



(12) **Offenlegungsschrift**

(21) Aktenzeichen: **10 2021 202 083.6**

(22) Anmeldetag: **04.03.2021**

(43) Offenlegungstag: **08.09.2022**

(51) Int Cl.: **B60W 30/08 (2012.01)**  
**G08G 1/16 (2006.01)**

(71) Anmelder:

**PSA Automobiles SA, Poissy, FR**

(74) Vertreter:

**Horst, Christian, Dr., 65428 Rüsselsheim, DE**

(72) Erfinder:

**Thiem, Christoph, 65428 Rüsselsheim, DE; Eberle,  
Ulrich, 65428 Rüsselsheim, DE**

(56) Ermittelter Stand der Technik:

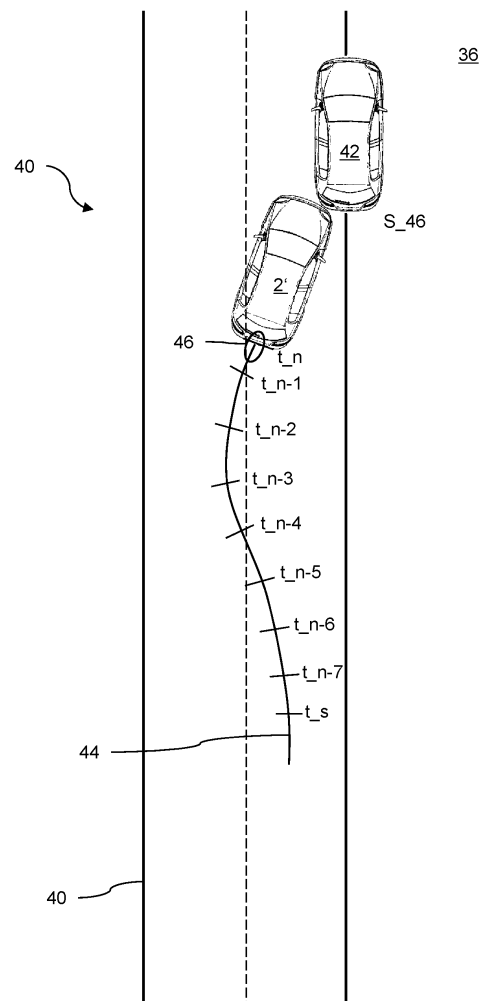
DE	10 2006 044 086	A1
DE	10 2008 001 256	A1
DE	10 2008 027 509	A1
DE	10 2019 206 908	A1

Rechercheantrag gemäß § 43 PatG ist gestellt.

**Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen.**

(54) Bezeichnung: **Computerimplementiertes Verfahren zum Trainieren wenigstens eines Algorithmus für eine Steuereinheit eines Kraftfahrzeugs, Computerprogrammprodukt, Steuereinheit sowie Kraftfahrzeug**

(57) Zusammenfassung: Beschrieben wird ein Verfahren zum Trainieren wenigstens eines Algorithmus für eine Steuereinheit eines autonomen Kraftfahrzeugs unter Verwendung des wenigstens einen Algorithmus, wobei der wenigstens eine Algorithmus in einer Simulationsumgebung durch ein selbstlernendes neuronales Netz trainiert wird, durch Auswahl einer kritischen Fahrsituation oder eines Unfalls des Kraftfahrzeugs; Nachbildung der kritischen Fahrsituation oder des Unfalls in der Simulationsumgebung; Berechnung eines Aufenthaltsbereichs des Kraftfahrzeugmodells zu einem früheren Zeitpunkt anhand der Fahr- und Umgebungsparameter unter Verwendung wenigstens eines Physikmodells; Simulation der kritischen Fahrsituation oder des Unfalls; Bestimmen von Abweichungen zwischen einem Zustand der kritischen Fahrsituation oder des Unfalls und der Simulation derselben sowie Anwenden eines Optimierungsalgorithmus zur Minimierung der Abweichungen zwischen der kritischen Fahrsituation oder des Unfalls und dessen Simulation.



## Beschreibung

**[0001]** Vorliegend werden ein computerimplementiertes Verfahren zum Trainieren wenigstens eines Algorithmus für eine Steuereinheit eines Kraftfahrzeugs, ein Computerprogrammprodukt, eine Steuereinheit sowie ein Kraftfahrzeug beschrieben.

**[0002]** Computerimplementiertes Verfahren zum Trainieren wenigstens eines Algorithmus für eine Steuereinheit eines Kraftfahrzeugs, Computerprogrammprodukte, Steuereinheiten sowie Kraftfahrzeuge der eingangs genannten Art sind im Stand der Technik bekannt.

**[0003]** Verfahren, Computerprogrammprodukte, Kraftfahrzeuge sowie Systeme der eingangs genannten Art sind im Stand der Technik bekannt. Die ersten teil-automatisiert (entspricht SAE Level 2 gemäß SAE J3016) fahrenden Kraftfahrzeuge sind in den vergangenen Jahren zur Serienreife gelangt. Automatisiert (entspricht SAE Level  $\geq 3$  gemäß SAE J3016) bzw. autonom (entspricht SAE Level 4/5 gemäß SAE J3016) fahrende Kraftfahrzeuge müssen anhand vielfältiger Vorgaben, zum Beispiel Fahrtziel und Einhaltung gängiger Verkehrsregeln, mit maximaler Sicherheit auf unbekannte Verkehrssituationen selbständig reagieren. Da die Verkehrswirklichkeit aufgrund der Unvorhersehbarkeit des Verhaltens anderer Verkehrsteilnehmer hoch komplex ist, gilt es als nahezu unmöglich, entsprechende Steuereinheiten von Kraftfahrzeugen mit herkömmlichen Methoden und auf der Grundlage von menschengemachten Regeln zu programmieren.

**[0004]** Stattdessen ist es bekannt, mit Methoden des maschinellen Lernens bzw. der künstlichen Intelligenz Algorithmen zu entwickeln. Solche Algorithmen können einerseits maßvoller auf komplexe Verkehrssituationen reagieren als traditionelle Algorithmen. Andererseits ist es mithilfe künstlicher Intelligenz prinzipiell möglich, die Algorithmen während des Entwicklungsprozesses und im Alltag durch ständiges Lernen weiterzuentwickeln. Alternativ wird ein Stand nach der Beendigung der Trainingsphase im Entwicklungsprozesses durch den Hersteller eingefroren.

**[0005]** Die Validierung entsprechend erzeugter automatisierter Fahrfunktionen ist allerdings sehr aufwendig und oft nur schwer realisierbar. Als Stand der Technik wird daher unter anderem auf szenarienbasierte Simulationen unter zu Hilfenahme eines sog. digitalen Zwillings zurückgegriffen. Der digitale Zwilling, also eine Simulation des zu trainierenden Fahrzeugmodells, weist den zu trainierenden Algorithmus auf, der so trainiert wird, dass der digitale Zwilling in der Simulation im Wesentlichen das Fahrverhalten des realen Kraftfahrzeugs nachahmt. Zum Training werden relevante Szenarien genutzt, um

das Verhalten des zu testenden automatisierten Fahrzeugmodells („System-under-Test“ oder „SuT“) zu untersuchen. Jedoch ist es schwierig, die Szenarien so auszuwählen und zu beschreiben, dass das Fahrzeugmodell wirklich in eine kritische Situation gerät bzw. einen möglichen hohen Anteil an verschiedensten kritischen Szenarien zu entwickeln.

**[0006]** Aus der DE 10 2006 044 086 A1 ist ein System zur virtuellen Simulation von Verkehrssituationen bekannt, mit einer Anzeigeeinheit zur bildhaften Anzeige der virtuellen Verkehrssituationen und einer Eingabeeinheit, und mit zumindest einem modellierten Referenzfahrzeug und zumindest einem weiteren modellierten Verkehrsteilnehmer, wobei das Referenzfahrzeug eine modellierte Detektionseinrichtung zur Erfassung der virtuellen Umgebung des Referenzfahrzeugs aufweist, und das Verhalten des Referenzfahrzeugs während einer simulierten Verkehrssituation zumindest zeitweise unter Berücksichtigung der durch die Detektionseinrichtung erfassten Informationen erfolgt.

**[0007]** Somit stellt sich die Aufgabe, Verfahren zum Trainieren wenigstens eines Algorithmus für eine Steuereinheit eines Kraftfahrzeugs, Computerprogrammprodukte sowie Kraftfahrzeuge der eingangs genannten Art dahingehend weiterzubilden, dass beim Trainieren des Algorithmus kritische Situationen besser nachgebildet werden können.

**[0008]** Die Aufgabe wird gelöst durch ein Verfahren zum Trainieren wenigstens eines Algorithmus für eine Steuereinheit eines Kraftfahrzeugs gemäß Anspruch 1, ein Computerprogrammprodukt gemäß dem nebengeordneten Anspruch 9, eine Steuereinheit gemäß dem nebengeordneten Anspruch 12, sowie ein Kraftfahrzeug gemäß dem nebengeordneten Anspruch 13. Weiterführende Ausgestaltungen und Weiterbildungen sind Gegenstand der abhängigen Ansprüche.

**[0009]** Nachfolgend wird ein computerimplementiertes Verfahren zum Trainieren wenigstens eines Algorithmus für eine Steuereinheit eines autonom oder teilautonom fahrenden Kraftfahrzeugs zur Umsetzung einer autonomen Fahrfunktion unter Eingriff in Aggregate des Kraftfahrzeugs auf der Grundlage von Eingangsdaten unter Verwendung des wenigstens einen Algorithmus beschrieben, wobei der wenigstens eine Algorithmus in einer Simulationsumgebung durch ein selbstlernendes neuronales Netz anhand eines Kraftfahrzeugmodells des Kraftfahrzeugs trainiert wird, umfassend folgende Schritte:

a) Bereitstellen wenigstens eines Computerprogrammproduktmoduls für die autonome Fahrfunktion, wobei das wenigstens eine Computerprogrammproduktmodul das Kraftfahrzeugmodell, den wenigstens einen zu trainierenden Algorithmus, das selbstlernende

neuronalen Netz sowie die Simulationsumgebung enthält;

b) Auswahl einer kritischen Fahrsituation oder eines Unfalls des Kraftfahrzeugs, wobei zu der kritischen Fahrsituation oder dem Unfall eine Mehrzahl von Fahrparametern des Kraftfahrzeugs und Umgebungsparametern bestimmt sind;

c) Nachbildung der kritischen Fahrsituation oder des Unfalls aus Schritt b) in der Simulationsumgebung und Festlegung eines Zielzeitpunktes ( $t_n$ ), wobei der Zielzeitpunkt ( $t_n$ ) von den Fahrparametern des Kraftfahrzeugs und den Umgebungsparametern aus Schritt b) umfasst sind;

d) Berechnung eines Aufenthaltsbereichs des Kraftfahrzeugmodells zu einem Zeitpunkt ( $t_{n-1}$ ), wobei der Zeitpunkt ( $t_{n-1}$ ) vor dem Zielzeitpunkt ( $t_n$ ) liegt, anhand der Fahrparameter und Umgebungsparameter unter Verwendung wenigstens eines Physikmodells;

e) Simulation der kritischen Fahrsituation oder des Unfalls vom Zeitpunkt ( $t_{n-1}$ ) zum Zielzeitpunkt ( $t_n$ ),

f) Bestimmen wenigstens einer Abweichung zwischen einem Zustand der kritischen Fahrsituation oder des Unfalls aus Schritt b) und der simulierten kritischen Fahrsituation oder des simulierten Unfalls aus Schritt e);

g) Anwenden eines Optimierungsalgorithmus zur Minimierung der wenigstens einen Abweichung zwischen der kritischen Fahrsituation oder des Unfalls aus Schritt b) und der simulierten kritischen Fahrsituation oder des simulierten Unfalls aus Schritt e).

**[0010]** Eine autonome Fahrfunktion übernimmt ganz oder teilweise die Steuerung eines Kraftfahrzeugs, indem sie eine Umgebung des Kraftfahrzeugs erfasst, daraus situativ ein sinnvolles Verhalten des Kraftfahrzeuges ableitet und das Kraftfahrzeug durch Eingriffe in Lenkung, Gas, Bremse und/oder andere Aggregate, z.B. Licht, Blinker etc. entsprechend des geplanten Verhaltens steuert. Dabei besteht, insbesondere im gemischten autonom - nicht-autonom-Verkehr generell der Wunsch, das Verhalten natürlicher Fahrer nachzuahmen, um durch ein unvorhersehbares Verhalten keine Sicherheitsrisiken hervorzurufen.

**[0011]** Die Fahrfunktion kann je nach Ausgestaltung auf Navigationsdaten zurückgreifen. Eine entsprechende autonome Fahrfunktion kann beispielsweise ein Stauassistent sein, der das Kraftfahrzeug in dichtem Verkehr bei langsamen Geschwindigkeiten selbsttätig steuert.

**[0012]** Als Eingangsdaten kommen generell aktuelle, die Umgebung betreffende Daten infrage, zum Beispiel mittels verschiedener Sensoren gewonnene Daten. Solche Sensoren können beispielsweise Kameras, Radar-, Lidar- und/oder Ultraschall-Sensoren umfassen, aber auch andere Sensoren wie zum Beispiel Positionssensoren, z.B. GPS, Magnetfelddetektierende Sensoren und dergleichen.

**[0013]** Weitere mögliche Eingangsdaten sind Farbplanungsdaten, die beispielsweise aus einem Navigationsziel gewonnen werden, sowie unter Umständen Verkehrsdaten, die den Verkehrsfluss auf der Route bestimmen. Weitere Daten können Kommunikationsdaten sein, die beispielsweise aus Car-to-Car- oder Car-to-Infrastructure-Systemen gewonnen werden, z.B. zu Ampelphasen o.ä..

**[0014]** Das selbstlernende neuronale Netz kann auf verschiedenen Lernprinzipien aufgebaut sein, insbesondere kann es Verfahren des verstärkenden Lernens anwenden. Verstärkendes Lernen (englisch reinforcement learning) steht für eine Reihe von Methoden des maschinellen Lernens, bei denen das neuronale Netz selbstständig eine Strategie erlernt, um erhaltene Belohnungen zu maximieren. Dabei wird dem neuronalen Netz nicht vorgezeigt, welche Aktion in welcher Situation die beste ist, sondern es erhält zu bestimmten Zeitpunkten eine Belohnung, die auch negativ sein kann. Anhand dieser Belohnungen approximiert es eine Nutzenfunktion, die beschreibt, welchen Wert ein bestimmter Zustand oder eine bestimmte Aktion hat.

**[0015]** Eine Möglichkeit, ein möglichst natürliches Fahrverhalten zu erreichen, ist es, einen virtuellen Zwilling eines realen Kraftfahrzeuges mit einem realen Fahrer in einer Simulationsumgebung zu definieren, die einer realen Umgebung nachgebildet ist. Der reale Fahrer und der virtuelle Zwilling bekommen identische Fahraufgaben, wobei die reale Fahrt die Referenz darstellt. Der virtuelle Zwilling hat die Aufgabe, das Fahrverhalten des realen Fahrers nachzubilden. Als Belohnungskriterien eignen sich hier eine Abweichung zwischen der Trajektorie des realen Fahrers und der Trajektorie des virtuellen Zwillings. Das selbstlernende neuronale Netz bekommt die Aufgabe, die Abweichung zu minimieren. Auf diese Weise imitiert das selbstlernende neuronale Netz das Verhalten realer Autofahrer.

**[0016]** Im Hinblick auf kritische Situationen oder gar Unfälle ist dies jedoch nicht so einfach, da es ethisch nicht vertretbar ist, reale Fahrer bewusst in gefährliche Situationen oder Unfallsituationen zu bringen. Mit dem hier vorgeschlagenen Verfahren ist es möglich, die bekannten Referenzfahrten durch Mithilfe des Verfahrens erzeugte simulierte Fahrten zu ersetzen, deren Verhalten dann von dem selbstlernenden

neuronalen Netz nachgeahmt werden sollen. Hierzu kann gemäß des Verfahrens auf vorhandene Daten zurückgegriffen werden, die Fahrparameter und Umgebungsparameter enthalten.

**[0017]** Als Startzeitpunkt  $t_n$  kann ein Zeitpunkt unmittelbar vor Eintreten der kritischen Situation oder des Unfalls gewählt werden, da oftmals nach einer kritischen Situation oder Unfall keine verlässliche Datenlage mehr vorliegt.

**[0018]** Durch Verwendung eines Physikmodells, das beispielsweise Bestandteil der Simulationsumgebung ist, kann eine wahrscheinliche Trajektorie vor dem Entstehen der kritischen Situation oder vor dem Unfall nach dem zuvor beschriebenen Verfahren rekonstruiert werden.

**[0019]** Durch die Verwendung eines Optimierungsalgorithmus ist es möglich, Abweichungen zwischen der kritischen Fahrsituation bzw. des Unfalls und der simulierten kritischen Fahrsituation bzw. des simulierten Unfalls zu minimieren, um ein möglichst realistisches Abbild der kritischen Fahrsituation bzw. des Unfalls in der Simulationsumgebung zu erhalten.

**[0020]** Anschließend können weitere Prozessschritte zum Trainieren des wenigstens einen Algorithmus durchgeführt werden, wie sie weiter oben erläutert wurden.

**[0021]** In einer ersten weiterführenden Ausgestaltung kann vorgesehen sein, dass h) die Schritte d) bis g) für einen Zeitraum zwischen ( $t_{n-1}$ ) und einem früheren Zeitraum ( $t_{n-2}$ ) durchgeführt werden.

**[0022]** Auf diese Weise wird ein iterativer Ansatz zum Zurückrechnen verwendet, mit dem die simulierte kritische Fahrsituation oder der simulierte Unfall rekonstruiert werden kann.

**[0023]** Diese Verfahrensschritte können abhängig von einer Inkrementgröße  $|t_n - t_{n-1}|$  mehrfach oder vielfach wiederholt werden, bis ein sinnvoller Startzeitpunkt vor der kritischen Fahrsituation oder dem Unfall erreicht worden ist, beispielsweise einige Sekunden vor der kritischen Fahrsituation oder dem Unfall.

**[0024]** In einer weiteren weiterführenden Ausgestaltung kann vorgesehen sein, dass die wenigstens eine Abweichung bewertet wird, wobei Schritt h) nur eingeleitet wird, wenn die wenigstens eine Abweichung unter einem Schwellwert liegt.

**[0025]** Auf diese Weise kann erreicht werden, dass die simulierte Trajektorie der kritischen Situation oder des Unfalls hinreichend präzise dem realen Vorbild entsprechen.

**[0026]** Als Abweichung kommen sowohl Abweichungen zum vorherigem Schritt als auch Abweichungen über mehrere Iterationsstufen in Frage, z.B. zwischen Startzeitpunkt und Zielzeit. Es können mehrere Abweichungen gemessen werden, z.B. Distanzen, Geschwindigkeiten oder Zeiten.

**[0027]** In einer weiteren weiterführenden Ausgestaltung kann vorgesehen sein, dass i) die Fahrparameter und/oder die Umgebungsparameter variable Fahrparameter und/oder variable Umgebungsparameter aufweisen, wobei wenigstens einer der variablen Fahrparameter und/oder Umgebungsparameter variiert werden, wobei die Simulation mit den variierten Parametern erneut durchgeführt wird.

**[0028]** Es können drei Arten von Parametern unterschieden werden:

1. Festgelegte Parameter - solche Parameter haben einen Wert, der für die kritische Fahrsituation oder den Unfall festgelegt ist. Das können z. B. alle Werte der Verkehrsinfrastruktur wie Straßenverlauf und -breite und dergleichen sein oder auch ein Reibwert der Straße, falls dieser bekannt ist.

2. Variable Umgebungsparameter - Hier können Wertebereiche (kontinuierlich oder diskret) festgelegt werden, deren konkrete Werte jedoch vor dem eigentlichen Start der Simulation festgelegt werden. Dies kann z.B. durch eine Versuchsplanung geschehen. Beispiele wären ebenfalls der Reibwert der Straße, weitere Verkehrsteilnehmer wie Fußgänger, Autofahrer und dergleichen, aber auch Sonnenstand, Sensorzustand etc.

3. Designparameter - Hier können Wertebereiche vorgegeben werden. Diese Wertebereiche für die werden von dem (numerischen) Optimierungsalgorithmus in der Simulationsphase variiert, um den Zielpunkt zu erreichen. Hierzu gehören beispielsweise Trajektorien verschiedener Verkehrsteilnehmer.

**[0029]** In einer weiteren weiterführenden Ausgestaltung kann vorgesehen sein, dass der Optimierungsalgorithmus gemäß Schritt g) die variablen Fahrparameter und/oder die variablen Umgebungsparameter zwischen zwei Iterationen der Optimierung variiert.

**[0030]** Dies kann als Methode des verstärkenden Lernens verwendet werden und erhöht die Automatisierung des Verfahrens.

**[0031]** In einer weiteren weiterführenden Ausgestaltung kann vorgesehen sein, dass die wenigstens eine Abweichung bewertet wird, wobei, wenn die wenigstens eine Abweichung kleiner als der Schwellwert ist, Schritt h) durchgeführt wird.

**[0032]** Somit kann die Präzision des Verfahrens sichergestellt werden.

**[0033]** In einer weiteren weiterführenden Ausgestaltung kann vorgesehen sein, dass wenn die wenigstens eine Abweichung unter dem Schwellwert liegt, der wenigstens eine Algorithmus gespeichert wird.

**[0034]** Die Speicherung kann eine Zwischenspeicherung sein. Nach dem Abspeichern kann ein weiterer Iterationsschritt durchgeführt werden bzw. nach Validierung der Fahrfunktion für diesen Fall eine andere Situation ausgewählt werden, an der der wenigstens eine vortrainierte Algorithmus weitertrainiert wird.

**[0035]** In einer weiteren weiterführenden Ausgestaltung kann vorgesehen sein, dass der wenigstens eine Algorithmus in weiteren Simulationen der gleichen kritischen Fahrsituation oder des Unfalls oder anderen kritischen Fahrsituationen oder Unfällen verwendet und trainiert wird.

**[0036]** Auf diese Weise kann der wenigstens eine Algorithmus an vielen verschiedenen kritischen Situationen oder Unfallsituationen trainiert werden.

**[0037]** In einer weiteren weiterführenden Ausgestaltung kann vorgesehen sein, dass wenn die wenigstens eine Abweichung von einem Startzeitpunkt ( $t_s$ ) bis zum Zielzeitpunkt ( $t_n$ ) nicht unter den Schwellwert liegt, die Fahrfunktion für die ausgewählte kritische Fahrsituation oder den Unfall validiert wird.

**[0038]** Ab einer gewissen Zeit kann es gemäß einer weiterführenden Ausgestaltung vorkommen, dass die wenigstens eine Abweichung zu groß wird. Ein Grund dafür kann sein, dass das Kraftfahrzeugmodell die kritische Situation oder den Unfall durch ein anderes Fahrverhalten als der menschliche Fahrer vermeiden kann. Dieses Ergebnis kann als Validierung der entsprechenden Situation verwendet werden, da das neuronale Netz die kritische Situation oder den Unfall zu vermeiden weiß.

**[0039]** Ein erster unabhängiger Gegenstand betrifft eine Vorrichtung zum Trainieren wenigstens eines Algorithmus für eine Steuereinheit eines autonom oder teilautonom fahrenden Kraftfahrzeugs zur Umsetzung einer autonomen Fahrfunktion unter Eingriff in Aggregate des Kraftfahrzeugs auf der Grundlage von Eingangsdaten unter Verwendung des wenigstens einen Algorithmus beschrieben, wobei eine Simulationsumgebung zum Trainieren des wenigstens einen Algorithmus durch ein selbstlernendes neuronales Netz anhand eines Kraftfahrzeugmodells des Kraftfahrzeugs vorgesehen ist, wobei die Simulationsumgebung dazu ausgestaltet ist, folgende Schritte durchzuführen:

a) Bereitstellen wenigstens eines Computerprogrammproduktmoduls für die autonome Fahrfunktion, wobei das wenigstens eine Computerprogrammproduktmodul das Kraftfahrzeugmodell, den wenigstens einen zu trainierenden Algorithmus, das selbstlernende neuronale Netz sowie die Simulationsumgebung enthält;

b) Auswahl einer kritischen Fahrsituation oder eines Unfalls des Kraftfahrzeugs, wobei zu der kritischen Fahrsituation oder dem Unfall eine Mehrzahl von Fahrparametern des Kraftfahrzeugs und Umgebungsparametern bestimmt sind;

c) Nachbildung der kritischen Fahrsituation oder des Unfalls aus Schritt b) in der Simulationsumgebung und Festlegung eines Zielzeitpunktes ( $t_n$ ), wobei der Zielzeitpunkt ( $t_n$ ) von den Fahrparametern des Kraftfahrzeugs und den Umgebungsparametern aus Schritt b) umfasst sind;

d) Berechnung eines Aufenthaltsbereichs des Kraftfahrzeugmodells zu einem Zeitpunkt ( $t_{n-1}$ ), wobei der Zeitpunkt ( $t_{n-1}$ ) vor dem Zielzeitpunkt ( $t_n$ ) liegt, anhand der Fahrparameter und Umgebungsparameter unter Verwendung wenigstens eines Physikmodells;

e) Simulation der kritischen Fahrsituation oder des Unfalls vom Zeitpunkt ( $t_{n-1}$ ) zum Zeitpunkt ( $t_n$ ),

f) Bestimmen wenigstens einer Abweichung zwischen einem Zustand der kritischen Fahrsituation oder des Unfalls aus Schritt b) und der simulierten kritischen Fahrsituation oder des simulierten Unfalls aus Schritt e);

g) Anwenden eines Optimierungsalgorithmus zur Minimierung der wenigstens einen Abweichungen zwischen der kritischen Fahrsituation oder des Unfalls aus Schritt b) und der simulierten kritischen Fahrsituation oder des simulierten Unfalls aus Schritt e).

**[0040]** In einer ersten weiterführenden Ausgestaltung kann vorgesehen sein, dass die Simulationsumgebung dazu eingerichtet ist,

h) die Schritte d) bis g) für einen Zeitraum zwischen ( $t_{n-1}$ ) und einem früheren Zeitraum ( $t_{n-2}$ ) durchzuführen.

**[0041]** In einer weiteren weiterführenden Ausgestaltung kann vorgesehen sein, dass die Simulationsumgebung dazu eingerichtet ist, die wenigstens eine Abweichung zu bewerten, wobei Schritt h) nur eingeleitet wird, wenn die wenigstens eine Abweichung unter einem Schwellwert liegt.

**[0042]** In einer weiteren weiterführenden Ausgestaltung kann vorgesehen sein, dass i) die Fahrparameter und/oder die Umgebungsparameter variable Fahrparameter und/oder variable Umgebungsparameter aufweisen, wobei die Simulationsumgebung dazu eingerichtet ist, wenigstens einen der variablen Fahrparameter und/oder Umgebungsparameter zu variieren und die Simulation mit den variierten Parametern erneut durchzuführen.

**[0043]** In einer weiteren weiterführenden Ausgestaltung kann vorgesehen sein, dass die Simulationsumgebung dazu eingerichtet ist, dass der Optimierungsalgorithmus gemäß Schritt g) die variablen Fahrparameter und/oder die variablen Umgebungsparameter zwischen zwei Iterationen der Optimierung variiert.

**[0044]** In einer weiteren weiterführenden Ausgestaltung kann vorgesehen sein, dass die Simulationsumgebung dazu eingerichtet ist, die wenigstens eine Abweichung zu bewerten, wobei, wenn die wenigstens eine Abweichung kleiner als der Schwellwert ist, Schritt h) durchgeführt wird.

**[0045]** In einer weiteren weiterführenden Ausgestaltung kann vorgesehen sein, dass die Simulationsumgebung dazu eingerichtet ist, wenn die wenigstens eine Abweichung unter dem Schwellwert liegt, den wenigstens einen Algorithmus zu speichern.

**[0046]** In einer weiteren weiterführenden Ausgestaltung kann vorgesehen sein, dass die Simulationsumgebung dazu eingerichtet ist, den wenigstens einen Algorithmus in weiteren Simulationen der gleichen kritischen Fahrsituation oder des Unfalls oder anderen kritischen Fahrsituationen oder Unfällen zu verwenden und zu trainieren.

**[0047]** In einer weiteren weiterführenden Ausgestaltung kann vorgesehen sein, dass die Simulationsumgebung dazu eingerichtet ist, die Fahrfunktion für die ausgewählte kritische Fahrsituation oder den Unfall zu validieren, wenn die wenigstens eine Abweichung von einem Startzeitpunkt ( $t_s$ ) bis zum Zielzeitpunkt ( $t_n$ ) nicht unter den Schwellwert liegt.

**[0048]** Ein weiterer unabhängiger Gegenstand betrifft ein Computerprogrammprodukt, mit einem dauerhaften, computerlesbaren Speichermedium, auf dem Befehle eingebettet sind, die, wenn sie von wenigstens einer Recheneinheit ausgeführt werden, bewirken, dass die wenigstens eine Recheneinheit dazu eingerichtet ist, das Verfahren der vorgenannten Art auszuführen.

**[0049]** Das Verfahren kann auf einer oder auf mehreren Recheneinheiten verteilt ausgeführt werden, sodass bestimmte Verfahrensschritte auf der einen Recheneinheit und andere Verfahrensschritte auf

wenigstens einer weiteren Recheneinheit ausgeführt werden, wobei berechnete Daten sofern notwendig zwischen den Recheneinheiten übermittelt werden können.

**[0050]** Die Recheneinheit kann Bestandteil der Steuereinheit sein.

**[0051]** In einer ersten weiterführenden Ausgestaltung kann vorgesehen sein, dass die Befehle das Computerprogrammproduktmodul der zuvor beschriebenen Art aufweisen.

**[0052]** Ein weiterer unabhängiger Gegenstand betrifft eine Steuereinheit mit einem dauerhaften, computerlesbaren Speichermedium, wobei auf dem Speichermedium ein Computerprogrammprodukt der zuvor beschriebenen Art gespeichert ist.

**[0053]** Ein weiterer unabhängiger Gegenstand betrifft ein Kraftfahrzeug mit einer Steuereinheit der zuvor beschriebenen Art.

**[0054]** In einer ersten weiterführenden Ausgestaltung kann vorgesehen sein, dass Recheneinheit Bestandteil der Steuereinheit ist.

**[0055]** In einer weiteren weiterführenden Ausgestaltung kann vorgesehen sein, dass die Recheneinheit mit Umgebungssensoren vernetzt ist.

**[0056]** Weitere Merkmale und Einzelheiten ergeben sich aus der nachfolgenden Beschreibung, in der - gegebenenfalls unter Bezug auf die Zeichnung - zumindest ein Ausführungsbeispiel im Einzelnen beschrieben ist. Beschriebene und/oder bildlich dargestellte Merkmale bilden für sich oder in beliebiger, sinnvoller Kombination den Gegenstand, gegebenenfalls auch unabhängig von den Ansprüchen, und können insbesondere zusätzlich auch Gegenstand einer oder mehrerer separater Anmeldung/en sein. Gleiche, ähnliche und/oder funktionsgleiche Teile sind mit gleichen Bezugszeichen versehen. Dabei zeigen schematisch:

**Fig. 1** ein Kraftfahrzeug, das zum automatisierten bzw. autonomen Fahren eingerichtet ist;

**Fig. 2** ein Computerprogrammprodukt für das Kraftfahrzeug aus **Fig. 1**;

**Fig. 3** eine Simulationsumgebung mit dem Kraftfahrzeug aus **Fig. 1**, sowie

**Fig. 4** ein Ablaufdiagramm des Verfahrens.

**[0057]** **Fig. 1** zeigt ein Kraftfahrzeug 2, das zum automatisierten bzw. autonomen Fahren eingerichtet ist.

**[0058]** Das Kraftfahrzeug 2 weist eine Steuereinheit 4 mit einer Recheneinheit 6 und einem Speicher 8

auf. Der Speicher 8 ist ein dauerhafter Speicher, dessen Daten nicht verloren gehen, wenn der Speicher 8 stromlos ist. Im Speicher 8 ist ein Computerprogrammprodukt gespeichert, das im Nachfolgenden im Zusammenhang mit den **Fig. 2** bis **Fig. 4** noch eingehender beschrieben wird.

**[0059]** Die Steuereinheit 4 ist einerseits mit einer Reihe von Umgebungssensoren verbunden, die eine Erfassung der aktuellen Lage des Kraftfahrzeugs 2 sowie der jeweiligen Verkehrssituation erlauben. Hierzu zählen Umgebungssensoren 10, 11 an der Front des Kraftfahrzeugs 2, Umgebungssensoren 12, 13 am Heck des Kraftfahrzeugs 2, eine Kamera 14 sowie ein GPS-Modul 16. Die Umgebungssensoren 10 bis 13 können z.B. Radar-, Lidar- und/oder Ultraschallsensoren umfassen.

**[0060]** Des Weiteren sind Sensoren zur Erfassung des Zustands des Kraftfahrzeugs 2 vorgesehen, unter anderem Raddrehzahlsensoren 16, Beschleunigungssensoren 18 und Pedalsensoren 20, die mit der Steuereinheit 4 verbunden sind. Mithilfe dieser Kraftfahrzeugsensoren 16, 18, 20 kann der momentane Zustand des Kraftfahrzeugs 2 zuverlässig erfasst werden.

**[0061]** Während des Betriebs des Kraftfahrzeugs 2 hat die Recheneinheit 6 das im Speicher 8 abgelegte Computerprogrammprodukt geladen und führt dieses aus. Auf der Grundlage eines Algorithmus und der Eingangssignale entscheidet die Recheneinheit 6 über die Steuerung des Kraftfahrzeugs 2, die die Recheneinheit 6 über Eingriff in die Lenkung 22, Motorsteuerung 24 sowie Bremsen 26 erreichen würde, die jeweils mit der Steuereinheit 4 verbunden sind.

**[0062]** Daten der Sensoren 10 bis 20 werden fortlaufend im Speicher 8 zwischengespeichert und nach einer vorgegebenen Zeitdauer verworfen, damit diese Umgebungsdaten zur weiteren Auswertung zur Verfügung stehen können.

**[0063]** Der Algorithmus wurde nach dem nachfolgend beschriebenen Verfahren trainiert.

**[0064]** **Fig. 2** zeigt ein Computerprogrammprodukt 28 mit einem Computerprogrammproduktmodul 30.

**[0065]** Das Computerprogrammproduktmodul 30 weist ein selbstlernendes neuronales Netz 32 auf, das einen Algorithmus 34 trainiert. Das selbstlernende neuronale Netz 32 lernt nach Methoden des bestärkenden Lernens, d. h. der Algorithmus 34 versucht durch Variation des neuronalen Netzes 32, Belohnungen für ein verbessertes Verhalten entsprechend einer oder mehrerer Metriken oder Maßstäbe, also für Verbesserungen des Algorithmus 34 zu erhalten. In anderen Ausführungsformen können

auch bekannte Lernverfahren des überwachten und unüberwachten Lernens, sowie Kombinationen dieser Lernverfahren, verwendet werden.

**[0066]** Das neuronale Netz 32 kann im Wesentlichen eine Matrix aus Werten, in der Regel Gewichte genannt, sein, die eine komplexe Filterfunktion definieren, die das Verhalten des Algorithmus 34 abhängig von Eingangsgrößen, welche vorliegend über die Umgebungssensoren 10 bis 20 aufgenommen werden, bestimmt und Steuersignale zur Steuerung des Kraftfahrzeugs 2 generiert.

**[0067]** Bei dem hier vorliegenden Algorithmus handelt es sich um ein sog. tiefes neuronales Netz (deep neural network), das neben einer Eingabeebene (input layer) und einer Ausgabeebene (output layer) wenigstens eine versteckte Ebene (hidden layer) aufweist.

**[0068]** Das Computerprogrammproduktmodul 30 kann sowohl im Kraftfahrzeug 2 als auch außerhalb des Kraftfahrzeugs 2 zum Einsatz gebracht werden. Somit ist es möglich, das Computerprogrammproduktmodul 30 sowohl in einer realen Umgebung als auch in einer Simulationsumgebung zu trainieren, in der ein virtueller Zwilling des Kraftfahrzeugs 2 trainiert wird.

**[0069]** Insbesondere beginnt das Training gemäß der hier beschriebenen Lehre in einer Simulationsumgebung, da dies sicherer ist als ein Training in einer realen Umgebung.

**[0070]** Das Computerprogrammproduktmodul 30 ist dazu eingerichtet, eine Metrik aufzustellen, die verbessert werden soll. Eine solche Metrik kann das Erreichen eines bestimmten Zielzustandes, der des realen Kraftfahrzeugs 2 sein. Wenn die Metrik eine gewisse Schwelle überschritten hat, z.B. eine Abweichung kleiner als ein gewisser Schwellwert, kann die Metrik als erfüllt gelten und der Algorithmus diesbezüglich eingefroren bzw. gespeichert werden. Dann kann entweder hinsichtlich einer anderen Metrik optimiert und anhand einer anderen Mission weiter trainiert werden oder der Algorithmus kann in einer realen Umgebung getestet werden.

**[0071]** Das Computerprogrammproduktmodul 30 weist eine Fahrfunktion 31 auf, z.B. ein autonomes Fahrprogramm, das auf der Grundlage von Eingangsdaten, zum Beispiel Umgebungsdaten der Umgebungssensoren 10 bis 13, der Kamera 14 sowie des GPS-Moduls 15 und ggf. Umgebungsdatenbanken sowie Fahrdaten der Sensoren 16 bis 18 auf der Grundlage einer Mission, beispielsweise einen gewissen Zielort zu erreichen, Eingriffe in Lenkung 22, Motorsteuerung 24 sowie Bremsen 26 plant.

**[0072]** Desweiteren ist ein neuronales Netz 32 vorgesehen, das vorliegend als tiefes neuronales Netz mit wenigstens einer versteckten Ebene ausgebildet ist und Bestandteil eines Algorithmus 34 ist, der Teile der Fahrfunktion 31 umsetzt.

**[0073]** Eine Simulationsumgebung 36 weist das Kraftfahrzeugmodell 2' auf, das eine virtuelle Repräsentanz des Kraftfahrzeugs 2 darstellt. Das Kraftfahrzeugmodell 2' bildet das Kraftfahrzeug 2 hinsichtlich seiner Fahreigenschaften nach.

**[0074]** Des Weiteren ist ein Physikmodell 38 vorgesehen, das das dynamische Verhalten der Bestandteile der Simulationsumgebung 36 auf der Grundlage von Fahrparametern pF, zum Beispiel Geschwindigkeit, Kurvengeschwindigkeit, Beschleunigung, Bremsbetätigung, Masse etc., und Umgebungsparametern pU, zum Beispiel Wetter und Straßenreiwerte, simuliert. Die Parameter pF und pU sind teilweise fest vorgegeben und teilweise inkrementell oder in Stufen variabel

**[0075]** Darüber hinaus ist ein Optimierungsalgorithmus 39 vorgesehen, der die Fahrparameter pF und/oder Umgebungsparameter pU variiert, um das Verhalten des Kraftfahrzeugmodells 2' zu beeinflussen.

**[0076]** Fig. 3 zeigt eine Wiedergabe der Simulationsumgebung 36 mit einem Kraftfahrzeugmodell 2', das ein virtueller Zwilling des Kraftfahrzeugs 2 aus Fig. 1 ist.

**[0077]** Vorbild für die Simulationsumgebung 36 ist eine Unfallbeschreibung in einer Unfalldatenbank, z.B. GIDAS, in der ein Unfall zwischen zwei Kraftfahrzeugen stattfand.

**[0078]** In einer solchen Datenbank sind beispielsweise folgende Informationen abgelegt:

Technische Bestandsaufnahme zu Unfall:

- Kraftfahrzeug: Baujahr 2021, Bremse: ABS, Motorisierung: 103 KW (1997 ccm), Höchstgeschwindigkeit: 195 km/h
- Ausstattung: Antischlupfregelung, Elektronisches Stabilitätsprogramm, Bremsassistent, Spurverlasswarnung, Reifendruckkontrolle, Geschwindigkeitsregelanlage
- Zustand nach Unfall:
  - Frontairbags Fahrer- und Beifahrerseite (aktiviert), Knieairbag Fahrerseite (aktiviert), Seitenairbag sitzintegriert 1. Reihe, Seiten-Kopfairbag 1. und 2. Reihe
  - Gurtstraffer, Gurtkraftbegrenzer 1. Sitzreihe (Auslösung an der Unfallstelle nicht feststellbar)

- keine offensichtlichen technischen Mängel am Kraftfahrzeug feststellbar

- Beschädigungen im gesamten Frontbereich durch Auffahren auf das stehende Kraftfahrzeug

- Beschädigungen im Innenraum im Armaturenbereich

**[0079]** Desweiteren sind in der Regel Skizzen zur Unfallumgebung, z.B. Straßenverlauf, Straßenbeschaffenheit, Wetter, Sonnenstand, Uhrzeit, und fahrdynamische Daten, z.B. Geschwindigkeiten, Trajektorien etc. gespeichert.

**[0080]** In der Simulationsumgebung 36 ist zur Nachbildung des zuvor beschriebenen Unfalls eine Straße 40 abgebildet, auf der sich neben dem Kraftfahrzeug 2 ein weiteres Kraftfahrzeug 42 am Rand der Straße 40 befindet. Gezeigt ist der Zeitpunkt des Zusammenstoßes des Kraftfahrzeugs 2 mit dem Kraftfahrzeug 40.

**[0081]** Um das neuronale Netz 32 des Algorithmus 34 zu trainieren, wird wie bereits zuvor beschrieben das Verfahren des verstärkenden Lernens angewendet, wobei der Algorithmus 34 dazu trainiert wird, das Verhalten des realen Kraftfahrzeugs 2 möglichst genau nachzuahmen. Da jedoch von dem zu simulieren den Unfall keine realen Fahrdaten vorhanden sind, die direkt nachgeahmt werden können, werden stattdessen anhand der Daten aus der Unfalldatenbank Vergleichs-Fahrdaten erzeugt, die der Algorithmus 34 nachahmen soll.

**[0082]** Dazu werden die bekannten Daten zu einer Trajektorie 44 des Kraftfahrzeugs 2 in der Simulationsumgebung 36 iterativ nachgeahmt und mithilfe des Optimierungsalgorithmus 39 soweit als möglich optimiert, sodass das Kraftfahrzeugmodell 2' das bekannte Verhalten des Kraftfahrzeugs 2 bestmöglich emuliert.

**[0083]** Um dies zu bewerkstelligen, wird die Trajektorie 44 in eine Vielzahl kurzer Abschnitte aufgeteilt, die voneinander jeweils ein Zeitinkrement  $[t_{n-1}; t_n]$  mit einer Dauer von  $\Delta t = |t_n - t_{n-1}|$  beabstandet sind. Dann wird die Fahrt des Kraftfahrzeugmodells 2' mithilfe des Physikmodells 38 zunächst von einem ein Zeitinkrement  $\Delta t$  vor einer Zielzeit  $t_n$  (Zeitpunkt der Kollision) liegenden Zeitraum  $t_{n-1}$  zur Zielzeit  $t_n$  simuliert und solange durch Variation der veränderbaren Fahrparameter pF variiert, bis das simulierte Kraftfahrzeugmodell 2' bis auf eine Abweichung 46 geringer als ein vorgegebener Schwellwert  $S_{46}$  mit den gleichen Werten an der gleichen Stelle mit dem Kraftfahrzeug 42 zusammenstößt.

**[0084]** Ist dies erreicht, wird der Zwischenstand abgespeichert und der Vorgang in dem davorliegenden

den Zeitinkrement  $[t_{n-2}; t_{n-1}]$  wiederholt. Dies findet so oft statt, bis eine Startzeit  $t_s$  erreicht wurde.

**[0085]** Zur Validierung der Fahrfunktion 31 wird dann der Algorithmus 34 angewendet und trainiert, indem er auf das mithilfe des zuvor beschriebenen Verfahrens entwickelte Fahrverhalten des Kraftfahrzeugmodells 2', das das Fahrverhalten des in den Unfall verwickelten Kraftfahrzeugs 2 imitiert, trainiert wird.

**[0086]** Ab einem gewissen Abstand kann es vorkommen, dass der Algorithmus 34 in der Lage ist, den Unfall zu vermeiden. Dies kann als Validierung des Algorithmus 34 verstanden werden.

**[0087]** Fig. 4 zeigt ein Ablaufdiagramm des Verfahrens.

**[0088]** Nach dem Start wird eine Unfallsituation aus einer Datenbank ausgewählt und in die Simulationsumgebung eingepreßt. Die Fahrsituation wird bis zur Zielzeit  $t_n$  nachgebildet.

**[0089]** Danach beginnt die Simulation der Fahrt des Kraftfahrzeugmodells für eine Zeitinkrementdauer  $\Delta t$  von einer Zeit  $t_{n-1}$  bis zur Zielzeit  $t_n$ . Der Aufenthaltsbereich des Kraftfahrzeugmodells 2' wird mit den Daten zu dem realen Kraftfahrzeug 2 verglichen. Liegt dieses nicht innerhalb einer akzeptablen Abweichung 46, wird der Optimierungsalgorithmus 39 angewendet und variable Fahrparameter  $p_F$  variiert.

**[0090]** Wird eine akzeptable Abweichung 46 erreicht, wird das zuvor beschriebene Verfahren ab dem zuvor liegenden Zeitpunkt  $t_{n-2}$  ausgehend solange wiederholt, bis das Verfahren bei der Startzeit  $t_s$  angekommen ist.

**[0091]** Danach wird das neuronale Netz 32 so lange trainiert, bis der Unfall vermieden werden kann.

**[0092]** Obwohl der Gegenstand im Detail durch Ausführungsbeispiele näher illustriert und erläutert wurde, so ist die Erfindung nicht durch die offenbarten Beispiele eingeschränkt und andere Variationen können vom Fachmann hieraus abgeleitet werden. Es ist daher klar, dass eine Vielzahl von Variationsmöglichkeiten existiert. Es ist ebenfalls klar, dass beispielhaft genannte Ausführungsformen nur Beispiele darstellen, die nicht in irgendeiner Weise als Begrenzung etwa des Schutzbereichs, der Anwendungsmöglichkeiten oder der Konfiguration der Erfindung aufzufassen sind. Vielmehr versetzen die vorhergehende Beschreibung und die Figurenbeschreibung den Fachmann in die Lage, die beispielhaften Ausführungsformen konkret umzusetzen, wobei der Fachmann in Kenntnis des offenbarten Erfindungsgedankens vielfältige Änderungen beispielsweise

hinsichtlich der Funktion oder der Anordnung einzelner, in einer beispielhaften Ausführungsform genannter Elemente vornehmen kann, ohne den Schutzbereich zu verlassen, der durch die Ansprüche und deren rechtliche Entsprechungen, wie etwa einer weitergehenden Erläuterung in der Beschreibung, definiert wird.

#### Bezugszeichenliste

2	Kraftfahrzeug
2'	Kraftfahrzeugmodell
4	Steuereinheit
6	Recheneinheit
8	Speicher
10-13	Umgebungssensor
14	Kamera
15	GPS-Modul
16	Raddrehzahlsensor
18	Beschleunigungssensor
20	Pedalsensor
22	Lenkung
24	Motorsteuerung
26	Bremsen
28	Computerprogrammprodukt
30	Computerprogrammproduktmodul
31	Fahrfunktion
32	neuronales Netz
34	Algorithmus
36	Simulationsumgebung
38	Physikmodell
39	Optimierungsalgorithmus
40	Straße
42	geparktes Kraftfahrzeug
44	Trajektorie
46	Abweichung
$t_n$	Zielzeit
$t_{n-1}, t_{n-2} \dots$	Zeit
$t_s$	Startzeit
$\Delta t$	Zeitinkrementdauer

pF	Fahrparameter
pU	Umgebungsparameter
S_46	Schwellwert

**ZITATE ENTHALTEN IN DER BESCHREIBUNG**

**Zitierte Patentliteratur**

- DE 102006044086 A1 [0006]

## Patentansprüche

1. Computerimplementiertes Verfahren zum Trainieren wenigstens eines Algorithmus (34) für eine Steuereinheit (4) eines autonom oder teilautonom fahrenden Kraftfahrzeugs (2) zur Umsetzung einer autonomen Fahrfunktion (31) unter Eingriff in Aggregate (22, 24, 26) des Kraftfahrzeugs (2) auf der Grundlage von Eingangsdaten unter Verwendung des wenigstens einen Algorithmus (34), wobei der wenigstens eine Algorithmus (34) in einer Simulationsumgebung (36) durch ein selbstlernendes neuronales Netz (32) anhand eines Kraftfahrzeugmodells (2') des Kraftfahrzeugs (2) trainiert wird, umfassend folgende Schritte:

a) Bereitstellen wenigstens eines Computerprogrammproduktmoduls (30) für die autonome Fahrfunktion (31), wobei das wenigstens eine Computerprogrammproduktmodul (30) das Kraftfahrzeugmodell (2'), den wenigstens einen zu trainierenden Algorithmus (34), das selbstlernende neuronale Netz (32) sowie die Simulationsumgebung (36) enthält;

b) Auswahl einer kritischen Fahrsituation oder eines Unfalls des Kraftfahrzeugs (2), wobei zu der kritischen Fahrsituation oder dem Unfall eine Mehrzahl von Fahrparametern (pF) des Kraftfahrzeugs (2) und Umgebungsparametern (pU) bestimmt sind;

c) Nachbildung der kritischen Fahrsituation oder des Unfalls aus Schritt b) in der Simulationsumgebung (36) und Festlegung eines Zielzeitpunktes ( $t_n$ ), wobei der Zielzeitpunkt ( $t_n$ ) von den Fahrparametern (pF) des Kraftfahrzeugs (2) und den Umgebungsparametern (pU) aus Schritt b) umfasst sind;

d) Berechnung eines Aufenthaltsbereichs ( $\Delta$ ) des Kraftfahrzeugmodells (2') zu einem Zeitpunkt ( $t_{n-1}$ ), wobei der Zeitpunkt ( $t_{n-1}$ ) vor dem Zielzeitpunkt ( $t_n$ ) liegt, anhand der Fahrparameter (pF) und Umgebungsparameter (pU) unter Verwendung wenigstens eines Physikmodells (38);

e) Simulation der kritischen Fahrsituation oder des Unfalls vom Zeitpunkt ( $t_{n-1}$ ) zum Zielzeitpunkt ( $t_n$ );

f) Bestimmen wenigstens einer Abweichung (46) zwischen einem Zustand der kritischen Fahrsituation oder des Unfalls aus Schritt b) und der simulierten kritischen Fahrsituation oder des simulierten Unfalls aus Schritt e);

g) Anwenden eines Optimierungsalgorithmus (39) zur Minimierung der wenigstens einen Abweichung (46) zwischen der kritischen Fahrsituation oder des Unfalls aus Schritt b) und der simulierten kritischen Fahrsituation oder des simulierten Unfalls aus Schritt e).

2. Computerimplementiertes Verfahren nach Anspruch 1, wobei h) die Schritte d) bis g) für einen Zeitraum zwischen ( $t_{n-1}$ ) und einem früheren Zeitraum ( $t_{n-2}$ ) durchgeführt werden.

3. Computerimplementiertes Verfahren nach Anspruch 1 oder 2, wobei die wenigstens eine Abweichung (46) bewertet wird, wobei Schritt h) nur eingeleitet wird, wenn die wenigstens eine Abweichung (46) unter einem Schwellwert ( $S_{46}$ ) liegt.

4. Computerimplementiertes Verfahren nach einem der vorangegangenen Ansprüche, wobei i) die Fahrparameter (pF) und/oder die Umgebungsparameter (pU) variable Fahrparameter (pF) und/oder variable Umgebungsparameter (pU) aufweisen, wobei wenigstens einer der variablen Fahrparameter (pF) und/oder Umgebungsparameter (pU) variiert werden, wobei die Simulation mit den variierten Parametern (pF, pU) erneut durchgeführt wird.

5. Computerimplementiertes Verfahren nach einem der vorangegangenen Ansprüche, wobei der Optimierungsalgorithmus (39) gemäß Schritt g) die variablen Fahrparameter (pF) und/oder die variablen Umgebungsparameter (pU) zwischen zwei Iterationen der Optimierung variiert.

6. Computerimplementiertes Verfahren nach einem der vorangegangenen Ansprüche, wobei die wenigstens eine Abweichung (46) bewertet wird, wobei, wenn die wenigstens eine Abweichung (46) kleiner als der Schwellwert ( $S_{46}$ ) ist, Schritt h) durchgeführt wird.

7. Computersimuliertes Verfahren nach einem der vorangegangenen Ansprüche, wobei, wenn die wenigstens eine Abweichung (46) unter dem Schwellwert ( $S_{46}$ ) liegt, der wenigstens eine Algorithmus (34) gespeichert wird.

8. Computersimuliertes Verfahren nach Anspruch 7, wobei der wenigstens eine Algorithmus (34) in weiteren Simulationen der gleichen kritischen Fahrsituation oder des Unfalls oder anderen kritischen Fahrsituationen oder Unfällen verwendet und trainiert wird.

9. Computersimuliertes Verfahren nach einem der vorangegangenen Ansprüche, wobei, wenn die wenigstens eine Abweichung (46) von einem Startzeitpunkt ( $t_s$ ) bis zum Zielzeitpunkt ( $t_n$ ) nicht unter dem Schwellwert ( $S_{46}$ ) liegt, die Fahrfunktion (31) für die ausgewählte kritische Fahrsituation oder den Unfall validiert wird.

10. Computerprogrammprodukt, mit einem dauerhaften, computerlesbaren Speichermedium (8), auf dem Befehle eingebettet sind, die, wenn sie von wenigstens einer Recheneinheit (4) ausgeführt werden, bewirken, dass die wenigstens eine Recheneinheit (4) dazu eingerichtet ist, das Verfahren nach einem der vorangegangenen Ansprüche auszuführen.

11. Computerprogrammprodukt nach Anspruch 10, wobei die Befehle das Computerprogrammproduktmodul (30) nach einem der Ansprüche 1 bis 9 aufweisen.

12. Steuereinheit (4) mit einem dauerhaften, computerlesbaren Speichermedium (8), wobei auf dem Speichermedium (8) ein Computerprogrammprodukt nach Anspruch 10 oder 11 gespeichert ist.

13. Kraftfahrzeug mit einer Steuereinheit (4) nach Anspruch 12.

14. Kraftfahrzeug nach Anspruch 13, wobei die Steuereinheit (4) mit Umgebungssensoren (10, 11, 12, 13, 14, 15) vernetzt ist.

Es folgen 3 Seiten Zeichnungen

Anhängende Zeichnungen

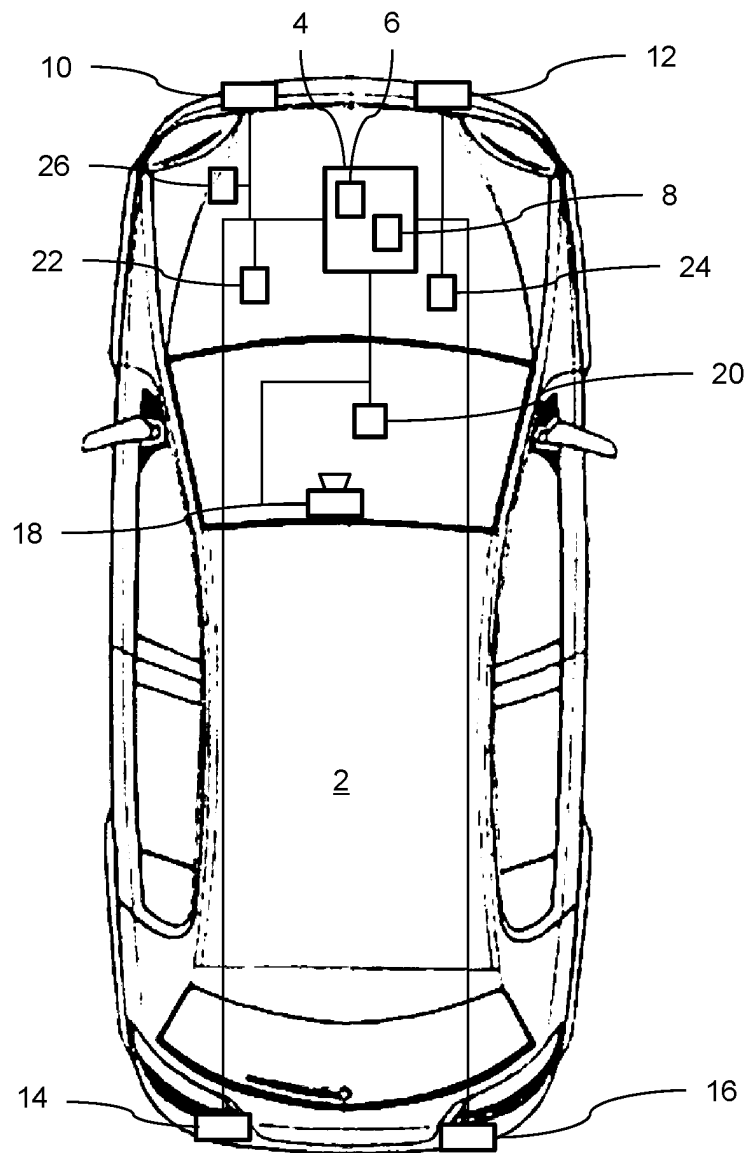


Fig. 1

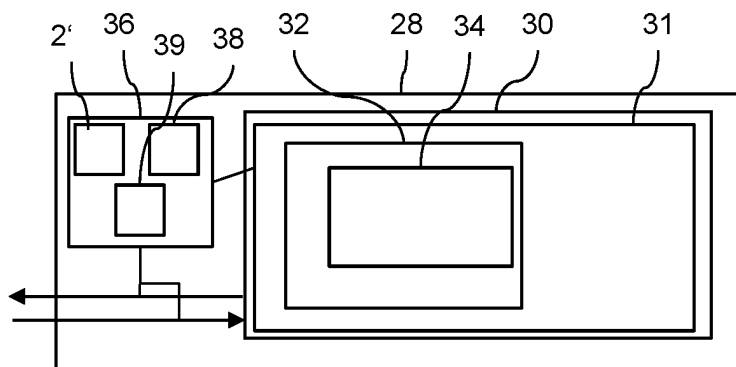


Fig. 2

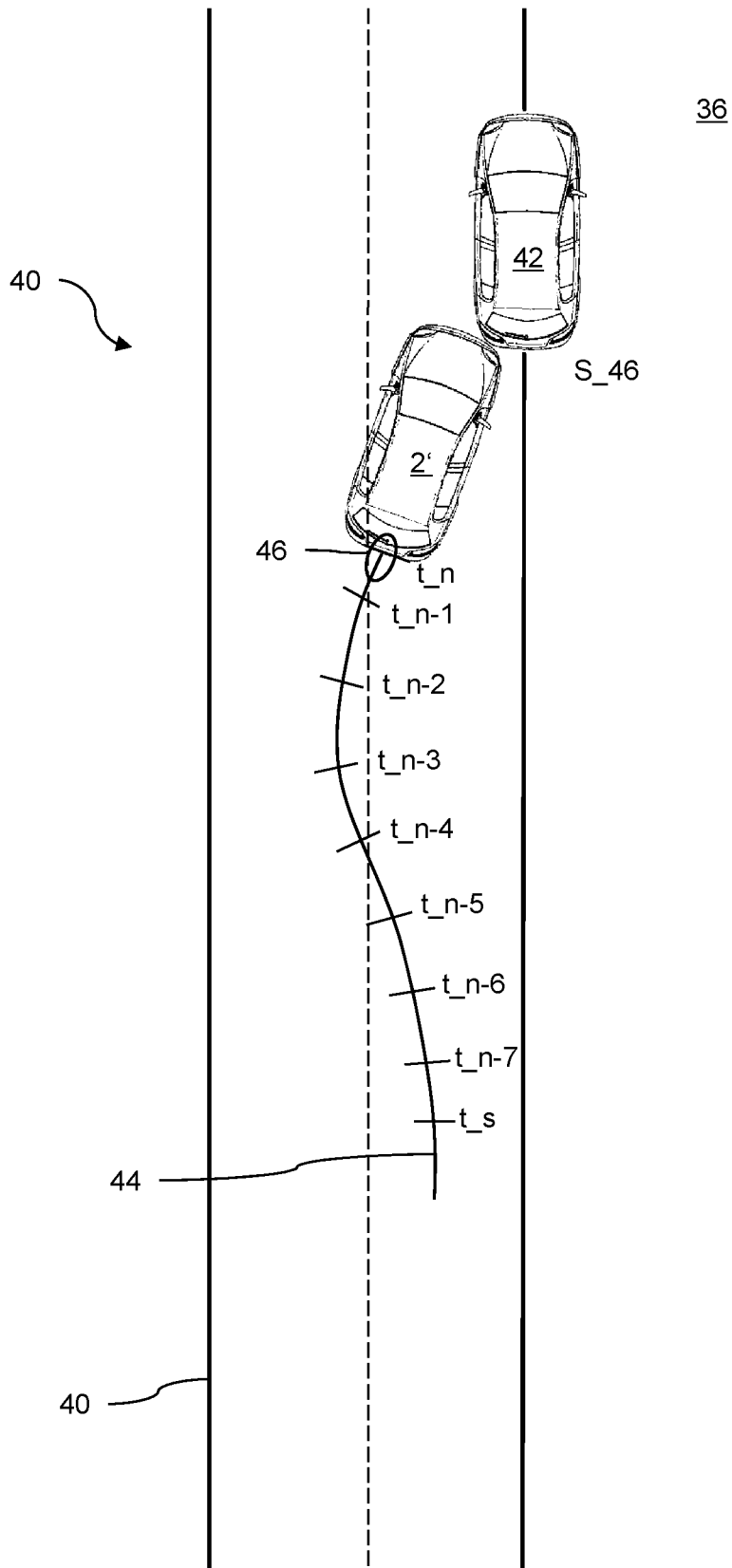


Fig. 3

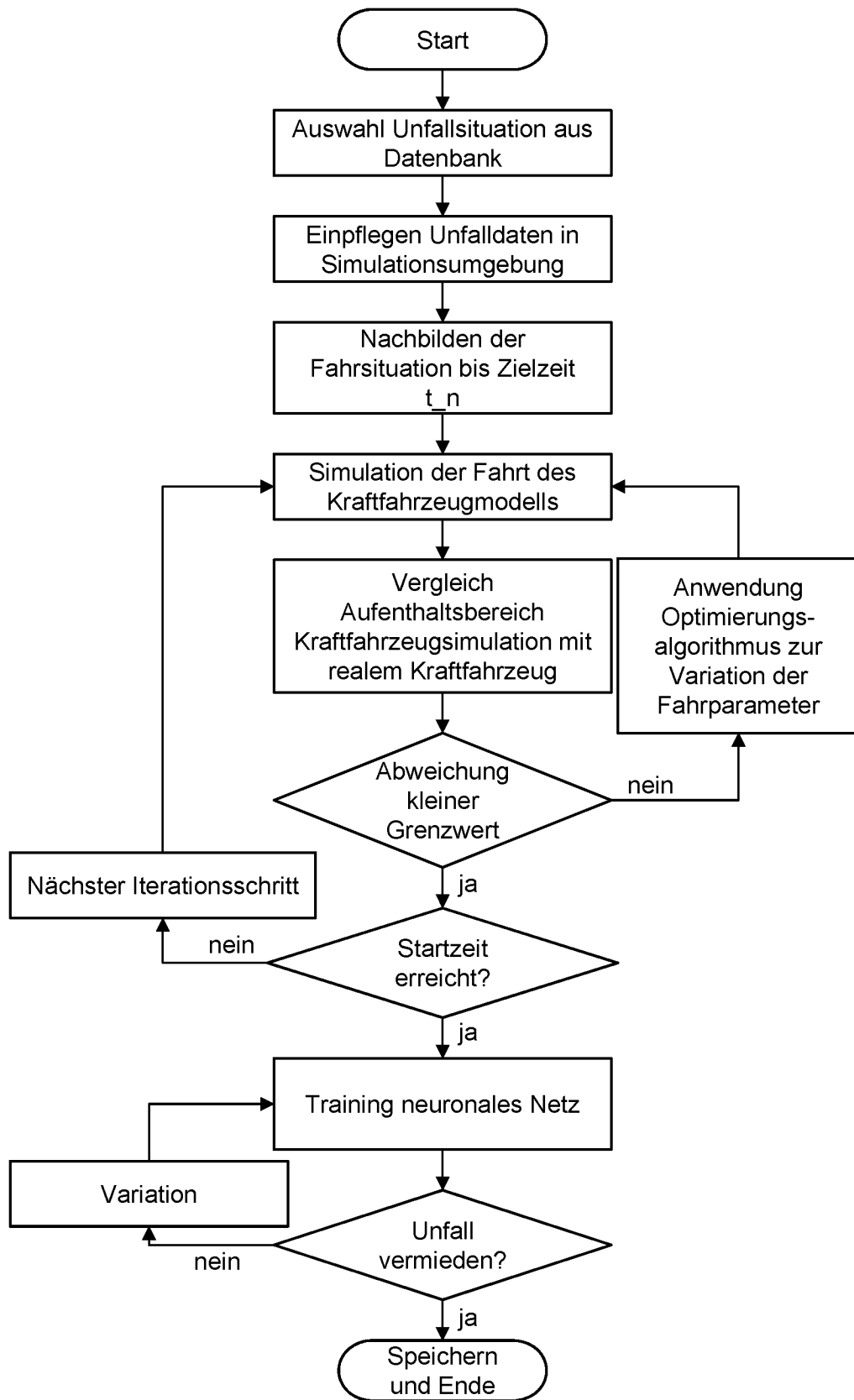


Fig. 4