu jeder WV  $\mathbf{p}$  gibt es einen Binärbaum  $t$  mit

$$\bar{h}(t, \mathbf{p}) < H(\mathbf{p}) + 1$$

18

Dann gilt (Induktion!)

$$\begin{aligned} \bar{h}(t, \mathbf{p}) &= \sum_{b \in E(t)} p_b \cdot h(b, t) \\ &= \sum_{b \in E(t_\ell)} p_b \cdot h(b, t) + \sum_{c \in E(t_r)} p_c \cdot h(c, t) \\ &= \sum_{1 \leq i \leq m} p_i \cdot (h(b_i, t_\ell) + 1) + \sum_{1 \leq j \leq n} q_j \cdot (h(c_j, t_r) + 1) \\ &= p + q + p \cdot \bar{h}(t_\ell, \mathbf{p}_\ell) + q \cdot \bar{h}(t_r, \mathbf{p}_r) \\ &\geq 1 + p \cdot H(\mathbf{p}_\ell) + q \cdot H(\mathbf{p}_r) \\ &= \underbrace{1 - H(p, q)}_{\geq 0} + H(\mathbf{p}) \geq H(\mathbf{p}) \end{aligned}$$

wegen Eigenschaften 3. und 9. der Entropiefunktion.

20

(Verlustfreie) Datenkompression durch Quellcodierung mit variabler Länge

- $A = \{a, b, c, \dots\}$  : endliche Menge von "Nachrichten" (Quellalphabet)
- $\{0, 1\}$  : "Kanalalphabet"
- (binäre) Codierung ist injektive Abbildung

$$\Phi : A \rightarrow \{0, 1\}^+$$

fortgesetzt zu Homomorphismus  $\Phi : A^* \rightarrow \{0, 1\}^*$ ,

falls die Decodierbedingung (*unique decipherability*) erfüllt ist:

$$(UD) \quad \text{für jedes } w \in \{0, 1\}^* \text{ gilt } \#\Phi^{-1}(w) \leq 1$$

- $\Phi(A) = \{\Phi(a), \Phi(b), \Phi(c), \dots\}$  : Menge der Codewörter, "Code"

21

- Beispiele ( $A = \{a, b, c, d\}$ )

$$\Phi_1 : \begin{cases} a \mapsto 00 \\ b \mapsto 01 \\ c \mapsto 10 \\ d \mapsto 11 \end{cases} \quad \Phi_2 : \begin{cases} a \mapsto 0 \\ b \mapsto 111 \\ c \mapsto 110 \\ d \mapsto 101 \end{cases} \quad \Phi_3 : \begin{cases} a \mapsto 01 \\ b \mapsto 011 \\ c \mapsto 110 \\ d \mapsto 101 \end{cases}$$

- $\Phi_1$  : Codierung mit konstanter Länge

$$\Phi_1(abadc) = 00 \cdot 01 \cdot 00 \cdot 11 \cdot 10 = 0001001110$$

- $\Phi_2$  : Codierung mit variabler Länge

$$\Phi_2(abadc) = 0 \cdot 111 \cdot 0 \cdot 101 \cdot 110 = 01110101110$$

- $\Phi_3$  : keine Codierung!

$$\Phi_3(bda) = 011 \cdot 101 \cdot 01 = 01110101 = 01 \cdot 110 \cdot 101 = \Phi_3(acd)$$

22

- Eine Codierung  $\Phi$  (ein Code  $\Phi(A)$ ) hat die *Präfix-Eigenschaft* (ist ein *Präfixcode*), wenn gilt:

(PP) kein Codewort ist Präfix eines anderen Codewortes

- $\Phi_1$  (und allgemein alle Codes konstanter Länge) sowie  $\Phi_2$  haben die (PE), die Abbildung  $\Phi_3$  nicht

- es gilt offensichtlich: (PP)  $\Rightarrow$  (UD), aber (UD)  $\not\Rightarrow$  (PP)

- Beispiele ( $A = \{a, b, c, d, e\}$ )

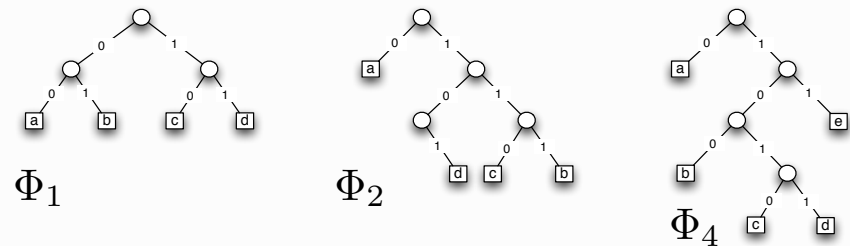
$$\Phi_4 : \begin{cases} a \mapsto 0 \\ b \mapsto 100 \\ c \mapsto 1010 \\ d \mapsto 1011 \\ e \mapsto 11 \end{cases} \quad \Phi_5 : \begin{cases} a \mapsto 00 \\ b \mapsto 101 \\ c \mapsto 010 \\ d \mapsto 001 \\ e \mapsto 11 \end{cases}$$

$\Phi_4$  hat (PP),  $\Phi_5$  hat (UD), aber nicht (PP)

23

Präfixcodes sind binäre Bäume in einem erweiterten Sinn:

innere Knoten können einen (rechten oder linken) oder zwei (rechten und linken) Nachfolger haben



24

## Vergleich von Codierungen — Datenkompression

	$\Phi_4$	$\Phi_5$
aaaea	$\mapsto$ 000110	0000001100
abaedbc	$\mapsto$ 0100011101110101011	0010001100101010

Welche Codierung ist "besser"?

Das hängt davon ab, mit welchen Wahrscheinlichkeiten die "Quellsymbole"  $a, b, c, d, e$  vorkommen!

25

### Beispiel

$\mathcal{Q} = \langle A, \mathbf{p} \rangle$  mit  $A = \{a, b, c, d\}$  und  $\mathbf{p} = (p_a, p_b, p_c, p_d) = (0.9, 0.05, 0.025, 0.025)$

–  $\Phi_1 : a \mapsto 00, b \mapsto 01, c \mapsto 10, d \mapsto 11$

$$\mu(\mathcal{Q}, \Phi_1) = 0.9 \cdot 2 + 0.05 \cdot 2 + 0.025 \cdot 2 + 0.025 \cdot 2 = 2$$

–  $\Phi_2 : a \mapsto 0, b \mapsto 111, c \mapsto 110, d \mapsto 101$

$$\mu(\mathcal{Q}, \Phi_2) = 0.9 \cdot 1 + 0.05 \cdot 3 + 0.025 \cdot 3 + 0.025 \cdot 3 = 1.2$$

### Beispiel

$\mathcal{Q} = \langle A, \mathbf{q} \rangle$  mit  $A = \{a, b, c, d\}$  und  $\mathbf{q} = (q_a, q_b, q_c, q_d) = (0.35, 0.25, 0.25, 0.15)$

–  $\Phi_1 : a \mapsto 00, b \mapsto 01, c \mapsto 10, d \mapsto 11$

$$\mu(\mathcal{Q}, \Phi_1) = 0.35 \cdot 2 + 0.25 \cdot 2 + 0.25 \cdot 2 + 0.15 \cdot 2 = 2$$

–  $\Phi_2 : a \mapsto 0, b \mapsto 111, c \mapsto 110, d \mapsto 101$

$$\mu(\mathcal{Q}, \Phi_2) = 0.35 \cdot 1 + 0.25 \cdot 3 + 0.25 \cdot 3 + 0.15 \cdot 3 = 2.3$$

27

- Eine *Quelle*  $\mathcal{Q} = \langle A, \mathbf{p} \rangle$  ist ein Paar, bestehend aus
  - einem *Quellalphabet*  $A = \{a, b, c, \dots\}$
  - einer *Wahrscheinlichkeitsverteilung*  $\mathbf{p} = (p_a, p_b, p_c, \dots)$  auf  $A$ .

- Für Codierungen  $\Phi : A \rightarrow \{0, 1\}^+$  einer Quelle  $\mathcal{Q} = \langle A, \mathbf{p} \rangle$  ist

$$\mu(\mathcal{Q}, \Phi) = \sum_{x \in A} p_x \cdot |\Phi(x)|$$

*mittlere* oder *erwartete* Codewortlänge des Codes  $\mathcal{C} = \Phi(A)$ .

- Für Präfixcodes:

gewichtete mittlere Höhe des  
entsprechenden Binärbaumes  
(im erweiterten Sinn)

mittlere Codewortlänge =

26

### Folgerung (SHANNON):

Bei gegebener Quellverteilung  $\mathbf{p} = (p_a, p_b, p_c, \dots)$  gibt die Entropie  $H(\mathbf{p})$  ein Maß dafür an, welche Kompression (mittlere Wortlänge) bei keiner Codierung unterschritten werden kann.

### Bemerkungen:

- für optimale Codierung kann man sich auf die Verwendung von (echten) Binärbäumen beschränken
- – Präfixcodes lassen sich "online" decodieren
  - Codes ohne (PP) lassen sich i.a. nicht "online" decodieren
- Verwendung von (UD)-Codes, die nicht die (PP) haben, kann die Situation nicht verbessern:
  - zu jeder Quelle  $\mathcal{Q} = \langle A, \mathbf{p} \rangle$  und zu jedem (UD)-Code  $\Phi(A)$  gibt es einen (PP)-Code  $\Psi(A)$  mit  $\mu(\mathcal{Q}, \Phi) = \mu(\mathcal{Q}, \Psi)$  (Satz von MACMILLAN)
- Konstruktion optimaler Präfixcodes: Verfahren von HUFFMAN

28

Existenz von Präfix-Codes mit gegebenen Wortlängen  $\ell_1, \ell_2, \dots, \ell_n$

- Es existiert ein Präfixcode  $\{w_1, w_2, \dots, w_n\} \subset \{0, 1\}^*$  mit Wortlängen  $|w_k| = \ell_k$  ( $1 \leq k \leq n$ ) genau dann, wenn

$$\sum_{1 \leq k \leq n} 2^{-\ell_k} \leq 1$$

(Ungleichung von KRAFT)

- Beachte: ist  $t$  ein Binärbaum (im ursprünglichen Sinn), so gilt immer

$$\sum_{b \in E(t)} 2^{-h(b,t)} = 1$$

29

- Für  $n = 1$  bzw.  $n = 2$  leisten  $\{0^{\ell_1}\}$  bzw.  $\{0^{\ell_1}, 1^{\ell_2}\}$  das Gewünschte
- Sei die Behauptung für  $n > 1$  bewiesen.  
 $1 \leq \ell_1 \leq \dots \leq \ell_n \leq \ell_{n+1}$  genüge der Ungleichung von KRAFT.

Die gilt dann auch für die  $n + 1$  Zahlen

$$\ell_1, \ell_2, \dots, \ell_{n-1}, \ell_n, \ell_n$$

und wegen  $2^{-\ell_n} + 2^{-\ell_n} = 2^{-(\ell_n-1)}$  auch für die  $n$  Zahlen

$$\ell_1, \ell_2, \dots, \ell_{n-1}, \ell_n - 1$$

Es existiert also ein Präfixcode

$$\{w_1, w_2, \dots, w_n\} \text{ mit } |w_k| = \ell_k \text{ (} 1 \leq k < n \text{) und } |w_n| = \ell_n - 1$$

Der Präfixcode

$$\{w_1, w_2, \dots, w_{n-1}, w_n 0, w_n 1^{\ell_{n+1} - \ell_n + 1}\}$$

leistet das Verlangte.

31

Beweis der Ungleichung von KRAFT (notwendig):

Es sei  $\ell_n = \max_{1 \leq k \leq n} \ell_k$ . Die Wortmengen

$$w_k \cdot \{0, 1\}^{\ell_n - \ell_k} \subseteq \{0, 1\}^{\ell_n} \text{ (} 1 \leq k \leq n \text{)}$$

sind wegen (PP) paarweise disjunkt, deshalb

$$\sum_{1 \leq k \leq n} 2^{\ell_n - \ell_k} \leq 2^{\ell_n}$$

30

Beispiel zu Ungleichung von KRAFT

Wortlängen	Code
2, 3, 3, 3, 5, 7	{00, 010, 110, 111, 01110, 0111111}
2, 3, 3, 3, 4	{00, 010, 110, 111, 0111}
2, 2, 3, 3	{00, 01, 110, 111}
2, 2, 2	{00, 01, 11}
1, 2	{0, 11}
0	{ $\epsilon$ }

32

Beweis des Theorems von SHANNON (obere Schranke)

$\mathcal{Q} = \langle A, \mathbf{p} \rangle$  : Quelle mit  $A = \{a_1, \dots, a_n\}$ ,  $\mathbf{p} = (p_1, p_2, \dots, p_n)$ ,  
wobei  $p_k > 0$  ( $1 \leq k \leq n$ ).

Mit

$$\ell_k = \lceil -\log p_k \rceil \quad (1 \leq k \leq n)$$

sei

$$-\log p_k \leq \ell_k < 1 - \log p_k \quad (1 \leq k \leq n)$$

und somit (linke Ungleichung)

$$\sum_{1 \leq k \leq n} 2^{-\ell_k} \leq 1$$

Es existiert also ein Präfixcode

$$\Phi : A \rightarrow \{w_1, w_2, \dots, w_n\} : a_i \mapsto w_i \quad (1 \leq i \leq n)$$

mit Wortlängen

$$|w_k| = \ell_k = 2^{\lceil -\log p_k \rceil} \quad (1 \leq k \leq n)$$

33

Konstruktion optimaler (Präfix-)Codes (HUFFMAN)

– optimale Codierung  $\Phi$  für eine Quelle  $\mathcal{Q} = \langle A, \mathbf{p} \rangle$

$$\mu(\mathcal{Q}, \Phi) = \min\{\mu(\mathcal{Q}, \Psi) ; \Psi \text{ Codierung für } \mathcal{Q}\}$$

– Das Theorem von SHANNON garantiert für optimales  $\Phi$ :

$$H(\mathbf{p}) \leq \mu(\mathcal{Q}, \Phi) < 1 + H(\mathbf{p})$$

– Die Konstruktion eines optimalen  $\Phi$  kann mit Hilfe eines  
"GREEDY"-Algorithmus ausgeführt werden, der sich am Beweis der  
Ungleichung von KRAFT orientiert

35

Eine Abschätzung für die mittlere Wortlänge dieses Codes ergibt sich aus der rechten Ungleichung

$$\mu(\mathcal{Q}, \Phi) = \sum_{1 \leq k \leq n} p_k \cdot \ell_k < \sum_{1 \leq k \leq n} p_k \cdot (1 - \log p_k) = 1 + H(\mathbf{p})$$

34

Für eine Quelle  $\mathcal{Q} = \langle A, \mathbf{p} \rangle$  mit  $\#A \geq 2$  gilt:

– ist  $\Phi$  optimal für  $\mathcal{Q}$  und sind  $a, b \in A$  mit  $p_a > p_b$ ,

so ist  $|\Phi(a)| \leq |\Phi(b)|$

Begründung:

andernfalls könnte man die Codierungen von  $a$  und  $b$  vertauschen, d.h.

$$\Psi : \begin{cases} a \mapsto \Phi(b) \\ b \mapsto \Phi(a) \\ c \mapsto \Phi(c) \quad (c \in A \setminus \{a, b\}) \end{cases}$$

und damit die mittlere Codelänge verkleinern:

$$\begin{aligned} \mu(\mathcal{Q}, \Psi) &= \mu(\mathcal{Q}, \Phi) - p_a \cdot (|\Phi(a)| - |\Psi(a)|) - p_b \cdot (|\Phi(b)| - |\Psi(b)|) \\ &= \mu(\mathcal{Q}, \Phi) - \underbrace{(p_a - p_b) \cdot (|\Phi(a)| - |\Phi(b)|)}_{>0} \end{aligned}$$

– ist  $\Phi$  optimal für  $\mathcal{Q}$ , so ist der Code  $\Phi(A)$  ein Binärbaum  
(im strikten Sinn)

36

- ist  $\Phi$  optimal für  $\mathcal{Q}$ , so kann man annehmen (d.h., durch Umordnung erreichen), dass es Symbole  $a, b \in A$  mit minimalen Wahrscheinlichkeiten  $p_a, p_b$  gibt, d.h.

$$p_a, p_b \leq \min_{c \in A \setminus \{a, b\}} p_c,$$

die als Geschwister codiert sind, d.h. es gibt ein  $w \in \{0, 1\}^*$  mit

$$\Phi(a) = w \cdot 0 \quad \text{und} \quad \Phi(b) = w \cdot 1$$

Begründung:

Knoten auf dem höchsten Niveau eines Binärbaumes (im engeren Sinne) treten immer als Geschwisterpaare auf.

Durch Umordnung der Wahrscheinlichkeiten auf dem höchsten Niveau kann man die angegebene Situation erreichen, ohne die mittlere Wortlänge zu ändern.

37

- $\Phi$  (striker) Präfixcode für  $\mathcal{Q}$ , bei dem  $\Phi(a)$  und  $\Phi(b)$  Geschwister sind, d.h.  $\Phi(a) = w \cdot 0$ ,  $\Phi(b) = w \cdot 1$  für ein  $w \in \{0, 1\}^*$

$$\Phi' : A \rightarrow \{0, 1\}^+ : \begin{cases} x \mapsto \Phi(x) & \text{für } x \in A \setminus \{a, b\} \\ \alpha \mapsto w \end{cases}$$

$\Rightarrow \Phi'$  (striker) Präfixcode für  $\mathcal{Q}'$

- $\Phi'$  (striker) Präfixcode für  $\mathcal{Q}'$

$$\Phi : A \rightarrow \{0, 1\}^+ : \begin{cases} x \mapsto \Phi'(x) & \text{für } x \in A \setminus \{a, b\} \\ a \mapsto \Phi'(\alpha) \cdot 0 \\ b \mapsto \Phi'(\alpha) \cdot 1 \end{cases}$$

$\Rightarrow \Phi$  (striker) Präfixcode für  $\mathcal{Q}$ , bei dem  $\Phi(a)$  und  $\Phi(b)$  Geschwister sind

39

- Fusion von Quellsymbolen  $a, b \in A$ :

$$\mathcal{Q} = \langle A, \mathbf{p} \rangle$$

$$A' = (A \setminus \{a, b\}) \cup \{\alpha\}$$

$$p'_x = \begin{cases} p_x & \text{für } x \in A \setminus \{a, b\} \\ p_a + p_b & \text{für } x = \alpha \end{cases}$$

$$\mathcal{Q}' = \langle A', \mathbf{p}' \rangle$$

38

- hierbei gilt

$$\begin{aligned} \mu(\mathcal{Q}, \Phi) - \mu(\mathcal{Q}', \Phi') &= p_a \cdot |\Phi(a)| + p_b \cdot |\Phi(b)| - p'_\alpha \cdot |\Phi'(\alpha)| \\ &= p_a \cdot |\Phi(a)| + p_b \cdot |\Phi(b)| - (p_a + p_b) \cdot (|\Phi(a)| - 1) \\ &= p_a + p_b = p'_\alpha \end{aligned}$$

Folgerung

- Entsteht die Quelle  $\mathcal{Q}' = \langle A', \mathbf{p}' \rangle$  aus der Quelle  $\mathcal{Q} = \langle A, \mathbf{p} \rangle$  durch Fusion zweier Symbole  $a, b \in A$  mit minimalen Wahrscheinlichkeiten  $p_a, p_b$ , so gilt (mit dem obigen Zusammenhang zwischen  $\Phi$  und  $\Phi'$ )

$$\Phi \text{ optimal für } \mathcal{Q} \Leftrightarrow \Phi' \text{ optimal für } \mathcal{Q}'$$

Diese Aussage erlaubt die rekursive Konstruktion optimaler Präfixcodes.

40

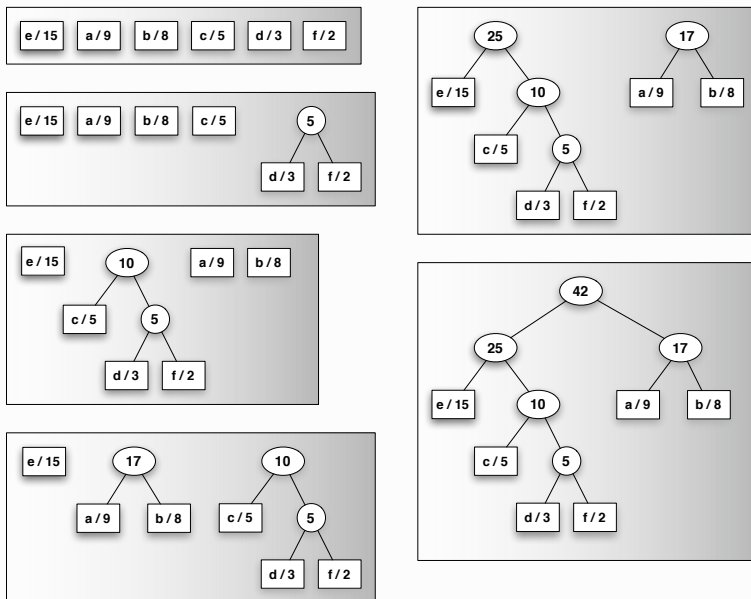
• Realisierung dieser Idee

$$Q^{(2)} \leftarrow Q^{(3)} \leftarrow \dots \leftarrow Q^{(n-1)} \leftarrow Q^{(n)} = Q$$

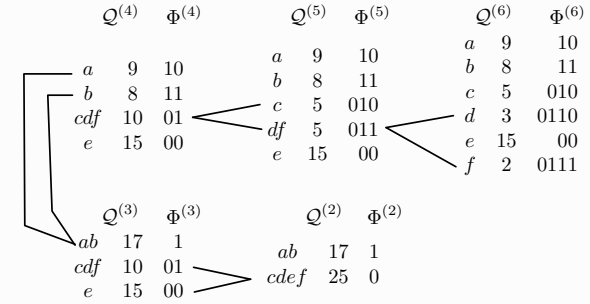
$$\Phi^{(2)} \rightarrow \Phi^{(3)} \rightarrow \dots \rightarrow \Phi^{(n-1)} \rightarrow \Phi^{(n)} = \Phi$$

Dabei

- $Q^{(k)}$  : Quelle mit  $k$  Symbolen
- $\Phi^{(k)}$  : optimaler Präfixcode für  $Q^{(k)}$
- $Q^{(k-1)} \leftarrow Q^{(k)}$  : Fusion von zwei Symbolen mit minimaler W.keit
- $\Phi^{(k-1)} \rightarrow \Phi^{(k)}$  : Konstruktion entlang Umkehrung der Fusion
- $Q^{(2)} = \langle \{a, b\}, (p_a, p_b) \rangle$
- $\Phi^{(2)} = \langle a \mapsto 0, b \mapsto 1 \rangle$



Beispiel zur Konstruktion von HUFFMAN  
(mit Symbolhäufigkeiten statt Wahrscheinlichkeiten)



$$\mu(Q^{(6)}, \Phi^{(6)}) = \frac{9 \cdot 2 + 8 \cdot 2 + 5 \cdot 3 + 3 \cdot 4 + 15 \cdot 2 + 2 \cdot 4}{42}$$

$$= \frac{5 + 10 + 17 + 25}{42} + 1 = \frac{99}{42} = 2.35714284 \dots$$

NB  $H(\frac{9}{42}, \frac{8}{42}, \frac{5}{42}, \frac{3}{42}, \frac{15}{42}, \frac{2}{42}) = 2.309050472 \dots$

Implementierung der HUFFMAN-Konstruktion

- Quelle  $Q = \langle A, p \rangle$  mit  $\#A = n$ ,  $p = (p_1, \dots, p_n)$
- konstruiere Folge  $F^{(n)}, F^{(n-1)}, \dots, F^{(3)}, F^{(2)}, F^{(1)}$  von Wäldern

$$F^{(k)} = \{(t_1^{(k)}, g_1), \dots, (t_k^{(k)}, g_k)\}$$

$t_1^{(k)}, \dots, t_k^{(k)}$  Binärbäume

$g_1, \dots, g_k$  Gewichte mit  $g_1 \geq g_2 \geq \dots \geq g_k$  und  $g_1 + \dots + g_k = 1$

-  $F^{(n)} = \{(\square, p_1), \dots, (\square, p_n)\}$

-  $F^{(k)} \rightarrow F^{(k-1)}$  : mit  $s = \langle \circlearrowleft, t_{k-1}^{(k)}, t_k^{(k)} \rangle$ ,  $h = g_{k-1} + g_k$

$$F^{(k-1)} = \left( F^{(k)} \setminus \{(t_{k-1}^{(k)}, g_{k-1}), (t_k^{(k)}, g_k)\} \right) \cup (s, h)$$

(nach Gewichten ordnen!)

- $t_1^{(1)}$  ist der HUFFMAN-Code

## Komplexität der HUFFMAN-Konstruktion

- Information über die Gewichte  $g_j$  in einer priority queue organisieren!  
Diese Queue als min-heap implementieren.
- Aufbau des heaps:  $\mathcal{O}(n)$
- $n - 1$  heap-Operationen ( $2 \times$  DELETMIN,  $1 \times$  REHEAP) mit Aufwand  $\mathcal{O}(\log n)$
- Gesamtaufwand (Vergleichsoperationen):  $\mathcal{O}(n \log n)$

45

## Literaturhinweise

- V. HEUN behandelt in *Grundlegende Algorithmen* die HUFFMAN-Konstruktion in Kapitel 6.5 (Datenkompression). Alles über Datenkompression erfährt man in
  - S. C. SALOMON, *Data Compression — The Complete Reference*, Springer, 1997.
- Für eine Diskussion des Entropiebegriffs, Quellcodierung etc. muss man in Bücher über Informationstheorie schauen, z.B.
  - D. WELSH, *Codes and Cryptography*, Oxford UP, 1988.
  - R. J. MCELIECE, *The Theory of Information and Coding*, Addison-Wesley, 1977.
- Alle soliden Lehrbücher über Algorithmen(-entwurf) behandeln mehr oder weniger ausführlich die HUFFMAN-Konstruktion im Kontext der GREEDY-Algorithmen, siehe z.B. Kapitel 16 in
  - T. H. CORMEN, C. L. LEISERSON, R. L. RIVEST, C. STEIN, *An Introduction to Algorithms* (2nd. ed.), MIT Press, 2001.Dort erfährt man auch etwas über den theoretischen Hintergrund (Matroide).

47

- HUFFMANS Konstruktion ist ein klassischer GREEDY-Algorithmus.
- Weitere bekannte GREEDY-Algorithmen
  - Zahldarstellung in Positionssystemen (incl. FIBONACCI)
  - Minimale Gerüste (Spannbäume) (KRUSKAL, PRIM)
  - Kürzeste Wege (DIJKSTRA)
  - Knapsack ("fractional")
- GREEDY gehört mit DIVIDE-AND-CONQUER, DYNAMIC PROGRAMMING, BACKTRACKING mit BRANCH-AND-BOUND, RANDOMIZATION zu den fundamentalen Entwurfsprinzipien für Algorithmen
- das Typische an GREEDY-Problemen: optimale Lösungen von Teilproblemen lassen sich immer zu global-optimalen Lösungen fortsetzen
- GREEDY kann man nicht immer einsetzen, aber wenn ja, ist es sehr effizient
- man kann genau charakterisieren, in welchen Situationen GREEDY funktioniert ( $\Rightarrow$  "Matroide")

46