

I. Schieferdecker, A. Rennoch, Mang Li und B. Stepien

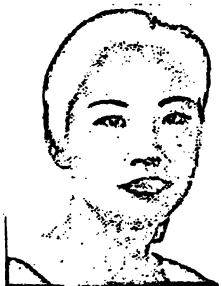
Eine Leistungstestmethodik und ihre Anwendung auf ATM Protokolle



Dr.-Ing. Ina Schieferdecker hat mathematische Informatik an der Humboldt Universität zu Berlin studiert und 1994 an der TU Berlin promoviert. Seit 1993 ist sie als Wissenschaftlerin am GMD Institut FOKUS in Berlin beschäftigt und Lehrbeauftragte an der TU Berlin. Ihre Arbeitsschwerpunkte beinhalten Test Methodiken, Performance Analysen und Formale Sprachen.



Axel Rennoch studierte Mathematik an der Freien Universität Berlin. In unterschiedlichen Projekten arbeitete er seit vielen Jahren auf dem Gebiet formaler Beschreibungstechniken zunächst am Hahn-Meitner-Institut und jetzt als wissenschaftlicher Mitarbeiter bei GMD FOKUS in Berlin.



Mang Li studierte Elektrotechnik am Institut für Nachrichtentechnik der TU Berlin. Sie ist seit 1996 bei der GMD FOKUS beschäftigt. Ihre Forschungsinteressen betreffen insbesondere Konformitäts- und Leistungstests.



Bernard Stepien studierte an der Universität Montpellier (Frankreich). Seit 1975 ist er als freier Mitarbeiter für Computer Anwendungen u.a. für Bell Canada, Nortel und die kanadische Regierung als auch als Gastwissenschaftler bei GMD FOKUS tätig. Seine Arbeitsgebiete betreffen formale Beschreibungssprachen und Kommunikationsprotokolle.

ZUSAMMENFASSUNG

Die Vergleichbarkeit von Zeit- und Kapazitätsmessungen bei Datenübertragungen erfordert eine eindeutige Beschreibung der durchzuführenden Experimente. In diesem Artikel stellen wir die Grundelemente für die Spezifikation solcher Leistungstests vor. Unser Ansatz basiert auf einer Erweiterung der bislang für Kommunikationsprotokolle einzigen standardisierten Testbeschreibungssprache TTCN. Wir zeigen, wie diese Erweiterung für die Definition und Implementierung verschiedener Leistungstests am AAL5 Protokoll eingesetzt wurde und diskutieren die beobachteten Meßergebnisse.

MOTIVATION

Die heutige Kommunikationstechnik zeichnet sich besonders durch große Datenübertragungsmengen, schnelle Übertragungsraten und hohe Ansprüche an die Zuverlässigkeit aus. Die Überprüfung dieser Qualitätsmerkmale sind Gegenstand von Leistungstests. Das zu testende System (SUT) kann dabei ein Kommunikationsprotokoll, ein Übertragungsdienst oder eine Anwendung (z.B. Multimedia Konferenz) sein, d.h. in der Regel aus vielen Programm- und Netzwerkkomponenten bestehen. Zu berücksichtigen sind dabei auch die möglichen Unterschiede in der Netzwerkbelastung.

Leistungstests sollen wie Funktionstests eindeutig spezifiziert, ausgeführt und vergleichend bewertet werden können. Bei den Funktionstests sind es insbesondere die Protokoll-Konformitätstests, für die im Rahmen der Standardisierung von ISO und ITU-T eine umfassende Methodik (CTMF [1]) erarbeitet wurde. Auf dieser Basis werden Konformitätstests in der Telekommunikationsindustrie weitgehend vorgenommen. Neben den Verfahrenswesen von Testlaboren und ihrer Kunden wurden in CTMF vor allem die technischen Aspekte von Testarchitekturen und Testbeschreibungen festgelegt.

Die grundlegenden Ideen von CTMF, d.h. die Konzepte, Architekturen und Terminologie, lassen sich auf Leistungstests übertragen: Das Konzept der Beobachtungspunkte (PCOs) kann zu Meßpunkten weiterentwickelt werden. Die hohe Flexibilität der Multi-Party Testmethode gestattet z.B. die Einbeziehung von Verkehrsgeneratoren oder Meßwerkzeugen.

SPEZIFIKATION VON LEISTUNGSTESTS

Die in CTMF genormte Testbeschreibungssprache TTCN (Teil 3 von [1]) umfaßt zunächst den Aufbau (Konfiguration) und den Informationsaustausch (Datenformate und Ablaufsteuerung) der Tests. Sie ist kontextbezogen und aus Gründen der Portabilität abstrakt, d.h. die Tests müssen in eine ausführbare Programmiersprache übersetzt werden. In bezug auf Leistungstests fehlt TTCN als auch den TTCN basierten Werkzeugen die Mächtigkeit für Zeit- und Kapazitätsmessungen.

Bisher gibt es zwei Vorschläge für Erweiterungen bei TTCN ([5][6]). In [6] wird insbesondere vorgeschlagen, die Ausführung von Testaktivitäten mit einem Zeitfenster zu koppeln. Unser Ansatz geht in erster Linie von der Beschreibung der erforderlichen Messungen aus. Seine tragenden Elementen zur Definition von Leistungstests betreffen:

- Eigenschaften und Wege von Datenströmen (Meß- und Hintergrundströme)
- Ort und Merkmale von Messungen
- Steuerung von Lastgeneratoren und Meßwerkzeugen

- Beeinflussung und Beurteilung von Testabläufen aufgrund von Meßergebnissen

Um eine kompakte und flexible Definition dieser Aspekte zu ermöglichen, haben wir - in Anlehnung an die Struktur von TTCN - die jeweiligen Merkmale in verschiedenen kombinierbaren Tabellen aufgegliedert. Die Deklaration der Datenströme wird zurückgeführt auf eine Anzahl vorher zu parameterisierender Datenstrommodelle. Diese Modelle beschreiben die spezifische Verteilung eines Datenaufkommens, z.B. konstante Datenströme (CBR) oder MMPP (Markov-modulated Poisson processes) Datenströme.

Für jedes der Modelle sind generische Tabellen erforderlich, die die kennzeichnenden Parameterwerte des Datenstroms festlegen (Datenrate, Nutzdaten etc.). Neben diesen Modellen sind Angaben zu machen über die PCOs an denen der Datenstrom ausgesendet bzw. hingelenkt werden soll (damit werden auch Multipoint-Verbindungen deklariert). Darüber hinaus müssen sog. Koordinierungsorte (CP) benannt werden an denen Lastgeneratoren / Meßwerkzeuge steuerbar sind.

Messungen sind die Grundlage für Leistungsbewertungen und die Steuerung der Testabläufe. Wir definieren sie zunächst äußerst elementar, d.h. frei von irgendeiner Auswertung oder statistischen Berechnung. Je nachdem, ob es sich z.B. um Zählungen oder Zeitmessungen handelt bezieht sich jede Messung auf mindestens eine, in der Regel aber auf zwei Ereignisse in einem Testablauf. Die mit einer Messung referenzierten Testereignisse sind durch Beobachtungspunkt (PCO), Ereignisname und Ereignisbedingung (TTCN constraint) deklariert (s. Tabelle 1). Darüber hinaus ist eine Angabe zum Typ der Messung und die Maßeinheit erforderlich. Hier sollen zunächst nur die Standard-Metriken Zähler, Zeitverzögerung, Jitter, Frequenz und Durchsatz mit vordefinierter Semantik verwendet werden. Ein Beispiel sei LILO_DELAY, also die Zeit zwischen dem letzten gesendeten und dem letzten empfangenen Bit einer Datenübertragung.

len, z.B. „Mittelwert von n Messungen“ oder „Maximum einer Meßreihe innerhalb der Zeit t“. Diese Charakteristiken unterliegen einer ständigen Aktualisierung und sind wie Programmvariablen zu betrachten. Vorgesehen ist insbesondere eine Verwendung in sog. Leistungsbedingungen, die wie die o.g. Ereignisbedingungen in den Testsequenzen zur Anwendung kommen. Leistungsbedingungen entstehen durch logische Verknüpfungen aus Leistungs-Charakteristiken, die Wertebeschränkungen unterliegen.

AAL5-LEISTUNGSTESTS

Als eines der ersten Anwendungen für Leistungstests nach dem beschriebenen Ansatz wählen wir das ATM Adaptation Layer Type 5 (AAL5) Protocol [2]. Im Mittelpunkt steht hierbei die anwendungsunabhängige Dienstschnittstelle des Common Part Convergence Sublayer (CP). Dieser Dienst ermöglicht eine Datenübertragung von Paketen bis zu einer Länge von 65535 Bytes.

Vom ATM Forum haben wir folgende Leistungstests für AAL5 übernommen: Throughput, Frame latency und Maximum frame burst size [4]. Der Dienst basiert auf den AAL5 Dienstprimitiven CPCS-Unitdata-invoke und CPCS-Unitdata-signal, die gleichzeitig den Ansatzpunkt für die Ableitung von Leistungsmessungen und -charakteristiken darstellen. Für den Test des Ende-zu-Ende-Dienstes werden an den ATM-Endsystemen getrennte Testerkomponenten (FT) erforderlich. Sie übernehmen die Beobachtung an den entsprechenden PCOs, ihre Steuerung erfolgt mittels Steuerungspunkten (MCP) von einer zentralen Testkomponente (MTC) aus. Die zur Anwendung gekommene sog. Verteilte Testmethode ist illustriert in Abb. 1. Sie ist vergleichbar mit der in [7] dargestellten „QoS testing architecture“. Mögliche Hintergrundlastströme werden durch separate Testerkomponenten (BT) an eigenen Beobachtungspunkten (PCO_B) erzeugt oder empfangen. Die jetzt implementierte AAL5 Leistungstest-Suite geht z. Zt. von einer zu testenden 1-zu-1 Verbindung aus.

Measurement Declaration						
Name	Metric	event 1	constraint 1	event 2	constraint 2	Comments
frameLatency	LILO_DELAY	PCO_F1a CPCS_UNITDATA_inv	Frame_Vs2FL(seq Num)	PCO_F1b CPCS_UNITDATA_sig	Frame_Vr2FL	Frame latency of marked frames.

Tabelle 1: Definition einer Messung

Die Steuerung von Messungen und Lastgeneratoren wird Teil der Testsequenz (Behaviour description), da sie bereits sämtliche Sende- und Empfangereignisse sowie die Timerverwaltung ordnet. Eine Zuordnung von Messungen mit ihren zugrundegelegten Ereignissen wurde bereits bei der Deklaration der Messungen vorgestellt. Zur vollen Kontrolle der Meßwerkzeuge ist darüberhinaus eine (De-) Aktivierung der Messung nötig, d.h. eine Messung wird - ähnlich wie bei Timern - gestartet und beendet (cancel). Erst dadurch ist ein zeitlicher Bezug für Zählungen (etc.) möglich. Eine weitergehende Steuerung der Meß- und Lastwerkzeuge findet in Abhängigkeit von den Spezifika der Tools durch Kommunikation an CPs statt. Der Verlauf von Testsequenzen wird im wesentlichen durch die in einem Zustand möglichen Ereignisse entschieden. Dabei sind die Ereignisse stets an Bedingungen geknüpft, die sich in Funktionstests auf empfangene Parameter- oder Variablenwerte beziehen. Bei Leistungstests ist es wünschenswert, auch auf Messungen zu reagieren. Dazu führen wir sog. Leistungs-Charakteristiken ein, die statistische Berechnungen auf der Grundlage der vorgenommenen Messungen bereitstellen.

Infolge der gewählten Testkonfiguration ist die Testsequenz auf drei selbstständige Prozesse aufzuteilen, d.h. die Testspezifikation muß auf Sprachelemente für Parallelität (Concurrent TTCN) zurückgreifen können. Der Prozeß der koordinierenden MTC (Test

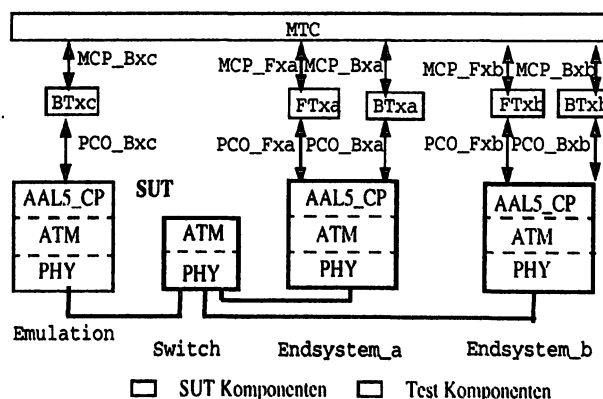


Abb. 1: Test Konfiguration

case behaviour) ist insbesondere für den Start und die Kontrolle zweier untergeordneter Prozesse (Test steps) verantwortlich. Von dieser Stelle aus wird auch der Beginn und das Ende der Messung ein- bzw. ausgeschaltet. Ein direkter Zugriff auf PCOs besteht jedoch nur für die beiden Test Steps, in denen auch die Semantik der Messung definiert wird. Jeweils ein Test Step enthält die Spezifikation der Sende- bzw. Empfangsseite des AAL5 PDU-Stroms.

Im Falle der Frame Latency Messung erfolgt auf der Senderseite in einem festzulegenden zeitlichen Abstand das Initiieren der CPCS_Unitdata_inv Primitive. Durch Angabe eines Zählerwertes wird es möglich, die Messung in Abständen, d.h. lediglich für „markierte“ AAL5 PDUs vorzunehmen. Diese gekennzeichneten PDUs können an der Empfängerseite von den anderen PDUs unterschieden werden. Die Definition der Frame Latency Messung kann sich nun direkt auf die Empfangsbedingung (constraint des Empfangsereignisses) der markierten PDU beziehen (s. Tabelle 2, Zeile 7).

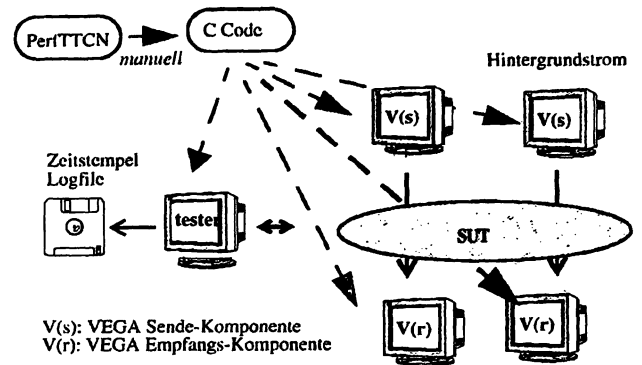


Abb. 2: Aufbau des Meßexperiments

che ist TTCN unter Hinzunahme der von uns vorgeschlagenen Erweiterungen (sog. PerITTCN [5]) benutzt. Für die Implementierung der Tests wurde zunächst die Idee verfolgt, einen (kommer-

Test Step Dynamic Behaviour					
Test Step Name:	FTxbRcvFrame_FL (mcpFxb: CP; pcoFxb: AAL5CP_UT)				
Group:	TestBody/RcvFrame/				
Objective:	Used in Frame latency test.				
Default:					
Comments:					
Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1	Lxb01	mcpFxb?TerminatePTC	TermPTC_sr1		
2		mcpFxb!PTCTerminated	PTCTerm_sr1	R	PTC is terminated.
3		mcpFxb?StartRcvFrame	StartRF_sr1		
4		mcpFxb!RcvFrameStarted	RFStarted_sr1		
5	Lxb02	pcoFxb?CPCS_UNITDATA_sig	Frame_Vr1TP		unmarked frame is rcvd.
6		GOTO Lxb02			
7		pcoFxb?CPCS_UNITDATA_sig	Frame_Vr2FL		marked frame is rcvd.
8		GOTO Lxb02			
9		mcpFxb?StopRcvFrame	StopRF_sr1	(P)	End of experiment.
10		mcpFxb!RcvFrameStopped	RFStopped_sr1		
11		GOTO Lxb01			
12		?OTHERWISE			
13		mcpFxb!Error	Er_sr1	F	
14		?OTHERWISE			
15		mcpFxb!Error	Er_sr1	F	
Detailed Comments:					

Tabelle 2: Spezifikation einer Frame Latency Messung

Throughput-Messungen entsprechen weitgehend denen von Frame Latency, jedoch entfällt hier die Markierung von ausgewählten (markierten) PDUs, da es stets auf die Gesamtrate übertragener PDUs ankommt. Ebenfalls ohne Markierungen kommen auch Tests zur Maximum Frame Burst Size aus. In diesen Testfällen erfolgt immer eine unverzügerte Aussendung einer bestimmten Anzahl von PDUs, die bis zum Eintreten erster Paketverluste gesteigert wird.

IMPLEMENTIERUNG

Die vorgestellten Tests sind auf der Grundlage unseres Ansatzes für Leistungstests vollständig definiert. Als Spezifikationsspra-

zielen) TTCN-Compiler für eine generische C-Code Generierung zu verwenden. Die Berücksichtigung von neuen - nur die Leistungstests (d.h. Messungen) betreffenden - Spezifikationselementen sollte anschließend manuell erfolgen. Es hat sich dabei herausgestellt, daß der automatisch generierte Code nicht nur äußerst umfangreich ist, sondern wegen seiner Komplexität eine unnötig hohe Laufzeit hat und daher die Meßwerte erheblich verfälscht. Als Folge wurde die gesamte Test Suite unter Berücksichtigung der Semantik von PerITTCN manuell als C-Programm implementiert.

Die Erzeugung von Meß- und Hintergrund Datenströmen ist mit Hilfe des verteilten Lastgenerators VEGA [3] vorgenommen.

VEGA ist in der Lage MMPP basierten Datenverkehr über TCP/IP aber auch AAL5/ATM zu generieren. Der technische Aufbau des Experiments ist in Abb. 2 illustriert.

Für die Frame Latency Messungen wurden Experimente mit unterschiedlichen Kombinationen von Paketgrößen (1000, 5000 und 9000 Bytes) und zeitlichen Abständen (10, 50 und 100 ms) zwischen dem Senden von AAL5 PDUs durchgeführt. Die beobachteten Frame Latency Meßwerte sind zusammengefaßt in Abb. 3. Jeder der dargestellten neun Meßwerte bezieht sich auf eine

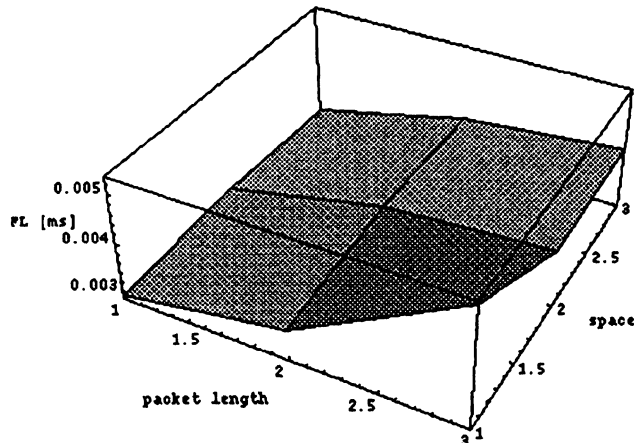


Abb. 3: Frame Latency Meßwerte

Kombination von Paketgröße und Zeitangabe (space). Z.B. wurde der in der Flächenmitte gelegene Meßwert mit den Grundflächenkoordinaten (2,2) bei einer Paketlänge von 5000 Bytes und einem Abstand von 50 ms beobachtet. Dabei ist offensichtlich, daß Frame Latency mit zunehmender Paketlänge steigt, und das um so stärker, je kleiner der zeitliche Abstand der AAL5 PDUs ist.

Analoge Experimente erfolgten zur Ermittlung der Maximum frame burst size (siehe Abb. 4). Während unterschiedliche Paketgrößen eher keinen Einfluß auf die Maximum frame burst size haben, so wird doch hier die Vermutung bestätigt, daß sich ein größerer zeitlicher Abstand der PDUs positiv auf die Maximum frame burst size auswirkt. Die Messung mit einer Paketgröße von 5000 bytes und einem zeitlichem Abstand von 100 ms fällt aus dieser Beobachtung jedoch heraus.

AUSBLICK

Es wurde gezeigt, daß es relativ weniger aber gezielter Erweiterungen bedarf, um aus der Mächtigkeit einer funktionaler Testnotation (z.B. bei Verwendung von TTCN) ein hinreichendes Beschreibungsmittel für Leistungstests abzuleiten. Damit ist die

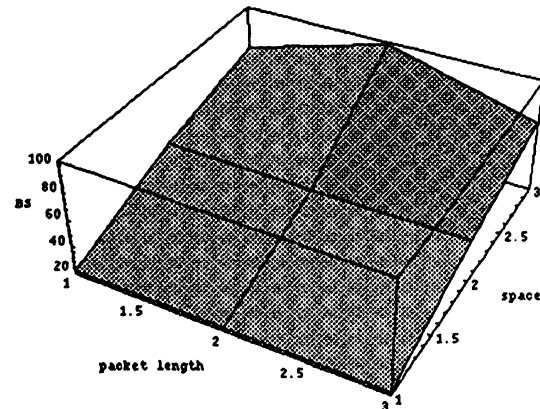


Abb. 4: Maximum Frame Burst Size Meßwerte

Grundlage für den Aufbau einer umfassenden Leistungstest Methodik in Anlehnung an den weit verbreiteten Konformitätstest Standard IS 9646 geschaffen.

Leider ist z. Zt. eine vollständig manuelle Implementierung der abstrakten Tests wegen der beschränkten Einsatzmöglichkeit von Compilern für Testspezifikationen unvermeidlich. Eine erfolgreiche Fortsetzung des Ansatzes ist damit an die Entwicklung von geeigneten Werkzeugen für die (halb-) automatische Implementierung der Leistungstests gekoppelt. Diese Arbeiten werden gegenwärtig bei der GMD FOKUS durchgeführt.

LITERATUR

- [1] ISO/IEC (1996): ISO/IEC 9646 Information Technology - Open Systems Interconnection - Conformance testing methodology and framework. Part 1 - 7.
- [2] ITU-T I.363.5 - B-ISDN ATM Adaptation Layer (AAL) Specification Type 5, August 1996.
- [3] Kanzow, P. (1994): Konzepte für Generatoren zur Erzeugung von Verkehrslasten bei ATM-Netzen. MSc-thesis, Technical University Berlin.
- [4] ATM Forum (1997): Performance Testing Specification (96-0810R4); Proposed modifications to Performance Testing Baseline: Throughput and Latency Metrics (97-0426).
- [5] Schieferdecker, I., Stepien, B. und Rennoch, A. (1997): PerFTTCN - a TTCN language extension for performance testing. 10th Intern. Workshop on Testing of Communicating Systems (IWTC'S'97), Cheju Island, Korea.
- [6] Walter, T., Grabowski, J. (1997): Real-time TTCN for testing real-time and multimedia systems. 10th Intern. Workshop on Testing of Communicating Systems (IWTC'S'97), Cheju Island, Korea.
- [7] Walter, T., Schieferdecker, I., Grabowski, J. (1998): Test Architectures for Distributed Systems - State of the Art and Beyond. 11th Intern. Workshop on Testing of Communicating Systems (IWTC'S'98), Tomsk, Russia.