



High Performance Motion Control

Modellbasierte Entwicklung

Virtueller Sensorik



Perfection in Automation



Führend in Technologie



Weltweit nah
am Kunden

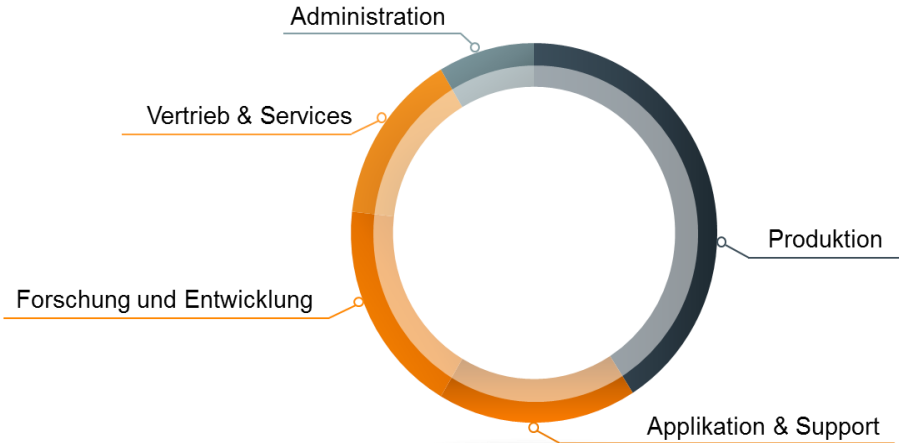


Wirtschaftlich
unabhängig

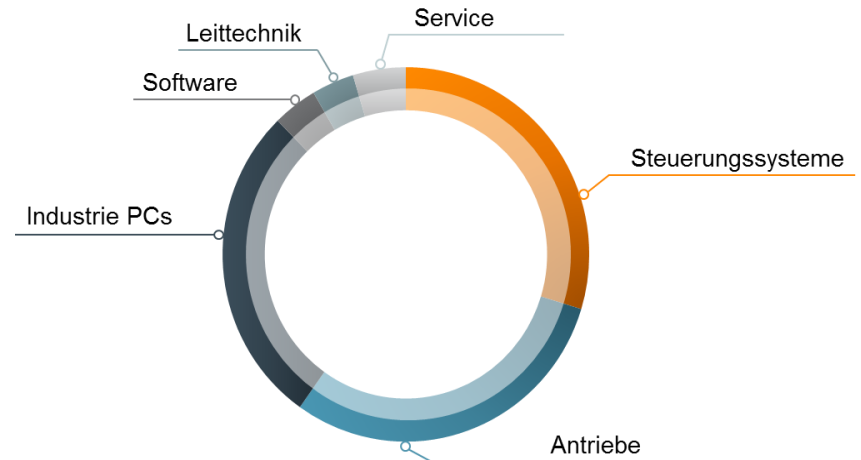
Unser Team für Ihren Erfolg



2.820 Mitarbeiter



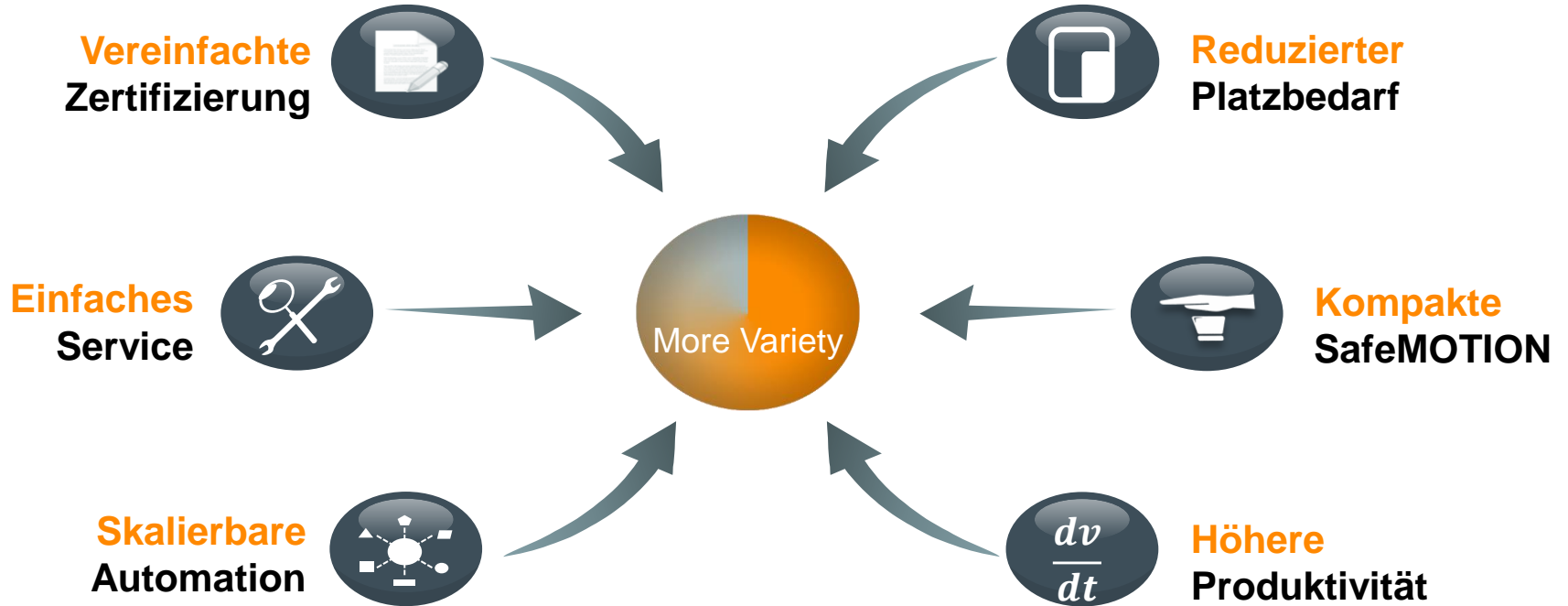
€ 585 Mio. Umsatz in 2015

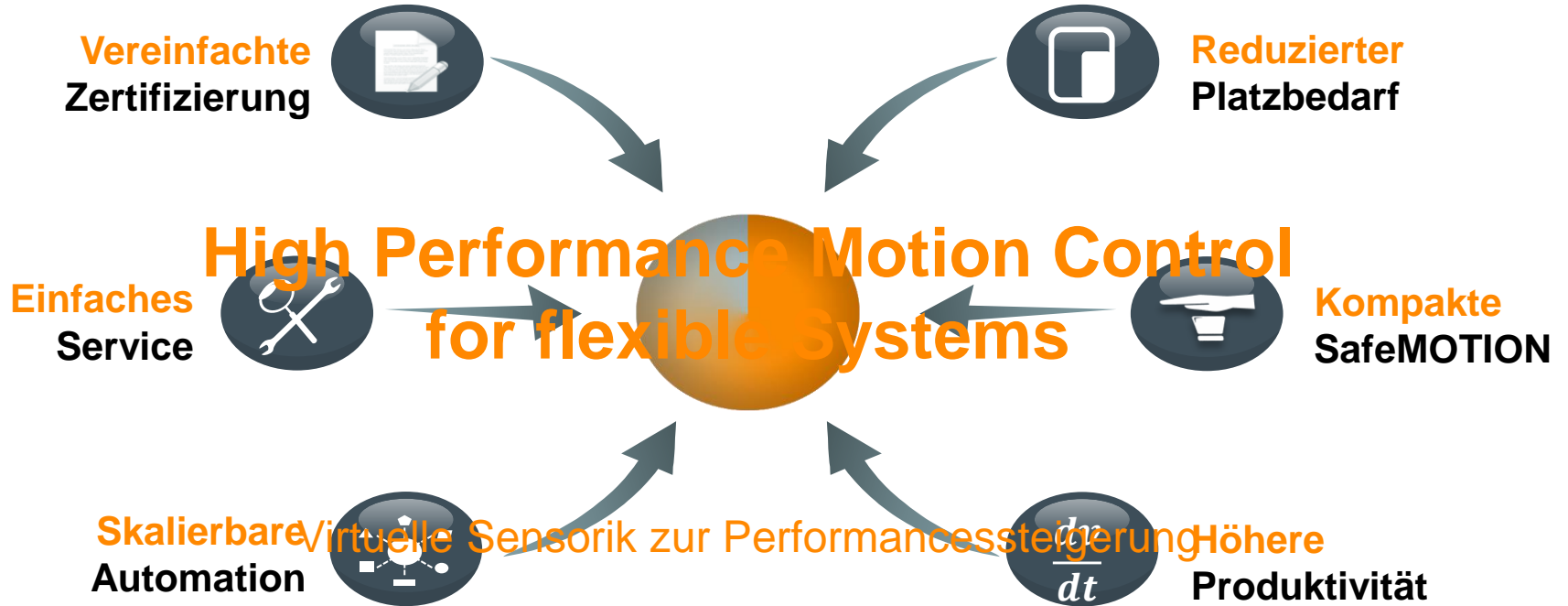




Neue Generation von Servoverstärkern





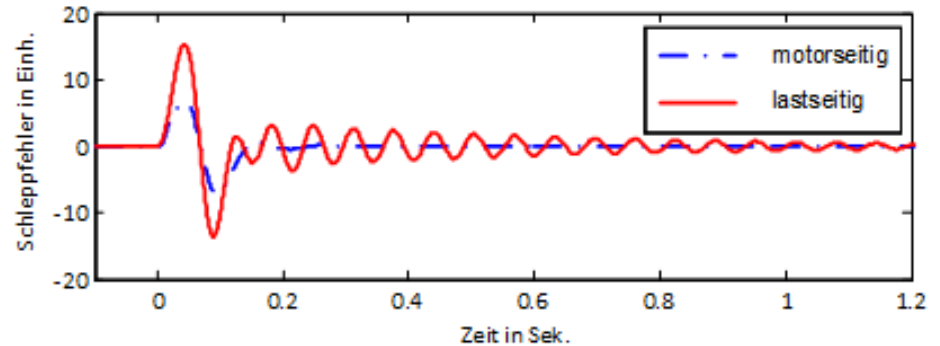
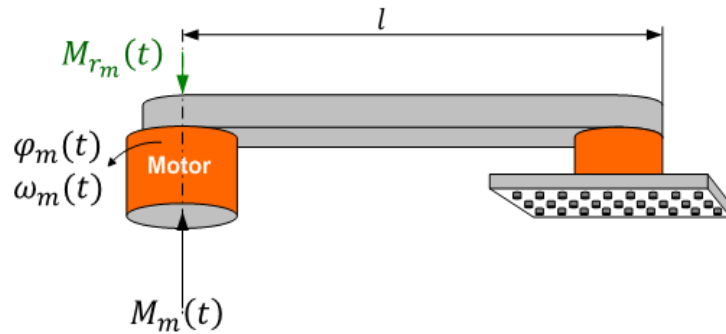




- Motivation: Virtuelle Sensorik zur Performancessteigerung
- Entwicklungsschritte eines Virtuellen Sensors
- Adaptierte Reglerkaskade mit Virtuellem Sensor
- Ergebnisse mit adaptierter Reglerkaskade



- Kostenoptimales mechatronisches Design und zunehmend steigende Maschinentaktraten
 - Geringere Systemsteifigkeiten
 - Antriebe werden im Grenzbereich betrieben
 - Direktantriebe: Größeres „Inertia Mismatch“

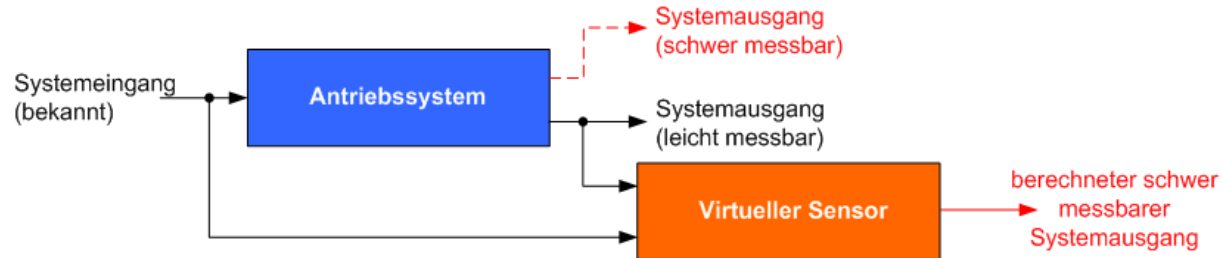


- Erhöhte (lastseitige) Schwingneigung in Servoantrieben
- Abhilfe: Virtueller Lastpositionssensor



Grundprinzip eines Virtuellen Sensors

- Mathematische Ermittlung von schwer messbaren Systemgrößen
 - Unter Verwendung leicht messbarer Systemgrößen
 - Unter der Verwendung einer mathematischen Beschreibung des Systems





Modellbasierte Entwicklung

● Modellbildung

- Physikalisches Modell
- Identifikation der Streckenparameter

● Lösungsansatz und Simulation

- Verhalten des geschlossenen Regelkreises mit Virtuellem Sensor

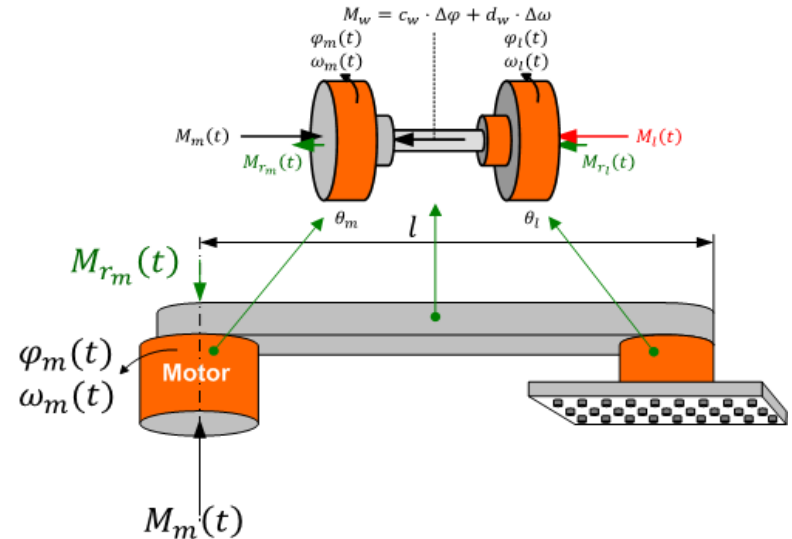
● Optimierung von Struktur & Parametern

● Codegenerierung (C, C++, VHDL, Verilog)

- Wrapper Funktion auf Basis einer Simulink S-function

● Verifikation des Codes (C, C++)

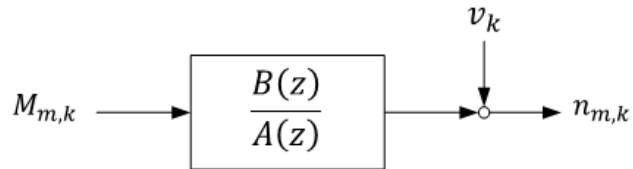
● Auswertung von Daten in MATLAB



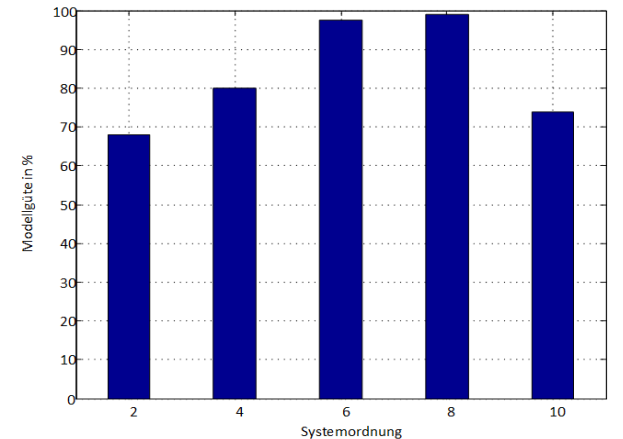
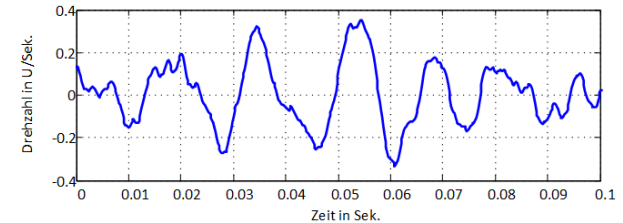
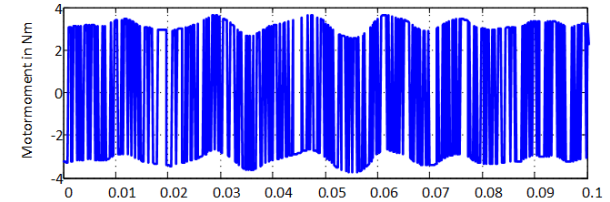


MATLAB® unterstützte Identifikation

- Systemidentifikation mit PBRs Anregung
- Identifikation eines OE – Modells (iterativer LS)



- Ordnungsbestimmung an Hand von Validationsdaten
 - NRMSE FIT Wert (Normalized Root Mean Square Error)

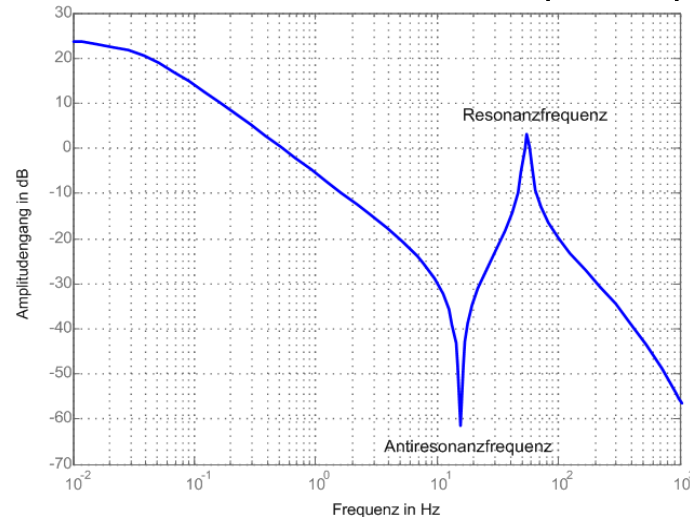




MATLAB® unterstützte Identifikation

- Ordnungsreduktion durch Pol- Nullstellenordnung (balancierte Eingangs-Ausgangsdarstellung)
- Suche nach den dominierenden Pol- Nullstellenpaaren (konjugiert komplex)

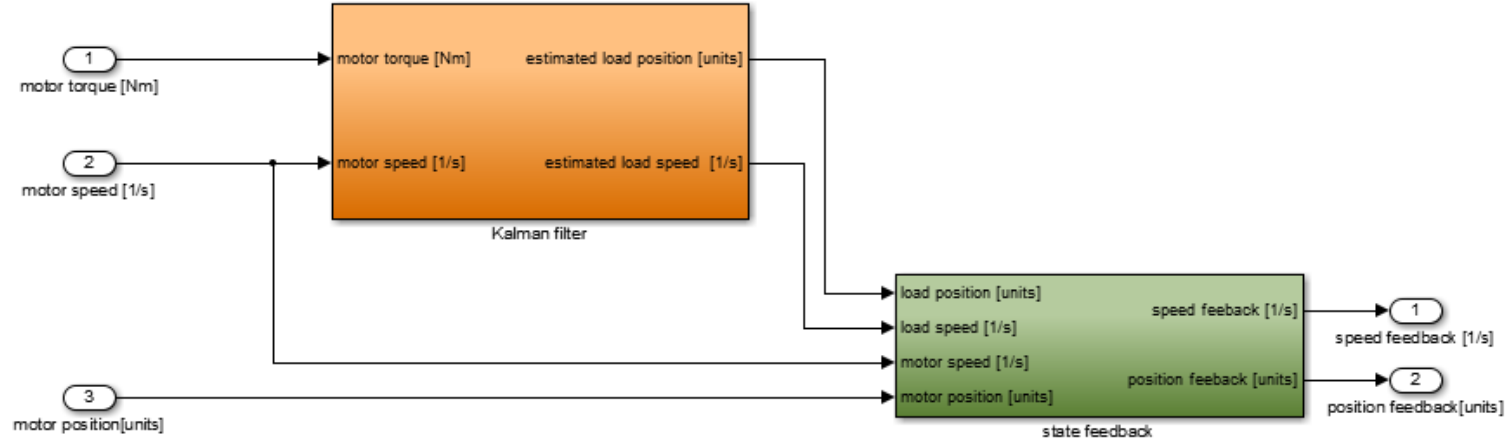
$$\tilde{G}_{n_m, M_m}(z) = \frac{\tilde{B}(z)}{\tilde{A}(z)} \quad \longrightarrow \quad \tilde{G}_{n_m, M_m}(z) = \tilde{V} \cdot \frac{z^2 + \tilde{b}_1 z^1 + \tilde{b}_0}{(z - 1) \cdot (z^2 + \tilde{a}_1 z + \tilde{a}_0)}$$





Lösungsansatz und Simulation /Optimierung von Parametern

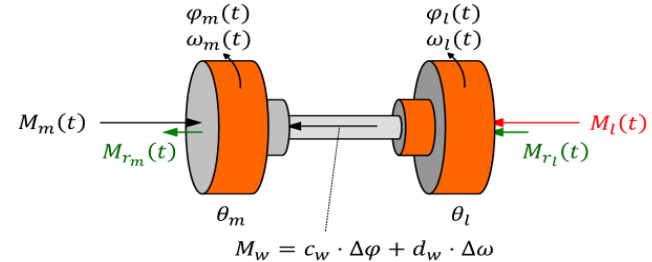
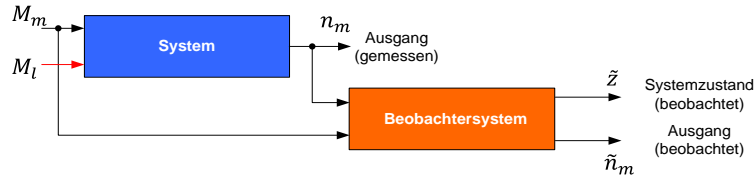
- Entwurf eines Kalmanfilters als Virtueller Sensor
- Verwendung der virtuellen Lastposition zur Performancesteigerung
- Optimierung der Parametrierung des Kalmanfilters und der adaptierten Regelung (Zustandsregelung) in einer Simulationsumgebung
 - Implementierung eines C MEX S-Function Wrappers





MATLAB® unterstützter Entwurf eines Kalman-Filters als Virtueller Sensor

● Zustandsbeobachter mittels Kalman-Filter



● Systembeschreibung in zeitdiskreter Form

$$x_{k+1} = A \cdot x_k + \begin{pmatrix} B_1 & B_2 \end{pmatrix} \cdot \begin{pmatrix} M_{m,k} \\ M_{l,k} \end{pmatrix}$$

$$y_k = C \cdot x_k$$

● Systemerweiterung

$$z_{k+1} = \begin{pmatrix} x_{k+1} \\ \eta_{k+1} \end{pmatrix} = \underbrace{\begin{pmatrix} A & B_2 \\ 0 & 1 \end{pmatrix}}_{\tilde{A}} \cdot \begin{pmatrix} x_k \\ \eta_k \end{pmatrix} + \underbrace{\begin{pmatrix} B_1 \\ 0 \end{pmatrix}}_{\tilde{B}} \cdot M_{m,k}$$

$$y_k = \underbrace{\begin{pmatrix} C & 0 \end{pmatrix}}_{\tilde{C}} \cdot \begin{pmatrix} x_k \\ \eta_k \end{pmatrix}$$



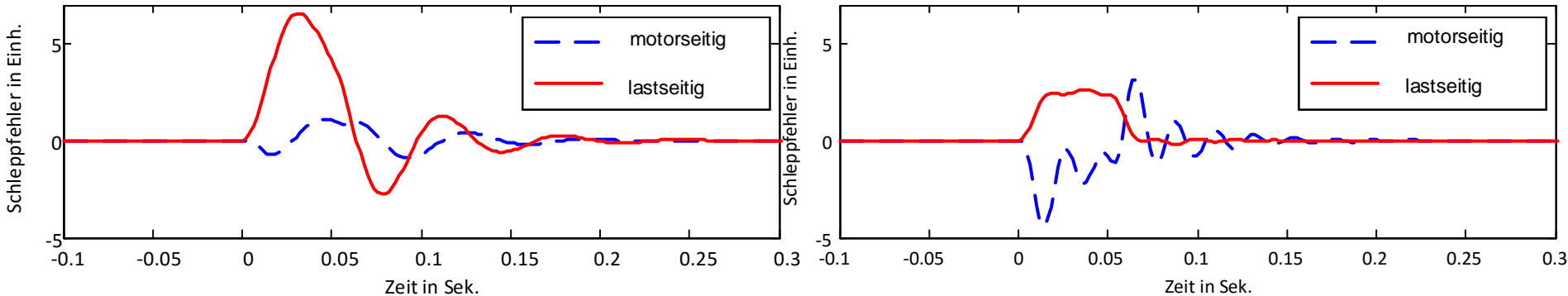
MATLAB® unterstützter Entwurf des Zustandsreglers

● Resultat bei Lastschlag

- Berechnung der Stellgröße $M_m = -k \cdot x + V \cdot r$, wobei k so berechnet wird, dass das Gütefunktional J ein Minimum wird

$$J = \int_0^{\infty} \left((\varphi_m(\tau) \quad \varphi_l(\tau)) \cdot \begin{pmatrix} (1-\lambda) \cdot q_0 & 0 \\ 0 & \lambda \cdot q_0 \end{pmatrix} \cdot \begin{pmatrix} \varphi_m(\tau) \\ \varphi_l(\tau) \end{pmatrix} + r_0 \cdot M_m^2(\tau) \right) \cdot d\tau$$

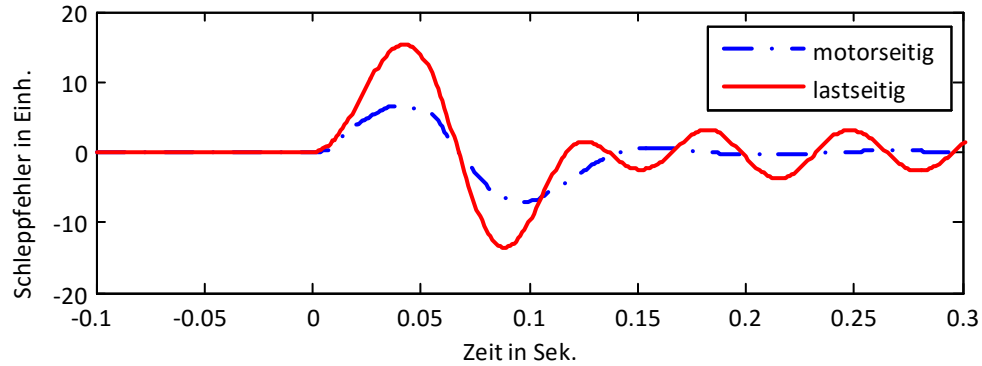
- Ergebnis bei lastseitiger (links) vs. motorseitiger (rechts) Gewichtung



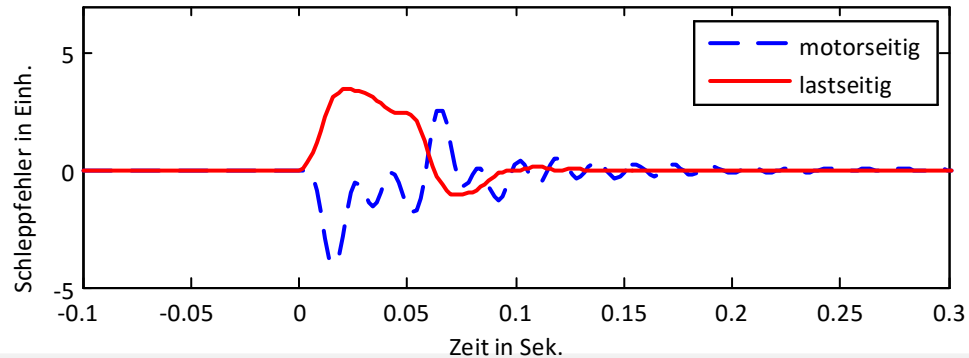


Ergebnis der Simulation

● Ohne Virtuellem Sensor

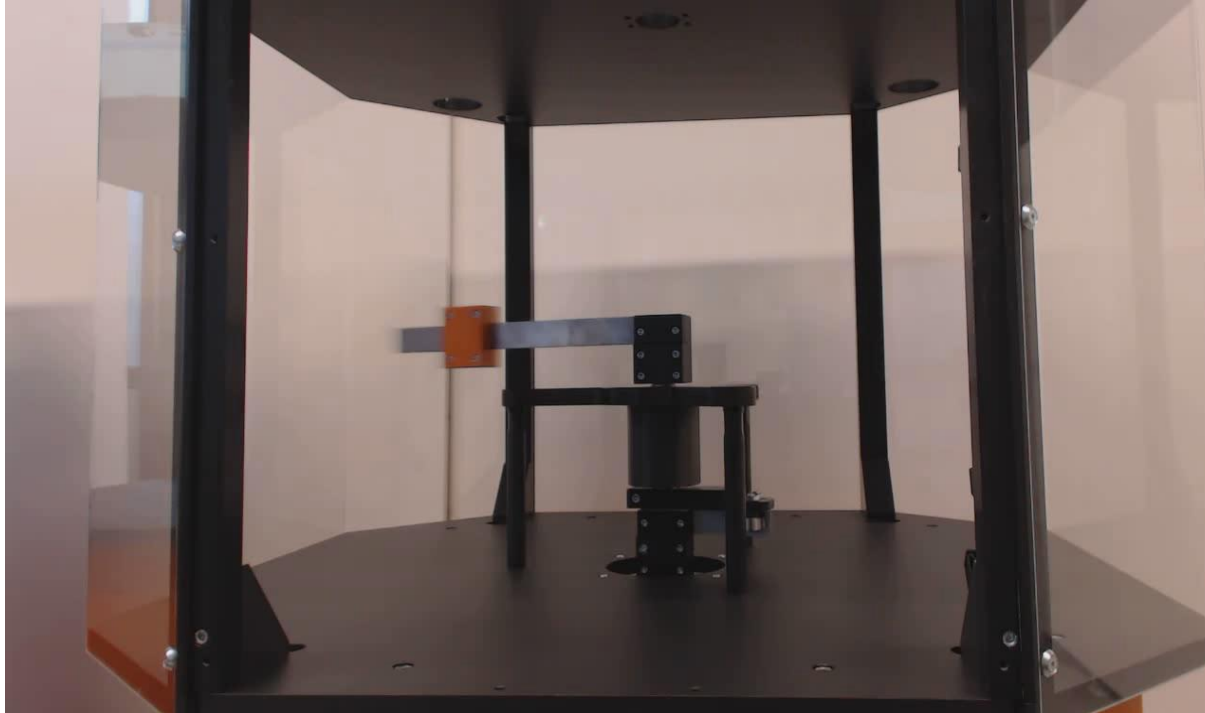


● Mit Virtuellem Sensor





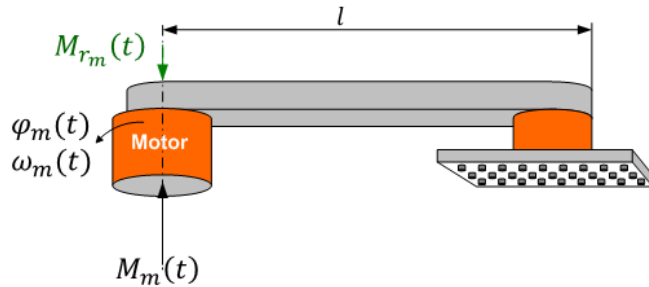
Ergebnis am realen Demonstrationsbeispiel



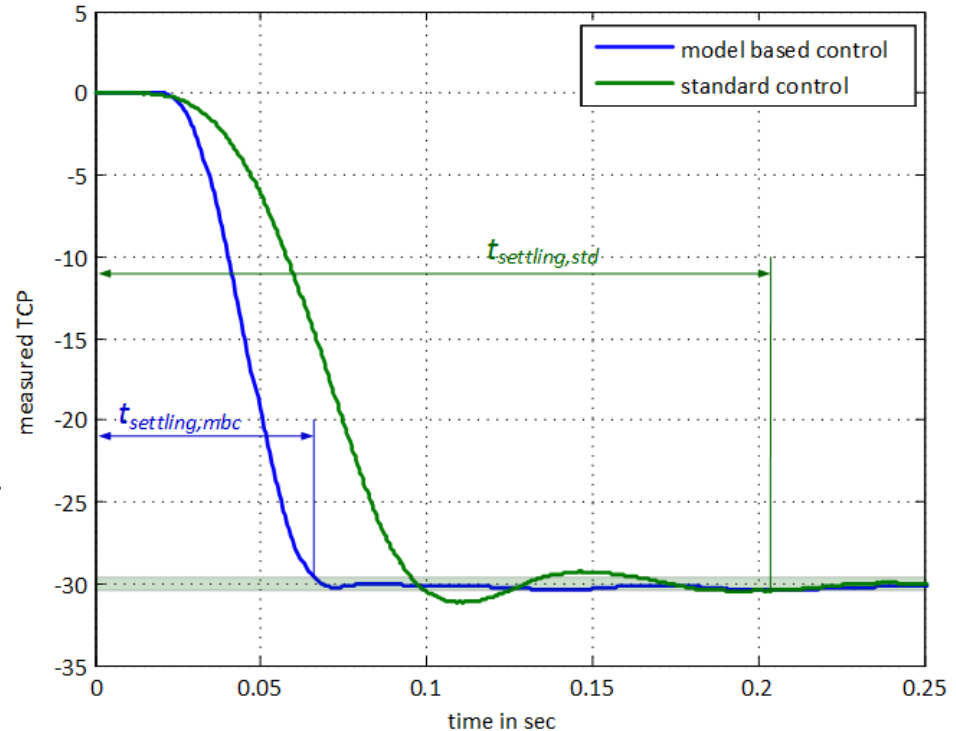


Ergebnis einer realen Applikation

Positionieren eines Auslegers



- Kurzes schrittweises Positionieren
- Nach jedem Positioniervorgang erfolgt ein Bearbeitungsschritt
- Viele einzelne Bearbeitungsschritte (ca. 50)
- Ziel 1: Schnelle Positionierung
- Ziel 2: Kürzest mögliche Einschwingzeit





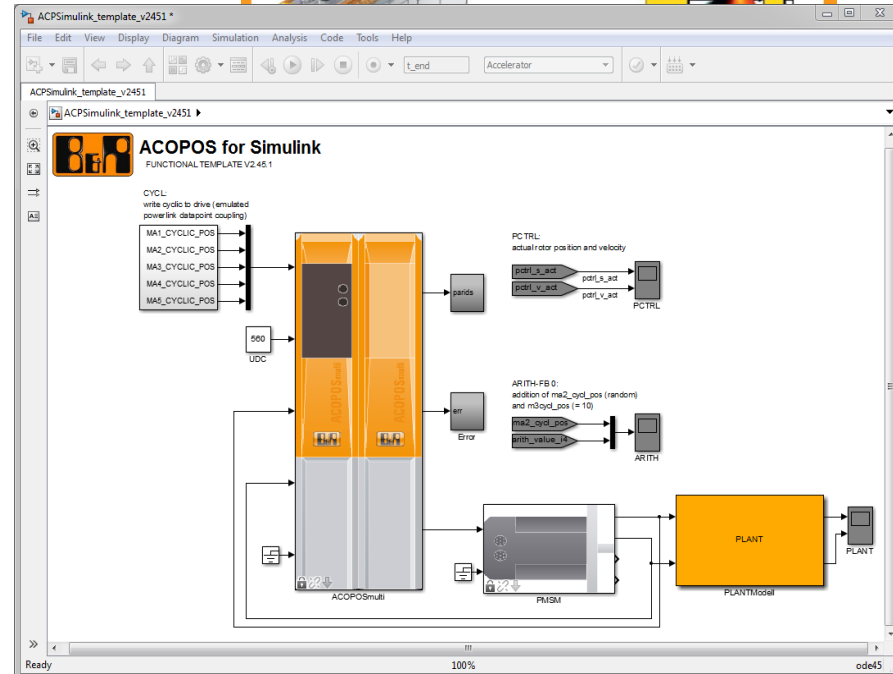
Modellbasierte Entwicklung für Maschinendesigner

Simulation des ACOPOS in Matlab / Simulink®

- Eigenständiges Matlab/Simulink® - Blockset
- Volle Funktionalität eines ACOPOS Servo-Antriebs

Eigenschaften

- Schnelle Simulation von ACOPOS + Motor
- Gewohnte Parametrierung des ACOPOS + Motor
- Simulation von bis zu 10 Achsen
- Schneller Einstieg auf Grund eines vorgefertigten Templates





LÖSUNGEN AUS EINER HAND WELTWEIT