6 Polynomielle Algorithmen

Definition 6.1. Sei $D \subseteq \mathbb{R}$ und seien $f, g : D \to \mathbb{R}_{\geq 0}$.

(i) Existieren $\alpha, \beta > 0$ mit

$$\forall x \in D : f(x) \leq \alpha g(x) + \beta$$

so schreiben wir "$f = O(g)$".

(ii) Gilt $f = O(g)$ und $g = O(f)$, so schreiben wir "$f = \Theta(g)$".

Bemerkung 6.2. Die verschiedenen speziellen Probleme, auf die ein **Algorithmus** angewendet werden kann heißen **Instanzen** des Algorithmus. Die **Kodierungslänge** einer Instanz ist ein Maß für die in ihr enthaltene Menge an Information und entspricht der Anzahl der Bits, die zu ihrer Beschreibung nötig sind.

Die Laufzeit eines Algorithmus auf einer Instanz ist beschränkt durch das Produkt aus

- der Zahl der ausgeführten **elementaren Operationen**

  (Beispiele: einfache arithmetische Operationen, Vergleiche, Setzungen, Zugriff, etc.)

- und der maximalen Zeit, die eine dieser elementaren Operationen benötigt.

Ein Algorithmus, dessen Instanzen Listen rationaler Zahlen sind, besitzt **polynomielle Laufzeit**, wenn ein $k \in \mathbb{N}$ existiert, so dass er für eine Instanz mit Kodierungslänge n insgesamt $O(n^k)$ elementare Operationen ausführt und alle Zahlen, auf die er elementare Operationen anwendet, Kodierungslänge $O(n^k)$ haben.

Es ist keine Implementierung/Variante des Simplexalgorithmus mit polynomieller Laufzeit bekannt.

Definition 6.3. Wir definieren folgende Kodierungslängen $\langle \cdot \rangle$.

(i) $\langle n \rangle = 1 + \lceil \log_2(|n| + 1) \rceil$ für $n \in \mathbb{Z}$.

(ii) $\langle \frac{p}{q} \rangle = \langle p \rangle + \langle q \rangle$ für $p, q \in \mathbb{Z}$, $q \neq 0$ und $\text{ggt}(p, q) = 1$.

(iii) $\langle (x_1, \ldots, x_n)^T \rangle = \langle x_1 \rangle + \ldots + \langle x_n \rangle$ für $(x_1, \ldots, x_n)^T \in \mathbb{Q}^n$.

(iv) $\langle A \rangle = \sum_{i=1}^{m} \sum_{j=1}^{n} \langle a_{i,j} \rangle$ für $A = (a_{i,j}) \in \mathbb{Q}^{m \times n}$.

Bemerkung 6.4.

- Für ein rationales LP

  $$(P) : \min\{c^T x \mid Ax = b, x \geq 0\}$$

  setzt man

  $$\langle (P) \rangle = \langle A \rangle + \langle c \rangle + \langle b \rangle.$$

- $|n| \leq 2^{\langle n \rangle - 1} - 1$.

- Um $\left\langle \frac{p}{q} \right\rangle$ nach oben zu beschränken, muss sowohl $\langle p \rangle$ als auch $\langle q \rangle$ beschränkt sein.

Lemma 6.5. *Seien $r_1, \ldots, r_n \in \mathbb{Q}$, $x, y \in \mathbb{Q}^n$ und $A = (a_{i,j}) \in \mathbb{Q}^{n \times n}$.*

(i) $|r| \leq 2^{\langle r \rangle - 1} - 1$ *und* $|1/r| \leq 2^{\langle r \rangle - 1} - 1$.

(ii) $\langle r_1 \cdot r_2 \cdot \ldots \cdot r_n \rangle \leq \langle r_1 \rangle + \langle r_2 \rangle + \ldots + \langle r_n \rangle$.

(iii) $\langle r_1 + r_2 + \ldots + r_n \rangle \leq 2(\langle r_1 \rangle + \langle r_2 \rangle + \ldots + \langle r_n \rangle)$.

(iv) $||x||_2 \leq ||x||_1 \leq 2^{\langle x \rangle - n} - 1$.

(v) $\langle x + y \rangle \leq 2(\langle x \rangle + \langle y \rangle)$.

(vi) $\langle x^T y \rangle \leq 2(\langle x \rangle + \langle y \rangle)$.

(vii) $|\det(A)| \leq 2^{\langle A \rangle - n^2} - 1$.

(viii) $\langle \det(A) \rangle \leq 2\langle A \rangle$.

□

Satz 6.6. *Ist $v = (v_1, \ldots, v_n)^T$ eine Ecke des Polyeders*

$$P = \{x \in \mathbb{R}^n \mid Ax = b, x \geq 0\}$$

mit $A \in \mathbb{Q}^{m \times n}$ und $b \in \mathbb{Q}^m$, so gilt $\langle v_i \rangle \leq 4\langle A \rangle + 2\langle b \rangle$ und daher $\langle v \rangle \leq 4n\langle A \rangle + 2n\langle b \rangle$.

□

Folgerung 6.7. *Sei $(P) : \min\{c^T x \mid Ax = b, x \geq 0\}$ mit $A \in \mathbb{Q}^{m \times n}$, $b \in \mathbb{Q}^m$ und $c \in \mathbb{Q}^n$. Ist (P) lösbar, so stammt der optimale Wert von (P) stammt aus der endlichen Menge*

$$S = \{r \in \mathbb{Q} \mid \langle r \rangle \leq 8n\langle A \rangle + 4n\langle b \rangle + 2\langle c \rangle\}.$$

□

Input: (P) wie in Folgerung 6.7.
Output: Eine optimale Lösung von (P).
begin
  $S \leftarrow \{r \in \mathbb{Q} \mid \langle r \rangle \leq 8n\langle A \rangle + 4n\langle b \rangle + 2\langle c \rangle\}$;
  while $|S| > 1$ **do**
    Sei $s \in S$ mit
    $$||S'| - |S''|| \leq 1$$
    für $S' = \{t \in S \mid t < s\}$ und $S'' = \{t \in S \mid t \geq s\}$;
    if $\{x \in \mathbb{R}^n \mid Ax = b, x \geq 0, c^T x \leq s\} = \emptyset$ **then**
      $S \leftarrow S''$;
    else
      $S \leftarrow S'$;
    end
  end
  Sei s das eindeutige Element von S;
  Bestimme x mit $Ax = b, x \geq 0, c^T x = s$;
  return x;
end

Algorithm 1: Binäre Suche

Die Anzahl der Iterationen der binären Suche ist höchstens

$$\lceil \log_2(|S|+1) \rceil$$

und damit nach Folgerung 6.7 polynomiell in $\langle (P) \rangle$.

Satz 6.8. *Es existiert ein polynomieller Algorithmus zur Lösung rationaler linearer Programme genau dann, wenn ein Algorithmus existiert, der in polynomieller Zeit entscheidet, ob ein rationales Polytop P leer oder nicht leer ist und — falls P nicht leer ist — einen Punkt in P bestimmt.*

□

6.1 Ellipsoidmethode

Die Ellipsoidmethode geht ursprünglich auf Yudin und Nemirovskii (1976) und Shor (1977) zurück und wurde zur Lösung konvexer "feasibility"-Probleme entwickelt, d.h. Fragen der Form "Ist eine gegebene konvexe Menge leer?" 1979 beobachtete Khachiyan, dass man mit ihr lineare Programme in polynomieller Zeit lösen kann.

Definition 6.9. Für einen Vektor $a \in \mathbb{R}^n$ und eine symmetrische positiv definite Matrix $A \in \mathbb{R}^{n \times n}$ ist die Menge

$$E = E(A,a) = \left\{ x \in \mathbb{R}^n \mid (x-a)^T A^{-1} (x-a) \leq 1 \right\}$$

ein *Ellipsoid mit Zentrum a.*

Bemerkung 6.10. Ist A eine symmetrische positiv definite Matrix, so sind alle Eigenwerte von A positive reelle Zahlen und es existiert eine Orthonormalbasis aus Eigenvektoren von A, d.h. es existiert eine (Basiswechsel)Matrix O mit $O^{-1} = O^T$ und $O^T A O$ ist Diagonalmatrix der Eigenwerte.

Es folgt die Existenz einer symmetrischen positiv definiten Matrix $A^{\frac{1}{2}}$ mit $A = A^{\frac{1}{2}} A^{\frac{1}{2}}$. Nun gilt

$$E(A,a) = A^{\frac{1}{2}} S(0,1) + a$$

wobei

$$S(0,1) = \left\{ x \in \mathbb{R}^n \mid x^T x \leq 1 \right\}.$$

Das n-dimensionale Volumen von $E(A,a)$ ist daher

$$\det\left(A^{\frac{1}{2}}\right) \cdot \text{volume}(S(0,1)) = \sqrt{\det(A)} \cdot \text{volume}(S(0,1)) \leq \sqrt{\det(A)} 2^n.$$

Nun zur Idee der Ellipsoidmethode. Sei $P = \{x \in \mathbb{R}^n \mid Ax \leq b, x \geq 0\}$ gegeben. Die Aufgabe besteht darin, zu testen, ob P leer ist und ggf. einen Punkt aus P zu bestimmen. Ist P nicht leer, so existieren Elemente, deren Kodierungslänge in der Kodierungslänge von P beschränkt sind. Wähle einen ersten Ellipsoiden, der garantiert Elemente von P enthält, falls P nicht leer ist, bzw. der garantiert P enthält, falls P beschränkt ist. Teste den Mittelpunkt und iteriere bis das Volumen klein genug ist. Da das n-dimensionale Volumen von nicht volldimensionalen Polyedern 0 ist, machen diese Probleme.

Satz 6.11. *Sei $A \in \mathbb{Q}^{m\times n}$ und $b \in \mathbb{Q}^m$.*
Das System

$$Ax \leq b, x \geq 0$$

besitzt genau dann eine Lösung, wenn das System

$$Ax \leq b + \epsilon\mathbf{1}, 0 \leq x \leq R\mathbf{1}$$

eine Lösung besitzt, wobei

$$\begin{aligned} \mathbf{1} &= (1,\ldots,1)^T, \\ \frac{1}{\epsilon} &= 2m2^{4\langle A\rangle+10\langle b\rangle} \text{ und} \\ R &= 2^{4\langle A\rangle+2\langle b\rangle} \end{aligned}$$

gilt.
Besitzt $Ax \leq b$, $x \geq 0$ eine Lösung, so ist das n-dimensionale Volumen von

$$\{x \in \mathbb{R}^n \mid Ax \leq b + \epsilon\mathbf{1}, 0 \leq x \leq R\mathbf{1}\}$$

mindestens $\left(\frac{\epsilon}{n2^{\langle A\rangle}}\right)^n$.

□

Satz 6.12. *Existieren*

- *ein polynomieller Algorithmus, der zu einem gegebenen rationalen Polyeder $P = \{x \in \mathbb{R}^n \mid Ax \leq b, x \geq 0\}$ entscheidet, ob P leer ist sowie*
- *ein polynomieller Algorithmus, der die Lösungsmenge rationaler linearer Gleichungssysteme bestimmt,*

so existiert ein polynomieller Algorithmus, der zu einem gegebenen rationalen Polyeder $P = \{x \in \mathbb{R}^n \mid Ax \leq b, x \geq 0\}$ entscheidet, ob P leer ist und — falls P nicht leer ist — eine Ecke von P bestimmt.

□

Satz 6.13. *Sei $E = E(A, a) \subseteq \mathbb{R}^n$ ein Ellipsoid und $c \in \mathbb{R}^n \setminus \{0\}$. Es gilt*

$$E \cap \{x \in \mathbb{R}^n \mid c^T x \leq c^T a\} \subseteq E(A', a')$$

für

$$\begin{aligned} d &= \frac{Ac}{\sqrt{c^T Ac}} \\ a' &= a - \frac{1}{n+1}d \\ A' &= \frac{n^2}{n^2-1}\left(A - \frac{2}{n+1}dd^T\right). \end{aligned}$$

Weiter ist A' positiv definit und

$$\frac{\det(A')}{\det(A)} \leq f(n) < 1.$$

□

Bemerkung 6.14. $E(A', a')$ ist sogar der eindeutige Ellipsoid minimalen Volumens, der $E \cap \{x \in \mathbb{R}^n \mid c^T x \leq c^T a\}$ enthält.

Input: $A \in \mathbb{Q}^{m \times n}$ und $b \in \mathbb{Q}^m$.
Output: Entscheidet, ob $P = \{x \in \mathbb{R}^n \mid Ax \leq b, x \geq 0\}$ leer ist.
begin
  $\frac{1}{\epsilon} \leftarrow 2m2^{4\langle A\rangle + 10\langle b\rangle}$;
  $R \leftarrow 2^{4\langle A\rangle + 2\langle b\rangle}$;

$$P' \leftarrow \{x \in \mathbb{R}^n \mid Ax \leq b + \epsilon \mathbf{1}, 0 \leq x \leq R\mathbf{1}\};$$

$$N \leftarrow \left\lceil \log_{\sqrt{f(n)}} \left(\left(\frac{\epsilon}{n 2^{<A>} 2\sqrt{n} R} \right)^n \right) \right\rceil;$$

  $A_0 \leftarrow nR^2 I_n$; $a_0 \leftarrow 0$; $E_0 \leftarrow E(A_0, a_0)$;
  $k \leftarrow 0$;
  while $k \leq N$ **do**
    if $a_k \in P'$ **then return** (P ist nicht leer);
    Sei $c^T x \leq \beta$ eine Ungleichung aus der Beschreibung von P', die a_k verletzt, d.h. $c^T x \leq c^T a_k$ für alle $x \in P'$;
    Setze A_{k+1} und a_{k+1} wie in Satz 6.13 so, dass

$$E_k \cap \{x \in \mathbb{R}^n \mid c^T x \leq c^T a_k\} \subseteq E(A_{k+1}, a_{k+1})$$

    gilt;
    $E_{k+1} \leftarrow E(A_{k+1}, a_{k+1})$;
    $k \leftarrow k + 1$;
  end
  return (P ist leer);
end

Algorithm 2: Ellipsoid Methode

Satz 6.15. *Die* Ellipsoid Methode *arbeitet korrekt.*

□