Zuverlässigkeitsbewertung einer Getriebesteuerungs-Software durch Auswertung der Betriebserfahrung

Sven Söhnlein[1], Francesca Saglietti[1], Franz Bitzer[2], Siegfried Baryschew

[1]*Lehrstuhl für Software Engineering,*
Universität Erlangen-Nürnberg,
91058 Erlangen,
{soehnlein, saglietti}@informatik.uni-erlangen.de

[2]*LPE2-FB/Functions Basic Development,*
ZF Friedrichshafen AG,
88038 Friedrichshafen,
franz.bitzer@zf.com

Zusammenfassung

Dieser Artikel schildert einen neuen Ansatz zum Zuverlässigkeitsnachweis sicherheitskritischer Softwaresysteme mittels statistischer Auswertung der Betriebserfahrung. Die praktische Anwendung dieses Verfahrens wird anhand einer Getriebesteuerungs-Software demonstriert, welche momentan im Rahmen einer industriellen Forschungskooperation durchgeführt wird.

1. Einleitung

Für den Einsatz komplexer Softwaresysteme in sicherheitskritischen Anwendungsbereichen ist angesichts der schwerwiegenden Folgen von Versagen ein rigoroser Nachweis hoher Zuverlässigkeit angebracht und oft auch vorgeschrieben. Die während einer vorangegangenen Test- oder Betriebsphase beobachtete fehlerfreie Funktionsweise deutet zwar auf ein zuverlässiges Produkt hin, jedoch fehlen bisher Methoden um einen quantitativen Nachweis auf Basis der gewonnen Betriebserfahrung zu erbringen.

Einen Ansatz für die quantitative Zuverlässigkeitsbewertung bietet die Anwendung der statistischen Stichprobentheorie [1, 4, 8, 9, 10, 11]. Während der Einsatz dieser Vorgehensweise für ein neues System zu einem extrem hohen Testaufwand führt [2, 6, 7], kann die Auswertung bereits gewonnener Betriebserfahrung zu einer deutlichen Ersparnis beitragen.

Die wirtschaftliche Attraktivität der Auswertung umfangreicher Betriebserfahrung, welche im Rahmen eines industriellen Forschungsprojektes mit ZF Friedrichshafen AG gerade praktisch erprobt wird, gewinnt zunehmend an Bedeutung. Deshalb beschäftigt sich dieser Artikel mit der praktischen Beschreibung einer Vorgehensweise zur Ermittlung von Zuverlässigkeitskenngrößen auf Basis der mit einer Getriebesteuerungs-Software gesammelten operationalen Daten.

Der Artikel ist wie folgt gegliedert: In Kapitel 2 werden die wesentlichen Grundlagen der statistischen Stichprobentheorie für softwarebasierte Systeme dargestellt. Kapitel 3 enthält die allgemeine Beschreibung einer Leitlinie zur Extraktion statistisch relevanter Betriebserfahrung, wobei anschließend in Kapitel 4 auf die praktische Anwendung dieser Leitlinie am Beispiel einer Getriebesteuerungssoftware eingegangen wird.

2. Softwarezuverlässigkeitsbewertung durch statistisches Testen

In diesem Kapitel werden die wichtigsten Grundlagen des statistischen Testens kurz vorgestellt. Für eine ausführlichere Beschreibung sei z. B. auf [1, 4] verwiesen. Anhand der statistischen Stichprobentheorie wird nach Beobachtung von n korrekt ausgeführten Test- bzw. Betriebsfällen zu einer vorgegebenen Aussagesicherheit β eine obere Schranke $\tilde{p}$ der Versagenswahrscheinlichkeit p eines Systems bestimmt, d.h.

$$P(p \le \tilde{p}) = \beta \qquad (1)$$

Um diese Theorie anzuwenden, müssen allerdings die folgenden Voraussetzungen erfüllt sein:

Voraussetzung 1 - Unabhängige Auswahl der Testfälle:
Die Auswahl eines Testfalls darf keinen Einfluss auf die Auswahl weiterer Testfälle haben.

Voraussetzung 2: Unabhängige Ausführung der Testfälle:
Die Ausführung eines Testfalls darf keinen Einfluss auf das Ergebnis weiterer Testfälle haben.

Voraussetzung 3 - Betriebstreue:
Im Laufe des beobachteten Tests bzw. Betriebs kommen Eingangsdaten mit der gleichen Wahrscheinlichkeit zum Zuge, mit der sie im künftigen Betrieb erwartet werden. Falls das Betriebsprofil aus einer früheren Betriebsphase von dem zu erwartenden Profil des künftigen Einsatzes in einem neuen System abweicht, ist eine entsprechende Anpassung bzw. Umrechnung zur Sicherstellung dieser Voraussetzung notwendig (siehe [12]).

Voraussetzung 4 - Versagensfreie Test- bzw. Betriebserfahrung:
Bei keinem der beobachteten Test- bzw. Betriebsfälle werden Versagen beobachtet. Die allgemeine Theorie über statistische Konfidenzintervalle (vgl. u.a. [15]) erlaubt zwar die Beobachtung einzelner Versagen, allerdings auf Kosten der daraus herleitbaren Zuverlässigkeitskenngrößen. Um im Fall sicherheitskritischer Anwendungen besonders hohe Zuverlässigkeitsaussagen zu erzielen, beschränken sich die folgenden Betrachtungen auf den Fall versagensfreier Betriebserfahrung.

Voraussetzung 5 - Invarianz der Versagens-wahrscheinlichkeit:
Eine weitere notwendige Annahme zur Anwendung der Theorie setzt eine invariante Versagenswahrscheinlichkeit p über dem gesamten Eingaberaum voraus. Nach der Theorie von Eckhardt und Lee (siehe [3]) ist diese Annahme für große Systeme mit entsprechend hoher Funktionsvielfalt im Allgemeinen unrealistisch, da in solchen Systemen die Problemkomplexität über dem gesamten Eingaberaum stark variieren kann. Die Annahme wird allerdings bei Betrachtung einzelner Ausführungspfade bzw. feingranularer Komponenten beschränkter Funktionalität realistischer, da man hier von einer geringeren Varianz der Eingabekomplexität ausgehen kann.

Falls ein bestimmter Umfang n an Test- bzw. Betriebsfällen gesammelt wurde und die oben genannten Voraussetzungen als erfüllt betrachtet werden können, lässt sich anhand der statistischen Stichprobentheorie folgender Zusammenhang zwischen n, $\tilde{p}$ und β definieren (siehe [1, 4]):

$$(1-\tilde{p})^n = 1-\beta \qquad (2)$$

Umgekehrt lässt sich somit die erforderliche Anzahl an erfolgreich bearbeiteten Testfällen bestimmen, um Aussage (1) für $\tilde{p} \ll 1$ nachzuweisen:

$$n \cong \frac{\ln(1-\beta)}{-\tilde{p}} \qquad (3)$$

Beispielsweise wären 46 052 korrekt ausgeführte Test- bzw. Betriebsfälle nötig, um eine obere Schranke von $\tilde{p} = 10^{-4}$ mit Aussagesicherheit $\beta = 0.99$ nachzuweisen.
Für die Aussagesicherheit β gilt damit

$$\beta = P(p \le \tilde{p}) = 1 - \exp(-n \cdot \tilde{p}) \qquad (4)$$

3. Extraktion statistisch relevanter Betriebserfahrung

Um den in Abschnitt 2 beschriebenen Ansatz auf die mit Komponenten gewonnene Betriebserfahrung anzuwenden, müssen die gesammelten operationalen Daten im Hinblick auf die Voraussetzungen 1-4 analysiert und gegebenenfalls adäquat gefiltert werden. Die folgende Leitlinie beschreibt hierfür die wesentlichen Schritte zur Extraktion statistisch relevanter operationaler Daten, sowie zu deren Auswertung und Ergänzung:

Schritt 1 - Identifikation der zu bewertenden Systemkomponente:
Hierbei muss genau abgegrenzt werden, auf welchen Ausführungspfad bzw. welche Teilfunktionalität sich die angestrebte Zuverlässigkeitsaussage bezieht.

Schritt 2 - Identifikation betrieblich unabhängiger Abläufe:
Im Hinblick auf Voraussetzung 2 müssen „gedächtnislose“ Ausführungssequenzen bestimmt werden, das heißt Folgen von Operationen, deren Verhalten von vorhergehenden Operationen nicht abhängt.

Schritt 3 - Definition der Struktur eines relevanten Ablaufs:
Hierzu müssen alle relevanten Eingabeparameter identifiziert werden und Eingabeparameter ohne Einfluss auf die in Schritt 1 festgelegte Teilfunktionalität ausgegrenzt werden.

Schritt 4 - Bestimmung des Betriebsprofils:
In diesem Zusammenhang muss die Auftrittshäufigkeit einzelner funktionaler Anforderungen an die Software im Betrieb ermittelt werden.

Schritt 5 - Filterung der operationalen Daten:
Im Hinblick auf die Voraussetzungen 1 und 3 muss aus den vorliegenden operationalen Daten eine unabhängige Teilmenge durch Entfernung betrieblich nicht repräsentativer bzw. statistisch abhängiger Abläufe extrahiert werden.

Schritt 6 - Validierung der Daten:
Dies betrifft die Sicherstellung des korrekten Verhaltens entsprechender Abläufe hinsichtlich Voraussetzung 4.

Schritt 7 – Zuverlässigkeitsbewertung:
An diesem Punkt wird die Zuverlässigkeitsaussage für das Gesamtsystem gemäß Abschnitt 2 ermittelt. Im Falle komponentenbasierter Systeme können die in [13, 14] hergeleiteten Verfahren angewendet werden.

Schritt 8 - Ergänzung der Betriebserfahrung:
Falls die extrahierte Betriebserfahrung nicht ausreicht, um eine vorgegebene Zuverlässigkeitskenngröße nachzuweisen, müssen zusätzliche Testfälle generiert werden, welche ebenfalls die Voraussetzungen 1-4 erfüllen müssen. Hierbei kann das in [14] beschriebene Verfahren verwendet werden, um den Umfang der zusätzlichen Testfallmenge im Falle komponentenbasierter Systeme zu minimieren.

4. Anwendung für eine Getriebesteuerungs-Software

Am praktischen Beispiel einer software-basierten Getriebesteuerung wird in diesem Abschnitt gezeigt, wie die in Abschnitt 3 beschriebe Leitlinie im Hinblick auf die Analyse und Extraktion relevanter Betriebserfahrung angewendet werden kann. Im Rahmen eines industriellen Forschungsprojektes wird hierzu eine Studie der Universität Erlangen-Nürnberg in Kooperation mit dem Automobilzulieferer ZF Friedrichshafen AG durchgeführt. Aus Gründen der Geheimhaltung firmeninterner Informationen sind die Daten im Folgenden anonymisiert dargestellt.

Die Software steuert zwölf Vorwärtsgänge und zwei Rückwärtsgänge, welche entweder manuell vom Fahrer oder automatisch durch eine Software-Komponente (der so genannten „Fahrstrategie“) gewählt werden. Neben dem Ist-Gang (im Folgenden durch IG abgekürzt) und dem Soll-Gang (im Folgenden durch SG abgekürzt) sind für die Funktionalität der Steuerung weitere Parameter ρ_i, wie beispielsweise die momentane Geschwindigkeit und Gaspedalstellung, relevant.

Während vorangegangener Testfahrten wurde ein großer Umfang an Daten über die relevanten Parameter in Abhängigkeit von der Zeit aufgezeichnet. Tabelle 1 zeigt hierzu exemplarisch aufgezeichnete Werte, wobei die Gänge durch Buchstaben (a, b, c, …, m) und die Parameter ρ_i durch Prozentangaben kodiert sind.

Tabelle 1: Ausschnitt aus den aufgezeichneten Daten

Zeit	SG
…	…
5.9	d
6.0	d
6.1	e
6.2	e
6.3	e
6.4	e
6.5	e
6.6	e
6.7	e
6.8	e
6.9	e
…	…

Zeit	IG
…	…
35.9	d
36.0	d
36.1	d
36.2	d
36.3	d
36.4	d
36.5	d
36.6	d
36.7	d
36.8	e
36.9	e
…	…

Zeit	ρ_1
…	…
25.9	10.60
26.0	11.29
26.1	11.70
26.2	11.90
26.3	12.01
26.4	12.01
26.5	11.80
26.6	12.90
26.7	12.51
26.8	12.60
26.9	12.80
…	…

…

Basierend auf diesen Daten wurde die Leitlinie wie folgt umgesetzt:

Schritt 1 - Identifikation der zu bewertenden Systemkomponente:
Die zu Grunde liegende Architektur (siehe Abbildung 1) wurde zunächst im Hinblick auf die Abgrenzung der zu bewertenden Funktionalität untersucht. Nach Absprache mit den Entwicklern wurde die Bewertung auf die reine Schaltfunktionalität (Getriebesteuerung) eingeschränkt. Das heißt, die „Fahrstrategie“ ist nicht Gegenstand der Untersuchung.

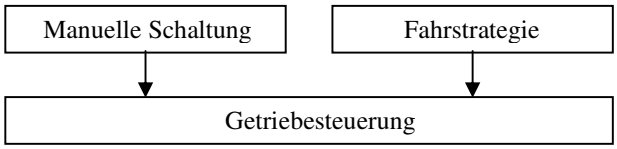

Abbildung 1: Systemarchitektur

Schritt 2 - Identifikation betrieblich unabhängiger Abläufe:
Zur Sicherstellung „gedächtnisloser“ Schaltsequenzen (siehe Voraussetzung 2), also Schaltungen, die von vorhergehenden Abläufen unabhängig sind, ist eine Initialisierungsphase notwendig, in der bestimmte Hintergrundparameter zunächst kalibriert werden. Danach können die Schaltungen als „gedächtnislos“ betrachtet werden. Wie in Abbildung 2 illustriert, ist die Funktionalität der Schaltung von c nach d dann unabhängig davon, ob vorher Gang a oder b eingelegt war.

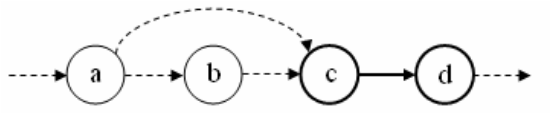

Abbildung 2: Gedächtnislose Schaltungen nach einer Initialisierungsphase

Schritt 3 - Definition der Struktur eines relevanten Ablaufs:
Relevante Betriebsfälle können zu jedem Zeitpunkt identifiziert und extrahiert werden, in dem ein neuer Schaltbefehl gegeben wird. Darüber hinaus wurden insgesamt vier weitere Parameter ρ_1, ρ_2, ρ_3 und ρ_4 identifiziert, welche für die Schaltfunktionalität zu diesen Zeitpunkten relevant sind. Tabelle 2 zeigt hierzu einen Ausschnitt aus den gesammelten Daten, wobei hier ein relevanter Betriebsfall zum Zeitpunkt 926.8 identifiziert werden kann, welcher zum Zeitpunkt 927.5 erfolgreich abgeschlossen wurde.

Tabelle 2: Ausschnitt aus den aufgezeichneten Daten mit einem relevanten Betriebsfall

Zeit	SG	IG	ρ_1	ρ_2	ρ_3	ρ_4
…	…	…	…	…	…	…
926.0	g	g	4.50	0.4	21.6	0
926.1	g	g	4.19	0.4	21.6	0
926.2	g	g	3.89	0.4	21.6	0
926.3	g	g	3.69	0.8	21.6	0
926.4	g	g	3.39	0.4	21.6	0
926.5	g	g	3.00	0.4	21.6	0
926.6	g	g	3.00	0.4	21.6	0
926.7	g	g	2.50	0.4	21.6	0
926.8	**f**	**g**	**2.39**	**0.4**	**21.6**	**0**
926.9	f	g	2.00	0.0	21.6	0
927.0	f	g	2.00	0.0	21.6	0
927.1	f	g	2.00	0.4	21.6	0
927.2	f	g	1.09	0.4	21.2	0
927.3	f	g	1.09	0.4	21.2	0
927.4	f	g	1.09	0.4	21.2	0
927.5	f	f	1.09	0.0	21.2	0
927.6	f	f	1.00	0.0	21.2	0
…	…	…	…	…	…	…

Auf Basis der definierten Struktur wurden die operationalen Daten gefiltert und alle Schaltbefehle mit den zugehörigen Parameterwerten extrahiert (siehe Tabelle 3).

Tabelle 3: Ausschnitt aus den extrahierten Betriebsfällen

Zeit	SG	IG	ρ_1	ρ_2	ρ_3	ρ_4
…	…	…	…	…	…	…
5940,6	k	l	64,35	88,8	0,0	0
6012,3	l	j	57,55	0,4	0,0	0
6016,2	j	h	42,23	0,4	16,0	0
…	…	…	…	…	…	…

Schritt 4 - Bestimmung des Betriebsprofils:
Auf Basis der extrahierten Betriebsfälle wurde das Betriebsprofil in zwei Schritten erstellt. Zunächst wurden hierfür die Auftrittsfrequenzen einzelner Schaltkombinationen (IG, SG) im Betrieb ermittelt (siehe Ausschnitt Tabelle 4 und Abbildung 3).

Tabelle 4: Ausschnitt der Auftrittsfrequenzen einzelner Schaltkombinationen

	d	e	f	g	h	i	…
…	…	…	…	…	…	…	…
d	--	6.06	16.13	0.00	1.35	0.00	…
e	…	--	16.13	13.33	0.00	0.00	…
f	…	6.06	--	18.33	17.57	1.10	…
g	…	63.64	19.35	--	25.68	15.38	…
h	…	0.00	45.16	33.33	--	25.27	…
i	…	0.00	0.00	33.33	43.24	--	…
j	…	0.00	0.00	0.00	8.11	52.75	…
…	…	…	…	…	…	…	…

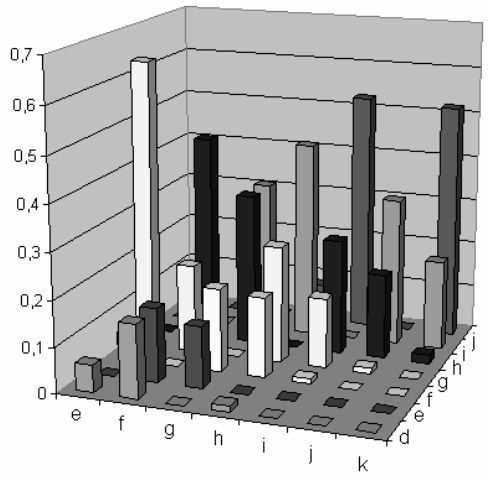

Abbildung 3: Grafische Darstellung der Auftrittsfrequenzen einzelner Schaltkombinationen

Darauf aufbauend wurde für jeden Schaltbefehl (IG, SG) das Profil jedes zugehörigen Parameters ρ_i, $i \in \{1\ldots4\}$ durch eine toolgestützte Anwendung von Verteilungsanpassungstechniken [5] ermittelt. Diese basieren auf

- der Auswahl einer Verteilungsfamilie,
- der Parameterbestimmung,
- der „Goodness-of-fit“ Bewertung.

Abbildung 4 zeigt hierzu ein Beispiel für einen “density / histogram overplot“ des Parameters ρ_1 für den Schaltbefehl (CG,DG) = (k,j).

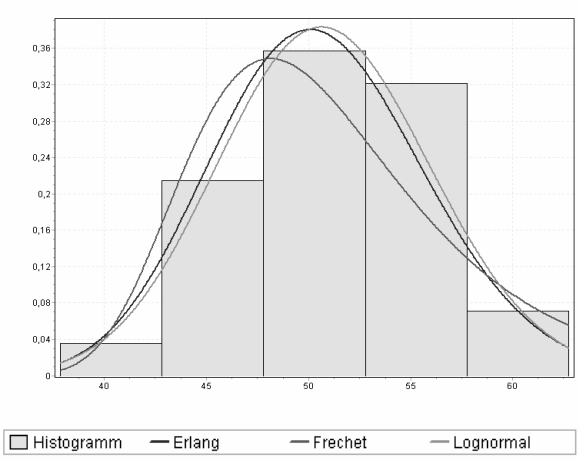

Abbildung 4: Density / histogram overplot für Parameter ρ_1 und (CG,DG) = (k,j)

Tabelle 5: “Goodness-of-fit” Test für Parameter ρ_1 und (CG,DG) = (k,j)

Verteilung: Fréchet				
Verteilungsparameter: $0.17241\cdot10^{-9}$; $0.90725\cdot10^{-9}$; $-0.90725\cdot10^{-9}$				
Kolmogorow-Smirnow Test				
α	0,1	0,05	0,02	0,01
Kritischer Wert	0,22497	0,24993	0,27942	0,29971
Ablehnen?	Nein	Nein	Nein	Nein
Anderson-Darling Test				
α	0,1	0,05	0,02	0,01
Kritischer Wert	1,9286	2,5018	3,2892	3,9074
Ablehnen?	Nein	Nein	Nein	Nein
χ^2 Test				
α	0,1	0,05	0,02	0,01
Kritischer Wert	4,6052	5,9915	7,824	9,2103
Ablehnen?	Nein	Nein	Nein	Nein

Die Anpassungsgüte wurde anschließend jeweils durch die Anwendung klassischer “Goodness-of-fit”-Tests bewertet. Hierzu gehören der Kolmogorow-Smirnow Test [5], der Anderson-Darling Test [5] und der χ^2 Test [5], wie in Tabelle 5 bezüglich der Fréchet Verteilung zu sehen ist.

In Fällen wo die Anpassung zu einer generischen Verteilung nicht möglich war, wurde eine empirische Verteilung durch lineare Interpolation der gesammelten Werte definiert.

Schritt 5 - Filterung der operationalen Daten:
Im Hinblick auf die Voraussetzung einer statistisch unabhängigen Stichprobe muss aus den vorliegenden operationalen Daten eine Teilmenge durch Entfernung betrieblich nicht repräsentativer bzw. statistisch abhängiger Abläufe extrahiert werden. Hierzu wird momentan ein Verfahren entwickelt, welches es erlaubt, operationale Daten im Hinblick auf zu Grunde liegende Korrelationen zu analysieren und eine weitestgehend unkorrelierte Teilmenge durch Anwendung heuristischer Optimierungsverfahren zu extrahieren.

Schritt 6 - Validierung der Daten:
Die Validierung der Daten (also die Sicherstellung korrekt durchgeführter Schaltoperationen) wird durch ZF Friedrichshafen AG erfolgen und basiert auf einer Reihe von Kriterien, wie beispielsweise der Einlegung eines Gangs innerhalb einer vorgegebenen Zeit.

Schritt 7 – Zuverlässigkeitsbewertung:
Die Zuverlässigkeitsbewertung kann schließlich durch die in Abschnitt 2 (bzw. die in [13, 14]) beschriebene Theorie erfolgen.

Schritt 8 - Ergänzung der Betriebserfahrung:
Falls die extrahierte Betriebserfahrung nicht ausreicht, um eine vorgegebene Zuverlässigkeitskenngröße nachzuweisen, können zusätzliche Testfälle auf Basis des erstellten Betriebsprofils automatisch generiert werden.

5. Zusammenfassung

In diesem Artikel wurde eine allgemeine Leitlinie zur Analyse und Extraktion statistisch relevanter Daten aus operationalen Informationen vorgestellt. Die Anwendung wurde anhand einer software-basierten Getriebesteuerung aus einem laufenden industriellen Forschungsprojekt illustriert. Die vorgestellte systematische Vorgehensweise bietet den Vorteil, den i. a. mit statistischen Verfahren verbundenen hohen Testaufwand durch Auswertung bereits vorliegender Betriebserfahrung deutlich zu reduzieren.

Literatur

[1] Amman, P. E., Brilliant, S. B., Knight, J. C.: The Effect of Imperfect Error Detection on Reliability Assessment via Life Testing, IEEE Transactions on Software Engineering, V. 20, No. 2, Februauy 1994.

[2] Butler, R., Finelli, G.: The Infeasibility of Quantifying the Reliability of Life-critical Real-time Software, Software Engineering, 19(1), 1993.

[3] Eckhardt, D.E., Lee, L.D.: A Theoretical Basis for the Analysis of Multiversion Software Subject to Coincident Errors, IEEE Transactions on Software Engineering, Vol. SE-11, No. 12, December 1985.

[4] Ehrenberger, W.: Software-Verifikation, Hanser Verlag, 2002.

[5] Law, A. M., Kelton, W. D.: Simulation, Modeling and Analysis, McGraw-Hill, 2000.

[6] Littlewood, B., Strigini, L.: Validation of Ultra-high Dependability for Software-based Systems, Communications of the ACM, 36(11), 1993.

[7] Littlewood, B., Strigini, L.: Software Reliability and Dependability: A Roadmap, The Future of Software Engineering, ACM Press, 2000.

[8] Littlewood, B., Wright, D.: Stopping Rules for Operational Testing of Safety Critical Software, Proc. 25th International Symposium Fault Tolerant Computing (FTCS 25), Pasadena, CA 1995.

[9] Miller, K. W., Morell, L. J., Noonan, R. E., Park, S. K., Nicol, D. M., Murrill, B. W., Voas, J. F.: Estimating the Probability of Failure When Testing Reveals No Failures, IEEE Transactions on Software Engineering, V. 18, No. 1, January 1992.

[10] Parnas, D., van Schouwen, J., Kwan, S.: Evaluation of Safety-critical Software, Communications of the ACM, 33(6), 1990.

[11] Quirk, W. J. (ed.): Verification and Validation of Real-time Software, Springer-Verlag, 1985.

[12] Saglietti, F.: Evaluation of Pre-developed Software for Usage in Safety-critical Systems, 26th Euromicro Conference (EUROMICRO 2000), IEEE, 2000.

[13] Söhnlein S., Saglietti F.: Nachweis hoher Softwarezuverlässigkeit auf der Basis von Test- und Betriebserfahrung mit wiederverwendbaren Komponenten, Proc. Sicherheit 2008, Lecture Notes in Informatics, Gesellschaft für Informatik, 2008.

[14] Söhnlein, S.; Saglietti, F.: Auswertung der Betriebserfahrung zum Zuverlässigkeitsnachweis sicherheitskritischer Softwaresysteme, Proc. Automotive 2008 - Safety & Security, Sicherheit und Zuverlässigkeit für automobile Informationstechnik, Stuttgart, 2008.

[15] Störmer, H.: Mathematische Theorie der Zuverlässigkeit, Oldenbourg, 1970Zuverlässigkeit für automobile Informationstechnik, Stuttgart, 2008.