

doi.org/10.37544/1436-4980-2026-06-75
Datum der Einreichung: 15.03.2026
Datum der Annahme: 31.03.2026
Datum der Veröffentlichung: 27.07.2026

Von der virtuellen Inbetriebnahme zur effizienten Betriebsabsicherung

Simulationsbasierte Online-Kollisionsvermeidung

A. Heine, L. Klingel, S. Chen, A. Verl

ZUSAMMENFASSUNG Simulationsmodelle der virtuellen Inbetriebnahme dienen dazu, Steuerungssysteme in einer digitalen Umgebung zu testen und Kollisionen frühzeitig zu erkennen. Für eine sichere Kollisionsvermeidung sind aber Modelle mit Online-Anbindung erforderlich, da Offline-Modelle durch laufende Hardware- und Softwareanpassungen ihre Gültigkeit verlieren. Ihre Entwicklung ist aufwendig, jedoch können viele Bestandteile aus bestehenden Offline-Modellen übernommen werden. Diese Arbeit beschreibt ein Vorgehensmodell zur systematischen Überführung und veranschaulicht es an einem Anwendungsfall.

STICHWÖRTER

Simulation, Automatisierung, Produktionstechnik

Simulation-based online collision avoidance – From virtual commissioning to efficient operational assurance

ABSTRACT Simulation models for virtual commissioning are used to test control systems in a digital environment and detect collisions at an early stage. Safe collision avoidance, however, requires models with online connectivity, as offline models lose their validity due to ongoing hardware and software modifications. Their development is complex, but many components can be transferred from existing offline models. This paper describes a methodology for systematic transfer and illustrates it using a conceptual use case.

1 Einleitung

Mit dem Fortschritt der industriellen Automatisierung und der Verbreitung cyber-physischer Systeme werden auch Bewegungssteuerungen nicht nur präziser, sondern auch zunehmend komplexer und hochdynamisch [1]. Diese zunehmende Komplexität von Bewegungsabläufen mit mehrachsigen Kinematiken oder der Kooperation zwischen bewegten Systemen erhöht das Risiko betrieblicher Kollisionen, die zu erheblichen wirtschaftlichen Schäden führen können [2–4].

Im Entwicklungsprozess von Produktionssystemen werden häufig Simulationsmodelle genutzt, um Bewegungsabläufe gegen Kollisionen abzusichern [5]. Dafür kommen 3D-Bewegungssimulationen zum Einsatz. Diese Simulationsmodelle bilden sowohl die kinematischen als auch die geometrischen Eigenschaften eines Produktionssystems ab [5]. Zur virtuellen Absicherung eines Produktionssystems wird eine virtuelle Inbetriebnahme (VIBN) durchgeführt, bei der unter Einbindung der realen Steuerung das Verhalten und die Wechselwirkungen zwischen Software und Hardware und deren Auswirkungen auf das Produktionssystem getestet werden [6, 7]. Bei der VIBN wird letztendlich simulationsbasiert überprüft, ob unter Verwendung der realen Steuerung und Steuerungsprogramme Kollisionen auftreten.

Vorab durchgeführte Simulationen in der VIBN können Kollisionen jedoch nicht vollständig absichern, wenn die Ursachen erst während des Betriebs, also zur Laufzeit, auftreten. Eine Ursache

kann die manuelle Modifikation des Steuerungsprogramms durch das Bedienpersonal sein [8]. Eine weitere Ursache kann die automatische Änderung eines adaptiven Systems sein, das beispielsweise zur Optimierung der Bahnplanung in Echtzeit auf das Steuerungsprogramm zugreift [9, 10].

Um auf derartige Echtzeit-Anpassungen im Steuerungsprogramm und die so entstehenden neuen potenziellen Kollisionen reagieren zu können, ist der Einsatz einer simulationsbasierten Online-Kollisionsvermeidung sinnvoll [11]. Die Konzeption, Implementierung und Parametrierung einer solchen Architektur stellen jedoch einen technisch anspruchsvollen und zeitintensiven Prozess dar. Daher bietet es sich in einem methodisch sinnvollen Vorgehen an, das bereits verfügbare und erprobte Echtzeitsimulationsmodell samt Schnittstellen aus der VIBN zu übernehmen, um Risiken in der Integration und den Entwicklungsaufwand zu reduzieren [12].

Das folgende Kapitel stellt anhand des Stands der Technik dar, wie Echtzeitsimulationsmodelle aus der VIBN in die Betriebsphase überführt werden können und welche Forschungslücke für die Kollisionsvermeidung besteht. Darauf aufbauend wird anschließend ein Konzept vorgestellt, das beschreibt, wie ein derartiges Vorgehen konkret für eine Online-Kollisionsvermeidung umgesetzt werden kann. Die Ausarbeitung wird anhand eines konzeptionellen Anwendungsfalls veranschaulicht, bevor der in diesem Artikel geleistete Beitrag abschließend zusammengefasst wird.

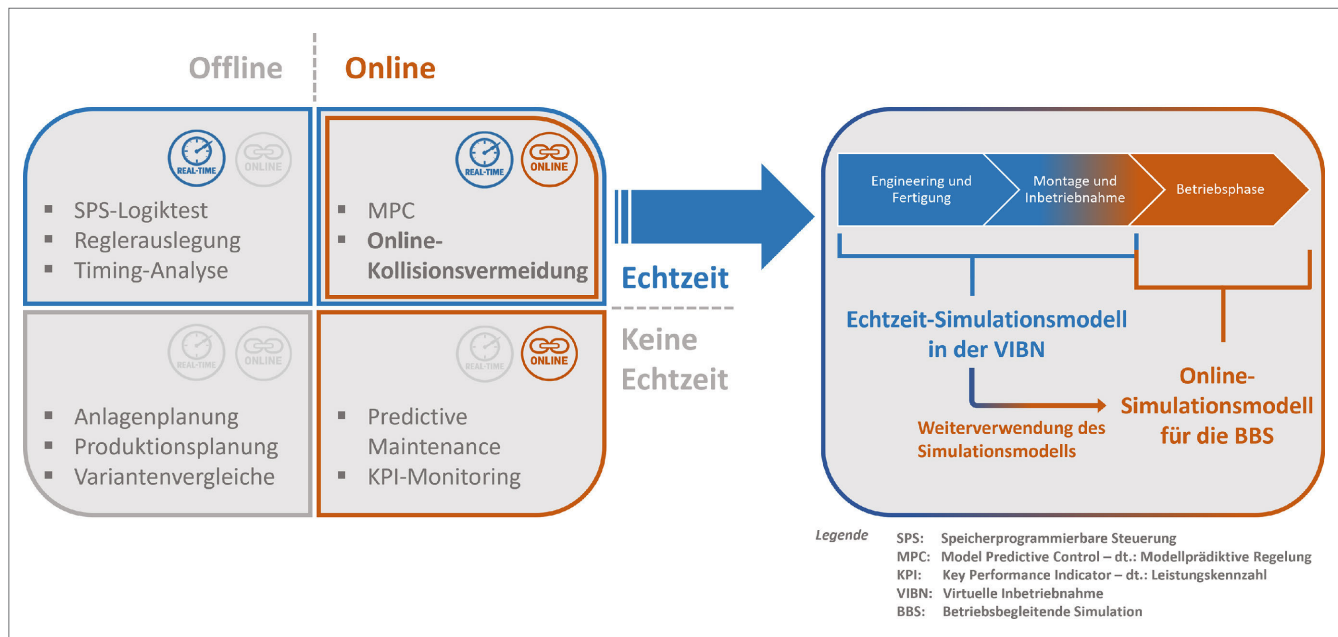


Bild 1 Klassifikation von Simulationsmodellen nach Echtzeitverhalten und Online-Interaktionsfähigkeit nach [13] sowie deren Einsatz in der virtuellen Inbetriebnahme (VIBN) und der betriebsbegleitenden Simulation (BBS); Weiterverwendung von Echtzeit-Simulationsmodellen aus der VIBN als Basis für die BBS. Grafik: ISW

2 Stand der Forschung und Technik

Nach *We et al.* können Simulationsmodelle durch Echtzeitverhalten und Online-Interaktionsfähigkeit differenziert werden [13]. Diese vier Kategorien sind in **Bild 1** dargestellt.

Wenn man Bewegungsabläufe im Steuerungssystem zuverlässig auf Kollisionszustände prüfen möchte, ist ein Simulationsmodell notwendig, das online und im Echtzeittakt der Steuerung arbeitet, um Schnittstellenkomplikationen zu vermeiden, keine Zustände zu überspringen sowie auf Zustandsänderungen reagieren zu können. Im Bereich der industriellen Steuerungs- und Automatisierungstechnik liegt dieser Takt meist im Bereich von wenigen Millisekunden [14]. Echtzeit-Simulationsmodelle kommen in der Betriebsphase bisher seltener zum Einsatz als in der VIBN [15]. Sie können aber auch, wie am Beispiel der Online-Kollisionsvermeidung, als betriebsbegleitende Simulation (BBS) genutzt werden, wenn sie eine Online-Anbindung ermöglichen. Zwischen VIBN und BBS bestehen dabei häufig gemeinsame Komponenten und Schnittstellen [16]. Aus diesen Gründen ist die Weiterverwendung von in der VIBN erstellten Simulationsmodellen in der BBS sinnvoll. Im Folgenden soll der Stand der Forschung zur Weiterverwendung von Simulationsmodellen aus der VIBN im Kontext der Online-Kollisionsvermeidung untersucht und die Forschungslücke identifiziert werden.

Bereits im Jahr 2008 formulierten *Kain et al.* die Idee, Simulationsmodelle aus der VIBN (Offline- und Echtzeit-Modelle) nicht nur zur reinen Verifikations- und Testfunktion zu verwenden, sondern diese Modelle dauerhaft als BBS (online und in Echtzeit) in die Betriebsphase zu überführen [17–19]. Dadurch können Modell und reales System laufend synchronisiert und so können wiederum kontinuierliches Monitoring und adaptive Steuerungsoptimierungen durchgeführt werden. Damit dies funktioniert, fordern *Kain et al.* eine Verbindung von Simulations- und Echtzeitdaten, angepasste Modellkomplexität sowie laufende Validierung und passende Schnittstellen.

Spätere Arbeiten erweiterten diese Idee und entwickelten Verfahren zur automatisierten Modellgenerierung aus digitalen Konstruktionsdaten und funktionalen Mock-ups, wodurch Simulationsmodelle für VIBN und BBS schneller erzeugt und genutzt werden können [20, 21].

Parallel dazu beschäftigten sich mehrere Beiträge mit Architekturfragen und Realisierungen für online und feldnahe Simulationsmodelle in der BBS. Konzepte zur Kopplung von Steuerung und Simulation, Partitionierung und Parallelisierung der Simulation für höhere Rechenperformance sowie Strategien zur Synchronisation zwischen realer Steuerung und digitalem Modell wurden vorgeschlagen und prototypisch demonstriert [22, 23]. Auch *Klingel et al.* stellen einen prototypischen Ansatz vor, Modelle aus der VIBN in eine industrielle Steuerung zu integrieren. Dabei wird eine mögliche Architektur zur Kopplung von Steuerung und Simulation beschrieben und auf deren Parallelisierungsfähigkeit zur Steigerung der Rechenperformance untersucht [24]. Diese Ansätze sind zwar vielversprechend, adressieren jedoch nicht die Kollisionsvermeidung.

Zuletzt wurden konkrete Kollisionsvermeidungen basierend auf Simulationsmodellen der VIBN präsentiert [25]. Dieses Konzept wurde sodann an einer realen Roboterzelle realisiert [26]. Nachteile dieser Vorarbeiten sind zum einen die Verwendung von zu vereinfachten Hüllkörpern (primitive Geometrien wie Kugeln, Quader oder Zylinder) für die Approximation des Simulationsmodells sowie die Verwendung von aktuellen Positionswerten statt einer Prädiktion. Diese Kombination führt zu fehlerhaft detektierten Kollisionen. Aufbauend auf diesen Arbeiten wurde ein Ansatz präsentiert, der um eine Bewegungsprädiktion und ein geometrisch-genauerer Simulationsmodell erweitert wurde [11].

Zusammenfassend verdeutlicht **Bild 2**, dass im aktuellen Stand der Technik trotz der Fortschritte die Mehrzahl der Forschungsbeiträge entweder auf konkrete Anwendungsfälle der Online-Kollisionsvermeidung beschränkt ist oder allgemeine Aspekte der BBS (Architektur, Synchronisation, Modellwiederverwendung)

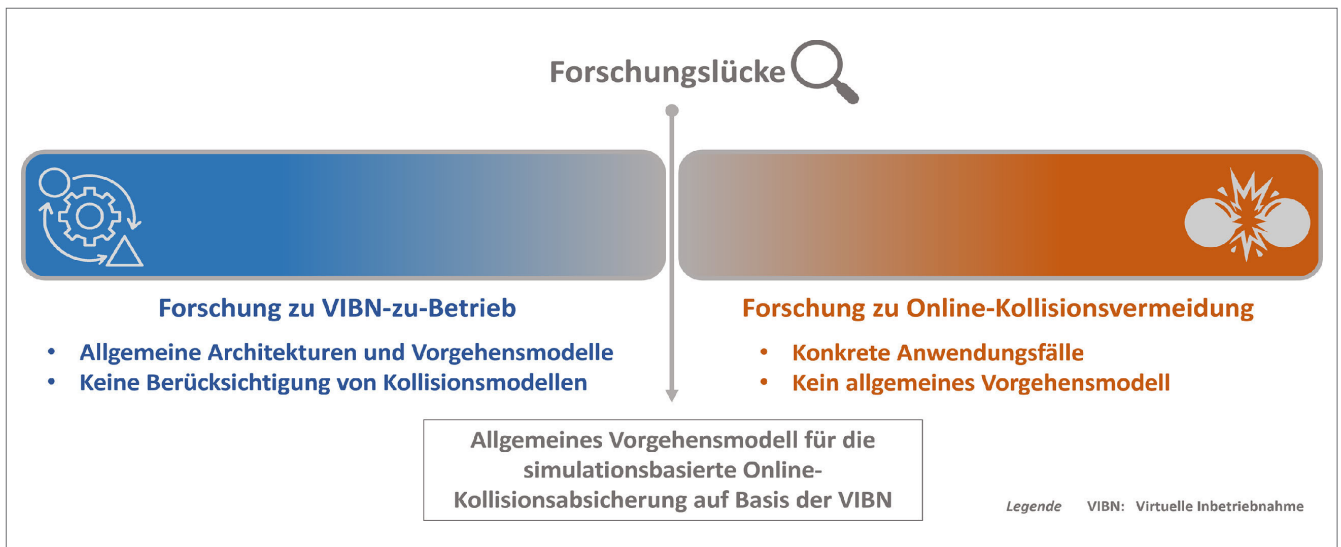


Bild 2 Fehlendes Vorgehensmodell für die simulationsbasierte Online-Kollisionsabsicherung auf Basis der VIBN als identifizierte Forschungslücke.
Grafik: ISW

behandelt, aber nicht systematisch den Aufbau und die Integration einer Kollisionsvermeidung als Bestandteil der VIBN-zu-Betrieb-Kette berücksichtigt.

3 Konzept für die simulationsbasierte Online-Kollisionsvermeidung

Das in dieser Arbeit vorgestellte Konzept ist ein allgemeines Vorgehensmodell zur Überführung von Simulationsmodellen aus der VIBN für die BBS unter Betrachtung der Online-Kollisionsvermeidung. Konkret werden dafür Teilmodelle identifiziert, die sowohl in der VIBN, als auch in der Online-Kollisionsvermeidung verwendet werden. Weiterhin sollen das Verfahren zur Transformation und die damit einhergehende Beschreibung der Schnittstellen der Teilmodelle konzipiert werden.

Als Basis für das Vorgehensmodell dienen die Definitionen der Teilmodelle der VIBN nach VDI 3693 [27]. Als übergeordnete Teilmodelle berechnen E/A (Eingang/Ausgang)- sowie Geräte-modelle in der VIBN basierend auf Sollwerten der Steuerung die Istwerte. Während E/A-Modelle einfache Übertragungsglieder modellieren, dienen Gerätemodelle der Abbildung komplexeren Verhaltens wie physischen Geräten.

Als dritte übergeordnete Teilmodelle bilden Prozessmodelle dagegen das logische und physikalische Verhalten des Systems ab. Sie können weiter untergliedert werden in kontinuierliche Prozesse, diskrete Prozesse und Netzwerkmodelle.

Kontinuierliche Prozesse beschreiben physikalische Größen stetig über die Zeit, während diskrete Prozesse Zustandssprünge abbilden (zum Beispiel im Takt der Steuerung). Netzwerkmodelle fokussieren sich auf die Kommunikationsaspekte vernetzter Komponenten.

Diskrete Modelle lassen sich in vier grundlegende Typen unterteilen. Eine erste Gruppe bilden Zustandsautomaten. Sie beschreiben ein System durch eine endliche Menge klar definierter Zustände sowie durch eindeutig festgelegte Übergänge zwischen diesen Zuständen. Eine zweite Gruppe sind Kinematikmodelle. Diese konzentrieren sich ausschließlich auf geometrische Bewegungsgrößen wie Lage, Orientierung und Geschwindigkeit. Kräfte oder physikalische Einwirkungen werden nicht berücksichtigt.

Zudem gibt es Kinematikmodelle mit 3D-Modell. Sie erweitern die rein geometrische Betrachtung um räumliche Geometrien. Dadurch werden etwa Visualisierungen, Kollisionserkennung oder die Modellierung von Sensoren möglich. Schließlich gehören auch Dynamikmodelle zu den diskreten Modellen. Im Gegensatz zu den zuvor genannten Ansätzen berücksichtigen sie Massen, Trägheiten sowie wirkende Kräfte und Momente. Auf diese Weise ermöglichen sie eine physikalisch korrekte Beschreibung des Bewegungs- und Kraftverhaltens eines Systems.

Alle wiederzuverwendenden Komponenten sind in **Bild 3** orange umrahmt. Weiterhin sind die drei notwendigen Schritte zur Transformation abgebildet, die im Folgenden genauer erläutert werden.

Das Verfahren beginnt mit einer umfassenden Analyse der vorhandenen Teilmodelle des VIBN-Simulationsmodells, um festzulegen, wie die Komponenten in die betriebsbegleitende Simulation übernommen werden können. Anschließend wird aus den geometrischen und kinematischen Informationen das erforderliche Kollisionsmodell abgeleitet und ausgelegt. Abschließend werden die ausgewählten Komponenten in die Simulationsumgebung integriert und angebunden:

3.1 Analyse der Teilmodelle:

Ein zentraler Aspekt bei der Analyse der Teilmodelle ist die Berücksichtigung der Taktzeit der eingesetzten Steuerung. Die Berechnung der Kollision kann, abhängig von Modellkomplexität, Anzahl der beteiligten Objekte und geforderter Genauigkeit, mehrere Millisekunden Rechenzeit beanspruchen [11]. Die verfügbare Zykluszeit der Steuerung begrenzt somit unmittelbar die geometrische Auflösung des Kollisionsmodells. Je kürzer die Taktzeit und je geringer die zur Verfügung stehenden Rechenressourcen, desto stärker müssen Modellierungstiefe und Prüfalgorithmen vereinfacht werden. Umgekehrt ermöglicht eine größere zeitliche Reserve eine detailliertere geometrische Abbildung und komplexere Prädiktionsmechanismen.

Darüber hinaus ist die Offenheit der Steuerungsarchitektur zu bewerten. Insbesondere in Brownfield-Systemen ist der direkte Zugriff auf Steuerungsprogramme oder interne Datenstrukturen

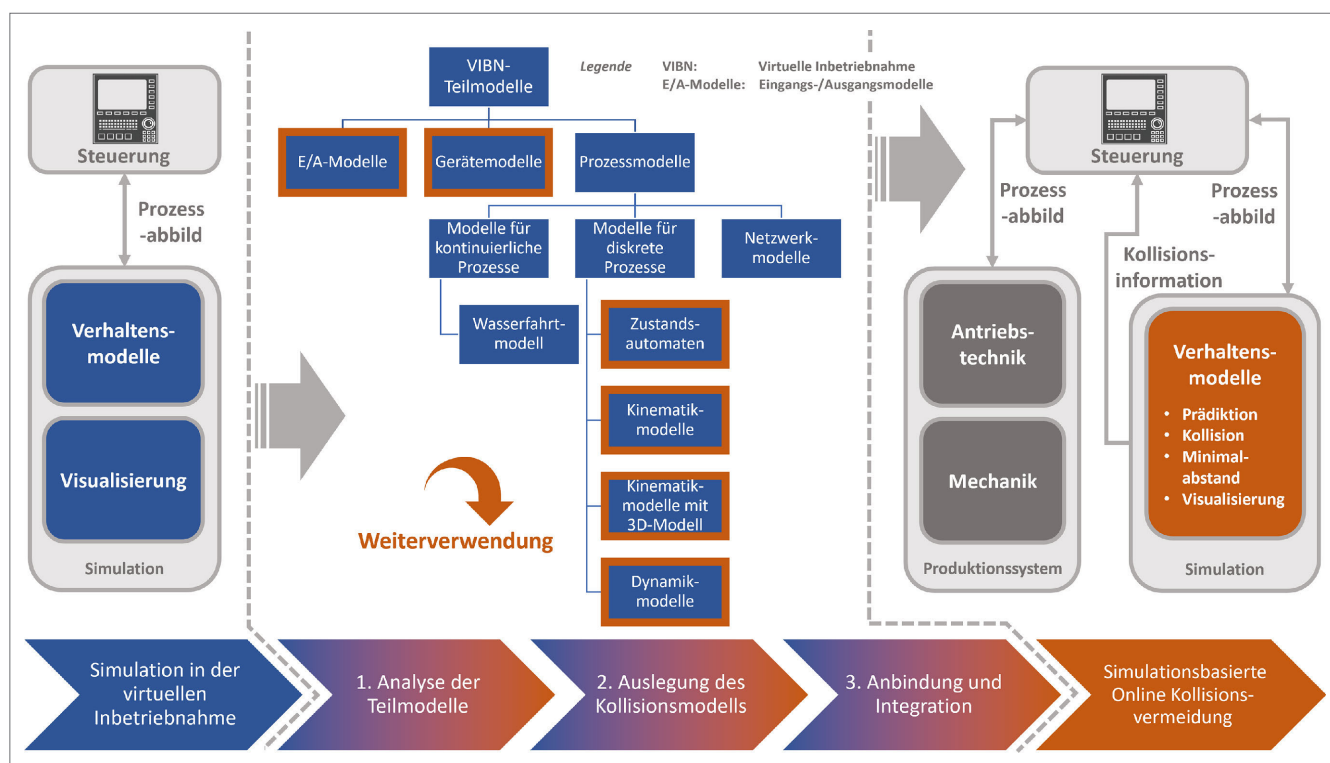


Bild 3 Allgemeines Vorgehensmodell für die simulationsbasierte Online-Kollisionsvermeidung auf Basis von Teilmodellen der virtuellen Inbetriebnahme nach VDI 3693. Grafik: ISW

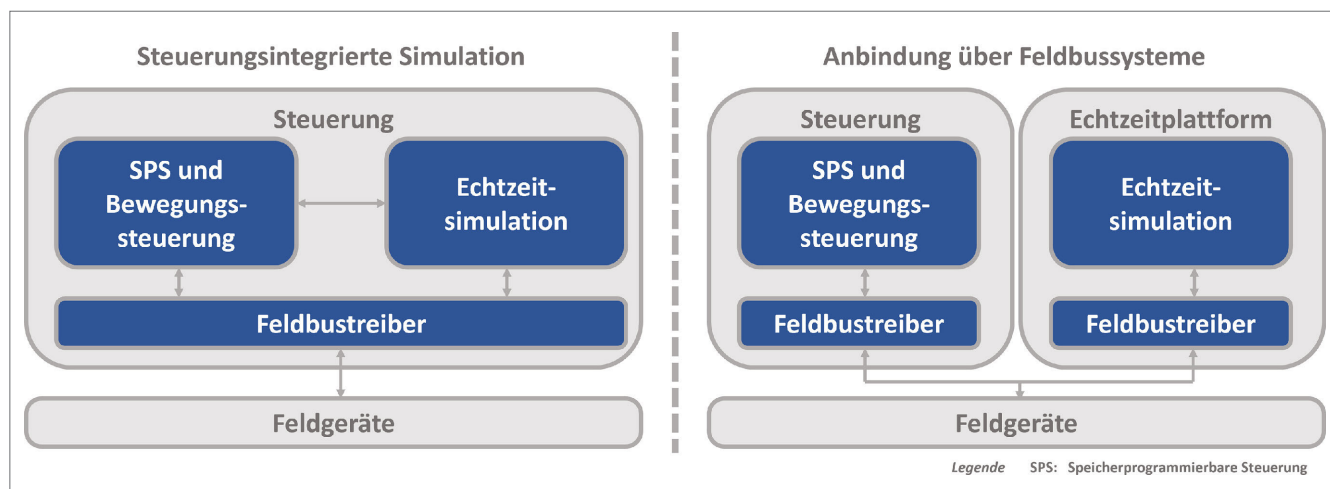


Bild 4 Anbindungsmöglichkeiten zwischen Simulation und Bewegungssteuerung in der Betriebsphase. Grafik: ISW

oft eingeschränkt. Ist die Steuerung offen und stehen ausreichende Rechenkapazitäten zur Verfügung, kann eine grundlegende Kollisionsvermeidung direkt in die Steuerung integriert werden. Dies bietet Vorteile hinsichtlich Latenz und Determinismus, da zusätzliche Kommunikationswege entfallen. Sind die Rechenressourcen begrenzt oder ist eine Integration aus organisatorischen oder technischen Gründen nicht möglich, muss die Kollisionsberechnung auf eine externe Recheneinheit ausgelagert werden. Eine grafische Darstellung dieser beiden Methoden ist **Bild 4** zu entnehmen.

In diesem Fall sind die Echtzeitanforderungen der Gesamtarchitektur sorgfältig zu analysieren. Sowohl die externe Hardware als auch die Kommunikation über den Feldbus müssen deterministisch erfolgen, wobei, abhängig von der sicherheitstechnischen

Einstufung der Anwendung, harte oder zumindest weiche Echtzeitbedingungen einzuhalten sind.

Um Kollisionszustände zuverlässig zu vermeiden und nicht nur zu erkennen, ist es zudem erforderlich, Steuerungsdaten mit einem konstanten zeitlichen Vorlauf, aber im Takt der Steuerung (Lookahead) bereitzustellen. Nur wenn zukünftige Solltrajektorien oder Bewegungsbefehle vor ihrer physischen Ausführung verfügbar sind, kann die Kollisionsberechnung prädiktiv erfolgen und rechtzeitig Gegenmaßnahmen, etwa durch Anpassung der Bewegung oder Einleitung eines sicheren Stopps, eingeleitet werden. Steht ein solcher Lookahead nicht zur Verfügung, sind alternative Prädiktionsverfahren einzusetzen, beispielsweise auf Basis aktueller Bewegungszustände und modellierter Dynamik aus den Kinematik- und Dynamikmodellen. Ergänzend kann im Rahmen

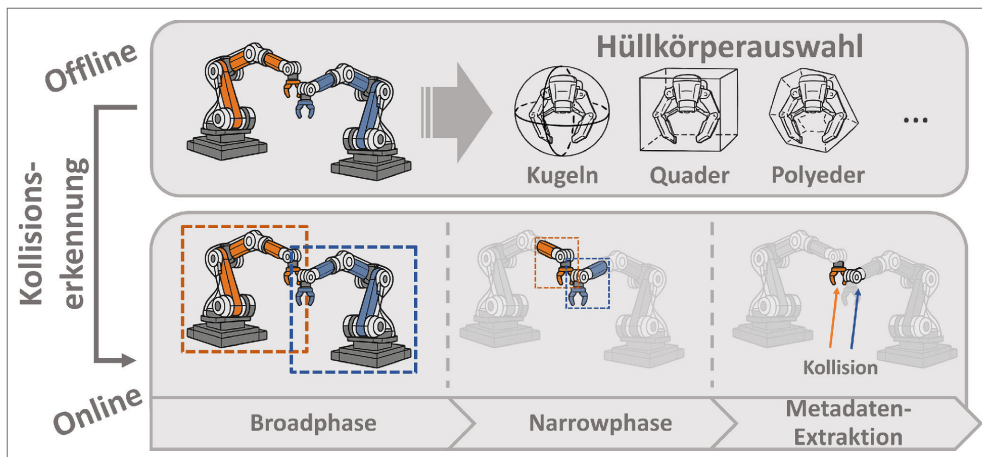


Bild 5 Gesamtprozess der Auslegung und Anwendung des Kollisionsmodells; oben: Offline-Prozess zur Auswahl geeigneter Hüllkörper aus den detaillierten 3D-Geometrien; unten: anschließender Online-Betrieb, in dem in jedem Takt der betriebsbegleitenden Simulation die Kollisionserkennung ausgeführt wird. Grafik: ISW

der späteren Auslegung des Kollisionsmodells mit konservativ vergrößerten Hüllkörpern gearbeitet werden, um geometrische Pufferzonen zu schaffen und Unsicherheiten in der Vorhersage zu kompensieren.

3.2 Auslegung des Kollisionsmodells

Für die Auslegung des Kollisionsmodells sind primär die geometrischen und kinematischen Informationen aus den Kinematik- und 3D-Modellen erforderlich. Die Erkennung von Kollisionen lässt sich in drei aufeinanderfolgende Schritte gliedern: Zunächst werden die Steuerungsbefehle aus den E/A- und Gerätemodellen entlang der in den Kinematikmodellen definierten kinematischen Ketten in konkrete translatorische und rotatorische Posen für jede relevante Geometrie (wie einzelnes Bauteil oder Bauteilverbund) überführt. Jedes Bauteil oder Bauteilverbund wird im Kollisionsmodell durch einen Hüllkörper approximiert. Das bedeutet, dass alle Punkte, Kanten und Flächen des 3D-Modells durch eine primitive Geometrie wie Kugeln, Quader oder Polyeder eingeschlossen wird. Die neue Pose des Bauteils oder des Bauteilverbunds wird direkt auf den Hüllkörper übertragen. Auf Basis der neu berechneten Posen der Hüllkörper erfolgt eine grobe Vorselektion (Broad Phase), in der möglichst viele Hüllkörperpaare frühzeitig ausgeschlossen werden, um den nachfolgenden Rechenaufwand zu reduzieren. In der anschließenden engen Prüfphase (Narrow Phase) werden die verbleibenden Paare mit präzisen Algorithmen auf tatsächliche Überschneidungen untersucht. Bei Feststellung einer Kollision wird die Berechnung abgebrochen und das Kollisionsereignis und möglicherweise relevante Metadaten (betroffene Objekte, Kollisionsstelle, Zeitpunkt) ausgegeben. Dieser Prozess ist in **Bild 5** zu sehen.

Die Auslegung des Kollisionsmodells bietet erhebliche Freiheitsgrade, die systematisch gegen Echtzeit- und Genauigkeitsanforderungen abzuwägen sind. Zu entscheiden ist, welche Bauteile in welche Hüllkörper zusammengefasst werden, wie fein die Auflösung der Hüllkörper sein muss und welche Kombination von Broad-/Narrow-Phase-Algorithmen zum Einsatz kommt. Diese Entscheidungen können vollständig automatisiert durch eine Physik-Engine getroffen werden oder gezielt manuell erfolgen. Eine manuelle Modellierung und Auswahl der Algorithmen ist zwar aufwendiger, ermöglicht aber oft eine signifikante Reduktion des Rechenaufwands bei gleicher oder besserer Genauigkeit als generische Engine-Voreinstellungen [11]. Zusätzlich können informationsreiche Quellen aus Zustandsautomaten und Dynamikmodel-

len eingebunden werden, um die Prüfung weiter zu optimieren. Beispiele hierfür sind Vorabprüfungen häufiger Zustände aus den Zustandsautomaten offline oder ein prädiktives Ausschließen physikalisch unmöglicher Konstellationen anhand von Geschwindigkeit und Trägheit.

3.3 Anbindung und Integration

Im letzten Schritt wird die definierte Kollisionslogik so in das Gesamtsystem eingebettet, dass eine deterministische und echtzeitfähige Interaktion mit der Steuerung gewährleistet ist. Die Integration hat so zu erfolgen, dass die Gesamtzykluszeit, bestehend aus Datenübertragung, Kollisionsberechnung und Rückmeldung an die Steuerung, innerhalb der zuvor definierten Echtzeitgrenze bleibt. Vor allem bei ausgelagerter Kollisionsberechnung müssen Kommunikationslatenzen, Jitter und Synchronisationsmechanismen explizit berücksichtigt und validiert werden.

Funktional ist zu definieren, wie auf eine erkannte oder prädizierte Kollision reagiert wird. Die Bandbreite reicht von adaptiven Bahnkorrekturen über Geschwindigkeitsreduktionen bis hin zur Einleitung eines sicheren Stopps. Ein weiterer Integrationsaspekt betrifft die Parametrierbarkeit und Wartbarkeit des Systems. Änderungen an Geometrien, Bewegungsabläufen oder Prozesszuständen sollten durch einfache Anpassungen im Kollisionsmodell anpassbar sein. Dazu können geeignete Konfigurationsmechanismen im Kollisionsmodell und Monitoringfunktionen vorgesehen werden [11, 28].

Abschließend ist die integrierte Lösung unter realistischen Betriebsbedingungen zu testen und zu validieren. Neben funktionalen Tests sind vor allem Grenzfälle, Kommunikationsstörungen und Lastszenarien zu untersuchen, um die Robustheit der Gesamtlösung sicherzustellen.

4 Anwendungsfall und Ergebnis

Der Anwendungsfall aus **Bild 6** beschreibt eine Industrieroboterzelle mit drei Greifarmrobotern der Firma Stäubli.

Als gemeinsame Arbeitsfläche diente eine Tischplatte mit verschiedenen Hindernissen. Im ersten Schritt des Vorgehensmodells wurde eine Steuerungstaktzeit von 2 ms ermittelt. Aufgrund der kurzen Taktzeit wurde die Umsetzung als steuerungsintegrierte Simulation realisiert. Auf einem Industrie-PC wurden in Echtzeit mittels TwinCAT das CNC-Basisprogramm und die Kollisionsabsicherung berechnet. Diese kommunizierte in Echt-

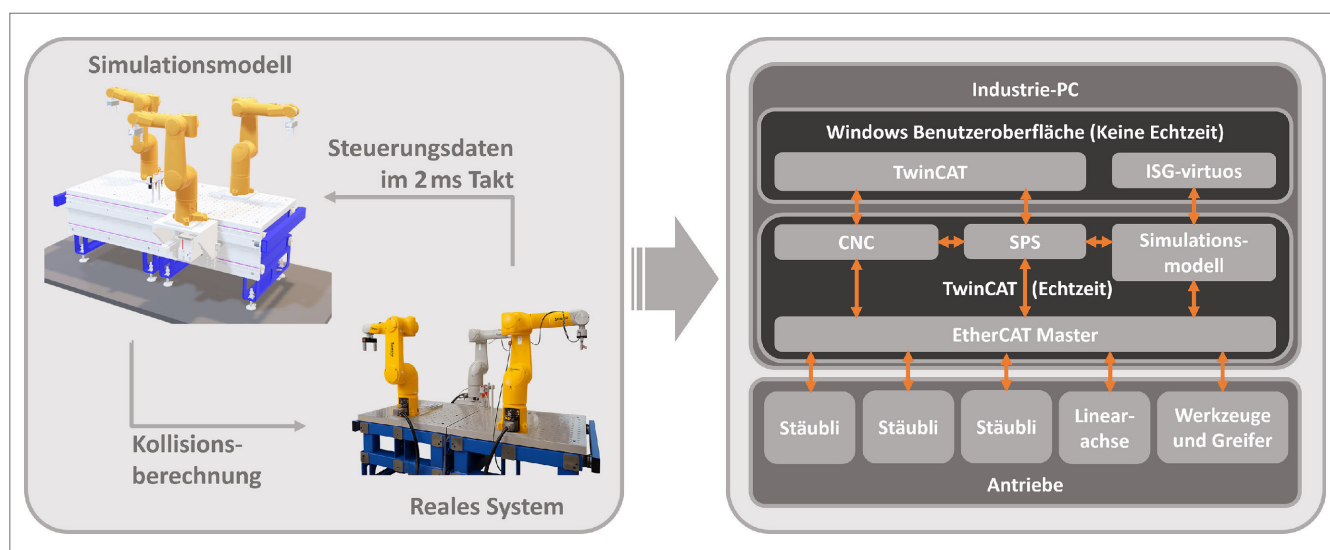


Bild 6 Architektur des Anwendungsfalls einer Industrieroboterzelle mit simulationsbasierter Online-Kollisionsvermeidung unter Anwendung des Vorgehensmodells. Grafik: ISW

zeit mit der Antriebstechnik. Das Simulationsmodell lief auf einem dedizierten Echtzeit-Solver (ISG-virtuos) und kommunizierte über einen EtherCAT-Master mit der Steuerung. Die Steuerungsdaten wurden per Lookahead vorab eingelesen.

Nach Schritt 2 des Vorgehensmodells erfolgte die Auslegung des Kollisionsmodells auf einem Windows-Betriebssystem (keine Echtzeit) mit ISG-virtuos. Die Kollisionserkennung war so ausgelegt, dass in der Broad-Phase mit achsenausgerichteten Quadern und in der Narrow-Phase mit präzisen konvexen Hüllen (kleinste Polyeder) gerechnet wurde. Wenn die Kollision erkannt wurde, leitete die SPS über die CNC einen Notstopp der Antriebstechnik ein. Neben dieser Notfalleinrichtung konnten weitere Informationen wie Abstände zwischen den Bauteilen und Zeitpunkte der Kollisionen extrahiert werden. Das Ergebnis war eine zuverlässige Kollisionserkennung mit zwei Sekunden Prognosezeit, die in den Versuchen alle Kollisionen innerhalb der Echtzeitbedingungen detektierte und verhinderte.

5 Zusammenfassung und Ausblick

Kollisionen durch bewegte Systeme in Produktionsanlagen verursachen erhebliche wirtschaftliche Schäden und stellen zugleich ein ernsthaftes Sicherheitsrisiko dar. Simulationsbasierte Ansätze zur Kollisionsprüfung in der virtuellen Inbetriebnahme haben sich bewährt und bieten einen hohen Nutzen bei der Absicherung von Anlagen vor der Inbetriebnahme. Änderungen während des Betriebs durch Hardware- oder Softwareanpassungen können aber dazu führen, dass offline durchgeführte Absicherungen nicht mehr greifen. Vor diesem Hintergrund rückt die simulationsbasierte Online-Kollisionsvermeidung in den Fokus aktueller Forschung. Bisher fehlen dafür einheitliche Vorgehensmodelle und praktikable Strategien zur Wiederverwendung vorhandener Simulationsmodelle.

Aus diesem Grund wurde ein allgemeines, dreistufiges Vorgehensmodell entwickelt, das die Überführung von Simulationsmodellen aus der VIBN in eine betriebsbegleitende Kollisionsvermeidung systematisiert. Basis sind die Teilmodelle nach VDI 3693. Im ersten Schritt werden Steuerungs- und Kommunikationsanforderungen analysiert. Im zweiten Schritt erfolgt die Auslegung

des Kollisionsmodells auf Basis geometrischer und kinematischer Daten. Der abschließende Integrationsschritt beschreibt die Einbettung in die Gesamtarchitektur sowie die definierten Reaktionsstrategien und stellt die erforderlichen Validierungs- und Sicherheitsprozesse sicher.

Die praktische Demonstration des Modells am Anwendungsfall hat gezeigt, dass die strukturierte Übertragung von VIBN-Modellen in eine onlinefähige Kollisionsvermeidung realisierbar ist und sich spezifische Anforderungen während des Prozesses gezielt berücksichtigen lassen, um zuverlässige, sicherheitsrelevante Maßnahmen zu gewährleisten.

Für zukünftige Untersuchungen stehen mehrere Forschungsfelder offen. Dazu zählen die Entwicklung robuster Prädiktionsverfahren zur Kompensation fehlender Lookahead-Daten, die Automatisierung der Kollisionsmodell-Auslegung und systematische Studien darüber, welche Hüllkörper und Kollisionsalgorithmen in industriellen Einsatzszenarien am effizientesten sind. Zudem sollte untersucht werden, ob und in welchem Umfang maschinelles Lernen die Erkennung oder Vorhersage von Kollisionen beschleunigen kann. Weiterhin könnte die Einbindung von Sensoren und Sensorfusion dynamische Objekte, die zuvor nicht im Kollisionsmodell definiert waren, zur Laufzeit zuverlässig erkennen und in das Kollisionsmodell integrieren. Insgesamt bildet das vorgestellte Vorgehensmodell eine belastbare Grundlage für weiterführende Arbeiten und die schrittweise Industrialisierung simulationsbasierter Online-Kollisionsvermeidung.

FÖRDERHINWEIS

Die Autoren danken der Deutschen Forschungsgemeinschaft (DFG) für die Förderung des Projekts „SimPro“ mit der Projektnummer 545637120, in welchem die vorgestellten Ergebnisse entstanden sind.

Literatur

- [1] Herrera-Vidal, G.; Coronado-Hernández, J. R.; Maheut, J.: Complexity management challenges in the industry 4.0 era: A systematic review in production systems. *Results in Engineering* 26 (2025), #105329
- [2] Brecher, C.; Weck, M.: *Werkzeugmaschinen Fertigungssysteme 3. Mechatronische Systeme, Steuerungstechnik und Automatisierung*. Heidelberg: Springer 2021
- [3] Yang, G.; An, L.; Zhao, C.: Collision/Obstacle Avoidance Coordination of Multi-Robot Systems: A Survey. *Actuators* 14 (2025) 2, #85
- [4] Abele, E.; Korff, D.: Avoidance of collision-caused spindle damages—Challenges, methods and solutions for high dynamic machine tools. *CIRP Annals* 60 (2011) 1, pp. 425–428
- [5] Verein Deutscher Ingenieure: VDI 3633 Blatt 8. Simulation von Logistik-, Materialfluss- und Produktionssystemen. Berlin: Beuth Verlag 2020
- [6] Verein Deutscher Ingenieure: *Virtuelle Inbetriebnahme*. Berlin: Beuth Verlag 2018
- [7] Verl, A.; Röck, S.; Scheifele, C.: *Echtzeitsimulation in der Produktionsautomatisierung*. Heidelberg: Springer 2024
- [8] Schumann, M.; Witt, M.; Klimant, P.: A Real-time Collision Prevention System for Machine Tools. *Procedia CIRP* 7 (2013), pp. 329–334
- [9] Kruger, G. H.; Shih, A. J.; Hattingh, D. G. et al.: Intelligent machine agent architecture for adaptive control optimization of manufacturing processes. *Advanced Engineering Informatics* 25 (2011) 4, pp. 783–796
- [10] Möhring, H.-C.; Wiederkehr, P.; Erkorkmaz, K. et al.: Self-optimizing machining systems. *CIRP Annals* 69 (2020) 2, pp. 740–763
- [11] Klingel, L.; Heine, A.; Acher, S. et al.: Simulation-Based Predictive Real-Time Collision Avoidance for Automated Production Systems. 2023 IEEE 19th International Conference on Automation Science and Engineering (CASE), Auckland, New Zealand, 2023, pp. 1–6
- [12] Ramonat, M.; Deubert, D.; Klingel, L. et al.: Weiterverwendung von Simulationsmodellen aus der VIBN: Anwendungsfälle, Anforderungen und Handlungsempfehlungen für die Betriebsphase. 26. VDI Kongress Automation 2025 – Leitkongress der Mess- und Automatisierungstechnik, 2025
- [13] We, K.-S.; Kim, J.-C.; Lee, C.-G.: A novel simulation framework for supporting real-time cyber-physical interactions. 2011 IEEE International Conference on Service-Oriented Computing and Applications (SOCA), Irvine, CA, USA, 2011, pp. 1–3
- [14] Lechler, A.: *Konzeption einer funktional einheitlichen Applikationsschnittstelle für Ethernet-basierte Bussysteme*. Dissertation, Universität Stuttgart, 2011
- [15] Mourtzis, D.: Simulation in the design and operation of manufacturing systems: state of the art and new trends. *International Journal of Production Research* 58 (2020) 7, pp. 1927–1949
- [16] VDMA: *Leitfaden Virtuelle Inbetriebnahme. Handlungsempfehlungen zum wirtschaftlichen Einstieg*. Stand: 2020. Internet: vdma.eu/c/document_library/get_file?uuid=66118d8d-68fc-55a7-a25b-a90ffe89f1f2&groupId=34570. Zugriff am 15.04.2026
- [17] Kain, S.; Heuschmann, C.; Schiller, F.: Von der Virtuellen Inbetriebnahme zur Betriebsparallelen Simulation: Herausforderungen im Anlagenbetrieb und Nutzenpotentiale der Betriebsparallelen Simulation. *atp magazin* (2008) 08, S. 59–62
- [18] Kain, S.; Dominka, S.; Merz, M. et al.: Reuse of HiL simulation models in the operation phase of production plants. 2009 IEEE International Conference on Industrial Technology – (ICIT), Churchill, Victoria, Australia, 2009, pp. 1–6
- [19] Kain, S.; Schiller, F.; Domink, S.: Methodology for Reusing Real-time HiL Simulation Models in the Commissioning and Operation Phase of Industrial Production Plants. In: Naik, G. R. (Hrsg.): *Intelligent Mechatronics*. InTech 2011, dx.doi.org/10.5772/14673
- [20] Pfeifer, D.; Scheid, J.; Kneifl, J. et al.: An improved development process of production plants using digital twins with extended dynamic behaviour in virtual commissioning and control – Simulation@Operations. *PAMM* 23 (2023) 3, doi.org/10.1002/pamm.202300225
- [21] Härle, C.; Barth, M.; Fay, A.: Assistance system for the automated composition and configuration of a Co-Simulation. The 34th Annual European Simulation and Modelling Conference 2020, Toulouse, France, 2020, pp. 155–162
- [22] Deubert, D.; Schultze, S.; Selig, A. et al.: Synchronized online simulation at field level for virtual sensing of web tension. *Discover Applied Sciences* 7 (2025) 6, doi.org/10.1007/s42452-025-07088-w
- [23] Deubert, D.; Selig, A.; Verl, A.: Real-Time Online Simulation at Field Level in Industrial Automation. In: Holder, D.; Wulle, F.; Lind, J. (Hrsg.): *Advances in Automotive Production Technology – Digital Product Development and Manufacturing*. Cham: Springer Nature Switzerland 2025, pp. 88–103
- [24] Klingel, L.; Kübler, K.; Verl, A.: A Multicore Control System Architecture as an Operating Platform for Industrial Digital Twins. *Procedia CIRP* 118 (2023), pp. 294–299
- [25] Csiszar, A.; Boanta, C.; Hoher, S. et al.: Optimal Design Concept of a Reconfigurable Haptic Device. In: Zaeh, M. F. (Hrsg.): *Enabling Manufacturing Competitiveness and Economic Sustainability*. Cham: Springer International Publishing 2014, pp. 267–272
- [26] Klingel, L.; Verl, A.: Simulationsbasierte Online-Absicherung von CNC-gesteuerten Industrierobotern. In: Brecher, C.; Herfs, W.; Petrovic, O. (Hrsg.): *Algorithmik und KI für moderne Roboterapplikationen – ein Überblick*. Düsseldorf: VDI Verlag 2023, S. 74–82
- [27] Verband Deutscher Ingenieure: VDI/VDE 3693 Blatt 1: *Virtuelle Inbetriebnahme – Modellarten und Begriffe*. Berlin: Beuth Verlag 2025
- [28] Ericson, C.: *Real-Time Collision Detection*. Amsterdam: Elsevier Morgan Kaufmann 2004

Alexander Heine, M.Sc. 
alexander.heine@isw.uni-stuttgart.de

Lars Klingel, M.Sc. 

Shengjian Chen, M.Sc. 

Prof. Dr.-Ing. Alexander Verl 
Universität Stuttgart
Institut für Steuerungstechnik der Werkzeugmaschinen
und Fertigungseinrichtungen (ISW)
Seidenstr. 36, 70174 Stuttgart
www.isw.uni-stuttgart.de

LIZENZ



Dieser Fachaufsatz steht unter der Lizenz Creative Commons
Namensnennung 4.0 International (CC BY 4.0)