

TPTR - Ein Transportprotokoll mit einstellbarer Zuverlässigkeit für drahtlose Streaming-Anwendungen¹

Jan Kritzner

Markus Kampmann, Joachim Sachs

Lehrstuhl für Informatik IV
RWTH Aachen
Ahornstrasse 55
52074 Aachen

kritzner@informatik.rwth-aachen.de

Ericsson Research
Ericsson Allee 1
52134 Herzogenrath
markus.kampmann@ericsson.com
joachim.sachs@ericsson.com

Abstract: Dieser Beitrag stellt Elemente des neuen TPTR-Protokolls für drahtlose Streaming-Anwendungen vor, welches mit Prioritäten arbeitet und für die einzelnen Prioritätsklassen eine einstellbare Zuverlässigkeit ermöglicht. Die vorgestellten Konzepte verbessern die Wiedergabe beim Streaming im Vergleich zu vorhandenen Algorithmen und ermöglichen eine kontrollierte Anpassung der Wiedergabequalität bei temporären Linkverlusten.

1 Einleitung

Streaming in drahtlosen Zugangsnetzwerken stellt besondere Anforderungen an das Übertragungsprotokoll, da sich die verfügbare Rate permanent ändern kann und Linkverluste aufgrund von Zellwechseln oder Störungen möglich sind. Die heutzutage in der Regel benutzten Protokolle sind in der Regel nicht zuverlässig und nehmen keine Rücksicht auf den aktuellen Linkstatus, was zu Paketverlusten und reduzierter Wiedergabequalität führt. Oft ist es auch nicht möglich, alle Daten rechtzeitig zur Wiedergabe zu senden. Aufgrund dieser Probleme wurde das neue TPTR-Protokoll (Transport Protocol with Tunable Reliability) entworfen. Obwohl für Streaming in drahtlosen Netzen entwickelt, ist es auch in anderen Szenarien zur Echtzeitdatenübertragung anwendbar.

Abbildung 1 gibt einen Überblick über das neu entwickelte Protokollkonzept. Zunächst erkennt man eine Aufteilung des Transportprotokolls in drei Teilschichten, genannt Application Framing (AF), Windowing, Reliability, Timing and Flow-Control (WRTF) und Congestion Control (CC). Diese Einteilung entspricht älteren Ansätzen [LHB99], allerdings wurde die Funktionsweise der Teilschichten neu festgelegt.

Die AF-Teilschicht nimmt von der Applikation die Datenframes entgegen und unterteilt sie in Pakete. Zusätzlich werden die zugehörigen Prioritätswerte von der Applikation dem WRTF mit Hilfe von *Priorities.Req*-Primitiven übermittelt. Um immer rechtzeitig eine

¹ Diese Arbeit wurde im Rahmen des IPonAir-Projektes vom BMBF unterstützt.

ausreichende Anzahl an Daten zur Verfügung zu haben, hat das Transportprotokoll mit Hilfe von *ClearToSend.Ind* die Möglichkeit, ein weiteres Füllen des Sendepuffers zu veranlassen. Der WRTF-Block bildet die zentrale sendeseitige TPTR-Funktionseinheit. In ihm wird entschieden, ob und wann welches Paket übertragen wird. Ein neu entwickelter Scheduler, der später im Detail vorgestellt wird, arbeitet prioritätsbasiert und mit zusätzlichen Parametern, über die sich die Zuverlässigkeit der einzelnen Prioritätsklassen einstellen lässt; mit Hilfe der *Prioritize.Ind*-Primitive kann eine Neuberechnung der Prioritätswerte durch die Applikation erfolgen. Drittes Element der Transportschicht ist die CC-Teilschicht, welche die Schnittstelle zum unterliegenden Netzwerk bildet und Werte wie verfügbare Bandbreite und Netzwerkverzögerungen ermittelt und mittels *Bandwidth.Ind* und *FTT.Ind* weiterleitet.

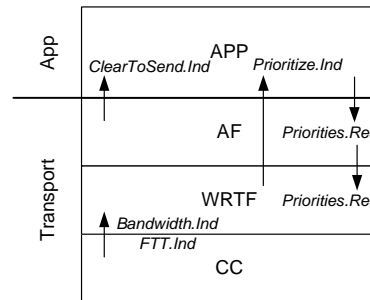


Abbildung 1: Teilschichten des Transportprotokolls und zusätzliche Primitive

Einige Aspekte werden in den folgenden Abschnitten detaillierter beschrieben. Kapitel 2 geht kurz auf die Berechnung von Prioritätswerten ein und Kapitel 3 beschreibt das Transportprotokoll, insbesondere das neu entwickelte Schedulingverfahren. Das letzte Kapitel fasst die Ergebnisse zusammen und gibt einen Ausblick auf weitere Arbeiten.

2 Berechnung von Prioritätswerten

Das unterliegende Netz bietet möglicherweise nicht genug Übertragungskapazität, um alle Daten rechtzeitig zum Abspielen zu übertragen. Aus diesem Grunde werden Prioritäten eingeführt, die dem Beitrag der Daten zur subjektiven Wiedergabequalität entsprechen sollen. Reicht die Übertragungskapazität nicht aus, wird auf die Übertragung weniger wichtiger Daten verzichtet. Audiodaten weist man in der Regel eine höhere Priorität zu als Videodaten, da Tonaussetzer eher als störend wahrgenommen werden als verlorene Bilder. Bei den Videodaten werden zwei verschiedene Arten von Bildern betrachtet: I-Frames sind einzeln dekodierbar, P-Frames enthalten nur die Änderungen im Vergleich zu vorherigen Bildern. Aufgrund dieser Bildabhängigkeiten ergeben sich verschiedene Prioritäten; I-Frames sind dabei wichtiger als P-Frames. Da die Darstellbarkeit von P-Frames von der Übertragung anderer Bildern abhängt, kann der Scheduler Prioritäten von der Applikation dynamisch neu berechnen lassen. Ein Verfahren dazu, welches auf der internen Blockstruktur des H.263-Codecs beruht und den Anteil der dekodierbaren Blöcke zu maximieren versucht, wurde in [KHK04] entwickelt.

3 Das TPTR-Protokoll

Der grundlegende Aufbau des Transportprotokolls ist bereits in der Einführung erläutert worden. Eine Möglichkeit, die hier vorgestellten Konzepte in die Praxis umzusetzen ist

die Weiterentwicklung des RTP-Protokolls, wobei der Aufbau der Datenpakete erhalten bleibt. Der Algorithmus geht davon aus, dass neuere Konzepte, wie z.B. Übertragungswiederholungen für RTP, vorhanden sind. Sein Einsatz ist jedoch mit eingeschränktem Nutzen auch ohne diese Ergänzungen möglich. Die CC-Teilschicht kann aus anderen definierten Elementen bestehen. Beispiele seien TCP-kompatible oder andere, nicht auf Paketverlusten basierende Verfahren [Ba04], auf die hier nicht näher eingegangen werden soll. Sie dient hauptsächlich der Linkschätzung.

Zentrales Element der WRTF-Teilschicht ist ein neu entwickelter Scheduling-Algorithmus. Er basiert zum einen auf dem Earliest Deadline First (EDF) [LL73] Algorithmus, bei dem Daten in ihrer Abspieldreihenfolge übertragen werden. Des Weiteren greift er Konzepte von Priority Based Scheduling (PBS) [Fe99] auf, bei dem der Empfängerpuffer jeweils mit den wichtigsten Daten gefüllt wird, die er fassen kann. Neuere Ansätze wie [KWF03] sind ähnlich, beschränken PBS jedoch auf einen kleineren Teil des Datenstroms.

Der neue Algorithmus wird im Folgenden als Priority Based Scheduling with Tunable Reliability (PBS/TR) bezeichnet. Er ermöglicht die Steuerung der Zuverlässigkeit einzelner Prioritätsklassen durch Zeitlimits. Seine allgemeine Funktionsweise soll anhand von Abbildung 2 erläutert werden: Den Paketen (dargestellt durch Rechtecke) sind jeweils eine Prioritätsklasse (Zahl darunter) und ein Präsentationszeitpunkt (Presentation Timestamp, PTS, Zahl darunter) zugeordnet.

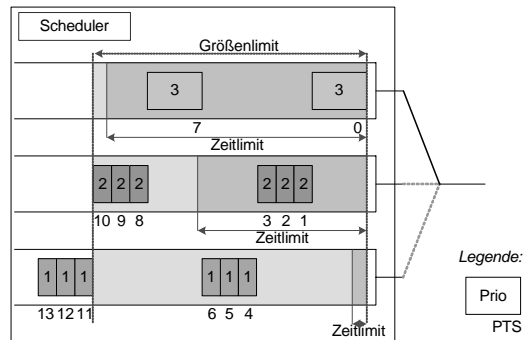


Abbildung 2: Scheduler des neuen TPTR-Protokolls

Übertragungswiederholungen werden unterstützt, indem als fehlerhaft erkannte Pakete wie nicht übertragene Pakete behandelt werden. Jeder Prioritätsklasse wird nun eine separate Warteschlange zugeordnet, und ein gemeinsames Größenlimit beschränkt die Gesamtmenge der Daten in allen Warteschlangen. Zusätzlich wird nun für jede Prioritätsklasse als Parameter ein Zeitlimit definiert. Der zugehörige Zeitrahmen beginnt mit dem gemeinsamen, niedrigsten PTS, der in einer der Warteschlangen vorhanden ist, (im Bild rechts dargestellt) und umfasst von da an, für jede Prioritätsklasse einzeln, einen gewissen Anteil der Multimediadaten.

Der Algorithmus arbeitet nun so, dass zunächst von der höchsten Prioritätsstufe alle Pakete innerhalb des Zeitlimits übertragen werden, dann von der nächsthöheren und so fort. Sind die Zeitlimits aller Prioritätsklassen erfüllt, schaltet der Scheduler in die EDF-Betriebsart um und überträgt – soweit noch Pakete innerhalb des Größenlimits sind – ohne Berücksichtigung von Prioritäten das Paket mit dem niedrigsten PTS.

Der niedrigste vorhandene PTS ändert sich dabei jedoch nicht durch die Übertragung der Pakete, sondern durch das Abspilverhalten der Echtzeitdaten am Client, ist also zeitdefiniert. Übertragene Pakete werden im Sendepuffer gehalten, bis sie auch am

Client durch Abspielen aus dem Puffer gelöscht werden. Der Server baut quasi ein Abbild des Clientbuffers auf, welches jedoch um eine Forward-Trip-Time in der Zeit verschoben ist. Dieses Abbild kann für die Flusskontrolle und Verhinderung von Überläufen am Empfängerpuffer verwendet werden, wenn das Größenlimit am Server der Puffergröße am Client entspricht.

Ist nicht genügend Bandbreite verfügbar, ändert sich der niedrigste PTS im Buffer aufgrund des Abspielverhaltens am Client möglicherweise so, dass nicht übertragene Pakete aus der Warteschlange gelöscht und somit vom Server verworfen werden. Dieser Fall tritt aufgrund der Übertragungsreihenfolge zunächst für Pakete niedriger Priorität ein.

Über die Zeitlimits lässt sich nun insbesondere im Hinblick auf drahtlose Streaming-Umgebungen eine weitgehende Anpassung an die Linkeigenschaften erzielen. So kann das Zeitlimit geteilt durch die Dauer einer Round Trip Time als die mögliche Anzahl an Übertragungswiederholungen interpretiert werden oder – im Falle eines temporären Linkverlusts – als die gewünschte Zeitdauer, für die Daten weiterhin ausgespielt werden können. Über die Zeitlimits wird im Falle eines Linkverlusts die Wiedergabequalität in Stufen herabgesetzt. Zunächst wird der Multimediasstrom für eine kurze Zeit vollständig wiedergegeben, besteht also aus Audiodaten sowie I- und P-Frames. Nach kurzer Zeit werden nur noch die Daten der Prioritätsklassen im Puffer vorhanden sein, deren Zeitlimit groß genug ist. Dies würde zur Wiedergabe der Audiodaten und einer „Diashow“ der I-Frames führen. Noch später – wenn die Unterbrechung das den I-Frames zugehörige Zeitlimit überschreitet – friert das Bild ein und die restlichen Audiodaten können ausgespielt werden. Konventionelle prioritätsbasierte Schedulingverfahren ermöglichen solche Optimierungen nicht. Simulationen in einem gemischt drahtgebundenen und drahtlosen Netzwerk mit Paketverlusten zeigen den Vorteil des neuen Schedulingverfahrens auf. Ein Video wird dabei zunächst mit hinreichender Bandbreite übertragen, danach bewegt sich der Nutzer mit seinem Empfangsgerät in eine neue Zelle, in der nicht mehr genügend Bandbreite zur Verfügung steht. Während des Wechsels zwischen den Funkzellen besteht für wenige Sekunden keine Funkverbindung zwischen Sender und Empfänger; der genaue Simulationsaufbau ist in [Kr04] nachzulesen. Die Zahlen repräsentieren einen beispielhaften Simulationslauf und geben zunächst an, welcher Anteil der Videodaten insgesamt übertragen werden konnte. Bilder werden als „dekodierbar“ bezeichnet, wenn ihre Abhängigkeiten komplett erfüllt sind, d.h. wenn sowohl das vorhergehende I-Frame als auch alle P-Frames seit dem letzten I-Frame am Empfänger vorhanden sind.

<i>Algorithmus</i>	<i>Übertragene Daten</i>	
	<i>Gesamt</i>	<i>Dekodierbar</i>
EDF	62.4%	45.5%
PBS	59.8%	56.9%
PBS/TR	62.8%	60.5%

Abbildung 3: Vergleich verschiedener Scheduling-Verfahren

Die Ergebnisse belegen, dass der neue PBS/TR Algorithmus bezüglich der dekodierbaren Bilder die besten Ergebnisse liefert. Bezüglich der Gesamtmenge der übertragenen Daten verhalten sich EDF und PBS/TR relativ gleich. Die Prioritätsunterstützung des

neu entwickelten Algorithmus führt jedoch dazu, dass die vorhandenen Daten in der Regel auch darstellbar sind. Normales PBS entspricht vom Prinzip her PBS/TR mit für alle Klassen identischen Zeitlimits, die genau der Empfängerpuffergröße entsprechen. Die Ursache der schlechten Werte liegt darin, dass zu früh zu viele Daten hoher Priorität übertragen werden. Für weitere Details sei auf oben zitierte Arbeit verwiesen.

4 Zusammenfassung und Ausblick

Das vorgestellte TPTR-Protokoll ist aufgrund des neuartigen Schedulers sehr gut für Echtzeitdatenübertragung in drahtlosen Zugangsnetzen geeignet. Insbesondere kurze Ausfallzeiten, wie sie durch temporären Linkverlust auftreten, werden durch die geänderte Übertragungsreihenfolge als weniger störend wahrgenommen. Die einzeln einstellbaren Zeitparameter ermöglichen eine kontrollierte, stufenweise Verringerung der Darstellungsqualität je nach Ausfalldauer. Die Unterstützung von Prioritäten spiegelt die Anforderungen von Streaming-Applikationen wieder, da die zu übertragenen Daten eine unterschiedliche Wichtigkeit für die Präsentation haben. Die Paketabhängigkeiten werden durch die Prioritäten widergespiegelt.

Eine Vielzahl weiterer Optimierungen und Ergänzungen ist möglich. Zurzeit wird der Einfluss von Proxies und verschiedener Congestion-Control-Verfahren simulativ bewertet. Weitere mögliche Arbeiten sind im Bereich der dynamischen Optimierung der Zeitparameter des Schedulers in Reaktion auf Messungen von Linkeigenschaften zu sehen. Des Weiteren werden Vorteile erwartet, wenn Algorithmen für die CC-Teilschicht eine eventuelle Abhängigkeit der Übertragungsdauer von der Paketgröße (durch Übertragungswiederholungen auf Schicht 2) berücksichtigen.

Literaturverzeichnis

- [Ba04] Baldo, N.; Horn, U.; Kampmann, M.; Hartung, F.: „RTCP Feedback based Transmission Rate Control for 3G Wireless Multimedia Streaming“. Erscheint in: Proceedings of the PIMRC'04, Barcelona, September 2004.
- [Fe99] Feng, W.C.; Liu, M.; Krishnaswami, B.; Prabhudev, A.: „A priority based technique for the best-effort delivery of stored video“. SPIE/IS&T Multimedia Computing and Networking 1999, 1999; S. 286-300.
- [KHK04] Kritzner, J.; Horn, U.; Kampmann, K.: „Priority Generation for Video Streaming Using Stream Decodability“. Eingereicht zur PV 2004, Irvine, USA, December 2004.
- [Kr04] Kritzner, J.; Horn, U.; Kampmann, M.; Sachs, J.: „Priority Based Packet Scheduling with Tunable Reliability for Wireless Streaming“. Erscheint in: Proceedings of the HSNMC 04, Toulouse, Frankreich, Juli 2004.
- [KWF03] Krasic, C.; Walpole, J.; Feng, W.-C.: “Quality adaptive media streaming by priority drop”. Proceedings of the 13th NOSSDAV, June, 2003
- [LHB99] Li, J.-R.; Ha, S.; Bhargavan, V.: “HPF: A Transport Protocol For Supporting Heterogeneous Racket Flows in the Internet”. Proceeding of the Conference on Computer Communications, März 1999.
- [LL73] Liu, C.L.; Layland, J.W.: “Scheduling algorithms for multiprogramming in a hard-real-time environment”. Journal of the ACM (JACM) 20, 1973, S. 46-61.