Algorithmen für schwierige Probleme

Britta Dorn

Wintersemester 2011/12

16. November 2011

Linear Programming (LP)

Lineares Programm:

gegeben: Matrix $A \in \mathbb{R}^{m,n}$, zwei Vektoren $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$.
gesucht: Vektor $x \in \mathbb{R}^n$ (Einträge ≥ 0), der lineare Bedingungen erfüllt:

$$\begin{array}{ccccccc} a_{11}x_1 & + & \dots & + & a_{1n}x_n & \leq b_1 \\ a_{21}x_1 & + & \dots & + & a_{2n}x_n & \leq b_2 \\ \vdots & & \vdots & & \vdots & \vdots \\ a_{m1}x_1 & + & \dots & + & a_{mn}x_n & \leq b_m. \end{array}$$

Finde unter allen zulässigen Vektoren x einen, der Skalarprodukt $c^T x = c_1x_1 + \cdots + c_nx_n$ maximiert.

Kurz: $\max\{c^T x \mid Ax \leq b,\ x \geq 0\}$ komponentenweise lesen...

maximiere $c^T x$
subject to $Ax \leq b$, $x \geq 0$ (constraints)

Linear Programming (LP)

maximiere $c^T x$
subject to $Ax \leq b$, $x \geq 0$ (constraints)

äquivalente Formulierungen:

- minimiere
- Bedinungen $\geq$ statt $\leq$
- $=$ statt $\leq$
- $x \geq 0$ muss nicht sein
- ...

LP (linear programming) Probleme in Polynomzeit lösbar!
(Innere-Punkte-Methode, Ellipsoidmethode)
Simplex-Verfahren (trotz exp. Laufzeit)

Integer Linear Programming (ILP)

ILP (Integer Linear Programming): nur ganzzahlige Variablen zugelassen (macht Problem i.a. schwieriger!)
Beispiel:

ILP für Vertex Cover

Gegeben: Graph $G = (V, E)$.
Variablen: für jeden Knoten $v \in V$ eine Variable x_v, kann Wert 1 haben (im Vertex Cover) oder Wert 0 (nicht im Vertex Cover)

Minimiere $\sum_{v \in V} x_v$
subject to $x_u + x_v \geq 1 \quad \forall e = \{u, v\} \in E$
$x_v \in \{0, 1\} \, \forall v \in V$

ILPs sind i.a. schwer zu lösen (entsprechendes Entscheidungsproblem ist NP-vollständig).

Integer Linear Programming (ILP)

Wichtiges Theorem für uns:

Theorem [Lenstra, 1983]

ILPs können mit $O(p^{9p/2} \cdot L)$ arithmetischen Operationen mit Integern der Größe $O(p^{2p} \cdot L)$ Bits gelöst werden, wobei p die Anzahl der ILP Variablen ist und L die Anzahl der Eingabebits.

Also: Falls p klein ist (oder nur von einem kleinen Parameter abhängt), dann kann man ILP effektiv lösen!