

Räumliche Kraftmessung und -regelung mit strukturintegrierter Sensorik in Hexapoden

22

Christian Friedrich, Jens Müller und Steffen Ihlenfeldt

Zusammenfassung

Strukturintegrierte Kraftsensorik besitzt zahlreiche Vorteile gegenüber klassischen Ansätzen zur Kraftmessung. Gleichzeitig wirken jedoch prozess- und strukturbedingte Einflüsse auf die Kraftsensoren, die in Abhängigkeit der Sensorposition durch ein Messmodell kompensiert werden müssen. Für dynamische Messungen während der Maschinenbewegung müssen diese Messmodelle zwingend in Echtzeit im Steuerungskern laufen und weiterhin durch ein Verfahren zur Parameteridentifikation schnell und einfach an der betriebsbereiten Maschine aktualisiert werden können. Ausgehend von einer systematischen Einordnung möglicher Messsystemvarianten werden die notwendigen Messmodelle und Varianten ihrer Steuerungsintegration vorgestellt. Den Abschluss bildet die Verwendung der integrierten Sensorik für die Kraftregelung und ihre steuerungstechnische Umsetzung in einer kommerziellen Werkzeugmaschinensteuerung.

22.1 Einleitung

Die zwischen Werkzeug und Werkstück wirkende Prozesskraft ist eine der wesentlichen Größen in der Produktionstechnik, deren Messung bzw. Regelung einen Beitrag zur Steigerung von Qualität und Produktivität, Senkung der Kosten oder gar zur Realisierung völlig neuer Prozesse leisten kann [1–3]. Dabei wächst insbesondere der Bedarf

C. Friedrich · J. Müller · S. Ihlenfeldt (✉)

Professur für Werkzeugmaschinenentwicklung und adaptive Steuerungen, Technische Universität Dresden, Dresden, Deutschland

E-Mail: Steffen.Ihlenfeldt@TU-Dresden.de

an räumlicher Messung in bis zu sechs Freiheitsgraden (FG). Mit der Verwendung autarker 6D-Kraftmessplattformen (KMP) am Endeffektor sind häufig eine Verkleinerung des Arbeitsraums, Einschränkungen bezüglich der Werkzeug- bzw. Werkstückbefestigung am Endeffektor (z. B. von Frässpindeln), Einschränkungen bei der Robustheit der Fertigungseinrichtung (z. B. bzgl. aggressiver Kühlenschmierstoffe oder Späne) und nicht zuletzt hohe Investitionskosten verbunden. Die alternativ häufig eingesetzte Kraftschätzung aus Motorströmen eignet sich aufgrund schwer modellier- und parametrierbarer nichtlinearer oder stochastischer Störeinflüsse im Antriebsstrang meist nur für Anwendungsfälle mit niedrigen Genauigkeitsanforderungen. Ähnliches gilt, wenn auch in geringerem Umfang, für die Drehmomentmessung in den Gelenken bei seriellen Kinematiken, die ferner nicht ohne Weiteres auf die hier betrachteten Parallelkinematiken übertragbar ist. Dennoch besteht erhebliches Potenzial in der Integration einachsiger Kraftsensoren in die Maschinenstruktur und der steuerungsintegrierten Verarbeitung der Messsignale zu Kräften und Momenten an der Wirkstelle. Dies gilt insbesondere für parallele Stabstrukturen – und für die 6D-Messung an Hexapodstrukturen und Hexapoden – bei denen eine nahezu reibungsfreie Messung mittels einachsiger Kraftsensoren möglich wird. Die 6D-Kraftmessung mit Hexapoden tritt jedoch bisher nur als reine Messeinrichtungen in Form von starren, unbewegten Hexapodstrukturen mit rein statischer Kalibrierung in Erscheinung [4–10]. Eine Integration in eine Maschinenstruktur erfolgt ebenso wenig wie die Bewegung der Messhexapoden selbst. Somit bleiben die zugehörigen Messmodelle trivial, da sie keine Verrechnung von Pose oder Dynamik mit den Kraftsensorsignalen erfordern.

Mit der Strukturintegration der Kraftsensorik in die bewegte Maschine ist die echtzeitfähige Berechnung der Messmodelle zur Korrektur interner Einflüsse hingegen zwingend erforderlich. In Abhängigkeit des konkreten Anwendungsfalls folgen daraus Echtzeitanforderungen an die Modellrechnung und weitere Anforderungen an die Steuerung, bspw. eine synchrone Datenerfassung von Antriebs- und Kraftsensordaten. Bis vor nicht allzu langer Zeit waren für neue mess-, steuerungs- und Regelungstechnische Lösungen spezielle Echtzeitsysteme für die Forschung und proprietäre Linux-basierte Steuerungen, wie DSpace, NI Labview, Xenomai oder VxWorks, Mittel der Wahl. Mit der rasanten Entwicklung der Steuerungstechnik im Rahmen der aktuellen Trends Industrie 4.0, Cyber-physische Produktionssysteme (CPPS) oder Industrial Internet of Things (IIOT) haben sich die Möglichkeiten der industriellen Steuerungstechnik in Bezug auf unterstützte Schnittstellen, Offenheit für Nutzeranwendungen, Messdatenerfassung und Rechenleistung jedoch signifikant verbessert, was sich in Produkten, wie bspw. Bosch Open Core Engineering und ctrlX oder Beckhoff TwinCAT 3.1 eXtended Automation Engineering, niederschlägt. Aus diesem Grund ist die Integration und Verarbeitung umfangreicher anwendungsspezifischer Messmodelle in Echtzeit und synchron zur Maschinenbewegung nun bereits für einige kommerzielle Werkzeugmaschinensteuerungen grundsätzlich möglich, auch wenn teils noch erheblicher Forschungsbedarf besteht.

Dieser Beitrag greift diese Fragen auf und befasst sich mit der Umsetzung der strukturintegrierten Kraftmessung am Hexapod-Versuchsträger Felix I mit besonderem Blick

auf die Integration der dazu notwendigen Messmodelle in die Maschinensteuerung. Im ersten Schritt werden, aufbauend auf einer systematischen Einordnung der Möglichkeiten zur Sensorintegration, die wesentlichen Grundlagen zu den Messmodellen für verschiedene Messsystemvarianten abgeleitet. Anschließend werden Steuerungsanforderungen in Abhängigkeit der Anwendungsfälle definiert und Lösungen zur Steuerungsintegration für Kraftmessung und -regelung vorgestellt.

Für weitere Ergebnisse zum Lösungsansatz der strukturintegrierten Kraftmessung [11], insbesondere zur erreichbaren Messgenauigkeit mit einem Hexapod-Endeffektor [12], zur Evaluierung der Steifigkeitsänderung durch Sensorintegration [13, 21] sowie zur Parameteridentifikation durch optimale Messposen bzw. Messtrajektorien [14] und zur Kraftregelung [22], soll auf die Literatur verwiesen werden.

22.2 Systematik und Umsetzung am Versuchsträger

Zwischen den bekannten Lösungen einer Kraftmessplattform am Endeffektor (R) und der Kraftschätzung aus den Motorströmen (M) lassen sich die Messsystemvarianten zur strukturintegrierten Kraftmessung mittels Längskraftmessung in Hexapodstrukturen systematisch einordnen (Abb. 22.1).

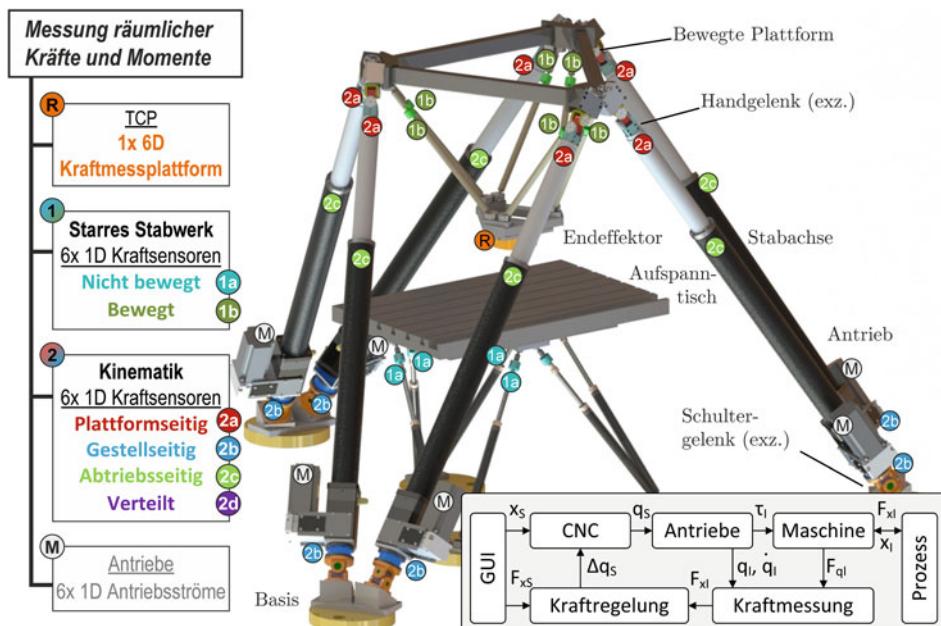


Abb. 22.1 Systematik zur Sensorintegration in Hexapodstrukturen und Hexapoden [11–14, 20]

Als Leichtbaustrukturen finden starre Stabtragwerke seit jeher konstruktive Anwendung, so auch als Aufspanntisch oder als Endeffektor in Maschinen. Durch die Ermöglichung räumlicher Kraftmessung in sechs FG sind sie in der Ausführung als Hexapodstruktur besonders gut für die Sensorintegration geeignet und können als 1a: *unbewegte* (z. B. Aufspanntisch, Spindelhalterung) oder 1b: *bewegte starre* Strukturen (z. B. Endeffektor) eingesetzt werden. Für in die *Kinematik* integrierte Sensoren ergeben sich ebenfalls verschiedene Möglichkeiten: Kraftsensoren sind in den Stabachsen 2a: *plattformseitig*, 2b: *gestellseitig*, 2c: *abtriebsseitig* oder 2d: *verteilt integrierbar*. Schließlich ordnen sich auch die Lösungen aus dem Stand der Technik in diese Systematik ein: Starre Stabwerke, die als eigenständige Sensoren auftreten [4–10], zählen zu Variante 1a. Als existierende Hexapoden mit integrierten Sensoren zählen eine Taumelpresse [15] zu Variante 2a und der Hexapod-Radlader [16, 17] zu Variante 2c. Keine dieser Arbeiten nutzt die Sensoren jedoch für eine Kraft-/Momentenmessung am TCP; vielmehr werden sie für die Antriebsregelung eingesetzt.

Am Versuchsträger Hexapod Felix I sind mit dem Aufspanntisch (1a), der Endeffektorplattform (1b) und der Sensorintegration gestellseitig in den Stabachsen (2b) drei strukturintegrierte Messsystemvarianten zugleich mit den Standardvarianten der Kraftmessplattform (R) und Messung der Motorströme (M) realisiert (Abb. 22.2). Dies ist für Untersuchungen und Vergleich der Möglichkeiten einer Steuerungsintegration verschieden komplexer Messmodelle besonders vorteilhaft.



Abb. 22.2 Umgesetzte Lösungen zur strukturintegrierten Kraftmessung am Versuchsträger Felix I mit 1a: Aufspanntisch (je 5 kN), 1b: Endeffektor (je 5 kN), 2b: Stabachsen mit gestellseitig integrierten Sensoren (je 10 kN) sowie (nicht dargestellt) R: Kraftmessplattform und M: Messung der Motorströme [20]

In jeder Phase des Produktlebenszyklus müssen Fragestellungen aus verschiedenen Sichtweisen und für unterschiedliche Verhaltensbereiche beantwortet werden. Zu diesem Zweck bildet der Digitale Zwilling einer Maschine die Maschine selbst und die auf ihr laufenden Prozesse für die aufgabenbezogene, relevante und hinreichend abstrahierte Beschreibung, Nachbildung, Analyse und Beeinflussung ihrer Bestandteile, Funktionen und Eigenschaften ab.

Während der *Entwurfsphase* besteht insbesondere für starre Strukturen, wie bspw. Aufspanntische oder Endeffektoren, in Grenzen aber auch für Kinematiken die Möglichkeit zur gezielten Beeinflussung charakteristischer Eigenschaften. So sind bspw. die Raumrichtungen mit hoher Empfindlichkeit bzw. hoher Steifigkeit gezielt einstellbar oder es kann isotropes Verhalten für Kräfte und Momente realisiert werden. Da diese Eigenschaften wesentlich durch die Sensoranordnung und damit durch die Geometrie der Stabstruktur bestimmt sind, ist für die Entwurfsphase ein nicht-echtzeitfähiges geometrisches Modell in Form der geometrischen Jacobimatrix ausreichend. Auf dieser Basis kann eine Optimierung der sich gegensätzlich verhaltenden Zielkriterien maximale Struktursteifigkeit und maximale Sensorempfindlichkeit unter Berücksichtigung der konstruktiven Randbedingungen, wie dem verfügbaren Einbauraum, erfolgen [7]. Da lediglich die Anordnung der Sensoren berücksichtigt wird, bleibt das Gesamtsystem hinsichtlich Größe und Messbereich skalierbar.

In der *Gestaltungsphase* werden mit konkreten Maschinenabmessungen und Sensordaten Aussagen über wichtige technische Kennzahlen, wie Messbereich, Überlastbereich, Auflösung oder Steifigkeit, in den kartesischen Raumrichtungen möglich. Modellabgleich und Echtzeitfähigkeit sind hier noch nicht gefordert; mit einem kinetischen Modell können aber bereits Komponentenmassen und -trägheiten berücksichtigt werden. An dieser Stelle kann die Gestaltung durch FE-Modelle zur besseren Einschätzung von Nachgiebigkeiten oder Mehrkörpersimulations-Modelle zur Beurteilung der Dynamik ergänzt werden.

Für die Realisierung praxisrelevanter Anwendungen sind in der *Inbetriebnahme- und Betriebsphase* für den hier betrachteten Anwendungsfall schließlich zwingend echtzeitfähige, steuerungsintegrierte Modelle einzusetzen. Diese umfassen die kinematischen, differenziell kinematischen und kinetischen Transformationen der Maschine sowie darauf aufbauend die Messmodelle mit der Sensoranordnung sowie Sensor- und Korrekturparametern. Unter Echtzeitfähigkeit wird in diesem Zusammenhang die Berechnung im Lagereglertakt von bspw. 1 ms verstanden. Für die Entwicklung der strukturintegrierten Kraftmessung wird der Newton-Euler Formalismus eingesetzt, da bei diesem Verfahren direkt Schnittkräfte und -momente an allen für die Sensorintegration in Frage kommenden Positionen vorliegen. Weil für Parallelkinematiken aufgrund der Schließbedingung keine rekursive Lösung möglich ist, ergibt sich ein Gleichungssystem mit 180 Gleichungen, welches im Steuerungstakt durch einen Gauss-Algorithmus gelöst wird [5, 8]. Schließlich können sich einige Parameter des Modells, wie Massen und Trägheitsmomente, während des Betriebs ändern, z. B. durch Werkzeug- oder Werkstückwechsel, und einen erneuteten Parameterabgleich erfordern. Für diesen Zweck besteht die Möglichkeit zur einfachen

und schnellen Parameteridentifikation an der betriebsbereiten Maschine auf Basis optimaler quasi-statischer oder dynamischer Anregungstrajektorien [9]. Voraussetzung dafür ist die Umstellung des Messmodells in eine Form, bei der die Parameter für die Identifikation zugänglich sind.

22.3 Messmodelle

Während für die Transformation der gemessenen Sensorkräfte in kartesische Kräfte und Momente an der Wirkstelle für reine Messhexapoden der Variante 1a eine statische Kalibriermatrix genügt, erfordern die übrigen Messsystemvarianten aufwändigere Messmodelle. Abhängig von der konkreten Sensorplatzierung sind dabei struktur- und prozessbedingte Einflüsse zu berücksichtigen, die eine Modellierung der *Sensoranordnung*, der *Starrkörperdynamik* und *weiterer Einflüsse* umfassen. Gemäß der Systematik ergeben sich als Parameter und Zustandsgrößen zunächst

- für alle Varianten: die (für starre Stabwerke konstante) Jacobimatrix zur Abbildung der Sensorpositionen und -orientierungen (36 Parameter) sowie die Nullpunktoffsets der Sensoren (6 Parameter),
- für bewegte starre Stabwerke (Variante 1b): zusätzlich Masse, Schwerpunktlage und Trägheitsmoment der als Starrkörper modellierten freien Seite des Stabwerks (10 Parameter) sowie in Echtzeit die dazu notwendigen Zustandsgrößen Pose, Geschwindigkeit und Beschleunigung des Stabwerks (18 Zustandsgrößen),
- für Hexapodkinematiken (Varianten 2a-d): zusätzlich Massen, Schwerpunktlagen und Trägheitsmomente aller weiteren 24 bewegten Körper (240 Parameter) sowie deren Posen, Geschwindigkeiten und Beschleunigungen in Echtzeit (450 Zustandsgrößen) sowie schließlich
- bei Verwendung der Motorströme (Variante M): zusätzliche Reibparameter der Antriebsstränge sowie elektrische und dynamische Kenngrößen der Motoren.

Hinzu kommen weitere Einflüsse, wie bspw. Querkräfte und -momente auf die Sensoren, elastische oder thermische Effekte. Insbesondere für die Sensorintegration in Maschinenkinematiken kommen so oft mehrere Hundert teils aufwändige und in Echtzeit zu lösende Gleichungen zusammen, die auf der Steuerung implementiert werden müssen.

Nicht bewegte starre Stabwerke (1a)

Das Messmodell zur Transformation der gemessenen Sensorkräfte f_q in kartesische Kräfte und Momente an der Wirkstelle f_{Ext} besteht für nicht bewegte starre Stabwerke

zunächst lediglich aus der geometrischen Transformation der Kräfte in Form der geometrischen Jacobimatrix des Stabwerks J_S und der Einführung von sechs Nullpunktoffsets f_{q0} für eine einfache Tarierung:

$${}^A f_{\text{Ext}} = {}^A X_S^F J_S^{-T} (f_q - f_{q0}) = {}^A X_S^F \begin{pmatrix} n_1 & \cdots & n_6 \\ h_1 \times n_1 & \cdots & h_6 \times n_6 \end{pmatrix} (f_q - f_{q0}). \quad (22.1)$$

Für die Definition der Jacobimatrix sind dabei n_i die Richtungsvektoren der Kraftsensoren und h_i ihre Verbindung zum Koordinatenursprung. Beide Größen lassen sich direkt aus den Konstruktionsdaten gewinnen, mit den gestellseitigen bzw. plattformseitigen Anlenkpunkten $\{S_i\}$ und $\{H_i\}$, Abb. 22.3. Die Überführung der Kräfte f und Momente m vom Koordinatensystem des Stabwerks $\{S\}$ in das Aufgabenkoordinaten- system $\{A\}$ gelingt schließlich mit Hilfe der räumlichen Krafttransformation ${}^A X_S^F$ über die Transformationsmatrix ${}^A R_S$ und den Ortsvektor ${}^A o_S$ wie folgt

$${}^A f = {}^A X_S^F {}^S f = \begin{pmatrix} {}^A f \\ {}^A m \end{pmatrix} = \begin{pmatrix} {}^A R_S & 0 \\ {}^S({}^A o_S) {}^A R_S & {}^A R_S \end{pmatrix} \cdot \begin{pmatrix} {}^S f \\ {}^S m \end{pmatrix}. \quad (22.2)$$

Sämtliche nicht modellierten Einflüsse, bspw. Gravitationskräfte aufgrund der Eigenmasse, werden für diese Stabwerke durch die Parameter f_{q0} abgebildet. Im Falle einer Lage- oder Eigenschaftsveränderung ist lediglich ein Tarieren erforderlich.

Bewegte Starre Stabwerke (1b)

Bei bewegten starren Stabwerken wirken externe Einflüsse, wie Gravitation und Dynamik, auf die Kraftsensoren, welche entsprechend im Messmodell berücksichtigt werden

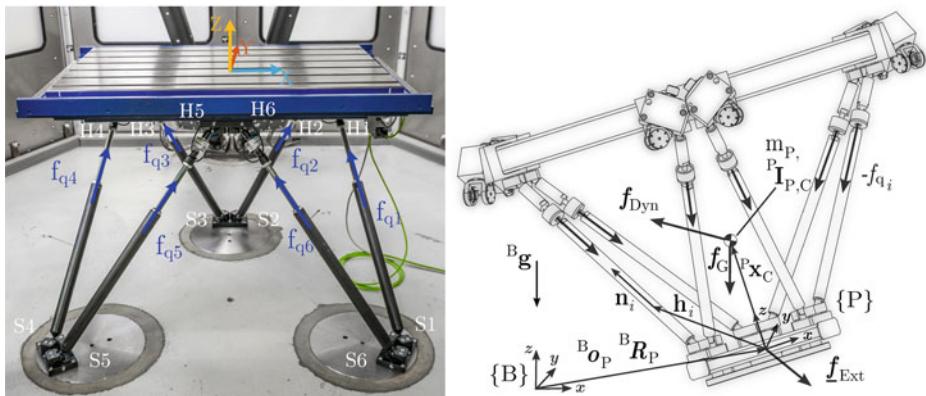


Abb. 22.3 Starre Stabwerke mit Kraftsensoren und Modellgrößen; links: Aufspanntisch (1a), rechts: Endeffektorplattform (1b) [12, 20]

müssen. Auf Basis der Krafttransformation mit Hilfe der Jacobimatrix ${}^P\mathbf{J}_S$ und unter Berücksichtigung eines einfachen Starrkörpermodells für den Plattformteil unterhalb der Sensoren ergibt sich das Messmodell mit Position \mathbf{x} , Geschwindigkeit \mathbf{v} , Winkelgeschwindigkeit $\boldsymbol{\omega}$, Erdbeschleunigung \mathbf{g} und Kreuzproduktmatrix \mathbf{S} zu

$$\begin{aligned} {}^P\mathbf{f}_{\text{Ext}}(\underline{\mathbf{x}}, \underline{\mathbf{v}}, \dot{\underline{\mathbf{v}}}) &= \underbrace{\begin{bmatrix} m_p E & m_p \mathbf{S}^T(r) \\ m_p \mathbf{S}(r) & {}^P\mathbf{I}_P \end{bmatrix}}_{M_p} \begin{bmatrix} {}^P\dot{\mathbf{v}} \\ {}^P\dot{\boldsymbol{\omega}} \end{bmatrix} + \underbrace{\begin{bmatrix} \mathbf{0} & m_p \mathbf{S}({}^P\boldsymbol{\omega}) \mathbf{S}^T(r) \\ \mathbf{0} & S({}^P\boldsymbol{\omega}) {}^P\mathbf{I}_P \end{bmatrix}}_{C_p} \begin{bmatrix} {}^P\mathbf{v} \\ {}^P\boldsymbol{\omega} \end{bmatrix} \\ &\quad + \underbrace{\begin{bmatrix} -m_p {}^P\mathbf{g} \\ -m_p \mathbf{S}(r) {}^P\mathbf{g} \end{bmatrix}}_{g_p} - {}^P\mathbf{J}_S^{-T} \underbrace{(\mathbf{f}_q - \mathbf{f}_{q0} - \mathbf{f}_K)}_{\dot{\mathbf{f}}_q} \\ &= M_p {}^P\dot{\mathbf{v}} + C_p {}^P\mathbf{v} + g_p - {}^P\mathbf{J}_S^{-T} \mathbf{f}_q^*, \end{aligned} \quad (22.3)$$

wobei für eine ausführliche Herleitung auf die Literatur verwiesen wird [12]. Die Starrkörperparameter Masse m_p , Schwerpunktlage \mathbf{r} und Trägheitsmoment \mathbf{I}_P können durch eine Prozedur zur Parameteridentifikation an der betriebsbereiten Maschine ermittelt werden [14]. Dies ist bspw. nach Werkzeug-/Werkstückwechsel erforderlich, wenn sich die Plattformeigenschaften verändern und das Modell aktualisiert werden muss. Für eine reine Tarierung, bspw. aufgrund von Sensordrifts, genügt es hingegen, die Nullpunktoffsets \mathbf{f}_{q0} neu zu berechnen.

Maschinenkinematik (2a-2d)

Das Messmodell für eine Sensorintegration in die Maschinenkinematik erfordert zunächst die vollständige Lösung der kinematischen, differenziell kinematischen und kinetischen Transformationen der Maschine. Dabei ist eine Lösung mit unterschiedlichen Modellierungsansätzen möglich, die sich auch hinsichtlich der benötigten Rechenoperationen und damit in der Rechenzeit unterscheiden [18, 19]. Für die strukturintegrierte Kraftmessung bietet sich eine Lösung des kinetischen Problems nach dem Newton-Euler-Verfahren an, da auf diese Weise bereits die Schnittkräfte an den Sensorpositionen bekannt sind. Mit diesem Ansatz ergibt sich für den Versuchsträger Hexapod Felix I mit exzentrischen Gelenken ein Gleichungssystem mit 180 Gleichungen und 180 Unbekannten, das im Steuerungstakt zu lösen ist. Für Details zur Aufstellung wird aufgrund des Umfangs auch hier auf die Literatur verwiesen [11].

Ist die Maschinendynamik berechnet, kann damit in Abhängigkeit der Sensorposition eine virtuelle Sensorkraft $\hat{\mathbf{f}}_q$ bestimmt werden

$$\hat{\mathbf{f}}_{q_i}(\mathbf{x}, \mathbf{v}, \dot{\mathbf{v}}) = \begin{cases} {}^B\mathbf{z}_{HS_i}^T {}^B\mathbf{f}_{HY_i} & \text{Variante 2a} \\ {}^B\mathbf{z}_{SS_i}^T {}^B\mathbf{f}_{SY_i} & \text{Variante 2b,} \\ {}^B\mathbf{z}_{SS_i}^T {}^B\mathbf{f}_{Mi} & \text{Variante 2c} \end{cases} \quad (22.4)$$

um welche die gemessenen Kraftsignale zu korrigieren sind. Für Variante 2d mit verteilten Sensoren sind die Einträge mit Gl. (22.4) für jede Achse i individuell einzusammeln. Die verbleibenden Differenzen entsprechen den gesuchten externen Kräften und Momenten und sind mit der nun poseabhängigen Jacobimatrix der Maschine $\mathbf{J}_{\text{Hex}}(\mathbf{x})$ in den Arbeitsraum zu transformieren:

$${}^{\text{B}} \mathbf{f}_{\text{Ext}}(\mathbf{x}, \mathbf{v}, \dot{\mathbf{v}}) = \mathbf{J}_{\text{Hex}}^{-T}(\mathbf{x}) \left(\mathbf{f}_{\text{q}} - \hat{\mathbf{f}}_{\text{q}}(\mathbf{x}, \mathbf{v}, \dot{\mathbf{v}}) - \mathbf{f}_{\text{q0}} - \mathbf{f}_{\text{K}} \right). \quad (22.5)$$

22.4 Steuerungsintegration der Kraftmessung

Nur für einige wenige Anwendungen genügt es, die Kraftmessdaten in Nicht-Echtzeit auszuwerten. Für die meisten praxisrelevanten Anwendungen, wie Prozesskraftüberwachung und -aufzeichnung, kraftbasiertes Teachen oder Prozesskraftregelung, ist die Berechnung des Messmodells hingegen zwingend in Echtzeit auszuführen. Weiterhin zählt eine synchrone Erfassung von Antriebs- und Kraftsensordaten zu den Voraussetzungen für die erfolgreiche Umsetzung des strukturintegrierten Ansatzes.

Unabhängig von der Implementierung echtzeitfähiger Algorithmen selbst besteht die Herausforderung in ihrer Integration in die Maschinensteuerung. Diese kann in Abhängigkeit der Steuerungszugänglichkeit direkt, über interne oder über externe Schnittstellen erfolgen (Tab. 22.1). Die größten Freiheiten bei Implementierung und Funktionsumfang bestehen bei Vollzugriff auf die Steuerung. Auch ist dieser für die Kraftregelung Voraussetzung, sofern keine steuerungsseitige Schnittstelle für die Kraftregelung zur Verfügung steht, was auf die meisten Steuerungen für Werkzeugmaschinen zutrifft. Während bei eingeschränktem Zugriff mit Hilfe von Echtzeitbussen die volle Kraftmessfunktionalität realisierbar bleibt, ist bei Zugriff nur über den User-Space eine Messung bei dynamischer Bewegung oder eine Prozesskraftmessung aufgrund unzureichender Taktrate und Synchronizität zu den Bewegungsdaten nicht möglich. Die größte Herausforderung stellen Steuerungen ohne Zugriffsmöglichkeit dar; hier ist nur ein Abgriff an den Hardwareschnittstellen mit entsprechend fraglicher technischer und rechtlicher Realisierbarkeit denkbar.

Für die praktische Umsetzung der strukturintegrierten Kraftmessung durch einen Maschinenhersteller sind die Steuerungsintegration als SPS/C++-Modul sowie die Realisierung als Feldbusteilnehmer (z. B. EtherCAT-Client) am attraktivsten. Nachfolgend soll die Umsetzung mit Vollzugriff auf der kommerziellen Echtzeitsteuerung Beckhoff TwinCAT 3.1 vorgestellt werden. Diese stellt eine Echtzeiterweiterung für das Betriebssystem Microsoft Windows dar und ist durch die Unterstützung der meisten Feldbusse sowie durch ein offenes Steuerungskonzept charakterisiert. Dabei werden sog. TcCOM-Module mit dem Steuerungsprogramm über standardisierte Schnittstellen, Datenbereiche und Zustände verwaltet, in einer oder mehreren Echtzeittasks zur Ausführung gebracht und über ein Prozessabbild mit der Hardware verbunden. Diese Vereinheitlichung erlaubt

Tab. 22.1 Auswahl an Möglichkeiten zur Anbindung der Funktionalität der strukturintegrierten Kraftmessung an die Maschinensteuerung in Abhängigkeit ihrer Zugänglichkeit; RT: Echtzeit, SC: Synchronizität, R: Lesezugriff, W: Schreibzugriff [20]

	Beschreibung	RT	SC	R	W
Vollzugriff	Realisierung als SPS- oder C++ -Modul innerhalb der Steuerung, Anschluss der Kraftsensoren an die Steuerung und direkter Zugriff auf die Antriebsdaten	✓	✓	✓	✓
Eingeschränkter Zugriff	Realisierung als Feldbus-Teilnehmer, z. B. als EtherCAT-Client, mit Direktanschluss der Kraftsensoren und Verbindung zur Steuerung via Echtzeitbus (z. B. EtherCAT)	✓	✓	✓	✗
	Realisierung als autarkes Gerät mit Direktanschluss der Kraftsensoren und Abgriff der Antriebsdaten über den User-Space der Steuerung, bspw. mittels OPC/UA, Ethernet oder ADS	✗	✗	✓	✗
Kein Zugriff	Realisierung als autarkes Gerät mit Direktanschluss der Kraftsensoren und Abgriff der Antriebsdaten direkt an den Encodern via Y-Kabel	✓	✓	✓	✗
	Realisierung als autarkes Gerät mit Direktanschluss der Kraftsensoren und Kopie der Antriebsdaten vom Antriebsbus der Steuerung (Realisierbarkeit fraglich)	✓	✓	✓	✗

die Erstellung der TcCOM-Module durch verschiedene Compiler, sodass bislang verschiedene Steuerungsarchitekturen, wie SPS, MC, CNC oder C++, nun in beliebiger Anzahl und Kombination als Softwaremodule in den Echtzeitkern eingebunden werden können.

Im Detail verfügt der Versuchsträger Felix I über eine CNC für die Bewegungssteuerung, drei SPSEN zur Logiksteuerung, z. B. für das Maschinenbedienfeld, sowie drei C++-Module für kinematische Transformationen, Kraftmessung und Kraftregelung. Die synchrone Datenerfassung wird über die Feldbusse CAN-Bus, Profibus sowie EtherCAT und Sercos II realisiert, Abb. 22.4.

Auch bei erfolgreicher Integration der Modelle in die Steuerung sind für eine praxisrelevante Anwendbarkeit der Kraftmessung noch zahlreiche weitere Funktionen zu implementieren, wie bspw. Datenvorverarbeitung (Skalierung, Filterung), Datenaufzeichnung oder M-Funktionen. Eine Übersicht anhand verschiedener Anwendungsfälle gibt Tab. 22.2.

Konkreter werden die Modelle zur kinematischen, differenziell kinematischen und kinetischen Transformation für Hexapoden mit exzentrischen Gelenken und darauf

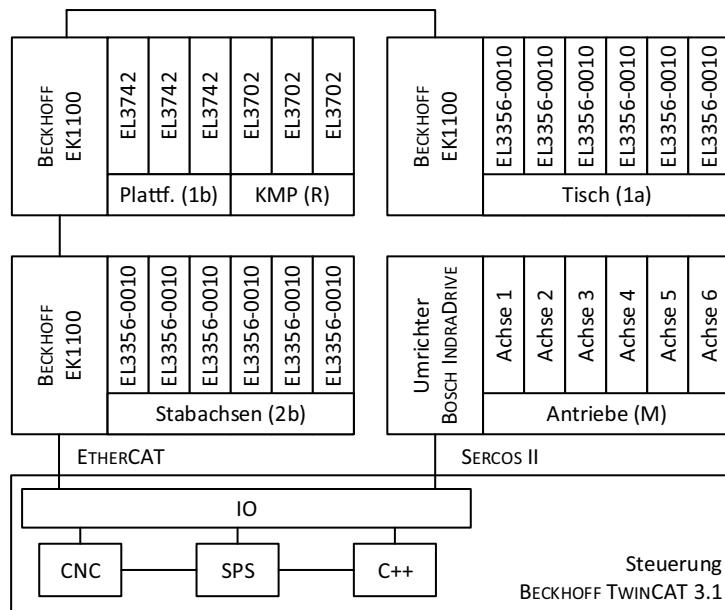


Abb. 22.4 Bustopologie zur echtzeitfähigen synchronen Datenerfassung von Kraftsensor- und Antriebsdaten (Ausschnitt aus der Bustopologie am Versuchsträger Felix I) [20]

Tab. 22.2 Anwendungsfälle der Kraftmessung und Anforderungen an die Steuerungsintegration; EZA: Echtzeitanforderungen, IPO: Interpolator, FFT: Fast Fourier Transformation [20]

Beschreibung	Funktionen	EZA
Anzeige von Kräften und Momenten am TCP auf der Bedienoberfläche der Steuerung	<ul style="list-style-type: none"> • Kin. Transformation • Messmodelle • Bedienoberfläche 	Niedrig (100 ms)
Aufzeichnung von Kräften und Momenten an definierten Posen im Stillstand aus dem NC-Programm und Abspeicherung in einer Datei	<ul style="list-style-type: none"> • G-Code M-Fktn • Datenakquisition • Dateizugriff 	Moderat SPS-Takt (2–20 ms)
Kontinuierliche Kraft-/Momentenüberwachung während eines Prozesses	<ul style="list-style-type: none"> • Limitdefinition • Fehlerreaktion 	Hoch IPO-Takt (1–4 ms)
Start/Stopp kontinuierlicher Aufzeichnung in ausgewählten Prozessabschnitten aus dem NC-Programm und Abspeicherung in einer Datei	<ul style="list-style-type: none"> • G-Code M-Fktn • Datenstreaming 	Hoch IPO-Takt (1–4 ms)
Frequenzanalysen zur Abschätzung dynamischen Übertragungsverhaltens	<ul style="list-style-type: none"> • Blockweises Datenstreaming • FFT 	Sehr hoch (<250 µs)

aufbauend die Messmodelle zur strukturintegrierten Kraftmessung zunächst als unabhängige C++-Bibliothek realisiert. Eine Eigenentwicklung ist dabei sowohl aufgrund der Einschränkungen durch die Echtzeitumgebung, wie fehlende bzw. eingeschränkte dynamische Speicherallokation, fehlende Ausnahmebehandlung oder fehlende Verfügbarkeit von Standard-Bibliotheken, als auch durch kinematische Besonderheiten (exzentrische Gelenke) notwendig. Insbesondere aber erlaubt der Einsatz entsprechender Software-/Code-Wrapper und Compilereinstellungen die Verwendung desselben getesteten Codes in mehreren Versuchsträgern, Simulationsumgebungen (z. B. Matlab, MathCAD) und in der Bedienoberfläche, was die Funktionssicherheit beträchtlich erhöht.

Abb. 22.5 zeigt die Einordnung der Funktionalität in den Steuerungskern: Die Ansteuerung der Maschine erfolgt zunächst klassisch, d. h. über eine Bedienoberfläche wird vom Bediener ein Bewegungsprogramm an die numerische Steuerung (CNC) übergeben, welches mittels der Antriebsregler zur Ausführung gebracht wird. Dabei kann seitens der CNC sowohl im Arbeitsraum (kinematische Transformation im NC-Kern über C++-Modul) als auch im Gelenkkraum (kinematische Transformation in der Bedienoberfläche) operiert werden. Das Messmodell läuft in einer eigenen C++-Task und empfängt synchron Position und Geschwindigkeit der Antriebe über Sercos II und Kraftmesswerte über EtherCAT. Sofern vom Eingang unterstützt, kann dabei mittels Oversampling die Abtastrate der Kraftsensoren auf bis zu 100 kHz gesteigert werden, was hier für eine Mittelwertbildung genutzt wird. Anschließend werden die Eingangsdaten vorverarbeitet, die Messmodelle berechnet und die Ergebnisse so bereitgestellt, dass sie auf der Bedienoberfläche angezeigt und über einen Datenlogger aufgezeichnet werden können. Die Einbindung eigener M-Befehle im NC-Kern erlaubt es, die Kraftsensoren aus dem G-Code heraus zu tarieren und quasi-statische bzw. kontinuierliche Messungen zu starten.

Im Ergebnis zeigt Abb. 22.6 beispielhaft anhand der Variante 1b (Endeffektor) die Korrektur der durch die Maschinenbewegung in sechs Freiheitsgraden wirkenden Kräfte auf die Sensoren mit dem steuerungsintegrierten Messmodell und die dadurch erreichte genaue Messung während der Bewegung. Für weitere Details zur Stabwerksauslegung, Validierung durch quasi-statische, dynamische und Prozesskraftmessungen sowie eine Einordnung im Vergleich zu einer Referenz-Kraftmessplattform wird auf die Literatur verwiesen [12].

22.5 Kraftregelung

Neben der Prozesskraftmessung und -überwachung besteht das größte Potenzial der strukturintegrierten Kraftmessung in ihrer Anwendung zur Prozesskraftregelung. Anwendungsfälle können dabei das kraftgeführte Teachen, die hybride Kraft-/Positionsregelung mit asynchroner Sollkraft sowie die hybride/parallele Kraft-/Positionsregelung mit bewegungssynchroner Sollwertvorgabe sein (vgl. Tab. 22.3). Dabei erfordern die Anwendungsfälle verschiedene Teifunktionen, wie bspw. selektierbare Koordinaten, Sollwertvorgabe

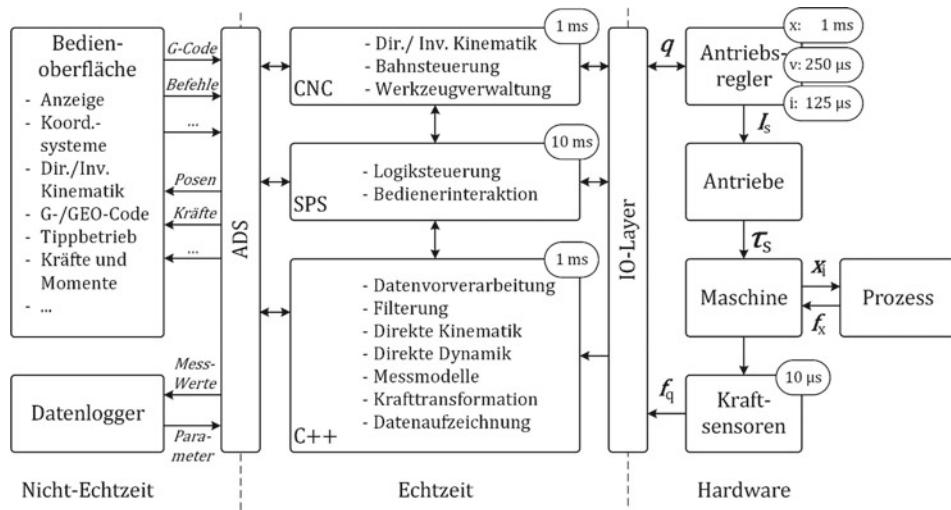


Abb. 22.5 Einbindung von Messmodellen und Kraftmessfunktionalität in den Steuerungskern [20]

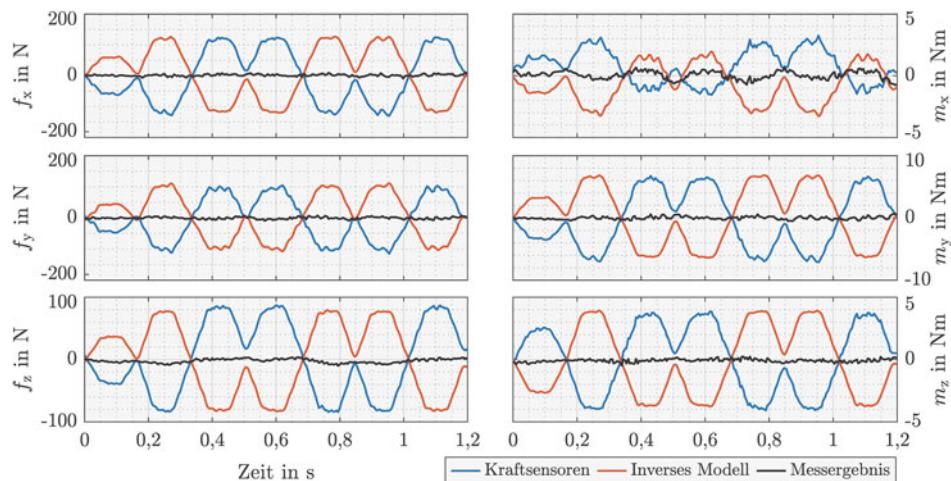


Abb. 22.6 Kompensation der durch die dynamische Maschinenbewegung in 6 Freiheitsgraden entstehenden Kräfte auf die Sensoren mittels eines steuerungsintegrierten Messmodells für strukturintegrierte Kraftsensoren im Endeffektor (Variante 1b) [12, 20]

Tab. 22.3 Anwendungsfälle, Teilfunktionen und Echtzeitanforderungen der Kraftregelung [20]

Beschreibung	Funktionen	EZA
Kraftgeführtes Teachen	<ul style="list-style-type: none"> • Direkte Kraftregelung • Sicherheitsfunktionen • Selektierbare Koordinaten 	Moderat SPS-Takt (2–20 ms)
Hybride Kraft-/Positionsregelung mit asynchroner Sollkraft	<ul style="list-style-type: none"> • Hybride Kraftregelung • Aufgabenkoordinatensystem • Kontaktsuche und Offsetverwaltung • Einfache Kraftsollwertvorgabe 	Moderat SPS-Takt (2–20 ms)
Hybride und parallele Kraft-/Positionsregelung mit bewegungssynchroner Sollkraftvorgabe	<ul style="list-style-type: none"> • Hybrid/parallele Kraftregelung • Synchrone Sollwertvorgabe • Zustandssteuerung via G-Code • G-Code Parameter 	Hoch IPO-Takt (1–4 ms)

und Stellwertaufschaltung, und stellen unterschiedliche Anforderungen an die Echtzeitfähigkeit und die Steuerungsintegration. Während das Teachen aufgrund des fehlenden Sollwertes, Prozesses und Umgebungskontakts, außer zum Bediener, relativ geringe Echtzeit- und Funktionsanforderungen stellt, sind für die hybride/parallele Kraft-/Positionsregelung mit bewegungssynchroner Sollwertvorgabe Echtzeitverarbeitung im Interpolator-Takt und die G-Code-Integration von Sollwertvorgabe und Ansteuerung erforderlich.

Vor dem Hintergrund der Realisierbarkeit mit einer kommerziellen Werkzeugmaschinensteuerung werden die vorhandenen dezentralen Einzelgelenkregler der Maschine beibehalten. Die Umsetzung erfolgt aus diesem Grund als direkter Kraftregelkreis in Aufgabenkoordinaten, der um den inneren Positionsregelkreis in Gelenkraumkoordinaten geschlossen wird. Es ist nicht sinnvoll, immer alle Koordinaten kraftzuregeln, obwohl dies mit der parallelen Kraftregelung oder der Impedanzregelung möglich wäre, sondern diese anhand der Aufgabe in Kraft- und Bewegungsfreiheitsgrade aufzuteilen. Dies leistet die hybride Kraft-/Positionsregelung mit Hilfe der diagonal besetzten Selektionsmatrix S . Für den vorliegenden Fall befindet sich die Maschine durch die Antriebe immer in Positionsregelung. Bei Bedarf wird der äußere Kraftregler für eine oder mehrere Koordinatenrichtungen überlagert. Es handelt sich deshalb bei der umgesetzten Variante um eine Kombination aus hybrider und paralleler Kraftregelung. Abb. 22.7 zeigt den Regelkreis in der Übersicht.

Für die Umsetzung der positionsbasierten Kraftregelung wird zunächst die Kraftregelungsaufgabe im Aufgabenkoordinatensystem $\{A\}$ mit Hilfe der natürlichen und künstlichen Randbedingungen definiert. Anschließend werden die kraftgeregelten Koordinaten mit der Selektionsmatrix S ausgewählt, die Sollwerte f_S vorgegeben und die Regelabweichung $\Delta^A f$ mittels den aus der Kraftmessung bekannten Istwerten f_I berechnet

$$\Delta^A f = S(^A f_S - ^A f_I). \quad (22.6)$$

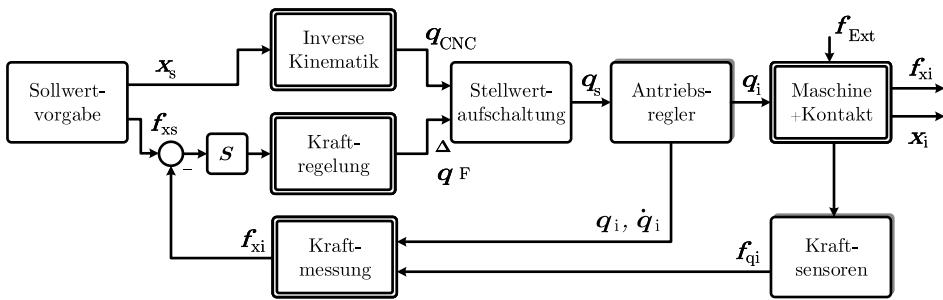


Abb. 22.7 Prinzip der hybriden/parallelen Kraftregelung mit strukturintegrierter Kraftmessung [20]

Zur Generierung der gesuchten Positionoffsets Δq_F aus der Kraftregelabweichung $\Delta^A f$ für die Aufschaltung der kraftbasierten Stellwerte auf die Antriebe bestehen verschiedene Möglichkeiten: Die Transformationen mit Hilfe der transponierten Jacobimatrix $J_{\text{Hex},G}^T$ und den Transformationsmatrizen X

$$\begin{aligned}\Delta^B f &= {}^B X_P^F {}^P X_A^F \Delta^A f \\ \Delta \tau_q &= J_{\text{Hex},G}^T(x_{\text{CNC}}) \Delta^B f \\ \Delta q_F &= K_{Fq} \Delta \tau_q\end{aligned}\quad (22.7)$$

oder der inversen Jacobimatrix $J_{\text{Hex},A}^{-1}$

$$\begin{aligned}\Delta^A x_F &= K_{Fx} \Delta^A f \\ \Delta^B x_F &= {}^B X_P^P X_A^A \Delta^A x_F \\ \Delta q_F &= J_{\text{Hex},A}^{-1}(x_{\text{CNC}}) \Delta^B x_F\end{aligned}\quad (22.8)$$

benötigen zwar nur geringe Rechenleistung, sind aber auch nur in der Nähe der CNC-Pose gültig. Abgesehen vom höheren Rechenaufwand erweist sich die Verwendung der vollständigen direkten (DK) und inversen kinematischen Transformation (IK) als besonders vorteilhaft, da die Regelung direkt in Aufgabenkoordinaten erfolgt, der Parallelbetrieb zur CNC möglich ist und die Stellwerte auch bei größerem Abstand zur CNC-Pose keine Ungenauigkeiten aufweisen. Im Detail wird durch ein Regelgesetz $K_{F,x}$ ein kartesischer Kraftregelungsoffset Δx_F bestimmt, der nach Transformation in das Basiskoordinatensystem $\{B\}$ auf die aktuelle CNC-Pose zur neuen Zielpose x_F addiert wird. Nach Inverstransformation der Zielpose können die Achssollwerte direkt aufgeschalten oder alternativ vorher zu Offsets Δq_F konvertiert werden:

$$\begin{aligned}
 \Delta^A \boldsymbol{x}_F &= \boldsymbol{K}_{Fx} \Delta^A \boldsymbol{f} \\
 \Delta^B \boldsymbol{x}_F &= {}^B \boldsymbol{X}_P {}^P \boldsymbol{X}_A \Delta^A \boldsymbol{x}_F \\
 {}^B \boldsymbol{x}_F &= D\boldsymbol{K}(\boldsymbol{q}_{\text{CNC}}) + \Delta^B \boldsymbol{x}_F \\
 \Delta \boldsymbol{q}_F &= I\boldsymbol{K}({}^B \boldsymbol{x}_F) - \boldsymbol{q}_{\text{CNC}}
 \end{aligned} \tag{22.9}$$

Ein direktes Aufschalten der kartesischen Offsets $\Delta \boldsymbol{x}_F$ ist durch die meist fehlenden Schnittstellen der CNC i. d. R. nicht möglich. Als Regelgesetz kann für \boldsymbol{K}_{Fx} bspw. ein PI-Regler oder ein Admittanz-Regler zum Einsatz kommen. Die Möglichkeiten zur Sollwertvorgabe und Stellwertaufschaltung werden im nächsten Abschnitt diskutiert.

22.6 Steuerungsintegration der Kraftregelung

Im Gegensatz zur Kraftmessung ist für die Kraftregelung immer ein schreibender Zugriff auf die Steuerung notwendig. Dabei unterscheiden sich die Anforderungen an die Steuerungsintegration für die in Tab. 22.3 aufgezeigten Anwendungsfälle. Zur Realisierung der Kraft-/Positionsregelung mit bewegungssynchroner Sollkraftvorgabe erfolgt die G-Code-Integration der Sollwertvorgabe für die Kräfte analog zur Vorgabe der Positionssollwerte, sodass diese in einem NC-Satz gemeinsam programmiert werden können. Dazu werden im CNC-Kern zusätzlich zu den realen Positionierachsen {X, Y, Z, U, V, W} virtuelle Kraft-Achsen {XF, YF, ZF, UF, VF, WF} angelegt, die durch die Platzierung in einem CNC-Kanal mit den Positionierachsen die synchrone Interpolation der Kraft- und Wegsollwerte möglich machen.

Anschließend wird ein C++-Modul zwischen CNC und Antrieben implementiert, welches die Kraftregelung, die Zustandssteuerung (kein Kontakt, möglicher Kontakt und Kontakt) und die Aufschaltung der Kraftregelungsoffsets auf die Antriebssollwerte realisiert (Abb. 22.8). Weitere Antriebsgrößen, wie Statuswort, Steuerwort oder Istwerte, werden für die realen Achsen ohne Veränderung durchgeleitet. Für die virtuellen Kraft-Achsen werden die Kraftistwerte aus dem Modul Kraftmessung an die CNC weitergegeben und Antriebskenngrößen, wie die Sercos-Phase, simuliert.

Als Voraussetzung erfolgt vorab die Integration der kinematischen Vorwärts- und Rückwärtstransformation der Hexapod-Parallelkinematik zur Umrechnung zwischen Antriebs- und Arbeitsraumkoordinaten in den CNC-Kern. Diese wird entsprechend der Transformationsschnittstelle der verwendeten Echtzeitsteuerung Beckhoff TwinCAT 3.1 mit Hilfe der Funktionen *Forward* und *Backward* im Kern implementiert.

Bei der Umsetzung werden durch die Integration des C++-Moduls in die CNC-GEO-Task Taskgrenzen und damit Totzeiten vermieden. Für die Zustände reiner Kraftregelung wird die Schleppfehlerüberwachung der CNC deaktiviert, da die programmierte Endposition nicht immer erreicht werden kann. Für die Zustände reiner Positionsregelung können

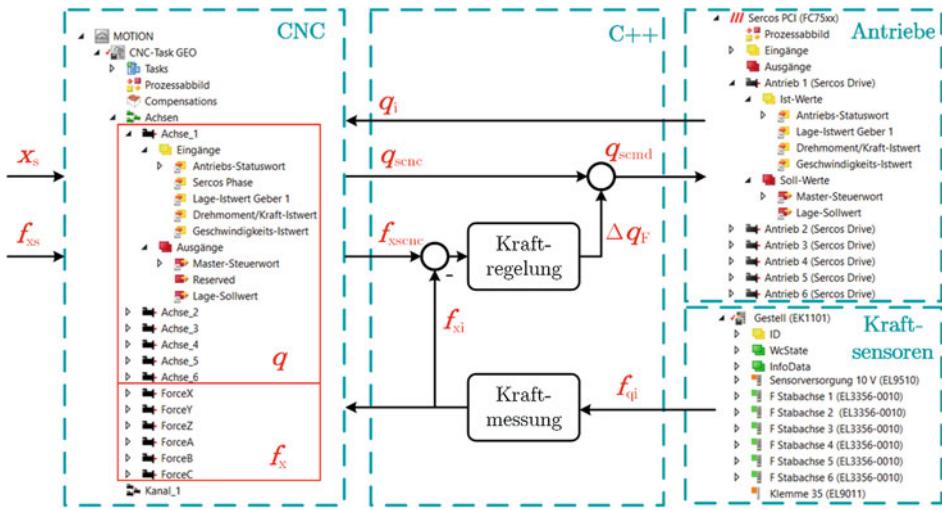


Abb. 22.8 Synchronie Sollwertvorgabe durch reale Positionier- und virtuelle Kraftachsen sowie „Man-in-the-Middle“-C++-Modul zur Stellgrößenaufschaltung in der Steuerung Beckhoff TwinCAT [20]

die Kraftachsen in den Nachführbetrieb geschaltet werden, um eine Anzeige der Istkraft mittels der virtuellen Achsen zu erreichen.

Listing 1 zeigt ein Beispiel zur Verwendung der Kraftregelung mit G-Code: Es werden die kinematische Koordinatentransformation in Euler-Winkel aktiviert, ein um $C = 30^\circ$ verdrehtes Basiskoordinatensystem definiert, die Kraftregelung im Selektions-Vektor für die $-Z$ -Richtung angewählt und die Startposition $X = 0, Z = 0$ mm angefahren. Anschließend wird die Kraftregelung mit automatischer Kontaktsuche aktiviert und schließlich während der Fahrt von $X = 0$ auf $X = 500$ mm die Kraft in $-Z$ -Richtung von 100 auf 300 N linear erhöht. Parallel wird die Z -Bewegung von $Z = 0$ auf $Z = 10$ mm vorgesteuert.

Listing 1: Beispiel für die Steuerung der Kraftregelung aus dem G-Code [20]

```

N00 #TRAFO OFF           ; (deactivate kinematic transformation)
N05 #KIN ID[65]          ; (select transformation with euler angles)
N10 ID[65].PARAM[6] = 30 ; (rot base around Z with 30°)
N15 ID[65].PARAM[15] = -1 ; (select force control in -Z-coordinates)
N20 #TRAFO ON            ; (activate trafo)
N25 G1 X=0 Z=0 ZF=100 F50 ; (move to start position X=0 mm, Z=0 mm)
N30 M51                  ; (activate force control, contact search)
N35 G1 X=500 Z=10 ZF=300 F50 ; (move to X=500 mm, Z=10 mm, with force ZF=300
N and velocity 50 mm/min) ; (and velocity 50 mm/min)
N40 M50                  ; (deactivate force control, clear offset)
N45 M30                  ; (end program)

```

Abschließend werden weitere Funktionen zur Steuerung und Parametrierung der Kraftregelung im G-Code verfügbar gemacht. Im Detail lassen sich über die Parameter Arbeitsraumgrenzen für die Kraftregelung, „weiche“ und „harte“ Reglerparameter für jeden der sechs Freiheitsgrade sowie Ursprungskoordinaten für die Regelung in radialen Koordinaten festlegen. Mittels weiterer M-Befehle wird der variable Taskspace de-/aktiviert sowie zwischen radialer und normaler Regelung umgeschaltet.

Exemplarisch zeigt schließlich Abb. 22.9 die Ergebnisse einer Konturverfolgungsaufgabe zur Validierung der Kraftregelung mit bzw. ohne Positions vorsteuerung der Kontur. Dazu wird eine am Endeffektor befestigte Rolle nach Kontakt suche und Einregelung einer konstanten Sollkraft von $ZF = -250$ N mit Positionsregelung in X-Richtung und Kraftregelung in Z-Richtung entlang einer der Steuerung unbekannten Kontur bewegt. Durch die konstante Sollkraft ergibt die Addition von CNC-Sollwert (grün) und Kraftregler-Offset (rot) für beide Experimente in den oberen Abbildungen die Ist-Kontur des Bauteils (blau). Dabei stellt sich eine maximale Regelabweichung von 5,9 % (ohne Vorsteuerung) bzw. 2,7 % (mit Vorsteuerung) ein (Abb. 22.9 unten). Für weitere Details zur Kraftregelung mit strukturintegrierter Kraftsensorik wird auf die Literatur verwiesen [20].

22.7 Fazit

In diesem Beitrag wurde die Messung von Kräften und Momenten an der Wirkstelle mit Hilfe von strukturintegrierten Kraftsensoren für Parallelkinematiken untersucht. Im Anschluss an die Aufstellung und Steuerungsintegration der Messmodelle wurde der Funktionsnachweis durch vergleichende Untersuchungen und die Anwendung einer Kraftregelung erbracht.

Die Messung und Regelung räumlicher Kräfte und Momente mit strukturintegrierter Kraftsensorik in Parallelkinematiken ist mit Hilfe eines Digitalen Zwillinges in Form eines steuerungsintegrierten echtzeitfähigen Messmodells möglich und erreicht dabei eine ähnliche Performance wie eine am Endeffektor angebrachte kommerzielle Kraftmessplattform.

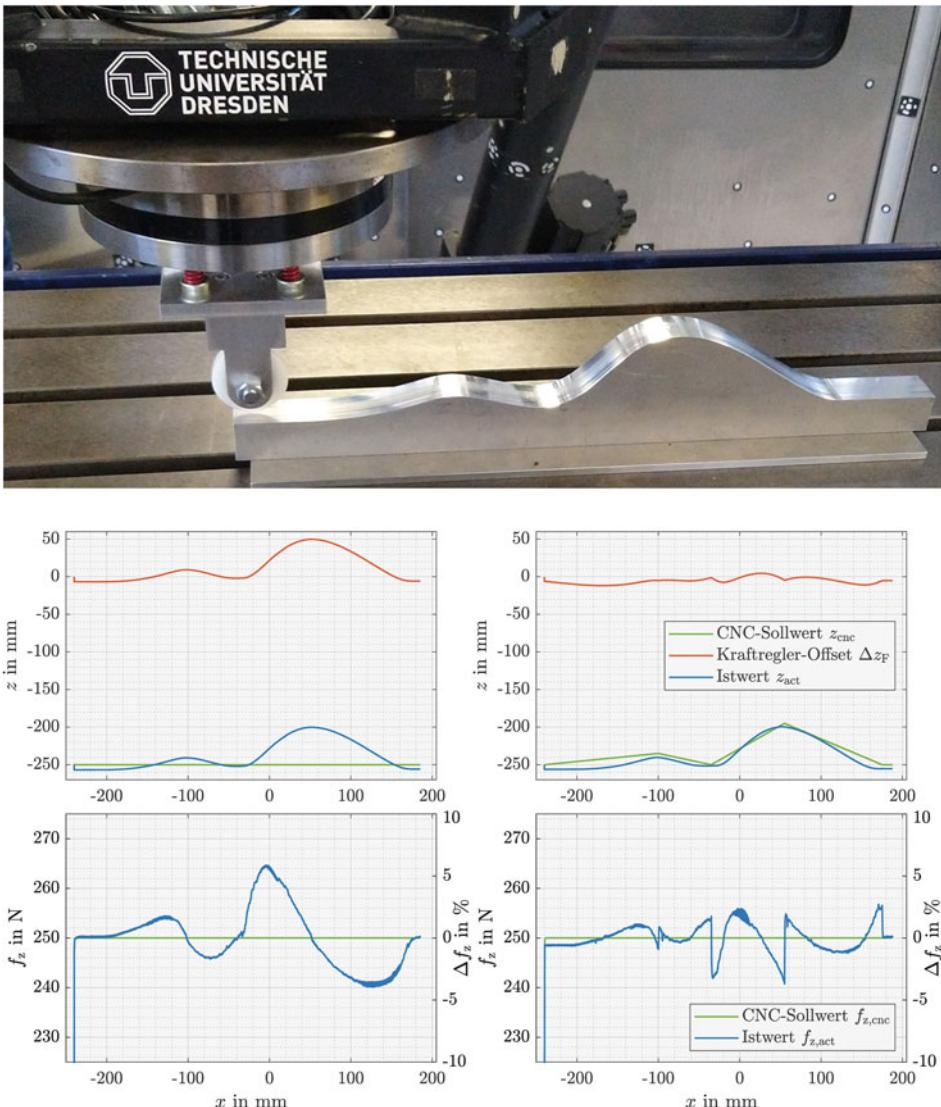


Abb. 22.9 Ergebnisse einer Konturverfolgungsaufgabe mit hybrider/paralleler Kraftregelung und Kraftmessung in der Endeffektor-Plattform (Messsystem 1b); oben: Weganteile in Z-Richtung aus CNC und Kraftregler sowie resultierender Istweg; unten: Soll- und Istkraftverlauf; links: ohne Vorsteuerung in Z, rechts: mit Vorsteuerung der Bauteilkontur mit 5 Stützstellen [20]

Gleichzeitig bestehen dabei wesentliche Vorteile, wie die hohe Robustheit durch größere Entfernung von der Wirkstelle mit Blick auf Späne und Kühlschmierstoffe, die Vermeidung von Einschränkungen bei Arbeitsraum und Befestigungsmöglichkeiten sowie die geringen Kosten durch die Verwendung einachsiger Standardkraftsensoren.

Literatur

1. Denkena B, Schmidt J, Krüger M (2014) Data mining approach for knowledge-based process planning. *Procedia Technol* 15:406–415
2. Stefanescu DM (2011) Handbook of force transducers: Principles and components. Springer Science & Business Media
3. Teti R, Jemielniak K, O'Donnell G, Dornfeld D (2010) Advanced monitoring of machining operations. *CIRP Annals* 59(2):717–739, ISSN 0007–8506
4. Desogus S, Germak A, Mazzoleni F, Quagliotti D, Barbato G, Barbieri A, Bigolin G, Bin C (2010) Developing multicomponent force transducers at INRIM. *IMEKO World Congress*, S 17–19
5. Genta G, Prato A, Mazzoleni F, Germak A, Galetto M (2018) Accurate force and moment measurement in spring testing machines by an integrated hexapod-shaped multicomponent force transducer. *Meas Sci Technol* 29(9):095902
6. Matich S, Hessinger M, Kupnik R (2017) Werthschützky und C. Hatzfeld: Miniaturized multiaxial force/torque sensor with a rollable hexapod structure. *tm-Technisches Messen*, 84(s1):138–142
7. Nitsche J, Baumgarten S, Petz M, Röske D, Kumme R, Tutsch R (2017) Measurement uncertainty evaluation of a hexapod-structured calibration device for multi-component force and moment sensors. *Metrologia* 54(2):171
8. Palumbo S, Germak A, Mazzoleni F, Desogus S, Barbato G (2016) Design and metrological evaluation of the new 5 MN hexapod-shaped multicomponent build-up system. *Metrologia* 53(3):956
9. Röske D (2003) Metrological characterization of a hexapod for a multi-component calibration device. In XVII IMEKO World Congress (Metrology in the 3rd millennium), S 347–351
10. Zhao Y, Zhang C, Zhang D, Shi Z, Zhao T (2016) Mathematical model and calibration experiment of a large measurement range flexible joints 6-UPUR six-axis force sensor. *Sensors* 16(8):1271
11. Friedrich C, Kauschinger B, Ihlenfeldt S (2016) Decentralized structure-integrated spatial force measurement in machine tools. *Mechatronics* 40:17–27
12. Friedrich C, Kauschinger B, Ihlenfeldt S (2019) Spatial force measurement using a rigid hexapod-based end-effector with structure-integrated force sensors in a hexapod machine tool. *Measurement* 145C:350–360
13. Friedrich C, Kauschinger B, Ihlenfeldt S (2020) Stiffness evaluation of a hexapod machine tool with integrated force sensors. *Journal of Machine Engineering* 20(1):58–69
14. Friedrich C, Ihlenfeldt S (2021) Model calibration for a rigid hexapod-based end-effector with integrated force sensors. *MDPI Sensors*, 21
15. Hesselbach J, Behrens BA, Dietrich F, Rathmann S, Poelmeyer J (2007) Flexible forming with hexapods. *Prod Eng* 1(4):429–436, ISSN 1863–7353

16. Großmann K, Friedrich C (2012) Steuerung einer neuartigen Hexapod-Arbeitsausrüstung für Radlader. Teil 1 – Systembeschreibung, Busauslegung und Steuerungsarchitektur. ZWF 12:897–902
17. Großmann K, Friedrich C (2013) Steuerung einer neuartigen Hexapod-Arbeitsausrüstung für Radlader. Teil 2 – Regelungskonzept, Hexapodkinematik und Sicherheitsfunktionen. ZWF 01/02:37–41
18. Woernle C (2016) Mehrkörpersysteme: Eine Einführung in die Kinematik und Dynamik von Systemen starrer Körper. Springer
19. Featherstone R (2008) Rigid body dynamics algorithms. Springer Nature
20. Friedrich C (2022) Räumliche Kraftmessung und -regelung mit Parallelkinematiken unter Verwendung strukturintegrierter Kraftsensorik. Dissertation. Technische Universität Dresden. <https://tud.qucosa.de/landing-page/https%3A%2F%2Ftud.qucosa.de%2Fapi%2Fqucosa%253A82770%2Fmets%2F/>
21. Friedrich C, Ihlenfeldt S (2022) Spatial Compliance Measurement of a Clamping Table with integrated Force Sensors. Journal of Machine Engineering 22:1. <https://doi.org/10.36897/jme/146533>
22. Friedrich C, Schlüter F, Ihlenfeldt S (2023) Spatial hybrid/parallel force control of a hexapod machine tool using structure-integrated force sensors and a commercial numerical control. Mechatronics 94. <https://doi.org/10.1016/j.mechatronics.2023.103007>. <https://www.sciencedirect.com/science/article/pii/S0957415823000636>

Open Access Dieses Kapitel wird unter der Creative Commons Namensnennung 4.0 International Lizenz (<http://creativecommons.org/licenses/by/4.0/deed.de>) 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 Kapitel 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.

