4 Roboterregelung

Ein Roboter kann eine Reihe von fundamental unterschiedlichen Aufgaben ausführen, wobei für jede Aufgabe ein geeignetes Regelungskonzept eingesetzt werden muss.

Zunächst kann ein Roboter vorgegebene Bewegungen – beschrieben im Konfigurations- oder im Arbeitsraum – ausführen, um so ein Objekt zu manipulieren oder ein Werkzeug entlang der Oberfläche eines Objektes zu führen. Dies wird durch die sogenannte *Bewegungsregelung* bewerkstelligt.

Ist der Endeffektor des Roboters im physischen Kontakt mit der Umgebung, so können mithilfe einer *Kraftregelung* gewünschte Kräfte und Drehmomente auf die Umgebung ausgeübt werden. Dies ist zum Beispiel notwendig, wenn zwei Objekte mit definierten Kräften zusammengefügt werden sollen oder wenn die Oberfläche eines Werkstücks mit vorgegebenen Kräften geschliffen oder poliert werden soll.

Darüber hinaus wird häufig bei kollaborativen Robotern die sogenannte *Impedanzregelung* eingesetzt. Eine mechanische Impedanz beschreibt den Zusammenhang zwischen der Positionsabweichung eines mechanischen Systems aus einer Ruhelage und den daraus folgenden Reaktionskräften. Ein Beispiel für ein mechanisches System ist das Feder-Masse-Dämpfer-System: Wird die Masse aus der Ruhelage heraus bewegt, so entstehen Reaktionskräfte, die der Auslenkung entgegenwirken. Mithilfe der Impedanzregelung kann sich ein Roboter im Konfigurations- oder im Arbeitsraum wie ein virtuelles mechanisches System verhalten, z. B. wie ein Feder-Masse-Dämpfer-System mit vorgegebener Masse, Dämpfung und Steifigkeit. Dies ist nützlich, wenn ein Roboter ein Objekt bearbeiten soll, dessen Form nicht exakt bekannt ist, oder wenn der Roboter als haptische Benutzerschnittstelle verwendet wird.

Schließlich können die obigen Regelungskonzepte auch kombiniert werden, indem unterschiedliche Regler für die jeweiligen Koordinaten des Arbeitsraums verwendet werden. So kann ein Werkzeug bei der Oberflächenbearbeitung durch den Roboter mithilfe einer Kraftregelung angedrückt und mit einer Bewegungsregelung entlang der Oberfläche geführt werden.

Dieses Kapitel befasst sich mit der Bewegungsregelung von Robotersystemen für kinematisch nicht-redundante und redundante Roboter. In den beiden folgenden Abschnitten 4.1 und 4.2 werden zunächst geeignete Beschreibungen für das dynamische Robotermodell im Konfigurationsraum bzw. im Arbeitsraum eingeführt. Anschließend werden häufig verwendete Regler für die Bewegungsregelung und Impedanzregelung vorgestellt.

4.1 Konfigurationsraum

4.1.1 Dynamisches Modell im Konfigurationsraum

Die Bewegungsgleichungen eines Roboters mit $n = \dim(\mathbf{q})$ Freiheitsgraden lassen sich im Konfigurationsraum in der Form

$$\mathbf{D}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} \tag{4.1}$$

anschreiben [4.1]. Es wird angenommen, dass der Roboter *vollaktuiert* ist, d.h. es gibt für jeden mechanischen Freiheitsgrad einen unabhängigen Stelleingang $\boldsymbol{\tau}^{\mathrm{T}} = [\tau_1, \tau_2, \ldots, \tau_n]$. Hier ist zu beachten, dass sich die generalisierten Kräfte $\boldsymbol{\tau}$ im Allgemeinen noch in die dissipativen Kräfte und Momente $\boldsymbol{\tau}_d$, die Stelleingänge $\boldsymbol{\tau}_c$ und die extern auf den Roboter wirkenden Kräfte und Momente $\boldsymbol{\tau}_{ext}$ aufspalten lassen, d.h.

$$\boldsymbol{\tau} = \boldsymbol{\tau}_d + \boldsymbol{\tau}_c + \boldsymbol{\tau}_{ext} \ . \tag{4.2}$$

Für die Massenmatrix $\mathbf{D}(\mathbf{q})$, die Matrix der Zentrifugal- und Coriolisterme $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ und den Vektor der Potentialkräfte $\mathbf{g}(\mathbf{q})$ gelten folgende Eigenschaften und Zusammenhänge:

- $\mathbf{D}(\mathbf{q})$ ist eine positiv definite und damit invertierbare Matrix. D. h. für alle $\dot{\mathbf{q}} \neq \mathbf{0}$ gilt $\dot{\mathbf{q}}^{\mathrm{T}}\mathbf{D}(\mathbf{q})\dot{\mathbf{q}} > 0$, denn die kinetische Energie ist immer positiv.
- $\mathbf{N}(\mathbf{q}, \dot{\mathbf{q}}) = \dot{\mathbf{D}}(\mathbf{q}) - 2\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ ist eine schiefsymmetrische Matrix, d.h. $\mathbf{N}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{N}^{\mathrm{T}}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{0}$. Damit gilt $\dot{\mathbf{q}}^{\mathrm{T}}\mathbf{N}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} = 0$ für alle $\dot{\mathbf{q}}$ sowie $\dot{\mathbf{D}}(\mathbf{q}) = \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{C}^{\mathrm{T}}(\mathbf{q}, \dot{\mathbf{q}})$.
- $\mathbf{g}^{\mathrm{T}}(\mathbf{q}) = \frac{\partial}{\partial \mathbf{q}} V(\mathbf{q})$ mit der potenziellen Energie $V(\mathbf{q}) > 0$.

4.1.2 Dezentrale Regler

Wie bereits in Abschnitt 1.3.1 eingeführt wurde, kann die Regelung im Konfiguration nun zentral oder dezentral erfolgen. Bei der *dezentralen Regelung* wird jedes Gelenk des Roboters getrennt voneinander geregelt und die mechanische Kopplung der Glieder bei der Roboterbewegung wird als Störung aufgefasst. Dadurch reduziert sich der Reglerentwurf auf den Eingrößenfall (SISO) mit der jeweiligen (verallgemeinerten) Gelenkskoordinate q_j als Ausgangsgröße und der generalisierten Kraft bzw. dem Moment τ_j, $j = 1, \ldots, n$ als Stelleingang.

Typischerweise liegt dem Entwurf ein einfaches Modell zweiter Ordnung zu Grunde. In der Praxis hat es sich bewährt, hochdynamische unterlagerte Geschwindigkeitsregler einzusetzen, welche etwa um den Faktor 5-10 schneller als der äußere Regelkreis sind. Diese unterlagerten Regler werden meist in Form von einfachen PID- oder PD-Reglern mit Reibungskompensation implementiert. Im Sinne einer klassischen Kaskadenregelung lassen sich dann die unterlagerten Geschwindigkeitsregelkreise idealisiert als Durchschaltung

$$\dot{q}_j = \dot{q}_{d,j} = u_j \ , \quad j = 1, \ldots, n \tag{4.3}$$

mit den gewünschten Gelenksgeschwindigkeiten $u_j = \dot{q}_{d,j}$ als neue Eingangsgrößen betrachten. Mit den idealisierten unterlagerten Geschwindigkeitsregelkreisen reduziert sich das mathematische Modell des Roboters (4.1) zum sogenannten *kinematischen Robotermodell*

$$\dot{\mathbf{q}} = \mathbf{u} \ , \tag{4.4}$$

welches im Wesentlichen n unabhängige Integratoren darstellt. Um nun auf die gewünschte Trajektorie im Konfigurationsraum $q_{d,j}(t)$ zu regeln, kann man das einfache PI-Regelgesetz

$$u_j = -k_1(q_j - q_{d,j}(t)) - k_0 \int (q_j - q_{d,j}(t))\,\mathrm{d}t \tag{4.5}$$

mit geeigneten Koeffizienten $k_0 > 0$ und $k_1 > 0$ verwenden. Liegt auch die Sollgeschwindigkeit $\dot{q}_{d,j}(t)$ der Trajektorie vor, so kann das Regelgesetz noch weiter verbessert werden indem der Term $\dot{q}_{d,j}(t)$ als Vorsteuerung zu (4.5) addiert wird.

4.1.3 PD-Regeler mit Gravitationskompensation

Ein sehr bekanntes und einfaches Verfahren zur Stabilisierung eines *Arbeitspunktes* im Konfigurationsraum (d. h. $\mathbf{q} = \mathbf{q}_d$ und $\dot{\mathbf{q}} = \dot{\mathbf{q}}_d = \mathbf{0}$) ist die *PD-Regelung mit Gravitationskompensation*

$$\boldsymbol{\tau} = \underbrace{\mathbf{g}(\mathbf{q})}_{\text{Gravitationskomp.}} - \underbrace{\mathbf{K}_d\dot{\mathbf{e}}_q}_{\text{D-Anteil}} - \underbrace{\mathbf{K}_p\mathbf{e}_q}_{\text{P-Anteil}} \tag{4.6}$$

mit dem Regelfehler $\mathbf{e}_q = \mathbf{q} - \mathbf{q}_d$ und $\dot{\mathbf{e}}_q = \dot{\mathbf{q}} - \dot{\mathbf{q}}_d = \dot{\mathbf{q}}$ und den positiv definiten (symmetrischen) Reglermatrizen $\mathbf{K}_d$ und $\mathbf{K}_p$. Aufgrund des Terms $\mathbf{g}(\mathbf{q})$ in (4.6) muss das Regelgesetz von einer zentralen Recheneinheit ausgewertet werden, in welcher die Messwerte aller Gelenkswinkel $\mathbf{q}$ vorliegen. Damit zählt dieses Verfahren zu den zentralen Reglern im Konfigurationsraum.

Setzt man (4.6) in (4.1) ein, so erhält man für den geschlossenen Kreis

$$\mathbf{D}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \mathbf{g}(\mathbf{q}) - \mathbf{K}_d\dot{\mathbf{q}} - \mathbf{K}_p\mathbf{e}_q \ . \tag{4.7}$$

Man erkennt, dass durch das Regelgesetz (4.6) der Gravitationsterm $\mathbf{g}(\mathbf{q})$ gekürzt und mit $\boldsymbol{\tau}_p = \mathbf{K}_p\mathbf{e}_q$ ein weiterer Term hinzugefügt wird, der als Federkraft mit der Steifigkeitsmatrix $\mathbf{K}_p$ interpretiert werden kann. Im Speziellen lassen sich die Einträge von $\mathbf{K}_p$ als Federsteifigkeiten von fiktiven Federelementen im Konfigurationsraum interpretieren. Diese Elemente sind zwischen der Konfiguration des Roboters $\mathbf{q}$ und der gewünschten Konfiguration des Roboters $\mathbf{q}_d$ gespannt. Für die Kraft $\boldsymbol{\tau}_p$ gilt

$$\boldsymbol{\tau}_p = \mathbf{K}_p\mathbf{e}_q = \left(\frac{\partial}{\partial\mathbf{q}}V_p\right)^{\mathrm{T}} \quad \text{mit} \quad V_p = \frac{1}{2}\mathbf{e}_q^{\mathrm{T}}\mathbf{K}_p\mathbf{e}_q \ , \tag{4.8}$$

d. h. $\boldsymbol{\tau}_p$ stellt eine Potenzialkraft dar, wie sie in der Vorlesung *Modellbildung* eingeführt wurde, siehe [4.1].

Aus (4.7) ist nicht unmittelbar einsichtig, dass die Ruhelage $\mathbf{q} = \mathbf{q}_d$ und $\dot{\mathbf{q}} = \mathbf{0}$ des geschlossenen Kreises auch tatsächlich asymptotisch stabil ist. Um dies zu zeigen, wird in einem ersten Schritt für den geschlossenen Kreis (4.7) eine Energiefunktion der Form

$$E_c = \underbrace{\frac{1}{2}\dot{\mathbf{q}}^{\mathrm{T}}\mathbf{D}(\mathbf{q})\dot{\mathbf{q}}}_{\text{kinetische Energie}} + \underbrace{\frac{1}{2}\mathbf{e}_q^{\mathrm{T}}\mathbf{K}_p\mathbf{e}_q}_{\substack{\text{potentielle Energie}\\ \text{der Feder}}} > 0 \tag{4.9}$$

mit der potentiellen Energie der Feder V_p aus (4.8) angesetzt. Bildet man die zeitliche Ableitung von (4.9), so folgt

$$\begin{aligned}
\dot{E}_c &= \frac{1}{2}\ddot{\mathbf{q}}^{\mathrm{T}}\mathbf{D}(\mathbf{q})\dot{\mathbf{q}} + \frac{1}{2}\dot{\mathbf{q}}^{\mathrm{T}}\dot{\mathbf{D}}(\mathbf{q})\dot{\mathbf{q}} + \frac{1}{2}\dot{\mathbf{q}}^{\mathrm{T}}\mathbf{D}(\mathbf{q})\ddot{\mathbf{q}} + \frac{1}{2}\dot{\mathbf{e}}_q^{\mathrm{T}}\mathbf{K}_p\mathbf{e}_q + \frac{1}{2}\mathbf{e}_q^{\mathrm{T}}\mathbf{K}_p\dot{\mathbf{e}}_q \\
&= \dot{\mathbf{q}}^{\mathrm{T}}\mathbf{D}(\mathbf{q})\ddot{\mathbf{q}} + \frac{1}{2}\dot{\mathbf{q}}^{\mathrm{T}}\dot{\mathbf{D}}(\mathbf{q})\dot{\mathbf{q}} + \dot{\mathbf{e}}_q^{\mathrm{T}}\mathbf{K}_p\mathbf{e}_q \\
&= \dot{\mathbf{q}}^{\mathrm{T}}\underbrace{\left(-\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} - \mathbf{K}_d\dot{\mathbf{q}} - \mathbf{K}_p\mathbf{e}_q\right)}_{\text{gemäß (4.7)}} + \frac{1}{2}\dot{\mathbf{q}}^{\mathrm{T}}\dot{\mathbf{D}}(\mathbf{q})\dot{\mathbf{q}} + \dot{\mathbf{q}}^{\mathrm{T}}\mathbf{K}_p\mathbf{e}_q \\
&= -\dot{\mathbf{q}}^{\mathrm{T}}\mathbf{K}_d\dot{\mathbf{q}} + \underbrace{\frac{1}{2}\dot{\mathbf{q}}^{\mathrm{T}}\underbrace{\left(\dot{\mathbf{D}}(\mathbf{q}) - 2\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\right)}_{\mathbf{N}(\mathbf{q},\dot{\mathbf{q}})=-\mathbf{N}^{\mathrm{T}}(\mathbf{q},\dot{\mathbf{q}})}\dot{\mathbf{q}}}_{=0} \leq 0\ .
\end{aligned} \tag{4.10}$$

Man erkennt, dass wegen der positiven Definitheit von $\mathbf{K}_d$ die Energie E_c von (4.9) im geschlossenen Kreis nur kleiner werden kann und daraus lässt sich über die *Lyapunovtheorie* die asymptotische Stabilität des Arbeitspunktes $\mathbf{q} = \mathbf{q}_d$ und $\dot{\mathbf{q}} = \mathbf{0}$ beweisen.

> *Bemerkung* 4.1. Man kann die Stabilität auch zeigen, wenn statt $\mathbf{g}(\mathbf{q})$ nur der gewünschte Arbeitspunkt bei der Gravitationskompensation $\mathbf{g}(\mathbf{q}_d)$ in (4.6) eingesetzt wird. Im Weiteren gibt es auch adaptive Regelungsverfahren, die es erlauben, gewisse Parameter des Roboters (z. B. die Lastmasse) in Echtzeit mitzuschätzen.

4.1.4 Computed-Torque-Regler

Der PD-Regler mit Gravitationskompensation in Abschnitt 4.1.3 kann nur für die Stabilisierung von Arbeitspunkten $\mathbf{q}_d$ oder für langsam variierende Sollkonfigurationen $\mathbf{q}_d(t)$ eingesetzt werden. Eine Bewegungsregelung im eigentlichen Sinn ist hingegen eine Trajektorienfolgeregelung, beispielsweise mit dem *Computed-Torque-Regler*. Mit diesem Verfahren folgt der Roboter im Konfigurationsraum einer *vorgegebenen Solltrajektorie* $\mathbf{q}_d(t)$. Dabei wird angenommen, dass die Solltrajektorie $\mathbf{q}_d(t)$ zumindest zweifach stetig differenzierbar ist, d.h. $\dot{\mathbf{q}}_d(t)$ und $\ddot{\mathbf{q}}_d(t)$ sind zumindest stetig. Der Computed-Torque-Regler lautet

$$\boldsymbol{\tau} = \mathbf{D}(\mathbf{q})(\ddot{\mathbf{q}}_d(t) - \mathbf{K}_{2q}\dot{\mathbf{e}}_q - \mathbf{K}_{1q}\mathbf{e}_q) + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) \tag{4.11}$$

mit dem Trajektorienfehler $\mathbf{e}_q = \mathbf{q} - \mathbf{q}_d(t)$ bzw. $\dot{\mathbf{e}}_q = \dot{\mathbf{q}} - \dot{\mathbf{q}}_d(t)$ und den positiv definiten Reglermatritzen $\mathbf{K}_{1q}$ und $\mathbf{K}_{2q}$. Ein Vergleich zwischen (4.1) und (4.11) zeigt die strukturelle Ähnlichkeit des Computed-Torque-Reglers zum Robotermodell im Konfigurationsraum. Tatsächlich werden durch den Computed-Torque-Regler die nichtlinearen Terme des Robotermodells $\mathbf{D}(\mathbf{q})$, $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$ und $\mathbf{g}(\mathbf{q})$ kompensiert bzw. gekürzt und es bleibt ein lineares Systemverhalten übrig. Deshalb wird dieses Verfahren auch *inverse*

Dynamik bezeichnet. Dazu müssen allerdings alle statischen und dynamischen Parameter (Abmessungen, Schwerpunkte, Massen, Massenträgheitsmomente) des Roboters sehr genau bekannt sein, was in der Praxis eine Herausforderung darstellt. Weiters handelt es sich beim Computed-Torque-Regler ebenfalls um ein zentrales Regelungskonzept im Konfigurationsraum.

Setzt man (4.11) in (4.1) ein, so ergibt sich

$$\mathbf{D}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \mathbf{D}(\mathbf{q})(\ddot{\mathbf{q}}_d(t) - \mathbf{K}_{2q}\dot{\mathbf{e}}_q - \mathbf{K}_{1q}\mathbf{e}_q) + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) \tag{4.12}$$

bzw.

$$\mathbf{D}(\mathbf{q})\Big(\underbrace{\ddot{\mathbf{q}} - \ddot{\mathbf{q}}_d(t)}_{\ddot{\mathbf{e}}_q} + \mathbf{K}_{2q}\dot{\mathbf{e}}_q + \mathbf{K}_{1q}\mathbf{e}_q\Big) = \mathbf{0}\ . \tag{4.13}$$

Da $\mathbf{D}(\mathbf{q})$ für alle $\mathbf{q}$ positiv definit und damit invertierbar ist, muss für die Fehlerdynamik

$$\ddot{\mathbf{e}}_q + \mathbf{K}_{2q}\dot{\mathbf{e}}_q + \mathbf{K}_{1q}\mathbf{e}_q = \mathbf{0} \tag{4.14}$$

gelten. Wählt man für $\mathbf{K}_{1q}$ und $\mathbf{K}_{2q}$ Diagonalmatrizen in der Form $\mathbf{K}_{1q} = \mathrm{diag}(k_{1q,1}, \ldots, k_{1q,n})$ und $\mathbf{K}_{2q} = \mathrm{diag}(k_{2q,1}, \ldots, k_{2q,n})$ mit positiven Einträgen $k_{1q,j}, k_{2q,j} > 0$, $j = 1, \ldots, n$, dann sind die Fehlerdynamiken der einzelnen Gelenkskoordinaten q_j mit

$$\ddot{e}_{q,j} + k_{2q,j}\dot{e}_{q,j} + k_{1q,j}e_{q,j} = 0\ , \quad j = 1, \ldots, n \tag{4.15}$$

entkoppelt. Schreibt man (4.15) als System von Differenzialgleichungen erster Ordnung (Zustandsdarstellung) in der Form

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} e_{q,j} \\ \dot{e}_{q,j} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 \\ -k_{1q,j} & -k_{2q,j} \end{bmatrix}}_{\mathbf{A}} \begin{bmatrix} e_{q,j} \\ \dot{e}_{q,j} \end{bmatrix}\ , \tag{4.16}$$

so erkennt man, dass die Matrix $\mathbf{A}$ in erster Standardform vorliegt und das Polynom $p_j(s) = s^2 + k_{2q,j}s + k_{1q,j}$ das charakteristische Polynom von $\mathbf{A}$ darstellt. Für $k_{2q,j} > 0$ und $k_{1q,j} > 0$ ist $p_j(s)$ ein Hurwitzpolynom und mit den Koeffizienten $k_{2q,j}$ und $k_{1q,j}$ lassen sich die *Eigenwerte der Trajektorienfehlerdynamik* wie gewünscht vorgeben (Polvorgabe).

> *Bemerkung* 4.2 (Computed-Torque-Regler mit Integralanteil). Man könnte in (4.11) noch einen Integralanteil im Regler hinzufügen, womit das erweiterte Regelgesetz
>
> $$\boldsymbol{\tau} = \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{D}(\mathbf{q})\left(\ddot{\mathbf{q}}_d(t) - \mathbf{K}_{2q}\dot{\mathbf{e}}_q - \mathbf{K}_{1q}\mathbf{e}_q - \mathbf{K}_{0q}\int \mathbf{e}_q\,\mathrm{d}t\right)\ . \tag{4.17}$$
>
> lautet. Wählt man auch $\mathbf{K}_{0q} = \mathrm{diag}(k_{0q,1}, \ldots, k_{0q,n})$ als Diagonalmatrix mit positiven Einträgen $k_{0q,j}$, $j = 1, \ldots, n$, dann müssen die Koeffizienten $k_{0q,j}$, $k_{1q,j}$ und $k_{2q,j}$ so festgelegt werden, dass die entkoppelten Fehlerdynamiken gewünschte Eigenwerte

aufweisen (Polvorgabe) und die charakteristischen Polynome

$$p_j(s) = s^3 + k_{2q,j}s^2 + k_{1q,j}s + k_{0q,j} \tag{4.18}$$

Hurwitzpolynome sind.

Bemerkung 4.3 (Kompensation von dissipativen und externen Kräften und Momenten). Man beachte, dass sich $\boldsymbol{\tau}$ gemäß (4.2) im Allgemeinen aus dissipativen Kräften und Momenten $\boldsymbol{\tau}_d$, Stelleingängen $\boldsymbol{\tau}_c$ und extern auf den Roboter wirkenden Kräften und Momenten $\boldsymbol{\tau}_{ext}$ zusammensetzt. Sind Schätzungen $\hat{\boldsymbol{\tau}}_d$ bzw. $\hat{\boldsymbol{\tau}}_{ext}$ von $\boldsymbol{\tau}_d$ bzw. $\boldsymbol{\tau}_{ext}$ oder zumindest von Teilen bekannt, dann können diese im Regelgesetz berücksichtigt werden und $\boldsymbol{\tau}_c$ lautet dann

$$\boldsymbol{\tau}_c = \boldsymbol{\tau} - \hat{\boldsymbol{\tau}}_d - \hat{\boldsymbol{\tau}}_{ext} \tag{4.19}$$

mit $\boldsymbol{\tau}$ gemäß (4.6), (4.11) oder (4.17).

4.2 Arbeitsraum

4.2.1 Dynamisches Modell im Arbeitsraum

Das dynamische Modell des Roboters (4.1) wurde als Funktion der Koordinaten im Konfigurationsraum $\mathbf{q}$ und $\dot{\mathbf{q}}$ angeschrieben. Eine Bewegungsregelung im Arbeitsraum findet allerdings in den zugehörigen Koordinaten des Arbeitsraums $\mathbf{x}_e$ gemäß (1.3) und $\dot{\mathbf{x}}_e$ gemäß (2.33) statt. Es ist daher vorteilhaft, das dynamische Modell des Roboters (4.1) in den Koordinaten des Arbeitsraums anzuschreiben.

Der Zusammenhang zwischen den Koordinaten im Konfigurationsraum und den Koordinaten im Arbeitsraum sowie deren Ableitungen sind durch (1.3) und (2.33) gemäß

$$\mathbf{x}_e = \mathbf{f}(\mathbf{q}) \tag{4.20a}$$
$$\dot{\mathbf{x}}_e = \mathbf{J}_A(\mathbf{q})\dot{\mathbf{q}} \tag{4.20b}$$
$$\ddot{\mathbf{x}}_e = \mathbf{J}_A(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{J}}_A(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} \tag{4.20c}$$

gegeben. Aus (4.20c) folgt unter der Annahme der Regularität von $\mathbf{J}_A(\mathbf{q})$, d. h. außerhalb der Singularitäten, die Beziehung

$$\ddot{\mathbf{q}} = \mathbf{J}_A^{-1}(\mathbf{q})\Big(\ddot{\mathbf{x}}_e - \dot{\mathbf{J}}_A(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}\Big)\ . \tag{4.21}$$

Setzt man (4.21) in (4.1) ein, so erhält man

$$\mathbf{D}(\mathbf{q})\mathbf{J}_A^{-1}(\mathbf{q})\ddot{\mathbf{x}}_e + \Big(\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) - \mathbf{D}(\mathbf{q})\mathbf{J}_A^{-1}(\mathbf{q})\dot{\mathbf{J}}_A(\mathbf{q},\dot{\mathbf{q}})\Big)\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} \tag{4.22}$$

und durch Multiplikation der Gleichung mit $\mathbf{J}_A^{-\mathrm{T}}(\mathbf{q}) = \left(\mathbf{J}_A^{-1}\right)^{\mathrm{T}}(\mathbf{q}) = \left(\mathbf{J}_A^{\mathrm{T}}\right)^{-1}(\mathbf{q})$ von links ergibt sich das *mathematische Modell des Roboters im Arbeitsraum* zu

$$\mathbf{\Lambda}(\mathbf{x}_e)\ddot{\mathbf{x}}_e + \boldsymbol{\mu}(\mathbf{x}_e, \dot{\mathbf{x}}_e)\dot{\mathbf{x}}_e + \mathbf{F}_g(\mathbf{x}_e) = \mathbf{F} \tag{4.23}$$

mit

$$\mathbf{\Lambda}(\mathbf{x}_e) = \mathbf{J}_A^{-\mathrm{T}}(\mathbf{q})\mathbf{D}(\mathbf{q})\mathbf{J}_A^{-1}(\mathbf{q}) \tag{4.24a}$$

$$\boldsymbol{\mu}(\mathbf{x}_e, \dot{\mathbf{x}}_e) = \mathbf{J}_A^{-\mathrm{T}}(\mathbf{q})\Big(\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{D}(\mathbf{q})\mathbf{J}_A^{-1}(\mathbf{q})\dot{\mathbf{J}}_A(\mathbf{q}, \dot{\mathbf{q}})\Big)\mathbf{J}_A^{-1}(\mathbf{q}) \tag{4.24b}$$

$$\mathbf{F}_g(\mathbf{x}_e) = \mathbf{J}_A^{-\mathrm{T}}(\mathbf{q})\mathbf{g}(\mathbf{q}) \tag{4.24c}$$

und der generalisierten Kraft $\mathbf{F}$ im Arbeitsraum

$$\mathbf{F} = \mathbf{J}_A^{-\mathrm{T}}(\mathbf{q})\boldsymbol{\tau} \ , \tag{4.25}$$

welche am Endeffektor des Roboters wirkt. Die Beziehung zwischen den Kräften und Momenten im Konfigurationsraum und jenen im Arbeitsraum über die Manipulator Jacobi-Matrix wurde in der Vorlesung *Modellbildung* hergeleitet [4.1]. Man beachte, dass in (4.24) der Zusammenhang $\dot{\mathbf{q}} = \mathbf{J}_A^{-1}(\mathbf{q})\dot{\mathbf{x}}_e$ gilt und $\mathbf{q}$ noch *formal* durch $\mathbf{q} = \mathbf{f}^{-1}(\mathbf{x}_e)$ ersetzt werden muss, siehe (4.20).

Die Multiplikation von (4.22) mit $\mathbf{J}_A^{-\mathrm{T}}(\mathbf{q})$ von links wurde ausgeführt, um beim Robotermodell im Arbeitsraum (4.23) eine ähnliche Struktur zu erhalten wie im Konfigurationsraum (4.1). In (4.23) ist $\mathbf{\Lambda}(\mathbf{x}_e)$ die Massenmatrix, $\boldsymbol{\mu}(\mathbf{x}_e, \dot{\mathbf{x}}_e)$ ist die Coriolismatrix und $\mathbf{F}_g(\mathbf{x}_e)$ ist der Vektor der Gravitationskräfte im Arbeitsraum. Es kann gezeigt werden, dass folgende Eigenschaften auch für das Robotermodell (4.23) erhalten bleiben.

- $\mathbf{\Lambda}(\mathbf{x}_e)$ ist eine positiv definite und damit invertierbare Matrix.
- $\boldsymbol{\mu}(\mathbf{x}_e, \dot{\mathbf{x}}_e)$ ist eine schiefsymmetrische Matrix, d.h. $\boldsymbol{\mu}(\mathbf{x}_e, \dot{\mathbf{x}}_e) + \boldsymbol{\mu}^{\mathrm{T}}(\mathbf{x}_e, \dot{\mathbf{x}}_e) = \mathbf{0}$.

4.2.2 Computed-Torque-Regler

Der *Computed-Torque-Regler im Arbeitsraum* basiert direkt auf dem mathematischen Modell des Roboters in den Koordinaten des Arbeitsraums (4.23) und lautet

$$\mathbf{F} = \mathbf{\Lambda}(\mathbf{x}_e)(\ddot{\mathbf{x}}_{e,d}(t) - \mathbf{K}_{2x}\dot{\mathbf{e}}_x - \mathbf{K}_{1x}\mathbf{e}_x) + \boldsymbol{\mu}(\mathbf{x}_e, \dot{\mathbf{x}}_e)\dot{\mathbf{x}}_e + \mathbf{F}_g(\mathbf{x}_e) \tag{4.26}$$

mit dem Trajektorienfehler im Arbeitsraum $\mathbf{e}_x = \mathbf{x}_e - \mathbf{x}_{e,d}(t)$ bzw. $\dot{\mathbf{e}}_x = \dot{\mathbf{x}}_e - \dot{\mathbf{x}}_{e,d}(t)$ und den Reglermatrizen $\mathbf{K}_{1x}$ und $\mathbf{K}_{2x}$. Die Fehlerdynamik des geschlossenen Regelkreises folgt durch Einsetzen des Reglers (4.26) in die Gleichung des Robotermodells (4.23) zu

$$\underbrace{\ddot{\mathbf{x}}_e - \ddot{\mathbf{x}}_{e,d}}_{\ddot{\mathbf{e}}_x} + \mathbf{K}_{2x}\dot{\mathbf{e}}_x + \mathbf{K}_{1x}\mathbf{e}_x = \mathbf{0} \ . \tag{4.27}$$

Werden weiters die Reglermatrizen als Diagonalmatrizen in der Form $\mathbf{K}_{1x} = \mathrm{diag}(k_{1x,1}, \ldots, k_{1x,m})$ und $\mathbf{K}_{2x} = \mathrm{diag}(k_{2x,1}, \ldots, k_{2x,m})$ mit positiven Einträgen $k_{1x,j}$, $k_{2x,j}$, $j = 1, \ldots, m$, gewählt, so sind die m Fehlerdynamiken des m-dimensionalen Arbeitsraums

$$\ddot{e}_{x,j} + k_{2x,j}\dot{e}_{x,j} + k_{1x,j}e_{x,j} = 0 \ , \quad j = 1, \ldots, m \tag{4.28}$$

entkoppelt. Dies bedeutet, dass die Fehlerdynamik in jeder Koordinate individuell vorgegeben werden kann und dass die Regelfehler unabhängig voneinander mit der jeweiligen Dynamik asymptotisch abklingen.

Während (4.23) und (4.26) das Regelungskonzept mathematisch klar darstellen, ist die Formulierung (4.26) des Computed-Torque-Reglers im Arbeitsraum jedoch ungeeignet für eine Implementierung in der Praxis. Die Gründe dafür sind, dass in den Matrizen (4.24) mehrmals die Inverse der analytischen Manipulator Jacobi-Matrix $\mathbf{J}_A(\mathbf{q})$ auftritt und die Gleichungen (4.20) nicht eindeutig umgekehrt werden können, siehe dazu Abschnitt 2.2. Weiters liegen meist nur die Gelenkswinkel $\mathbf{q}$ als Messung am Roboter vor, nicht aber die Pose des Endeffektors im Arbeitsraum $\mathbf{x}_e$. Daher kann die Messung der Gelenkswinkel $\mathbf{q}$ direkt im Regelgesetz verwendet werden. Setzt man nun (4.24) und (4.25) in (4.26) ein, so erhält man den Computed-Torque-Regler im Arbeitsraum *ausgedrückt im Konfigurationsraum* in der Form

$$\begin{aligned}\boldsymbol{\tau} &= \mathbf{J}_A^{\mathrm{T}}(\mathbf{q})\mathbf{F} \\ &= \mathbf{D}(\mathbf{q})\mathbf{J}_A^{-1}(\mathbf{q})\Big(\ddot{\mathbf{x}}_{e,d}(t) - \mathbf{K}_{2x}\dot{\mathbf{e}}_x - \mathbf{K}_{1x}\mathbf{e}_x - \dot{\mathbf{J}}_A(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}\Big) + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q})\end{aligned} \tag{4.29}$$

mit

$$\mathbf{e}_x = \mathbf{f}(\mathbf{q}) - \mathbf{x}_{e,d}(t)\ ,\ \dot{\mathbf{e}}_x = \mathbf{J}_A(\mathbf{q})\dot{\mathbf{q}} - \dot{\mathbf{x}}_{e,d}(t)\ . \tag{4.30}$$

Man beachte, dass in der Formulierung (4.29) und (4.30) lediglich die Vorwärtskinematik $\mathbf{f}(\mathbf{q})$ und die differentielle Kinematik mit $\mathbf{J}_A(\mathbf{q})$ des Roboters benötigt werden. Auch diese Formulierung führt zur selben Fehlerdynamik (4.27).

4.2.3 Impedanzregelung

Bei der Impedanzregelung im Arbeitsraum wird nicht nur eine gewünschte Roboterbewegung $\mathbf{x}_{e,d}(t)$ vorgegeben, sondern ein gewünschtes mechanisches Verhalten bezüglich der Koordinaten im Arbeitsraum. Ein Beispiel für ein mechanisches Verhalten ist das *Feder-Masse-Dämpfer-System*

$$\mathbf{M}_d\ddot{\mathbf{e}}_x + \mathbf{D}_d\dot{\mathbf{e}}_x + \mathbf{K}_d\mathbf{e}_x = \mathbf{F}_{ext} \tag{4.31}$$

mit $\mathbf{e}_x = \mathbf{x}_e - \mathbf{x}_{e,d}$ und den vorgegebenen positiv definiten Matrizen $\mathbf{M}_d$, $\mathbf{D}_d$ und $\mathbf{K}_d$ für die Masse, Dämpfung bzw. Steifigkeit des Systems [4.2]. Die externe Kraft $\mathbf{F}_{ext}$ in (4.31) wirkt auf das System und lenkt die virtuelle Masse von der (zeitveränderlichen) virtuellen Ruhelage $\mathbf{x}_{e,d}$ aus.

Das Regelgesetz für die Impedanzregelung im Arbeitsraum lautet

$$\mathbf{F} = \boldsymbol{\Lambda}(\mathbf{x}_e)\Big(\ddot{\mathbf{x}}_{e,d}(t) + \mathbf{M}_d^{-1}(\mathbf{F}_{ext} - \mathbf{D}_d\dot{\mathbf{e}}_x - \mathbf{K}_d\mathbf{e}_x)\Big) + \boldsymbol{\mu}(\mathbf{x}_e,\dot{\mathbf{x}}_e)\dot{\mathbf{x}}_e + \mathbf{F}_g(\mathbf{x}_e) \tag{4.32}$$

und ausgedrückt im Konfigurationsraum ist dies

$$\begin{aligned}\boldsymbol{\tau} = {} & \mathbf{D}(\mathbf{q})\mathbf{J}_A^{-1}(\mathbf{q})\Big(\ddot{\mathbf{x}}_{e,d}(t) + \mathbf{M}_d^{-1}(\mathbf{F}_{ext} - \mathbf{D}_d\dot{\mathbf{e}}_x - \mathbf{K}_d\mathbf{e}_x) - \dot{\mathbf{J}}_A(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}\Big) \\ & + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q})\ .\end{aligned} \tag{4.33}$$

Beide Formulierungen führen zum gewünschten Impedanzverhalten und weisen (4.31) als Fehlerdynamik des geschlossenen Regelkreises auf. Dies bedeutet, dass eine externe

Kraft $\mathbf{F}_{ext}$, welche am Endeffektor des Roboters angreift, eine Auslenkung $\mathbf{e}_x$ aus der virtuellen Ruhelage $\mathbf{x}_{e,d}$ gemäß der vorgegebenen Dynamik (4.31) hervorruft. Für die Impedanzregelung ist allerdings die Messung (oder Schätzung) der externen Kräfte und Drehmomente $\mathbf{F}_{ext}$ in allen m Dimensionen des Arbeitsraums notwendig.

4.2.4 Kinematisch redundante Roboter $n > m$

In diesem Abschnitt werden die vorgestellten Regelungskonzepte im Arbeitsraum auf kinematisch redundante Roboter erweitert, d. h. es gilt $n > m$. Damit ist die analytische Manipulator Jacobi-Matrix $\mathbf{J}_A(\mathbf{q})$ nicht mehr quadratisch und die Inverse ist nicht mehr definiert. Analog zu den Abschnitten 2.4.2 und 2.4.3 wird daher die Pseudoinverse $\mathbf{J}_A^\dagger(\mathbf{q})$ aus (2.71) anstelle der Inversen $\mathbf{J}_A^{-1}(\mathbf{q})$ verwendet. Weiters wird ein zusätzlicher Stelleingang $\boldsymbol{\tau}_n$ eingeführt, über welchen die Bewegung des Roboters im Nullraum geregelt wird.

Mit diesen Erweiterungen folgt der Computed-Torque-Regler für kinematisch redundante Roboter (ausgedrückt im Konfigurationsraum) zu, siehe (4.29)

$$\begin{aligned}\boldsymbol{\tau} = \mathbf{D}(\mathbf{q})\mathbf{J}_A^\dagger(\mathbf{q})\Big(\ddot{\mathbf{x}}_{e,d}(t) - \mathbf{K}_{2x}\dot{\mathbf{e}}_x - \mathbf{K}_{1x}\mathbf{e}_x - \dot{\mathbf{J}}_A(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}\Big)\\ + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \boldsymbol{\tau}_n\end{aligned} \tag{4.34}$$

und der Impedanzregler lautet, siehe (4.33)

$$\begin{aligned}\boldsymbol{\tau} = \mathbf{D}(\mathbf{q})\mathbf{J}_A^\dagger(\mathbf{q})\Big(\ddot{\mathbf{x}}_{e,d}(t) + \mathbf{M}_d^{-1}(\mathbf{F}_{ext} - \mathbf{D}_d\dot{\mathbf{e}}_x - \mathbf{K}_d\mathbf{e}_x) - \dot{\mathbf{J}}_A(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}\Big)\\ + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \boldsymbol{\tau}_n\ .\end{aligned} \tag{4.35}$$

Die Beschleunigungen des Nullraumreglers $\ddot{\mathbf{q}}_n$ gemäß (2.82) werden über eine Projektionsmatrix in der Form

$$\boldsymbol{\tau}_n = \Big(\mathbf{E} - \mathbf{J}_A^\dagger(\mathbf{q})\mathbf{J}_A(\mathbf{q})\Big)\ddot{\mathbf{q}}_n \tag{4.36}$$

aufgeschaltet. Wie in den Abschnitten 2.4.2 und 2.4.3 kann mit dem Nullraumregler ein (anwendungsspezifisch gewähltes) sekundäres Kriterium erfüllt werden.

4.3 Literatur

[4.1] W. Kemmetmüller und A. Kugi, *Skriptum zur VU Modellbildung (SS 2020)*, Institut für Automatisierungs- und Regelungstechnik, TU Wien, 2020. Adresse: https://www.acin.tuwien.ac.at/bachelor/modellbildung/.

[4.2] C. Ott, *Cartesian Impedance Control of Redundant and Flexible-Joint Robots*. Springer, Berlin Heidelberg, 2008.

[4.3] K. M. Lynch und F. C. Park, *Modern Robotics: Mechanics, Planning, and Control*. Cambridge University Press, Cambridge, 2017. Adresse: http://hades.mech.northwestern.edu/images/7/7f/MR.pdf.

[4.4] B. Siciliano, L. Sciavicco, L. Villani und G. Oriolo, *Robotics: Modelling, Planning and Control*. Springer, London, 2009.