Herausforderungen in der Absicherung von Fahrerassistenzsystemen bei der Benutzung maschinell gelernter und lernender Algorithmen

Maren Henzel*, Prof. Dr. rer. nat. Hermann Winner† und Dr.-Ing. Benedikt Lattke+

Zusammenfassung: Der Einsatz maschinell gelernter und lernender Algorithmen verstärkt sich im Bereich der Fahrerassistenzsysteme zunehmend basierend auf den hieraus resultierenden Vorteilen wie der Individualisierung des Systems oder der Identifikation komplexer Zusammenhänge. Bei leistungsfähigen maschinell gelernten Algorithmen handelt es sich jedoch zumeist um Modelle mit einer hohen Komplexität, die einer Black-Box gleichzusetzen sind. Dies stellt besondere Herausforderungen an die Absicherung der Assistenzsysteme. Werden darüber hinaus im Fahrbetrieb weiterlernende Modelle eingesetzt, besteht nicht nur die Herausforderung in der einmaligen Absicherung dieser Algorithmen zum Auslieferungszeitpunkt, sondern zusätzlich in der ständigen Absicherung des geänderten Verhaltens im Betrieb. Die bisher bekannten Lösungsansätze dieser Problematik werden vorgestellt sowie kritisch im Hinblick auf ihren Einsatz zur Absicherung von Fahrerassistenzsystemen diskutiert.

Schlüsselwörter: Absicherung, Fahrerassistenzsysteme, ISO26262, maschinelles Lernen, Systementwicklung

1 Einleitung

Maschinell gelernte Algorithmen halten in den letzten Jahren immer stärkeren Einzug in Fahrzeugsysteme. Schätzungen zufolge wird die im Jahr 2015 vorhandene Anzahl von sieben Millionen Fahrzeugsystem-Einheiten, die sich künstlicher Intelligenz bedienen, bis 2025 auf 225 Millionen Einheiten anwachsen. Dabei wird der Einsatz der gelernten Algorithmen neben dem Infotainment-Bereich auch im Rahmen von Fahrerassistenzsystemen und (teil-)automatisierten Fahrfunktionen erwartet. [1] Durch den Einsatz der gelernten und lernenden Algorithmen ergeben sich Veränderungen im bisherigen Systementwicklungsprozess von Fahrerassistenzsystemen. Hierfür sprechen ebenfalls

* Maren Henzel ist Wissenschaftliche Mitarbeiterin des Fachgebiets Fahrzeugtechnik an der Technischen Universität Darmstadt, Otto-Berndt-Str. 2, 64287 Darmstadt (e-mail: henzel@fzd.tu-darmstadt.de).

† Prof. Dr. rer. nat Hermann Winner Leiter des Fachgebiets Fahrzeugtechnik an der Technischen Universität Darmstadt, Otto-Berndt-Str. 2, 64287 Darmstadt (e-mail: winner@fzd.tu-darmstadt.de).

+ Dr.-Ing. Benedikt Lattke ist Projektleiter ADAS & Automation der Division Chassis & Safety bei Continental, Guerickestr. 7, 60488 Frankfurt am Main (e-mail: benedikt.lattke@continental-corporation.com)

Aussagen des Fahrzeugherstellers Tesla, bei denen die Funktionalität des Flottenlernens zur Verbesserung der Umfelderfassung für den „Autopiloten“ im Rahmen des neuen Softwareupdates erläutert werden [2]. Neben den sich beispielsweise ändernden Anforderungen des Datenschutzes durch die Erhebung von Fahrerdaten und deren Speicherung in Modellen, werden auch Änderungen in der Absicherung der gelernten und lernenden Systeme erwartet.
Zur tiefergehenden Bewertung der Gründe für die zu erwartenden Veränderungen in der Absicherung werden Grundlagen des maschinellen Lernens sowie der heutige Stand der Technik zur Absicherung von Fahrerassistenzsystemen (FAS) vorgestellt, um die hieraus resultierenden Problemstellungen für die Verwendung maschineller Lernverfahren im FAS-Bereich zu diskutieren. Darauf basierend werden Ansätze zur Lösung der Absicherungsproblematik vorgestellt und kritisch diskutiert.
Die vorliegende Betrachtung beschränkt sich auf maximal teilautomatisierte Systeme nach der Definition von [3].

2 Maschinelle Lernverfahren

Maschinelle Lernverfahren bilden eine Methode zum Aufbau künstlicher Intelligenz. Hierzu werden große Datenmengen (Ein- und Ausgangsgrößen) einer Problemstellung zur automatisierten Modellerstellung benutzt, um mittels dieser gelernten Modelle Vorhersagen anhand von neuen Eingangsdaten zu generieren [4].
Die benutzten Lernverfahren werden anhand unterschiedlicher Merkmale, wie beispielsweise die Art oder das Wissen über die vorherzusagende Ausgangsgröße, kategorisiert. Für die vorliegende Betrachtung eignet sich die Unterscheidung nach der Art der Ausgangsgröße bzw. Art der gewünschten Abbildung in Klassifikation, Regression und Clustering, wie in Abbildung 1 dargestellt. Bei einer Klassifikation wird die Abbildung von Eingangsgrößen auf begrenzte Räume der Ausgangsgröße, sogenannte Klassen, erlernt. Im Gegensatz zur Klassifikation ist die Ausgangsgröße bei einer Regression kontinuierlich, d.h. das Ziel ist die Abbildung der Eingangsgrößen auf eine kontinuierlich(e) Ausgangsgröße(n). Bei Klassifikation und Regression besitzt der Datensatz, der zum Trainieren der Modelle verwendet wird, eine direkte Zuordnung zwischen Eingangs- und Ausgangsgrößen. Wenn diese nicht vorhanden ist, ist es möglich, die Muster bzw. Ähnlichkeit innerhalb des Datensatzes festzustellen und basierend hierauf sogenannte Cluster zu erzeugen, die ähnliche Daten enthalten. Wird ein neuer Datenpunkt auf das gelernte Modell anwendet, wird dieser den identifizierten Clustern zugeordnet. Dieses Lernverfahren ist als Clustering bekannt [5].
Innerhalb der verschiedenen Lernverfahren werden unterschiedliche Algorithmen benutzt, die sich unter anderem hinsichtlich ihrer Komplexität und Leistungsfähigkeit unterscheiden. Die für die nachfolgende Betrachtung relevanten Algorithmen, deren Zuordnung zu den Lernverfahren sowie eine kurze Erläuterung sind Tabelle 1 zu entnehmen.

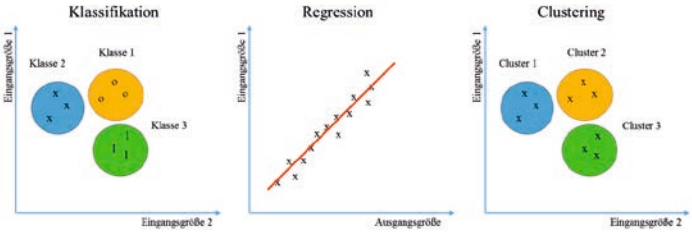

Abbildung 1: Übersicht der Lernverfahren

Tabelle 1: Erläuterung der angesprochenen Algorithmen

Algorithmus	Lernverfahren	Erläuterung
Decision Trees Unterkategorien: classification and regression trees (CART), generalized unbiased interaction detection and estimation (GUIDE)	Klassifikation, Regression	Entscheidungsbaum mit gelernten Entscheidungskriterien [6] Komplexität abhängig von der Anzahl der Verzweigungen (bzw. Knoten) [7]
Support Vector Machines	Klassifikation, Regression	Erlernen von (Hyper-)Trennebenen mit dem größten Abstand zwischen den Klassen z.T. Transformation der Datenbasis in einen höherdimensionalen Raum, um aus einem nicht-trennbaren Datensatz einen trennbaren Datensatz zu erhalten. (sogn. Kernel trick) [8]
Symbolic Regression	Regression	Erlernen einer Funktion $y=f(x)$, die den Zusammenhang zwischen abhängigen und unabhängigen Eingangsdaten x und Ausgangsdaten y repräsentiert. [9] Beispiel: $y = 0{,}45x_1+5{,}78x_2+1{,}3x_1x_2$
Neuronale Netze	Klassifikation, Regression, Clustering	Erlernen eines Netzes mit einer Eingangsschicht, einer Ausgangsschicht und ein oder mehreren Zwischenschichten (sogn. verdeckte Schichten) in Anlehung an die Funktionsweise des menschlichen Gehirns.[10]

3 Absicherung von Fahrerassistenzsystemen

3.1 Bestehende Absicherungsstrategien

Zu Beginn heutiger FAS-Absicherungsstrategien steht analog zu den Vorgaben der ISO 26262 [11], die sich mit dem Vorgehen zur Erreichung funktionaler Sicherheit beschäftigt und als Norm mit hoher Relevanz für die FAS-Absicherung identifiziert wurde [12], die Durchführung einer Risiko- bzw. Gefahrenanalyse, aus der sich Sicherheitsziele ableiten. Es besteht die Notwendigkeit, deren Erfüllung vor Auslieferung an den Kunden zu beweisen [13]. Eine Auflistung sowie ein Vergleich verschiedener Verfahren zur Risikoanalyse sind in [14] zu finden. Für die Beweisführung der Erfüllung der sich hieraus ableitenden Sicherheitsziele und des zuvor definierten Funktionsumfangs existieren verschiedene Verfahren. Es besteht einerseits die Möglichkeit, durch analytische Beweisführung die Erfüllung der Anforderungen zu zeigen. Dies erfordert das Verstehen des betrachteten Systems und der zugehörigen Systemgrenzen. Andererseits ist es möglich, durch zuvor definierte Testfälle, die Anforderungserfüllung zu beweisen. Die zur Testumsetzung eingesetzten Verfahren sind [15] und [16] zu entnehmen. Die Herausforderung in der Anwendung der Testverfahren liegt in der Definition der korrekten Testfälle, um einerseits alle Aspekte der Anforderungen abzuprüfen und anderseits die Anzahl der benötigten Testfälle möglichst gering zu halten. Aufgrund des zu erwartenden zeitlichen und finanziellen Aufwands der Testdurchführung ist es empfehlenswert, die analytische Beweisführung dem Abtesten der Anforderungen vorzuziehen.
Zusätzlich zur Beweisführung der Erfüllung der Sicherheitsziele wird im Bereich von aktiven FAS ebenfalls entweder ein analytischer Beweis oder die Erfüllung von Testfällen zum Nachweis der Kontrollierbarkeit benötigt [17].

3.2 Problematik der bestehenden Absicherungsstrategien im Hinblick auf maschinelles Lernen

Die Ergebnisse der Methoden des maschinellen Lernens, die für ihre hohe Vorhersageleistung bekannt sind, wie beispielsweise Support Vector Machines (SVM), sind in der Regel schwer für den Menschen interpretierbar. Komplementär hierzu zeichnen sich Methoden, die eine hohe Transparenz und Interpretierbarkeit besitzen, wie beispielsweise Decision Trees, normalerweise durch eine begrenzte Vorhersageleistung aus [18]. Durch die Wahl einer schwer interpretierbaren Methode wird die analytische Beweisführung der Erfüllung von Sicherheitszielen verhindert. Hierdurch steigen die resultierenden Kosten für die testfallgetriebene Absicherung, wenn dies durch den Einsatz von Black-Box-Modellen, aufgrund der hohen Anzahl an benötigten Testfällen, überhaupt möglich ist. Aus diesem Grund gewinnt die Gegenüberstellung der zu erwarteten Leistungsfähigkeit des Algorithmus zu den erwarteten Kosten in der Absicherung bereits in einer frühen Entwicklungsphase an Relevanz. Im folgenden Abschnitt werden daher bestehende Lösungsansätze vorgestellt, die einen Kompromiss zwischen der Leistungsfähigkeit und der Interpretierbarkeit eines Algorithmus beinhalten bzw. die Transparenz oder Leistungsfähigkeit bestehender Algorithmen erhöhen.
Weitere Lösungsmöglichkeiten zur Verringerung der Anzahl an benötigten Testfällen, stellen das Begrenzen des Arbeitsbereichs eines Black-Box-Algorithmus oder die

Definition von Worst-Case-Testfällen, analog zur Prüfung mechanischer Komponenten, dar. Auch diese Möglichkeiten werden jeweils in einem eigenen Unterkapitel im folgenden Abschnitt vorgestellt.
Neben der Problematik der Interpretierbarkeit von maschinell gelernten Algorithmen besteht im Fall des Einsatzes von weiterlernenden bzw. zeitvarianten Algorithmen die Herausforderung in der Absicherung des veränderten Modells während des Betriebs, d.h. nach Auslieferung an den Kunden. Eine Analyse der ISO26262 zu Vorgaben bezüglich Systemen, deren Parametrierung oder Struktur sich während des Betriebs verändert ergab, dass dies kein durch die Norm abgedeckten Fall darstellt. Diese Art der Algorithmen ist jedoch besonders im Hinblick auf die individuelle Adaption von Systemreaktionen an den Fahrer, wie ein Spurhalteassistent, der seine Lenkungs-Regelparameter an das individuelle Fahrerverhalten anpasst, interessant. Welche Ansätze zur Lösung dieser Problematik bestehen, wird ebenfalls im nächsten Abschnitt diskutiert.

4 Lösungsansätze zur Absicherung von Fahrerassistenzsystemen mit maschinell gelernten Algorithmen

4.1 Erreichen von Interpretierbarkeit bei gleichzeitiger Leistungsfähigkeit

Die grundsätzlichen Möglichkeiten, um einen interpretierbaren und gleichzeitig leistungsfähigen maschinell gelernten Algorithmus zu erhalten sind:

- Auswahl von Algorithmen mit hoher Grund-Interpretierbarkeit (White-Box-Algorithmen) und Verbesserung der Leistungsfähigkeit
- Auswahl von Algorithmen mit begrenzter Interpretierbarkeit (Grey- oder Black-Box-Algorithmen) und Erhöhung der Interpretierbarkeit

4.1.1 Verbesserung der Leistungsfähigkeit interpretierbarer Algorithmen

Algorithmenklassen, die für ihre hohe Transparenzeigenschaft auf Basis ihrer Grundstruktur bekannt sind, sind beispielsweise Decision Trees oder Symbolic Regression (SR) Algorithmen, die zur Klassifikation oder Regression genutzt werden [7]. Zur Verbesserung deren beschränkter Leistungsfähigkeit wurden folgende Ansätze identifiziert:

- Verwendung von komplexeren Modellen in den Knoten von Decision Trees [19]
- Verwendung von komplexeren Vorhersagemodellen für ganze Äste von Decision Trees, die für die Erfüllung von Sicherheitszielen nicht relevant sind
- Verbesserung der durch das einfache Design verbleibenden (Vorhersage-)Fehler durch komplexere Modelle [7]

Für die Umsetzung des ersten Ansatzes wird ein Entscheidungsbaum trainiert, der eine begrenzte Anzahl an Knoten besitzt, um die Interpretierbarkeit des Baums zu erhalten. Zur Verbesserung der Leistungsfähigkeit werden, anders als bei einfachen Entscheidungsbäumen wie CART (classification and regression tree), innerhalb der Knoten komplexere Modelle eingesetzt, falls sich keine einfache Trennung einer Variablen für einen Knoten

findet. Dabei ist wichtig, dass auch die komplexeren Modelle interpretierbar gewählt werden, wie beispielsweise im Rahmen des GUIDE Algorithmus (generalized unbiased interaction detection and estimation). Die hierin verwendeten komplexeren Modelle bestehen in diesem Fall aus einem linearen Zusammenhang zweier Variablen. Neben dem Verwenden von komplexeren Modellen in den Knoten der Bäume ist es ebenfalls im Fall von Regressions-Entscheidungsbäumen möglich, in den „Blättern" der Vorhersagemodelle komplexere Algorithmen zu verwenden. [20] Ergebnisse hinsichtlich der Verbesserung der Leistungsfähigkeit zwischen einem interpretierbaren Entscheidungsbaum mit und ohne Verbesserung der Leistungsfähigkeit durch den Einsatz komplexerer Vorhersagemodelle finden sich in [19, 20].
Beim Ersatz ganzer Teilbereiche eines Entscheidungsbaums nach einem Knoten durch komplexere, leistungsfähigere Algorithmen besteht die Herausforderung in der Identifikation der ersetzbaren Bereiche. Eine Möglichkeit liegt in der Analyse der Sicherheitsziele des Systems und der Identifikation der Bereiche des Entscheidungsbaums, die zu deren Erfüllung nicht relevant sind. Dieser Ansatz wurde bisher nicht in der Literatur gefunden und bedarf weiterer Untersuchungen hinsichtlich der erreichbaren Leistungsfähigkeit.
Für die Verbesserung der Leistungsfähigkeit von interpretierbaren Symbolic-Regression-Algorithmen, die sich zum Beispiel auf die Verwendung von Multiplikation, Addition und Subtraktion der Einzelfunktionen beschränken, werden die verbleibenden (Vorhersage-) Fehler durch komplexere, nicht interpretierbare Modelle wie beispielsweise SVM minimiert. Dabei wird der Geltungsbereich der SVM auf den maximal aufgetretenen Fehler bzw. der maximal akzeptierten Verschlechterung der Ergebnisse des Symbolic-Regression-Algorithmus, bei der keine Verletzung der Sicherheitsziele vorliegt, beschränkt. Hierdurch bleibt die Gesamt-Interpretierbarkeit der Symbolic-Regression erhalten. Dieser Ansatz wurde als leistungsfähiger und besser interpretierbar als der GUIDE-Algorithmus bewertet, wobei jedoch eine höhere Rechenleistung und –zeit zur Modellerstellung benötigt wird [7]. Die Verbesserung der Leistungsfähigkeit beim Einsatz anderer Algorithmen als SVM sowie deren Auswirkung auf die Rechenleistung ist zu überprüfen.

4.1.2 Verbesserung der Interpretierbarkeit leistungsfähiger Algorithmen

Durch die Beschränkung des menschlichen Vorstellungsvermögens auf wenige Dimensionen ist es nicht möglich, ein Verständnis für die Vorgänge und Zusammenhänge in einem hochdimensionalen, komplexen Algorithmus zu erhalten [21]. Daher beruhen die Ansätze zur Verbesserung der Interpretierbarkeit eines solchen Algorithmus auf dem Grundgedanken, dessen Dimensionalität zu reduzieren:

- Herunterbrechen eines hochdimensionalen Algorithmus auf mehrere niedrigdimensionale Teilmodelle [18]
- Vereinfachung eines Algorithmus durch Reduzierung auf die stärksten Merkmale und Analysieren des Zusammenhangs einzelner Merkmale zur Ausgabegröße
- Niedrigdimensionale Visualisierung der Vorgänge in einem komplexen Algorithmus zum Verständnis der Entscheidungen [22]
- Implementierung eines Zuverlässigkeitsmaßes

Der erste Ansatz wird bisher für Klassifikationen angewendet. Dabei wird anstelle des hochdimensionalen Eingangsdatenbereichs lediglich eine Teilmenge von maximal zwei bis drei Dimensionen verwendet. Auf dieser Teilmenge werden Modelle trainiert, die durch ihre niedrige Dimensionalität menschlich verständlich visualisierbar sind. Hierdurch ist es möglich, jedes Teilmodell abzusichern. Die einzelnen Teilmodelle werden zu einem Gesamtmodell kombiniert, um die Leistungsfähigkeit der einzelnen Modelle zu verbessern. Allerdings wird bei diesem Ansatz angenommen, dass sich das hochdimensionale Problem in niedrigdimensionale Teile trennen lässt [23]. Ein Vergleich der Leistungsfähigkeit zwischen einem einzelnen hochdimensionalen Modell und der Kombination der einzelnen Teilmodelle ist in [18, 23] zu finden.
Der Ansatz der Dimensionsreduktion auf die wichtigsten Merkmale eines auf einem hochdimensionalen Eingangsraum gelernten Modells und die anschließende zwei- bis dreidimensionale Analyse der Zusammenhänge zwischen Eingangsmerkmalen und Ausgabegrößen wurde bisher nicht in der Literatur gefunden und bedarf weiterer Untersuchungen hinsichtlich der erreichbaren Interpretierbarkeit bei der durch die Dimensionsreduktion verminderten Leistungsfähigkeit. Der Ansatz ist prinzipiell für alle Lernverfahren anwendbar.
Der generelle Ansatz der Visualisierung der Vorgänge in einem komplexen Algorithmus zum Verständnis von dessen Entscheidungen bzw. zum Erhöhen von dessen Interpretierbarkeit wird im Zusammenhang mit Neuronalen Netzen vorgestellt [22]. Es wird eine Methode zur Visualisierung von Strukturen in hochdimensionalen Datensätzen angewendet, die die Struktur der Daten nicht verändert, wie beispielsweise Neuronale Netze, sondern die Struktur der Daten in einem niedrigdimensionalen Raum (zwei oder drei Dimensionen) darstellt. Dabei sind unterschiedliche Verfahren bekannt, die beispielsweise darauf beruhen, dass Datenpunkte, die sich in einem hochdimensionalen Raum nahe sind, im niedrigdimensionalen Raum ebenfalls mit geringem Abstand zueinander dargestellt werden. Dabei ist es wichtig, globale und lokale Strukturen zu konservieren. Die Visualisierungsverfahren besitzen die Möglichkeit, beispielsweise auf verdeckte Schichten von neuronalen Netzen angewendet zu werden, um die Vorgänge innerhalb der Netze menschlich interpretierbarer darzustellen [21, 22]. Ein Beispiel für die Anwendung eines Verfahrens zur Strukturvisualisierung bzw. Dimensionsreduktion wird in Abbildung 2 gegeben. Dabei wird ein graphenbasiertes Verfahren auf die versteckte Schicht eines Neuronalen Netzes, das zur Klassifikation in zehn Klassen gelernt wurde, angewendet. Die Einfärbung der Datenpunkte zeigt ihre Klassenzugehörigkeit. In der Eingangsschicht ist es zwar möglich, Schwerpunkte der einzelnen Farbklassen zu identifizieren, eine deutliche Trennung der Bereiche ist jedoch nicht auszumachen. In der versteckten Schicht wird diese Trennung der einzelnen Farbbereiche (= Klassen) nach den Transformationen durch das neuronale Netz bereits deutlicher. Allerdings sind weitere Untersuchungen notwendig, ob eine analytische Beweisführung möglich ist, die die benutzte Dimensionsreduktion bzw. Visualisierungsmethode aller zur Absicherung relevanten Strukturen konserviert. Falls diese Beweisführung gelingt, ist zu überprüfen, ob das hieraus gewonnene Verständnis über die Trennbereiche bzw. Vorgehensweise des Algorithmus ausreicht, um die Erfüllung von Sicherheitszielen zu beweisen.

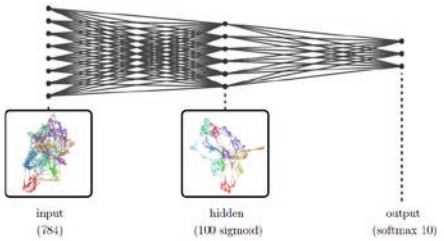

Abbildung 2: Anwendung einer Methode zur Dimensionsreduktion auf ein Neuronales Netz [22]

Durch die Implementierung eines Zuverlässigkeitsmaßes, das bei fehlenden, stark verrauschten oder unbekannten Eingangsinformationen eine verminderte Zuverlässigkeit des gelernten Modells bzw. der hieraus resultierenden Ausgabe im Vergleich zu vollständig vorliegenden Eingangsinformationen angibt, ist es ebenfalls möglich, die Interpretierbarkeit von Algorithmen zu erhöhen. Der Algorithmus selbst erlernt, was er weiß, und besitzt hierdurch die Möglichkeit auszugeben, was er nicht weiß („I don't know"- Ausgabe). Dieser Ansatz wurde unter anderem in [24 bis 26] vorgestellt. Eine andere Möglichkeit der Zuverlässigkeitsangabe besteht bei Klassifikationen und Clustering-Lernverfahren darin, die Wahrscheinlichkeit der Zugehörigkeit einer neuen Eingabe zu allen Ausgabeklassen bzw. -clustern anzugeben [27]. Hierdurch wird erkennbar, inwieweit ähnliche Daten bereits im gelernten Modell enthalten sind bzw. wie stark die Zuordnung zu der bestehenden Datenbasis stattfindet. Beide Möglichkeiten eignen sich alleinstehend nicht zum Beweis von Sicherheitszielen, sind aber als Unterstützung zum besseren Verständnis des Algorithmus hilfreich und mit den anderen Ansätzen zu einer ganzheitlichen Absicherungsstrategie zu kombinieren.

4.2 Begrenzung des Arbeitsbereichs

Durch Begrenzung des Wertebereichs von Ein-, Zustands- und Ausgangsgrößen auf Werte des Trainingsdatensatzes sowie eine dynamische Begrenzung der Änderungsrate dieser Größen, ist es möglich, die Absicherbarkeit eines Algorithmus zu erhöhen [7]. Hierdurch sind die Arbeitsgrenzen des Algorithmus fixiert und die Absicherung hat lediglich innerhalb der Grenzen zu erfolgen. Dies reduziert die Anzahl an benötigten Testfällen zur Absicherung, falls der Algorithmus an sich nicht interpretierbar ist.

4.3 Definition von Worst-Case-Testfällen

Um die Anzahl an benötigten Testfällen zu verringern, besteht eine Möglichkeit darin, Worst-Case-Testfälle zu identifizieren, die eine Vielzahl von anderen Testfällen, die eine geringere Auswirkung beisitzen, ersparen. Eine Analogie hierzu ist in der thermischen Belastungsprüfung von physikalischen Bauteilen zu finden. Anstatt jeden Temperaturschritt einzeln zu überprüfen, wird lediglich der Bereich nahe der ungefähr bekannten, kritischen Temperatur(en) abgeprüft. Zur Übertragung auf die Testfallgenerierung von maschinellen Lernverfahren ist die Analyse, wie Fehlverhalten gezielt erzeugt wird,

notwendig. Hierzu gilt es die Eigenschaften und die Schwächen der unterschiedlichen Algorithmentypen zu identifizieren, analog zu einer grundlegenden Analyse von physikalischen Materialeigenschaften. Eine solche Analyse hinsichtlich der möglichen auftretenden Probleme des Algorithmentyps wurde durch [28] geleistet, deren Fokus auf Reinforcement-Lernverfahren liegt. Die identifizierten Schwächen gilt es anwendungsfallunabhängig auf ihre Robustheit im Bezug auf verschiedene Einflussgrößen hin zu überprüfen. Hierdurch ist es bereits möglich, eine Menge an anwendungsfallunabhängigen Basis-Testfällen zu definieren. Mit dem generierten Wissen wird anschließend eine anwendungsfallabhängige Risiko- und Gefahrenanalyse durchgeführt, um abzuprüfen, welche der identifizierten Schwächen nicht und welche im Besonderen auftreten. Zusätzlich werden hierdurch eventuelle anwendungsfallspezifische Risiken identifiziert und hierauf basierend die Testfallentwicklung verfeinert. Dieses Verfahren wurde im Zusammenhang mit maschinellem Lernen noch nicht in der Literatur beschrieben und bedarf weiterer Untersuchungen hinsichtlich der Anwendbarkeit und beispielsweise der Möglichkeit, algorithmenspezifische Schwächen zu identifizieren.

4.4 Absicherung von im Betrieb lernenden Algorithmen

Zur Gewährleistung der Sicherheit von im Betrieb lernenden Strukturen wurden zwei grundsätzliche Ansätze in der Literatur identifiziert:

- Begrenzung des Arbeitsbereichs und Absicherung aller möglichen Zustände innerhalb dieses Arbeitsbereichs
- Absicherung bei jeder Änderung des Algorithmus

Die erste Möglichkeit wurde bereits in Kapitel 4.2. kurz erläutert. Eine beispielhafte Umsetzung der Begrenzung speziell für online lernende Algorithmen findet sich in [29]. Das Problem dieses Lösungsansatzes stellt die Beschränkung des Vorteils der individuellen Anpassung der lernenden Struktur dar [30].
Zur Umsetzung der zweiten Möglichkeit wurden folgende Ansätze identifiziert:

- Überwachen des Auftretens möglicher Fehler und im Fehlerfall Zurückwechseln auf ein Back-Up-Modell [31]
- Echtzeitbasierte Validierung [32]
- Selbstqualifikation des Algorithmus

Der erste Ansatz basiert auf einer dauerhaften und robusten Überwachung von möglichen auftretenden Fehlern und dem Wechsel auf ein sich nicht änderndes, bereits im Vorfeld abgesichertes Modell [31]. Die Herausforderung liegt hierbei in der Definition aller möglichen auftretenden Fehlerfälle und ihrer zuverlässigen Identifikation. Vor allem im Hinblick auf vielfältige, voneinander abhängige Umwelteinflüsse, wie sie im Straßenverkehr auftreten, wird dies problematisch.
Im Rahmen der echtzeitbasierten Validierung wird das Einhalten eines sicheren Bereichs, der durch Anforderungen definiert wurde, ständig überwacht. Hierauf basierend werden entweder vorgeschlagene Änderung des Algorithmus angenommen oder verworfen [32]. Die Herausforderung des zweiten Ansatzes liegt in der Definition aller benötigten Anforderungen in der Weise, dass sie online überprüfbar sind. Beide Ansätze benötigen darüber hinaus eine hohe Rechenleistung im Fahrzeug für die Durchführung der Online-

Überprüfung. Eine Möglichkeit, dieser Problematik zu begegnen, liegt in der Beschränkung der Änderung des Algorithmus auf Stillstandszeiten des Fahrzeugs, während der mehr Rechenzeit und –leistung als im Fahrbetrieb zur Verfügung steht.
Der Ansatz der Selbstqualifikation eines Algorithmus beruht auf der Verbindung der Idee des bereits in Kapitel 4.1.2 vorgestellten Ansatzes des Zuverlässigkeitsmaßes mit einer Überwachung und Beschränkung der Änderungsdynamik des Algorithmus. Hierdurch wird es dem Algorithmus ermöglicht, neben der Optimierung der eigentlichen Leistung, eine zusätzliche Sicherheitsbewertung der geplanten Änderung vorzunehmen. Allerdings genügt dieser Ansatz nicht als alleiniges Sicherheitskriterium, da unsichere Zustände auch mit einer geringeren Änderungsdynamik erreicht werden. Ein Beispiel hierfür stellt eine individuell an den Fahrer angepasste Zeitlücke eines Adaptive Cruise Control Systems dar, die eine kritische Zeitlücke langsam unterschreitet, da der Fahrer einen unterhalb der kritischen Zeitlücke liegenden Abstand zum Vorderfahrzeug präferiert. Der Ansatz der ganzheitlichen Selbstqualifikation wurde nicht in der Literatur identifiziert und bedarf weiterer Untersuchung beispielsweise hinsichtlich seiner Erweiterungsfähigkeit und weiteren Methoden zur Selbstqualifikation.

5 Zusammenfassung und Ausblick

Die verschiedenen Verfahren zur Verminderung der Testfallanzahl für gelernte Algorithmen sowie die Lösungsansätze für den Bereich der online lernenden Algorithmen wurden diskutiert sowie die weiteren Forschungsfragen adressiert.
Im Fall von gelernten Algorithmen wurden drei generelle Lösungsmöglichkeiten identifiziert. Das Erhalten eines interpretierbaren und gleichzeitig leistungsfähigen Algorithmus entweder durch die Erhöhung der Interpretierbarkeit bei leistungsfähigen Algorithmen oder die Erhöhung der Leistungsfähigkeit bei interpretierbaren Algorithmen stellt die erste Lösungsmöglichkeit dar. Die einzelnen Ansätze dieser Möglichkeiten wurden teilweise prototypisch für verschiedene Problemstellungen angewendet. Als zweite Möglichkeit wurde die Begrenzung des Arbeitsbereichs identifiziert. Diese Lösungsmöglichkeit besitzt einen geringen Aufwand, allerdings wird der Nutzen des gelernten Modells stark eingeschränkt. Der dritte Ansatz sieht die Definition von Worst-Case-Testfällen vor, die die Testfälle, die nicht die schlimmsten anzunehmenden Auswirkungen besitzen, redundant werden lassen. Hierzu wurden noch keine Anwendungen identifiziert. Für online lernende Algorithmen wurden neben der Begrenzung des Arbeitsbereichs die Möglichkeiten der echtzeitbasierten Fehlerüberwachung, der Runtime Verification & Validation und der Selbstqualifikation des Algorithmus vorgestellt.
Aus der vorherigen Betrachtung lässt sich schließen, dass bereits im Rahmen der Algorithmenauswahl für FAS eine Absicherungsstrategie zu entwickeln bzw. die Aufwendung zur Absicherung bereits als Kriterium bei der Algorithmenauswahl heranzuziehen ist. Zusätzlich gilt zu analysieren, ob die benötigten Aufwendungen nicht den erwarteten Nutzen des Einsatzes des maschinell gelernten Algorithmus übersteigen. Eine Kombination der vorgestellten Werkzeuge für ein ganzheitliches Absicherungskonzept ist zu erwarten.

Literaturangaben

[1] Ambroggi, L. de: Artificial Intelligence Systems for Autonomous Driving On the Rise, IHS Says, 2016, abgerufen am: 30.11.2016
[2] The Tesla Team: Upgrading Autopilot: Seeing the World in Radar, 2016. https://www.tesla.com/de_DE/blog/upgrading-autopilot-seeing-world-radar?redirect=no, abgerufen am: 12.12.2016
[3] Gasser, T. M., Arzt, C., Ayoubi, M., Bartels, A., Bürkle, L., Eier, J., Flemisch, F., Häcker, D., Hesse, T., Huber, W., Lotz, C., Maurer, M., Ruth-Schumacher, S., Schwarz, J. u. Vogt, W.: Rechtsfolgen zunehmender Fahrzeugautomatisierung. Gemeinsamer Schlussbericht der Projektgruppe. Berichte der Bundesanstalt für Strassenwesen - Fahrzeugtechnik (F), Bd. 83. Bremerhaven: Wirtschaftsverl. NW Verl. für neue Wissenschaft 2012
[4] Copeland, M.: What's the Difference Between Artificial Intelligence, Machine Learning, and Deep Learning?, 2016. https://blogs.nvidia.com/blog/2016/07/29/whats-difference-artificial-intelligence-machine-learning-deep-learning-ai/, abgerufen am: 12.12.2016
[5] Suthaharan, S.: Machine learning models and algorithms for big data classification. Thinking with examples for effective learning. Integrated series in information systems, Volume 36. New York: Springer Science+Business Media 2016
[6] Mitchell, T. M.: Machine learning. McGraw-Hill series in computer science: Artificial Intelligence. New York, NY: McGraw-Hill 1997
[7] Otte, C.: Safe and Interpretable Machine Learning: A Methodological Review. In: Moewes, C. u. Nürnberger, A. (Hrsg.): Computational Intelligence in Intelligent Data Analysis. Studies in Computational Intelligence. Berlin, Heidelberg: Springer Berlin Heidelberg 2013, S. 111–122
[8] Markowetz, F.: Klassifikation mit Support Vector Machines. Genomische Datenanalyse 2003. Berlin
[9] Flasch, O.: A friendly introduction to rgp. Gummersbach 2014
[10] Scherer, A.: Neuronale Netze. Grundlagen und Anwendungen. Computational Intelligence. Wiesbaden: Vieweg+Teubner Verlag 1997
[11] International Organization for Standardization: ISO 26262:2011. Road vehicles : Functional safety. Geneva: International Organization for Standardization 2011
[12] Weitzel, A., Winner, H., Peng, C., Geyer, S., Lotz, F. u. Sefati, M.: Absicherungsstrategien für Fahrerassistenzsysteme mit Umfeldwahrnehmung. Berichte der Bundesanstalt für Strassenwesen - Fahrzeugtechnik (F), Bd. 98. Bremerhaven: Wirtschaftsverl. NW Verl. für neue Wissenschaft 2014
[13] Bundesministerium der Justiz und für Verbraucherschutz: Verordnung über die Zulassung von Fahrzeugen zum Straßenverkehr (Fahrzeug-Zulassungsverordnung – FZV). 2011
[14] Ständer, T.: Eine modellbasierte Methode zur Objektivierung der Risikoanalyse nach ISO 26262. 2011
[15] Berg, G., Nitsch, V. u. Färber, B.: Vehicle in the Loop. In: Winner, H., Hakuli, S., Lotz, F. u. Singer, C. (Hrsg.): Handbook of Driver Assistance Systems: Basic

Information, Components and Systems for Active Safety and Comfort. Cham: Springer International Publishing 2016, S. 199–210
[16] Rüger, F., Sieber, M., Siegel, A., Siederberger Karl-Heinz u. Färber. Berthold: Kontrollierbarkeitsbewertung von FAS der aktiven Sicherheit in frühen Phasen des Entwicklungsprozesses mit dem Vehicle in the Loop (VIL). In: 9. Workshop Fahrerassistenzsysteme, S. 137–146
[17] Weitzel, D. A.: Objektive Bewertung der Kontrollierbarkeit nicht situationsgerechter Reaktionen umfeldsensorbasierter Fahrerassistenzsysteme. VDI-Verlag 2013
[18] Nusser, S.: Robust Learning in Safety-Related Domains. Machine Learning Methods for Solving Safety-Related Application Problems. 2009
[19] Loh, W.-Y.: Regression by Parts: Fitting Visually Interpretable Models with GUIDE, S. 447–469
[20] Loh, W.-Y.: Classification and regression trees. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 1 (2011) 1, S. 14–23
[21] Olah, C.: Visualizing MNIST: An Exploration of Dimensionality Reduction, 2014. http://colah.github.io/posts/2014-10-Visualizing-MNIST/
[22] Olah, C.: Visualizing Representations: Deep Learning and Human Beings, 2015. http://colah.github.io/posts/2015-01-Visualizing-Representations/
[23] Nusser, S., Otte, C., Hauptmann, W. u. Kruse, R.: Learning verifiable ensembles for classification problems with high safety requirements. Intelligent Soft Computation and Evolving Data Mining: Integrating Advanced Technology (2010), S. 405–431
[24] Zhang, C. u. Chaudhuri Kamalika: A Potential-based Framework for Online Learning with Mistakes and Abstentions. In: Lee, D. D., Sugiyama, M., Luxburg, U. V., Guyon, I. u. Garnett, R. (Hrsg.): Neural Information Processing Systems 29. Barcelona 2016
[25] Li, L., Littman, M. L. u. Walsh, T. J.: Knows what it knows: a framework for self-aware learning. Proceedings of the 25th international conference on Machine learning. 2008, S. 568–575
[26] Demaine, E. D. u. Zadimoghaddam, M.: Learning disjunctions: near-optimal trade-off between mistakes and I don't knows. Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms. 2013, S. 1369–1379
[27] Cord, M. u. Cunningham, P.: Machine Learning Techniques for Multimedia: Case Studies on Organization and Retrieval. Springer Berlin Heidelberg 2008
[28] Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J. u. Mané, D.: Concrete Problems in AI Safety. CoRR abs/1606.06565 (2016)
[29] Gillula, J. H. u. Tomlin, C. J.: Guaranteed Safe Online Learning via Reachability: tracking a ground target using a quadrotor. IEEE International Conference on Robotics and Automation (ICRA), 2012. 14-18 May 2012, Saint Paul, Minnesota, USA. Piscataway, NJ: IEEE 2012, S. 2723–2730
[30] Dahm, W.: Perspectives on Verification and Validation in Complex Adaptive Systems. Workshop on Verification and Validation in Computational Science. Notre Dame University, Indiana 2011
[31] Isermann, R.: Fault-diagnosis systems. An introduction from fault detection to fault tolerance. Berlin, New York: Springer 2006

[32] Tamura, G., Villegas, N., Müller, H., Sousa, J., Becker, B., Karsai, G., Mankovskii, S., Pezzè, M., Schäfer, W., Tahvildari, L. u. Wong, K.: Towards Practical Runtime Verification and Validation of Self-Adaptive Software Systems. In: Lemos, R. de, Giese, H., Müller, H. u. Shaw, M. (Hrsg.): Software Engineering for Self-Adaptive Systems II. Lecture Notes in Computer Science. Springer Berlin Heidelberg 2013, S. 108–132