# Validierung Autonomer Systeme

Johannes Haberstock
Institut für Automatisierungstechnik
und Softwaresysteme
Universität Stuttgart
johannes@adinfinitum.de

Andreas Löcklin
Institut für Automatisierungstechnik
und Softwaresysteme
Universität Stuttgart
andreas.loecklin@ias.uni-stuttgart.de

Michael Weyrich
Institut für Automatisierungstechnik
und Softwaresysteme
Universität Stuttgart
michael.weyrich@ias.uni-stuttgart.de

***Abstract*—Autonomous Systems are ubiquitous in today's society and have an ever-increasing influence. However, the challenge to validate those systems to ensure safe and fault-free behavior is not solved yet. Due to lack of transparency and complex and adaptive software, traditional validation methods aren't sufficient anymore. This is especially critical considering the fact that Autonomous Systems often operate and interact with humans and therefore are highly safety-critical. This paper shows the problems of traditional validation methods when facing Autonomous Systems and points out how future approaches should look like to ensure verified and reliable behavior.**

***Stichworte—Autonome Systeme, Validierung, Autonomes Fahren, Intelligentes Testen, Kognitives Testen***

## I. Einleitung

Autonome Systeme erfahren ein ständig steigendes, wirtschaftliches Wachstum und sind aus unserer heutigen Gesellschaft nicht mehr wegzudenken. Sie finden Anwendung in den verschiedensten Gebieten von der Luft- und Raumfahrt über automatisiertes Fahren bis hin zu autonomen Industrieanlagen. Die Vorteile liegen auf der Hand, so ist zum Beispiel zu erwarten, dass der Einsatz automatisierter Fahrzeuge bis zu 90 % aller Unfälle verhindert und bis zu 50 % der Pendelzeit pro Benutzer und Tag reduzieren wird. [1]

Autonome Systeme sind flexibler als traditionelle Systeme, da sie sich ihrer Umwelt anpassen, um das vorgeschriebene Ziel zu erreichen. Ihre Stärke liegt in der Fähigkeit sich an verschiedene, unsichere Umgebungen anzupassen, die zur Designzeit eventuell noch nicht vollständig bekannt waren. Diese Stärke wird aber zugleich zu ihrer großen Schwäche im Validierungsprozess. Es ist nicht mehr ausreichend sie nur zur Designzeit zu validieren, sondern die Validierung muss auch während des Einsatzes fortlaufend durchgeführt werden. Zu Beginn herrschte in der Industrie der Irrtum unbenannte und unbewachte autonome Systeme könnten mit den gleichen Vorgehensweisen getestet werden wie herkömmliche Systeme. [2]

Auch kam es in der Vergangenheit aufgrund unzureichender Validierung immer wieder zu schweren Unfällen. Besonders prominent sind die Unfälle selbstfahrender Fahrzeuge, wie sie sich in den Vereinigten Staaten bereits öfter ereigneten. Vorkommnisse wie diese führen dazu, dass automatisierte Fahrzeuge immer weniger von der Gesellschaft akzeptiert werden und Studien belegen, dass allgemein zunehmend Misstrauen und Abneigung gegenüber Autonomen Systemen entsteht [1]. Es liegt in der Verantwortung der Ingenieure, in den kommenden Jahren dieses Vertrauen wieder herzustellen und durch neuartige Validierungsmethoden algorithmische Transparenz und einen sicheren Einsatz zu garantieren.

## II. Ziele und Methodik

Der Zweck dieses Veröffentlichung ist es, die Herausforderungen bei der Validierung Autonomer Systeme aufzuzeigen sowie Lösungsansätze für eine gute und aussagekräftige Validierung zu präsentieren. Um dieses Ziel zu erreichen soll folgenden Forschungsfragen nachgegangen werden:

**Frage 1**: Was bedeutet Autonomie? Was sind die charakteristischen Eigenschaften Autonomer Systeme?

**Frage 2**: Wie funktioniert klassische Validierung bei herkömmlichen, nicht autonomen Systemen und welche Methoden gibt es?

**Frage 3**: Sind klassische Validierungsmethoden anwendbar oder übertragbar auf Autonome Systeme? Wo liegen die Grenzen und Herausforderungen?

**Frage 4:** Wie könnten neuartige Methoden aussehen, die geeignet sind Autonome Systeme zu validieren?

Die Literaturrecherche wurde auf Google Scholar ausschließlich unter dem Suchbegriff „*Validation of Autonomous Systems*“ durchgeführt. Des Weiteren wurden die Publikationen des Instituts für Automatisierungstechnik und Softwaresysteme der Uni Stuttgart herangezogen, insbesondere die Veröffentlichung [3] „Validation of Autonomous Systems“.

## III. Grundlagen

Im folgenden Kapitel werden einige Begriffe näher erläutert, die für das weitere Verständnis notwendig sind.

### A. Verifikation, Validierung und Testen

Verifikation und Validierung (im Folgenden „V&V“) ist nach ISO/IEC/IEEE 24765 der „Prozess der Prüfung, ob Anforderungen an ein System oder eine Komponente komplett und korrekt sind, ob das Produkt in jeder Entwicklungsphase die Anforderungen erfüllt (…) und ob das finale System oder Komponente mit den spezifizierten Anforderungen übereinstimmt.“ Während die Verifikation dabei sicher stellt, dass das „System richtig gebaut“ wurde, beschäftigt sich die Validierung mit der Frage, ob „das richtige System gebaut wurde“ [4]. Testen ist eine Methode der V&V, bei der ein System oder eine Komponente als Testobjekt ausgeführt wird, um bestimmte Eigenschaften oder Verhaltensweisen zu prüfen.

### B. Autonome Systeme

Da für die weiteren Erläuterungen dieser Veröffentlichung ein gutes Verständnis der Charakteristiken Autonomer Systeme notwendig ist, soll im Folgenden kurz auf eine Definition und die daraus folgenden Eigenschaften eingegangen werden. Veröffentlichung [5] fasst eine vielfältige Anzahl von Beschreibungen zusammen und bietet

so eine möglichst allgemeine Definition des Begriffs „Autonome Systeme“:

„*An industrial autonomous system is a delimited technical system, which systematically and without external intervention, achieves its set objectives despite uncertain environmental conditions.”* [5, S. 5]

Diese Definition umfasst folgende vier wichtige Charakteristiken:

- Systematische Prozessausführung
- Anpassungsfähigkeit an ungewisse Umgebungen
- Selbststeuerung
- Eigenständigkeit

Die Systematische Prozessausführung beschreibt die Fähigkeit, modellierte Prozesse auszuführen, woraus sich eine hohe Anzahl von Handlungsmöglichkeiten ergibt. Die Anpassungsfähigkeit an gewisse Umgebungen, um das angestrebte Ziel zu erreichen, ist eine weitere sehr wichtige Eigenschaft Autonomer Systeme und resultiert in einem hochadaptiven Charakter des Systems. Selbststeuerung und Eigenständigkeit bedeutet, dass die Systeme ohne menschliche Aufsicht und Überwachung auskommen müssen. Dies erschwert die Validierung hinsichtlich Sicherheit deutlich, da der Mensch keine Rückfallebene mehr darstellt. [5]

## IV. Traditionelle Validierungsmethoden angewandt auf Autonome Systeme

Veröffentlichung [3] bietet eine Übersicht über traditionelle Validierungsmethoden, angewandt auf Autonome Systeme (vgl. Abbildung 1). Dabei wird auf der horizontalen Achse nach Transparenz der Validierung unterschieden. Während der Tester bei White Box Ansätzen Einblicke in die getestete Software hat, bieten Black Box Ansätze weniger bis keine Transparenz. Die vertikale Achse zeigt die Handhabung beziehungsweise die Automatisierungsmöglichkeit der Techniken auf. Die Methoden der unteren Hälfte werden manuell ausgeführt, die der oberen können automatisiert erfolgen.

Um Fehler in Systemen zu identifizieren, die durch Softwareänderungen hinzukamen, werden oft Regressionstests angewendet. Diese führen nach einer Veränderung eine neuerliche Überprüfung durch und stellen so die Funktionen sicher. Bei herkömmlichen, wie den hier aufgeführten Validierungsmethoden, werden Regressionstests meist durch eine erneute Durchführung der Testfälle oder durch ein erneutes Testen der veränderten Komponenten realisiert. Diese Art von Regressionstests ist bei Autonomen Systemen in der Regel nicht ausreichend, wie im folgenden Abschnitt erläutert wird [3].

## V. Probleme Traditioneller Validierungsmethoden

Obwohl die im letzten Abschnitt erwähnten traditionellen Validierungsmethoden weiterhin eine große Bedeutung für die Validierung Autonomer Systeme haben, stoßen sie doch an ihre Grenzen und können keine ausreichende Testabdeckung liefern. Die schwerwiegendsten Limitierungen werden im Folgenden in drei Punkte zusammengefasst:

Traditionelle Validierungsmethoden sind nicht ausreichend, um die ständig wachsende Komplexität Autonomer Systeme zu testen. Viele Anwendungen erfordern ein komplexes Umfeld, in der die Systeme autonom agieren


Abbildung 1: Klassische Validierungsmethoden für Autonome Systeme, in Anlehnung an [3]

sollen. In der Regel sind dabei die Randbedingungen nicht eindeutig definierbar oder einschränkbar. Sie sind veränderlich und somit sind die Eingangssignale der Systeme hochdimensional und zur Entwurfszeit nicht vollständig bekannt. Dies erfordert eine adaptive Anpassung der Software an die Situation und Umgebung sowie eine fortlaufende Selbstdiagnose und Selbstrekonfiguration. Der Einsatz von lernbasierten Verfahren führt zu nichtdeterministischen Verhaltensweisen, was bedeutet, dass auch Testdurchläufe mit denselben Testparametern verschiedene Ergebnisse liefern können. Autonome Systeme besitzen also eine hohe Komplexität und Vielschichtigkeit und sind daher nicht transparent und für einen Menschen schwer zu begreifen. Ein Testen der Systeme allein mit Brute-Force-Methoden ist keine Option, da ein Durchtesten aller Zustände aufgrund der praktisch endlosen Vielfalt unmöglich ist. Zudem würde selbst eine vollständige Testabdeckung aller Zustände aufgrund des nichtdeterministischen Verhaltens keine Aussagen über die Funktion oder Güte treffen können. [2, 3, 6]

Ein weiterer Grund der Problematik der Validierung liegt darin, dass Autonome Systeme hochgradig dynamisch und adaptiv sind, hauptsächlich aus zwei Gründen: Viele Anwendungen erfordern Upgrades und Updates Over-the-Air, um durch Sicherheitsupdates den hohen Anforderungen der Cybersecurity gerecht zu werden und um kontinuierliche Updates für neue Funktionen einpflegen zu können. Hinzu kommt der Einsatz lernender Systeme, wobei zum Beispiel Neuronale Netze eine fortlaufende Anpassung an die Umgebung gewährleisten. Die hier besprochenen Systeme befinden sich also ständig im Wandel, sowohl in der Entwicklungsphase als auch im Feld und bedürfen somit einer fortlaufenden Validierung. Auf Grund der mangelnden algorithmischen Transparenz ist es für einen menschlichen Prüfer meist unmöglich, das System ausreichend zu verstehen, um Testfälle zur Sicherstellung von Funktionen ableiten zu können. Es werden Regressionstests benötigt, welche die sich fortlaufend ändernden Systeme validieren. Allerdings sind klassische Regressionstests im Sinne von Wiederholtests oder erneuertem Testen einzelner Komponenten nicht ausreichend, wie im folgenden Punkt näher beschrieben wird. [2, 3, 7]

Die dritte Limitierung hat ihren Ursprung in der Tatsache, dass komplexe Systeme oft verteilt sind. Die Problematik bezüglich der Validierung liegt darin, dass Softwareänderungen oder Fehler an einer Stelle sich auf völlig andere

Komponenten durchschlagen können und dort möglicherweise unerwünschtes Verhalten und Fehlfunktionen hervorrufen. Klassische Regressionsmethoden scheitern hier, da ein ständiges Durchtesten des gesamten Systems unverhältnismäßig und unmöglich ist. Auch ein erneutes Testen einer fehlerhaften Komponente kommt nicht in Frage, da der Ursprung des Fehlers an einer völlig anderen Stelle im System liegen könnte. Um dennoch Sicherheit und ein zuverlässiges Verhalten garantieren zu können, werden daher neuartige Regressionsmethoden benötigt. [3, 8]

## VI. Intelligente Validierungstechniken

Wie im vorangegangenen Abschnitt erläutert, stoßen traditionelle Validierungsmethoden in der Anwendung auf Autonome Systeme schnell an ihre Grenzen. Um Forschungsfrage vier zu beantworten gilt es nun, intelligentere Validierungsmethoden zu finden, die mit Hilfe von Regressionstests eine fortlaufende Validierung gewährleisten und somit Garantien über Verhaltensweisen und Sicherheit liefern können. Sie gliedern sich auf der Vierfeldertafel (Abbildung 1) im Bereich „Black Box“ und „Automatisch“ ein. Der hohe Automatisierungsgrad bringt eine deutliche Zeitersparnis bei der Verwaltung von Testfällen mit sich, da deren Design, Erstellung und Ausführung einen sehr hohen Arbeitsaufwand birgt. Zudem werden durch teil- oder vollautomatisierte Testprozesse menschliche Fehler minimiert, die bei der Ableitung von Testfällen auch bei erfahrenen Testingenieuren auftreten. Ein weiteres Ziel Intelligenter Validierungstechniken soll sein, durch neuartige Regressionsmethoden eine hohe Transparenz und Rückverfolgbarkeit in den Algorithmen zu liefern. Hierbei sind Ansätze basierend auf künstlicher Intelligenz grundsätzlich sehr vielversprechend, da sie komplexe Zusammenhänge in verteilten Systemen erkennen und daraus aussagekräftige Testfälle erzeugen können. [3]

Der Prozess der Validierung soll dabei auf allen Ebenen des Systems erfolgen: Auf System-, Komponenten- und Modulebene. Dabei gilt es in jeder Instanz aussagekräftige Testfälle zu finden und zu generieren. Auch hier können KI-basierte Ansätze sehr hilfreich sein, die Anpassung oder Erstellung in die richtige Richtung zu leiten. Denkbar sind einfache Unterstützungshilfen wie die automatische Bewertung einzelner Testfälle hinsichtlich ihrer Wichtigkeit bis hin zur vollautomatischen Generierung mithilfe eines intuitiven User Interfaces für unerfahrene Testingenieure.

Im Folgenden werden zwei konkrete Beispiele für intelligente Validierungstechniken aufgezeigt

### A. Kognitive Tests

In Veröffentlichung [3] werden kognitive Tests als konkrete Fallstudie der intelligenten Validierungstechniken erläutert. Das Ziel ist es hierbei, Testfälle sinnvoll auszuwählen und zu priorisieren, um Autonome Systeme effizient und transparent validieren zu können. Im Gegensatz zu Brute-Force-Methoden, die für eine ausreichende Abdeckung eine unrealistisch hohe Anzahl an Tests benötigen würden, werden hier durch eine gezielte Auswahl kritischer Fälle durch eine kleine Anzahl eine hohe Testabdeckung erreicht. Da Autonome Systeme wie oben gezeigt hoch komplex sind und meist keine algorithmische Transparenz bieten, erfolgt die Auswahl mithilfe von künstlicher Intelligenz. Es werden Bereiche im Zustandsraum ausgewählt, die hinsichtlich der Sicherheit besonders kritisch sind. Dazu wird zunächst ein Soll-Verhalten des Systems durch eine formale Regelstruktur beschrieben. Diese Regelstruktur kann aus Gesetzen, Erfahrung menschlicher Expertise oder Vorgaben zum Beispiel aus Ethik-Kommissionen abgeleitet werden. Im nächsten Schritt vergleicht eine KI das Ist-Verhalten, also das Ergebnis von Testdurchläufen, mit dem Soll-Verhalten und zieht aus der Abweichung Rückschlüsse über die Kritikalität. Die Auswahl der Testfälle soll dabei möglichst transparent und für eine menschliche Prüfung zugänglich sein, was vor allem durch die anschauliche Formulierung der Regelstruktur geschieht. [9]

### B. Software in the Loop

Software in the Loop (im Folgenden "SIL") ist eine weitverbreitete Methode, autonome Fahrzeuge zu validieren und zu testen. Auf einer hochauflösenden, digitalen Karte wird die gesamte Umgebung des Fahrzeuges möglichst realitätsnah nachgebildet und simuliert. Dazu gehören Straßen, Verkehrsschilder, andere Verkehrsteilnehmer sowie die Simulation der Fahrzeugphysik. In dieser Abbildung der realen Welt wird nun dieselbe Software eingespielt, die auch im realen Fahrzeug verwendet wird. Die Software wird also in einer Loop, in einer simulierten Welt getestet. Es können nun Fahrszenarien oder Missionen erstellt werden, die das Fahrzeug fehler- und unfallfrei zu absolvieren hat.

Im ersten Schritt werden die Fahrszenarien von Hand entwickelt und erstellt. Dieser Prozess ist sehr zeitaufwendig und fehlerbehaftet. Um den Arbeitsaufwand zu reduzieren, sollten die Tests automatisiert ausgeführt werden. Die Testdurchläufe können anschließend automatisch bewertet werden und deren Erfolg durch entsprechende Key Performance Indices aussagekräftig beurteilt werden. Des Weiteren können Fahrszenarien automatisch variiert oder sogar neu erzeugt werden. Hierbei gilt es, intelligente Testmethoden zu finden, um besonders kritische und aussagekräftige Szenarien zu generieren. Die Methodik der Kognitiven Tests könnte auch hier große Erfolge liefern, da die KI-Komponente die komplexen Zusammenhänge der Szenarien untersuchen und Rückschlüsse auf deren Kritikalität schließen kann. Aktuell wird bereits die Technik des Search Based Testings angewendet. Bei dieser Methode werden bestimmte Parameter im Zustandsraum verändert, um Bereiche zu identifizieren, die besonders kritisch sind [10]. Auch die Methodik der Uncertainty Quantification eignet sich für die Anwendung auf SIL [11].

Außerdem kann SIL im Sinne des Intelligenten Testens durch die ganze Entwicklungsphase hindurch angewandt werden und nicht nur auf das fertige Produkt. Sind gewisse Softwarekomponenten noch nicht weit genug entwickelt, können diese durch Module mit vereinfachten Funktionen ersetzt werden. So kann zum Beispiel die Trajektorienplanung ersetzt werden durch eine künstliche, vereinfachte Trajektorie, die in konstanter Geschwindigkeit dem Straßenverlauf folgt. Es empfiehlt sich zur fortlaufenden Validierung Regressionstests zu verwenden. Je kritischer ein Szenario ist, desto häufiger sollte es getestet werden. Unkritische Szenarien können seltener durchgeführt werden. Auch kann somit durchgehend die Funktionalität des Systems überprüft werden. Dazu muss nach jeder Softwareänderung, zum Beispiel vor jedem Commit in das Repository, ein besonders kritischer Grund-Testsatz erfolgreich abgeschlossen werden ("Gatekeeper"). Vor jedem Release Candidate könnte die erfolgreiche Ausführung eines umfassenderen Testsatzes gefordert werden.

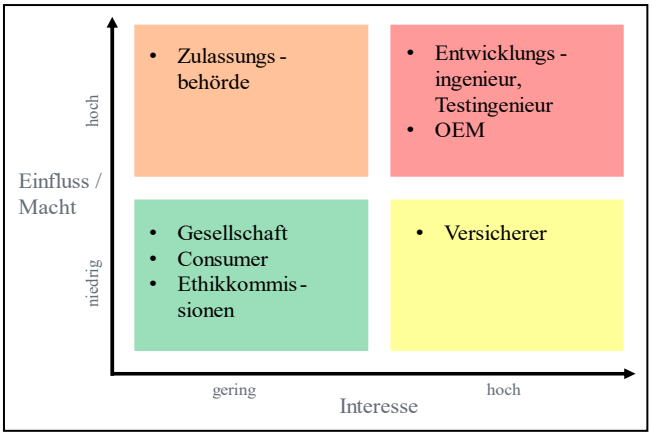


Abbildung 2: Stakeholder Analyse

## VII. Stakeholder Analyse und Value Proposition Canvas

In dieser Veröffentlichung wurde erörtert, wie zukünftige Validierungsmethoden für Autonome Systeme gestaltet sein müssen, um Sicherheit und gesicherte Verhaltensweisen zu garantieren. Nachfolgend soll aufgrund dieser Aussagen ein Geschäftsmodel mit dem Gegenstand der Validierung entwickelt werden.

Um potenzielle Geschäftsideen strukturiert entwickeln und prüfen zu können, wird hierzu das sogenannte Lean Canvas verwendet, welches vor allem in der Start-up-Szene geläufig ist. Zunächst soll eine Stakeholder Analyse durchgeführt werden und anschließend mit Hilfe des Value Proposition Canvas (im Folgenden „VPC“) ein mögliches Produkt abgeleitet werden.

### A. *Stakeholder Analyse*

Zur Stakeholder Analyse wird eine Matrix (siehe Abbildung 2) verwendet, die potenzielle Stakeholder anhand ihrer Interessen und ihres Einflusses auf die Entwicklung des Produkts trennt. Die zu lösende Herausforderung soll dabei die „Vereinfachung der Validierung Autonomer Systeme“ sein.
Im Bereich des hohen Einflusses und hoher Interesse befinden sich die Schlüsselspieler, die sich meist am besten als mögliche Kunden eignen. Im beschriebenen Beispiel führt der Entwicklungs- beziehungsweise Testingenieur die Validierung eines Autonomen Systems durch. Genau wie der einzelne Ingenieur hat aber auch der OEM ein großes Interesse, die Validierung zu vereinfachen.
Neben den Schlüsselspielern müssen aber auch alle anderen Stakeholder analysiert und beachtet werden. Ein hohes Interesse, aber einen geringen Einfluss haben beispielsweise Versicherer, da sie an robusteren Systemen aufgrund guter Validierung zwar sehr interessiert sind, aber keinen direkten Einfluss darauf haben. Es ist ausreichend, Stakeholder in dieser Gruppe regelmäßig über den Fortschritt der Entwicklung zu informieren.
Einen hohen Einfluss aber dafür wenig Interesse an einer Vereinfachung der Validierung haben Zulassungsbehörden, da die Aufgabe, eine sichere und aussagekräftige Validierung zu liefern, beim Hersteller liegt. Der hohe Einfluss ist in der legislativen Gewalt der Behörde begründet. Das Lean Canvas rät zu einer guten Betreuung und Absprache mit Stakeholdern aus dieser Gruppe.
Die vierte Kategorie mit geringem Interesse als auch geringem Einfluss bildet die Gesellschaft beziehungsweise der einzelne Konsument eines fertigen Systems als auch eine Ethikkommission. In den meisten Fällen hat diese Gruppe kein Bewusstsein für die Wichtigkeit und Auswirkung der Validierung und somit auch kein direktes Interesse an deren Vereinfachung. [12]
Gegenstand der folgenden Produktentwicklung soll der Bereich der Schlüsselspieler sein.

### B. *Value Proposition Canvas*

Es empfiehlt sich einen virtuellen Stakeholder klar und konkret zu formulieren um dann mit Hilfe des Value Proposition Canvas ein möglichst effizientes und aussagekräftiges Produkt ableiten zu können. Näher betrachtet werden soll hier ein Entwicklungs- beziehungsweise Testingenieur, der im Bereich der Intralogistik tätig ist. Das Unternehmen stellt Automated Guided Vehicles (nachfolgend „AGV“) her wie zum Beispiel einen autonomen Gabelstapler, der auch in der Umgebung von Arbeitern eingesetzt wird. Die Anwendung ist also hoch sicherheitskritisch und es wird weiter angenommen, der Testingenieur habe keine weitreichende Erfahrung in der Validierung komplexer Autonomer Systeme. Diese Annahme ist entscheidend für die weitere Produktentwicklung, da definitiv ein Mangel an Fachpersonal in diesem Gebiet herrscht und diese Problematik durch entsprechende Produkte oder Tools teilweise gelöst werden könnte.

Die rechte Seite des VPC (siehe Abbildung 3) bildet die Customer Seite, auf welcher die Jobs-to-be-done, Pains sowie Gains für den Testingenieur eingetragen werden. Der wichtigste Job ist die Validierung und das Testen des Systems. Weiterhin wird angenommen, der Ingenieur hat die Aufgabe die Güte des AGV zu bewerten und mit den Systemen anderer Hersteller zu vergleichen.
Die Pains, die er dabei verspürt, sind die schwere Durchführbarkeit von Realtests. Reale Tests zu planen, designen und durchzuführen stellt einen großen Arbeitsaufwand dar. Die Anzahl der Testszenarien, sei es für den Realtest oder simulierte Tests, ist dabei riesig und deshalb schwer zu verwalten. Fehlerquellen sind aufgrund der in Abschnitt V genannten Problematiken schwer identifizierbar. Die Gütebewertung und der Verglich mit anderen Systemen ist sehr schwer realisierbar, da es keine einheitlichen Bewertungskriterien oder einheitliche Vergleichsbasis gibt. Gains, die aus einem potenziellen Produkt entstehen könnten und über die Jobs-to-be-done hinausgehen sind zum Beispiel eine definierte Kategorisierung von Systemen oder ein automatisiertes Reporting von Fehlern sowie eine automatisierte Dokumentation der Testdurchläufe. Weiterhin könnte ein Tool mit intuitiven User Interface alle hier aufgeführten Aufgaben erleichtern und gerade für unerfahrene Testingenieure zugänglicher machen.

Aus den Bedürfnissen des Kunden auf der rechten Seite des VPC werden nun entsprechend auf der linken Seite, der Produktseite, Lösungen für die Pains und Gains entwickelt um auf deren Basis eine Produktidee abzuleiten.
Begonnen wird mit der Betrachtung der Pains. Um den Aufwand der Realtests zu minimieren sollte das Produkt eine Unterstützung bei der Planung bieten und Testserien verwalten. Die riesige Menge an möglichen Testszenarien wird überschaubarer gemacht, indem eine automatische Bewertung der Szenarien gegeben werden soll sowie eine Indizierung und Priorisierung hinsichtlich ihrer Kritikalität. Da Fehlerquellen wie oben beschrieben schwer, beziehungsweise für einen Menschen oft gar nicht identifizierbar sind,

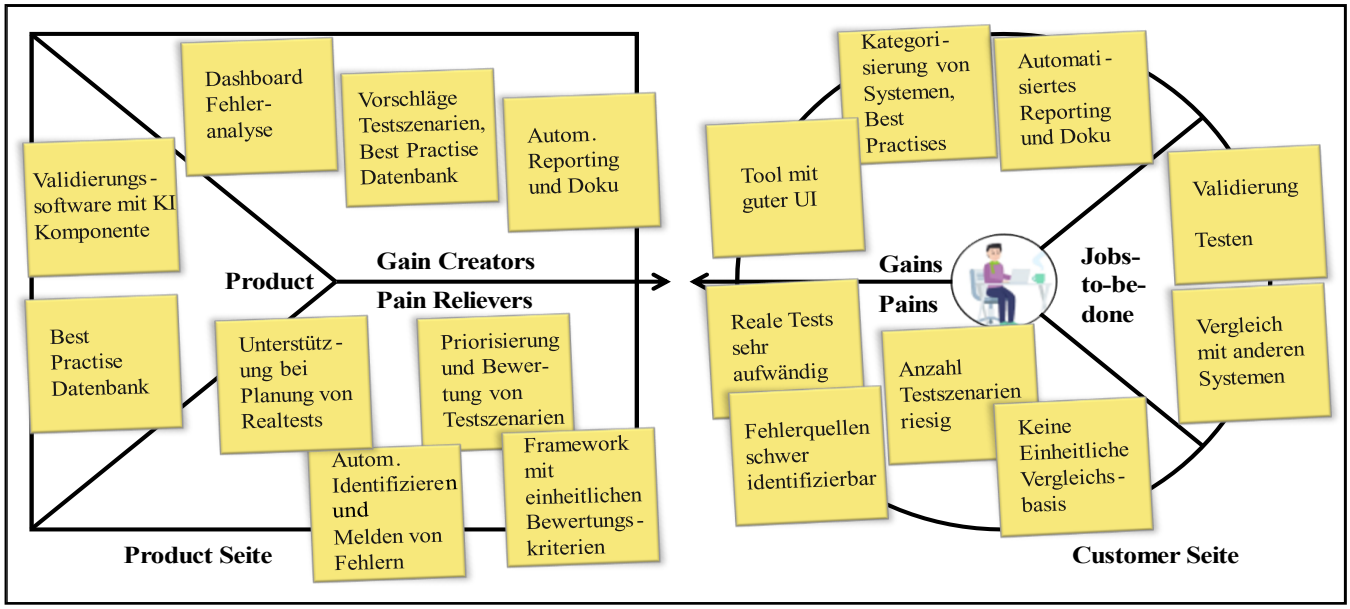


Abbildung 3: Value Proposition Canvas

soll eine automatische Identifizierung und Fehlermeldung den Testingenieur unterstützen. Mithilfe eines Frameworks mit einheitlichen Bewertungskriterien wird eine aussagekräftige Gütebewertung des Systems ermöglicht. Dadurch wird eine einheitliche Vergleichsbasis zu Produkten von anderen Herstellern geschaffen.
Als Gain Creator könnte im Produkt eine Software zum Einsatz kommen, die bei der Verwaltung und Auswahl von Testszenarien Unterstützung bietet. Weiterführend soll sie auch auf Basis einer Datenbank mit kategorisierten Szenarien Beispielszenarien vorschlagen und mit einer Best-Practise-Datenbank Vorschläge zu Validierungstechniken liefert. Ein Dashboard zur Fehleranalyse mit intuitivem User Interface soll den unerfahrenen Testingenieur einen Überblick über fehlgeschlagene Testdurchläufe und ein Verständnis über den Zustand des Systems ermöglichen. Zudem soll das Tool erfolgreiche sowie fehlgeschlagene Tests automatisch dokumentieren und übersichtlich darstellen.

Das hier entwickelte Produkt ist nun eine Validierungssoftware, die den unerfahrenen Testingenieur bei der Planung und Validierung Autonomer Systeme unterstützt. Sie gibt auf Basis einer Best-Practise-Datenbank Vorschläge zu geeigneten Validierungstechniken und Testszenarien und bietet Unterstützung bei der Planung von Tests.
Durch das Bewertungsframework wird die Güte und Sicherheit des Systems in aussagekräftigen Größen ausgedrückt, was das Tool nicht nur für den Ingenieur sondern besonders auch für das Management attraktiv macht, da der Projektfortschritt besser kontrolliert und gesteuert werden kann. Um Hilfestellungen bei der Fehleridentifikation und Auswahl kritischer Testfälle zu bieten, sollen Ansätze mit künstlicher Intelligenz eingesetzt werden. Die KI-Unterstützung soll dabei allerdings einen relativ geringen Komplexitätsgrad aufweisen, um keine großen Unsicherheiten zu erzeugen und dem Ingenieur den Validierungsvorgang möglichst transparent zu gestalten. Das Tool soll keine vollautomatische Erzeugung und Ausführung von Testszenarien bieten, sondern lediglich die oben beschriebenen Hilfestellungen zur Validierung leisten. Die durchgeführte VPC Analyse hat gezeigt, dass solch ein Tool bereits einen enormen Mehrwert bietet und dafür auch ein Kundenkreis existiert.
Eine schrittweise Erweiterung des Tools hinsichtlich höherer Komplexität durch hohen Automatisierungsgrad und vollautomatischer Testentwicklung ist denkbar für weitere, zukünftige Produktiterationen. Allerdings ist dies nach aktuellem Stand der Technik nicht oder nur unter massiven Aufwand realisierbar und wird nach Auffassung des Autors auch noch lange Zeit dauern.

## VIII. Zusammenfassung und Ausblick

Auch wenn Autonome Systeme nicht immer von den Benutzern aktiv als solche wahrgenommen werden, sind sie doch allgegenwärtig und aus der heutigen Gesellschaft nicht mehr wegzudenken. In der Zukunft wird sich dieser Trend voraussichtlich noch weiter verschärfen. So wird beispielsweise der Einsatz automatisierter Shuttlefahrzeuge den Personentransport in Innenstädten völlig verändern, die Verkehrssituation entlasten und die Anzahl der Verkehrsunfälle drastisch senken [1]. Um die ständig wachsende Komplexität zu beherrschen sind nicht nur fortschrittliche Entwicklungsmethoden, sondern auch neuartige und geeignete Validierungstechniken notwendig. Dies stellt eine enorme Herausforderung dar. Die zuverlässige Validierung Autonomer Systeme zur Sicherstellung der Funktionen ist insbesondere deshalb essenziell, da sie oft in Interaktion mit Menschen arbeiten und somit höchst sicherheitskritisch sind.

Die Validierung technischer Systeme ist aufgrund der ständig wachsenden Komplexität seit jeher schwierig und aufwändig, sodass ihr oft zu wenig Bedeutung zugestanden wird. In dieser Veröffentlichung wurde gezeigt, dass diese Problematik bei Autonomen Systemen besonders zum Tragen kommt und erörtert, weshalb traditionelle Validierungsmethoden an ihre Grenzen stoßen. Charakteristisch für Autonome Systeme sind eine hohe Adaptivität und nichtdeterminischtische Verhaltensweisen, welche sich auch während des Einsatzes verändern. Sie müssen zum Teil in offenen Umgebungen agieren, welche zur Designzeit nicht vollständig bekannt sind oder wegen ihrer hohen Komplexität nicht vollständig beschreibbar sind. Während bei herkömmlichen Systemen, vor allem im industriellen Bereich, oft das Umfeld vereinfacht werden kann, um die Vielfalt an Einsatzsituationen zu verringern, ist dies bei Autonomen Systemen in der Regel nicht möglich. Es wird

viel Aufwand in die Forschung gesteckt, um die Validierung und das Testen Autonomer Systeme voranzutreiben. Im Zuge dieser Veröffentlichung wurde aufgezeigt, wie in Zukunft Intelligente Validierungsmethoden durch den Einsatz von künstlicher Intelligenz Antworten auf die oben genannten Probleme liefern und Transparenz bei der Absicherung von Funktionen bringen können.

Während Autonome Systeme immer mehr Auswirkung auf unser Leben haben, entsteht ihnen gegenüber zunehmend Misstrauen und Abneigung. Es ist die Aufgabe der Ingenieure, Möglichkeiten zu finden dieses Vertrauen wiederherzustellen. Dabei ist der Aufbau von Vertrauen eng verknüpft mit Fragen der Validierung. Es wird noch ein langer Weg sein bis dieses Ziel erreicht sein wird, zumal der Mensch eine viel geringere Fehlertoleranz gegenüber Robotern hat und von ihnen mindestens eine Größenordnung höhere Qualität erwartet.
Allerdings ist dabei auch eine Lernkurve der Akzeptanz zu verzeichnen. Die vielen Vorteile eines Systems werden oft trotz des verbleibenden Restrisikos akzeptiert. Beispielsweise geschehen in der Luftfahrt immer wieder schreckliche Unfälle und trotzdem schrecken die wenigsten Menschen davor zurück, in den Urlaub zu fliegen. Auch nutzen fast alle Menschen in ihrem Smartphone die Google-Play-Dienste oder die Funktionen ihres iPhones trotz der Angst vor Datenmissbrauch.
Ein Aspekt, der vermutlich noch lange Zeit eine offene Problematik darstellen wird oder sogar unlösbar sein wird, ist die Frage der ethischen Dilemmata. Angenommen es wäre möglich, einen fehler- und risikofreien Einsatz eines Autonomen Systems absolut sicher zu garantieren, so können dennoch ethische Dilemmata auftreten, wie sie vor allem beim automatisierten Fahren oft diskutiert werden. Es bleibt die Frage offen, wie in diesem Fall Vertrauen geschaffen werden kann.

Aufgabe der Forschung und des Gesetzgebers für die nächsten Jahre wird es auch sein, geeignete Standards, Normierungen und Zertifizierungsstrategien zu entwickeln. So ist Beispielsweise die Norm zur funktionalen Sicherheit von Straßenfahrzeugen ISO 26262 offensichtlich nicht mehr geeignet, um automatisierte Fahrzeuge zu entwickeln. Der V-Prozess ist für hochkomplexe Softwaresysteme nur eingeschränkt anwendbar und die vorgeschriebenen Validierungsmethoden nicht ausreichend. [6] bieten einen vielversprechenden Ansatz, den V-Prozess der ISO 26262 für die Anwendung auf autonome Fahrzeuge anzupassen. Besonders bedeutend wird auch die ISO/PAS 21448 sein, aus welcher derzeit (Stand Frühjahr 2021) die Norm ISO 21448 „Road vehicles – Safety oft he intended functionality“ (kurz SOTIF) entwickelt wird.

Vielversprechend für Autonome Systeme im Allgemeinen ist insbesondere die 2021 erschienene Veröffentlichung „Towards a framework for vertification“ [13], welche aktuelle Zertifizierungsstandards der Robotik aufführt sowie deren Grenzen bei der Anwendung auf Autonome Systeme. Die Autoren bieten Strategien, wie die Standards angepasst werden sollten und welche dabei die Herausforderungen für die zukünftige Forschung sind.

## IX. Literatur


[1] P. Gao, *Automotive revolution: perspective towards 2030: how the convergence of disruptive technology-driven trends could transform the auto industry*. McKinsey and Company, 2016. [Online]. Verfügbar unter: https://www.voced.edu.au/content/ngv:74173

[2] P. Helle, W. Schamai und C. Strobel, „Testing of Autonomous Systems - Challenges and Current State-of-the-Art“ (en), *INCOSE International Symposium*, Jg. 26, Nr. 1, S. 571–584, 2016, doi: 10.1002/j.2334-5837.2016.00179.x.

[3] C. Ebert und M. Weyrich, „Validation of Autonomous Systems“, *IEEE Softw.*, Jg. 36, Nr. 5, S. 15–23, 2019, doi: 10.1109/MS.2019.2921037.

[4] A. Locklin, M. Muller, T. Jung, N. Jazdi, D. White und M. Weyrich, „Digital Twin for Verification and Validation of Industrial Automation Systems – a Survey“ in *2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, Vienna, Austria, 08.09.2020 - 11.09.2020, S. 851–858, doi: 10.1109/ETFA46521.2020.9212051.

[5] Manuel Müller, Timo Müller, Behrang Ashtari Talkhestani, Philipp Marks, Nasser Jazdi und Michael Weyrich, „Industrial autonomous systems: a survey on definitions, characteristics and abilities“ (de), *at - Automatisierungstechnik*, Jg. 69, Nr. 1, S. 3–13, 2021, doi: 10.1515/auto-2020-0131.

[6] P. Koopman und M. Wagner, „Challenges in Autonomous Vehicle Testing and Validation“, *SAE International Journal of Transportation Safety*, Jg. 4, Nr. 1, S. 15–24, 2016. [Online]. Verfügbar unter: http://www.jstor.org/stable/26167741

[7] A. Zeller, N. Jazdi und M. Weyrich, „Functional verification of distributed automation systems“ (en), *Int J Adv Manuf Technol*, Jg. 105, Nr. 9, S. 3991–4004, 2019, doi: 10.1007/s00170-019-03791-2.

[8] C. Ebert und M. Weyrich, „Validation of Automated and Autonomous Vehicles“, *ATZ Electron Worldw*, Jg. 14, Nr. 9, S. 26–31, 2019, doi: 10.1007/s38314-019-0090-9.

[9] C. Ebert, M. Fouad, B. Lindemann und M. Weyrich, „Validierung Autonomer Systeme - Transparenz und Effizienz durch kognitive Testmethoden“, *OBJEKTspektrum*, Jg. 06, S. 32–37, 2020.

[10] C. Gladisch, T. Heinz, C. Heinzemann, J. Oehlerking, A. von Vietinghoff und T. Pfitzer, „Experience Paper: Search-Based Testing in Automated Driving Control Applications“ in *2019 34th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, San Diego, CA, USA, 2019, S. 26–37, doi: 10.1109/ASE.2019.00013.

[11] S. Riedmaier, B. Danquah, B. Schick und F. Diermeyer, „Unified Framework and Survey for Model Verification, Validation and Uncertainty Quantification“, *Arch Computat Methods Eng*, S. 1–34, 2020, doi: 10.1007/s11831-020-09473-7.

[12] t2informatik. Wir entwickeln Software., „Was ist eine Stakeholder-Matrix? - Wissen kompakt - t2informatik“, *t2informatik GmbH*, 25. Juni 2018, 2018. [Online]. Verfügbar unter: https://t2informatik.de/wissen-kompakt/stakeholder-matrix/. Zugriff am: 4. Februar 2021.

[13] M. Fisher, V. Mascardi, K. Y. Rozier, B.-H. Schlingloff, M. Winikoff und N. Yorke-Smith, „Towards a framework for certification of reliable autonomous systems“ (en), *Auton Agent Multi-Agent Syst*, Jg. 35, Nr. 1, S. 1–65, 2021, doi: 10.1007/s10458-020-09487-2.