



Echtzeit-Co-Simulation von Produktionsanlagen

3

Christian Scheifele und Alexander Verl

Zusammenfassung

Bei der virtuellen Planung, Auslegung und Inbetriebnahme steigen die Anforderungen an die Aussagekraft Digitaler Zwillinge, um eine umfassende Absicherung und Auslegung im digitalen Entwicklungsprozess zu erreichen. Die Zunahme der Modellkomplexität eines Digitalen Zwillings stellt die Echtzeitsimulation durch den steigenden Berechnungsaufwand vor neue Herausforderungen. Dieser Beitrag stellt eine Plattform zur Echtzeit-Co-Simulation von Digitalen Zwillingen für die Virtuelle Inbetriebnahme vor. Die im Beitrag vorgestellte parallelisierte Modellberechnung auf Basis einer Co-Simulationsarchitektur beabsichtigt die zur Modellberechnung nutzbare Rechenleistung zu steigern. Der Beitrag führt (i) in die Integration leistungsfähiger technologiespezifischer Simulationslösungen durch Integrationsschnittstellen und (ii) die Echtzeit-Co-Simulation von Produktionsanlagen auf Basis von Partitionierungs-, Parallelisierungs-, Synchronisations- und Datenaustauschmechanismen ein.

C. Scheifele (✉)

ISG Industrielle Steuerungstechnik GmbH, Stuttgart, Deutschland

E-Mail: christian.scheifele@isg-stuttgart.de

A. Verl

Institut für Steuerungstechnik der Werkzeugmaschinen und Fertigungseinrichtungen (ISW),

Universität Stuttgart, Stuttgart, Deutschland

© Der/die Autor(en) 2024

A. Verl, S. Röck und C. Scheifele (Hrsg.),

Echtzeitsimulation in der Produktionsautomatisierung,

https://doi.org/10.1007/978-3-662-66217-5_3

3.1 Einleitung

Die Digitale Fabrik ist der Oberbegriff für „[...] ein umfassendes Netzwerk von digitalen Modellen, Methoden und Werkzeugen [...]“ [1], welche sich in die Lebenszyklusphasen einer Produktionsanlage integrieren. Im Kontext der Digitalen Fabrik sind in den letzten Jahren eine Vielzahl von hochspezialisierten und technologiespezifischen Simulationslösungen entstanden. Mit diesen Entwicklungen können verschiedene Aspekte einer Produktionsanlage präzise virtuell abgebildet werden. Vor allem in Entwicklungsprozessen im Maschinen- und Anlagenbau kommen heute Simulationslösungen, beruhend auf digitalen Modellen und Methoden, zum Einsatz. Abb. 3.1 zeigt anhand des V-Modells als etablierte Entwicklungsmethodik für mechatronische Systeme die Verortung der Virtuellen Inbetriebnahme (VIBN) im Entwicklungsprozess von Produktionsanlagen.

Während in den frühen Phasen des Entwicklungsprozesses auf eine Simulationsdisziplin spezialisierte hochauflösende Simulationsmodelle zum Einsatz kommen, gewinnt im Verlauf des Entwicklungsprozesses bis hin zur Systemintegration die Betrachtung des

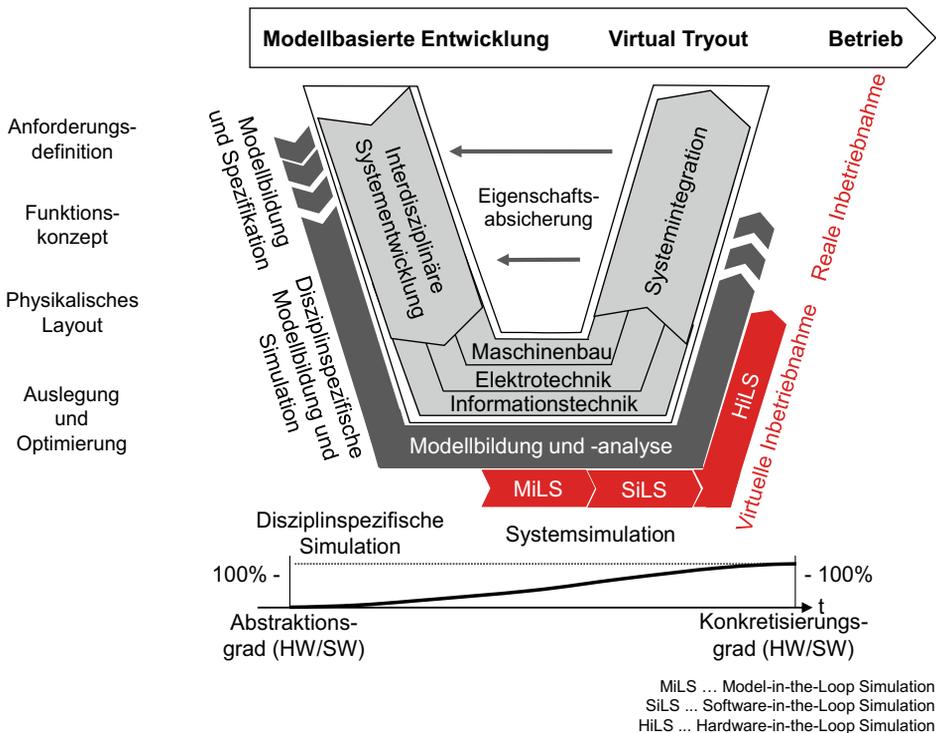


Abb. 3.1 V-Modell erweitert um digitale Modelle und Methoden, insbesondere der Virtuellen Inbetriebnahme

mechatronischen Gesamtsystems an Bedeutung. In den späten Phasen des Entwicklungsprozesses liegt der Fokus auf der virtuellen Absicherung der Interaktionen und Wechselwirkungen der Disziplinen eines mechatronischen Systems. Zur virtuellen Abbildung des Gesamtverhaltens der Maschine oder Anlage wird daher eine Systemsimulation – auch als gekoppelte Simulation oder Multidomain-Simulation bezeichnet – notwendig. Hierbei werden Modelle aus unterschiedlichsten Simulationsdisziplinen zu einem Gesamtmodell des Systems gekoppelt. Dies führt zu einem Simulationsmodell, welches das Zusammenspiel der mechatronischen Disziplinen und damit die Korrelationen abbilden kann. In diesem Beitrag wird hierfür der Begriff des Digitalen Zwillings verwendet. Ein solcher Digitaler Zwilling kommt beispielsweise in der Automatisierungstechnik bei einer VIBN zum Einsatz. Die VIBN ermöglicht es, die Inbetriebnahme der Automatisierungstechnik bereits vor der Verfügbarkeit der realen Lösung am Digitalen Zwilling durchzuführen.

Die VIBN unterscheidet zwischen verschiedenen Testkonfigurationen. Die Software-in-the-Loop Simulation (SiLS) bildet die Komponenten eines mechatronischen Systems vollständig virtuell mit Softwarekomponenten ab. Durch die Anbindung einer virtuellen/emulierten Steuerung kommen die originalen Steuerungsalgorithmen des Steuerungssystems zum Einsatz. Die Simulationszeit kann frei vom Simulator gesteuert werden.

In einer Hardware-in-the-Loop Simulation (HiLS) wird dagegen die reale Steuerungshardware über den industriellen Feldbus mit dem Digitalen Zwilling verbunden. Der Simulator bildet das Verhalten der Feldbuskomponenten am Feldbus ab, damit die Steuerung ohne aufwendige Änderung der Feldbuskonfiguration zum Einsatz kommen kann. Die Steuerung erkennt keinen Unterschied zwischen dem Feldbusverhalten der realen und virtuellen Feldbuskomponenten, wenn diese im Sinne einer Emulation das reale Verhalten korrekt nachbilden. Für den verlustfreien Signalaustausch mit dem Steuerungssystem muss die Simulation im Takt des realen Steuerungssystems laufen, was eine Simulation des Systemverhaltens mit harten Echtzeitkriterien (dies bedeutet, dass Simulationsergebnisse zuverlässig innerhalb einer festen Zeitschranke vorliegen müssen) notwendig macht.

Im Rahmen von komplexen Szenarien, bei denen mehrere Steuerungen gleichzeitig an den Digitalen Zwilling angeschlossen sind, treten auch kombinierte Formen der SiLS und HiLS auf. So können die einzelnen Steuerungssysteme entweder als reale Hardwarekomponenten (HiLS) oder als emulierte bzw. virtuelle Steuerungen (SiLS) gekoppelt werden. Sobald allerdings eine reale Hardwarekomponente im Einsatz ist, bestimmt diese die Taktung des Simulators, was zu harten Echtzeitanforderungen an die eingesetzten Simulationsmodelle führt.

Zur umfassenden virtuellen Absicherung und Auslegung müssen die Interaktionen zwischen Prozess, Maschine, Steuerungssystem und Bediener durch den Digitalen Zwilling abgebildet werden. Es besteht eine direkte Korrelation zwischen dem im Rahmen einer VIBN erreichbaren Konkretisierungsgrad von Hardware- (HW) und Softwarekomponenten (SW) der Produktionsanlage und der Modellgenauigkeit (siehe Abb. 3.2). Die Einhaltung der harten Echtzeitkriterien im Rahmen der HiLS trotz steigender

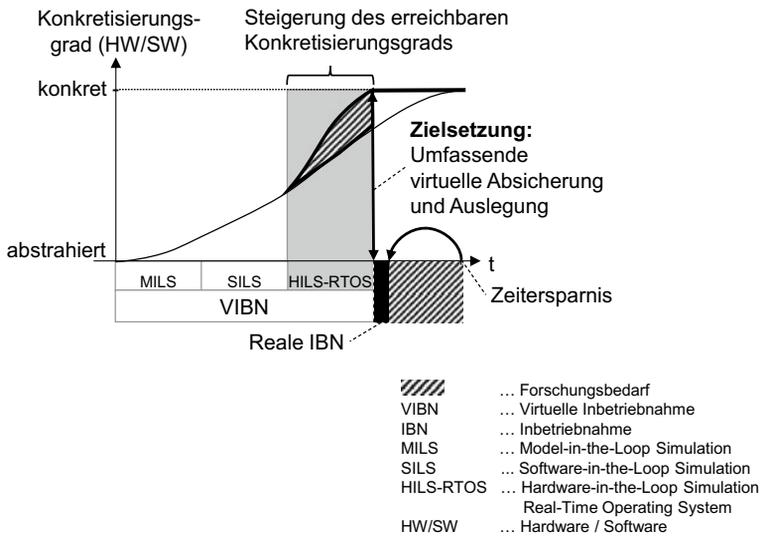


Abb. 3.2 Zielsetzung einer umfassenden virtuellen Absicherung und Auslegung im Rahmen der Virtuellen Inbetriebnahme

Modellgenauigkeit stellt eine große Herausforderung und aktuell ein Defizit der VIBN dar.

Laut einer Studie des VDMA in 2017 [2] erwartet die Mehrheit der Maschinen- und Anlagenbauer eine steigende Nachfrage nach einer VIBN im Entwicklungsprozess ihrer Produktionsanlagen (57 % der befragten Unternehmen). Handlungsbedarf sieht der VDMA unter anderem im Bereich der Abbildung physikalischer Effekte durch bestehende Technologien (21 % der befragten Unternehmen). Daraus lässt sich die Anforderung an eine höhere Modelltiefe von Digitalen Zwillingen ableiten [3]. Hierzu sind neue Ansätze zur Steigerung der in der HiLS nutzbaren Rechenleistung gefragt. Um hochauflösende Modelle zu erhalten, wird die Integration von disziplinspezifischen, leistungsfähigen und performanten Modellen der Digitalen Fabrik in den Digitalen Zwilling vorgeschlagen.

Dieser Beitrag stellt eine Plattform zur Echtzeit-Co-Simulation für die VIBN vor. Die vorgestellte parallelisierte Modellberechnung auf Basis einer Co-Simulationsarchitektur beabsichtigt die zur Modellberechnung nutzbare Rechenleistung zu steigern. Um die nutzbare Rechenleistung und damit die mögliche Modellkomplexität zu erhöhen, werden Partitionierungs-, Parallelisierungs-, Synchronisierungs- und Datenaustauschmechanismen vorgeschlagen. Für die Integration leistungsfähiger technologiespezifischer Simulationslösungen werden Integrationsschnittstellen für eine blockbasierte Modellierung vorgestellt.

3.2 Stand der Technik

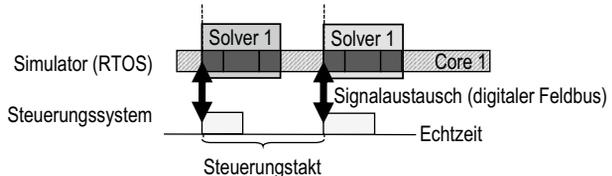
Um die Jahrtausendwende wurden in verschiedenen wissenschaftlichen Arbeiten die Architektur von Simulatoren für die HiLS untersucht. *Wünsch* teilt in [4] wissenschaftliche Ansätze für die HiLS unter anderem anhand der Simulationszeitbasen für Steuerungstakte von hundert, zehn und einer Millisekunde (ms) ein. Untersuchte Echtzeitansätze für zehn und hundert Millisekunden basieren dabei meist auf einem Windows-Betriebssystem. Um Simulationszeiten bis in den Bereich unter einer Millisekunde zu erreichen, ist die Verwendung eines Echtzeitbetriebssystems (RTOS) notwendig [5]. Während *Zäh* [6] und *Mewes* [7] Windows-basierte HiLS für Steuerungstakte im Bereich zwischen zehn und hundert Millisekunden vertreten, formulieren *Pritschow* und *Röck* [5] für CNC-Maschinen die Notwendigkeit einer zeitdeterministischen Simulation auf einem Echtzeitbetriebssystem für Steuerungstakte bis in den Bereich unter einer Millisekunde. Eine ausführliche Darstellung wissenschaftlicher Ansätze zeigt [4].

Nach *Pritschow* und *Röck* [5] muss eine HiLS das Maschinenverhalten innerhalb eines Steuerungstakts (heute im Bereich bis unter einer ms bei CNC-Werkzeugmaschinen) berechnen, um den Anforderungen an eine zeitsynchronen und verlustfreien Datenaustausch zwischen Steuerung und Simulation über den Feldbus gerecht zu werden (siehe Abb. 3.3).

Wenn die Simulationsarchitektur auf einem einzigen Rechenkern (Solver), wie in Abb. 3.3 gezeigt, basiert, spricht man von einer geschlossenen Simulationsarchitektur [8]. Der Rechenkern unterwirft das Gesamtmodell einem einheitlichen Simulationstakt, der mit der Steuerung und dem digitalen Feldbus synchronisiert ist. Nach *Pritschow* und *Röck* wird das Gesamtmodell sequenziell auf einem Prozessorkern berechnet. Hierdurch ist die verfügbare Rechenleistung begrenzt.

Dieser Nachteil begrenzt die Modelltiefe der Verhaltensmodelle und den Modellumfang des Gesamtsystems [3, 9, 10]. Als Modelltiefe wird in diesem Beitrag die Abbildungsgenauigkeit eines Verhaltensmodells bezeichnet, während der Modellumfang die Größe eines Simulationsmodells beschreibt. Trotz bekannter Verfahren zur Modellreduktion, effizienten numerischen Methoden und optimierten Simulationsansätzen können detaillierte Modelle der Strukturmechanik (z. B. Finite-Elemente-Methode, flexible Mehrkörpersysteme), zur hochgenauen Prozesssimulationen (z. B. Materialabtrag, Spanbildung, Materialfluss) und für die präzise 3D-Kinematiksimulation mit

Abb. 3.3 Zeitsynchrone und verlustfreie Kommunikation zwischen industriellem Steuerungssystem und Systemsimulation (Digitaler Zwilling)



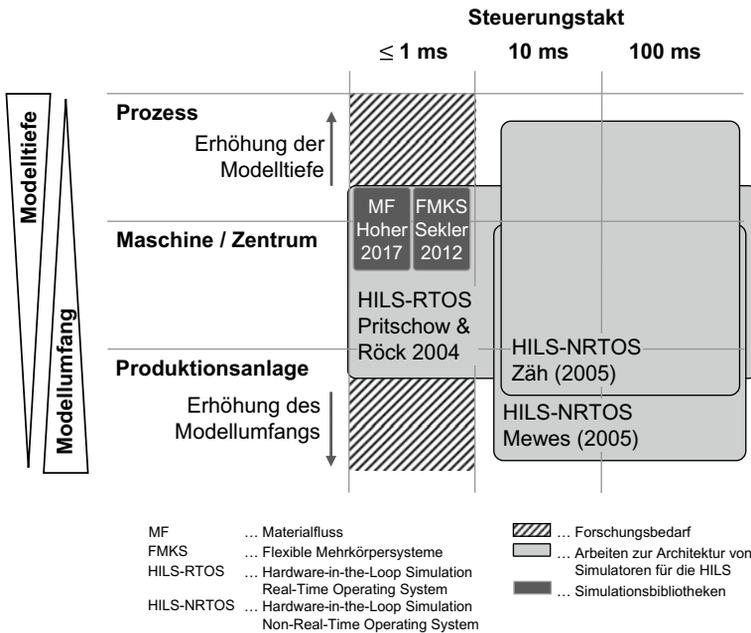


Abb. 3.4 Forschungsbedarf und Betrachtungsfokus im Kontext ausgewählter wissenschaftlicher Arbeiten im Bereich der Hardware-in-the-Loop Simulation

Kollisionserkennung aktuell nur mit starken Vereinfachungen und Anpassungen berücksichtigt werden. Abb. 3.4 stellt den Forschungsbedarf und Betrachtungsfokus des Beitrags in Anlehnung an ausgewählte wissenschaftliche Arbeiten im Bereich der HiLS mit industriellen Steuerungssystemen [5, 6, 7] sowie Simulationsdisziplin-spezifische Untersuchung von Simulationsbibliotheken zur Materialflusssimulation [8] und zur flexiblen Mehrkörpersimulation [11] dar.

Während bei der Berechnung auf einem einzelnen Prozessorkern die Berechnung sequentiell durchgeführt wird, kann bei Mehrkernprozessoren eine parallelisierte Berechnung durchgeführt werden [12]. Durch Modellpartitionierung und parallelisierte Simulation in einer Co-Simulationsarchitektur können Rechenzeiten signifikant reduziert werden [13]. In verschiedensten Simulationsdomänen wurden daher in den vergangenen Jahren Co-Simulationsarchitekturen wissenschaftlich untersucht [14].

In Vorarbeiten wird zur Behebung der aktuellen Einschränkungen der geschlossenen Simulationsarchitektur bei der HiLS mit industriellen Steuerungssystemen eine Partitionierung des Gesamtmodells in unabhängig lösbare Teilmodelle zur parallelen Modellberechnung auf Mehrkernprozessoren durch Kopplung und Synchronisation von Teilmodell-Berechnungen in einer Co-Simulation vorgeschlagen [3, 9, 10]. Um die Notwendigkeit einer Einhaltung der harten Echtzeitbedingungen bei der HiLS zu verdeutlichen, wird dieser Ansatz als Echtzeit-Co-Simulation bezeichnet und in aktuellen

Arbeiten am ISW wissenschaftlich für die HiLS mit industriellen Steuerungssystemen untersucht.

Bisherige Arbeiten auf dem Gebiet der Co-Simulation (z. B. [13, 15]) lassen sich weitestgehend der Offline-Co-Simulation mechanischer Systeme zuordnen. Offline-Co-Simulationsansätze gehen von einer virtuellen Zeitachse für die Simulationszeit aus. In einer Offline-Co-Simulation werden daher die einzelnen Teilmodelle in der Regel sequentiell berechnet und dabei Echtzeitanforderungen vernachlässigt. Für die HiLS mit realer Steuerungshardware werden jedoch steuerungsspezifische Echtzeitanforderungen an den Simulator gestellt. In diesem Beitrag wird daher ein Echtzeit-Co-Simulationsansatz mit parallelisierter Berechnung untersucht.

In Vorarbeiten [3] wurde eine erste Realisierung der Echtzeit-Co-Simulation am Beispiel der physikbasierten Materialflusssimulation vorgestellt. Die eingesetzte Co-Simulation basiert auf zwei miteinander gekoppelten Rechenkernen. Jeder Rechenkern hat einen anderen Zeitkontext (z. B. realer Steuerungstakt, ein Vielfaches des Steuerungstakts) und ein anderes Betriebssystem (z. B. Windows, RTOS). Die Simulation der einzelnen Teilmodelle erfolgt parallel. Zur Steigerung der Modellgenauigkeit werden über eine Multi-Rate Methode fehlende Koppelsignale im schnelleren Rechenkern extrapoliert. Multi-Rate Methoden sind seit langem Gegenstand wissenschaftlicher Untersuchungen [16].

In verschiedenen Beiträgen [9, 10] wird auf die Notwendigkeit einer Integrationschnittstelle für Systemsimulationswerkzeuge zur Integration technologiespezifischer Simulationslösungen hingewiesen. Dies wird damit begründet, dass hochspezialisierte, technologiespezifische Simulationslösungen für eine hochauflösende Abbildung des Systemverhaltens benötigt werden. Derzeit werden für jedes Simulationstool für die VIBN eigenständige Simulationsbibliotheken neu entwickelt. Im Bereich der VIBN existieren derzeit keine oder nur rudimentär definierte Werkzeugketten und Modellaustauschstandards. Durch eine Integrationschnittstelle können Teilmodelle von den entsprechenden Technologieexperten einfach integriert werden, um den steigenden Anforderungen an Genauigkeit und Funktionalität eingesetzter Modelle gerecht zu werden. Ein möglicher Standard für die Integration disziplinspezifischer Modelle in eine Systemsimulation ist das Konzept des Functional Mock-up Interface (FMI) [17].

3.3 Konzept einer Plattform zur Echtzeit-Co-Simulation

Zur Erweiterung der Simulationsarchitektur von einer geschlossenen Architektur zu einer Co-Simulationsarchitektur ist eine Betrachtung der Modellierung des Gesamtmodells, der Modellpartitionierung und Initialisierung sowie der Echtzeitberechnung und Echtzeit-Synchronisation der Co-Simulation notwendig (siehe Abb. 3.5).

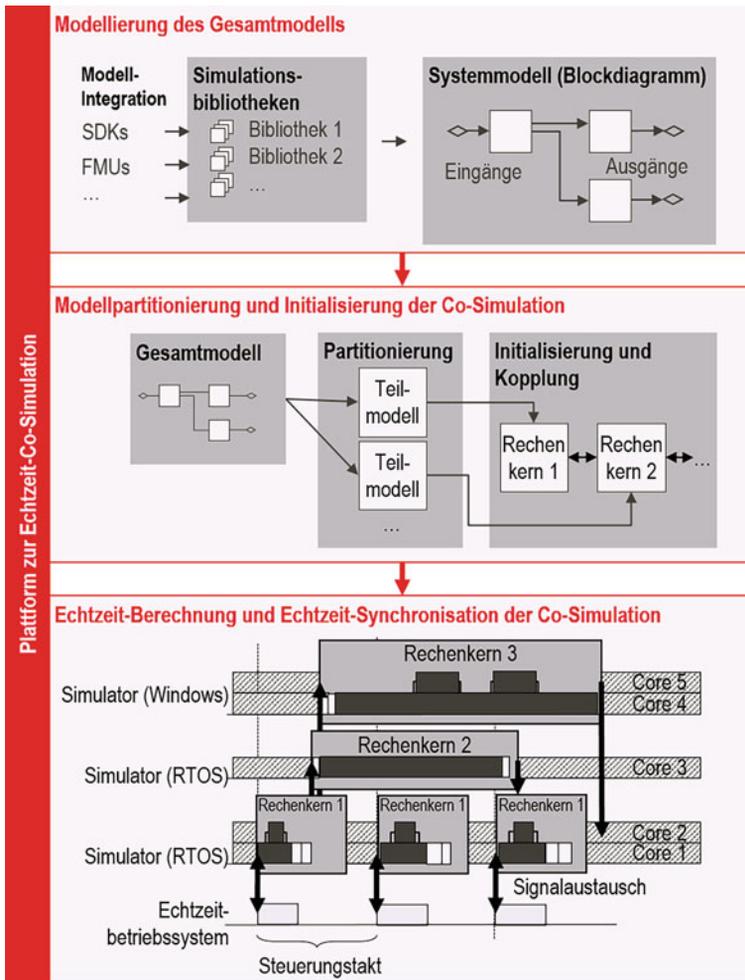


Abb. 3.5 Konzept der Plattform zur Echtzeit-Co-Simulation für die Virtuelle Inbetriebnahme industrieller Steuerungssysteme

3.4 Modellierung und Modellpartitionierung

Nach *Gipser* [18] wird ein dynamisches System definiert, indem seine Objekte, die Kopplungen zwischen den Objekten, deren Eingangsgrößen und deren Ausgangsgrößen spezifiziert und beschrieben werden. *Gipser* unterscheidet zwischen internen Kopplungen (Verbindungen zwischen Objekten) und externen Kopplungen (Eingangs- und Ausgangsgrößen). Für eine ganzheitliche Beschreibung von Produktionsanlagen sind

unterschiedlichste Simulationsdisziplinen in einer Systemsimulation miteinander zu koppeln. Neben der Beschreibung des Prozesses ist das Verhalten der Maschine (z. B. Logik, Kinematik, Dynamik) einschließlich des Verhaltens der industriellen Steuerungskomponenten am industriellen Feldbus mit entsprechender Sensorik und Aktorik zu simulieren. Bei der Simulation von Produktionsanlagen wird die Abbildung des dynamischen Verhaltens des Materialflusses zwischen den einzelnen Maschinen notwendig. Je nach gewünschter Modelltiefe wird für jede Simulationsdisziplin der passende Simulationsansatz ausgewählt. Bei der Zusammensetzung des Simulationsmodells entsteht eine Simulationskonfiguration (siehe Abb. 3.6). Der Übergang zwischen Simulationskonfigurationen kann als Skalierung der Modelltiefe des Simulationsmodells bezeichnet werden.

Zur Modellierung des Systemverhaltens können unterschiedliche Modellierungsansätze verwendet werden. Modellierungsansätze verwenden mathematische Gleichungen die entweder direkt verwendet oder durch eine grafische Darstellung repräsentiert werden [18]. Ein gängiger grafischer Modellierungsansatz ist das Blockschaltbild (oder Signalfussdiagramm) zur Abbildung der Wirkstruktur eines dynamischen Systems. Diese Modellierungsmethode ist im Maschinen- und Anlagenbau weit verbreitet.

Bei der Echtzeit-Co-Simulation soll analog zur geschlossenen Simulationsarchitektur die Modellierung und Konfiguration des Digitalen Zwillings weiterhin in einer einzigen, integrierenden Modellierungsoberfläche auf Basis eines Blockschaltbilds erfolgen. Um die von der Plattform bereitgestellten Simulationsbibliotheken um technologiespezifische

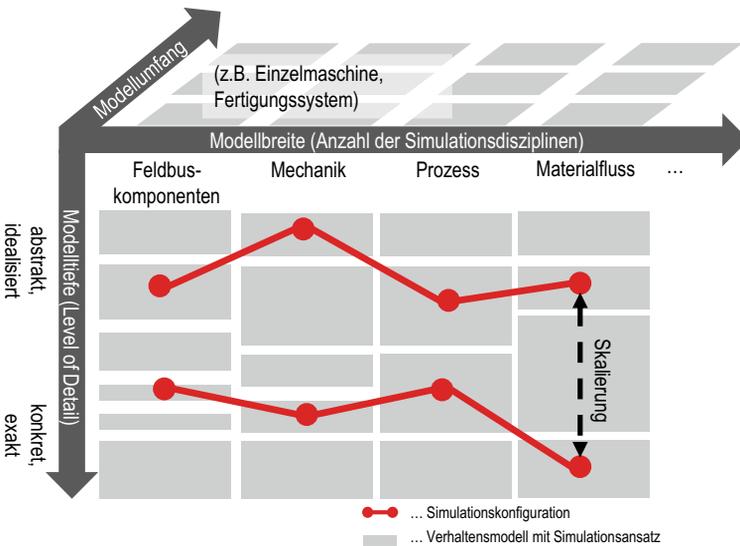


Abb. 3.6 Simulationskonfigurationen eines Digitalen Zwillings und der Übergang zwischen Simulationskonfigurationen (Skalierung)

Simulationslösungen zu erweitern, müssen Integrationsschnittstellen bereitgestellt werden. Die Modellierung der technologiespezifischen Simulationsmodelle kann entweder direkt im Blockschaltbild der Plattform oder im disziplinspezifischen Simulationswerkzeug mit einem individuellen Modellierungsansatz erfolgen. Bei externer Modellierung wird das Simulationsmodell als abgeschlossene Black-Box im Blockschaltbild integriert.

Bei den bisherigen Anwendungen der Co-Simulation bestand die Idee darin, einzelne Teilmodelle von Komponenten zu einem ganzheitlichen Systemmodell zusammenzuführen. Eine Komponente wird hier bereits als Partition gesehen. Bei der Echtzeit-Co-Simulation von Produktionsanlagen wird dagegen die Partitionierung des Block-basierten Systemmodells in mehrere Teilmodelle zur Parallelisierung der Echtzeit-Berechnung notwendig. Die dabei erzeugten Partitionen bilden Teilmodelle über die Komponentengrenzen hinweg.

Zur Modellpartitionierung ist eine Partitionierungsstrategie erforderlich, die das Gesamtmodell in geeignete Teilmodelle aufteilt. Zur Erfüllung der Echtzeit-Anforderungen einer HiLS ist die Definition von Simulationsebenen erforderlich, die die erforderliche Zykluszeit und Echtzeiteigenschaften eingesetzter Modelle definieren (z. B. im Takt des Steuerungssystems oder in einem Vielfachen des Steuerungstakts). Ein Vielfaches des Steuerungstakts ist nur möglich, wenn nicht in jedem Steuerungstakt eine Simulationsantwort vorhanden sein muss. Zustandsmodelle für die Feldbuskommunikation von Antrieben können beispielsweise nicht ausgelagert werden, da das Steuerungssystem eine Rückmeldung der Simulation in jedem Steuerungstakt benötigt. Multi-Rate Methoden bieten in manchen Anwendungsfällen die Möglichkeit durch die Prädiktion der Koppelsignale, Modelle in einem Vielfachen des Steuerungstakts zu berechnen und trotzdem die verlustfreie Kommunikation mit dem Steuerungssystem herzustellen [3]. Für einen wirtschaftlichen Einsatz sind automatisierte Partitionierungsmechanismen erforderlich, um eine Erhöhung der Modellierungszeiten zu verhindern. Zur Automatisierung der Partitionierung müssen die Simulationsbibliotheken um Konfigurations- und Parametrierungsparameter für die Echtzeit-Co-Simulation erweitert werden.

In der Modellierungsumgebung muss für den Anwender ersichtlich sein, wie das Gesamtmodell auf die einzelnen Rechenkerne verteilt wird. Dabei müssen Anpassungen und Umkonfigurationen durch den Benutzer ermöglicht werden. Zusätzlich sind Mechanismen zur Überprüfung der Partitionierung durch den Anwender oder durch die automatische Modellpartitionierung notwendig.

3.5 Initialisierung, Echtzeit-Berechnung und Echtzeit-Synchronisation

Das Gesamtmodell wird basierend auf einer Partitionierungsstrategie in Partitionen aufgeteilt, um die Berechnung auf mehrere Rechenkern mit unterschiedlicher Taktung und Ausführungsplattform zu verteilen bzw. zu parallelisieren. Bei der Initialisierung werden die Koppelsignale aus dem Gesamtmodell automatisch abgeleitet und entsprechende Kommunikationskanäle zwischen den Rechenkernen automatisch konfiguriert. Auch über die Grenzen der Ausführungsplattformen (z. B. Windows, RTOS) muss eine performante Kommunikation ermöglicht werden. Zur Prädiktion von fehlenden Koppelsignalen zwischen Rechenkernen mit unterschiedlichster Taktung kann die Anwendung von Multi-Rate Methoden im schnelleren Rechenkern notwendig sein.

Im Rahmen der Echtzeitberechnung müssen die einzelnen Rechenkern entsprechend den Anforderungen einer HiLS synchronisiert werden. Die Anforderungen einer HiLS sind:

- die Aufrechterhaltung der verlustfreien und taktsynchronen Kommunikation mit den industriellen Steuerungssystemen
- die Gewährleistung des Zeitdeterminismus und der Reproduzierbarkeit von Simulationsläufen

Aufgrund der Parallelisierung und der Einbindung von Windows-Rechenkernen kann die vollständige zur Verfügung stehende Rechenleistung für die Echtzeitberechnung ausgenutzt werden. Durch die Verfügbarkeit Windows-basierter Rechenkern können auch technologiespezifische, Windows-basierte Simulationsbibliotheken in den Digitalen Zwilling integriert werden.

3.6 Integration technologiespezifischer Simulationslösungen

Die Kopplung von Simulationsmodellen aus verschiedenen Simulationsdisziplinen über Toolgrenzen hinweg ist seit langem ein Thema wissenschaftlicher Arbeiten. *Pritschow* sieht beispielsweise in [19] ein zentrales Problem in der Zusammenführung von Modellen aus verschiedenen Simulationsdisziplinen und formuliert zwei Handlungsfelder für zukünftige wissenschaftliche Untersuchungen:

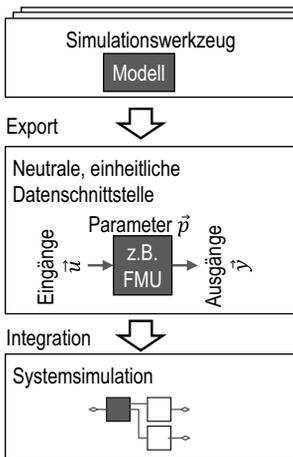
- Modelldaten werden über eine neutrale, einheitliche Datenschnittstelle mit einem einheitlichen Datenformat zur Beschreibung der Modelle verschiedener Disziplinen zu einer Systemsimulation zusammengeführt. Das Gesamtsystem wird dann in einem einzigen, integrierenden Simulationswerkzeug berechnet.

- Direkte Kopplung von Simulationswerkzeugen aus verschiedenen Simulationsdisziplinen zu einer Systemsimulation. Die Simulation läuft verteilt auf mehreren Simulationswerkzeugen.

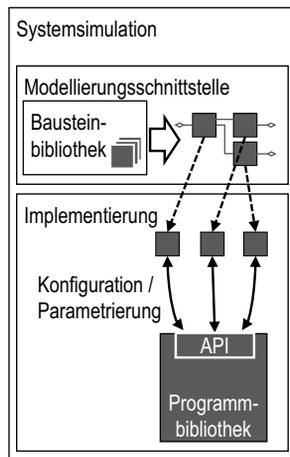
Im Kontext der Digitalen Fabrik sind in der Zwischenzeit technologiespezifische simulationsbasierte Lösungen entstanden. Zur umfassenden Simulation komplexer Prozesse müssen bestehende Lösungen in den Digitalen Zwilling integriert werden. Abb. 3.7 stellt drei grundlegende Ansätze für Blockschaltbilder zur Integration technologiespezifischer Simulationslösungen dar.

Black-Box-Integration abgeschlossener Modelle: Für die Integration von Simulationsmodellen, die in einem technologiespezifischen Simulationswerkzeug mit einem werkzeugabhängigen Modellierungsansatz modelliert wurden, wird mittels eines Modelaustauschstandards (z. B. FMI) die Modellintegration ermöglicht. Das Systemsimulationswerkzeug stellt eine Integrationschnittstelle über einen Black-Box Ansatz zur Verfügung.

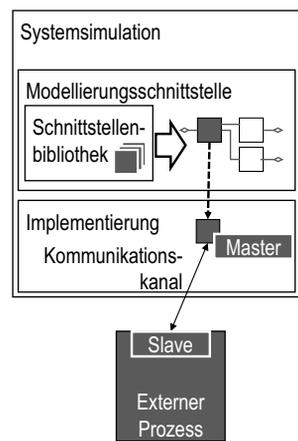
a) Black-Box Integration abgeschlossener Modelle



b) Bausteinbibliothek zur Modellierung und Konfiguration von Programmbibliotheken



c) Anbindung von eigenständigen Simulationslösungen (Tool-Kopplung)



FMU: Functional-MockupUnit
API: Application-Programming-Interface

Abb. 3.7 Grundlegende Ansätze für Blockschaltbilder zur Integration technologiespezifischer Simulationslösungen (a) Black-Box Integration abgeschlossener Modelle; (b) Bausteinbibliothek zur Modellierung und Konfiguration von Programmbibliotheken; (c) Anbindung von eigenständigen Simulationslösungen (Tool-Kopplung)

Bausteinbibliothek zur Modellierung und Konfiguration von Programmbibliotheken: Steht eine technologiespezifische Simulationslösung als konfigurierbare und parametrierbare Programmbibliothek (z. B. Physik-Engine) zur Verfügung, kann über die Anwendungsprogrammierschnittstelle (API) im Systemsimulationswerkzeug eine Bausteinbibliothek zur Modellierung und Konfiguration entwickelt werden.

Anbindung von eigenständigen Simulationslösungen (Tool-Kopplung): Zur Anbindung einer eigenständigen Simulationslösung als externer Prozess ist eine Kommunikationsschnittstelle erforderlich. Im Idealfall stellt das Systemsimulationswerkzeug den Kommunikations-Master dar, der die externe Simulation taktet und den Signalaustausch steuert.

3.7 Realisierung der Integrationsschnittstellen

Bei der Realisierung der Plattform zur Echtzeit-Co-Simulation wurden drei beispielhafte technologiespezifische Simulationslösungen anhand der zuvor eingeführten Integrations-schnittstellen in die Echtzeit-Co-Simulation integriert. Diese werden im Folgenden näher betrachten:

- *Black-Box-Integration abgeschlossener Modelle:* Integration eines abgeschlossenen Teilmodells in die Systemsimulation am Beispiel einer Functional Mock-up Unit (FMU).
- *Integration von Programmbibliotheken am Beispiel der Abbildung physikalischer Effekte:* Für die Integration hochauflösender physikalischer Effekte wurde eine Bausteinbibliothek zur benutzerfreundlichen Modellierung und Konfiguration der leistungsfähigen Funktionalitäten einer Physik-Engine (in diesem Fall mit AgX Dynamics und NVIDIA PhysX) erstellt.
- *Integration von Steuerungsalgorithmen:* Im Rahmen eines komplexen Szenarios kann die Anbindung mehrerer Steuerungen an den Digitalen Zwilling notwendig sein. Konzentriert sich die virtuelle Validierung auf ein einziges Steuerungssystem oder sind die industriellen Steuerungssysteme nur teilweise als reale Hardwarekomponenten verfügbar, bietet es sich an, einzelne Steuerungssysteme durch eine emulierte bzw. virtuelle Steuerung zu ersetzen. Am Beispiel einer virtuellen CNC wurde eine

Bausteinbibliothek erstellt, welche die Konfiguration und Einbindung der Programm-bibliothek „kernelv“ (die Simulations-DLL des ISG-kernels) in Verbindung mit der Echtzeitberechnung ermöglicht.

Anwendungsbeispiel „Functional Mock-Up Unit“

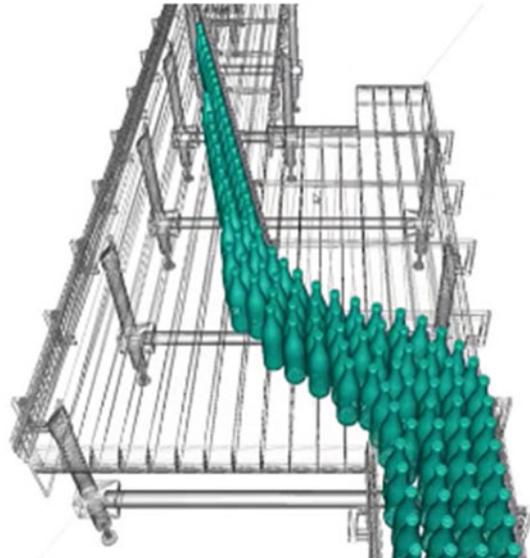
Zur hochauflösenden Abbildung spezifischer Effekte bietet sich die Integration abgeschlossener Modellen aus technologiespezifischen Simulationswerkzeugen in den Digitalen Zwilling an. FMI ist eine standardisierte Schnittstelle zum Austausch von Modellen über Tool-Grenzen hinweg. Mittlerweile wird FMI von einigen führenden Simulationswerkzeugen unterstützt (z. B. MATLAB/Simulink, MapleSim, Altair Activate). Zur Untersuchung der Anforderungen an eine FMU bei der Integration in die Echtzeitberechnung im Rahmen einer VIBN wurde eine FMI-Integrationschnittstelle in der Echtzeit-Co-Simulations Plattform realisiert.

Anwendungsbeispiel „kinetisches Prozessverhalten“

Abb. 3.8 und 3.9 zeigen zwei Anwendungsbeispiele, die eine hochauflösende Abbildung des dynamischen Materialflusses notwendig machen. Zum einen wurde der Materialfluss in der Getränkechnik am Beispiel eines Glide-Liners dargestellt. Für jeden Simulationsschritt werden die Kollisionen zwischen den einzelnen Materialien anhand der geometrischen Beschreibungen mit einem Windows-basierten Rechenkern (physikalische Simulation, Zykluszeit ~16 ms) erkannt und aufgelöst. Der Windows-basierte Rechenkern ist mit einem Rechenkern basierend auf RTOS verbunden (Antriebsmodelle, Kinematik

Abb. 3.8

Anwendungsbeispiel
„Glide-Liner“



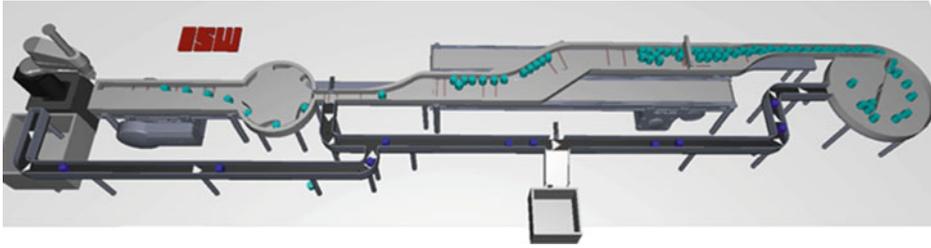


Abb. 3.9 Anwendungsbeispiel „Mehrskaliges Materialflussmodell am Beispiel der Verpackungstechnik“

etc., Zykluszeit~1 ms). Im zweiten Anwendungsfall wurde an einem Beispiel aus der Verpackungstechnik ein mehrskaliges Materialflussmodell integriert, bestehend aus einem physikalischen Windows-basierten Materialfluss (Zykluszeit~16 ms, Windows-basierte Physik-Engine) und einer kinematischen Rückführung im Takt des Steuerungssystems (Zykluszeit~1 ms).

Die Modellierung der physikalischen Effekte erfolgt auf Basis einer Bausteinbibliothek im Blockschaltbild des Digitalen Zwillings. Abb. 3.10 zeigt ein Blockschaltbild eines Materialflussmodells.

Anwendungsbeispiel „elastostatisches Prozessverhalten“

Bei der CNC-Bearbeitung mit Robotersystemen ist das Schlauchpaket zur Kabelführung oft massiven Kräften ausgesetzt. Um die Belastungen des Schlauchpakets abbilden zu können, müssen die Wechselwirkungen zwischen Prozess, Schlauchpaket, Robotersystem, Steuerung und Bediener in einer einzigen integrierenden Simulation dargestellt

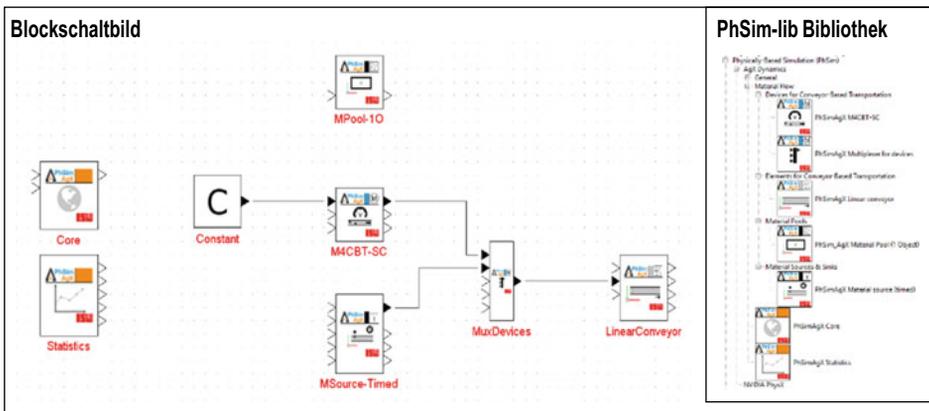


Abb. 3.10 Physik-Bibliothek für Materialflusssysteme

werden. Durch die Abbildung des flexiblen Verhaltens können Spannungen und Beschädigungen des Schlauchpakets aufgrund einer ungünstigen Bahnplanung simuliert und im Rahmen einer VIBN bereits simulativ erkannt werden. Im Rahmen der prototypischen Umsetzung der Echtzeit-Co-Simulation wurde eine Simulationsbibliothek erstellt, die das elastostatische Verhalten von Schlauchpaketen in der Echtzeitberechnung abbilden kann.

Anwendungsbeispiel „blockbasierte virtuelle CNC“

Im konkreten Anwendungsbeispiel wurden einzelne CNC-Steuerungssysteme durch emulierte bzw. virtuelle Steuerungen ersetzt. Die in Abb. 3.11 dargestellte virtuelle CNC-Bibliothek im Blockschaltbild modelliert und übernimmt die Konfigurationsdaten direkt aus dem Digitalen Zwilling. Beispielsweise sind kinematische Parameter im Digitalen Zwilling bereits bekannt und können für die Konfiguration der virtuellen CNC ohne weitere Benutzereingaben verwendet werden.

Ist keine reale Steuerung an die Simulation gekoppelt, liegt eine reine SiLS mit virtueller Zeitachse vor. Selbst sehr rechenaufwändige Simulationsmodelle können dabei in den Digitalen Zwilling integriert werden, da durch die virtuelle Zeitachse keine Echtzeitanforderungen an die Simulation gestellt werden. Sobald ein reales Steuerungssystem mit der Simulation gekoppelt wird bestimmt diese die Taktung des Simulators, wodurch Echtzeitanforderungen erfüllt werden müssen.

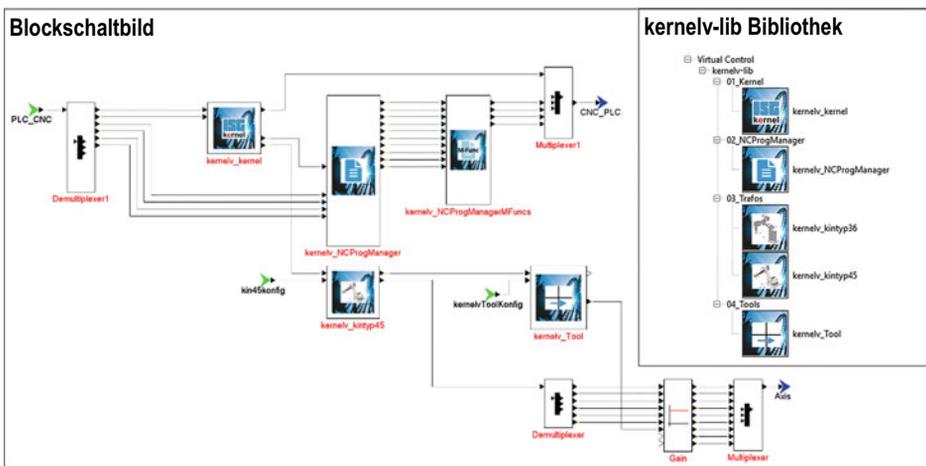


Abb. 3.11 Bausteinbibliothek einer virtuellen CNC zur Integration in den Digitalen Zwilling

3.8 Zusammenfassung und Ausblick

Das in diesem Beitrag vorgestellte Konzept einer Plattform zur Echtzeit-Co-Simulation ermöglicht die Steigerung der Aussagekraft Digitaler Zwillinge bei der virtuellen Planung und Auslegung von Produktionsanlagen im Rahmen einer VIBN. Durch Integrationschnittstellen können hochspezialisierte Simulationslösungen in die Systemsimulation integriert werden. Durch die Echtzeit-Co-Simulation und die damit verbundenen Ansätze zur Parallelisierung der Berechnungen kann die bei der Echtzeitberechnung zur Verfügung stehende Rechenkapazität massiv gesteigert werden. Ziel ist es, die Wechselwirkungen zwischen Prozess, Maschine, Steuerung und Bediener im Rahmen einer HiLS (Zykluszeiten ~ 1 ms) mit einer hohen Modelltiefe, Modellumfang und Modellbreite (vgl. Abb. 3.6) in einem einzigen integrierten Simulationsmodell der Produktionsanlage abzubilden.

Der Beitrag betrachtet neben der Vorstellung eines Konzepts für eine Plattform zur Echtzeit-Co-Simulation die Integration leistungsfähiger technologiespezifischer Simulationslösungen durch Integrationschnittstellen. Zukünftige Arbeiten fokussieren sich auf die wissenschaftliche Betrachtung von Partitionierungs-, Parallelisierungs-, Synchronisations- und Datenaustauschmechanismen.

Literatur

1. VDI 4499: Digitale Fabrik – Grundlagen (2008) <http://www.beuth.de>
2. VDMA: IT-Report Simulation + Visualisierung (2017) Einsatz von Simulationswerkzeugen, Einbindung in die Organisation und Entwicklungstendenzen. <http://www.vdma.org>
3. Scheifele C, Verl A (2016) Hardware-in-the-Loop simulation for machines based on a multi-rate approach. In: Proceedings of The 9th Eurosim Congress on Modeling and Simulation, EUROSIM, S 715–720
4. Wünsch G (2007) Methoden für die virtuelle Inbetriebnahme automatisierter Produktionssysteme. TU München, Dissertation
5. Pritschow G, Röck S (2004) “Hardware in the Loop” simulation of machine tools. CIRP Annals 53(1):295–298
6. Zäh M, Schack R, Carnevale M, Müller S (2005) Ansatz zur Projektierung der Digitalen Fabrik. ZWF – Zeitschrift für wirtschaftlichen Fabrikbetrieb 5:286–290
7. Mewes J (2005) Virtuelle Inbetriebnahme mit realen Automatisierungssystemen und virtuellen Maschinen. Deutsch-Niederländische Automatisierungstage
8. Hoher S (2017) Ein gekoppeltes Materialflussmodell zur durchgängigen Entwicklungsunterstützung von Materialflusssteuerungen. Universität Stuttgart, Dissertation
9. Scheifele C, Riedel O, Verl A (2017) Virtuelle Inbetriebnahme komplexer Produktionsanlagen mittels echtzeitfähiger Co-Simulation. In: AUTOMATION 2017: 18. Leitkongress der Mess- und Automatisierungstechnik, S 45–46
10. Scheifele C, Verl A (2017) Von der Virtuellen Maschine zur Virtuellen Produktion mit Ansätzen der Echtzeit-Co-Simulation. In: Industrie 4.0. Vision und Realität. Fortschritt-Berichte VDI. Reihe 2, Fertigungstechnik. 696:81–90
11. Sekler P (2012) Modellbasierte Berechnung der Systemeigenschaften von Maschinenstrukturen auf der Steuerung. Universität Stuttgart, Dissertation

12. Dronka S (2004) Die Simulation gekoppelter Mehrkörper- und Hydraulik-Modelle mit Erweiterung für Echtzeitsimulation. TU Dresden, Dissertation
13. Günther F (2017) Beitrag zur Co-Simulation in der Gesamtsystementwicklung des Kraftfahrzeugs. TU München, Dissertation
14. Gomes C, Thule C, Broman D, Larsen P, Vangheluwe H (2018) Co-Simulation: A survey. *ACM Computing Surveys (CSUR)*. 51(3) Article No. 49:1–33
15. Benedikt M, Watzenig D, Hofer A (2013) Modelling and analysis of the non-iterative coupling process for co-simulation. *Math Comput Model Dyn Syst* 19(5):451–70
16. Gear C, Wells D (1984) Multirate linear multistep methods. *BIT Numerical Mathematics*, S 484–502
17. Blochwitz T, Otter M, Akesson J. et al (2012) Functional mockup interface 2.0: The standard for tool independent exchange of simulation models. In: *Proceedings of the 9th International MODELICA Conference*, S 173–184
18. Gipser M (1999) *Systemdynamik und Simulation*. Vieweg+Teuber Verlag
19. Pirtschow G, Berkemer T, Bürger T, Croon N, Korajda B, Röck S (2003) Die simulierte Werkzeugmaschine. In: *Fertigungstechnisches Kolloquium. FtK, Stuttgart*, S 219–246

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.

