

Anwendung eines Grundvokabulars für den szenarienbasierten Testansatz automatisierter Fahrfunktionen anhand eines Beispiels

Markus Steimle; Gerrit Bagschik; Till Menzel;
Jan Timo Wendler; Markus Maurer

Technische Universität Braunschweig, Institut für Regelungstechnik
{steimle,bagschik,menzel,wendler,maurer}@ifr.ing.tu-bs.de

Kurzfassung

Steimle u. a. (2018) begründen die Notwendigkeit einer einheitlichen, deutschen Terminologie für den szenarienbasierten Testansatz automatisierter Fahrfunktionen. Durch den Vorschlag eines Grundvokabulars und einer Gliederung haben sie einen Beitrag zur Etablierung einer konsistenten Begrifflichkeit für eine solche Terminologie geleistet. Zur Verdeutlichung der definierten Begriffe werden die Begriffsdefinitionen in diesem Beitrag mit einem durchgängigen Beispiel erläutert und in Kontext gesetzt.

1 Einleitung

Die ersten beiden Absätze dieser Einleitung wurden bereits in Steimle u. a. (2018) veröffentlicht.

„Fahrerassistenzsysteme und automatisierte Fahrfunktionen sind zentrale Bestandteile der Forschung im Automobilbereich (Bundesministerium für Verkehr und digitale Infrastruktur, 2017). So hat die Bundesregierung bereits im September 2015 die ‚Strategie automatisiertes und vernetztes Fahren‘ beschlossen (Bundesministerium für Verkehr und digitale Infrastruktur, 2015, S. 12). Durch diese Strategie soll Deutschland seine Position als Leitanbieter automatisierter Fahrzeugtechnologien weiter festigen und sich als Leitmarkt etablieren. Um dieses Ziel zu forcieren, initiierte das Bundesministerium für Verkehr und digitale Infrastruktur eine ausgeprägte Förderlandschaft für deutsche Institutionen und Unternehmen. (Bundesministerium für Verkehr und digitale Infrastruktur, 2017)

Für die Markteinführung einer automatisierten Fahrfunktion stellt nicht nur die Entwicklung der Funktion eine Herausforderung dar, sondern insbesondere deren Absicherung und Freigabe (Gasser u. a., 2015, S. 12). Eine herkömmliche, streckenbasierte Absicherung ist aufgrund der zu fahrenden Kilometer aus Zeit- und Kostengründen nicht anwendbar und wurde von Wachenfeld & Winner (2015) als Freigabefälle für das autonome Fahren formuliert (Wachenfeld & Winner, 2015, S. 462 f.). Zur Freigabe automatisierter Fahrfunktionen sind folglich neue Methoden nötig. Vielversprechend erscheint der szenarienbasierte Testansatz (Schuldt, 2017, S. 210). Durch diesen können zur Überprüfung der Fahrfunktion relevante Anwendungsszenarien entlang des Entwicklungsprozesses detailliert sowie gezielt variiert und durchgeführt werden. Deshalb beschäftigen sich derzeit verschiedene Projekte mit der szenarienbasierten Absicherung automatisierter Fahrfunktionen, wie z. B. die Forschungsprojekte PEGASUS (PEGASUS Projektbüro, 2018) und aFAS (aFAS Projektbüro, 2018) sowie die Projekte ENABLE-S3 (ENABLE-S3 PROJECT COORDINATION, 2018) und Ko-HAF (Ko-HAF Projektbüro, 2018).“ (Steimle u. a., 2018, S. 1 f.)

Nach Steimle u. a. (2018, S. 2) bildet zum Erreichen der Ziele aus der „Strategie automatisiertes und vernetztes Fahren“ und zur Sicherstellung vergleichbarer Ergebnisse verschiedener Herangehensweisen szenarienbasierter Test- und Freigabeprozesse eine einheitliche, deutsche Terminologie die Basis. Dazu haben Steimle u. a. (2018) ein Grundvokabular und eine Gliederung für eine einheitliche, deutsche Terminologie für den szenarienbasierten Testansatz automatisierter Fahrfunktionen vorgeschlagen. In diesem Beitrag wird das Grundvokabular aus Steimle u. a. (2018) anhand eines durchgängigen Beispiels, die Entwicklung und Absicherung eines ACC-Systems, erläutert und in Kontext gesetzt.

2 Anwendung des Grundvokabulars anhand eines Beispiels

In diesem Abschnitt werden die Begriffsdefinitionen aus Steimle u. a. (2018) im Rahmen der Entwicklung und Absicherung eines ACC-Systems exemplarisch erläutert und gegeneinander abgegrenzt. Es werden selektiv beispielhafte Inhalte aufgeführt, die die Begriffe verdeutlichen und gegeneinander abgrenzen sowie im weiteren Verlauf des Beispiels für andere Definitionen relevant sind, sodass kein Anspruch auf Vollständigkeit des Beispiels besteht. Zur Kennzeichnung der Stellen, an denen die jeweiligen Begriffe aus Steimle u. a. (2018) exemplarisch

erläutert werden, sind diese an den entsprechenden Stellen im Text fett gedruckt.

2.1 Szenarienerstellung

In diesem Unterabschnitt werden die Begriffe der Szenarienerstellung aus Steimle u. a. (2018, Abschnitt 2.1) im Rahmen der Entwicklung und Absicherung eines ACC-Systems exemplarisch erläutert und gegeneinander abgegrenzt.

Nach Bagschik u. a. (2017, S. 2) können durch die Definition von Szenarien auf unterschiedlichen Abstraktionsebenen „Szenarien von Beginn des Entwicklungsprozesses an bereits in der Konzeptphase identifiziert werden und im weiteren Verlauf detailliert und konkretisiert werden.“ In diesem Beitrag werden exemplarisch jeweils ein funktionales, ein logisches und ein konkretes Szenario erstellt. Für eine genaue Aufteilung der funktionalen, logischen und konkreten Szenarien auf die verschiedenen Zeitpunkte im Entwicklungsprozess sei auf Bagschik u. a. (2017) verwiesen.

Abbildung 1 zeigt die Begriffe im Zusammenhang mit einem Szenario und deren Relationen als UML-Diagramm. Abbildung 2 zeigt die Begriffe der verschiedenen Abstraktionsebenen von Szenarien und deren Relationen als UML-Diagramm.

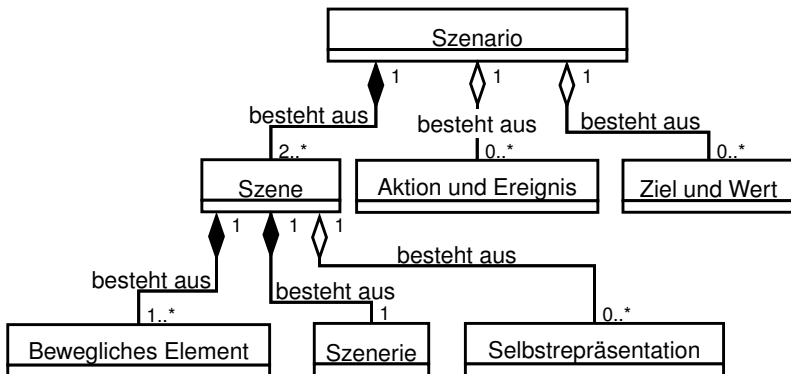


Abbildung 1: Begriffe eines Szenarios dargestellt als UML-Diagramm (Steimle u. a., 2018, S. 5)

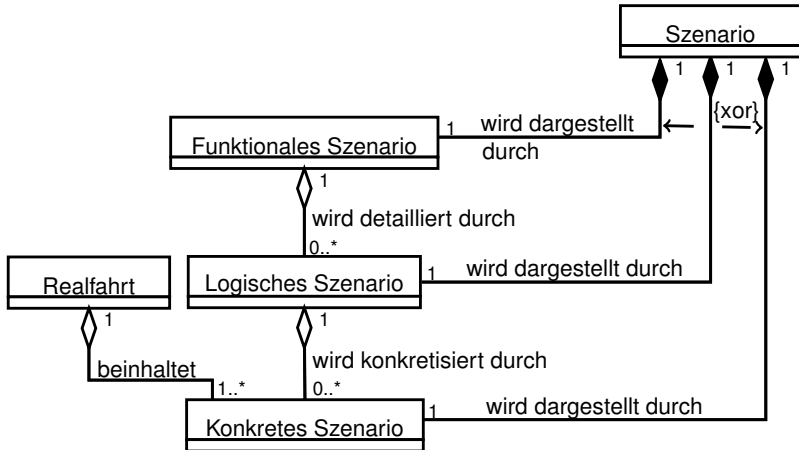


Abbildung 2: Begriffe in Verbindung zu verschiedenen Abstraktionsebenen von Szenarien dargestellt als UML-Diagramm (Steimle u. a., 2018, S. 6)

Ein mögliches **funktionales Szenario**, das sprachlich beschrieben wird, lautet, dass das Fahrzeug mit dem Testobjekt mit einer bestimmten Startgeschwindigkeit auf dem rechten Fahrstreifen ohne Verkehr im Sommer auf einer zweistreifigen Autobahn mit einer Kurve fährt. Nach einer gewissen Zeit aktiviert der Fahrer das ACC-System, indem er eine Wunschgeschwindigkeit vorgibt.

Im nächsten Schritt werden die funktionalen Szenarien durch logische Szenarien detailliert. Aus dem vorgestellten funktionalen Szenario wird exemplarisch folgendes **logische Szenario**, welches als „ACC-Freifahrt_logisch“ bezeichnet wird, abgeleitet. Aufgrund der Übersichtlichkeit wird ein sehr limitierter Parameterraum genutzt. In der Realität sind deutlich mehr Parameter notwendig. Abbildung 3 zeigt dieses logische Szenario als grafische Übersicht. Die **Szenarie** sieht wie folgt aus. Die Autobahn besteht aus zwei Fahrstreifen. Ein Fahrstreifen besitzt eine Breite von 2,5 m bis 3,75 m und einen Kurvenradius von 300 m bis 900 m. Die Umgebungstemperatur liegt zwischen 15 °C und 25 °C. Das Fahrzeug mit dem Testobjekt ist ein **bewegliches Element**, das mit einer Startgeschwindigkeit zwischen 80 km/h und 130 km/h fährt. Weitere bewegliche Elemente sind nicht vorhanden. Die **Selbstrepräsentation** des eigenen Fahrzeugs beinhaltet die Funktionsbereitschaft

<u>Logisches Szenario „ACC-Freifahrt logisch“</u>	
<u>Szenerie:</u>	
Breite Fahrstreifen:	[2,5..3,75] m
Kurvenradius:	[300..900] m
Umgebungstemperatur:	[15..25] °C
<u>Bewegliche Elemente:</u>	
Fahrzeug mit dem Testobjekt; Startgeschwindigkeit:	[80..130] km/h
<u>Selbstrepräsentation:</u>	
Funktionsbereitschaft ACC-System gemäß Spezifikation	
<u>Aktionen / Ereignisse:</u>	
Aktivierung ACC-System mit Wunschgeschwindigkeit:	[70..200] km/h
Zeitpunkt:	1 s
<u>Individuelle Ziele und Werte:</u>	
Ziel des Systems: Regelung auf die Wunschgeschwindigkeit:	[70..200] km/h
Ziel des Fahrers: Erreichen des Reiseziels:	Braunschweig
Wert des Fahrers: Gewünschter Abstand zu einem vorausfahrenden Fahrzeug: [gering, mittel, groß]	

Abbildung 3: Übersicht des abgeleiteten logischen Szenarios (detaillierte Beschreibung des Beispiels im Text)

des ACC-Systems gemäß Spezifikation. 1 s nach Start des Szenarios aktiviert der Fahrer das ACC-System durch Vorgabe einer Wunschgeschwindigkeit zwischen 70 km/h und 200 km/h. Ein individuelles **Ziel** des Systems stellt die Regelung auf diese Wunschgeschwindigkeit dar. Ein individuelles **Ziel** des Fahrers stellt das Erreichen des Reiseziels Braunschweig dar. Ein individueller **Wert** des Fahrers ist die Wahl des Abstands (gering, mittel, groß) zu einem vorausfahrenden Fahrzeug.

Im nächsten Schritt werden die logischen Szenarien durch konkrete Szenarien konkretisiert. Aus dem vorgestellten logischen Szenario wird exemplarisch folgendes **konkrete Szenario**, welches als „ACC-Freifahrt_konkret“ bezeichnet wird, abgeleitet. Abbildung 4 zeigt dieses konkrete Szenario als grafische Übersicht. Die **Szenerie** besteht aus einer Autobahn mit zwei Fahrstreifen mit jeweils einer Breite von 3,2 m. Die Autobahn besitzt einen Kurvenradius von 500 m. Die Umgebungstemperatur beträgt 20 °C. Ein **bewegliches Element** ist das Fahrzeug mit dem Testobjekt, das mit einer Startgeschwindigkeit von 100 km/h fährt. Weitere bewegliche Elemente sind nicht vorhanden. Die **Selbstrepräsentation** des eigenen Fahrzeugs beinhaltet die Funktionsbereitschaft des ACC-Systems gemäß Spezifikation. 1 s nach Start des Szenarios erfolgt die **Aktion** bzw. das **Ereignis** der Aktivierung des ACC-Systems durch Vorgabe einer Wunschgeschwindigkeit von 130 km/h. Ein individuelles **Ziel** des Systems stellt die Regelung auf die Wunschgeschwindigkeit von 130 km/h dar. Ein individuelles **Ziel** des Fahrers stellt das Erreichen des Reiseziels Braunschweig dar. Ein individueller **Wert** des Fahrers ist die Wahl eines großen Abstands zu einem vorausfahrenden Fahrzeug.

Durch eine **Realfahrt** auf einer Autobahn können Daten aufgezeichnet werden. Durch eine Analyse der Daten können konkrete Szenarien identifiziert und vorhandenen oder zu erstellenden logischen Szenarien zugeordnet werden. So kann zum Beispiel die Breite der Fahrstreifen (in dieser Aufzeichnung 3,4 m) und der Kurvenradius (in dieser Aufzeichnung 600 m) extrahiert werden. Entsprechen die weiteren Gegebenheiten ebenfalls dem obigen logischen Szenario, so kann dieses identifizierte konkrete Szenario diesem logischen Szenario zugeordnet werden.

Konkretes Szenario „ACC-Freifahrt konkret“	
<u>Szenerie:</u>	
Breite Fahrstreifen:	3,2 m
Kurvenradius:	500 m
Umgebungstemperatur:	20 °C
<u>Bewegliche Elemente:</u>	
Fahrzeug mit dem Testobjekt; Startgeschwindigkeit:	100 km/h
<u>Selbstrepräsentation:</u>	
Funktionsbereitschaft ACC-System gemäß Spezifikation	
<u>Aktionen / Ereignisse:</u>	
Aktivierung ACC-System mit Wunschgeschwindigkeit:	130 km/h
Zeitpunkt:	1 s
<u>Individuelle Ziele und Werte:</u>	
Ziel des Systems: Regelung auf die Wunschgeschwindigkeit	130 km/h
Ziel des Fahrers: Erreichen des Reiseziels:	Braunschweig
Wert des Fahrers: Gewünschter Abstand zu einem vorausfahrenden Fahrzeug: groß	

Abbildung 4: Übersicht des abgeleiteten konkreten Szenarios (detaillierte Beschreibung des Beispiels im Text)

2.2 Design- und Implementierungsphase

In diesem Abschnitt werden die Begriffe der Design- und Implementierungsphase aus Steimle u. a. (2018, Abschnitt 2.2) im Rahmen der Entwicklung und Absicherung eines ACC-Systems exemplarisch erläutert und gegeneinander abgegrenzt. Abbildung 5 zeigt diese Begriffe und deren Relationen als UML-Diagramm.

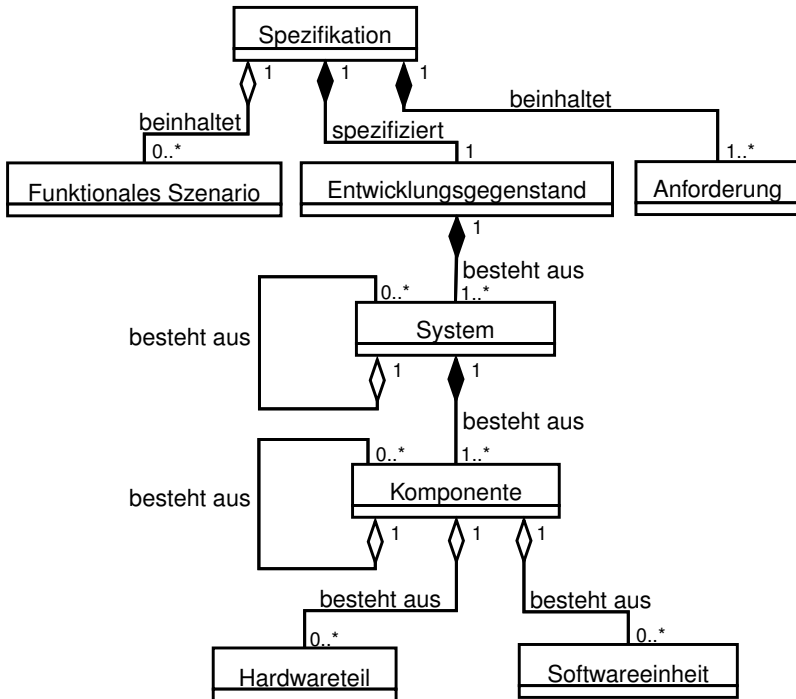


Abbildung 5: Begriffe der Design- und Implementierungsphase dargestellt als UML-Diagramm (Steimle u. a., 2018, S. 8)

Der **Entwicklungsgegenstand** ist das ACC-System. Die nachfolgend aufgeführten **Anforderungen** an das ACC-System sind in Anlehnung an die Norm ISO 15622 (2010) erstellt. Sie bilden das ACC-System nicht vollständig, sondern nur grundlegende Eigenschaften davon ab.

- Erste Anforderung: Das Fahrzeug mit aktiviertem ACC-System soll im Modus Geschwindigkeitsregelung mit einer maximalen Abweichung von ± 5 Prozent auf eine Wunschgeschwindigkeit, die vom Fahrer vorgegeben wird, regeln. Das heißt:
 - Wenn die aktuelle Geschwindigkeit des Fahrzeugs geringer als die Wunschgeschwindigkeit ist, soll das ACC-System auf die Wunschgeschwindigkeit beschleunigen.
 - Wenn die Wunschgeschwindigkeit geringer als die aktuelle Geschwindigkeit des Fahrzeugs ist, soll das ACC-System auf die Wunschgeschwindigkeit verzögern.
- Zweite Anforderung: Die durchschnittliche automatische Verzögerung des Fahrzeugs durch das ACC-System darf $3,5 \text{ m/s}^2$ in einem Intervall von 2 s nicht überschreiten.
- Dritte Anforderung: Die automatische Beschleunigung des Fahrzeugs durch das ACC-System darf $2,0 \text{ m/s}^2$ nicht überschreiten.

Die **Spezifikation** beinhaltet diese Anforderungen und spezifiziert dadurch das funktionale Verhalten des ACC-Systems auf Fahrzeugebene. Durch die erste Anforderung ist auch ein (abstrahiertes) funktionales Szenario in der Spezifikation enthalten.

Eine exemplarische Unterteilung des ACC-Systems in verschiedene Elemente ist in Abbildung 6 abgebildet. Aufgrund der Übersichtlichkeit sind nur beispielhafte Elemente dargestellt. In der Realität sind deutlich mehr Elemente notwendig. Ein **System** des ACC-Systems ist der Verbund aus dem ACC-Steuergerät und dem Radarsensor. Das ACC-Steuergerät ist eine **Komponente**, welche aus Hardwarekomponenten, Hardwareteilen, Softwarekomponenten und Softwareeinheiten besteht. Eine **Hardwarekomponente** ist der gesamte Mikrocontroller. **Hardwareteile** sind die Recheneinheit und der Flash-Speicher des Mikrocontrollers sowie ein Widerstand. Eine **Softwarekomponente** besteht aus der gesamten Software zur Realisierung der ACC-Funktionalität. Eine **Softwareeinheit** ist die Funktion „Berechnung_Sollbeschleunigung“, durch die vom ACC-System die Sollbeschleunigung des Fahrzeugs berechnet wird. Dazu erhält die Funktion die aktuelle Geschwindigkeit des Fahrzeugs und die vom Fahrer vorgegebene Wunschgeschwindigkeit und errechnet daraus die Sollbeschleunigung des Fahrzeugs.

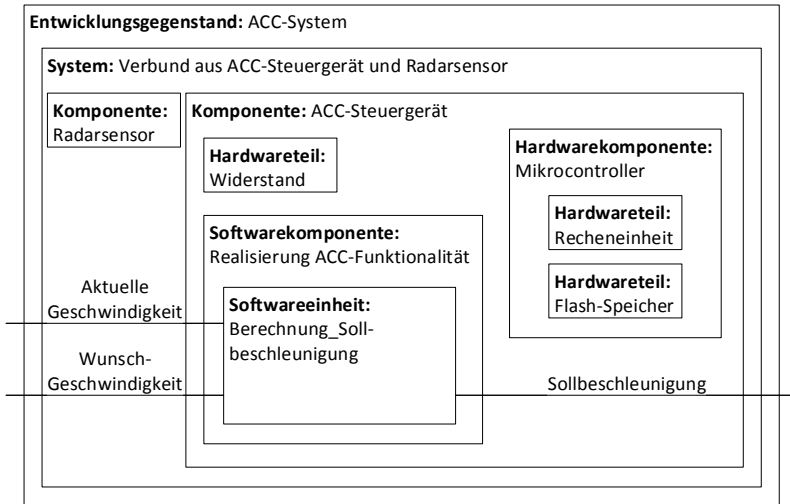


Abbildung 6: Exemplarische Unterteilung des ACC-Systems in verschiedene Elemente (System, Komponente, Hardwareteil und Softwareeinheit)

2.3 Begriffe des Testprozesses

In diesem Abschnitt werden die Begriffe des Testprozesses aus Steimle u. a. (2018, Abschnitt 2.3) im Rahmen der Entwicklung und Absicherung eines ACC-Systems exemplarisch erläutert und gegeneinander abgegrenzt. Steimle u. a. (2018) betrachten die Phase der Testplanung nicht weiter, weshalb sie auch hier nicht berücksichtigt wird.

2.3.1 Testspezifizierung

In diesem Abschnitt werden die Begriffe aus der Phase der Testspezifizierung aus Steimle u. a. (2018, Abschnitt 2.3.2) im Rahmen der Entwicklung und Absicherung eines ACC-Systems exemplarisch erläutert und gegeneinander abgegrenzt. Abbildung 7 zeigt diese Begriffe und deren Relationen als UML-Diagramm.

Zum Erstellen der Testspezifikation ist zuerst ein Testziel zu definieren. Hier ist das **Testziel** die Überprüfung der Anforderungen aus der Spezifikation und das Aufdecken von Abweichungen. Aus den ermit-

telten logischen Szenarien, den Anforderungen aus der Spezifikation und den Testzielen wird die Testspezifikation hergeleitet, welche die konkreten Testfälle und die Testobjekte spezifiziert.

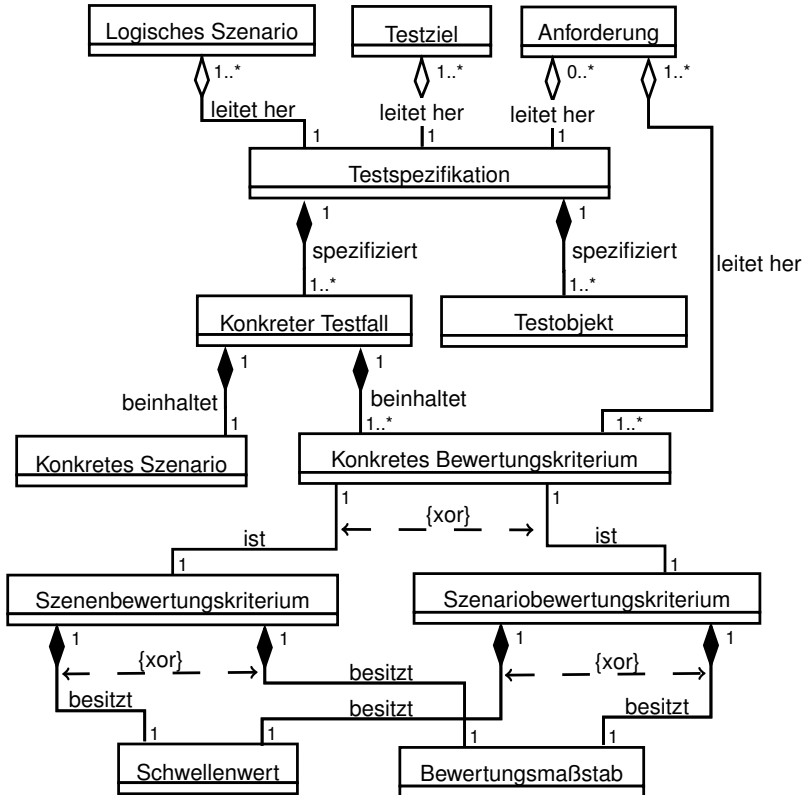


Abbildung 7: Begriffe der Testspezifizierung dargestellt als UML-Diagramm (Steimle u. a., 2018, S. 12)

Mit dem logischen Szenario „ACC-Freifahrt_logisch“ aus Abschnitt 2.1, den Anforderungen aus Abschnitt 2.2 und dem obigen Testziel wird ein konkreter Testfall für die Testspezifikation abgeleitet. Dazu wird auf Basis des obigen logischen Szenarios das konkrete Szenario „ACC-Freifahrt_konkret“ aus Abschnitt 2.1 verwendet, indem für dieses konkrete Szenario folgende konkrete Bewertungskriterien definiert werden.

Aus der ersten Anforderung aus Abschnitt 2.2 wird für diesen konkreten Testfall das erste **konkrete Bewertungskriterium** abgeleitet, welches in Abbildung 8 grafisch dargestellt ist. Es besagt, dass nach Erreichen der vom Fahrer vorgegebenen Wunschgeschwindigkeit diese mit einer maximalen Abweichung von ± 5 Prozent eingehalten werden muss. Dies ist nach Erreichen der Wunschgeschwindigkeit in jeder Szene zu bewerten (**Szenenbewertungskriterium**). Dafür wird ein **Bewertungsmaßstab** definiert, durch den die Einhaltung der Wunschgeschwindigkeit bewertet wird, sodass eine größere Abweichung von dieser schlechter bewertet wird, als eine geringere Abweichung. Wenn die vorgegebene Wunschgeschwindigkeit präzise eingehalten wird, dann wird das konkrete Bewertungskriterium zu 100 Prozent eingehalten. Wenn die aktuelle Geschwindigkeit um mehr als 5 Prozent von der eingestellten Wunschgeschwindigkeit abweicht, dann wird das konkrete Bewertungskriterium zu 0 Prozent eingehalten. Dazwischen liegt in diesem Beispiel eine lineare Bewertung vor.

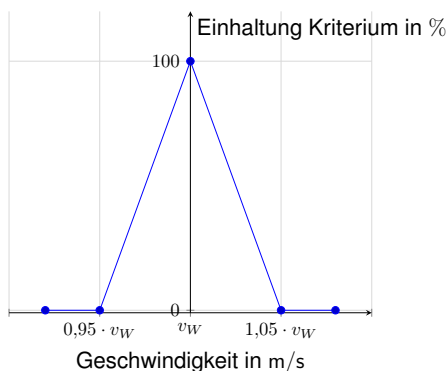
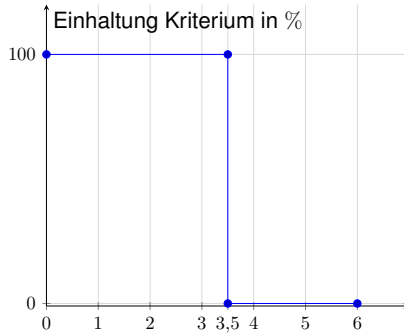


Abbildung 8: Erstes konkretes Bewertungskriterium (Szenenbewertungskriterium mit Bewertungsmaßstab), v_W : Wunschgeschwindigkeit

Aus der zweiten Anforderung aus Abschnitt 2.2 wird für diesen konkreten Testfall das zweite **konkrete Bewertungskriterium** abgeleitet, welches in Abbildung 9 grafisch dargestellt ist. Es besagt, dass die durchschnittliche automatische Verzögerung in einem Intervall von 2 s durch das System nicht größer als $3,5 \text{ m/s}^2$ (**Schwellenwert**) sein darf. Hierzu ist eine Abfolge an Szenen in einem Zeitintervall von 2 s zu überprüfen (**Szenariobewertungskriterium**).



Durchschnittliche automatische Verzögerung innerhalb 2 s in m/s^2

Abbildung 9: Zweites konkretes Bewertungskriterium (Szenariobewertungskriterium mit Schwellenwert)

Aus der dritten Anforderung aus Abschnitt 2.2 wird für diesen konkreten Testfall das dritte **konkrete Bewertungskriterium** abgeleitet, welches in Abbildung 10 grafisch dargestellt ist. Es besagt, dass die automatische Beschleunigung durch das ACC-System $2,0 m/s^2$ (**Schwellenwert**) nicht überschreiten darf. Dies ist in jeder Szene zu überprüfen (**Szenenbewertungskriterium**).

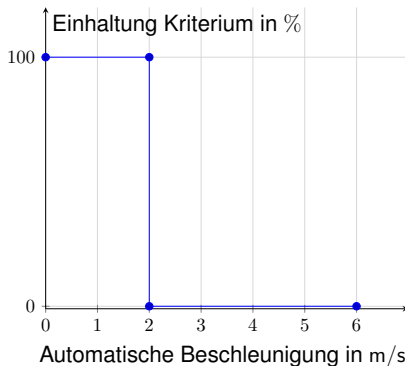


Abbildung 10: Drittes konkretes Bewertungskriterium (Szenenbewertungskriterium mit Schwellenwert)

Die Kombination des konkreten Szenarios „ACC-Freifahrt_konkret“ und dieser konkreten Bewertungskriterien ergibt einen **konkreten Testfall**, der als „Testfall_ACC-Freifahrt“ bezeichnet wird. Für diesen kon-

konkreten Testfall dienen in diesem Beispiel je nach Prüfebene als **Testobjekte** die Softwareeinheit mit dem Namen „Berechnung_Sollbeschleunigung“ und die Komponente mit dem Namen „ACC-Steuergerät“ aus Abschnitt 2.2. Der konkrete Testfall wird mit den Testobjekten in die **Testspezifikation** aufgenommen.

2.3.2 Testinstrumentierung

In diesem Abschnitt werden die Begriffe aus der Phase der Testinstrumentierung aus Steimle u. a. (2018, Abschnitt 2.3.3) im Rahmen der Entwicklung und Absicherung eines ACC-Systems exemplarisch erläutert und gegeneinander abgegrenzt. Abbildung 11 zeigt diese Begriffe und deren Relationen als UML-Diagramm.

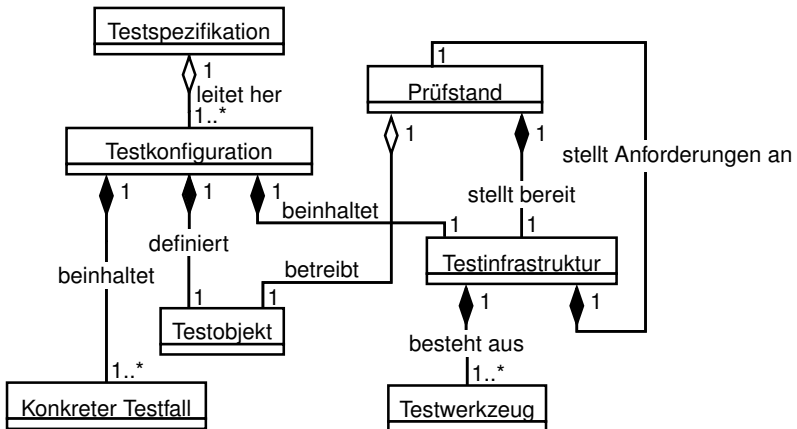


Abbildung 11: Begriffe der Testinstrumentierung dargestellt als UML-Diagramm (Steimle u. a., 2018, S. 15)

Während der Testinstrumentierung werden auf Basis der Testspezifikation die Testkonfigurationen hergeleitet. In diesem Beispiel werden drei Testkonfigurationen zur Ausführung des konkreten Testfalls „Testfall_ACC-Freifahrt“ aus Abschnitt 2.3.1 auf den Prüfebene Ebene und Komponentenebene abgeleitet. Abbildung 12 zeigt exemplarisch den Aufbau dieser drei Testkonfiguration, die im Folgenden beschrieben werden.

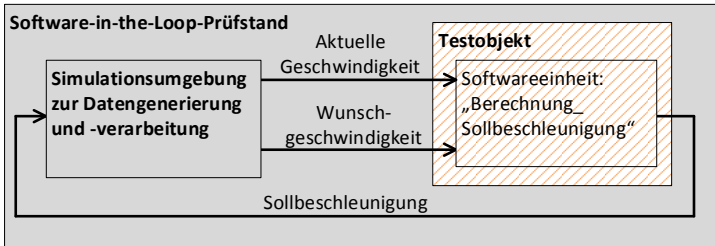
Die erste **Testkonfiguration** wird auf Unitebene für die Softwareeinheit „Berechnung_Sollbeschleunigung“ aus Abschnitt 2.2 erstellt.

Diese ist in Abbildung 12 oben grafisch dargestellt. Das **Testobjekt** (orange schraffiert in Abbildung 12) ist somit die Softwareeinheit „Berechnung_Sollbeschleunigung“. Diese Testkonfiguration besteht aus einem Software-in-the-Loop-Prüfstand, auf dem die Simulationsumgebung zur Datengenerierung und -verarbeitung bereitgestellt und das Testobjekt betrieben wird. Das Originalsteuergerät wird in diesem Fall simuliert. Die Ein- und Ausgänge der Softwareeinheit sind direkt mit der Simulationsumgebung verbunden. Die zur Durchführung des konkreten Testfalls benötigte **Testinfrastruktur** (grau hinterlegt in Abbildung 12) beinhaltet folglich die **Testwerkzeuge** Software-in-the-Loop-Prüfstand mit Simulationsumgebung. Da nur die Softwareeinheit benötigt wird, kann dieser konkrete Testfall mit dieser Testkonfiguration im Vergleich zu den folgenden Testkonfigurationen im Entwicklungsprozess zeitlich gesehen am frühesten durchgeführt werden.

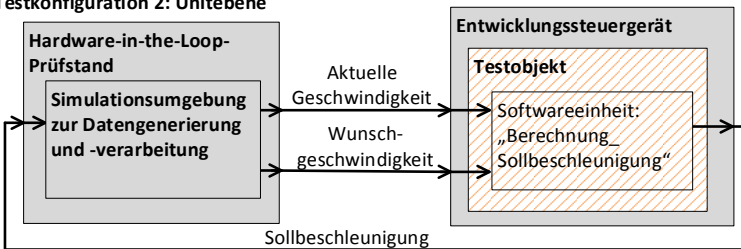
Eine zweite **Testkonfiguration** wird ebenfalls auf Unitebene für die Softwareeinheit „Berechnung_Sollbeschleunigung“ aus Abschnitt 2.2 erstellt. Diese ist in Abbildung 12 in der Mitte grafisch dargestellt. Das **Testobjekt** (orange schraffiert in Abbildung 12) ist somit die Softwareeinheit „Berechnung_Sollbeschleunigung“. Diese Testkonfiguration besteht aus einem Hardware-in-the-Loop-Prüfstand, dessen Schnittstellen mit einem Entwicklungssteuergerät, auf dem die Softwareeinheit ausgeführt wird, verbunden sind und auf dem die Simulationsumgebung zur Datengenerierung und -verarbeitung bereitgestellt wird. Die Ein- und Ausgänge der Simulationsumgebung sind ebenfalls mit den Schnittstellen des Hardware-in-the-Loop-Prüfstandes verbunden. Mit dem Entwicklungssteuergerät wird das Originalsteuergerät emuliert. Dadurch kann dieser Testfall im Entwicklungsprozess zeitlich schon vor dem Vorhandensein des originalen ACC-Steuergerätes durchgeführt werden. Die **Testinfrastruktur** (grau hinterlegt in Abbildung 12) beinhaltet folglich die **Testwerkzeuge** Hardware-in-the-Loop-Prüfstand mit Simulationsumgebung und das Entwicklungssteuergerät.

Eine dritte **Testkonfiguration** wird auf Komponentenebene für die Komponente mit dem Namen „ACC-Steuergerät“ aus Abschnitt 2.2 erstellt. Diese ist in Abbildung 12 unten grafisch dargestellt. Das **Testobjekt** (orange schraffiert in Abbildung 12) ist somit die Komponente mit dem Namen „ACC-Steuergerät“. Diese Testkonfiguration besteht aus einem Hardware-in-the-Loop-Prüfstand, dessen Schnittstellen mit dem originalen ACC-Steuergerät, auf dem die Originalsoftware ausgeführt wird, verbunden sind und auf dem die Simulationsumgebung zur Datengenerierung und -verarbeitung bereitgestellt wird. Die Ein- und

Testkonfiguration 1: Unitebene



Testkonfiguration 2: Unitebene



Testkonfiguration 3: Komponentenebene

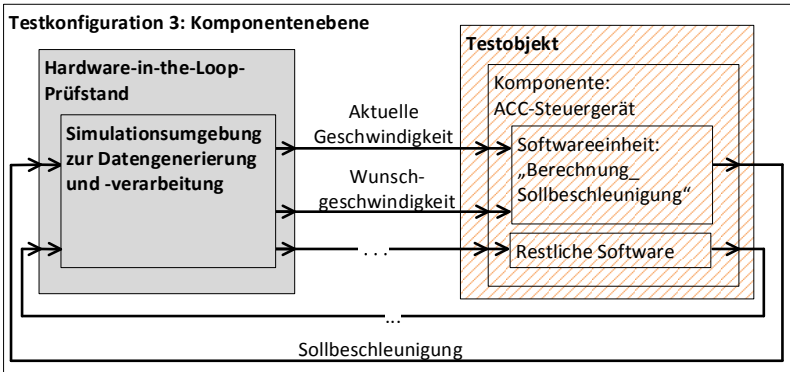


Abbildung 12: Exemplarischer Aufbau drei verschiedener Testkonfigurationen auf unterschiedlichen Prüfebene(n) (grau hinterlegt: Testinfrastruktur; orange schraffiert: Testobjekt)

Ausgänge der Simulationsumgebung sind ebenfalls mit den Schnittstellen des Hardware-in-the-Loop-Prüfstandes verbunden. Das ACC-Steuer-

ergerät empfängt Daten aus der Simulationsumgebung und liefert Stellgrößen an die Simulationsumgebung zurück. Daher sind die benötigten Schnittstellen ebenfalls mit der Simulationsumgebung verbunden. Die **Testinfrastruktur** (grau hinterlegt in Abbildung 12) beinhaltet folglich die **Testwerkzeuge** Hardware-in-the-Loop-Prüfstand mit Simulationsumgebung.

Durch die in allen drei Testkonfigurationen notwendige Testinfrastruktur werden Anforderungen an die **Prüfstände** gestellt, an dem die konkreten Testfälle durchgeführt werden. Die erste Testkonfiguration benötigt einen entsprechend geeigneten Software-in-the-Loop-Prüfstand. Die zweite und dritte Testkonfiguration benötigt jeweils einen entsprechend geeigneten Hardware-in-the-Loop-Prüfstand. Die zweite Testkonfiguration benötigt am Prüfstand zusätzlich ein Entwicklungssteuergerät.

2.3.3 Testdurchführung

In diesem Abschnitt werden die Begriffe aus der Phase der Testdurchführung aus Steimle u. a. (2018, Abschnitt 2.3.4) im Rahmen der Entwicklung und Absicherung eines ACC-Systems exemplarisch erläutert und gegeneinander abgegrenzt. Abbildung 13 zeigt diese Begriffe und deren Relationen als UML-Diagramm.

Nachfolgend wird die dritte Testkonfiguration aus Abschnitt 2.3.2 betrachtet, mit der der konkrete Testfall „Testfall_ACC-Freifahrt“ durchgeführt wird. Durch die Simulationsumgebung werden in jedem Simulationsschritt (danach ergibt sich jeweils eine Szene des konkreten Szenarios) Daten verarbeitet und generiert. In diesem Beispiel werden die aktuelle Geschwindigkeit des simulierten Fahrzeugs, die durch den Fahrer bzw. durch das Fahrermodell vorgegebene Wunschgeschwindigkeit sowie weitere Daten zum Betrieb des ACC-Steuergeräts durch den Hardware-in-the-Loop-Prüfstand als **Testobjekteingabewerte** an das Testobjekt (ACC-Steuergerät) gegeben. Das ACC-Steuergerät verarbeitet diese Daten und errechnet Stellgrößen, die über den Hardware-in-the-Loop-Prüfstand an die Simulationsumgebung zurückgegeben und dort verarbeitet werden. Dadurch entsteht die nächste Szene (**Testobjektausgabewert**) des konkreten Szenarios. Somit liegt eine Closed-Loop-Simulation vor. Im nächsten Simulationsschritt wiederholt sich dieser Ablauf. Die Testobjektausgabewerte werden in diesem Beispiel aufgezeichnet. Der Aufbau dieser Testkonfiguration ist exemplarisch in Abbildung 12 unten aufgezeigt.

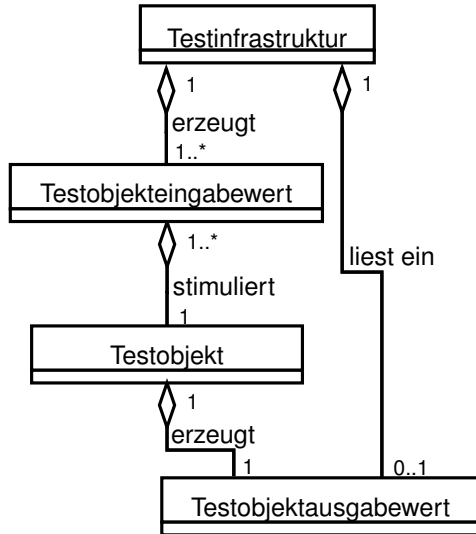


Abbildung 13: Begriffe der Testdurchführung dargestellt als UML-Diagramm (Steimle u. a., 2018, S. 17)

2.3.4 Testbewertung

In diesem Abschnitt werden die Begriffe aus der Phase der Testbewertung aus Steimle u. a. (2018, Abschnitt 2.3.5) im Rahmen der Entwicklung und Absicherung eines ACC-Systems exemplarisch erläutert und gegeneinander abgegrenzt. Abbildung 14 zeigt diese Begriffe und deren Relationen als UML-Diagramm.

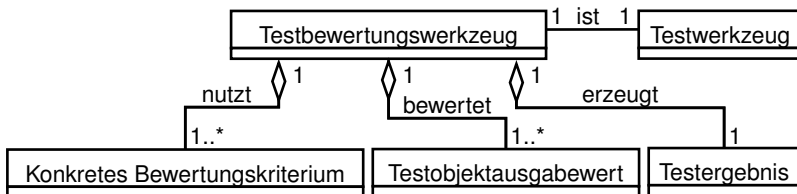


Abbildung 14: Begriffe der Testbewertung dargestellt als UML-Diagramm (Steimle u. a., 2018, S. 18)

In der Testbewertung werden die in der Testdurchführung aus Abschnitt 2.3.3 aufgezeichneten Testobjektausgabewerte durch ein **Testbewertungswerkzeug** mit den konkreten Bewertungskriterien des konkreten Testfalls aus Abschnitt 2.3.1 bewertet. Dazu wird jede Szene auf Einhaltung der Szenenbewertungskriterien überprüft und auf Basis einer Abfolge an Szenen die Szenariobewertungskriterien.

In diesem Beispiel wird nach Erreichen der Wunschgeschwindigkeit in jeder Szene durch das erste konkrete Bewertungskriterium (Szenenbewertungskriterium) aus Abschnitt 2.3.1 überprüft, ob diese mit einer maximalen Abweichung von ± 5 Prozent eingehalten wird und der zugehörige Bewertungsmaßstab angewendet. Weiterhin wird in einer Abfolge an Szenen durch das zweite konkrete Bewertungskriterium (Szenariobewertungskriterium) aus Abschnitt 2.3.1 überprüft, ob die durchschnittliche automatische Verzögerung des Fahrzeugs innerhalb 2 s immer kleiner als oder gleich $3,5 \text{ m/s}^2$ ist. Zusätzlich wird in jeder Szene durch das dritte konkrete Bewertungskriterium (Szenenbewertungskriterium) aus Abschnitt 2.3.1 überprüft, ob die automatische Beschleunigung des Fahrzeugs immer kleiner als oder gleich $2,0 \text{ m/s}^2$ ist.

Auf Basis aller Einzelbewertungen wird ein **Testergebnis** erzeugt. Werden während des konkreten Szenarios alle konkreten Bewertungskriterien eingehalten, so gilt der konkrete Testfall als bestanden. Durch den Bewertungsmaßstab kann zusätzlich eine Aussage getroffen werden, wie gut der Testfall bestanden wurde.

3 Zusammenfassung und Ausblick

In diesem Beitrag wurde das von Steimle u. a. (2018) vorgeschlagene Grundvokabular zur Terminologie für den szenarienbasierten Testansatz automatisierter Fahrfunktionen angewendet, indem den definierten Begriffen ein Beispiel, die Entwicklung und Absicherung eines ACC-Systems, zugewiesen wurde. Aufgrund des Umfangs einer vollständigen Entwicklung und Absicherung eines ACC-Systems, wurden nicht die komplette Entwicklung und Absicherung, sondern beispielhafte Inhalte betrachtet, die die Begriffe exemplarisch verdeutlichen und gegeneinander abgrenzen. Nach Steimle u. a. (2018, S. 18) soll das erstellte Grundvokabular zukünftig erweitert werden. Aufbauend auf diesen Erweiterungen soll auch das in diesem Beitrag aufgezeigte Beispiel entsprechend erweitert werden, sodass auch die neuen Begriffe verdeutlicht und gegeneinander abgegrenzt werden.

Literatur

aFAS Projektbüro (2018). *Forschungsprojekt aFAS*. URL: <https://www.afas-online.de> (abgerufen am 08.01.2018).

Bagschik, G., Menzel, T., Reschka, A. & Maurer, M. (2017). „Szenarien für Entwicklung, Absicherung und Test von automatisierten Fahrzeugen“. In: *11. Workshop Fahrerassistenzsysteme*. Hrsg. von Uni-DAS e. V., S. 125–135.

Bundesministerium für Verkehr und digitale Infrastruktur (2015). *Strategie automatisiertes und vernetztes Fahren: Leitanbieter bleiben, Leitmarkt werden, Regelbetrieb einleiten*. URL: https://www.bmvi.de/SharedDocs/DE/Publikationen/DG/broschuere-strategie-automatisiertes-ernetztes-fahren.pdf?__blob=publicationFile (abgerufen am 06.09.2017).

Bundesministerium für Verkehr und digitale Infrastruktur (2017). *BMVI vergibt dreizehn Förderbescheide für neue Forschungsprojekte*. URL: <http://www.bmvi.de/SharedDocs/DE/Pressemitteilungen/2017/102-dobrindt-22-millionen-euro-fuer-forschungsprojekte-zum-automatisierten-fahren.html?nn=13326> (abgerufen am 07.09.2017).

ENABLE-S3 PROJECT COORDINATION (2018). *Project ENABLE-S3*. URL: <https://www.enable-s3.eu/> (abgerufen am 08.01.2018).

Gasser, T. M., Schmidt, E. A., Bengler, K., Chiellino, U., Diederichs, F., Eckstein, L., Flemisch, F., Fraedrich, E., Fuchs, E., Gustke, M., Hoyer, R., Hüttinger, M., Jipp, M., Köster, F., Kühn, M., Lenz, B., Lotz-Keens, C., Maurer, M., Meurer, M., Meuresch, S., Müller, N., Reitter, C., Reschka, A., Riegelhuth, G., Ritter, J., Siedersberger, K.-H., Stankowitz, W., Trimpop, R. & Zeeb, E. (2015). *Bericht zum Forschungsbedarf - Runder Tisch Automatisiertes Fahren - AG Forschung*. URL: http://www.bmvi.de/SharedDocs/DE/Anlage/Digitales/bericht-zum-forschungsbedarf-runder-tisch-automatisiertes-fahren.pdf?__blob=publicationFile (abgerufen am 27.09.2017).

Ko-HAF Projektbüro (2018). *Projekt Ko-HAF*. URL: <http://www.ko-haf.de/startseite/> (abgerufen am 08.01.2018).

ISO 15622 (2010). *ISO 15622: Intelligent transport systems - Adaptive Cruise Control systems - Performance requirements and test procedures*. Standard ISO 15622:2010. International Organization for Standardization.

PEGASUS Projektbüro (2018). *Forschungsprojekt PEGASUS*. URL: <http://www.pegasusprojekt.de> (abgerufen am 08.01.2018).

Schuldt, F. (2017). „Ein Beitrag für den methodischen Test von automatisierten Fahrfunktionen mit Hilfe von virtuellen Umgebungen“. Dissertation. Braunschweig: Technische Universität Braunschweig.

Steimle, M., Bagschik, G., Menzel, T., Wendler, J. T. & Maurer, M. (2018). „Ein Beitrag zur Terminologie für den szenarienbasierten Testansatz automatisierter Fahrfunktionen“. In: *AAET - Automatisiertes und vernetztes Fahren*. Hrsg. von ITS Niedersachsen, (zur Veröffentlichung angenommen).

Wachenfeld, W. & Winner, H. (2015). „Die Freigabe des autonomen Fahrens“. In: *Autonomes Fahren: Technische, rechtliche und gesellschaftliche Aspekte*. Hrsg. von Maurer, M., Gerdes, J. C., Lenz, B. & Winner, H. Springer Open. Berlin: Springer Vieweg, S. 439–464.