

## Grundlagen der Optimierung

### Innere-Punkte-Verfahren

Wir betrachten ein allgemeines Innere-Punkte-Verfahren zur Lösung des LP (12.1), also

$$\min c^\top x \quad \text{sodass} \quad Ax = b \quad \text{und} \quad x \geq 0.$$

#### Algorithmus 12.9 (Allgemeines Innere-Punkte-Verfahren)

- 1: Wähle  $(x_0, \lambda_0, s_0) \in \mathcal{F}^0$  und setze  $k := 0$
- 2: **while**  $\mu_k := (x_k)^\top s_k / n > \varepsilon$  **do**
- 3: Wähle  $\sigma_k \in [0, 1]$  und löse

$$\begin{pmatrix} 0 & A^\top & I \\ A & 0 & 0 \\ S_k & 0 & X_k \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta \lambda \\ \Delta s \end{pmatrix} = - \begin{pmatrix} A^\top \lambda_k + s_k - c \\ Ax_k - b \\ X_k S_k e - \sigma_k \mu_k e \end{pmatrix} \quad (12.9)$$

- 4: Setze

$$x_{k+1} := x_k + t_k \Delta x$$

$$\lambda_{k+1} := \lambda_k + t_k \Delta \lambda$$

$$s_{k+1} := s_k + t_k \Delta s$$

Dabei ist  $t_k \in (0, 1]$  eine Schrittweite, die  $x_{k+1} > 0$  und  $s_{k+1} > 0$  garantiert. (Der Newtonschritt wird also bei Bedarf **gedämpft**.)

- 5: Setze  $k := k + 1$
- 6: **end while**

**Zur Erinnerung:** Für  $x_k \in \mathbb{R}^n$  ist  $X_k$  die Diagonalmatrix gebildet aus den Einträgen des Vektors  $x_k$ . Analog ist  $S_k$  gebildet aus den Einträgen von  $s_k$ , und es gilt  $e := (1, \dots, 1)^\top \in \mathbb{R}^n$ .

Der Algorithmus 12.9 lässt noch Freiheiten in der Wahl der Zentrierungsparameters  $\sigma_k$  und der Schrittweite  $t_k$ . Effizient sind sogenannte **Pfadverfolgungsverfahren**, bei denen die Iterierten  $(x_k, \lambda_k, s_k)$  in einer definierten Umgebung des zentralen Pfades gehalten werden. Die Größe der Umgebung sinkt mit der gewichteten Dualitätslücke  $\mu_k$  gegen null. Für Details siehe [Geiger and Kanzow, 2002, Kapitel 4.2] oder Wright [1997].

## Literatur

- C. Geiger and C. Kanzow. *Theorie und Numerik restringierter Optimierungsaufgaben*. Springer, New York, 2002.
- S. Wright. *Primal-Dual Interior Point Methods*. SIAM, Philadelphia, 1997.