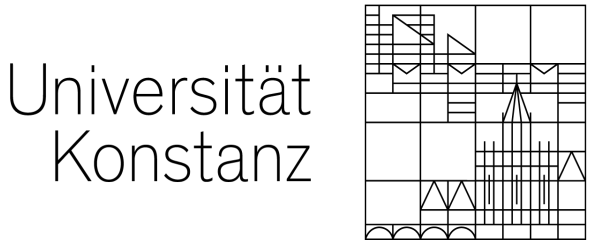
Numerische Verfahren der restringierten Optimierung

Stefan Volkwein



4. Februar 2015

Prof. Dr. Stefan Volkwein, Fachbereich Mathematik und Statistik, Universität Konstanz
E-mail address: stefan.volkwein@uni-konstanz.de

Das Manuskript entstand zur zweistündigen Vorlesung *Numerische Verfahren der restringierten Optimierung* im Wintersemester 2014 am Fachbereich Mathematik und Statistik der Universität Konstanz.

Inhaltsverzeichnis

KAPITEL 1

Optimalitätsbedingungen für die restringierte Optimierung

Das Ziel in diesem Abschnitt ist die Herleitung notwendiger und hinreichender Optimalitätsbedingungen erster und zweiter Ordnung. Für mehr Details und weitere Beispiele verweisen wir auf [**8**, § 10].

1. Nebenbedingungen

Wir betrachten

$$\textbf{(P)} \qquad \min J(x) \quad \text{u.d.N.} \quad x \in \mathbb{R}^n,\ e(x) = 0 \text{ und } g(x) \leq 0,$$

wobei $J : \mathbb{R}^n \to \mathbb{R}$ das *Kosten-* oder *Zielfunktional* ist, $e = (e_1, \ldots, e_m)^T : \mathbb{R}^n \to \mathbb{R}^m$, $m \leq n$, die *Gleichungs-Nebenbedingungen* und $g = (g_1, \ldots, g_p)^T : \mathbb{R}^n \to \mathbb{R}^p$ die *Ungleichungs-Nebenbedingungen* sind. In (**P**) wie im weiteren steht die Abkürzung "u.d.N." für "unter der Nebenbedingung". Wir schreiben $g(x) \leq 0$ genau dann, wenn $g_i(x) \leq 0$ für alle Komponenten $i = 1, \ldots, p$ von g gilt. Ein Punkt $x \in \mathbb{R}^n$ heißt *zulässig* für (**P**), sofern $e(x) = 0$ und $g(x) \leq 0$ erfüllt sind. Die *zulässige Menge* für (**P**) bezeichnen wir mit

$$\mathcal{F}(\mathbf{P}) = \big\{x \in \mathbb{R}^n \,|\, e(x) = 0 \text{ und } g(x) \leq 0\big\}.$$

Bei den Ungleichungen unterscheiden wir zwei Fälle. An $x \in \mathcal{F}(\mathbf{P})$ heißt eine Ungleichungs-Nebenbedingung $g_i(x)$ *aktiv* für ein $i \in \{1, \ldots, p\}$, wenn $g_i(x) = 0$ gilt, und *inaktiv*, wenn $g_i(x) < 0$ erfüllt ist.

In der folgenden Definition wollen wir den Begriff einer Lösung von (**P**) einführen.

Definition 1.1. *Sei x^* ein Punkt im $\mathbb{R}^n$.*

1) *Der Punkt x^* wird* lokale Lösung *von (**P**) genannt, wenn $x^* \in \mathcal{F}(\mathbf{P})$ und $J(x^*) \leq J(x)$ für alle $x \in U(x^*) \cap \mathcal{F}(\mathbf{P})$ gelten, wobei $U(x^*) \subseteq \mathbb{R}^n$ eine Umgebung des Punktes x^* bezeichnet.*
2) *Der Punkt x^* ist eine* strikte lokale Lösung *von (**P**), wenn $x \in \mathcal{F}(\mathbf{P})$ und $J(x^*) < J(x)$ für alle $x \in U(x^*) \cap \mathcal{F}(\mathbf{P})$ erfüllt sind, wobei $U(x^*) \subseteq \mathbb{R}^n$ wieder eine Umgebung des Punktes x^* bezeichnet.*
3) *Der Punkt x^* ist eine* globale Lösung *von (**P**), wenn $x^* \in \mathcal{F}(\mathbf{P})$ und $J(x^*) \leq J(x)$ für alle $x \in \mathcal{F}(\mathbf{P})$ gelten. Wir nennen x^* eine* strikte globale Lösung *von (**P**), wenn $x \in \mathcal{F}(\mathbf{P})$ und $J(x^*) < J(x)$ für alle $x^* \in \mathcal{F}(\mathbf{P})$ erfüllt sind.*

Globale Lösungen sind im allgemeinen schwieriger zu bestimmen als lokale.

Beispiel 1.2. Wir betrachten das Problem

$$\min \|x\|_2^2 \quad \text{u.d.N.} \quad x \in \mathbb{R}^n \text{ und } \|x\|_2^2 \geq 1, \tag{1.1}$$

wobei $\|\cdot\|_2$ die Euklidische Norm bezeichnet. Offenbar ist (1.1) ohne Ungleichungs-Nebenbedingungen ein konvexes, quadratisches Problem mit der eindeutigen Lösung $x^* = 0$. Wegen der Bedingung $\|x\|_2 \geq 1$ löst (**P**) jedes $x \in \mathbb{R}^n$ mit $\|x\|_2 = 1$, insbesondere gibt es unendlich viele Lösungen für $n \geq 2$. ◊

Wenn wir a-priori wissen, dass eine Lösung von (**P**) aktiv ist in allen Komponenten $i \in \{1, \ldots, p\}$, so können wir die Ungleichungs-Nebenbedingungen ignorieren. Wir erhalten dann ein Problem mit Gleichungs-Nebenbedingungen. Diese Probleme werden wir zunächst genauer untersuchen.

2. Die Tangentialebene

Wir setzen nun voraus, dass die beiden Funktionen J und e stetig differenzierbar seien.

Um die *Tangentialebene* einzuführen, werden wir den Begriff der Kurven auf Hyperflächen verwenden. Eine *Kurve* in einer Hyperfläche $\mathcal{H} \subset \mathbb{R}^n$ ist eine Familie von Punkten $x(t) \in \mathcal{H}$, wobei $x : [a, b] \to \mathcal{H}$ eine stetige Abbildung ist. Die Kurve heißt *differenzierbar in* t, wenn $\dot{x}(t) = \frac{\mathrm{d}x}{\mathrm{d}t}(t)$ existiert, und sie *zweimal differenzierbar in* t, wenn $\ddot{x}(t) = \frac{\mathrm{d}^2x}{\mathrm{d}t^2}(t)$ existiert. Wir sagen, die Kurve $x(t)$ *geht durch einen Punkt* $\bar{x} \in \mathcal{H}$, wenn für ein $\bar{t} \in [a, b]$ die Bedingung $x(\bar{t}) = \bar{x}$ gilt. Die Tangentialebene an $\bar{x} \in \mathcal{H}$ ist die Menge der Tangentialvektoren $\dot{x}(\bar{t})$ aller durch den Punkt $\bar{x}$ gehenden differenzierbaren Kurven in $\mathcal{H}$.

Wir definieren die (glatte) Fläche

$$\mathcal{E} = \big\{x \in \mathbb{R}^n \,|\, e(x) = 0\big\} \tag{1.2}$$

in $\mathbb{R}^n$. Unser Ziel ist es nun, die Tangentialebene an einem Punkt $\bar{x} \in \mathcal{E}$ mit Hilfe der Gradienten von e_i, $1 \leq i \leq m$, zu beschreiben. Daher betrachten wir die Menge

$$\operatorname{Kern} \nabla e(\bar{x}) = \big\{v \in \mathbb{R}^n \,|\, \nabla e(\bar{x})v = 0\big\},$$

wobei $\nabla e(\bar{x}) \in \mathbb{R}^{m \times n}$ die Funktional-Matrix

$$\nabla e(\bar{x}) = \begin{pmatrix} \nabla e_1(\bar{x}) \\ \vdots \\ \nabla e_m(\bar{x}) \end{pmatrix}$$

bezeichnet und $\nabla e_i(\bar{x}) \in \mathbb{R}^{1 \times n}$ der Gradient von e_i an $\bar{x}$ ist, $1 \leq i \leq m$.

Wir wollen in Satz 1.4 zeigen, dass $\operatorname{Kern} \nabla e(\bar{x})$ die Tangentialebene an $\mathcal{E}$ im Punkt $\bar{x}$ darstellt. Dazu muss die Funktional-Matrix ∇e am Punkt $\bar{x}$ aber folgende Eigenschaften haben:

Definition 1.3. *Ein Punkt $\bar{x} \in \mathcal{E}$ heißt* regulärer Punkt *(oder einfach* regulär*) bezüglich der Nebenbedingung $e(x) = 0$, wenn die Gradienten $\nabla e_1(\bar{x}), \ldots, \nabla e_m(\bar{x})$ linear unabhängig in $\mathbb{R}^n$ sind.*

In [**12**, Kapitel 1] wird folgender Satz beweisen.

Satz 1.4. *Sei $\bar{x} \in \mathcal{E}$ ein regulärer Punkt. Dann ist die Tangentialebene an $\bar{x}$ gleich der Menge* $\operatorname{Kern} \nabla e(\bar{x})$.

Bemerkung 1.5. Die Voraussetzung, dass $\bar{x}$ regulär ist, stellt keine Voraussetzung an die Menge $\mathcal{E}$ dar, sondern an die Darstellung mittels der Funktion e. Die Tangentialebene an $\bar{x}$ ist unabhängig von der Repräsentation von $\mathcal{E}$ durch die Abbildung e, die Menge $\operatorname{Kern} \nabla e(\bar{x})$ aber offensichtlich nicht. ◊

BEISPIEL 1.6. Seien $n = 2$, $m = 1$ und $e(x_1, x_2) = x_1$. Dann ist die Menge $\mathcal{E}$ die x_2-Achse. Wegen $\nabla e(0, x_2) = (1, 0) \neq (0, 0)$ sind alle Punkte in $\mathcal{E}$ regulär. Repräsentieren wir aber die Menge $\mathcal{E}$ mit der Funktion $e(x_1, x_2) = x_1^2$, erhalten wir allerdings $\nabla e(0, x_2) = (0, 0)$ für alle Punkte aus $\mathcal{E}$. Damit ist in diesem Fall kein Punkt aus $\mathcal{E}$ regulär. ◊

3. Notwendige Bedingungen 1. Ordnung für Gleichungs-Restriktionen

Wir betrachten nun an der Stelle von (**P**) das (in der Regel einfachere) Problem

$$(\mathbf{P}_{Gl}) \qquad \min J(x) \quad \text{u.d.N.} \quad x \in \mathbb{R}^n \text{ und } e(x) = 0.$$

Das Haupt-Resultat dieses Abschnittes lautet wie folgt: Sei $x^* \in \mathbb{R}^n$ ein lokales Minimum von $(\mathbf{P}_{Gl})$ und ein regulärer Punkt. Dann existiert ein *Lagrange-Multiplikator* $\lambda^* = (\lambda_1^*, \ldots, \lambda_m^*)^T \in \mathbb{R}^m$ mit

$$\nabla J(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla e_i(x^*) = 0. \tag{1.3}$$

Die Gleichung (1.3) lässt sich wie folgt interpretieren:

1) Der Gradient von J an x^* liegt im Span der Gradienten der Nebenbedingung:
$$\nabla J(x^*) \in \operatorname{Span}\big\{\nabla e_1(x^*), \ldots, \nabla e_m(x^*)\big\}.$$
2) Schreiben wir
$$\sum_{i=1}^{m} \lambda_i^* \nabla e_i(x^*) = (\lambda^*)^T \nabla e(x^*)$$
und transponieren die Gleichung (1.3), so erhalten wir
$$\nabla e(x^*)^T \lambda^* = -\nabla J(x^*)^T.$$
Damit gilt $\nabla J(x^*)^T \in \operatorname{Bild} \nabla e(x^*)^T$. Der transponierte Gradient $\nabla J(x^*)^T$ liegt damit im Bild der adjungierten/transponierten Matrix $\nabla e(x^*)^T$. Da x^* regulär ist, ist $\nabla e(x^*)$ surjektiv und daher $\nabla e(x^*)^T$ injektiv. Es kann daher also höchstens nur ein λ^* geben.
3) Multiplizieren wir Gleichung (1.3) mit $v \in \operatorname{Kern} \nabla e(x^*)$, so folgt sofort
$$\nabla J(x^*) v = 0 \quad \text{für alle } v \in \operatorname{Kern} \nabla e(x^*).$$
Für Variationen $v \in \operatorname{Kern} \nabla e(x^*)$ erfüllt $x = x^* + v$ die Nebenbedingung $e(x) = 0$ bis zur ersten Ordnung. Damit darf J bezüglich dieser Variation bis zur ersten Ordnung nicht wachsen oder fallen.

BEISPIEL 1.7. Betrachte das Optimierungs-Problem

$$\min J(x) = x_1 + x_2 \quad \text{u.d.N.} \quad x_1^2 + x_2^2 = 2.$$

Wie wir uns leicht grafisch überlegen können, ist $x^* = (-1, -1)$ die eindeutige Lösung. Wir berechnen $\nabla J(x^*) = (1, 1)$ und $\nabla e(x^*) = (-2, -2) \neq (0, 0)$. Damit ist x^* ein regulärer Punkt, und mit $\lambda^* = 1/2$ folgt (1.3). ◊

BEISPIEL 1.8. Wir wollen nun das Problem

$$\min J(x) = x_1 + x_2 \quad \text{u.d.N.} \quad \begin{pmatrix} e_1(x) \\ e_2(x) \end{pmatrix} = \begin{pmatrix} (x_1 - 1)^2 + x_2^2 - 1 \\ (x_1 - 2)^2 + x_2^2 - 4 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

untersuchen. Hier ist die Menge $\mathcal{E}$ ein-elementig, und $x^* = (0,0)$ der einzige zulässige Punkt und damit die (triviale) Lösung des Minimierungs-Problems. Der Gradient von J ist wie im vorangegangenen Beispiel gegeben durch $\nabla J(x^*) = (1,1)$. Ferner ergeben sich wegen $\nabla e_1(x_1,x_2) = (2(x_1-1), 2x_2)$ und $\nabla e_2(x_1,x_2) = (2(x_1-2), 2x_2)$ die Gradienten von e_1 und e_2 an x^* zu $\nabla e_1(x^*) = (-2,0)$ beziehungsweise $\nabla e_2(x^*) = (-4,0)$. Damit sind die Gradienten $\nabla e_1(x^*)$ und $\nabla e_2(x^*)$ linear abhängig in $\mathbb{R}^2$, der Punkt x^* kann also nicht regulär sein. Offenbar löst x^* löst die Minimierungs-Aufgabe, es gibt aber kein $\lambda^* \in \mathbb{R}^2$, so dass (1.3) erfüllt ist. ◊

Wir kommen nun zur Formulierung des Haupt-Resultates von diesem Abschnitt. Für einen Beweis verweisen wir auf [**12**, Kapitel 1].

Satz 1.9 (Notwendige Bedingungen 1. Ordnung). *Sei der Punkt x^* eine lokale Lösung von $(\mathbf{P}_{Gl})$ und ein regulärer Punkt. Dann existiert genau ein $\lambda^* = (\lambda_1^*, \ldots, \lambda_m^*)^T \in \mathbb{R}^m$, der sogenannte Lagrange-Multiplikator, so dass*

$$\nabla J(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla e_i(x^*) = \nabla J(x^*) + (\lambda^*)^T \nabla e(x^*) = 0. \tag{1.4}$$

Bemerkung 1.10.

a) Die notwendigen Optimalitätsbedingungen erster Ordnung

$$\nabla J(x^*) + (\lambda^*)^T \nabla e(x^*) = 0$$

zusammen mit der Nebenbedingung

$$e(x^*) = 0$$

ergeben ein (nichtlineares) Gleichungs-System mit $n+m$ Gleichungen für die $n+m$ Unbekannten $x^* \in \mathbb{R}^n$ und $\lambda^* \in \mathbb{R}^m$.

b) Der Beweis von Satz 1.9 zeigt auch, wie der Lagrange Multiplikator λ^* berechnet werden kann. Wir sortieren den Vektor x um, indem wir $x = (x_B, x_R) \in \mathbb{R}^n$ schreiben mit $x_B \in \mathbb{R}^m$, $x_R \in \mathbb{R}^{n-m}$, so dass $\nabla_B e(x^*) \in \mathbb{R}^{m\times m}$ invertierbar ist. Hierbei bezeichnet ∇_B den Gradient mit den partiellen Ableitungen bezüglich des Vektors x_B. Für gegebene optimale Lösung x^* von $(\mathbf{P}_{Gl})$ löst λ^* das lineare System

$$\nabla_B e(x^*)^T \lambda^* = -\nabla_B J(x^*)^T$$

Offenbar ist $\lambda^* = 0$ im Fall von $\nabla_B J(x^*) = 0$. ◊

Wir bezeichnen mit $\langle \cdot, \cdot \rangle_{\mathbb{R}^m}$ das Euklidische Skalarprodukt im $\mathbb{R}^m$. Mit der Einführung der *Lagrange-Funktion*

$$L(x,\lambda) = J(x) + \lambda^T e(x) = J(x) + \langle \lambda, e(x) \rangle_{\mathbb{R}^m} \tag{1.5}$$

lassen sich die Optimalitätsbedingungen (1.4) und die Nebenbedingung kompakt schreiben mit Hilfe des Gradienten von L:

$$\nabla_x L(x^*,\lambda^*) = \nabla J(x^*) + (\lambda^*)^T \nabla e(x^*) = 0, \tag{1.6a}$$

$$\nabla_\lambda L(x^*,\lambda^*) = e(x^*)^T = 0, \tag{1.6b}$$

also insgesamt $\nabla L(x^*,\lambda^*) = 0$.

Beispiel 1.11. Wir betrachten das Problem

$$\min x_1x_2 + x_2x_3 + x_1x_3 \quad \text{u.d.N.} \quad x_1 + x_2 + x_3 = 3.$$

Um die Form $(\mathbf{P}_{Gl})$ zu erhalten, setzen wir $n = 3$, $m = 1$, $J(x) = x_1x_2+x_2x_3+x_1x_3$ und $e(x) = x_1 + x_2 + x_3 - 3$ für $x = (x_1, x_2, x_3) \in \mathbb{R}^3$. Wegen $\nabla e(x) = (1, 1, 1)$ ist jeder Punkt $x \in \mathbb{R}^3$ regulär. Zum Aufstellen des Gleichungssystems (1.6) führen wir die Lagrange-Funktion gemäß (1.5) ein:

$$L(x, \lambda) = J(x) + \lambda e(x) = x_1x_2 + x_2x_3 + x_1x_3 + \lambda(x_1 + x_2 + x_3 - 3).$$

Dann folgen die beiden Gleichungen:

$$\begin{aligned}\nabla_x L(x, \lambda) &= (x_2 + x_3 + \lambda, x_1 + x_3 + \lambda, x_1 + x_2 + \lambda) = 0,\\ \nabla_\lambda L(x, \lambda) &= x_1 + x_2 + x_3 - 3 = 0.\end{aligned}$$

Nun lassen sich $x^* = (x_1^*, x_2^*, x_3^*) \in \mathbb{R}^3$ und $\lambda^* \in \mathbb{R}$ als Lösung des Gleichungs-Systems

$$\begin{pmatrix} 0 & 1 & 1 & 1\\ 1 & 0 & 1 & 1\\ 1 & 1 & 0 & 1\\ 1 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} x_1^*\\ x_2^*\\ x_3^*\\ \lambda^* \end{pmatrix} = \begin{pmatrix} 0\\ 0\\ 0\\ 3 \end{pmatrix}$$

berechnen. Wir erhalten $x_i^* = 1$ für $i = 1, 2, 3$ und $\lambda^* = -2$. Somit haben wir eine Lösung von (1.6) gefunden. Wir wissen aber nicht, ob es sich bei dem Punkt x^* um eine Minimum, Maximum oder einen Sattelpunkt handelt. ◊

4. Bedingungen 2. Ordnung für Gleichungs-Restriktionen

Seien sowohl das Zielfunktional J als auch die Gleichungs-Nebenbedingung e zweimal stetig differenzierbar. Die folgenden beiden Aussagen sind ebenfalls in [**12**, Kapitel 1] bewiesen.

Satz 1.12 (Notwendige Bedingungen 2. Ordnung). *Sei x^* ein lokales Minimum von J unter der Nebenbedingung $e(x) = 0$. Ferner sei x^* ein regulärer Punkt. Dann ist die $n \times n$-Matrix*

$$\nabla_{xx} L(x^*, \lambda^*) = \nabla^2 J(x^*) + (\lambda^*)^T \nabla^2 e(x^*) = \nabla^2 J(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla^2 e_i(x^*)$$

positiv semi-definit auf $\operatorname{Kern} \nabla e(x^*)$*, das heißt,*

$$v^T \nabla_{xx} L(x^*, \lambda^*) v \geq 0 \quad \textit{für alle } v \in \operatorname{Kern} \nabla e(x^*).$$

wobei λ^* den nach Satz 1.9 eindeutig bestimmten Lagrange-Multiplikator zu x^* bezeichnet und ∇^2 für die zweite Ableitung steht.

Satz 1.13 (Hinreichende Bedingungen 2. Ordnung). *Seien $x^* \in \mathcal{E} \subset \mathbb{R}^n$ und $\lambda^* \in \mathbb{R}^m$ zwei Punkte mit*

$$\nabla J(x^*) + (\lambda^*)^T \nabla e(x^*) = 0. \tag{1.7}$$

Ferner seien x^* ein regulärer Punkt und die Matrix $\nabla_{xx} L(x^*, \lambda^*)$ positiv definit auf $\operatorname{Kern} \nabla e(x^*)$*:*

$$v^T \nabla_{xx} L(x^*, \lambda^*) v > 0 \quad \textit{für alle } v \in \operatorname{Kern} \nabla e(x^*) \setminus \{0\}.$$

Dann ist x^* ein striktes lokales Minimum von J unter der Nebenbedingung $e(x) = 0$, das heißt, x^* ist eine strikte lokale Lösung von $(\mathbf{P}_{Gl})$.

BEISPIEL 1.14. Wir wenden uns nun noch einmal dem Beispiel 1.11 zu. Die Lösung der notwendigen Optimalitätsbedingungen sind $x^* = (1,1,1)$ und $\lambda^* = -2$. Dann folgt

$$\nabla_{xx} L(x^*, \lambda^*) = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}.$$

Die symmetrische Matrix $\nabla_{xx} L(x^*, \lambda^*)$ ist indefinit mit den drei Eigenwerten $\mu_1 = -1$, $\mu_2 = -1$ und $\mu_3 = 2$. Um die hinreichenden Bedingungen 2. Ordnung nachzuprüfen, wählen wir $v = (v_1, v_2, v_3) \in \operatorname{Kern} \nabla e(x^*) \setminus \{0\}$ beliebig. Dann folgen $v_1 + v_2 + v_3 = 0$ und

$$v^T \nabla_{xx} L(x^*, \lambda^*) v = v_1(v_2 + v_3) + v_2(v_1 + v_3) + v_3(v_1 + v_2) = -v_1^2 - v_2^2 - v_3^2 < 0.$$

Damit ist $\nabla_{xx} L(x^*, \lambda^*)$ negativ definit und an x^* liegt kein Minimum, sondern ein Maximum vor. ◊

5. Sensitivitätsanalyse

Sei $x^* \in \mathbb{R}^n$ ein regulärer Punkt und eine lokale Lösung von

$$\min J(x) \quad \text{u.d.N.} \quad e(x) = 0. \tag{1.8}$$

Ferner seien J und e zweimal stetig differenzierbar. Mit $\lambda^* \in \mathbb{R}^m$ bezeichnen wir den nach Satz 1.9 eindeutig bestimmten Lagrange-Multiplikator zu x^*. Wir wollen nun das Problem

$$\min J(x) \quad \text{u.d.N.} \quad e(x) = c \tag{1.9}$$

mit $c \in \mathbb{R}^m$ lösen.

LEMMA 1.15. *Seien $Q \in \mathbb{R}^{n \times n}$ und $A \in \mathbb{R}^{m \times n}$ gegeben, wobei* $\operatorname{Rang} A = m$ *gilt und Q positiv definit auf dem Teilraum* $\operatorname{Kern} A$ *ist. Dann ist die Matrix*

$$\begin{pmatrix} Q & A^T \\ A & 0 \end{pmatrix}$$

invertierbar.

BEWEIS. Angenommen, das Paar $(x, \lambda) \in \mathbb{R}^n \times \mathbb{R}^m$ löst das System

$$Qx + A^T \lambda = 0, \tag{1.10a}$$

$$Ax \qquad\quad = 0. \tag{1.10b}$$

Zu zeigen ist, dass $x = 0$ und $\lambda = 0$ gelten muß. Wir multiplizieren (1.10a) mit x^T von links und erhalten die Gleichung

$$x^T Q x + x^T A^T \lambda = 0.$$

Mit (1.10b) folgen $x^T A^T = 0$ und daher $x^T Q x = 0$. Wegen (1.10b) haben wir aber $x \in \operatorname{Kern} A$, so dass $x = 0$ ist. Aus (1.10a) ergibt sich mit $x = 0$ nun $A^T \lambda = 0$. Nach Voraussetzung gilt $\operatorname{Rang} A = m$. Damit ist A surjektiv und deshalb A^T injektiv. Das bedeutet aber, dass $\lambda = 0$ sein muß. Wir haben damit gezeigt, dass $(x, \lambda) = (0, 0)$ erfüllt ist. □

Nun können wir folgenden Satz beweisen.

SATZ 1.16. *Sei $x^* \in \mathbb{R}^n$ eine lokale Lösung von (1.8) und sei x^* ein regulärer Punkt. Der Vektor $\lambda^* \in \mathbb{R}^m$ bezeichne den Lagrange-Multiplikator zu x^*. Es gelte*

$$v^T \nabla_{xx} L(x^*, \lambda^*) v > 0 \quad \text{für alle } v \in \operatorname{Kern} \nabla e(x^*) \setminus \{0\}. \tag{1.11}$$

Dann gibt es eine Umgebung $U(0) \subset \mathbb{R}^m$ von $0 \in \mathbb{R}^m$, so dass (1.9) eine Lösung $x(c)$ für alle $c \in U(0)$ besitzt, die stetig von c abhängt mit $x(0) = x^*$. Ferner gilt

$$\nabla_c J(x(c))\big|_{c=0} = -(\lambda^*)^T.$$

BEWEIS. Wir betrachten das Gleichungs-System

$$\nabla J(x) + \lambda^T \nabla e(x) = 0, \tag{1.12a}$$

$$e(x) = c. \tag{1.12b}$$

Nach Voraussetzung löst (x^*, λ^*) das System (1.12) für $c = 0$. Die Jacobi-Matrix der Abbildung

$$F(x, \lambda, c) = \begin{pmatrix} \nabla J(x)^T + \nabla e(x)^T \lambda \\ e(x) - c \end{pmatrix}$$

am Punkt $(x^*, \lambda^*, 0) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^m$ bezüglich der Variablen (x, λ) ist

$$\nabla_{(x,\lambda)} F(x^*, \lambda^*, 0) = \begin{pmatrix} \nabla_{xx} L(x^*, \lambda^*) & \nabla e(x^*)^T \\ \nabla e(x^*) & 0 \end{pmatrix} = \nabla^2 L(x^*, \lambda^*).$$

Wegen (1.11) und der Tatsache, dass x^* ein regulärer Punkt ist, ist $\nabla^2 L(x^*, \lambda^*)$ nach Lemma 1.15 invertierbar. Nach dem Satz über Implizite Funktionen gibt es eine Lösung $(x(c), \lambda(c))$ von (1.12) in einer Umgebung $U(0) \subset \mathbb{R}^m$ von $0 \in \mathbb{R}^m$. Diese Lösung hängt sogar stetig differenzierbar von c ab. Da $x \mapsto \nabla e(x)$ stetig ist, ist $\nabla e(x(c))$ surjektiv für alle c in einer eventuell kleineren Umgebung $\tilde{U}(0) \subseteq U(0)$ von 0. Da $\nabla_{xx} L(x, \lambda)$ stetig ist, lässt sich auch zeigen, dass eine Umgebung $\hat{U}(0) \subset \tilde{U}(0)$ existiert mit

$$v^T \nabla_{xx} L(x(c), \lambda(c)) v > 0 \quad \text{für alle } v \in \operatorname{Kern} \nabla e(x(c)) \setminus \{0\}$$

für alle $c \in \hat{U}(c)$, siehe [**11**, Lemma 2.12]. Damit erfüllen die Punkte $(x(c), \lambda(c))$, $c \in \hat{U}(0)$, die hinreichenden Bedingungen 2. Ordnung. Also ist $x(c)$ eine strikte lokale Lösung von (1.9).
Mit der Kettenregel erhalten wir

$$\nabla_c J(x(c))\big|_{c=0} = \nabla J(x^*) \nabla_c x(0)$$

und

$$\nabla_c e(x(c))\big|_{c=0} = \nabla e(x^*) \nabla_c x(0).$$

Wegen (1.12b) folgen $\nabla e(x^*) \nabla_c x(0) = I \in \mathbb{R}^{m \times m}$ und mit (1.12a)

$$\nabla_c J(x(c))\big|_{c=0} = -(\lambda^*)^T,$$

was zu zeigen war. □

6. Ungleichungs-Nebenbedingungen

Seien J und e zweimal stetig differenzierbar. Wir betrachten das Optimierungs-Problem

$$\min J(x) \quad \text{u.d.N.} \quad e(x) = 0 \text{ und } g(x) \leq 0. \tag{P}$$

DEFINITION 1.17. *Sei $x^* \in \mathbb{R}^n$ ein Punkt mit*

$$e(x^*) = 0 \quad \textit{und} \quad g(x^*) \leq 0. \tag{1.13}$$

Mit $\mathcal{A} \subseteq \{1, \ldots, p\}$ bezeichnen wir die Menge der aktiven Indizes *$i \in \mathcal{A}$, für die $g_i(x^*) = 0$ gilt. Der Punkt x^* heißt* regulärer Punkt für (1.13)*, wenn $\nabla e_i(x^*)$, $1 \leq i \leq m$, und $\nabla g_i(x^*)$, $i \in \mathcal{A}$, linear unabhängig sind.*

Einen Beweis der folgenden Aussage finden Sie zum Beispiel in [**12**, Kapitel 1].

SATZ 1.18 (Karush-Kuhn-Tucker). *Sei $x^* \in \mathbb{R}^n$ ein lokales Minimum von* (P) *und sei x^* ein regulärer Punkt für* (1.13)*. Dann existieren Vektoren $\lambda^* \in \mathbb{R}^m$ und $\mu^* \in \mathbb{R}^p$ mit $\mu^* \geq 0$, so dass*

$$\nabla J(x^*) + (\lambda^*)^T \nabla e(x^*) + (\mu^*)^T \nabla g(x^*) = 0, \tag{1.14a}$$

$$(\mu^*)^T g(x^*) = 0. \tag{1.14b}$$

BEMERKUNG 1.19. Die Gleichung (1.14b) wird *Komplementaritäts-Bedingung* genannt. Aus $\mu^* \geq 0$ und $g(x^*) \leq 0$ folgen, dass (1.14b) äquivalent mit der Tatsache ist, dass $\mu^* > 0$ nur dann gelten kann, wenn $i \in \mathcal{A}$ erfüllt ist. ◊

Bei Ungleichungs-Restriktionen führen wir eine gegenüber (1.5) erweiterte Lagrange-Funktion wie folgt ein:

$$L(x, \lambda, \mu) = J(x) + \lambda^T e(x) + \mu^T g(x) \quad \text{für } (x, \lambda, \mu) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p.$$

Offenbar lässt sich (1.14a) in der Form $\nabla_x L(x^*, \lambda^*, \mu^*) = 0$ schreiben.

BEISPIEL 1.20. Wir betrachten das folgende Beispiel:

$$\min \frac{1}{2}(x_1^2 + x_2^2 + x_3^2) \quad \text{u.d.N.} \quad x_1 + x_2 + x_3 \leq -3.$$

Aus (1.14a) erhalten wir die drei Gleichungen

$$x_1^* + \mu^* = 0, \quad x_2^* + \mu^* = 0, \quad x_3^* + \mu^* = 0$$

für ein lokales Minimum $x^* = (x_1^*, x_2^*, x_3^*)$. Es gibt zwei Möglichkeiten:

1) Die Nebenbedingung ist an x^* inaktiv, d.h.,

$$x_1^* + x_2^* + x_3^* < -3.$$

Wegen (1.14b) folgt sofort $\mu^* = 0$. Also erhalten wir direkt $x^* = (0, 0, 0)$, was aber der Ungleichungs-Nebenbedingung widerspricht. Also entfällt diese Möglichkeit.

2) Die Nebenbedingung ist aktiv. Dann haben wir vier Gleichungen zur Verfügung, um die Variablen $x_1^*, x_2^*, x_3^*, \mu^*$ zu berechnen. Wir bekommen das lineare Gleichungs-System

$$\left(\begin{array}{ccc|c} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ \hline 1 & 1 & 1 & 0 \end{array}\right) \begin{pmatrix} x_1^* \\ x_2^* \\ x_3^* \\ \mu^* \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ -3 \end{pmatrix}$$

Die Koeffizienten-Matrix ist nach Lemma 1.15 invertierbar. Wir berechnen die Lösung $x_i^* = -1$, $1 \leq i \leq 3$, und $\mu^* = 1 \geq 0$. Es gibt damit nur einen Kandidaten x^* für ein lokales Minimum. ◊

Notwendige und hinreichende Bedingungen zweiter Ordung für Probleme mit Ungleichungs-Restriktionen werden im wesentlichen dadurch hergeleitet, dass nur die aktiven Nebenbedingungen betrachtet werden.

SATZ 1.21. *Seien J, e, g zweimal stetig differenzierbar und x^* ein regulärer Punkt von* (1.13)*. Ferner nehmen wir an, dass x^* ein relatives Minimum für* (**P**) *ist. Dann existieren Lagrange-Multiplikatoren $\lambda^* \in \mathbb{R}^m$ und $\mu^* \in \mathbb{R}^p$ mit $\mu^* \geq 0$, so dass* (1.14) *gilt und die Hesse-Matrix*

$$\begin{aligned}\nabla_{xx} L(x^*, \lambda^*, \mu^*) &= \nabla^2 J(x^*) + (\lambda^*)^T \nabla^2 e(x^*) + (\mu^*)^T \nabla^2 g(x^*) \\ &= \nabla^2 J(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla^2 e_i(x^*) + \sum_{i=1}^{p} \mu_i^* \nabla^2 g_i(x^*)\end{aligned}$$

positiv semi-definit ist auf dem Tangentialraum zu den aktiven Nebenbedingungen.

BEWEIS. Wenn x^* ein lokales Minimum bezüglich der Nebenbedingung (1.13) ist und die Menge der aktiven Indizes mit $\mathcal{A} = \{i_1, \ldots, i_\ell\}$ bezeichnet wird, ist es auch eines für das Problem

$$\min J(x) \quad \text{u.d.N.} \quad e(x) = 0,\ g_{i_1}(x) = \ldots = g_{i_\ell}(x) = 0$$

(siehe auch im Beweis von Satz 1.18). Daher folgt die Aussage aus Satz 1.12. □

Um hinreichende Kriterien herzuleiten, müssen wir den Fall berücksichtigen, dass die zu aktiven Ungleichungen assozierten Lagrange-Multiplikatoren den Wert Null haben können. Daher muß $\nabla_{xx} L(x^*, \lambda^*, \mu^*)$ positiv definit auf einem größerem Unterraum sein. Ein Beweis des folgenden Satzes kann man wieder in [**12**, Kapitel 1] nachlesen.

SATZ 1.22. *Seien J, e, g zweimal stetig differenzierbar. Hinreichende Bedingung, dass $x^* \in \mathbb{R}^n$ ein striktes lokales Minimum von* (**P**) *ist, ist die Existenz von Vektoren $\lambda^* \in \mathbb{R}^m$ und $\mu^* \in \mathbb{R}^p$, so dass*

$$\mu^* \geq 0, \tag{1.15a}$$

$$(\mu^*)^T g(x^*) = 0, \tag{1.15b}$$

$$\nabla J(x^*) + (\lambda^*)^T \nabla e(x^*) + (\mu^*)^T \nabla g(x^*) = 0 \tag{1.15c}$$

gelten und die Hesse-Matrix

$$\nabla_{xx} L(x^*, \lambda^*, \mu^*) = \nabla^2 J(x^*) + (\lambda^*)^T \nabla^2 e(x^*) + (\mu^*)^T \nabla^2 g(x^*)$$

positiv definit auf dem Unterraum

$$\tilde{\mathcal{K}} = \left\{ v \in \mathbb{R}^n \,|\, \nabla e(x^*) v = 0,\ \nabla g_i(x^*) v = 0 \text{ für } i \in \tilde{\mathcal{A}} \right\}$$

mit $\tilde{\mathcal{A}} = \{ i \in \mathcal{A} \,|\, \mu_i^* > 0 \}$.

BEMERKUNG 1.23. Wegen $\tilde{\mathcal{A}} \subset \mathcal{A}$ folgt

$$\mathcal{K} = \left\{ v \in \mathbb{R}^n \,|\, \nabla e(x^*) v = 0,\ \nabla g_i(x^*) v = 0 \text{ für } i \in \mathcal{A} \right\} \subset \tilde{\mathcal{K}},$$

das heißt, die Menge $\tilde{\mathcal{K}}$ ist im allgemeinen größer als die Menge $\mathcal{K}$. ◊

BEMERKUNG 1.24. Gilt $\tilde{\mathcal{A}} = \mathcal{A}$, so ist die Voraussetzung, dass $\nabla_{xx} L(x^*, \lambda^*, \mu^*)$ positiv definit auf

$$\left\{ v \in \mathbb{R}^n \,|\, \nabla e(x^*) v = 0,\ \nabla g_i(x^*) v = 0 \text{ für } i \in \mathcal{A} \right\}$$

ist, eine hinreichende Bedingung für ein striktes lokales Minimum an x^*. ◊

BEISPIEL 1.25. Wir setzen das Beispiel 1.20 fort und berechnen

$$\nabla^2 J(x^*) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{und} \quad \nabla^2 g(x^*) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

Also ist $\nabla_{xx} L(x^*, \lambda^*, \mu^*)$ gleich der Identität in $\mathbb{R}^{3\times 3}$. Da die Nebenbedingung aktiv ist und $\mu^* = 1 > 0$ gilt, ist eine hinreichende Bedingung für ein striktes lokales Minimum an x^* nach Satz 1.22, dass $\nabla_{xx} L(x^*, \lambda^*, \mu^*)$ positiv definit auf dem Unterraum $\operatorname{Kern} \nabla g(x^*)$ ist. Offenbar ist aber $\nabla_{xx} L(x^*, \lambda^*, \mu^*)$ positiv auf $\mathbb{R}^3$, und damit liegt an x^* ein striktes lokales Minimum vor. ◊

Wir geben noch ein Sensitivitätsresultat an. Ein Beweis basiert auf ähnlichen Argumenten wie denen im Beweis von Satz 1.16.

SATZ 1.26. *Seien J, e, g zweimal stetig differenzierbar. Für $(c, d) \in \mathbb{R}^m \times \mathbb{R}^p$ betrachten wir die Familie von Problemen*

$$\min J(x) \quad \textit{u.d.N.} \quad e(x) = c \textit{ und } g(x) \le d. \tag{1.16}$$

Der Punkt $x^* \in \mathbb{R}^n$ sei regulär und eine lokale Lösung von (1.16) für $(c, d) = (0, 0)$. Ferner erfülle x^* zusammen mit den nach Satz 1.18 zugehörigen Lagrange-Multiplikatoren $\lambda^* \in \mathbb{R}^m$ und $\mu^* \in \mathbb{R}^p$ mit $\mu^* \ge 0$ die hinreichenden Bedingungen 2. Ordnung für ein striktes lokales Minimum (siehe Satz 1.22). Ferner gelte $\mu_i^* > 0$ für alle aktiven Ungleichungs-Restriktionen. Dann existiert eine Umgebung $U \subset \mathbb{R}^{m+p}$ von $(0, 0)$, so dass (1.16) eine lokale Lösung $x = x(c, d)$ zu jedem $(c, d) \in U$ besitzt. Diese Lösung $x(c, d)$ hängt stetig von (c, d) ab mit $x(0, 0) = x^*$. Ferner gelten

$$\nabla_c J(x(c,d))\big|_{(c,d)=(0,0)} = -(\lambda^*)^T \quad \textit{und} \quad \nabla_d J(x(c,d))\big|_{(c,d)=(0,0)} = -(\mu^*)^T.$$

KAPITEL 2

Lineare Programmierung: Innere-Punkte Verfahren

In diesem Abschnitt werden wir uns mit Innere Punkte Verfahren zur Behandlung von linearen Ungleichungs-Nebenbedingungen beschäftigen. Dabei werden wir *Innere-Punkte Verfahren* verwenden.

1. Primal-Duale Verfahren

Wir betrachten das Problem

$$\min c^T x \quad \text{u.d.N.} \quad Ax = b \text{ und } x \geq 0 \tag{2.1}$$

mit $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$ sowie $b \in \mathbb{R}^m$. Da sowohl die Zielfunktion sowie die Nebenbedingungen linear sind, wird (2.1) auch als Problem der *Linearen Programmierung* bezeichnet.

Im Kontext von Abschnitt 1 setzen wir

$$J(x) = c^T x,\ e(x) = b - Ax,\ g(x) = -x \text{ und } p = n.$$

Dann erhalten wir $\nabla J(x) = c^T$, $\nabla e(x) = -A$ sowie $\nabla g(x) = -I$. Sei $x^* \in \mathbb{R}^n$ eine lokale Lösung von (2.1). Da alle Nebenbedingungen, d.h., e und g, linear sind, existieren Lagrange-Multiplikatoren $\lambda^* \in \mathbb{R}^m$ und $\mu^* \in \mathbb{R}^n$ mit $\mu^* \geq 0$, so dass die notwendigen Optimalitäts-Bedingungen erster Ordnung erfüllt sind:

$$\nabla J(x^*) + (\lambda^*)^T \nabla e(x^*) + (\mu^*)^T \nabla g(x^*) = 0. \tag{2.2}$$

Einen Beweis finden wir z.B. in [**9**, S. 351-353]. Für unser Beispiel lautet (2.2):

$$A^T \lambda^* + \mu^* = c. \tag{2.3a}$$

Ferner erfüllt x^* die Gleichungs-Restriktionen:

$$Ax^* = b. \tag{2.3b}$$

Aus der Komplementaritäts-Bedingung $(\mu^*)^T g(x^*) = 0$ folgt

$$\sum_{i=1}^{n} \mu_i^* g_i(x^*) = 0.$$

Nun gelten $\mu_i^* \geq 0$ und $g_i(x^*) \leq 0$ für alle $i = 1, \ldots, n$. Daher können wir die Komplementaritäts-Bedingung auch in der Form

$$\mu_i^* g_i(x^*) = 0 \quad \text{für } i = 1, \ldots, n \tag{2.3c}$$

schreiben. Schließlich sind x^* und μ^* nicht-negativ. Wir drücken das wie folgt aus:

$$(x^*, \mu^*) \geq 0. \tag{2.3d}$$

Primal-Duale Verfahren bestimmen eine Lösung (x^*, λ^*, μ^*) von (2.3) durch Anwendung des Newton-Algothmus auf (2.3a)-(2.3c), wobei die Suchrichtung so modifiziert wird, dass (2.3d) in jeder Iteration strikt erfüllt ist. Daher werden diese Methoden auch *Innere-Punkte Verfahren* genannt.

Die Bedingung (2.3d) macht das Problem (2.3) deutlich schwieriger und ist Ursache für Entwicklung von unterschiedlichen Varianten der Inneren-Punkte Verfahren. Wir führen die Abbildung $F : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n$ durch

$$F(x, \lambda, \mu) = \begin{pmatrix} A^T\lambda + \mu - c \\ Ax - b \\ XMe \end{pmatrix}, \quad (x, \lambda, \mu) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n$$

ein, wobei

$$X = \operatorname{diag}(x_1, \ldots, x_n) \in \mathbb{R}^{n\times n},\ M = \operatorname{diag}(\mu_1, \ldots, \mu_n) \in \mathbb{R}^{n\times n},\ e = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} \in \mathbb{R}^n$$

gesetzt worden sind. Dann lässt sich das Problem (2.3) in der Form

$$F(x, \lambda, \mu) = 0, \tag{2.4a}$$

$$(x, \mu) \geq 0 \tag{2.4b}$$

schreiben. Wie wir bereits oben erwähnt haben, generieren Primal-Duale Verfahren Iterierte (x^k, λ^k, μ^k), die (2.4b) strikt erfüllen, das heißt, es gelten $x^k > 0$ und $\mu^k > 0$ für alle k. Damit werden insbesondere keine Iterierten erzeugt, die (2.4b) nicht erfüllen.

Zulässige Innere-Punkte Verfahren fordern, dass sowohl (2.3a) als auch (2.3b) für alle Iterationen gelten. Wir führen aus diesem Grund zwei Mengen ein:

$$\mathcal{F} = \big\{(x, \lambda, \mu) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n \mid A^T\lambda + \mu = c,\ Ax = b,\ (x, \mu) \geq 0\big\},$$
$$\mathcal{F}^\circ = \big\{(x, \lambda, \mu) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n \mid A^T\lambda + \mu = c,\ Ax = b,\ (x, \mu) > 0\big\}$$

und bezeichnen $\mathcal{F}$ als *zulässige Menge* und $\mathcal{F}^\circ$ als *strikt zulässige Menge*. Die Forderung, dass die Iterierte (x^k, λ^k, μ^k) strikt zulässig ist, lässt sich wie folgt schreiben:

$$(x^k, \lambda^k, \mu^k) \in \mathcal{F}^\circ.$$

Wir wollen nun einen Newton-Schritt für das nicht-lineare System (2.4a) betrachten. Sei die Iterierte (x^k, λ^k, μ^k), $k \geq 0$, gegeben. Dann berechnen wir $(\Delta x^k, \Delta\lambda^k, \Delta\mu^k)$ als Lösung des linearen Gleichungs-Systems

$$\nabla F(x^k, \lambda^k, \mu^k) \begin{pmatrix} \Delta x^k \\ \Delta\lambda^k \\ \Delta\mu^k \end{pmatrix} = -F(x^k, \lambda^k, \mu^k)$$

mit der nichtsymmetrischen Funktional-Matrix

$$\nabla F(x, \lambda, \mu) = \begin{pmatrix} 0 & A^T & I \\ A & 0 & 0 \\ M & 0 & X \end{pmatrix}.$$

Wir bemerken an dieser Stelle, dass die Funktionalmatrix durch die Diagonalmatrizen M und X von den Argumenten x sowie μ, aber nicht von λ abhängen. Die neue Iterierte ist nun gegeben durch

$$(x^{k+1}, \lambda^{k+1}, \mu^{k+1}) = (x^k, \lambda^k, \mu^k) + (\Delta x^k, \Delta\lambda^k, \Delta\mu^k).$$

Im Allgemeinen werden wir keinen vollen Newton-Schritt ausführen können, ohne (2.4b) zu verletzen. Daher bestimmen wir einen Schrittweiten-Parameter $\alpha_k \in (0,1]$, so dass

$$(x^{k+1}, \lambda^{k+1}, \mu^{k+1}) = (x^k, \lambda^k, \mu^k) + \alpha_k(\Delta x^k, \Delta\lambda^k, \Delta\mu^k)$$

die Bedingung (2.4b) erfüllt. Oft muss aber dann der Parameter α_k sehr klein gewählt werden, um (2.4b) für die nächste Iterierte zu garantieren.

Primal-Duale Verfahren modifizieren daher wie folgt:

1) Sie "zwingen" die Suchrichtung in das Innere von $\mathcal{F}^\circ$, so dass wir ein größeres α_k wählen können, ohne (2.4b) zu verletzen.
2) Sie verhindern, dass die Komponenten von x und μ "zu nahe" an die Null kommen.

Wir führen den *zentralen Pfad* $\mathcal{C}$ ein, eine durch $\tau > 0$ parametrisierte Kurve in $\mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n$, wobei die Punkte $(x^\tau, \lambda^\tau, \mu^\tau) \in \mathcal{C}$ Lösungen des folgenden nichtlinearen Gleichungs-Systems sind:

$$A^T\lambda + \mu = c, \tag{2.5a}$$

$$Ax = b, \tag{2.5b}$$

$$x_i\mu_i = \tau, \quad i \in \{1, \dots, n\}, \tag{2.5c}$$

$$(x, \mu) > 0. \tag{2.5d}$$

In (2.5d) bedeutet $(x, \mu) > 0$, dass sowohl $x > 0$ als auch $\mu > 0$ gelten. Anstatt von (2.3c) verlangen wir, dass alle Produkte von x_i^τ und μ_i^τ gleich $\tau > 0$ sind. Der zentrale Pfad ist daher die Menge

$$\mathcal{C} = \big\{(x^\tau, \lambda^\tau, \mu^\tau) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n \,|\, (x^\tau, \lambda^\tau, \mu^\tau) \text{ löst (2.5) für ein } \tau > 0\big\}.$$

Es kann gezeigt werden, dass für jedes $\tau > 0$ genau eine Lösung $(x^\tau, \lambda^\tau, \mu^\tau)$ existiert, wenn $\mathcal{F}^\circ \neq \emptyset$ gilt. Wir können (2.5) in der Form

$$\tilde{F}(x^\tau, \lambda^\tau, \mu^\tau) = F(x^\tau, \lambda^\tau, \mu^\tau) - \begin{pmatrix} 0 \\ 0 \\ \tau e \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \tag{2.6}$$

schreiben. Damit approximiert (2.5) das System (2.3) zunehmend besser für $\tau \to 0$. Wenn für eine Folge $\{\tau_n\}_{n=0}^\infty$ mit $\tau_n > 0$ für alle n und $\lim_{n\to\infty} \tau_n = 0$ die Folge $\{(x^{\tau_n}, \lambda^{\tau_n}, \mu^{\tau_n})\}_{n=0}^\infty$ für $n \to \infty$ gegen ein Grenzelement (x^*, λ^*, μ^*) konvergiert, so löst (x^*, λ^*, μ^*) das System (2.3).

Primal-Duale Verfahren wählen für $\tau > 0$ Newton-Schritte zum zentralen Pfad $\mathcal{C}$. Sei $\sigma \in [0, 1]$, und die sogenannte *gewichtete Dualitäts-Lücke* definiert durch

$$\eta = \frac{1}{n}\sum_{i=1}^n x_i\mu_i = \frac{x^T\mu}{n}.$$

Im Englischen werden die Parameter σ und η *centering parameter* beziehungsweise *duality measure* genannt. Wir schreiben $\tau = \sigma\eta$ und wenden bei festem τ einen Newton-Schritt auf (2.6) an, das heißt, auf das System $\tilde{F}(x^\tau, \lambda^\tau, \mu^\tau) = 0$:

$$\begin{pmatrix} 0 & A^T & I \\ A & 0 & 0 \\ M^k & 0 & X^k \end{pmatrix} \begin{pmatrix} \Delta x^k \\ \Delta\lambda^k \\ \Delta\mu^k \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -X^kM^ke + \sigma\eta e \end{pmatrix},$$

wobei $X^k = \operatorname{diag}(x_1^k, \ldots, x_n^k)$ und $M^k = \operatorname{diag}(\mu_1^k, \ldots, \mu_n^k)$ gelten. Das Lösungs-Tripel $(\Delta x^k, \Delta\lambda^k, \Delta\mu^k)$ ist damit ein Newton-Schritt in Richtung des Punktes $(x^{\sigma\eta}, \lambda^{\sigma\eta}, \mu^{\sigma\eta})$ mit $x_i^{\sigma\eta}\mu_i^{\sigma\eta} = \sigma\eta$ bei festen Werten von σ und η. Im Fall von $\sigma = 0$ erhalten wir wieder den Newton-Schritt für (2.4a), für $\sigma = 1$ hingegen einen Schritt in Richtung von $(x^\eta, \lambda^\eta, \mu^\eta)$. Oft wird $\sigma \in (0, 1)$ gewählt.

Wir wollen nun den Primal-Dualen Algorithmus formulieren.

ALGORITHMUS 2.1 (Primal-Duales Verfahren).

1) *Wähle* $(x^0, \lambda^0, \mu^0) \in \mathcal{F}^\circ$ *und setze* $k = 0$.
2) *Löse das lineare Gleichungs-System*

$$\begin{pmatrix} 0 & A^T & I \\ A & 0 & 0 \\ M^k & 0 & X^k \end{pmatrix} \begin{pmatrix} \Delta x^k \\ \Delta\lambda^k \\ \Delta\mu^k \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -X^k M^k e + \sigma_k \eta_k e \end{pmatrix} \tag{2.7}$$

mit $\sigma_k \in [0, 1]$ *und* $\eta_k = (x^k)^T \mu^k / n$. *Setze*

$$(x^{k+1}, \lambda^{k+1}, \mu^{k+1}) = (x^k, \lambda^k, \mu^k) + \alpha_k (\Delta x^k, \Delta\lambda^k, \Delta\mu^k), \tag{2.8}$$

wobei $\alpha_k \in (0, 1]$ *derart bestimmt wird, dass* $(x^{k+1}, \mu^{k+1}) > 0$ *erfüllt ist.*
3) *Sofern kein Abbruch-Kriterium eintritt, setze* $k = k + 1$ *und gehe zurück zu Schritt* 2).

BEMERKUNG 2.2.

1) Die Wahl der Parameter σ_k und α_k führt zu unterschiedlichen Varianten von Algorithmus 2.1.
2) Für den Startwert haben wir $(x^0, \lambda^0, \mu^0) \in \mathcal{F}^\circ$. Mittels Induktion können wir zeigen, dass $(x^k, \lambda^k, \mu^k) \in \mathcal{F}^\circ$ für alle $k \geq 1$ gilt.
3) Bei *nicht-zulässigen Innere-Punkte Verfahren* fordern wir nur $(x^0, \mu^0) > 0$. Mit der Notation

$$r_b(x) = Ax - b \quad \text{und} \quad r_c(\lambda, \mu) = A^T\lambda + \mu - c$$

lösen wir dann an der Stelle von (2.7) das lineare System

$$\begin{pmatrix} 0 & A^T & I \\ A & 0 & 0 \\ M^k & 0 & X^k \end{pmatrix} \begin{pmatrix} \Delta x^k \\ \Delta\lambda^k \\ \Delta\mu^k \end{pmatrix} = \begin{pmatrix} -r_c(\lambda^k, \mu^k) \\ -r_b(x^k) \\ -X^k M^k e + \sigma_k \eta_k e \end{pmatrix}.$$

Gilt $\alpha_{\bar{k}} = 1$ für ein $\bar{k} \geq 0$, so folgen $r_b(x^k) = 0$ und $r_c(\lambda^k, \mu^k) = 0$ für alle $k > \bar{k}$. Das liegt an der Linearität der beiden Gleichungen (2.5a) und (2.5b). ◊

2. Pfad-Verfolgungs Verfahren

Pfad-Verfolgungs Verfahren restringieren die Iterierten auf eine Umgebung des zentralen Pfades $\mathcal{C}$ und folgen der Kurve $\mathcal{C}$ zu einer Lösung des linearen Optimierungs-Problems. Dabei wird der Ausdruck

$$\eta_k = \frac{1}{n} \sum_{i=1}^{n} x_i^k \mu_i^k$$

sukzessive für $k \to \infty$ verkleinert. Für $\theta, \gamma \in (0, 1]$ definieren wir folgende Umgebungen des zentralen Pfades $\mathcal{C}$:

$$\mathcal{U}_2(\theta) = \left\{(x, \lambda, \mu) \in \mathcal{F}^\circ \mid \|XMe - \eta e\|_2 \leq \theta\eta\right\}$$
$$\mathcal{U}_{-\infty}(\gamma) = \left\{(x, \lambda, \mu) \in \mathcal{F}^\circ \mid x_i\mu_i \geq \gamma\eta \text{ für } i = 1, \ldots, n\right\}$$

Typische Werte sind $\theta = 0.5$ und $\gamma = 10^{-3}$. In $\mathcal{U}_{-\infty}(\gamma)$ fordern wir $x_i\mu_i \geq \gamma\eta$ für jede Komponente des Vektors XMe. Wir nähern uns im Grenzfall $\gamma \to 0$ der Menge $\mathcal{F}$. In $\mathcal{U}_2(\theta)$ sind die Anforderungen im allgemeinen restriktiver.

BEISPIEL 2.3. Wir wählen als Beispiel im $\mathbb{R}^2$ die Vektoren $x = (11, 1)^T$ und $\mu = (1, 1)^T$. Dann gilt $(x, \mu) > 0$. Ferner bekommen wir

$$\eta = \frac{11 \cdot 1 + 1 \cdot 1}{2} = 6.$$

Mit $\gamma = 1/6$ folgt $x_i\mu_i \geq \gamma\eta$ für $i = 1, 2$. Für die Euklidische Norm hingegen erhalten wir

$$\|XMe - \eta e\|_2 = \sqrt{(11-6)^2 + (1-6)^2} = \sqrt{50} > 7 > \theta\eta \quad \text{für alle } \theta \in (0, 1].$$

Damit erfüllt das Paar (x, μ) die Ungleichungs-Bedingung in $\mathcal{U}_{-\infty}(\gamma)$ für $\gamma = 1/6 \in (0, 1]$, allerdings nicht die entsprechende Bedingung in $\mathcal{U}_2(\theta)$ für ein $\theta \in (0, 1]$. $\Diamond$

ALGORITHMUS 2.4 (Zulässiges Pfad-Verfolgungs Verfahren).

1) *Wähle* $\gamma \in (0, 1)$, $0 < \underline{\sigma} < \overline{\sigma} < 1$, $\varepsilon \in (0, 1)$, $w^0 = (x^0, \lambda^0, \mu^0) \in \mathcal{U}_{-\infty}(\gamma)$ *und setze* $k = 0$.
2) *Ist* $\eta_k = (x^k)^T\mu^k/n \leq \varepsilon$ *erfüllt, dann STOPP.*
3) *Wähle* $\sigma_k \in [\underline{\sigma}, \overline{\sigma}]$ *und bestimme eine Lösung*

$$\Delta w^k = (\Delta x^k, \Delta\lambda^k, \Delta\mu^k)$$

des linearen Gleichungs-Systems (2.7). *Sei* α_k *die größte Schrittweite* $\alpha \in (0, 1]$ *mit*

$$w^k(\alpha) = (x^k + \alpha\Delta x^k, \lambda^k + \alpha\Delta\lambda^k, \mu^k + \alpha\Delta\mu^k) \in \mathcal{U}_{-\infty}(\gamma).$$

4) *Setze* $w^{k+1} = w^k(\alpha_k)$, $k = k + 1$ *und gehe zurück zu Schritt* 2).

BEMERKUNG 2.5.

1) Algorithmus 2.4 ist ein Spezialfall von Algorithmus 2.1. Bei der Wahl von σ_k bestehen weiter Freiheitsgrade, allerdings sind $\sigma_k = 0$ und $\sigma_k = 1$ ausgeschlossen. Ferner haben wir gleichmäßige Schranken für die σ_k's: $0 < \underline{\sigma} \leq \sigma_k \leq \overline{\sigma} < 1$ für alle $k \geq 0$.
2) Die Wahl von α_k ist in Algorithmus 2.4 fest vorgeschrieben. Die Konvergenzeigenschaften des Verfahrens ändern sich aber nicht, wenn wir geeignete Backtracking-Strategien verwenden:

$i_{\max} \in \mathbb{N}$; $\epsilon \in (0, 1)$; $i = 0$; $\alpha_k^{(0)} = 1$;
while $\left(w^k(\alpha_k^{(i)}) \notin \mathcal{U}_{-\infty}(\gamma)$ **and** $i \leq i_{\max}$ **and** $\alpha_k^{(i)} > \epsilon\right)$
$\alpha_k^{(i+1)} = \beta\alpha_k^{(i)}$; $i = i + 1$;
end;
$\alpha_k = \alpha_k^{(i)}$;

mit einem Parameter $\beta \in (0, 1)$. Bei einer Wahl $\beta \approx 1$ lässt sich das α_k aus Algorithmus 2.4, Schritt 3), recht gut bestimmen, es sind aber eventuell viele Iterationen in der while-Schleife der Backtracking-Strategie notwendig. Ist $\beta \approx 0$, so ist die while-Schleife schnell beendet, der Schrittweiten-Parameter α_k wird aber in der Regel deutlich kleiner sein als der von Algorithmus 2.4 in Schritt 3). $\Diamond$

3. Konvergenz-Analyse für Algorithmus 2.4

In diesem Abschnitt beschäftigen wir uns mit der Untersuchung der Konvergenz von Algorithmus 2.4. Es wird sich herausstellen, dass der Algorithmus folgende beiden Eigenschaften besitzt:

- das Verfahren bricht nach endlich vielen Iterationen mit Schritt 2.) ab,
- die Anzahl der Iterationen hängt polynomial von der Anzahl der Unbekannten ab (*polynomiale Komplexität*).

Wir bemerken an der Stelle, dass das Simplex-Verfahren aus der Linearen Programmierung kein polynomiales Verfahren ist. Es gibt dazu das Gegenbeispiel von Klee und Minty.

Zur Untersuchung der Konvergenz-Eigenschaften sind einige Hilfsresultate notwendig.

LEMMA 2.6. *Seien $u, v \in \mathbb{R}^n$ zwei Vektoren mit $u^T v \geq 0$. Dann gilt*

$$\|UVe\|_2 \leq 2^{-3/2} \|u + v\|_2^2$$

wobei $U = \operatorname{diag}(u_1, \ldots, u_n)$, $V = \operatorname{diag}(v_1, \ldots, v_n)$ Diagonalmatrizen aus $\mathbb{R}^{n \times n}$ sind und $e = (1, \ldots, 1)^T \in \mathbb{R}^n$ gilt.

BEWEIS. Zunächst gilt für beliebige Zahlen $\alpha, \beta \in \mathbb{R}$:

$$\frac{1}{4}(\alpha + \beta)^2 = \frac{1}{4}(\alpha - \beta)^2 + \alpha\beta \geq \alpha\beta. \tag{2.9}$$

Wegen $u^T v \geq 0$ erhalten wir

$$0 \leq u^T v = \sum_{u_i v_i \geq 0} u_i v_i + \sum_{u_i v_i < 0} u_i v_i = \sum_{i \in \mathcal{P}} u_i v_i - \sum_{i \in \mathcal{M}} |u_i v_i| \tag{2.10}$$

mit der Menge $\mathcal{P} = \{i \in \{1, \ldots, n\} \mid u_i v_i \geq 0\}$ der nicht-negativen Indizes und der Menge $\mathcal{M} = \{i \in \{1, \ldots, n\} \mid u_i v_i < 0\}$ der negativen Indizes. Weiter haben wir die Ungleichung

$$\|x\|_2 \leq \|x\|_1 = \sum_{i=1}^{n} |x_i| \quad \text{für alle } x \in \mathbb{R}^n \tag{2.11}$$

zur Verfügung. Die Beziehung (2.11) sehen wir wie folgt: Für $x = 0$ ist nichts zu zeigen. Sei nun $\alpha = \|x\|_1 > 0$. Dann gilt $|x_i|/\alpha \leq 1$ für alle $1 \leq i \leq n$ und wir erhalten

$$\frac{\|x\|_2^2}{\|x\|_1^2} = \sum_{i=1}^{n} \left(\frac{x_i}{\alpha}\right)^2 \leq \sum_{i=1}^{n} \frac{|x_i|}{\alpha} = 1,$$

woraus direkt (2.11) folgt. Bevor wir nun die Aussage des Lemmas beweisen können, führen wir noch eine Notation ein: Für den Vektor $w \in \mathbb{R}^n$ mit den Komponenten $w_i = u_i v_i$, $i = 1, \ldots, n$, schreiben wir $[u_i v_i]_{i \in \mathcal{P}}$ für den Vektor, der nur die nicht-negativen Komponenten von w_i enthält und $[u_i v_i]_{i \in \mathcal{M}}$ für den Vektor, der nur die

negativen Komponenten enthält. Nun ergibt sich aus (2.9)-(2.11):

$$\begin{aligned}\|UVe\|_2 &= \Big(\|[u_iv_i]_{i\in\mathcal{P}}\|_2^2 + \|[u_iv_i]_{i\in\mathcal{M}}\|_2^2\Big)^{1/2}\\ &\overset{(2.11)}{\leq} \Big(\|[u_iv_i]_{i\in\mathcal{P}}\|_1^2 + \|[u_iv_i]_{i\in\mathcal{M}}\|_1^2\Big)^{1/2}\\ &\overset{(2.10)}{\leq} \Big(2\,\|[u_iv_i]_{i\in\mathcal{P}}\|_1^2\Big)^{1/2} = \sqrt{2}\,\|[u_iv_i]_{i\in\mathcal{P}}\|_1\\ &\overset{(2.9)}{\leq} \sqrt{2}\,\big\|\big[\tfrac{1}{4}(u_i+v_i)^2\big]_{i\in\mathcal{P}}\big\|_1 = 2^{-3/2}\sum_{i\in\mathcal{P}}(u_i+v_i)^2\\ &\leq 2^{-3/2}\sum_{i=1}^{n}(u_i+v_i)^2 = 2^{-3/2}\,\|u+v\|_2^2,\end{aligned}$$

was zu zeigen war. □

Seien $\Delta x^k = (\Delta x_1^k, \ldots, \Delta x_n^k)^T$ und $\Delta\mu^k = (\Delta\mu_1^k, \ldots, \Delta\mu_n^k)^T$ die Lösungen von (2.7). Wir führen die beiden folgenden $n \times n$-Diagonal-Matrizen ein:

$$\Delta X^k = \operatorname{diag}(\Delta x_1^k, \ldots, \Delta x_n^k) \quad \text{und} \quad \Delta M^k = \operatorname{diag}(\Delta\mu_1^k, \ldots, \Delta\mu_n^k).$$

LEMMA 2.7. *Sei $(x^k, \lambda^k, \mu^k) \in \mathcal{U}_{-\infty}(\gamma)$. Dann folgt*

$$\|\Delta X^k \Delta M^k e\|_2 \leq 2^{-3/2}\left(1 + \frac{1}{\gamma}\right) n\eta_k,$$

wobei $\eta_k = (x^k)^T\mu^k/n$ die aktuelle gewichtete Dualitätslücke bezeichnet.

BEWEIS. Nach Voraussetzung gelten $x_i^k\mu_i^k \geq \gamma\eta_k > 0$, so dass alle Komponenten auf den Diagonalen von X^k und M^k positiv sind. Aus der dritten Blockzeile von (2.7) ergibt sich

$$M^k\Delta x^k + X^k\Delta\mu^k = -X^kM^ke + \sigma_k\eta_ke.$$

Multiplizieren wir diese Gleichung mit $(X^kM^k)^{-1/2}$ von links und verwenden die Abkürzung $D^k = (X^k)^{1/2}(M^k)^{-1/2}$ so erhalten wir

$$\begin{aligned}&(M^k)^{-1/2}(X^k)^{-1/2}M^k\Delta x^k + (M^k)^{-1/2}(X^k)^{1/2}\Delta\mu^k\\ &\qquad = (X^kM^k)^{-1/2}\big(-X^kM^ke + \sigma_k\eta_ke\big).\end{aligned}$$

Da X^k und M^k Diagonalmatrizen sind, erhalten wir

$$(D^k)^{-1}\Delta x^k + D^k\Delta\mu^k = (X^kM^k)^{-1/2}\big(-X^kM^ke + \sigma_k\eta_ke\big). \tag{2.12}$$

Weiter haben wir

$$(\Delta x^k)^T\Delta\mu^k = 0. \tag{2.13}$$

Denn aus der ersten Blockzeile in (2.7) folgen $A^T\Delta\lambda^k + \Delta\mu^k = 0$ und daher

$$(\Delta x^k)^TA^T\Delta\lambda^k + (\Delta x^k)^T\Delta\mu^k = 0.$$

Wegen der zweiten Blockzeile in (2.7) gilt $(\Delta x^k)^TA^T = 0$ und damit bekommen wir (2.13). Setzen wir $u = (D^k)^{-1}\Delta x^k$ und $v = D^k\Delta\mu^k$, so bekommen wir mit

Lemma 2.6 und (2.12)

$$\begin{aligned}\|\Delta X^k \Delta M^k e\|_2 &= \|((D^k)^{-1}\Delta X^k)(D^k \Delta M^k)e\|_2 \\ &\le 2^{-3/2}\,\|(D^k)^{-1}\Delta x^k + D^k \Delta\mu^k\|_2^2 \\ &= 2^{-3/2}\,\|(X^k M^k)^{-1/2}(-X^k M^k e + \sigma_k\eta_k e)\|_2^2.\end{aligned} \tag{2.14}$$

Aus $(x^k)^T\mu^k = n\eta_k$, $e^T e = n$, $x_i^k\mu_i^k \ge \gamma\eta_k$ für $i = 1,\dots,n$ und $\sigma_k \in (0,1)$ erhalten wir

$$\begin{aligned}\|\Delta X^k \Delta M^k e\|_2 &\le 2^{-3/2}\,\|-(X^k M^k)^{1/2}e + \sigma_k\eta_k (X^k M^k)^{-1/2}e)\|_2^2 \\ &= 2^{-3/2}\left((x^k)^T\mu^k - 2\sigma_k\eta_k e^T e + \sigma_k^2\eta_k^2 \sum_{i=1}^n \frac{1}{x_i^k\mu_i^k}\right) \\ &\le 2^{-3/2}\left((x^k)^T\mu^k - 2\sigma_k\eta_k e^T e + \sigma_k^2\eta_k \frac{n}{\gamma}\right) \\ &= 2^{-3/2}\left(n\eta_k - 2\sigma_k\eta_k n + \frac{n\sigma_k^2\eta_k}{\gamma}\right) \\ &= 2^{-3/2} n\eta_k\left(1 - 2\sigma_k + \frac{\sigma_k^2}{\gamma}\right) < 2^{-3/2} n\eta_k\left(1 + \frac{1}{\gamma}\right),\end{aligned}$$

was zu zeigen war. □

Wir geben als nächstes eine obere Schranke für die Schrittweite α_k an. Dieses Resultat kann als der wesentliche Schritt zum Nachweis der polynomialen Komplexität von Algorithmus 2.4 angesehen werden.

LEMMA 2.8. *Sei die Iterierte $(x^k, \lambda^k, \mu^k) \in \mathcal{U}_{-\infty}(\gamma)$ gegeben. Dann gilt*

$$(x^k(\alpha), \lambda^k(\alpha), \mu^k(\alpha)) \in \mathcal{U}_{-\infty}(\gamma)$$

für alle $\alpha \in (0, \bar\alpha_k]$ mit

$$\bar\alpha_k = 2^{3/2}\gamma\,\frac{\sigma_k}{n}\,\frac{1-\gamma}{1+\gamma}.$$

BEWEIS. Aus der dritten Blockzeile von (2.7) ergibt sich

$$\mu_i^k \Delta x_i^k + x_i^k \Delta\mu_i^k = -x_i^k\mu_i^k + \sigma_k\eta_k \quad \text{für } i = 1,\dots,n. \tag{2.15}$$

Anwendung von Lemma 2.7 führt auf

$$|\Delta x_i^k \Delta\mu_i^k| \le \|\Delta X^k \Delta M^k e\|_2 \le 2^{-3/2}\left(1 + \frac{1}{\gamma}\right) n\eta_k \quad \text{für } i = 1,\dots,n. \tag{2.16}$$

Mit $x_i^k\mu_i^k \ge \gamma\eta_k$ für $i = 1,\dots,n$, (2.15) und (2.16) folgt

$$\begin{aligned}x_i^k(\alpha)\mu_i^k(\alpha) &= (x_i^k + \alpha\Delta x_i^k)(\mu_i^k + \alpha\Delta\mu_i^k) \\ &= x_i^k\mu_i^k + \alpha\big(x_i^k\Delta\mu_i^k + \mu_i^k\Delta x_i^k\big) + \alpha^2\Delta x_i^k\Delta\mu_i^k \\ &\overset{(2.15)}{\ge} (1-\alpha)x_i^k\mu_i^k + \alpha\sigma_k\eta_k - \alpha^2|\Delta x_i^k\Delta\mu_i^k| \\ &\overset{(2.16)}{\ge} (1-\alpha)\gamma\eta_k + \alpha\sigma_k\eta_k - 2^{-3/2}\alpha^2 n\eta_k\left(1 + \frac{1}{\gamma}\right)\end{aligned} \tag{2.17}$$

für $i = 1,\dots,n$ und für $\alpha \in [0,1]$. Die dritte Blockzeile von (2.7) lautet in Matrix-Schreibweise

$$M^k\Delta x^k + X^k\Delta\mu^k = -X^k M^k e + \sigma_k\eta_k e.$$

Summation über die n Komponenten dieser Gleichung liefert

$$(\mu^k)^T \Delta x^k + (x^k)^T \Delta \mu^k = -(1-\sigma_k)(x^k)^T \mu^k.$$

Zusammen mit (2.13) ergibt sich daher

$$\begin{aligned}(x^k(\alpha))^T \mu^k(\alpha) &= (x^k)^T \mu^k + \alpha\big((\mu^k)^T \Delta x^k + (x^k)^T \Delta \mu^k\big) \\ &= (x^k)^T \mu^k (1 - \alpha(1-\sigma_k)).\end{aligned} \tag{2.18}$$

Daher ist die Bedingung

$$x_i^k(\alpha)\mu_i^k(\alpha) \geq \gamma \eta_k(\alpha) = \gamma \, \frac{(x^k(\alpha))^T \mu^k(\alpha)}{n} \overset{(2.18)}{=} \gamma\big(1 - \alpha(1-\sigma_k)\big)\eta_k \tag{2.19}$$

für $i = 1, \ldots, n$ erfüllt, sofern wegen (2.17)

$$\gamma(1-\alpha)\eta_k + \alpha\sigma_k\eta_k - 2^{-3/2}\alpha^2\left(1 + \frac{1}{\gamma}\right)n\eta_k \geq \gamma\big(1 - \alpha(1-\sigma_k)\big)\eta_k$$

erfüllt ist. Eine Umformung der Terme zeigt, dass dies äquivalent ist zu

$$\alpha\sigma_k\eta_k(1-\gamma) \geq 2^{-3/2}\alpha^2 n\eta_k\left(1 + \frac{1}{\gamma}\right).$$

Letzteres führt auf die Bedingung

$$\alpha \leq 2^{3/2}\gamma \, \frac{\sigma_k}{n} \, \frac{1-\gamma}{1+\gamma} = \bar{\alpha}_k.$$

Nun ist nur noch $(x^k(\alpha), \lambda^k(\alpha), \mu^k(\alpha)) \in \mathcal{F}^\circ$ für alle $\alpha \in [0, \bar{\alpha}_k]$ zu zeigen. Nach Voraussetzung haben wir $Ax^k = b$ und $A^T\lambda^k + \mu^k = c$. Wegen (2.7) gelten dann offenbar

$$Ax^k(\alpha) = b \quad \text{und} \quad A^T\lambda^k(\alpha) + \mu^k(\alpha) = c$$

für alle $\alpha \geq 0$. Aus $\gamma \in (0,1)$ bekommen wir $\gamma(1-\gamma) \leq 1/4$. Also

$$\alpha \leq 2^{3/2}\gamma \, \frac{\sigma_k}{n} \, \frac{1-\gamma}{1+\gamma} \leq 2^{3/2} \, \frac{1}{4} \, \frac{\overline{\sigma}}{n} \, \frac{1}{1+\gamma} < \frac{1}{\sqrt{2}n} < 1.$$

Wegen $x^k = x^k(0) > 0$ und $\mu^k = \mu^k(0) > 0$ folgt aus (2.19) und dem gerade bewiesenem Teil für alle $\alpha \in (0, \bar{\alpha}_k] \subsetneq [0,1]$:

$$x_i^k(\alpha)\mu_i^k(\alpha) \geq \gamma\big(1 - \underbrace{\alpha(1-\sigma_k)}_{<1}\big)\eta_k > 0 \quad \text{für } i = 1, \ldots, n.$$

Also kann kein Index $i \in \{1, \ldots, n\}$ und kein $\alpha \in [0, \bar{\alpha}_k]$ existieren mit $x_i^k(\alpha) = 0$ oder $\mu_i^k(\alpha) = 0$. □

Nun können wir die Reduktion von η_k abschätzen.

Satz 2.9. *Sei $\{(x^k, \lambda^k, \mu^k)\}_{k=0}^\infty$ eine durch Algorithmus 2.4 erzeugte Folge. Dann gilt*

$$\eta_{k+1} \leq \left(1 - \frac{\delta}{n}\right)\eta_k \quad \textit{für alle } k \geq 0$$

für eine von k unabhängige Konstante $\delta > 0$.

BEWEIS. Wegen Lemma 2.8 erhalten wir

$$\alpha_k \geq \bar{\alpha}_k = 2^{3/2}\gamma\,\frac{\sigma_k}{n}\,\frac{1-\gamma}{1+\gamma} \quad \text{für alle } k \geq 0.$$

Aus (2.18) ergibt sich daher

$$\begin{aligned} \eta_{k+1} &= \eta_k(\alpha_k) = \frac{(x^k(\alpha_k))^T \mu^k(\alpha_k)}{n} \\ &= \big(1-\alpha_k(1-\sigma_k)\big)\eta_k \leq \left(1 - \frac{2^{3/2}}{n}\,\gamma\,\frac{1-\gamma}{1+\gamma}\sigma_k(1-\sigma_k)\right)\eta_k. \end{aligned} \tag{2.20}$$

Die quadratische Funktion $\sigma \mapsto \sigma(1-\sigma)$ ist strikt konkav. Daher nimmt sie ihr Minimum in dem kompakten Intervall $[\underline{\sigma}, \overline{\sigma}] \subset (0,1)$ an einem der Endpunkte an. Also gilt

$$\sigma_k(1-\sigma_k) \geq \min\big\{\underline{\sigma}(1-\underline{\sigma}), \overline{\sigma}(1-\overline{\sigma})\big\} \quad \text{für alle } \sigma_k \in [\underline{\sigma}, \overline{\sigma}].$$

Setzen wir

$$\delta = 2^{3/2}\,\gamma\,\frac{1-\gamma}{1+\gamma}\,\min\big\{\underline{\sigma}(1-\underline{\sigma}), \overline{\sigma}(1-\overline{\sigma})\big\} > 0,$$

so folgt die Behauptung des Satzes aus (2.20). □

Nun sind wir in der Lage, das wesentliche Konvergenzresultat für Algorithmus 2.4 zu beweisen. Dieses besagt, dass der Algorithmus 2.4 nach $O(n|\log(\varepsilon)|)$ Iterationen dem Abbruchkriterium aus Schritt 2) genügt, wobei die in der O-Notation steckende Konstante von der Qualität des Startvektors abhängt.

SATZ 2.10. *Sei $\{(x^k, \lambda^k, \mu^k)\}_{k=0}^{\infty}$ eine durch Algorithmus 2.4 erzeugte Folge, wobei der Startvektor (x^0, λ^0, μ^0) der Bedingung*

$$\eta_0 \leq \frac{1}{\varepsilon^{\varrho}} \tag{2.21}$$

für eine positive Konstante $\varrho > 0$ genügt. Dann existiert ein $K \in \mathbb{N}$ mit $K = O(n|\log(\varepsilon)|)$ und

$$\eta_k \leq \varepsilon \quad \textit{für alle } k \geq K.$$

BEWEIS. Nach Satz 2.9 haben wir

$$\eta_{k+1} \leq \left(1 - \frac{\delta}{n}\right)\eta_k.$$

Also folgt

$$\log \eta_{k+1} \leq \log\left(1 - \frac{\delta}{n}\right) + \log \eta_k.$$

Wiederholte Anwendung ergibt mit (2.21)

$$\begin{aligned} \log \eta_k &\leq \log\left(1 - \frac{\delta}{n}\right) + \log \eta_{k-1} \leq 2\log\left(1 - \frac{\delta}{n}\right) + \log \eta_{k-2} \\ &\leq k \log\left(1 - \frac{\delta}{n}\right) + \log \eta_0 \leq k \log\left(1 - \frac{\delta}{n}\right) + \varrho \log \frac{1}{\varepsilon}. \end{aligned}$$

Wegen $1 + \beta \leq e^{\beta}$ gilt

$$\log(1+\beta) \leq \beta \quad \text{für alle } \beta > -1.$$

Daher erhalten wir

$$\log \eta_k \leq k\left(-\frac{\delta}{n}\right) + \varrho \log \frac{1}{\varepsilon}.$$

Es gilt also $\eta_k \leq \varepsilon$, sofern

$$k\left(-\frac{\delta}{n}\right) + \varrho \log \frac{1}{\varepsilon} \leq \log \varepsilon$$

erfüllt ist. Also bekommen wir die Bedingung

$$k \geq K = (1+\varrho)\frac{n}{\delta} \log \frac{1}{\varepsilon},$$

was zu zeigen war. □

Bemerkung 2.11. Wir betonen abschließend noch, dass die beiden Sätze 2.9 und 2.10 auch noch gelten, wenn wir α_k in Algorithmus 2.4 durch die in Lemma 2.8 gegebene explizite Schranke $\bar{\alpha}_k$ ersetzen. ◊

Nun kommen wir zu einem nicht-zulässigen Verfahren. Dazu haben wir bereits die Residuenvektoren

$$r_b(x) = Ax - b \in \mathbb{R}^m \quad \text{und} \quad r_c(\lambda, \mu) = A^T\lambda + \mu - c \in \mathbb{R}^n$$

eingeführt. Wir werden die Kurzschreibweise $r_b^k = r_b(x^k)$ sowie $r_c^k = r_c(\lambda^k, \mu^k)$ verwenden. In Algorithmus 2.4 galten stets $r_b^k = 0$ und $r_c^k = 0$ für alle $k \in \mathbb{N}$. Die Wahl eines Startwertes $(x^0, \lambda^0, \mu^0) \in \mathcal{U}_{-\infty}(\gamma)$ kann aber unter Umständen nicht so einfach sein. Im weiteren müssen die beiden Residuenvektoren r_b^k und r_c^k nicht mehr notwendig gleich null sein, so dass im Hinblick auf die Wahl des Startwertes mehr Freiheiten zugelassen sind. Allerdings muss die Menge $\mathcal{U}_{-\infty}(\gamma)$ modifiziert werden:

$$\mathcal{U}_{-\infty}(\gamma,\beta) = \Bigg\{ (x,\lambda,\mu) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n \mid x_i\,\mu_i \geq \gamma\eta \text{ für } 1 \leq i \leq n \text{ und } \\ \|(r_b(x), r_c(\lambda,\mu))\|_2 \leq \frac{\|(r_b^0, r_c^0)\|_2}{\eta_0}\,\beta\eta \Bigg\}$$

mit $\gamma \in (0,1)$, $\beta \geq 1$ und $\eta = x^T\mu/n$. Offenbar ist $\beta \geq 1$ notwendig dafür, dass auch der Startwert (x^0, λ^0, μ^0) in $\mathcal{U}_{-\infty}(\gamma,\beta)$ liegt. In $\mathcal{U}_{-\infty}(\gamma,\beta)$ haben wir die zusätzliche Forderung

$$\|(r_b(x), r_c(\lambda,\mu))\|_2 \leq \frac{\|(r_b^0, r_c^0)\|_2}{\eta_0}\,\beta\eta,$$

um die Verletztheit der beiden linearen Gleichungen $Ax = b$ und $A^T\lambda + \mu = c$ zu messen.

Gilt $\lim_{k\to\infty} \eta_k = 0$, so folgt auch für nicht-zulässiges (x^k, λ^k, μ^k)

$$\lim_{k\to\infty} r_b^k = 0 \quad \text{und} \quad \lim_{k\to\infty} r_c^k = 0.$$

Jeder Häufungspunkt der Folge $\{(x^k, \lambda^k, \mu^k)\}_{k=0}^{\infty}$ erfüllt daher die Optimalitätsbedingungen

$$\begin{aligned} A^T\lambda + \mu &= c, \\ Ax &= b, \\ x_i\mu_i &= 0, \quad i = 1,\ldots,n, \\ (x,\mu) &\geq 0. \end{aligned}$$

Algorithmus 2.12 (Nicht-zulässiges Pfad-Verfolgungs Verfahren).

1) *Wähle* $\gamma \in (0,1)$, $\beta \geq 1$, $0 < \underline{\sigma} < \overline{\sigma} \leq 1/2$, $\varepsilon \in (0,1)$, $w^0 = (x^0, \lambda^0, \mu^0)$ *mit* $(x^0, \mu^0) > 0$ *und* $x_i^0\mu_i^0 \geq \gamma\eta_0$ *für* $i = 1,\ldots,n$ *und setze* $k = 0$.

2) *Ist* $\eta_k = (x^k)^T\mu^k/n \leq \varepsilon$ *erfüllt, dann STOPP.*
3) *Wähle* $\sigma_k \in [\underline{\sigma}, \overline{\sigma}]$ *und bestimme eine Lösung*

$$\Delta w^k = (\Delta x^k, \Delta\lambda^k, \Delta\mu^k)$$

des linearen Gleichungs-Systems

$$\begin{pmatrix} 0 & A^T & I \\ A & 0 & 0 \\ M^k & 0 & X^k \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta\lambda \\ \Delta\mu \end{pmatrix} = \begin{pmatrix} -r_c^k \\ -r_b^k \\ -X^kM^ke + \sigma_k\eta_k e \end{pmatrix}. \tag{2.22}$$

Sei α_k *die größte Schrittweite* $\alpha \in (0,1]$ *mit*

$$w^k(\alpha) = (x^k + \alpha\Delta x^k, \lambda^k + \alpha\Delta\lambda^k, \mu^k + \alpha\Delta\mu^k) \in \mathcal{U}_{-\infty}(\gamma,\beta) \tag{2.23}$$

und

$$\eta_k(\alpha) \leq (1 - 0.01\alpha)\eta_k. \tag{2.24}$$

4) *Setze* $w^{k+1} = w^k(\alpha_k)$, $k = k+1$ *und gehe zurück zu Schritt* 2).

Bemerkung 2.13. 1) Wegen $\beta \geq 1$ folgt $(x^0, \lambda^0, \mu^0) \in \mathcal{U}_{-\infty}(\gamma,\beta)$. Daher liegt wegen (2.23) die gesamte Folge $\{w^k\}_{k=0}^{\infty}$ in $\mathcal{U}_{-\infty}(\gamma,\beta)$.
2) Die Bedingung (2.24) garantiert eine hinreichende Abnahme der gewichteten Dualitätslücke η_k.
3) Die schwer zu berechnende Schrittweite α_k kann wieder durch einen expliziten Ausdruck ersetzt werden. Für Details verweisen wir an dieser Stelle auf [**3**]. ◊

Wir zitieren hier den folgenden Satz aus [**3**, S. 159].

Satz 2.14. *Sei* $\{w^k\}_{k=0}^{\infty}$ *eine durch Algorithmus* 2.12 *erzeugte Folge. Dann gelten folgende Aussagen:*

1) *Die Folge* $\{\eta_k\}_{k=0}^{\infty}$ *konvergiert linear gegen Null.*
2) *Die Folge* $\{\|(r_b^k, r_c^k)\|\}_{k=0}^{\infty}$ *konvergiert r-linear gegen Null, das heißt, die Folge* $\{\|(r_b^k, r_c^k)\|\}_{k=0}^{\infty}$ *nicht-negativer Zahlen wird durch eine linear gegen Null konvergierende Folge* $\{c_k\}_{k=0}^{\infty}$ *majorisiert* ($c_k \geq 0$ *für alle* $k \in \mathbb{N}$, $\lim_{k\to\infty} c_k = 0$ *und* $\|(r_b^k, r_c^k)\|_2 \leq c_k$ *für alle* $k \in \mathbb{N}$).

4. Der Prädiktor-Korrektor Algorithmus von Mehrotra

Die meisten Innere-Punkte-Verfahren in Programm-Bibliotheken basieren auf einer von Mehrotra vorgeschlagenen Variante, die im wesentlichen zwei Gesichtspunkte hat:

1) Hinzufügen eines Korrektur-Schrittes bei der Berechnung der Suchrichtung;
2) adaptive Wahl des Parameters σ_k.

Motivieren kann man das Verfahren, in dem der zentrale Pfad $\mathcal{C}$ so verschoben wird, dass er an der aktuellen Iterierten (x^k, λ^k, μ^k) beginnt und weiterhin in der Menge der zulässigen Lösungen endet. Daher haben wir eine modifizierte Kurve

$$\mathcal{H} = \left\{(\hat{x}(s), \hat{\lambda}(s), \hat{\mu}(s)) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n : s \in [0,1)\right\}$$

mit $(\hat{x}(0), \hat{\lambda}(0), \hat{\mu}(0)) = (x^k, \lambda^k, \mu^k)$ und, sofern der Grenzwert existiert,

$$\lim_{s\nearrow 1}(\hat{x}(s), \hat{\lambda}(s), \hat{\mu}(s)) \in \mathcal{C}$$

Damit startet die Kurve $\mathcal{H}$ von einem Punkt, der nicht auf $\mathcal{C}$ liegt, endet aber in einem Punkt, der zur Menge $\mathcal{C}$ gehört.

Das Verfahren kombiniert drei Schritte zur Bestimmung der Suchrichtung:

1) *Prädiktor-Schritt*, der erlaubt, σ_k zu berechnen;
2) *Korrektur-Schritt*, der Information zweiter Ordnung von $\mathcal{H}$ (d.h., Information über die Krümmung), ausnutzt, um näher an die Lösung zu kommen;
3) *Zentrierender Schritt*, in dem σ_k in die dritte Blockgleichung in (2.22) eingesetzt wird.

Konkret lösen wir zunächst (2.22) mit der Wahl $\sigma_k = 0$, d.h.,

$$\begin{pmatrix} 0 & A^T & I \\ A & 0 & 0 \\ M^k & 0 & X^k \end{pmatrix} \begin{pmatrix} \Delta x^{\text{aff}} \\ \Delta\lambda^{\text{aff}} \\ \Delta\mu^{\text{aff}} \end{pmatrix} = \begin{pmatrix} -r_c^k \\ -r_b^k \\ -X^k M^k e \end{pmatrix}. \tag{2.25}$$

Die Richtung $(\Delta x^{\text{aff}}, \Delta\lambda^{\text{aff}}, \Delta\mu^{\text{aff}})$ wird im Englischen *affine scaling direction* genannt, daher die Bezeichnung mit dem Index aff.

Dann bestimmen wir die größtmögliche Schrittweiten $\alpha^{\text{aff}}_{\text{prim}}$, $\alpha^{\text{aff}}_{\text{dual}} \in (0, 1]$, so dass

$$x^k(\alpha^{\text{aff}}_{\text{prim}}) = x^k + \alpha^{\text{aff}}_{\text{prim}} \Delta x^{\text{aff}} \geq 0, \quad \mu^k(\alpha^{\text{aff}}_{\text{dual}}) = \mu^k + \alpha^{\text{aff}}_{\text{dual}} \Delta\mu^{\text{aff}} \geq 0.$$

Wir haben explizite Formeln für die Schrittweiten zur Verfügung:

$$\alpha^{\text{aff}}_{\text{prim}} = \min\left\{1, \min_{i\,:\,\Delta x_i^{\text{aff}} < 0} \frac{-x_i^k}{\Delta x_i^{\text{aff}}}\right\}, \tag{2.26a}$$

$$\alpha^{\text{aff}}_{\text{dual}} = \min\left\{1, \min_{i\,:\,\Delta \mu_i^{\text{aff}} < 0} \frac{-\mu_i^k}{\Delta \mu_i^{\text{aff}}}\right\}. \tag{2.26b}$$

Dann folgen offenbar

$$x_i^k + \alpha^{\text{aff}}_{\text{prim}} \Delta x_i^{\text{aff}} \geq x_i^k - \frac{x_i^k}{\Delta x_i^{\text{aff}}} \Delta x_i^{\text{aff}} = 0,$$

$$\mu_i^k + \alpha^{\text{aff}}_{\text{dual}} \Delta \mu_i^{\text{aff}} \geq \mu_i^k - \frac{\mu_i^k}{\Delta \mu_i^{\text{aff}}} \Delta \mu_i^{\text{aff}} = 0.$$

Mit den berechneten Schrittweiten berechnen wir

$$\eta^{\text{aff}} = \frac{1}{n} \left(x^k(\alpha^{\text{aff}}_{\text{prim}})\right)^T \mu^k(\alpha^{\text{aff}}_{\text{dual}}) \tag{2.27}$$

und mit $\eta_k = (x^k)^T \mu^k / n$ setzen wir

$$\sigma = \left(\frac{\eta^{\text{aff}}}{\eta_k}\right)^3.$$

Ist nun $\eta^{\text{aff}} \ll \eta_k$, so ist σ klein (und umgekehrt).

Im Korrektur-Schritt wählen wir auf der rechten Seite von (2.25) den Vektor $(0, 0, -\Delta X^{\text{aff}} \Delta M^{\text{aff}} e)^T$ und im letzten Schritt $(0, 0, \sigma\eta_k e)^T$. Insgesamt haben wir dann das System

$$\begin{pmatrix} 0 & A^T & I \\ A & 0 & 0 \\ M^k & 0 & X^k \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta\lambda \\ \Delta\mu \end{pmatrix} = \begin{pmatrix} -r_c^k \\ -r_b^k \\ -X^k M^k e - \Delta X^{\text{aff}} \Delta M^{\text{aff}} e + \sigma\eta_k e \end{pmatrix} \tag{2.28}$$

und setzen dann

$$\alpha_k^{\text{prim}} = \min\left\{1, \min_{i\,:\,\Delta x_i<0} \frac{-x_i^k}{\Delta x_i}\right\}, \tag{2.29a}$$

$$\alpha_k^{\text{dual}} = \min\left\{1, \min_{i\,:\,\Delta \mu_i<0} \frac{-\mu_i^k}{\Delta \mu_i}\right\}. \tag{2.29b}$$

BEMERKUNG 2.15. Um den Korrekturschritt zu motivieren, betrachten wir $(x_i^k + \Delta x_i^{\text{aff}})(\mu_i^k + \Delta\mu_i^{\text{aff}}) = x_i^k\mu_i^k + x_i^k\Delta\mu_i^{\text{aff}} + \Delta x_i^{\text{aff}}\mu_i^k + \Delta x_i^{\text{aff}}\Delta\mu_i^{\text{aff}} = \Delta x_i^{\text{aff}}\Delta\mu_i^{\text{aff}}$, wobei wir die dritte Blockzeile von (2.25) verwendet haben. Wird also ein voller Schritt gewählt, d.h., $\alpha_{\text{prim}}^{\text{aff}} = \alpha_{\text{dual}}^{\text{aff}} = 1$, so geht das Produkt $x_i^k\mu_i^k$ im Prädiktor-Schritt statt auf den Wert 0 über in das Produkt $\Delta x_i^{\text{aff}}\Delta\mu_i^{\text{aff}}$, $i = 1, \ldots, n$. Der Korrektur-Schritt versucht dieses zu kompensieren, so dass das Produkt der Komponenten $x^k + \Delta x$ mit $\mu^k + \Delta\mu$ näher an Null ist. ◊

ALGORITHMUS 2.16 (Mehrotra Prädiktor-Korrektur Verfahren).

1) *Wähle* $(x^0, \lambda^0, \mu^0) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n$ *mit* $(x^0, \mu^0) > 0$, $\varepsilon \in (0, 1)$ *und setze* $k = 0$.
2) *Ist* $\eta_k = (x^k)^T\mu^k/n \leq \varepsilon$ *erfüllt, dann STOPP.*
3) *Löse* (2.25) *für* $(\Delta x^{\text{aff}}, \Delta\lambda^{\text{aff}}, \Delta\mu^{\text{aff}})$.
4) *Berechne die Schrittweiten* $\alpha_{\text{prim}}^{\text{aff}}$, $\alpha_{\text{dual}}^{\text{aff}}$, η^{aff} *gemäß* (2.26) *sowie* (2.27) *und setze* $\sigma = (\eta^{\text{aff}}/\eta_k)^3$.
5) *Löse* (2.28) *für* $(\Delta x, \Delta\lambda, \Delta\mu)$.
6) *Berechne die Schrittweiten* α_k^{prim} *und* α_k^{dual} *mit* (2.29).
7) *Setze* $x^{k+1} = x^k + \alpha_k^{\text{prim}}\Delta x$, $(\lambda^{k+1}, \mu^{k+1}) = (\lambda^k, \mu^k) + \alpha_k^{\text{dual}}(\Delta\lambda, \Delta\mu)$, $k = k + 1$, *und gehe zurück zu Schritt* 2).

BEMERKUNG 2.17. Für Algorithmus 2.16 ist keine Konvergenzanalysis vorhanden. Es gibt auch Beispiele, in denen Algorithmus 2.16 divergiert, was allerdings durch kleine Modifikationen verhindert werden kann. In vielen Anwendungen ist aber das Konvergenzverhalten des Prädiktor-Korrektor Verfahrens sehr gut. ◊

KAPITEL 3

Quadratische Programmierung

In diesem Kapitel beschäftigen wir uns mit der Quadratischen Programmierung (im Englischen *quadratic programming*), wo die Zielfunktion quadratisch ist und lineare Nebenbedingungen vorliegen. Diese Problemklasse ist insbesondere auch deshalb von großer Bedeutung, da sie uns im Kapitel 4 als Teilproblem von einem iterativem Verfahren, dem SQP-Verfahren, wieder beschäftigen wird.

Das Standard-Problem in diesem Abschnitt lautet wie folgt

$$(\mathbf{QP}) \quad \min q(x) = \frac{1}{2}x^T Q x + x^T d \quad \text{u.d.N.} \quad \begin{cases} a_i^T x = b_i, & i = 1, \dots, m, \\ a_i^T x \geq b_i, & i = m+1, \dots, m+p. \end{cases}$$

Wir setzen voraus, dass $Q \in \mathbb{R}^{n \times n}$ symmetrisch und positiv semi-definit ist. Ferner gelte $a_i \in \mathbb{R}^n$ für $i = 1, \dots, m+p$. Dann ist $(\mathbf{QP})$ ein konvexes Optimierungsproblem, da die Nebenbedingungen eine konvexe Menge beschreiben und die Zielfunktion konvex ist.

1. Gleichungsrestringierte Probleme

In diesem Abschnitt beschränken wir uns auf Probleme ohne Ungleichungen. Daher betrachten wir

$$(\mathbf{QP}_{Gl}) \qquad \min q(x) = \frac{1}{2}x^T Q x + x^T d \quad \text{u.d.N.} \quad Ax = b,$$

wobei $A \in \mathbb{R}^{m \times n}$ gegeben ist durch

$$A = \begin{pmatrix} a_1^T \\ \vdots \\ a_m^T \end{pmatrix}$$

mit $\operatorname{Rang} A = m$, d.h., A hat vollen Rang. Ferner gelte $m \leq n$.

Um die notwendigen Bedingungen erster Ordnung (siehe Satz 1.9) aufzustellen, führen wir die affin-lineare Abbildung $e : \mathbb{R}^n \to \mathbb{R}^m$ durch $e(x) = Ax - b$ für $x \in \mathbb{R}^n$ ein. Wegen $\nabla e(x) = A$ folgt, dass die Jacobi-Matrix von e vollen Rang besitzt für alle $x \in \mathbb{R}^n$. Damit sind alle Punkte in $\mathbb{R}^n$ reguläre Punkte bezüglich der Nebenbedingung $e(x) = 0$. Ist also $x^* \in \mathbb{R}^n$ eine lokale Lösung von $(\mathbf{QP}_{Gl})$, so existiert ein Lagrange-Multiplikator $\lambda^* \in \mathbb{R}^m$ mit

$$\nabla q(x^*) + (\lambda^*)^T \nabla e(x^*) = 0.$$

Speziell für $(\mathbf{QP}_{Gl})$ bekommen wir daher die notwendige Bedingung:

$$Qx^* + d + A^T \lambda^* = 0.$$

Mit der Gleichungs-Nebenbedingung erhalten wir das lineare System

$$\begin{pmatrix} Q & A^T \\ A & 0 \end{pmatrix} \begin{pmatrix} x^* \\ \lambda^* \end{pmatrix} = \begin{pmatrix} -d \\ b \end{pmatrix}. \tag{3.1}$$

Da (3.1) die notwendigen Bedingungen erster Ordung darstellen, wird die Koeffizienten-Matrix in (3.1) auch *KKT-Matrix* bezeichnet, wobei die Abkürzung KKT für *Karush-Kuhn-Tucker* steht (vergleiche Satz 1.18).

Mit $x^* = x + \Delta x$, $e(x) = Ax - b$ und $\nabla q(x) = (Qx + d)^T$ lässt sich (3.1) auch in der Form

$$\begin{pmatrix} Q & A^T \\ A & 0 \end{pmatrix} \begin{pmatrix} \Delta x \\ \lambda^* \end{pmatrix} = - \begin{pmatrix} \nabla q(x)^T \\ e(x) \end{pmatrix}. \tag{3.2}$$

Wir erhalten bei der Wahl eines beliebigen $x \in \mathbb{R}^n$ die Lösung (x^*, λ^*) von (3.1), indem wir (3.2) lösen und dann $x^* = x + \Delta x$ setzen.

Um hinreichende Bedingungen für die Invertierbarkeit der KKT-Matrix anzugeben, führen wir eine Matrix $Z \in \mathbb{R}^{n \times (n-m)}$ ein, deren Spalten eine Basis für $\operatorname{Kern} \nabla e(x)$ bilden. Damit gilt

$$AZ = 0 \in \mathbb{R}^{m \times (n-m)}. \tag{3.3}$$

LEMMA 3.1. *Die Matrix A habe vollen Rang m, und Z^TQZ sei positiv definit. Dann ist die KKT-Matrix*

$$K = \begin{pmatrix} Q & A^T \\ A & 0 \end{pmatrix}$$

invertierbar. Insbesondere existiert ein eindeutiges Paar (x^*, λ^*), welches (3.1) *löst.*

BEWEIS. Der Beweis folgt bereits aus Lemma 1.15. Wir wollen ihn aber hier noch einmal mit etwas anderen Argumenten durchführen. Seien $(x, \lambda) \in \mathbb{R}^n \times \mathbb{R}^m$ beliebig gewählt mit

$$\begin{pmatrix} Q & A^T \\ A & 0 \end{pmatrix} \begin{pmatrix} x \\ \lambda \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \tag{3.4}$$

Aus der zweiten Blockzeile in (3.4) erhalten wir $Ax = 0$, das heißt, $x \in \operatorname{Kern} A$. Damit existiert ein $w \in \mathbb{R}^{n-m}$, so dass wir x in der Form $x = Zw$ schreiben können. Aus $Ax = 0$ folgt $x^TA^T = 0$. Daher führt (3.4) auf die skalare Gleichung

$$\begin{aligned} 0 &= \begin{pmatrix} x \\ \lambda \end{pmatrix}^T \begin{pmatrix} Q & A^T \\ A & 0 \end{pmatrix} \begin{pmatrix} x \\ \lambda \end{pmatrix} = \begin{pmatrix} x \\ \lambda \end{pmatrix}^T \begin{pmatrix} Qx + A^T\lambda \\ 0 \end{pmatrix} = x^TQx \\ &= w^TZ^TQZw. \end{aligned}$$

Nach Voraussetzung ist Z^TQZ positiv definit, also muss $w = 0$ gelten. Damit ist auch $x = Zw = 0$. Es bleibt also nur noch zu zeigen, dass auch $\lambda = 0$ gilt. Aus der ersten Blockzeile in (3.4) ergibt sich mit $x = 0$ die Gleichung $A^T\lambda = 0$. Da A vollen Rang besitzt, ist A surjektiv. Deshalb ist die Matrix A^T injektiv. Das bedeutet aber $\lambda = 0$. □

BEMERKUNG 3.2. Die Matrix Z^TQZ wird *reduzierte Hesse-Matrix* genannt. Wir werden später noch auf diese Matrix zurückkommen. ◊

BEISPIEL 3.3. Wir betrachten das Problem

$$\min q(x) \quad \text{u.d.N.} \quad x_1 + x_3 = 3,\ x_2 + x_3 = 0$$

mit $x = (x_1, x_2, x_3) \in \mathbb{R}^3$ und

$$q(x) = 3x_1^2 + 2x_1x_2 + x_1x_3 + \frac{5}{2}\,x_2^2 + 2x_2x_3 + 2x_3^2 - 8x_1 - 3x_2 - 3x_3.$$

Zuerst schreiben wir das Problem in der Form $(\mathbf{QP}_{Gl})$. Wir setzen daher $n = 3$, $m = 2$ und

$$Q = \begin{pmatrix} 6 & 2 & 1 \\ 2 & 5 & 2 \\ 1 & 2 & 4 \end{pmatrix},\ d = \begin{pmatrix} -8 \\ -3 \\ -3 \end{pmatrix},\ A = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix},\ b = \begin{pmatrix} 3 \\ 0 \end{pmatrix}.$$

Eine Basis von Kern A ist gegeben durch $Z = (-1, -1, 1)^T \in \mathbb{R}^{3\times 1}$. Dann folgt $Z^TQZ = 13 > 0$. Nach Lemma 3.1 existiert genau eine Lösung (x^*, λ^*) von (3.1), und zwar

$$x^* = \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} \quad \text{und} \quad \lambda^* = \begin{pmatrix} 3 \\ -2 \end{pmatrix}.$$

In diesem Beispiel ist Q selbst positiv definit. ◊

Ist (x^*, λ^*) eine Lösung von (3.1) unter den Voraussetzungen von Lemma 3.1, so gelten auch die hinreichenden Bedingungen zweiter Ordnung, das heißt, x^* ist ein striktes lokales Minimum von $(\mathbf{QP}_{Gl})$. Wir können diese Tatsache aber auch auf einem anderen, direktem Weg beweisen.

Satz 3.4. *Es seien die Voraussetzungen von Lemma* 3.1 *erfüllt. Dann ist die eindeutige Lösung x^* von* (3.1) *auch eine eindeutige globale Lösung von* $(\mathbf{QP}_{Gl})$.

Beweis. Sei $x \in \mathbb{R}^n$ ein zulässiger Punkt, das heißt, es gilt $Ax = b$. Ferner sei $\Delta x = x^* - x$. Dann folgen $Ax^* = Ax = b$ und daher $A\Delta x = 0$. Somit liegt Δx im Kern von A. Ferner erhalten wir

$$\begin{aligned} q(x) &= \frac{1}{2}\,(x^* - \Delta x)^T Q(x^* - \Delta x) + d^T(x^* - \Delta x) \\ &= \frac{1}{2}\,\Delta x^T Q \Delta x - \Delta x^T Q x^* - d^T \Delta x + q(x^*). \end{aligned} \tag{3.5}$$

Aus (3.1) schließen wir $Qx^* = -d - A^T\lambda^*$. Also bekommen wir mit $\Delta x \in \operatorname{Kern} A$

$$\Delta x^T Q x^* = \Delta x^T\big(-d - A^T\lambda^*\big) = -\Delta x^T d.$$

Einsetzen in (3.5) liefert

$$q(x) = \frac{1}{2}\,\Delta x^T Q \Delta x + q(x^*).$$

Wegen $\Delta x \in \operatorname{Kern} A$ ergibt sich die Darstellung $\Delta x = Zu$ für ein $u \in \mathbb{R}^{n-m}$, wobei die Spalten der Matrix $Z \in \mathbb{R}^{n\times(n-m)}$ eine Basis des Kerns von A bilden. Damit lässt sich q an der Stelle x schreiben als

$$q(x) = \frac{1}{2}\,u^T Z^T Q Z u + q(x^*).$$

Da Z^TQZ positiv definit ist, gilt $q(x) > q(x^*)$ für alle $u \in \mathbb{R}^{n-m} \setminus \{0\}$ und daher für alle $x \in \mathbb{R}^n \setminus \{x^*\}$ mit $Ax = b$. Das bedeutet, dass x^* das eindeutige globale Minimum von $(\mathbf{QP}_{Gl})$ bezeichnet. □

BEMERKUNG 3.5. Wenn die reduzierte Hesse-Matrix nicht-positive Eigenwerte hat, so besitzt $(\mathbf{QP}_{Gl})$ keine beschränkte Lösung, ausgenommen in einem Spezialfall. Angenommen, das Paar (x^*, λ^*) löst (3.1). Sei $u \in \mathbb{R}^{n-m}$ ein Vektor mit $u^T Z^T Q Z u \leq 0$. Wir setzen $\Delta x = Zu$. Dann folgt für alle $\alpha > 0$

$$A(x^* + \alpha \Delta x) = b,$$

so dass $x^* + \alpha \Delta x$ für alle $\alpha > 0$ zulässig ist, aber

$$q(x^* + \alpha \Delta x) = q(x^*) + \alpha \Delta x^T \big(Qx^* + d\big) + \frac{\alpha^2}{2} \Delta x^T Q \Delta x = q(x^*) + \frac{\alpha^2}{2} \Delta x^T Q \Delta x$$

gilt, wobei wir die Beziehungen $Qx^* + d = -A^T \lambda^*$ und $\Delta x^T A^T \lambda^* = u^T Z^T A^T \lambda^* = 0$ genutzt haben. Damit können wir zu jedem x^*, das die KKT-Bedingungen (3.1) erfüllt, eine Richtung Δx finden, in die q nicht wächst. Es existiert sogar im Fall, wenn $Z^T Q Z$ mindestens einen negativen Eigenwert besitzt, eine Richtung, in die q sogar streng monoton fallend ist. Der einzige Fall, in dem $(\mathbf{QP}_{Gl})$ eine Lösung besitzt, tritt ein, wenn $Z^T Q Z$ positiv semidefinit ist. Aber dann ist x^* auch kein striktes lokales Minimum. ◊

2. Lösung des KKT-Systems

Zunächst wollen wir bemerken, dass im Fall $m \geq 1$ die KKT-Matrix stets indefinit ist. Es gilt sogar das folgende Resultat (ohne Beweis):

LEMMA 3.6. *Die Matrix A habe vollen Rang m, und $Z^T Q Z$ sei positiv definit. Dann hat die nach Lemma* 3.1 *reguläre KKT-Matrix genau n positive und m negative Eigenwerte.*

Wir wollen hier zwei Methoden zum Lösen des KKT-Systems besprechen, die im Englischen mit *range space method* und *null space method* bezeichnet werden.

Range space method. Ist Q symmetrisch und positiv definit, so können wir folgende Blockelimination beim KKT-System durchführen: Wir multiplizieren die erste Zeile von (3.2) mit AQ^{-1} von links und erhalten

$$AQ^{-1} Q \Delta x + AQ^{-1} A^T \lambda^* = -AQ^{-1} \nabla q(x)^T.$$

Subtraktion der zweiten Zeile von (3.2) führt auf

$$AQ^{-1} A^T \lambda^* = -AQ^{-1}(Qx + d) + e(x) = e(x) - Ax - AQ^{-1} d. \tag{3.6}$$

Offenbar ist die Matrix $AQ^{-1}A^T \in \mathbb{R}^{m \times m}$ symmetrisch und positiv definit, da wir vorausgesetzt haben, dass Q symmetrisch und positiv definit ist. Damit können wir das lineare Gleichungs-System (3.6) mit dem CG-Verfahren oder mit Hilfe der Cholesky-Zerlegung lösen. Ist λ^* berechnet, so bekommen wir für Δx das System

$$Q \Delta x = -(Qx + d) - A^T \lambda^*$$

Auch hier können wir das CG-Vefahren oder die Cholesky-Zerlegung zur Bestimmung von Δx verwenden.

Erforderlich ist bei der Anwendung der Range-Space-Methode die Realisierung von Q^{-1}. Daher wird dieses Verfahren zur Lösung des KKT-Systems angewendet, wenn

- Q gut konditioniert ist,
- Q^{-1} ohne viel Aufwand zu invertieren ist, explizit bekannt ist oder durch Quasi-Newton Updates approximiert wird,
- im Fall von $m \ll n$.

Null space method. Für diese Strategie ist $\det Q \neq 0$ nicht erforderlich, so dass dieses Verfahren im allgemeinen öfter angewendet werden kann. Vorausgesetzt werden die Annahmen von Lemma 3.1: Rang $A = m$ und $Z^T QZ$ ist positiv definit, wobei $Z \in \mathbb{R}^{n\times(n-m)}$ eine Matrix ist, deren Spalten eine Basis des Nullraums von A bilden. Wir schreiben Δx in

$$\begin{pmatrix} Q & A^T \\ A & 0 \end{pmatrix} \begin{pmatrix} \Delta x \\ \lambda^* \end{pmatrix} = - \begin{pmatrix} Qx + d \\ Ax - b \end{pmatrix} \tag{3.7}$$

in der Form

$$\Delta x = Y \Delta x_Y + Z \Delta x_Z, \tag{3.8}$$

wobei wir die Matrix $Z = [z_1 \,|\, \ldots \,|\, z_{n-m}] \in \mathbb{R}^{n\times(n-m)}$ bereits eingeführt haben und $Y = [y_1 \,|\, \ldots \,|\, y_m] \in \mathbb{R}^{n\times m}$ derart gewählt ist, dass die zusammengesetzte Matrix $[Y|Z] \in \mathbb{R}^{n\times n}$ regulär ist. Ferner gelten $\Delta x_Y \in \mathbb{R}^m$ und $\Delta x_Z \in \mathbb{R}^{n-m}$. Offenbar bilden die Spalten von Y eine Basis von Bild A^T. Weiter bekommen wir $A[Y|Z] = [AY|0]$, $AY \in \mathbb{R}^{m\times m}$ und Rang $(AY) = \text{Span}\,\{y_1, \ldots, y_m\} = m$. Offenbar folgt daher aus $Av = 0$ die Eigenschaft $v \in \text{Span}\,\{y_1, \ldots, y_m\} \cap \text{Span}\,\{z_1, \ldots, z_{n-m}\} = \{0\}$, das heisst, die Matrix AY ist invertierbar. Wir schließen aus der zweiten Blockzeile in (3.7)

$$(AY)\Delta x_Y = -(Ax - b). \tag{3.9}$$

Das System (3.9) besitzt genau eine Lösung $\Delta x_Y \in \mathbb{R}^m$. Wir setzen nun die Zerlegung (3.8) in die erste Blockzeile von (3.7) ein:

$$QY\Delta x_Y + QZ\Delta x_Z + A^T\lambda^* = -Qx - d.$$

Multiplikation mit Z^T von links führt wegen $AZ = 0 \in \mathbb{R}^{m\times(n-m)}$ auf

$$\begin{aligned} (Z^TQZ)\Delta x_Z &= -Z^TQY\Delta x_Y - Z^TA^T\lambda^* - Z^T(Qx + d) \\ &= -Z^T\big(QY\Delta x_Y + Qx + d\big). \end{aligned} \tag{3.10}$$

Unter den Voraussetzungen von Lemma 3.1 ist die Matrix Z^TQZ positiv definit. Daher können wir zur Berechnung von Δx_Z aus (3.10) das CG-Verfahren oder die Cholesky-Faktorisierung verwendet werden. Damit ist Δx aus (3.8) mittels (3.9) und (3.10) berechenbar. Multiplizieren wir die erste Blockzeile in (3.7) mit Y^T von links, so erhalten wir das lineare System

$$(AY)^T\lambda^* = -Y^T(Qx + d + Q\Delta x). \tag{3.11}$$

Wegen $\det(AY) \neq 0$ ist λ^* durch (3.11) eindeutig bestimmt.

Beispiel 3.7. Wir betrachten das Problem von Beispiel 3.3. Als Matrix Z wählen wir $Z = (-1, -1, 1)^T$. Damit können wir die Matrix

$$Y = \frac{1}{3}\begin{pmatrix} 2 & -1 \\ -1 & 2 \\ 1 & 1 \end{pmatrix}$$

wählen. Es folgt

$$AY = \frac{1}{3}\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}\begin{pmatrix} 2 & -1 \\ -1 & 2 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

Wir wählen $x = 0$ in (3.7). Damit folgen

$$e(x) = Ax - b = -b = \begin{pmatrix} -3 \\ 0 \end{pmatrix} \quad \text{und} \quad \nabla q(x)^T = Qx + d = d = \begin{pmatrix} -8 \\ -3 \\ -3 \end{pmatrix}.$$

Aus (3.9) bekommen wir $\Delta x_Y = b = (3, 0)^T$. Wir berechnen die reche Seite von (3.10):

$$\begin{aligned} &-Z^T QY \begin{pmatrix} 3 \\ 0 \end{pmatrix} - Z^T \begin{pmatrix} -8 \\ -3 \\ -3 \end{pmatrix} \\ &\quad = \left(\frac{1}{3}, \frac{1}{3}, -\frac{1}{3}\right) \begin{pmatrix} 6 & 2 & 1 \\ 2 & 5 & 2 \\ 1 & 2 & 4 \end{pmatrix} \begin{pmatrix} 2 & -1 \\ -1 & 2 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 3 \\ 0 \end{pmatrix} - (1, 1, -1) \begin{pmatrix} -8 \\ -3 \\ -3 \end{pmatrix} \\ &\quad = \left(\frac{7}{3}, \frac{5}{3}, -\frac{1}{3}\right) \begin{pmatrix} 2 & -1 \\ -1 & 2 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 3 \\ 0 \end{pmatrix} + 8 = (7, 5, -1) \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} = 0. \end{aligned}$$

Damit ist die Lösung von (3.10) durch $\Delta x_Z = 0$ gegeben. Also folgt

$$\Delta x = Y \Delta x_Y + Z \Delta x_Z = Y \begin{pmatrix} 3 \\ 0 \end{pmatrix} + \begin{pmatrix} -1 \\ -1 \\ 1 \end{pmatrix} 0 = \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix}.$$

Nun zur Berechnung von λ^* gemäß (3.11): Wegen $AY = I$ folgt

$$\lambda^* = -Y^T \begin{pmatrix} -8 \\ -3 \\ -3 \end{pmatrix} - Y^T Q \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} = \begin{pmatrix} -3 \\ 2 \end{pmatrix}.$$

Wegen $x^* = x + \Delta x$ bekommen wir wegen $x = 0$ die Lösung

$$x^* = \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix} \quad \text{und} \quad \lambda^* = \begin{pmatrix} -3 \\ 2 \end{pmatrix}.$$

als Lösung von (3.7). ◊

Ist $n - m$ klein, so ist die Null-Space-Methode oft sehr effizient. Allerdings ist die Berechnung von Z notwendig. Die Matrix Z ist nicht eindeutig bestimmt und die Matrix $Z^T QZ$ eine schlecht konditionierte Matrix. Sind allerdings die Spalten von Z orthonormal, so folgt für die Konditionszahl $\kappa_2(Z^T QZ) = \kappa_2(Q)$.

3. Ungleichungsrestringierte Probleme

Wir wollen die Optimalitätsbedingungen für das Problem (**QP**) formulieren. Dazu setzen wir

$$e(x) = Ax - b \quad \text{und} \quad g(x) = r - Cx,$$

wobei

$$A = \begin{pmatrix} a_1^T \\ \vdots \\ a_m^T \end{pmatrix} \in \mathbb{R}^{m \times n}, \; C = \begin{pmatrix} a_{m+1}^T \\ \vdots \\ a_{m+p}^T \end{pmatrix} \in \mathbb{R}^{p \times n}$$

und

$$b = \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix} \in \mathbb{R}^m, \; r = \begin{pmatrix} b_{m+1} \\ \vdots \\ b_{m+p} \end{pmatrix} \in \mathbb{R}^p$$

gelten. Die Lagrange-Funktion ist

$$L(x, \lambda, \mu) = \frac{1}{2} x^T Q x + x^T d + \langle Ax - b, \lambda \rangle_{\mathbb{R}^m} + \langle r - Cx, \mu \rangle_{\mathbb{R}^p}.$$

Die KKT-Bedingungen lauten daher

$$Qx^* + d + A^T \lambda^* - C^T \mu^* = 0 \qquad \text{in } \mathbb{R}^n, \tag{3.12a}$$

$$Ax^* = b \qquad \text{in } \mathbb{R}^m, \tag{3.12b}$$

$$Cx^* \geq r \qquad \text{in } \mathbb{R}^p, \tag{3.12c}$$

$$\mu^* \geq 0 \qquad \text{in } \mathbb{R}^p, \tag{3.12d}$$

$$(\mu^*)^T (r - Cx^*) = 0. \tag{3.12e}$$

Für konvexe quadratische Optimierungsprobleme, wenn also Q positive semidefinite ist, sind die notwendigen Optimalitätsbedingungen bereits hinreichend dafür, dass x^* eine globale Lösung von (**QP**) ist.

Satz 3.8. *Wenn x^* die Bedingungen* (3.12) *erfüllt zusammen mit einem $\lambda^* \in \mathbb{R}^m$ und einem $\mu^* \in \mathbb{R}^p$ mit $\mu^* \geq 0$ und wenn Q positiv semidefinit auf* $\operatorname{Ker} \nabla e(x^*)$ *ist, dann ist x^* eine globale Lösung von* (**QP**).

Beweis. Sei x ein zulässiger Punkt für (**QP**). Dann gelten $Ax = b$ in $\mathbb{R}^m$ und $Cx \geq r$ in $\mathbb{R}^p$. Wir setzen $\Delta x = x - x^*$. Dann folgen $A\Delta x = 0$ und

$$\big(C\Delta x\big)_i = \big(Cx - Cx^*\big)_i \geq r_i - \big(Cx^*\big)_i = 0 \quad \text{für alle } i \in \mathcal{A}(x^*),$$

wobei $\mathcal{A}(x^*) \subset \{1, \ldots, p\}$ die Menge der an x^* aktiven Indizes bezeichnet. Es gilt weiter $\mu_i^* = 0$ für alle $i \in \mathcal{I}(x^*) = \{1, \ldots, p\} \setminus \mathcal{A}(x^*)$. Zusammen mit (3.12a) und (3.12d) erhalten wir

$$\begin{aligned} \Delta x^T (Qx^* + d) &= -\Delta x^T A^T \lambda^* + \Delta x^T C^T \mu^* = \Delta x^T C^T \mu^* \\ &= \sum_{i \in \mathcal{A}(x^*)} \big(C\Delta x\big)_i \mu_i^* + \sum_{i \in \mathcal{I}(x^*)} \big(C\Delta x\big)_i \mu_i^* \geq 0. \end{aligned} \tag{3.13}$$

Da Q positiv semidefinit auf $\operatorname{Ker} \nabla e(x^*)$ ist, schließen wir aus (3.13), dass

$$q(x) = q(x^*) + \Delta x^T (Qx^* + d) + \frac{1}{2} \Delta x^T Q \Delta x \geq q(x^*) + \frac{1}{2} \Delta x^T Q \Delta x \geq q(x^*),$$

so dass x^* eine globale Lösung von (**QP**) ist. □

Bemerkung 3.9.

1) Eine kleine Modifikation des Beweises von Satz 3.8 zeigt, dass x^* die eindeutige, globale Lösung von (**QP**) ist, wenn Q positiv definit auf $\operatorname{Ker} \nabla e(x^*)$ ist.
2) Wenn Q nicht positiv definit auf $\operatorname{Ker} \nabla e(x^*)$ ist, dann kann (**QP**) mehrere Lösungen besitzen.

Verfahren zum Lösen der KKT-Bedingungen sind zum Beispiel *aktive Mengenstrategien*, *projezierte Gradienten-Verfahren* oder *Innere Punkte Methoden*.

4. Innere-Punkte Verfahren für Quadratische Programmierung

Wir betrachten

$$\min q(x) = \frac{1}{2}\, x^T Q x + x^T d \quad \text{u.d.N.} \quad Ax \geq b \tag{3.14}$$

wobei Q symmetrisch und positiv semi-definit ist und $d \in \mathbb{R}^n$, $A \in \mathbb{R}^{m\times n}$ sowie $b \in \mathbb{R}^m$ gelten.

Notwendige Optimalitäts-Bedingungen erster Ordnung: Sei x^* eine Lösung von (3.14). Dann löst (x^*, μ^*), μ^* der assozierte Lagrange-Multiplikator, das System

$$\begin{aligned} Qx - A^T\mu + d &= 0 && \text{in } \mathbb{R}^n,\\ b - Ax &\leq 0 && \text{in } \mathbb{R}^m,\\ (b - Ax)_i\mu_i &= 0 && \text{für } i = 1,\dots,m,\\ \mu &\geq 0 && \text{in } \mathbb{R}^m. \end{aligned}$$

Mit Einführung der *Slack-Variablen* $y = Ax - b \in \mathbb{R}^m$ folgt

$$Qx - A^T\mu + d = 0 \qquad \text{in } \mathbb{R}^n, \tag{3.15a}$$

$$b - Ax + y = 0 \qquad \text{in } \mathbb{R}^m, \tag{3.15b}$$

$$y_i\mu_i = 0 \qquad \text{für } i = 1,\dots,m, \tag{3.15c}$$

$$y \geq 0 \qquad \text{in } \mathbb{R}^m, \tag{3.15d}$$

$$\mu \geq 0 \qquad \text{in } \mathbb{R}^m. \tag{3.15e}$$

Das System (3.15) ist auch hinreichend, da die Zielfunktion und die Menge der zulässigen Punkte konvex sind. Wie in Abschnitt 2 schreiben wir (3.15) in der folgenden Form

$$F(x,y,\mu) = \begin{pmatrix} Qx - A^T\mu + d \\ b - Ax + y \\ YMe \end{pmatrix} = 0 \quad \text{und} \quad (y,\mu) \geq 0,$$

wobei

$$Y = \operatorname{diag}(y_1,\dots,y_m),\ M = \operatorname{diag}(\mu_1,\dots,\mu_m) \quad \text{und} \quad e = (1,\dots,1)^T \in \mathbb{R}^m$$

Sei $(x,y,\mu) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^m$ eine aktuelle Iterierte. Dann ist die gewichtete Dualitätslücke η definiert durch

$$\eta = \frac{1}{m}\sum_{i=1}^{m} y_i\mu_i = \frac{y^T\mu}{m}.$$

Der zentrale Pfad $\mathcal{C}$ besteht aus der Menge von Punkten $(x_\tau, y_\tau, \mu_\tau)$, $\eta > 0$, so dass

$$F(x,y,\mu) = \begin{pmatrix} 0 \\ 0 \\ \tau e \end{pmatrix} \quad \text{und} \quad (y_\tau,\mu_\tau) > 0$$

gilt.

Ein Newton-Schritt, ausgehend von (x,y,μ) auf den Punkt $(x_{\sigma\eta}, y_{\sigma\eta}, \mu_{\sigma\eta}) \in \mathcal{C}$ zu mit $\sigma \in [0,1]$, genügt dem lineaen Gleichungs-System

$$\begin{pmatrix} Q & 0 & -A^T \\ -A & I & 0 \\ 0 & M & Y \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta\mu \end{pmatrix} = \begin{pmatrix} -r_d(x,\mu) \\ -r_b(x,y) \\ -YMe + \sigma\eta e \end{pmatrix}, \tag{3.16}$$

wobei

$$r_d(x,\mu) = Qx - A^T\mu + d \quad \text{und} \quad r_b(x,y) = b - Ax + y \tag{3.17}$$

gelten. Die nächste Iterierte ist dann für $\alpha \in (0,1]$ gegeben durch

$$(x^+, y^+, \mu^+) = (x, y, \mu) + \alpha(\Delta x, \Delta y, \Delta\mu), \tag{3.18}$$

so dass $(y^+, \mu^+) > 0$ erfüllt ist.

Lösung des primal-dualen Systems. Der Hauptaufwand der Innere-Punkte-Verfahren besteht in der Regel in der Lösung des Systems (3.16). Aufgrund der Hessematrix Q in der Koeffizientenmatrix kann die Lösung von (3.16) viel aufwendiger sein als die Lösung des KKT-System in Abschnitt 2 im Zusammenhang mit Innere-Punkte-Verfahren in der Linearen Programmierung. Daher ist es wichtig, die spezielle Struktur von (3.16) auszunutzen, indem wir eine geeignete direkte Zerlegung oder einen passenden Vorkonditionierer im Kontext von iterativen Verfahren verwenden.

Aus der dritten Blockzeile von (3.16) erhalten wir

$$\begin{aligned}\Delta y &= M^{-1}\big(-MYe + \sigma\eta e - Y\Delta\mu\big) = -Ye + \sigma\eta M^{-1}e - M^{-1}Y\Delta\mu \\ &= -y + \sigma\eta M^{-1}e - M^{-1}Y\Delta\mu.\end{aligned}$$

Ferner gilt (vergleiche zweite Blockzeile von (3.16))

$$\begin{aligned}A\Delta x - \Delta y &= A\Delta x + y - \sigma\eta M^{-1}e + M^{-1}Y\Delta\mu \\ &= \big(A \,|\, M^{-1}Y\big)\begin{pmatrix}\Delta x \\ \Delta\mu\end{pmatrix} - \big(-y + \sigma\eta M^{-1}e\big).\end{aligned}$$

Damit können wir das System (3.16) in der Form

$$\begin{pmatrix} Q & -A^T \\ A & M^{-1}Y \end{pmatrix}\begin{pmatrix}\Delta x \\ \Delta\mu\end{pmatrix} = \begin{pmatrix} -r_d(x,\mu) \\ r_b(x,y) - y + \sigma\eta M^{-1}e \end{pmatrix}. \tag{3.19}$$

Mit der zweiten Blockzeile in (3.19) erhalten wir

$$\Delta\mu = Y^{-1}M\big(r_b - y + \sigma\eta M^{-1}e + A\Delta x\big)$$

so dass die erste Blockzeile von (3.19) auf das System

$$\big(Q + A^TY^{-1}MA\big)\Delta x = -r_d + A^TY^{-1}M\big(r_b(x,y) - y + \sigma\eta M^{-1}e\big). \tag{3.20}$$

Das System (3.20) kann mit einem (modifizierten) Cholesky-Verfahren (siehe [**9**]) gelöst werden. Diese Vorgangsweise ist insbesondere geeignet, wenn die Matrix $A^TY^{-1}MA$ im Vergleich zur Matrix Q nicht so dicht besetzt ist und das System (3.20) deutlich kleiner als das in (3.19) ist. Als iteratives Verfahren kann ein projiziertes CG-Verfahren eingesetzt werden (siehe [**9**]), in dem nur Matrix-Vektor-Produkte auszuwerten sind.

Das System (3.16) kann auch in der Form

$$\begin{pmatrix} Q & 0 & -A^T \\ 0 & M & Y \\ A & -I & 0 \end{pmatrix}\begin{pmatrix}\Delta x \\ \Delta y \\ \Delta\mu\end{pmatrix} = \begin{pmatrix} -r_d(x,\mu) \\ -YMe + \sigma\eta e \\ r_b(x,y) \end{pmatrix}$$

geschrieben werden. Das sind aber die KKT-Bedingungen für das konvexe, quadratische Optimierungsproblem

$$\begin{aligned}&\min \frac{1}{2}\begin{pmatrix}\Delta x\\ \Delta y\end{pmatrix}^T\begin{pmatrix}Q & 0\\ 0 & Y^{-1}M\end{pmatrix}\begin{pmatrix}\Delta x\\ \Delta y\end{pmatrix}+\begin{pmatrix}\Delta x\\ \Delta y\end{pmatrix}^T\begin{pmatrix}r_d(x,\mu)\\ Me-\sigma\eta Y^{-1}e\end{pmatrix}\\ &\text{u.d.N. } \begin{pmatrix}A\,|\,-I\end{pmatrix}\begin{pmatrix}\Delta x\\ \Delta y\end{pmatrix}=r_b(x,y),\end{aligned}$$

welches unter Verwendung geeigneter Optimierungsverfahren gelöst werden kann, zum Beispiel mit dem projizierten CG-Verfahren.

Schrittlängen-Bestimmung. Innere Punkte-Verfahren für die Lineare Programmierung sind effizienter, wenn für die primalen und dualen Variablen unterschiedliche Schrittweiten-Parameter (α^{pri} beziehungsweise α^{dual}) verwendet werden.

Seien die nächsten Iterierten durch

$$(x^+,y^+)=(x,y)+\alpha^{\text{pri}}(\Delta x,\Delta y)\quad\text{und}\quad \mu^+=\mu+\alpha^{\text{dual}}\Delta\mu, \tag{3.21}$$

wobei $\alpha^{\text{pri}}>0$ und $\alpha^{\text{dual}}>0$ so gewählt sind dass $(y^+,\mu^+)>0$ erfüllt ist. Aus (3.16) und (3.17) folgen

$$\begin{aligned}r_b^+ &= -A\big(x+\alpha^{\text{pri}}\Delta x\big)+\big(y+\alpha^{\text{pri}}\Delta y\big)+b = r_b(x,y)-\alpha^{\text{pri}}\big(A\Delta x-\Delta y\big)\\ &= \big(1-\alpha^{\text{pri}}\big)r_b(x,y)\end{aligned} \tag{3.22a}$$

sowie

$$\begin{aligned}r_d^+ &= Q\big(x+\alpha^{\text{pri}}\Delta x\big)-A^T\big(\mu+\alpha^{\text{dual}}\Delta\mu\big)+d\\ &= r_d+\alpha^{\text{pri}}Q\Delta x-\alpha^{\text{dual}}A^T\Delta\mu\\ &= r_d+\alpha^{\text{dual}}\big(Q\Delta x-A^T\Delta\mu\big)+\big(\alpha^{\text{pri}}-\alpha^{\text{dual}}\big)Q\Delta x\\ &= \big(1-\alpha^{\text{dual}}\big)r_d+\big(\alpha^{\text{pri}}-\alpha^{\text{dual}}\big)Q\Delta x.\end{aligned} \tag{3.22b}$$

Gilt $\alpha^{\text{pri}}=\alpha^{\text{dual}}=\alpha$, so fallen beide Residuen linear für alle $\alpha\in(0,1)$. Allerdings kann für unterschiedliche Schrittweiten α^{pri} und α^{dual} die Norm $\|r_d^+\|_2$ anwachsen, so dass das Innere-Punkte-Verfahren divergiert. Eine Möglichkeit besteht darin, Schrittweiten gemäß (3.18) zu wählen, wobei wir $\alpha=\min\{\alpha_\tau^{\text{pri}},\alpha_\tau^{\text{dual}}\}$ setzen mit

$$\begin{aligned}\alpha_\tau^{\text{pri}} &= \max\big\{\alpha\in(0,1)\,|\,y+\alpha\Delta y\geq(1-\tau)y\big\},\\ \alpha_\tau^{\text{dual}} &= \max\big\{\alpha\in(0,1)\,|\,\mu+\alpha\Delta\mu\geq(1-\tau)\mu\big\}.\end{aligned} \tag{3.23}$$

Dabei steuert $\tau\in(0,1)$, wie weit wir von dem maximalen Schritt, der die Bedingungen $y+\alpha\Delta y$ und $\mu+\alpha\Delta\mu$ erfüllt, (relativ) entfernt sind.

Die numerische Erfahrung hat allerdings gezeigt, dass die Wahl unterschiedlicher Schrittweiten für die primalen und dualen Variablen oft zu schnellerer Konvergenz der Innere-Punkte-Verfahrens führt. Eine Möglichkeit zur Wahl unterschiedlicher Schrittweiten ist, $(\alpha^{\text{pri}},\alpha^{\text{dual}})$ als (näherungsweise) Lösung der Minimierungsaufgabe

$$\begin{aligned}&\min \|Qx^+-A^T\mu^++d\|_2^2+\|Ax^+-y^+-b\|_2^2+\big(y^+\big)^T\mu^+\\ &\text{u.d.N. } 0\leq\alpha^{\text{pri}}\leq\alpha_\tau^{\text{pri}},\ 0\leq\alpha^{\text{dual}}\leq\alpha_\tau^{\text{dual}}\text{ und }(x^+,y^+,\mu^+)\text{ gemäß (3.18).}\end{aligned}$$

Ein praktischer Primal-dualer Algorithmus. Die am meisten verwendete Variante des Innere-Punkte-Verfahrens basiert auf dem Pädiktor-Korrektor Algorithmus von Mehrotra; vergleiche Abschnitt 4. Zuerst wird ein affiner Skalierungsschritt $(\Delta x^{\text{aff}}, \Delta y^{\text{aff}}, \Delta \mu^{\text{aff}})$ bestimmt, indem in (3.16) mit $\sigma = 0$. Die erhaltene Richtung wird dann in einem nachfolgenden Korrekturschritt verbessert, wobei $\sigma = (\eta^{\text{aff}}/\eta)^3$ gesetzt wird. Insgesamt lösen wir im Korrekturschritt das System

$$\begin{pmatrix} Q & 0 & -A^T \\ A & I & 0 \\ 0 & M & Y \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta \mu \end{pmatrix} = \begin{pmatrix} -r_d(x,\mu) \\ -r_b(x,y) \\ -MYe - \Delta M^{\text{aff}} \Delta Y^{\text{aff}} e + \sigma \eta e \end{pmatrix}. \tag{3.24}$$

Algorithmus 3.10 (Prädiktor-Korrektor Verfahren für quadratische Probleme).

1) *Wähle* (x^0, y^0, μ^0) *mit* $(y^+, \mu^+) > 0$ *und setze* $k = 0$.
2) *Setze* $x, y, \mu) = (x^k, y^k, \mu^k)$ *und löse* (3.16) *für* $(\Delta x^{\text{aff}}, \Delta y^{\text{aff}}, \Delta \mu^{\text{aff}})$ *mit* $\sigma = 0$.
3) *Berechne* $\eta = y^T \mu / m$.
4) *Setze* $\hat{\alpha}^{\text{aff}} = \max\{\alpha \in (0,1] \,|\, (y,\mu) + \alpha(\Delta y^{\text{aff}}, \Delta \mu^{\text{aff}}) \geq 0\}$
5) *Bestimme* $\eta^{\text{aff}} = (y + \alpha^{\text{aff}} \Delta y)^T (\mu + \alpha^{\text{aff}} \Delta \mu)/m$ *und wähle* $\sigma = (\eta^{\text{aff}}/\eta)^3$.
6) *Löse* (3.24) *für* $(\Delta x, \Delta y, \Delta \mu)$.
7) *Wähle* $\tau_k \in (0,1)$ *und setze* $\hat{\alpha} = \min\{\alpha^{\text{pri}}_{\tau_k}, \alpha^{\text{dual}}_{\tau_k}\}$*; vergleiche* (3.23).
8) *Setze* $(x^{k+1}, y^{k+1}, \mu^{k+1}) = (x^k, y^k, \mu^k) + \hat{\alpha}(\Delta x, \Delta y, \Delta \mu)$, $k = k+1$ *und gehe zurück zu Schritt* 2).

Im Algorithmus 3.10 können wir $t_k \to 1$ wählen, wenn die Iterierten konvergieren, um die Konvergenz zu beschleunigen. Wie im Fall der Linearen Programmierung hängt die Effizienz von Algorithmus 3.10 von der Wahl geeigneter Startwerte ab. Eine mögliche Heuristik verwendet einen gegebenen Startwert $(\bar{x}, \bar{y}, \bar{\mu})$, um diesen hinreichend weit weg von dem Rand der durch die Bedingung $(y, \mu) \geq 0$ definierten Menge zu verschieben, so dass zu Beginn von Algorithmus 3.10 große Schrittweiten möglich sind.

KAPITEL 4

SQP-Verfahren

In diesem Abschnitt werden wir uns mit einem der effizientesten Verfahren der nichtlinearen restringierten Optimierung beschäftigen, mit dem *SQP-Verfahren*. Dabei steht SQP für *sequential quadratic programming*.

1. Das lokale SQP-Verfahren

Wir betrachten

$$\textbf{(P)} \qquad \min J(x) \qquad \text{u.d.N.} \quad e(x) = 0,$$

wobei $J : \mathbb{R}^n \to \mathbb{R}$ und $e : \mathbb{R}^n \to \mathbb{R}^m$ zweimal stetig differenzierbar sind und die zweiten Ableitungen Lipschitz-stetig sind.

Wesentliche Idee des SQP-Verfahrens ist es, dass (**P**) an jeder Iterierten x^k durch ein quadratisches Modell ersetzt wird und der Minimierer dazu benutzt wird, die neue Iterierte x^{k+1} zu berechnen.

Die Lagrange-Funktion zu (**P**) lautet

$$L(x, \lambda) = J(x) + \lambda^T e(x).$$

Wir bezeichnen mit

$$A(x) = \begin{pmatrix} \nabla e_1(x) \\ \vdots \\ \nabla e_m(x) \end{pmatrix} \in \mathbb{R}^{m \times n} \tag{4.1}$$

die Jacobi-Matrix von e am Punkt x. Die KKT-Bedingungen $\nabla L(x, \lambda) = 0$ ergeben

$$F(x, \lambda) = \begin{pmatrix} \nabla J(x)^T + A(x)^T \lambda \\ e(x) \end{pmatrix} \stackrel{!}{=} 0. \tag{4.2}$$

Hat $A(x^*)$ vollen Rang m, so ist x^* ein regulärer Punkt und es existiert zu jeder Lösung $x^* \in \mathbb{R}^n$ von (**P**) ein zugehöriger Lagrange-Multiplikator $\lambda^* \in \mathbb{R}^m$ mit $F(x^*, \lambda^*) = 0$. Wir lösen (4.2) mit dem Newton-Verfahren. Die Jacobi-Matrix von F ist

$$\nabla^2 L(x, \lambda) = \begin{pmatrix} \nabla_{xx} L(x, \lambda) & A(x)^T \\ A(x) & 0 \end{pmatrix}. \tag{4.3}$$

Damit ist der Newton-Schritt von der Iterierten (x^k, λ^k) gegeben durch

$$\begin{pmatrix} x^{k+1} \\ \lambda^{k+1} \end{pmatrix} = \begin{pmatrix} x^k \\ \lambda^k \end{pmatrix} + \begin{pmatrix} \Delta x^k \\ \Delta \lambda^k \end{pmatrix}, \tag{4.4a}$$

wobei

$$\nabla^2 L(x^k, \lambda^k) \begin{pmatrix} \Delta x^k \\ \Delta \lambda^k \end{pmatrix} = -\nabla L(x^k, \lambda^k)^T \tag{4.4b}$$

gilt. Das Verfahren (4.4) heißt daher oft auch *Lagrange-Newton-SQP Verfahren*, da es als Newton-Verfahren in den Variablen (x^k, λ^k) interpretiert werden kann. Ist die Hesse-Matrix $\nabla^2 L(x^k, \lambda^k)$ invertierbar, so ist die Iteration (4.4) wohldefiniert. Voraussetzungen für die Regularität von $\nabla^2 L(x^k, \lambda^k)$ sind in Lemma 3.1 gegeben.

VORAUSSETZUNG 4.1.
1) *Die Jacobi-Matrix $A_k = \nabla e(x^k)$ hat vollen Rang.*
2) *Die Hesse-Matrix $\nabla_{xx} L(x^k, \lambda^k)$ ist positiv definit auf dem Kern der Matrix $A(x^k)$.*

BEMERKUNG 4.2.
1) Gilt

$$\Delta x^T \nabla_{xx} L(x^*, \lambda^*) \Delta x \geq \kappa \|\Delta x\|^2 \quad \text{für alle } \Delta x \in \operatorname{Kern} A(x^*)$$

für ein $\kappa > 0$ (hinreichende Bedingungen zweiter Ordnung), so folgt die Voraussetzung 4.1-2) in einer Umgebung von (x^*, λ^*).
2) Aufgrund der Theorie des Newton-Verfahrens ergibt sich für das SQP-Verfahren lokal quadratische Konvergenz in (x, λ), das heißt, es gibt ein $C > 0$ mit

$$\|(x^{k+1}, \lambda^{k+1}) - (x^*, \lambda^*)\| \leq C \|(x^k, \lambda^k) - (x^*, \lambda^*)\|^2 \quad \text{für alle } k \geq 0,$$

sofern $\|(x^0, \lambda^0) - (x^*, \lambda^*)\|$ hinreichend klein sind. ◊

Wir wollen nun eine andere Motivation für (4.4) geben. Dazu betrachten wir das quadratische Problem

$$\min_{\Delta x^k} \frac{1}{2} (\Delta x^k)^T \nabla_{xx} L(x^k, \lambda^k) \Delta x^k + \nabla J(x^k) \Delta x^k \tag{4.5a}$$

$$\text{u.d.N. } A_k \Delta x^k + e(x^k) = 0. \tag{4.5b}$$

Hier erkennen wir, warum das Verfahren SQP-Algorithmus heißt: Es sind in jeder Iteration quadratische Probleme zu lösen.

Die Optimalitäts-Bedingungen für das quadratische Problem (4.5) lauten

$$\nabla_{xx} L(x^k, \lambda^k) \Delta x^k + \nabla J(x^k)^T + A_k^T \mu^k = 0, \tag{4.6a}$$

$$A_k \Delta x^k = -e(x^k) \tag{4.6b}$$

mit einem Lagrange-Multiplikator $\mu^k \in \mathbb{R}^m$. Die Voraussetzung 4.1 garantiert, dass es eine eindeutige Lösung $(\Delta x^k, \mu^k)$ von (4.6) existiert. Wenn wir $A_k^T \lambda^k$ in der ersten Blockzeile auf beiden Seiten von (4.4b) addieren, so erhalten wir

$$\nabla_{xx} L(x^k, \lambda^k) \Delta x^k + A_k^T (\underbrace{\lambda^{k+1} - \lambda^k}_{=\Delta\lambda^k}) + A_k^T \lambda^k = -\nabla J(x^k)^T - A_k^T \lambda^k + A_k^T \lambda^k.$$

Insgesamt ergibt damit (4.4b)

$$\begin{pmatrix} \nabla_{xx} L(x^k, \lambda^k) & A_k^T \\ A_k & 0 \end{pmatrix} \begin{pmatrix} \Delta x^k \\ \lambda^{k+1} \end{pmatrix} = - \begin{pmatrix} \nabla J(x^k)^T \\ e(x^k) \end{pmatrix}. \tag{4.7}$$

Unter der Voraussetzung $\det(\nabla^2 L(x^k, \lambda^k)) \neq 0$ folgt $\lambda^{k+1} = \mu^k$. Damit sind das Newton- und das SQP-Verfahren äquivalent. Unter der Voraussetzung 4.1 an (x^k, λ^k) kann die nächste Iterierte (x^{k+1}, λ^{k+1}) als Lösung des quadratischen Problems (4.5) oder als Iterierte des Newton-Verfahrens (4.4) berechnet werden.

Aus der Sicht des Newton-Verfahrens lassen sich eher theoretische Resultate nachweisen (zum Beispiel die quadratische Konvergenz), während die Interpretation

als SQP-Algorithmus es ermöglicht, praktische Verfahren zu entwerfen und auch Ungleichungen zu berücksichtigen.

ALGORITHMUS 4.3 (Lokales SQP-Verfahren).

1) *Wähle Startwerte* $x^0 \in \mathbb{R}^n$, $\lambda^0 \in \mathbb{R}^m$ *und* $k_{\max} \in \mathbb{N}$.
2) *For* $k = 0, 1, \ldots, k_{\max}$
   - *Berechne* $J_k = J(x^k)$, $\nabla J_k = \nabla J(x^k)$, $\nabla_{xx} L(x^k, \lambda^k)$, $e_k = e(x^k)$ *und* $A_k = \nabla e(x^k)$*;*
   - *Löse* (4.5) *für* $(\Delta x^k, \mu^k)$*;*
   - *Setze* $x^{k+1} = x^k + \Delta x^k$ *und* $\lambda^{k+1} = \mu^k$*;*
   - *Prüfe die Abbruchkriterien;*

   end (For).

Wir haben bereits erwähnt, dass die lokal quadratische Konvergenz in (x^k, λ^k) von Algorithmus 4.3 aus der lokalen Äquivalenz mit dem Newton-Verfahren für die Gleichung (4.2) folgt.

Das Zielfunktional in (4.5a)

$$\frac{1}{2}(\Delta x^k)^T \nabla_{xx} L(x^k, \lambda^k) \Delta x^k + \nabla J(x^k) \Delta x^k$$

können wir wegen (4.5b) durch

$$\frac{1}{2}(\Delta x^k)^T \nabla_{xx} L(x^k, \lambda^k) \Delta x^k + \nabla_x L(x^k, \lambda^k) \Delta x^k$$

ersetzen; denn es gilt

$$\begin{aligned} \nabla_x L(x^k, \lambda^k) \Delta x^k &= \nabla J(x^k) \Delta x^k + (\lambda^k)^T \nabla e(x^k) \Delta x^k \\ &= \nabla J(x^k) \Delta x^k + (\lambda^k)^T (-e(x^k)) \\ &= \nabla J(x^k) \Delta x^k - (\lambda^k)^T e(x^k) \end{aligned}$$

und der Term $-(\lambda^k)^T e(x^k)$ ist konstant, beeinflußt daher die Minimierung nicht Damit können wir (4.5) auch durch

$$\begin{aligned} &\min_{\Delta x^k} \frac{1}{2}(\Delta x^k)^T \nabla_{xx} L(x^k, \lambda^k) \Delta x^k + \nabla_x L(x^k, \lambda^k) \Delta x^k \\ &\text{u.d.N. } A_k \Delta x^k + e(x^k) = 0. \end{aligned}$$

ersetzen.

Das SQP-Verfahren kann einfach auf nichtlineare Probleme mit Ungleichungs-Nebenbedingungen erweitert werden. Wir betrachten

$$\min J(x) \quad \text{u.d.N.} \quad e(x) = 0 \text{ in } \mathbb{R}^m \text{ und } g(x) \leq 0 \text{ in } \mathbb{R}^p. \tag{4.8}$$

Zur Lösung von (4.8) linearizieren wir sowohl die Gleichungs- als auch die Ungleichungs-Nebenbedingungen. Dann erhalten wir

$$\begin{cases} \min\limits_{\Delta x^k \in \mathbb{R}^n} J_k + \nabla J_k \Delta x + \dfrac{1}{2}\left(\Delta x^k\right)^T \nabla_{xx} L(x^k, \lambda^k) \Delta x^k \\ \text{u.d.N. } \nabla e(x^k) \Delta x^k + e(x^k) = 0 \text{ in } \mathbb{R}^m, \ \nabla g(x^k) \Delta x^k + g(x^k) \leq 0 \text{ in } \mathbb{R}^p. \end{cases} \tag{4.9}$$

Nun können wir Algorithmen für quadratische Programme verwenden, z.B., das Innere-Punkte-Verfahren aus Abschnitt 4. Die neue Iterierte ist durch das Paar $(x^k + \Delta x^k, \lambda^{k+1})$ gegeben, wobei Δx^k und λ^{k+1} die Lösung beziehungsweise der assoziierte Lagrange-Multiplikator von (4.9) sind. Ein lokales SQP-Verfahren für

(4.8) hat damit die Form von Algorithmus 4.3 mit der Modifikation, dass der Schritt durch die Lösung von (4.9) bestimmt wird.

2. Berechnung des SQP-Schrittes

Wie im vorigen Abschnitt wollen wir uns auch hier auf Gleichungs-Restriktionen beschränken.

a) Als erste Alternative können wir das System (4.7) entweder mit einem direktem Verfahren (zum Beispiel einer *LR*-Zerlegung) oder einem iterativen Gleichungslöser (zum Beispiel GMRES-, QMR-, MINRES- oder SYMMLQ-Verfahren, siehe [**4**]) lösen. Die Vorkonditionierung der iterativen Verfahren ist ein aktives Forschungsgebiet.

b) Wenn $\nabla_{xx} L(x^k, \lambda^k)$ positiv definit ist, können wir das System (4.7) entkoppeln und gemäß der Range-Space Methode in Abschnitt 2 lösen. Dann erhalten wir die beiden Gleichungen

$$\begin{aligned} \big(A_k \nabla_{xx} L(x^k, \lambda^k)^{-1} A_k^T\big)\lambda^{k+1} &= -A_k \nabla_{xx} L(x^k, \lambda^k)^{-1} \nabla J_k^T + e_k, \\ \nabla_{xx} L(x^k, \lambda^k)\Delta x^k &= -\nabla J_k^T - A_k^T \lambda^{k+1} \end{aligned}$$

mit $A_k = \nabla e(x^k)$, $\nabla J_k = \nabla J(x^k)$ und $e_k = e(x^k)$. Dieser Lösungsweg ist insbesondere dann sehr effizient, wenn $\nabla_{xx} L(x^k, \lambda^k)$ durch positiv definite Approximationen ersetzt wird, zum Beispiel durch Quasi-Newton Updates.

c) Die letzte Möglichkeit wird bei sehr vielen SQP-Verfahren genutzt. Hier wird die Idee der Null-Space Methode aus dem dritten Abschnitt angewendet. Wir müssen also Matrizen Z_k und Y_k bestimmen, wobei die Spalten von Z_k eine Basis des Nullraums von A_k und die Spalten von Y_k eine des Bildraums von A_k^T bilden. Mit

$$\Delta x^k = Y_k \Delta x_Y^k + Z_k \Delta x_Z^k$$

ergeben sich aus (4.6) die beiden Gleichungen

$$(A_k Y_k)\Delta x_Y^k = -e_k, \tag{4.10a}$$

$$\big(Z_k^T \nabla_{xx} L(x^k, \lambda^k) Z_k\big)\Delta x_Z^k = -Z_k^T \nabla_{xx} L(x^k, \lambda^k) Y_k \Delta x_Y^k - Z_k^T \nabla J_k^T. \tag{4.10b}$$

Der Lagrange-Multiplikator zu Problem (4.5) ergibt sich dann aus

$$(A_k Y_k)^T \lambda^{k+1} = -Y_k^T (\nabla J_k^T + \nabla_{xx} L(x^k, \lambda^k)\Delta x^k). \tag{4.10c}$$

Für diesen Lösungsweg benötigen wir nur, dass die reduzierte Hesse-Matrix $Z_k^T \nabla_{xx} L(x^k, \lambda^k) Z_k$ positiv definit ist. Eine Variante berechnet λ^{k+1} in (4.10c), indem auf der rechten Seite der Term $\nabla_{xx} L(x^k, \lambda^k)\Delta x^k$ weggelassen wird. Wegen $\Delta x^k \to 0$ macht dieses auch Sinn. Weiters können wir im Fall von $\operatorname{Rang} A_k^T = m$ die Wahl $Y_k = A_k^T$ treffen, was auf

$$\hat{\lambda}^{k+1} = -(A_k A_k^T)^{-1} A_k \nabla J_k^T \tag{4.11}$$

führt. Dieser Lagrange-Multiplikator wird als *Least-Squares Multiplikator* bezeichnet; denn er ergibt sich als Lösung des Least-Squares-Problems

$$\min_{\hat{\lambda} \in \mathbb{R}^m} \|\nabla J_k^T + A_k^T \hat{\lambda}\|_2. \tag{4.12}$$

Offenbar sind nämlich die Optimalitäts-Bedingungen für (4.12)

$$\left(\nabla J_k^T + A_k^T \hat{\lambda}\right)^T \left(A_k^T v\right) = 0 \quad \text{für alle } v \in \mathbb{R}^n,$$

was auf die Normalgleichungen

$$A_k A_k^T \hat{\lambda} = -A_k \nabla J_k^T$$

führt. Auch wenn x^k weit von der Lösung x^* von (**P**) entfernt ist, macht (4.11) Sinn, da in jeder Iteration der Lagrange-Multiplikator so gewählt wird, dass die Norm

$$\|\nabla_x L(x, \hat{\lambda})^T\|_2 = \|\nabla J(x)^T + A(x)^T \hat{\lambda}\|_2$$

minimiert wird, was durch die Optimalitäts-Bedingungen $\nabla_x L(x, \lambda) = 0$ motiviert ist, vergleiche (4.2). Wir berechnen daher $\hat{\lambda}^{k+1}$ durch (4.11), wobei wir auf der rechten Seite die Ausdrücke bereits an der neuen Iterierten auswerten:

$$\hat{\lambda}^{k+1} = -(A_{k+1} A_{k+1}^T)^{-1} A_{k+1} \nabla J_{k+1}^T.$$

Damit wird das SQP-Verfahren in ein Verfahren transformiert, das nur auf der primalen Variablen x^k arbeitet, denn $\hat{\lambda}^k$ hängt nur von x^k, nicht aber von $\hat{\lambda}^{k-1}$ ab.

Eine weitere Variante vernachlässigt $Z_k^T \nabla_{xx} L(x^k, \lambda^k) Y_k \Delta x_Y^k$ auf der rechten Seite von (4.10b). Es wird also nur das System

$$\left(Z_k^T \nabla_{xx} L(x^k, \lambda^k) Z_k\right) \Delta x_Z^k = -Z_k^T \nabla J_k^T \tag{4.13}$$

gelöst. Die Konvergenz dieser sogenannten *reduced SQP methods* wurde in [**7**] untersucht.

3. Die Hesse-Matrix des quadratischen Modells

In Abschnitt 1 haben wir über die Äquivalenz des SQP- mit dem Newton-Verfahren gesprochen. Unter sinnvollen Voraussetzungen erhalten wir daher lokal quadratische Konvergenz. Unter Umständen ist aber die Matrix

$$\nabla_{xx} L(x^k, \lambda^k) = \nabla^2 J(x^k) + \sum_{i=1}^{m} \lambda_i^k \nabla^2 e_i(x^k)$$

schwer zu berechnen oder nicht positiv definit auf dem Kern der linearisierten Nebenbedingungen. Eine Alternative ist daher, $\nabla_{xx} L(x^k, \lambda^k)$ durch eine Quasi-Newton Approximation B_k zu ersetzen. Die Quasi-Newton Updates haben sich bereits sehr effizient in der unrestringierten Optimierung erwiesen. Wir werden sie daher jetzt hier anwenden. Der Update für B_k beim Schritt von k nach $k+1$ verwendet die Vektoren

$$s^k = x^{k+1} - x^k \quad \text{und} \quad y^k = \nabla_x L(x^{k+1}, \lambda^{k+1})^T - \nabla_x L(x^k, \lambda^k)^T \tag{4.14}$$

zum Beispiel beim BFGS-Update wie folgt

$$B_{k+1} = B_k - \frac{B_k s^k (s^k)^T B_k}{(s^k)^T B_k s^k} + \frac{y^k (y^k)^T}{(y^k)^T s^k}.$$

Diese Variante können wir als Quasi-Newton Update für den Fall deuten, dass die Zielfunktion durch $L(x, \lambda)$ bei fixiertem λ gegeben ist. Das macht die Stärken, aber auch die Schwächen dieser Variante klar. Ist $\nabla_{xx} L(x^k, \lambda^k)$ in der Region, wo die

Minimierung durchgeführt wird, positiv definit, so geben die Quasi-Newton Approximationen B_k gute Informationen über die Krümmung und das Verfahren konvergiert schnell und robust gegen die Minimalstelle. Besitzt hingegen $\nabla_{xx}L(x^k,\lambda^k)$ negative Eigenwerte, so sind die positiv definiten Approximationen nicht sehr geeignet. Die Bedingung $(s^k)^T y^k > 0$ braucht noch nicht einmal in einer kleinen Umgebung der Lösung zu gelten. Diese Beobachtungen haben zu folgenden gedämpften BFGS-Update Formeln für SQP-Verfahren geführt:

1) Definiere die Vektoren s^k und y^k gemäß (4.14) und setze

$$r^k = \theta_k y^k + (1-\theta_k) B_k s^k,$$

wobei der Skalar $\theta_k \in [0,1]$ gegeben ist durch

$$\theta_k = \begin{cases} 1 & \text{falls } (s^k)^T y^k \geq 0.2 (s^k)^T B_k s^k, \\ \frac{0.8 (s^k)^T B_k s^k}{(s^k)^T B_k s^k - (s^k)^T y^k} & \text{falls } (s^k)^T y^k < 0.2 (s^k)^T B_k s^k. \end{cases} \tag{4.15}$$

2) Berechne B_{k+1} mit der Update-Formel

$$B_{k+1} = B_k - \frac{B_k s^k (s^k)^T B_k}{(s^k)^T B_k s^k} + \frac{r^k (r^k)^T}{(r^k)^T s^k}. \tag{4.16}$$

Die Formel (4.16) ist die BFGS-Update Formel, wobei y^k durch den Vektor r^k ersetzt worden ist. Für $\theta_k = 1$ folgt $r^k = y^k$. Im Fall von $\theta_k \neq 1$ erhalten wir mit (4.15) die Abschätzung

$$\begin{aligned} (s^k)^T r^k &= (s^k)^T \big(\theta_k y^k + (1-\theta_k) B_k s^k\big) = \theta_k (s^k)^T y^k + (1-\theta_k)(s^k)^T B_k s^k \\ &= \frac{0.8 (s^k)^T B_k s^k}{(s^k)^T B_k s^k - (s^k)^T y^k} (s^k)^T y^k + \frac{0.2 (s^k)^T B_k s^k - (s^k)^T y^k}{(s^k)^T B_k s^k - (s^k)^T y^k} (s^k)^T B_k s^k \\ &= \left(\frac{0.8 (s^k)^T y^k}{(s^k)^T B_k s^k - (s^k)^T y^k} + \frac{0.2 (s^k)^T B_k s^k - (s^k)^T y^k}{(s^k)^T B_k s^k - (s^k)^T y^k} \right) (s^k)^T B_k s^k \\ &= 0.2 (s^k)^T B_k s^k > 0. \end{aligned}$$

Damit ist B_{k+1} positiv definit. Für $\theta_k = 0$ folgt $B_k = B_{k+1}$. Andererseits führt $\theta_k = 1$ auf eine möglicherweise indefinite Matrix, die sich aus den unmodifizierten BFGS-Formeln ergibt. Mit $\theta_k \in (0,1)$ erhalten wir eine Interpolation der beiden Extremfälle.

Eine andere Variante bietet sich dadurch an, die reduzierte Hesse-Matrix der Lagrange-Funktion $Z_k^T \nabla_{xx} L(x^k,\lambda^k) Z_k$ zu approximieren, insbesondere dann, wenn die Dimension dieser Matrix klein ist. Die Herangehensweise ist in *Reduced-Hessian Quasi-Newton Methods* realisiert. Die Suchrichtung erfüllt

$$\lambda^k = -(A_k A_k^T)^{-1} A_k \nabla J_k^T, \tag{4.17a}$$

$$(A_k A_k^T) \Delta x_Y^k = -e_k, \tag{4.17b}$$

$$M_k \Delta x_Z^k = -Z_k^T \nabla J_k^T, \tag{4.17c}$$

wobei wir in (4.17c) im Vergleich mit (4.13) eine Quasi-Newton Approximation für die reduzierte Hesse-Matrix $Z_k^T \nabla_{xx} L(x^k,\lambda^k) Z_k$ verwendet haben. Im Folgenden wollen wir diskutieren, wie die Matrizen M_k konstruiert werden können. Sei $\alpha_k \Delta x^k$ der Schritt von (x^k,λ^k) nach (x^{k+1},λ^{k+1}). Wegen des Satzes von Taylor und der

Lipschitz-Stetigkeit von ∇L_{xx} folgt

$$\begin{aligned}
&\nabla_{xx} L(x^{k+1}, \lambda^{k+1})\big(\alpha_k \Delta x^k\big) \\
&\quad = \nabla_{xx} L(x^k, \lambda^{k+1})\big(\alpha_k \Delta x^k\big) + \big(\nabla_{xx} L(x^{k+1}, \lambda^{k+1}) - \nabla_{xx} L(x^k, \lambda^{k+1})\big)\big(\alpha_k \Delta x^k\big) \\
&\quad = \nabla_x L(x^k + \alpha_k \Delta x^k, \lambda^{k+1})^T - \nabla_x L(x^k, \lambda^{k+1})^T + \mathcal{O}\big(\alpha_k \|\Delta x^k\|\big)
\end{aligned}$$

mit $\Delta x^k = x^{k+1} - x^k = Z_k \Delta x_Z^k + Y_k \Delta x_Y^k$. Multiplikation mit Z_k^T ergibt

$$\begin{aligned}
&Z_k^T \nabla_{xx} L(x^{k+1}, \lambda^{k+1}) Z_k \big(\alpha_k \Delta x_Z^k\big) \\
&\qquad \approx -Z_k^T \nabla_{xx} L(x^{k+1}, \lambda^{k+1})\big(\alpha_k Y_k \Delta x_Y^k\big) \\
&\qquad\quad + Z_k^T \big(\nabla_x L(x^{k+1}, \lambda^{k+1})^T - \nabla_x L(x^k, \lambda^{k+1})^T\big).
\end{aligned} \tag{4.18}$$

Vernachlässigen des ersten Terms auf der rechten Seite von (4.18) führt auf

$$M_{k+1} s^k = y^k$$

mit

$$s^k = \alpha_k \Delta x_Z^k \quad \text{und} \quad y^k = Z_k^T \big(\nabla_x L(x^{k+1}, \lambda^{k+1})^T - \nabla_x L(x^k, \lambda^{k+1})^T\big). \tag{4.19}$$

Damit können wir die BFGS-Formeln

$$M_{k+1} = M_k - \frac{M_k s^k (s^k)^T M_k}{(s^k)^T M_k s^k} + \frac{y^k (y^k)^T}{(y^k)^T s^k}$$

verwenden, um die neue Approximation M_{k+1} zu berechnen. Es gibt Varianten von (4.19), zum Beispiel

$$y^k = Z_k^T \big(\nabla J_{k+1}^T - \nabla J_k^T\big)$$

oder

$$y^k = Z_k^T \big(\nabla_x L(x^k + Z_k \Delta x_Z^k, \lambda^{k+1})^T - \nabla_x L(x^k, \lambda^{k+1})^T\big) \tag{4.20}$$

(Cole und Coleman). In (4.20) ist die zusätzliche Auswertung des Terms $\nabla_x L(x^k + Z_k \Delta x_Z^k, \lambda^{k+1})$ notwendig. In einer Umgebung der Lösung gilt für (4.20) die Abschätzung

$$\begin{aligned}
(y^k)^T s^k &= \alpha_k \big(\nabla_x L(x^k + Z_k \Delta x_Z^k, \lambda^{k+1}) - \nabla_x L(x^k, \lambda^{k+1})\big) Z_k \Delta x_Z^k \\
&= \alpha_k (\Delta x_Z^k)^T \left(\int_0^1 Z_k^T \nabla_{xx} L(x^k + s Z_k \Delta x_Z^k, \lambda^{k+1}) Z_k \, \mathrm{d}s \right) \Delta x_Z^k > 0
\end{aligned}$$

für $(x^k + s Z_k \Delta x_Z^k, \lambda^{k+1}) \in U(x^*, \lambda^*)$. Damit sind die BFGS-Formeln wohldefiniert.

4. Merit- oder Straffunktionen

Um zu garantieren, dass das SQP-Verfahren von Startwerten, die weit weg von Lösungen liegen, konvergiert, wird häufig eine Merit- oder Straffunktion verwendet, um bei Liniensuch-Verfahren die Schrittweite zu kontrollieren oder bei Trust-Region Verfahren den Trust-Region zu modifizieren. Im unrestringierten Fall haben wir die Zielfunktion verwendet. Wir wollen hier nur zwei Meritfunktionen diskutieren: die nicht-differenzierbare ℓ_1-*Meritfunktion* sowie Fletchers exakte und differenzierbare *augmentierte Lagrange Funktion.*

Ziel der Meritfunktion ist die Garantie globaler Konvergenz ohne Schritte zu verwerfen, die zur Lösung führen. Die ℓ_1-Meritfunktion für Probleme mit Gleichungs-Restriktionen lautet

$$\Phi_1(x;\mu) = J(x) + \frac{1}{\mu}\,\|e(x)\|_1 = J(x) + \frac{1}{\mu}\sum_{i=1}^{m}|e_i(x)|, \tag{4.21}$$

wobei $\mu > 0$ ein Strafparameter ist. Die Abbildung Φ_1 ist insbesondere für Punkte x mit $e_i(x) = 0$ für mindestens ein $i \in \{1,\dots,m\}$ nicht differenzierbar. Eine Richtungs-Ableitung von Φ_1 existiert dagegen immer.

Beispiel 4.4. Wir berechnen für $J(x) = \|x\|_1$ die Richtungs-Ableitung

$$D(J(x);\Delta x) = \lim_{\varepsilon\searrow 0}\frac{1}{\varepsilon}\big(J(x+\varepsilon\Delta x) - J(x)\big)$$

in eine Richtung $\Delta x \in \mathbb{R}^n$. Wir erhalten

$$D(J(x);\Delta x) = \lim_{\varepsilon\searrow 0}\frac{1}{\varepsilon}\big(\|x+\varepsilon\Delta x\|_1 - \|x\|_1\big) = \lim_{\varepsilon\searrow 0}\frac{1}{\varepsilon}\sum_{i=1}^{n}\big(|x_i+\varepsilon\Delta x_i| - |x_i|\big).$$

Gilt $x_i > 0$ für ein $i \in \{1,\dots,n\}$, so folgt $|x_i+\varepsilon\Delta x_i| = x_i+\varepsilon\Delta x_i$ für ε hinreichend klein. Ist hingegen $x_i < 0$ für ein $i \in \{1,\dots,n\}$, so erhalten wir $|x_i+\varepsilon\Delta x_i| = |x_i| - \varepsilon\Delta x_i$ für ε klein genug. Im Fall von $x_i = 0$ für ein $i \in \{1,\dots,n\}$ gilt $|x_i+\varepsilon\Delta x_i| = \varepsilon\,|\Delta x_i|$. Insgesamt berechnen wir daher

$$D(J(x);\Delta x) = \sum_{i:x_i>0}\Delta x_i - \sum_{i:x_i<0}\Delta x_i + \sum_{i:x_i=0}|\Delta x_i|$$

als Richtungs-Ableitung von J am Punkt x in Richtung Δx. ◊

Lemma 4.5. *Seien Δx^k und λ^{k+1} Lösungen von (4.7). Dann gilt für die Richtungs-Ableitung von Φ_1 in Richtung Δx^k die Abschätzung*

$$D(\Phi_1(x^k;\mu);\Delta x^k) \le -(\Delta x^k)^T\nabla_{xx}L(x^k,\lambda^k)\Delta x^k - \left(\frac{1}{\mu} - \|\lambda^{k+1}\|_\infty\right)\|e(x^k)\|_1. \tag{4.22}$$

Bemerkung 4.6. Ist $\nabla_{xx}L(x^k,\lambda^k)$ positiv definit, so folgt aus (4.22), dass Δx^k eine Abstiegs-Richtung für Φ_1 an x^k ist, wenn μ hinreichend klein gewählt wird. Es lässt sich zeigen, dass dieser Schluß auch gilt, wenn die reduzierte Hesse-Matrix $Z_k^T\nabla_{xx}L(x^k,\lambda^k)Z_k$ positiv definit ist. In der Praxis wird

$$\mu = \frac{1}{\|\lambda^{k+1}\|_\infty + \delta}$$

mit einem $\delta > 0$ gewählt. Eine andere Möglichkeit ist es zu fordern, dass die Richtungsableitung von Φ_1 hinreichend negativ ist:

$$D\Phi_1(x^k;\mu);\Delta x^k) = \nabla J(x^k)\Delta x^k - \frac{1}{\mu}\,\|e(x^k)\|_1 \le -\frac{\varrho}{\mu}\,\|e(x^k)\|_1 \tag{4.23}$$

für ein $\varrho \in (0,1)$, wobei wir (4.25) verwendet haben. Diese Ungleichung gilt, sofern

$$\frac{1}{\mu} \ge \frac{\nabla J(x^k)\Delta x^k}{(1-\varrho)\|e(x^k)\|_1} \quad \text{beziehungsweise} \quad \mu \le \frac{(1-\varrho)\|e(x^k)\|_1}{\nabla J(x^k)\Delta x^k}; \tag{4.24}$$

denn wir haben mit (4.24) und $1-\varrho>0$

$$\begin{aligned} D\Phi_1(x^k;\mu);\Delta x^k) + \frac{\varrho}{\mu}\left\|e(x^k)\right\|_1 &= \nabla J(x^k)\Delta x^k - \frac{1-\varrho}{\mu}\left\|e(x^k)\right\|_1 \\ &\leq \nabla J(x^k)\Delta x^k - \frac{\nabla J(x^k)\Delta x^k}{\left\|e(x^k)\right\|_1}\left\|e(x^k)\right\|_1 = 0, \end{aligned}$$

was (4.23) ergibt. Diese Wahl für μ hängt nicht vom Lagrange-Multiplikator ab und wird in der Praxis häufig verwendet. ◊

Beweis von Lemma 4.5. Wir wenden den Satz von Taylor an:

$$\begin{aligned} \Phi_1(x^k+\alpha\Delta x^k;\mu) - \Phi_1(x^k;\mu) &= J(x^k+\alpha\Delta x^k) - J(x^k) \\ &\quad + \frac{1}{\mu}\left(\left\|e(x^k+\alpha\Delta x^k)\right\|_1 - \left\|e(x^k)\right\|_1\right) \\ &\leq \alpha\nabla J(x^k)\Delta x^k + \gamma\alpha^2\left\|\Delta x^k\right\|^2 \\ &\quad + \frac{1}{\mu}\left(\left\|e(x^k)+\alpha A_k\Delta x^k\right\|_1 - \left\|e(x^k)\right\|_1\right) \end{aligned}$$

wobei $\gamma>0$ eine Schranke für die zweiten Ableitungen von J und e bezeichnet. Für Δx^k aus (4.7) gilt $A_k\Delta x^k = -e(x^k)$. Also gilt für $\alpha\in[0,1]$

$$\Phi_1(x^k+\alpha\Delta x^k;\mu) - \Phi_1(x^k;\mu) \leq \alpha\left(\nabla J(x^k)\Delta x^k - \frac{1}{\mu}\left\|e(x^k)\right\|_1\right) + \alpha^2\gamma\left\|\Delta x^k\right\|^2.$$

Analog schließen wir

$$\Phi_1(x^k+\alpha\Delta x^k;\mu) - \Phi_1(x^k;\mu) \geq \alpha\left(\nabla J(x^k)\Delta x^k - \frac{1}{\mu}\left\|e(x^k)\right\|_1\right) - \alpha^2\gamma\left\|\Delta x^k\right\|^2.$$

Daher folgt

$$D(\Phi_1(x^k;\mu),\Delta x^k) = \nabla J(x^k)\Delta x^k - \frac{1}{\mu}\left\|e(x^k)\right\|_1. \tag{4.25}$$

Aus der ersten Blockzeile in (4.7) bekommen wir

$$D(\Phi_1(x^k;\mu),\Delta x^k) = -(\Delta x^k)^T\nabla_{xx}L(x^k,\lambda^k)\Delta x^k - (\Delta x^k)^T A_k^T\lambda^{k+1} - \frac{1}{\mu}\left\|e(x^k)\right\|_1,$$

und wegen der zweiten Blockzeile in (4.7) gilt

$$D(\Phi_1(x^k;\mu),\Delta x^k) = -(\Delta x^k)^T\nabla_{xx}L(x^k,\lambda^k)\Delta x^k + e(x^k)^T\lambda^{k+1} - \frac{1}{\mu}\left\|e(x^k)\right\|_1.$$

Aufgrund der Abschätzung

$$e(x^k)^T\lambda^{k+1} = \sum_{i=1}^m e_i(x^k)\lambda_i^{k+1} \leq \left\|\lambda^{k+1}\right\|_\infty \sum_{i=1}^m |e_i(x^k)| = \left\|\lambda^{k+1}\right\|_\infty \left\|e(x^k)\right\|_1$$

folgt (4.22). □

Eine weitere sehr effektive Strategie, um den Strafparameter μ zu wählen, wird sowohl im Kontext von Liniensuch- oder Trust-Region-Verfahren verwendet. Das Vorgehen basiert auf einem quadratischen Modell für Φ_1:

$$q_\mu(\Delta x) = J(x^k) + \nabla J(x^k)\Delta x + \frac{\sigma}{2}\Delta x^T\nabla_{xx}L(x^k,\lambda^k)\Delta x + \frac{1}{\mu}m(\Delta x), \tag{4.26}$$

wobei

$$m(\Delta x) = \left\|e(x^k)+A_k\Delta x\right\|_1$$

gilt und σ ein Parameter ist, den wir später definieren. Haben wir einen Schritt Δx^k berechnet, so wählen wir μ hinreichend klein, so dass

$$q_\mu(0) - q_\mu(\Delta x^k) \geq \frac{\varrho}{\mu}\big(m(0) - m(\Delta x^k)\big) \tag{4.27}$$

erfüllt ist mit einem $\varrho \in (0,1)$. Für Δx^k aus (4.7) gilt $A_k \Delta x^k = -e(x^k)$ und somit $m(\Delta x^k) = 0$. Daher gilt wegen (4.26)

$$\begin{aligned} q_\mu(0) - q_\mu(\Delta x^k) &= \frac{1}{\mu}\big(m(0) - m(\Delta x^k)\big) - \nabla J(x^k)\Delta x^k - \frac{\sigma}{2}(\Delta x^k)^T \nabla_{xx} L(x^k, \lambda^k)\Delta x^k \\ &= \frac{1}{\mu}\|e(x^k)\|_1 - \nabla J(x^k)\Delta x^k - \frac{\sigma}{2}(\Delta x^k)^T \nabla_{xx} L(x^k, \lambda^k)\Delta x^k \\ &= \left(\frac{1}{\mu} + \frac{\nabla J(x^k)\Delta x^k + \frac{\sigma}{2}(\Delta x^k)^T \nabla_{xx} L(x^k, \lambda^k)\Delta x^k}{(1-\varrho)\|e(x^k)\|_1}(\varrho - 1)\right)\|e(x^k)\|_1. \end{aligned}$$

Ist die Ungleichung

$$\frac{1}{\mu} \geq \frac{\nabla J(x^k)\Delta x^k + \frac{\sigma}{2}(\Delta x^k)^T \nabla_{xx} L(x^k, \lambda^k)\Delta x^k}{(1-\varrho)\|e(x^k)\|_1} \tag{4.28}$$

erfüllt, schliessen wir

$$q_\mu(0) - q_\mu(\Delta x^k) \geq \left(\frac{1}{\mu} + \frac{\varrho - 1}{\mu}\right)\|e(x^k)\|_1 = \frac{\varrho}{\mu}\big(m(0) - m(\Delta x^k)\big),$$

was (4.27) entspricht. Erfüllt der Parameter μ aus der vorangegangenen SQP-Iteration die Bedingung (4.28), so ändern wir μ nicht. Andernfalls wird μ verkleinert, so dass (4.28) gilt. Die Konstante σ ermöglicht es, den Fall zu behandeln, wenn die Hesse-Matrix $\nabla_{xx} L(x^k, \lambda^k)$ nicht positiv definit ist. Wir setzen daher

$$\sigma = \begin{cases} 1 & \text{falls } (\Delta x^k)^T \nabla_{xx} L(x^k, \lambda^k)\Delta x^k > 0, \\ 0 & \text{andernfalls.} \end{cases}$$

Erfüllt μ (4.28), so garantiert die Wahl für σ die Ungleichung

$$D(\Phi_1(x^k; \mu); \Delta x^k) \leq -\frac{\varrho}{\mu}\|e(x^k)\|_1. \tag{4.29}$$

Die Ungleichung (4.29) folgt aus

$$\begin{aligned} q_\mu(0) - q_\mu(\Delta x^k) &= -\nabla J(x^k)\Delta x^k - \frac{\sigma}{2}(\Delta x^k)^T \nabla_{xx} L(x^k, \lambda^k)\Delta x^k + \frac{1}{\mu}\|e(x^k)\|_1 \\ &= -D(\Phi_1(x^k; \mu), \Delta x^k) - \frac{\sigma}{2}(\Delta x^k)^T \nabla_{xx} L(x^k, \lambda^k)\Delta x^k \end{aligned}$$

und

$$D(\Phi_1(x^k; \mu), \Delta x^k) \leq q_\mu(\Delta x^k) - q_\mu(0) \leq \frac{\varrho}{\mu}\big(m(\Delta x^k) - m(0)\big) = -\frac{\varrho}{\mu}\|e(x^k)\|_1.$$

Wegen (4.29) ist Δx^k eine Abstiegsrichtung für Φ_1. Dies ist nicht immer erfüllt, wenn $\sigma = 1$ ist und $(\Delta x^k)^T \nabla_{xx} L(x^k, \lambda^k)\Delta x^k < 0$ gilt. Ein Vergleich von (4.24) und (4.28) zeigt, dass wir im Fall $\sigma > 0$ in der Strategie, die (4.26) verwendet, einen kleineren Strafparameter zulassen. Damit wird mehr Gewicht auf die Erfüllung der Nebenbedingungen gelegt. Das ist sinnvoll bei Schritten, die eine Reduktion der Nebenbedingungen, aber einen Anstieg im Zielfunktional bewirken. Diese Schritte werden dann durch die Meritfunktion eher akzeptiert.

Nun kommen wir zu Fletchers augmentierter Lagrange-Funktion:

$$\Phi_F(x;\mu) = J(x) + \lambda(x)^T e(x) + \frac{1}{2\mu}\,\|e(x)\|^2, \tag{4.30}$$

wobei $\mu > 0$ einen Penalty-Parameter bezeichnet und

$$\lambda(x) = -\big(A(x)A(x)^T\big)^{-1} A(x)\nabla J(x)^T \tag{4.31}$$

der Least-Squares Multiplikator ist. Offenbar ist Φ_F differenzierbar. Es folgt

$$\nabla\Phi_F(x^k;\mu) = \nabla J(x^k) + \left(A_k^T\lambda^k + \nabla\lambda(x^k)^T e(x^k) + \frac{1}{\mu}\,A_k^T e(x^k)\right)^T.$$

Löst Δx^k das System (4.7), so gilt

$$\nabla\Phi_F(x^k;\mu)\Delta x^k = \nabla J(x^k)\Delta x^k - (\lambda^k)^T e(x^k) + e(x^k)^T\nabla\lambda(x^k)\Delta x^k - \frac{1}{\mu}\,\|e(x^k)\|^2.$$

Wir schreiben $\Delta x^k = Z_k\Delta x_Z^k + Y_k\Delta x_Y^k$, wobei Z_k eine Basis des Nullraums von A_k ist und $Y_k = A_k^T$ gesetzt ist. Wegen (4.10a) erhalten wir

$$A_k^T\Delta x_Y^k = -A_k^T\big(A_kA_k^T\big)^{-1}e(x^k),$$

und wegen (4.31) gilt

$$\nabla J(x^k)A_k^T\Delta x_Y^k = -\nabla J(x^k)A_k^T\big(A_kA_k^T\big)^{-1}e(x^k) = (\lambda^k)^T e(x^k).$$

Damit folgt

$$\begin{aligned}\nabla\Phi_F(x^k;\mu)\Delta x^k &= \nabla J(x^k)Z_k\Delta x_Z^k + \nabla J(x^k)A_k^T\Delta x_Y^k - (\lambda^k)^T e(x^k)\\ &\quad + e(x^k)^T\nabla\lambda(x^k)\Delta x^k - \frac{1}{\mu}\,\|e(x^k)\|^2\\ &= \nabla J(x^k)Z_k\Delta x_Z^k + e(x^k)^T\nabla\lambda(x^k)\Delta x^k - \frac{1}{\mu}\,\|e(x^k)\|^2.\end{aligned} \tag{4.32}$$

Aus der ersten Blockzeile in (4.7) folgt mit $\Delta x^k = A_k^T\Delta x_Y^k + Z_k\Delta x_Z^k$

$$\begin{aligned}\nabla_{xx}L(x^k,\lambda^k)\Delta x^k &= \nabla_{xx}L(x^k,\lambda^k)Z_k\Delta x_Z^k + \nabla_{xx}L(x^k,\lambda^k)A_k^T\Delta x_Y^k\\ &= -\nabla J(x^k)^T - A_k^T\lambda^{k+1}\end{aligned}$$

und somit

$$\begin{aligned}(\Delta x^k)^T\nabla_{xx}L(x^k,\lambda^k)Z_k\Delta x_Z^k &= \big(\nabla_{xx}L(x^k,\lambda^k)\Delta x^k\big)^T Z_k\Delta x_Z^k\\ &= \big(-\nabla J(x^k)^T - A_k^T\lambda^{k+1}\big)^T Z_k\Delta x_Z^k\\ &= -\nabla J(x^k)Z_k\Delta x_Z^k\end{aligned}$$

wegen $A_kZ_k\Delta x_Z^k = 0$. Aus (4.32) erhalten wir daher wieder mit $\Delta x^k = A_k^T\Delta x_Y^k + Z_k\Delta x_Z^k$

$$\begin{aligned}\nabla\Phi_F(x^k;\mu)\Delta x^k = &-(\Delta x_Z^k)^T Z_k^T\nabla_{xx}L(x^k,\lambda^k)Z_k\Delta x_Z^k\\ &- (\Delta x_Y^k)^T A_k\nabla_{xx}L(x^k,\lambda^k)Z_k\Delta x_Z^k + e(x^k)^T\nabla\lambda(x^k)\Delta x^k\\ &- \frac{1}{\mu}\,\|e(x^k)\|^2.\end{aligned}$$

Damit ist Δx^k eine Abstiegsrichtung für die Abbildung Φ_F, wenn die reduzierte Hesse-Matrix $Z_k^T \nabla_{xx} L(x^k, \lambda^k) Z_k$ positiv definit ist und μ der Bedingung

$$\begin{aligned} \frac{1}{\mu} \quad > & \frac{-\frac{1}{2}(\Delta x^k)^T Z_k^T \nabla_{xx} L(x^k, \lambda^k) Z_k \Delta x^k + e(x^k)^T \nabla \lambda(x^k) \Delta x^k}{\|e(x^k)\|^2} \\ & + \frac{-(\Delta x_Y^k)^T A_k \nabla_{xx} L(x^k, \lambda^k) Z_k \Delta x_Z^k}{\|e(x^k)\|^2} + \delta \end{aligned} \tag{4.33}$$

genügt für ein $\delta > 0$. Im Falle von $e(x^k) = 0$ ist Δx^k eine Abstiegsrichtung für jedes $\mu > 0$, siehe (4.10a).

Satz 4.7. *Angenommen, x^k ist kein stationärer Punkt des Problem* (**P**) *und die Hesse-Matrix $Z_k^T \nabla_{xx} L(x^k, \lambda^k) Z_k$ ist positiv definit. Dann ist die Suchrichtung Δx^k aus* (4.7) *eine Abstiegsrichtung für Φ_1, wenn* (4.22) *gilt, und für Φ_F, wenn* (4.33) *erfüllt ist*

5. Ein SQP-Verfahren mit Liniensuche

Es gibt viele Varianten für SQP-Verfahren. Sie können sich zum Beispiel durch folgende Aspekte unterscheiden:

- Approximation der Hesse-Matrix,
- Wahl der Merit-Funktion,
- Berechnung des Schrittweiten-Parameters,
- Update für den Multiplikator λ^k,
- unterschiedliche Formeln für die Quasi-Newton Approximation,
- andere Parameter,
- Globalisierung mit Trust-Region oder Liniensuch-Strategien,
- Berechnung des SQP-Schrittes.

Wir wollen nun ein Beispiel für ein SQP-Verfahren angeben.

Algorithmus 4.8 (SQP-Verfahren für nichtlineare Optimierung).

1) *Wähle $\eta \in (0, 1/2)$, $\tau \in (0, 1)$, $(x^0, \lambda^0) \in \mathbb{R}^n \times \mathbb{R}^m$, $k_{\max} \in \mathbb{N}$;*
2) *Wähle positiv definite und symmetrische Startmatrix $B_0 \in \mathbb{R}^{n \times n}$ als Approximation der reduzierten Hesse-Matrix; berechne J_0, ∇J_0, e_0 sowie A_0;*
3) **for** $k = 0$ **to** $k_{\max}$
   - **if** *Konvergenz, breche ab;*
   - *Berechne den SQP-Schritt Δx^k;*
   - *Bestimme $\mu_k > 0$, so dass Δx^k eine Abstiegsrichtung für die Merit-Funktion $\Phi(\cdot\,; \mu)$ ist;*
   - *Setze $\alpha^{(0)} = 1$ und $i = 0$;*
   - **while** *($\Phi(x^k + \alpha^{(i)} \Delta x^k; \mu_k) > \Phi(x^k; \mu_k) + \eta \alpha^{(i)} D(\Phi(x^k; \mu); \Delta x^k)$*
     Setze $\alpha^{(i+1)} = \tau_\alpha \alpha^{(i)}$ mit $\tau_\alpha \in (0, \tau)$ und $i = i + 1$;
   - **end**
   - *Setze $\alpha_k = \alpha^{(i)}$ und $x^{k+1} = x^k + \alpha_k \Delta x^k$;*
   - *Berechne J_{k+1}, ∇J_{k+1}, e_{k+1} sowie A_{k+1};*
   - *Bestimme Least-Squares Multiplikator λ^{k+1}:*

$$\lambda^{k+1} = -\left(A_{k+1} A_{k+1}^T\right)^{-1} A_{k+1} \nabla J_{k+1}^T;$$

   - *Setze $s^k = \alpha_k \Delta x^k$, $y^k = Z_k^T (\nabla_x L(x^{k+1}, \lambda^{k+1})^T - \nabla_x L(x^k, \lambda^{k+1})^T)$;*
   - *Berechne B_{k+1} aus B_k mittels BFGS-Update;*

end (for)

Wir können bei der Lösung der quadratischen Teilprobleme durch die Verwendung von Warm-up Strategien deutlich effizienter werden. Ferner kann ein Limited-Memory BFGS-Verfahren [**9**] verwendet werden, insbesondere im Kontext von hochdimensionalen Optimierungsaufgaben. Wird die Hesse-Matrix $\nabla_{xx} L_k$ verwendet, so gehen wir davon aus, dass eine Modifikation der Hesse-Matrix durchgeführt wird, sofern die Matrix nicht positiv definit auf dem Unterraum Kern A_k ist.

6. Trust-Region SQP-Verfahren

Trust-Region SQP-Verfahren besitzen mehrere Vorteile. Auch wenn die Hesse-Matrix $\nabla_{xx} L_k$ nicht positiv definit auf dem Unterraum Kern A_k oder gar singulär ist, kann eine Strategie verfolgt werden, die globale Konvergenz garantiert. Die einfachste Weise, einen Trust-Region-Algorithmus zu entwerfen, besteht darin, zum quadratischen Teilproblem (4.9) eine Trust-Region-Restriktion dazuzufügen:

$$\min_{\Delta x^k \in \mathbb{R}^n} J_k + \nabla J_k \Delta x^k + \frac{1}{2}\left(\Delta x^k\right)^T \nabla_{xx} L(x^k, \lambda^k) \Delta x^k \tag{4.34a}$$

$$\text{u.d.N.} \quad \nabla e(x^k)\Delta x^k + e(x^k) = 0 \text{ in } \mathbb{R}^m, \tag{4.34b}$$

$$\nabla g(x^k)\Delta x^k + g(x^k) \leq 0 \text{ in } \mathbb{R}^p, \tag{4.34c}$$

$$\|\Delta x^k\|_2 \leq \Delta_k. \tag{4.34d}$$

Selbst wenn die Bedingungen (4.34b) und (4.34c) kompatibel sind, kann es sein, dass das Problem (4.34) keine Lösung besitzt, da aufgrund von der Restriktion (4.34) die Menge der zulässigen Lösungen leer ist. Um den Konflikt der Bedingungen (4.34b)-(4.34d) zu lösen, kann nicht einfach der Trust-Region Radius Δ_k vergrößert werden, denn sonst kann keine globale Konvergenz garantiert werden. Daher wird die folgende Strategie verwendet: Es besteht keine Notwendigkeit, die Gleichungsnebenbedingung (4.34b) exakt zu erfüllen, sondern im Laufe der SQP-Iterationen die Gleichungsnebenbedingung zunehmend besser zu garantieren. Es gibt hier drei unterschiedliche Strategien: Relaxierungsmethoden, Penalty-Verfahren oder Filter-Algorithmen.

Wir wollen hier kurz auf Relaxierungsmethoden eingehen. Dabei beschränken wir uns auf ($\mathbf{P}_{Gl}$), das heißt, auf Optimierungsprobleme mit Gleichungsrestriktionen. Erweiterungen auf Probleme mit Ungleichungs-Nebenbedingungen basieren auf Innere-Punkte-Verfahren. Sei die Iterierte x^k gegeben. Wir berechnen im SQP-Schritt die Lösung des Teilproblems

$$\min_{\Delta x^k \in \mathbb{R}^n} J_k + \nabla J_k \Delta x^k + \frac{1}{2}\left(\Delta x^k\right)^T \nabla_{xx} L(x^k, \lambda^k) \Delta x^k \tag{4.35a}$$

$$\text{u.d.N.} \quad \nabla e(x^k)\Delta x^k + e(x^k) = r_k \text{ in } \mathbb{R}^m, \tag{4.35b}$$

$$\|\Delta x^k\|_2 \leq \Delta_k. \tag{4.35c}$$

Die Wahl des Vektors r_k erfordert eine gute Strategie, da die Effizienz des Verfahrens wesentlich davon abhängt. Wir wählen r_k als kleinsten Vektor, so dass (4.35b) und (4.35c) erfüllt sind für einen leicht reduzierten Trust-Region Radius Δ_k. Daher lösen wir zunächst das Teilproblem

$$\min_{v \in \mathbb{R}^n} \|A_k v + e_k\|_2^2 = v^T A_k^T A_k v + 2 e_k^T A_k v + \|e_k\|_2^2 \text{ u.d.N. } \|v\|_2 \leq 0.8\,\Delta_k. \tag{4.36}$$

Sei v_k die Lösung von (4.36). Dann definieren wir

$$r_k = A_k v_k + e_k. \tag{4.37}$$

Dann lösen wir (4.35), bestimmen den Schritt Δx^k und setzen $x^{k+1} = x^k + \Delta x^k$. Nun kann λ^{k+1} mit Hilfe der Least-Squares-Formel berechnet werden. Wir bemerken, dass nun (4.35b) und (4.35c) konsistent sind, da sie für $\Delta x^k = v_k$ erfüllt sind.

Auf den ersten Blick erscheint die Vorgangsweise nicht effizient zu sein, da in der Regel die Probleme (4.35) und (4.36) nicht einfach zu lösen sind, insbesondere wenn $\nabla_{xx} L_k$ indefinit ist. Es sind aber sehr effiziente Verfahren entwickelt worden, die beiden Optimierungsaufgaben inexakt zu lösen.

Zur Lösung von (4.36) verwenden wir ein Dogleg-Verfahren [**9**]. Dazu benötigen wir den Cauchy-Punkt v^{CP}, welcher der Minimierer des Zielfunktionals in (4.36) entlang der Richtung $-A_k^T e_k$ ist, und — im Falle der Existenz — den Newton-Punkt v^{NP}, den unrestringierten Minimierer von (4.36). Da die Hesse-Matrix von (4.36) singulär ist, gibt es unendlich viele Möglichkeiten zur Wahl von v^{NP}, die alle die Gleichung $A_k v^{\mathrm{NP}} + e_k = 0$ erfüllen. Wir wählen die Lösung mit der minimalen euklidischen Norm, indem wir die Singulärwertzerlegung zur Lösung verwenden. Nun sei v_k der Minimierer von (4.36) entlang des Pfads, der durch v^{CP} und v^{NP} definiert wird:

$$\tilde{v}(\tau) = \begin{cases} \tau v^{\mathrm{CP}} & \text{für } 0 \le \tau \le 1, \\ v^{\mathrm{CP}} + (\tau - 1)\left(v^{\mathrm{NP}} - v^{\mathrm{CP}}\right) & \text{für } 1 \le \tau \le 2. \end{cases}$$

Eine bevorzugte Technik zur Berechnung einer approximativen Lösung Δx^k für (4.35) ist das projezierte konjugierte Gradienten-Verfahren. Wir wenden dieses Verfahren zur Lösung des Problems (4.35a)-(4.35b) an, wobei wir darauf achten, dass die Trust-Region-Bedingung (4.35c) erfüllt ist, und brechen ab, sobald der Trust-Region-Rand oder eine Richtung negativer Krümmung erreicht wird.

Eine Meritfunktion für die präsentierte Vorgangsweise ist zum Beispiel die nichtglatte ℓ_2-Funktion

$$\Phi_2(x;\mu) = J(x) + \frac{1}{\mu}\left\|e(x)\right\|_2, \quad \mu > 0.$$

Für Φ_2 verwenden wir das quadratische Modell

$$q_\mu(\Delta x) = J(x^k) + \nabla J(x^k)\Delta x + \frac{1}{2}\,\Delta x^T \nabla_{xx} L(x^k, \lambda^k)\Delta x + \frac{1}{\mu}\, m(\Delta x)$$

wobei wir

$$m(\delta x) = \left\|e_k + A_k \Delta x\right\|_2$$

setzen. Wir wählen den Strafparameter μ hinreichend klein, so dass die Ungleichung

$$q_\mu(0) - q_\mu(\Delta x^k) \ge \frac{\varrho}{\mu}\left(m(0) - m(\Delta x^k)\right), \quad \varrho \in (0,1), \tag{4.38}$$

erfüllt ist. Die Entscheidung, ob ein Schritt Δx^k akzeptiert wird, wird anhand des Quotienten

$$\varrho_k = \frac{\mathrm{ared}_k}{\mathrm{pred}_k} = \frac{\Phi_2(x^k;\mu) - \Phi_2(x^k + \Delta x^k;\mu)}{q_\mu(0) - q_\mu(\Delta x^k)}$$

durchgeführt.

Algorithmus 4.9 (Byrd-Omojokun Trust-Region SQP-Verfahren).

1) *Wähle Konstanten* $k_{\max} \in \mathbb{N}$, $\varepsilon > 0$ *und* $\eta, \gamma \in (0,1)$*;*
2) *Wähle einen Startwert* x^0 *und einen Trust-Radius Radius* Δ^0*;*
3) **for** $k = 0$ **to** $k_{\max}$
   Berechne J_k, e_k, ∇J_k *und* A_k*;*

Bestimme den Least-Squares Multiplikator $\hat{\lambda} = (A_k A_k^T)^T A_k \nabla J_k$*;*
if $\|\nabla J_k - A_k^T \lambda_k\|_\infty < \varepsilon$ **and** $\|e_k\|_\infty < \varepsilon$
    Abbruch mit der approximativen Lösung x^k*;*
end (if)
Löse das Teilproblem (4.36) *zur Berechnung von* v_k *und setze* r_k *gemäß* (4.37)*;*
Berechne $\nabla_{xx} L(x^k, \lambda^k)$ *oder eine Quasi-Newton Approximation der Hesse-Matrix;*
Löse das Problem (4.35) *unter Verwendung eines projezierten konjugierten Gradienten-Verfahren;*
Bestimme einen Strafparameter μ_k*, der* (4.38) *erfüllt;*
Berechne den Quotienten $\varrho_k = \mathrm{ared}_k / \mathrm{pred}_k$*;*
if $\varrho_k > \eta$
    Setze $x^{k+1} = x^k + \Delta x^k$ *und wähle einen neuen Trust-Region Radius mit* $\Delta_{k+1} \geq \Delta_k$*;*
else
    Setze $x^{k+1} = x^k$ *und wähle einen neuen Trust-Region Radius mit* $\Delta_{k+1} \leq \gamma \|\Delta x^k\|_2$*;*
end (if)
end (for)

KAPITEL 5

Optimal control in finite dimension

Some basic concepts in optimal control theory can be illustrated very well in the context of finite-dimensional optimization. In particular, we do not have to deal with partial differential equations and several aspects from functional analysis.

1. Finite-dimensional optimal control problem

Let us consider the minimization problem

$$\min J(y,u) \quad \text{subject to (s.t.)} \quad Ay = Bu \quad \text{and} \quad u \in U_{ad} \tag{5.1}$$

where $J : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ denotes the cost functional, $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$ and $\emptyset \neq U_{ad} \subset \mathbb{R}^m$ is the set of admissible controls.

We look for vectors $y \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$ which solve (5.1).

Beispiel 5.1. Often the cost functional is quadratic, e.g.,

$$J(y,u) = |y - y_d|^2 + \lambda |u|^2,$$

where $|\cdot|$ stands for the Euclidean norm and $y_d \in \mathbb{R}^n$, $\lambda \geq 0$ hold. ◊

Problem (5.1) has the form of an optimization problem. Now we assume that A is an invertible matrix. Then we have

$$y = A^{-1}Bu. \tag{5.2}$$

In this case there exists a unique vector $y \in \mathbb{R}^n$ for any $u \in \mathbb{R}^m$. Hence, y is a dependent variable. We call u the control and y the state. In this way, (5.1) becomes a finite-dimensional optimal control problem.

We define the matrix $S \in \mathbb{R}^{m\times n}$ by $S = A^{-1}B$. Then, S is the solution matrix of our control system: $y = Su$. Utilizing the matrix S we introduce the so-called reduced cost functional

$$\hat{J}(u) = J(Su, u).$$

This leads to the reduced problem

$$\min \hat{J}(u) \quad \text{s.t.} \quad u \in U_{ad}. \tag{5.3}$$

In (5.3) the state variable is eliminated.

2. Existence of optimal controls

Definition 5.2. *The vector $u^* \in U_{ad}$ is called an* optimal control *for* (5.1) *provided*

$$\hat{J}(u^*) \leq \hat{J}(u) \qquad \textit{for all } u \in U_{ad}.$$

The vector $y^* = Su^*$ is the associated optimal state.

SATZ 5.3. *Suppose that J is continuous on $\mathbb{R}^n \times U_{ad}$, that U_{ad} is nonempty, bounded, closed and that A is invertible. Then, there exists at least one optimal control for* (5.1).

PROOF. Since the cost functional J is continuous on $\mathbb{R}^n \times U_{ad}$, the reduced cost $\hat{J}$ is continuous on U_{ad}. Furthermore, $U_{ad} \subset \mathbb{R}^m$ is bounded and closed. This implies that U_{ad} is compact. Due to the theorem of Weierstrass $\hat{J}$ has a minimum $u^* \in U_{ad} \neq \emptyset$, i.e., $\hat{J}(u^*) = \min_{u \in U_{ad}} \hat{J}(u)$. □

In the context of partial differential equations the proof for the existence of optimal controls is more complicated. The reason for this fact is that bounded and closed sets in infinite-dimensional function spaces need not to be compact.

3. First-order necessary optimality conditions

To compute solutions to optimal control problems we make use of optimality conditions. For that purpose we study first-order conditions for optimality.

We use the following notation for a function $\hat{J} : \mathbb{R}^m \to \mathbb{R}$:

$$
\begin{aligned}
D_i = \frac{\partial}{\partial x_i}, \quad D_x = \frac{\partial}{\partial x}, \quad D_{xx} = \frac{\partial^2}{\partial x^2} \qquad & \text{(partial derivatives)}, \\
\hat{J}'(x) = (D_1\hat{J}(x), \dots, D_m\hat{J}(x)) \in \mathbb{R}^{1\times m} \qquad & \text{(derivative)}, \\
\nabla\hat{J}(x) = \hat{J}'(x)^\top \qquad & \text{(gradient)}.
\end{aligned}
$$

For functions $J : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ we denote by $D_yJ(y,u) \in \mathbb{R}^{1\times n}$ the derivative with respect to $y \in \mathbb{R}^n$, i.e., $D_yJ(y,u) = (D_{y_1}J(y,u), \dots, D_{y_n}J(y,u))$. The vector $\nabla_yJ(y,u) = D_yJ(y,u)^\top \in \mathbb{R}^{n\times 1}$ is the gradient of J with respect to y. Analogously, $D_uJ(y,u)$ and $\nabla_uJ(y,u)$ are defined.

The Euclidean inner product is denoted by

$$
\langle u, v\rangle_{\mathbb{R}^m} = u \cdot v = \sum_{i=1}^{m} u_i v_i \quad \text{for} \quad u = (u_1, \dots, u_m)^\top,\ v = (v_1, \dots, v_m)^\top.
$$

For the directional derivative in direction $h \in \mathbb{R}^m$ we have

$$
\hat{J}'(u)h = \langle \nabla\hat{J}(u), h\rangle_{\mathbb{R}^m} = \nabla\hat{J}(u) \cdot h.
$$

Throughout we assume that all partial derivatives of J exist and are continuous. From the chain rule it follows that $\hat{J}(u) = J(Su, u)$ is continuously differentiable.

BEISPIEL 5.4. Let us consider the cost functional

$$
\hat{J}(u) = \frac{1}{2}|Su - y_d|^2 + \frac{\lambda}{2}|u|^2,
$$

see Example 5.1. We obtain

$$
\begin{aligned}
\nabla\hat{J}(u) &= S^\top(Su - y_d) + \lambda u, \\
\hat{J}'(u) &= (S^\top(Su - y_d) + \lambda u)^\top, \\
\hat{J}'(u)h &= \langle S^\top(Su - y_d) + \lambda u, h\rangle_{\mathbb{R}^m}
\end{aligned}
$$

at $u \in \mathbb{R}^m$ and for $h \in \mathbb{R}^m$. ◊

SATZ 5.5. *Suppose that u^* is an optimal control for* (5.1) *and U_{ad} convex. Then the variational inequality*

$$\hat{J}'(u^*)(u-u^*) \geq 0 \qquad \text{for all } u \in U_{ad} \tag{5.4}$$

holds.

It follows from Theorem 5.5 that at u^* the cost functional $\hat{J}$ can not decrease in any feasible direction. The proof follows from a more general result (see [**10**, pag. 63]).
From the chain rule we derive

$$\begin{aligned} \hat{J}'(u^*)h &= D_y J(Su^*,u^*)Sh + D_u J(Su^*,u^*)h \\ &= \langle \nabla_y J(y^*,u^*), A^{-1}Bh\rangle_{\mathbb{R}^n} + \langle \nabla_u J(y^*,u^*), h\rangle_{\mathbb{R}^m} \\ &= \langle B^\top A^{-\top}\nabla_y J(y^*,u^*) + \nabla_u J(y^*,u^*), h\rangle_{\mathbb{R}^m}, \end{aligned} \tag{5.5}$$

where $(A^\top)^{-1} = (A^{-1})^\top := A^{-\top}$ holds. Thus, we derive from (5.4)

$$\langle B^\top A^{-\top}\nabla_y J(y^*,u^*) + \nabla_u J(y^*,u^*), u-u^*\rangle_{\mathbb{R}^m} \geq 0 \tag{5.6}$$

for all $u \in U_{ad}$. In the following subsection we will introduce the so-called adjoint or dual variable. Then, we can express (5.6) in a simpler way.

1.4. Adjoint variable and reduced gradient. In a numerical realization the computation of A^{-1} is avoided. The same holds for the matrix $A^{-\top}$. Thus, we replace the term $A^{-\top}\nabla_y J(y^*,u^*)$ by $p^* := -A^{-\top}\nabla_y J(y^*,u^*)$, which is equivalent with

$$A^\top p^* = -\nabla_y J(y^*,u^*). \tag{5.7}$$

DEFINITION 5.6. *Equation* (5.7) *is called the* adjoint *or* dual equation. *Its solution p^* is the* adjoint *or* dual variable *associated with (y^*,u^*).*

BEISPIEL 5.7. For the quadratic cost functional $J(y,u) = \frac{1}{2}|y-y_d|^2 + \frac{1}{2}\lambda|u|^2$ with $y, y_d \in \mathbb{R}^m$ and $\lambda \geq 0$ we derive the adjoint equation

$$A^\top p^* = y_d - y^*.$$

Here we have used $\nabla_y J(y,u) = y - y_d$. ◊

The introduction of the dual variable yields two advantages:

1) We obtain an expression for (5.6) without the matrix $A^{-\top}$.
2) The expression (5.6) can be written in a more readable form.

Utilizing $y^* = Su^*$ in (5.5) we find that

$$\nabla\hat{J}(u^*) = -B^\top p^* + \nabla_u J(y^*,u^*).$$

The vector $\nabla\hat{J}(u^*)$ is called the *reduced gradient*. The directional derivative of the reduced cost functional $\hat{J}$ at an arbitrary $u \in U_{ad}$ in direction h is given by

$$\hat{J}'(u)h = \langle -B^\top p + \nabla_u J(y,u), h\rangle_{\mathbb{R}^m},$$

where $y = Su$ and $p = -A^\top\nabla_y J(y,u)$ hold. From Theorem 5.5 and (5.6) we derive directly the following theorem.

SATZ 5.8. *Suppose that A is invertible, u^* is an optimal control for* (5.1) *and $y^* = Su^*$ the associated optimal state. Then, there exists a unique dual variable p^* satisfying* (5.7). *Moreover, the variational inequality*

$$\langle -B^\top p^* + \nabla_u J(y^*, u^*), u - u^* \rangle_{\mathbb{R}^m} \geq 0 \quad \text{for all } u \in U_{ad} \tag{5.8}$$

holds true.

We have derived an optimality system for the unknown variables y^*, u^* and p^*:

$$\begin{aligned} &Ay^* = Bu^*, \qquad u^* \in U_{ad} \\ &A^\top p^* = -\nabla_y J(y^*, u^*) \\ &\langle -B^\top p^* + \nabla_u J(y^*, u^*), v - u^* \rangle_{\mathbb{R}^m} \geq 0 \quad \text{for all } v \in U_{ad}. \end{aligned} \tag{5.9}$$

Every solution (y^*, u^*) to (5.1) must satisfy, together with the dual variable p^*, the necessary conditions (5.9).

If $U_{ad} = \mathbb{R}^m$ holds, then the term $u - u^*$ can attain any value $h \in \mathbb{R}^m$. Therefore, the variational inequality (5.8) implies the equation

$$-B^\top p^* + \nabla_u J(y^*, u^*) = 0.$$

BEISPIEL 5.9. We consider the cost functional

$$J(y, u) = \frac{1}{2}|Cy - y_d|^2 + \frac{\lambda}{2}|u|^2$$

with $C \in \mathbb{R}^{n \times n}$, y, $y_d \in \mathbb{R}^n$, $\lambda \geq 0$ and $u \in \mathbb{R}^m$. Then,

$$\nabla_y J(y, u) = C^\top (Cy - y_d), \qquad \nabla_u J(y, u) = \lambda u.$$

Thus, we obtain the optimality system

$$\begin{aligned} &Ay^* = Bu^*, \quad u^* \in U_{ad} \\ &A^\top p^* = C^\top (y_d - Cy^*) \\ &\langle -B^\top p^* + \lambda u^*, v - u^* \rangle_{\mathbb{R}^m} \geq 0 \quad \text{for all } v \in U_{ad} \end{aligned}$$

If $U_{ad} = \mathbb{R}^m$ holds, we find $-B^\top p^* + \lambda u^* = 0$. For $\lambda > 0$ we have

$$u^* = \frac{1}{\lambda} B^\top p^*. \tag{5.10}$$

Inserting (5.10) into the state equation, we obtain a linear system in the state and dual variables:

$$\begin{aligned} Ay^* &= \frac{1}{\lambda} BB^\top p^* \\ A^\top p^* &= C^\top (y_d - Cy^*). \end{aligned}$$

If (y^*, p^*) is computed, u^* is given by (5.10). ◊

1.5. The Lagrange function. The optimality condition can be expressed by utilizing the Lagrange function.

DEFINITION 5.10. *The function $\mathcal{L} : \mathbb{R}^{2n+m} \to \mathbb{R}$ defined by*

$$\mathcal{L}(y, u, p) = J(y, u) + \langle Ay - Bu, p \rangle_{\mathbb{R}^n}, \qquad (y, u, p) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n,$$

is called the Lagrange function *for* (5.1).

It follows that the second and third conditions of (5.9) can be expressed as

$$\nabla_y \mathcal{L}(y^*, u^*, p^*) = 0$$
$$\langle \nabla_u \mathcal{L}(y^*, u^*, p^*), u - u^* \rangle_{\mathbb{R}^m} \geq 0 \quad \text{for all } u \in U_{ad}.$$

BEMERKUNG 5.11. The adjoint equation (5.7) is equivalent to $\nabla_y \mathcal{L}(y^*, u^*, p^*) = 0$. Thus, (5.7) can be derived from the derivative of the Lagrange functional with respect to the state variable y. Analogously, the variational inequality follows from the gradient $\nabla_u \mathcal{L}(y^*, u^*, p^*)$. ◊

It follows from Remark 5.11 that (y^*, u^*) satisfies the necessary optimality conditions of the minimization problem

$$\min_{y,u} \mathcal{L}(y, u, p^*) \quad \text{s.t. } u \in U_{ad},\ y \in \mathbb{R}^n. \tag{5.11}$$

Notice that (5.11) has no equality constraints (in contrast to (5.1)). In most applications p^* is not known a-priori. Thus, (y^*, u^*) can not be computed from (5.11).

1.6. Discussion of the variational inequality. In many applications the set of admissible controls has the form

$$U_{ad} = \{u \in \mathbb{R}^m \mid u_a \leq u \leq u_b\}, \tag{5.12}$$

where $u_a \leq u_b$ are given vectors in $\mathbb{R}^m$ and "$\leq$" means less or equal in each component: $u_{a,i} \leq u_i \leq u_{b,i}$ for $i = 1, \ldots, m$. From (5.8) it follows that

$$\langle -B^\top p^* + \nabla_u J(y^*, u^*), u^* \rangle_{\mathbb{R}^m} \leq \langle -B^\top p^* + \nabla_u J(y^*, u^*), u \rangle_{\mathbb{R}^m}$$

for all $u \in U_{ad}$. This implies that u^* solves the minimization problem

$$\min_{u \in U_{ad}} \langle -B^\top p^* + \nabla_u J(y^*, u^*), u \rangle_{\mathbb{R}^m} = \min_{u \in U_{ad}} \sum_{i=1}^m (-B^\top p^* + \nabla_u J(y^*, u^*))_i u_i.$$

If U_{ad} is of the form (5.12), then the minimization of a component u_i is independent of u_j, $i \neq j$:

$$(-B^\top p^* + \nabla_u J(y^*, u^*))_i u_i^* = \min_{u_{a,i} \leq u_i \leq u_{b,i}} (-B^\top p^* + \nabla_u J(y^*, u^*))_i u_i,$$

$1 \leq i \leq m$. Thus,

$$u_i^* = \begin{cases} u_{b,i} & \text{if } (-B^\top p^* + \nabla_u J(y^*, u^*))_i < 0 \\ u_{a,i} & \text{if } (-B^\top p^* + \nabla_u J(y^*, u^*))_i > 0. \end{cases} \tag{5.13}$$

If $(-B^\top p^* + \nabla_u J(y^*, u^*))_i = 0$ holds, we have no information from the variational inequality. In many cases we can use the equation $(-B^\top p^* + \nabla_u J(y^*, u^*))_i = 0$ to obtain an explicit equation for one of the components of u^*.

1.7. The Karush–Kuhn–Tucker system. Define the vectors

$$\begin{aligned} \mu_a &:= (-B^\top p^* + \nabla_u J(y^*, u^*))_+ \\ \mu_b &:= (-B^\top p^* + \nabla_u J(y^*, u^*))_-. \end{aligned} \tag{5.14}$$

where $\mu_{a,i} = (-B^\top p^* + \nabla_u J(y^*, u^*))_i$ if the right-hand side is positive and $\mu_{a,i} = 0$ otherwise. Analogously, $\mu_{b,i} = |(-B^\top p^* + \nabla_u J(y^*, u^*))_i|$ if the right-hand side is negative and $\mu_{b,i} = 0$ otherwise. Utilizing (5.13) we have

$$\mu_a \geq 0, \quad u_a - u^* \leq 0, \quad \langle u_a - u^*, \mu_a \rangle_{\mathbb{R}^m} = 0$$
$$\mu_b \geq 0, \quad u^* - u_b \leq 0, \quad \langle u^* - u_b, \mu_b \rangle_{\mathbb{R}^m} = 0$$

These conditions are called *complementarity conditions*. The inequalities are clear. We prove $\langle u_a - u^*, \mu_a \rangle_{\mathbb{R}^m} = 0$. Suppose that $u_{a,i} < u_i^*$ holds. Due to (5.13) we have $(-B^\top p^* + \nabla_u J(y^*, u^*))_i \leq 0$. Thus, $\mu_{a,i} = 0$ which gives $(u_{a,i} - u_i^*)\mu_{a,i} = 0$. Now we assume $\mu_{a,i} > 0$. Using (5.14) we derive $(-B^\top p^* + \nabla_u J(y^*, u^*))_i > 0$. It follows from (5.13) that $u_{a,i} = u_i^*$ holds. Again, we have $(u_{a,i} - u_i^*)\mu_{a,i} = 0$. Summation over $i = 1, \ldots, m$ yields $\langle u_a - u^*, \mu_a \rangle_{\mathbb{R}^m} = 0$.
Notice that

$$\mu_a - \mu_b = -B^\top p^* + \nabla_u J(y^*, u^*).$$

Hence,

$$\nabla_u J(y^*, u^*) - B^\top p^* + \mu_b - \mu_a = 0. \tag{5.15}$$

Let us consider an augmented Lagrange functional

$$\tilde{\mathcal{L}}(y, u, p, \mu_a, \mu_b) = J(y, u) + \langle Ay - Bu, p \rangle_{\mathbb{R}^n} + \langle u_a - u, \mu_a \rangle_{\mathbb{R}^m} + \langle u - u_b, \mu_b \rangle_{\mathbb{R}^m}$$

Then, (5.15) can be written as

$$\nabla_u \tilde{\mathcal{L}}(y^*, u^*, p^*, \mu_a, \mu_b) = 0.$$

Moreover, the adjoint equation is equivalent with

$$\nabla_y \tilde{\mathcal{L}}(y^*, u^*, p^*, \mu_a, \mu_b) = 0.$$

Here, we have used that $\nabla_y \mathcal{L} = \nabla_y \tilde{\mathcal{L}}$. The vectors μ_a and μ_b are the Lagrange multipliers for the inequality constraints $u_a - u^* \leq 0$ and $u^* - u_b \leq 0$.

Satz 5.12. *Suppose that u^* is an optimal control for (5.1), A is invertible and U_{ad} has the form (5.12). Then, there exist Lagrange multipliers $p^* \in \mathbb{R}^n$ and $\mu_a, \mu_b \in \mathbb{R}^m$ satisfying*

$$\begin{aligned}
&\nabla_y \tilde{\mathcal{L}}(y^*, u^*, p^*, \mu_a, \mu_b) = 0 \\
&\nabla_u \tilde{\mathcal{L}}(y^*, u^*, p^*, \mu_a, \mu_b) = 0 \\
&\mu_a \geq 0, \quad \mu_b \geq 0 \\
&\langle u_a - u^*, \mu_a \rangle_{\mathbb{R}^m} = \langle u^* - u_b, \mu_b \rangle_{\mathbb{R}^m} = 0 \\
&Ay^* = Bu^*, \quad u_a \leq u \leq u_b.
\end{aligned} \tag{5.16}$$

The optimality system (5.16) is called the *Karush-Kuhn-Tucker (KKT) system.*

KAPITEL 6

The linear-quadratic regulator problem

In this section we introduce the optimal state-feedback and the linear-quadratic regulator (LQR) problem. Utilizing dynamic programming necessary optimality conditions are derived. It turns out that for the LQR problem the state-feedback solution can be determined by solving a differential matrix Riccati equation. The presented theory is taken from the book [**2**].

1. The linear-quadratic regulator (LQR) problem

The goal is to find a state-feedback control law of the form

$$u(t) = -Kx(t) \quad \text{for } t \in [0,T]$$

with $u : [0,T] \to \mathbb{R}^{m_u}$, $x : [0,T] \to \mathbb{R}^{m_x}$, $K \in \mathbb{R}^{m_u \times m_x}$ so that u minimizes the quadratic cost functional

$$J(x,u) = \int_0^T x(t)^T Q x(t) + u(t)^T R u(t)\,\mathrm{d}t + x(T)^T M x(T), \tag{6.1a}$$

where the state x and the control u are related by the linear initial value problem

$$\dot{x}(t) = Ax(t) + Bu(t) \text{ for } t \in (0,T] \quad \text{and} \quad x(0) = x_0. \tag{6.1b}$$

In (6.1a) the matrices Q, $M \in \mathbb{R}^{m_x \times m_x}$ are symmetric, positive semi-definite, $R \in \mathbb{R}^{m_u \times m_u}$ is symmetric, positive definite and in (6.1b) we have $A \in \mathbb{R}^{m_x \times m_x}$, $B \in \mathbb{R}^{m_x \times m_u}$ and $x_0 \in \mathbb{R}^{m_x}$. The final time T is fixed, but the final state $x(T)$ is free. Thus, we aim to track the state to the state $\bar{x} = 0$ as good as possible. The terms $x(t)^T Q x(t)$ and $x(T)^T M x(T)$ are measures for the control accuracy and the term $u(t)^T R u(t)$ measures the control effort. Problem (6.1) is called the *linear-quadratic regulator problem* (*LQR problem*).

2. The Hamilton-Jacobi-Bellman equation

In this section we derive first-order necessary optimality conditions for the LQR problem. Since generalizing the problem to a non-linear problem does not cause more difficulties in the deviation, we consider the problem to find a state-control feedback control law

$$u(t) = \Phi(x(t), t), \quad t \in [0,T],$$

such that the cost-functional

$$J_t(x,u) = \int_t^T L(x(s), u(s), s)\,\mathrm{d}s + g(x(T)) \tag{6.2a}$$

is minimized subject to the non-linear system dynamics

$$\dot{x}(s) = F(x(s), u(s), s) \text{ for } s \in (0,T] \quad \text{and} \quad x(t) = x_t. \tag{6.2b}$$

We suppose that the functions $L : \mathbb{R}^{m_x} \times \mathbb{R}^{m_u} \times [0,T] \to [0,\infty)$ and $g : \mathbb{R}^{m_x} \to [0,\infty)$ satisfy

$$L(0,0,s) = 0 \text{ for } s \in [0,T] \quad \text{and} \quad g(0) = 0$$

Moreover, let $F : \mathbb{R}^{m_x} \times \mathbb{R}^{m_u} \times [0,T] \to \mathbb{R}^{m_x}$ be continuous and locally Lipschitz-continuous with respect to the variable x. Moreover, $x_t \in \mathbb{R}^{m_x}$ holds. To derive optimality conditions we use the so-called *Bellman principle* (or *dynamic programming principle*). The essential assumption is that the system can be characterized by its state $x(t)$ at the time $t \in [0,T]$ which completely summarizes the effect of all $u(s)$ for $0 \le s \le t$. The dynamic programming principle was first proposed by Bellman [**1**].

Satz 6.1 (Bellman principle). *Let $t \in [0,T]$. If $u^*(s)$ is optimal for $s \in [t,T]$ and x^* is the associated optimal state, starting at the state $x_t \in \mathbb{R}^{m_x}$, then $u^*(s)$ is also optimal over the subinterval $[t+\Delta t, T]$ for any $\Delta t \in [0, T-t]$ starting at $x_{t+\Delta t} = x^*(t+\Delta t)$.*

Proof. We show Theorem 6.1 by contradiction. Suppose that there exists a control u^{**} so that

$$\begin{aligned} \int_{t+\Delta t}^{T} L(x^{**}(s), u^{**}(s), s)\,\mathrm{d}s + g(x^{**}(T)) \\ < \int_{t+\Delta t}^{T} L(x^*(s), u^*(s), s)\,\mathrm{d}s + g(x^*(T)), \end{aligned} \tag{6.3}$$

where

$$\dot{x}^*(s) = F(x^*(s), u^*(s), s) \quad \text{and} \quad \dot{x}^{**}(s) = F(x^{**}(s), u^{**}(s), s)$$

hold for $s \in [t+\Delta t, T]$. We define the control

$$u(s) = \begin{cases} u^*(s) & \text{if } s \in [t, t+\Delta t], \\ u^{**}(s) & \text{if } s \in (t+\Delta t, T]. \end{cases} \tag{6.4}$$

By $x(s)$ we denote the state satisfying $\dot{x}(s) = F(x(s), u(s), s)$ for $s \in [t,T]$ and $x(t) = x_t$. Then we derive from (6.3) and (6.4) that

$$\begin{aligned} &\int_t^T L(x(s), u(s), s)\,\mathrm{d}s + g(x(T)) \\ &= \int_t^{t+\Delta t} L(x^*(s), u^*(s), s)\,\mathrm{d}s + \int_{t+\Delta t}^{T} L(x^{**}(s), u^{**}(s), s)\,\mathrm{d}s + g(x^{**}(T)) \\ &< \int_t^{t+\Delta t} L(x^*(s), u^*(s), s)\,\mathrm{d}s + \int_{t+\Delta t}^{T} L(x^*(s), u^*(s), s)\,\mathrm{d}s + g(x^*(T)) \\ &= \int_t^T L(x^*(s), u^*(s), s)\,\mathrm{d}s + g(x^*(T)). \end{aligned} \tag{6.5}$$

Recall that $u^*(s)$ is optimal for $s \in [t,T]$ by assumption. From (6.5) it follows that the control u given by (6.4) yields a smaller value of the cost functional. This is a contradiction. □

Next we derive the Hamilton-Jacobi-Bellman equation for (6.2). Let $V^* : \mathbb{R}^{m_x} \times [0,T] \to \mathbb{R}$ denote the minimal value function given by

$$\begin{aligned} &V^*(x_t,t) \\ &= \min_{u:[t,T]\to\mathbb{R}^{m_u}} \Big\{ J_t(x,u) \,\Big|\, \dot{x}(s) = F(x(s),u(s),s),\ s \in (t,T] \text{ and } x(t) = x_t \Big\} \end{aligned} \tag{6.6}$$

for $(x_t,t) \in \mathbb{R}^{m_x} \times [0,T]$, where

$$J_t(x,u) = \int_t^T L(x(s),u(s),s)\,\mathrm{d}s + g(x(T)).$$

From the linearity of the integral and (6.6) we conclude

$$\begin{aligned} &V^*(x_t,t) \\ &= \min_{u:[t,t+\Delta t]\to\mathbb{R}^{m_u}} \Big\{ \int_t^{t+\Delta t} L(x(s),u(s),s)\,\mathrm{d}s + V^*(x(t+\Delta t),t+\Delta t) \,\Big| \\ &\qquad\qquad \dot{x}(s) = F(x(s),u(s),s),\ s \in (t,t+\Delta t] \text{ and } x(t) = x_t \Big\} \end{aligned} \tag{6.7}$$

for $(x_t,t) \in \mathbb{R}^{m_x} \times [0,T-\Delta t]$, where we have used the Bellman principle. Thus, by using the Bellman principle the problem of finding an optimal control over the interval $[t,T]$ has been reduced to the problem of finding an optimal control over the interval $[t,t+\Delta t]$.

Now we replace the integral in (6.7) by $L(x(t),u(t),t)\Delta t$, perform a Taylor approximation for $V^*(x(t+\Delta t),t+\Delta t)$ about the point $(x_t,t) = (x(t),t)$ and approximate $x(t+\Delta t) - x(t)$ by $F(x(t),u(t),t)\Delta t$. Then we find

$$\begin{aligned} V^*(x_t,t) &= \min_{u_t\in\mathbb{R}^{m_u}} \Big\{ L(x_t,u_t,t)\Delta t + V^*(x_t,t) + \frac{\partial V^*}{\partial t}(x_t,t)\,\Delta t \\ &\qquad\qquad + \nabla V^*(x_t,t)^T F(x_t,u_t,t)\Delta t + o(\Delta t) \Big\} \\ &= V^*(x_t,t) + \frac{\partial V^*}{\partial t}(x_t,t)\Delta t \\ &\quad + \Delta t \min_{u_t\in\mathbb{R}^{m_u}} \Big\{ L(x_t,u_t,t) + \nabla V^*(x_t,t)^T F(x_t,u_t,t) + \frac{o(\Delta t)}{\Delta t} \Big\} \end{aligned}$$

for any $\Delta t > 0$. Thus,

$$-\frac{\partial V^*}{\partial t}(x_t,t) = \min_{u_t\in\mathbb{R}^{m_u}} \Big\{ L(x_t,u_t,t) + \nabla V^*(x_t,t)^T F(x_t,u_t,t) + \frac{o(\Delta t)}{\Delta t} \Big\}.$$

Taking the limit $\Delta t \to 0$ and using $V^*(x_t,T) = g(x_t)$ we obtain

$$-\frac{\partial V^*}{\partial t}(x_t,t) = \min_{u_t\in\mathbb{R}^{m_u}} \big\{ L(x_t,u_t,t) + \nabla V^*(x_t,t)^T F(x_t,u_t,t) \big\} \tag{6.8a}$$

for all $(x_t,t) \in \mathbb{R}^{m_x} \times [0,T)$ and

$$V^*(x_t,T) = g(x_t) \tag{6.8b}$$

for all $x_t \in \mathbb{R}^{m_x}$.

To solve (6.8) we proceed in two steps. First we compute a solution u_t to

$$u^*(t) = \operatorname*{argmin}_{u_t\in\mathbb{R}^{m_u}} \big\{ L(x_t,u_t,t) + \nabla V^*(x_t,t)^T F(x_t,u_t,t) \big\}$$

and set

$$\Psi(\nabla V^*(x_t,t),x_t,t) = u^*(t), \tag{6.9}$$

which gives us a control law. Then we insert (6.9) into (6.8a) and solve

$$\begin{aligned} -\frac{\partial V^*}{\partial t}(x_t,t) = {} & L(x_t,\Psi(\nabla V^*(x_t,t),x_t,t),t) \\ & + \nabla V^*(x_t,t)^T F(x_t,\Psi(\nabla V^*(x_t,t),x_t,t),t) \end{aligned}$$

for all $(x_t,t) \in \mathbb{R}^{m_x} \times [0,T)$. Finally, we can compute the gradient $\nabla V^*(x_t,t)$ and deduce the state-feedback law

$$u^*(t) = \Phi(x_t,t) = \Psi(\nabla V^*(x_t,t),x_t,t) \quad \text{for all } (x_t,t) \in \mathbb{R}^{m_x} \times [0,T).$$

Bemerkung 6.2. 1) In general, it is not possible to solve (6.8) analytically. However, for the LQR problem we can derive an explicit solution for the state-feedback law.

2) Note that the Hamilton-Jacobi-Bellman equation are only necessary optimality conditions. ♢

3. The state-feedback law for the LQR problem

For the LQR problem we have

$$L(x,u,t) = x^T Q x + u^T R u, \quad g(x) = x^T M x, \quad F(x,u,t) = Ax + Bu$$

for $(x,u,t) \in \mathbb{R}^{m_x} \times \mathbb{R}^{m_u} \times [0,T]$. For brevity, we focus on the situation, where the matrices A, B, Q, R are time-invariant. However, most of the presented theory also holds for the time-varying case.

First we minimize

$$x^T Q x + u^T R u + \nabla V^*(x,t)^T \big(Ax + Bu\big)$$

with respect to u. First-order necessary optimality conditions are given by

$$u^T R \tilde{u} + \tilde{u}^T R u + \nabla V^*(x,t)^T B \tilde{u} = 0 \quad \text{for all } \tilde{u} \in \mathbb{R}^{m_u}.$$

By assumption, R is symmetric and positive definite. Then we find

$$\big(2Ru + B^T \nabla V^*(x,t)\big)^T \tilde{u} = 0 \quad \text{for all } \tilde{u} \in \mathbb{R}^{m_u}$$

and

$$u^* = -\frac{1}{2} R^{-1} B^T \nabla V^*(x,t). \tag{6.10}$$

For the minimal value function V^* we make the quadratic ansatz

$$V^*(x,t) = x^T P(t) x, \quad P(t) \in \mathbb{R}^{m_x \times m_x} \text{ symmetric.} \tag{6.11}$$

Then, we have $\nabla V^*(x,t) = 2P(t)x$ so that

$$u^* = -R^{-1} B^T P(t) x.$$

Note that

$$\begin{aligned}\frac{\partial V^*}{\partial t}(x_t,t) &= x_t^T\dot{P}(t)x_t,\\ L(x_t,-R^{-1}B^TP(t)x_t,t) &= x_t^TQx_t + x_t^TP(t)BR^{-1}B^TP(t)x_t\\ &= x_t^T\big(Q+P(t)BR^{-1}B^TP(t)\big)x_t,\\ F(x_t,-R^{-1}B^TP(t)x_t,t) &= Ax_t - BR^{-1}B^TP(t)x_t = \big(A-BR^{-1}B^TP(t)\big)x_t,\\ \nabla V^*(x_t,t) &= 2P(t)x_t.\end{aligned}$$

Consequently,

$$\begin{aligned}-x_t^T\dot{P}(t)x_t &= -\frac{\partial V^*}{\partial t}(x_t,t)\\ &= x_t^T\big(Q+P(t)BR^{-1}B^TP(t)\big)x_t + \big(2P(t)x_t\big)^T\big(A-BR^{-1}B^TP(t)\big)x_t\end{aligned}$$

for all $x_t\in\mathbb{R}^{m_x}$, which yields

$$\begin{aligned}&-x_t^T\dot{P}(t)x_t\\ &\quad= x_t^T\big(Q+P(t)BR^{-1}B^TP(t)+2P(t)A-2P(t)BR^{-1}B^TP(t)\big)x_t\\ &\quad= x_t^T\big(2P(t)A+Q-P(t)BR^{-1}B^TP(t)\big)x_t\end{aligned}$$

for all $x_t\in\mathbb{R}^{m_x}$. From $P(t)=P(t)^T$ we deduce that

$$2x_t^TP(t)Ax_t = x_t^TP(t)Ax_t + x_t^TA^TP(t)x_t = x_t^T\big(A^TP(t)+P(t)A\big)x_t.$$

Using $V^*(x_t,T)=x_t^TP(T)x_t$ and (6.8b) we get

(6.12a)
$$-x_t^T\dot{P}(t)x_t = x_t^T\big(A^TP(t)+P(t)A+Q-P(t)BR^{-1}B^TP(t)\big)x_t,\quad t\in[0,T)$$

(6.12b)
$$x_t^TP(T)x_t = x_t^TMx_t.$$

Since (6.12) holds for all $x_t\in\mathbb{R}^{m_x}$ we obtain the following *matrix Riccati equation*

(6.13a) $$-\dot{P}(t) = A^TP(t)+P(t)A+Q-P(t)BR^{-1}B^TP(t),\quad t\in[0,T)$$

(6.13b) $$P(T)=M.$$

Finally, the optimal state-feedback is given by

$$u^*(t) = -K(t)x(t)\quad\text{and}\quad K(t)=R^{-1}B^TP(t).$$

Beispiel 6.3. Let us consider the problem

$$\min\int_0^T |x(t)|^2+|u(t)|^2\,\mathrm{d}t\quad\text{s.t.}\quad\dot{x}(t)=u(t)\text{ for }t\in(0,T].$$

Choosing $m_x=m_u=1$, $A=M=0$ and $B=Q=R=1$ the matrix Riccati equation has the form

$$-\dot{P}(t) = 1-P(t)^2\text{ for }t\in[0,T)\quad\text{and}\quad P(T)=0.$$

This scalar ordinary differential equation can be solved by separation of variables. Its solution is

$$P(t)=\frac{1-e^{-2(T-t)}}{1+e^{-2(T-t)}}$$

with the optimal control $u^*(t)=-P(t)x(t)$. ◊

Literaturverzeichnis

[1] R.E. Bellman. The theory of dynamic programming. *Proc. Nat. Acad. Sci.*, USA, 38:716-719, 1952.

[2] P. Dorato, C. Abdallah, and V. Cerone. *Linear-Quadratic Control*. Prentice Hall, Englewood Cliffs, New Yersey 07632, 1995.

[3] C. Geiger and C. Kanzow. *Theorie und Numerik restringierter Optimierungsaufgaben*. Springer-Verlag, Berlin, 2002.

[4] C. T. Kelley. *Iterative Methods for Linear and Nonlinear Equations*. Frontiers in Applied Mathematics. SIAM, Philadelphia, 1995.

[5] C. T. Kelley. *Iterative Methods for Optimization*. Frontiers in Applied Mathematics. SIAM, Philadelphia, 1999.

[6] H. Kuhn und A. Tucker. Nonlinear Programming. In J. Neyman, Editor, *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, University of California Press, Berkeley, pp. 481-492, 1951

[7] F.-S. Kupfer. An infinite-dimensional convergence theory for reduced SQP methods in Hilbert spaces. *SIAM Journal on Optimization*, 6:126-163, 1996.

[8] D. G. Luenberger. *Linear and Nonlinear Progamming*. Addison-Wesley Publishing Company, Reading, Massachusetts, 1984.

[9] J. Nocedal and S. J. Wright. *Numerical Optimization*. 2. Auflage, Springer Series in Operation Research. Springer-Verlag, New York, 2006.

[10] F. Tröltzsch. *Optimal Control of Partial Differential Equations: Theory, Methods and Applications*. American Mathematical Society, Graduate Studies in Mathematics, vol. 112, 2010.

[11] S. Volkwein. *Some remarks on augmented Lagrange-Newton SQP methods*. SFB-Preprint No. 256, 2003.

[12] S. Volkwein. *Numerische Verfahren der restringierten Optimierung*. Vorlesungsmanuscript, 2009.
`http://www.math.uni-konstanz.de/numerik/personen/volkwein/teaching/scripts.php`