Ein nicht-innerer Prädiktor-Korrektor Pfadverfolgungs-Algorithmus für das monotone lineare Komplementaritätsproblem

Universität Würzburg
Fakultät für Mathematik und Informatik
Institut für Mathematik
Betreuer: Prof. Dr. Christian Kanzow

Bachelorarbeit

Sommersemester 2024

Bastian Baumann
Matrikel-Nummer: 1839447
Bachelor Mathematik
eMail: bastian.baumann@stud-mail.uni-wuerzburg.de

26. September 2024

Inhaltsverzeichnis

1 Einleitung

1.1 Zusammenfassung

In der vorliegenden Bachelorthesis befassen wir uns mit einem nicht-inneren Prädiktor-Korrektor Pfadverfolgungs-Algorithmus zur Lösung von monotonen linearen Komplementaritätsproblemen, der 1997 von Jim Burke und Song Xu eingereicht wurde [1].
Lineare Komplementaritätsprobleme umfassen insbesondere drei wichtige Grundprobleme: lineare Programme, quadratische Programme und Bimatrix-Spiele in der Spieltheorie. Wir wollen uns im Verlauf dieser Arbeit ausschließlich mit den ersten beiden Grundproblemen und mit der Theorie für lineare Komplementaritätsprobleme im Allgemeinen befassen und den Algorithmus später für lineare Programme in Python implementieren.

1.2 Problemstellung und Motivation

In diesem Kapitel werden wir die Problemstellung von Grund auf formulieren. Wir definieren also lineare Komplementaritätsprobleme (LCP; englisch: linear complementarity problem), quadratische Programme (QP; englisch: quadratic program) und lineare Programme (LP; englisch: linear program) und setzen diese in einen für die spätere Analyse sinnvollen Kontext. Wir halten uns vorerst bei der Formulierung dieser an Kapitel 1 des sehr detaillierten Buchs von Cottle, Pang und Stone [2].

Definition 1.1. *Es seien ein Vektor $q \in \mathbb{R}^n$ und eine Matrix $M \in \mathbb{R}^{n \times n}$ gegeben. Wir bezeichnen als **lineares Komplementaritätsproblem** $LCP(q, M)$ in Abhängigkeit von q und M, das Suchen nach einem Vektor $x \in \mathbb{R}^n$, der*

$$\begin{aligned} x &\geq 0, \\ q + Mx &\geq 0, \\ x^T(q + Mx) &= 0 \end{aligned} \tag{1.1}$$

*erfüllt, wobei auch die Nicht-Existenz eines solchen Vektors ein mögliches Ergebnis ist. Handelt es sich bei M um eine positiv semi-definite Matrix, sprechen wir von einem **monotonen** linearen Komplementaritätsproblem.*

Es sei angemerkt, dass in der Literatur alternative, äquivalente Definitionen existieren. In dem von uns betrachteten Artikel [1] wird $LCP(q, M)$ als die Suche nach einem Vektor $(x, y) \in \mathbb{R}^n \times \mathbb{R}^n$, der

$$\begin{aligned} Mx - y + q = 0, \\ x \geq 0,\ y \geq 0,\ x^T y = 0 \end{aligned} \tag{1.2}$$

erfüllt, beschrieben. Die Äquivalenz dieser Formulierungen folgt für $y = q + Mx$ direkt.

Als Nächstes definieren wir die für uns relevanten Untergruppen der LCPs, mit denen wir uns im dritten Kapitel intensiver beschäftigen wollen. Wir starten dafür mit dem allgemeineren Problem, den quadratischen Programmen.

Definition 1.2. *Es seien ein Vektor* $c \in \mathbb{R}^n$*, eine symmetrische Matrix* $Q \in \mathbb{R}^{n\times n}$*, eine Matrix* $A \in \mathbb{R}^{m\times n}$ *und ein Vektor* $b \in \mathbb{R}^m$ *gegeben. Wir bezeichnen als* ***quadratisches Programm*** $QP(c, Q, A, b)$ *in Abhängigkeit von* c*,* Q*,* A *und* b *das Minimierungsproblem*

$$\begin{aligned} \min f(x) = c^T x + \frac{1}{2} x^T Q x \\ \text{u.d.N.} \quad Ax \geq b, \\ x \geq 0. \end{aligned} \tag{1.3}$$

Handelt es sich bei Q *um eine positiv semi-definite Matrix, sprechen wir von einem* ***konvexen*** *quadratischen Programm.*

Damit kommen wir zu linearen Programmen, deren direkte Verbindung zu quadratischen Programmen direkt sichtbar ist.

Definition 1.3. *Es seien ein Vektor* $c \in \mathbb{R}^n$*, eine Matrix* $A \in \mathbb{R}^{m\times n}$ *und ein Vektor* $b \in \mathbb{R}^m$ *gegeben. Wir bezeichnen als* ***lineares Programm*** $LP(c, A, b)$ *in Abhängigkeit von* c*,* A *und* b *das Minimierungsproblem*

$$\begin{aligned} \min f(x) = c^T x \\ \text{u.d.N.} \quad Ax \geq b, \\ x \geq 0. \end{aligned} \tag{1.4}$$

Aus *Definition 1.2.* und *Definition 1.3.* ist sofort ersichtlich, dass es sich bei linearen Programmen um einen Spezialfall der (konvexen) quadratischen Programme handelt. Beide Problemarten treten im Allgemeinen mit verschiedenen Formen von Nebenbedingungen auf, worauf wir in Kapitel 3 näher eingehen werden. An dieser Stelle entscheiden wir uns für diese Art von Nebenbedingungen, da sich so die Möglichkeit der Umformung zu linearen Komplementaritätsproblemen leicht zeigen lässt.
Als Verbindungsstück machen wir Gebrauch von alternativen Optimalitätsbedingungen, die in der Literatur der Inneren-Punkte-Verfahren üblich sind - einem Spezialfall der **Karush-Kuhn-Tucker Bedingungen**, oder auch KKT-Bedingungen. Da eine exakte Einführung an dieser Stelle den Rahmen sprengen würde, verweisen wir dafür auf das Buch von Geiger und Kanzow [3] und belassen es bei ihrer Formulierung.
Es sei $x \in \mathbb{R}^n$ eine lokal optimale Lösung von $QP(c, Q, A, b)$. Dann existiert ein Vektor $y \in \mathbb{R}^m$, sodass das Tupel (x, y)

$$\begin{aligned} u := c + Qx - A^T y \geq 0,\ x \geq 0,\ x^T u = 0, \\ v := -b + Ax \geq 0,\ y \geq 0,\ y^T v = 0 \end{aligned} \tag{1.5}$$

erfüllt.
Handelt es sich zusätzlich bei Q um eine positiv semi-definite Matrix, bei $QP(c, Q, A, b)$ also um ein konvexes quadratisches Programm, so ist das Erfüllen von (1.5) hinreichend dafür, dass x eine global optimale Lösung von $QP(c, Q, A, b)$ ist.

Die Optimalitätsbedingungen von $QP(c, Q, A, b)$ in (1.5) formulieren das $LCP(q, M)$ mit

$$q = \begin{bmatrix} c \\ -b \end{bmatrix}, \ M = \begin{bmatrix} Q & -A^T \\ A & 0 \end{bmatrix}. \tag{1.6}$$

Insbesondere für konvexe QPs handelt es sich mit $z := (x, y) \in \mathbb{R}^{n+m}$ wegen

$$z^T M z = x^T Q x - x^T A^T y + y^T A x = x^T Q x \geq 0$$

dabei um ein monotones lineares Komplementaritätsproblem, dessen globale Lösung gerade der des konvexen quadratischen Programms entspricht.

Wir halten fest: Wir können unsere folgende theoretische Analyse auf monotone lineare Komplementaritätsprobleme beschränken, da die Ergebnisse sich auf konvexe quadratische Programme und damit auch auf lineare Programme übertragen lassen.

Zunächst noch ein paar Worte zu unserer Notation. Alle Vektoren sind Spaltenvektoren, wobei mit Superskript T der transponierte Vektor gemeint ist. Mit $\mathbb{R}^n$ bezeichnen wir den Vektorraum, der alle n-dimensionalen reellen Vektoren enthält, wobei mit $\mathbb{R}^{n \times n}$ der Vektorraum gemeint ist, der alle reellen $(n \times n)$-Matrizen enthält. Mit $\mathbb{R}^n_{\geq 0}$ bezeichnen wir den nicht-negativen Teilraum des $\mathbb{R}^n$, wobei mit $\mathbb{R}^n_{>0}$ dessen Inneres gemeint ist. Für gegebene Vektoren $x, y \in \mathbb{R}^n$ schreiben wir $x \leq y$, wenn $y - x \in \mathbb{R}^n_{\geq 0}$, beziehungsweise $x < y$, wenn $y - x \in \mathbb{R}^n_{>0}$ gilt. Für $x \in \mathbb{R}^n$ bezeichnen wir mit $||x||_1$, $||x||_2$ und $||x||_\infty$ die Betragssummennorm, die euklidische Norm und die Maximumsnorm von x.

1.3 Der Algorithmus von Burke und Xu

Wir legen nun die grundlegenden Bausteine zur Formulierung des Algorithmus. Die Grundidee bei allen Pfadverfolgungs-Algorithmen ist es, dem sogenannten **zentralen Pfad** zu folgen. Zur Herleitung dieser Menge formulieren wir eine modifizierte Version unseres Grundproblems und folgen dabei der Notation in Kanzow [4]. Wenn nicht anders erwähnt, handelt es sich bei den Beweisen in diesem Kapitel um eine Ausarbeitung der Beweise in Burke und Xu [1].
Wir definieren das gestörte lineare Komplementaritätsproblem (PLCP; englisch: perturbed linear complementarity problem).

Definition 1.4. *Es seien ein Vektor $q \in \mathbb{R}^n$, eine Matrix $M \in \mathbb{R}^{n \times n}$ und ein Parameter $\mu > 0$ gegeben. Wir bezeichnen als* ***gestörtes Lineares Komplementaritätsproblem*** *$PLCP(q, M, \mu)$ in Abhängigkeit von q, M und μ das Suchen nach einem Vektor $(x(\mu), y(\mu))$, der*

$$\begin{aligned} x > 0, y > 0, \\ Mx - y + q = 0, \\ Xy = \mu^2 e \end{aligned} \tag{1.7}$$

erfüllt, wobei auch die Nicht-Existenz eines solchen Vektors ein mögliches Ergebnis ist.

Wir verwenden an dieser Stelle die in der Literatur Innerer-Punkte-Verfahren übliche Notation mit $X = \text{diag}(x_1, x_2, ..., x_n)^T$ und $e = (1, 1, ..., 1)^T$, siehe zum Beispiel Wright [5]. Durch das Zulassen von $x = 0$ und $y = 0$ würden wir für $\mu = 0$ gerade $LCP(q, M)$ wie in (1.2) erhalten.

Gegen Ende dieses Unterkapitels werden wir sehen, dass unter sinnvollen Voraussetzungen für jedes $\mu > 0$ eine Lösung $(x(\mu), y(\mu))$ von $PLCP(q, M, \mu)$ existiert. Dies führt zur Wohldefiniertheit der Abbildung $\mu \to (x(\mu), y(\mu))$, deren Graph dann eine glatte Trajektorie formt, die wir als den zentralen Pfad bezeichnen wollen. Die genannten Eigenschaften geben dabei Grund zur Hoffnung, als Grenzwert dieser Abbildung für $\mu \to 0$ eine Lösung des Ausgangsproblems $LCP(q, M)$ zu finden. Wir formulieren den Zentralen Pfad erneut als Menge

$$\zeta := \{(x, y) \mid x > 0,\ y > 0,\ Mx - y + q = 0,\ Xy = \mu^2 e \text{ für ein } \mu > 0\}. \tag{1.8}$$

Als weiteres Hilfsmittel benötigen wir eine sogenannte **Chen-Harker-Kanzow-Smale Glättungsfunktion**, die in Kanzow [4] näher beschrieben wird. Der Algorithmus beruht auf der Funktion

$$\phi(a, b, \mu) := a + b - \sqrt{(a - b)^2 + 4\mu^2}, \tag{1.9}$$

deren für uns relevanten Eigenschaften wir in folgendem Lemma beweisen wollen. Es handelt sich dabei um die **geglättete Minimum-Funktion**.

Lemma 1.5. *Es sei* $\phi : \mathbb{R}^2 \times \mathbb{R}_{\geq 0} \to \mathbb{R},\ (a, b, \mu) \to a + b - \sqrt{(a - b)^2 + 4\mu^2}$.

(i) *Es gilt* $\phi(a, b, \mu) = 0$ *genau dann, wenn* $a \geq 0$, $b \geq 0$ *und* $ab = \mu^2$.
(ii) *Die Funktion* $\phi(a, b, \mu)$ *ist stetig differenzierbar für* $\mu > 0$.
(iii) *Für die partiellen Ableitungen gilt* $\frac{\partial \phi(a,b,\mu)}{\partial a}, \frac{\partial \phi(a,b,\mu)}{\partial b} \in (0, 2)$ *für* $\mu > 0$.
(iv) *Die Funktion* $\phi(a, b, \mu)$ *ist konkav für* $\mu > 0$.
(v) *Es gilt für* $\mu > 0$

$$\left|\left|\nabla^2 \phi(a, b, \mu)\right|\right|_2 \leq \frac{4}{\sqrt{(a - b)^2 + 4\mu^2}} \leq \frac{2}{\mu}.$$

(vi) *Es gilt für* $\mu > 0$

$$|\nabla_\mu \phi(a, b, \mu)| = \left| - \frac{4\mu}{\sqrt{(a - b)^2 + 4\mu^2}} \right| \leq 2.$$

Beweis. **(i)** Der Beweis von (i) folgt dem Beweis von Lemma 2.1. in Kanzow [4].
Es seien $a \geq 0$, $b \geq 0$ und $ab = \mu^2$. Dann gilt

$$\begin{aligned}
\phi(a, b, \mu) &= a + b - \sqrt{a^2 - 2ab + b^2 + 4ab} \\
&= a + b - \sqrt{(a+b)^2} \\
&= a + b - |a + b| \\
&= 0.
\end{aligned}$$

Sei nun $\phi(a, b, \mu) = 0$. Dann gilt

$$a + b = \sqrt{(a-b)^2 + 4\mu^2} \geq 0$$

Durch das Quadrieren beider Seiten erhält man sofort $ab = \mu^2$.
Für $\mu = 0$ folgt entweder $a = 0$ und damit wegen der Ungleichung $b \geq 0$, oder $b = 0$ und damit $a \geq 0$.
Für $\mu > 0$ muss $\text{sign}(a) = \text{sign}(b)$ gelten. Aus der Ungleichung folgt damit direkt $a > 0$ und $b > 0$, was zu zeigen war.
(ii) Partielles Ableiten ergibt

$$\nabla \phi(a, b, \mu) = \begin{bmatrix} 1 - \frac{a-b}{\sqrt{(a-b)^2+4\mu^2}} \\ 1 + \frac{a-b}{\sqrt{(a-b)^2+4\mu^2}} \\ -\frac{4\mu}{\sqrt{(a-b)^2+4\mu^2}} \end{bmatrix} \tag{1.10}$$

und damit wegen $\mu > 0$ die Aussage.
(iii) Es gilt für $\mu > 0$

$$\left| \frac{a-b}{\sqrt{(a-b)^2 + 4\mu^2}} \right| < 1$$

und damit

$$0 < 1 - \frac{a-b}{\sqrt{(a-b)^2 + 4\mu^2}} < 2 \quad \text{und} \quad 0 < 1 + \frac{a-b}{\sqrt{(a-b)^2 + 4\mu^2}} < 2.$$

(iv) Erneutes Ableiten ergibt für

$$\nabla^2 \phi(a, b, \mu) = \frac{4}{((a-b)^2 + 4\mu^2)^{\frac{3}{2}}} \begin{bmatrix} -\mu^2 & \mu^2 & (a-b)\mu \\ \mu^2 & -\mu^2 & -(a-b)\mu \\ (a-b)\mu & -(a-b)\mu & -(a-b)^2 \end{bmatrix}.$$

Wegen $\mu > 0$, also $-\mu^2 < 0$ und der Singularität der restlichen Hauptminoren, folgt die negative Semi-Definitheit von $\nabla^2 \phi(a, b, \mu)$ und damit die Konkavität von $\phi(a, b, \mu)$.
(v) Weiter folgt mit der Frobeniusnorm $|| \cdot ||_F := \sqrt{\text{spur}(A^T A)}$

$$\left\| \nabla^2 \phi(a, b, \mu) \right\|_2 \leq \left\| \nabla^2 \phi(a, b, \mu) \right\|_F = \frac{4((a-b)^2 + 2\mu^2)}{((a-b)^2 + 4\mu^2)^{\frac{3}{2}}} \leq \frac{4}{\sqrt{(a-b)^2 + 4\mu^2}} \leq \frac{2}{\mu}.$$

(vi) Zuletzt gilt

$$|\nabla_\mu \phi(a,b,\mu)| = \left| -\frac{4\mu}{\sqrt{(a-b)^2+4\mu^2}} \right| \leq \frac{4\mu}{\sqrt{4\mu^2}} \leq 2.$$

□

Wie bei allen Pfadverfolgungs-Algorithmen, ist es unser Ziel dem zentralen Pfad mittels des **Newton-Verfahrens** zu folgen. Dafür ist es zunächst notwendig unser Grundproblem in eine Art Nullstellenproblem umzuformen.

Wir definieren dafür die Funktion $F : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}_{\geq 0} \to \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}_{\geq 0}$

$$F(x,y,\mu) := \begin{bmatrix} Mx - y + q \\ \Phi(x,y,\mu) \\ \mu \end{bmatrix} \tag{1.11}$$

mit

$$\Phi(x,y,\mu) := \begin{bmatrix} \phi(x_1,y_1,\mu) \\ \dots \\ \phi(x_n,y_n,\mu) \end{bmatrix}. \tag{1.12}$$

Wir erkennen direkt, dass

$$F(x,y,\mu) = 0 \tag{1.13}$$

genau dann gilt, wenn (x,y) eine Lösung für $LCP(q,M)$ ist und

$$F(x,y,\mu) = \begin{bmatrix} 0 \\ 0 \\ \bar{\mu} \end{bmatrix} \tag{1.14}$$

mit $\bar{\mu} > 0$ genau dann, wenn (x,y) eine Lösung für $PLCP(q,M,\bar{\mu})$ ist und damit auf dem zentralen Pfad liegt.
Zuletzt formulieren wir noch eine **Umgebung für den zentralen Pfad**, die uns später gewährleistet, dass wir mit unserem Pfadverfolgungs-Verfahren nicht zu sehr vom zentralen Pfad abweichen und untersuchen diese auf Beschränktheit. Die Menge

$$\mathcal{N}(\beta) := \{(x,y) \mid Mx - y + q = 0,\ \Phi(x,y,\mu) \leq 0,\ ||\Phi(x,y,\mu)||_2 \leq \beta\mu \text{ für ein } \mu > 0\},$$

für ein gegebenes $\beta > 0$, besteht aus der Vereinigung der Scheiben

$$\mathcal{N}(\beta,\mu) := \{(x,y) \mid Mx - y + q = 0,\ \Phi(x,y,\mu) \leq 0,\ ||\Phi(x,y,\mu)||_2 \leq \beta\mu\}, \tag{1.15}$$

für $\mu > 0$. Für die zwei folgenden Resultate benötigen wir eine Annahme an die Lösbarkeit unseres Grundproblems.

Annahme (A)

Es sei $LCP(q, M)$ monoton und es existiere ein "innerer" Punkt der zulässigen Menge, also ein $(x, y) \in \mathbb{R}^n \times \mathbb{R}^n$, so dass

$$x > 0,\ y > 0,\ Mx - y + q = 0. \tag{1.16}$$

Unter Verwendung dieser Annahme lässt sich nun die Beschränktheit der Vereinigung der Mengen $\mathcal{N}(\beta, \mu)$ für beschränkte μ nachweisen.

Lemma 1.6. *Es gelte Annahme (A). Dann ist die Menge*

$$\bigcup_{0<\mu\leq\mu_0} \mathcal{N}(\beta, \mu)$$

für jedes $\beta > 0$ und $\mu_0 > 0$ beschränkt. Genauer gilt für jedes $(x, y) \in \bigcup_{0<\mu\leq\mu_0} \mathcal{N}(\beta, \mu)$

$$-\frac{\beta\mu_0}{2} \leq x_i \leq \frac{\bar{x}^T\bar{y} + \frac{\beta\mu_0}{2}\left(||\bar{x}||_1 + ||\bar{y}||_1\right) + n\max\left\{\mu_0^2, \frac{\beta^2\mu_0^2}{4}\right\}}{\bar{y}_i}$$
$$-\frac{\beta\mu_0}{2} \leq y_i \leq \frac{\bar{x}^T\bar{y} + \frac{\beta\mu_0}{2}\left(||\bar{x}||_1 + ||\bar{y}||_1\right) + n\max\left\{\mu_0^2, \frac{\beta^2\mu_0^2}{4}\right\}}{\bar{x}_i}$$

für alle $i = 1, 2, ..., n$, wobei $(\bar{x}, \bar{y})$ ein beliebiger innerer Punkt der zulässigen Menge ist.

Beweis. Es seien $0 < \mu \leq \mu_0$, $\beta > 0$ und (x, y), so dass $(x, y) \in \mathcal{N}(\beta, \mu)$. Außerdem sei $(\bar{x}, \bar{y})$ ein beliebiger innerer Punkt der zulässigen Menge, der nach Annahme (A) existiert. Wir betrachten zuerst folgende Äquivalenzumformungen

$$\begin{aligned}
& -\delta \leq a + b - \sqrt{(a-b)^2 + 4\mu^2} \\
\iff\quad & \sqrt{(a-b)^2 + 4\mu^2} \leq a + b + \delta \qquad (1.17) \\
\iff\quad & \left((a + \delta/2) - (b + \delta/2)\right)^2 + 4\mu^2 \leq \left((a + \delta/2) + (b + \delta/2)\right)^2 \\
\iff\quad & 4\mu^2 \leq 4\left(b + \delta/2\right)\left(a + \delta/2\right), \qquad (1.18)
\end{aligned}$$

die wegen $0 \leq \sqrt{(a-b)^2 + 4\mu^2} \leq a + b + \delta = (a + \delta/2) + (b + \delta/2)$ in (1.17) und $\mu > 0$ in (1.18) die Positivität von $(a + \delta/2)$ und $(b + \delta/2)$ für beliebige a, b implizieren, wenn $-\delta \leq \phi(a, b, \mu)$ gilt. Dies zeigt uns, dass aus $-\delta \leq \phi(a, b, \mu)$ direkt $-\delta/2 < \min\{a, b\}$ folgt und damit

$$x_i > -(\beta\mu)/2 \geq -(\beta\mu_0)/2 \quad \text{und} \quad y_i > -(\beta\mu)/2 \geq -(\beta\mu_0)/2 \tag{1.19}$$

für alle $i = 1, 2, ..., n$ gilt, da für $(x, y) \in \mathcal{N}(\beta, \mu)$ nach Definition $\phi(x_i, y_i, \mu) \leq 0$ und $-\phi(x_i, y_i, \mu) = |\phi(x_i, y_i, \mu)| \leq ||\Phi(x, y, \mu)||_\infty \leq ||\Phi(x, y, \mu)||_2 \leq \beta\mu$ gelten muss.
Seien nun $a \geq 0$ und $b \geq 0$, dann folgt aus $\phi(a, b, \mu) \leq 0$ direkt $0 \leq a+b \leq \sqrt{(a-b)^2 + 4\mu^2}$. Quadrieren und Wegstreichen wie oben führt zu $ab \leq \mu^2$ und damit zu

$$x_i y_i \leq \mu_0^2 \tag{1.20}$$

für alle $i = 1, 2, ..., n$ mit $x_i \geq 0$ und $y_i \geq 0$.
Zuletzt erhalten wir, wegen der durch Annahme (A) gewährleisteten Monotonie,

$$(\bar{x} - x)^T(\bar{y} - y) = (\bar{x} - x)^T((M\bar{x} + q) - (Mx + q)) = (\bar{x} - x)^T M(\bar{x} - x) \geq 0$$

und damit

$$\bar{x}^T y + x^T \bar{y} \leq \bar{x}^T \bar{y} + x^T y.$$

Diese Ungleichung liefert uns, gemeinsam mit (1.19) und (1.20),

$$\begin{aligned}
\sum_{y_i>0} \bar{x}_i y_i + \sum_{x_i>0} x_i \bar{y}_i &\leq \bar{x}^T \bar{y} + x^T y - \left(\sum_{y_i<0} \bar{x}_i y_i + \sum_{x_i<0} x_i \bar{y}_i \right) \\
&\leq \bar{x}^T \bar{y} + x^T y + \frac{\beta\mu_0}{2} \left(||\bar{x}||_1 + ||\bar{y}||_1 \right) \\
&\leq \bar{x}^T \bar{y} + \sum_{x_i y_i > 0} x_i y_i + \frac{\beta\mu_0}{2} \left(||\bar{x}||_1 + ||\bar{y}||_1 \right) \\
&\leq \bar{x}^T \bar{y} + n \max\{\mu_0^2, \frac{\beta^2\mu^2}{4}\} + \frac{\beta\mu_0}{2} \left(||\bar{x}||_1 + ||\bar{y}||_1 \right).
\end{aligned}$$

Für $x_i > 0$ gilt also

$$x_i \leq \frac{\bar{x}^T \bar{y} + \frac{\beta\mu_0}{2} \left(||\bar{x}||_1 + ||\bar{y}||_1 \right) + n \max\left\{\mu_0^2, \frac{\beta^2\mu_0^2}{4}\right\}}{\bar{y}_i}$$

und für $y_i > 0$

$$y_i \leq \frac{\bar{x}^T \bar{y} + \frac{\beta\mu_0}{2} \left(||\bar{x}||_1 + ||\bar{y}||_1 \right) + n \max\left\{\mu_0^2, \frac{\beta^2\mu_0^2}{4}\right\}}{\bar{x}_i},$$

wobei für negative Komponenten die Schranke schon durch (1.19) gegeben war. □

Diese Beschränktheit kann in Verbindung mit Annahme (A) dazu genutzt werden die Wohldefiniertheit des zentralen Pfads und die Lösbarkeit unseres Grundproblems zu gewährleisten.

Satz 1.7. *Es gelte Annahme (A). Dann gilt*

(i) *Für jedes $\mu > 0$ besitzt $PLCP(q, M, \mu)$ eine Lösung (x, y).*
(ii) *$LCP(q, M)$ besitzt eine Lösung (x, y).*

Beweis. Für diesen Satz verweisen wir auf Kojima, Megiddo, Noma, und Yoshi [6], da der Beweis an dieser Stelle den Rahmen sprengen würde.
Aussage (i) wird dabei in Lemma 4.3 in [6] gezeigt, wobei Aussage (ii) in Teil 3 von Theorem 4.4 in [6] nachgewiesen wird.

An dieser Stelle können wir die Idee unseres Algorithmus formulieren.

Nach sinnvoller Initialisierung wird zuerst ein **Prädiktor-Schritt** durchgeführt. Dabei handelt es sich um einen Newton-Schritt, der auf Gleichung (1.13) basiert. Es wird überprüft, ob der Prädiktor-Schritt in der vorher festgelegten Umgebung $\mathcal{N}(\beta, \mu_k)$ des zentralen Pfads endet. Ist dies der Fall, wird der Glättungsparameter μ_k so weit reduziert wie möglich, ohne dass der Schritt die Umgebung verlässt. Ist dies nicht der Fall, wird der Prädiktor-Schritt in dieser Iteration abgelehnt.

Daraufhin wird ein **Korrektor-Schritt** durchgeführt, bei dem es sich erneut um einen Newton-Schritt handelt, der auf obiger Gleichung basiert. Diesmal wird allerdings die rechte Seite etwas angepasst, was durch Gleichung (1.14) motiviert wird. Der sogenannte Zentrierungsparameter $\bar{\sigma}$ soll dabei dafür sorgen, dass der Korrektor-Schritt nicht nur in Richtung der globalen Lösung geht, sondern die aktuelle Iterierte auch in Richtung des zentralen Pfads zentriert, siehe beispielsweise Wright [5] für mehr Details. Die Schrittweite wird hierbei durch eine Armijo-Bedingung festgelegt, damit wir auch hier wieder nicht die Umgebung $\mathcal{N}(\beta, \hat{\mu_k})$ verlassen.

Wie wir später sehen werden, sorgt hierbei der Prädiktor-Schritt für die **schnelle lokale Konvergenz**, während der Korrektor-Schritt **globale Konvergenz** gewährleistet.

Wir formulieren Algorithmus 1 und beenden damit das Kapitel.

Algorithm 1:

1 **Schritt 0: (Initialisierung)** Setze $k := 0$.

2 Wähle $x^0 \in \mathbb{R}^n$, $y^0 = Mx^0 + q$ und μ_0, so dass $\Phi(x^0, y^0, \mu_0) < 0$.

3 Wähle $\beta > 2\sqrt{n}$, so dass $||\Phi(x^0, y^0, \mu_0)||_2 \leq \beta\mu_0$.

4 Wähle $\bar{\sigma}$, α_1 und α_2 aus $(0, 1)$.

5 **Schritt 1: (Prädiktor-Schritt)** Bestimme $(\Delta x^k, \Delta y^k, \Delta\mu_k)$ als Lösung des linearen Gleichungssystems

$$F(x^k, y^k, \mu_k) + \nabla F(x^k, y^k, \mu_k)^T \begin{bmatrix} \Delta x^k \\ \Delta y^k \\ \Delta\mu_k \end{bmatrix} = 0. \tag{1.21}$$

6 **if** $||\Phi(x^k + \Delta x^k, y^k + \Delta y^k, 0)||_2 = 0$, **then**

7 STOP,

$$(x^k + \Delta x^k, y^k + \Delta y^k, 0) \,\text{löst}\, LCP(q, M); \tag{1.22}$$

8 **else if** $||\Phi(x^k + \Delta x^k, y^k + \Delta y^k, \mu_k)||_2 > \beta\mu_k$, **then**

9 setze

$$\hat{x}^k := x^k,\ \hat{y}^k := y^k,\ \hat{\mu}_k := \mu_k \text{ und } \eta_k := 1; \tag{1.23}$$

10 **else**

11 wähle $\eta_k = \alpha_1^s$, so dass s die nicht-negative ganze Zahl mit

$$||\Phi(x^k + \Delta x^k, y^k + \Delta y^k, \alpha_1^t\mu_k)||_2 \leq \alpha_1^t\beta\mu_k \text{ für alle } t = 0, 1, ..., s \text{ und} \tag{1.24}$$

$$||\Phi(x^k + \Delta x^k, y^k + \Delta y^k, \alpha_1^{s+1}\mu_k)||_2 > \alpha_1^{s+1}\beta\mu_k \tag{1.25}$$

ist und setze

$$\hat{x}^k := x^k + \Delta x^k,\ \hat{y}^k := y^k + \Delta y^k,\ \hat{\mu}_k := \eta_k\mu_k; \tag{1.26}$$

12 **Schritt 2: (Korrektor-Schritt)** Bestimme $(\Delta\hat{x}^k, \Delta\hat{y}^k, \Delta\hat{\mu}_k)$ als Lösung des linearen Gleichungssystems

$$F(\hat{x}^k, \hat{y}^k, \hat{\mu}_k) + \nabla F(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)^T \begin{bmatrix} \Delta\hat{x}^k \\ \Delta\hat{y}^k \\ \Delta\hat{\mu}_k \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ (1 - \bar{\sigma})\hat{\mu}_k \end{bmatrix}. \tag{1.27}$$

13 Wähle $\hat{\lambda}_k$ als Maximum der Werte $1, \alpha_2, \alpha_2^2, ...$, so dass
$||\Phi(\hat{x}^k + \hat{\lambda}_k\Delta\hat{x}^k, \hat{y}^k + \hat{\lambda}_k\Delta\hat{y}^k, (1 - \bar{\sigma}\hat{\lambda}_k)\hat{\mu}_k)||_2 \leq (1 - \bar{\sigma}\hat{\lambda}_k)\beta\hat{\mu}_k$
gilt und setze

$$x^{k+1} := \hat{x}^k + \hat{\lambda}_k\Delta\hat{x}^k,\ y^{k+1} := \hat{y}^k + \hat{\lambda}_k\Delta\hat{y}^k,\ \mu_{k+1} := (1 - \bar{\sigma}\hat{\lambda}_k)\hat{\mu}_k; \tag{1.28}$$

und gehe mit $k := k + 1$ zurück zu Schritt 1 in Zeile 5.

2 Theoretische Analyse für monotone LCP

In diesem Kapitel beweisen wir die theoretischen Eigenschaften des Algorithmus. Zuerst weisen wir hierfür nach, dass der Algorithmus **wohldefiniert** und wie beschrieben ausführbar ist. Daraufhin zeigen wir die **globale lineare Konvergenz**, die durch den Korrektor-Schritt gewährleistet wird. Zuletzt belegen wir, unter bestimmten Voraussetzungen, die **lokal quadratische Konvergenz** des Algorithmus. Auch in diesem Kapitel handelt es sich bei den Beweisen, wenn nicht anders erwähnt, um eine Ausarbeitung der Resultate aus Burke und Xu [1].

2.1 Wohldefiniertheit

Die Initialisierung des Algorithmus ist unproblematisch. Da an den Startvektor x^0 quasi keine Voraussetzungen gestellt werden und (x^0, y^0) insbesondere nicht einmal nichtnegativ sein muss, erfordert das Bestimmen der Startwerte maximal eine Matrixmultiplikation und eine Vektoraddition, damit $y^0 = Mx^0 + q$ erfüllt ist.
Für μ_0 muss dabei nur

$$\mu_0 > \sqrt{\max\{x_i^0 y_i^0 \mid x_i > 0,\ y_i > 0\}}$$

gelten, damit $\Phi(x^0, y^0, \mu_0) < 0$ gewährleistet ist. Dies ist beispielsweise für $(x^0, y^0) = (0, q)$ für alle positiven μ_0 der Fall. Für β muss lediglich

$$\beta > \max\left\{2\sqrt{n},\ \frac{||\Phi(x^0, y^0, \mu_0)||_2}{\mu_0}\right\}$$

erfüllt sein, was leicht zu bestimmen ist. Für den Beweis der Wohldefiniertheit benötigen wir zusätzlich die (eindeutige) Lösbarkeit der Gleichungssysteme (1.21) und (1.27). Diese beweisen wir in folgendem Lemma und halten uns dabei an den Beweis zu Theorem 3.5 in Kanzow [4].

Lemma 2.1. *Es gelte Annahme (A). Dann ist die Jacobi-Matrix $\nabla F(x, y, \mu)^T$ für alle $(x, y) \in \mathbb{R}^{2n}$ und $\mu > 0$ regulär.*

Beweis. Die Jacobi-Matrix hat die Form

$$\nabla F(x, y, \mu)^T = \begin{bmatrix} M & -I & 0 \\ \nabla_x \Phi & \nabla_y \Phi & \nabla_\mu \Phi \\ 0 & 0 & 1 \end{bmatrix} \tag{2.1}$$

mit den zwei Diagonalmatrizen und dem Spaltenvektor

$$\nabla_x \Phi := \operatorname{diag}\left(\left.\frac{\partial \phi(a, b, \mu)}{\partial a}\right|_{\substack{a=x_i \\ b=s_i}}\right), \nabla_s \Phi := \operatorname{diag}\left(\left.\frac{\partial \phi(a, b, \mu)}{\partial b}\right|_{\substack{a=x_i \\ b=s_i}}\right), \nabla_\mu \Phi := \left(\left.\frac{\partial \phi(a, b, \mu)}{\partial \mu}\right|_{\substack{a=x_i \\ b=s_i}}\right)^T. \tag{2.2}$$

Im Folgenden werden wir bei obiger Notation Φ einen Index geben, um zu kennzeichnen welche x-, y- und μ-Werte dabei eingesetzt werden. Es ist direkt zu erkennen, dass

$\nabla F(x,y,\mu)^T$ genau dann regulär ist, wenn ihr Hauptminor zweiter Ordnung, den wir im Folgenden mit

$$\nabla \bar{F}_\mu(x,y)^T := \begin{bmatrix} M & -I \\ \nabla_x \Phi & \nabla_y \Phi \end{bmatrix} \tag{2.3}$$

bezeichnen wollen, regulär ist.
Sei nun $\nabla \bar{F}_\mu(x,y)^T p = 0$ mit $p = (p^{(1)}, p^{(2)})$ für $p^{(i)} \in \mathbb{R}^n$, $i = 1, 2$. Dann gilt

$$Mp^{(1)} - p^{(2)} = 0, \tag{2.4}$$

$$\nabla_x \Phi p^{(1)} + \nabla_y \Phi p^{(2)} = 0. \tag{2.5}$$

Aus *Lemma 1.5.* (iii) wissen wir, dass die Diagonalmatrizen $\nabla_x \Phi$ und $\nabla_y \Phi$ positiv definit sind. Daher können wir (2.5) umschreiben zu

$$p^{(2)} = -\nabla_y \Phi^{-1} \nabla_x \Phi p^{(1)}, \tag{2.6}$$

womit auch $\nabla_y \Phi^{-1} \nabla_x \Phi$ eine positiv definite Diagonalmatrix ist. Setzen wir (2.6) in (2.5) ein, erhalten wir

$$(M + \nabla_y \Phi^{-1} \nabla_x \Phi) p^{(1)} = 0. \tag{2.7}$$

Da wegen Annahme (A) M positiv semi-definit ist, handelt es sich bei $M + \nabla_y \Phi^{-1} \nabla_x \Phi$ um eine positiv definite und damit reguläre Matrix. Es folgt

$$p^{(1)} = 0.$$

Aus (2.4) folgt daher direkt

$$p^{(2)} = 0$$

und damit die Aussage. □

Mit (2.1) folgt außerdem für $Mx^k - y^k + q = 0$

$$M\lambda\Delta x^k - \lambda\Delta y^k = 0. \tag{2.8}$$

für beliebige Schrittweiten $\lambda \in [0,1]$. Dies gewährleistet, dass der oberste Block-Eintrag von $F(x^k, y^k, \mu_k)$ und $F(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)$ in jedem Iterationsschritt verschwindet.

Im folgenden Resultat können wir nun die Wohldefiniertheit des Algorithmus beweisen, indem wir noch zeigen, dass die Iterationsvorschriften im Prädiktor- und Korrektor-Schritt sinnvoll definiert sind und die Backtracking Routinen endlich terminieren.

Satz 2.2. *(Wohldefiniertheit) Es gelte Annahme (A). Wir betrachten Algorithmus 1, wie oben beschrieben. Ist $(x^k, y^k) \in \mathcal{N}(\beta, \mu_k)$ mit $\mu_k > 0$, dann löst entweder $(x^k + \Delta x^k, y^k + \Delta y^k)$ $LCP(q, M)$ oder sowohl $(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)$, als auch $(x^{k+1}, y^{k+1}, \mu_{k+1})$ sind wohldefiniert und die Backtracking Routine im Prädiktor- und Korrektor-Schritt ist endlich terminierend. Im zweiten Fall ist $(\hat{x}^k, \hat{y}^k) \in \mathcal{N}(\beta, \hat{\mu}_k)$ und $(x^{k+1}, y^{k+1}) \in \mathcal{N}(\beta, \mu_{k+1})$ mit $0 < \mu_{k+1} < \hat{\mu}_k$.*
Für $(x^0, y^0) \in \mathcal{N}(\beta, \mu_0)$ mit $\mu_0 > 0$ folgt damit, dass Algorithmus 1 ***wohldefiniert*** *ist.*

Beweis. Es sei $(x^k, y^k) \in \mathcal{N}(\beta, \mu_k)$ mit $\mu_k > 0$. Da Annahme (A) gilt, wissen wir wegen *Lemma 2.1.*, dass $(\Delta x^k, \Delta y^k, \Delta \mu_k)$ existiert und eindeutig ist.
Wegen $y^k + \Delta y^k = M(x^k + \Delta x^k) + q$ gilt

$$\left|\left|\Phi(x^k + \Delta x^k, y^k + \Delta y^k, 0)\right|\right|_2 = 0$$

genau dann, wenn $(x^k + \Delta x^k, y^k + \Delta y^k)$ $LCP(q, M)$ löst - siehe (1.13).
Löst $(x^k + \Delta x^k, y^k + \Delta y^k)$ $LCP(q, M)$ nicht, existieren aus Stetigkeitsgründen $\epsilon > 0$ und $\bar{\mu} > 0$, so dass

$$\left|\left|\Phi(x^k + \Delta x^k, y^k + \Delta y^k, \mu)\right|\right|_2 > \epsilon$$

für alle $\mu \in (0, \bar{\mu})$ gilt. In diesem Fall sieht man direkt, dass die Backtracking Routine in (1.24) und (1.25) endlich terminierend ist. Damit ist $(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)$ wohldefiniert mit $0 < \hat{\mu}_k \leq \mu_k$.
Damit ist auch $(\Delta \hat{x}^k, \Delta \hat{y}^k, \Delta \hat{\mu}_k)$ wieder eindeutig bestimmt und liefert uns in der zweiten Blockzeile von (1.27) mit der Notation aus (2.2)

$$\nabla \Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)^T \begin{bmatrix} \Delta \hat{x}^k \\ \Delta \hat{y}^k \\ \Delta \hat{\mu}_k \end{bmatrix} = \begin{bmatrix} \nabla_x \Phi_{\hat{k}} & \nabla_y \Phi_{\hat{k}} & \nabla_\mu \Phi_{\hat{k}} \end{bmatrix} \begin{bmatrix} \Delta \hat{x}^k \\ \Delta \hat{y}^k \\ \Delta \hat{\mu}_k \end{bmatrix} = -\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k),$$

womit wir für die Richtungsableitung von $||\Phi(x, y, \mu)||_2$ in Richtung $(\Delta \hat{x}^k, \Delta \hat{y}^k, \Delta \hat{\mu}_k)$

$$\begin{aligned} \left(D_{(\Delta \hat{x}^k, \Delta \hat{y}^k, \Delta \hat{\mu}_k)} ||\Phi(x, y, \mu)||_2\right)(\hat{x}^k, \hat{y}^k, \hat{\mu}_k) &= \frac{\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)}{||\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)||_2} \cdot \nabla \Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)^T \cdot (\Delta \hat{x}^k, \Delta \hat{y}^k, \Delta \hat{\mu}_k)^T \\ &= \frac{\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)}{||\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)||_2} \cdot -\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k) \\ &= -\left|\left|\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)\right|\right|_2 \end{aligned}$$

erhalten. Es handelt sich also um eine Abstiegsrichtung, womit unsere Backtracking Routine zu einem Backtracking Line Search wird. Für diesen können wir die endliche Terminierung mit einer Beweisidee von Burke [7] zeigen. Es handelt sich hierbei um die Wohldefiniertheit der sogenannten Armijo-Regel. Wegen $\bar{\sigma} \in (0, 1)$ gilt

$$\begin{aligned} \lim_{h \searrow 0} \frac{\left|\left|\Phi(\hat{x}^k + h\Delta \hat{x}^k, \hat{y}^k + h\Delta \hat{y}^k, \hat{\mu}_k + h\Delta \hat{\mu}_k)\right|\right|_2 - \left|\left|\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)\right|\right|_2}{h} &= -\left|\left|\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)\right|\right|_2 \\ &< -\bar{\sigma} \left|\left|\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)\right|\right|_2 \\ &< 0. \end{aligned}$$

Daher existiert ein $0 < \bar{h} < 1$, so dass für alle $h \in (0, \bar{h})$

$$\frac{\left|\left|\Phi(\hat{x}^k + h\Delta\hat{x}^k, \hat{y}^k + h\Delta\hat{y}^k, \hat{\mu}_k + h\Delta\hat{\mu}_k)\right|\right|_2 - \left|\left|\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)\right|\right|_2}{h} \leq -\bar{\sigma}\left|\left|\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)\right|\right|_2$$

gelten muss. Umformen dieser Ungleichung ergibt

$$\begin{aligned}\left|\left|\Phi(\hat{x}^k + h\Delta\hat{x}^k, \hat{y}^k + h\Delta\hat{y}^k, \hat{\mu}_k + h\Delta\hat{\mu}_k)\right|\right|_2 &\leq -\bar{\sigma}h\left|\left|\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)\right|\right|_2 + \left|\left|\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)\right|\right|_2 \\ &= (1-\bar{\sigma}h)\left|\left|\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)\right|\right|_2 \\ &\leq (1-\bar{\sigma}h)\beta\hat{\mu}_k,\end{aligned}$$

da wegen (1.24) oder - im Fall des abgelehnten Prädiktor-Schritts - wegen der Gültigkeit des vorherigen Korrektor-Schritts $\left|\left|\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)\right|\right|_2 \leq \beta\hat{\mu}_k$ gilt. Wie wir später in diesem Beweis noch sehen und verwenden werden, entspricht $\hat{\mu}_k + h\Delta\hat{\mu}_k$ gerade $(1-\bar{\sigma}h)\hat{\mu}_k$, wodurch wir insgesamt die endliche Terminierung dieser Backtracking Routine mit $0 < \mu_{k+1} < \hat{\mu}_k \leq \mu_k$ erhalten.
Wegen $(x^k, y^k) \in \mathcal{N}(\beta, \mu_k)$ impliziert die obige Argumentation, dass entweder $(x^k + \Delta x^k, y^k + \Delta y^k)$ $LCP(q, M)$ löst, oder $\hat{y}^k = M\hat{x}^k + q$ mit $||\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)||_2 \leq \beta\hat{\mu}_k$ und $y^{k+1} = Mx^{k+1} + q$ mit $||\Phi(x^{k+1}, y^{k+1}, \mu_{k+1})||_2 \leq \beta\mu_{k+1}$ gilt.
Es bleibt also wegen (2.8) nur noch zu zeigen, dass wenn $(x^k + \Delta x^k, y^k + \Delta y^k)$ $LCP(q, M)$ nicht löst, sowohl $\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k) \leq 0$ als auch $\Phi(x^{k+1}, y^{k+1}, \mu_{k+1}) \leq 0$ gilt und damit $(\hat{x}^k, \hat{y}^k) \in \mathcal{N}(\beta, \hat{\mu}_k)$ und $(x^{k+1}, y^{k+1}) \in \mathcal{N}(\beta, \mu_{k+1})$ - unser Algorithmus also die Umgebung des zentralen Pfads nicht verlässt.
Durch die komponentenweise Konkavität von Φ aus *Lemma 1.5.* (iv) gilt für $(x, y, \mu) \in \mathbb{R}^{2n+1}$ mit $\mu > 0$ und $(\Delta x, \Delta y, \Delta\mu) \in \mathbb{R}^{2n+1}$

$$\Phi(x + \Delta x, y + \Delta y, \mu + \Delta\mu) \leq \Phi(x, y, \mu) + \nabla\Phi(x, y, \mu)^T \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta\mu \end{bmatrix}.$$

Zusätzlich betrachten wir das zu lösende Gleichungssystem des Prädiktor-Schritts in (1.21) mit der Notation aus (2.2)

$$\begin{bmatrix} 0 \\ \Phi(x^k, y^k, \mu_k) \\ \mu_k \end{bmatrix} + \begin{bmatrix} M & -I & 0 \\ \nabla_x\Phi_k & \nabla_y\Phi_k & \nabla_\mu\Phi_k \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \Delta x^k \\ \Delta y^k \\ \Delta\mu_k \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.$$

Aus der dritten Blockzeile erhält man direkt $\Delta\mu_k = -\mu_k$ und durch Einsetzen dessen in die zweite Blockzeile damit das Block-Gleichungssystem

$$\begin{aligned} M\Delta x^k - \Delta y^k &= 0, \\ \Phi(x^k, y^k, \mu_k) + \nabla\Phi(x^k, y^k, \mu_k)^T \begin{bmatrix} \Delta x^k \\ \Delta y^k \\ -\mu_k \end{bmatrix} &= 0. \end{aligned} \tag{2.9}$$

Für den Prädiktor-Schritt gilt also entweder die Bedingung von (1.23), oder

$$\begin{aligned}
\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k) &= \Phi(x^k + \Delta x^k, y^k + \Delta y^k, \eta_k \mu_k) \\
&\leq \Phi(x^k, y^k, \mu_k) + \nabla \Phi(x^k, y^k, \mu_k)^T \begin{bmatrix} \Delta x^k \\ \Delta y^k \\ (n_k - 1)\mu_k \end{bmatrix} \\
&= \Phi(x^k, y^k, \mu_k) + \nabla \Phi(x^k, y^k, \mu_k)^T \begin{bmatrix} \Delta x^k \\ \Delta y^k \\ -\mu_k \end{bmatrix} + \eta_k \mu_k \nabla_\mu \Phi(x^k, y^k, \mu_k) \\
&= \eta_k \mu_k \nabla_\mu \Phi(x^k, y^k, \mu_k) \\
&< 0
\end{aligned}$$

wegen *Lemma 1.5.* (vi) und damit in beiden Fällen $\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k) \leq 0$.
Für den Korrektor-Schritt in (1.28) gilt wiederum

$$\begin{bmatrix} 0 \\ \Phi(\hat{x}^k, \hat{y}^k, \hat{\mu_k}) \\ \hat{\mu_k} \end{bmatrix} + \begin{bmatrix} M & -I & 0 \\ \nabla_x \Phi_{\hat{k}} & \nabla_y \Phi_{\hat{k}} & \nabla_\mu \Phi_{\hat{k}} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \Delta \hat{x}^k \\ \Delta \hat{y}^k \\ \Delta \hat{\mu}_k \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ (1 - \bar{\sigma})\hat{\mu} \end{bmatrix}.$$

Aus der dritten Blockzeile erhält man diesmal $\Delta \hat{\mu}_k = -\hat{\sigma} \hat{\mu}_k$ und durch Einsetzen dessen in die zweite Blockzeile damit das Block-Gleichungssystem

$$\begin{aligned}
M \Delta \hat{x}^k - \Delta \hat{y}^k &= 0 \\
\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu_k}) + \nabla \Phi(\hat{x}^k, \hat{y}^k, \hat{\mu_k})^T \begin{bmatrix} \Delta \hat{x}^k \\ \Delta \hat{y}^k \\ -\bar{\sigma} \hat{\mu}_k \end{bmatrix} &= 0.
\end{aligned} \tag{2.10}$$

Ähnlich zu oben bringt uns dies für den Korrektor-Schritt

$$\begin{aligned}
\Phi(x^{k+1}, y^{k+1}, \mu_{k+1}) &= \Phi(\hat{x}^k + \hat{\lambda}_k \Delta \hat{x}^k, \hat{y}^k + \hat{\lambda}_k \Delta \hat{y}^k, \hat{\mu}_k + \hat{\lambda}_k \Delta \hat{\mu}_k) \\
&\leq \Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k) + \hat{\lambda}_k \nabla \Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k)^T \begin{bmatrix} \Delta \hat{x}^k \\ \Delta \hat{y}^k \\ -\bar{\sigma} \hat{\mu}_k \end{bmatrix} \\
&= (1 - \hat{\lambda}_k) \Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k) \\
&\leq 0,
\end{aligned}$$

da wir $\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k) \leq 0$ schon gezeigt haben. □

2.2 Globale lineare Konvergenz

Für den Beweis der globalen linearen Konvergenz benötigen wir eine weitere Annahme, die wir an den schon im Beweis von *Lemma 2.1.* in (2.3) betrachteten Hauptminor zweiter Ordnung der Jacobi-Matrix stellen.

Annahme (B)

Es seien $\mu_0 > 0$ und $\beta > 0$. Es existiert eine Konstante $C > 0$, so dass

$$\left\| \nabla \bar{F}_\mu(x,y)^{-1} \right\|_2 \leq C \tag{2.11}$$

für alle $0 < \mu \leq \mu_0$ und $(x,y) \in \mathcal{N}(\beta,\mu)$.

Auf den ersten Blick wirkt diese Annahme deutlich unintuitiver als Annahme (A). In Burke und Xu [8] wird die Existenz einer solchen Schranke, unter bestimmten Voraussetzungen an M - den sogenannten FLP-Bedingungen in Fukushima, Luo und Pang [9] - gezeigt.
Darüberhinaus zeigen wir in *Lemma 2.3.* und *Lemma 2.4.* wie die eindeutige Lösbarkeit von $LCP(q,M)$ in Verbindung zu Annahme (B) steht.

Lemma 2.3. *Es gelten Annahme (A) und Annahme (B). Dann besitzt $LCP(q,M)$ eine eindeutige Lösung.*

Beweis. Wir bezeichnen mit $SOL(q,M)$ die Menge der Lösungen von $LCP(q,M)$. Annahme (A) in Verbindung mit *Satz 1.7.* gewährleistet, dass $SOL(q,M)$ nichtleer ist.
Seien nun also $(x_1,y_1),(x_2,y_2) \in SOL(q,M)$ zwei verschiedene Lösungen von $LCP(q,M)$. Wir betrachten die Konvexkombinationen

$$x_\lambda := \lambda x_1 + (1-\lambda)x_2 \geq 0, \quad y_\lambda := \lambda y_1 + (1-\lambda)y_2 \geq 0$$

für $\lambda \in [0,1]$. Für diese gilt

$$\begin{aligned} Mx_\lambda - y_\lambda + q &= M(\lambda x_1 + (1-\lambda)x_2) - (\lambda y_1 + (1-\lambda)y_2) + \lambda q + (1-\lambda)q \\ &= \lambda(Mx_1 - y_1 + q) + (1-\lambda)(Mx_2 - y_2 + q) \\ &= 0. \end{aligned}$$

Mit der Beweisidee von Theorem 3.1.7 in Cottle, Pang und Stone [2] erhalten wir für die Komplementaritätsbedingung, wegen der Monotonie von $LCP(q,M)$ und der damit verbundenen positiven Semi-Definitheit von M

$$0 \leq (x_1 - x_2)^T M(x_1 - x_2) = -x_1^T y_2 - x_2^T y_1 \leq 0$$

durch $y_1 - y_2 = M(x_1 - x_2)$. Es folgt $x_1^T y_2 = x_2^T y_1 = 0$ und damit

$$x_\lambda^T y_\lambda = \lambda^2 x_1^T y_1 + \lambda(1-\lambda)x_1^T y_2 + (1-\lambda)\lambda x_2^T y_1 + (1-\lambda)^2 x_2^T y_2 = 0.$$

Mit *Definition 1.1.* und (1.2) folgt $(x_\lambda, y_\lambda) \in SOL(q, M)$ und damit, dass $SOL(q, M)$ konvex und damit zusammenhängend sein muss.
Annahme (B) impliziert wiederum die Regularität eines jeden Elements des (in Qi [10] Seite 233) sogenannten B-Subdifferentials von F in jedem Punkt von $SOL(q, M)$. Mit Proposition 2.5 in Qi [10] folgt damit, dass $SOL(q, M)$ nur aus isolierten Punkten besteht. Da $SOL(q, M)$ konvex ist, kann es also nur ein Element besitzen und die Lösung von $LCP(q, M)$ muss eindeutig sein. □

Wissen wir auf der anderen Seite bereits von der Eindeutigkeit der Lösung von $LCP(q, M)$, so können wir unter bestimmten Voraussetzungen dadurch die Schranke in (2.11) implizieren.

Lemma 2.4. *Es existiere eine eindeutige Lösung $(\bar{x}, \bar{y})$ von $LCP(q, M)$, die die strenge Komplementaritätsbedingung $\bar{x} + \bar{y} > 0$ erfüllt. Dann existiert der Grenzwert*

$$L := \lim_{(x,y,\mu)\to(\bar{x},\bar{y},0)} \nabla \bar{F}_\mu(x, y)^T$$

und ist regulär. Wenn zusätzlich Annahme (A) erfüllt ist, dann muss für beliebige $\beta > 0$ und $\mu_0 > 0$ auch Annahme (B) erfüllt sein.

Beweis. Wir setzen $J_x := \{j : \bar{x}_j > 0\}$ und $J_y := \{j : \bar{y}_j > 0\}$. Wegen $\bar{x}^T\bar{y} = 0$ und $\bar{x} + \bar{y} > 0$ bilden J_x und J_y eine Partition der Indizes $\{1, 2, ..., n\}$. Weiter definieren wir die Diagonalmatrizen $D_x := 2\text{diag}(e_{J_y})$ und $D_y := 2\text{diag}(e_{J_x})$, wobei $e_K \in \mathbb{R}^n$ der Vektor ist, dessen Komponenten $(e_K)_i = 1$ sind, wenn $i \in K$ und $(e_K)_i = 0$ sind, wenn $i \notin K$. Einsetzen in (1.10) ergibt, wegen $\bar{x}^T\bar{y} = 0$ und $\bar{x}+\bar{y} > 0$, mit den Definitionen aus *Lemma 2.1.* für $\nabla \bar{F}_\mu(x, y)^T$ in (2.3)

$$\nabla_x\Phi \to D_x \;\text{ und }\; \nabla_y\Phi \to D_y$$

für $(x, y, \mu) \to (\bar{x}, \bar{y}, 0)$. Folglich konvergiert $\nabla \bar{F}_\mu(x, y)^T$ gegen den Grenzwert

$$L := \begin{bmatrix} M & -I \\ D_x & D_y \end{bmatrix}$$

für $(x, y, \mu) \to (\bar{x}, \bar{y}, 0)$.
Angenommen es existieren $(u, v) \in \mathbb{R}^n \times \mathbb{R}^n$, so dass

$$L \begin{bmatrix} u \\ v \end{bmatrix} = 0$$

und damit

$$Mu = v, \tag{2.12}$$
$$D_x u + D_y v = 0 \tag{2.13}$$

gilt. Wegen (2.13) folgt $u_j = 0$ für $\bar{y}_j > 0$ und $v_j = 0$ für $\bar{x}_j > 0$. Mit $J_x \cap J_y = \emptyset$ folgt für beliebige $\lambda \in \mathbb{R}$, dass

$$(\bar{x} + \lambda u)^T(\bar{y} + \lambda v) = \bar{x}^T\bar{y} + \lambda\bar{x}^T v + \lambda u^T\bar{y} + \lambda^2 u^T v = 0 \tag{2.14}$$

gelten muss. Weiter impliziert die strenge Komplementarität aus Stetigkeitsgründen für alle ausreichend kleinen λ aber auch

$$(\bar{x} + \lambda u) \geq 0 \quad \text{und} \quad (\bar{y} + \lambda v) \geq 0. \tag{2.15}$$

Zusätzlich gilt

$$M(\bar{x} + \lambda u) + q = (\bar{y} + \lambda v) \tag{2.16}$$

wegen (2.12). Mit (2.14), (2.15) und (2.16) ist $(\bar{x} + \lambda u, \bar{y} + \lambda v)$ also eine Lösung von $LCP(q, M)$. Da diese aber nach Voraussetzung eindeutig und λ beliebig ist, muss $(u, v) = 0$ gelten und damit L regulär sein.
Die letzte Aussage von *Lemma 2.4.* folgt aus der in *Lemma 1.6.* gezeigten Beschränktheit von $\bigcup_{0<\mu\leq\mu_0} \mathcal{N}(\beta, \mu)$, der Stetigkeit und der von uns in *Lemma 2.1.* gezeigten Regularität von $\nabla \bar{F}_\mu(x, y)^T$ und der in diesem Beweis gezeigten Existenz und Regularität des Grenzwerts L. □

Wir sind nun in der Lage die Hauptaussage dieser Analyse zu treffen - die **globale lineare Konvergenz** des Algorithmus - und diese zu belegen. Das Resultat des folgenden Satzes beruht dabei einzig und allein auf dem Korrektor-Schritt und ist komplett unabhängig davon, ob der Prädiktor-Schritt in irgendeiner Iteration angenommen wird.

Satz 2.5. *(Globale lineare Konvergenz) Es gelten Annahme (A) und Annahme (B). Es sei* $\{(x^k, y^k, \mu_k)\}$ *die von Algorithmus 1 erzeugte Iterationsfolge. Wenn der Algorithmus nicht nach endlich vielen Schritten gegen die Lösung von* $LCP(q, M)$ *konvergiert, dann gilt für* $k = 0, 1, ...$

$$(x^k, y^k) \in \mathcal{N}(\beta, \mu_k), \tag{2.17}$$

$$(1 - \bar{\sigma}\hat{\lambda}_{k-1})\eta_{k-1} \cdots (1 - \bar{\sigma}\hat{\lambda}_0)\eta_0\mu_0 = \mu_k, \tag{2.18}$$

mit

$$\hat{\lambda}_k \geq \bar{\lambda} := \min\left\{1, \frac{\alpha_2(1 - \bar{\sigma})\beta}{C^2(\beta + 2\sqrt{n}\bar{\sigma})^2 + \sqrt{n}\bar{\sigma}^2 + \bar{\sigma}(1 - \bar{\sigma})\beta}\right\}, \tag{2.19}$$

wobei C *die Konstante aus* (2.11) *ist. Damit gilt, dass* μ_k *linear gegen* 0 *geht. Die Iterationsfolge* $\{(x^k, y^k)\}$ *konvergiert gegen die eindeutige Lösung von* $LCP(q, M)$.

Beweis. (2.17) wurde bereits in *Satz 2.2.* gezeigt und (2.18) folgt direkt aus der Konstruktion des Algorithmus.
Wegen *Lemma 1.5.* (ii) erhalten wir für $i = 1, 2, ..., n$ und $\lambda \in [0, 1]$, durch komponentenweise Taylor-Entwicklung, die Gleichheit

$$\begin{aligned} \left|\phi(\hat{x}_i^k + \lambda\Delta\hat{x}_i^k, \hat{y}_i^k + \lambda\Delta\hat{y}_i^k, (1 - \bar{\sigma})\hat{\mu}_k)\right| = \Bigg|\phi(\hat{x}_i^k, \hat{y}_i^k, \hat{\mu}_k) + \lambda\nabla\phi(\hat{x}_i^k, \hat{y}_i^k, \hat{\mu}_k)^T \begin{bmatrix} \Delta\hat{x}_i^k \\ \Delta\hat{y}_i^k \\ -\bar{\sigma}\hat{\mu}_k \end{bmatrix} \\ + \frac{\lambda^2}{2} \begin{bmatrix} \Delta\hat{x}_i^k \\ \Delta\hat{y}_i^k \\ -\bar{\sigma}\hat{\mu}_k \end{bmatrix}^T \nabla^2\phi(\hat{x}_i^k + \theta_i\lambda\Delta\hat{x}_i^k, \hat{y}_i^k + \theta_i\lambda\Delta\hat{y}_i^k, (1 - \theta_i\bar{\sigma})\hat{\mu}_k) \begin{bmatrix} \Delta\hat{x}_i^k \\ \Delta\hat{y}_i^k \\ -\bar{\sigma}\hat{\mu}_k \end{bmatrix}\Bigg| \end{aligned} \tag{2.20}$$

für ein $\theta_i \in [0,1]$. Weiter erhalten wir zuerst durch die komponentenweise Gleichheit in (2.10) für den vorderen Term, in Verbindung mit der Cauchy-Schwarz-Formel und der Submultiplikativität der Spektralnorm für den hinteren Term und danach durch *Lemma 1.5.* (v), die Abschätzungen

$$\begin{aligned}
&\Bigg|\phi(\hat{x}_i^k, \hat{y}_i^k, \hat{\mu}_k) + \lambda \nabla \phi(\hat{x}_i^k, \hat{y}_i^k, \hat{\mu}_k)^T \begin{bmatrix} \Delta \hat{x}_i^k \\ \Delta \hat{y}_i^k \\ -\bar{\sigma}\hat{\mu}_k \end{bmatrix} \\
&+ \frac{\lambda^2}{2} \begin{bmatrix} \Delta \hat{x}_i^k \\ \Delta \hat{y}_i^k \\ -\bar{\sigma}\hat{\mu}_k \end{bmatrix}^T \nabla^2 \phi(\hat{x}_i^k + \theta_i \lambda \Delta \hat{x}_i^k, \hat{y}_i^k + \theta_i \lambda \Delta \hat{y}_i^k, (1-\theta_i \bar{\sigma})\hat{\mu}_k) \begin{bmatrix} \Delta \hat{x}_i^k \\ \Delta \hat{y}_i^k \\ -\bar{\sigma}\hat{\mu}_k \end{bmatrix} \Bigg| \qquad (2.21) \\
&\leq (1-\lambda) \left|\phi(\hat{x}_i^k, \hat{y}_i^k, \hat{\mu}_k)\right| + \frac{\lambda^2}{2} \left|\left|\nabla^2 \phi(\hat{x}_i^k + \theta_i \lambda \Delta \hat{x}_i^k, \hat{y}_i^k + \theta_i \lambda \Delta \hat{y}_i^k, (1-\theta_i \bar{\sigma})\hat{\mu}_k)\right|\right|_2 \left|\left| \begin{bmatrix} \Delta \hat{x}_i^k \\ \Delta \hat{y}_i^k \\ -\bar{\sigma}\hat{\mu}_k \end{bmatrix} \right|\right|_2^2 \\
&\leq (1-\lambda) \left|\phi(\hat{x}_i^k, \hat{y}_i^k, \hat{\mu}_k)\right| + \frac{\lambda^2}{1-\bar{\sigma}\hat{\mu}_k} \left|\left|(\Delta \hat{x}_i^k, \Delta \hat{y}_i^k, -\bar{\sigma}\hat{\mu}_k)\right|\right|_2^2 .
\end{aligned}$$

Es seien nun $\tilde{v}, t$ und $\hat{t}$ die Vektoren in $\mathbb{R}^n$, deren Komponenten durch $\tilde{v}_i := \left|\phi(\hat{x}_i^k, \hat{y}_i^k, \hat{\mu}_k)\right|$, $t_i := \left|\left|(\Delta \hat{x}_i^k, \Delta \hat{y}_i^k, -\bar{\sigma}\hat{\mu}_k)\right|\right|_2^2$ und $\hat{t}_i := \left|\left|(\Delta \hat{x}_i^k, \Delta \hat{y}_i^k)\right|\right|_2^2$ für $i = 1, 2, ..., n$ gegeben sind. Es folgt $t = \hat{t} + \bar{\sigma}^2 \hat{\mu}_k^2 e$ für $e = (1, 1, ..., 1)^T$ und damit

$$\begin{aligned}
& \left|\left|\Phi(\hat{x}_i^k + \lambda \Delta \hat{x}_i^k, \hat{y}_i^k + \lambda \Delta \hat{y}_i^k, (1-\bar{\sigma})\hat{\mu}_k)\right|\right|_2 \\
\leq\ & \left|\left|(1-\lambda)\tilde{v} + \frac{\lambda^2}{(1-\bar{\sigma}\lambda)\hat{\mu}_k} t\right|\right|_2 \\
\leq\ & (1-\lambda) \left|\left|\tilde{v}\right|\right|_2 + \frac{\lambda^2}{(1-\bar{\sigma}\lambda)\hat{\mu}_k} \left|\left|t\right|\right|_2 \\
=\ & (1-\lambda) \left|\left|\tilde{v}\right|\right|_2 + \frac{\lambda^2}{(1-\bar{\sigma}\lambda)\hat{\mu}_k} \left|\left|\hat{t} + \bar{\sigma}^2 \hat{\mu}_k^2 e\right|\right|_2 \\
\leq\ & (1-\lambda) \left|\left|\tilde{v}\right|\right|_2 + \frac{\lambda^2}{(1-\bar{\sigma}\lambda)\hat{\mu}_k} \left(\left|\left|\hat{t}\right|\right|_2 + \bar{\sigma}^2 \hat{\mu}_k^2 \left|\left|e\right|\right|_2\right) \\
=\ & (1-\lambda) \left|\left|\tilde{v}\right|\right|_2 + \frac{\lambda^2}{(1-\bar{\sigma}\lambda)\hat{\mu}_k} \left(\left|\left|\hat{t}\right|\right|_2 + \sqrt{n}\bar{\sigma}^2 \hat{\mu}_k^2\right) \\
\leq\ & (1-\lambda) \left|\left|\tilde{v}\right|\right|_2 + \frac{\lambda^2}{(1-\bar{\sigma}\lambda)\hat{\mu}_k} \left(\left|\left|\hat{t}\right|\right|_1 + \sqrt{n}\bar{\sigma}^2 \hat{\mu}_k^2\right) \\
=\ & (1-\lambda) \left|\left|\Phi(\hat{x}_i^k, \hat{y}_i^k, \hat{\mu}_k)\right|\right|_2 + \frac{\lambda^2}{(1-\bar{\sigma}\lambda)\hat{\mu}_k} \left(\left|\left| \begin{bmatrix} \Delta \hat{x}^k \\ \Delta \hat{y}^k \end{bmatrix} \right|\right|_2^2 + \sqrt{n}\bar{\sigma}^2 \hat{\mu}_k^2\right) \\
\leq\ & (1-\lambda)\beta\hat{\mu}_k + \frac{\lambda^2}{(1-\bar{\sigma}\lambda)\hat{\mu}_k} \left(C^2 \left(\beta + 2\sqrt{n}\bar{\sigma}\right)^2 + \sqrt{n}\bar{\sigma}^2\right) \hat{\mu}_k,
\end{aligned}$$

wobei die letzte Ungleichheit aus der Abschätzung

$$\begin{aligned}
\left|\left| \begin{bmatrix} \Delta \hat{x}^k \\ \Delta \hat{y}^k \end{bmatrix} \right|\right|_2 &\leq \left|\left|\nabla \bar{F}_{\hat{\mu}_k}(\hat{x}^k, \hat{y}^k)^{-1}\right|\right|_2 \left(\left|\left|\Phi(\hat{x}_i^k, \hat{y}_i^k, \hat{\mu}_k)\right|\right|_2 + \bar{\sigma}\hat{\mu}_k \left|\left|\nabla_\mu \Phi_{\hat{k}}\right|\right|_2\right) \\
&\leq C\left(\beta + 2\sqrt{n}\bar{\sigma}\right)\hat{\mu}_k \qquad (2.22)
\end{aligned}$$

folgt. Diese erhält man durch Isolieren von $\nabla \bar{F}_{\hat{\mu}_k}(\hat{x}^k, \hat{y}^k)^T$ in (2.10)

$$\nabla \bar{F}_{\hat{\mu}_k}(\hat{x}^k, \hat{y}^k)^T \begin{bmatrix} \Delta\hat{x}^k \\ \Delta\hat{y}^k \end{bmatrix} = \begin{bmatrix} 0 \\ -\Phi(\hat{x}^k, \hat{y}^k, \hat{\mu}_k) + \bar{\sigma}\hat{\mu}_k \nabla_\mu \Phi_{\hat{k}} \end{bmatrix}$$

und Anwenden von *Lemma 1.5.* (vi). Es lässt sich durch elementare Umformungen verifizieren, dass

$$(1-\lambda)\beta\hat{\mu}_k + \frac{\lambda^2}{(1-\bar{\sigma}\lambda)\hat{\mu}_k}\left(C^2\left(\beta + 2\sqrt{n}\bar{\sigma}\right)^2 + \sqrt{n}\bar{\sigma}^2\right)\hat{\mu}_k \leq (1-\bar{\sigma}\lambda)\beta\hat{\mu}_k$$

erfüllt sein muss, wenn

$$\lambda \leq \frac{(1-\bar{\sigma})\beta}{C^2\left(\beta + 2\sqrt{n}\bar{\sigma}\right)^2 + \sqrt{n}\bar{\sigma}^2 + \bar{\sigma}(1-\bar{\sigma})\beta}$$

gilt. Damit erhalten wir

$$\hat{\lambda}_k \geq \min\left\{1,\ \frac{\alpha_2(1-\bar{\sigma})\beta}{C^2\left(\beta + 2\sqrt{n}\bar{\sigma}\right)^2 + \sqrt{n}\bar{\sigma}^2 + \bar{\sigma}(1-\bar{\sigma})\beta}\right\}.$$

Zuletzt halten wir fest, dass die Iterationsfolge $\{(x^k, y^k)\}$ durch *Satz 2.2.* in Verbindung mit *Lemma 1.6.* beschränkt ist. Mit der Argumentation für (2.22) erhalten wir zusätzlich die Beschränkungen

$$\left\|\begin{bmatrix} \Delta x^k \\ \Delta y^k \end{bmatrix}\right\|_2 \leq C\left(\beta + 2\sqrt{n}\right)\hat{\mu}_k \quad \text{und} \quad \left\|\begin{bmatrix} \Delta\hat{x}^k \\ \Delta\hat{y}^k \end{bmatrix}\right\|_2 \leq C\left(\beta + 2\sqrt{n}\right)\hat{\mu}_k$$

wegen $0 < \bar{\sigma} < 1$ und $0 < \eta_k \leq 1$ für alle k. Mit den beiden schon gezeigten Eigenschaften (2.18) und (2.19) erhalten wir damit

$$\begin{aligned} \left\|(x^{k+1}, y^{k+1}) - (x^k, y^k)\right\|_2 &\leq \left\|\begin{bmatrix} \Delta x^k \\ \Delta y^k \end{bmatrix}\right\|_2 + \hat{\lambda}_k \left\|\begin{bmatrix} \Delta\hat{x}^k \\ \Delta\hat{y}^k \end{bmatrix}\right\|_2 \\ &\leq 2C\left(\beta + 2\sqrt{n}\right)\hat{\mu}_k \\ &\leq C\left(\beta + 2\sqrt{n}\right)(1-\bar{\sigma}\bar{\lambda})^k\mu_0. \end{aligned}$$

Bei der Iterationsfolge $\{(x^k, y^k)\}$ handelt es sich also um eine Cauchy-Folge, diese muss gegen die eindeutige Lösung von $LCP(q, M)$ konvergieren. Die untere Schranke an die Schrittweite gewährleistet dabei die Linearität, die oben erwähnte Beschränktheit den Grenzwert. □

2.3 Lokal quadratische Konvergenz

An dieser Stelle können wir ohne weitere Hilfsmittel die lokal quadratische Konvergenz untersuchen, mit deren Beweis wir die theoretische Analyse abschließen wollen.

Satz 2.6. *(Lokal quadratische Konvergenz) Es gelte Annahme (B). Es sei $\{(x^k, y^k, \mu_k)\}$ die von Algorithmus 1 erzeugte Iterationsfolge, die gegen $\{(x^*, y^*, 0)\}$ konvergiert, wobei $\{(x^*, y^*)\}$ die eindeutige Lösung von $LCP(q, M)$ ist. Ist außerdem die strenge Komplementaritätsbedingung $x^* + y^* > 0$ erfüllt, so gilt*

$$\mu_{k+1} = \mathcal{O}(\mu_k^2),$$

also konvergiert μ_k quadratisch gegen 0.

Beweis. Die strenge Komplementaritätsbedingung $x^* + y^* > 0$ gewährleistet in Verbindung mit der linken Ungleichung in *Lemma 1.5.* (v), dass für Iterierte (x^k, y^k, μ_k) nah an der Lösung Konstanten $\epsilon > 0$ und $L > 0$ existieren, so dass für $i = 1, 2, ..., n$

$$\left|\left|\nabla^2 \phi(x_i^k, y_i^k, \mu_k)\right|\right|_2 \leq L \tag{2.23}$$

gelten muss, sobald

$$\left|\left|(x^k, y^k, \mu_k) - (x^*, y^*, 0)\right|\right|_2 \leq \epsilon$$

gilt. Für k hinreichend groß und $\eta \in (0, 1]$ erhalten wir damit

$$\begin{aligned}
&\left|\phi(x_i^k + \Delta x_i^k, y_i^k + \Delta y_i^k, \eta\mu_k)\right| \leq \left|\phi(x_i^k, y_i^k, \mu_k) + \nabla\phi(x_i^k, y_i^k, \mu_k)^T \begin{bmatrix} \Delta x_i^k \\ \Delta y_i^k \\ (\eta - 1)\mu_k \end{bmatrix}\right| \\
&+ \frac{1}{2}\left|\begin{bmatrix} \Delta x_i^k \\ \Delta y_i^k \\ (\eta - 1)\mu_k \end{bmatrix}^T \nabla^2\phi(x_i^k + \theta_i \Delta x_i^k, y_i^k + \theta_i \Delta y_i^k, (1 + \theta_i(\eta - 1))\mu_k) \begin{bmatrix} \Delta x_i^k \\ \Delta y_i^k \\ (\eta - 1)\mu_k \end{bmatrix}\right| \\
&= \left|\phi(x_i^k, y_i^k, \mu_k) + \nabla\phi(x_i^k, y_i^k, \mu_k)^T \begin{bmatrix} \Delta x_i^k \\ \Delta y_i^k \\ -\mu_k \end{bmatrix} + \eta\mu_k \nabla_\mu \phi(x_i^k, y_i^k \mu_k)\right| \\
&+ \frac{1}{2}\left|\begin{bmatrix} \Delta x_i^k \\ \Delta y_i^k \\ (\eta - 1)\mu_k \end{bmatrix}^T \nabla^2\phi(x_i^k + \theta_i \Delta x_i^k, y_i^k + \theta_i \Delta y_i^k, (1 + \theta_i(\eta - 1))\mu_k) \begin{bmatrix} \Delta x_i^k \\ \Delta y_i^k \\ (\eta - 1)\mu_k \end{bmatrix}\right| \\
&\leq \eta\mu_k \left|\nabla_\mu \phi(x_i^k, y_i^k, \mu_k)\right| + \frac{L}{2}\left(| \left|(\Delta x_i^k, \Delta y_i^k, (\eta - 1)\mu_k)\right|\right|_2^2 \\
&\leq 2\eta\mu_k + \frac{L}{2}\left(\left|\left|(\Delta x_i^k, \Delta y_i^k)\right|\right|_2^2 + (1 - \eta)^2 \mu_k^2\right)
\end{aligned}$$

für ein $\theta_i > 0$. Die erste Ungleichung folgt hierbei aus der Dreiecksungleichung in Verbindung mit der Gleichheit in (2.20) für $\lambda = 1$ und $\eta = (1 - \bar{\sigma})$. Die vorletzte Ungleichung folgt aus der komponentenweisen Gleichheit in (2.9) und die letzte Ungleichung aus *Lemma 1.5.* (vi). Ähnliche Argumentation wie in (2.22) ergibt für den Prädiktor-Schritt

$$\left\| \begin{bmatrix} \Delta x^k \\ \Delta y^k \end{bmatrix} \right\|_2 \leq C \left(\beta + 2\sqrt{n}\right) \hat{\mu}_k$$

und damit in Verbindung mit der komponentenweisen Abschätzung oben

$$\begin{aligned} \left\|\Phi(x^k + \Delta x^k, y^k + \Delta y^k, \eta\mu_k)\right\|_2 &\leq 2\sqrt{n}\eta\mu_k + \frac{L}{2}\left(C^2\left(\beta + 2\sqrt{n}\right)^2 + \sqrt{n}(1-\eta)^2\right)\mu_k^2 \\ &\leq 2\sqrt{n}\eta\mu_k + \frac{L}{2}\left(C^2\left(\beta + 2\sqrt{n}\right)^2 + \sqrt{n}\right)\mu_k^2. \end{aligned} \tag{2.24}$$

Weiter gilt wegen $\beta > 2\sqrt{n}$ also

$$\begin{aligned} \left\|\Phi(x^k + \Delta x^k, y^k + \Delta y^k, \mu_k)\right\|_2 &\leq \left(2\sqrt{n} + \mu_k \frac{L}{2}\left(C^2\left(\beta + 2\sqrt{n}\right)^2 + \sqrt{n}\right)\right)\mu_k \\ &\leq \beta\mu_k \end{aligned}$$

für k hinreichend groß und damit μ_k hinreichend klein. Dies entspricht gerade Ungleichung (1.24) im Prädiktor-Schritt für $t = 0$, also $\eta = 1$.
Es lässt sich durch Elementare Umformungen verifizieren, dass

$$2\sqrt{n}\eta\mu_k + \frac{L}{2}\left(C^2\left(\beta + 2\sqrt{n}\right)^2 + \sqrt{n}\right)\mu_k^2 \leq \eta\beta\mu_k$$

gelten muss, wenn

$$\eta \geq \frac{L}{2(\beta - 2\sqrt{n})}\left(C^2\left(\beta + 2\sqrt{n}\right)^2 + \sqrt{n}\right)\mu_k$$

gilt. Es folgt

$$\alpha_1\eta_k \leq \frac{L}{2(\beta - 2\sqrt{n})}\left(C^2\left(\beta + 2\sqrt{n}\right)^2 + \sqrt{n}\right)\mu_k$$

und damit

$$\eta_k \leq \frac{L}{2\alpha_1(\beta - 2\sqrt{n})}\left(C^2\left(\beta + 2\sqrt{n}\right)^2 + \sqrt{n}\right)\mu_k$$

für k hinreichend groß. Mit den Iterationsvorschriften (1.26) und (1.28) erhalten wir

$$\mu_{k+1} = \mathcal{O}(\mu_k^2).$$

$\square$

3 Praktische Anwendung für lineare Programme

In diesem Kapitel untersuchen wir eine Variation von Algorithmus 1 für lineare Programme, die in Python implementiert und getestet wurde. Die durchgeführten Modifikationen sind dabei großteils heuristischer Natur und werden nicht mathematisch belegt. Stattdessen werden wir die Eigenschaften des Algorithmus am Ende des Kapitels durch das Anwenden auf eine große Testproblem-Sammlung untersuchen.

3.1 Implementation in Python

Es wäre naheliegend das in (1.6) gewonnene Verhältnis zwischen $LCP(q, M)$ und $QP(c, Q, A, b)$

$$q = \begin{bmatrix} c \\ -b \end{bmatrix},\ M = \begin{bmatrix} Q & -A^T \\ A & 0 \end{bmatrix}$$

für $Q = 0$ zu nutzen und den Algorithmus für LCP zu implementieren, wie er in Kapitel 1 formuliert wurde. Dies würde allerdings zu unnötigen Nicht-Negativitätsbedingungen und damit komplexeren Gleichungssystemen führen.
Wir folgen stattdessen dem für Innere-Punkte-Verfahren üblichen Ansatz und betrachten dafür das sogenannte lineare Programm in Standardform

$$\begin{aligned} \min f(x) &= c^T x \\ \text{u. d. N.}\quad Ax &= b, \\ x &\geq 0 \end{aligned} \tag{3.1}$$

mit gegebenen Parametern $c \in \mathbb{R}^n$, $b \in \mathbb{R}^m$ und $A \in \mathbb{R}^{m \times n}$. Beispielsweise in Wright [5] wird gezeigt, dass für $\lambda \in \mathbb{R}^m$ und $s \in \mathbb{R}^n$ die Optimalitätsbedingungen

$$\begin{aligned} A^T\lambda + s &= c, \\ Ax &= b, \\ x_i \geq 0,\ s_i \geq 0, x_i s_i &= 0 \qquad \forall i = 1, 2, ..., n \end{aligned} \tag{3.2}$$

genau dann eine Lösung besitzen, wenn (3.1) eine Lösung besitzt. Ähnlich zu Kapitel 1 betrachten wir auch hier das gestörte Problem. Die sogenannten zentralen Pfad-Bedingungen

$$\begin{aligned} A^T\lambda + s &= c, \\ Ax &= b, \\ x_i \geq 0,\ s_i \geq 0, x_i s_i &= \mu^2 \qquad \forall i = 1, 2, ..., n \end{aligned} \tag{3.3}$$

für $\mu > 0$ liefern uns für lineare Programme in Standardform mit der gleichen Argumentation wie in (1.11)

$$F(x, \lambda, s, \mu) := \begin{bmatrix} A^T\lambda + s - c \\ Ax - b \\ \Phi(x, s, \mu) \\ \mu \end{bmatrix} \tag{3.4}$$

mit Φ wie in (1.12). Mit diesem Ansatz lässt sich der folgende Algorithmus formulieren.

Algorithm 2:

1 **Schritt 0: (Initialisierung)** Setze $k := 0$.
2 Wähle $(x^0, \lambda^0, s^0) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n$, so dass $Ax^0 = b,\ A^T\lambda^0 + s^0 = c$ und μ_0, so dass $\Phi(x^0, s^0, \mu_0) < 0$.
3 Wähle β, so dass $||\Phi(x^0, s^0, \mu_0)||_2 \leq \beta\mu_0$.
4 Wähle $\epsilon > 0$ und $\bar{\sigma}$, α_1 und α_2 aus $(0, 1)$.
5 **Schritt 1: (Abbruch-Kriterium) if** $\mu_k < \epsilon^2$ **or** $||\Phi(x^k, y^k, 0)||_\infty < \epsilon$: STOP.
6 **Schritt 2: (Prädiktor-Schritt)** Bestimme $(\Delta x^k, \Delta\lambda^k, \Delta s^k, \Delta\mu_k)$ als Lösung des linearen Gleichungssystems

$$F(x^k, \lambda^k, s^k, \mu_k) + \nabla F(x^k, \lambda^k, s^k, \mu_k)^T \begin{bmatrix} \Delta x^k \\ \Delta\lambda^k \\ \Delta s^k \\ \Delta\mu_k \end{bmatrix} = 0. \tag{3.5}$$

7 **if** $||\Phi(x^k + \Delta x^k, s^k + \Delta s^k, 0)||_2 < \epsilon$, **then**
8 STOP;
9 **else if** $||\Phi(x^k + \Delta x^k, s^k + \Delta s^k, \mu_k)||_2 > \beta\mu_k$, **then**
10 setze

$$\hat{x}^k := x^k,\ \hat{\lambda}^k := \lambda^k,\ \hat{s}^k := s^k\ \hat{\mu}_k := \mu_k \text{ und } \eta_k := 1; \tag{3.6}$$

11 **else**
12 wähle $\eta_k = \alpha_1^l$, so dass l die nicht-negative ganze Zahl mit
$||\Phi(x^k + \Delta x^k, s^k + \Delta s^k, \alpha_1^j\mu_k)||_2 \leq \alpha_1^j\beta\mu_k$ für alle $j = 0, 1, ..., l$ und
$||\Phi(x^k + \Delta x^k, s^k + \Delta s^k, \alpha_1^{l+1}\mu_k)||_2 > \alpha_1^{l+1}\beta\mu_k$
13 ist und setze

$$\hat{x}^k := x^k + \Delta x^k,\ \hat{\lambda}^k := \lambda^k + \Delta\lambda^k,\ \hat{s}^k := s^k + \Delta s^k \text{ und } \hat{\mu}_k := \eta_k\mu_k; \tag{3.7}$$

14 **Schritt 3: (Korrektor-Schritt)** Bestimme $(\Delta\hat{x}^k, \Delta\hat{\lambda}^k, \Delta\hat{s}^k, \Delta\hat{\mu}_k)$ als Lösung des linearen Gleichungssystems

$$F(\hat{x}^k, \hat{\lambda}^k, \hat{s}^k, \hat{\mu}_k) + \nabla F(\hat{x}^k, \hat{\lambda}^k, \hat{s}^k, \hat{\mu}_k)^T \begin{bmatrix} \Delta\hat{x}^k \\ \Delta\hat{\lambda}^k \\ \Delta\hat{s}^k \\ \Delta\hat{\mu}_k \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ (1-\bar{\sigma})\hat{\mu}_k \end{bmatrix}. \tag{3.8}$$

15 Wähle $\hat{t}_k$ als Maximum der Werte $1, \alpha_2, \alpha_2^2, ...$, so dass $||\Phi(\hat{x}^k + \hat{t}_k\Delta\hat{x}^k, \hat{s}^k + \hat{t}_k\Delta\hat{s}^k, (1-\bar{\sigma}\hat{t}_k)\hat{\mu}_k)||_2 \leq (1-\bar{\sigma}\hat{t}_k)\beta\hat{\mu}_k$ und setze

$$x^{k+1} := \hat{x}^k + \hat{t}_k\Delta\hat{x}^k,\ \lambda^{k+1} := \hat{\lambda}^k + \hat{t}_k\Delta\hat{\lambda}^k,\ s^{k+1} := \hat{s}^k + \hat{t}_k\Delta\hat{s}^k$$
$$\text{und } \mu_{k+1} := (1-\bar{\sigma}\hat{t}_k)\hat{\mu}_k; \tag{3.9}$$

Schritt 4: (Update) Setze $k := k + 1$ und gehe zu Schritt 1 in Zeile 5.

Anmerkung (Lösbarkeit der linearen Gleichungssysteme). Für F wie in (3.4) erhalten wir für die Jacobi-Matrizen in (3.5) und (3.8)

$$\nabla F(x,\lambda,s,\mu)^T = \begin{bmatrix} 0 & A^T & I & 0 \\ A & 0 & 0 & 0 \\ \nabla_x\Phi & 0 & \nabla_s\Phi & \nabla_\mu\Phi \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{3.10}$$

Beispielsweise in Geiger und Kanzow [3] wird gezeigt, dass diese Matrix regulär ist, wenn A vollen Zeilenrang besitzt, da es sich bei $\nabla_x\Phi$ und $\nabla_s\Phi$ um positiv definite Diagonalmatrizen handelt. Es kann darüberhinaus gezeigt werden, dass die Komponenten Δx und Δs in (3.5) und (3.8) selbst dann eindeutig bestimmt sind, wenn A nicht vollen Zeilenrang besitzt, siehe Aufgabe 4.4. in Geiger und Kanzow [3].

Anmerkung (Effizientes Lösen der linearen Gleichungssysteme). Mit (3.10) erhalten wir die Block-Gleichungssysteme

$$\begin{bmatrix} 0 & A^T & I & 0 \\ A & 0 & 0 & 0 \\ \nabla_x\Phi_k & 0 & \nabla_s\Phi_k & \nabla_\mu\Phi_k \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \Delta x^k \\ \Delta\lambda^k \\ \Delta s^k \\ \Delta\mu_k \end{bmatrix} = \begin{bmatrix} -(A^T\lambda^k + s^k - c) \\ -(Ax^k - b) \\ -\Phi(x^k, s^k, \mu_k) \\ -\mu_k \end{bmatrix} \tag{3.11}$$

für den Prädiktor-Schritt in (3.5) und

$$\begin{bmatrix} 0 & A^T & I & 0 \\ A & 0 & 0 & 0 \\ \nabla_x\Phi_{\hat{k}} & 0 & \nabla_s\Phi_{\hat{k}} & \nabla_\mu\Phi_{\hat{k}} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \Delta\hat{x}^k \\ \Delta\hat{\lambda}^k \\ \Delta\hat{s}^k \\ \Delta\hat{\mu}_k \end{bmatrix} = \begin{bmatrix} -(A^T\hat{\lambda}^k + \hat{s}^k - c) \\ -(A\hat{x}^k - b) \\ -\Phi(\hat{x}^k, \hat{s}^k, \hat{\mu}) \\ -\hat{\mu} + (1-\bar{\sigma})\hat{\mu}_k \end{bmatrix} \tag{3.12}$$

für den Korrektor-Schritt in (3.8) mit

$$\nabla_x\Phi_k := \operatorname{diag}\left(\frac{\partial\phi(a,b,\mu_k)}{\partial a}\bigg|_{\substack{a=x_i^k\\ b=s_i^k}}\right),\ \nabla_s\Phi_k := \operatorname{diag}\left(\frac{\partial\phi(a,b,\mu_k)}{\partial b}\bigg|_{\substack{a=x_i^k\\ b=s_i^k}}\right),\ \nabla_\mu\Phi_k := \left(\frac{\partial\phi(a,b,\mu_k)}{\partial\mu_k}\bigg|_{\substack{a=x_i^k\\ b=s_i^k}}\right)^T,$$

wobei wir auch hier wieder mit $\hat{k}$ implizieren, dass $(\hat{x}^k, \hat{s}^k, \hat{\mu}_k)$ eingesetzt wurde. Ähnlich zu Kapitel 2, erhalten wir $\Delta\mu_k = -\mu_k$ und $\Delta\hat{\mu}_k = -\bar{\sigma}\hat{\mu}_k$, mit ähnlicher Argumentation wie in (2.8) und vereinfachter Notation also das Block-Gleichungssystem

$$\begin{bmatrix} 0 & A^T & I \\ A & 0 & 0 \\ \nabla_x\Phi & 0 & \nabla_s\Phi \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta\lambda \\ \Delta s \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -r_3 \end{bmatrix} \tag{3.13}$$

mit $-r_3 = -\Phi(x,s,\mu) + \mu\nabla_\mu\Phi$ in (3.11) und $-r_3 = -\Phi(x,s,\mu) + \mu\bar{\sigma}\nabla_\mu\Phi$ in (3.12). Das Gleichungssystem in (3.13) verfügt dabei über große Ähnlichkeit zu den Gleichungssystemen, die in Inneren-Punkte-Verfahren für lineare Programme gelöst werden. Wir folgen daher einem Lösungsansatz, der beispielsweise für das Prädiktor-Korrektor-Verfahren von Mehrotra verwendet wird. Wir betrachten dafür die einzelnen Block-Gleichungen

$$A^T\Delta\lambda + \Delta s = 0, \tag{3.14}$$

$$A\Delta x = 0, \tag{3.15}$$

$$\nabla_x\Phi\Delta x + \nabla_s\Phi\Delta s = -r_3. \tag{3.16}$$

Aus (3.16) folgt direkt

$$\Delta x = -\nabla_x \Phi^{-1}(\nabla_s \Phi \Delta s + r_3).$$

Einsetzen davon in (3.15) liefert

$$\begin{aligned} & -A\nabla_x \Phi^{-1}(\nabla_s \Phi \Delta s + r_3) = 0 \\ \iff \quad & -A\nabla_x \Phi^{-1}\nabla_s \Phi \Delta s = A\nabla_x \Phi^{-1} r_3. \end{aligned}$$

Aus (3.14) erhalten wir $\Delta s = -A^T \Delta\lambda$ und damit

$$(A\nabla_x \Phi^{-1}\nabla_s \Phi A^T)\Delta\lambda = A\nabla_x \Phi^{-1} r_3. \tag{3.17}$$

Wir gehen also folgendermaßen vor: Wir bestimmen $\Delta\lambda$ aus (3.17) mit einer sparse Cholesky-Zerlegung. Wir setzen $\Delta s := -A^T\Delta\lambda$ und $\Delta x := -\nabla_x\Phi^{-1}(\nabla_s\Phi\Delta s + r_3)$.
Angenommen der Zeilenrang $\text{rang}(A) = m$ sei voll, dann lösen wir ein symmetrisch positiv definites Gleichungssystem der Dimension m, statt eines im Allgemeinen indefiniten Gleichungssystems der Dimension $(n + m + n)$.

Anmerkung (Abgelehnter Prädiktor-Schritt). Wird in einer Iteration der Prädiktor-Schritt abgelehnt und es tritt (3.6) ein, so ist die linke Seite des Gleichungssystems für den Korrektor-Schritt in (3.8) identisch zu der im Prädiktor-Schritt. In diesem Fall ist also nur eine Matrix-Faktorisierung notwendig um Schritt 2 und Schritt 3 durchzuführen.

Anmerkung (Verbesserung der Robustheit). Damit wir das Gleichungssystem, wie oben beschrieben, mit einer Cholesky-Zerlegung lösen können, benötigen wir sowohl den vollen Zeilenrang von A, als auch die positive Definitheit der Diagonalmatrizen $\nabla_x\Phi$ und $\nabla_s\Phi$. Wie wir aus Kapitel 2 wissen, sind die Einträge der Diagonalmatrizen im Allgemeinen echt positiv, genauer sogar in $(0, 2)$ enthalten. Aufgrund von Maschinenungenauigkeit kann dies aber, besonders für kleine Werte von μ, problematisch zu gewährleisten sein. Darüberhinaus können wir bei den betrachteten Testproblemen nicht uneingeschränkt annehmen, dass A, selbst nach dem später noch besprochenen Preprocessing, vollen Zeilenrang hat.
Daher wurden zwei heuristische Änderungen vorgenommen: Wir beschränken die Diagonaleinträge von $\nabla_x\Phi$ und $\nabla_s\Phi$ nach unten und damit von $\nabla_x\Phi^{-1}$ nach oben. Ist die linke Seite von (3.17) dennoch nicht positiv definit, addieren wir ein Vielfaches der Einheitsmatrix auf unser Gleichungssystem. Der Vorfaktor startet dabei bei ϵ^2 und wird in einer Schleife um Faktor 10 erhöht, bis das Gleichungssystem positiv definit ist.

Anmerkung (Konvertierung auf Standardform). Die im nächsten Kapitel betrachteten Testprobleme verfügen über verschiedene Arten von Nebenbedingungen, die erstmal auf Standardform konvertiert werden müssen. Die Eingangsparameter der Testprobleme haben die folgende Form

$$\begin{aligned} & \min f(x) = \tilde{c}^T x \\ \text{u. d. N.} \quad & A_{\text{eq}} x = b_{\text{eq}}, \\ & A_{\text{ineq}} x \leq b_{\text{ineq}}, \\ & \text{lb} \leq x \leq \text{ub} \end{aligned} \tag{3.18}$$

mit gegebenen Parametern $\tilde{c} \in \mathbb{R}^n$, $b_{\text{eq}} \in \mathbb{R}^m$, $A_{\text{eq}} \in \mathbb{R}^{m\times n}$, $b_{\text{ineq}} \in \mathbb{R}^s$, $A_{\text{ineq}} \in \mathbb{R}^{s\times n}$, $\text{lb} \in \mathbb{R}^n$ und $\text{ub} \in \mathbb{R}^n$. Die Überführung in Standardform erfolgt folgendermaßen:

Die Ungleichungs-Restriktionen überführen wir, mit der Hilfe von Schlupfvariablen $\hat{x}$, in Gleichungsrestriktionen. Dabei führt uns

$$A_{\text{ineq}}x + I\hat{x} = b_{\text{ineq}}, \quad (x, \hat{x}) \geq 0$$

zu

$$\begin{bmatrix} A_{\text{eq}} & 0 \\ A_{\text{ineq}} & I \end{bmatrix} \begin{bmatrix} x \\ \hat{x} \end{bmatrix} = \begin{bmatrix} b_{\text{eq}} \\ b_{\text{ineq}} \end{bmatrix}, \quad (x, \hat{x}) \geq 0$$

mit $\hat{x} \in \mathbb{R}^s$. Wir setzen also

$$A := \begin{bmatrix} A_{\text{eq}} & 0 \\ A_{\text{ineq}} & I \end{bmatrix}, \quad b := \begin{bmatrix} b_{\text{eq}} \\ b_{\text{ineq}} \end{bmatrix} \quad \text{und} \quad c := \begin{bmatrix} \tilde{c} \\ 0 \end{bmatrix} \tag{3.19}$$

mit $A \in \mathbb{R}^{(m+s)\times(n+s)}$, $b \in \mathbb{R}^{(m+s)}$ und $c \in \mathbb{R}^{(n+s)}$.
Die Box-Restriktionen überführen wir, mit der Hilfe einer Fallunterscheidung, in Gleichungsrestriktionen und unterscheiden dabei 6 Fälle:

(i): $0 \leq x_i \leq \infty$	Entspricht bereits der Standardform.
(ii): $-\infty \leq x_i \leq \infty$	Erweitern von A durch Kopieren der i-ten Spalte mit negativem Vorzeichen für $x_i = x_{i+} - x_{i-}$. Auch c wird um $-c_i$ erweitert.
(iii): $lb_i \leq x_i \leq \infty$	Verschiebung $x_i' = x_i - \text{lb}_i$, wodurch es zu einer Anpassung von b um $-A_i \cdot \text{lb}_i$ kommt, wobei A_i die i-te Spalte von A ist.
(iv): $0 \leq x_i \leq \text{ub}_i$	Erweiterung von A_{ineq} und b_{ineq} um eine Zeile beziehungsweise einen Eintrag für $x_i \leq \text{ub}_i$. Auch c wird um eine Null erweitert.
(v): $-\infty \leq x_i \leq \text{ub}_i$	Vorzeichenwechsel der i-ten Spalte von A und des i-ten Eintrags von c. Danach wird wie bei (iii) für $-\text{ub}_i \leq x_i \leq \infty$ verfahren.
(vi): $\text{lb}_i \leq x_i \leq \text{ub}_i$	Verschiebung $x_i' = x_i - \text{lb}_i$, wodurch es zu einer Anpassung von b um $-A_i \cdot \text{lb}_i$ kommt, wobei A_i wieder die i-te Spalte von A ist. Danach wird wie bei (iv) für $0 \leq x_i' \leq \text{ub}_i - \text{lb}_i$ verfahren.

3.2 Numerische Ergebnisse

Algorithmus 2 wurde in Python 3.12.2 implementiert. Es wurden dabei importierte Methoden von SciPy [11] und NumPy [12] verwendet. Die sparse Cholesky-Zerlegung, die benutzt wurde, stammt von scikit-sparse [13]. Diese Faktorisierung merkt sich, bei Bestimmung der Startvektoren, die Struktur von A und verwendet diese, um effizient spätere Gleichungssysteme zu lösen, da $A(\nabla_x \Phi^{-1} \nabla_s \Phi)^{\frac{1}{2}}$ in jedem Schritt die gleiche, dünn besetzte Struktur wie A aufweist.
Die betrachteten Testprobleme entstammen der Netlib Bibliothek für lineare Programme [14]. Diese wurden von Matt Haberland [11] als SciPy Benchmark Files zu NumPy Arrays für Python umgewandelt. Bevor unsere oben beschriebene Routine zur Umwandlung in Standardform angewandt wird, behandeln wir die Probleme darüberhinaus mit einigen Preprocessing-Tools von SciPy.
Wir verwenden die Methode `presolve` von SciPy, die versucht, basierend auf Andersen und Andersen [15], beispielsweise Nullzeilen, duplizierte Zeilen oder Zeilen mit nur einem Eintrag zu entfernen. Zusätzlich hat `presolve` die Option `rr_method`, durch die Redundanzen, wie linear abhängige Zeilen, entfernt werden sollen. Dafür wird eine Methode

verwendet, die auf Andersen [16] beruht. In seltenen Fällen führt dieses Entfernen der Redundanzen allerdings zu einer Änderung der optimalen Lösung. Zusätzlich verwenden wir die Methode `autoscale`, die die Nebenbedingungen, basierend auf der *Equilibration Method* in Tomlin [17], skaliert. `autoscale` führte im Allgemeinen zu einer deutlichen Verschlechterung der Iterationszahlen. Diese Methode wurde daher nur dann getestet, wenn ein Testproblem sowieso schon die maximale Anzahl an Iterationen erreichte. In diesen einzelnen Fällen, führte es teilweise zu einer deutlichen Verbesserung.
Unsere Vorgehensweise beim Preprocessing für Tabelle 1 lässt dich daher wie folgt beschreiben: Standardmäßig wurde `presolve` inklusive `rr_method` angewandt und `autoscale` nicht durchgeführt. Wich der resultierende optimale Funktionswert dabei von dem, in der .readme der Netlib Datenbank [14] angegebenen, optimalen Funktionswert ab, wurde `rr_method` weggelassen. Erreichte ein Testproblem die maximale Anzahl an Iterationen, so wurde mit `autoscale` versucht, dies zu beheben. Gab es weiterhin Komplikationen bei einem Testproblem, wurde auf `presolve` komplett verzichtet.
Bei 'BANDM', 'BORE3D', 'D2Q06C', 'GANGES', 'GREENBEA', 'GREENBEB', 'GROW15', 'GROW22', 'MAROS-R7', 'MAROS', 'PILOTNOV', 'QAP12', 'QAP15', 'SCSD8', 'WOOD1P' und 'WOODW' wurde `rr_method` weggelassen. Bei 'PILOT-JA' und 'BNL2' wurde `presolve` gar nicht angewandt. Bei 'MODSZK1', 'PEROLD', 'PILOT', 'PILOT87' und 'STOCFOR3' wurde zusätzlich `autoscale` angewandt.
Nach erfolgreichem Preprocessing wird das lineare Programm von unserem Algorithmus, wie oben beschrieben, auf Standardform gebracht, wobei `autoscale` erst auf das Problem in Standardform angewandt wird. Danach werden die Startvektoren (x^0, λ^0, s^0) initialisiert: Wir bestimmen $AA^T x' = b$ mit einer sparse Cholesky-Zerlegung und setzen $x^0 := A^T x'$. Damit gilt $Ax^0 = b$. Wir lösen $AA^T \lambda^0 = Ac$ mit obiger Zerlegung und setzen $s^0 := c - A^T(\lambda^0)$. Damit gilt $A^T \lambda^0 + s^0 = c$. Desweiteren setzen wir

$$\mu_0 := \sqrt{\max\{x_i^0 y_i^0 \mid x_i^0 > 0,\ y_i^0 > 0\}} + \epsilon,$$

damit $\Phi(x^0, s^0, \mu_0) < 0$ gewährleistet ist. Der Algorithmus konvergierte auch ohne diese Bedingung, allerdings mit einer durchschnittlich höheren Anzahl an Iterationen. Wir setzen

$$\beta := \frac{||\Phi(x^0, y^0, \mu_0)||_2}{\mu_0},$$

da zusätzlich $\beta > 2\sqrt{n}$ nur für den Beweis der lokal quadratischen Konvergenz benötigt wurde und in unseren Tests für einen durchschnittlich schlechteren Verlauf gesorgt hat. Für die übrigen äußeren Parameter setzen wir

$$\epsilon := 10^{-4}, \quad \alpha_1 := 0.75, \quad \alpha_2 := 0.8 \;\text{ und }\; \bar{\sigma} := 0.5.$$

Außerdem beträgt die maximale Anzahl an Iterationen Maxiter = 1000. Führt der Algorithmus zu einem Fehler, wird er erneut mit einer niedrigeren maximalen Anzahl an Iterationen CrMaxiter = 100 und Genauigkeit 10^{-3} ausgeführt. Tabelle 1 beinhaltet die Testergebnisse mit obigen Einstellungen. Die Spalten haben dabei folgende Bedeutung:

Datei:	Name des Testproblems in der Netlib Datenbank,
Nonzeros:	Anzahl der in den Nebenbedingungen von 0 verschiedenen Einträge vor Preprocessing (entnommen der .readme der Netlib Datenbank),
Fun:	Optimaler Funktionswert nach Postprocessing (zum Vergleich mit Werten der .readme der Netlib Datenbank),
μ:	Wert von μ_{Iter} bei Terminierung,
$\vert\vert\Phi\vert\vert$:	Wert von $\left\vert\left\vert\Phi(x^{\text{Iter}}, s^{\text{Iter}}, \mu_{\text{Iter}})\right\vert\right\vert_{\infty}$ vor Postprocessing,
PS:	Anzahl der akzeptierten Prädiktor-Schritte,
Iter:	Anzahl der benötigten Iterationen,
Time:	Laufzeit in Sekunden exklusive Preprocessing und Überführung auf Standardform,
Status:	Erfolg (in weniger als Maxiter Iterationen terminiert), Maxiter (hat Maxiter Iterationen erreicht), CrErfolg (führte zu einem Fehler, ist mit robusteren Einstellungen terminiert), CrMaxiter (führte zu einem Fehler, hat CrMaxiter Iterationen erreicht), Fehler (führte auch mit robusteren Einstellungen zu einem Fehler).

Tabelle 1: Numerische Ergebnisse für Algorithmus 2

Datei	Nonzeros	Fun	μ	$\vert\vert\Phi\vert\vert$	PS	Iter	Time	Status
25FV47	11127	5.5018e+03	8.49e-05	7.49e-05	1	40	0.27	Erfolg
80BAU3B	29063	9.8722e+05	3.08e-05	3.50e-05	4	77	2.32	Erfolg
ADLITTLE	465	2.2549e+05	1.23e-03	1.05e-03	5	16	0.02	Erfolg
AFIRO	88	-4.6475e+02	7.40e-06	9.86e-06	4	6	0.01	Erfolg
AGG	2541	-3.5992e+07	9.59e-06	4.09e-06	8	37	0.10	Erfolg
AGG2	4515	-2.0239e+07	4.49e-04	8.68e-05	4	24	0.10	Erfolg
AGG3	4531	1.0312e+07	2.89e-04	5.69e-04	5	28	0.12	Erfolg
BANDM	2659	-1.5863e+02	6.21e-05	4.74e-05	0	43	0.06	Erfolg
BEACONFD	3476	3.3592e+04	2.77e-05	5.40e-05	3	17	0.02	Erfolg
BLEND	521	-3.0812e+01	1.43e-04	9.61e-06	2	14	0.01	Erfolg
BNL1	6129	1.9776e+03	3.59e-05	4.56e-05	0	196	0.42	Erfolg
BNL2	16124	1.8112e+03	4.19e-05	8.19e-05	0	85	1.22	Erfolg
BORE3D	1525	1.3731e+03	2.42e-04	1.13e-04	1	53	0.08	Erfolg
BRANDY	2150	1.5185e+03	6.32e-04	3.03e-04	0	48	0.05	Erfolg
CAPRI	1786	2.6901e+03	3.28e-06	7.14e-06	10	131	0.20	Erfolg
CYCLE	21322	-5.2264e+00	2.14e-06	7.95e-05	4	529	8.13	Erfolg
CZPROB	14173	2.1852e+06	1.23e-03	9.91e-04	1	50	0.09	Erfolg
D2Q06C	35674	1.2278e+05	8.95e-06	5.70e-06	0	223	4.74	Erfolg
D6CUBE	43888	3.1549e+02	9.05e-05	2.38e-04	1	21	0.15	CrErfolg
DEGEN2	4449	-1.4352e+03	2.86e-04	1.90e-04	7	15	0.06	Erfolg
DEGEN3	26230	-9.8729e+02	3.07e-04	1.95e-04	0	13	0.25	Erfolg
DFL001	41873	8.6519e+06	5.22e-02	4.49e+01	5	1000	85.65	Maxiter
E226	2767	-1.8752e+01	1.13e-04	8.60e-05	0	25	0.03	Erfolg
ETAMACRO	2489	-7.5572e+02	1.44e-05	1.96e-05	3	40	0.12	Erfolg
FFFFF800	6235	5.5568e+05	5.89e-08	3.56e-06	9	310	1.38	Erfolg
FINNIS	2714	1.7279e+05	3.46e-05	5.43e-05	2	45	0.08	Erfolg

Datei	Nonzeros	Fun	μ	$\|\|\Phi\|\|$	PS	Iter	Time	Status
FIT1D	14430	-9.1464e+03	5.50e-06	5.27e-09	7	124	0.36	Erfolg
FIT1P	10894	9.1464e+03	4.44e-04	1.37e-04	6	91	1.00	Erfolg
FIT2D	138018	-6.8464e+04	2.54e-06	5.09e-06	9	210	4.39	Erfolg
FIT2P	60784	6.8464e+04	3.58e-06	2.26e-06	9	71	31.82	Erfolg
GANGES	7021	-1.0956e+05	5.55e-08	1.21e-05	14	346	1.39	Erfolg
GFRD-PNC	3467	6.9022e+06	1.80e-03	3.47e-03	1	142	0.19	Erfolg
GREENBEA	31499	-7.2856e+07	1.74e+00	8.13e+01	0	100	2.57	CrMaxiter
GREENBEB	31499	-4.3023e+06	1.20e-05	8.45e-05	0	133	3.53	Erfolg
GROW15	5665	-1.0687e+08	4.87e-03	1.91e-03	0	113	0.23	Erfolg
GROW22	8318	-1.6083e+08	2.93e-02	2.80e-03	0	67	0.16	Erfolg
GROW7	2633	-4.7788e+07	2.60e-03	2.57e-05	0	111	0.17	Erfolg
ISRAEL	2358	-8.9664e+05	3.07e-05	2.88e-05	6	241	0.52	Erfolg
KB2	291	-1.7499e+03	1.48e-04	1.21e-05	6	361	0.41	Erfolg
LOTFI	1086	-2.5265e+01	1.28e-07	8.85e-05	2	297	0.39	Erfolg
MAROS-R7	151120	1.4972e+06	4.02e-07	2.65e-10	6	173	8.92	Erfolg
MAROS	10006	-5.8064e+04	9.54e-06	1.60e-05	0	251	1.58	Erfolg
MODSZK1	4158	3.2063e+02	1.56e-05	2.37e-05	0	27	0.05	Erfolg
PEROLD	6026	-1.0074e+04	1.25e-05	3.79e-04	3	1000	6.19	Maxiter
PILOT-JA	14706	-6.0731e+03	8.76e-04	4.47e-01	1	1000	12.33	Maxiter
PILOT-WE	9218	-2.7201e+06	5.19e-05	3.73e-05	1	741	2.32	Erfolg
PILOT	43220	-5.5749e+02	1.44e-06	9.78e-05	0	162	4.16	Erfolg
PILOT4	5145	-2.5423e+03	1.14e-04	7.14e-02	0	1000	5.58	Maxiter
PILOT87	73804	3.0171e+02	1.08e-06	9.95e-05	0	129	5.63	Erfolg
PILOTNOV	13129	-4.4973e+03	7.36e-08	3.15e-05	7	401	3.75	Erfolg
QAP12	44244	5.2288e+02	1.57e-05	2.11e-03	3	1000	145.72	Maxiter
QAP15	110700	1.0409e+03	3.00e-05	3.62e-03	2	1000	700.37	Maxiter
QAP8	8304	2.0350e+02	3.81e-05	7.63e-05	6	8	0.08	Erfolg
RECIPE	752	-2.6662e+02	9.32e-05	1.87e-04	4	12	0.02	Erfolg
SC105	281	-5.2202e+01	7.34e-06	1.44e-05	6	85	0.12	Erfolg
SC205	552	-5.2202e+01	1.40e-07	1.16e-07	7	194	0.32	Erfolg
SC50A	131	-6.4575e+01	2.46e-04	2.12e-04	4	31	0.03	Erfolg
SC50B	119	-7.0000e+01	4.30e-05	1.49e-07	3	23	0.02	Erfolg
SCAGR25	2029	-1.4753e+07	2.60e-05	2.52e-08	3	58	0.06	Erfolg
SCAGR7	553	-2.3314e+06	1.35e-05	1.30e-07	3	25	0.02	Erfolg
SCFXM1	2612	1.8417e+04	1.25e-05	2.58e-05	8	218	0.36	Erfolg
SCFXM2	5229	3.6660e+04	1.21e-05	2.43e-05	9	120	0.23	Erfolg
SCFXM3	7846	5.4901e+04	2.26e-04	6.84e-05	4	95	0.22	Erfolg
SCORPION	1708	1.8781e+03	6.55e-05	5.43e-05	5	54	0.06	Erfolg
SCRS8	4029	9.0430e+02	7.75e-07	2.01e-06	7	57	0.08	Erfolg
SCSD1	3148	8.6667e+00	1.08e-04	8.19e-05	2	11	0.01	Erfolg
SCSD6	5666	5.0500e+01	2.01e-04	2.28e-04	2	13	0.01	Erfolg
SCSD8	11334	9.0500e+02	4.79e-05	6.60e-05	6	9	0.02	Erfolg
SCTAP1	2052	1.4122e+03	1.71e-04	1.15e-04	3	23	0.02	Erfolg
SCTAP2	8124	1.7248e+03	1.35e-04	3.93e-05	1	21	0.04	Erfolg
SCTAP3	10734	1.4240e+03	3.85e-05	2.98e-05	3	16	0.05	Erfolg
SHARE1B	1182	-7.6589e+04	2.14e-06	4.11e-08	10	664	0.86	Erfolg
SHARE2B	730	-4.1573e+02	1.97e-04	8.23e-05	2	18	0.01	Erfolg
SHELL	4900	1.2088e+09	9.63e-03	7.80e-03	6	28	0.04	Erfolg
SHIP04L	8450	1.7933e+06	1.51e-04	9.26e-05	3	128	0.22	Erfolg

Datei	Nonzeros	Fun	μ	$\|\|\Phi\|\|$	PS	Iter	Time	Status
SHIP04S	5810	1.7987e+06	1.19e-03	9.45e-04	3	102	0.14	Erfolg
SHIP08L	17085	1.9091e+06	8.10e-04	6.52e-04	4	197	0.52	Erfolg
SHIP08S	9501	1.9201e+06	2.68e-03	9.60e-04	3	87	0.14	Erfolg
SHIP12L	21597	1.4702e+06	4.31e-04	3.54e-04	3	190	0.59	Erfolg
SHIP12S	10941	1.4892e+06	1.37e-03	5.20e-04	3	111	0.19	Erfolg
SIERRA	9252	1.5394e+07	1.52e-04	8.40e-05	0	31	0.08	Erfolg
STAIR	3857	-2.5127e+02	2.08e-05	4.16e-05	7	303	1.04	Erfolg
STANDATA	3038	1.2577e+03	1.29e-02	1.11e-03	0	17	0.02	Erfolg
STANDMPS	3686	1.4060e+03	3.14e-03	9.17e-04	0	25	0.03	Erfolg
STOCFOR1	474	-4.1132e+04	1.91e-03	3.17e-03	5	42	0.04	Erfolg
STOCFOR2	9492	-3.9024e+04	2.42e-04	9.35e-05	2	100	0.28	Erfolg
STOCFOR3	74004	-3.9977e+04	9.83e-06	3.05e-06	4	64	1.35	Erfolg
TRUSS	36642	4.5882e+05	4.70e-04	1.31e-04	3	26	0.23	Erfolg
TUFF	4523	2.9214e-01	5.56e-08	2.76e-05	8	460	1.01	Erfolg
VTP-BASE	914	1.2983e+05	2.92e-06	2.64e-05	5	180	0.20	Erfolg
WOOD1P	70216	1.4429e+00	6.64e-05	8.64e-05	0	20	0.71	Erfolg
WOODW	37478	1.3045e+00	1.74e-06	3.25e-06	0	85	0.83	Erfolg

3.3 Fazit

Wir haben in diesem Kapitel einen robusten Algorithmus zur Lösung linearer Programme in Python implementiert und getestet. Für diesen wurde, in vorherigen Kapiteln, anhand einer allgemeineren Problemstellung und, unter der Verwendung der Ergebnisse des zugrundeliegenden Artikels von Burke und Xu [1], der theoretische Grundstein gelegt.
Es war uns dabei möglich, den Großteil einer anspruchsvoll konzipierten Testproblem-Sammlung linearer Programme mit verschiedenen Arten von Nebenbedingungen, mit gewünschter Genauigkeit und einer akzeptablen Laufzeit zu lösen. Lediglich die hohe Anzahl abgelehnter Prädiktor-Schritte, besonders bei groß-dimensionierten Problemen, entsprach dabei nicht komplett unserer Zufriedenheit.
Der nicht unerhebliche Einfluss des Preprocessings auf die numerischen Ergebnisse unseres Algorithmus, gibt dabei aber durchaus Grund zur Hoffnung. Durch besser an den Algorithmus angepasste Preprocessing-Methoden, könnten sowohl die numerischen Ergebnisse im Allgemeinen als auch die Quote der akzeptierten Prädiktor-Schritte, eventuell noch erheblich verbessert werden. Es wäre daher äußerst interessant, inwiefern das - oder auch andere Modifikationen - die Performance des Algorithmus weiter verbessern könnte.

Literatur

[1] J. Burke und S. Xu. „A non-interior predictor-corrector path following algorithm for the monotone linear complementarity problem“. In: *Springer* Math. Program., Ser A 87 (2000), S. 113–130.

[2] R. W. Cottle, J.-S. Pang und R. E. Stone. *The Linear Complementarity Problem.* Society for Industrial und Applied Mathematics, (2009). ISBN: 9780898716863.

[3] C. Geiger und C. Kanzow. *Theorie und Numerik restringierter Optimierungsaufgaben.* Springer, (2002). ISBN: 9783540427902.

[4] C. Kanzow. „Some noninterior continuation methods for linear complementarity problems“. In: *SIAM* J. Matrix Anal. Appl. 17 (1996), S. 851–868.

[5] S. J. Wright. *Primal-Dual Interior-Point Methods.* Society for Industrial und Applied Mathematics, (1997). ISBN: 9780898713824.

[6] M. Kojima, N. Megiddo, T. Noma und A. Yoshise. „A unified approach to interior point algorithms for linear complementarity problems: A summary“. In: *Operations Research Letters* 10 (1991), S. 247–254.

[7] J. Burke. *Backtracking Line Search.* URL: https://sites.math.washington.edu/~burke/crs/516/notes/backtracking.pdf.

[8] J. Burke und S. Xu. „The global linear convergence of a non-interior path-following algorithm for linear complementarity problem“. In: *Mathematics of Operations Research* 23.3 (1998), S. 719–734.

[9] M. Fukushima, Z.-Q. Luo und J.-S. Pang. „A Globally Convergent Sequential Quadratic Programming Algorithm for Mathematical Programs with Linear Complementarity Constraints“. In: *Computational Optimization and Applications* 10 (1998), S. 5–34.

[10] L. Qi. „Convergence Analysis of Some Algorithms for Solving Nonsmooth Equations“. In: *Mathematics of Operations Research* 18.1 (1993), S. 227–244.

[11] P. Virtanen u. a. „SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python“. In: *Nature Methods* 17 (2020), S. 261–272.

[12] C. R. Harris u. a. „Array programming with NumPy“. In: *Nature* 585.7825 (Sep. 2020), S. 357–362.

[13] D. Cournapeau, N. Smith, D. S. Seljebotn, L. Barrett, Yuri, A. Lee, A. Grigorievskiy und J. Reimer. *scikit-sparse - Sparse matrix extensions for SciPy.* URL: https://scikit-sparse.readthedocs.io/.

[14] D. M. Gay. *Netlib Repository.* URL: https://www.netlib.org/lp/data/.

[15] E. D. Andersen und K. D. Andersen. „Presolving in linear programming“. In: *Math. Program.* 71.2 (Dez. 1995), S. 221–245.

[16] E. D. Andersen. „Finding all linearly dependent rows in large-scale linear programming“. In: *Optimization Methods and Software* 6.3 (1995), S. 219–227.

[17] J. A. Tomlin. „On scaling linear programming problems“. In: (1975). Hrsg. von M. L. Balinski und E. Hellerman, S. 146–166.

Eidesstattliche Erklärung

Hiermit erkläre ich, dass ich die Arbeit selbstständig verfasst, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt und die diesen Quellen und Hilfsmitteln wörtlich oder sinngemäß entnommenen Ausführungen als solche kenntlich gemacht habe. Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt.

Würzburg, 26.09.2024

....

Bastian Baumann