

Suchbasierte Testfallgenerierung und Wiederverwendbarkeit für autonome Fahrsysteme in Kreuzungen

Nicola Kolb

Lehrstuhl für Software & Systems Engineering
Technische Universität München
Boltzmannstr. 3
85748 Garching b. München

nicola.kolb@tum.de

Abstract: Internationale Standardisierungsbemühungen zeigen, dass sich der szenario-basierte Testansatz für das Testen automatisierter und autonomer Fahrsysteme auf Systemebene etabliert. Zur Testfallgenerierung werden im szenario-basierten Testansatz dabei unter anderem suchbasierte Verfahren eingesetzt. Basierend auf der Eingabe eines parametrisierten Szenarios und einer zugehörigen Fitness-Funktion sollen diejenigen konkreten Instanzierungen eines parametrisierten Szenarios identifiziert werden, die das zu testende System in Bezug auf sein sicheres Verhalten besonders herausfordern. Diese werden als Testfälle selektiert und sind *gut* in dem Sinn, dass sie potenziell unsicheres Verhalten des zu testenden Systems aufdecken können. Arbeiten in der Literatur zeigen, dass die Güte der generierten Testfälle unter anderem systemspezifisch sein kann, d.h. ein selektierter *guter* Testfall eines parametrisierten Szenarios für ein zu testendes System A ist nicht zwangsweise ein *guter* Testfall für ein zu testendes System B oder eine neue Systemversion A'. Die selektierten Testfälle sollten deshalb nicht unbedacht wiederverwendet werden. Der Vortrag behandelt Herausforderungen beim Einsatz des suchbasierten Testfallgenerierungsansatzes und bei der Wiederverwendbarkeit der generierten Testfälle im Kreuzungskontext in Erweiterung zum bisher hauptsächlich betrachteten Autobahnkontext.

1 Einleitung und Kontext

Der szenario-basierte Testansatz etabliert sich (siehe Standardisierungsanstrengungen [2]) als Ansatz für das Testen automatisierter und autonomer Fahrsysteme auf Systemebene. Hierbei wird die Betriebsumgebung des zu testenden Systems durch sog.

Szenariotypen beschrieben, die wiederkehrendes Verkehrsgeschehen erfassen. Diese lassen sich operationalisieren und detaillieren, indem geeignete Parameter ausgewählt werden und diesen Parameterdomänen zugewiesen werden. Die daraus resultierenden parametrisierten Szenarien umfassen eine Vielzahl an potentiellen konkreten Instanziierungen eines Szenarios, welche als Testfälle selektiert werden können [1].

Für die Selektion von Testfällen gibt es unterschiedliche Ansätze. Dabei werden unter anderem suchbasierte Verfahren eingesetzt (siehe z.B. [4, 8, 12, 13]). Der Suchraum, definiert durch ein parametrisierte Szenario, dient neben einer spezifischen Fitness-Funktion als Eingabe. Es wird nach derjenigen konkreten Szenario-Instanziierung, charakterisiert durch ihre spezifischen Parameterwertkonstellation, gesucht, die das zu testende System besonders herausfordert und somit potentiell fehlerhaftes Verhalten aufdecken kann [4]. Diese herausfordernden Instanziierungen werden im Folgenden als *gute* Testfälle bezeichnet (siehe [6, 4]). Wenn die herausforderndste Instanz (*worst-case*) selektiert und darin ein sicheres Verhalten des zu testenden Systems nachgewiesen werden kann, ist das der Sicherheitsargumentation dienlich [4].

Während sich bisherige Arbeiten oft auf den Autobahnkontext beziehen, wird in diesem Vortrag die suchbasierte Testfallgenerierung für den Kontext von innerstädtischen Kreuzungen behandelt. Denn die hier auftretenden Verkehrssituationen unterscheiden sich nicht nur infrastrukturbedingt erheblich vom Autobahnkontext, sondern nehmen auch in ihrer Komplexität zu (siehe Kapitel 3). Gleichzeitig versprechen autonome Fahrsysteme besonders im innerstädtischen Bereich ein enormes Potenzial zur Verbesserung der allgemeinen Verkehrssicherheit (siehe [9]). Das Testen dieser Systeme bzgl. ihres sicheren Verhaltens im Kreuzungskontext bedarf allein deshalb besonderer Aufmerksamkeit.

Bei der Testfallgenerierung für den Kreuzungskontext mittels suchbasierter Verfahren werden drei Herausforderungen offenkundig, auf die im Folgenden in Erweiterung zum bisher betrachteten Autobahnkontext eingegangen wird.

2 Verortung im szenario-basierten Ansatz

Ausgehend von einem möglichst vollständigen Katalog an Szenariotypen für den betrachteten Verkehrskontext (vgl. [7]) werden die Szenariotypen zu parametrisierten Szenarien detailliert (siehe Beispiel in Kapitel 3) [1]. Dabei wird für jeden Szenariotyp (mindestens) ein parametrisiertes Szenario abgeleitet (siehe Kapitel 3) [1,3]. Diese wiederum dienen als Eingabe für die Testfallselektion [4]. Aufgrund der angestrebten, endlichen Testaktivität soll aus der Menge aller möglichen konkreten Instanziierungen (= aller möglichen potentiellen Testfälle) der parametrisierten Szenarien jeweils eine Teilmenge an *guten* Testfällen selektiert werden [4]. Der Vortrag bezieht sich auf den Schritt der Testfallgenerierung (siehe blaue Markierung in Abbildung 1).

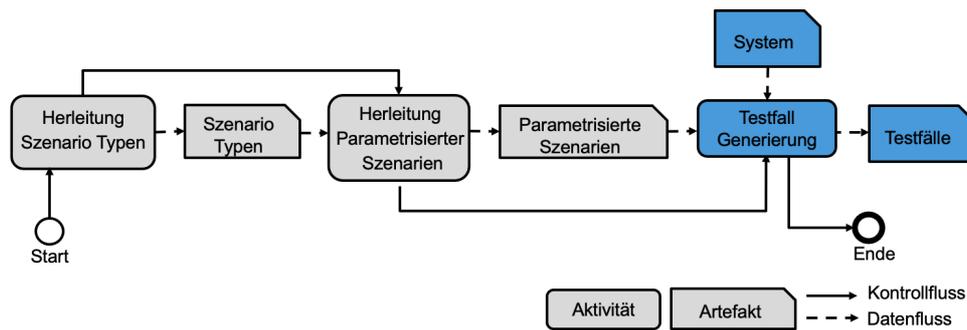


Abbildung 1: Vereinfachter Überblick zur Einordnung des Vortrags im szenario-basierten Ansatz (siehe erw. Versionen in [3][4])

3 Herausforderungen bei der Testfallgenerierung im Kreuzungskontext

Szenarien im Kreuzungskontext unterscheiden sich nicht zuletzt in ihrer Komplexität wesentlich von Szenarien im Autobahnkontext. Die Szenario-Teilnehmer bewegen sich nicht, wie auf der Autobahn der Fall, alle in dieselbe Richtung, sondern relativ zueinander in unterschiedliche Richtungen. Die Straßeninfrastruktur einer Kreuzung selbst schafft naturgemäß eine potenzielle Gefahrenzone, da sich die Wege der Verkehrsteilnehmer kreuzen. Darüber hinaus bieten innerstädtische Kreuzungen eine neue Vielfalt an Verkehrsteilnehmern, da sich auf diesen neben Fahrzeugen zum Beispiel auch vulnerable Teilnehmer wie Fußgänger oder Fahrradfahrer aufhalten. Gleichzeitig postulieren differenzierte Verkehrsregeln Verhaltensvorgaben für ein *sicheres* Fahrverhalten im Kreuzungsverkehr (siehe StVO). Diese Komplexität birgt und verschärft Herausforderungen für den Testfallgenerierungsschritt. Diese lassen sich mit Hilfe des folgenden Beispielszenarios erläutern (siehe Abbildung 2): „*In einer Kreuzung, biegt das Ego-Fahrzeug (zu testendes System) e links ab. Ein entgegenkommendes Fahrzeug c passiert die Kreuzung gerade aus.*“

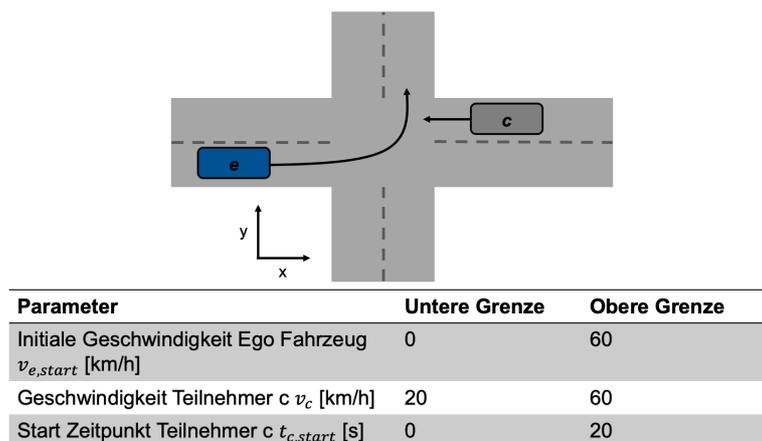


Abbildung 2: Beispiel für ein Kreuzungsszenario (vgl. [3]).

Bei der suchbasierten Testfallgenerierung dient das parametrisierte Szenario als Suchraum für die Selektion *guter* Testfälle. Von den potentiellen Testfällen müssen die Formkriterien des jeweiligen Szenarios erfüllt werden (z.B.: das Ego-Fahrzeug biegt **vor** dem entgegenkommenden Fahrzeug ab und nicht **dahinter**) [4]. Es wird nach denjenigen Parameterwertkonstellationen gesucht, die das zu testende System dergestalt herausfordern, dass es an die Grenzen seines sicheren Betriebsbereichs (siehe [11]) gelangt oder darüber hinaus schreitet (vgl. [3,4]). Wenn gezeigt werden kann, dass sich das zu testende System auch in der herausforderndsten Instanz eines parametrisierten Szenarios sicher verhält, ist dies für die Sicherheitsargumentation des zu testenden Systems dienlich [4]. Vor dem Hintergrund der Argumentation und bei der Anwendung der suchbasierten Testfallgenerierung im Kreuzungskontext werden folgende Herausforderungen offenkundig:

(1) Ein Szenario im Kreuzungskontext lässt sich in unterschiedlichen *Granularitäten* definieren, z.B. folgendermaßen: „Das Ego-Fahrzeug biegt an einer T-Kreuzung mit einem Fahrstreifen in jeder Richtung und mit Gegenverkehr links ab.“ Nach dieser Definition kann für das Szenario als Parameter die Anzahl der entgegenkommenden Fahrzeuge selektiert werden. Das Szenario lässt sich auch in einer anderen Granularität definiert: „Das Ego-Fahrzeug biegt an einer T-Kreuzung mit einem Fahrstreifen in jeder Richtung und einem einzigen entgegenkommenden Fahrzeug *c* links ab.“ Für ein Szenario auf diesem Granularitätslevel ergeben sich andere Parameter. Gleichzeitig hat die Granularität einen Einfluss auf die Formkriterien eines Testfalls [3]. Ein zu wählendes Granularitätslevel kann unter anderem hinsichtlich der Anforderungen an die Interpretierbarkeit der Resultate für die Sicherheitsargumentation vorzugswürdig sein (Beispiel: „Das Ego-Fahrzeug biegt an einer Kreuzung ab“ vs. „Das Ego-Fahrzeug biegt an einer Kreuzung nach links ab“ und „Das Ego-Fahrzeug biegt an einer Kreuzung nach rechts ab“). Das gewählte Granularitätslevel hat Einfluss auf die Anzahl der auszuführenden Suchen, den Suchraum sowie die Kriterien für die Testfallselektion (vgl. [3]).

(2) Das sichere Verhalten des zu testenden Systems in einem Szenario im Kreuzungskontext lässt sich bezüglich unterschiedlicher Sicherheitskriterien betrachten, z.B. in Bezug auf eine laterale oder longitudinale Verletzung der einzuhaltenen minimalen Sicherheitsdistanz zu anderen Verkehrsteilnehmern. Bei der Suche nach *guten* Testfällen, kann eine gleichzeitige Optimierung bzgl. beider Sicherheitskriterien (longitudinal und lateral) dazu führen, dass nicht zwangsweise *gute* Testfälle selektiert werden (z.B. ist eine longitudinale Verletzung der Sicherheitsdistanz ohne gleichzeitig Verletzung der lateralen Sicherheitsdistanz möglich). Stattdessen sollte ein Sicherheitskriterium optimiert werden. Die Erstellung des Suchproblems sowie deren Anzahl und die Interpretierbarkeit der Ergebnisse hängen von den Sicherheitskriterien ab (vgl. [3]).

(3) Bisherige Arbeiten in der Literatur zeigen, unter anderem durch Experimente, dass die Güte der abgeleiteten Testfälle systemspezifisch sein kann (siehe [5]): Ein selektierter *guter* Testfall für eine Systemversion A kann für eine überarbeitete Systemversion A' nicht mehr im ursprünglichen Sinne *gut* sein. Wird beispielsweise der selektierte Testfall zum Test der neuen Systemversion ausgeführt, kann es sein, dass das zu testende System in dem Testfall ein anderes Systemverhalten zeigt und nicht

dasjenige das ursprünglich getestet werden soll (z.B.: das Ego-Fahrzeug biegt *hinter* dem entgegenkommenden Fahrzeug ab und nicht mehr *davor*) [5]. Der Testfall erfüllt potentiell nicht mehr die spezifischen Formkriterien eines Szenarios (vgl. [4]) oder ist nicht mehr *herausfordernd* in dem Sinne, dass es eine potenzielle Verletzung des sicheren Betriebsbereichs provoziert [5]. Erneut, nach jedem Systemupdate, *gute* Testfälle zu generieren, ist, gerade vor dem Hintergrund beschriebenen Zunahme der Anzahl an der Suchprobleme im Kreuzungskontext problematisch. Hierfür bedarf es konstruktiver Lösungsansätze [10].

4 Literaturverzeichnis

- [1] Menzel, T., Bagschik, G., Maurer M. "Scenarios for development, test and validation of automated vehicles." 2018 IEEE Intelligent Vehicles Symposium (IV), 2018, 1821-1827.
- [2] "ISO/DIS 34502 Road vehicles – Scenario-based safety evaluation framework for Automated Driving Systems." International Organization for Standardization, 2021.
- [3] Kolb, N., Hauer, F., Pretschner, A. "Fitness Function Templates for Testing Automated and Autonomous Driving Systems in Intersection Scenarios." 2021 IEEE International Intelligent Transportation Systems Conference (ITSC), 2021, 217-222.
- [4] Hauer, F., Pretschner, A., Holzmüller, B. "Fitness functions for testing automated and autonomous driving systems." In: Romanovsky, A., Troubitsyna, E., Bitsch, F. (eds.) SAFECOMP 2019. LNCS, 11698, 69-84. Springer, Cham, 2019.
- [5] Hauer, F., Pretschner, A., Holzmüller, B. "Re-using concrete test scenarios generally is a bad idea." 2020 IEEE Intelligent Vehicles Symposium (IV), 2020, 1305-1310.
- [6] Pretschner, A. "Defect-Based Testing." Dependable Software Systems Engineering, 84, 141-163, 2015.
- [7] Hauer, F., et al. "Did we test all scenarios for automated and autonomous driving systems?." 2019 IEEE Intelligent Transportation Systems Conference (ITSC), 2019, 2950-2955.
- [8] Riedmaier, S., et al. "Survey on Scenario-Based Safety Assessment of Automated Vehicles." IEEE Access, 8, 87456-87477, 2020.
- [9] Statistisches Bundesamt. „Straßenverkehrsunfälle nach Unfallkategorie, Ortslage“, 2022. <https://www.destatis.de/DE/Themen/Gesellschaft-Umwelt/Verkehrsunfaelle/Tabellen> (aufgerufen am 03/09/2022).
- [10] Hauer, F. „On Scenario-Based Testing of Automated and Autonomous Driving Systems.“ Dissertation. Technische Universität München, 2021, insbes. S. 82.
- [11] Koopman, P., Wagner, M. "Challenges in autonomous vehicle testing and validation." SAE International Journal of Transportation Safety, 4(1), 15-24, 2016.

- [12] Ben Abdesslem, R., et al. „Testing advanced driver assistance systems using multi-objective search and neural networks.” Proceedings of the 31st IEEE/ACM international conference on automated software engineering, 2016, 63-74.
- [13] Buehler, O., Wegener, J. "Evolutionary functional testing of an automated parking system." Proceedings of the International Conference on Computer, Communication and Control Technologies (CCCT'03) and the 9th. International Conference on Information Systems Analysis and Synthesis (ISAS'03), 2003.