

Simulationsbasierte Evaluierung von Metriken in Echtzeit-Ethernet basierten Fahrzeugnetzen

Till Steinbach, Franz Korf, Thomas C. Schmidt
Department of Computer Science
Hamburg University of Applied Sciences, Germany
{till.steinbach, korf, schmidt}@informatik.haw-hamburg.de

Zusammenfassung—Durch die Zunahme von elektronischen Systemen, insbesondere im Fahrerassistenz- und Komfortbereich, kommen die etablierten Automotive-Kommunikationstechnologien an die Grenze ihrer Leistungsfähigkeit. Ein neuer Ansatz für die Kommunikation zwischen Steuergeräten im Automobil ist der Einsatz von Ethernet. Echtzeit-Erweiterungen haben den Einsatzbereich von standard switched Ethernet auf zeitkritische Anwendungen ausgedehnt.

Diese Arbeit leistet einen Beitrag zur Bewertung dieser neuen Konzepte, indem eine simulationsbasierte Evaluierungsstrategie für die Ermittlung von Kenngrößen (Metriken) Echtzeit-Ethernet-basierter Vermittlungsinfrastrukturen entwickelt wird. Eine gründliche Analyse des Simulationsmodells, welche die Simulationsergebnisse mit Berechnungen eines mathematischen Modells und Messungen auf echter Hardware vergleicht, belegt die Gültigkeit der Implementierung und der mit ihr ermittelten Kenngrößen.

I. EINFÜHRUNG & MOTIVATION

In einem modernen Kraftfahrzeug wird der Fahrer von einer Vielzahl an Fahrerassistenzsystemen unterstützt. Mit bis zu 70 Steuergeräten in Oberklasse-Modellen sind heutige Fahrzeuge komplexe verteilte Echtzeit-Systeme, deren Kommunikationsflüsse zwischen den verteilten Modulen einen signifikanten Einfluss auf ihre Zuverlässigkeit und die Sicherheit und den Komfort der Insassen haben.

Durch die Zunahme von elektronischen Systemen im Automobil sind auch die Anforderungen an leistungsfähige Kommunikations-Verbindungen zwischen den einzelnen Steuergeräten gestiegen. In Spitzenmodellen werden ca. 2500 Signale zwischen den Steuergeräten übertragen. Die Steigerung des Bedarfs an leistungsfähigen Kommunikations-Verbindungen wird sich auch in den kommenden Jahren, insbesondere mit kamerabasierten Assistenzsystemen, unvermindert fortsetzen. Es sind neue Entwürfe gefragt, welche diesen zunehmenden Anforderungen gerecht werden können.

Ein weiteres Argument für die Weiterentwicklung von Fahrzeugnetzen ist der heterogene Aufbau der Datenübertragung aktueller Serienfahrzeuge. Je nach Anwendung und zu übertragenden Daten kommen spezielle Bussysteme zum Einsatz welche für ihre jeweilige Anwendung besonders geeignet sind. Um jedoch eine Kommunikation zwischen den verschiedenen Technologien zu ermöglichen, müssen sogenannte Gateways eingesetzt werden, welche die Nachrichten zwischen den verschiedenen Systemen übersetzen. Die damit entstehende heterogene Struktur macht das Fahrzeugnetzwerk kompliziert und dadurch vor allem in der Entwicklung teuer. Erstrebenswert

ist eine neue Technologie, die sich weitgehend durchgängig im gesamten Fahrzeug einsetzen lässt.

Ein neuer Architekturansatz für Kommunikation im automotive Bereich basiert auf einem sogenannten *Backbone-Netzwerk*, einer *intelligenten Vermittlungsinfrastruktur*, welche die verteilten Steuergeräte des Fahrzeugs verbindet. Dieser zentrale Fahrzeug-Backbone muss nicht nur starker Last gewachsen sein, sondern muss dabei auch stets eine deterministische Nachrichtenübertragung garantieren.

Die Nutzung der *Ethernet-Technologie* in Fahrzeug-Netzwerken ist ein solcher Ansatz. Durch den Einsatz in Computernetzen hat Ethernet bereits bewiesen, ein flexibles, hoch skalierbares Übertragungsprotokoll zu sein. Standardmäßig kann Ethernet jedoch nicht die für eine Echtzeit-Kommunikation im Fahrzeug erforderliche Vorhersagbarkeit und Zuverlässigkeit in der Paketweiterleitung und dem Medienzugriff garantieren.

Echtzeit-Erweiterungen von Standard-Ethernet versprechen zuverlässige, vorhersagbare Übertragung und haben im Bereich der Prozessautomatisierung bereits umfangreichen Einsatz in Produktivsystemen gefunden. Time-Triggered Ethernet (TTEthernet) [1] ist eine Echtzeit-Ethernet-Erweiterung, welche darauf abzielt, die speziellen Bedürfnisse der automotive-Kommunikation zu befriedigen. In TTEthernet werden alle sendenden Teilnehmer über ein ausfallgesichertes Protokoll synchronisiert. Anschließend operieren alle Teilnehmer nach einem vordefinierten gemeinsamen Zeitplan, der das simultane Senden zweier Echtzeit-Nachrichten über dieselbe physikalische Verbindung ausschließt. Damit erweitert TTEthernet das Standard-Ethernet-Protokoll um die Fähigkeit, zeitkritische Daten deterministisch zu transportieren.

TTEthernet ist, wie Vorarbeiten zeigen [2], eine erfolgversprechende Technologie für eine zukünftige intelligente Vermittlungsinfrastruktur in Fahrzeugen. Es greift Konzepte aus verschiedenen in Flug- und Fahrzeugen bewährten Protokollen wie *FlexRay* [3], *TTP* [4] oder dem *AFDX* [5] auf, und ist daher gut für die Entwicklung von automotive-Anwendungen geeignet. Darüber hinaus bietet es den Transport von Nachrichten mit unterschiedlichen zeitlichen Anforderungen über eine einzige geteilte physikalische Infrastruktur. Eine Eigenschaft, die wegen der Vielschichtigkeit der Anwendungen in heutigen und zukünftigen Fahrzeugen besonders attraktiv ist.

Um den Einsatz dieser neuen Ethernet-basierten Vermittlungsinfrastruktur im Fahrzeug zu evaluieren, ist es wichtig, ihre charakteristischen Netzwerk-Metriken zu identifizieren.

Diese Arbeit leistet mit einer simulationsbasierten Evaluierungsstrategie für die Ermittlung dieser Kenngrößen einen Beitrag zur Bewertung.

Die speziellen Herausforderungen für einen Evaluierungsprozess liegen im Entwicklungszyklus eines Fahrzeuges. Während der Entwicklung sind diverse Dienstleister und Zulieferer beteiligt. Verschiedene Teile und Baugruppen werden in Kooperation mit mehreren Partnern entwickelt. Der Fahrzeughersteller (OEM) ist in diesem Zusammenwirken für das Design und die Konfiguration der Infrastruktur zur Fahrzeugkommunikation verantwortlich. Eigenschaften des Zeitverhaltens und der Auslastung sind insbesondere wegen der dezentralen Entwicklung schon zu einem sehr frühen Stadium wichtig, noch bevor echte Komponenten zur Verfügung stehen. Spätere Erweiterungen und Build-to-Order-Varianten (BTO) führen dabei zu einer extrem großen Anzahl möglicher Konfigurationen, welche bei der Analyse berücksichtigt werden müssen. Daher eignet sich eine simulationsbasierte Evaluierungsstrategie besonders gut für die Bewertung. Mit ihr kann der Effekt einer großen Anzahl von Konfigurationsparametern in umfassender Weise bereits vor der Realisierung überprüft und aus den Simulationsergebnissen direkt Anforderungen an Dienstleister und Zulieferer abgeleitet werden.

Der in diesem Beitrag vorgestellte Evaluierungsprozess ist in drei Phasen unterteilt. In der *Einrichtungsphase* wird zunächst werkzeuggestützt aus dem zu untersuchenden Netzwerkmodell ein Simulationsmodell der Vermittlungsinfrastruktur, der Datenquellen und -senken, sowie der Nachrichten erstellt. Aus den Anforderungen der zu Grunde liegenden Anwendungen können anschließend die Obergrenzen für die verschiedenen zu untersuchenden Metriken der Vermittlungsinfrastruktur gewonnen werden. In der *Simulationsphase* wird das abgeleitete Modell in der Simulationsumgebung vermessen und die Ergebnisse dokumentiert. In der *Auswertungsphase* werden anschließend die Simulationsergebnisse mit den zu Beginn definierten Anforderungen verglichen und bewertet. Als Ergebnis dieses Prozesses erhält der Anwender automatisiert einen Analysebericht über die vermessenen Metriken und ihre Konformität mit den Anforderungen. Durch eine präzise Abbildung der zeitlichen Eigenschaften im Simulationsmodell können die Ergebnisse direkt auf ein mit demselben Netzwerkmodell konfiguriertes reales Netzwerk übertragen werden. Dabei ist zu beachten das die Aussagekraft der Ergebnisse stets durch eine sorgfältige Evaluierung des Simulationsmodells sichergestellt sein muss.

Der Beitrag gliedert sich wie folgt: Im Abschnitt II wird der Hintergrund sowie die vergleichbaren und verwandten Arbeiten vorgestellt. Abschnitt III gibt eine Einführung in die echtzeitrelevanten Metriken und ihre Bedeutung für die automotive Kommunikation. Das Konzept der simulationsbasierten Evaluierung und die Architektur der Plattform wird in Abschnitt IV gezeigt, Abschnitt V zeigt ausgewählte Details aus der Implementierung. Abschließend werden in Abschnitt VI Ergebnisse der Evaluierung vorgestellt. Abschnitt VII gibt ein zusammenfassendes Fazit und einen Ausblick auf weitere Ziele.

II. HINTERGRUND & VERWANDTE ARBEITEN

Dieser Abschnitt gibt einen Überblick über die Merkmale von verteilten Anwendungen im Automobil und ihre Anforderungen sowie heutige Bussysteme und die in der Evaluierung verwendeten Techniken und Technologien. Abschließend werden Vorarbeiten und die mit dieser Arbeit verwandten und vergleichbaren Veröffentlichungen vorgestellt.

A. Merkmale des verteilten Echtzeit-Systems Automobil

Das Fahrzeugnetzwerk ist ein wohldefiniertes, geschlossenes System. In der Regel werden Komponenten nicht spontan in das System eingefügt oder entfernt. Damit kann der Determinismus und das Zeitverhalten des Kommunikationssystems gut im voraus berechnet werden. Das System kann statisch konfiguriert werden und muss sich nicht auf sich ändernde Topologien oder variierende Teilnehmerzahlen anpassen, wie es etwa in lokalen (LAN) oder ausgebreiteten Computernetzwerken (WAN) erforderlich ist.

Echtzeitanwendungen im Automobil haben hohe Anforderungen an das Zeitverhalten der Datenübertragung zwischen den Steuergeräten. So muss beispielsweise bei einem Aufprall die Aufnahme der Sensorwerte, die Datenübertragung und die Aktivierung des Airbags abgeschlossen sein, bevor die Relativbewegung der Insassen beginnt. Auch die Güte von Regelungssystemen wie beispielsweise der Antriebsschlupfregelung (ASR) hängt vom Zeitverhalten der Datenübertragung, insbesondere der Nachrichtenlaufzeit (Latenz) und der Varianz der Nachrichtenlaufzeit (Jitter) ab, denn sie hat wesentlichen Einfluss auf die Totzeit der verteilten Regelung. Die Totzeit ist ein Begriff aus der Regelungstechnik, der die Zeitspanne zwischen einer Änderung am System-Eingang und der entsprechenden Antwort am System-Ausgang bezeichnet. Je länger die Totzeit ist, desto schlechter wird das Regelverhalten. Dies kann soweit führen, dass der Regler unbrauchbar wird.

Die Automobilindustrie definiert einen umfangreichen Anforderungskatalog an Kommunikationssysteme für Echtzeitanwendungen im Fahrzeug [6]. Im Zentrum stehen:

- Übertragung von Daten mit unterschiedlichen Zeitanforderungen
- Synchroner und asynchroner Übertragung
- Niedrige Latenz und niedriger Nachrichtenjitter
- Hohe Übertragungsbandbreite
- Redundanzkonzept
- Austauschbarkeit des Physical-Layers
- Einfache Anwendbarkeit
- Verwendbarkeit von Standardkomponenten

Darüber hinaus gibt es eine ganze Reihe von allgemeinen Anforderungen an Elektronik im Fahrzeug wie: Kosteneffizienz, schnelles Start-Up, einfache Wartbarkeit, gute Skalierbarkeit für weitere Extras, Eignung für kritische Umweltbedingungen (Temperatur, Vibrationen, Feuchtigkeit, elektromagnetische Strahlung), geringes Gewicht, geringer Platzbedarf und niedriger Energieverbrauch.

Zur Zeit ist das Datenübertragungsnetz in Serienfahrzeugen heterogen aufgebaut. Je nach Anwendung und zu übertragenden Daten kommen spezielle Bussysteme zum Einsatz [7].

Tabelle I
BUSSYSTEME IM FAHRZEUG UND IHRE TYPISCHE VERWENDUNG

System	Typisches Anwendungsfeld
CAN	Vernetzung von Steuergeräten, event-triggered
LIN	Vernetzung von Aktoren und Sensoren
FlexRay	Vernetzung von Steuergeräten (höhere Bandbreite), event- und time-triggered
MOST	Vernetzung von Audio- und Video-, Sprach- und Datenanwendungen

Gängige Technologien sind der *LIN-Bus* [8], der *CAN-Bus* [9], *FlexRay* [3] und der *MOST-Bus* [10]. Tabelle I zeigt wichtige, heute eingesetzte Bussysteme und ihre Verwendung.

B. Die Echtzeit-Ethernet-Technologie

Ethernet ist eine bewährte Technologie, die mit dem Internet und der damit verbundenen Vernetzung von Rechnern ihren Durchbruch erlebte. Ethernet bietet eine hohe Übertragungsbandbreite. Während sich Systeme mit bis zu 10 Gbit/s bereits am Markt etabliert haben, werden in Laborbedingungen bereits 100 Gbit/s, auch über weite Strecken, erreicht. Durch die gute Verbreitung sind die Preise für Ethernet-Komponenten stark gesunken. Heute haben sehr viele Microcontroller bereits Hardware für die Anbindung an Ethernet integriert und es werden heute mehr Ethernet-Einheiten ausgeliefert als Einheiten der klassischen automotive-Technologien. Ein großer Vorteil bei der Verwendung von Ethernet ist die Entkopplung der Kommunikation vom Übertragungsmedium. Der Ethernet-Standard definiert neben der Nutzung von Kupferkabel in verschiedenen Konfigurationen beispielsweise auch die Übertragung über Lichtwellenleiter. Durch die weite Verbreitung von Ethernet gibt es bereits eine große Palette an Entwicklungs- und Diagnosewerkzeugen. Die Zahl an fachkundigen Entwicklern ist beim Ethernet-Protokoll weit größer als bei speziellen automotive-Lösungen wie LIN oder MOST.

Ursprünglich ist Ethernet nicht echtzeitfähig, da weder Teilnehmer noch Vermittlungsknoten einer deterministischen Nachrichtenvermittlung unterliegen. In Switched-Ethernet kann es dabei zu Datenstaus an den Switchen (Knotenpunkten) kommen, welche die Übertragung der Nachrichten verzögern. Verschiedene Protokollerweiterungen ermöglichen, Echtzeitfähigkeit auch auf Ethernet-Netzwerken herzustellen.

Grundsätzlich lassen sich die am Markt befindlichen Technologien in die drei Hauptfelder *time-triggered* Systeme, *token-basierte* Systeme und *bandbreitenbasierte* Systeme einordnen, da sie jeweils die wichtigsten Eigenschaften vereinen.

Time-triggered (zeitgesteuerte) Protokolle sind ein verbreiteter Ansatz, um Echtzeitfähigkeit auf Kommunikationssystemen herzustellen. Dabei synchronisieren sie sich über eine globale Zeit. Jeder Teilnehmer hat eine lokale Uhr, welche möglichst genau mit den Uhren der anderen Teilnehmer synchronisiert werden muss. Zugewiesene Zeitschlitze definieren für jeden Teilnehmer, wann er das Medium zum Senden nutzen darf. Damit ist sichergestellt, dass immer nur ein Teilnehmer zur Zeit auf das Medium zugreift. Diese Technik wird auch als *koordiniertes Time-Devision-Multiple-Access (TDMA)* be-

zeichnet. Diverse Echtzeit-Ethernet-Produkte, darunter auch TTEthernet basieren auf dem time-triggered Konzept.

Die Grundlage von Token-basierten Systemen ist der wechselseitige Ausschluss mit Hilfe einer zwischen allen Teilnehmern geteilten Ressource, dem *Token*. Das Token wandert von Teilnehmer zu Teilnehmer, wobei nur derjenige senden darf, der das Token hält. Damit ist sichergestellt, dass das Medium stets frei ist, wenn Daten gesendet werden. Das System bleibt über das Token synchronisiert. In der Prozessautomatisierung wird das Token-basierte *EtherCAT-Protokoll* [11] eingesetzt.

Bandbreitenbasierte Systeme arbeiten ohne eine Form der Synchronisierung. Es werden virtuelle Bandbreitenkontingente konfiguriert. Jeder Teilnehmer sendet mit einem solchen Kontingent. In diesen Systemen überwachen die Switches, dass die Teilnehmer ihre Kontingente nicht überschreiten. Da keine Synchronisierung stattfindet, ist in Systemen mit Bandbreitenkontingenten die berechnete worst-case Verzögerungszeit und damit auch der Jitter weit höher als in der Realität zu beobachten. Dies hängt damit zusammen, dass im worst-case alle Teilnehmer zur gleichen Zeit ihre Nachrichten versenden. Das AFDX-Protokoll [5] ist ein typisches bandbreitenbasiertes Protokoll.

C. Time-triggered Ethernet

TTEthernet wird von der Firma TTTech [12] entwickelt. Die TTEthernet-Spezifikation [1] wird momentan von der Society of Automotive Engineers (SAE) standardisiert [13]. TTEthernet basiert auf standard Switched-Ethernet. Die Topologie wird aus Switchen zusammengesetzt, welche die Nachrichten weiterleiten. Redundanz kann durch redundante Verbindungen zwischen den Komponenten erreicht werden.

Die Datenübertragung in TTEthernet ist synchron und zyklisch aufgebaut. Für die Echtzeit-Kommunikation werden jedem Teilnehmer offline konfigurierte Zeitslots zugewiesen. Dieses Netzwerk-Zugriffsverfahren wird als koordiniertes TDMA bezeichnet und erlaubt vorhersagbare Übertragungszeiten in definierten Grenzen ohne Warteschlangenbildung in den Switchen und damit Kommunikation mit niedriger Latenz und niedrigem Jitter. Damit jeder Teilnehmer zur richtigen Zeit seinen Slot in Anspruch nehmen kann, haben alle Teilnehmer eine lokale Uhr und einen Zeitplan (Schedule) für die Übertragung. Da eine globale Zeit über alle Teilnehmer erforderlich ist, definiert die TTEthernet-Spezifikation ein fehlergesichertes Synchronisierungsprotokoll.

Neben den *time-triggered* Nachrichten definiert TTEthernet noch zwei weitere Nachrichtenarten. *Rate-constrained* Nachrichten sind für die Übertragung von Daten mit weniger harten Echtzeitanforderungen gedacht und basieren auf dem AFDX-Protokoll [5]. *Best-effort* Nachrichten basieren auf Standard-Ethernet und haben in der Übertragung die niedrigste Priorität. Sie werden verwendet, um Daten im Hintergrund zu übertragen. Eine Besonderheit von TTEthernet-Netzwerken ist die Fähigkeit, Teilnehmer zu integrieren, die das time-triggered Protokoll nicht unterstützen. Diese Teilnehmer bleiben unsynchronisiert und kommunizieren ausschließlich über best-effort Nachrichten.

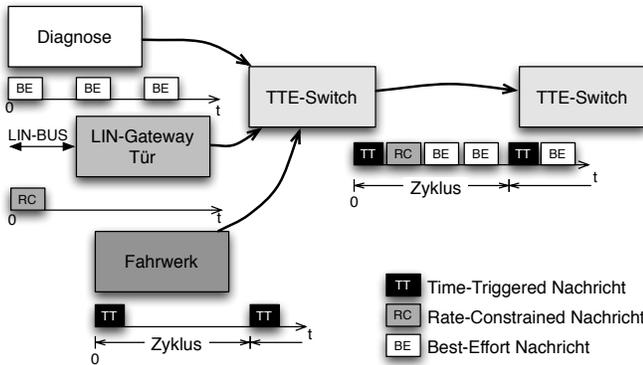


Abbildung 1. TTEthernet-Beispielanwendung — Fahrzeugkommunikation mit drei unterschiedlichen Trafficklassen

Abbildung 1 visualisiert eine Beispiel-Anwendung aus dem automotive-Umfeld, in der die drei Nachrichtenarten auf demselben physikalischen Link koexistieren. Während time-triggered Nachrichten — beispielsweise für die Übertragung von Fahrwerksinformationen — unter strikter Einhaltung des Übertragungszyklus und mit der höchsten Priorität weitergeleitet werden, werden rate-constrained Pakete in freien Zeitschlitzern übertragen. Die danach noch freie Bandbreite wird für die Übertragung von best-effort Datenverkehr verwendet.

D. Grundlagen der Simulation

Die Event-basierte Simulation hat sich de facto als Standard für die Simulation von Netzwerken etabliert. Dies hängt damit zusammen, dass die Paketweiterleitung sehr gut in ein diskretes Event-basiertes Modell überführt werden kann. Nimmt man den Paketfluss zwischen zwei Teilnehmern als Beispiel, sind nur einige wenige Zeitpunkte zwischen Versand und Empfang der Pakete, wie das Eintreffen in den Instanzen zwischen den Teilnehmern, von Interesse. Alle weiteren Zeitpunkte dazwischen können in der Simulation vernachlässigt werden. Im Gegensatz zu physikalischen Modellen, in denen ein bedeutender Rückfluss des Zustandes in das System stattfindet und eine große zeitliche Auflösung zu Ungenauigkeiten führt, hat das System in der Event-basierten Simulation bedeutend weniger Zustände.

Für den Evaluierungsprozess wurde *OMNeT++* [14], ein Simulationsframework für Simulationen auf Basis von diskreten Events, ausgewählt. *OMNeT++* verfolgt einen Framework-Ansatz. Es bietet nicht direkt Komponenten zur Simulation an, sondern unterstützt den Anwender mit Werkzeugen, um Simulationskomponenten zu erstellen. Frameworks — wie das *INET-Framework* — fügen *OMNeT++* Komponenten aus speziellen Anwendungsgebieten hinzu. Diese Frameworks werden unabhängig entwickelt und meist von Dritten herausgegeben.

Das *INET-Framework* [15] ist ein Framework für *OMNeT++*, das die Entwicklung von Simulationen im Bereich der Netzwerkprotokolle vereinfacht. Es besteht aus einer Reihe von Modulen, die die Funktionalität von verschiedenen Elementen — vom physikalischen Layer bis zur Applikationsschicht des OSI-Modells — zur Verfügung stellt. Das *INET-Framework* ist modular aufgebaut und setzt intensiv Vererbung ein. Daher ist es besonders gut für die Verwendung in der

Simulation von Echtzeit-Ethernet geeignet. Da TTEthernet auf Standard-Ethernet basiert, können große Teile übernommen und um die TTEthernet-Logik erweitert werden.

E. Vorarbeiten, Verwandte und vergleichbare Arbeiten

In Vorarbeiten wurde für einen Vergleich zwischen TTEthernet und FlexRay auf Basis analytischer Methoden bereits ein mathematisches Modell für die TTEthernet-Technologie erstellt [2]. Dieses Modell wurde in dieser Arbeit insbesondere für die Validierung der Simulationsergebnisse herangezogen.

Der Einsatz von neuen Vermittlungsinfrastrukturen in Kraftfahrzeugen wird durch verschiedene Institutionen untersucht. Insbesondere ist die BMW-Group zu erwähnen, welche im Jahre 2007 als technische Demonstration einen IP-basierten Fahrzeug-Prototyp vorstellte und den Einsatz von Ethernet für Anwendungen im Fahrzeug anstrebt. Insbesondere in Zusammenarbeit zwischen der Technischen Universität München und BMW (CAR@TUM) entstanden verschiedene Forschungsprojekte und Veröffentlichungen zu Grundlagenthemen wie dem Design von vereinheitlichten Kommunikationsinfrastrukturen [16], [17], der Abbildung von Kommunikationsstrukturen aktueller Fahrzeuge auf Ethernet-basierte Vermittlungsinfrastrukturen [18] oder der Koexistenz von Traffic-Klassen mit verschiedenen Eigenschaften in geschichteten Netzwerken [19]. Die Arbeiten basieren fast ausschließlich auf Standard-Ethernet und grenzen sich damit von dieser Arbeit ab.

In der Prozessautomatisierung sind die Evaluierungen von Echtzeit-Ethernet-Protokollen schon weit fortgeschritten. Die Dissertation von Jürgen Jasperneite [20] beschäftigt sich mit der Leistungsbewertung von Ethernet für die prozessnahe Echtzeitkommunikation.

Nach aktuellem Kenntnisstand sind keine vollständigen Implementierungen von time-triggered Echtzeit-Ethernet-Erweiterungen für *OMNeT++* öffentlich verfügbar. Für andere Simulationswerkzeuge wie den *OPNET-Modeler* gibt es Modelle für Protokolle aus der Anlagensteuerung [21], [22]. Da die Protokolle jedoch von TTEthernet abweichen, sind diese Modelle nicht als Vorlage für ein Simulationsmodell geeignet.

III. NETZWERK METRIKEN IN DER AUTOMOTIVE KOMMUNIKATION

Für die Bewertung von Kommunikationstechnologien und ihrer Konfiguration haben Metriken eine große Bedeutung. Sie können die Grundlage für Aussagen zum technologischen und wirtschaftlichen Erfolg von zukünftigen Vermittlungsinfrastrukturen in Fahrzeugen bilden. Insbesondere für einen objektiven, kritischen Blick auf Optimierungen und Lösungsvorschläge und die Bewertung von Protokoll-Verbesserungen sind Metriken wichtig. Auch für das Festlegen von Anforderungen sind Metriken bedeutsam. Werden Anforderungen in definierten Metriken formuliert, sind sie allgemein verständlich, mess- und verifizierbar. Damit eignen sich Metriken als gemeinsame Sprache für die in der Automobilbranche übliche verteilte Entwicklung.

Metriken ordnen Systemzuständen reale Zahlenwerte zu. Damit sind sie geeignet als Qualitätsangabe (Absolutbewertung) sowie für die Vergleichbarkeit verschiedener Systeme

(Relativbewertung). Die Ermittlung von Metriken im Rahmen der Leistungs- und Verlässlichkeitsbewertung kann sowohl durch empirische Untersuchung im Betrieb als auch am Modell vorgenommen werden. Oft sind empirische Untersuchungen aufwendiger als Versuche am Modell oder können gar nicht durchgeführt werden, weil das reale System noch nicht existiert. Insbesondere bei großen und komplexen Systemen kann daher eine empirische Untersuchung unwirtschaftlich sein. Zudem müssen empirische Studien, insbesondere für die Verlässlichkeitsbewertung, über lange Zeit durchgeführt werden, um statistisch verwertbare Aussagen zu erhalten. Eine Alternative zur Messung am realen System bietet daher die in dieser Arbeit durchgeführte Untersuchung am Modell.

A. Evaluierung von Metriken in der Simulation

Im Gegensatz zur mathematischen Berechnung in der analytischen Evaluierung basiert die Evaluierung in der Simulation auf Versuchen am Modell. Dabei wird das Verhalten des Untersuchungsobjektes und der Umgebung durch Simulation der Abläufe und Prozesse in einem Simulationsprogramm nachgestellt.

Ein bedeutender Vorteil der Simulation, verglichen mit analytischen Verfahren, ist der Detaillierungsgrad. Mit simulationsbasierten Ansätzen lässt sich ein System in nahezu jeder gewünschten Tiefe betrachten. Der Aufwand, der dafür in die Modellierung gesteckt werden muss, hängt vom zu erzielenden Detailgrad ab. Zudem ist die Komplexität der Modelle nicht durch die mathematische Berechenbarkeit beschränkt. Die Grenze der Komplexität einer Simulation liegt in erster Linie in der vom Benutzer tolerierten Laufzeit und dem im Simulationssystem verfügbaren Speicher.

Somit stellt vor allem die Rechenzeit einen bedeutenden Nachteil der Simulation dar. Sie liegt weit über der der analytischen Evaluierung. Mit einer sorgfältigen Auswahl der Modellierungs- und Simulationstechniken kann die Rechenzeit jedoch auf ein notwendiges Mindestmaß reduziert werden. Um die Präzision des Simulationsmodells zu verbessern, können Messungen am Simulationsobjekt zur Gewinnung realitätsnaher Einflussgrößen hinzugezogen werden. Dies gilt insbesondere für die Modellierung der Arbeitslast.

B. Auswahl von Metriken

Nicht alle Metriken sind geeignet, um in einem simulationsbasierten Evaluierungsprozess ermittelt zu werden. Für die Bewertung der Qualität von Kommunikationsdiensten sind — neben den großen Drei: *Reliability*, *Maintainability* und *Availability* — die wichtigsten Metriken die *Latenz*, die *Varianz der Latenz (Jitter)* und die nutzbare *Bandbreite*. Im folgenden werden die Metriken, die im Evaluierungsprozess dieser Arbeit berücksichtigt werden, vorgestellt.

1) *Zuverlässigkeit (Reliability)*, *Wartbarkeit (Maintainability)*, *Verfügbarkeit (Availability)*: Die Zuverlässigkeit ist ein statistischer Faktor, der die Wahrscheinlichkeit angibt, ob ein System zu einem gegebenen Zeitpunkt t funktioniert. Unter der Annahme, dass das System zum Zeitpunkt t_0 funktionsfähig war, und das System eine durchschnittliche Fehlerrate von

λ Ausfällen pro Stunde besitzt, kann die Zuverlässigkeit $R(t)$ nach [23] mit der Formel 1 berechnet werden:

$$R(t) = e^{-\lambda \cdot (t-t_0)} \quad (1)$$

Die Fehlerrate gibt die Klasse der Zuverlässigkeit an. Ab einer Fehlerrate niedriger 10^{-9} wird von *ultrahigh reliability* gesprochen. Für Komponenten werden die Fehlerraten meist in Parts per Million (ppm) angegeben. Da die Lebensdauer von Systemen im Automobil bei mehreren tausend Betriebsstunden über einen Zeitraum von über 10 Jahren liegt und für viele Anwendungen mehrere Elemente benötigt werden, werden im Automobilbereich für einige Komponenten Fehlerraten im einstelligen ppm Bereich gefordert [24]. Das bedeutet, dass über die komplette Lebensdauer bei einer Million ausgelieferten Teilen weniger als 10 einen Defekt aufweisen dürfen [23].

Mit Wartbarkeit wird üblicherweise die Zeit beschrieben, die benötigt wird, um das System aus dem Fehlerzustand in den betriebsbereiten Zustand zu überführen. Wenn eine sicherheitskritische Komponente im Fahrzeug ausfällt, ist üblicherweise das Fahrzeug nicht mehr betriebsbereit. Die Dauer der Reparatur ist insbesondere unter ökonomischen Gesichtspunkten von Interesse, für die simulationsbasierte Evaluierung jedoch eher nicht geeignet.

Die Verfügbarkeit ist die Relation zwischen der Ausfallzeit und der Betriebsdauer. Sie kann daher durch Verbesserung der Zuverlässigkeit oder Verkürzung der Wartung verbessert werden.

2) *Latenz*: Die Latenz oder Nachrichtenlaufzeit beschreibt die zeitliche Verzögerung zwischen dem Aussenden eines Paketes beim Sender und dem Eingang des Paketes beim Empfänger. Sie setzt sich aus verschiedenen Komponenten wie *Bearbeitungsverzögerungen*, *Warteschlangenverzögerungen*, *Sendeverzögerungen* und *Signallaufzeiten* zusammen.

Die Latenz ist eine wichtige Metrik für die zeitkritische Kommunikation. Insbesondere bei der Regelung von Prozessen spielt die Latenz eine wichtige Rolle. Sie hat direkten Einfluss auf die Totzeit und ist damit eine maßgebende Größe für die Güte einer verteilten Regelung. Insbesondere bei Prozessen im Automobil wie dem Antiblockiersystem (ABS), der Antriebsschlupfregelung (ASR) oder dem Elektronischen Stabilitätsprogramm (ESP) ist eine geringe Totzeit und damit auch eine geringe Latenz wichtig. Ein allgemeiner Richtwert für die maximale Totzeit in Regelprozessen kann nicht festgelegt werden. Er hängt vor allem von der Geschwindigkeit des zu regelnden Prozesses ab und ist daher stets in Relation zur Anwendung zu betrachten. Die Latenz kann von Sender zu Empfänger (*Ende-zu-Ende Latenz*) oder in beide Richtungen (*Roundtrip Latenz*) gemessen und angegeben werden. In zeitkritischen Systemen wird üblicherweise nur die Ende-zu-Ende Latenz angegeben, da Verbindungen unidirektional oder asymmetrisch konfiguriert sein können.

Neben der Nachrichtenlaufzeit können noch weitere Faktoren die Latenz bestimmen. Dazu gehört beispielsweise die *Error Detection Latency*, die Dauer der Fehlererkennung. Dies betrifft Systeme, die Fehler, beispielweise über Redundanz, kompensieren können, zur Fehlererkennung jedoch eine gewisse Zeit benötigen.

3) *Jitter*: Der Nachrichten-Jitter beschreibt üblicherweise die Varianz der Nachrichten-Laufzeit. Für Echtzeit-Systeme ist es wichtig, den maximalen Jitter, also die Differenz zwischen maximaler und minimaler Paketlaufzeit, zu kennen. Daher ist es in diesen Systemen gängig, den Jitter als die Differenz zwischen oberer und unterer Grenze der Nachrichtenlaufzeit zu definieren.

Ein übliches Verfahren für verteilte Regler ist die zyklische Übertragung von Informationen. Der Sender sendet einen Wert in regelmäßigen Abständen an die Empfänger. Der Nachrichten-Jitter führt dazu, dass der Zyklus beim Empfänger nicht mehr sauber eingehalten werden kann. In synchronisierten Systemen kann durch die Verzögerung der Verarbeitung von eingehenden Nachrichten ein leichter Jitter zuungunsten der Latenz ausgeglichen werden. Ein starker Jitter führt dabei jedoch zu instabilen Regelprozessen. Eine übliche obere Schranke für den Jitter in automotive Anwendungen ist 10% der Latenz.

4) *Bandbreite*: Die Bandbreite, oft auch als Kapazität oder Maximaldurchsatz bezeichnet, beschreibt die maximale Menge an Informationen, die pro Zeiteinheit übertragen werden können. Damit ist die Bandbreite ein Kriterium der Leistungsfähigkeit eines Übertragungsmediums. Die nutzbare Bandbreite sollte stets netto, ohne alle Protokollinformationen, angegeben werden. Da jede Technologie einen anderen Protokoll-Overhead besitzt, ist nur so ein gerechter Vergleich möglich. Bei Übertragungstechniken mit einer zyklischen Übertragung ist zudem die maximale Anzahl verschiedener Botschaften in einem Zyklus von Interesse. Sie hängt neben der Bandbreite des Mediums auch von der Länge der Nachrichten ab.

Weitere Metriken wie die Zykluslänge oder die Anzahl möglicher Botschaften lassen sich gut mit analytischen Methoden durch Analyse der Scheduling-Konfiguration ermitteln. Andere Metriken wie der Energieverbrauch, die Start-Up Geschwindigkeit, die Elektromagnetische Verträglichkeit (EMV), das Gewicht oder die Abmessungen erfordern reale Hardware für empirische Untersuchungen oder zumindest Modelle der Implementierungen.

IV. KONZEPT & ARCHITEKTUR

Aufgrund der Erfahrungen aus den Vorarbeiten mit der analytischen und der simulativen Evaluierung wird ein simulationsbasierter Evaluierungsprozess entwickelt. Dies liegt insbesondere am benötigten Detaillierungsgrad, der in einem analytischen Modell nur mit großen Aufwand erreicht werden könnte.

Abbildung 2 zeigt die Struktur des Evaluierungsprozesses, sowie die Ein- und Ausgangsgrößen. Die Simulation wird durch eine Netzwerkkonfiguration gespeist, die zuvor manuell oder mit Werkzeugunterstützung aus dem Fahrzeugmodell erstellt wurde. Zusätzlich definieren die Anwendungen dieses Modells die Anforderungen an die Vermittlungsinfrastruktur, die in der Evaluierung analysiert werden.

Der Evaluierungsprozess selbst ist in drei Phasen unterteilt. Zunächst wird die Simulation auf Basis der Netzwerkkonfigu-

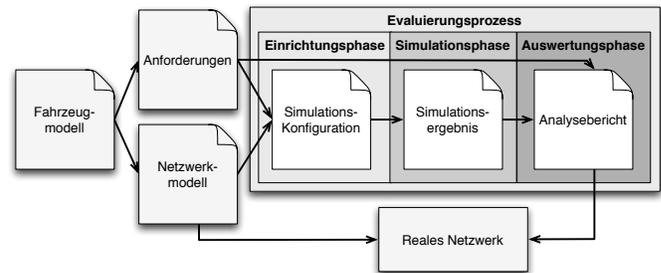


Abbildung 2. Überblick über die Phasen des Evaluierungsprozesses und seine Ein- und Ausgangsgrößen

ration und Anwendungsanforderungen konfiguriert. Anschließend wird in der Simulationsphase das abgeleitete Modell in der Simulationsumgebung vermessen und dokumentiert. Abschließend werden die Ergebnisse in der Auswertungsphase mit den zuvor erstellten Anforderungen verglichen und bewertet. In einer übersichtlichen Darstellung wird dem Nutzer gezeigt, an welcher Stelle das Modell nicht den Anforderungen genügt. Diese Information soll dem Nutzer helfen, in einem iterativen Vorgehen die Konfiguration anzupassen, bis sie alle Vorgaben erfüllt. Als Ergebnis erhält der Nutzer eine Übersicht über alle für ihn relevanten Metriken zusammen mit einer Netzwerkkonfiguration, welche direkt in vorhandene Hardware eingespielt werden kann.

Die Basis für den Evaluierungsprozess ist eine Netzwerkkonfiguration im Format von TTEthernet. Diese mit den TTEthernet-Werkzeugen erstellte XML-Datei enthält bereits die wichtigsten Informationen für die Simulation. Diese XML-Netzwerkkonfiguration wird vom Evaluierungs-Framework eingelesen und in das Format der Simulationsumgebung übersetzt. Als weitere Eingabe in den Evaluierungsprozess wird eine Beschreibung der Datenflüsse und ihrer Anforderungen definiert. Diese Beschreibung ist notwendig, um in der Simulation für die spätere Anwendung charakteristische Datenflüsse zu generieren. Nur so ist ein gutes Simulationsergebnis, welches die späteren realen Eigenschaften wiedergibt, erzielbar.

Im nächsten Schritt werden die globalen Parameter der Simulation, beispielsweise die Simulationsdauer oder die zeitliche Auflösung, festgelegt. Anschließend wird die Simulation — je nach Anforderung unterschiedlich — durchgeführt. Um eine Konfiguration zu überprüfen, das Verhalten des Netzwerkes zu untersuchen oder den Einfluss von Konfigurationsparametern auf die Weiterleitung zu beobachten, bietet sich dabei die grafische Simulationsoberfläche an. Hier kann detailliert der Nachrichtenfluss beobachtet werden. Fehler, die im Switch erkannt werden, werden direkt an den Nutzer gemeldet. Für Simulationen über lange Zeiträume bietet sich dagegen eine Ausführung ohne grafische Oberfläche oder auch eine Verteilung über mehrere Rechner an. Dadurch kann die Simulationsdauer signifikant reduziert werden.

Im letzten Schritt generiert das System einen Bericht über die simulierten Metriken. Als Eingabe dient hier, neben den Ergebnissen der Simulation, die Beschreibung der Datenflüsse und Anforderungen. Erfüllt die Konfiguration alle Anforderungen, kann der Nutzer auf Basis des Berichtes entscheiden, ob die evaluierte Konfiguration in dieser Form eingesetzt

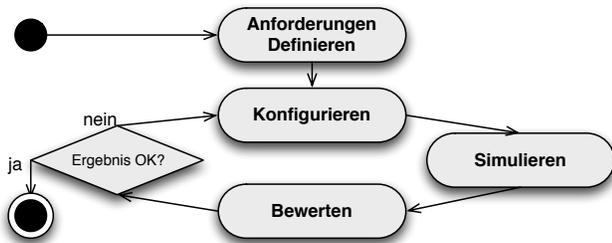


Abbildung 3. Zyklischer Arbeitsablauf im Evaluierungsprozess

werden soll. Werden Anforderungen nicht erfüllt, muss die Konfiguration angepasst und erneut evaluiert werden. Die Evaluierung unterstützt den Nutzer nun so lange in einem zyklischen Prozess, bis das gewünschte Ergebnis erzielt wird (siehe Abbildung 3).

Die Abbildungsgenauigkeit des Modells wird bewusst reduziert, um von den Implementierungsdetails zu abstrahieren. Als Vorlage für das Modell dient die Spezifikation und nicht die Hardware-Implementierung der TTEthernet-Komponenten. Bei einer Modellierung anhand der Spezifikation wird das Modell so umgesetzt, dass es den vorgegebenen Regeln entspricht. Damit abstrahiert es von der genauen Umsetzung in der Hardware.

Ein Vorteil dieser Abstraktionsebene ist, dass die Implementierung ohne die sonst bei Hardware-Entwürfen notwendigen Kompromisse umgesetzt werden kann. Der Nachteil ist eine mögliche Abweichung der Simulationsergebnisse von der späteren Implementierung. Solche Abweichungen lassen sich jedoch durch Korrekturfaktoren im Modell aufheben. Auf verschiedene Varianten vom Realsystem kann dann durch zielsystemspezifische Korrekturfaktoren eingegangen werden.

Damit der Evaluierungsprozess möglichst gut in die Werkzeugkette von TTEthernet integriert werden kann, wird das Netzwerk in einer sogenannten Network-Configuration beschrieben. Die Network-Configuration ist ein XML-Format zur Beschreibung von TTEthernet-Konfigurationen. Die Simulation in OMNeT++ selbst wird in Network Description-Files (NED) konfiguriert. NEDs enthalten die Struktur der Simulation. In einem automatisierten Prozess wird über eine Modelltransformation die NED aus der Network-Configuration generiert. Dadurch kann sichergestellt werden, dass die Simulation genau der gewünschten Hardwarekonfiguration entspricht. Darüber hinaus kann die Simulation in OMNeT++ durch die Werkzeuge für TTEthernet konfiguriert werden.

A. Elemente des Simulationsmodells

Die Simulation setzt sich aus drei übereinander gelagerten Schichten zusammen. Die Basis bildet die OMNeT++-Simulationsumgebung.

Basierend auf OMNeT++ wird das INET-Framework eingesetzt. Das INET-Framework ist besonders gut für die Verwendung in einer Simulation von Echtzeit-Ethernet geeignet, da es modular aufgebaut ist und intensiv Vererbung verwendet. Da TTEthernet auf Standard-Ethernet basiert, kann durch Vererbung die grundlegende Netzwerkfunktionalität vom Physikalischen- und Linklayer aus dem INET-Framework

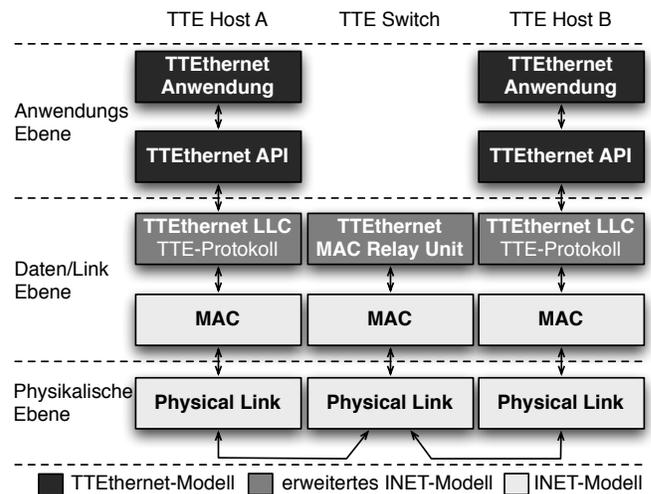


Abbildung 4. TTEthernet-Integration in das INET-Framework

abgeleitet werden.

Auf Basis des INET-Frameworks wird die TTEthernet-Simulationsschicht [25] implementiert. Sie erweitert das Modell um die Echtzeiteigenschaften des TTEthernet-Protokolls. Dabei muss besonders auf eine präzise Abbildung des Zeitverhaltens geachtet werden. Nur so lassen sich die Ergebnisse der Simulation zuverlässig auf ein echtes Szenario übertragen.

Abbildung 4 zeigt die wichtigsten Komponenten des TTEthernet-Modells und ihre Integration in die Schichten der Simulationsplattform. Dadurch, dass TTEthernet auf Standard-Ethernet basiert, müssen ausschließlich Komponenten des INET-Frameworks oberhalb der MAC-Ebene erweitert oder angepasst werden. Für das Modell des Ethernet-Hosts wird der Protokollstack in einer Erweiterung des Ethernet-LLC implementiert. Darüber liegen das TTEthernet-API und die Anwendungen in neuen Modulen. Für den TTEthernet-Switch kann die Protokoll-Implementierung direkt als Erweiterung der MACRelayUnit aus dem INET-Framework umgesetzt werden.

B. Automatisierte Auswertung der Simulationsergebnisse

Im Auswertungsprozess wird festgestellt, ob die Anforderungen erfüllt sind. Dafür werden die aufgezeichneten Werte gegen die definierten Bedingungen der Anforderungen überprüft. Das Zielformat des Analyseberichtes ist das PDF, da es ein verbreitetes Format ist, das sich gut zur Dokumentation eignet. Für die Ausgabe im PDF gibt es bereits diverse C++-Bibliotheken. Um dem Nutzer jedoch Freiheiten bei der Gestaltung der Berichte zu geben, wird ein auf TeX [26] basierendes Dokument erstellt. Dafür muss das XML-Format in die TeX-Syntax übersetzt werden.

Mit der XSL-Transformation gibt es bereits ein standardisiertes Verfahren für die Übersetzung von XML. Da XSLT jedoch ineffizient ist, wenn das Ergebnis der Transformation selbst nicht XML ist, wird ein Zwischenschritt für die Übersetzung eingefügt. Das Dokument wird zunächst per XSLT in die TeXML übersetzt. Dabei kann der Nutzer durch Anpassungen an der XSLT-Beschreibung das Layout des späteren Dokuments komplett anpassen. TeXML [27] ist eine XML-basierte Beschreibungssprache für LaTeX-Dokumente.

V. AUSGEWÄHLTE DETAILS DER IMPLEMENTIERUNG

A. Uhrenmodell

Die global synchronisierte Zeit und das Scheduling von jeder Komponente sind die wichtigsten Unterschiede von TTEthernet gegenüber Standard-Ethernet. Daher benötigt jeder Teilnehmer des Netzwerks ein eigenes Uhrenmodell, das über ein fehlergesichertes Synchronisationsprotokoll mit den Uhren der anderen Teilnehmer synchronisiert wird. Die kleinste Einheit der Uhr in TTEthernet ist ein *Tick*. Dabei ist die Zeit des Ticks konfigurierbar.

Bei der Modellierung der Uhr muss beachtet werden, dass jeder Oszillator eine gewisse Ungenauigkeit besitzt, welche als *Clock-Drift* bezeichnet wird. Diese Clock-Drift hat signifikanten Einfluss auf das Verhalten des Protokolls und muss daher überlegt in das Uhrenmodell integriert werden. Aus offensichtlichen Gründen kann die Clock-Drift nicht simuliert werden, indem jeder Tick als ein separates Event ausgeführt wird. Dies würde die Simulation über die Maßen verlangsamen.

Im vorgeschlagenen Konzept werden mehrere Ticks simultan in einem Schritt simuliert. Dabei wird ein *Clock-Drift-Faktor* eingeführt, welcher die Ungenauigkeit der Uhr nachbildet. Dieser Faktor wird für ein konfigurierbares Intervall als konstant angenommen. Dadurch entfallen diverse Simulationsschritte. Die Gleichung 2 zeigt die Formel für die Berechnung des aktuellen Wertes der Uhr.

$$t' = t + \delta \cdot (\Delta t_{Tick} + \Delta t_{Drift}) \quad (2)$$

Dabei ist t' die Zeit für das nächste geplante Event, t ist die aktuelle Simulationszeit, δ ist die Anzahl von Ticks im Schedule für das nächste Event, Δt_{Tick} ist die konfigurierbare Zeit für einen Tick und Δt_{Drift} ist die aktuelle mittlere Clock-Drift. Dieses Modell der Uhren ist eine zulässige und ausreichend genaue Vereinfachung des Verhaltens von echten Uhren, da während eines Zyklus die Varianz der mittleren Clock-Drift in echten Uhren nur gering ist.

Während die Clock-Drift Uhren ungenau werden lässt, sorgen die zyklischen Synchronisationsnachrichten dafür, dass die Uhr stets auf einen bekannten Punkt im Zyklus zurückgesetzt wird.

B. Switchmodell

TTEthernet benötigt eine Switch-Erweiterung, welche das time-triggered Protokoll verarbeitet. Der TTEthernet-Switch kombiniert ein Standard-Switched-Ethernet-Gerät für die Weiterleitung von best-effort Daten mit einem Modul, welches das Synchronisationsprotokoll und das Weiterleiten von zeitkritischen Daten (time-triggered und rate-constrained) unterstützt. Dieses Modul enthält weiterhin die Schedules und die lokale Uhr.

Für die Implementierung des TTEthernet-Switches wird das MACRelayUnit Interface aus dem INET-Framework verwendet. Die Logik für die Weiterleitung des zeitkritischen Datenverkehrs verwendet das beschriebene Uhrenmodell (siehe Abschnitt V-A).

Wie durch die TTEthernet-Spezifikation vorgegeben, werden ankommende Pakete anhand ihrer Zieladresse klassifiziert

und dann, basierend auf ihrer Nachrichtenart, weitergeleitet oder bearbeitet:

- Synchronisationsnachrichten werden überprüft und an das Synchronisationsmodul des Gerätes weitergeleitet.
- Time-triggered Nachrichten werden in einem Puffer bis zu ihrem geplanten Weiterleitungszeitpunkt zwischengespeichert und dann abgesendet.
- Rate-constrained Nachrichten werden so schnell wie möglich weitergeleitet (jedoch mit geringerer Priorität als time-triggered Nachrichten, um diese nicht zu verzögern).
- Best-effort Nachrichten werden in der übrigen Zeit weitergeleitet.

Direkt nach dem Empfang eines Ethernet-Frames überprüft der TTEthernet-Switch die Nachricht auf Konformität mit den vorkonfigurierten Regeln. Für zeitkritischen Datenverkehr sind die Konformitätsüberprüfungen in der Critical-Traffic-Configuration (CTC)-Tabelle abgelegt. Der Switch kann unter anderem überprüfen, ob die Nachricht auf dem richtigen Port und zum korrekten Zeitpunkt angekommen ist. Während für rate-constrained Nachrichten die Einhaltung des Bandwidth Allocation Gap (BAG) überprüft wird, wird für time-triggered Daten geprüft, ob die Nachricht zur geplanten Zeit eintraf. Diese Analyse ist wichtig, um das System vor der Störung durch fehlerhafte oder bösartige Sender zu schützen. So können Nachrichten von Geräten, die vorgeben, time-triggered Sender zu sein, verworfen werden, sofern sie an einem falschen Port eintreffen. Best-effort Frames sind von der Überprüfung ausgeschlossen. Sie dürfen zu jedem Zeitpunkt des Schedules an jedem Port des TTEthernet-Switches eintreffen. Damit können auch Geräte in das Netzwerk integriert werden, die das time-triggered Protokoll nicht unterstützen.

1) *Hostmodell*: Für die Simulation von TTEthernet-Endsystemen muss der Standard-Ethernet-Protokollstack um die time-triggered Funktionalität erweitert werden. Dafür muss das Nachrichten-Scheduling und der zugehörige Scheduler, die Klassifizierung von eingehenden Nachrichten und das Synchronisationsmodul implementiert werden. Das Nachrichten-Scheduling legt fest, wann eine time-triggered Nachricht gesendet oder empfangen wird. Es wird offline konfiguriert. Für unterschiedliche Events, darunter das Versenden oder Empfangen von Nachrichten, aktiviert der Scheduler sogenannte *Tasks*.

Für die direkte Integration von Anwendungen in das Simulationsmodell wird das TTEthernet-API für das Simulationsmodell implementiert. Über Code-Integration können darüber Anwendungen, die gegen das API programmiert wurden, direkt in der Simulation verwendet werden. Dabei verwendet das Hostmodell die gleichen Datenstrukturen wie die reale Implementierung. Alle Nachrichten werden in Puffern zwischen Anwendungs- und Protokoll-Layer ausgetauscht.

2) *Konfiguration der Simulationskomponenten*: Ein wesentliches Ziel bei der Umsetzung ist die enge Integration des Simulationsprozesses in die TTEthernet-Toolchain von TTEch. Um dieses Ziel zu erreichen, wird die XML-Network-Configuration verwendet, um die Komponenten in der Simulation zu konfigurieren.

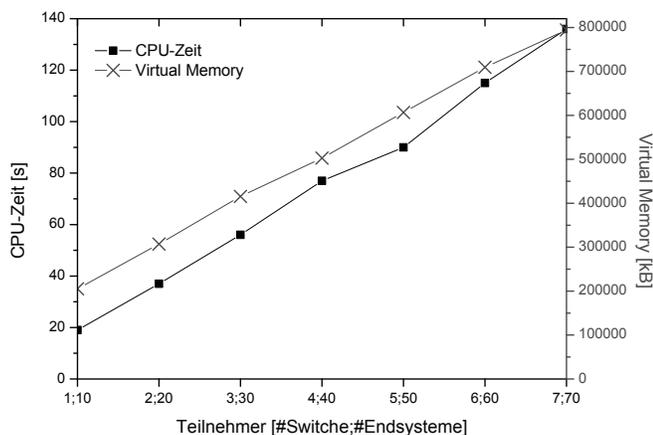


Abbildung 5. Lineares Skalierungsverhalten des Simulationsmodells bei abstrakten Topologien verschiedener Größe

Die Herausforderung beim Einlesen der Konfiguration liegt im Format. Die XMI-Dateien basieren auf den Ecore-Datentypen [28]. Daher kann kein normaler C++ XML-Parser verwendet werden. Auch die Referenzen werden in einem speziellen EMF-konformen Format, sogenannten Fragment-based-URIs, gespeichert. Diese Form der Referenzen weicht von der normalen Syntax in XML-Dokumenten, die in der Xpath [29] definiert sind, ab.

Da das EMF auf Java basiert, stehen hierfür Parser zur Verfügung. Da die Simulation aber in C++ implementiert ist, ist eine Java-Lösung nicht sinnvoll. Wie bei der NED-Generierung wird daher auch für die Konfiguration das EMF4CPP-Framework [30] eingesetzt. Dadurch kann das Einlesen der Konfiguration direkt in der C++ Simulationsumgebung erfolgen. Der Einsatz nur eines Parsers für alle Werkzeuge des Evaluierungsprozesses reduziert zudem die Komplexität des Gesamtprojektes.

VI. EVALUIERUNG & ERGEBNISSE

In diesem Abschnitt werden ausgewählte Simulationsergebnisse, sowie eine Evaluierung der Simulationsperformance aufgezeigt. Weiterhin wird die vorgestellte Simulationsumgebung realen Hardwaremessungen und Ergebnissen aus dem analytischen Modell gegenübergestellt.

A. Simulationsergebnisse und Performance

Die entwickelte TTEthernet-Implementierung wird zunächst auf Basis künstlich generierter Topologien für Fahrzeug-Backbones getestet. Um zu überprüfen, ob das Modell den Anforderungen an die Skalierbarkeit gerecht wird, werden Topologien innerhalb der erwarteten maximalen Backbone-Größe generiert. Basierend auf aktuellen Designs kann vorläufig mit einer Topologie von maximal 70 Hosts und 7 Switchen gerechnet werden.

Die Simulation läuft auf Standard-PC-Hardware mit zwei Kernen à 2,4 GHz und 2 GB Arbeitsspeicher. Experimente zeigen eine lineare Abhängigkeit zwischen dem Speicherverbrauch und der Netzwerkgröße sowie der Anzahl von time-triggered Botschaften im Zyklus. Abbildung 5 zeigt die

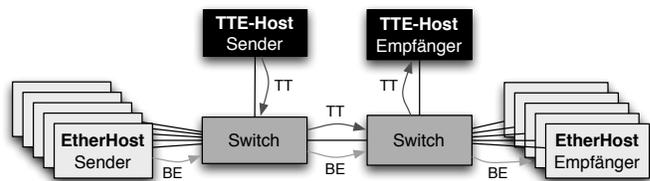


Abbildung 6. Simulierte Topologie für den Nachweis der Protokollkonformität — Parallele Übertragung von time-triggered und best-effort Nachrichten

benötigte CPU-Zeit und den Speicherverbrauch bei Simulationen unterschiedlicher Größe. Dabei wird jeweils über 30 Echtzeitsekunden simuliert. Auf der X-Achse ist die Topologiegröße (Anzahl Switche und Anzahl Hosts) aufgetragen. Es wird dabei eine synthetisch generierte Topologie ausgemessen. Simulationen auf verschiedenen Topologien gleicher Größe zeigen, dass die Rechenzeit und der Speicherverbrauch ausschließlich von der Größe und nicht von der Struktur der Topologie abhängt.

Empirische Untersuchungen innerhalb der maximal erwarteten Netzwerkgröße von 70 Hosts und 7 Switchen für einen zukünftigen Fahrzeug-Backbone zeigen, dass die Simulation für die Experimente ausreichend performant ist.

Um die Metriken und Protokollkonformität des TTEthernet-Modells zu zeigen, wird weiterhin die Latenz und der Jitter eines Beispiel-Netzwerkes mit TTEthernet- und Standard-Ethernet-Switchen aus dem INET-Framework verglichen. Die dafür verwendete Topologie besteht aus zwei verbundenen Switchen, an denen zwei TTEthernet-Hosts für time-triggered und zehn Standard-Ethernet-Hosts für die Erzeugung von best-effort Hintergrunddatenverkehr hängen (siehe Abbildung 6). Alle Links im simulierten Netzwerk haben eine Bandbreite von 100 Mbit/s. Die Clock-Drift wird mit 200 ppm konfiguriert. Die Verzögerung des Kabels beträgt 100 ns auf jedem Link.

Um den Einfluss des best-effort Datenverkehrs auf die Übertragung über den Link zwischen den beiden Switchen zu messen, wird dieser zusätzlich zu den time-triggered Nachrichten zwischen 0 % und 100 % mit best-effort Frames ausgelastet. Während dieser Simulation wird die minimale und maximale Ende-zu-Ende Latenz über 10.000 Nachrichten gemessen. Die Extremwerte der Ende-zu-Ende Latenz sind die wichtigste Metrik für die Echtzeit-Fahrzeugkommunikation (siehe Abschnitt III-B).

1) *Exemplarische Simulationsergebnisse:* Die Ende-zu-Ende Latenz zwischen den time-triggered Hosts wird für die INET-Ethernet-Switch-Implementierung und das TTEthernet-Modell gemessen. Die Ergebnisse (siehe Abbildung 7) zeigen die maximale und minimale Nachrichtenlatenz für die Beispieltopologie (Abbildung 6) in einer Messung über 10.000 Nachrichten, während das Netzwerk zusätzlich mit einer variablen Menge von best-effort Frames belastet wird.

TTEthernet verspricht zuverlässige Kommunikation innerhalb definierter zeitlicher Grenzen. Abbildung 7 zeigt, dass durch das synchronisierte Protokoll, die Priorisierung und die Mechanismen zur deterministischen Medienreservierung eine gleichförmige, von der Linkauslastung unabhängige Latenz er-

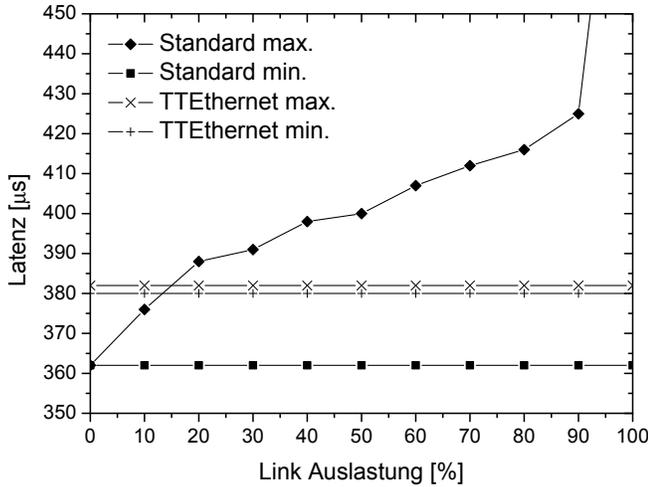


Abbildung 7. Vergleich der Ende-zu-Ende Latenz — Standard INET-Switch und TTEthernet-Switch

reicht wird. Die geringen Schwankungen zwischen minimaler und maximaler Latenz entstehen durch die Clock-Drift der lokalen Uhren (siehe Abschnitt V-A). Der Einfluss der Clock-Drift ist in den Vorarbeiten bereits analytisch verifiziert worden [2].

Dem gegenüber steigt beim INET-Ethernet-Switch die Differenz zwischen minimaler und maximaler Latenz mit zunehmender Link-Auslastung. Dieser Effekt entsteht durch das Aufstauen von Nachrichten an den ausgehenden Ports der Switches. Wird der Link voll ausgelastet, ist die Kommunikation nicht mehr zuverlässig. Es werden Nachrichten verworfen, da die Übertragungskapazität nicht balanciert ist.

Eine weitere wichtige Analyse, welche die Topologie und Konfiguration eines Netzwerkes charakterisiert, ist die *Latenzverteilung*. Graphen zur Latenzverteilung können die Entscheidung unterstützen, ob ein Netzwerk die Anforderungen der Applikationen an die Weiterleitung von event-triggered Datenverkehr erfüllt. Abbildung 8 zeigt die Latenz für die drei Nachrichtenarten von TTEthernet auf demselben voll ausgelasteten TTEthernet Netzwerk. Wie gefordert haben die time-triggered Nachrichten eine nahezu konstante Latenz. Die rate-constrained Nachrichten werden mit einer variablen, jedoch unter $500\mu\text{s}$ begrenzten Verzögerung ausgeliefert. Durch das Aufstauen und die Mechanismen zur Linkreservierung ist im Vergleich dazu beim best-effort Datenverkehr der Jitter hoch.

B. Verifikation des Simulationsmodells

Um zu überprüfen, ob das TTEthernet-Simulationsmodell das Verhalten des Protokolls korrekt abbildet, werden die Simulationsergebnisse mit Ergebnissen aus dem analytischen Modell [2] und Messungen mit echter TTEthernet-Hardware verglichen. Da zum Zeitpunkt der Messungen für die Überprüfung mit echter Hardware nur ein Switch zur Verfügung stand, ist eine einfachere Topologie für die Verifikation genutzt worden. Die Verifikation kann wie hier beschrieben jedoch auf jegliche Topologie angewendet werden.

Die Evaluierung nutzt eine Topologie mit einem Switch und mehreren Hosts. Dabei läuft die Kommunikation zwischen

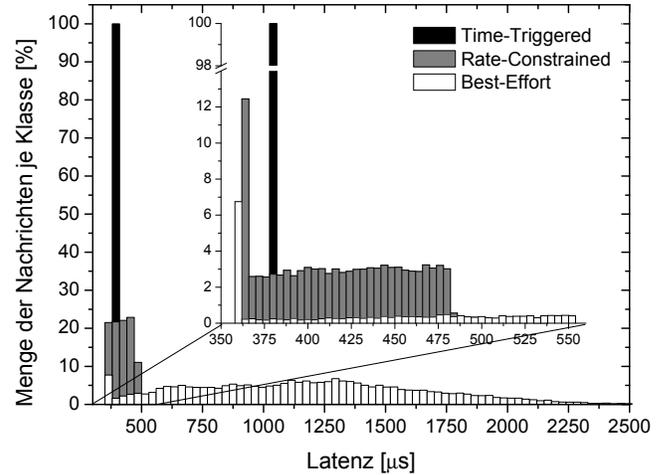


Abbildung 8. Latenzverteilung der drei Traffic-Klassen von TTEthernet bei einem voll ausgelasteten Link

den Teilnehmern stets mit einem Switching-Hop. Während der Evaluierung werden im Switch verschiedene Schedules mit unterschiedlichen Weiterleitungsverzögerungen eingesetzt.

Die Simulation läuft über 10.000 Nachrichten jeweils für einen Switch-Schedule mit $350\mu\text{s}$ Verzögerung zwischen Empfang und Weiterleitung des Frames und für einen Schedule mit $9\mu\text{s}$. Eine Verzögerung von $9\mu\text{s}$ ist zur Zeit das Minimum für eine zuverlässige Weiterleitung.

Der Schedule mit $9\mu\text{s}$ führt zu einer simulierten Ende-zu-Ende Latenz von $19,5\mu\text{s}$ für einen Frame mit minimalem Payload und $252,0\mu\text{s}$ für einen Frame mit maximalem Payload. Für den Schedule mit $350\mu\text{s}$ Weiterleitungsverzögerung beträgt die Ende-zu-Ende Latenz, abhängig von der Payload-Größe, $360,5\mu\text{s}$ bis $593,0\mu\text{s}$. Im Folgenden werden diese Simulationsergebnisse mit dem mathematischen Modell und der Hardware-Messung verglichen.

1) *Vergleich mit mathematischem Modell*: Das mathematische Modell der TTEthernet-Komponenten ist in einer Vorarbeit für einen theoretischen Vergleich der Leistungsmerkmale von TTEthernet und FlexRay entwickelt und veröffentlicht worden [2].

Die folgenden Gleichungen sind ein Teil dieses Frameworks. Mit ihrer Hilfe lässt sich das zeitliche Verhalten einer TTEthernet-Komponente berechnen: Die Latenz t_L wird dabei für minimalen und maximalen Payload berechnet. Sie setzt sich zusammen aus der physikalischen Signallaufzeit auf dem Kabel $t_{WD} \cdot l_W$, der Zeit für die Übertragung der Daten — jeweils beim Sender und dem weiterleitenden Switch — $2 \cdot l_F \cdot t_b$ und der Verzögerung durch den Switch-Schedule t_{SD} :

$$t_L(t_{SD}) = t_{WD} \cdot l_W + 2 \cdot l_F \cdot t_b + t_{SD} \quad (3)$$

$$t_{L_{min}}(t_{SD}) = 10 \frac{\text{ns}}{\text{m}} \cdot 0,5 \text{ m} + 2 \cdot 5,12 \mu\text{s} + t_{SD} \quad (4)$$

$$t_{L_{max}}(t_{SD}) = 10 \frac{\text{ns}}{\text{m}} \cdot 0,5 \text{ m} + 2 \cdot 121,44 \mu\text{s} + t_{SD} \quad (5)$$

Wie erwartet besteht eine lineare Abhängigkeit zur Frame-Länge l_F . Für den Schedule mit $t_{SD} = 350\mu\text{s}$ liegt die berechnete Latenz je nach Payload-Größe zwischen $360,245\mu\text{s}$

Tabelle II
VERGLEICH DER LATENZGRENZWERTE BEI MINIMALEM UND
MAXIMALEM PAYLOAD FÜR VERSCHIEDENE SWITCH-SCHEDULES

Weiterleitungs- verzögerung	Payload	Simulation	Theorie	Messung
350 μ s Schedule	minimum	360,5 μ s	360,245 μ s	360 μ s
	maximum	593,0 μ s	592,885 μ s	592 μ s
9 μ s Schedule	minimum	19,5 μ s	19,245 μ s	-
	maximum	252,0 μ s	251,885 μ s	-

und 592,885 μ s. Dieselbe Berechnung für den Schedule mit der minimalen Verzögerung von $t_{SD} = 9 \mu$ s ergibt eine Latenz zwischen 19,245 μ s und 251,885 μ s.

Die Ergebnisse des mathematischen Frameworks sind geringfügig (<300 ns) niedriger als die Simulationsergebnisse. Der Grund sind Vereinfachungen im mathematischen Modell bei der Abbildung von Hardware-Verzögerungen. Diese Verzögerungen werden in der Simulation präzise berücksichtigt.

2) *Vergleich mit Messungen auf TTEthernet-Hardware:* Um die Qualität des Simulationsmodells besser bewerten zu können, werden die Ergebnisse mit dem Verhalten echter TTEthernet-Hardware verglichen. Dazu wird eine in Vorarbeiten erstellte Messmethode [31] verwendet. Sie ist auf einem eingebetteten System mit zwei Ethernet-Ports und einem Echtzeit-Linux-Kernel umgesetzt. Das *Network under Test* — also das auszumessende Netzwerk — wird zwischen den beiden Ports des Systems angeschlossen. Für die Messung werden Nachrichten mit Zeitstempel von einem Port abgesendet, durchlaufen das Netzwerk und erreichen den zweiten Port des Systems. Beim Empfang wird ein zweiter Zeitstempel aufgezeichnet. Diese Methode erlaubt die Messung von Latenz und Jitter des angeschlossenen Netzwerkes mit einer Genauigkeit im einstelligen Mikrosekunden-Bereich.

Für das simulierte Netzwerk mit einem Schedule von 350 μ s Verzögerung ergibt sich eine gemessene Latenz zwischen 360 μ s und 592 μ s je nach Framegröße. Bei Vernachlässigung der niedrigeren Präzision der Messmethode gegenüber der Simulation sind die Ergebnisse als vergleichbar anzusehen. Der Schedule mit 9 μ s Verzögerung konnte nicht gemessen werden, da die Messmethode auf einer Softwareimplementierung des TTEthernet-Stacks basiert, dessen Präzision der Synchronisierung nicht ausreicht. Infolgedessen ergeben sich bei diesem Schedule partielle Paketverluste bei der Softwarebasierten Messmethode. Durch eine Hardware-nahe Implementierung der Messung auf einem ARM9-System soll die Messung jedoch zukünftig ohne Paketverlust möglich sein.

3) *Bewertung der Verifikation:* Das Simulationsmodell implementiert das erwartete Verhalten eines TTEthernet-Systems mit einer hohen Genauigkeit. Die Ergebnisse der Experimente zeigen, dass das Modell konform mit der TTEthernet-Spezifikation, dem mathematisch-analytischen Modell und der Hardware-Messung arbeitet. Die Latenz der time-triggered Nachrichten ist unabhängig vom simultan gesendeten best-effort Datenverkehr. Tabelle II zeigt die Ergebnisse der verschiedenen Messungen und Experimente für die Schedules mit 9 μ s und 350 μ s im Vergleich.

VII. ZUSAMMENFASSUNG & AUSBLICK

Durch die Zunahme von elektronischen Systemen im Automobil sind die Anforderungen an Kommunikationsverbindungen im Automobil stark gestiegen. Dies motiviert neue Konzepte für den Datenaustausch zwischen den verteilten Steuergeräten. Ein solcher neuer Architekturansatz ist ein Backbone-Netzwerk, das als zentrales Element aller Datenverbindungen hoher Last gewachsen sein muss und dabei eine vorhersagbare, deterministische Nachrichtenübertragung garantiert. Zudem bringen neue Fahrerassistenzsysteme wie die Fahrzeugumfeldererkennung, welche mithilfe kamerabasierter Objekterkennung den Fahrer unterstützt, heutige automotive Bussysteme an ihre Grenzen. Eine Zunahme der Datenmenge in diesem Segment ist zudem bereits absehbar. Eine mögliche Technologie für zukünftige Fahrzeugnetzwerke ist Echtzeit-Ethernet. Ethernet ist eine flexibles, hoch skalierbare Datenübertragungstechnologie, welches durch Echtzeiterweiterungen um die Funktionalität der deterministischen, zuverlässigen und zeitlich hoch präzisen Paketvermittlung erweitert wird. Um frühzeitig Aussagen zur Eignung von Ethernet-basierter Fahrzeugkommunikation machen zu können, spielt die Simulation von konkreten Anwendungsszenarien eine wichtige Rolle für OEM und Zulieferer. Ein flexibler Evaluierungsprozess unterstützt dabei die Entwicklung kontinuierlich.

Der in dieser Arbeit entwickelte Prozess basiert auf drei Phasen. In der Einrichtungsphase wird die Simulation aus vorhandenen Konfigurationsformaten eingerichtet. In der Simulationsphase wird die Simulation auf Basis dieser Konfigurationen stimuliert. In der Auswertungsphase bekommt der Nutzer die Simulationsergebnisse in Form eines Analyseberichtes aufbereitet. Dabei wird darauf Wert gelegt, dass die Simulationsergebnisse direkt auf die Konfiguration und damit auf die Eigenschaften realer Hardware zurückführbar sind.

Die Herausforderung beim Simulationsmodell liegt insbesondere in der präzisen Umsetzung des Zeitverhaltens. Da TTEthernet ein synchron arbeitendes Protokoll ist, muss eine geeignete Abbildung der lokalen Uhren in jedem Gerät und der in jedem Zeitgeber vorhandenen Clock-Drift gefunden werden. Die erzielten Ergebnisse zeigen, dass das Simulationsmodell spezifikationskonform arbeitet. Die Performance-Untersuchung an Modellen innerhalb der erwarteten Topologiegröße für automotive Anwendungen zeigt, dass das reine Modell der Komponenten ausreichend gut skaliert. Die benötigte Rechenzeit und der Speicherverbrauch wachsen linear mit der Modellgröße. Die Verifikation zeigt auf Basis einer synthetischen Topologie, dass das Simulationsmodell eine hohe zeitliche Präzision aufweist. Dazu werden die Simulationsergebnisse mit den Ergebnissen aus dem analytischen Modell und Messungen mit echter TTEthernet-Hardware verglichen.

In einer Fallstudie wurde der Evaluierungsprozess mit einer realen automotive Anwendung getestet. Dabei wurde eine reale Fahrzeuganwendung mit hohen Bandbreitenanforderungen — die kamerabasierte Fahrzeugumfeldererkennung — durch den Evaluierungsprozess verarbeitet (siehe Abbildung 9). Damit kann die praktische Relevanz eines simulationsbasierten

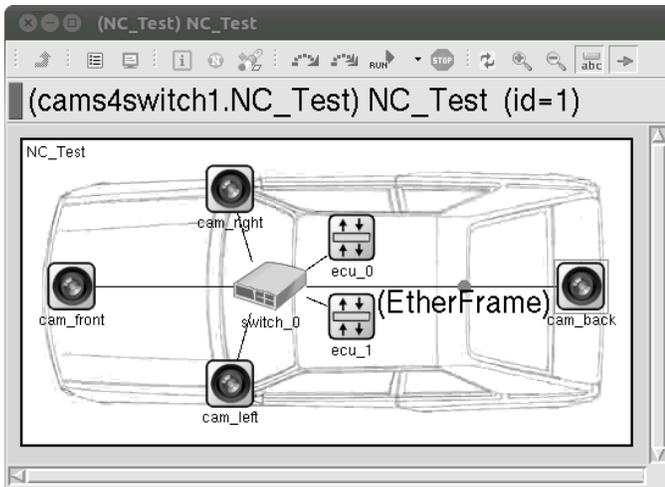


Abbildung 9. Grafische Darstellung der Anwendung in der Simulation

Ansatzes für die automotiv Entwicklung gezeigt werden.

Diese Arbeit zeigt ein präzises, gut skalierendes Simulationsmodell für Echtzeit-Ethernet-basierte Kommunikation in Fahrzeugen. Durch die sorgfältige Validierung auf Basis von mathematischen Berechnungen in einem analytischen Modell und Messungen auf echter TTEthernet-Hardware kann sichergestellt werden, dass die Ergebnisse der Simulation auf reale Probleme anwendbar sind.

Zukünftige Simulationsarbeiten auf Basis konkreter Automobiltopologien und realistischer Anwendungsdaten begleiten die Analyse, wie time-triggered Ethernet-Systeme in einem zukünftigen Fahrzeug-Backbone umgesetzt werden können. Dabei unterstützt der in dieser Arbeit vorgestellte Evaluierungsprozess und die darin enthaltene Simulationsumgebung die Entwicklung von prototypischen Umsetzungen und konkreten Lösungen wie der Konsolidierung aktueller automotiv Bussysteme zu einer intelligenten Ethernet-basierten Vermittlungsinfrastruktur.

LITERATUR

- [1] W. Steiner, "TTEthernet Specification," TTTech Computertechnik AG, Nov. 2008. [Online]. Available: <http://www.tttech.com>
- [2] T. Steinbach, F. Korf, and T. C. Schmidt, "Comparing Time-Triggered Ethernet with FlexRay: An Evaluation of Competing Approaches to Real-time for In-Vehicle Networks," in *8th IEEE Intern. Workshop on Factory Communication Systems*. Piscataway, New Jersey: IEEE Press, May 2010, pp. 199–202.
- [3] FlexRay Consortium, "Protocol Specification," FlexRay Consortium, Stuttgart, Specification 2.1, Dec. 2005.
- [4] TU Wien, "The ttp protocols," 1997. [Online]. Available: <http://www.vmars.tuwien.ac.at/projects/ttp/ttpmain.html>
- [5] Aeronautical Radio Incorporated, "Aircraft Data Network," ARINC, Annapolis, Maryland, Standard 664, 2002.
- [6] R. Belschner, J. Berwanger, C. Bracklo, C. Ebner, B. Hedenetz, W. Kuffner, P. Lohrmann, J. Minuth, M. Peller, A. Schedl, and V. Seefried, "Anforderungen an ein zukünftiges Bussystem für fehlertolerante Anwendungen aus Sicht Kfz-Hersteller," *VDI-Berichte*, vol. 1547, pp. 23–41, 2000.
- [7] T. Dohmke, "Bussysteme im Automobil: CAN, FlexRay und MOST," Technische Universität Berlin, DaimlerChrysler AG, Tech. Rep., 2002.
- [8] LIN-Administration, "Local Interconnect Network." [Online]. Available: <http://www.lin-subbus.org/>
- [9] International Organization for Standardization, "Road vehicles – Controller area network (CAN)," ISO, Genf, ISO 11898, 2003.
- [10] MOST Cooperation, "Media Oriented Systems Transport." [Online]. Available: <http://www.mostcooperation.com/>
- [11] EtherCAT Technology Group, "EtherCAT." [Online]. Available: <http://www.ethercat.org>
- [12] TTTech Computertechnik AG, "TTEthernet" Wien. [Online]. Available: <http://www.tttech.com>
- [13] SAE - AS-2D Time Triggered Systems and Architecture Committee, "Time-Triggered Ethernet (AS 6802)," 2009. [Online]. Available: <http://www.sae.org>
- [14] OMNeT++ Community, "OMNeT++ 4.0." [Online]. Available: <http://www.omnetpp.org>
- [15] OMNeT++ Community, "INET Framework for OMNeT++ 4.0." [Online]. Available: <http://inet.omnetpp.org/>
- [16] B. Müller-Rathgeber, M. Eichhorn, and H.-U. Michel, "A unified car-it communication-architecture: Design guidelines and prototypical implementation," in *IEEE Intelligent Vehicles Symposium, 2008*, Jun. 2008, pp. 709–714.
- [17] B. Müller-Rathgeber, M. Eichhorn, and H.-U. Michel, "A unified car-it communication-architecture: Network switch design guidelines," in *IEEE International Conference on Vehicular Electronics and Safety, 2008*, Sep. 2008, pp. 16–21.
- [18] B. Müller-Rathgeber and H.-U. Michel, "Automotive network planning - a genetic approach," in *IEEE Intelligent Vehicles Symposium, 2009*, Jun. 2009, pp. 1088–1092.
- [19] J. Hillebrand, M. Rahmani, R. Bogenberger, and E. Steinbach, "Coexistence of Time-Triggered and Event-Triggered Traffic in Switched Full-Duplex Ethernet Networks," in *International Symposium on Industrial Embedded Systems, 2007. SIES '07.*, Jul. 2007, pp. 217–224.
- [20] J. Jaspersteite, *Leistungsbewertung eines lokalen Netzwerkes mit Class-of-Service Unterstützung für die prozessnahe Echtzeitkommunikation*, ser. Kommunikationstechnik. Aachen: Shaker Verlag, Oct. 2002.
- [21] L. Seno and C. Zunino, "A simulation approach to a Real-Time Ethernet protocol: EtherCAT," in *IEEE Int. Conference on Emerging Technologies and Factory Automation 2008*, Sep. 2008, pp. 440–443.
- [22] P. Ferrari, A. Flammini, and S. Vitturi, "Performance analysis of PROFINET networks," *Computer Standards & Interfaces*, vol. 28, pp. 369–385, Apr. 2006.
- [23] H. Kopetz, *Real-time Systems: Design Principles for Distributed Embedded Applications*, 8th ed., ser. The Kluwer international series in engineering and computer science: real-time systems. Boston: Kluwer Academic, 2004.
- [24] K. Reif, *Automobilelektronik*. Wiesbaden: Vieweg und Teubner, 2009.
- [25] T. Steinbach, H. Dieumo Kenfack, F. Korf, and T. C. Schmidt, "An Extension of the OMNeT++ INET Framework for Simulating Real-time Ethernet with High Accuracy," in *Proceedings of the 4th International ICST Conference on Simulation Tools and Techniques*, 2011, to appear.
- [26] D. E. Knuth, *Digital Typography*. Stanford: Center for the Study of Language and Information (CSLI), Jun. 1998.
- [27] O. Parashchenko, "TeXML 2.0.2," Jan. 2011. [Online]. Available: <http://getfo.org/texml/>
- [28] Eclipse Foundation, "Eclipse Modeling Framework." [Online]. Available: <http://www.eclipse.org/modeling/emf/>
- [29] World Wide Web Consortium, "XML Path Language (XPath)," W3C Recommendation, Nov. 1999. [Online]. Available: <http://www.w3.org/TR/xpath/>
- [30] A. Senac and D. Sevilla, "EMF4CPP." [Online]. Available: <http://www.catedrasaes.org/trac/wiki/EMF4CPP>
- [31] F. Bartols, T. Steinbach, F. Korf, and T. C. Schmidt, "Performance Analysis of Time-Triggered Ether-Networks Using Off-the-Shelf-Components," in *14th IEEE International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing Workshops (ISORCW) 2011*, Mar. 2011, pp. 49–56.