

Netzwerkeffizienz stabiler Overlay-Streaming-Topologien

Network Efficiency of Stable Overlay Streaming Topologies

Dipl.-Inf. Jens Wildhagen: Fachgebiet Integrierte Hard- und Softwaresysteme, Technische Universität Ilmenau, 98693 Ilmenau, Germany

E-Mail: jens.wildhagen@tu-ilmenau.de

Jens Wildhagen studierte Informatik mit Nebenfach Mathematik an der Technischen Universität Ilmenau. Er ist seit 2006 Mitarbeiter des Fachgebietes Integrierte Hard- und Softwaresysteme (Prof. A. Mitschele-Thiel). Seine Forschungsinteressen liegen in den Bereichen der dezentralen verteilten Systeme („Peer-to-Peer“), dem Netzwerkmanagement und der Netzwerkanalyse.

Dipl.-Inf. Thorsten Strufe: Fachgebiet Telematik/Rechnernetze, Technische Universität Ilmenau, Langewiesener Str., 37, 98693 Ilmenau, Germany

Tel: +49-6377-694552, Fax: +49-3677-694540, E-Mail: thorsten.strufe@tu-ilmenau.de

Dipl.-Inf. Thorsten Strufe studierte an der TU Ilmenau Informatik. Seit 2001 ist er Mitarbeiter des Fachgebietes Telematik/Rechnernetze (Prof. D. Reschke bis 2005 und Prof. G. Schäfer seitdem). Seine Forschungsinteressen liegen in den Bereichen der dezentralen verteilten Systeme („Peer-to-Peer“), dem Multimedia-Streaming und der Netzwerkanalyse. Dabei liegt der Fokus vor allem auf der Konstruktion netzwerk-effizienter und stabiler Overlay-Topologien.

Prof. Dr.-Ing. Günter Schäfer: Fachgebiet Telematik/Rechnernetze, Technische Universität Ilmenau, 98693 Ilmenau, Germany

E-Mail: guenter.schaefer@tu-ilmenau.de

Nach seinem Studium der Informatik an der Universität Karlsruhe (TH) arbeitete Prof. Dr.-Ing. Günter Schäfer dort als wissenschaftlicher Mitarbeiter am Institut Telematik und promovierte 1998. 1999 – 2000 war er an der Ecole Nationale Supérieure des Telecommunications (Paris) und von 2000 – 2005 wissenschaftlicher Mitarbeiter des TKN der TU-Berlin. Seit 2005 leitet Prof. Schäfer das Fachgebiet „Telematik/Rechnernetze“ an der Technischen Universität Ilmenau.

Keywords: C.2.4 [Distributed Systems], Multimedia, Streaming, P2P, Netzwerkeffizienz, Stability, Sabotage-Resistency

Schlagworte: C.2.4 [Distributed Systems], Multimedia, Streaming, P2P, Netzwerkeffizienz, Stabilität, Sabotage-Resistenz

MS-ID:

thorsten.strufe@tu-ilmenau.de

9. Juli 2007

Heft: / ()

In order to distribute multimedia content, in peer-to-peer based approaches, all participants choose other peers as neighbours, thus spanning an overlay network of logical nodes and connecting edges. In building such an overlay for live multimedia streaming, two important properties of the evolving topologies have to be considered: The *efficiency* regarding the transmission cost in the network and the *stability* towards random node failures and deliberate attacks.

The majority of existing approaches focuses on the construction of efficient overlays and only a few aim to achieve resilience towards the failure of nodes. Furthermore, no approaches exist which propose a satisfying solution to the creation of both efficient and stable topologies. In order to close this gap, this article presents a new approach for an adjustable construction of both efficient and stable overlay streaming topologies. For this purpose, the properties of stability and efficiency are formalized as a set of cost metrics and a distributed algorithm for dynamic construction of such topologies is introduced, which locally minimizes the total cost. A simulation study shows that this approach allows for a good trade-off between the construction of efficient and stable topologies.

Zusammenfassung

Zur Verteilung von Multimediadaten werden mit Peer-to-Peer-Ansätzen Overlay-Netze konstruiert, welche die Teilnehmer logisch durch Kanten verbinden und so eine abstrakte Nachbarschaft definieren. Bei der Konstruktion solcher Overlays für multimediale Live-Streaming-Anwendungen sind zwei Eigenschaften der Topologien von besonderer Bedeutung: die *Netzwerkeffizienz* bei der Paketverteilung und die *Stabilität* sowohl gegenüber zufälligen Knotenausfällen als auch vorsätzlichen Sabotageangriffen.

In bisherigen Arbeiten zur Steigerung der Stabilitätseigenschaften entstehender Topologien finden Effizienzeigenschaften zumeist keine Berücksichtigung. Um diese Lücke zu schließen wird in dem vorliegenden Artikel ein Verfahren vorgestellt, das es ermöglicht, einen Kompromiss zwischen der Effizienz und der Stabilität konstruierter Topologien zu finden. Hierzu werden zunächst Stabilitäts- und Effizienzeigenschaften in Form von Kostenmetriken operationalisiert und darauf aufbauend ein verteilter Algorithmus zur dynamischen Topologieoptimierung vorgestellt, der eine Gesamtkostenfunktion optimiert, die durch eine parametrisierbare, gewichtete Kombination der Einzelmetriken definiert ist. Mit Hilfe einer Simulationsstudie wird gezeigt, dass auf diese Weise gute Kompromisse zwischen Effizienz und Stabilität bei der Topologiekonstruktion gefunden werden können.

1 Einleitung

In den vergangenen Jahren wurde für den Internet-basierten Transport von Multimedia-Inhalten hin zu großen Benutzerpopulationen der Einsatz des sogenannten “*Application Level Multicast*”-Ansatzes (ALM) vorgeschlagen [7]. Der prinzipielle Vorteil dieses Ansatzes ist, dass die den Datenstrom empfangenden Systeme diesen für andere Systeme replizieren und somit das “Angebot” an potentiellen Quellen für einen bestimmten Datenstrom automatisch mit der Nachfrage nach diesem Datenstrom steigt. Der Ansatz weist daher theoretisch eine beliebige Skalierbarkeit in der Anzahl der Empfänger auf.

Bei der Realisierung eines ALM-basierten Verteildienstes für die Live-Übertragung multimedialer Daten (“Live-Streaming”) sind neben den üblicherweise an Übertragungsdienste für Multimedia-Daten gestellten Dienstgüteeigenschaften nach einer möglichst geringen *Ende-zu-Ende-Verzögerung (Delay)* und *Schwankung dieser Verzögerung (Jitter)* auch Anforderungen in Bezug auf die Effizienz der Verteiltopologie zu beachten – letztere charakterisiert durch das *Verhältnis der Pfadlängen in der Topologie im Vergleich zum kürzesten Pfad (“Path Stretch”)* und die *Anzahl von Kopien identischer Pakete auf einzelnen Teilstrecken (“Link Stress”)* [15]. Von ebenso großer Bedeutung für einen kommerziellen Einsatz dieses Ansatzes ist jedoch die *Gewährleistung einer hohen Verfügbarkeit* des Verteildienstes bei zufälligen Störungen sowie bei vorsätzlichen Sabotageangriffen.

2 Grundlagen

In der Vergangenheit wurden mit unterschiedlichen Methoden be-

reits effiziente oder stabile Overlays konstruiert. Die bisherigen Verfahren zur Stabilisierung und Sicherung der Qualität übertragener Daten basieren hierbei in der Regel auf dem Hinzufügen von Redundanzen oder auf der Minimierung der Auswirkung einzelner ausfallender Teilnehmer auf das Gesamtsystem. Die erstgenannte Strategie nutzt die natürliche Robustheit multimedialer Daten gegen eine geringe Paketverlustrate und zielt darauf ab, die Auswirkungen, die einzelne ausfallende Knoten auf das restliche System haben, zu minimieren. Zu diesem Zweck wird zum einen versucht, durch die Konstruktion flacher Bäume die Anzahl der Vorgänger so gering wie möglich zu halten [2]. Um zum anderen weniger abhängig von einzelnen Vorgängern zu sein, versuchen weitere Ansätze den Datenstrom über möglichst unterschiedliche Pfade zu beziehen und dadurch den Knotenzusammenhang des Overlays zu erhöhen [5, 13]. Beide Verfahren verhelfen potentiellen Angreifern jedoch zu dem Wissen über die Wichtigkeit von Knoten und versetzen sie so in die Lage besonders zentrale Knoten als Ziele auszuwählen. Insgesamt richten sich die vorgeschlagenen Ansätze damit lediglich gegen zufällige Ausfälle und nicht gegen vorsätzliche Angriffe.

Aufgrund der zeitlichen Anforderungen an den Transport von Multimediadaten und der Vermeidung von unnötigen Übertragungsvorgängen im Netzinneren (bezogen auf die Übertragungsvorgänge pro Netzwerk-Link) ist es für ALM weiterhin besonders wichtig, dass sich die Strukturen der konstruierten Topologien den Strukturen der unterliegenden Transportnetze anpassen. Ineffiziente Topologien mit einer Vielzahl von Verbindungen zwischen weit entfernten Knoten führen nicht nur zu

erhöhten Ende-zu-Ende-Verzögerungen und verstärktem Schwanken dieser Verzögerung (Jitter), sondern bewirken zugleich eine stärkere, unnötige Belastung des Transportnetzes. Die Konstruktion effizienter Topologien zielt daher darauf ab, Pakete vorrangig über kurze Ende-zu-Ende-Verbindungen zu übertragen und lange Distanzen im Netz zu vermeiden.

Um möglichst effiziente Topologien konstruieren zu können, folgt die Nachbarwahl bestehender Ansätze grundsätzlich einer der folgenden drei Strategien: Entweder wird mit Hilfe eines eigenen Signalisierungs-Overlays zunächst die eigene Position bestimmt und die Verbindung zu einem der Knoten in der eigenen Region aufgebaut [7, 17], oder der Knoten tritt dem Streaming-Overlay direkt bei und wird in diesem durch den lokalen Tausch mit Nachbarn sukzessive an eine kostengünstige Position gebracht [11, 12]. In hybriden Verfahren sucht jeder Knoten zuerst einen nahe liegenden Nachbarn und nach dem Beitritt werden die lokalen Verbindungen optimiert [10]. Für die erste Strategie bedarf es eines Verfahrens, um die Positionen der einzelnen Teilnehmer initial zu ermitteln. Der zuverlässigste Ansatz hierfür ist es, eine Distanzbestimmung aller Knoten untereinander durchzuführen [7]. Er ist mit wachsender Teilnehmerzahl auf Grund seiner hohen Nachrichtenkomplexität jedoch nicht mehr durchzuführen. Eine vereinfachte Lösung des Verfahrens verfolgt lediglich das Ziel, Knoten der näheren eigenen Umgebung im Netzwerk zu identifizieren, diese zu gruppieren und daraufhin Verbindungen präferiert innerhalb der entstehenden Gruppen aufzubauen [14, 1]. Diese Ansätze kommen mit einem signifikant geringeren Nachrichtenaufwand aus, der jedoch zu einer

geringeren Qualität der Lösung führen kann.

Indirekt kann eine Reduktion der überbrückten Distanzen auch dadurch erreicht werden, dass der ALM auf einem Peer-to-Peer-Suchmechanismus, der die Lokaltäten der Knoten berücksichtigt, aufgesetzt wird und alle Pakete durch diesen geroutet werden [5, 6]. Diese Lösungen beziehen jedoch in der Regel Knoten zur Weiterleitung ein, die den empfangenen Datenstrom selbst nicht nutzen, was für diese nicht wünschenswert ist.

In unseren Vorarbeiten haben wir bisher auf die Konstruktion von Topologien fokussiert, die stabil gegen Angriffe sind. In [3] wurde eine Klasse von Topologien für einen Spezialfall vorgestellt und ihre optimale Resistenz gegen Sabotageangriffe bewiesen. Weiterhin wurde in [18] ein Verfahren für die verteilte Konstruktion stabiler Streaming-Topologien vorgeschlagen und bewertet, das jedoch Effizienzkriterien noch nicht mit einbezieht.

Eine gemeinsame Betrachtung der Kriterien Effizienz und Stabilität gegen Sabotageangriffe von Topologien ist daher das Ziel des vorliegenden Artikels.

3 Effizienz und Stabilität

Grundlegend ist ein Overlay ein ungerichteter, schleifenfreier Graph $G = (V, E)$, bestehend aus einer endlichen Knotenmenge $V = \{v_1, \dots, v_n\}$ (den n teilnehmenden Endsystemen) und einer Kantenmenge $E \subseteq \{(u, v) | u, v \in V, u \neq v\}$ (den Verbindungen zwischen den Endsystemen).

Alle Endsysteme sind in der Lage, über das unterliegende Netzwerk paarweise miteinander Verbindungen aufzubauen, so dass

der Graph $G = (V, E)$ im Allgemeinen zusammenhängend ist. Im konkreten Fall wählt jeder Knoten jedoch nur die Teilmenge $N_v \subseteq V$ als *Nachbarn*, wodurch in Overlays nur eine Teilmenge der Kanten vorkommt.

Da sich die Teilnehmer im Internet in unterschiedlichen Entfernungen zueinander befinden, ist zusätzlich die nichtnegative *Distanzfunktion*, die den Kosten der jeweiligen Kante entspricht, definiert: $d : E \rightarrow \mathbb{R}^+$.

Zusätzlich sind die Netzzugänge der Teilnehmer in der Bandbreite beschränkt. Dafür ist die *Knotenkapazität* wie folgt definiert: $c : V \rightarrow \mathbb{R}^+$.

Im Overlay existiert eine Datenquelle $v_s \in V$, welche den Datenstrom mit der Bitrate R_0 erzeugt und als Ursprung den anderen Knoten zur Verfügung stellt. Dieser Datenstrom $\mathcal{S} = \{P_1, \dots, P_n\}$ besteht aus n Datenpaketen, die auf jedem Knoten beliebig dupliziert und weitergeleitet werden können. Mittels Unterteilung des Datenstroms in l aufeinander folgende Sequenzen mit jeweils k Paketen und durch Zusammenfassung jedes i -ten Paketes aus allen Sequenzen kann er in k gleich große Teildatenströme, oder „*Stripes*“, unterteilt werden:

$$\mathcal{S} = \{\{p_1^1, \dots, p_k^1\}, \dots, \{p_1^l, \dots, p_k^l\}\} \\ \text{mit } p_j^i = P_{(i-1) \cdot k + j}.$$

Mit C wird die Knotenkapazität der Quelle bezeichnet: $C = c(v_s)$, die Quelle hat in Konsequenz maximal $C \cdot k$ ausgehende Kanten und die Bandbreite des Netzzugangs der Quelle beträgt damit $C \cdot R_0$.

Im allgemeinen Fall, dass alle Knoten den gesamten Datenstrom erhalten, werden die Pakete des Datenstromes entlang n Spannbäumen T_i mit der Wurzel v_s , über den Graphen verteilt. Bei Vernachlässigung der Knotendynamik gilt in diesem Fall, dass die

jeweils l Spannbäume eines Stripes identisch sind und lediglich k unterschiedliche Spannbäume existieren: $T_1 = (V, E_1), \dots, T_k = (V, E_k)$. Aufgrund der Bandbreitenbeschränkung der einzelnen Knoten gilt darüber hinaus, dass die Summe der Grade in den Spannbäumen T_i für jeden Knoten $v \in V$ höchstens $c(v)$ ist:

$$\sum_{i=1}^k \deg_{T_i}(v) \leq c(v) \quad \forall v \in V.$$

Werden die Distanzen als Kosten berücksichtigt, ergeben sich die Gesamtkosten der Topologie in diesem Fall zu:

$$tc(\mathcal{T}) = \sum_{i=1}^k d(T_i) = \sum_{i=1}^k \sum_{e \in E_i} d(e).$$

Die Sequenz $\mathcal{T} = (T_1, \dots, T_k)$ der Spannbäume wird im Weiteren als Streaming-Topologie \mathcal{T} bezeichnet.

Verlässt ein Knoten v das Overlay, so führt dies bis zur erneuten vollständigen Verbindung der Topologie zu einem Verlust der Pakete des Stripes i bei allen seinen Nachfolgern $\text{suc}_{T_i}(v)$. Die Anzahl der dadurch nicht mehr empfangenen Stripes beschreibt die Funktion $a_{T_i}(\mathcal{A}) := |\text{suc}_{T_i}(\mathcal{A}) \cup \mathcal{A}|$. Die Anzahl der durch eine Ausfallmenge \mathcal{A} in der gesamten Topologie \mathcal{T} verursachten Paketverluste, ergibt sich zu:

$$a_{\mathcal{T}}(\mathcal{A}) := \sum_{i=1}^k a_{T_i}(\mathcal{A}).$$

Als Maß für die Robustheit eines Overlays wird damit die *Angriffsstabilität* der Topologie definiert. Sie beschreibt die Anzahl der Knoten, die aus einem Overlay entfernt werden müssen, um eine Schranke Θ_{drop} insgesamt im System verlorener Pakete zu überschreiten:

Gegeben ist die Topologie \mathcal{T} und eine Paketverlustschranke $\Theta_{Drop} \in (0, 1)$.

Gesucht ist die minimale Ausfallmenge $\mathcal{A} \subseteq \mathcal{T} \setminus \{v_s\}$ von Knoten, so dass gilt:

$$a_{\mathcal{T}}(\mathcal{A}) \geq \Theta_{Drop} \cdot k \cdot n.$$

Die Angriffsstabilität bewertet somit lediglich das Verhalten der Topologien gegenüber optimalen Angriffen, die auf globalem Wissen basieren. Zufällige Knotenausfälle stellen den Sonderfall eines Angriffs dar, bei dem der Angreifer zufällige Knoten als Ziele auswählt. Die Angriffsstabilität gibt damit als Worst-Case-Metrik für Knotenausfälle das Verhalten der Topologien im schlimmsten Fall an.

Link-Stress und Path-Stretch als bekannte Metriken für die Bewertung der Effizienz der konstruierten Topologien sind stark von der Struktur des unterliegenden Transportnetzes abhängig: In großen Infrastrukturnetzen, die aus vielen Netzzwischensystemen und Verbindungen zwischen diesen bestehen, überdecken sich die kürzesten Pfade zwischen unterschiedlichen Endsystemen seltener, als in kleineren Infrastrukturnetzen.

Zudem sind die direkten Netzwerkpfade zwischen der Datenquelle und vielen der Endsysteme größer als in kleinen Infrastrukturnetzen, während die Netzwerkpfade zwischen nahe liegenden Endsystemen (die etwa am gleichen Access-Router angeschlossen sind) gleich bleiben.

Diese Tatsachen führen dazu, dass sowohl der Link-Stress als auch der Path-Stretch der gleichen Overlay-Konstruktionsprozedur in großen Infrastrukturnetzen mit vielen Netzwerklinks geringer als in kleinen Infrastrukturnetzen sind.

Aus diesem Grund bedarf es einer Metrik, mit Hilfe derer exaktere und vergleichbare Aussagen über die Effizienz einer Konstruktionsprozedur getroffen werden können. Daher

schlagen wir als Maß für die Netzwerkeffizienz die Anzahl der gegenüber einer optimalen Topologie zusätzlich auf Punkt-zu-Punkt-Verbindungen übertragenen Netzwerk-Hops vor. Als Optimalitätsmaß wird eine rein theoretische Lösung ohne Berücksichtigung jedweder Bandbreitenbegrenzungen gewählt. Der so theoretisch minimale Wert lässt sich mit globalem Wissen über das gesamte Netzwerk über das Overlay-Modell berechnen: Der effizienz-optimale Verteilbaum, der die geringsten Netzwerkkosten erzeugt, entspricht dem minimalen Spannbaum (MST) über alle Knoten und kann als Referenzwert $tc(MST(G))$ für jedes Overlay in einem zugehörigen Infrastrukturnetzwerk angegeben werden.

Das Maß für die Netzwerkeffizienz einer Topologie ergibt sich zu:

$$hp(G, \mathcal{T}) := \frac{tc(\mathcal{T})}{tc(MST(G))}.$$

4 Ein Ansatz zur gemeinsamen Optimierung

Ein neu zum System hinzukommender Knoten verbindet sich initial mit anderen Knoten in der Topologie, von denen er daraufhin die Datenstropfpakete erhält. Zur Verbesserung der Eigenschaften der so zufällig entstehenden Topologie wird diese daraufhin optimiert. Um eine Skalierbarkeit auf große Gruppen zu ermöglichen und einen einzelnen Single-Point-of-Failure, der zusätzlich jedem Knoten bekannt ist, zu vermeiden, wird die Optimierung, basierend auf lokalem Wissen, verteilt implementiert.

Stabilität und Effizienz der Topologien sollen durch eine lokale Optimierung mit Hilfe einer Gesamtkostenfunktion erreicht wer-

den. Zur Optimierung der Topologie analysiert jeder Knoten seine aktuelle Situation und versucht die auftretenden Kosten zu minimieren. Dabei werden die Kosten aller Kanten zu den Kindern analysiert und einzelne Kinder gegebenenfalls untereinander weitervermittelt. Alle Veränderungen der Topologie sind damit von Vaterknoten initiiert. Um ein Abwägen zwischen den Optimierungszielen zu ermöglichen, werden die Kosten einer Kante in Bezug auf die Effizienz K_d über den Faktor s gewichtet mit den Stabilitätskosten K_s zusammengefasst. Für die Kante des Knotens v zu seinem Kind w im Spannbaum des Stripes i ergeben sich damit die Gesamtkosten zu:

$$K_i(e) = s \cdot K_s(e) + (1 - s) \cdot K_d(e).$$

Zur Kostenminimierung müssen die alternativen Situationen überprüft werden, um den besten alternativen Vater u und dadurch die lokale Veränderung zu ermitteln, welche zur stärksten Kostensenkung führt.

Um eine Stabilität der Topologien gegen Ausfälle zu erreichen, müssen die Abhängigkeiten zwischen den Knoten minimiert werden.

Zunächst ist es für jeden Knoten wichtig, von keinem anderen Knoten große Anteile des Datenstroms zu beziehen, damit dessen Ausfall nicht zu hohem Paketverlust führt. Um dieses Ziel zu erreichen, werden der Datenstrom an der Quelle in Stripes unterteilt und für jeden Knoten Kosten für die Weiterleitung mehrerer Stripes eingeführt.

$$K_1(e) := 1 - \frac{fo_{T_i}(v)}{c(v)}.$$

$fo_{T_i}(v)$ (fanout) beschreibt dabei die Anzahl der ausgehenden Kanten des Knotens v im Spannbaum i und $fo_{\mathcal{T}}(v, u)$ die Anzahl Kanten von Knoten v zu Knoten u in

der Topologie \mathcal{T} . Durch die Minimierung dieser Kosten wird im besten Fall von jedem Knoten nur ein Stripe weitergeleitet.

Global ist es für die Topologie wichtig, dass zum einen Knoten, die im betrachteten Spannbäum Daten weiterleiten können, der Quelle möglichst nahe sind, um die verfügbare Bandbreite zu erhöhen, und zum anderen die verbleibenden Knoten als Blätter der Quelle möglichst weit entfernt sind. Aus diesem Grund erhalten Knoten w , die den Stripe i weiterleiten können, die Kosten $K_2(e) = 0$, alle anderen Knoten erhalten $K_2(e) = 1$, was dazu führt, dass bei Minimierung der Kosten die Spannbäume flach gehalten werden.

Kommt es durch Engpässe bei der Optimierung zu der Situation, dass der Knoten v zu dem Knoten u mehr als eine Kante unterhält, so erhalten alle Kanten zu diesem Knoten durch die Abhängigkeitskosten K_3 die Bewertung:

$$K_3(e) := \frac{\text{fo}_{\mathcal{T}_i}(v, u)}{k}.$$

Außerdem sollten Knoten w mit vielen Nachfolgern von einem Vater v nicht tiefer gehängt werden, um die durchschnittliche Tiefe der Knoten in den Bäumen und damit die Abhängigkeit nicht zu erhöhen. Für die Angriffsstabilität ist es zusätzlich wichtig, dass der Ausfall eines bestimmten Knotens nicht zu deutlich höheren Paketverlusten führt, als der eines beliebigen anderen. Wenn einzelne Knoten mit einer hohen Anzahl von Nachfolgern $a_{\mathcal{T}}(v)$ existieren, so wird ein Angreifer in die Lage versetzt, durch das Ausschalten dieser wenigen Knoten große Teile des Systems vom Dienst zu trennen und leicht großen Schaden anzurichten. Beide Ziele werden durch die Balancierung der Topologie erreicht und es ergibt sich die

Kostenfunktion:

$$K_4 := \frac{\left(\frac{\text{suc}_{\mathcal{T}_i}(v)}{\text{fo}_{\mathcal{T}_i}(v)} - 1\right) - \text{suc}_{\mathcal{T}_i}(w)}{\left(\frac{\text{suc}_{\mathcal{T}_i}(v)}{\text{fo}_{\mathcal{T}_i}(v)} - 1\right)}$$

Zur Untersuchung des Einflusses der einzelnen Kostenfunktionen wurden diese zunächst einzeln bewertet [16]. Dabei wurde ermittelt, dass die Kantenbewertung mit einzelnen Kostenfunktion nicht zu einer zufriedenstellenden Stabilität der Topologie führt. Vielmehr müssen alle Kostenfunktionen gemeinsam berücksichtigt werden, damit die Topologien eine hohe Stabilität erreichen. Aus den vier Kostenarten ergeben sich damit die Stabilitätskosten insgesamt zu:

$$K_s(e) = \sum_{j=1}^4 K_j(e).$$

Zur Konstruktion zusätzlich netzwerkeffizienter Topologien ist es wichtig, dass alle Kanten in den Spannbäumen eine möglichst geringe Distanz $d(e)$ aufweisen, um die Datenstrompakete auf kurzen Wegen zu übertragen. Aus diesem Grund werden alle Distanzen in den Spannbäumen minimiert.

$$K_d := 1 - \frac{d(e_{(v,w)}) + d(e_{(w,u)})}{d(e_{(v,w)}) + d(e_{(v,u)})}.$$

Diese lokale Optimierung führt auch global für die Topologie \mathcal{T} zu einem niedrigen $\text{tc}(\mathcal{T})$, geringem Link-Stress und $\text{hp}(G, \mathcal{T})$.

Da ein Vater nicht alle lokalen Stabilitätsinformationen seiner Kinder und ihrer Nachfolger kennt, wird die Gesamtoptimierung in zwei Schritten durchgeführt. Zunächst wird die Kante ausgewählt, welche die höchsten Gesamtkosten K_i verursacht. In einem zweiten Schritt wird bei der Optimierung der lokalen Situation von den Stabilitätskosten lediglich K_4 berücksichtigt und insgesamt so optimiert, dass der Nutzen $G_i = K_i - (s \cdot K_4 + (1 -$

$s) \cdot K_d)$ maximiert wird. Da durch die Optimierung die Baumhöhe lediglich vergrößert werden kann, was bei einer Senkung von Stress und tc zu einer Erhöhung des Path-Stretch führt, werden als Nutzenschranke Θ_{pass} eingeführt und Optimierungen nur dann ausgeführt, wenn ihre Nutzensteigerung G diese Schwelle überschreitet.

Die Entscheidung über die Weiterleitung eines Knotens kann nicht über eine konstante Schwelle parametrisiert werden, da einerseits vermieden werden muß, dass ein Knoten, der noch gar keine Bandbreite verbraucht hat, sofort alle Knoten, die eine Verbindung zu ihm aufbauen, weiterleitet. Andererseits muß ein Knoten mit ausgelasteter Bandbreite in der Lage sein, eines seiner Kinder weiterzuleiten. Die Schwelle muß folglich durch eine Funktion über die Bandbreite berechnet werden. Sie soll mit steigender anteiliger Bandbreitenutzung eines Vaterknotens $b \in (0, 1)$ stetig von 1 auf 0 fallen, wobei: $b = \frac{\text{deg}(v)}{c(v)}$. Zusätzlich soll mit dem Gewicht t eine Optimierung auf geringeren Stretch oder geringere tc ermöglicht werden. Hierfür wurde die Funktion $\Theta_{pass}(b) = (1 - b^{(2 \cdot t)^3} \cdot (1 - t^3)) + t^3$ gewählt, wobei der Funktionswert von $\Theta_{pass}(1)$ auf $-\infty$ definiert ist. Stellt ein Knoten fest, dass er Bandbreite frei hat, so fordert er von einem seiner Kinder einen beliebigen Nachfolger an. Hierdurch wird erreicht, dass bei freiwerdenden Bandbreiten die Höhe der Topologie gesenkt werden kann.

Mit den so definierten Funktionen für Kosten K und die Kostensenkung G optimiert der vorgeschlagene Algorithmus zur verteilten Topologieoptimierung die lokale Nachbarschaft jedes teilnehmenden Knotens. Die Zeitkomplexität dieser Optimierung ist in jedem Optimierungsschritt

für jeden Knoten quadratisch in der Anzahl seiner maximal $c(v) \cdot k$ Kinderknoten im optimierten Spannbaum. Die Nachrichtenkomplexität des Protokolls liegt in $O(c(v) \cdot k)$. Da C und k im Modell konstante Werte sind ergibt sich für jeden Knoten eine Zeitkomplexität von $O(1)$ und für das Gesamtsystem eine Nachrichtenkomplexität von $O(n)$.

5 Evaluation

Zur Überprüfung der Hypothese, dass mit Hilfe lokaler Optimierungen auf Basis der vorgestellten Kostenfunktionen stabile und effiziente Topologien konstruiert werden können, wurde eine zweiteilige Simulationsstudie durchgeführt.

Zu diesem Zweck wurden zunächst Algorithmus und Signalisierungsprozedur im Simulationswerkzeug OMNeT++ implementiert. Als Backbone kam ein durch den BRITE-Topologiegenerator konstruiertes Netzwerk, welches dem Internet-Modell aus [4] folgend generiert wurde, mit 750 Netzzwischensystemen zum Einsatz. Mit den Zwischensystemen wurden gleichverteilt eine Datenstromquelle und zwischen 50 und 250 Endsysteme verbunden, die einem Nutzermodell nach [8] folgend, dem System beitreten. Natürlich existiert in realen Systemen keine konstante und für alle Teilnehmer gleiche Bandbreite. In den Simulationen wird mit $c(v) = 2$ daher lediglich als Untergrenze angenommen, dass jeder Knoten in der Lage ist die zum Empfang notwendige Bandbreite auch anderen Knoten für die Weiterleitung zur Verfügung zu stellen.

Um zur Erlangung der für die Effizienzoptimierung benötigten Entfernungen eine permanente Distanzmessung zwischen den Knoten zu vermeiden, wurde ein

synthetisches Koordinatensystem basierend auf Vivaldi [9] implementiert. Vivaldi benötigt zur Adaption der Koordinaten einen Signalisierungsaufwand von $O(1)$ Nachrichten pro teilnehmenden Knoten. Damit wird die Skalierbarkeit auf die Anzahl der Teilnehmer des Systems durch den Einsatz von Vivaldi nicht eingeschränkt. Als Metrik kommt in der Implementierung die Anzahl der IP-Hops im Netzwerk zum Einsatz. Die Abweichung zwischen geschätzten und gemessenen Entfernungen zum Ende der Simulationszeit betrug dabei im Maximum ca. 218% und im Schnitt ca. 45%.

Zur Untersuchung wurde der Aufbau der Streaming-Topologien bei gegebenem Nutzermodell mit wachsenden Nutzergruppen simuliert und die entstehenden Topologien danach offline analysiert.

Um die Effizienz der entstandenen Topologien zu bewerten, wurden die zur Übertragung des Streams benötigten Kosten $tc(\mathcal{T})$ und die Güte in Form des $hp(G, \mathcal{T})$ ermittelt.

Durch eine erste Simulationsreihe wurde daraufhin ein günstiges Θ_{pass} ermittelt, welches zur Konstruktion von Topologien mit niedrigem $hp(G, \mathcal{T})$ führte. Insgesamt konnte der minimale Spannbaum bei steigenden Gruppengrößen immer weniger gut angenähert werden und das minimale $hp(G, \mathcal{T})$ stieg stetig mit wachsender Teilnehmerzahl. Diese Tatsache ist nicht überraschend, da die Kapazitäten der Knoten bei der realen Optimierung berücksichtigt werden müssen, während sie für den minimalen Spannbaum keine Rolle spielen.

Im zweiten Schritt sollte der Trade-Off zwischen der Effizienz und der Stabilität untersucht werden, wozu das Gewicht s variiert wurde.

Die Resultate dieser Untersuchung (vgl. Abb. 1) zeigten, daß

die Topologien bei niedrigem Gewicht s erwartungsgemäß effizient wurden. Mit steigendem Gewicht und zunehmender Optimierung der Topologien auf Stabilität stieg diese in allen Simulationen erwartungsgemäß stetig an. Für kleine Topologien ($N=50$) ergaben sich bereits ab einem s von ca. 0.5 minimale Effizienzwerte, während die Effizienz größerer Gruppen bei steigendem s weiter sank. Für alle Gruppengrößen gilt hierbei, dass die Anzahl verschickter Netzwerkpakete immer geringer war, als in einem Client-Server-Szenario.

Bild 1

Insgesamt kann festgestellt werden, dass die Effizienzoptimierung den Erwartungen entspricht und die Topologien bei geringerer Gewichtung der Effizienz höhere Kosten im Netzwerk erzeugen.

Zur Untersuchung der Stabilitätseigenschaften der Topologien wurden sie zusätzlich auf ihre Angriffsstabilität untersucht. Zu diesem Zweck wurde ein auf globalem Wissen basierender Angriff implementiert und die daraus resultierende Paketverlustrate im Gesamtsystem gemessen. Die so ermittelte Angriffsstabilität gibt an, wie viele Knoten mindestens aus der Topologie entfernt werden müssen, um die insgesamt noch ausgelieferten Pakete unter eine vorgegebene Schwelle zu senken. Im Ergebnis zeigt sich somit der maximale durch korrelierten Knotenausfall auftretende Paketverlust im Gesamtsystem für den Zeitraum bis zur Reparatur der Topologie.

Diese zweite Auswertung zeigte, dass sich die Topologien bei einer reinen Stabilitätsoptimierung ($s = 1.0$) einer optimalen Topologie annäherten und die Angriffsstabilität den Ergebnissen aus vorherigen Arbeiten entsprachen [3].

Eine Senkung von s führte wie erwartet zu einer Verringerung der Stabilität der Topologien (vgl. Abb. 2). Starke Veränderungen konnten aber erst ab Gewichtungen von $s < 0.2$ beobachtet werden.

Bild 2

Bereits mit geringer Gewichtung s führte die Optimierung folglich zu Topologien mit einer hohen Stabilität gegen gezielte Sabotageangriffe. Zusammenfassend ist festzustellen, dass der Algorithmus mit einer Gewichtung von $s = 0.2$ zu Topologien führt, die sowohl angriffsstabil als auch netzwerkeffizient sind.

6 Resümee

In der vorliegenden Arbeit wurde ein formales Modell für die Analyse von Overlay-Streaming-Systemen aufgestellt. Mit Hilfe des Modells wurde als neue Effizienzmetrik $hp(G, \mathcal{T})$, das Verhältnis der Distanzsummen von konstruierter Topologie \mathcal{T} und einem minimalen Spannbaum in G , eingeführt.

Zur Konstruktion von Topologien, die einen guten Kompromiss zwischen Angriffsstabilität und Netzwerkeffizienz realisieren, wurde ein verteilter Algorithmus beschrieben, der durch die Verbindung der Effizienz- und Stabilitätskosten in einer gewichteten Summe eine Abwägung zwischen beiden Optimierungszielen ermöglicht.

In einer Simulationsstudie wurde gezeigt, dass der Algorithmus trotz einer Beschränkung auf lokales Wissen mit Hilfe der Kostenoptimierung in der Lage ist, stabile und effiziente Topologien zu konstruieren. Dabei stieg den Erwartungen entsprechend die Angriffsstabilität bei einer erhöhten Gewichtung der Stabi-

lität und sank das $hp(G, \mathcal{T})$ der Topologien bei einer erhöhten Gewichtung der Effizienz. Als Gewicht für die Kostensumme wurde ein Bereich ermittelt, in dem die Topologien sowohl gute Effizienz- als auch Stabilitätseigenschaften aufweisen. In allen Simulationsszenarien lag das hp dabei unter dem hp eines Client-Server-Unicast für die gleiche Knotengruppe.

Offene Fragen für zukünftige Arbeiten sind die Bewertung der Stabilität gegenüber zufälligen Ausfällen von Knoten, für welche die Angriffsstabilität lediglich eine obere Schranke angibt, sowie die Analyse der Signalisierungsprozedur auf Schwachstellen gegenüber vorsätzlichen Angriffen.

Literatur

- [1] Suman Banerjee, Bobby Bhattacharjee, and Christopher Kommareddy. Scalable application layer multicast. In *ACM SIGCOMM Computer Communication Review*, pages 205–217, 2002.
- [2] S. Birrer, D. Lu, F.E. Bustamante, Y. Qiao, and P. Dinda. FatNemo: Building a resilient multi-source multicast fattree. In *9th International Workshop on Web Content Caching and Distribution*, pages 182–196, 2004.
- [3] Micheal Brinkmeier, Thorsten Strufe, and Günter Schäfer. Towards Stable P2P-Streaming-Topologies. Technical report, Technische Universität Ilmenau, 2006.
- [4] T. Bu and D. Towsley. On distinguishing between Internet power law topology generators. In *21st Annual Joint Conference of the IEEE*

Computer and Communications Societies (INFOCOM), pages 638–647, 2002.

- [5] M. Castro, P. Druschel, A. Kermarrec, A. Nandi, A. Rowstron, and A. Singh. SplitStream: High-bandwidth multicast in cooperative environments. In *19th ACM Symposium on Operating Systems Principles*, pages 298–313, 2003.
- [6] Miguel Castro, Peter Druschel, Anne-Marie Kermarrec, and Anthony Rowstron. Scribe: a large-scale and decentralized application-level multicast infrastructure. *IEEE Journal on Selected Areas in Communications*, 20(8):1489 – 1499, 2002.
- [7] Y. H. Chu, S. G. Rao, S. Seshan, and H. Zhang. A Case for End System Multicast. *IEEE Journal on Selected Areas in Communications*, 20(8):1456–1471, Oct 2002.
- [8] Cristiano P. Costa, Italo S. Cunha, Alex Borges, Claudiney V. Ramos, Marcus M. Rocha, Jussara M. Almeida, and Berthier Ribeiro-Neto. Analyzing client interactivity in streaming media. In *13th international conference on World Wide Web*, pages 534–543, 2004.
- [9] F. Dabek, R. Cox, F. Kaashoek, and R. Morris. Vivaldi: A Decentralized Network Coordinate System. In *ACM SIGCOMM*, 2004.
- [10] Paul Francis. Yoid: Extending the internet multicast architecture. 2000.
- [11] John Jannotti, David K. Gifford, Kirk L. Johnson,

- M. Frans Kaashoek, and James W. O’Toole, Jr. Overcast: Reliable Multicasting with an Overlay Network. In *4th Symposium on Operating System Design and Implementation (OSDI)*, pages 197–212, 2000.
- [12] Zhi Li and Prasant Mohapatra. HostCast: a new Overlay Multicasting Protocol. In *IEEE International Conference on Communications (ICC)*, volume 1, pages 702 – 706, 2003.
- [13] J. Liang and K. Nahrstedt. DagStream: Locality Aware and Failure Resilient Peer-to-Peer Streaming. In *Multimedia Computing and Networking (MMCN)*, volume 6071 of *Proceedings of SPIE*, pages 224 – 238, 2006.
- [14] Sylvia Ratnasamy, Mark Handley, Richard Karp, and Scott Shenker. Topologically-Aware Overlay Construction and Server Selection. In *21st Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, volume 3, pages 1190 – 1199, 2002.
- [15] Alfonse Shimbel. Structural parameters of communication networks. *Bulletin of Mathematical Biophysics*, 15:501 – 507, 1953.
- [16] Thorsten Strufe. Ein Peer-to-Peer-basierter Ansatz für die Live-Übertragung multimedialer Daten. 2007.
- [17] Thorsten Strufe, Günter Schäfer, and Arthur Chang. BCBS: An Efficient Load Balancing Strategy for Cooperative Overlay Live-Streaming. In *IEEE International Congress on Communications (ICC)*, pages 304–309, 2006.
- [18] Thorsten Strufe, Jens Wildhagen, and Günter Schäfer. Towards the construction of Attack Resistant and Efficient Overlay Streaming Topologies. In *2nd International Workshop on Security and Trust Management*, pages 108–118, 2006.

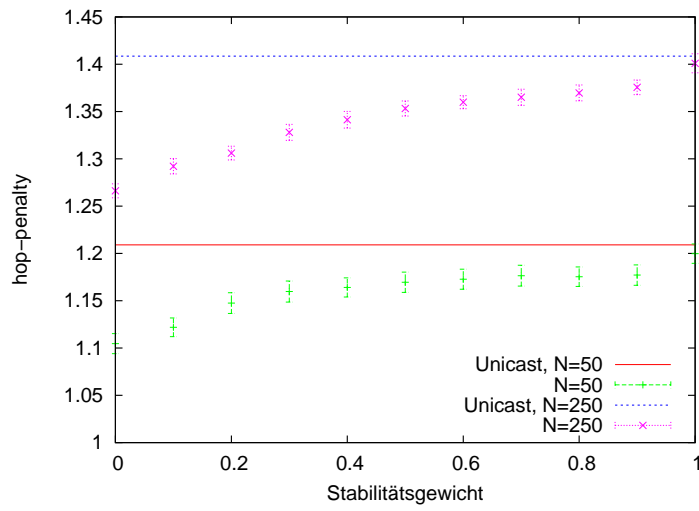


Abbildung 1: Hop-Penalty hp entstehender Topologien im Vergleich zu Client-Server-Unicast-Streaming (16 Simulationsläufe, 98% Konfidenz)

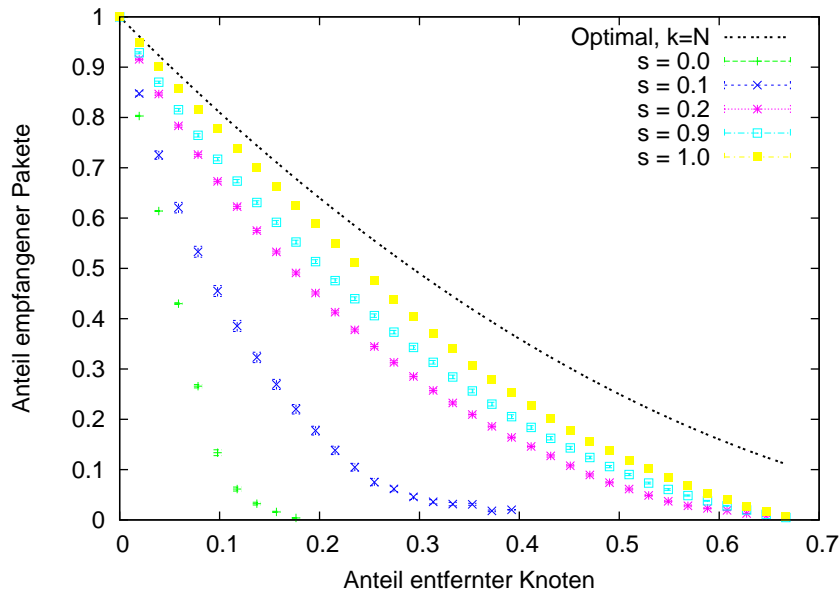


Abbildung 2: Minimaler Anteil empfangener Pakete im System nach Anteil entfernter Knoten bei unterschiedlicher Stabilitätsgewichtung (jeweils 16 Simulationsläufe, 98% Konfidenz)