

Echtzeit-Mehrkörpersimulation von Fahrzeug-Radaufhängungen verschiedener Topologien unter Berücksichtigung ihrer Elastokinematik

Real-Time Multibody Simulation of Vehicle Wheel Suspensions of Different Topologies Taking Into Account their Elastokinematics

Jan-Lukas Archut, Burkhard Corves

RWTH Aachen University, Institut für Getriebetechnik, Maschinendynamik und Robotik, 52062 Aachen, Deutschland, {archut, corves}@igmr.rwth-aachen.de

Kurzfassung

In der Fahrzeugentwicklung wird der Einsatz virtueller Entwicklungsmethoden immer wichtiger. Die Erprobung am Fahr-simulator ist ein wichtiges Werkzeug in der Fahrdynamikentwicklung, da die Bewertung der Fahrdynamik einen stark subjektiven Charakter hat. Um die dabei bestehenden Echtzeitanforderungen zu erreichen, sind die verwendeten Fahrdy-namikmodelle meist vereinfacht, besonders hinsichtlich der Elastokinematik der Radaufhängungen. In diesem Beitrag wird die Echtzeit-Mehrkörpersimulation von Radaufhängungen unter Berücksichtigung ihrer Elastokinematik behandelt. Die elastischen Lager werden durch (nicht-)lineare Bushing-Kraftelemente modelliert. Der genutzte Ansatz für die echt-zeitfähige Integration der numerisch steifen Bewegungsgleichung sind linear-implizite Integrationsverfahren. Das Modell erlaubt es, Radaufhängungen mit unterschiedlichen Topologien zu parametrieren und zu simulieren. Es wird gezeigt, wie der Rechenaufwand durch Berücksichtigung der Topologie reduziert werden kann. Mit den beschriebenen Methoden ist es möglich, sowohl eine Doppelquerlenker- als auch eine Mehrlenkerachse in Echtzeit auf einem normalen PC zu simulieren, ohne dass Vereinfachungen der Elastokinematik erforderlich sind.

Abstract

Virtual methods are becoming more and more important in the vehicle development process. Driving simulators are an important tool in vehicle dynamics development as the evaluation of vehicle dynamics through the driver is very subjective. To achieve the required computation times in such real-time applications, vehicle dynamics models are usually simplified, especially with respect to the elastokinematics of the wheel suspensions. This contribution focuses on real-time multibody simulation of wheel suspensions taking into account their elastokinematics. The elastic bearings are modeled by (non)linear bushing force elements. Linear-implicit integration methods allow real-time integration of the numerically stiff equation of motion. With the model, wheel suspensions with different topologies can be parameterized and simulated. It is shown how the computational effort can be reduced by taking the topology into account. Using the described methods, it is possible to simulate both a double wishbone and a multi-link suspension in real time on a normal PC without the need for simplifications of the elastokinematics.

1 Einleitung

Virtuelle Entwicklungsmethoden versprechen eine schnellere, zielgerichtetere und kostengünstigere Produktentwicklung. In der Fahrzeugentwicklung gewinnen sie auch bei der Auslegung der Fahrdynamik an Bedeutung. Die Fahrdynamik eines Kraftfahrzeugs wird wesentlich durch dessen Radaufhängung beeinflusst. Sie führt das Rad relativ zum Fahrzeugaufbau und definiert damit die Radstellung. Die Radstellung wirkt sich wiederum auf die Reifenkräfte aus, die in longitudinaler, lateraler und vertikaler Richtung von der Straße auf das Fahrzeug wirken. In der Fahrzeugtechnik hat sich eine Vielzahl von Radaufhängungskonzepten mit unterschiedlichen Topologien entwickelt [1]. Die Radstellung wird nicht nur durch die Kinematik, sondern auch durch die Elastokinematik dieser Radaufhängungen beeinflusst. Die Körper der Radaufhängungen sind häufig mittels elastischer Lager miteinander verbunden [1]. Daher weisen die Radaufhängungen eine gewisse Nachgiebigkeit unter dem Einfluss äußerer Kräf-

te auf. Es ergeben sich zusätzliche Radstellungsänderungen, die sich der rein kinematischen Führung des Rads überlagern. Die Auslegung der Elastokinematik hat das Ziel, diese zusätzlichen Radstellungsänderungen entweder zu kompensieren oder in gewünschte Radstellungsänderungen umzuwandeln, um die Fahrdynamik des Fahrzeugs zu verbessern [1]. Neben Ansätzen zur objektiven Bewertung der Fahrdynamik durch Kenngrößen spielt die subjektive Wahrnehmung durch die Fahrerin oder den Fahrer eine entscheidende Rolle. Daher ist die Erprobung am Fahr-simulator ein wichtiges Werkzeug in der virtuellen Fahrdy-namikentwicklung. Fahrdynamikmodelle für solche Echt-zeitanwendungen sind meist stark vereinfacht, um die geforderten Rechenzeiten einzuhalten. Die Vereinfachungen betreffen häufig die elastokinematischen Eigenschaften der Radaufhängung. Um die Aussagekraft der Fahr-simulatoren und damit deren Einsetzbarkeit im Entwicklungsprozess zu erweitern, sind daher echtzeitfähige Mehrkörpersimulationsmodelle von Fahrzeug-Radaufhängungen ohne Verein-

DOI: 10.17185/dupublico/77401



fachung der Elastokinematik erstrebenswert.

Elastische Lager in Radaufhängungen weisen hohe Steifigkeiten und Dämpfungen auf. Dadurch ergeben sich hochfrequente und/oder stark gedämpfte Bewegungen der im Verhältnis leichten Körper der Radaufhängung. Im Gegensatz dazu ist die Dynamik der Einfederbewegung des Rads langsam. Dementsprechend weist das System Dynamik auf unterschiedlichen Zeitskalen auf. Aus numerischer Sicht bedeutet dies, dass die Bewegungsgleichung des Mehrkörpersimulationsmodells eine hohe numerische Steifigkeit aufweist. Daher ist die echtzeitfähige Mehrkörpersimulation von Fahrzeug-Radaufhängungen mit elastischen Lagern herausfordernd.

In der Literatur gibt es einige Ansätze für die echtzeitfähige Mehrkörpersimulation von Radaufhängungen mit besonderem Fokus auf ihrer Elastokinematik. Dazu gehören zum Beispiel die Beschreibung der Radaufhängung als „Makro-Gelenk“ [2], die quasi-statische Approximation der schnellen Bewegung der leichten Körper [3] oder eine Steifigkeitsreduktion der elastischen Lager durch Ersatzsteifigkeiten und -dämpfungen [4]. In diesen Ansätzen werden die Dynamik und/oder die Elastokinematik der Radaufhängung modifiziert beziehungsweise vereinfacht dargestellt. Eine Alternative ist die Verwendung linear-impliziter numerischer Integrationsverfahren. Sie können die echtzeitfähige Integration einer numerisch steifen Bewegungsgleichung ermöglichen, ohne dass Modellvereinfachungen notwendig sind. Beispiele solcher Integrationsverfahren sind die linear-implizite Euler-Methode, Rosenbrock-Wanner- beziehungsweise W-Methoden oder die nicht-iterative HHT- α -Methode [5–7].

In diesem Beitrag präsentieren die Autoren ein Mehrkörpersimulationsmodell für die echtzeitfähige Simulation der Dynamik und Elastokinematik von Radaufhängungen. In diesem Modell werden nichtlineare Bushing-Kraftelemente verwendet, um die elastischen Gelenke in den Radaufhängungen zu modellieren. Für die Integration der numerisch steifen Bewegungsgleichung werden linear-implizite Verfahren eingesetzt, in diesem Fall das Verfahren LSRT2 [8]. Mit dem Modell werden die Dynamik und Elastokinematik der Radaufhängungen in Echtzeit berechnet, ohne dass Vereinfachungen der Elastokinematik erforderlich sind. Das Modell ist in Matlab/Simulink implementiert. Mit der Modellimplementierung lassen sich Radaufhängungen mit verschiedenen Topologien simulieren. Dazu muss lediglich die Bedatung des Modells angepasst werden. Rechenzeiten und Genauigkeit des Modells werden anhand einer Doppelquerlenkerachse und einer Mehrlenkerachse demonstriert.

Der Beitrag ist wie folgt strukturiert. In Abschnitt 2 werden die Mehrkörpersysteme der Radaufhängungen zusammen mit ihrer Bewegungsgleichung eingeführt. Danach wird in Abschnitt 3 auf die Herausforderungen bei der echtzeitfähigen Integration steifer Bewegungsgleichungen eingegangen und das hier verwendete linear-implizite Verfahren LSRT2 beschrieben. Anhand der Topologie der Radaufhängungssysteme wird ein Ansatz gezeigt, mit dem die linearen Gleichungssysteme bei der linear-impliziten Integration effizienter gelöst werden können. Für die linear-

implizite Integration muss die Bewegungsgleichung in linearisierter Form vorliegen. Das Datenmodell und der Berechnungsablauf dieser Linearisierung werden in Abschnitt 4 beschrieben. In Abschnitt 5 werden die Ergebnisse des Modells in Form der Rechenzeiten und der Genauigkeit bei einem dynamischen Lastfall vorgestellt. Der Beitrag endet mit der Zusammenfassung in Abschnitt 6.

2 Mehrkörpersysteme von Radaufhängungen mit elastischen Lagern

In diesem Beitrag werden zwei unterschiedliche Radaufhängungssysteme betrachtet, eine Doppelquerlenkerachse und eine Mehrlenkerachse. **Bild 1** zeigt das vereinfachte Topologiediagramm der Doppelquerlenkerachse.

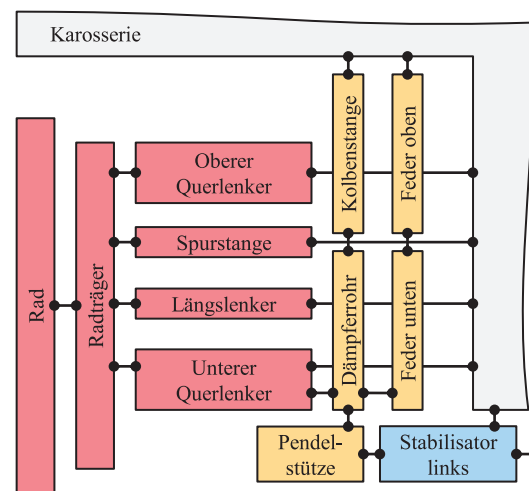


Bild 1 Vereinfachtes Topologiediagramm einer Doppelquerlenkerachse (nur linke Seite dargestellt)

Der Radträger ist über den oberen Querlenker, die Spurstange, den Längslenker und den unteren Querlenker mit der Karosserie verbunden. Die Feder/Dämpfer-Einheit besteht aus vier Körpern, der Kolbenstange, dem Dämpferrohr und der in eine obere und eine untere Hälfte geteilten Feder. Feder und Dämpfer sind am oberen Ende an die Karosserie angebunden und stützen sich am unteren Ende auf dem unteren Querlenker ab. Ein Stabilisator verbindet die linke und die nicht abgebildete rechte Seite der Achse. Er ist in eine linke und eine rechte Stabilisatorhälfte aufgeteilt. Die Stabilisatorhälften sind jeweils über die Pendelstütze mit der Kolbenstange verbunden. Insgesamt besteht das Modell aus 24 starren Körpern. Alle Körper sind ausschließlich durch Kraftelemente miteinander gekoppelt. Die Kopplungen sind in Bild 1 durch die schwarzen Linien repräsentiert, wobei eine Linie auch für mehrere Kraftelemente stehen kann. Die elastischen Lager in der Radaufhängung werden mit Bushing-Kraftelementen modelliert. Sie weisen lineare oder nichtlineare Steifigkeits- und Dämpfungseigenschaften in allen drei translatorischen und rotatorischen Richtungen auf und sind durch entsprechende Kennlinien parametrisiert. Auch das Radlager, das das Rad mit dem Radträger verbindet, ist elastisch durch

ein Bushing-Kraftelement modelliert. Die Radaufhängung weist nur wenige ideale Gelenke auf, zum Beispiel die Schubgelenke zwischen Kolbenstange und Dämpferrohr beziehungsweise zwischen den Federhälften. Diese Gelenke werden ebenfalls durch entsprechend parametrisierte Bushing-Kraftelemente abgebildet. Die Kraftcharakteristiken der Aufbaufeder und des Aufbaudämpfers werden mittels beidseitig gelenkig gelagerter Kraftelemente, im Folgenden P2P-Kraftelemente (*point-to-point*) genannt, modelliert. Weiterhin wird der Federweg durch Bumpstops und Reboundstops begrenzt. Hierbei handelt es sich um P2P-Kraftelemente mit hoher, progressiver Steifigkeit, die erst ab einem bestimmten Federweg in Eingriff kommen. Insgesamt enthält das Modell 55 Kraftelemente.

Die Topologie der Mehrlenkerachse unterscheidet sich von der Doppelquerlenkerachse, siehe **Bild 2**. Radträger und die Lenker der Achse sind nicht direkt an die Karosserie angebunden, sondern an einen Achsträger. Ebenfalls unterscheiden sich die Anbindung des Dämpferrohrs und die Anbindung des Stabilisators über die Pendelstütze. Die Mehrlenkerachse besteht aus insgesamt 27 Körpern und 63 Kraftelementen.

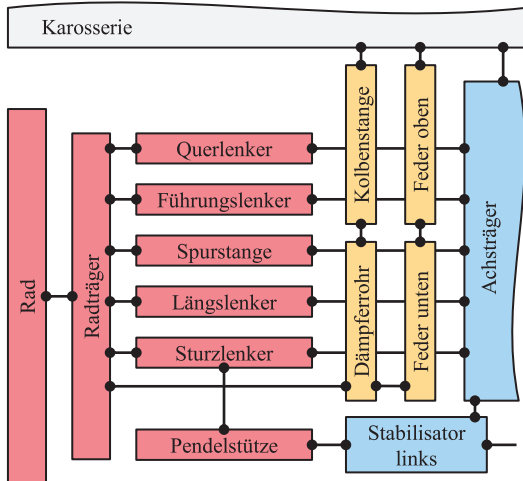


Bild 2 Vereinfachtes Topologiediagramm einer Mehrlenkerachse (nur linke Seite dargestellt)

Wie beschrieben sind alle Körper der Radaufhängungen ausschließlich durch Kraftelemente miteinander gekoppelt. Daher weist das Modell keine kinematischen Zwangsbedingungen auf, sodass die Bewegungsgleichung direkt als eine nichtlineare gewöhnliche Differentialgleichung in der Form

$$\dot{\mathbf{r}} = \mathbf{K}(\mathbf{r})\mathbf{v} \quad (1a)$$

$$\dot{\mathbf{v}} = \mathbf{q}(\mathbf{r}, \mathbf{v}, \mathbf{u}) \quad (1b)$$

formuliert werden kann. Hierbei bezeichnen \mathbf{r} und \mathbf{v} die Vektoren der generalisierten Koordinaten beziehungsweise Geschwindigkeiten. Sie setzen sich aus den Vektoren der einzelnen Körper gemäß

$$\mathbf{r} = (\mathbf{r}_1^T \ \cdots \ \mathbf{r}_m^T)^T, \quad \mathbf{v} = (\mathbf{v}_1^T \ \cdots \ \mathbf{v}_m^T)^T \quad (2)$$

zusammen, wobei m für die Anzahl der Körper steht. Die

Vektoren eines einzelnen Körpers l sind definiert als

$$\mathbf{r}_l = ({}^0x_l \quad {}^0y_l \quad {}^0z_l \quad \alpha_{z,l} \quad \alpha_{y',l} \quad \alpha_{x'',l})^T \quad (3a)$$

$$\mathbf{v}_l = ({}^0v_{x,l} \quad {}^0v_{y,l} \quad {}^0v_{z,l} \quad {}^K\omega_{x,l} \quad {}^K\omega_{y,l} \quad {}^K\omega_{z,l})^T \quad (3b)$$

0x_l , 0y_l , 0z_l and ${}^0v_{x,l}$, ${}^0v_{y,l}$, ${}^0v_{z,l}$ bezeichnen translatorische Verschiebungen und Geschwindigkeiten des Körpers im globalen Koordinatensystem. Die rotatorische Kinematik des Körpers wird durch die Euler-Winkel $\alpha_{z,l}$, $\alpha_{y',l}$ and $\alpha_{x'',l}$ und die Winkelgeschwindigkeiten ${}^K\omega_{x,l}$, ${}^K\omega_{y,l}$, ${}^K\omega_{z,l}$ im Körper-Koordinatensystem beschrieben. Insgesamt weist die Doppelquerlenkerachse somit 144 und die Mehrlenkerachse 162 Freiheitsgrade auf. Die Kinematikmatrix $\mathbf{K}(\mathbf{r})$ beschreibt die lineare Transformation der generalisierten Geschwindigkeiten zur Zeitableitung der generalisierten Koordinaten. Der Vektor der generalisierten Beschleunigungen $\mathbf{q}(\mathbf{r}, \mathbf{v}, \mathbf{u})$ ist die Zeitableitung der generalisierten Geschwindigkeiten und hängt von den generalisierten Koordinaten und Geschwindigkeiten sowie den äußeren Anregungen \mathbf{u} in Form von zum Beispiel Radkräften ab.

Wegen der hohen Steifigkeiten und Dämpfungen der elastischen Lager weist die Bewegungsgleichung eine hohe numerische Steifigkeit auf. Ein Maß für die numerische Steifigkeit einer Differentialgleichung ist das Verhältnis $\max_{i,j}(|\lambda_i|/|\lambda_j|)$ der Eigenwerte λ der linearisierten Differentialgleichung. Bei einem numerisch steifen System sind die Realteile aller Eigenwerte kleiner als Null und das Verhältnis $\max_{i,j}(|\lambda_i|/|\lambda_j|)$ ist sehr viel größer als Eins [9]. Bei den hier betrachteten Radaufhängungssystemen ist dieses Verhältnis größer als 10^5 , die Systeme weisen also eine hohe numerische Steifigkeit auf.

3 Linear-implizite Integration der Bewegungsgleichung

Numerisch steife Mehrkörpersysteme stellen eine Herausforderung in Bezug auf die echtzeitfähige numerische Zeitintegration dar. Bei Echtzeitanwendungen wie zum Beispiel in Fahrsimulatoren müssen Ausgabegrößen zu vorgegebenen Zeitschritten mit konstantem zeitlichen Abstand zur Verfügung gestellt werden. Die Berechnung dieser Ausgabegrößen, sprich in dem hier betrachteten Fall die numerische Zeitintegration, muss daher zuverlässig innerhalb der vorgegebenen Schrittweite erfolgen. Implizite Integrationsverfahren können zwar die stabile numerische Integration numerisch steifer Mehrkörpersysteme ermöglichen, erfordern jedoch die Lösung nichtlinearer Gleichungssysteme. Diese Lösung erfolgt in der Regel iterativ und ist mit hohem Rechenaufwand verbunden. Je nach Zustand und Anregung des Mehrkörpersystems kann der Rechenaufwand zudem während der Simulation schwanken. Daher kann bei impliziten Verfahren nicht ohne Weiteres gewährleistet werden, dass die numerische Integration zuverlässig innerhalb der von der Echtzeitanwendung vorgegebenen Schrittweite abgeschlossen ist. Explizite Integrationsmethoden weisen häufig unzureichende Stabilitätseigenschaften auf, um numerisch steife Mehrkörpersysteme stabil zu integrieren.

Ein Ansatz für die echtzeitfähige Integration numerisch steifer Mehrkörpersysteme sind linear-implizite Integrationsverfahren [5]. Sie basieren auf impliziten Integrationsverfahren. Statt die nichtlineare Bewegungsgleichung direkt zu integrieren, wird hier die Bewegungsgleichung in linearisierter Form integriert. Dadurch werden die nichtlinearen Gleichungssysteme, die bei impliziten Integrationsverfahren gelöst werden müssen, in lineare Gleichungssysteme überführt. Diese lassen sich mit einem geringeren und während der gesamten Simulation konstanten Rechenaufwand lösen. Linear-implizite Integrationsverfahren sind daher sehr gut für Echtzeitanwendungen geeignet.

In der Literatur gibt es eine Vielzahl linear-impliziter Integrationsverfahren. Eine Auswahl wurde von den Autoren des vorliegenden Beitrags für die Simulation der hier betrachteten Doppelquerlenkerachse verglichen [10]. Einen guten Kompromiss zwischen Genauigkeit und Recheneffizienz bietet das Verfahren LSRT2 [8]. Das Verfahren gehört zur Klasse der Rosenbrock-Wanner-Verfahren, einer linear-impliziten Version der Runge-Kutta-Methoden [7]. Es handelt sich um ein Einschrittverfahren mit $s = 2$ Stufen und hoher numerischer Dämpfung [8]. Dieses Verfahren wird auch im vorliegenden Beitrag verwendet. Es basiert auf einer Linearisierung der Bewegungsgleichung (1) gemäß

$$\mathbf{q}(\mathbf{r}, \mathbf{v}, \mathbf{u}) \approx \mathbf{q}_{\text{lin}} + \mathbf{J}_{\text{r,lin}}(\mathbf{r} - \mathbf{r}_{\text{lin}}) + \mathbf{J}_{\text{v,lin}}(\mathbf{v} - \mathbf{v}_{\text{lin}}) + \mathbf{J}_{\text{u,lin}}(\mathbf{u} - \mathbf{u}_{\text{lin}}) \quad (4)$$

mit $\mathbf{q}_{\text{lin}} = \mathbf{q}(\mathbf{r}_{\text{lin}}, \mathbf{v}_{\text{lin}}, \mathbf{u}_{\text{lin}})$ und den Jacobi-Matrizen

$$\mathbf{J}_{\text{r,lin}} = \frac{\partial \mathbf{q}}{\partial \mathbf{r}}, \quad \mathbf{J}_{\text{v,lin}} = \frac{\partial \mathbf{q}}{\partial \mathbf{v}}, \quad \mathbf{J}_{\text{u,lin}} = \frac{\partial \mathbf{q}}{\partial \mathbf{u}} \quad (5)$$

im Linearisierungszustand $\mathbf{r}_{\text{lin}}, \mathbf{v}_{\text{lin}}$ und \mathbf{u}_{lin} .

Ausgehend von den bekannten generalisierten Koordinaten \mathbf{r}^k und Geschwindigkeiten \mathbf{v}^k im Zeitschritt k ergeben sich die Werte im nächsten Zeitschritt $k + 1$ gemäß

$$\mathbf{r}^{k+1} = \mathbf{r}^k + \Delta t_{\text{int}} \sum_{i=1}^s (b_i \Delta \mathbf{r}^{k,i}) \quad (6a)$$

$$\mathbf{v}^{k+1} = \mathbf{v}^k + \Delta t_{\text{int}} \sum_{i=1}^s (b_i \Delta \mathbf{v}^{k,i}) \quad (6b)$$

Dabei ist Δt_{int} die konstante Integrationsschrittweite. Zur Berechnung von $\Delta \mathbf{v}^{k,i}$ ist das lineare Gleichungssystem

$$\begin{aligned} & [\mathbf{E} - \Delta t_{\text{int}} \gamma_{ii} \mathbf{J}_{\text{v,lin}} - (\Delta t_{\text{int}} \gamma_{ii})^2 \mathbf{J}_{\text{r,lin}} \mathbf{K}_{\text{lin}}] \Delta \mathbf{v}^{k,i} = \\ & \mathbf{q}(\tilde{\mathbf{r}}^{k,i}, \tilde{\mathbf{v}}^{k,i}, \tilde{\mathbf{u}}^{k,i}) + \Delta t_{\text{int}} \gamma_{ii} \mathbf{J}_{\text{u,lin}} \Delta \mathbf{u}^{k,i} + \\ & \Delta t_{\text{int}} \begin{pmatrix} \mathbf{J}_{\text{r,lin}} & \mathbf{J}_{\text{v,lin}} & \mathbf{J}_{\text{u,lin}} \end{pmatrix} \sum_{j=1}^{i-1} \left(\gamma_{ij} \begin{pmatrix} \Delta \mathbf{r}^{k,j} \\ \Delta \mathbf{v}^{k,j} \\ \Delta \mathbf{u}^{k,j} \end{pmatrix} \right) + \\ & \Delta t_{\text{int}} \gamma_{ii} \mathbf{J}_{\text{r,lin}} \mathbf{K}_{\text{lin}} \left[\tilde{\mathbf{v}}^{k,i} + \Delta t_{\text{int}} \sum_{j=1}^{i-1} \left(\gamma_{ij} \Delta \mathbf{v}^{k,j} \right) \right] \end{aligned} \quad (7)$$

zu lösen. Die Matrix \mathbf{E} bezeichnet hierbei die Einheitsma-

trix. Anschließend wird $\Delta \mathbf{r}^{k,i}$ mittels

$$\Delta \mathbf{r}^{k,i} = \mathbf{K}_{\text{lin}} \left[\tilde{\mathbf{v}}^{k,i} + \Delta t_{\text{int}} \gamma_{ii} \Delta \mathbf{v}^{k,i} + \Delta t_{\text{int}} \sum_{j=1}^{i-1} \left(\gamma_{ij} \Delta \mathbf{v}^{k,j} \right) \right] \quad (8)$$

berechnet. \mathbf{K}_{lin} ist die Kinematikmatrix im Linearisierungszustand. Die in den obigen Gleichungen verwendeten Stufenvektoren $\tilde{\mathbf{r}}^{k,i}$, $\tilde{\mathbf{v}}^{k,i}$ und $\tilde{\mathbf{u}}^{k,i}$ sind definiert als

$$\tilde{\mathbf{r}}^{k,i} = \mathbf{r}^k + \Delta t_{\text{int}} \sum_{j=1}^{i-1} (a_{ij} \Delta \mathbf{r}^{k,j}) \quad (9a)$$

$$\tilde{\mathbf{v}}^{k,i} = \mathbf{v}^k + \Delta t_{\text{int}} \sum_{j=1}^{i-1} (a_{ij} \Delta \mathbf{v}^{k,j}) \quad (9b)$$

$$\tilde{\mathbf{u}}^{k,i} = \mathbf{u}(t^k + c_i \Delta t_{\text{int}}). \quad (9c)$$

Die Parameter s , b_i , c_i , a_{ij} and γ_{ij} der Methode sind in **Tabelle 1** aufgelistet.

Tabelle 1 Parameter der Methode LSRT2 [8]

s	b_1	b_2	c_1	c_2	a_{21}	$\gamma_{11} = \gamma_{22}$	γ_{21}
2	0	1	0	$\frac{1}{2}$	$\frac{1}{2}$	$1 - \frac{\sqrt{2}}{2}$	$-(1 - \frac{\sqrt{2}}{2})$

In jeder Stufe $i = 1 \dots s$ eines Integrationsschritts sind lineare Gleichungssysteme der Form

$$\mathbf{A} \mathbf{x} = \mathbf{b} \quad (10)$$

mit der Koeffizientenmatrix

$$\mathbf{A} = [\mathbf{E} - \Delta t_{\text{int}} \gamma_{ii} \mathbf{J}_{\text{v,lin}} - (\Delta t_{\text{int}} \gamma_{ii})^2 \mathbf{J}_{\text{r,lin}} \mathbf{K}_{\text{lin}}] \quad (11)$$

zu lösen (vergleiche (7)). Die Lösung erfolgt mittels LR-Zerlegung und anschließendem Vorwärts-/Rückwärtseinsetzen. Wegen $\gamma_{11} = \gamma_{22}$ ist die Koeffizientenmatrix in beiden Stufen gleich. Außerdem ändert sich die Koeffizientenmatrix nur, wenn die Linearisierung (4) und damit die Jacobi-Matrizen $\mathbf{J}_{\text{r,lin}}$ und $\mathbf{J}_{\text{v,lin}}$ sowie die Kinematikmatrix \mathbf{K}_{lin} neu berechnet werden. Nur dann ist es auch notwendig, die LR-Zerlegung neu zu berechnen.

Aus (11) lässt sich schließen, dass sich die Besetzungsstruktur der Koeffizientenmatrix aus der Besetzungsstruktur der Jacobi-Matrizen $\mathbf{J}_{\text{r,lin}}$ und $\mathbf{J}_{\text{v,lin}}$ und der Kinematikmatrix \mathbf{K}_{lin} ergibt. Sie ist daher eng mit der Topologie des betrachteten Radaufhängungssystems verbunden. **Bild 3** zeigt die Besetzungsstruktur für die Doppelquerlenkerachse (siehe Topologiediagramm in Bild 1) und die Mehrlenkerachse (siehe Topologiediagramm in Bild 2). Es sei angemerkt, dass die Besetzungsstruktur für die gesamte Achse, die Topologiediagramme jeweils nur für die halbe Achse dargestellt sind. Anhand von Bild 3 lässt sich erkennen, dass die Koeffizientenmatrix als Blockmatrix mit den farblich markierten Blöcken aufgefasst werden kann. Die farblichen Markierungen der Blöcke auf der Diagonalen entsprechen den eingefärbten Körpern in den Topologiediagrammen (Bilder 1 und 2).

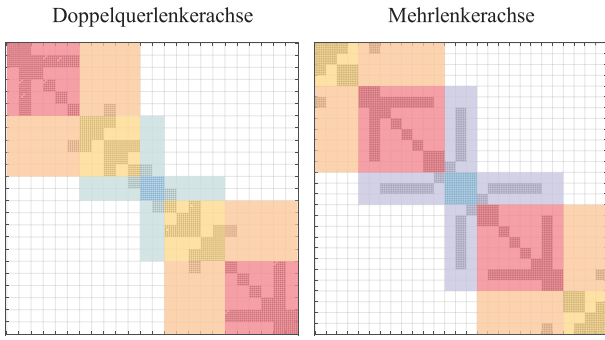


Bild 3 Besetzungsstruktur der Koeffizientenmatrix (11) für die Doppelquerlenkerachse und die Mehrlenkerachse

In Blau sind die Körper eingefärbt, die die beiden Hälften der Achse verbinden. Dies sind der Stabilisator und bei der Mehrlenkerachse zusätzlich der Achsträger. In Rot ist das Subsystem Rad, Radträger und Achslenker und in Gelb die Feder-Dämpfer-Einheit eingefärbt. Die Nichtnullelemente außerhalb der Diagonalen resultieren aus den Kopplungen dieser Subsysteme. Bei der Doppelquerlenkerachse ist der Stabilisator auf beiden Seiten mit den Feder-Dämpfer-Einheiten gekoppelt. Rad, Radträger und Lenker sind ebenfalls mit der Feder-Dämpfer-Einheit, jedoch nicht mit dem Stabilisator gekoppelt. Dementsprechend ist die Blockmatrix nur auf der Diagonalen und den Nebendiagonalen besetzt. Ähnliches gilt für die Mehrlenkerachse.

Diese Besetzungsstruktur kann ausgenutzt werden, um das Gleichungssystem effizienter zu lösen. Dazu wird das lineare Gleichungssystem mit der Koeffizientenmatrix als Blockmatrix aufgeschrieben:

$$\begin{pmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{A}_{21} & \mathbf{A}_{22} & \mathbf{A}_{23} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{32} & \mathbf{A}_{33} & \mathbf{A}_{34} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_{43} & \mathbf{A}_{44} & \mathbf{A}_{44} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{A}_{54} & \mathbf{A}_{55} \end{pmatrix} \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \mathbf{x}_4 \\ \mathbf{x}_5 \end{pmatrix} = \begin{pmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \mathbf{b}_3 \\ \mathbf{b}_4 \\ \mathbf{b}_5 \end{pmatrix} \quad (12)$$

Das Gleichungssystem wird nun in mehrere Gleichungssysteme aufgeteilt, die nacheinander gelöst werden. Zunächst wird das Gleichungssystem

$$\tilde{\mathbf{A}}_{33} \mathbf{x}_3 = \tilde{\mathbf{b}}_3 \quad (13)$$

mit den Koeffizientenmatrizen

$$\tilde{\mathbf{A}}_{33} = \mathbf{A}_{33} - \mathbf{A}_{32} \tilde{\mathbf{A}}_{22}^{-1} \mathbf{A}_{23} - \mathbf{A}_{34} \tilde{\mathbf{A}}_{44}^{-1} \mathbf{A}_{43} \quad (14a)$$

$$\tilde{\mathbf{A}}_{22} = \mathbf{A}_{22} - \mathbf{A}_{21} \mathbf{A}_{11}^{-1} \mathbf{A}_{12} \quad (14b)$$

$$\tilde{\mathbf{A}}_{44} = \mathbf{A}_{44} - \mathbf{A}_{45} \mathbf{A}_{55}^{-1} \mathbf{A}_{54} \quad (14c)$$

und den Vektoren auf der rechten Seite

$$\tilde{\mathbf{b}}_3 = \mathbf{b}_3 - \mathbf{A}_{32} \tilde{\mathbf{A}}_{22}^{-1} \tilde{\mathbf{b}}_2 - \mathbf{A}_{34} \tilde{\mathbf{A}}_{44}^{-1} \tilde{\mathbf{b}}_4 \quad (15a)$$

$$\tilde{\mathbf{b}}_2 = \mathbf{b}_2 - \mathbf{A}_{21} \mathbf{A}_{11}^{-1} \mathbf{b}_1 \quad (15b)$$

$$\tilde{\mathbf{b}}_4 = \mathbf{b}_4 - \mathbf{A}_{45} \mathbf{A}_{55}^{-1} \mathbf{b}_5 \quad (15c)$$

gelöst. Anschließend werden die Gleichungssysteme

$$\tilde{\mathbf{A}}_{22} \mathbf{x}_2 = \tilde{\mathbf{b}}_2 - \mathbf{A}_{23} \mathbf{x}_3 \quad (16a)$$

$$\tilde{\mathbf{A}}_{44} \mathbf{x}_4 = \tilde{\mathbf{b}}_4 - \mathbf{A}_{43} \mathbf{x}_3 \quad (16b)$$

und

$$\mathbf{A}_{11} \mathbf{x}_1 = \mathbf{b}_1 - \mathbf{A}_{12} \mathbf{x}_2 \quad (17a)$$

$$\mathbf{A}_{55} \mathbf{x}_5 = \mathbf{b}_5 - \mathbf{A}_{54} \mathbf{x}_4 \quad (17b)$$

gelöst. Voraussetzung ist hierbei, dass die Matrizen \mathbf{A}_{ij} und $\tilde{\mathbf{A}}_{ij}$ invertierbar sind. Die Rechenzeiterparnis bei diesem Vorgehen ergibt sich dadurch, dass statt eines großen linearen Gleichungssystems mehrere kleine Gleichungssysteme gelöst werden. Dies ist mit einem geringeren Aufwand für die LR-Zerlegung und das Vorwärts-/Rückwärtseinsetzen verbunden. Der Rechenaufwand reduziert sich weiter dadurch, dass bei den Matrizenmultiplikationen mit den Nebendiagonalmatrizen \mathbf{A}_{ij} in den Gleichungen (14-17) nur die Nichtnullelemente der Nebendiagonalmatrizen berücksichtigt werden müssen. Es sei angemerkt, dass die Aufteilung der Koeffizientenmatrix nach Bild 3 nicht zwangsläufig die recheneffizienteste Aufteilung ist. Sie ist jedoch entsprechend der Topologie der Achsen sehr anschaulich.

4 Datenmodell und Linearisierung der Bewegungsgleichung

Zur Anwendung des zuvor dargestellten Integrationsverfahrens müssen die generalisierten Beschleunigungen \mathbf{q}_{lin} , die Jacobi-Matrizen $\mathbf{J}_{r,lin}$, $\mathbf{J}_{v,lin}$, $\mathbf{J}_{u,lin}$ und die Kinematikmatrix \mathbf{K}_{lin} berechnet werden. In diesem Abschnitt werden das dahinterliegende Datenmodell und die Berechnung dieser Größen beschrieben. **Bild 4** zeigt das Datenmodell zur Parametrierung eines Radaufhängungssystems.

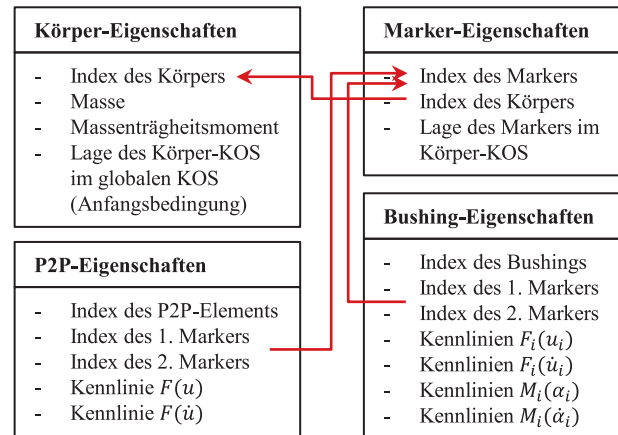


Bild 4 Vereinfachtes Datenmodell zur Parametrierung eines Radaufhängungssystems

In dem Datenmodell sind alle Modellelemente mit ihren Eigenschaften hinterlegt. Jedes Modellelement erhält neben den numerischen Parametern einen eindeutigen Index zur Identifikation. Über diese Indizes wird die Topologie der Radaufhängung im Datenmodell abgebildet. Jeder Marker ist ein lokales Koordinatensystem auf einem Körper, siehe **Bild 5**, und eindeutig diesem Körper zugeordnet. Der Index dieses Körpers ist dementsprechend in den Marker-Eigenschaften abgelegt. An den Markern sind die Kraftelemente angebunden, sodass einem Kraftelement genau zwei Marker zugeordnet werden. Die Indizes dieser

Marker sind daher in den Bushing- und P2P-Eigenschaften abgelegt. Um ein Radaufhängungssystem zu parametrieren, werden alle Elemente mit Indizes versehen und die Referenzierungen sowie die numerischen Parameter der Modellelemente wie beschrieben im Datenmodell hinterlegt. Auf diese Weise lassen sich beliebige Topologien zusammensetzen.

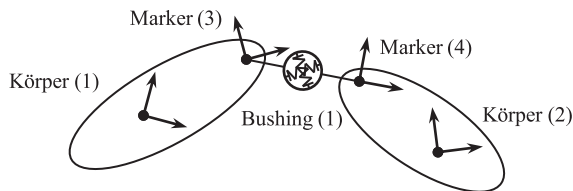


Bild 5 Kopplung zweier Körper durch ein Kraftelement

Die Eigenschaften aller Körper, Marker, Bushing- beziehungsweise P2P-Elemente aus Bild 4 werden in Tabellen gelistet. Diese Tabellen sind Grundlage zur Linearisierung der Bewegungsgleichung, das heißt der Berechnung der generalisierten Beschleunigungen, der Jacobi-Matrizen und der Kinematikmatrix. Der zugehörige Berechnungsablauf ist in **Bild 6** dargestellt.

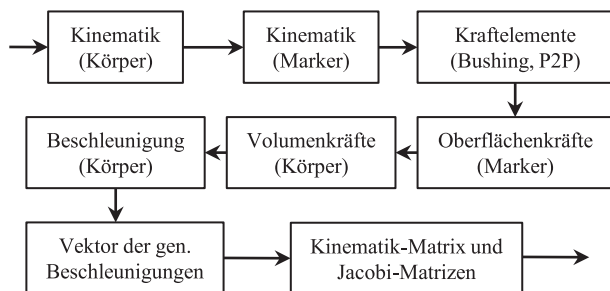


Bild 6 Berechnungsablauf zur Linearisierung der Bewegungsgleichung

Zunächst wird die Kinematik aller Körper berechnet. Die einzelnen Körper sind durch Kraftelemente gekoppelt, jedoch nicht kinematisch miteinander verbunden. Die Kinematik der Körper wird daher getrennt voneinander auf Basis der entsprechenden generalisierten Koordinaten und Geschwindigkeiten berechnet. Die Ergebnisse der Berechnung werden in einer Tabelle gelistet, in der jeder Körper einen Eintrag, versehen mit seinem Index, erhält. Danach erfolgt die Berechnung der Marker-Kinematik. Notwendige Größen der Körper-Kinematik werden entsprechend der Körper-Indizes aus der Tabelle der Körper-Kinematik ausgelesen. Auch die Ergebnisse der Marker-Kinematik werden in einer entsprechenden Tabelle gelistet. Nach gleicher Logik erfolgt die Berechnung der Kraftelemente, der Oberflächen- und der Volumenkräfte. Anschließend werden, für jeden Körper getrennt, die Newton/Euler-Gleichungen aufgestellt und die Beschleunigungen der Körper berechnet. Diese werden dann im globalen Vektor der generalisierten Beschleunigungen \mathbf{q}_{lin} zusammengefasst. Die Bestimmung der Jacobi-Matrizen $\mathbf{J}_{r,lin}$, $\mathbf{J}_{v,lin}$ und $\mathbf{J}_{u,lin}$ erfolgt analytisch. In jedem Berechnungsblock in Bild 6 werden dazu alle benötigten partiellen Ableitungen der jeweiligen Ausgangsgrößen berechnet und mit aus-

gegeben. Diese werden im letzten Berechnungsblock zusammengefasst und in die entsprechende Stelle der Jacobi-Matrizen geschrieben. Ebenfalls wird im letzten Berechnungsblock die Kinematikmatrix des Gesamtsystems aufgestellt.

Dieser Berechnungsablauf und dessen Implementierung kann für beliebige Topologien verwendet werden. Auch die Rechenschritte zur Integration der Bewegungsgleichung (siehe Abschnitt 3) sind unabhängig von der Topologie, sofern auf die topologiespezifische Aufteilung und Lösung des linearen Gleichungssystems gemäß Bild 3 verzichtet wird. Um mit dem Modell unterschiedliche Topologien zu simulieren, muss daher lediglich die Parametrierung nach Bild 4 angepasst werden.

5 Ergebnisse

In diesem Abschnitt werden die mit dem Modell erzielten Ergebnisse in Form der Rechenzeiten und der Genauigkeit bei einem dynamischen Lastfall vorgestellt.

5.1 Rechenzeiten

Zur Bewertung der Rechenzeiten wird der sogenannte Echtzeitfaktor verwendet. Er ist das Verhältnis zwischen der benötigten Rechenzeit und der eingestellten Dauer einer Simulation, das heißt ein Wert unter Eins bedeutet Echtzeitfähigkeit. Zur Bestimmung der Echtzeitfaktoren wird das Simulink-Modell mit dem Simulink Coder kompiliert und C-Code generiert [11]. Um geringere Rechenzeiten zu erzielen, wird dabei optional die Intel oneAPI Bibliothek eingebunden [12]. Sie enthält Implementierungen der Programmibliotheken BLAS (Basic Linear Algebra Subprograms) und LAPACK (Linear Algebra Package). Beim Kompilierungsprozess werden Lineare-Algebra-Berechnungen ab einer bestimmten Größe durch BLAS- bzw. LAPACK-Operationen ersetzt, die weniger Berechnungszeit benötigen. Die Ausführung des generierten C-Codes erfolgt auf einem PC mit einer Intel Core i7-3770 CPU mit 3,4GHz. Als Integrationsschrittweite wird ein Wert von $\Delta t_{int} = 1,0\text{ms}$ eingestellt, die Linearisierung erfolgt in jedem Zeitschritt. Zwar ist es mit den Gleichungen aus Abschnitt 3 möglich, die Linearisierung erst nach einigen Integrationsschritten neu zu berechnen, doch in vorherigen Untersuchungen hat sich herausgestellt, dass eine Integrationsschrittweite von $\Delta t_{int} = 1,0\text{ms}$ mit Linearisierung in jedem Zeitschritt der beste Kompromiss aus Genauigkeit und Echtzeitfähigkeit ist [10]. **Tabelle 2** zeigt die erzielten Echtzeitfaktoren.

Tabelle 2 Erzielte Echtzeitfaktoren

	Doppelquerlenker- achse	Mehrlenker- achse
Ohne Ausnutzung der Besetzungsstruktur von (11):		
ohne BLAS/LAPACK	0,82	1,08
mit BLAS/LAPACK	0,71	0,91
Mit Ausnutzung der Besetzungsstruktur von (11):		
ohne BLAS/LAPACK	0,69	0,86
mit BLAS/LAPACK	0,59	0,73

Es werden die Ergebnisse ohne und mit Einbindung der Programmbibliotheken BLAS und LAPACK sowie ohne und mit Ausnutzung der Besetzungsstruktur der Koeffizientenmatrix (11) bei der Lösung des linearen Gleichungssystems (10) aufgelistet. Durch die Verwendung der Programmbibliotheken BLAS und LAPACK werden die Rechenzeiten um bis zu 15% reduziert. Damit ist es möglich, für beide Radaufhängungen Echtzeitfaktoren von unter Eins zu erzielen. Wird außerdem die Besetzungsstruktur der Koeffizientenmatrix (11) ausgenutzt, werden die Echtzeitfaktoren um zusätzlich bis zu 20% reduziert. Damit ergeben sich Echtzeitfaktoren von 0,59 beziehungsweise 0,73, die deutlich unter Eins liegen.

5.2 Simulation eines dynamischen Lastfalls

Um die Genauigkeit der Integrationsmethode zu zeigen, wird mit beiden Radaufhängungen ein dynamischer Lastfall simuliert. Hierbei handelt es sich um einen Kraftsprung der longitudinalen Radkraft gemäß

$$\mathbf{F}_x(t) = \begin{cases} 0 \text{ N}, & t < 5,0 \text{ s} \\ -2500 \text{ N}, & t \geq 5,0 \text{ s} \end{cases} \quad (18)$$

wobei die positive Koordinatenrichtung in Fahrtrichtung zeigt. Die nach oben gerichtete vertikale Radkraft beträgt konstant $\mathbf{F}_z(t) = 5000 \text{ N}$. Die Radkräfte greifen an beiden Rädern am Radmittelpunkt an.

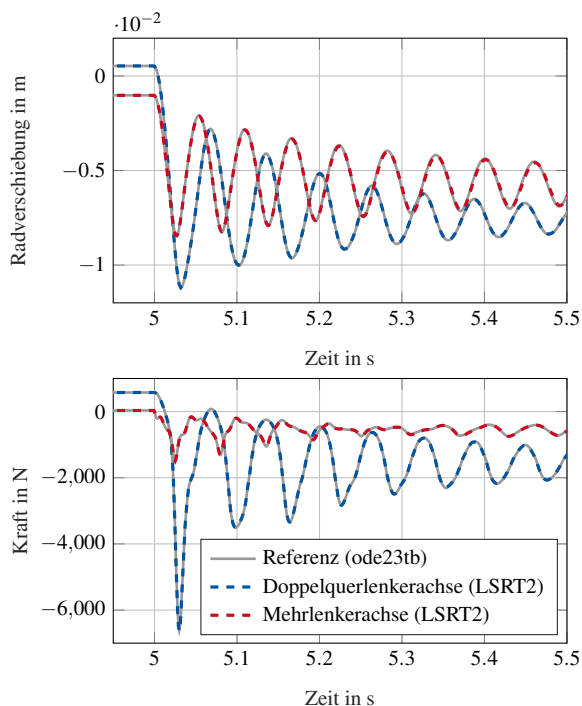


Bild 7 Longitudinale Radverschiebung und Kraft im Bushing zwischen Längslenker und Karosserie beziehungsweise Achsträger (linke Achsseite)

Die Integration der Bewegungsgleichung erfolgt wie bei der Bestimmung der Echtzeitfaktoren mit dem LSRT2-Verfahren bei einer Integrationsschrittweite von $\Delta t_{\text{int}} = 1,0 \text{ ms}$ und Linearisierung in jedem Zeitschritt. Außerdem werden Simulationen mit dem Matlab/Simulink Sol-

ver ode23tb durchgeführt [13]. Dabei handelt es sich um einen impliziten Solver für numerisch steife Systeme. Die Simulationen mit diesem Solver benötigten Rechenzeiten, die weit weg von Echtzeitfähigkeit sind, dienen aber als Referenz zur Bewertung der Genauigkeit.

Bild 7 zeigt die Simulationsergebnisse für die Doppelquerlenker- und die Mehrlenkerachse. Es werden die Verschiebung des Radmittelpunkts sowie die Kraft im Bushing zwischen Längslenker und Karosserie beziehungsweise Achsträger in longitudinaler Richtung dargestellt. Nach dem Kraftsprung ergibt sich eine schwach gedämpfte Schwingung der Radverschiebung und der Kraft. Entsprechend der Elastokinematik der Radaufhängungen pendelt sich die Radverschiebung auf einen Wert von einigen Millimetern ein.

In **Tabelle 3** stehen die normierten mittleren Fehlerquadrate der mit dem LSRT2-Verfahren erzielten Ergebnisse in Bezug auf die Referenzergebnisse. Die Werte beziehen sich auf den in Bild 7 dargestellten Zeitbereich und sind auf den Mittelwert der Referenzergebnisse normiert.

Tabelle 3 Auf den Mittelwert normierte mittlere Fehlerquadrate für den in Bild 7 dargestellten Zeitbereich

	Doppelquerlenkerachse	Mehrlenkerachse
Radverschiebung	0,00243	0,01286
Kraft	0,00751	0,01545

Der Vergleich mit der Referenz zeigt, dass die Integration mit dem LSRT2-Verfahren eine hohe Genauigkeit aufweist. Die Fehlerwerte betragen bei der Doppelquerlenkerachse weniger als 1% und bei der Mehrlenkerachse etwas mehr als 1%. In Bild 7 liegen die Abweichungen im Rahmen der Linienstärke.

6 Zusammenfassung

Der vorliegende Beitrag präsentiert die echtzeitfähige Simulation numerisch steifer Mehrkörpersysteme am Beispiel von Fahrzeug-Radaufhängungen. Die hohe numerische Steifigkeit resultiert hier aus elastischen Lagern, die zur elastokinematischen Auslegung der Radaufhängungen eingesetzt werden. Der Ansatz für die echtzeitfähige Integration der numerisch steifen Bewegungsgleichung liegt in der Verwendung linear-impliziter Integrationsverfahren, in diesem Fall des Verfahrens LSRT2. Diese Verfahren überführen die nichtlinearen Gleichungssysteme, die bei impliziten Integrationsverfahren gelöst werden müssen, in lineare Gleichungssysteme. Es wird gezeigt, wie die Topologie der Radaufhängung in der Lösung dieser linearen Gleichungssysteme berücksichtigt werden kann, um den Rechenaufwand zu reduzieren. Mit dem dahinterliegenden Datenmodell lassen sich Radaufhängungen mit beliebigen Topologien aufbauen. Auch das Berechnungsmodell der Linearisierung erlaubt es, die für die Integration benötigten Größen für beliebige Topologien zu berechnen. Voraussetzung dafür ist, dass die Körper durch elastische Lager miteinander verbunden sind. Die Anwendbarkeit des Ansatzes wird anhand zweier Radaufhängungen mit un-

terschiedlicher Topologie, einer Doppelquerlenker- und einer Mehrlenkerachse, nachgewiesen. Mit Echtzeitfaktoren von 0,59 für die Doppelquerlenkerachse beziehungsweise 0,73 für die Mehrlenkerachse lassen sich beide Radaufhängungen in Echtzeit simulieren. Die Simulationsergebnisse beim hier untersuchten dynamischen Lastfall weisen eine hohe Genauigkeit auf.

Neben elastischen Lagern beeinflussen auch elastische Bauteilverformungen die Elastokinematik von Fahrzeug-Radaufhängungen. Die weiteren Untersuchungen werden sich daher mit der Einbindung flexibler Körper in das vorliegende Modell befassen.

7 Literatur

- [1] Matschinsky, W.: *Radführungen der Straßenfahrzeuge: Kinematik, Elasto-Kinematik und Konstruktion*. Berlin, Heidelberg: Springer-Verlag, 2007, ISBN 978-3-540-71197-1.
- [2] Eichberger, A.; Rulka, W.: *Process Save Reduction by Macro Joint Approach: The Key to Real Time and Efficient Vehicle Simulation*. Vehicle System Dynamics, 41 (2004) 5, S. 401-413, DOI 10.1080/00423110412331300354.
- [3] Burgermeister, B.; Arnold, M.; Eichberger, A.: *Smooth velocity approximation for constrained systems in real-time simulation*. Multibody System Dynamics, 26 (2011) 1, S. 1-14, DOI 10.1007/s11044-011-9243-1.
- [4] Kracht, F. E.: *Modellbildung und Simulation der Dynamik und Elastokinematik von Radaufhängungen für Echtzeitanwendungen*. Dissertation, University of Duisburg-Essen, Germany, 2020, DOI 10.17185/DUEPUBLICO/71445.
- [5] Arnold, M.; Burgermeister, B.; Eichberger, A.: *Linearly implicit time integration methods in real-time applications: DAEs and stiff ODEs*. Multibody System Dynamics, 17 (2007) , S. 99-117, DOI 10.1007/s11044-007-9036-8.
- [6] Kim, M.; Song, H.; Kim, S.-S.: *A non-iterative implicit integration method using a HHT- α integrator for real-time analysis of multibody systems*. Journal of Mechanical Science and Technology, 33 (2019) 3, S. 1087-1096, DOI 10.1007/s12206-019-0208-2.
- [7] Rang, J.: *Improved traditional Rosenbrock-Wanner methods for stiff ODEs and DAEs*. Journal of Computational and Applied Mathematics, 286 (2015) , pp. 128-144, DOI 10.1016/j.cam.2015.03.010.
- [8] Bursi, O. S.; Gonzalez-Buelga, A.; Vulcan, L.; Neild, S. A.; Wagg, D. J.: *Novel coupling Rosenbrock-based algorithms for real-time dynamic substructure testing*. Earthquake Engineering & Structural Dynamics, 37 (2008) 3, S. 339-360, DOI 10.1002/eqe.757.
- [9] Dahmen, W.; Reusken, A.: *Numerik für Ingenieure und Naturwissenschaftler*. Berlin, Heidelberg: Springer-Verlag, 2008, ISBN 978-3-540-76493-9.
- [10] Archut, J.-L.; Corves, B.: *Comparison of Integration Methods for Real-Time Capable Multibody Simulation of an Elastokinematic Wheel Suspension System*. In: Joint International Conference on Multibody System Dynamics 2022. 16.-20. Oktober 2022 New Delhi.
- [11] MathWorks: *Simulink Coder Documentation*. <https://de.mathworks.com/help/rtw/>
- [12] Intel: *Accelerate Fast Math with Intel oneAPI Math Kernel Library*. <https://software.intel.com/oneapi/onemkl>
- [13] MathWorks: *Solve stiff differential equations - trapezoidal rule + backward differentiation formula - MATLAB ode23tb*. <https://de.mathworks.com/help/matlab/ref/ode23tb.html>

DuEPublico

Duisburg-Essen Publications online

UNIVERSITÄT
DUISBURG
ESSEN

Offen im Denken

ub | universitäts
bibliothek

In: Neunte IFToMM D-A-CH Konferenz 2023

Dieser Text wird via DuEPublico, dem Dokumenten- und Publikationsserver der Universität Duisburg-Essen, zur Verfügung gestellt. Die hier veröffentlichte Version der E-Publikation kann von einer eventuell ebenfalls veröffentlichten Verlagsversion abweichen.

DOI: 10.17185/duepublico/77401

URN: urn:nbn:de:hbz:465-20230314-173845-3



Dieses Werk kann unter einer Creative Commons Namensnennung 4.0 Lizenz (CC BY 4.0) genutzt werden.