

Auslegung von Hüllkörpern und Algorithmen für die virtuelle Inbetriebnahme

Hochauflösende Kollisionserkennung in Echtzeit

L. Klingel, P. Kocsis, A. Heine, C. Scheifele, A. Verl

ZUSAMMENFASSUNG Die virtuelle Inbetriebnahme ist eine etablierte Methode im Maschinen- und Anlagenbau. Um physikalisches Verhalten in einer Echtzeitsimulation für die virtuelle Inbetriebnahme abbilden zu können, ist eine Kollisionserkennung nötig. Die Auslegung von Echtzeit-Kollisionsmodellen ist komplex und es bestehen große Abhängigkeiten zwischen Hüllkörpern und Kollisionsalgorithmen. Dieser Beitrag bietet anwendungsnahe Handlungsempfehlungen bei der Auslegung von Kollisionsmodellen für die virtuelle Inbetriebnahme. Zudem wird eine Realisierung am Beispiel eines industriellen Simulationswerkzeugs für die virtuelle Inbetriebnahme präsentiert und diskutiert.

STICHWÖRTER

Automatisierung, Produktionstechnik, Simulation

High resolution collision detection in real-time - Design of bounding volumes and algorithms for virtual commissioning

ABSTRACT Virtual commissioning is an established method in mechanical and plant engineering. Collision detection is necessary to map physical behavior in a real-time simulation for virtual commissioning. The design of real-time collision models is complex and there are major dependencies between enveloping bodies and collision algorithms. This article provides application-oriented recommendations for the design of collision models for virtual commissioning. In addition, it presents and discusses the example of an industrial simulation train for virtual commissioning.

1 Einleitung

Virtuelle Methoden gewinnen in der Produktionstechnik zunehmend an Bedeutung. Eine mittlerweile etablierte Methode ist die virtuelle Inbetriebnahme (VIBN) [1]. Bei der VIBN wird das Steuerungssystem mit einer Simulation verbunden, um die Steuerungsfunktionen und das Zusammenspiel der Steuerung mit den weiteren Komponenten im Automatisierungssystem virtuell testen zu können. Auf diese Weise kann die Entwicklung beschleunigt und die Validierung vorgezogen werden [2]. Zudem können die Simulationsmodelle im Anschluss für den simulationsgestützten Betrieb wiederverwendet werden [3]. Bei der VIBN steht immer die performante Berechnung der Simulation im Fokus, da die Simulation mit weiteren Systemen kommuniziert. Bei der Hardware-in-the-Loop-Simulation wird die Simulation beispielsweise über den Feldbus mit der realen Steuerung verbunden, woraus Echtzeitanforderungen an die Simulation resultieren [4].

Ein wichtiger Bestandteil einer Simulation für die VIBN ist die Kollisionserkennung. Diese ist elementar für die Auslegung von kollisionsfreien Bewegungen, aber auch ein wichtiger Baustein für die physikbasierte Simulation, wie sie beispielsweise bei der kombinierten Simulation von mehreren Maschinen und der Abbildung des Materialflusses benötigt wird [5].

In der Softwaretechnik existiert eine Vielzahl an Bibliotheken bis hin zu umfassenden Physik-Engines, um ein physikalisches Verhalten oder Kollisionen zu berechnen. Allerdings liegt bei die-

sen der Fokus oft auf einer ansprechenden visuellen Darstellung [6] und weniger auf der genauen und zeitdeterministischen Berechnung. Daher existieren Arbeiten im Bereich der VIBN, die genau diese Lücke und den Einsatz der simulativen Kollisionserkennung für die Produktionstechnik schließen wollen.

Dieser Beitrag stellt zunächst den aktuellen Stand der Technik vor, bevor die Auslegung von Kollisionsmodellen diskutiert wird. Anschließend wird an einem industriellen Simulationswerkzeug für die VIBN aufgezeigt, wie hochauflösende Kollisionserkennung in Echtzeit realisiert werden kann.

2 Stand der Technik

Dieses Kapitel stellt den aktuellen Stand der Technik in der Echtzeit-Kollisionserkennung für die VIBN vor. Dabei führt der erste Abschnitt kurz in die Grundlagen der Kollisionserkennung ein und im zweiten Abschnitt werden aktuelle industrielle Umsetzungen und deren wissenschaftlichen Vorarbeiten vorgestellt.

2.1 Grundlagen

Kollisionsmodelle werden genutzt, um in der Simulation zu überprüfen, ob bewegliche Körper mit anderen beweglichen oder starren Körpern kollidieren. Ein Standardwerk für die Echtzeitkollisionserkennung ist die Arbeit von Ericson [6], auf den sich dieser Abschnitt bezieht.

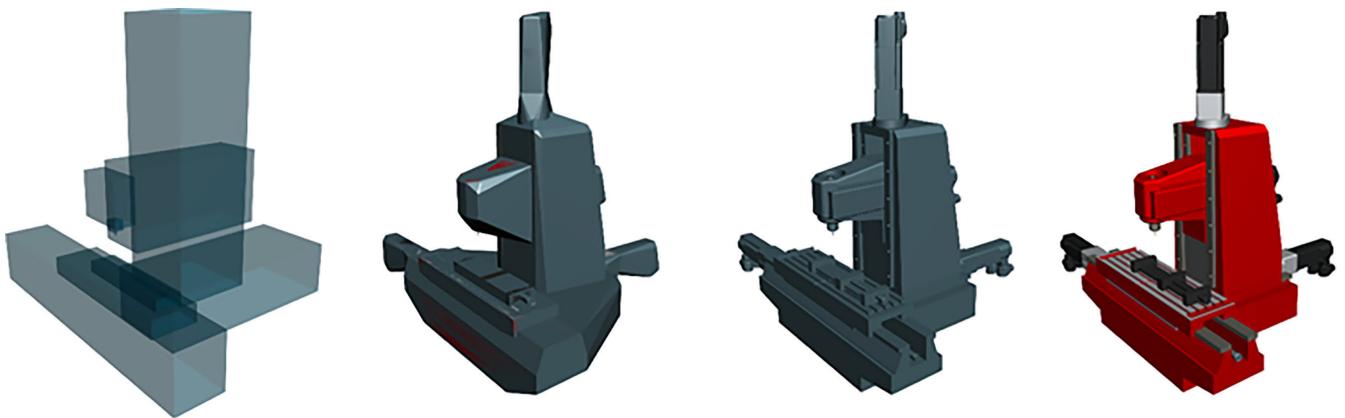


Bild 1. Darstellung verschiedener Hüllkörper am Beispiel einer Fräsmaschine. *Grafik: ISW*

In einer Kollisionserkennung wird in einer Szene zu jedem Takt geprüft, ob sich die modellierten Geometrien überschneiden und daher eine Kollision vorliegt. Bei den zum Einsatz kommenden Geometrien handelt es sich jedoch nicht um detaillierte 3D-Modelle, wie sie in Computerspielen und in der Filmindustrie vorkommen, da sich die Darstellungsform und die Aufbereitung der 3D-Präsentationen zur Visualisierung nicht für eine echtzeitfähige Kollisionsberechnung eignen [6]. Daher werden bei der Echtzeit-Kollisionserkennung sogenannte Hüllkörper verwendet. **Bild 1** zeigt dies am Beispiel einer Fräsmaschine.

Die bewegten Objekte sind jeweils von einem Hüllkörper umgeben. Bei bewegungsgesteuerten Produktionssystemen sind dies in der Regel die Achsen und die zugehörigen geometrischen Komponenten. Mithilfe dieser Hüllkörper kann eine Kollisionserkennung effizient berechnet werden.

Der Nachteil von Hüllkörpern ist, dass sie die reale Geometrie lediglich approximieren. Sie decken also immer einen größeren Bereich ab als die reale Geometrie. Es existieren verschiedene Geometrien, die als Hüllkörper genutzt werden können.

Für die Beschreibung von Kugeln (nicht in Bild 1 dargestellt) wird nur der geometrische Mittelpunkt und der Radius benötigt. Bei Bounding-Boxen (Bild 1 links) wird zwischen achsgerichteten Bounding-Boxen und orientierten Bounding-Boxen unterschieden. Eine weitere Möglichkeit ist die konvexe Hülle (Bild 1, zweite Grafik von links), die, wie der Name suggeriert, das kleinstmögliche konvexe Polygon darstellt, das alle Punkte des Bauteils einschließt.

Neben der Berücksichtigung der Hüllkörper spielt der verwendete Algorithmus eine entscheidende Rolle für die Performance einer Kollisionserkennung. In einer Simulation wird dieser Algorithmus in jedem Zyklus auf alle Kollisionspaare angewendet und geprüft, ob und welche Hüllkörper sich überschneiden. Dabei steigt die Anzahl der notwendigen Kollisionsprüfungen annähernd quadratisch mit der Anzahl der Objekte in der Szene. Verbreitete Algorithmen zur Berechnung der geometrischen Überschneidung von komplexen konvexen Körpern sind Ansätze, die auf dem Paradigma des „Separating Axis Theorem“ (SAT) beruhen sowie der Gilbert-Johnson-Keerthi (GJK)-Algorithmus. Die beiden Algorithmen basieren auf einem gegenteiligen Ansatz, da der GJK eine vorliegende Kollision sucht und das SAT aussagt, dass keine Kollision vorliegt, wenn eine separierende Achse oder im dreidimensionalen eine Ebene gefunden wird.

Um nicht alle möglichen Kollisionspaare in jedem Zyklus überprüfen zu müssen, wird die Kollisionserkennung meist in

zwei Phasen unterteilt, die Broad Phase und die Narrow Phase. In der Broad Phase werden mit sehr performanten aber eher ungenauen Ansätzen Kollisionen zwischen einzelnen Paaren ausgeschlossen, zum Beispiel weil zwei Objekte sehr weit voneinander entfernt sind. In der Narrow Phase wird dann die enge Auswahl genauer berechnet.

2.2 Wissenschaftliche und industrielle Vorarbeiten

Wie bereits angesprochen, können Physik-Engines nicht ohne weiteres für die Echtzeitsimulation verwendet werden. Diese sind nicht auf die zeitdeterministische Berechnung ausgelegt, sondern der Fokus liegt stärker auf der Visualisierung. Dennoch basieren Simulationsumgebungen für die VIBN meist auf Physik-Engines, was daran liegt, dass sie sehr umfangreiche Methoden zur physikbasierten Simulation bieten. Um die Einsetzbarkeit in der Echtzeitsimulation zu gewährleisten, haben verschiedene wissenschaftliche Arbeiten untersucht, wie Physik-Engines parametrierter oder verändert werden müssen, um in der Echtzeitsimulation für die VIBN eingesetzt werden zu können.

Hier sind in erster Linie die Arbeiten von *Lacour* und *Hoher* [5, 7] zu nennen, die mit der Motivation, den Materialfluss zwischen Maschinen in der VIBN simulativ abbilden zu können, in ihren Arbeiten den Einsatz von Physik-Engines untersucht haben. *Hoher* [7] entwickelte dafür eigens ein gekoppeltes Materialflussmodell, das auf einer Mischung aus kinematischem und dynamischem Verhalten aufbaut, um die zeitdeterministische Berechnung zu gewährleisten. In jüngerer Vergangenheit wurden weitere Ansätze für die performante Simulation des Materialflusses durch *Jaensch* und *Kienzlen* vorgestellt [8, 9]. Diese basieren zum einen auf einem Flussmodell und zum anderen auf graph-basierten, neuronalen Netzen [8]. Diese Methoden können die Berechnungszeit zwar deutlich verkürzen, sind allerdings stark auf den Materialfluss spezialisiert und weniger generalisierbar.

Die Materialflusssimulation und generell die Simulation von physikalischem Verhalten basiert neben der Kollisionserkennung zusätzlich auf der Auflösung dieser Kollisionen, beispielsweise um simulativ eine Berührung und ein Abstoßen von mehreren Körpern abzubilden. Die Kollisionsauflösung stellt hinsichtlich des Determinismus der Berechnung oft eine größere Herausforderung als die Kollisionserkennung dar. Somit ist das Ziel für die reine Kollisionserkennung, wie sie beispielsweise bei der Überprüfung von Kollisionen durch Bewegungen von Robotern notwendig ist,

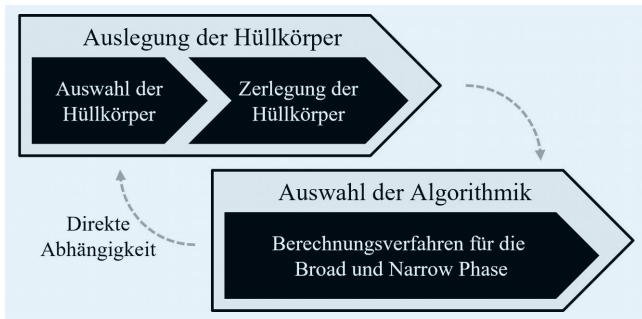


Bild 2. Schritte bei der Auslegung von Kollisionsmodellen für die virtuelle Inbetriebnahme. *Grafik: ISW*

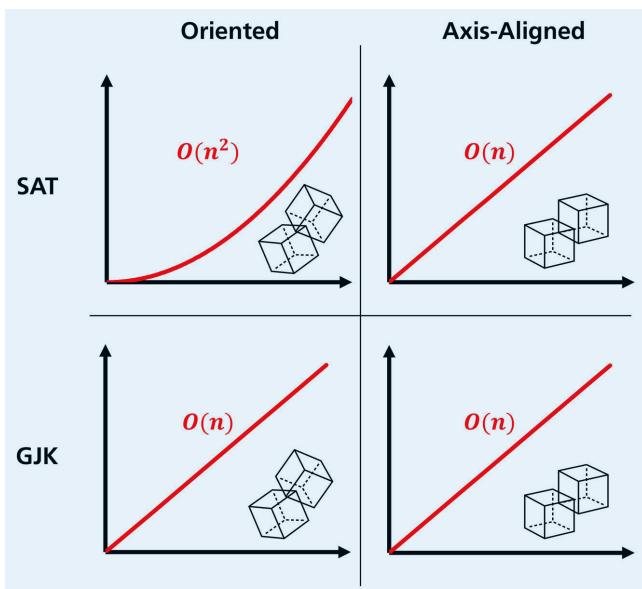


Bild 3. Einordnung der Rechenkomplexitäten der Kollisionsalgorithmen in Bezug auf verschiedene Hüllkörper. *Grafik: ISW*

geometrisch hochauflösender zu agieren als es bei der reinen Physiksimulation, beispielsweise beim Materialfluss, möglich ist.

Bei der Kollisionserkennung, ob als Teil der Physiksimulation oder separat, spielt immer die Abwägung zwischen Genauigkeit und Rechenzeit die zentrale Rolle. Daher wird im nächsten Kapitel diskutiert, welche Design-Entscheidungen bei der Auslegung von Kollisionsmodellen berücksichtigt werden müssen.

3 Auslegung von Kollisionsmodellen für die Echtzeitsimulation

Die Auslegung von Kollisionsmodellen ist elementar für den effizienten Einsatz in der Echtzeitsimulation. Um eine möglichst genaue geometrische Approximation zu erreichen, muss diese im direkten Kontext mit der Algorithmik betrachtet werden, dies ist in **Bild 2** dargestellt.

Bei den Hüllkörpern bilden in den meisten Fällen konvexe Hüllen einen guten Kompromiss aus Genauigkeit und Rechenzeit. Es ist von größter Relevanz, dass die Hüllkörper konvex sind, da die eingangs erwähnten Berechnungsverfahren zur Überprüfung der geometrischen Überschneidung auf der Annahme von konvexen Körpern basieren. Sollte die Genauigkeit einer konvexen Hülle nicht ausreichen, kann eine konvexe Hüllzerlegung in mehrere

Teilkörper erfolgen. Die konvexe Hüllzerlegung kann genutzt werden, um einen konvexen oder auch nicht-konvexen Körper in mehrere Teilhüllen zu unterteilen. Dies führt zu einem höheren Detailgrad, aber auch zu einem höheren Rechenaufwand im Gesamtkontext der Kollisionserkennung. Bei der Einführung einer konvexen Hüllzerlegung ist also wieder darauf zu achten, einen Kompromiss zwischen Detailgrad und Rechenaufwand zu finden. Hierfür können Cluster Evaluationsmethoden zur Hilfe gezogen werden.

Bei der Broad Phase existieren zwei Ansätze: Die räumliche Teilung und die Objektunterteilung, die sogenannten Hüllkörperhierarchien. Die räumliche Teilung ist sinnvoll, wenn keine örtliche Kohärenz zwischen den Körpern besteht. Bei der Kollisionserkennung von zusammenhängenden, bewegten Kinematiken in der Produktion eignet sich eine Hüllkörperhierarchie, da hier eine starke örtliche Kohärenz zwischen den Körpern besteht. Dabei ist zu beachten, welcher Algorithmus und welche Hüllkörper in welcher Phase (Broad und Narrow Phase) sinnvoll kombiniert werden können. In **Bild 3** ist die Komplexität der Algorithmen SAT und GJK im Kontext von achsgerichteten und orientierten Hüllkörpern dargestellt.

Dabei ist hinzuzufügen, dass achsgerichtete Hüllkörper in den meisten Fällen eine schlechtere geometrische Approximation als orientierte Hüllkörper darstellen und eine höhere Rechenzeit für die Generierung des Hüllkörpers benötigen. Der SAT funktioniert besonders effizient bei achsgerichteten Hüllkörpern und eignet sich daher besonders gut für die Broad Phase, während der GJK mit der effizienten Berechnung auch bei orientierten Hüllkörpern für die Narrow Phase geeignet ist.

Neben der Auswahl des geeigneten Algorithmus für die jeweilige Phase, unter Verwendung der passenden Hüllkörper, ist ein relevanter Parameter die Festlegung der Tiefe der Hüllkörperhierarchie. Grundsätzlich ist diese Auslegung sehr komplex, es lässt sich aber festhalten, dass eine Hüllkörperhierarchie ausgewogen sein sollte und nicht zu oberflächlich (wenige Stufen) und nicht zu tief (viele Stufen). Dies ist notwendig, damit die Worst-Case-, Best-Case- und Average-Case-Fälle der Rechenzeiten näher beieinanderliegen. Zudem sollte bei der Auslegung darauf geachtet werden, dass sich für rotatorische Bewegungen, wie etwa bei einem Knickarmroboter, Kugeln als Hüllkörper eignen, da diese invariant gegen Rotation sind. Falls diese den zugrunde liegenden Körper zu ungenau abbilden, eignen sich auch orientierte Hüllkörper- wie die orientierte Bounding-Box. Bei translatorischen Bewegungen, wie bei einer Fräsmaschine oder einem Portal, lohnt sich die Auswahl einer achsgerichteten Bounding-Box.

4 Industrielle Anwendung für die virtuelle Inbetriebnahme

Dieses Kapitel stellt die industrielle Anwendung der bisher beschriebenen Grundlagen und Konzepte am Beispiel des Simulationswerkzeugs „ISG-virtuos“ [10] vor. ISG-virtuos zeichnet sich durch die echtzeitfähige zeitdeterministische Berechnung von Simulationsmodellen aus. Um in diesem Kontext auch physikalisches Verhalten abzubilden, wurde auf den Forschungsvorhaben von *Hoher* aufgebaut und die Physik-Engine „MFPhys“ entwickelt. Die Kollisionserkennung innerhalb MFPhys basiert auf der Physik-Engine „Bullet,“ während die Kollisionsauflösung durch eine Eigenentwicklung realisiert wird. Ursprünglich kann in MFPhys nur mit primitiven Körpern, in diesem Fall Boxen und Zylindern,

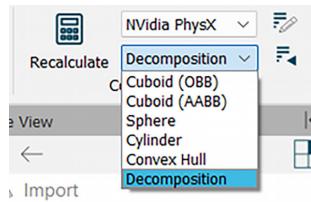


Bild 4. Auswahlmöglichkeiten in „ISG-virtuos“: verschiedene Physik-Engines und Hüllkörper. *Grafik: ISG*

gearbeitet werden, was einige Vorteile für die deterministische Berechnung mit sich bringt, jedoch die Möglichkeiten hinsichtlich der geometrischen Approximationsfähigkeit einschränkt.

Ausgehend von dem Bedarf an hochauflösenden Hüllkörpern und den zusätzlichen Fortschritten und Entwicklungen bei Physik-Engines im Allgemeinen, sind zwei neue Funktionen in ISG-virtuos entstanden. Zum einen ist es möglich, neben MFPhys über die HiFi (High-Fidelity)-Funktion weitere Physik-Engines wie „Nvidia PhysX“ und „Algoryx“ zu verwenden, was auf den technologischen Vorarbeiten der Echtzeit-Co-Simulation von

Scheifele basiert [11]. Zum anderen werden mittlerweile weitere Hüllkörper angeboten, sowohl bei MFPhys als auch bei den HiFi-Physik-Engines. Es ist aber hinzuzufügen, dass die hochauflösenden Hüllkörper im zeitdeterministischen MFPhys ausschließlich für die Kollisionserkennung und nicht für die Kollisionsauflösung eingesetzt werden können. **Bild 4** zeigt die verschiedenen Möglichkeiten, wie sie dem Nutzer in der grafischen Benutzeroberfläche zur Auswahl stehen.

Neben den bereits aufgeführten Auswahlmöglichkeiten ist eine weitere Neuerung dargestellt und zwar die Möglichkeit zur konvexen Zerlegung von Hüllkörpern. Hierfür wird die Bibliothek „V-HACD“ verwendet. Durch die konvexe Zerlegung ist es möglich Geometrien noch genauer abzubilden, als es mit der ursprünglichen konvexen Hülle möglich ist. Da die Kollisionsalgorithmen stets auf der Annahme konvexer Körper basieren, ist es relevant, dass im Rahmen der Zerlegung wiederum konvexe Körper entstehen. In **Bild 5** sind verschiedene Hüllkörper beispielhaft dargestellt.

Oben sind die Bounding-Box (a) und die konvexe Hülle (b) zu sehen, während unten zwei konvexe Zerlegungen mit einer

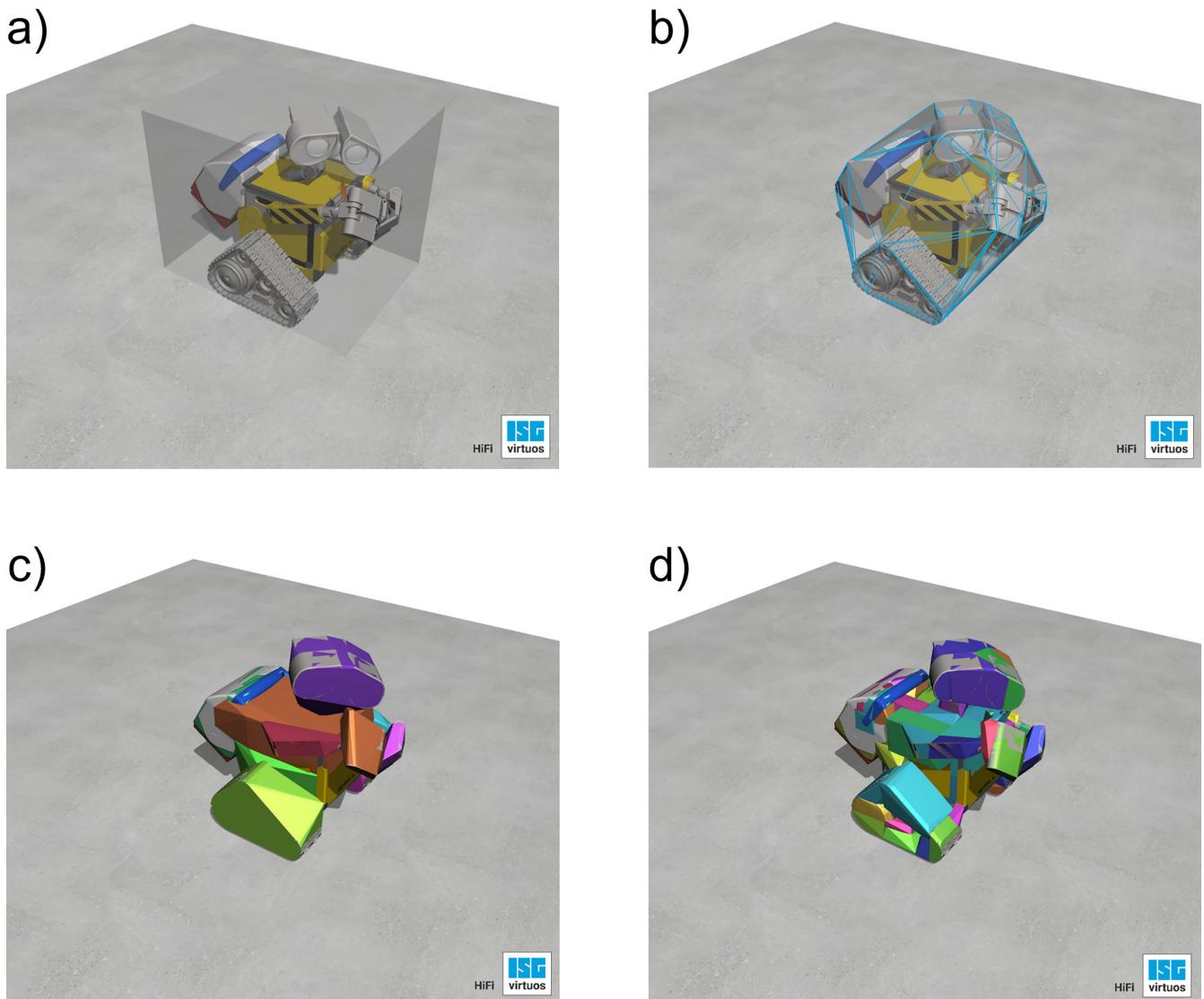


Bild 5. Beispielhafte Darstellung mehrerer Hüllkörper mit und ohne konvexe Zerlegung in ISG-virtuos. *Grafik: ISG*



Bild 6. Darstellung der konvexen Zerlegung am Beispiel einer Teilgeometrie eines Industrieroboters. *Grafik: ISG*

erher gröberen (c) und feineren (d) Auflösung dargestellt sind. In **Bild 6** ist dies zusätzlich an einem Beispiel aus dem produktions-technischen Umfeld, in diesem Fall eines Industrieroboters, dar-gestellt.

Hier wird der stark konkav Teil des Roboters in 64 konvexe Körper unterteilt, da er sonst durch die Abbildung mittels einer konvexen Hülle nur sehr grob approximiert werden könnte. Wie bei der Auslegung der Hüllkörper und Kollisionsalgorithmen an sich, muss auch bei der Auslegung der Zerlegung ein Kompromiss aus geometrischer Abbildungsgenauigkeit und daraus resultieren-der Rechenzeit getroffen werden. Dies muss in weiteren Arbeiten untersucht werden und kann dann beispielsweise als zusätzliche Funktion in ISG-virtuos zur optimalen Zerlegungsauslegung inte-griert werden.

5 Zusammenfassung und Ausblick

Die Kollisionserkennung ist ein wichtiger Bestandteil einer Echtzeitsimulation für die VIBN. Die Auslegung solcher Kollisionsmodelle ist jedoch ein komplexes Unterfangen, da die Auswahl der Hüllkörper und deren geometrische Approximation und die Kollisionsalgorithmen in starker Abhängigkeit stehen, was einen großen Einfluss auf die Gesamtrechenzeit hat.

Im Konzept dieses Beitrags wurden Handlungsempfehlungen zur Auswahl und Kombination dieser Verfahren gegeben. Dabei stellte sich heraus, dass die Kombination von achsenausgerichteten Hüllkörpern und die Berechnung auf Basis des SAT in der Broad Phase und konvexe Hüllen mit dem GJK in der Narrow Phase eine gute Wahl sind. Für produktions-technische Anwendungsfälle wird die Anwendung einer Hüllkörperhierarchie empfohlen.

Im Anschluss wurde die industrielle Realisierung der vorab diskutierten Paradigmen und Ansätze am Beispiel des Simulati-onswerkzeugs ISG-virtuos vorgestellt. Dabei lag der Fokus dieser Realisierung verstärkt auf der konvexen Zerlegung und weniger auf der Optimierung durch Hüllkörperhierarchien. Zudem konnte in dieser Realisierung zwischen verschiedenen Physik-Engines ge-wählt werden.

In zukünftigen Arbeiten müssen in der praktischen Realisie- rung weitere Untersuchungen und Performanceoptimierungen durchgeführt werden. Darüber hinaus ist es notwendig, dass die Auslegung der konvexen Zerlegung in ein Optimierungsproblem überführt wird, um Nutzer bei der Parametrierung automatisiert zu unterstützen.

LITERATUR

- [1] Verein Deutscher Ingenieure: VDI: VDI/VDE 3693 Blatt 1: Virtuelle Inbetriebnahme – Modellarten und Glossar. Berlin: Beuth Verlag 2016
- [2] VDMA: Leitfaden Virtuelle Inbetriebnahme. Handlungsempfehlungen zum wirtschaftlichen Einstieg. Stand: 2020. Internet: www.vdma.org/c/document_library/get_file?uuid=66118d8d-65fc-55a7-a25b-a90ffe89f1f2&groupId=34570. Zugriff am 23.05.2025
- [3] 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, doi.org/10.1109/CASE56687.2023.10260637
- [4] Pritschow, G.; Röck, S.: "Hardware in the Loop" Simulation of Machine Tools. CIRP Annals 53 (2004) 1, pp. 295–298
- [5] Lacour, F-F: Modellbildung für die physikbasierte Virtuelle Inbetrieb-nahme materialflussintensiver Produktionsanlagen. München: Utz V-erlag 2012
- [6] Ericson, C.: Real-Time Collision Detection. Amsterdam: Elsevier 2005
- [7] Hoher, S.: Ein gekoppeltes Materialflussmodell zur durchgängigen Entwicklungsunterstützung von Materialflussteuerungen. Stuttgart: Fraunhofer Verlag 2017
- [8] Jaensch, F.; Herburger, K.; Bobe, E. et al.: Complex Physics with Graph Networks for Industrial Material Flow Simulation. Procedia CIRP 118 (2023), pp. 50–55
- [9] Kienzlen, A.; Verl, A.: Methods for Localization in Multi-Scale Material Flow Simulation for Virtual Commissioning. 2024 4th International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME), Male, Maldives, 2024, pp 01-07, doi.org/10.1109/ICECCME62383.2024.10797016
- [10] ISG Industrielle Steuerungstechnik GmbH: ISG-virtuos. Stand: 2025. Internet: www.isg-stuttgart.de/produkte/softwareprodukte/isg-virtuos. Zugriff am 23.05.2025
- [11] Scheifele, C.: Plattform zur Echtzeit-Co-Simulation für die virtuelle Inbetriebnahme. Dissertation, Universität Stuttgart, 2019

Lars Klingel, M.Sc.
lars.klingel@isw.uni-stuttgart.de
 Tel. +49 711 / 685-82393

Alexander Heine, B.Sc.

Prof. Alexander Verl

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

Peter Kocsis, M.Sc.

Dr.-Ing. Christian Scheifele
 ISG Industrielle Steuerungstechnik GmbH
 Gropiusplatz 10, 70563 Stuttgart
www.isg-stuttgart.de



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