Informatik III

CoNe
Freiburg

Albert-Ludwigs-Universität Freiburg
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

Christian Schindelhauer

Wintersemester 2006/07

16. Vorlesung

15.12.2006

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Das Rekursionstheorem**
  - Selbstreferenz
  - Das Theorem
  - Terminologie
  - Anwendungen
- **Kolmogorov-Komplexität**
  - Optimale Kompression
  - Zufall und Komprimierbarkeit

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

Information und Zufall

- **Welche Sequenzen sind zufällig?**
- **Welche haben viel oder wenig Information?**

  – 000000000000000000000000000000000000000000000000000000
  – 010101010101010101010101010101010101010101010101010101
  – 101011001111010101011010101101010110101010111100110101
  – 011010100101100010111100011101010111000010111101100010
  – 100011010011010110100101010101101010000000000000000000

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Bessere Beschreibungen:**
  - 000000000000000000000000000000000000000000000000000000
    - = 0^{54}
  - 010101010101010101010101010101010101010101010101010101
    - = $(01)^{27}$
  - 101011001111010101011010101101010110101010111100110101
    - = ww^{rev}, für 101011001111010101011010101
  - 100011010011010110100101010101101010000000000000000000
    - = $1000w0ww(01)^{5}10101\ 0^{12}$, für w=11010
  - 011010100101100010111100011101010111000010111101100010
    - = ????

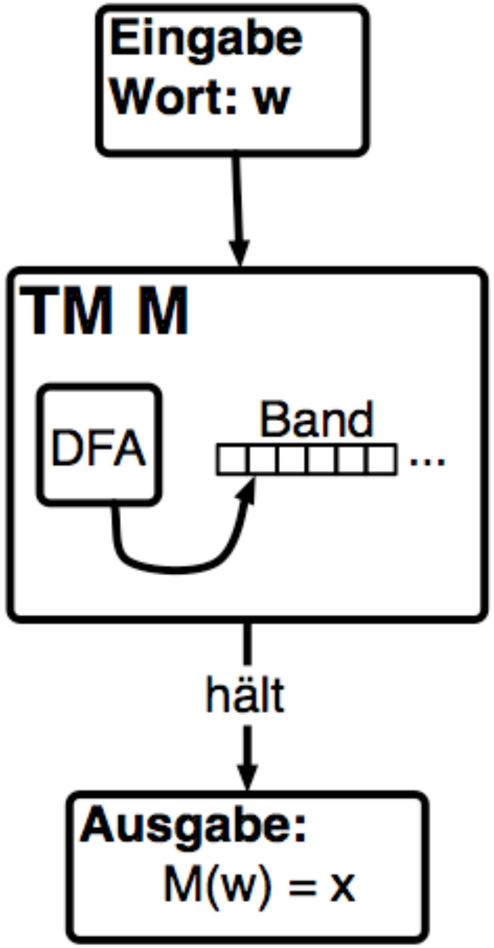
- **Gibt es nicht komprimierbare Binärfolgen?**

Wie wir <M,w> kodieren

- **Mit Hilfe eines größeren Alphabets kann man die Wortlänge weiter komprimieren.**
  - z.B. 137913 = 100001101010111001
- **Um ein festes Maß der Beschreibungskomplexität zu finden, kodieren wir Turing-Maschinen und Eingaben binär wie folgt:**
  - Sei 10110110...10 die Binärkodierung der TM M
  - Sei 01101011...010 das Wort w ∈ {0,1}*
  - Dann ist
    <M,w> = **11 00 11 11 00 11 11 00 ... 11 00 01 01101011...010**
- **Jedes Bit von <M> wird verdoppelt**
  - Aus 1 wird 11
  - Aus 0 wird 00
- **Dann das Trennsymbol 01 eingefügt**
- **Dann wird w angehängt**

Beschreibungskomplexität
Definition

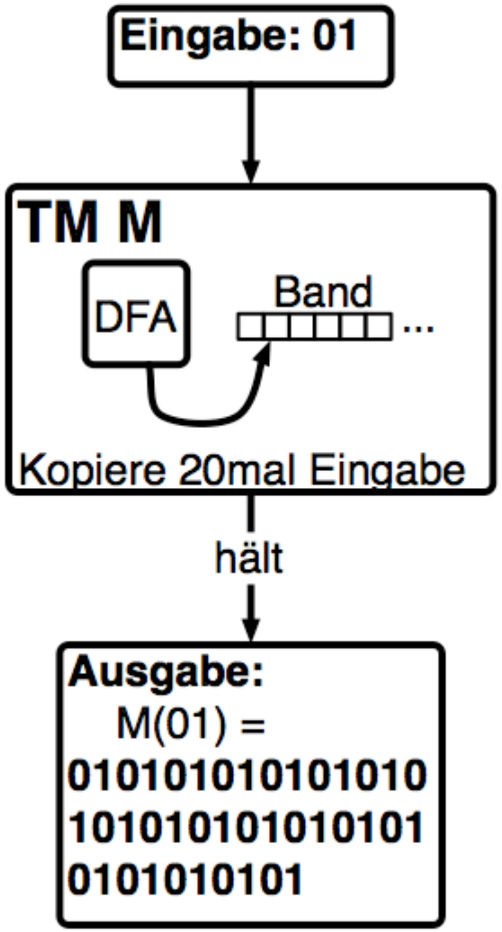
- **Definition**
  - Sei x eine binäre Zeichenkette
  - Die **minimale Beschreibung d(x)** von x ist
  - **die kürzeste binäre Zeichenkette <M,w>,**
    - wobei M eine TM ist und
    - M auf Eingabe w hält und x auf das Band schreibt.
  - Die **Beschreibungskomplexität** (**Kolmogorov-Komplexität oder Kolmogorov-Chaitin-Komplexität**) ist definiert als
    - **K(x) = |d(x)|**



Beschreibungskomplexität: Beispiel

- **Definition**
  - Sei x eine binäre Zeichenkette
  - Die **minimale Beschreibung** von x ist **d(x)**
  - **die kürzeste binäre Zeichenkette <M,w>,**
    - wobei M eine TM ist und
    - M auf Eingabe w hält und x auf das Band schreibt.

- Die **Beschreibungs-Komplexität** ist definiert als
  - **K(x) = |d(x)|**



w = 01

<M> = 1100101010110

<M,w> =
111100001100110
011001111000101

x =
01010101010101010101
01010101010101010101

Aufwärmübungen

- **Theorem**
  - Die Beschreibungskomplexität einer Zeichenkette ist höchstens um eine Konstante größer als die Länge der Zeichenkete, d.h.
  - $\exists c \in \mathbf{N}: \forall x \in \{0,1\}^*: K(x) \leq |x| + c$
- **Beweis**
  - Betrachte TM M =
    - “Auf Eingabe w:
      - kopiere w auf das Ausgabeband
      - halte”
  - Sei <M> die Beschreibung von M mit der Länge $k=|\langle M\rangle|$
  - Dann ist <M,x> eine Beschreibung von x.
  - $|\langle M,x\rangle| = 2k + 2 + |x|$
  - Nun ist 2k+2 eine Konstante
    - die wir c nennen.
  - Die kürzeste Beschreibung von x hat also höchstens die Länge $|x|+c$.

Die Beschreibungs-Komplexität ist nicht berechenbar

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Theorem**
  - Die Beschreibungs-Komplexität ist nicht berechenbar.
- **Zum Beweis benötigen wir folgendes Lemma:**
- **Lemma**
  - Angenommen, die Beschreibungs-Komplexität ist berechenbar. Dann gibt es eine TM M, die auf Eingabe n ein Wort x mit Beschreibungskomplexität n ausgibt, wenn so ein Wort existiert.
- **Beweis**
  - Betrachte folgende TM B
    - B = “Auf Eingabe n:
      - Für t=1,2,3,...:

        Für jede TM M und jedes Wort w mit |<M,w>| = n:

        Simuliere M auf Eingabe w für t Schritte.

        Falls M hält innerhalb von t Schritten hält,

        Sei x die Ausgabe

        Berechne K(x)

        Falls K(x) = n, halte und gib x aus”

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

Die Beschreibungs-Komplexität ist nicht berechenbar

- **Theorem**
  - Die Beschreibungs-Komplexität ist nicht berechenbar.
- **Beweis**
  - Angenommen, K(x) ist berechenbar.
  - Lemma (der letzten Seite) sagt:
    - Dann gibt es eine TM M, die auf Eingabe n ein Wort x mit Beschreibungskomplexität n ausgibt.
  - Betrachte folgende TM R,
    - R = “Auf Eingabe w,
      - Schreibe 2|w| auf das Band.
      - Simuliere M auf n= 2|w|
      - Gib x aus.”
  - Da M x ausgibt, ist die Beschreibungskomplexität von K(x)=2|w|
  - Mit Hilfe von R kann man aber x auch so beschreiben:
    - |<R,w>| = |w| + 2 + 2<R>
  - Falls |w| > 2+2|<R>| ist aber |<R,w>| < K(x)
    - Es gibt dann eine kürzere Beschreibung als die kürzeste Beschreibung
    - Ein Widerspruch.

Eigenschaften der Beschreibungskomplexität

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Theorem**
  - $\exists c \in \mathbf{N}: \forall x,y \in \{0,1\}^*: \quad K(xy) \leq 2K(x) + K(y) + c$
- **Beweis:**
  - Betrachte minimale Beschreibung für x: <M,w>
  - und minimale Beschreibung für y: <M’,w’>
  - Ersetze in <M,w> jede 0 durch 00 und jede 1 durch 11
    - Sei <<M,w>> das Ergebnis
  - Betrachte z = <<M,w,>> 01 <M’,w’> und die TM R:
  - R = “Auf Eingabe z = <<M,w,>>01<M’,w’>
    - Simuliere M auf Eingabe w und
    - Simuliere M’ auf Eingabe w’
    - Gibt konkateniertes Ergebnis aus”
  - Die Komplexität dieser Beschreibung <R,z> ist:
    - $2|\langle R\rangle| + 2 + 2\,K(x) + 2 + K(y) \leq 2K(x)+K(y) + c$
    - für $c = 2|\langle R\rangle| + 4$

Eigenschaften der Beschreibungskomplexität

- **Theorem**
  - $\exists c \in$ **N:** $\forall x,y \in \{0,1\}^*$: $K(xy) \leq 2K(x) + K(y) + c$
- **kann verbessert werden zu:**
- **Theorem**
  - $\exists c \in$ **N:** $\forall x,y \in \{0,1\}^*$: $K(xy) \leq K(x) + K(y) + 2 \log |x| + c$
- Es gilt aber **nicht:**
  - $K(xy) \leq K(x) + K(y) + c$
- **Warum?**

Universalität der Beschreibungskomplexität

- **Definition**
  - Sei $K_p(x)$ die Beschreibungskomplexität bezüglich eines Maschinenmodells p, welches jede Berechnung einer Turing-Maschine berechnen kann.
- **Theorem**
  - Für jedes Maschinenmodell p gilt:
    - $\exists c \in \mathbf{N}$: $\forall x \in \{0,1\}^*$: $K_p(x) \leq K(x) + c$
- **Beweis**
  - Betrachte TM U, die auf Eingabe von TM M und Wort w kodiert als $\langle M,w \rangle$ das Ergebnis M(w) berechnet
  - Sei V die Maschine im Maschinenmodell p, die U berechnet.
  - Sei $\langle M,w \rangle$ die minimale Beschreibung von x
  - Dann ist $\langle V,\langle M,w \rangle\rangle$ auch eine Beschreibung von x aber in p.
  - Es gilt:
    - $| \langle V,\langle M,w \rangle\rangle | = 2|\langle V \rangle| + 2 + K(x) \leq K(x) + c$
    - für $c = 2|\langle V \rangle| + 2$

Komprimierbarkeit

- **Definition**
  - Sei x ein Wort.
  - x ist **c-komprimierbar**, falls
    - $K(x) \leq |x| - c$
  - Falls x nicht c-komprimierbar ist, wird x als **c-unkomprimierbar** bezeichnet
  - Falls x nicht 1-komprimierbar ist, wird x als **unkomprimierbar** bezeichnet.
- **Theorem**
  - Es gibt unkomprimierbare Wörter.
- **Beweis**
  - Die Anzahl der Wörter der Länge = n ist 2^n
  - Die Anzahl der Wörter mit Beschreibungskomplexität $\leq$ n-1 ist
    - $1 + 2 + 4 + 8 + \ldots + 2^{n-1} = 2^n - 1$
  - Damit muss es mindestens ein Wort geben, das unkomprimierbar ist.

Die Anzahl unkomprimierbarer Wörter

- **Theorem**
  - Es gibt mindestens $2^n-2^{n-c+1}+1$ viele **c-unkomprimierbare Wörter**.
- **Beweis**
  - Die Anzahl der Wörter der Länge = n ist 2^n
  - Die Anzahl der Wörter mit Beschreibungskomplexität $\leq$ n-c ist
    - $1 + 2 + 4 + 8 + ... + 2^{n-c} = 2^{n-c+1}-1$
  - Damit muss es mindestens $2^n-2^{n-c+1}+1$ Wörter geben, für die kein Code mit Beschreibungskomplexität $\leq$ n-c zur Verfügung steht.
- **Damit gilt:**
  - Die Wahrscheinlichkeit, dass ein Wort c-komprimierbar ist,
    - ist höchstens $1/2^{c-1}$.
  - Ein zufälliges Wort ist also mit großer Wahrscheinlichkeit c-unkomprimierbar
- **Zufall und Beschreibbarkeit:**
  - Die Unkomprimierbarkeit ist typisch für zufällige Worte.
  - Während alle maschinell erzeugten Wort kleine Beschreibungskomplexität haben.

Ende der 16. Vorlesung

Albert-Ludwigs-Universität Freiburg
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

Christian Schindelhauer
Wintersemester 2006/07
16. Vorlesung
15.12.2006