

混在通信環境における自律協調型メディア配信システム

三島和宏^{†1} 遠峰隆史^{†1}
朝枝仁^{†1} 中村修^{†2}

インターネットを介したメディア配信を行う際、1) ネットワーク環境、2) 計算機環境、3) メディアフォーマットを送受信者間であらかじめ特定しておく必要があるが、ネットワークインフラストラクチャの発達およびプロセッサやメモリ性能向上に伴う計算機環境の発達により、ユーザ間の通信環境を単一の基準で特定することは困難となった。このため、理想的なメディア配信を実現するにあたり、個々のユーザの持つ通信環境に応じて動的に適應できる配信技術が必要である。本研究では、メディア配信における通信環境の状態変化を判別する環境適応型の自律協調型メディア配信を実現した。本手法の実現のため、ネットワーク資源や計算機資源の把握と指標化を行い、それらの状態に応じたノード間ネゴシエーションを用いた配信制御を行う機構を開発した。

Adaptive Media Streaming System for the Heterogeneous Communication Environment

KAZUHIRO MISHIMA,^{†1} TAKASHI TOMINE,^{†1} HITOSHI ASaeda^{†1}
and OSAMU NAKAMURA^{†2}

A data sender and receiver using a media transport system on the Internet need to know the condition of 1) network environment, 2) computational environment, and 3) acceptable media format, prior to their data transmission. Due to the heterogeneity of network infrastructures and computer resources, their communication environments are always changeable. As the ideal media transport system, it is necessary to provide a system that dynamically negotiates the communication factors, e.g. data transmission rate, between a data sender and receiver during their communication. This paper proposes an adaptive media streaming system that recognizes the network condition and dynamically changes the communication factors between a data sender and receiver. This system indicates the conditions for the data transmission, and decides the appropriate communication factors.

1. はじめに

インターネットテクノロジーの発達により、インターネットを介した映像や音声などのメディア配信が広く行われるようになった。メディア配信では、ノード間でのフォーマット同一性、ノード間でのネットワーク資源確保、各ノードでの計算機資源確保などが要求され、条件を満たせない場合、映像・音声の送受信や再生に支障が発生する。

現在、通信環境として、10Gigabit EthernetやGigabit Ethernetなどの非常に広帯域なネットワークが利用可能となった。また、高性能な計算機を用いることにより、従来の機器と比較してより高度な処理が可能となり、様々

な種類のメディア配信システムやフォーマットをユーザが自由に選択・使用することが可能となった。これに対し、PDAや携帯電話などの携帯端末を利用した通信環境も徐々に利用されるようになったが、携帯端末では、高品位な映像・音声を送受信するためのネットワーク環境や、再生するための計算機環境を通常の計算機と比較して十分に得られるとは限らない。

このような様々な要素が混在する通信環境下では、メディア配信を単一の通信帯域、メディアフォーマットで一律に提供することは適切とは言えず、ノードの資源を含む通信環境に応じたメディア配信を行う必要がある。本研究では、エンドノード間での自律協調型メディア配信の実現のために、各ノードにおける計算機ならびにネットワーク資源情報の指標化を行い、ノード間での情報交換・共有、取得した情報を基にした配信制御を行う“自律協調型メディア配信手法”を提案する。

^{†1} 慶應義塾大学 大学院 政策・メディア研究科
Graduate School of Media and Governance, Keio University
^{†2} 慶應義塾大学 環境情報学部
Keio University Faculty of Environmental and Information
Studies

2. 既存技術とその問題点

既存のインターネットを介したメディア配信において、通信環境の変化に適応するための手法としては、1) メディアセッション情報記述プロトコル、および2) 個々のメディア配信アプリケーションにおける実装が挙げられる。ここでは、それぞれの手法において、通信状況の変化に応じたメディアフォーマットの選択（フォーマット）、送受信者のシステム環境の変化に応じた転送レート制御（システム）、送受信者間のネットワーク環境の変化に応じた転送レート制御（ネットワーク）という3つの観点について、3段階（○、△、×）で評価した結果を示す。

2.1 メディアセッション情報記述プロトコル

メディアセッション情報記述プロトコルは、メディア配信に用いるメディアフォーマットの情報を記述するために構成されたプロトコルセットである。表1に、インターネットを介したメディア配信で主に利用されている3つのプロトコル、SDP¹⁾、H.245²⁾、MPEG21³⁾+MPEG7⁴⁾について評価を行った。

いずれのプロトコルにおいてもメディアフォーマットに対する記述は一樣に行うことができる。これは、セッション開始時にノード間の持つメディアフォーマット情報を交換するために用いられるためである。しかし、計算機システムやネットワーク状態の変化に対して動的に対応することができないため、既存のメディアセッション記述手法を用いた制御では、十分な制御ができない。

2.2 メディア配信アプリケーション

2.1項と同様に、表2に、インターネットで広く利用されている3つのメディア配信アプリケーション（DVTS⁵⁾、Windows Media Player、VideoLAN Clinet）について評価を行う。

いずれのメディア配信アプリケーションにおいても、メディアフォーマットに対する制御やネットワーク状態に応じたレート制御を一樣に行うことができる。しかし、計算機システム状態やネットワーク状態の変化に対する処理が不十分であるため、既存のアプリケーションでは、動

表1 情報記述プロトコルの評価

	フォーマット	システム	ネットワーク
SDP	○	×	△
H.245	○	×	△
MPEG21+7	○	○	△

表2 メディア配信アプリケーションの評価

	フォーマット	システム	ネットワーク
DVTS	○	×	△
WMP	○	×	○
VLC	○	×	△

的な変化に対して十分な制御ができない。

3. 環境資源を考慮する自律協調型メディア配信手法の提案

本研究では、ノード間での自律協調型メディア配信の実現を目指し、1) 計算機の資源量を用いた計算機状態把握と配信制御、2) パケットの伝送特性を用いた動的レート制御の2つの制御手法を用いた配信手法を提案する。

3.1 資源量を用いた計算機状態把握と配信制御

本手法では、1) プロセッサ資源、2) メモリ資源、3) ネットワーク資源の3点について、処理可能な資源が残存しているかどうかに着目し、メディア配信に必要な資源がノード上で確保できているかどうかについて把握を行い、取得した値と評価値を基にして制御する。

メディア配信では、映像・音声の送受信や再生のため、データ送受信状態ではすべての資源が定常状態と比較して大きく使用される。また、配信に使用されるメディアフォーマットによって、配信されるデータの処理に必要な資源量が異なるため、消費される資源量は変化する。

高品位なメディア配信ではより多くの計算機資源を必要とし、資源量を用いた指標はより大きく変化する。このことから、資源量に着目した制御は多くのメディア配信アプリケーションにおける配信制御手法として有用性がある。

3.2 伝送特性を用いた動的レート制御

パケット伝搬遅延時間の変動の多くは、計算機やネットワーク状態が高負荷になり、負荷軽減処理としてバッファリングが発生することにより起こる。このような伝搬遅延時間の変動をジッタと呼ぶ。図1は、35Mbpsに帯域幅が制限されたネットワーク上に30Mbps消費するDVTSストリームが流れている状態で、パケットベースキュー⁶⁾を用いたNetperf⁷⁾によるバーストトラフィックが瞬間的に生じた場合の送信ノードでのジッタとパケットロスの値を示したグラフである。このグラフからバーストトラフィックの発生とともにジッタは大きく正負に変動していることが確認できる。このようにジッタは、ネットワークの変化、特に帯域幅の不足や伝搬の遅延時間変化の際に大きな変動をする。

リアルタイムアプリケーションにおける伝搬遅延時間の揺らぎは、エンドノード間で容易に計測が行え、再生と密接な関係があることから制御に対する有用性がある。本手法ではネットワークの変化の検知とメディアストリームのトラフィック減少制御にジッタを、その後の増加制御には補助的な指標としてパケットロスの値を用いる。

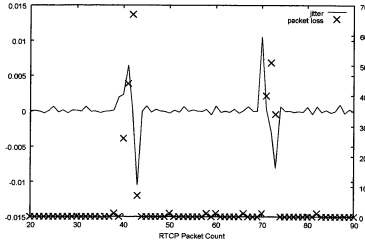


図1 瞬間的な輻射時のジッタとパケットロス

4. AMSS: Adaptive Media Streaming System

本機構は、送信ノードならびに受信ノードの2つの計算機により構成される。送信ノードでは複数のメディアフォーマットの送信に対応できる入力デバイスと配信アプリケーションが実装されている。受信ノードでは複数のメディアフォーマットの受信に対応できるCODECが実装されており、受信した映像・音声データは自身のモニターで表示される。ノード間はIPネットワークを通じて、RTP⁸⁾パケット化された映像・音声データが配信される。

本機構では、複数のメディアフォーマットを動的に切り替え、環境変化に対応する。本機構は、1) HDV(MPEG2-TS)フォーマット、2) DVフォーマット、3) RGB(非圧縮ビットマップ)フォーマットの3つのメディアフォーマットを扱うことができるメディア配信機能を備える。また、用いるメディアフォーマットに対して1)から3)の順で使用する優先順位を設定する。

本機構における各モジュールの関係を図2に示す。資源計測モジュールは、リクエスト・レスポンス制御モジュールに情報を通知し、相手ノードに対して必要な処理を要求する。送信ノードではリクエスト・レスポンス制御モジュールからカプセル化フィルタに対してメッセージ通知を行うことで、動的なフォーマットスイッチングならびにパラメータ変更を実現する。

4.1 統一的な資源の指標化

ユーザの持つ環境資源は使用する計算機やオペレーティングシステムによって取得に用いる手法や取得可能なデータ形式が異なる。これらの差異を吸収するため、本機構では資源に関して統一的な指標化を行う。

4.1.1 計算機資源の指標化

本機構では、計算機の持つプロセッサ、メモリ、ネットワークのそれぞれの資源使用率をまず求める。各資源ごとの使用率 P は、計測時点(j)での資源使用量 U と使用可能な資源の総量 A により求める。

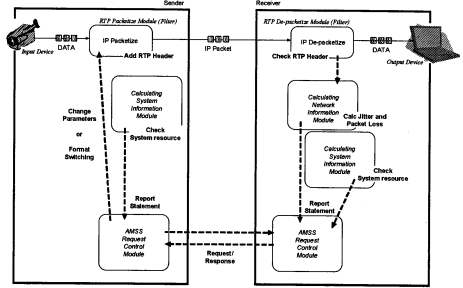


図2 各モジュール間関係概要

表3 メディアフォーマットによる値

	プロセッサ	メモリ	ネットワーク
DV	350	1.0	35
HDV	450	1.5	30
RGB	100	0.5	60

表4 プロセッサ種別による値

	プロセッサ	メモリ
Pentium3	4.0	0.5
Pentium4	2.0	0.2
PentiumM	3.5	0.4
Xeon(P4)	2.0	0.2

$$P_j = \frac{U_j}{A_j} \quad (1)$$

メディア配信の制御は、それぞれの資源値に対して計測時点(j)での判断値 D_j と許容値 T を用いて行われる。判断値 D_j は、求められた各資源の使用率は n 区間分の移動平均により平滑化し、値 F と値 R を用いて求める。

$$D_j = \frac{1}{n} \sum_{i=j-n+1}^j P_i + \frac{F}{R} > T \quad (2)$$

メディア配信システムが使用する資源量は、利用するメディアフォーマットやプロセッサ性能などによって異なる。例えば、同一プロセッサの場合、プロセッサのクロック性能に比例してメディア配信に使用されるプロセッサ使用率は低くなる。そこで、本機構ではメディアフォーマットや計算機の持つ性能によって変化する値 F 、値 R を用いる。

表3は各メディアフォーマットが使用する各資源量について、表4はプロセッサの性能について、それぞれ事前の実験により求めた値である。判断値との比較を行う値 D_j を求める際に用いる値 F は表3の値を、値 R は表4の値とプロセッサのクロック周波数 C を乗じた値をそれぞれ用いる。

$$R = E \times C \quad (3)$$

要求	処理内容
REGISTER	初期ノードへの参加要求
NODELIST	初期ノードへのノードリストの配信要求
START	メディア配信の開始要求
CAPEX	ノード能力チェック要求
RESTART	メディア配信の再開要求
CHGFMT	フォーマットスイッチング要求
CHGCFG	フォーマット設定変更要求
STOP	送信停止要求
LEAVE	ノード離脱要求

また、メディア配信時に使用されるネットワーク帯域幅は計算機資源に影響されないため、 R の値は1を用いる。

4.1.2 ネットワーク資源の指標化

メディア配信中のネットワーク状態の変化を把握するためにネットワークの伝送特性を用いる。本機構では、指標としてパケット喪失数ならびにパケットの伝搬遅延時間の揺らぎを用いる。パケット喪失数 L は、時刻 s と時刻 t ($s < t$) における到達パケットのシーケンス番号を用いて求める。

$$L = S_j - S_i \quad (4)$$

パケットの伝搬遅延時間の揺らぎ J は、時刻 m と時刻 n ($m < n$) におけるパケットの送信ノードでの送出時刻 S と受信ノードでの到達時刻 R を用いて求める。

$$J = (R_n - S_n) - (R_m - S_m) \quad (5)$$

4.2 ノード間ネゴシエーションを用いた配信制御

本機構では、ノード間で必要な情報を交換し、相互ノードの情報を共有し、環境変化に応じて動的に制御する。

4.2.1 ノード間での交換と共有

環境資源情報やその他接続に必要な情報を送受信ノード間で、HTTP⁹⁾のリクエスト・レスポンスを拡張した手法を用いて交換する。情報の交換は表5に挙げるリクエストメソッドを用いて行われ、相手ノードに対する要求を“リクエスト”，相手ノードからの要求処理に対する返答を“レスポンス”と呼ぶ。リクエストには必ず何らかの要求を示す“メソッド”と行うべき処理に必要な情報を記述した“メッセージ”が記述される。レスポンスには相手ノードにて要求した処理を実行した結果が記述されるとともに、相手ノードにおいて必要な情報が“メッセージ”として追加的に記述される。

4.2.2 資源の記述手法

リクエストメソッドには、必要に応じて各環境資源情報がメッセージとして含まれる。各環境資源情報の記述には、SDPを拡張した記述手法を用いる。a (attribute) フィールドを用いて記述され、セッション全体に対して記述する場合はSDPのセッションレベル属性として、メディアフォーマットごとに記述する場合はSDPのメディアレベル属性を用いる。図3に記述例を示す。

4.2.3 ネゴシエーション動作の流れ

本機構におけるリクエスト・レスポンスの流れを図4に示す。メディア配信を行っている間に計算機やネットワーク状態が変化した場合は、CHGCFG要求ないしはCHGFMT要求を発行し、状態変化にあわせたメディア配信を継続させる。

5. 評価実験

AMSSを用いた自律協調型メディア配信の有効性を検証するために、図5に示す実験環境を構築し、1) 状況に適したフォーマットの選択、2) 状況に応じた配信制御の2点に関する実験を行った。実験ネットワークは、送受信ノード各2台を帯域制御を行うTraffic Shaperを介して接続し、dummysnet¹⁰⁾を用いて擬似的にネットワーク帯域幅を50Mbpsに制限している。

5.1 実験1：状況に適したフォーマットの選択

送信開始前のネゴシエーションの動作について、送信ノード1ならびに受信ノード1を用いて検証した。

受信ノード1では、HDVフォーマットの受信に対して十分な資源が確保できない計算機を用いるため、設定された優先度に従い、HDVフォーマット以外が選択される。

図6に受信ノードでのAMSSの動作ログを示す。受

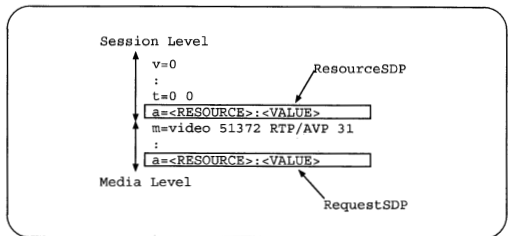


図3 拡張SDPメッセージ

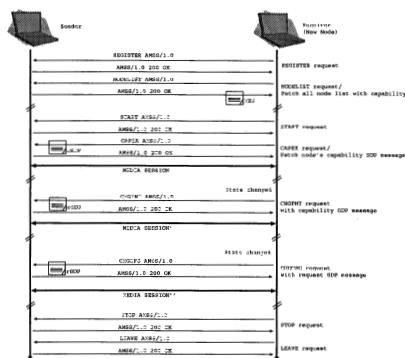


図4 リクエスト・レスポンスの流れ

信ノードにおける環境資源値の取得ならびに値による状況評価の結果、DV フォーマットが選択され、送信ノードに対して DV フォーマットでのメディア配信開始要求 (CAPEX 要求) を発行 (時刻:1136636065) している。

5.2 実験 2 : 状況に応じた配信制御

資源状態に基づくフォーマット変更要求である CHGFMT リクエスト、ネットワーク伝送特性に基づくパラメータ変更要求である CHGCFG リクエストの動作について、送信ノード 1、送信ノード 2、受信ノード 1 ならびに受信ノード 2 を用いて検証した。

受信ノード 1 では、実験 1 同様、HDV フォーマットの受信に対して十分な資源が確保できない計算機を用いる。また、送信ノード 2 ならびに受信ノード 2 の間で AMSS 非実装のアプリケーションを用いた DV フォーマットでのメディア配信を行うことで一時的にネットワーク全体でメディア配信に対して十分なネットワーク帯域幅を確保できない状態を構成する。

- (1) 実験に関する各ログの記録を開始する。
- (2) AMSS 実装アプリケーションを起動する。
- (3) CHGFMT 要求の動作を検証するため、強制的に HDV フォーマットでのメディア配信を AMSS 実装アプリケーションに開始させる。
- (4) 計算機資源不足により AMSS がメディアフォーマット切替処理 (CHGFMT 要求の発行) を行う。
- (5) AMSS 非実装アプリケーションによる DV ストリームの送信を開始する。
- (6) AMSS 非実装アプリケーションによる DV ストリームの送信を終了する。
- (7) AMSS 実装アプリケーションを終了する。

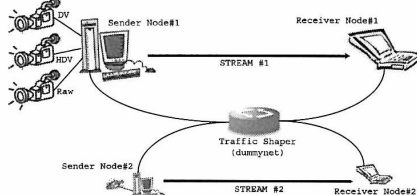


図 5 実験環境

```
1136636062 : Go Experiment Scenario #1.
:
1136636063 : START submitted.
1136636063 : FormatID is DV.
1136636065 : CAPEX submitted.
:
1136636065 : RecvStart.
```

図 6 受信ノードでの動作ログ (実験 1)

(8) 実験に関する各ログの記録を終了する。

受信ノードにおける AMSS の動作ログを図 7 に示し、実験中の受信ノードにおける各資源使用量の変化を図 8 に示す。

動作ログより、ステップ 3 として、強制的に MPEG2-TS フォーマットを選択し、送信ノードに対して CAPEX リクエストを発行 (時刻:1136746608) している。その後、ステップ 4 として、受信ノードにおいて環境資源値の取得ならびに値による状況評価の結果、許容値の超過が認識 (時刻:1136746612) された。再度フォーマットの選択を行った結果、DV フォーマットが選択され、送信ノードに対して CHGFMT リクエストを発行 (時刻:1136746612) している。このように、受信ノードでは、CAPEX リクエストならびに CHGFMT リクエスト発行後、各フォーマットでの受信を行う設定が行われた。また、資源許容量の超過を認識してから新たなフォーマットへの切り替え・再生まで要した時間は、249 ミリ秒である。

さらに、ステップ 5 として、DV ストリームの送信開始と同時に、受信ノードでパケットの伝搬遅延時間の揺らぎならびにパケットロスの増加を検知し、フレームレート変更のリクエストを CHGCFG リクエストにより発行し、メディア配信のパラメータ制御を行っている。(時刻:1136746637 から 1136746681)

各資源量の推移のグラフでは、AMSS の起動と同時にアプリケーションによるメモリ確保が、メディア配信の

```
1136746607 : Go Experiment Scenario #3.
:
1136746608 : START submitted.
1136746608 : FormatID is MPEG2-TS.
1136746608 : CAPEX submitted.
:
1136746612 : Over-threshold-count.(59645844/diff:59645844)
1136746612 : FormatID is DV.
1136746612 : CHGFMT submitted.
:
1136746612 : RecvStart.(59646093/diff:249)
1136746637 : CHGCFG/DV-rate-down submitted.
:
1136746681 : CHGCFG/DV-rateup submitted.
```

図 7 受信ノードでの動作ログ (実験 2)

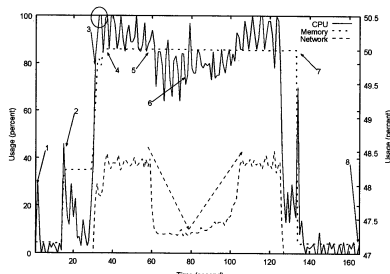


図 8 各資源量の変化 (実験 2)

開始と同時にメディア配信のためのメモリ確保が行われている。また、MPEG2-TS フォーマットによる受信を開始した段階で、プロセッサ使用率が連続的に 100%を超過しており、この時点で許容値を超えたためフォーマット変更処理が行われた。また、2 本目の DV ストリームが送信されると同時にネットワーク状態の変化を検知し、DV のフレームレートを下げる処理を行ったため、ネットワーク使用率が減少している。2 本目の DV ストリームの停止後は、レート増加処理によって、徐々にネットワーク使用率が増加している。

6. おわりに

本研究では、計算機環境の持つプロセッサ、メモリなどの計算機資源、ならびにネットワーク資源と言ったメディア配信を取り巻く通信環境の変化に適応する自律協調型メディア配信システムを実現した。これより、インターネットを介したメディア配信において求められる、1) ネットワーク環境、2) 計算機環境、3) メディアフォーマットの 3 つの要件に対する自律的な判断と状況に応じたノード間の協調動作により、ユーザが資源状態を意識することなく、ネットワークならびに計算機状態に応じたメディア配信環境が実現できた。

提案した自律協調型メディア配信システムは、計算機資源やネットワーク資源などの通信環境の変化に応じた配信制御を行うメディア配信システムである。この実装を用いることにより、複数のメディアフォーマットでの送受信、必要に応じたメディアフォーマットの動的な切り替え、1 対 1 の通信ノード間での自律協調的な環境資源の計測、1 対 1 の通信ノード間での情報交換 (ネゴシエーション)、環境資源の統一的な指標化・ノード環境に依存しない情報定義、ノード間で協調した配信制御が実現できた。

本研究が提案する混在通信環境における自律協調型メディア配信手法では、1) 計算機資源の変化に応じたメディアフォーマットの変更、2) 計算機資源量に基づくメディアフォーマットの決定、3) ネットワーク状態に応じたパラメータの変更を行ない、資源変化の検知と状態に適応したメディア配信を可能とした。

6.1 今後に向けて

本機構は、RTP を用いて配信を行うメディアフォーマットについては設計を変更することなく適用できる。更なる拡張性のため、今後、本機構の他フォーマットへの対応も検討したい。この際、より多くのフォーマットへの対応を行うため、本機構が指標とした値以外の指標についての検討も必要と考える。また、各フォーマット独自の制御手法が存在する可能性も考慮し、フォーマット個別に対する制御を行うだけでなく、新しい制御グループを定義し、

特徴の似たメディアフォーマットについては抽象的に制御を表現できる手法を追加検討したい。

本研究で指標化した資源値は、いずれのオペレーティングシステムにおいても取得可能なものである。今後は様々なオペレーティングシステム環境において資源情報を取得するための汎用モジュールを構成することで更に多くのシステム環境に対して適応可能となる。このため、オペレーティングシステムに応じた資源情報の取得ならびにオペレーティングシステムに依存しない AMSS への抽象化されたインタフェースの確保を検討したい。

謝 辞

本研究の一部は、情報通信研究機構 (NICT) の委託研究「ダイナミックネットワーク技術の研究開発」の助成を受けて実施したものである。

参 考 文 献

- 1) M. Handley and V. Jacobson. SDP: Session Description Protocol. RFC 2327, IETF, April 1998.
- 2) The ITU Telecommunication Standardization Sector (ITU-T). H.245. <http://www.itu.int/rec/recommendation.asp?type=folders&lang=e&parent=T-REC-H.245>, 2005.
- 3) Moving Picture Experts Group. MPEG-21. <http://www.chiariglione.org/mpeg/standards/mpeg-21/mpeg-21.htm>, Oct 2002.
- 4) Moving Picture Experts Group. MPEG-7. <http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm>, Jul 2001.
- 5) A. Ogawa, K. Kobayashi, K. Sugiura, O. Nakamura, and J. Murai. Design and Implementation of DV Stream over Internet. *Proceedings of Internet Workshop '99*, pages 255–260, Feb 1999.
- 6) S. Keshav. A control-theoretic approach to flow control. *Proceedings of the conference on Communications architecture & protocols*, pages 3–15, 1993.
- 7) HP. The Public Netperf Homepage. <http://www.netperf.org/netperf/>.
- 8) Audio-Video Transport Working Group, H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RTP: A Transport Protocol for Real-Time Applications. RFC 1889, IETF, January 1996.
- 9) R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-Lee. Hypertext Transfer Protocol – HTTP/1.1. RFC 2616, IETF, June 1999.
- 10) Luigi Rizzo. Dummynet: a simple approach to the evaluation of network protocols. *ACM Computer Communication Review*, 27(1):31–41, 1997.