

TECHNISCHE UNIVERSITÄT DRESDEN

FAKULTÄT INFORMATIK
INSTITUT FÜR SYSTEMARCHITEKTUR
LEHRSTUHL RECHNERNETZE

Kurzfassung der Dissertation

zur Erlangung des akademischen Grades
Doktoringenieur (Dr.-Ing.)

**Emulation von Netzwerkverhalten für
Skalierbarkeitstests IP-basierter
Audio/Video-Kommunikationssysteme**

Dipl.-Inf. Robert Lübke
geboren am 28. September 1985 in Schwedt / Oder

Betreuender Hochschullehrer:
Prof. Dr. rer. nat. habil. Dr. h. c. Alexander Schill, TU Dresden

Dresden, im April 2015

1 Motivation

Das Internet hat in den letzten Jahren einen ungeheuren Boom im Bereich der Audio/Video-Anwendungen erfahren. Mit Voice over IP (VoIP) und Videotelefonie setzte sich die Audio/Video-Kommunikation in Echtzeit über das Internet durch. Multimediakommunikation findet heute neben dem privaten Bereich hauptsächlich im Geschäftskontext statt. Video- und Web-Konferenzsysteme ermöglichen Besprechungen von weltweit verteilten Mitarbeitern globaler Unternehmen und erhöhen somit die Produktivität. Dabei stellen sie durchaus Dienstanforderungen, welche von denen der traditionellen Anwendungen wie E-Mail, World Wide Web und File-Sharing stark abweichen. Insbesondere bei der **Echtzeitkommunikation** spielen die Ende-zu-Ende-Verzögerung und deren Schwankung eine entscheidende Rolle. Die Qualität der Kommunikation und damit die Zufriedenheit der Kommunikationsteilnehmer sinkt, sobald sich diese Werte erhöhen. Gelegentliche Datenverluste sind dagegen tolerierbar ohne starke Auswirkungen auf die Qualität.

Aufgrund der großen Konkurrenz auf dem Markt geschieht die (Weiter-)Entwicklung von Multimedia-Anwendungen im Internet heutzutage innerhalb kurzer Zyklen. Dies erfordert zwingend eine systematische Herangehensweise und **kontinuierliches Testen** der konzipierten und implementierten Komponenten. Die dabei auftretenden und in dieser Arbeit adressierten Probleme werden im Folgenden kurz erläutert. Mit der Popularität von IP-basierten Audio/Video-Kommunikationssystemen steigt auch die Anzahl der dazugehörigen Rechenknoten, sodass sie im praktischen Einsatz riesige Ausmaße annehmen können. Aktuelle Video- und Web-Konferenzsysteme unterstützen beispielsweise bis zu 1000 Teilnehmer innerhalb einer einzigen Sitzung. Dies ist eine besondere Herausforderung für den Testprozess, welche **Skalierbarkeitsuntersuchungen** derartiger Systeme bereits im Entwicklungsstadium notwendig macht, um limitierenden Faktoren und Engpässen frühzeitig entgegenwirken zu können. Manuelle Tests von solch großen Szenarien erfordern einen enorm **hohen Koordinationsaufwand**, weswegen sie meist wirtschaftlich nicht vertretbar sind. Ein anderes Problem ist die sich immer stärker ausprägende **Heterogenität der Endpunkte**, die für die Multimediakommunikation eingesetzt werden. Neben PCs werden auch Notebooks, Smartphones und Tablets unterschiedlicher Hersteller und mit unterschiedlichen Betriebssystemen verwendet. Diese verschiedenen Endpunktarten müssen in Tests des Gesamtsystems involviert sein, um deren Funktionsfähigkeit nachzuweisen. Auch der Aspekt der Mobilität darf beim Testen nicht vernachlässigt werden. Generell sind die verschiedenen Endpunkte über unterschiedliche Netzwerkzugangstechnologien wie DSL, Kabel oder Mobilfunk an das Internet angeschlossen. Diese Technologien weisen zahlreiche **verschiedene Netzwerkparameter und -effekte** auf, die im Rahmen von Tests berücksichtigt werden müssen, denn sie wirken sich letztlich auf das Verhalten des Audio/Video-Kommunikationssystems aus. Abbildung 1.1 zeigt abschließend ein typisches Anwendungsszenario von IP-basierter Kommunikation über das Internet und verdeutlicht insbesondere die Heterogenität der beteiligten

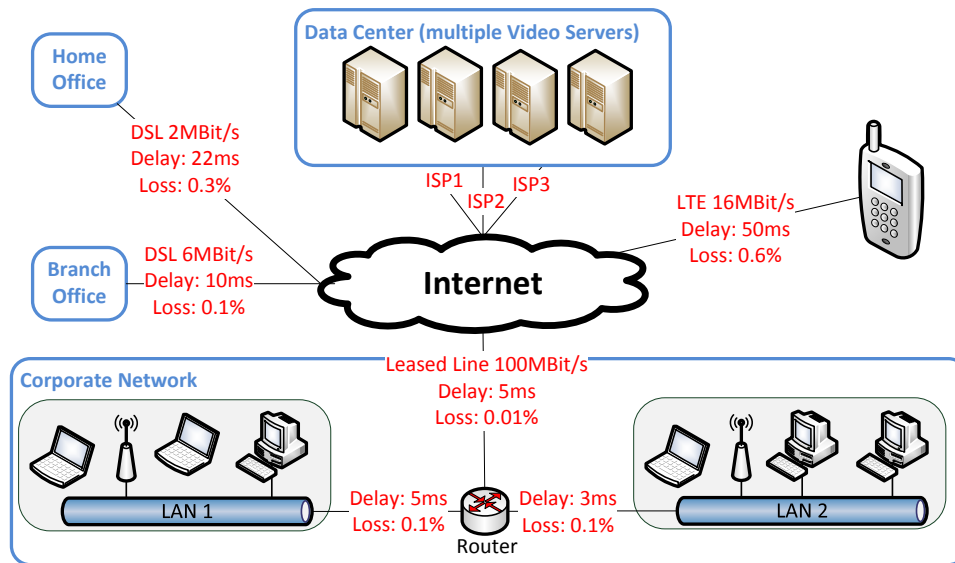


Abbildung 1.1: Anwendungsszenario

Geräte als auch der Netzwerktechnologien. Im Rahmen von Tests gilt es also, solche Szenarien mit bis zu 1000 beteiligten Teilnehmern realitätsgetreu und reproduzierbar nachzubilden.

Die in dieser Dissertation vorgeschlagene Lösung für die angesprochenen Probleme des Testens komplexer Multimediaanwendungen im Internet kombiniert die Durchführung von potentiell sehr groß angelegten Tests mit der Nachbildung unterschiedlichen Netzwerkverhaltens mittels Emulation und der Nachbildung von Anwendungsverhalten.

2 Ziele und Forschungsthemen der Arbeit

Das Ziel dieser Dissertation ist die Entwicklung einer Testplattform für IP-basierte Audio/Video-Kommunikationssysteme, die zusätzlich das Netzwerkverhalten berücksichtigt. Dabei sollen das Verhalten und die Charakteristik des genutzten Netzwerks sowie konkrete Netzwerktechnologien realitätsnah nachgebildet werden. Im Rahmen der Tests soll das Anwendungsverhalten möglichst unter Nutzung der originalen zu testenden Anwendung geschehen, um eine hohen Realitätsbezug zu erreichen. Die Plattform soll insbesondere für sehr große Skalierbarkeitstests des IP-basierten Audio/Video-Kommunikationssystems einsetzbar sein. Dabei soll der Koordinationsaufwand für den Nutzer gering gehalten werden und die Testplattform soll sich möglichst nahtlos in den Arbeitsablauf der Softwareentwickler integrieren. Die Testplattform soll weiterhin berücksichtigen, dass üblicherweise eine hohe Heterogenität an Plattformen, Betriebssystemen und Geräten der Nutzerendpunkte des Audio/Video-Kommunikationssystems vorherrscht.

Die Analyse des aktuellen Stands der Technik und Forschung zeigt, dass zwar viele Lösungen zum Nachbilden von Netzwerkcharakteristiken existieren (siehe Abschnitt 3), aber diese erfüllen nicht die hohen Anforderungen, die bei Tests IP-basierter Audio/Video-Kommunikationssysteme an die

Netzwerknachbildung gestellt werden. Außerdem bieten sie keine Möglichkeit der Nachbildung von Anwendungsverhalten unter Nutzung der originalen zu testenden Software. Diese Kombination wurde bislang noch nicht ausreichend betrachtet. Im Rahmen der Dissertation sollen daher die folgenden Forschungsthese nachgewiesen werden:

These I: Mittels Software-basierter Netzwerkemulation lässt sich Netzwerkverhalten mit ausreichender Genauigkeit für Skalierbarkeitstests IP-basierter Audio/Video-Kommunikationssysteme nachbilden.

These II: Auf Basis Software-basierter Netzwerkemulation lassen sich Skalierbarkeitstests heterogener IP-basierter Audio/Video-Kommunikationssysteme mit komplexer Netzwerktopologie und realem Anwendungsverhalten reproduzierbar und effizient durchführen.

Mit der Forschungsthese I soll also die allgemeine Einsatzfähigkeit von Software-basierter Netzwerkemulation im Kontext der Software-Tests nachgewiesen werden, wobei insbesondere die Genauigkeit der Netzwerknachbildung den Anforderungen entsprechen muss. Darauf aufbauend soll mit These II nachgewiesen werden, dass sich die Netzwerkemulation auch gleichzeitig mit der Nachbildung von Anwendungsverhalten kombinieren lässt, wie es bei Software-Tests notwendig ist. Der Fokus liegt dabei sowohl auf der Heterogenität der beteiligten Endpunkte als auch auf der Reproduzierbarkeit und Effizienz der Testdurchführung. Die Innovation dieser Arbeit besteht also darin, die Ergebnisse des bereits umfassend bearbeiteten Forschungsgebietes Netzwerkemulation zu nutzen und für Tests IP-basierter Audio/Video-Kommunikationssysteme anzuwenden.

3 Existierende Ansätze zur Netzwerknachbildung

Im Rahmen von Tests von Echtzeitkommunikationssystemen sollte aufgrund der Komplexität nur das tatsächlich relevante Netzwerkverhalten nachgebildet werden. Dazu gehören die Limitierung der Datenrate, die Paketverzögerung sowie deren Varianz, der Paketverlust, die Änderung der Paketreihenfolge, Paketvervielfältigung sowie -verfälschung. Bei allen diesen Netzwerkeigenschaften ist zu beachten, dass sie sich lediglich auf einen Netzwerkknoten oder eine -verbindung beziehen. Da die Kommunikation in Rechnernetzen in der Regel über mehrere Knoten und Verbindungen hinweg geschieht (Multi-Hop-Verbindungen), muss in der Nachbildung ebenfalls die Kumulation der Eigenschaften beachtet werden. Zusätzlich zu den genannten Eigenschaften müssen bei der Nachbildung mobiler Übertragungstechnologien die Effekte Verbindungsabbruch und -übergabe (Handover) berücksichtigt werden.

Als konkrete Methoden der Netzwerknachbildung bieten sich Simulation, Emulation sowie ein hybrider Ansatz an. Die **Netzwerksimulation** verfolgt einen synthetischen Ansatz und benötigt kein physisches Netzwerk als Grundlage. Vielmehr wird das nachzubildende System in ein Softwaremodell transferiert, das den Simulationsberechnungen zu Grunde liegt. Die Güte dieses Modells ist ausschlaggebend für den Realitätsbezug und die Detailtreue der Simulation. Es

existieren bereits einige Netzwerksimulatoren, die reale Netze mit hochkomplexen Modellen sehr detailgetreu nachbilden können. Bei Experimenten mit Simulatoren wie ns-2 [BEF⁺00], ns-3 [HRFR06], OMNet++ [VH08] und JiST [BHvR05] muss aufgrund der Komplexität der Berechnungen oft ein Kompromiss zwischen Modellgüte und Größe bzw. Dauer des Experiments gefunden werden. Zum Testen von komplexen Softwaresystemen eignet sich Netzwerksimulation nicht, denn die zu testende Software müsste dazu ebenfalls in das Berechnungsmodell transferiert werden. Dies widerspricht dem Grundgedanken des Softwaretests, bei dem möglichst die originale Anwendung untersucht werden sollte.

Der zweite Ansatz ist die **Emulation der Netzwerkeigenschaften** in einem bestehenden physischen Testnetzwerk mit besseren Eigenschaften. Dabei wird die Charakteristik von realem Verkehr dieses Testnetzwerks so modifiziert, dass sie einer gegebenen Konfiguration entspricht. Dazu werden zwischen zwei Protokollschichten im Netzwerk-Stack die Daten abgegriffen, in eine Emulationskomponente geleitet und dort entsprechend geändert.

Die dritte Art der Nachbildung folgt einem **hybriden Ansatz**, der Simulation und Emulation kombiniert. Eine Möglichkeit ist es, beide Verfahren zeitlich voneinander getrennt durchzuführen. Zuerst werden in einer Simulation konkrete Netzwerkparameter und -einstellungen berechnet, die in der danach folgenden Emulation verwendet werden. Dieser in [CPGD10] vorgestellte Ansatz ist vorteilhaft, wenn komplexe und realitätsnahe Simulationsmodelle in Kombination mit echtem Netzwerkverkehr genutzt werden sollen. Eine zeitliche Entkopplung ist im Rahmen von Software-Tests jedoch nicht immer möglich. Alternativ lassen sich auch die physischen Netzwerkschnittstellen während einer Simulation nutzen. Entsprechend erweiterte Netzwerksimulatoren können somit errechnete Netzwerkpakete als realen Verkehr ins Netzwerk einspeisen. Dieser in ns-3 [HRFR06] verfügbare Ansatz ist aber nur realistisch, wenn die Abläufe in Echtzeit geschehen. Wird die Simulation z. B. durch große und komplexe Berechnungsmodelle verlangsamt, entspricht der abgegebene Netzwerkverkehr nicht mehr den realen Bedingungen.

Von den drei vorgestellten Möglichkeiten stellt sich die Emulation als bester Ansatz zur Netzwerknachbildung im Rahmen von Software-Tests heraus. Emulation ist wiederum auf drei verschiedene Arten möglich. **Hardware-basierte Emulatoren** können die Verbindung zwischen bestimmten Netzwerkports künstlich verschlechtern. Sie gelten als sehr performant und hochgenau. Es mangelt ihnen jedoch an der für Software-Test benötigten Flexibilität in der Konfiguration. Diese Anforderung wird von dynamisch konfigurierbaren **Software-basierten Netzwerkekulatoren** wie NetEm [SH05], WANEm [KN11], DummyNet [CR10] und KauNet [GHB08] erfüllt. Sie greifen in den Netzwerk-Stack des Betriebssystems ein und manipulieren so den ein- und ausgehenden Verkehr. Ihre Leistung und Genauigkeit sind von der konkreten Implementation sowie der verwendeten Hardware abhängig. **Emulationsumgebungen** wie PlanetLab [PR06], Distem [SBJN13], GLab [SRZ⁺14], GENI [BCL⁺14] und FIRE [CF14] sind Testnetzwerke mit oftmals sehr vielen verteilten Knoten. Sie entstehen meist durch Initiativen und Zusammenschlüsse mehrerer Universitäten, die ihre Testumgebungen zusammenschalten, um größere und verteilte Testplattformen zu erhalten. In vielen dieser Umgebungen wird ebenfalls Netzwerkekulation eingesetzt, um bestimmtes Netzwerkverhalten innerhalb der durchgeführten Experimente nach-

zubilden [CR11]. Allerdings liegt der Fokus der Emulationsumgebungen auf der Entwicklung von Netzwerkprotokollen und nicht auf dem Testen von verteilten Systemen.

Letztlich kann herausgestellt werden, dass Software-basierte Netzwerke Emulatoren einen Großteil der gestellten Anforderungen erfüllen und somit den besten Ansatz zur Netzwerknachbildung im Kontext von Software-Tests verfolgen.

4 Methodologie

Das zum Nachweis der Forschungsthese eingesetzte Vorgehen gliedert sich in die in Abbildung 4.1 gezeigten vier Teilschritte. Um in Experimenten reale Rechnernetze detailgetreu nachahmen zu können, muss vorher eine genaue Aufnahme der nachzubildenden Eigenschaften, Bedingungen und Effekte erfolgen. Im ersten Teilschritt gilt es daher gemäß Forschungsthese I, die konkreten Netzwerkmetriken möglichst genau durch eine integrierte Netzwerkmessung zu ermitteln. Mit den untersuchten und selbst entwickelten Methoden lassen sich sowohl reale als auch emulierte Netzwerke analysieren. Im zweiten Schritt werden existierende Emulatoren zur Netzwerknachbildung sowie eigene Erweiterungen bezüglich ihrer Fähigkeiten, Leistung und Genauigkeit evaluiert. Somit können fundierte Aussagen zur erreichbaren Genauigkeit und zum Einsatz für Tests IP-basierter Audio/Video-Kommunikationssysteme getätigt werden.

Zum Nachweis der Forschungsthese II werden im dritten Schritt Methoden zur Ermittlung typischer Parameter verschiedener Netzwerktechnologien aufgezeigt. Dazu werden relevante Technologien beleuchtet, Aufbauten und Vorgehen zur Netzwerkmessung vorgeschlagen und letztlich Messungen durchgeführt. Die so ermittelten Werte dienen bei der realitätsnahen Netzwerknachbildung als Eingabekonfiguration von Netzwerke Emulatoren. Letztlich wird im vierten Schritt ein konkretes System zur Emulation von Netzwerk- und Anwendungsverhalten für Tests sehr großer IP-basierter Audio/Video-Kommunikationssysteme entwickelt. Dabei wird umfangreich evaluiert, dass alle zuvor analysierten Anforderungen von der Test- und Emulationsplattform erfüllt werden.

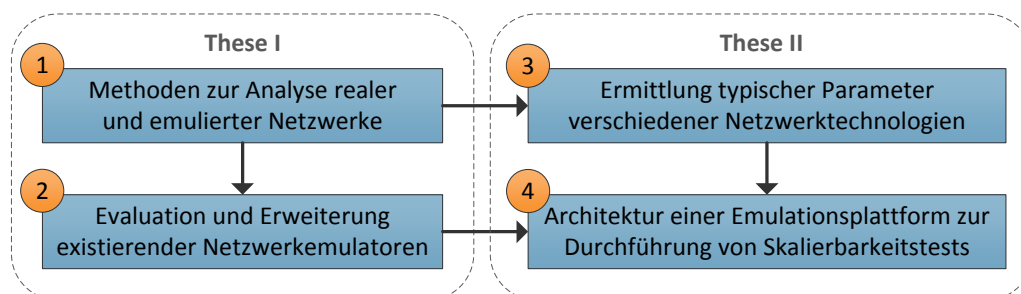


Abbildung 4.1: Vorgehen zum Nachweis der Forschungsthese

5 Ergebnisse und Beiträge

Das zuvor beschriebene Vorgehen zum Nachweis der Forschungsthese führt letztlich zu diversen Ergebnissen und Forschungsbeiträgen, die im Folgenden kurz erläutert werden sollen.

5.1 Methoden zur Analyse realer und emulierter Netzwerke

Die Ermittlung der kompletten Charakteristik eines realen oder emulierten Rechnernetzes ist mit dem aktuellen Stand der Technik zwar möglich, aber dies ist mit einem hohen Aufwand verbunden. Es existieren diverse Messwerkzeuge, die sich allerdings auf eine oder wenige Netzwerkmetriken beschränken. Daher wurde das eigene Messwerkzeug *Nora* entwickelt, welches die integrierte Messung aller relevanten Netzwerkparameter ermöglicht. *Nora* implementiert existierende und etablierte Messkonzepte aus der Forschung, die angepasst und verbessert wurden, und bietet darüber hinaus auch gänzlich neue Konzepte zur Messung mit TCP-Verkehr.

Zur Evaluation der eigenen Konzepte wurden Vergleichsmessungen mit diversen anderen Messwerkzeugen vorgenommen. Die Grundlage dafür stellte ein hochgenauer und leistungsstarker Hardware-basierter Netzwerkemulator. Mit verschiedenen Vorgaben der unterschiedlichen Netzwerkparametern wurde für alle untersuchten Messwerkzeuge die Abweichung von konfigurierten und tatsächlich gemessenen Werten bestimmt. Diese Abweichungen sind für Paketverlust (*One Way Loss Ratio*, Abk. OWLR), Paketumordnung (*One Way Reordering Rate*, Abk. OWRR) und Paketvervielfältigung (*One Way Duplication Rate*, Abk. OWDR) in Abbildung 5.1a und 5.1b dargestellt. Die gemittelten Ergebnisse von *Nora* unter Nutzung von UDP- und TCP-Verkehr entsprechen demnach denen des etablierten Messwerkzeugs *iperf*. Abbildung 5.1c zeigt die mit *Nora* gemessenen Werte für Verzögerung und Jitter sowie die geringe Abweichung von der konfigurierten Normalverteilung. Generell kann den eigenen Konzepten zur integrierten Netzwerkmesung eine dem Stand der Technik entsprechende Genauigkeit nachgewiesen werden.

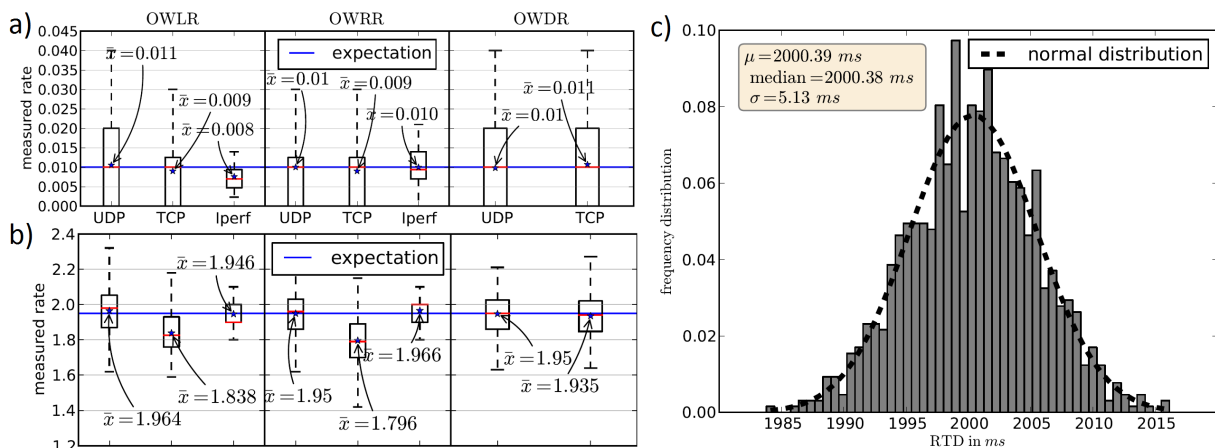


Abbildung 5.1: Evaluation der Genauigkeit des eigenen Messwerkzeugs *Nora*

5.2 Evaluation und Erweiterung existierender Netzwerkemulatoren

Eine Feature-basierte Untersuchung aktueller Software-basierter Netzwerkemulatoren ergab, dass keine existierende Lösung alle relevanten Netzwerkeigenschaften und -effekte unterstützt. Die Emulatoren NetEm, DummyNet und KauNet stellten sich als die vielversprechendsten Kandidaten heraus und wurden daher weitergehenden Leistungs- und Genauigkeitsmessungen unterzogen. Als Referenzpunkt wurde der leistungsfähige Hardware-Emulator Linktropy 7500pro der Firma Apposite eingesetzt.

Die evaluierten Netzwerkemulatoren zeigen deutliche Unterschiede in der Genauigkeit, wobei kein System alle Aspekte zufriedenstellend nachbilden kann. Die Wahl eines passenden Emulators für eigene Netzwerkkperimente ist daher ausschlaggebend für die Qualität der Ergebnisse. Die qualitativen Messergebnisse sind in Tabelle 5.1 festgehalten. Lediglich DummyNet ist in der Lage, die geforderten Netzwerkpfade und -topologien nachzubilden. Dessen Erweiterung KauNet kann durch einen musterbasierten Emulationsansatz perfekte Resultate bei der Nachbildung der Paketeffekte erreichen. Daher soll die Kombination beider Emulatoren als Ausgangsbasis für die weitere Arbeit dienen.

Tabelle 5.1: Empfohlene Anwendungsfelder der evaluierten Netzwerkemulatoren und der eigenen Erweiterung KauNet+

	Linktropy	NetEm	DummyNet	KauNet	KauNet+
Datenrate	+	x	o	-	o
Paketverzögerung	+	+	o	o	o
Paketeffekte	o	o	-	+	+
Netzwerkpfade	x	x	+	x	+

+: hervorragend, o: anwendbar, -: nicht empfohlen, x: nicht unterstützt

Da jedoch weder DummyNet noch KauNet die Emulation von Paketervielfältigung unterstützen, wurde diese Netzwerkeigenschaft in einer eigenen Erweiterung umgesetzt und hinsichtlich der Genauigkeit evaluiert. Die KauNet+ genannte Erweiterung emuliert letztlich alle für Tests IP-basierter Audio/Video-Kommunikationssysteme relevanten Netzwerkeigenschaften und -effekte mit einer ausreichenden Genauigkeit und kann somit als Grundlage für die zu entwickelnde Emulationsplattform dienen.

5.3 Architektur der Emulationsplattform

Weitere wichtige Ergebnisse dieser Dissertation sind die einzelnen vorgeschlagenen Konzepte für eine Test- und Emulationsplattform IP-basierter Audio/Video-Kommunikationssysteme. Die generelle Architektur der NESSEE genannten Plattform ist in Abbildung 5.2 dargestellt.

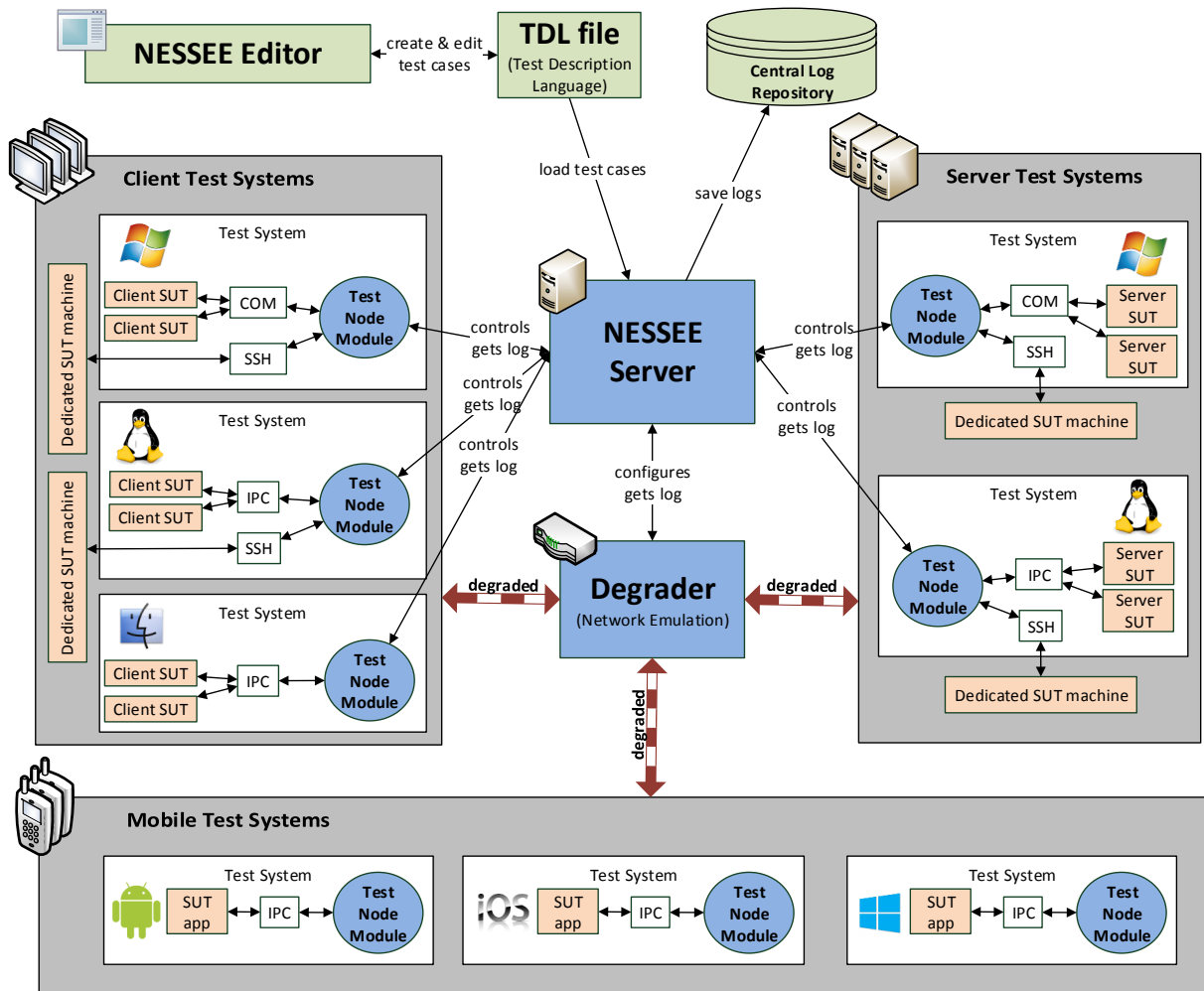


Abbildung 5.2: Allgemeine Architektur der entwickelten Emulationsplattform NESSEE

Der Fokus der Plattform liegt auf dem Testen von Client/Server-basierten Anwendungen, die im folgenden **Software Under Test (SUT)** genannt werden. Die Testsysteme, auf denen letztlich die SUT ausgeführt wird, werden daher ebenfalls in **Client- und Server-Testsysteme** aufgeteilt. Auf einem Client-Testsystem werden üblicherweise mehrere Instanzen der SUT gestartet, um eine hohe Anzahl an emulierten Clients mit möglichst wenig Testsystemen zu erreichen. Neben gängigen Desktop- und Server-Betriebssystemen lassen sich selbst mobile Geräte in die Architektur eingliedern. Auf jedem Testsystem koordiniert das sogenannte **Test Node Module (TNM)** die SUT-Instanzen und kommuniziert mit dem zentralen **NESSEE Server**, der als Koordinator für die gesamte Plattform agiert. Durch den agentenbasierten TNM-Ansatz lassen sich Systeminformationen wie Speicherverbrauch und Prozessorauslastung sowie Informationen zum Ausführungszustand der SUT aufzeichnen und zum NESSEE Server senden, wo sie im **Central Log Repository** zur späteren Analyse gespeichert werden. Der durchzuführende Test wird in der sogenannten **Test Description Language (TDL)** beschrieben. Hierbei handelt es sich um eine selbst entwickelte generische Sprache zur Beschreibung von Tests IP-basierter Echtzeitkom-

munikationssysteme. Das zu testende Anwendungsverhalten kann mit Skripten programmiert werden, die auch Teil der TDL sind. Zur Erstellung der entsprechenden TDL-Dateien für den NESSEE Server kann der Nutzer das grafische Authoring Tool **NESSEE Editor** verwenden. Alle Testsysteme benutzen den sogenannten **Degrader** als Standard Gateway, wodurch ihr Netzwerkverkehr über diesen geleitet wird. Der Degrader, auf dem der selbst weiterentwickelte Netzwerkemulator KauNet+ zum Einsatz kommt, ist verantwortlich für die Nachbildung des gewünschten Netzwerkverhaltens. Basierend auf Quell- und Ziel-IP-Adressen sowie Ports wird auf dem Degrader der Netzwerkverkehr der einzelnen SUT-Instanzen künstlich verschlechtert. Die dafür benötigte Konfiguration erhält der Degrader vom NESSEE Server.

Die einzelnen Komponenten und vorgeschlagenen Konzepte werden umfassend evaluiert. So kann nachgewiesen werden, dass alle an eine Testplattform für IP-basierte Audio/Video-Kommunikationssysteme gestellten Anforderungen erfüllt sind. Insbesondere die gute Skalierbarkeit der gesamten Plattform erlaubt es, Tests mit bis zu 6.000 emulierten Videokonferenzteilnehmern durchzuführen.

5.4 Ermittlung typischer Parameter verschiedener Netzwerktechnologien

Ein weiterer Beitrag sind die entwickelten Konzepte zur Ermittlung typischer Werte von Netzwerkmetriken für verschiedene Netzwerkzugangstechnologien sowie die durch entsprechende Messdurchführungen gewonnenen Werte. Mit diesen Werten lässt sich die zuvor besprochene Test- und Emulationsplattform NESSEE so konfigurieren, dass innerhalb der Experimente eine reale Technologie nachgebildet werden kann.

Es werden sowohl relevante kabelgebundene Technologien wie DSL, das Kabelfernsehtnetz und Ethernet als auch Funktechnologien wie Satellitenkommunikation, WLAN und Mobilfunk näher beleuchtet. Den entsprechenden Spezifikationen dieser Technologien sind meist nur theoretische oder unter perfekten Bedingungen erreichbare Netzwerkeigenschaften zu entnehmen, welche nicht die Realität widerspiegeln. Daher werden ein Messaufbau und -vorgehen vorgeschlagen, mit dem existierende und auch zukünftige Technologien genau analysiert werden können. So lässt sich deren Verhalten in Testdurchläufen genau nachbilden.

6 Zusammenfassung

Die Durchführung von Tests großer verteilter Systeme stellt sich oftmals als schwierig, zeitaufwendig und fehleranfällig heraus. Insbesondere etablierte Audio/Video-Kommunikationssysteme wie Google Hangouts, Skype, WebEx oder Citrix GoToMeeting haben viele Millionen Nutzer weltweit und stellen entsprechend hohe Anforderungen an die Software-Qualität. Ebenso sind in solchen Szenarien Skalierbarkeitstests unerlässlich, denn die Funktionalität muss auch unter Last und in Stresssituationen gewährleistet werden. Wie alle verteilten Systeme müssen auch

Audio/Video-Kommunikationssysteme reproduzierbar unter Nachbildung verschiedener Netzwerkbedingungen getestet werden. Die Besonderheit der Audio/Video-Echtzeitkommunikation ist die ungesicherte, zumeist UDP-basierte Übertragung, bei der sich die vielfältigen Netzwerkeffekte direkt auf die Anwendung auswirken und nicht von Protokollen wie TCP kompensiert werden. Die von den Teilnehmern der Audio/Video-Kommunikation empfundene Qualität wird also direkt durch die Netzwerkeigenschaften beeinflusst. Daher ist insbesondere bei solchen Systemen eine detailgetreue Emulation innerhalb des Testprozesses unerlässlich. Die Analyse verwandter Arbeiten ergab, dass keine Lösung existiert, welche diese Anforderungen erfüllen kann.

Die Dissertation verfolgt daher das Ziel, Konzepte einer Testplattform für IP-basierte Audio/Video-Kommunikationssysteme zu entwerfen sowie diese umzusetzen und zu evaluieren. Dabei liegt der Fokus hauptsächlich auf der Durchführung von sehr großen Skalierbarkeitstests mit beispielsweise 1.000 nachgebildeten Teilnehmern einer Audio/Video-Konferenz, wobei gleichzeitig der Koordinationsaufwand für den Tester gering gehalten werden soll. Eine besondere Herausforderung stellte die notwendige realitätsnahe Nachbildung des Verhaltens und der Charakteristik des zum Einsatz kommenden Netzwerks sowie konkreter Netzwerktechnologien dar.

In der Dissertation können die eingangs beschriebenen Forschungstheseen letztlich nachgewiesen werden. So wird einerseits gezeigt, dass sich die Software-basierte Netzwerkemulation sehr gut zur realitätsnahen Nachbildung von Netzwerkverhalten im Kontext von Software-Tests eignet. Als Nachweis dient die eigene Erweiterung KauNet+, mit der das komplexe Verhalten des Netzwerks genau genug für Skalierbarkeitstests IP-basierter Audio/Video-Kommunikationssysteme reproduziert werden kann. Zum Nachweis von These II wurde eine Emulationsumgebung entwickelt, mit der neben der Netzwerkemulation auch Anwendungsverhalten realitätsnah nachgebildet werden kann. Die damit durchführbaren Tests sind reproduzierbar und durch eine Reduzierung des Koordinationsaufwandes auch sehr effizient.

Neben mehreren kleineren Forschungsbeiträgen stellt die Konzeption, Umsetzung und Evaluierung einer Test- und Emulationsplattform für IP-basierte Audio/Video-Kommunikationssysteme den Hauptbeitrag der Dissertation dar. Die implementierte Testplattform namens NESSEE wird beim Industriepartner Citrix Online hauptsächlich für Skalierbarkeitstests, aber auch für funktionales Testen und Netzwerktests eingesetzt. Mit NESSEE werden Teile der vom Unternehmen entwickelten Audio- und Videoinfrastruktur getestet, die unter anderem in Produkten für Online Meetings, Trainings und Webinare eingesetzt werden.

Literaturverzeichnis

- [BCL⁺14] Mark Berman, Jeffrey S. Chase, Lawrence Landweber, Akihiro Nakao, Max Ott, Dipankar Raychaudhuri, Robert Ricci, and Ivan Seskar. Geni: A federated testbed for innovative network experiments. *The International Journal of Computer and Telecommunications Networking*, 2014. Letzter Abruf am 23.04.2015. URL: <http://dx.doi.org/10.1016/j.bjp.2013.12.037>.
- [BEF⁺00] Lee Breslau, Deborah Estrin, Kevin Fall, Sally Floyd, John Heidemann, Ahmed Helmy, Polly Huang, Steven McCanne, Kannan Varadhan, Ya Xu, and Haobo Yu. Advances in network simulation. *IEEE Computer*, 33(5):59–67, May 2000. Letzter Abruf am 23.04.2015. URL: <http://www.isi.edu/~johnh/PAPERS/Breslau00a.html>.
- [BHvR05] Rimon Barr, Zygmunt J. Haas, and Robbert van Renesse. Jist: an efficient approach to simulation using virtual machines. *Software: Practice and Experience*, 35(6):539–576, 2005. Letzter Abruf am 23.04.2015. URL: <http://dx.doi.org/10.1002/spe.647>.
- [CF14] M. Campanella and F. Farina. The federica infrastructure and experience. *The International Journal of Computer and Telecommunications Networking*, 2014.
- [CPGD10] Emmanuel Conchon, Tanguy Pérennou, Johan Garcia, and Michel Diaz. W-nine: A two-stage emulation platform for mobile and wireless systems. *EURASIP J. Wirel. Commun. Netw.*, 2010:3:1–3:12, April 2010. Letzter Abruf am 23.04.2015. URL: <http://dx.doi.org/10.1155/2010/149075>.
- [CR10] Marta Carbone and Luigi Rizzo. Dummynet Revisited. *ACM SIGCOMM Computer Communication Review*, 40(2):12–20, April 2010. Letzter Abruf am 23.04.2015. URL: <http://doi.acm.org/10.1145/1764873.1764876>.
- [CR11] Marta Carbone and Luigi Rizzo. An emulation tool for planetlab. *Comput. Commun.*, 34(16):1980–1990, October 2011. Letzter Abruf am 23.04.2015. URL: <http://dx.doi.org/10.1016/j.comcom.2011.06.004>.
- [GHB08] Johan Garcia, Per Hurtig, and Anna Brunstrom. KauNet: A Versatile and Flexible Emulation System. In *Proceedings of the 5th Swedish National Computer Networking Workshop (SNCNW 2008)*, Karlskrona, Sweden, April 2008.
- [HRFR06] Thomas R. Henderson, Sumit Roy, Sally Floyd, and George F. Riley. Ns-3 project goals. In *Proceeding from the 2006 Workshop on Ns-2: The IP Network Simulator*, WNS2 '06, New York, NY, USA, 2006. ACM. Letzter Abruf am 23.04.2015. doi: 10.1145/1190455.1190468.
- [KN11] H.K. Kalita and M.K. Nambiar. Designing WANem : A Wide Area Network emulator tool. In *Communication Systems and Networks (COMSNETS), 2011 Third International Conference on*, pages 1–4, January 2011. Letzter Abruf am 23.04.2015. doi:10.1109/COMSNETS.2011.5716495.
- [PR06] Larry Peterson and Timothy Roscoe. The design principles of planetlab. *SIGOPS Oper. Syst. Rev.*, 40(1):11–16, January 2006. Letzter Abruf am 23.04.2015. doi: 10.1145/1113361.1113367.

- [SBJN13] Luc Sarzyniec, Tomasz Buchert, Emmanuel Jeanvoine, and Lucas Nussbaum. Design and Evaluation of a Virtual Experimental Environment for Distributed Systems. In *PDP2013 - 21st Euromicro International Conference on Parallel, Distributed and Network-Based Processing*, pages 172 – 179, Belfast, Royaume-Uni, February 2013. IEEE. Letzter Abruf am 23.04.2015. doi:10.1109/PDP.2013.32.
- [SH05] Stephen Hemminger. Network Emulation with NetEm. In *linux.conf.au*, Canberra, Australia, April 2005. Letzter Abruf am 23.04.2015. URL: http://devresources.linux-foundation.org/shemminger/LCA2005_netem.pdf.
- [SRZ⁺14] Dennis Schwerdel, Bernd Reuther, Thomas Zinner, Paul Müller, and Phuoc Tran-Gia. Future internet research and experimentation: The g-lab approach. *The International Journal of Computer and Telecommunications Networking*, 2014. Letzter Abruf am 23.04.2015. URL: <http://dx.doi.org/10.1016/j.bjp.2013.12.023>.
- [VH08] András Varga and Rudolf Hornig. An overview of the omnet++ simulation environment. In *Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems & Workshops*, Simutools '08, pages 60:1–60:10, ICST, Brussels, Belgium, Belgium, 2008. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering). Letzter Abruf am 23.04.2015. URL: <http://dl.acm.org/citation.cfm?id=1416222.1416290>.