

Fahrdynamiksimulation unter Verwendung einer Game-Engine

Tobias Neumeier

Zusammenfassung

Die vorliegende Arbeit behandelt die Modellierung eines Fahrzeuges, dessen Fahrdynamik mithilfe einer Game-Engine dargestellt wird. Das Modell beschreibt vor allem die grundlegende Längs- und Querdynamik während Fahrmanövern auf ebenem Untergrund und der schiefen Ebene. Zunächst wurden dafür die Grundlagen der Fahrdynamik analysiert. Darunter die Freiheitsgrade, Kinematik und Kraftverteilung des Fahrzeuges. Unter Zuhilfenahme der gewonnenen Erkenntnisse erfolgte anschließend die Modellierung und Simulation eines linearen Fahrzeugmodells. Für die Modellierung wurde die Game-Engine Unity3D und Microsoft Visual Studios verwendet.

I. EINLEITUNG

Zu Beginn gilt es die Motivation für diese Arbeit zu erläutern und anschließend die dazugehörige Forschungsfrage zu stellen. Die Simulation von Fahrzeugkomponenten und ganzen Fahrzeugmodellen ist für die Automobilentwicklung seit Jahren unerlässlich. Die virtuelle Modellierung bietet viele Vorteile im Vergleich mit Prototypen und Sensorik. Funktionen werden schneller validiert, Material- und Arbeitskosten reduziert, was bei sinkenden Produktlebenszyklen als erforderlich gilt. Außerdem können virtuelle Modelle ohne großen Aufwand in zukünftige Derivate integriert werden. Das Fahrzeugmodell lässt sich in verschiedene Teilbereiche untergliedern. Einer der wichtigsten Bereiche ist die Modellierung der Fahrdynamik, welche das Verhalten eines Fahrzeugs unter Einwirkung von Kräften beschreibt. Diese werden vorwiegend durch Kurvenfahrten oder Beschleunigungs- und Bremsvorgänge hervorgerufen. Die Relevanz dieses Teilmodells wird besonders durch die Notwendigkeit in vielen Fahrzeugassistenzsystemen und der damit verbundenen Auswirkung auf die Fahrsicherheit begründet. Das Fahrdynamikmodell kann neben dem Fahrzeugübertragungsverhalten, das die Kräfte simuliert, auch einzelne Komponenten und Randbedingungen nachbilden. Dadurch lässt sich die Vielfalt und Genauigkeit des Modells erhöhen. Außerdem steigt die Anzahl der möglichen Szenarien und durchführbaren Fahrmanöver, welche nicht in der Realität getestet werden müssen. Welcher Komplexitätsgrad notwendig ist, wird durch den Zweck des Modells definiert. Die Dynamik eines Fahrzeuges lässt sich auf verschiedenste Arten simulieren. Mögliche Ansätze sind ein- oder dreidimensionale Simulationsprogramme. 1D-Simulationen wie in Matlab oder Scilab überzeugen vor allem mit wenig Rechendauer und der damit verbundenen Echtzeitfähigkeit, jedoch fehlt die übersichtliche grafische Ausgabe. In 3D-Simulationen wäre dies gegeben. Allerdings kann die Echtzeitfähigkeit bei hoher Komplexität und niedriger Rechenleistung nicht gewährleistet werden. Hinzu kommen bei beiden Ansätzen hohe Lizenzkosten. Unter

Berücksichtigung aller genannten Vor- und Nachteile richtet sich der Fokus auf einen weiteren Ansatz, die Simulation in einer Game-Engine. Es stellt sich die Frage: Kann eine Spiele-Engine für eine Fahrdynamiksimulation verwendet werden? Die folgenden Kapitel gehen dieser Frage nach, indem die Fahrdynamik beschrieben und anschließend in die Game-Engine Unity3D implementiert wird.

II. GRUNDLAGEN DER FAHRZEUGDYNAMIK

Bei der Simulation der Fahrdynamik beschäftigt man sich mit der Beschreibung von Bewegungen. Dabei wichtige Größen sind Weg, Geschwindigkeit und Beschleunigung. Primär wird in diesem Abschnitt auf die Bewegung des Fahrzeugaufbaus eingegangen. Außerdem erfolgt eine Abgrenzung auf die wesentlichsten Merkmale der Fahrdynamik.

A. Freiheitsgrade eines Fahrzeugs

Das Darstellen der Fahrdynamik basiert auf dem Bewegungsverhalten der Fahrzeuge und wird durch jeweils drei translatorischen und rotatorischen Freiheitsgrade ausgedrückt. Die translatorischen Freiheitsgrade werden Zucken in Fahrzeuginnenrichtung, Schieben in Fahrzeugquerrichtung und Heben in Fahrzeughochrichtung genannt. Die rotatorischen Bewegungen werden mit Gieren (um die Hochachse), Nicken (um die Querachse) und Wanken (um die Längsachse) bezeichnet, welche in Abbildung 1 zu sehen sind.

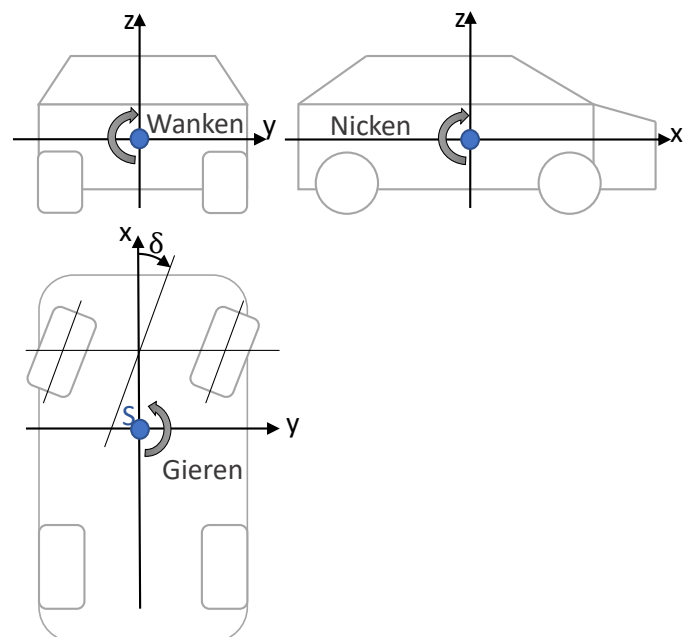


Abbildung 1: Rotatorische Freiheitsgrade vgl. [1]

Außerdem wird die Beschreibung der Fahrzeugbewegung abgebildet, welche ein fahrzeugfestes Koordinatensystem nutzt. Der Koordinatenursprung wird in den Schwerpunkt S gelegt [2].

Um Vorgänge besser eingliedern zu können wird die Fahrdynamik in drei Teilaspekte unterteilt. Der erste Teilaspekt, die Längsdynamik, beschreibt die Geradeausfahrt auf ebener Fahrbahn, Beschleunigungs- und Bremsvorgänge. Bezogen auf die Kinematik des Fahrzeugaufbaus ist deshalb nur die Längs- und Nickbewegung relevant. Teilaspekt zwei, die Vertikaldynamik, untersucht Fahrzustände die durch Unebenheiten in der Fahrbahn hervorgerufen werden. Die wichtigsten kinematischen Größen sind hier die Aufbaubeschleunigung sowie Nick- und Wankbewegungen. Der letzte Teilaspekt bezieht sich auf die Querdynamik. Primär betrachtet werden Lenkmanöver auf ebener Fahrbahn. Schwerpunktmäßig werden Beschleunigungs-, Brems- und Lenkmanöver betrachtet, was für die Verifizierung der Game-Engine als Simulationsumgebung ausreicht [1].

B. Kinematik des Fahrzeugaufbaus

Das Berechnen der primären Fahrdynamikgrößen greift auf die Kinematik der ebenen Starrkörperbewegung zurück. Grundlegend für die Bewegungsberechnung ist die Beschleunigung a , welche mit der am Fahrzeug wirkenden Kräfte bestimmt wird. Daraus lässt sich durch integrieren über die Zeit t die Geschwindigkeitswert v berechnen:

$$v = \int a \cdot dt \quad (1)$$

und durch erneute Integration der zurückgelegte Weg s berechnen:

$$s = \int v \cdot dt \quad (2)$$

Anders als bei den Freiheitsgraden, welche sich auf ein fahrzeuggebundenes Koordinatensystem stützen, wird die Lage mit einem an die Umgebung angelehnten Koordinatensystem berechnet. So lässt sich die Lage des Fahrzeugs mit einem Vektor auf einen festen Bezugspunkt beschreiben. Die Konvertierung zwischen den beiden Koordinatensystemen erfolgt durch Matrixtransformationsmethoden [1].

C. Kraftverteilung eines Fahrzeugs

Nachdem die grundlegenden kinematischen Eigenschaften beschrieben wurden, kann man diese durch Berechnungen mit auftretenden Kräften ergänzen, validieren oder ersetzen. Die dafür benötigten Kräfte zeigt Abbildung 2.

Die Masse des Fahrzeugs wirkt auf dessen Schwerpunkt mit der Gewichtskraft F_G . Wird nun der Fall der schiefen Ebene betrachtet, lässt sich die Kosinuskomponente (Normalkraft) F_N und die Sinuskomponente (Hangabtriebskraft) F_H mit der Gewichtskraft und dem Winkel des Gefälles α berechnen. h_a ist die Wirkhöhe der aerodynamischen Kraft D_A bzw. des Strömungswiderstandes. Neben der Beschreibung des Fahrzeugaufbaus, bringt auch die separate Modellierung der Reifen einen erheblichen Anstieg an Realitätsnähe. Weshalb

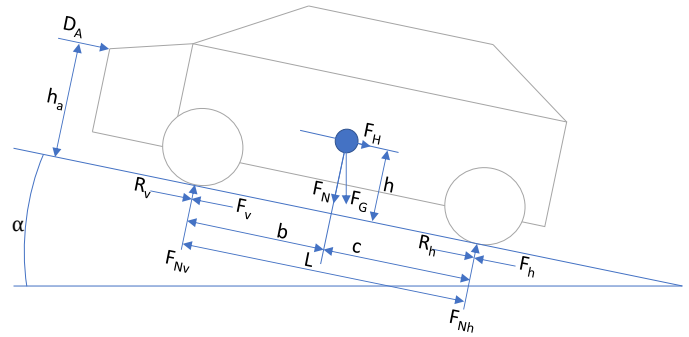


Abbildung 2: Kraftverhältnisse in der schiefen Ebene vgl. [3]

an dieser Stelle auch auf die Reifen eingegangen wird. Der Abstand zwischen den Reifen wird mit L gekennzeichnet. b und c sind der Längsabstand des Schwerpunktes zu den Vorder- und Hinterachsen. Die Rollwiderstände werden mit R_v und R_h bezeichnet. Diese wirken gegen die Zugkräfte F_v und F_h der beiden Achsen. Durch Berücksichtigen des Reifenschlupfs, müssen vor der Berechnung der Zugkräfte, die von den jeweiligen Achsen ausgehenden Normalkräfte F_{Nv} und F_{Nh} berechnet werden. Bei einer statischen Lastverteilung wird dies mit folgenden Formeln realisiert [3]:

$$F_{Nv} = F_N \cdot \frac{c}{L} \quad (3)$$

$$F_{Nh} = F_N \cdot \frac{b}{L} \quad (4)$$

Bei der schiefen Ebene muss der Gradient berücksichtigt werden:

$$F_{Nv} = F_N \cdot \frac{c}{L} - F_N \cdot \frac{c}{L} \cdot \alpha \quad (5)$$

$$F_{Nh} = F_N \cdot \frac{b}{L} - F_N \cdot \frac{b}{L} \cdot \alpha \quad (6)$$

Ähnlich dazu ändert sich auch bei einer Schräglage des Fahrzeugs die Kraftverteilung auf die Reifen. Dabei nimmt die Kraft auf die weiter unten liegenden Reifen F_{Nu} zu, während die Kraft der höheren Reifen F_{No} abnimmt. Die Formeln für diese Kräfte lauten:

$$F_{Nu} = \frac{F_N}{2} - F_N \cdot \frac{h}{t} \cdot |\beta| \quad (7)$$

$$F_{No} = \frac{F_N}{2} + F_N \cdot \frac{h}{t} \cdot |\beta| \quad (8)$$

h ist dabei die Höhe des Schwerpunktes über der Fahrbahn und t entspricht der Spurbreite des Fahrzeugs. Wichtig für die Implementierung ist, dass die obere bzw. untere Reifenkraft keinem Reifen zugeordnet ist und dem entsprechen situationsbezogen angewendet werden muss. Querneigungswinkel β und der Gradient der schiefen Ebene α werden in rad beschrieben. Gültig sind diese Formel bis zu einem Winkel von 20° [3].

Unter dynamischen Bedingungen kann die Last auf die Vorderräder (beim Bremsen), die Hinterräder (beim Beschleunigen) und von einer Seite zur anderen (in Kurven) übertragen werden. Die Bestimmung der dynamischen Lastübertragung

ist ein wichtiger Schritt bei der Analyse von Beschleunigung, Bremsen und Kurvenfahrten. Da die Achslasten die an jedem Rad wirkenden Zug- und Lenkkräfte bestimmen, welche sich wiederum auf das Verhalten des Fahrzeugs, wie Höchstgeschwindigkeit oder Bremsleistung auswirken. Die maßgeblichen Gleichungen für das Berechnen der Achslasten sind [3]:

$$F_{Nv} = F_N \cdot \frac{c}{L} - F_N \cdot \frac{h}{L \cdot g} \cdot a \quad (9)$$

$$F_{Nh} = F_N \cdot \frac{b}{L} + F_N \cdot \frac{h}{L \cdot g} \cdot a \quad (10)$$

Nachdem alle für die Reifen relevanten Normalkräfte ermittelt wurden, kann durch Mittelwertbildung über die jeweiligen Kräfte die resultierende Normalkraft für jedes einzelne Rad berechnet werden. Die bereits erwähnten Zugkräfte F_v und F_h wirken in Richtung positiver x-Achse und werden über das vom Motor erzeugte Moment¹ M_{Motor} , des Reifenradius r , der Getriebeübersetzung der Vorder- und Hinterachse i_v und i_h mit folgender Formel berechnet.

$$F = \frac{M_{Motor} \cdot i}{r} \quad (11)$$

Dagegen wirken die Hangabtriebskraft, der aerodynamische Fahrwiderstand und die Reibverluste. Der aerodynamische Fahrwiderstand ist abhängig von dem Strömungswiderstandskoeffizient c_w , der Luftwiderstandsfläche A , der Luftdichte ρ_{Luft} und der Referenzgeschwindigkeit v_{ref} :

$$D_A = c_w \cdot A \cdot \rho_{Luft} \cdot \frac{v_{ref}^2}{2} \quad (12)$$

wobei c_w Strömungswiderstandskoeffizient Die Reibverluste bildet sich aus dem Produkt des Reibungskoeffizient μ und Normalkraft [4]:

$$F_R = \mu \cdot F_N \quad (13)$$

Hier wird die resultierende Kraft, die Antriebskraft, durch summieren der genannten Kräfte berechnet.

III. MODELLBILDUNG UND SIMULATION

Zunächst werden der Modellansatz und die Modellierung der Fahrzeugdynamik beschrieben. Daraufhin wird erläutert, welche Schritte in der Game-Engine notwendig sind um die Fahrdynamik echtzeitfähig darstellen zu können.

A. Lineares Fahrzeugmodell

Ein lineares Fahrzeugmodell oder Einspurmodell ist das einfachste Modell für zweispurige Kraftfahrzeuge. Dabei beschreibt es primär die stationäre und instationäre Dynamiken und bildet die Grundlage für komplexere Fahrzeugmodelle. Für dieses Modell wird ein Fahrzeug mit Elektromotor (ohne Boost-Funktion), einem 1-Gang-Getriebe und Allradantrieb angenommen. Außerdem wird das Fahrzeug nicht durch die

Traktion eingeschränkt, d.h. die Reifen übertragen die gesamte Motorleistung auf die Fahrbahn.

Zur Beschreibung und Bewertung der Querdynamik ist es notwendig alle modellrelevanten Größen zu definieren. Der Fahrzeugaufbau wird als ein starrer Körper mit einer Gesamtmasse m betrachtet. Die Reifen werden gesondert betrachtet. Eingangsgrößen in das Modell sind lediglich der Lenkradwinkel δ und die Beschleunigung a . Die Reaktion auf die Eingangsgrößen sind die Giergeschwindigkeit und die Geschwindigkeit des Fahrzeugs v [1].

B. Implementierung in die Game Engine

Anstelle von kommerziellen Tools wird hier Unity, eine 3D-Spiele-Engine für die Entwicklung von Videospielen, für die Modellierung der Straßenumgebung und für die Fahrdynamikmodellierung verwendet. Der Hauptgründe für die Wahl dieser Spiele-Engine ist zum Einen der benutzerfreundliche GUI-Editor, zum Anderen ist diese Software kostenfrei und auf nahezu jedem Betriebssystem lauffähig. Außerdem können damit ausführbare Dateien generiert werden, welche neben den Betriebssystemen für Rechner auch auf Android und iOS eine Simulationsumgebung erstellen können. Dies könnte einen wesentlichen Einfluss auf die Validierung des Modells haben, da die Möglichkeit besteht diese durch "gamified and crowd-sourced virtual Testing" der Allgemeinheit zu überlassen. Ein weiteres Argument für Unity 3D ist, dass diese Engine eine Vielzahl von Fahrdynamikberechnungen bereits liefert [5].

Der erste Schritt der Implementierung ist die Generierung der Umgebung. Dabei wird auf den Unity Asset Store zurückgegriffen, welcher z.B. Pakete wie EasyRoads3D für den Straßenbau beinhaltet. Alle Objekte in der Umgebung werden mit dem Umgebungskoordinatensystem platziert, auch ein späterer Zugriff erfolgt über dieses System. Nach der Umgebungsgenerierung folgt die Erstellung des Game Objekts. In diesem Fall ist dies das Fahrzeug, deren Dynamik dargestellt werden soll. Objekte die dem Game Objekt untergeordnet sind, orientieren sich an dem fahrzeuggebundenen Koordinatensystem. Das verwendete Game Objekt muss über einen Fahrzeugaufbau und vier separate Reifen verfügen. Objekte dieser Art werden im Unity Asset Store angeboten. Optional kann mit der Software Blender eine derartiges Objekt erstellt und in Unity importiert werden. Diese Arbeit bezieht das kostenlose Asset Store Modell *car_1* aus dem Paket *Racing cars*. Die simulationsrelevanten Komponenten werden mit Collider erweitert, welche der Kollisionsdetektion und Kollision mit anderen Objekten dient. Für den Fahrzeugaufbau ist dies ein Box oder Mesh Collider, die Reifen erhalten einen Wheel Collider. Kameras dienen der Erfassung der Umgebung und zeigt diese dem Anwender an. Die Main Kamera befindet sich in einem 30 – 40° Winkel über dem Fahrzeug und zeigt in Fahrtrichtung. Eine weitere Kamera zeigt die Situation aus der Draufsicht. Die genannten Komponenten werden in Abbildung 3 als Explosionsdarstellung wiedergegeben und nummeriert. Dabei kennzeichnet Nummer 1 die Main Kamera, 2 die Zweitkamera, 3 den Fahrzeugaufbau, 4 den Fahrzeugcollider, 5 einen der Reifen, 6 den dazugehörigen Reifencollider und 7 das fahrzeugfeste Koordinatensystem.

¹Das Motormoment wird unter Verwendung der Geschwindigkeit und der Momentkennlinie des Motor bestimmt

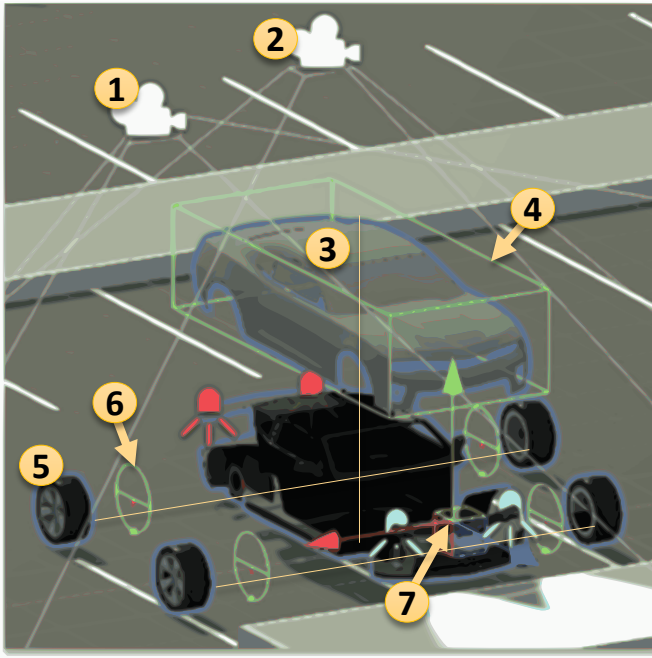


Abbildung 3: Explosionsdarstellung des Modells

Wie schon erwähnt, übernimmt Unity3D einen Teil der Fahrdynamiksimulation. Parameter wie Reifenmasse, Radius, Aufhängungsabstand, Dämpfungsrate und Federungswert werden dem Collider übergeben, welcher dann auf z.B. Unebenheiten der Fahrbahn reagiert. Diese Parameter dienen allerdings nicht zur Berechnung der resultierenden Antriebskraft oder Querdynamik. Dafür sind zusätzliche Berechnungen notwendig, welche innerhalb eines C#-Skriptes ausgeführt werden. Simuliert wird ein Audi e-tron (die Auswahl wurde rein zufällig getroffen). Für die Berechnungen geht man von den Parametern des Herstellers oder allgemein geltenden Annäherungen aus. So liegt z.B. der für die Berechnung der Luftwiderstandskraft benötigte C_W Wert bei 0,28 und der Rollreibungskoeffizient beträgt bei Asphalt und Reifen 0,015. Weitere relevante Fahrzeugdaten liefert die Quelle *AUDI E-TRON QUATTRO (2018): TECHNISCHE DATEN* [6]. Mit den erhaltenen Daten und den in Kapitel II-C genannten Gleichungen, werden weitere Teile der Fahrdynamik berechnet. Der erste Schritt ist die Berechnung der Normalkräfte, gefolgt von der des aerodynamischen Fahrwiderstandes, der Zugkräfte und der auftretenden Reibverluste. Zu beachten ist, dass Verluste und Widerstände abhängig von der Ausrichtung des Geschwindigkeitsvektors sind. Denn diese wirken in die entgegengesetzte Richtung, was in der Simulation mit dem Faktor $-\frac{v}{v}$ berücksichtigt wird. Im Fall des Stillstandes wirkt die Reibungskraft nicht effektiv, weshalb sie gleich 0 gesetzt wird. Für die Berechnung der resultierenden Antriebskraft, wird noch die vom Motor ausgehende Kraft benötigt. Diese wird durch Betätigung der nach oben, bzw. unten für den invertierten Wert, gerichteten Pfeiltaste generiert. Für die Berechnung der Motorkraft liegt nur die Kennlinie des Tesla Modell S vor. Weshalb die Kennlinie des e-trons durch Hochskalierung der Tesla-Kennlinie, unter Verwendung von Eckdaten beider Fahrzeuge, abgeschätzt wird. Die Fahrzeuggeschwindigkeit

lässt sich durch Einsetzen der resultierende Beschleunigung aus Gleichung 14 in Gleichung 15 abschätzen.

$$a_{res} = \frac{F_{res}}{m} \quad (14)$$

$$v_n = a + t + v_{n-1} \quad (15)$$

v_n ist nach der Berechnung die aktuelle Geschwindigkeit, t die vergangene Zeit seit der letzten Berechnung und v_{n-1} ist der vorhergehende Geschwindigkeitswert. Anzumerken ist, dass diese Berechnung nur für die Abschätzung der Geschwindigkeit gültig ist und stark von der Frame-Rate abhängt, da diese gleich der maximalen Berechnungsrate ist. Genauere Berechnungen sind mit Gleichung 1 möglich, benötigen jedoch höhere Rechenleistung. Eine weitere Funktion in Richtung der Längsachse ist das Bremsen. Hierbei wirkt die Bremskraft (konstante Bremsbeschleunigung) mit der Reibungskraft und tritt durch Drücken der Leertaste in Kraft. Die Berechnung der Querdynamik beschränkt sich in dieser Arbeit auf Lenkmanöver. Was wie bei der Beschleunigung durch Einlesen von Pfeiltasten geschieht. Doch in diesem Fall werden nur die Werte maximaler und minimaler Lenkradwinkel in das Modell gespeist. Die Querbewegung wird vereinfachend als konstant angenommen. Für den Fall, dass das Fahrzeug den Kontakt zum Untergrund verloren hat, wird lediglich der Luftwiderstand berücksichtigt. Der Kontakt zum Untergrund wird durch einen Kollisionstriggers an der Unterseite des Fahrzeuges überprüft. Der letzte Schritt in dieser Simulation, besteht darin die berechneten Daten dem Modell zu übergeben. Dabei greift man auf die beiden Unity-Methoden *Rotate* und *Translate* zurück. Diese ändern über das fahrzeuggebundene Koordinatensystem die Giergeschwindigkeit und die Geschwindigkeit in Längsrichtung.

IV. SIMULATIONSERGEBNISSE UND BEWERTUNG

Dieser Abschnitt behandelt die Darstellung und Bewertung des erstellten Modells, indem die berechneten Daten kritisch hinterfragt werden. Das Verhalten entlang der Längsachse wird durch den Output von Geschwindigkeit und Beschleunigung beschrieben, welche wie in Abbildung 4 dargestellt werden. Außerdem werden die vergangene Zeit und die Rotation um die Hochachse ausgegeben.



Abbildung 4: Visualisierung des Fahrzeugmodells in Unity

Mit diesen Daten kann man das Modell im Vergleich zur Realität bewerten. Der erste Bewertungspunkt bezieht sich auf die maximale Geschwindigkeit, welcher in der Simulation $204 \frac{km}{h}$ beträgt. Zum Vergleich wird v_{max} berechnet. Mit einem Reifenradius von $r = 37,05 cm$ (Reifentyp: 255/55R19), einem Drehzahlbereich von 0 bis $14.000 U/min$ und einer Gesamtübersetzung von ca. $i = 9,144$ ergibt sich mit folgender Formel 17 [4]

$$v_{max} = \frac{2\pi \cdot n \cdot r}{i} \quad (16)$$

eine Höchstgeschwindigkeit von

$$\begin{aligned} v_{max} &= \frac{2\pi \cdot \frac{14000}{60 s} \cdot 0,3705 m}{9,144} \\ &= 59,40 \frac{m}{s} = 213,85 \frac{km}{h} \end{aligned} \quad (17)$$

Ein weiteres Bewertungskriterium ist die Beschleunigung des Fahrzeugs. Diese wird mit zwei Geschwindigkeiten, welche in Tabelle I aufgeführt sind, validiert. Es wird zwischen den Hersteller- und Simulationsdaten verglichen [6].

Tabelle I: Vergleich der Beschleunigungszeit zwischen Hersteller und Simulation

Beschleunigung	Hersteller	Simulation
0 – $60 \frac{km}{h}$	3,1 s	3,05 s
0 – $80 \frac{km}{h}$	4,3 s	4,5 s
0 – $100 \frac{km}{h}$	5,8 s	6,3 s
0 – $180 \frac{km}{h}$	19,7 s	29,2 s

Das Modell liefert bis $100 \frac{km}{h}$ tolerierbare Ergebnisse, danach stieg die Abweichung an, was durch sich durch eine Aufsummierung der Fehler über die Zeit und den starken Anstieg des Luftwiderstandes erklären lässt. Die Berechnung der Fallbeschleunigung und der Reifendynamik obliegt der Game-Engine weshalb, unter Voraussetzung der korrekten Parametrisierung, keine ausschlaggebenden Fehler bemerkt wurden. Die Querdynamik wurde aufgrund fehlender Vergleichsdaten nicht validiert. Zusammenfassen für diesen Abschnitt lässt sich sagen, dass die Simulation bis $100 \frac{km}{h}$ die Fahrdynamik bezüglich der beschränkten Längsdynamik korrekt wiedergibt.

V. AUSBLICK

Die grundlegenden Elemente der Fahrzeugdynamik sind bereits in das Modell implementiert. Allerdings ist es in diesem Entwicklungsstadium nur für eine grobe Einschätzung des Verhaltens nützlich und dient daher eher als Grundlage für ein komplexeres Modell. Ein solches würde die restlichen translatorischen und rotatorischen Freiheitsgrade behandeln. Im ersten Schritt die Querdynamik mit Beschreibung des Querverhaltens durch Überarbeitung der Reifen und deren Haftung. Hierbei wäre es durchaus sinnvoll eine Differenzierung des Untergrundtyps durchzuführen, um beispielsweise auf Schotter oder Gras ein unterschiedliches Haftverhalten im Vergleich zu Asphalt hervorzuheben. Schritt zwei bezieht sich auf die Freiheitsgrade der Hochachse (z-Achse), welche aktuell die Unity interne Berechnung übernimmt. Damit sind

die Reaktionen von Dämpfern und Federn der Reifen, aber auch die auf das Fahrzeug wirkende Gravitation gemeint. Diese würden durch eigene Modelle ersetzt werden. Der letzte Schritt wäre eine höhere Detaillierung des Fahrzeugs. Darunter die Einführung von Trägheitsmomenten und Untergliederung des Fahrzeugaufbaus in Karosserie, Motor, Getriebe oder losen Objekten wie Gepäck. Neben der Erweiterung des Fahrzeugmodells wäre auch die Vervollständigung der Fahrmanöver, z.B. die Schräglage des Fahrzeugs, vorstellbar. Allerdings sollte dabei die Echtzeitfähigkeit nie außer Acht gelassen werden, denn diese charakterisiert letztendlich die Game-Engine im Vergleich zu einem dreidimensionalen Simulator. Sollte dies nicht mehr der Fall sein ist eine Kombination mit einem eindimensionalen Simulator wie Simulink vorstellbar. Außerdem wäre dort die Simulation, durch bessere Übersicht und bereits optimierten Rechenmethoden, deutlich einfacher.

VI. CONCLUSION

Diese Arbeit hatte das Ziel, herauszufinden ob eine Game-Engine zu einer Fahrdynamiksimulation fähig ist. Vorab muss man dazu sagen, dass eine Simulation eine Annäherung an die Realität ist und damit dieser nur in einem gewissen Maß entspricht. Mit dem erstellten Simulationsmodell ist es möglich die geforderten Freiheitsgrade eines Fahrzeugs zu beschreiben und in Unity zu visualisieren. Ein lineares Fahrzeugmodell welches den Fahrzeugaufbau und Reifen betrachtet, wurde mit den in Kapitel II genannten Grundlagen vollständig umgesetzt. Allerdings hat das Modell nur einen Geltungsbereich bis $100 \frac{km}{h}$. Darüber hinaus weicht die Simulation bei der Geschwindigkeitsberechnung deutlich von der Realität ab. Somit kann man sagen, dass es möglich ist, eine Fahrdynamiksimulation zumindest für einen gewissen Bereich in einer Game-Engine zu realisieren. Will man diesen Bereich erweitern muss das Modell deutlich höhere Komplexitätsebenen erreichen.

LITERATUR

- [1] Erich Schindler. *Fahrdynamik. Grundlagen des Lenkverhaltens und ihre Anwendung für fahrzeugregelsysteme*. Expert Verlag, 2007.
- [2] Michael Unterreiner. „Across-frequency in convolutive blind source separation“. dissertation. Universität Duisburg-Essen, 2013. URL: <https://d-nb.info/1051579686/34>.
- [3] Ted Zuvich. *Vehicle Dynamics for Racing Games*. URL: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.486.3116&rep=rep1&type=pdf>.
- [4] Paul Balzer. *Zur Längsdynamik des Tesla Roadsters S*. 2012. URL: <https://www.cbcity.de/zur-laengsdynamik-des-tesla-roadster-s> (besucht am 04.06.2020).
- [5] Masahiro Yamaura. *ADAS Virtual Prototyping using Modelica and Unity Co-simulation via OpenMETA*. Toyota InfoTechnology Center, 2016. URL: <https://www.ep.liu.se/ecp/124/006/ecp16124006.pdf>.

-
- [6] Gerd Stegmaier. *AUDI E-TRON QUATTRO (2018): TECHNISCHE DATEN*. 2018. URL: <https://www.auto-motor-und-sport.de/elektroauto/audi-e-tron-quattro-2018-elektroauto-suv/technische-daten/> (besucht am 28.05.2020).