



Datenbasierte, adaptive Regelung von Servomotoren für Industrieroboter

Christopher Ranisch · Heiko Koch · Timm Streul

Eingegangen: 28. November 2021 / Angenommen: 24. Januar 2022 / Online publiziert: 8. März 2022
 © Der/die Autor(en) 2022

Zusammenfassung Datenbasierte Schätzalgorithmen (z. B. künstliche neuronale Netze oder Support Vector Machines) sind eine robuste Alternative zu konventionellen, physikalischen Modellen für adaptive Regler. Dieser Typ Algorithmus kann anhand von empirischen Daten nichtlineare und zeitvariante dynamische Systeme identifizieren sowie zukünftige Ausgangsgrößen prognostizieren. Tiefere Kenntnisse über die Physik des technischen Systems sind somit nicht mehr notwendig, und es können einfach und schnell Modelle abgeleitet werden. Speziell für Industrieroboter eröffnen datenbasierte Schätzalgorithmen neue Möglichkeiten. So muss die nichtlineare Dynamik des Roboters je nach Pose und Last nicht durch komplexe kinematische Gleichungen berechnet, sondern kann aufgrund von Sensordaten geschätzt werden. Dies spart Berechnungszeit bei gleichbleibend hoher Modellgüte. Des Weiteren kann der Regler deutlich schneller bei geänderten Randbedingungen des Roboters, wie beispielsweise stark unterschiedlichen mechanischen Lasten am Endeffektor, adaptiert werden. In diesem Beitrag wird exemplarisch die Auslegung eines datenbasierten, modellprädiktiven Reglers sowohl für einen sechs- als auch für einen einachsigen Gelenkprüfstand diskutiert. Hierbei erfolgen der Entwurf und die Validierung des Reglers anhand einer Co-Simulation bestehend aus Mehrkörper- und Systemsimulation. Designparameter der Regelung (Größe des Modells, Trainingsdaten, Optimierer etc.) werden diskutiert. Schließlich erfolgt eine experimentelle Validierung an einem einachsigen Gelenkprüfstand.

Schlüsselwörter Industrierobotik · Modellbasierte prädiktive Regelung · Lokale, lineare Modellbäume · Datenbasierte Modelle

Data-driven, adaptive control of servo drives for industrial robots

Abstract Data-driven estimation algorithms (i.e., artificial neural networks or support vector machines) are a robust alternative to conventional, physical models for adaptive controllers. These algorithms can identify non-linear and time-variant dynamic systems and predict the systems' future outputs using empirical data. Thus, deep knowledge of the system's physics is not required anymore, and models can be implemented and derived easily and quickly. Especially for industrial robots, data-driven estimation algorithms offer new possibilities. For example, depending on the pose and load, the non-linear dynamics of the robot do not need to be calculated by means of complex kinematic equations but can be estimated based on sensor data. This saves computational time and provides a constant model accuracy. Furthermore, the controller can adapt to changed boundary conditions, such as varying mechanical loads on the end effector. This paper discusses the design of a data-driven, model predictive controller for a six- and a single-axis joint test rig. Here, the design and validation of the controller are based on a co-simulation consisting of a multibody and system simulation. All design parameters of the control (order of magnitude of the model, training data, optimization, etc.) are discussed. Finally, the control will be validated in an experiment using a single-axis joint test rig.

Keywords Industrial robotics · Model-based predictive control · Local linear model trees · Data-driven models

C. Ranisch (✉) · H. Koch · T. Streul
 Fachbereich Elektrotechnik und Informationstechnik,
 Hochschule Darmstadt, Birkenweg 8, 64295 Darmstadt,
 Deutschland
christopher.ranisch@h-da.de

Einleitung und Problemstellung

Industrieroboter sind ein essentieller Bestandteil in modernen Produktionsstraßen für unterschiedliche Waren und Produkte und zur mechanischen Bearbeitung von Werkstücken. Die Aufgabenfelder umfassen bspw. Schweißen, Kleben, Fräsen, Lackieren, Pick-And-Place oder Palettieren. Infolge der flexiblen Einsatzmöglichkeiten, hohen Zuverlässigkeit bzw. Geschwindigkeit und Erleichterung der Arbeit sind Industrieroboter daher unumgänglich für moderne und vernetzte industrielle Unternehmen. Je nach Aufgabe des Roboters steigen aber infolge von immer komplexeren Waren und Bearbeitungsprozessen von Werkstücken die Anforderungen im Hinblick auf Prozessgenauigkeit und -geschwindigkeit. So müssen für eine ökologisch nachhaltige und effiziente Produktion bereits ab dem ersten Werkstück alle Qualitätsanforderungen erfüllt werden. Die Teilkomponenten eines Industrieroboters (Sensorik, Servomotoren, Getriebe, digitale Regler) müssen also den genannten Anforderungen genügen. Speziell die eingesetzten Regler müssen hochdynamisch auf Laständerungen des Roboters reagieren und entsprechende Stellgrößen berechnen [1]. Adaptive, modellbasierte Regler eröffnen die Möglichkeit – je nach Applikation – situativ auf geänderte Umgebungsbedingungen des Roboters reagieren zu können [2]. Inspiriert durch die aktuellen Forschungsarbeiten im Bereich adaptiver Regler für Industrieroboter, wird in diesem Beitrag eine robuste, datenbasierte, modellbasierte prädiktive Regelung (MPR) entwickelt. Diese soll für unterschiedlichen Lasten am Endeffektor eine gleichbleibend hohe Prozessgenauigkeit gewährleisten können. Alle Schritte in der Entwicklung der Regelung (Training, Validierung und Integration des datengetriebenen Modells, Architektorentwicklung, Auswahl des Optimierungsalgorithmus und Sensitivitätsanalyse) werden diskutiert. Als Schätzalgorithmus für die Systemdynamik werden hierbei so genannte lokale, lineare Modellbäume (Engl.: Local-Linear-Model-Trees (LoLiMot)) eingesetzt. Diese Klasse eines datenbasierten Modells kann robust nichtlineare und parametervariante dynamische Systeme schätzen [3]. Die Architektur der implementierten Regelung lehnt sich an eine MPR [4] an. Kritische Einflussparameter (z. B. Abtastzeit des Reglers, Anzahl der Aufrufe des Optimierers) auf die Güte der Regelung werden ebenfalls diskutiert. Letztendlich wird der Algorithmus an einem einachsigen Roboterarm experimentell validiert und mit einer konventionellen P/PI Kaskadenregelung verglichen.

Stand der Technik und Forschung

Ein sehr weit verbreiteter und etablierter Regelungsalgorithmus für industrielle Roboterarme, ist eine kaskadierte P/PI-Regelung mit Geschwindigkeitsvorsteuerung [5]. Hierbei werden digitale PID-Regler für die einzelnen Regelkreise (Geschwindigkeit und Po-

sition der Achse) eingesetzt. Beiden Reglern ist eine Momenten- bzw. Stromregelung der Servomotoren unterlagert, d. h. es handelt sich hierbei um einen Regelkreis mit drei Schleifen pro Gelenkachse des Roboters. Die Auslegung der Regelparameter für die Gelenke erfolgt hierbei anhand der kinematischen Beschreibung des Roboters oder experimentell durch die Identifikation des Roboters [6, 7]. Hierfür muss also zunächst die direkte und inverse Kinematik anhand der geometrischen Parameter und definierten Koordinatensysteme berechnet werden. Ein entscheidender Nachteil dieses Ansatzes ist, dass plötzliche Änderungen in der Kinematik (z. B. die Aufnahme eines Werkstücks) nicht oder nur unzureichend über die Regler kompensiert werden können [8]. Dieses Verhalten der Regler führt daher zwangsläufig zu einer verschlechterten Prozessgenauigkeit. Ein weiterer Nachteil ist die Komplexität in der Auslegung der Regler. So muss erst eine hinreichend genaue kinematische Beschreibung des Roboters erfolgen, um die einzelnen Regler zu parametrisieren. Ist diese nicht vorhanden, so kann der Roboter im Betrieb instabil werden. Um aber für Roboter mit ständig wechselnden mechanischen Lasten ein robustes Systemverhalten zu gewährleisten, werden verstärkt adaptive Regelungsalgorithmen untersucht und experimentell validiert [2, 6]. Dieser Typ Regelung kann situativ und abhängig von der zugewiesenen Aufgabe optimalen Stellgrößen für die Servomotoren berechnen. Hierbei wurden verschiedene Ansätze (z. B. Fuzzy-Regelung, situativ schaltende oder adaptive PID-Regler, modellbasierte Regler) für roboterspezifische Applikationen entwickelt [9–11]. Alle bisherigen Ansätze benötigen hierfür numerische Modelle zur Schätzung der Systemdynamik. Die hinterlegten Modelle, sind in der Regel physikalische Modelle des Roboters, welche die Kinematik beschreiben [7]. Weitere eingesetzte Modelltypen sind künstliche neuronale Netze (KNN), zentrale Mustergeneratoren oder Beobachter auf Basis von Kalman-Filtern [11–14]. Letztere bieten den Vorteil, dass keine analytische Systembeschreibung vorliegen muss, sie schätzen anhand von Datensätzen offline oder online die Systemdynamik. Je nach Modellklasse können so unterschiedliche Klassen von dynamischen Systemen identifiziert und geschätzt werden [15]. Des Weiteren können diese Modelle online lernen und so auf geänderte Randbedingungen, Parameter des Systems oder Umwelteinflüsse reagieren [16].

Der Großteil der bisherigen Beiträge im Bereich der adaptiven Regelung von Industrierobotern fokussiert sich auf modellbasierte Regler mit beschreibender Kinematik des Roboters oder Kalman Filter als Beobachter. In diesem Artikel wird ein datenbasierter, adaptiver Regler auf Basis von LoLiMot Modellen für einen Roboter mit wechselnden Lastmassen untersucht. Die Auslegung der Regelung erfolgt zunächst anhand einer Simulation für einen sechsachsigen Roboter. Hierbei werden Kriterien an die Trainingsda-

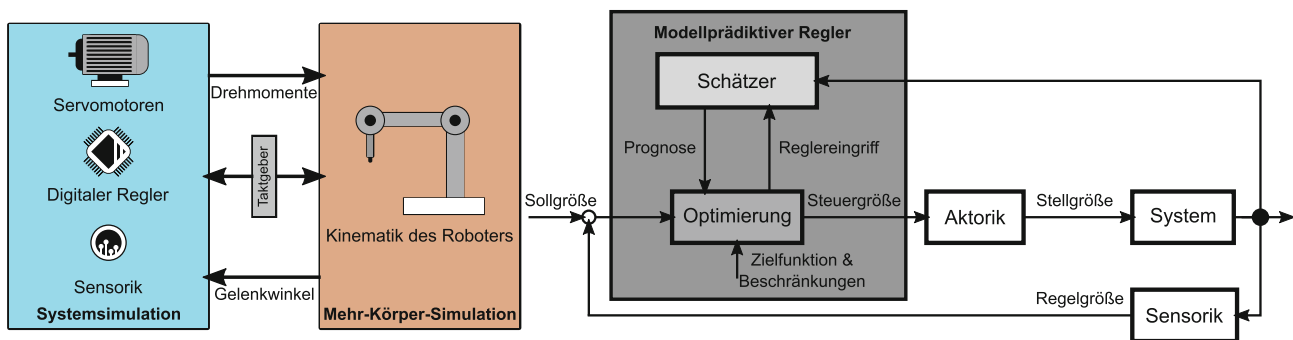


Abb. 1 Simulationsarchitektur und Signalfluss der Co-Simulation zur Synthese der MPR mit linearen, lokalen Modellnetzen für einen Knickarmroboter vom Typ ABB IRB140 (*links*) und

Architektur des MPR Reglers für die Positionsregelung des Roboters (*rechts*). (Nach [4])

ten (Anregungssignale, Systemparameter, Konsistenz, Einzigartigkeit etc.) für die LoLiMot Modelle diskutiert. Ebenso wird die Auslegung des Reglers (Optimierer, Begrenzungen etc.) näher betrachtet. Schlussendlich wird der Regler an einem einachsigen Gelenkprüfstand experimentell validiert und mit einer P/PI Kaskadenregelung verglichen. Hierbei wird die Masse am Endeffektor asymmetrisch variiert um die Vorteile einer datenbasierten Regelung aufzuzeigen.

Methodik und Diskussion

Simulationsumgebung zur Entwicklung eines datenbasierten, adaptiven Reglers für Industrieroboter

Als Anwendungsszenario wird im Folgenden die Positionsregelung für einen Knickarmroboter vom Typ ABB IRB140 mit sechs Gelenkachsen ausgelegt. Dieser hat laut Spezifikation eine maximale Nutzlast von 6 kg am Endeffektor und soll als exemplarische Aufgabe verschiedene Werkstücke (0,5–5 kg) transportieren. Durch die vergleichsweise hohe Variation der Masse am Endeffektor sollen die Parametervariationen der Kinematik und die nichtlineare Dynamik des Industrieroboters abgebildet werden. Zunächst wird für die Synthese der Regelung eine Co-Simulation mit der in Abb. 1 (*links*) aufgezeigten Simulationsarchitektur implementiert. Der digitale Regler, die elektrischen Antriebe sowie die datenbasierten Schätzalgorithmen werden in der Systemsimulation abgebildet. Über eine Mehr-Körper-Simulation (MKS) wird die Kinematik des Industrieroboters implementiert. Die Masse am Endeffektor kann hierbei entsprechend variiert werden. Alle hierfür notwendigen CAD-Daten des Roboters werden vom Hersteller online zur Verfügung gestellt. Eingangsvektor für die Systemsimulation sind die Gelenkwinkel q_i der sechs Achsen des Roboters. Ausgangsvektor sind die jeweiligen Drehmomente T_i der Servomotoren. Diese werden dann entsprechend in der MKS verarbeitet und führen zu entsprechenden Änderungen der Gelenkwinkel q_i und -geschwindigkeiten \dot{q}_i . Für die Servomotoren wird eine vor-

ab parametrisierte Stromregelung angenommen und die Dynamik kann als proportionale Verzögerungsstrecke 1. Ordnung genähert werden. Beide Simulationen werden über einen gemeinsamen Taktgeber synchronisiert, sodass die Berechnungsschritte und der Austausch von Signalen angepasst sind. Dies ist dadurch begründet, dass unterschiedliche Lösungsverfahren mit fester und variabler Schrittweite für die Simulationen verwendet werden.

Die Architektur des digitalen Reglers entspricht einer MPR [4] (vgl. Abb. 1 *rechts*), wobei die Modellintegration, der Optimierer sowie das Referenzmodell für den vorgestellten Anwendungsfall weiterentwickelt bzw. angepasst sind. In der konventionellen MPR werden in der Regel lineare Modelle (z. B. Zustandsraumdarstellung) zur Schätzung der Systemgrößen verwendet. Dieser Beitrag fokussiert explizit datenbasierte Modelle als Schätzer. Alle Teilschritte im Entwurf der Regelung (Training und Validierung der datenbasierten Schätzalgorithmen, Auswahl des Optimierers etc.) werden im Folgenden diskutiert.

Training von datenbasierten Schätzalgorithmen und Entwurf eines modellprädiktiven Reglers

Im ersten Auslegungsschritt werden die Schätzer trainiert und validiert. Für das Training werden synthetische Trainingsdaten mit der Co-Simulation aus Abb. 1 generiert. Als Schätzalgorithmus für die aktuelle Pose werden LoLiMot Modelle für jede einzelne Gelenkachse verwendet. Die allgemeine Struktur für ein LoLiMot Modell kann Abb. 2 entnommen werden. Jedes einzelne Neuron dieser Netzklasse beinhaltet ein lokales, lineares Modell (LLM) mit Gültigkeitsfunktion Φ_i . Der Ausgang \hat{y} eines Netzes berechnet sich über die mit w gewichtete Superposition der Ausgänge der einzelnen Neuronen in Abhängigkeit der jeweiligen Gültigkeitsfunktionen Φ_i aller Neuronen. Vereinfacht kann Φ_i somit als Arbeitspunkt des Schätzmodells aufgefasst werden [12]. Die Signalvektoren \underline{x} und \underline{z} beschreiben die Eingangs- und Parametersignale des zu identifizierenden dynamischen Systems [12, 17]. Für

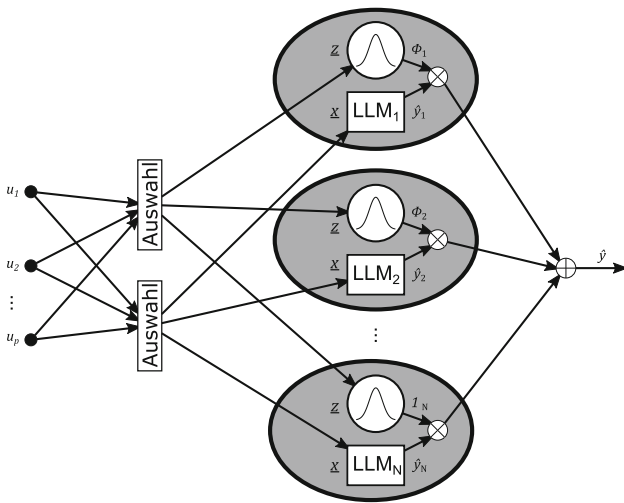


Abb. 2 Aufbau eines LoLiMot Modells mit N Neuronen, dem Eingangsvektor \underline{x} , dem Parametervektor \underline{z} und Gültigkeitsfunktionen Φ_i [12]

die geschätzte Systemgröße des LoLiMot mit N Neuronen und Signalvektoren der Länge n_x bzw. n_z folgt somit:

$$\hat{y} = \sum_{i=1}^N (w_{i,0} + w_{i,1}x_1 + \dots + w_{i,n_x}x_{n_x}) \Phi_i(\underline{z}) \quad (1)$$

Wobei der Gewichtungsvektor \underline{w} ein linearer Parameter des LoLiMot Netzes ist und vor dem Training gewählt wird. Die Berechnung der Schätzung, der Aufbau eines LoLiMot Modells und die theoretischen Grundlagen und Hintergründe zur nichtlinearen Systemidentifikation wurden von Nelles et al. [12] bereits umfassend beschrieben.

Entscheidend für die Qualität der Prognose des LoLiMot Modells, sind die Trainingsdaten. Im Idealfall wird über die Trainingsdaten der gesamte mögliche Parameter- und Frequenzbereich des Systems abgedeckt. Für das hier diskutierte Anwendungsszenario (Transport von Werkstücken) muss also ein Training für jeweils verschiedene Massen m_i am Endeffektor vorgenommen werden, über einen möglichst vollständigen Winkelbereich aller Gelenke des Roboters. Die Variation der Masse und die Gelenkwinkel, welche bei einem Durchlauf nicht trainiert werden stellen zeitvariante Parameter für das Modellnetz dar. Solche zeitvarianten Parameter werden in der Regel über Dreiecks- oder Sinussignale trainiert. Für das Training von Gelenk i wird über den Motor ein Drehmoment gestellt. Hierfür eignet sich eine amplitudenmodulierte Pseudo-Randomisierte-Binärsequenz¹ (PRBS) besonders gut. Alle übrigen Gelenkachsen des Roboters werden mit einer Sollvorgabe zur Winkelgeschwindigkeit angesteuert. Andere Anregungssignale (z. B.

Dreieck oder Sprung) führten zu einer schlechteren Netz- und Prognosequalität und eignen sich somit nicht für diese Anwendung. Die Trainingsdaten für jede Gelenkachse sind in zwei einzelne Segmente unterteilt. Für Gelenkachse 1 sind die Trainingsdaten bspw. wie folgt aufgliedert:

Segment 1 Gelenkachse 1 wird mit einem amplitudenmodulierten PRBS Drehmomentsignal angeregt und alle übrigen Gelenkachsen (2–6) werden mit einer Geschwindigkeitsvorgabe angeregt. Diese sind für jede Achse zeitlich versetzte, einzelne Sinuswellen. Hierdurch wird der Einfluss von jeder einzelnen Gelenkachse auf Gelenkachse 1 abgebildet. Die Gelenke 2–6 werden hierbei im maximal möglichen Winkelbereich angesteuert. Die Masse wird am Endeffektor im Bereich 0,5–5 kg über die Zeit variiert.

Segment 2 Gelenkachse 1 wird ebenfalls, wie in Segment 1, mit einem amplitudenmodulierten PRBS Drehmomentsignal angeregt. Die übrigen Gelenkachsen (2–6) werden mit sinusförmigen Profilen gleichzeitig angesteuert, um so die Wechselwirkungen der jeweiligen Gelenkachsen untereinander abzubilden. Hierbei wird die Anregungsfrequenz der Sinuswelle von Achse 2 bis 6 für jede Achse verdoppelt. Die Masse am Endeffektor wird ebenfalls im Bereich 0,5–5 kg über die Zeit variiert.

Die beiden Datensegmente werden dann zu einem Segment für das Training des Netzes aneinandergereiht. Dies ist möglich, da alle Gelenke am Ende von Segment 1 und 2 jeweils wieder in der Ursprungsposition des Roboters sind, mit Ausnahme des Trainingsgelenks. Die Sprünge im Verlauf der Trainingsdaten an der Lage des Trainingsgelenks haben sich beim Training mit unterschiedlichen Gelenken als unproblematisch erwiesen. Um zusätzlich die Effekte von unterschiedlichen Massen am Endeffektor abzubilden, wird die Lastmasse m_i im Bereich von 0,5–5 kg treppenförmig über die Zeit variiert (vgl. Abb. 3). Ein Datensatz bildet somit alle Effekte (Massenträgheiten am Endeffektor, Wechselwirkungen der Gelenke) ab. Wie in Abb. 3 zu erkennen ist, wird die Masse am Endeffektor nicht kontinuierlich variiert. Aufgrund der guten Interpolationsfähigkeit des LoLiMot Algorithmus ist die eher grobe Auflösung der Massenvariation aber ausreichend, um robuste Prognosen zu erzielen. Des Weiteren sind die geforderten Winkel für Segment 1 und 2 so gewählt, dass alle möglichen Posen des Roboters abgedeckt sind. Der LoLiMot Algorithmus erhält somit Information über alle mögliche Zustände und Gravitationskräfte des Roboters.

Neben den offline Trainingsdaten, benötigt der LoLiMot Algorithmus für die online Prognose historische Sensordaten der Systemzustände (Gelenkwinkel) und Stellgrößen (Drehmomente). Darüber hinaus muss eine Abschätzung für zukünftige Gelenkwinkel des Roboters $\theta_{i,p}$ erfolgen, da diese ebenfalls vom Schätzalgorithmus benötigt werden. Für den LoLiMot Algorithmus

¹ Ein PRBS Signal, ist ein binäres Signal, welches das Frequenzspektrum von weißem Rauschen approximiert.

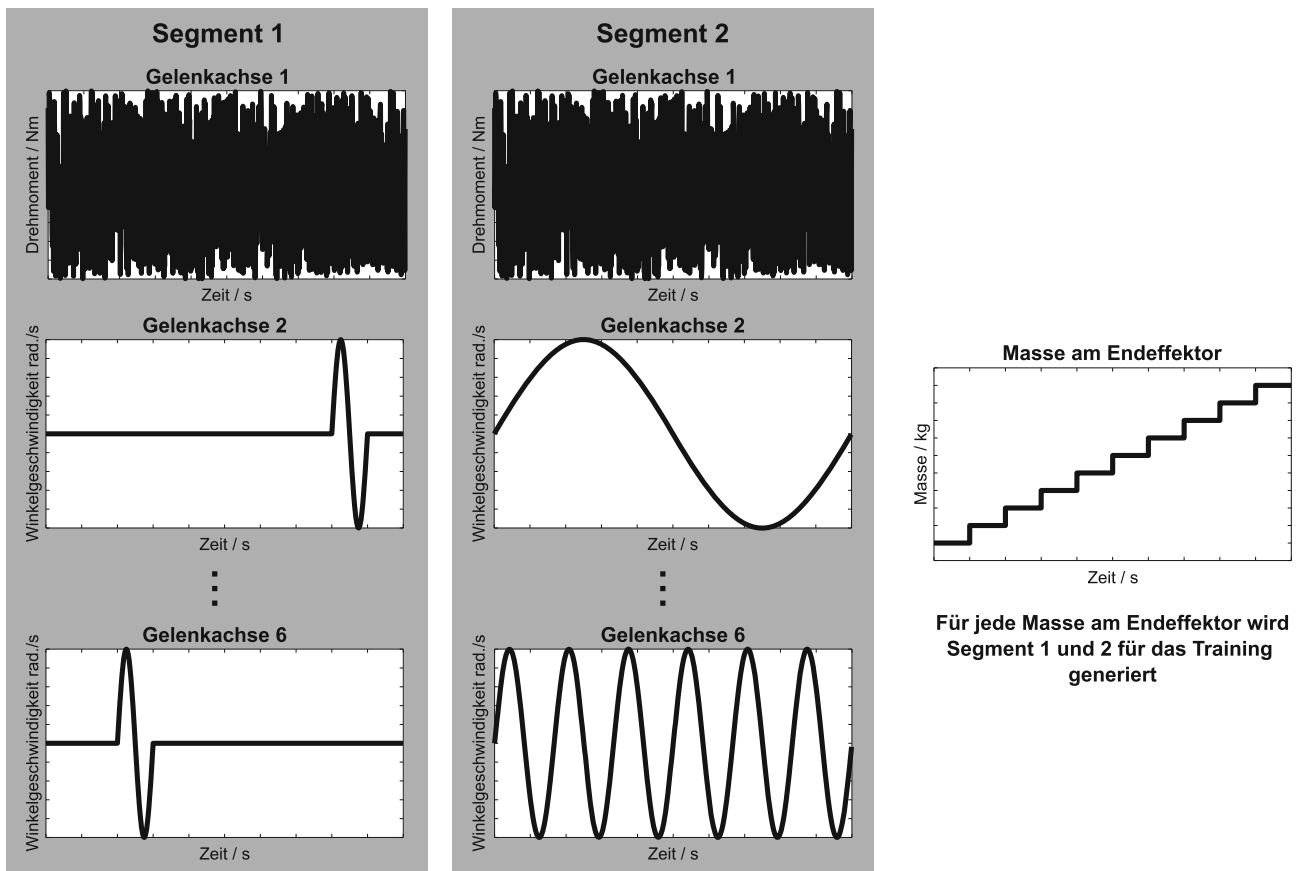


Abb. 3 Aufbau und Segmentierung der Daten für das Training des LoLiMot Algorithmus zur Schätzung der Gelenkwinkel des Roboters für eine Gelenkachse

mus erfolgt dies über den Gradienten der vergangenen zwei Messwerte $\theta_{-1,m}, \theta_{0,m}$, den aktuellen Messwert $\theta_{1,m}$ und Faktor k , um eine Schätzung der zukünftigen Gelenkwinkel des Roboters zu erzeugen. Der Faktor k beschreibt hierbei den diskreten Abstand zwischen den letzten Messungen bzw. Abtastwerten der Gelenkwinkel des Roboters, somit berechnet sich die Projektion des i -ten Gelenkwinkels zu:

$$\theta_{i,p} = \theta_{1,m} + k \cdot \Delta\theta = \theta_{1,m} + k(\theta_{0,m} - \theta_{-1,m}). \quad (2)$$

Bei ausreichend kleiner Abtastzeit des Reglers und nicht zu hoher Systemdynamik, ist der Einfluss des Projektionsfehlers aus Gl. 2 vernachlässigbar klein und kann durch die MPR kompensiert werden. An Knickstellen im Referenzprofil, sowie am Anfang der Regelung kann es zu höheren Abweichungen kommen, welche zuerst kompensiert werden müssen. Für die Auslegung gilt es jedoch abzuwägen und eine sinnvolle Abschätzung zu treffen, da sonst zu hohe Stellgrößen berechnet werden oder Diskontinuitäten in der Schätzung auftreten.

Als zweiter Schritt wird der Regler der MPR parametrisiert. Hierzu zählen die Integration eines Referenzmodells und die Auswahl eines Optimierungsalgorithmus. Das Referenzmodell kann als physikalisches Modell oder Datensatz bzw. Array hinterlegt werden. Wo-

bei die zweite Variante erheblich Berechnungszeit einspart und vergleichsweise einfach zu implementieren ist. Im Folgenden ist das Referenzmodell als Übertragungsfunktion hinterlegt, welches die geforderten Ausgangsgrößen $y_{i,\text{ref}}$ für alle Gelenke berechnet. Mit den geschätzten Systemzuständen \hat{y}_i und der gewünschten Dynamik $y_{i,\text{ref}}$ werden dann über den Optimierer alle Steuergrößen u_i für die Servomotoren berechnet. Hierfür wird zunächst die euklidische Norm zwischen $y_{i,\text{ref}}$ und \hat{y}_i innerhalb des Prädiktionshorizonts N_p gebildet.

$$\|y_{i,\text{norm}}\| = \sqrt{\sum_{i=1}^{N_p} \|y_{i,\text{ref}} - \hat{y}_i\|} \quad (3)$$

Der Prädiktionshorizont beschreibt die Anzahl an Abtastschritten, für welche das LoLiMot-Modell die Ausgangsgröße schätzt. Für den Prädiktionshorizont muss gelten, dass $N_p \geq N_{\text{sys}}$, wobei N_{sys} die Ordnung des zu regelnden Systems beschreibt. Des Weiteren gilt, dass der Regler nur einen begrenzten Stellhorizont ($N_c \leq N_p$) berechnen kann [17]. Über die Norm für jedes Gelenk i wird die Kostenfunktion somit über den Prädiktionshorizont N_p

$$J(y_i(0), u_i) = \sum_{i=0}^{N_p} \|y_{i,\text{ref}} - \hat{y}_i\| + r \cdot \sum_{i=0}^{N_p} \|u_i\| \quad (4)$$

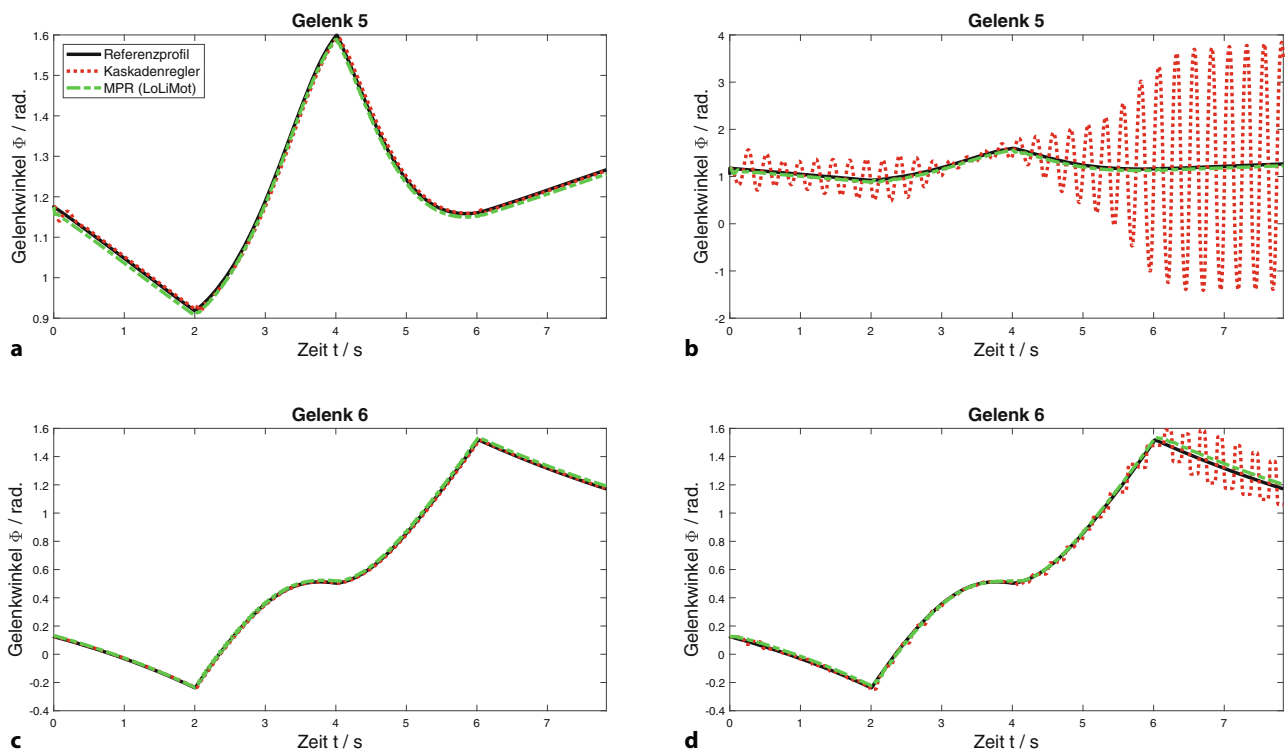


Abb. 4 Vergleich zwischen Referenzprofil, Kaskadenregelung und der entwickelten MPR mit LoLiMot für einen Roboter vom Typ ABB IRB140 für die Gelenkachsen 5 und 6 mit einer

mechanischen Last von 0,5 kg (a,c) und mit einer mechanischen Last von 5 kg (b,d) am Endeffektor

definiert. Für die Optimierungsvorschrift folgt mit der Kostenfunktion J somit

$$u_i = \underset{u_i}{\operatorname{argmin}} J(y_i(0), u_i) \quad (5)$$

Die Kostenfunktion J muss über den Optimierer also minimiert werden, sodass bei berechneten Stellgrößen u_i die Norm zwischen geforderter Systemdynamik $y_{i,\text{ref}}$ und geschätzter Systemantwort \hat{y}_i minimal wird. Für den Vorhersagehorizont N_p wird dem Roboter so die gewünschte Systemdynamik aufgeprägt. Die Stellgrößen u_i werden noch über den Faktor r gewichtet. In Gl. 5 können zusätzlich Begrenzungen oder Strafterme für die Stellgrößen u_i ergänzt werden. Für den Roboter können zu große Sprünge der Stellgrößen zu instabilem Verhalten führen, daher werden diese betragsmäßig $|u_i| \leq u_{i,\text{max}}$ begrenzt. Der verwendete Algorithmus zur Berechnung der optimalen Stellgrößen u_i basiert auf dem Verfahren des Goldenen Schnitts und der parabolischen Interpolation [18, 19].

Simulative Verifikation der Regelung an einem sechsachsigen Industrieroboter

Wie bereits kurz erläutert, ist die Testapplikation für den Roboter der Transport von verschiedenen Werkstücken. Um die entwickelte MPR mit LoLiMot Modellen zu validieren, muss der Roboter in der Simulationsumgebung je ein Gewicht von 0,5 kg und 5 kg transportieren. Das Referenzprofil wurde dabei

so implementiert, dass jede Gelenkachse angesprochen wird. Alle möglichen Wechselwirkungen auf die Güte der adaptiven Regelung können so abgebildet bzw. analysiert werden. Abb. 4 zeigt die Ergebnisse der MPR (grün, gestrichelt) im Vergleich zu einer konventionellen Kaskadenregelung (rot, gestrichelt) für die Gelenkachsen 5 und 6 des ABB IRB 140, anhand eines Referenzprofils (schwarz, durchgezogen). Für ein Lastgewicht von 0,5 kg am Endeffektor sind beide Regler ähnlich performant und es treten nur geringe Winkelfehler zwischen Referenzprofil und Gelenkwinkel auf. Wird das Lastgewicht jedoch auf 5 kg geändert, so zeigt sich die MPR als äußerst robust. Trotz geänderter Systemdynamik (Kinematik des Roboters) kann eine gleich hohe Güte erzielt werden. Im Vergleich hierzu neigt die Kaskadenregelung besonders bei Gelenkachse 5 zur Instabilität und Schwingungen treten auf. Dieser Effekt ist speziell für Gelenkachse 5 sichtbar, da der Einfluss der Massenträgheit und Erdbeschleunigung auf Achse 5 am höchsten sind.

Ein essentieller Parameter in der Auslegung für die MPR mit LoLiMot ist die Anzahl der Modelle im Modellnetz. Je nach Anzahl der lokalen Submodelle (LLM) des Modells und Anzahl der historischen Messwerte steigt der Berechnungsaufwand und der Speicherbedarf für die online Prognose erheblich an. Dies limitiert wiederum die maximal mögliche Abtastzeit des Reglers. Darüber hinaus kann bei zu hoher Modellanzahl das Modellnetz zu einer sogenannten Überanpassung (Engl.: Overfitting) neigen und

Effekte² abbilden, die nicht dem System zugeordnet werden können. Es empfiehlt sich daher sowohl die Anzahl der Submodelle als auch die Anzahl der historischen Messwerte zu begrenzen. Zur Bestimmung der Modellanzahl (LLMs) und Anzahl der benötigten Messwerte existiert eine Vielzahl an modellbasierten und datengetriebenen Verfahren zur Bestimmung der Ordnung [20–22]. Für LoLiMot Netze im diskutierten Anwendungsszenario eignen sich die Berechnung der Lipschitz-Nummer und das Verfahren des False-Nearest-Neighbours besonders gut.

Ein weiterer Einflussfaktor auf die Güte der Regelung ist die Anzahl der Funktionsaufrufe des Optimierungsalgorithmus, da für jeden Abtastzeitpunkt die optimalen Stellgrößen für die Gelenkachsen berechnet werden müssen. Je nach Abtastzeit des Reglers muss die Anzahl der Funktionsaufrufe begrenzt werden, um die Echtzeitfähigkeit zu gewährleisten.

Experimentelle Verifikation der Regelung an einem einachsigen Gelenkprüfstand

Im letzten Schritt soll die MPR mit LoLiMot an einem einachsigen Gelenkprüfstand experimentell validiert werden. Der Prüfstand zur Validierung der MPR mit LoLiMot kann Abb. 5 entnommen werden. Für die Gelenkachse kann die Masse durch Montage von Gewichtsplatten auf beiden Seiten asymmetrisch variiert werden. Die Ansteuerung des Servomotors und das Einlesen der Sensordaten erfolgt über eine digitale/analoge Signalschnittstelle. Der Regler und die LoLiMot Modelle sind auf einer eigenen Recheninstanz implementiert, d. h. die Signalschnittstelle liest und schreibt nur die Sensor- und Steuersignale.

Das Training für die LoLiMot Modelle erfolgt analog zum Schema der Daten aus Abb. 3, d. h. die Gelenkachse wird mit einem Drehmoment angeregt und die Masse für einzelne Sequenzen an den Montagepunkten asymmetrisch variiert. Neben den Gelenkwinkeln erhält das LoLiMot Modell also auch Informationen über die Masse und Verteilung der Masse am Endeffektor als Parameter. Aufgrund der hohen Dynamik im Ansprechverhalten des Servomotors für das Gelenk muss die Pulsdauer zwischen einzelnen Signalwerten des Drehmoments auf 1 s geändert werden. Die maximalen Stellgrößen für den digitalen Regler werden zusätzlich auf 20 % des möglichen Stellbereichs begrenzt und im Optimierungsalgorithmus hinterlegt (siehe Gl. 4).

Zu Validierung des Reglers wird ein Referenzprofil für die Gelenkachse vorgegeben. Das Profil soll zunächst ohne Masse und anschließend mit einer unsymmetrisch montierten Lastmasse von ca. 2 kg abgefahren werden. Hierbei soll die Gelenkachse zunächst aus der Nullposition auf einen Winkel von +81° fahren

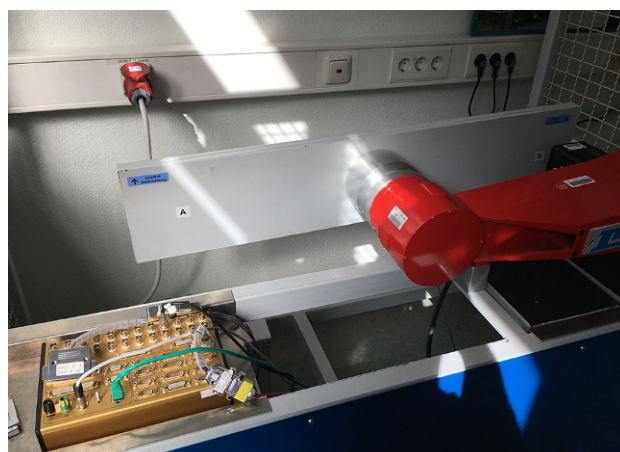


Abb. 5 Prüfaufbau des einachsigen Robotergelenks mit unsymmetrischer Massenvariation am Endeffektor zur experimentellen Validierung der entwickelten MPR mit LoLiMot Schätzmodellen

und anschließend rampenförmig mit $\pm 9^\circ$ um 90° verfahren. Abb. 6 zeigt das Referenzprofil (schwarz), das Sensorsignal der Kaskadenregelung (rot, gestrichelt), das Sensorsignal der MPR mit LoLiMot (grün, gestrichelt) und die berechneten Stellgrößen der Regler für den unbelasteten Fall. Der Vergleich der beiden Regler zeigt, dass die MPR mit LoLiMot dynamischer ist und die geforderten Sollwinkel schneller erreicht als die Kaskadenregelung. Die Stellgrößen der MPR mit LoLiMot sind kleiner oder ähnlich groß wie die Stellgrößen der Kaskadenregelung und die Begrenzung von 20 % des möglichen Stellbereichs wird ebenfalls eingehalten. Neben einer höheren Dynamik der Regelung wird also auch teilweise weniger Leistung für die Ansteuerung benötigt und so die Effizienz gesteigert.

In Abb. 7 sind das Referenzprofil (schwarz), das Sensorsignal der Kaskadenregelung (rot, gestrichelt), das Sensorsignal der MPR mit LoLiMot (grün, gestrichelt) und die zugehörigen Stellgrößen für eine Belastung von ca. 2 kg am Endeffektor dargestellt. Trotz geänderter Systemdynamik des Roboters erweist sich die MPR mit LoLiMot als äußerst robust und erzielt eine gleichbleibend hohe Regelgüte. Die Stellgrößen der MPR sind ebenfalls in einer ähnlichen Größenordnung für den belasteten Fall und die Effizienz bleibt gleich. Auch für den belasteten Fall ist die Kaskadenregelung vergleichsweise träge und kann die geforderten Sollwinkel nicht erreichen.

Zusammenfassung

Im Rahmen dieses Artikels wurde ein datenbasiertes, adaptives Regelungskonzept für mehrachsige Industrieroboter untersucht und an einem einachsigen Prüfaufbau experimentell validiert. Für die Entwicklung und Parametrisierung des adaptiven Reglers wurde zunächst eine Simulation bestehend aus der Kinematik des Roboters, den Servomotoren, Reglern und Sensorik vorgestellt. Davon wurden die Anforde-

² Typische Effekte, die durch Überanpassung der Modellnetze abgebildet werden sind bspw. das Rauschen von Sensoren oder A/D- bzw. D/A-Wandlern.

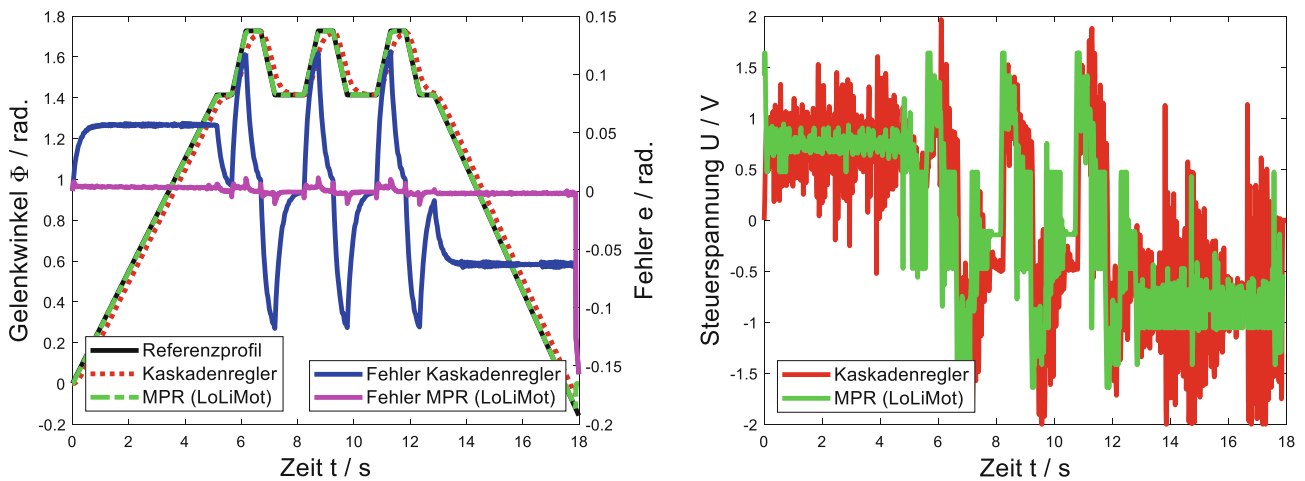


Abb. 6 Vergleich der Regelleistung und zugehörigen Stellgrößen zwischen Kaskadenregelung und MPR mit LoLiMot ohne mechanische Last am Endeffektor der Gelenkachse

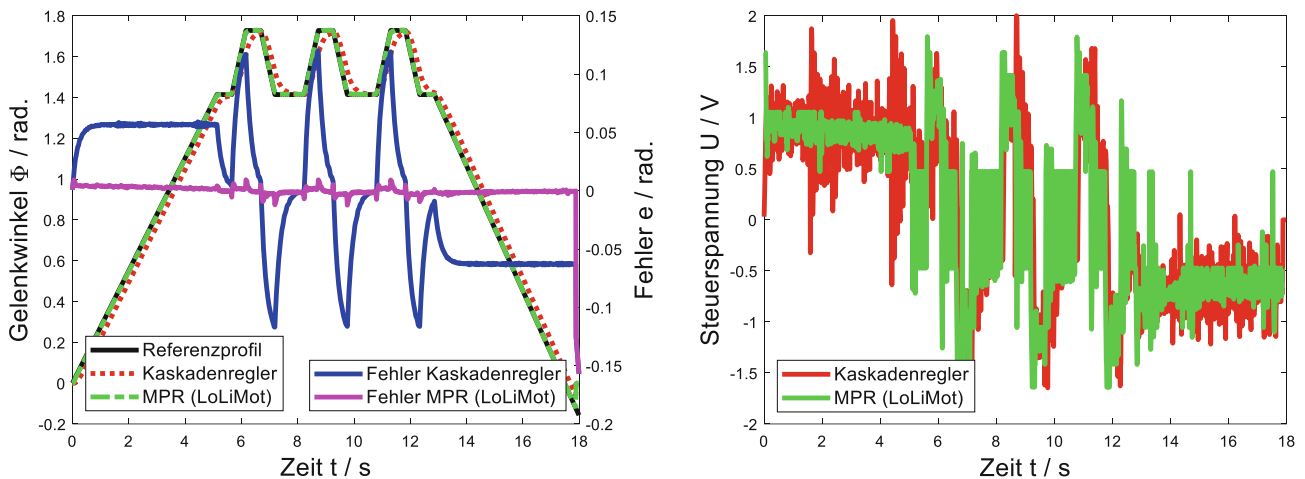


Abb. 7 Vergleich der Regelleistung und zugehörigen Stellgrößen zwischen Kaskadenregelung und MPR mit LoLiMot mit einer zusätzlichen unsymmetrischen mechanischen Last von 1 kg am Endeffektor der Gelenkachse

rungen an die datenbasierten Schätzer (Qualität der Trainingsdaten, Ordnung des Modells, Anregungssignal etc.) abgeleitet. Des Weiteren wurden kritische Einflussparameter auf die Stabilität und Güte der Regelung identifiziert und analysiert. Im direkten Vergleich zu einer Kaskadenregelung zeigte sich die datenbasierte, adaptive Regelung am einachsigen Prüfaufbau dynamischer und robust gegenüber geänderten Randbedingungen oder Systemparametern des Roboters. Selbst bei wechselnder Last wurde eine gleich hohe Regelleistung erzielt. Des Weiteren liegt die benötigte Leistung in einer ähnlichen Größenordnung zur Kaskadenregelung, d. h. bei höherer Regelgüte wird mit dem vorgestellten Regler dieselbe Leistung benötigt. Die Hauptkenntnisse dieses Artikels lassen sich wie folgt zusammenfassen

- Über die Trainingsdaten muss möglichst der komplette Parameterbereich des Roboters (Last am Endeffektor) abgebildet werden. Je nach Parameterbe-

reich müssen die Trainingsdaten hierfür entsprechend generiert werden.

- Sprünge der Lastmasse am Endeffektor können durch die sehr gute Interpolationsfähigkeit der lokalen Modellnetze kompensiert werden. Wie in Abb. 3 dargestellt kann der Algorithmus trotz der Sprünge der Lastmasse robust und zuverlässig die Systemgrößen schätzen.
- Um die Wechselwirkungen der Gelenkachsen untereinander abzubilden empfiehlt es sich die Trainingsdaten in zwei einzelne Segmente zu unterteilen. Hierbei muss der Einfluss jedes einzelnen Gelenks als auch die Einflüsse der Gelenke untereinander abgebildet werden (vgl. Abb. 3).
- Die Modellanzahl und Anzahl historischer Messwerte muss vorab über eines der etablierten Verfahren [20–22] bestimmt werden. Bei einer zu hohen Modellordnung oder zu vielen historischen Messwerten muss die Abtastzeit des Reglers entsprechend angepasst werden.

- Als Anregungssignal für die Trainingsdaten der linearen, lokalen Modellnetze eignet sich eine amplitudenmodulierte Pseudo-Randomisierte-Binärssequenz am besten. Andere Signale (z. B. Dreieck oder Sprung) führten zu schlechteren Netzqualitäten.
- Die Abtastzeit und Anzahl der Funktionsaufrufe für den Optimierungsalgorithmus sind kritisch in Bezug auf die Echtzeitfähigkeit des Reglers. Hier muss die Abtastzeit mit den lokalen Modellnetzen synchronisiert werden. Des Weiteren wird die Anzahl der Funktionsaufrufe des Optimierers hierdurch begrenzt.

Die entwickelte Regelung bietet nun Potenzial zur Weiterentwicklung und Optimierung von einzelnen Teilaspekten. Hierunter fallen die folgenden Aspekte des Reglers:

- Der Optimierer kann speziell für die gegebene Kostenfunktion und den Suchbereich angepasst werden, um so effizienter und schneller optimale Stellgrößen zu berechnen.
- Für jede Gelenkachse muss sowohl der Schätzer als auch die Optimierung in jedem Zeitschritt berechnet werden. Hierbei können bestimmte Rechenschritte parallelisiert werden, um wieder Rechenressourcen zu sparen und die Abtastzeit einzuhalten.

Funding Open Access funding enabled and organized by Projekt DEAL.

Open Access Dieser Artikel wird unter der Creative Commons Namensnennung 4.0 International Lizenz veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Artikel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.

Weitere Details zur Lizenz entnehmen Sie bitte der Lizenzinformation auf <http://creativecommons.org/licenses/by/4.0/deed.de>.

Literatur

1. Celikag H, Sims ND, Ozturk E (2019) Chatter suppression in robotic milling by control of configuration dependent dynamics. *Procedia CIRP* 82:521–526. <https://doi.org/10.1016/j.procir.2019.04.053>
2. Yang T, Sun N, Fang Y, Xin X, Chen H (2022) New adaptive control methods for n-link robot manipulators with online gravity compensation: design and experiments. *IEEE Trans Ind Electron* 69(1):539–548. <https://doi.org/10.1109/TIE.2021.3050371>
3. Nelles O, Fink A, Isermann R (2000) Local linear model trees (LOLIMOT) toolbox for nonlinear system identification. *IFAC Proc* 33(15):845–850. [https://doi.org/10.1016/S1474-6670\(17\)39858-0](https://doi.org/10.1016/S1474-6670(17)39858-0)
4. Adamy J (2018) Nichtlineare Systeme und Regelungen, 3. Aufl. Springer Vieweg, Berlin <https://doi.org/10.1007/978-3-642-45013-6>
5. Zidan A, Kaczor D, Tappe S, Ortmaier T (2019) Optimization of a P/PI Cascade Motion Controller for a 3-DOF Delta Robot. In: Schüppstuhl T, Tracht K, Roßmann J (Hrsg) Tagungsband des 4. Kongresses Montage Handhabung Industrieroboter. Springer Vieweg, Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-59317-2_22
6. Ren Y, Chen Z, Liu Y, Gu Y, Jin M, Liu H (2017) Adaptive hybrid position/force control of dual-arm cooperative manipulators with uncertain dynamics and closed-chain kinematics. *J Franklin Inst* 354(17):7767–7793. <https://doi.org/10.1016/j.jfranklin.2017.09.015>
7. Huang L, Jiang R (2013) A new method of inverse kinematics solution for industrial 7DOF robot. In: Proceedings of the 32nd Chinese Control Conference, S 6063–6065
8. Pagilla PR, Tomizuka M (1995) Adaptive control of two robot arms carrying an unknown object. In: Proceedings of 1995 IEEE International Conference on Robotics and Automation, Bd. 1, S 597–602 <https://doi.org/10.1109/ROBOT.1995.525349>
9. Wang J, Wei Q, Zhao Q, Lou Z-E (2021) Adaptive PID switched tracking control of robot manipulator. In: 2021 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC), S 380–384 <https://doi.org/10.1109/SPAC53836.2021.9539974>
10. Arena P, Sueri P, Taffara S, Patanè L (2021) MPC-based control strategy of a neuro-inspired quadruped robot. In: 2021 International Joint Conference on Neural Networks (IJCNN), S 1–8 <https://doi.org/10.1109/IJCNN52387.2021.9533394>
11. Li G, Yu J, Chen X (2021) Adaptive fuzzy neural network command filtered impedance control of constrained robotic manipulators with disturbance observer. In: IEEE Transactions on Neural Networks and Learning Systems. <https://doi.org/10.1109/TNNLS.2021.3113044>
12. Pan Q, Ma B, Dong B, Ren X, Li Y (2021) Observer-critic-based event-triggered decentralized optimal control of modular robot manipulators. In: 2021 40th Chinese Control Conference (CCC), S 3912–3918 <https://doi.org/10.23919/CCC52363.2021.9550013>
13. Ruan X, Song K (2011) An adaptive extended Kalman filter for attitude estimation of self-balancing two-wheeled robot. In: 2011 International Conference on Electric Information and Control Engineering, S 4760–4763 <https://doi.org/10.1109/ICEICE.2011.5777681>
14. Bätz G, Weber B, Scheint M, Wollherr D, Buss M (2013) Dynamic contact force/torque observer: sensor fusion for improved interaction control. *Int J Robot Res*. <https://doi.org/10.1177/0278364913482015>
15. Miao Y, Li Z (1991) Neural networks based adaptive predictors for nonlinear dynamical systems. In: 1991 IEEE International Joint Conference on Neural Networks, Bd. 1, S 777–782 <https://doi.org/10.1109/IJCNN.1991.170495>
16. Lin Y, Chang J, Lin C (2013) Identification and prediction of dynamic systems using an interactively recurrent self-evolving fuzzy neural network. *IEEE Trans Neural Netw Learning Syst* 24(2):310–321. <https://doi.org/10.1109/TNNLS.2012.2231436>

17. Marzooghi H, Raoufat M, Dehghani M, Elahi GH (2012) Dynamic modeling of solid oxide fuel cell stack based on local linear model tree algorithm. *Int J Hydrogen Energy* 37(5):4367–4376. <https://doi.org/10.1016/j.ijhydene.2011.11.149>
18. Pulikkoonattu R, Subramanian HK, Laxman S (2008) Least square based piecewise parabolic interpolation for timing synchronization. In: 2008 IEEE Radio and Wireless Symposium, S 155–158 <https://doi.org/10.1109/RWS.2008.4463452>
19. Chang Y (2009) N-dimension golden section search: its variants and limitations. In: 2009 2nd International Conference on Biomedical Engineering and Informatics, S 1–6 <https://doi.org/10.1109/BMEI.2009.5304779>
20. Hassibi B, Stork DG, Wolff GJ (1993) Optimal brain surgeon and general network pruning. In: IEEE International Conference on Neural Networks, Bd. 1, S 293–299 <https://doi.org/10.1109/ICNN.1993.298572>
21. Macas M, Moretti F (2016) Sensitivity based selection of inputs and delays for NARX models. In: Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies, S 15–20 <https://doi.org/10.4108/eai.3-12-2015.2262480>
22. Nelles O (2001) Nonlinear system identification <https://doi.org/10.1007/978-3-662-04323-3>



Christopher Ranisch, Nach seinem Studium der Elektrotechnik an der Hochschule Darmstadt (B.Eng. 2016, M.Sc. 2017) war Herr Ranisch zunächst von 2017 bis 2021 als Wissenschaftlicher Mitarbeiter in der Abteilung Strukturtechnik und Schwingungstechnik am Fraunhofer-Institut LBF in Darmstadt tätig. Fokus der Forschung lag hierbei auf der Entwicklung und Analyse von elektrodynamischen Aktoren und der Implementierung von adaptiven, modellbasierten Reglern. Seit 2021 ist Herr Ranisch Laboringenieur am Fachbereich Elektrotechnik und Informationstechnik und forscht zu verschiedenen Fragestellungen im Bereich der Antriebstechnik und Robotik.



Heiko Koch, hat 2013 an der Technischen Universität Chemnitz auf dem Gebiet der sensorgestützten Regelung von Robotern promoviert. Seit 2018 lehrt er an der Hochschule Darmstadt im Fachbereich Elektrotechnik und Informationstechnik im Bereich der Robotik.



Timm Streul, hat Mechatronik (Bachelor of Science, Vertiefung Robotik, 2020) und Elektrotechnik (Master of Science, Vertiefung Automatisierungstechnik, 2021) an der Hochschule Darmstadt studiert. Die Themenschwerpunkte der bisherigen Arbeiten von Herrn Streul umfassen die Entwicklung von Simulationsmodellen mechatronischer Systeme und adaptiver Regler auf Basis von Künstlicher Intelligenz.