
Simulative Untersuchungen zur Wirtschaftlichkeit von DVB-H basierten Datendiensten

Bernhard Hechenleitner, Hilmar Linder

Fachhochschule Salzburg GmbH, Urstein Süd 1, A-5412 Puch/Salzburg, AUSTRIA

ABSTRACT

Motivation: Im Rahmen des Mobile TV Austria Feldversuchs wurden simulative Untersuchungen zur Bestimmung der optimalen Übertragungsparameter durchgeführt, mit dem Ziel die Datenkorrekturkosten für IPDC/DVB-H basierte Datendienste zu minimieren.

Ergebnisse: In Simulationsläufen mit unterschiedlichen Parametereinstellungen wurden jeweils die anfallenden Kosten von Point-to-point bzw. Point-to-multipoint Datenkorrekturen verglichen und bewertet. Um die Datenkorrekturkosten minimal zu halten, wurden aus den Simulationsergebnissen Empfehlungen für die Wahl wichtiger Übertragungsparameter abgeleitet.

Schlussfolgerung: Durch eine gezielte Auswahl und Optimierung von Übertragungsparametern lassen sich die für die Fehlerkorrektur anfallenden Kosten drastisch reduzieren und somit wirtschaftlich tragfähige Datendienste auf Basis von DVB-H implementieren.

Kontakt: bernhard.hechenleitner@fh-salzburg.ac.at

1 EINLEITUNG

Pünktlich zum Start der Fußball-EM 2008 soll die Übertragung von Videos auf DVB-H fähige Endgeräte in den Ballungsräumen rund um die Austragungsorte verfügbar sein. Durch die Verschmelzung von IP und Broadcast-basierten Übertragungstechnologien auf Basis von DVB-H [1] sind neben audiovisuellen Programmen auch effiziente rundfunkbasierte Datendienste möglich. Im Unterschied zur Videoübertragung spielt Zuverlässigkeit bei der Datenübertragung eine ungleich wichtigere Rolle. Im Rahmen des Mobile TV Austria Feldversuchs wurden File Repair Mechanismen für IPDC/DVB-H [2] mittels Simulation verglichen und anhand von Kostenfunktionen auf deren Wirtschaftlichkeit hin überprüft bzw. Implementierungsvorschläge erarbeitet. Ein File Repair Mechanismus bietet Empfängern die Möglichkeit, bei einem Datendownload fehlerhaft empfangene Datenblöcke an einen File Repair Server zurückzumelden. Dieser kann in Abhängigkeit von Anzahl und Verteilung der zu wiederholenden Datenblöcke diese entweder an jeden Empfänger einzeln (Point-to-point, ptp) oder mittels IPDC/DVB-H an alle als Broadcast (Point-to-multipoint, ptm) aussenden.

2 REPERATUR MECHANISMEN UND KANALMODELLE

2.1 File Repair Mechanismus

Der prinzipielle Zweck des File Repair Mechanismus ist die Korrektur von verlorenen oder unwiederbringlich zerstörten Datenfragmenten einer Datenübertragungssession. Zur Erfüllung dieser Aufgabe ist zumindest ein File Repair Server vorgesehen, der die Repair-Anfragen der Empfänger entgegennimmt und die angeforderten Datenfragmente (File Repair) entweder direkt (z.B. über den UMTS Kanal) an die betroffenen Empfänger schickt (Point-to-point, ptp) oder die fehlenden Datenblöcke über das Broadcastnetzwerk nochmals an alle Empfänger sendet (Point-to-multipoint, ptm).

2.1.1 Prinzipieller Ablauf einer File-Repair [3]:

- (1) Der Empfänger identifiziert die fehlenden Datenfragmente.
- (2) Für die Absetzung von File Repair Request Messages ist nach Abschluss der Datenübertragung ein Zeitfenster vorgesehen, innerhalb dessen die betroffenen Empfänger zu zufälligen Zeitpunkten die Anfragen an den File Repair Server senden. Dementsprechend berechnet jeder Empfänger jeweils eine zufällige Zeitdauer (Back-off Time).
- (3) Nach Ablauf der Back-off Time sendet der Empfänger die Repair Request Messages an den File Repair Server.
- (4) Der File Repair Server antwortet mit einer Repair Response Message, welche entweder die angeforderten Daten (Point-to-point Repair) oder die Session Description der Point-to-multipoint Repair Session enthält.

Alle File Repair Request Messages sollen innerhalb einer TCP Session mittels HTTP 1.1 GET Requests [4] transportiert werden. Die fehlenden Datenfragmente werden darin durch Angabe der fehlerhaften Source Block Numbers (SBN) und Encoding Symbol IDs (ESI) angegeben.

Im Falle einer ptp File Repair Session werden die angeforderten Fragmente direkt an die Empfänger als HTTP Responses gesendet. Wird eine ptm File Repair Session durchgeführt, so werden die angeforderten Fragmente wie in einer normalen File Delivery Session via IPDC/DVB-H übertragen.

2.1.2 Kostenberechnungen Ein File Repair Server kann vor Ablauf des Zeitfensters der Fehler-Rückmeldungen die erwarteten Repair Kosten abschätzen. Der Zeitpunkt dieser Schätzung wird durch den Parameter α definiert, welcher den Bruchteil des Rückmeldungs-Zeitfensters angibt, zu welchem die Schätzung durchgeführt wird. Nach Ende des Zeitfensters der Fehler-Rückmeldungen kann die Berechnung der tatsächlichen Kosten für die erste Runde des File Repair erfolgen. Die Formeln zur Berechnung der erwarteten und tatsächlichen Datenkorrekturkosten sind in [3] definiert.

2.2 Fehlermodell eines DVB-H Übertragungskanal

Das in dieser Studie verwendete Fehlermodell für den Übertragungskanal basiert auf dem Four-state Run Length Model aus [5] und [6]. Dieses Markov-Modell erzeugt Ströme von fehlerbehafteten TS Paketen unter Zuhilfenahme der vier Zustände *Good (short)*, *Bad (short)*, *Good (long)* und *Bad (long)*. In den Zuständen *Bad (short)* und *Bad (long)* werden jeweils fehlerhafte TS Pakete generiert, wohingegen in den beiden anderen Zuständen fehlerfreie TS Pakete erzeugt werden. Die Übergangswahrscheinlichkeiten ergeben sich aus den Gleichungen in Tabelle 1, welche die Konstanten aus Tabelle 2 verwenden [5].

Tabelle 1. Berechnung von Übergangswahrscheinlichkeiten

Nach	Good (short)	Bad (short)	Good (long)	Bad (long)
Von				
Good (short)	α_g	$\pi_1=(1-\alpha_g)p_b$		$\pi_2=(1-\alpha_g)(1-p_b)$
Bad (short)	$\pi_5=(1-\alpha_b)p_g$	α_b	$\pi_6=(1-\alpha_b)(1-p_g)$	
Good (long)		$\pi_4=(1-\beta_g)p_b$	β_g	$\pi_3=(1-\beta_g)(1-p_b)$
Bad (long)	$\pi_8=(1-\beta_b)p_g$		$\pi_7=(1-\beta_b)(1-p_g)$	β_b

Tabelle 2. Konstanten

Konstante	Wert
α_g	0.650
β_g	0.999
p_g	0.657
α_b	0.650
β_b	0.982
p_b	0.840

2.3 Multi-Protocol Encapsulation Forward Error Correction (MPE-FEC)

Die in einer Datenübertragungssession zu übertragenden Daten werden zunächst gemäß eines sogenannten Blocking Algorithmus in Source Blocks und Encoding Symbols aufgeteilt. Dieser Blocking Algorithmus hängt von den Parametern der verwendeten Vorwärtsfehlerkorrektur auf Anwendungsebene (AL FEC) ab. Wird kein AL FEC verwendet (wie im Rahmen dieser Studie), so erfolgt die Bildung von Source Blocks, Encoding Symbols und letztendlich FLUTE/UDP/IP Paketen wie folgt: abhängig von der gewünschten Encoding Symbol Size wird das zu übertragende Binärobjekt in Source Blocks und diese wiederum in die einzelnen Symbole der jeweiligen Größe fragmentiert. Jedes Encoding Symbol stellt die Payload eines FLUTE Pakets dar. Ein FLUTE Paket wird mittels UDP transportiert und letztendlich in Form von IP Paketen an den IP Encapsulator der DVB-H Sendeinfrastruktur weitergeleitet.

Bei zusätzlicher Anwendung von Multi-Protocol Encapsulation – Forward Error Correction (MPE-FEC) für eine Vorwärtsfehlerkorrektur werden die IP Pakete vertikal in die Spalten der Application Data Table (ADT) eines MPE-FEC Frames eingepasst und die ADT wenn nötig mit Padding aufgefüllt. Nach Berechnung der Korrekturdaten in Form der Reed-Solomon (RS) Data Table werden die IP Pakete in MPE Sections enkapsuliert und die MPE Sections wiederum auf die diesem Dienst zugewiesenen MPEG-2 Transport Stream (TS) Pakete abgebildet¹.

Ein MPE-FEC Frame kann aus 256, 512, 768 oder 1024 Zeilen bestehen und dementsprechend 47.75 KB, 95.5 KB, 143.25 KB oder 191 KB IP Daten beinhalten. Der verwendete Reed-Solomon Code zur Vorwärtsfehlerkorrektur RS(255,191,32) kann mit Hilfe der Korrekturdaten der RS Data Table beim Empfänger bis zu 64 Byte pro Zeile des MPE-FEC Frames korrigieren.

3 SIMULATIONSMODELL

3.1 Simulationskomponenten

Das für diese Studie konzipierte und implementierte Simulations-Framework verwendet als zentralen Simulationskern den diskreten, ereignisgesteuerten Simulator OMNeT++ [7].

Abbildung 1 zeigt die Komponenten der verwendeten Simulationstopologie bei vier Empfängern. Die Komponente *Sender* stellt den IPDC/DVB-H Sender dar, welcher Datenobjekte an die Empfänger sendet. Die Broadcast-Übertragung wird durch die Komponente *PacketBroadcast* durchgeführt.

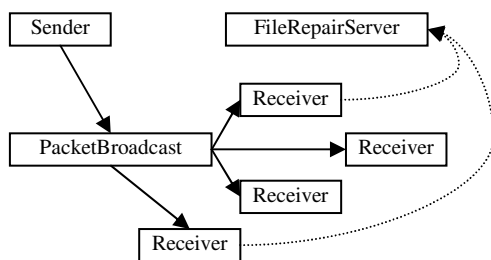


Abbildung 1. Simulationstopologie

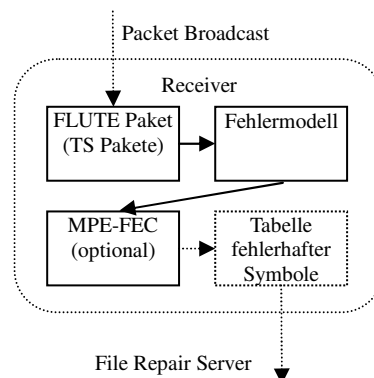


Abbildung 2. Details der Komponente *Receiver*

¹ ohne MPE-FEC wird jeweils ein IP Paket ohne diesen Zwischenschritt in einer MPE Section enkapsuliert

Diese Komponente repliziert ein gesendetes Paket für jeden Empfänger. Die Komponente *FileRepair-Server* nimmt die entsprechenden File Repair Request Messages der Empfänger entgegen und führt die Kostenberechnungen durch. In dieser Abbildung senden exemplarisch zwei Empfänger File Repair Requests an den File Repair Server.

Abbildung 2 zeigt die wichtigsten Teilkomponenten eines Empfängers. Jedes in einem FLUTE Paket enthaltene Symbol wird in Form von TS Paketen transportiert (siehe Abschnitt 2.3). Auf den Strom ankommender TS Pakete wird das Fehlermodell aus Abschnitt 2.2 angewendet. Abhängig von den Zuständen des Fehlermodells führt dies zu keinen oder mehreren fehlerhaften TS Paketen. Wenn MPE-FEC zur Vorwärtsfehlerkorrektur verwendet wird, können einige oder alle dieser fehlerhaften Pakete korrigiert werden, abhängig von der Größe des verwendeten MPE-FEC Frames. Nicht korrigierbare Symbole werden in der Tabelle fehlerhafter Symbole gelistet und können nach Abschluss der Übertragung beim File Repair Server angefordert werden.

3.2 Simulationsparameter

Im Rahmen der vorliegenden Studie wurden jeweils die anfallenden Kosten für die erste Runde einer File Repair Session berechnet. Im Falle einer ptp Repair Session ist im Normalfall keine weitere Runde nötig, bei einem ptm Repair Verfahren können mehrere Runden nötig sein, bis die gesendeten Daten bei allen Empfängern korrekt empfangen wurden.

Es wurden mehrere Simulationsläufe durchgeführt. Tabelle 3 zeigt die Parameter der verschiedenen Läufe. Die Größe der zu übertragenden Datei war bei allen Läufen 4 MB, die Größe der einzelnen Encoding Symbols wurde zwischen 100 Byte und 1400 Byte variiert. Es wurden Simulationen mit und ohne MPE-FEC durchgeführt. Bei aktivierter MPE-FEC war die Anzahl der Zeilen des MPE-FEC Frames entweder 256, 512 oder 1024.

Tabelle 3. Simulationsparameter

Datei [MB]	Symbol [Byte]	MPE-FEC	
4	500	ohne mit, 256 Zeilen mit, 512 Zeilen mit, 1024 Zeilen	
		1400	ohne mit, 512 Zeilen mit, 1024 Zeilen
			100 mit, 512 Zeilen

Tabelle 4. Konstante Parameter

Komponente	Parameter	Wert
Sender	Maximum source block length	100 Symbole
Sender	TS data rate	9.95 Mbps
Receivers	DVB transmission mode	8K
FileRepairServer	Transmission cost of 1 octet ptp	EUR 7.33e-9
FileRepairServer	Transmission cost of 1 octet ptm	EUR 3.07e-7
FileRepairServer	Offset time	50 s
FileRepairServer	Maximum back-off time	600 s
FileRepairServer	α	0.2

Tabelle 4 zeigt weitere Parameter einzelner Simulationskomponenten, die bei allen Läufen konstant gehalten wurden. Die Übertragungskosten von 1 Byte über das ptm Netzwerk errechnete sich aus den Preisangaben eines nationalen DVB-H Sendeanbieters². Ebenso wurden die Übertragungskosten von 1 Byte über das ptp Netzwerk anhand der Kostenangaben eines nationalen Mobilfunkbetreibers festgelegt³. Die Berechnung der erwarteten Kosten erfolgte jeweils nach Ablauf eines Fünftel ($\alpha = 0.2$) des maximalen Rückmeldungs-Zeitfensters (siehe Abschnitt 2.1.2). Der variable Simulationsparameter pro Simulationslauf war jeweils die Anzahl der Empfänger.

² EUR 8300 exkl. MWSt / 100 kbps / Monat für DVB-H Datendienste

³ EUR 22 inkl. MWSt / 3 GB / Monat für GPRS/UMTS Datendienste

4 ERGEBNISSE

Mittels Simulationen wurden Kostenvergleiche zwischen Point-to-point (ptp) und Point-to-multipoint (ptm) File Repair in Abhängigkeit von der Anzahl der Empfänger durchgeführt. Für die Simulation des Fehlermodells des Übertragungskanals wurden aktuelle wissenschaftliche Arbeiten berücksichtigt ([5], [6]). Als Grundlage für den Kostenvergleich dienten die Kostenformeln aus [3]. Für die Berechnung der Datenkorrekturkosten wurde jeweils die erste Runde einer File Repair Session berücksichtigt.

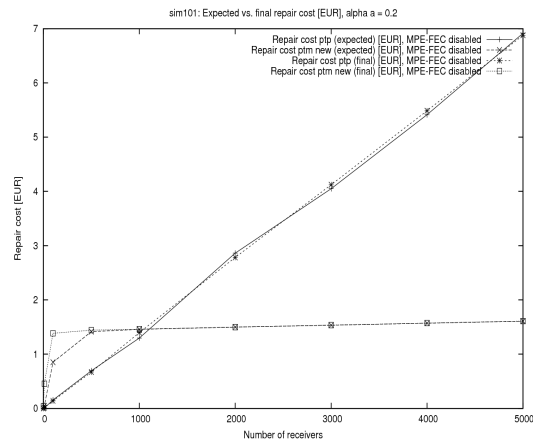


Abbildung 3. Kostenvergleich ptp/ptm

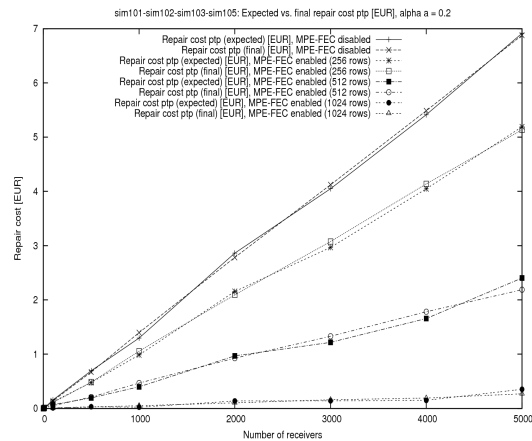


Abbildung 4. Kosten ptp ohne/mit MPE-FEC

Abbildung 3 zeigt jeweils die Gegenüberstellung von erwarteten und tatsächlichen Datenkorrekturkosten für ptp und ptm Repair Sessions bei deaktivierter Vorwärtsfehlerkorrektur mittels MPE-FEC. Während die ptp Kosten in etwa linear mit der Anzahl der Empfänger ansteigen, nähern sich die ptm Kosten relativ rasch einem Maximalwert, welcher der Korrektur der gesamten Datei entspricht. Es ist außerdem ersichtlich, dass die Ergebnisse der Berechnung der erwarteten Kosten bereits bei einem Fünftel des gesamten Rückmeldungs-Zeitfensters ($\alpha = 0.2$) sehr nahe an den tatsächlichen finalen Kosten liegen. Für die Ergebnisse in Abbildung 4 wurden zusätzlich Simulationen mit MPE-FEC der Größen 256, 512 und 1024 Zeilen durchgeführt. Es ist zu erkennen, dass ein größeres MPE-FEC Frame geringere Korrekturkosten bedeutet.

Die Auswertung der Simulationsergebnisse aller durchgeführten Simulationsläufe lässt folgende Schlussfolgerungen zu, welche für die durch die Simulationsparameter gegebenen Rahmenbedingungen (siehe Abschnitt 3.2) gelten:

- (1) Die Korrekturkosten von Point-to-point File Repair Sessions steigen linear mit der Anzahl der Empfänger an.
- (2) Die Korrekturkosten von Point-to-multipoint File Repair Sessions nähern sich mit steigender Anzahl der Empfänger rasch den Kosten einer neuerlichen Übertragung der kompletten Datei an.
- (3) Ohne Verwendung von MPE-FEC sind ab ca. 1000 Empfänger die Korrekturkosten einer Point-to-multipoint File Repair Session deutlich geringer als die Point-to-point Korrekturkosten. Dieser Unterschied kann jedoch durch den Einsatz von MPE-FEC stark reduziert werden.

- (4) Je größer die verwendeten MPE-FEC Frames, desto geringer der Anstieg der Korrekturkosten, insbesondere der Point-to-point Korrekturkosten. Die größte MPE-FEC Frame Größe von 1024 Zeilen liefert die besten Resultate.
- (5) Je nach Anzahl der Empfänger können bei Einsatz von MPE-FEC die Point-to-point Korrekturkosten deutlich unter den Kosten einer Point-to-multipoint Korrektur liegen.
- (6) Die Größe der verwendeten Encoding Symbols hat starke Auswirkungen auf die Korrekturkosten. Dabei liefern weder sehr große noch sehr kleine Encoding Symbol Sizes gute Ergebnisse. Es empfiehlt sich die Verwendung von mittelgroßen Symbolen der Größe 500 Byte.

5 ZUSAMMENFASSUNG

Im Rahmen dieser Studie wurde der File Repair Mechanismus von IPDC/DVB-H simulativ untersucht. Dafür wurden die relevanten Mechanismen der Datenübertragung von Binärobjekten und der Durchführung von File Repairs gemäß [3] in Form eines Simulations-Frameworks implementiert. Um einen möglichst realistischen fehlerbehafteten Übertragungskanal zu simulieren, wurde ein aktuelles Fehlermodell [5] für DVB-H implementiert. In mehreren Simulationsläufen wurden die Auswirkungen der bei DVB-H optionalen Vorwärtsfehlerkorrektur (MPE-FEC) auf die entstehenden Datenkorrekturkosten untersucht. Dazu wurden jeweils die anfallenden Kosten einer Point-to-point bzw. einer Point-to-multipoint Datenkorrektur in Abhängigkeit von der Länge der Encoding Symbols ermittelt. Aufgrund der Simulationsergebnisse konnten Empfehlungen für wichtige Übertragungsparameter abgeleitet werden, mit dem Ziel die Datenkorrekturkosten zu minimieren und dadurch wirtschaftlich tragfähige Datendienste auf Basis von DVB-H implementieren zu können. Ein weiteres Ergebnis der Simulationen ist, dass die Abschätzung der erwarteten Korrekturkosten bereits bei einem Berechnungszeitpunkt von ca. einem Fünftel des gesamten Rückmeldungs-Zeitfensters sehr zuverlässige Ergebnisse liefert. Je größer die verwendeten MPE-FEC Frames, desto ungenauer werden die Kostenprognosen.

6 DANKSAGUNG

Diese Arbeit wurde durch das RTR Projekt „DVB-H Pilotversuch – Mobile TV Austria“ unterstützt.

REFERENZEN

- [1] ETSI (2004): Digital Video Broadcasting (DVB); Transmission System for Handheld Terminals (DVB-H) (ETSI EN 302 304 V1.1.1).
- [2] ETSI (2006): Digital Video Broadcasting (DVB); IP Datacast over DVB-H: Architecture (ETSI TR 102 469 V1.1.1).
- [3] ETSI (2006): Digital Video Broadcasting (DVB); IP Datacast over DVB-H: Content Delivery Protocols (ETSI TS 102 472 V1.2.1).
- [4] Fielding, R. et al. (1999): Hypertext Transfer Protocol -- HTTP/1.1 (IETF RFC 2616). The Internet Society.
- [5] Poikonen, J., Paavola, J. (2006): Error Models for the Transport Stream Packet Channel in the DVB-H Link Layer. In: Proceedings of the IEEE International Conference on Communications 2006 (ICC '06), Volume 4:1861-1866.
- [6] Poikonen, J., Paavola, J. (2005): Comparison of Finite-State Models for Simulating the DVB-H Link Layer Performance. Technical Report, University of Turku.
- [7] Varga, A. (2008): OMNeT++ Community Site. <http://www.omnetpp.org/>