Clusterforschungsprojekt FEST
Abschlussbericht

Zuwendungsempfänger	TU-Ilmenau
Förderkennzeichen	01M3072A
Vorhabenbezeichnung	Funktionale Verifikation von Systemen (FEST)
Titel	Abschlussbericht

Inhalt

1 Kurzdarstellung

1.1 Aufgabenstellung

Ziel des Teilprojektes „Zeitverifikation“ des Clusterforschungsprojektes FEST war die Erforschung von Methoden zur Modellierung und Analyse von funktionalen Eigenschaften, speziell Zeiteigenschaften auf Systemebene, um eine Prüfung bereits vor den Implementierungsschritten vornehmen zu können. Die Sicht auf die Systemebene ist in diesem Zusammenhang dadurch gekennzeichnet, dass ein zu Produkt in seiner Gesamtheit betrachtet wird, um Aussagen über die Auswirkung des Zusammenspiels von Einzelkomponenten durchführen zu können. Entsprechend wurden im Rahmen des Antrags drei Aufgaben definiert.

A 1.3: Zeiteigenschaften und Restriktionen in Modellen

Gegenstand der Aufgabe war die Erforschung von Notationen in Modellierungen, um eine Beschreibung und Analyse von Systemrestriktionen, speziell Zeiteigenschaften, in Systemmodellen zu einem frühen Entwicklungszeitpunkt zu ermöglichen.
Ziel der Aufgabe war die Entwicklung einer Beschreibungsmethodik, welche die Syntax und Semantik, sowie Struktur und Hierarchiebeziehungen in den betreffenden Modellen unterstützt. Hierbei waren insbesondere Modelle des Multidomänenwerkzeugs MLDesigner zu berücksichtigen.

A 2.2: Verfeinerung von Zeitrestriktionen

Gegenstand der Aufgabe war die Erforschung der Verfeinerung von Zeiteigenschaften bei gleichzeitiger Verfeinerung funktionaler Blöcke.
Ziel des Arbeitspaketes war die Entwicklung einer Methodik, um bei einer Verfeinerung von Funktionen durch hierarchische Blöcke eine Abbildung von Zeitrestriktionen zu ermöglichen.

A 2.3: Transformation von zeitbewerteten Modellen

Gegenstand der Aufgabe war die Erforschung von Transformationen aus Beschreibungsmitteln in mathematische Analysemodelle.
Ziel der Aufgabe war die Entwicklung einer Transformationsmethodik, um Multi-Domänen-Modelle des Werkzeugs MLDesigner in mathematische Beschreibungsmodelle, wie zeitbewertete höhere Petrinetze bzw. zeitbewertete Automaten zum Zweck der Analyse überführen zu können.

1.2 Voraussetzungen

Voraussetzung für die Durchführung des Vorhabens war die Verfügbarkeit einer Simulationsumgebung, welche für die Validierung von ausführbaren Spezifikationen geeignet ist. Ansätze unter Verwendung der UML [UML04] sind aus verschiedenen gründen nicht geeignet. Zum einen fehlt eine Standardisierung der Action Language, des Weiteren gibt es Mehrdeutigkeiten und die Interoperabelität ist derzeit noch nicht vollständig gegeben [RuHQJZ05]. Der Ansatz gemäß SystemC ist noch in der Entwicklungsphase, eine leistungsfähige Entwicklungsumgebung und Debugmöglichkeiten auf Modellebene sind nicht gegeben.

1.3 Planung und Ablauf des Vorhabens

Zeitplanung

Die Planung der Meilensteine zu den Aufgaben erfolgte in zwei Arbeitspaketen. Das Arbeitspaket AP1 „Modellierung und Abstraktion“ umfasst diejenigen Aufgaben, die sich mit der Modellierung und Abstraktion beschäftigen. Im Rahmen von Verifikationsaktivitäten also mit der Abbildung von Simulationsmodellen in Modelle, die zur Durchführung der Verifikation selbst genutzt werden können. Das Arbeitspaket AP2 „Methoden und Integration“ umfasst Aufgaben, welche die Verifikationsmethoden bzw. Algorithmen selbst betreffen.

Die Bearbeitung der Aufgaben wird jeweils in Phasen mit den Aktivitäten Erforschen, Integrieren und Validieren eingeteilt, wobei sich das Integrieren mit dem Erforschen und Validieren überlappt. Diese Einteilung spiegelt sich auch in der Meilensteinplanung gemäß dem Projektantrag, siehe Tabelle 1 und Tabelle 2,wieder.

AP1: Modellierung und Abstraktion

Meilenstein	Beschreibung	Art	Partner	Termin
M1-A1.3-1	Konzeption der Constraint Language	Bericht	TUI	Q2
M1-A1	Spezifikation einer Modellierungsumgebung	Bericht	Alle in AP1	Q4
M2-A1	Methodik zur Modellierung der Modellierung und Abstraktion	Bericht	Alle in AP1	Q8
M3-A1	Ergebnisse der Validierung der Modellierungs- und Abstraktionstechniken	Bericht	Alle in AP1	Q12
M3-A1.3-1	Validierung des implementierten Konzepts zur Behandlung von Zeiteigenschaften und deren Restriktionen	Bericht, Software	TUI	Q12

Tabelle 1: Übersicht der Meilensteine zum Arbeitspaket 1

AP2: Methoden und Integration

Meilenstein	Beschreibung	Art	Partner	Termin
M1-A2	Spezifikation der Algorithmen und Verfahren	Bericht	Alle in AP2	Q4
M1-A2.2-1	Spezifikation des Dekompositionsverfahrens unter Verwendung der Constraint Language für die Zeitverifikation	Bericht	TUI	Q6
M1-A2.3-1	Spezifikation der Algorithmen und Verfahren mit Hilfe der Modellierungen aus AP1 und Untersuchung der Transformationenverfahren an Beispielmodellen	Bericht	TUI	Q6
M2-A2	Integration der Verfahren	Software	Alle in AP2	Q8
M2-A2.3-1	Implementierung der Transformationsverfahren	Software	TUI	Q10
M3-A2	Ergebnisse der validierten Verfahren und Algorithmen	Bericht	Alle in AP2	Q12
M3-A2.2-1	Implementierung und Validierung des Dekompositionsverfahrens für die Zeitverifikation	Bericht, Software	TUI	Q12
M3-A2.3-1	Validierung der implementierten Verfahren und Algorithmen	Bericht	TUI	Q12

Tabelle 2: Übersicht der Meilensteine zum Arbeitspaket 2

Personalbesetzung

Für Bearbeitung der Aufgaben wurde ein Personalaufwand von 36 Personenmonaten veranschlagt, welche durch Arbeiten von studentischen bzw. wissenschaftlichen Hilfskräften ergänzt wird.

Finanzielle Mittel

Die laut dem Projektantrag und Bewilligungsbescheid geplante Verwendung finanzieller Mittel gemäß der Aufschlüsselung nach den Richtlinien für Zuwendungsanträge auf Ausgabenbasis ist in Tabelle 3 dargestellt.

Angaben in EUR	Bewilligte Mittel (Gesamt)
Gesamtausgaben	**194.900,00 €**
A: Personalausgaben (Projekt)	
812	150.400,00 €
822	24.000,00 €
B: Sächliche Ausgaben (Projekt)	
843	9.500,00 €
846	7.500,00 €
850	3.500,00 €

Tabelle 3: Ausgabenplanung

1.4 Wissenschaftlicher Stand, an dem angeknüpft wurde

Stand der Wissenschaft und Technik

Für die Zeitverifikation im hierarchischen Mehrebenenentwurf existieren derzeit Verfahren zur bevorzugten Anwendung in den unteren Abstraktionsebenen. Diese eignen sich nicht für den Einsatz auf Systemebene und ermöglichen dort nur eine stark reduzierte Verifikation. Wichtige Vorläuferarbeiten sind RAVEN [Ruf99], welches die Zeitverifikation mit Clocked- Computation-Tree-Logic und diskreten Zustandsautomaten bearbeitet. UPPAAL [BeDL04] ist ein Werkzeug zur Zeitverifikation, das mit Hierarchical-Timed-Automata und diskreten Intervallen arbeitet. Weitere Forschungsergebnisse sind in den Prototypen T-MSC eingeflossen [FeKDHS01], [FeHFS02], das die Zeitverifikation mit so genanten Message-Sequence-Charts und die Überführung in Zeitintervall-Petrinetze ermöglicht. Der Entwurf auf Systemebene erfolgt domänenübergreifend. Für die Zeitverifikation auf Systemebene wird deshalb die Integration unterschiedlicher Modellierungsdomänen benötigt. Diese Eigenschaft fehlt allen beschriebenen Prototypen. Auch existiert keine ausreichende Unterstützung in einem EDA-Tool. Die im hierarchischen Mehrebenenentwurf bestehende Notwendigkeit zur Unterstützung mehrerer Abstraktionsebenen im Verifikationsprozess ist ebenfalls im heutigen Stand der Technik nicht ausreichend vorhanden.

Bisherige Arbeiten des Projektpartners

Im Institut Theoretische und Technische Informatik der TU Ilmenau werden Grundlagen für Entwurfssysteme für komplexe eingebettete Systeme (z.B. autonome mobile Systeme [Ze03], Nanopräzisionsmaschinen [FeDDL03] und SoC) auf Missions- und Systemlevel für verschiedene Entwurfsdomänen erforscht. Gegenstand sind beispielsweise Multi-Level- und Multi-Resolution-Technologien für die Modellierung sowie effiziente Simulationsalgorithmen für Multi-Domain-Umgebungen. Diese waren

und sind Basis des Design-Tools ML-Designer der MLDesign GmbH Ilmenau. Weiterhin existieren Forschungsergebnisse auf dem Gebiet der Modellierung, Verifikation und Modelltransformation von kontinuierlich-diskreten Systemen [FrZ03] unter Berücksichtigung der Modellvielfalt und ihrer zeitlichen Eigenschaften, die im Wesentlichen im DFG-Schwerpunkt 1040 [FeKDHS01] und im Graduiertenkolleg GRK 164/1-96 entstanden. Beispielsweise wurden umfangreiche Untersuchungen zur Verifikation technischer Systeme mit Hilfe zeitintervall-bewerteter Petrinetze durchgeführt. Zusammenfassend resultieren umfangreiche Erfahrungen sowohl im Bereich der Erstellung und Verifikation von Modellen als auch bezüglich Modellierungswerkzeugen und Modellierungstechnologien.

1.5 Zusammenarbeit mit anderen Stellen

Im Rahmen des Projektes FEST wurden Kooperationen mit den Forschungspartnern entsprechend der definierten Schnittstellen durchgeführt. In Kooperation mit der Universität Tübingen wurde untersucht inwiefern das Modell (Syntax und Semantik) des tübinger Model-Checkers RAVEN für die Prüfung von Systemmodellen entsprechend der Semantik von Modellen des Werkzeugs ML Designer geeignet ist. Dabei hat sich herausgestellt, dass die Transformation, wegen einer Zusätzlichen Modellierung eines Konzeptes für Zeitvariablen, sehr aufwendig ist. Die Erstellung einer Transformation von allgemeinen Timed Automata in Modelle der Werkzeuge RAVEN bzw. SystemC würde eine eigenständige, komplexe Forschungsarbeit darstellen.
Mit der Universität Frankfurt am Main wurde untersucht, inwiefern eine Kompatibilität zwischen der Eigenschaftsbeschreibung für Modelle des Werkzeugs MLDesigner und der Eigenschaftsbeschreibung für das Model-Checking analoger bzw. hybrider Systeme gegeben ist. Zusammen mit der Universität Frankfurt und der Universität Tübingen wurde ein Konzept zur Modellierung von Mixed-Signal-Eigenschaften erarbeitet und veröffentlicht.
Mit der Firma MLDesign GmbH fand im Rahmen des Projektes ein Austausch bezüglich der Besonderheiten bezüglich der Semantik der Multi-Domänen-Modelle statt.

2 Eingehende Darstellung

2.1 Erzielte Ergebnisse

Im Folgenden wird auf die konkret erzielten Ergebnisse im Rahmen der Aufgaben A1.3, A2.2 und A2.3 eingegangen.

2.1.1 Aufgabe A1.3

Im Rahmen der Aufgabe A1.3 wurde eine Methodik zur Beschreibung von Zeiteigenschaften für Simulationsmodelle entwickelt. Die Möglichkeiten der Multi-Domänen-Simulation des Werkzeugs wurden mehrfach erfolgreich eingesetzt, um das funktionale Verhalten von Systemen während der Konzeptionsphase aufgrund bereits bekannter Kennwerte analysieren zu könnten. Eine Methodik für die systematische Analyse von Systemen unter Verwendung eines standardisierten Modellierungsverfahrens bezüglich der Modellstrukturierung stand jedoch erst im späteren Verlauf des Projektes zur Verfügung [BaHS07, Sa07]. Daher konnten Effekte, die sich ggf. aus einer Standardisierung von Architektur- und Ausführungskomponenten ergeben nicht berücksichtigt werden. Es ist zu erwarten, dass sich aus der Standardisierung der zu Untersuchenden Modellklasse vorteile für die Analyse ergeben. Sämtliche Betrachtungen beziehen sich daher auf das zur Simulation derartiger Modelle verwendete Berechnungsmodell der Simulationsdomäne Discrete-Eevent (DE). Auf die Abbildung des komplexen Typensystems wurde, im Hinblick auf die Komplexitätsbeschränkungen der genutzten Model-Checker verzichtet. Um eine geeignete Abstraktion zu ermöglichen wurde auf spezielle Modellelemente, wie spezielle Ereignisse und shared Memories verzichtet. Die Einbindung von speziellen Ereignissen ist durch eine Vorverarbeitung möglich. Shared Memories könnten zwar prinzipiell abgebildet werden, allerdings wäre hierzu noch die Erforschung eines geeigneten Konzepts für die Verarbeitung von komplexen Datentypen notwendig. Zeiten können während der Transfomation in ein ganzzahliges Wertesystem angebildet werden, bei Bedarf ist auch die Verwendung eines Einheitensystems möglich.

Die Zeiteigenschaften der entwickelten Methodik werden dabei auf Ereignissen an Modulinterfaces (Ports), also auf die Modellstruktur bezogen. Um die Kompatibilität zu den Multi-Domänenmodellen des Tools MLDesigner zu gewährleisten wurden die strukturellen Merkmale als Bezugspunkt festgelegt. Als besonders kompliziert hat sich der Umstand erwiesen, dass die Funktionalität elementarer Module von MLDesigner Modellen nicht in mathematisch interpretierbarer Form vorliegen. Dies ist dadurch begründet, dass der Informationsaustausch zwischen Simulationsmodell und Simulationskernel nicht symbolisch, sondern unter Verwendung einer Software-API erfolgt, und die Funktionalität selbst direkt als Quellcode realisiert ist. Daher ist es nicht ausreichend reine Zeitannotationen vorzunehmen. Die Beschreibung der Zeiteigenschaften umfasst neben den Zeitannotationen auch funktionale/temporale Aspekte. Die Eigenschaftsbeschreibung selbst wird als Template realisiert, so dass komplexere Eigenschaftsbeschreibungen durch das Definieren neuer Templates ergänzt werden können [PaFSV05]. Zudem ist es möglich für jedes Template, entsprechend der Verwendung eine Syntheseimplementierung und eine Analyseimplementierung zu hinterlegen. Durch den Bezug der Eigenschaftsbeschreibung zum Modell kann automatisch das richtige Template gewählt werden. Es ist möglich bestehende Bibliothekselemente von vorn herein mit Implementierungstemplates zu versehen. Durch das Einbinden von Modellparametern in die Constraintbeschreibung bleiben diese Templates parametrisierbar. Exemplarisch wurde die Anwendbarkeit der Modellierung von Systemeigenschaften und –Restriktionen durch Templates anhand von UML Modellen untersucht [Kl06].

Als ergänzender Ansatz für die Beschreibung von Zeiteigenschaften wurde die Verwendung einer zeitbeschränkten linearen temporalen Logik mit synchronisiertem kontinuierlichem Zeitmodell für Events (XFLTL) untersucht. Im Rahmen dieser Untersuchungen ist Software-Bibliothek zur simulationsbegleitenden Eigenschaftsprüfung (Assertion Checking) entstanden. Auch wenn derzeit noch Einschränkungen Anwendung dieser Bibliothek bestehen, kann der Ansatz als geeignet bewertet werden. Der Ansatz wurde u.a. genutzt, um Eigenschaften eines Mixed-Signal-Modells simulationsbegleitend zu prüfen [JeLP+07]. Prinzipiell könnte man die Beschreibung mittels XFLTL auch für die formale Verifikation verwenden. Es würde sich die Möglichkeit einer rein Formalen Eigenschaftsbeschreibung ergeben, die in späteren Entwicklungsphasen nahezu direkt auf PSL abbildbar wäre. Die Verwendung zur formalen Prüfung im aktuellen Projekt war aufgrund eines geeigneten Model-Checkers nicht möglich. Die Definition von Eigenschaften auf der Basis diskreter Ereignisse (XFLTL) ist zur Methodik zur Spezifikation und Prüfung von Mixed-Signal Eigenschaften kompatibel.

2.1.2 Aufgabe A2.2

Im Rahmen der Aufgabe A2.2 wurden Möglichkeiten der Verfeinerung von Zeitrestriktionen bei der Verfeinerung funktionaler Elemente untersucht. Während für Automatenmodelle Abstraktionsregeln existieren, die sich prinzipiell auch in umgekehrter Weise für die Verfeinerung eignen, muss man den zugrunde legenden Modellen beachten, dass diese sich vorwiegend auf Ereignisse beziehen. Für die Untersuchung von Systemmodellen spielt die bloße Verfeinerung einzelner FSM's jedoch keine entscheidende Rolle. Bei der Betrachtung auf Systemebene ist einer Verfeinerung von Zeitrestriktionen daher nur bei der gleichzeitigen Verfeinerung der Systemstruktur interessant.

Für die Verfeinerung allgemeiner temporaler Eigenschaften bei gleichzeitiger Verfeinerung der Modellstruktur konnte kein geeigneter Ansatz gefunden werden, der ausgehend von zeitbeschränkten temporalen Formeln entsprechend der XFLTL und Kenntnis der Modellstruktur einen Rückschluss auf die Erfüllung einer anderen XFLTL Formel erlaubt. Daher wurde die zur Dekompositionsanalyse nutzbaren Eigenschaften eingeschränkt.

Aufbauend auf die strukturelle Abhängigkeit zwischen Ereignissen wurden zwei Methoden der strukturellen Analyse von Verfeinerungen betrachtet, welche sich auf die Verfeinerung von Prozessen beziehen. Dabei wurde ausgehend von einer Methodik, die einzelne Ereignisse betrachtet eine Methodik entwickelt, die auch für Ereignisströme und in sich nebenläufige Prozesse geeignet ist.

Um Kausalität als strukturelle Eigenschaft zwischen Ereignissen zu modellieren wird oft die Abbildung auf eine Ereigniskette vorgenommen. Um diese eine solche Ereigniskette mittels Model-Checking zu erkennen müssen Ereignisinstanzen des betreffenden Typs dahingehend unterschieden werden, zu welcher Ereigniskettenistanz sie gehören. Dazu werden beispielsweise Transaktionssequenzen verwendet. Das in [PaFe07] vorgestellte Modell ist in der Lage solche Ereignisketten nachzuvollziehen ohne Transaktionssequenzen oder vergleichbare Mechanismen verwenden zu müssen. Aussagen über die Robustheit und Effizienz der vorgeschlagenen Methode stehen jedoch noch aus.

2.1.3 Aufgabe A2.3

In Aufgabe A2.3 wurde eine Methodik für die Transformation von Multi-Domänen-Modelle des Werkzeugs MLDesigner in Analysemodelle der zeitbewerteten Automaten entwickelt. Dazu wurde der Eigenschaftsgraph als eine Zwischendarstellung entwickelt, welche neben der kompletten Struktur alle für die Verifikation relevanten Informationen enthält. Auf der Basis des Eigenschaftsgraphen eine Normalisierung vorgenommen. Dies bedeutet, dass Eigenschaften die sich auf Signale verschiedener

Hierarchieebenen beziehen entsprechend zugeordnet werden und anschließend eine Auflösung der Hierarchie unter Beibehaltung der Verknüpfungen zwischen Funktionen und Eigenschaften durchgeführt wird.
Die Methodik zur Modelltransformation wurde auf zwei Ebenen betrachtet. Zunächst wurde untersucht welche Optionen für die Modelltransformation auf informationstechnischer Sicht bestehen. Dabei zeigten sich regelbasierte Transformationen für die meisten Aufgaben als besonders geeignet. Dazu werden Ausgangs- und Zielmodell mit der Syntax und Semantik eines gemeinsamen Metamodells beschrieben. Anschließend ist, eine Beschreibung von Regeln auf der Semantik des Metamodells möglich. Vorteil dieser Methode ist, dass die Regelbeschreibung im Vergleich zur direkten Implementierung sehr kompakt ist. Daher wurde zunächst ein Framework für die Modelltransformation erstellt, das eine Kaskadierung mehrerer Transformationsschritte, sowie eine bidirektionale Transformation erlaubt [Mu06]. In einem zweiten Schritt wurden die Modelle und Regeln implementiert und die Generatoren für die Erzeugung des Formalen Modells auf den Templates angekoppelt. Das verwendete Framework ist also relativ einfach auf andere Modelle und/oder Transformationsregeln anpassbar. Das Transformationstool wird in [MuPF07] vorgestellt. Die nachfolgenden Ausführungen beziehen sich auf die Gesamtmethodik im Kontext der Verifikation.

Als Modellierungsdomänen werden DE, FSM und SDF unterstützt, wobei für die Beschreibung der Funktionalität der SDF Domänen eine geeignete Zustandsbeschreibung vorliegen muss. Zur Wahrung der Semantik muss die oberste Hierarchieebene als Modellelement der DE Domäne definiert sein. Für die Transformation normalisierten Eigenschaftsgraphen ist die Überführung Struktur sowie der Funktionsbeschreibungen und Eigenschaftsbeschreibungen erforderlich. Dabei werden die im Rahmen der Aufgabe 1.3 entwickelten Templates in das Zielmodell überführt.

Als Zielmodelle werden zeitbewertete Automaten, speziell die Uppaal Timed Automata unterstützt. Für Intervall-Petrinetze ist eine Transformation definiert worden. Da für Intervall-Petrinetze keine Unterstützung für hierarchische Modelle verfügbar war, wurde von einer Implementierung dieser Transformationsvariante abgesehen, eine Erweiterung ist jedoch leicht möglich. Für beide Zielmodelle existieren Model-Checker, die eine Prüfung temporaler Eigenschaften ermöglichen. Zwar existieren für beide Zielmodelle keine Möglichkeiten Zeitbeschränkungen auf der Basis temporaler Zeiteigenschaften direkt zu prüfen, durch die Generierung von Monitoren für die zu analysierenden Zeiteigenschaften ist jedoch eine Prüfung auf der Basis einer Erreichbarkeitsanalyse möglich. Für die zeitbewerteten Automaten ist diese Einschränkung zu relativieren, da hier für die Beobachtung von Ereignissen im Rahmen des Model-Checking nur durch die Beobachtung von Synchronisationszuständen möglich ist.

Wichtiger ist, dass die Semantik der Ereignisdiskreten Simulation weitestgehend zur Semantik des Zielmodells kompatibel ist. Dies betrifft die Möglichkeit Ereignisse als eigenständige Modellelemente modellieren zu können. Dazu stehen bei den zeitbewerteten Automaten Channels zur Verfügung, die sich wie Ereignisse verhalten. Bei den Intervall-Petrinetzen können Transitionen äquivalent verwendet werden. Zudem können voneinander abhängige Ereignisse ohne Zeitverzug erzeugt werden. Dies ist für die Kopplung verschiedener Domänen, aber auch für die Trennung von Funktion und Zeit erforderlich, wie sie zur effizienten Simulation eingesetzt wird. Bei der Untersuchung von Model-Checkern für digitale Hardware war insbesondere die Modellierung von Zeiten durch die Synchronisation auf einen globalen Takt problematisch. Zwar besteht auch die Möglichkeit Zeitvariablen als Zähler separat zu modellieren, und zeitlose Zustandsveränderungen über kombinatorische Netzwerke zu realisieren, hierbei würde jedoch ebenfalls die Möglichkeit der Prüfung zeitbeschränkter temporaler Eigenschaften verloren gehen.

Das Problem der direkten Prüfung von Zeiteigenschaften in den Timed Automata und Intervall-Petrinetzen ergibt sich in der Art und Weise, wie Zeiten auf der Ebene des Zustandsraumes abgebildet werden. Zeiten werden durch Clock Zones, als konvexe Mengen von Zeitvariablen, modelliert. Für

die Zustandsübergänge sind anschließend nur noch relative Zeitintervalle bekannt. Summiert man diese Pfade, um Zeitinformationen zu rekonstruieren erhält man verfälschte Ergebnisse, da Wechselwirkungen zwischen Zeitvariablen nicht beachtet werden. Die im Rahmen des Projektes entwickelte Analysemethode [PaFe07] enthält diese Einschränkung nicht, erfordert jedoch eine größere Komplexität und wird derzeit noch validiert.

Sämtliche auf Model-Checking basierten Analyseverfahren benutzen entweder ein synchrones Zeitmodell oder Zeitvariablen, welche den Funktionselementen zugeordnet werden. Im Zeitmodell des Simulationswerkzeuges werden Zeiten als Eigenschaften den Ereignissen zugeordnet, so dass zu einem Zeitpunkt eine beliebige endliche Anzahl von Instanzen eines Ereignistyps simultan mit verschiedenen Zeitbezügen existent sein kann (sich in der Warteschlange befindet). Dieses nebenläufige Verhalten kann im Analysemodell durch Parameter statisch abgebildet werden. Mögliche Inkonsistenzen, die aus einem zu geringen Grad der Nebenläufigkeit entstehen können während der Analyse erkannt werden. Das entwickelte Beschreibungsmodell [PaFe07] besitzt diese Einschränkung nicht, es können mehrere simultan aktive Ereignisinstanzen abgebildet werden.

2.2 Voraussichtlicher Nutzen, Verwertbarkeit im Verwertungsplan

Als Ergebnis des Projektes steht eine prototypisch Methode zur Analyse von Zeiteigenschaften für Multi-Domänen-Modelle des Werkzeugs MLDesigner zur Verfügung. Wesentliche Betrachtungen zu den verschiedenen Arbeiten können auch auf abstrakte Modelle (Transaction Level) der Modellierungssprache SystemC bezogen werden. SystemC verwendet ein ähnliches, ereignisorientiertes Modellierungskonzept und enthält ähnliche Einschränkungen im Bezug auf die Modellierung von Funktionsblöcken durch C++ Quellcode und die Verwendung komplexer Datentypen.
Es haben sich im Verlauf des Forschungsprojektes neue Fragestellungen ergeben. Für zukünftige Forschungsprojekte gibt es Anknüpfungspunkte an die entwickelte Methodik. Durch Fortschritte im Bereich der automatisierten Datenabstraktion und die Definition einer mathematisch interpretierbaren Sprache für die Funktionsbeschreibung würde sich ein weites Einsatzgebiet im industriellen Umfeld ergeben.

Die entwickelte, prototypische Modellbibliothek für das Assertion-Checking wurde im Rahmen einer gemeinsamen Veröffentlichung genutzt um Eigenschaften in einer Mixed Signal Umgebung prüfen zu können. Durch eine Weiterentwicklung wäre deren Verwendung in System-Level-Simulationsumgebungen, wie MLDesigner oder auch in SystemC denkbar. In diesem Zusammenhang sind Projekte mit Toolherstellern geplant.

Das entwickelte Modell zur Beschreibung und Analyse von zeitbeschränkten temporalen Eigenschaften von Diskrete Event Modellen unter strukturellen Randbedingungen erscheint vielversprechend, erfordert aber noch weitere Forschungsaktivitäten, vor allem in Bezug auf die vergleichende Bewertung mit klassischen Ansätzen. In diesem Zusammenhang ist ein Projekt in Kooperation mit anderen Forschergruppen in Planung.

Die Ergebnisse und Erfahrungen des Forschungsprojektes werden in Lehrveranstaltungen einfließen. Weiterhin ist als Ergebnis des Projektes der Abschluss eines Promotionsvorhabens geplant.

2.3 Fortschritt bei anderen Stellen

Beim Projektpartner in Darmstadt wurde eine Methode zur kompositionalen Analyse von Eigenschaften auf der Basis einer kompositionalen Eigenschaftssynthese entwickelt. Dieser Ansatz nutzt ein synchrones Zeitmodell und wird auf Gatterebene verwendet. Interessant wäre eine Adaption dieser Methode auf Ereignismodelle mit kontinuierlichem Zeitmodell. In diesem Zusammenhang wäre gegebenenfalls die Synthese komplexer Module auf der Basis von Eigenschaftsbeschreibungen möglich.

Im Bereich der Modellierung mittels SystemC wurden Fortschritte gemacht. In diesem Zusammenhang ist die Unterstützung von Mixed Signal Modellen zu Erwähnen, aber auch Fortschritte im Bereich der Standardisierung und Werkzeugunterstützung. Interessant erscheint in diesem Zusammenhang eine Adaption der entwickelten Methoden und Konzepte auf SystemC Modelle für abstrakte Systemmodelle.

Im Bezug auf die Definition einer systematischen Methodik zur Erstellung von Systemmodellen zur Validierung in frühen Phasen der Systementwicklung wurden Fortschritte gemacht. [BaHS07, Fi07, Sa07]. Durch eine Standardisierung solcher Modelle ergibt sich die Möglichkeit die Nutzung von Verifikationsmethoden innerhalb eines Verifikationsflows zu definieren und für Komplexe Systeme in einen konkreten Designflow zu integrieren. Hierbei sollten auch die Ergebnisse der Universität Tübingen im Bezug auf die Definition von Eigenschaften durch UML Diagramme berücksichtigt werden.

2.4 Erfolgte oder geplante Veröffentlichungen

[BaPS07]
Tommy Baumann, Alexander Pacholik, Horst Salzwedel. Performance Exploration with MLDesigner using Standardized Communication Interfaces. University Booth at DATE '07, 16.-20. April 2007, Acropolis, Nice, France.

[PaFSV05]
Alexander Pacholik, Wolfgang Fengler, Horst Salzwedel, Oleg Vinogradov.: Real Time Constraints in System Level Specifications Improving the Verification Flow of Complex Systems. In: Proceedings of Net.ObjectDays 2005, Erfurt 19.-22. 9. 2005, ISBN 3-9808628-4-4, S. 283-294.

[FePV06]
W. Fengler, A. Pacholik, O. Vinogradov. Development of language of time-constraints for the design of reactive systems. Naučnaja sessija. MIFI Moskovskij Inženerno-Fizičeskij Institut, 23.-27.01.2006, Moskva.

[PaFe07]
Alexander Pacholik, Wolfgang Fengler. A System Model for Formal VerificationVerification of TLM based Transaction Properties. In Communications and Networking Symposium, Spring Sim'07, 23.03.2007-27.03.2007 Norfolk.

[MuPF07]
M. Müller, A. Pacholik, W. Fengler. Tool Support for Formal System Verification. 52. IWK – Internationales Wissenschaftliches Kolloquium 10 - 13 September 2007, to appear.

[JeLP07]
Alexander Jesser, Stefan Laemmermann, Alexander Pacholik, Roland Weiss, Juergen Ruf, Wolfgang Fengler, Lars Hedrich, Thomas Kropf, Wolfgang Rosenstiel. Analog Simulation Meets Digital Verification- A Formal Assertion Approach for Mixed-Signal Verification. The 14th Workshop on Synthesis And System Integration of Mixed Information technologies, 15.-16-10.2007, Hokkaido, Japan, to appear.

3 Literaturverzeichnis:

[BaPS07]
Tommy Baumann, Alexander Pacholik, Horst Salzwedel. Performance Exploration with MLDesigner using Standardized Communication Interfaces. University Booth at DATE '07, 16.-20. April 2007, Acropolis, Nice, France.

[BaHS07]
T. Baumann, M. Hauguth, H. Salzwedel. Overcoming the Gap between Design at Electronic System Level (ESL) and Implementation for Networked Electronics. 2007 Western MultiConference on Modeling & Simulation, WMC '07, 14.-18. January 2007, San Diego, California.

[BeDL04]
Gerd Behrmann, Alexandre David, Kim G. Larsen: „A Tutorial on Uppaal", verfügbar unter http://www.it.uu.se/research/group/darts/papers/texts/new-tutorial.pdf

[FeDDL03]
Wolfgang Fengler, Bernd Däne, Vesselka Duridanova, Thomas Licht. Design Methodology for an Embedded System for High-Performance Computing. 27th IFAC/IFIP/IEEE Workshop on Real-Time Programming, ISBN 0-08-044203-X, pp. 99-104, Lagow, Poland, May 14-17, 2003.

[FeHFS02]
O. Fengler, T. Hummel, W. Fengler. Modellierung kooperierender Prozesse mit gefärbten Sequenzdiagrammen. in: J. Ruf (Hrsg.): 5. GI/ITG/GMM-Workshop: Methoden und Beschreibungssprachen zur Modellierung und Verifikation von Schaltungen und Systemen. Shaker-Verlag Aachen (ISBN 3-8265-9859-8), S. 199-208, 2002.

[FeKDHS01]
W. Fengler, E. Kallenbach, V. Duridanova, T. Hummel, E. Saffert. Zwischenbericht zum Forschungsvorhaben DFG: FE373/13-1: Entwurf eingebetteter paralleler Steuerungssystemefür integrierte multi-axiale Antriebssysteme. S. 1-20, TU Ilmenau, 2001.

[Fi07]
Nils Fischer. Entwurf einer Plug-and-Play Entwicklungsumgebung zur Optimierung vernetzter Avionik-Systemarchitekturen. Diplomarbeit TU Ilmenau 2007

[FrZ03]
A. Franck, V. Zerbe. A Combined Continuous-Time/Discrete-Event Computation Model for Heterogeneous Simulation Systems. APPT'03: International Workshop on Advanced Parallel Processing Technologies, Xiamen (China), 17.-19. 9. 2003.

[JeLP+07]
Alexander Jesser, Stefan Laemmermann, Alexander Pacholik, Roland Weiss, Juergen Ruf, Wolfgang Fengler, Lars Hedrich, Thomas Kropf, Wolfgang Rosenstiel. Analog Simulation Meets Digital Verification- A Formal Assertion Approach for Mixed-Signal Verification. The 14th Workshop on Synthesis And System Integration of Mixed Information technologies, 15.-16-10.2007, Hokkaido, Japan, to appear.

[Kl06]
Johannes Klöckner. Transformation einer UML-Systembeschreibung in eine Struktur zur Verifikation von Zeiteigenschaften. Diplomarbeit TU Ilmenau 2006.

[M1-A1.3-1]
Alexander Pacholik, Wolfgang Fengler, Horst Salzwedel. Meilenstein M1-A1.3-1 zum Clusterforschungsbericht FEST: „Konzeption der Constraint Language". Technischer Bericht, TU Ilmenau 2005.

[M1-A2.2-1]
Alexander Pacholik, Wolfgang Fengler, Horst Salzwedel. Meilenstein M1-A2.2-1 zum Clusterforschungsbericht FEST: „Spezifikation des Dekompositionsverfahrens unter Verwendung der Constraint Language für die Zeitverifikation". Technischer Bericht, TU Ilmenau 2006.

[M1-A2.3-1]
Alexander Pacholik, Wolfgang Fengler, Horst Salzwedel. Meilenstein M1-A2.3-1 zum Clusterforschungsbericht FEST: „Spezifikation der Algorithmen und Verfahren mit Hilfe der Modellierungen aus AP1 und Untersuchung der Transformationenverfahren an Beispielmodellen". Technischer Bericht, TU Ilmenau 2006.

[M3-A1.3-1]
Alexander Pacholik, Wolfgang Fengler, Horst Salzwedel. Meilenstein M3-A2.2-1 zum Clusterforschungsbericht FEST: „Validierung des implementierten Konzepts zur Behandlung von Zeiteigenschaften und deren Restriktionen". Technischer Bericht, TU Ilmenau 2007.

[M3-A2.2-1]
Alexander Pacholik, Wolfgang Fengler, Horst Salzwedel. Meilenstein M3-A2.2-1 zum Cluster-

forschungsbericht FEST: „Implementierung und Validierung des Dekompositionsverfahrens für die Zeitverifikation“. Technischer Bericht, TU Ilmenau 2007.

[M3-A2.3-1]
Alexander Pacholik, Wolfgang Fengler, Horst Salzwedel. Meilenstein M3-A2.3-1 zum Clusterforschungsbericht FEST: „Validierung der implementierten Verfahren und Algorithmen“. Technischer Bericht, TU Ilmenau 2007.

[Mu06]
Marcus Müller. Transformation von modellbasierten Systembeschreibungen: Entwurf und Implementation eines Java-Frameworks. Diplomarbeit TU Ilmenau 2006.

[MuPF07]
M. Müller, A. Pacholik, W. Fengler. Tool Support for Formal System Verification. 52. IWK – Internationales Wissenschaftliches Kolloquium 10 - 13 September 2007, to appear.

[PaFe07]
Alexander Pacholik, Wolfgang Fengler. A System Model for Formal VerificationVerification of TLM based Transaction Properties. In Communications and Networking Symposium, Spring Sim’07, 23.03.2007-27.03.2007 Norfolk

[PaFSV05]
Alexander Pacholik, Wolfgang Fengler, Horst Salzwedel, Oleg Vinogradov.: Real Time Constraints in System Level Specifications Improving the Verification Flow of Complex Systems. In: Proceedings of Net.ObjectDays 2005, Erfurt 19.-22. 9. 2005, ISBN 3-9808628-4-4, S. 283-294.

[Ruf99]
J. Ruf. Techniken zur Modellierung und Verifikation von Echtzeitsystemen. Dissertation zur Erlangung des akademischen Grades eines Doktors des Ingenieurwissenschaften. Fakultät für Informatik, Universität Karlsruhe, 1999.

[RuHQJZ05]
C. Rupp, J. Hahn, S. Queins, B. Zengler. UML 2 glasklar - Praxiswissen für die UML-Modellierung und –Zertifizierung. Carl Hanser Verlag, 2.Auflage 2005, ISBN 3-446-22952-3

[Sa07]
Horst Salzwedel. Complex Systems Design Automation in the Presence of Bounded and Statistical Uncertainties. 52. IWK – Internationales Wissenschaftliches Kolloquium 10 - 13 September 2007, to appear

[UML04]
Object Management Group. UML 2.0 Superstructure. OMG Final Adopted Specification. Ptc/04-10-02., 2004, http://www.omg.org

[Ze03]
V. Zerbe: Mission Level Design of Complex Autonomous Systems. Invited paper at XLVII ETRAN Conference Herceg Novi, Montenegro, June 8-13, 2003.