

2 Modellprädiktive Regelung

Die modellprädiktive Regelung¹ (MPC) ist ein modernes Regelungsverfahren, das auf der Lösung dynamischer Optimierungsprobleme basiert. In diesem Kapitel werden die für das Verständnis von MPC nötigen Grundlagen der Optimierung und das Prinzip der numerischen Lösung erläutert. Die Erweiterung auf stochastische Unsicherheiten erfolgt im nächsten Kapitel.

2.1 Dynamische Optimierung auf bewegtem Horizont

Eine typische Aufgabenstellung in der Regelungstechnik ist das Anfahren sowie das Stabilisieren von Arbeitspunkten. Man unterscheidet dabei zwischen Steuerung² einerseits und Regelung³ andererseits. Bei der Steuerung findet keine Rückkopplung der Zustände statt. Das bedeutet, dass die Stellgröße nicht an das tatsächliche Systemverhalten angepasst wird. Wenn das Systemmodell ungenau ist oder Störungen auf das System wirken, wird der Sollzustand nicht erreicht. Wesentliches Merkmal der Regelung ist die Rückkopplung im geschlossenen Regelkreis. Durch die Rückführung der Zustände kann die Abweichung vom Sollzustand berechnet und die Stellgröße aktualisiert werden. Damit ist es möglich, Modellfehler und Störungen auszugleichen, so dass der Sollzustand exakt erreicht wird.

Als Systemmodell wird im Folgenden ein allgemeines nichtlineares System in Zustandsraumdarstellung betrachtet. Das Systemverhalten wird hier durch n Differentialgleichungen erster Ordnung

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \quad (2.1)$$

beschrieben. Dabei bezeichnet $\mathbf{x} \in \mathbb{R}^n$ den Vektor der Systemzustände und $\mathbf{u} \in \mathbb{R}^m$ die Eingänge oder Stellgrößen des Systems. In der Praxis unterliegen die Eingänge fast immer Stellgrößenbeschränkungen

$$\mathbf{u}(t) \in U, \quad t \geq 0, \quad (2.2)$$

wobei die zulässige Menge meist durch Minimal- und Maximalwerte

$$U = [\mathbf{u}_{\min}, \mathbf{u}_{\max}] \quad (2.3)$$

¹Englisch „model predictive control“

²Englisch „feedforward control“ oder „open-loop control“

³Englisch „feedback control“ oder „closed-loop control“

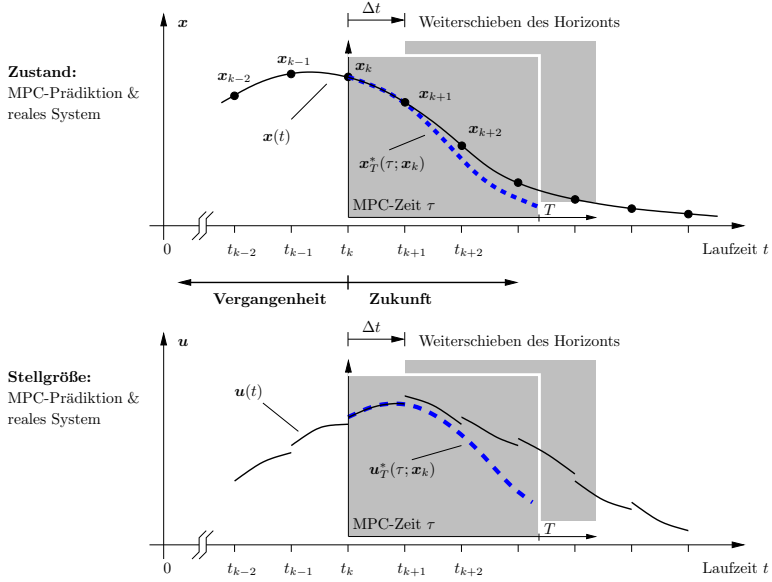


Abbildung 2.1: Prinzip der modellprädiktiven Regelung [Gra13, S. 141].

gegeben ist. Als Ziel der Regelung wird im Folgenden die Stabilisierung des Systems (2.1)

$$\mathbf{0} = \mathbf{f}(\mathbf{x}_R, \mathbf{u}_R) \quad (2.4)$$

in einer Ruhelage $(\mathbf{x}_R, \mathbf{u}_R)$ betrachtet.

Die modellprädiktive Regelung erreicht dieses Ziel (2.4), indem in jedem diskreten Zeitschritt $t_k = k \Delta t$ mit der Abtastzeit Δt ein dynamisches Optimierungsproblem

$$\min_{\bar{\mathbf{u}}(\cdot)} \quad J_T(\mathbf{x}_k, \bar{\mathbf{u}}) = V(\bar{\mathbf{x}}(T)) + \int_0^T l(\bar{\mathbf{x}}(\tau), \bar{\mathbf{u}}(\tau)) d\tau \quad (2.5a)$$

$$\text{u.B.v.} \quad \dot{\bar{\mathbf{x}}}(\tau) = \mathbf{f}(\bar{\mathbf{x}}(\tau), \bar{\mathbf{u}}(\tau)), \quad \bar{\mathbf{x}}(0) = \mathbf{x}_k = \mathbf{x}(t_k) \quad (2.5b)$$

$$\bar{\mathbf{u}}(\tau) \in U, \quad \tau \in [0, T] \quad (2.5c)$$

gelöst wird. Ausgehend vom Zustand $\mathbf{x}_k$ wird das Systemverhalten über den Zeithorizont T prädiziert und die optimale Steuerung $\bar{\mathbf{u}}^*(\tau)$ gesucht, die das Kostenfunktional J_T minimiert und die Stellgrößenbeschränkungen einhält. Die Schreibweise $\bar{\mathbf{x}}, \bar{\mathbf{u}}$ dient dazu, die MPC-internen Größen vom realen System (2.1) zu unterscheiden.

Für die Dauer der Abtastzeit Δt wird der Anfang der optimalen Steuertrajektorie $\bar{\mathbf{u}}^*$ als Stellgröße

$$\mathbf{u}(t_k + \tau) = \bar{\mathbf{u}}^*(\tau; \mathbf{x}_k), \quad \tau \in [0, \Delta t] \quad (2.6)$$

verwendet. Im nächsten Zeitschritt $t_{k+1} = t_k + \Delta t$ wird das Optimierungsproblem (2.5) mit dem neuen Anfangszustand $\mathbf{x}_{k+1}$ erneut gelöst. Dieses Prinzip wird in Abbildung 2.1 für einen Zustand und eine Stellgröße veranschaulicht.

Das Kostenfunktional J_T bestimmt die Eigenschaften der optimalen Lösung. Ein falsch gewähltes Kostenfunktional kann im Extremfall dazu führen, dass das Problem nicht lösbar ist. Eine häufige Wahl ist die quadratische Bestrafung der Zustände und der Stellgrößen im Integralterm $l(\bar{\mathbf{x}}, \bar{\mathbf{u}})$ und im Endkostenterm $V(\bar{\mathbf{x}})$, d. h.

$$J_T(\mathbf{x}_k, \bar{\mathbf{u}}) = \Delta \bar{\mathbf{x}}(T)^T \mathbf{P}_{\text{cost}} \Delta \bar{\mathbf{x}}(T) + \int_0^T \Delta \bar{\mathbf{x}}(\tau)^T \mathbf{Q}_{\text{cost}} \Delta \bar{\mathbf{x}}(\tau) + \Delta \bar{\mathbf{u}}(\tau)^T \mathbf{R}_{\text{cost}} \Delta \bar{\mathbf{u}}(\tau) d\tau, \quad (2.7)$$

wobei die Gewichtung über die symmetrischen Matrizen $\mathbf{P}_{\text{cost}} \in \mathbb{R}^{n \times n}$, $\mathbf{Q}_{\text{cost}} \in \mathbb{R}^{n \times n}$ und $\mathbf{R}_{\text{cost}} \in \mathbb{R}^{m \times m}$ eingestellt wird. Mit $\Delta \bar{\mathbf{x}} = \bar{\mathbf{x}} - \mathbf{x}_R$ und $\Delta \bar{\mathbf{u}} = \bar{\mathbf{u}} - \mathbf{u}_R$ werden die Abweichungen der Zustände und der Stellgrößen vom gewünschten Ziel (2.4) bezeichnet. Diese Form des Kostenfunktionals wird auch beim Entwurf linear-quadratischer Regler für lineare Systeme verwendet. Eine hohe Gewichtung der Stellgrößen kann im Sinne einer Energieoptimalität interpretiert werden. Soll dagegen der Sollzustand schnell erreicht werden, dann müssen die Zustände hoch gewichtet werden.

Das Optimierungsproblem für MPC kann auch anders als in (2.5) formuliert werden. Der Hauptunterschied liegt darin, welche Annahmen nötig sind, um Stabilität im geschlossenen Kreis beweisen zu können [MRRS00]. Eine für die Praxis wichtige Alternative ist die zeitdiskrete Formulierung mit N Prädiktionsschritten. Das Optimierungsproblem lautet dann

$$\min_{\bar{\mathbf{u}}} \quad J_N(\mathbf{x}_k, \bar{\mathbf{u}}) = V(\bar{\mathbf{x}}_{N-1}) + \sum_{j=0}^{N-1} l(\bar{\mathbf{x}}_j, \bar{\mathbf{u}}_j) \quad (2.8a)$$

$$\text{u.B.v.} \quad \bar{\mathbf{x}}_{j+1} = \mathbf{f}_d(\bar{\mathbf{x}}_j, \bar{\mathbf{u}}_j), \quad \bar{\mathbf{x}}_0 = \mathbf{x}_k = \mathbf{x}(t_k) \quad (2.8b)$$

$$\bar{\mathbf{u}}_j \in U, \quad j = 0, \dots, N-1 \quad (2.8c)$$

mit dem zeitdiskreten Systemmodell $\mathbf{f}_d$ und der stückweise konstanten Stellgröße $\hat{\mathbf{u}}^T = [\bar{\mathbf{u}}_0^T, \dots, \bar{\mathbf{u}}_{N-1}^T]$. Allerdings ist die Dynamik der meisten physikalischen Systeme durch Differentialgleichungen gegeben und die Herleitung eines zeitdiskreten Systemmodells oft nur näherungsweise möglich, beispielsweise mit einem Euler-Verfahren. Aus diesem Grund liegt der Schwerpunkt dieser Arbeit auf der Betrachtung zeitkontinuierlicher Systeme.

2.2 Optimalitätsbedingungen

Die notwendigen Optimalitätsbedingungen für das Optimierungsproblem (2.5) können kompakt mit der Hamilton-Funktion

$$H(\bar{\mathbf{x}}(\tau), \bar{\mathbf{u}}(\tau), \boldsymbol{\lambda}(\tau)) = l(\bar{\mathbf{x}}(\tau), \bar{\mathbf{u}}(\tau)) + \boldsymbol{\lambda}(\tau)^T \mathbf{f}(\bar{\mathbf{x}}(\tau), \bar{\mathbf{u}}(\tau)) \quad (2.9)$$

geschrieben werden. Die Hamilton-Funktion besteht aus dem Integralkostenterm $l(\bar{\mathbf{x}}, \bar{\mathbf{u}})$, dem Systemmodell $\mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}})$ und den zeitabhängigen Lagrange-Multiplikatoren $\boldsymbol{\lambda}(\tau) \in \mathbb{R}^n$. Aus den Ableitungen nach $\boldsymbol{\lambda}$ und nach $\bar{\mathbf{x}}$ ergeben sich die kanonischen Gleichungen für die optimalen Zustände⁴

$$\dot{\bar{\mathbf{x}}}^*(\tau) = \frac{\partial H}{\partial \boldsymbol{\lambda}} = \mathbf{f}(\bar{\mathbf{x}}^*(\tau), \bar{\mathbf{u}}^*(\tau)) \quad (2.10a)$$

$$\dot{\boldsymbol{\lambda}}^*(\tau) = -\frac{\partial H}{\partial \bar{\mathbf{x}}} = -H_{\bar{\mathbf{x}}}(\bar{\mathbf{x}}^*(\tau), \bar{\mathbf{u}}^*(\tau), \boldsymbol{\lambda}^*(\tau)) \quad (2.10b)$$

mit $H_{\bar{\mathbf{x}}} = \frac{\partial l}{\partial \bar{\mathbf{x}}} + \left(\frac{\partial \mathbf{f}}{\partial \bar{\mathbf{x}}}\right)^T \boldsymbol{\lambda}$. Die Differentialgleichung (2.10b) stellt das adjungierte System mit dem adjungierten Zustand $\boldsymbol{\lambda}$ dar. Weiter muss die optimale Lösung die Randbedingung

$$\bar{\mathbf{x}}^*(0) = \mathbf{x}_k \quad (2.11)$$

und die Transversalitätsbedingung

$$\boldsymbol{\lambda}^*(T) = \left. \frac{dV(\bar{\mathbf{x}}(\tau))}{d\bar{\mathbf{x}}} \right|_{\tau=T} \quad (2.12)$$

mit dem Endkostenterm $V(\bar{\mathbf{x}}(\tau))$ erfüllen. Aus Pontryagin's Maximumprinzip folgt, dass die Hamilton-Funktion (2.9) für die optimale Stellgröße $\bar{\mathbf{u}}^*$ ein absolutes Minimum annehmen muss, d. h.

$$H(\bar{\mathbf{x}}^*(\tau), \bar{\mathbf{u}}^*(\tau), \boldsymbol{\lambda}^*(\tau)) = \min_{\bar{\mathbf{u}} \in U} H(\bar{\mathbf{x}}^*(\tau), \bar{\mathbf{u}}(\tau), \boldsymbol{\lambda}^*(\tau)), \quad \tau \in [0, T] \quad (2.13)$$

muss gelten. Für den Fall ohne Stellgrößenbeschränkungen vereinfacht sich die letzte Bedingung zur Stationaritätsbedingung

$$\mathbf{0} = \frac{\partial H}{\partial \bar{\mathbf{u}}} = H_{\bar{\mathbf{u}}}(\bar{\mathbf{x}}^*(\tau), \bar{\mathbf{u}}^*(\tau), \boldsymbol{\lambda}^*(\tau)) \quad (2.14)$$

mit $H_{\bar{\mathbf{u}}} = \frac{\partial l}{\partial \bar{\mathbf{u}}} + \left(\frac{\partial \mathbf{f}}{\partial \bar{\mathbf{u}}}\right)^T \boldsymbol{\lambda}$. Die Optimalitätsbedingungen können verwendet werden, um eine optimale Lösung für das Optimierungsproblem (2.5) zu berechnen. Eine analytische Lösung ist nur in wenigen Fällen möglich, so dass in der Regel numerische Verfahren eingesetzt werden. Auf die Überprüfung weiterer hinreichender Bedingungen wird zumeist verzichtet und angenommen, dass eine (eindeutige) Lösung für das Problem existiert.

2.3 Berücksichtigung von Zustandsbeschränkungen

Neben den Beschränkungen der Stellgrößen (2.15) spielen in vielen praktischen Systemen auch Beschränkungen der Zustände eine große Rolle. Ein Beispiel ist das Umfahren von Hindernissen bei der Regelung eines mobilen Roboters. In allgemeiner Form können die Beschränkungen durch

$$\mathbf{h}(\bar{\mathbf{x}}(\tau)) \leq \mathbf{0} \quad (2.15)$$

⁴Die partiellen Ableitungen $\frac{\partial H}{\partial \bar{\mathbf{x}}}$ bzw. $\frac{\partial H}{\partial \boldsymbol{\lambda}}$ werden der Einfachheit halber als Spaltenvektor definiert.

mit $\mathbf{h} : \mathbb{R}^n \mapsto \mathbb{R}^q$ ausgedrückt werden. Die Erweiterung des Optimierungsproblems (2.5) um die Beschränkungen führt allerdings zu komplexen Optimalitätsbedingungen mit Fallunterscheidungen zwischen beschränkten und unbeschränkten Bereichen [BH75] und erschwert damit die numerische Lösung. Eine Alternative ist die Verwendung von äußeren Straffunktionen in der Kostenfunktion. Für jede Beschränkung

$$h_i(\bar{\mathbf{x}}(\tau)) \leq 0, \quad i = 1, \dots, q \quad (2.16)$$

wird dazu ein Strafterm

$$\psi(h_i(\bar{\mathbf{x}}(\tau))) = (\max\{0, h_i(\bar{\mathbf{x}}(\tau))\})^r, \quad r \geq 1 \quad (2.17)$$

eingeführt, der bei Verletzen der Beschränkung zu einer Bestrafung führt. Das Kostenfunktional wird um den Term

$$J_\varepsilon(\mathbf{x}_k, \bar{\mathbf{u}}) = \int_0^T l_\varepsilon(\bar{\mathbf{x}}(\tau)) \, d\tau = \int_0^T \sum_{i=1}^q \varepsilon_i \psi(h_i(\bar{\mathbf{x}}(\tau))) \, d\tau \quad (2.18)$$

mit den Gewichten ε_i ($i = 1, \dots, q$) erweitert. Statt (2.5a) wird also

$$J_{T,\varepsilon}(\mathbf{x}_k, \bar{\mathbf{u}}) = J_T(\mathbf{x}_k, \bar{\mathbf{u}}) + J_\varepsilon(\mathbf{x}_k, \bar{\mathbf{u}}) \quad (2.19)$$

minimiert. Der Nachteil dieser Methode ist, dass die optimale Lösung die Beschränkungen durchaus noch leicht verletzen kann. Die Gewichte müssen deshalb ausreichend groß gewählt werden.

2.4 Numerische Lösung

Bei vielen dynamischen Optimierungsproblemen können die Optimalitätsbedingungen nicht analytisch gelöst werden und man ist auf numerische Lösungsverfahren angewiesen. Grundsätzlich wird zwischen direkten und indirekten Verfahren unterschieden. Indirekte Verfahren lösen die Optimalitätsbedingungen numerisch. In diese Klasse gehören das Diskretisierungs-, das Kollokations-, das Schieß- und das Gradientenverfahren. Direkte Verfahren diskretisieren dagegen die Stelltrajektorie $\bar{\mathbf{u}}(\tau)$ und reduzieren damit das Problem auf ein endlich-dimensionales, das mit den Mitteln der statischen Optimierung gelöst werden kann. Je nachdem ob die Kostenfunktion und die Dynamik ebenfalls diskretisiert werden, spricht man von Voll- oder Teildiskretisierung. Auf diesem Weg lassen sich auch die Zustandsbeschränkungen (2.15) direkt berücksichtigen.

Für die hier verwendete MPC-Formulierung (2.5) eignet sich insbesondere das Gradientenverfahren. Ausgehend von einer Startschätzung $\bar{\mathbf{u}}^{(0)}(\tau)$ werden in jedem Schritt j die Differentialgleichung des Systems (2.10a)

$$\dot{\bar{\mathbf{x}}}(\tau) = \mathbf{f}(\bar{\mathbf{x}}(\tau), \bar{\mathbf{u}}(\tau)), \quad \bar{\mathbf{x}}(0) = \mathbf{x}_k \quad (2.20)$$

Initialisierung:	
$j \leftarrow 0$	Iterationszähler
ϵ	Abbruchkriterium
$\bar{\mathbf{u}}^{(0)}(\tau)$	Startschätzung der Steuertrajektorie
$\bar{\mathbf{x}}^{(0)}(\tau)$	Integration von (2.20) in Vorwärtszeit
repeat	
$\boldsymbol{\lambda}^{(j)}(\tau)$	Integration von (2.21) in Rückwärtszeit
$\mathbf{g}^{(j)}(\tau) \leftarrow (2.22)$	Berechnung des Gradienten
$\alpha^{(j)} \leftarrow \arg \min_{\alpha > 0} J(\bar{\mathbf{u}}^{(j)} - \alpha \mathbf{g}^{(j)})$	Linienuche (approximativ)
$\bar{\mathbf{u}}^{(j+1)}(\tau) = \bar{\mathbf{u}}^{(j)}(\tau) - \alpha^{(j)} \mathbf{g}^{(j)}(\tau)$	neue Steuertrajektorie mit Schrittweite $\alpha^{(j)}$
$\bar{\mathbf{x}}^{(j+1)}(\tau)$	Integration von (2.20) in Vorwärtszeit
$j \leftarrow j + 1$	
until $ J(\bar{\mathbf{u}}^{(j+1)}) - J(\bar{\mathbf{u}}^{(j)}) \leq \epsilon$	Abbruchkriterium

Tabelle 2.1: Algorithmus des Gradientenverfahrens [Gra13, S. 155].

in Vorwärtszeit und die Differentialgleichung des adjungierten Systems (2.10b)

$$\dot{\boldsymbol{\lambda}}(\tau) = -H_{\bar{\mathbf{x}}}(\bar{\mathbf{x}}(\tau), \bar{\mathbf{u}}(\tau), \boldsymbol{\lambda}(\tau)), \quad \boldsymbol{\lambda}(T) = V_{\bar{\mathbf{x}}}(T) \quad (2.21)$$

in Rückwärtszeit integriert, um die Trajektorien $\bar{\mathbf{x}}^{(j)}(\tau)$ und $\boldsymbol{\lambda}^{(j)}(\tau)$ zu bestimmen. Diese Lösung erfüllt im Allgemeinen die Stationaritätsbedingung (2.14) bzw. die Bedingung (2.13) nicht, aber der negative Gradient

$$-\mathbf{g}^{(j)}(\tau) = -\mathbf{H}_{\bar{\mathbf{u}}}(\bar{\mathbf{x}}^{(j)}(\tau), \bar{\mathbf{u}}^{(j)}(\tau), \boldsymbol{\lambda}^{(j)}(\tau)) \quad (2.22)$$

kann verwendet werden, um einen Abstieg im Kostenfunktional (2.5a) zu erreichen. Der negative Gradient legt die Suchrichtung für das Optimum fest, die Bestimmung der optimalen Schrittweite

$$\alpha^{(j)} \leftarrow \arg \min_{\alpha > 0} J(\bar{\mathbf{u}}^{(j)} - \alpha \mathbf{g}^{(j)}) \quad (2.23)$$

stellt ein unterlagertes skalares Optimierungsproblem dar, das meist approximativ gelöst wird. Die Stellgrößenbeschränkungen (2.2) können beim Gradientenverfahren berücksichtigt werden, indem die Steuerung $\bar{\mathbf{u}}^{(j)}(\tau)$ in jedem Schritt auf die zulässige Menge U projiziert wird. Der Algorithmus ist in Tabelle 2.1 zusammengefasst und wird in dieser Arbeit als Basis für die MPC-Implementierung verwendet.

2.5 Echtzeitfähige MPC-Implementierung

Für eine echtzeitfähige Implementierung der modellprädiktiven Regelung muss der Rechenaufwand der numerischen Lösung begrenzt werden. Bei Verwendung des Gradientenverfahrens ist es naheliegend die Optimierung nach einer festen Anzahl Iterationen abzubrechen. Die so erhaltene Näherungslösung kann im nächsten Zeitschritt zur Reinitialisierung verwendet werden.

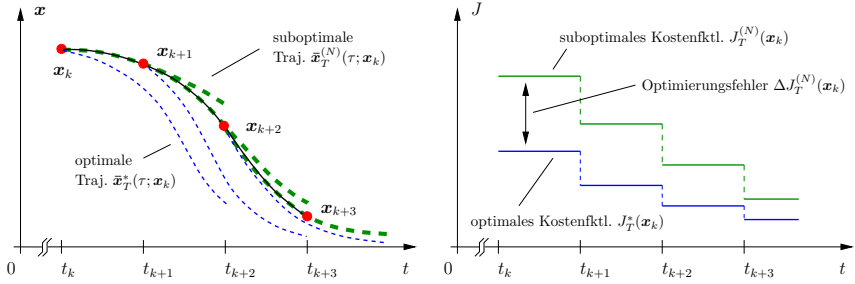


Abbildung 2.2: Suboptimale MPC-Implementierung [Gra13, S. 156].

Statt der optimalen Steuerung (2.6) wird die suboptimale Lösung nach N Iterationen als Stellgröße

$$\mathbf{u}(t_k + \tau) = \bar{\mathbf{u}}^{(N)}(\tau; \mathbf{x}_k), \quad \tau \in [0, \Delta t] \quad (2.24)$$

verwendet. Die Abweichung der Kosten vom optimalen Wert $J_T^*(\mathbf{x}_k, \bar{\mathbf{u}}^*)$

$$\Delta J_T^{(N)}(\mathbf{x}_k, \bar{\mathbf{u}}) = J_T^{(N)}(\mathbf{x}_k, \bar{\mathbf{u}}) - J_T^*(\mathbf{x}_k, \bar{\mathbf{u}}^*) \geq 0 \quad (2.25)$$

wird als Optimierungsfehler bezeichnet. Wenn die suboptimale Lösung als Startwert im nächsten Abtastschritt verwendet wird, kann der Optimierungsfehler sukzessive reduziert werden. Das Prinzip wird in Abbildung 2.2 dargestellt. Es lässt sich zeigen, dass auch mit dieser suboptimalen Strategie die Stabilität des Systems (2.4) erreicht wird [GEK10; GK12].

Die Liniensuche (2.23), also die Bestimmung der optimalen Schrittweite $\alpha^{(j)}$, kann durch Verwendung einer festen Schrittweite abgekürzt werden. In diesem Fall muss α aber sehr klein gewählt werden, was zu einer langsamen Konvergenz führt. Eine bessere Alternative ist es, die Kostenfunktion in Suchrichtung (2.22) durch ein Polynom zu approximieren und das Minimum des Polynoms als Schrittweite zu verwenden.

Wenn die Kostenfunktion (2.7) für drei Schrittweiten $\alpha_1 < \alpha_2 < \alpha_3$ ausgewertet wird, kann sie näherungsweise durch eine quadratische Funktion

$$J(\bar{\mathbf{u}}^{(j)} - \alpha^{(j)} \mathbf{g}^{(j)}) \approx \phi(\alpha) = c_0 + c_1 \alpha + c_2 \alpha^2 \quad (2.26)$$

mit den Koeffizienten c_0, c_1, c_2 beschrieben werden. Für die Koeffizienten können explizite Formeln angegeben werden [GK12]. Als Schrittweite wird der Wert $\alpha^{(j)}$ verwendet, der die Funktion $\phi(\alpha)$ innerhalb der Grenzen von α_1 und α_3 minimiert, d. h.

$$\alpha^{(j)} = \arg \min_{\alpha \in [\alpha_1, \alpha_3]} \phi(\alpha). \quad (2.27)$$

Damit das Minimum im Laufe der Iterationen zwischen die Intervallgrenzen fällt, werden

die Grenzen nach jeder Iteration gemäß

$$[\alpha_1, \alpha_3] \leftarrow \begin{cases} \kappa[\alpha_1, \alpha_3] & \alpha^{(j)} \geq \alpha_3 - \epsilon_\alpha(\alpha_3 - \alpha_1) \text{ und } \alpha_3 \leq \alpha_{\max} \\ \frac{1}{\kappa}[\alpha_1, \alpha_3] & \alpha^{(j)} \leq \alpha_1 + \epsilon_\alpha(\alpha_3 - \alpha_1) \text{ und } \alpha_1 \geq \alpha_{\min} \\ [\alpha_1, \alpha_3] & \text{sonst} \end{cases} \quad (2.28)$$

$$\alpha_2 \leftarrow \frac{\alpha_1 + \alpha_3}{2} \quad (2.29)$$

mit dem Anpassungsfaktor $\kappa > 1$ und der Intervalltoleranz $\epsilon_\alpha \in (0, 1)$ adaptiert. Damit das Intervall nicht unbegrenzt wächst, können mit $\alpha_{\min}$ und $\alpha_{\max}$ minimale und maximale Werte für die Intervallgrenzen festgelegt werden.

Das Programm **GRAMPC**, das am Institut für Mess-, Regel- und Mikrotechnik entwickelt wurde, basiert auf dem suboptimalen Gradientenverfahren und der adaptiven Liniensuche [KG14]. Die Software ist in C geschrieben und verfügt über eine Anbindung an MATLAB/SIMULINK. Mit dem echtzeitfähigen Verfahren ist die modellprädiktive Regelung nichtlinearer Systeme mit Abtastzeiten im Millisekunden-Bereich möglich. Aufgrund der sehr effizienten Implementierung wurde diese Software auch als Grundlage für das in Kapitel 4 entwickelte stochastische Verfahren eingesetzt.

Modellprädiktive Regelung nichtlinearer Systeme mit
Unsicherheiten

Völz, A.

2016, IX, 55 S. 7 Abb., Softcover

ISBN: 978-3-658-16278-8