Gliederung

Nichtdeterministische Turing-Maschinen I

Definition (Nichtdeterministische Turing-Machine)

Eine **Nichtdeterministische Turing-Maschine (kurz NTM)** ist ein Septupel $M = (Z, \Sigma, \Gamma, \delta, z_0, \Box, E)$ mit

- ...
- $\delta \subseteq (Z \setminus E) \times \Gamma \times Z \times \Gamma \times \{L, R, N\}$
- ...

Die "Folgekonfiguration"-Relation $\vdash_M^1$ von M spannt einen Berechnungsbaum auf

- $z_0 w \vdash_M^* k$ bedeutet: k kann von Startkonfiguration erreicht werden (Berechnungspfad)
- haltende/akzeptierende Konfig., halten auf/akzeptieren von Wörtern analog zu DTM
- Zertifikat für w in $T(M)$ ist endlicher Pfad von $z_0 w$ in akzeptierende Konfiguration
- akzeptierte Sprache analog zu DTM: $T(M) := \{w \in \Sigma^* \mid \exists_{\alpha,\beta \in \Gamma^*} \exists_{z \in E} : z_0 w \vdash_M^* \alpha z \beta\}$
- die von M berechnete Funktion ist $f : \mathbb{N} \to \mathbb{N}$ sodass, für alle $x \in \mathbb{N}$ und $y \in \mathbb{N}$,

$$f(x) = y \qquad \Leftrightarrow \qquad \{y' \in \Gamma^* \mid \exists_{z \in E}\, z_0\, \mathrm{BIN}(x) \vdash_M^* z y'\} = \{\mathrm{BIN}(y)\}$$

Nichtdeterministische Turing-Maschinen II

Bemerkung: DTM sind spezielle NTM (ohne Gebrauch des Nichtdeterminismus)

Theorem

Für jede NTM N gibt es eine DTM M mit $T(M) = T(N)$.

Beweis (Idee)

Zeigen: $T(N)$ ist Wertebereich einer berechenbaren Funktion ($\rightsquigarrow$ $T(N)$ semi-entscheidbar)

$$f(x, z) = \begin{cases} x & \text{falls } z \text{ ein Zertifikat für } x \text{ in } T(N) \text{ ist} \\ \bot & \text{sonst} \end{cases}$$

f kann von DTM berechnet werden indem sie dem Pfad im Berechnungsbaum von N folgt.

Einführung Komplexitätstheorie - TSP

Quelle: http://de.wikipedia.org/wiki/Datei:TSP_Deutschland_3.png

Algorithmische Komplexität

Bisher: qualitativ: berechenbar/entscheidbar oder nicht?
Jetzt: quantitativ: wie schnell/effizient kann ein entscheidbares Problem entschieden werden?
... es gibt viele Algorithmen zur Lösung berechenbarer Probleme wie z.B.

- Sortieren
- Potenzieren einer natürlichen Zahl
- ...

Einige davon sind

- schneller (weniger Elementaroperationen) oder
- platzsparender (weniger Speicher) als Andere.

Zentrale Frage

Wann ist ein Algorithmus **effizient** bzw. ein Berechnungsproblem **effizient lösbar**?

(Praktisch meist von Anwendung abhängig)

O-Notation zur Laufzeitanalyse

Problem: wie misst man Laufzeit von Algorithmen?
Beobachtung: Laufzeit muss (mindestens) von der Eingabegröße n abhängen
Ziel: "Effizienz" von Algorithmen unabhängig von Rechentechnik & Programmiersprache
$\leadsto$ "Landau-Symbole" / O-Notation

Definition

Seien $f, g : \mathbb{N} \to \mathbb{N}$. Dann,

- $f \in O(g)$ falls $\exists_{c \in \mathbb{N}^+} \exists_{n_0 \in \mathbb{N}} \forall_{n \geq n_0} f(n) \leq c \cdot g(n)$,
- $f \in \Theta(g)$ falls $f \in O(g)$ und $g \in O(f)$,

Beispiele

- $10\sqrt{n} \in O(n)$
- $2n \in \Theta(n)$
- $\log_2 n \in O(\sqrt{n})$
- $10^6 \in \Theta(1)$
- $n \log_2 n \in O(n^2)$
- $n\sqrt{n} \in O(n^2)$
- $n^{10} \in O(2^n)$
- $3n^4 + 5n^3 + 7 \log_2 n \in O(n^4)$

Deterministische Zeitklassen

Definition (time_M, $\mathrm{DTIME}\,(f(n))$)

Für jede (Mehrband-) DTM M sei $\mathrm{time}_M(n)$ die maximale Anzahl Konfigurationsübergänge von M auf Eingaben x der Länge n (Schritte bevor M auf x hält).

Für eine monoton wachsende Funktion $f : \mathbb{N} \to \mathbb{N}$ ist $\mathrm{DTIME}\,(f(n))$ die Klasse aller Sprachen $L \subseteq \Sigma^*$, die von einer deterministischen Mehrband-TM M akzeptiert werden, welche für jedes $x \in \Sigma^*$ maximal $O(f(|x|))$ Schritte ausführt, das heißt,

$$\mathrm{DTIME}\,(f(n)) := \{L \subseteq \Sigma^* \mid \exists_{\text{DTM } M}\, L = T(M) \wedge \mathrm{time}_M(n) \in O(f(n))\}$$

Definition (P)

$\mathrm{P} := \bigcup_{k \geq 1} \mathrm{DTIME}\,(n^k)$. “deterministisch, in Polynomzeit”

Nichtdeterministische Zeitklassen

Definition (time_N, $\mathrm{NTIME}(f(n))$)

Für jede (Mehrband-) **N**TM N sei $\mathrm{time}_N(n)$ die maximale Länge eines Berechnungspfades von N auf Eingaben x der Länge n.

Für eine monoton wachsende Funktion $f : \mathbb{N} \to \mathbb{N}$ ist $\mathrm{NTIME}(f(n))$ die Klasse aller Sprachen $L \subseteq \Sigma^*$, die von einer **nichtdeterministischen** Mehrband-TM N akzeptiert werden, deren Berechnungspfade für jede Eingabe $x \in \Sigma^*$ maximal Länge $O(f(|x|))$ haben, das heißt,

$$\mathrm{NTIME}(f(n)) := \{L \subseteq \Sigma^* \mid \exists_{\text{NTM } N}\, L = T(N) \land \mathrm{time}_N(n) \in O(f(n))\}$$

Definition (NP)

$\mathrm{NP} := \bigcup_{k \geq 1} \mathrm{NTIME}\left(n^k\right)$. "**nicht**deterministisch, in Polynomzeit"

Bemerkung: $\mathrm{P} \subseteq \mathrm{NP}$ klar, da jede DTM eine NTM ist.

Alternative Definition von NP

Theorem (Alternative Definition für NP („Guess and Check“))

Eine Sprache $L \subseteq \Sigma^*$ ist in NP, gdw. ein Polynom $p : \mathbb{N} \to \mathbb{N}$ und eine polynomiell zeitbeschränkte **D**TM M (d.h. $\mathrm{time}_M(n) \in O(n^c)$) existieren, sodass für jedes $x \in \Sigma^*$ gilt

$$x \in L \Leftrightarrow \exists_{u \in \Sigma^{p(|x|)}} \langle x, u \rangle \in T(M).$$

Beweis (Skizze)

"$\Rightarrow$": Sei $L \in \text{NP}$, d.h. es gibt eine polynomiell zeitbeschränkte NTM N mit $T(N) = L$.
Wir wählen u als Kodierung eines akzeptierenden Berechnungspfads ("Zertifikat") für x in $T(N)$.
Das Zertifikat ist polynomiell lang, da N polynomiell zeitbeschränkt ist.
$\leadsto$ $x \in L$ gdw. es ein solches Zertifikat $u \in \Sigma^{p(|x|)}$ für x in $T(N)$ gibt.

"$\Leftarrow$": Sei M eine DTM wie im Theorem, zeitbeschränkt durch Polynom q.
Wir konstruieren eine NTM N die:
1. das Zertifikat u der Länge $p(|x|)$ nichtdeterministisch erzeugt ("rät") und
2. sich danach wie M auf $\langle x, u \rangle$ verhält.

$\leadsto$ N terminiert in $p(|x|) + q(|x| + |u|)$ Schritten (also polynomieller Zeit) und
$x \in L \Leftrightarrow \exists_{u \in \Sigma^{p(|x|)}} \langle x, u \rangle \in T(M) \Leftrightarrow x \in T(N)$. Also $L \in \mathrm{NTIME}(p(n) + q(n + p(n)))$, also $L \in \text{NP}$.

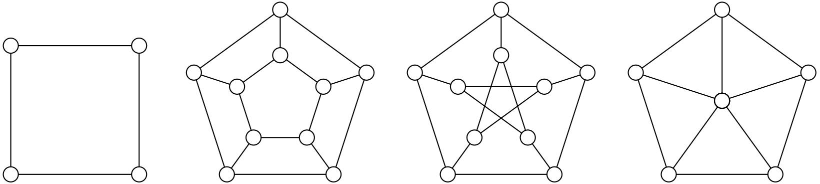
3-Coloring versus 2-Coloring

3-Coloring (2-Coloring)

Eingabe: ungerichteter Graph $G = (V, E)$

Frage: Lassen sich die Knoten von G mit **drei** (**zwei**) Farben so färben, dass keine zwei mit einer Kante verbundenen Knoten die gleiche Farbe haben?

Beispiele: Dreifärbbar? Zweifärbbar?



Mitteilung: Beide Probleme liegen in NP und 2-Coloring sogar in P

Frage: geben Sie einen deterministischen Polynomzeitalgorithmus für 2-Coloring an

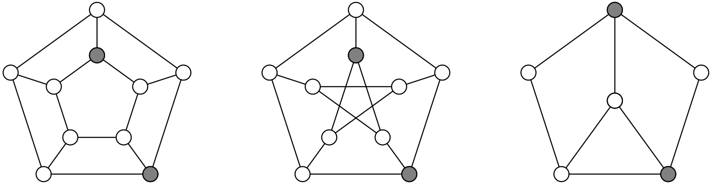
Longest Path versus Shortest Path

Shortest Path (Longest Path)

Eingabe: ungerichteter Graph $G = (V, E)$, zwei Knoten s, t und eine natürliche Zahl $k \leq |V|$

Frage: Existiert ein „einfacher" Pfad zwischen s und t der Länge **höchstens** (**mind.**) k?

Beispiel: Pfad der Länge ≤ 2? Pfad der Länge ≥ 9 (oder ≥ 5)?



Mitteilung: Beide Probleme liegen in NP und Shortest Path liegt sogar in P (Breitensuche)!

3-SAT versus 2-SAT

3-SAT (2-SAT)

Eingabe: aussagenlogische Formel F in „konjunktiver Normalform“ mit ≤ 3 (bzw. ≤ 2) Literalen pro Klausel.

Frage: Ist F **erfüllbar**, d.h. gibt es eine $\{0, 1\}$-wertige Belegung der in F verwendeten Booleschen Variablen derart, dass F zu **wahr** (d.h. 1) ausgewertet wird?

Beispiele

- $(x_1 \vee x_2 \vee x_3) \wedge (\overline{x_1} \vee x_3 \vee x_4) \wedge (\overline{x_2} \vee \overline{x_3} \vee \overline{x_4}) \wedge (x_2 \vee x_3 \vee x_4)$ ist erfüllbar z.B. mit $x_1 = 0$, $x_2 = 0$, $x_3 = 1$ (und x_4 beliebig).
- $(x_1 \vee \overline{x_2}) \wedge (x_1 \vee \overline{x_3}) \wedge (\overline{x_1} \vee x_2) \wedge (\overline{x_1} \vee \overline{x_2}) \wedge (x_2 \vee x_3)$ nicht erfüllbar

Mitteilung: Beide Probleme liegen in NP und 2-SAT liegt sogar in P

P versus NP

Die bekannteste offene Frage der (Theoretischen) Informatik ist: $\mathrm{P} \stackrel{?}{=} \mathrm{NP}$.

Zur Einordnung von P versus NP: „Geglaubtes Schaubild“ (unter $\mathrm{P} \subsetneq \mathrm{NP}$):