Seminar: Intelligente Algorithmen
Stefan Kopp, Alfred Kranstedt, Nadine Leßmann

Lineare Programmierung

Frank Schönmann

WS 2003/04

1 Einführung in die lineare Programmierung (LP)

Grundidee der linearen Programmierung ist die Optimierung einer linearen Funktion mit n Freiheitsgraden, die durch lineare Gleichungen bzw. Ungleichungen eingeschränkt ist. Formal lässt sich ein LP-Problem folgendermaßen definieren:

$$\text{maximiere } f(x_1, \ldots, x_n) = \sum_{i=1}^{n} c_i x_i$$

unter Einhaltung der Nebenbedingungen

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i \qquad \text{für } i = 1, 2, \ldots, m$$
$$x_j \geq 0 \qquad \text{für } j = 1, 2, \ldots, n$$

Das selbe Problem lässt sich in Matrix-Schreibweise definieren als

$$\text{maximiere } f(\mathbf{x}) = \mathbf{c}^T \mathbf{x}$$

unter den Nebenbedingungen (nach geeigneter Definition von $\leq$ und $\geq$)

$$A\mathbf{x} \leq \mathbf{b} \qquad \mathbf{x} \geq 0$$

Da dieses Problem bereits in einer standardisierten Form vorliegt, kann man es kompakt als $(A, \mathbf{b}, \mathbf{c})$ zusammenfassen mit $A \in \mathbb{R}^{m \times n}, \mathbf{b} \in \mathbb{R}^m$ und $\mathbf{c}, \mathbf{x} \in \mathbb{R}^n$.

Es gibt zwei äquivalente Formulierungen von LP-Problemen, die sogenannten Normalformen. In der Standardform sind alle Bedingungen durch Ungleichungen definiert, in der Slackform durch Gleichungen. Jedes LP-Problem lässt sich durch geeignete Umformungen in beide Normalformen bringen. Mögliche Abweichungen von der Standardform sind

1. Minimierungsproblem statt Maximierungproblem,

2. Variablen ohne Nichtnegativitätsbedingung,

3. Gleichheitsbedingungen statt Ungleichheitsbedingungen und
4. Größer-als-Bedingungen statt Kleiner-als-Bedingungen.

Um ein LP-Problem in Standardform in die Slackform zu überführen, verwendet man die Äquivalenz der folgenden beiden Ausdrücke:

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i \qquad \Longleftrightarrow \qquad 0 \leq s_i = b_i - \sum_{j=1}^{n} a_{ij} x_j$$

Die s_i nennt man Slack-Variablen. Außer den Nichtnegativitätsbedingungen erhält man nur noch Gleichungen als Nebenbedingungen.

2 Geometrische Interpretation

Anschaulich betrachtet beschreibt eine lineare Gleichung eine Hyperebene; eine lineare Ungleichung damit einen Halbraum, nämlich entweder alle Punkte die diesseits oder jenseits der Hyperebene liegen. Ein lineares Ungleichungssystem beschreibt daher den Schnitt der einzelnen Halbräume. Es entsteht ein konvexes Polytop (Simplex).

Die optimale Lösung des LP-Problems ist dann der Schnitt einer sich nähernden Hyperebene (der Zielfunktion) mit einem Eckpunkt oder einer Seite des Polytops (siehe Abb. 1).

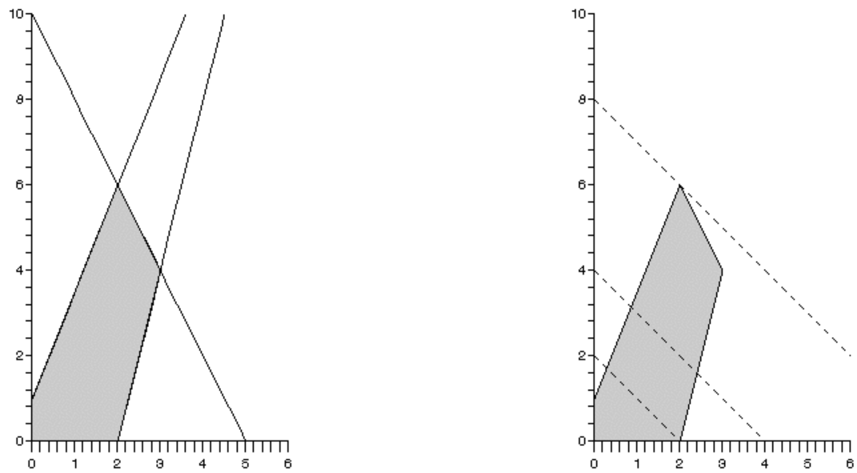

Abb. 1: Geometrische Interpretation der Lösung von LP-Problemen

3 Methoden zur Lösung von LP-Problemen

In der Praxis verwendet man den sog. Simplex-Algorithmus zur Lösung von LP-Problemen. Dessen Laufzeit ist zwar im *worst case* exponentiell, im Durchschnitt aber polynomiell. Es existieren allerdings auch Algorithmen zur Lösung solcher Probleme, die selbst im *worst case* effizient sind.

Anschaulich beginnt der Simplex-Algorithmus an einer Ecke des Polytops und führt eine Kette von Iterationen durch, bei denen der Algorithmus jeweils zu einer benachbarten Ecke springt, deren Wert nicht kleiner als der aktuelle ist. Das Verfahren terminiert in einem lokalen Maximum, das wegen Konvexität des Polytops und Linearität der Zielfunktion ein globales Optimum ist.

4 Lösung des TSP mithilfe von LP

Ein Spezialfall der LP ist die diskrete lineare Programmierung (*integer linear programming*). Zusätzlich zu den Einschränkungsungleichungen muss hier gelten $\mathbf{x} \in \mathbb{Z}^n$. In einigen Fällen gilt sogar $\mathbf{x} \in \{0,1\}^n$. Solche Probleme löst man in der binären diskreten linearen Programmierung. Beide Fälle sind (vermutlich) nicht effizient lösbar, da es sich um $\mathcal{NP}$-harte Probleme handelt.

Um das TSP mithilfe linearer Programmierung zu lösen, stellt man jede Lösung durch einen Inzidenzvektor $\mathbf{x} \in \mathcal{S}$ dar. Zusammen mit dem Kostenvektor $\mathbf{c} \in \mathbb{R}^{n(n-1)/2}$ ergeben sich als zu minimierende Gesamtkosten einer Tour $\mathbf{c}^T\mathbf{x}$.

Da dieses Problem nicht ohne weiteres lösbar ist, formuliert man ein verwandtes LP-Problem, das einfacher zu erfassen ist: minimiere $\mathbf{c}^T\mathbf{x}$ mit $A\mathbf{x} \leq \mathbf{b}$, so dass das Ungleichungssystem von allen $\mathbf{x} \in \mathcal{S}$ erfüllt wird. So erhält man einerseits eine untere Schranke für das eigentliche Problem, andererseits eine Lösung, in deren Nähe man weitersuchen kann.

Als initiales LP-Problem verwendet man hier die Bedingungen $0 \leq x_e \leq 1$ und, da jede Stadt v von genau zwei Kanten besucht wird, $\sum\{x_e \mid v \in e\} = 2$.

5 cutting planes

Findet man eine optimale Lösung $\mathbf{x}^*$ für das vereinfachte Problem, gilt es zwei Fälle zu betrachten: ist $\mathbf{x}^* \in \mathcal{S}$, so hat man optimale Lösung für das TSP gefunden. Andernfalls existiert eine lineare Ungleichung, die von allen Punkten in $\mathcal{S}$ erfüllt wird, aber von $\mathbf{x}^*$ nicht. Diese Ungleichung nennt man *cutting plane* und fügt sie zum Ungleichungssystem hinzu. Dieser Schritt wird wiederholt bis zur Entdeckung einer gültigen Lösung.

Eine der Schwierigkeiten hierbei ist das automatische Finden der *cutting planes*. Hierfür existieren verschiedene Methoden; es werden vor allen Dingen Hypergraph-Schnitte verwendet. Am intuitivsten sind dabei die sogenannten *subtour inequalities*:

Trennt man die Menge der Städte V in zwei echte Teilmengen S und T, so muss jede Tour diese beiden Teilmengen durch mindestens zwei Kanten verbinden, um eine gültige Tour zu erhalten. Dies drückt man aus durch $x(S, V - S) \geq 2$ mit

$$x(S,T) = \sum\{x_e \mid e \cap S \neq \emptyset \neq e \cap T\}$$

6 Zusammenfassung

Lineare Programmierung ist eines der Standardverfahren zur exakten Lösung größerer TSP-Instanzen. Es handelt sich dabei um ein Problem der binären diskreten linearen Programmierung. Diese Gruppe von Problemen in $\mathcal{NP}$- hart und daher nicht effizient lösbar. Zur Lösung kommt man durch iteratives Beschneiden eines hochdimensionalen Polytops mit geeigneten *cut*-Bedingungen.