

# TCP スループットの確率的拡散予測に基づく映像配信制御

吉田 裕志<sup>†</sup> 里田 浩三<sup>†</sup> 野垣 智士<sup>†</sup>

<sup>†</sup> NEC サービスプラットフォーム研究所 〒211-8666 神奈川県川崎市中原区下沼部 1753  
E-mail: †h-yoshida@jh.jp.nec.com, k-satoda@cb.jp.nec.com, nogaki@ak.jp.nec.com

インターネットやモバイルネットワークなどのベストエフォート型ネットワーク上での映像配信サービスが普及してきた。これらの映像配信サービスでは、その通信プロトコルに HTTP/TCP が利用されることが多いが、十分な TCP スループットが得られず再生停止が頻発し、ユーザの体感品質を損なうことがある。これに対して、TCP スループットに応じて映像ビットレートを制御する適応配信システムが数多く研究開発されてきている。しかしながら、従来の適応配信システムは、今現在の TCP スループットに適応した制御であり、将来の TCP スループット変動を考慮していないため、TCP スループット変動が激しい環境では高い体感品質が得られていない。本稿では将来の TCP スループットの広がり (確率的拡散) を予測して適応配信に活用することで、安定した視聴を実現する適応配信システムを提案する。TCP スループットの変動を Wiener 過程でモデリングすることで、TCP スループットの確率的拡散を 80%以上という高い精度で予測できることを示す。さらに、この確率的拡散予測結果を用いることで、TCP スループット変動が激しい環境でも高い再生停止回避能力を実現できることを示す。

## Video Streaming Control

### by Predicting Stochastic Diffusion of TCP Throughput

Hiroshi YOSHDIA<sup>†</sup>, Kozo SATODA<sup>†</sup>, and Satoshi NOGAKI<sup>†</sup>

<sup>†</sup> Service Platforms Res. Labs., NEC Corp. 1753 Shimonumabe, Nakahara-ku, Kawasaki, 211-8666 Japan  
E-mail: †h-yoshida@jh.jp.nec.com, k-satoda@cb.jp.nec.com, nogaki@ak.jp.nec.com

Recently HTTP/TCP has been widely used for video streaming services over best-effort networks such as the Internet. However, the varying and insufficient TCP throughput causes frequent playout interruptions. Concerning this issue, a number of adaptive video streaming methods have been researched and developed to adjust to the TCP throughput fluctuations. These conventional methods do not achieve sufficient improvement yet because they adapt to the “present” TCP throughput not to the “future” one. This paper proposes a novel adaptive streaming method that attains stable user experience by predicting stochastic diffusion of the “future” TCP throughput. Experimental results show that a Wiener process model predicts the stochastic TCP throughput diffusion with a high accuracy of over 80%. Furthermore, it is shown that the adaptive streaming method using the predicted stochastic TCP throughput diffusion is able to suppress playout interruptions.

## 1. はじめに

インターネットやモバイルネットワークなどの IP ネットワークを介した映像配信サービスでは、そのプロトコルとして HTTP/TCP が用いられることが多くなってきた。システム構築が安価であることやファイアウォール透過性の問題がない

ことが主な理由である [1]。HTTP で映像を配信する場合、そのトランスポート層である TCP のスループットが映像配信の品質を左右する。配信する映像のビットレートに対して TCP スループットが不十分であれば、クライアント側で再生バッファが枯渇して再生が停止してしまい、ユーザの体感品質を劣化させることになる。Wang らは、TCP スループットが映像

ビットレートの約2倍あれば、数秒程度の初期バッファリングで再生停止を回避できることを示したが[2]、近年の映像の高精細化に伴う映像ビットレートの増加により、インターネットやモバイルネットワークなどのベストエフォート型ネットワークでは十分なTCPスループットが得られない場合が多い。

そこで、TCPスループットに応じて映像ビットレートを制御するような商用の配信システムが登場してきている[1], [3], [4]。いずれも、一つの映像コンテンツに対して複数のビットレートから選択して配信可能なように構成された配信システムであり、TCPスループットに応じて、配信する映像のビットレートを決定する。本稿では、このようにネットワークの状態(ここではTCPスループット)に応じて複数の映像ビットレートから一つを選択することをレート制御(rate control)と呼び、レート制御などを活用して変動するネットワーク状態に適した映像を配信する方式を適応配信(adaptive streaming)方式と呼ぶ。

これまでに、数多くの適応配信方式が提案[5]~[7]および評価[8]されてきている。[5]および[6]は、映像を配信してからクライアントで受信して再生するまでの一連の流れを閉ループ制御系としてモデリングし、PID制御<sup>(注1)</sup>を用いて映像ビットレートを制御する手法をとっている。[7]は、レート制御を独自に定義した体感品質に関する評価関数を最大にする最適化問題に帰着している。[8]では、Smooth Streaming, Netflix Player, Adobe OSMFの三つの適応配信方式の評価を実施している。いずれの適応配信方式もある程度有効性が示されたものの、大きなTCPスループット変動への適応性が低く、高い体感品質を得るためにはまだ改善が必要であると、この評価は結論付けている。

[8]で示されたように、従来の適応配信方式では、TCPスループット変動が激しい場合に対応できず、高い視聴品質が得られていない。これは、従来の適応配信方式は今現在のTCPスループットに適応するレート制御を用いることが大きな原因である。将来のTCPスループット変動の予測に基づいて適応配信を実施すれば、より高い視聴品質を実現できるようになる。

しかしながら、TCPスループットは、RTT、パケットロス、可用帯域、クロストラヒック、電波状態など時々刻々変動する複数の要因が複雑に絡み合った結果として表れるため、将来のTCPスループットを確定的に予測することは困難である[9]。我々はTCPスループット自体を予測するのではなく、TCPスループットの確率的な広がり<sup>(注2)</sup>を予測することを提案する。本稿では、この確率的な広がり<sup>(注2)</sup>のことを**確率的拡散(stochastic diffusion)**と呼ぶ。そして、TCPスループットの確率的拡散

(注1)：フィードバック制御系補償器の一つ。目標値との偏差の比例(Proportional)、積分(Integral)、微分(Derivative)の線形結合によって制御操作量を決定する。

(注2)：これはちょうど台風の進路予想と似ている

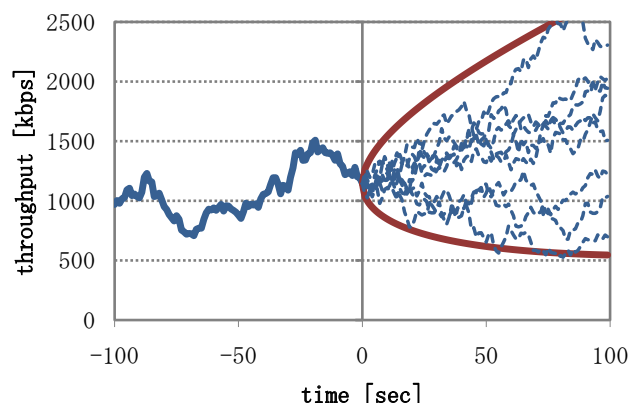


図1 TCPスループット変動と確率的拡散のイメージ

予測に基づき、レート制御を実施することで、高品質な映像視聴を実現する手法を提案する。

本稿の構成は次のとおりである。2章では、Wiener過程に基づいてTCPスループットの確率的拡散を予測する手法を提案する。その予測結果を用いて変動するTCPスループットに順応する適応配信方式を3章で提案する。4章で、TCPスループットの確率的拡散予測に基いた適応配信方式を映像配信サーバに実装して実験した結果について述べる。そして、5章で本稿をまとめる。

## 2. TCPスループットの確率的拡散予測

筆者らは、将来のTCPスループットの予測を適応配信に活用することで、高い視聴品質を実現できると考えている。しかし、予測が外れた場合、特に実際の値が予測値を下回ってしまった場合には体感品質を大幅に劣化させてしまう可能性がある。一方、上述したように、TCPスループットは複数の要因が複雑に絡み合った結果として表れるため、将来のTCPスループットを予測することは困難である[9]。

本稿では、将来のTCPスループットを**確定的に予測する**のではなく、**確率的な広がり(確率的拡散)**を予測することを考える。確定的に予測するということは、図1の青破線部のように無限に存在する将来のTCPスループットの可能性の中から、唯一に予測値を確定することである。一方、確率的拡散を予測するということは、予測値を唯一に確定せず、図1の赤実線部のように広がり<sup>(注2)</sup>を予測することである。

本章では、TCPスループット予測の関連研究について述べた後、TCPスループットの確率的拡散を予測する手法を提案する。そして、提案手法における確率的拡散の予測精度を実網で検証する。

### 2.1 関連研究

これまでに様々なTCPスループット予測方式が研究されてきている[11]~[14]。それらはHistory-Based(HB)方式とFormula-Based(FB)方式の二つに大別できる[9], [10]。HB方式はTCPスループットの過去の時系列データに基づいて将来のTCPスループットを予測する方式であり、時系列解析

に Auto-Regressive Moving Average (ARMA) モデルやマルコフ変調ポアソン過程 (MMPP) を用いる方式 [11], 機械学習の一つである Support Vector Regression (SVR) を用いる方式 [12] がある. 一方, FB 方式は TCP スループットの挙動を数式として表し, 必要なパラメータを代入して予測する方式であり, [13] や [14] の数式が有名である.

HB 方式は, 予測に要する分だけ時系列データを蓄積しなければならないので, 予測にはある程度の時間が必要であるが, FB 方式と比較して予測精度が高いことが特徴である. 反対に, FB 方式は数式に必要なパラメータを代入することで直ちに予測値が得られるが, その精度は低い.

しかしながら, いずれの関連研究も TCP スループットを確定的に予測する方式についての研究であり, 確率的拡散を予測する手法ではない.

一方, TCP スループットではないが, Zhang らは無線 IP ネットワークにおける端末のハンドオーバー要求数の確率的拡散を予測する手法を提案している [15]. 予測手段は上述の HB 方式であり, 時系列解析に Wiener 過程を用いている.

### 2.2 Wiener 過程に基づく確率的拡散予測

本稿では, Wiener 過程を用いた確率的拡散予測方式 [15] に基づき, TCP スループットの確率的拡散を予測する. 予測に用いられる確率過程モデルとして上記以外も考えるが, 最も基本的であり且つ数学解析が十分に確立された Wiener 過程で予測できるかを検証することにする. また, Wiener 過程から派生して得られる幾何ブラウン運動は株式価格のように非負の値を変動する確率過程モデルとして用いられ, 同じく非負の値を変動する TCP スループットの変動には Wiener 過程よりも好適であるかもしれないが, それは今後の検討課題としておく.

まず, Wiener 過程の定義を紹介する. 確率過程  $W(t)$  が分散  $\sigma^2$  の Wiener 過程である場合, 次の条件を満足する.

- (1)  $W(t)$  は独立増分である
- (2)  $\forall s, t > 0 \Rightarrow W(s+t) - W(s)$  は  $N(0, \sigma^2 t)$  に従う
- (3)  $W(0) = 0$  かつ  $W(t)$  は  $t = 0$  で連続である

さらに, Wiener 過程にドリフトを考慮したものを一般化された Wiener 過程と呼ぶ. 一般化された Wiener 過程は, ドリフトを  $\mu$  とすると, 上記 (2) の  $N(0, \sigma^2 t)$  を  $N(\mu t, \sigma^2 t)$  に変更することで得られる. 以降, 本稿では一般化された Wiener 過程を対象にするが, これを単に Wiener 過程と書くことにする.

TCP スループットが Wiener 過程に従って変動すると仮定すると, 時間  $t$  後の TCP スループット  $x$  の確率密度関数  $f(x, t)$  は式 (1) で表される.

$$f(x, t) = \frac{1}{\sqrt{2\pi\sigma^2 t}} \exp\left(-\frac{(x - x_0 - \mu t)^2}{2\sigma^2 t}\right) \quad (1)$$

ただし,  $x_0$  は現在時刻での TCP スループットである. 式 (1) によると, 確率密度関数の分散は  $\sigma^2 t$  であり, 時間  $t$  に比例して増大する. 確率的拡散の指標としては, 分散の平方根であ

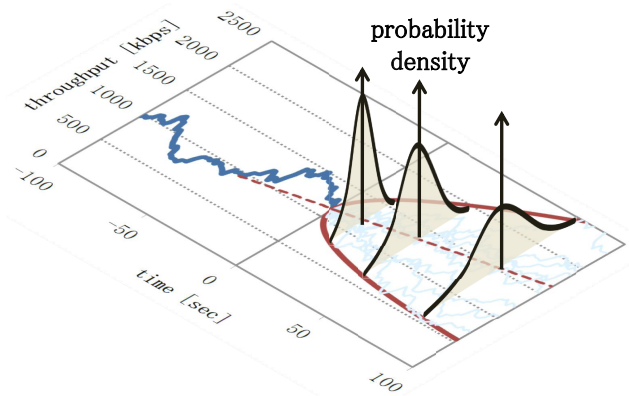


図2 TCP スループットの確率的拡散モデル (Wiener モデル)

る標準偏差が適当であり, その値は  $\sigma\sqrt{t}$  となる. したがって, 確率的拡散は経過時間の  $1/2$  乗に比例することになる. これにドリフトを考慮すると, 図 1 の赤実線で描かれた曲線の関数  $x^\pm(t)$  (曲線の上側を  $x^+$ , 下側を  $x^-$  とする) は式 (2) で計算できる.

$$x^\pm(t) = x_0 + \mu t \pm \alpha \sigma \sqrt{t} \quad (2)$$

ただし,  $\alpha$  は確率的拡散の予測範囲を定める定数である. 例えば,  $\alpha = 3$  であれば  $x^\pm(t)$  で囲まれる範囲に TCP スループットが収まる確率は 99.7% となる (正規分布の  $[-3\sigma, 3\sigma]$  の範囲). すなわち, 確率的拡散を予測するという行為は, 式 (2) の  $x^\pm(t)$  を計算することに等しい.

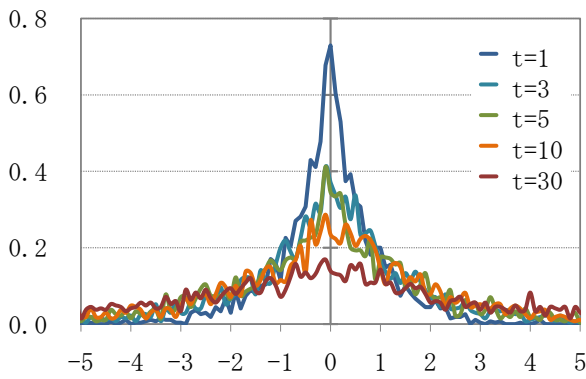
TCP スループットの確率的拡散と Wiener 過程に従う確率密度関数  $f(x, t)$  との関係を表した模式図が図 2 である. この図は, 図 1 を平面に配置して, その垂直方向に確率密度をとっている. 図では異なる三つの時刻での確率密度関数を示している. 式 (1) で示されてるとおり,  $t = 0$  に近いほど確率密度関数はとがった形になり,  $t$  が大きくなる程平たい形になる.

### 2.3 実網での確率的拡散の予測精度検証

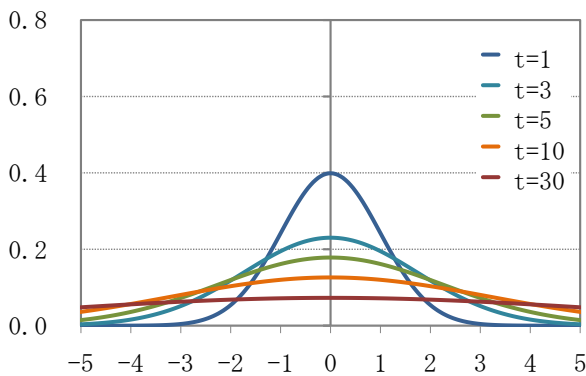
これまでに, 式 (2) の  $x^\pm(t)$  を計算することによって, TCP スループットの確率的拡散を予測できることを述べた. ここでは, 実網における TCP スループットの確率的拡散の予測精度を検証する. 以下で, 検証の手順を説明した後, 予測精度の結果を示す.

#### a) 実網での TCP スループットデータの採集

TCP スループット変動が激しいベストエフォート型ネットワークの代表例として, インターネット上に配置されたサーバからモバイル端末へファイルをダウンロードしたときの TCP スループット, およびモバイル端末からサーバへのファイルをアップロードしたときの TCP スループットを計測した. このとき, TCP スループットの計測は  $I(= 2)$  [sec] ごとに単純移動平均で計算した. この  $I$  [sec] を 1 単位時間として定義し, 以降本章では時間に関してはこの単位時間を用いる. また, モバイル端末は都内 19 箇所から 3G/3.5G および Wi-Fi (Hotspot) を利用してサーバに接続し, ダウンロードとアップロードそれ



(a) 実測 TCP スループット



(b) Wiener 過程による理論値

図 3 確率密度関数 (ヒストグラム)

それぞれ 10 分間=300 単位時間ずつ計測した。すなわち、累計 19 箇所×2(ダウンロード/アップロード)×300 単位時間=11400 単位時間分の TCP スループットデータを取得したことになる。

b) 確率密度関数 (ヒストグラム) の計算

続いて、実網での TCP スループットデータの確率密度関数を計算する。そのために、式 (2) の  $\sigma$  および  $\mu$  を同定する必要がある。

$\sigma$  は、過去  $n (= 30)$  単位時間分の TCP スループットデータを式 (3) に代入して計算する。この式は不偏分散の算出式である。ここで、 $x(-k)$  は現在から  $k$  単位時間前の TCP スループットであり、 $\bar{x}$  は当該  $n$  個の TCP スループットの平均値である。

$$\sigma^2 = \frac{1}{n-1} \sum_{k=0}^{n-1} (\bar{x} - x(-k))^2 \tag{3}$$

また、 $\mu$  は過去  $m (= 10)$  単位時間分の TCP スループットデータから最小二乗法にて算出した直線の傾きとする (式 (4))。

$$\mu = \frac{m \sum_{k=0}^{m-1} (-k)x(-k) - \left( \sum_{k=0}^{m-1} (-k) \right) \left( \sum_{k=0}^{m-1} x(-k) \right)}{m \sum_{k=0}^{m-1} k^2 - \left( \sum_{k=0}^{m-1} (-k) \right)^2} \tag{4}$$

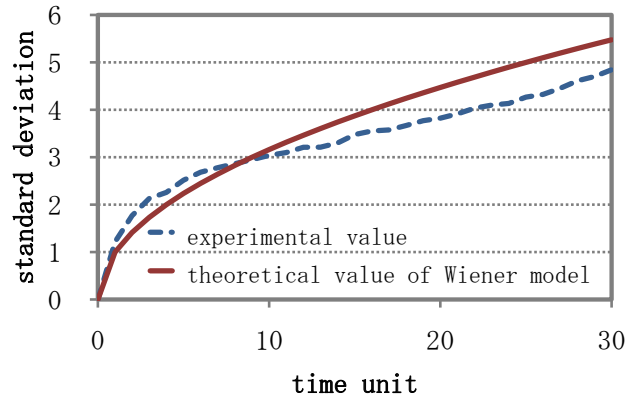


図 4 確率的拡散予測結果

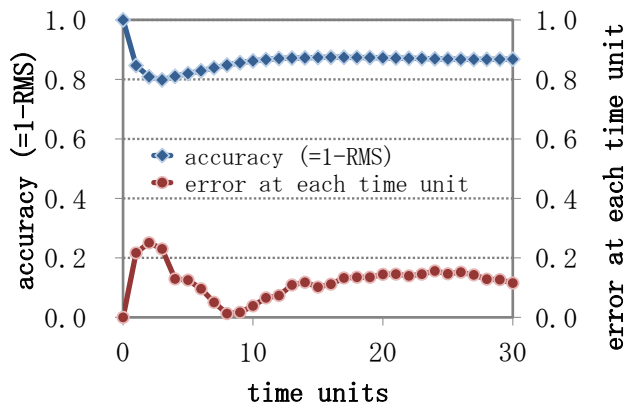


図 5 確率的拡散予測精度

ここで、上記  $\sigma$  および  $\mu$  を用いて  $t$  単位時間先の TCP スループットデータ  $x(t)$  を次式のように正規化する。

$$z(t) = \frac{x(t) - x_0 - \mu t}{\sigma} \tag{5}$$

本検証では、上記 a) で採集した TCP スループットデータに対して、 $t = 0, 1, 2, \dots, 30$  の  $z(t)$  を算出していき、 $z(t)$  に関するヒストグラムを作成する。その結果得られたヒストグラムが図 3(a) である (図では  $t = 1, 3, 5, 10, 30$  のみ表示)。データ量が十分であるとき、このヒストグラムは確率密度関数に近似される。一方、図 3(b) は TCP スループット変動が Wiener 過程であると仮定した場合の理論値である。実測値の方が中心のピークが高くなっているが  $t$  の増加に対する広がりによく近似できている。

c) 確率的拡散の予測精度

最後に Wiener 過程に基づく確率的拡散予測の予測精度を算出する。採集した TCP スループットデータに対する  $z(t) (t = 0, 1, 2, \dots, 30)$  のヒストグラム  $\approx$  確率密度関数それぞれについて標準偏差を算出し、右軸に  $t$ 、縦軸に算出した標準偏差をプロットする (図 4 の青破線)。仮に、Wiener 過程に基づく確率的拡散予測が正しければ式 (2) より  $z(t) \approx \sqrt{t}$  となる。そこで、図 4 には理論値として関数  $\sqrt{t}$  を赤実線で表示している。図を見て明らかのように、Wiener 過程に基づく確

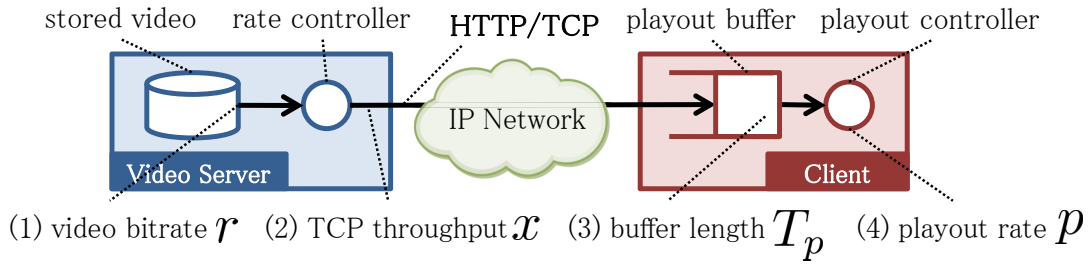


図6 映像配信システム概略

率的拡散予測は高精度に予測できることがわかる。

各単位時間  $t$  における  $z(t)$  と  $\sqrt{t}$  の誤差を  $e(t)$  とし、単位時間 0 から  $t$  までの  $e(k)$  ( $k = 1, 2, \dots, t$ ) の二乗平均平方根 (RMS) を 1 から引いた値を予測精度  $A(t)$  として考えた。  $e(t)$ ,  $A(t)$  それぞれの計算式は式 (6), 式 (7) となる。

$$e(t) = \frac{|z(t) - \sqrt{t}|}{\sqrt{t}} \quad (t > 0), \quad e(t) = 0 \quad (t = 0) \quad (6)$$

$$A(t) = 1 - \sqrt{\frac{\sum_{k=0}^t \{e(k)\}^2}{t+1}} \quad (7)$$

図 4 の予測値から各単位時間における誤差と予測精度を算出した結果を図 5 に示した。いずれの区間  $[0, t]$  ( $t = 0, 1, 2, \dots, 30$ ) でも予測精度は 80% 以上であり、高精度で予測できていることが分かる。特に、  $t > 10$  の範囲では予測精度は安定して高く約 87% であった。

従来の確定的な予測方式との予測精度の違いについては、予測対象が (確定的な) 値と (確率密度) 関数という違いがあるため、直接的な比較は難しい。そこで、後述の 4 章において、確定的な予測を適用した適応配信と確率的拡散予測に基づく適応配信とで評価実験を実施し、その違いを考察する。

### 3. 確率的拡散予測に基づく適応配信方式

前章では TCP スループットの確率的拡散を 80% 以上という高い精度で予測できることを示した。本章では、その予測結果を活用した適応配信方式を提案する。

#### 3.1 映像配信システムの構成

本稿で対象とする映像配信システムの構成について説明する。映像配信システムは、図 6 に示すように、配信サーバ、IP ネットワーク、クライアントの 3 要素から構成されている。

配信サーバは、蓄積映像 (stored video) とレート制御部 (rate controller) を備える。蓄積映像は、一つのコンテンツに対して複数の映像ビットレートから一つを選択して配信できる構造となっている。具体的には、予め複数の映像ビットレートで符号化しておく方法、スケーラブルコーデックを用いた階層符号化を利用する方法、もしくはリアルタイムに所望の映像ビットレートで符号化する方法がある。レート制御部は、後述 (3.3) する制御方式にしたがって、配信する映像のビットレートを決定する。

ここで想定する IP ネットワーク (IP network) は、HTTP/TCP による通信が可能なネットワークである。特に、インターネットやモバイルネットワークなどベストエフォート型ネットワークを対象にしている。

クライアントは、再生バッファ (playout buffer) と再生部 (playout controller) を備える。再生バッファは、受信した映像データを再生するまで一時的に保存するためのバッファである。再生部は、再生バッファから映像データを取り出して再生する。このとき、再生バッファが空だと再生が停止することになる。

#### 3.2 映像配信システムの数学モデル

図 6 を参照し、対象の映像配信システムにおける映像データの流れを数学モデルで記述する。配信サーバから送出された映像データは、IP ネットワークを経由してクライアントに到着する。この映像データは、再生バッファに蓄積された後、順次再生部にて取得、再生される。以上のような映像データの流れを記述するには、クライアントにおける再生バッファの入出力関係を考えるとよい。

ある時刻  $t$  [sec] に、再生バッファに入力される映像ビットレート (video bitrate) を  $r(t)$  [bps], TCP スループット (TCP throughput) を  $x(t)$  [bps] とする。同時刻に、再生バッファに蓄積されている残余再生時間 (buffer length) を  $T_p(t)$  [sec] とする。再生バッファに蓄積されている映像データをサイズ [byte] ではなく、再生時間 [sec] で表現している点に注意されたい。また、同時刻の再生部の再生レート (playout rate) を  $p(t)$  [-] と表す。再生レートは等倍再生を 1 とした場合の倍率を表し、無次元量である。例えば、  $p = 2$  なら実時間 1[s] あたりに再生時間 2[s] 分の映像データを再生バッファから取得することになる。

筆者らは、映像データの流れを流体に近似することによって End-to-End の映像データの流れを解析する手法を提案している [5]。流体近似により、映像データの流れについて連続体・連続時間の解析が適用できるようになる。以下では、本手法に基づいて、映像データの流れを支配する方程式を導出する。

時刻  $t$  [sec] から微小時間  $\Delta t$  [sec] の間に受信した映像データサイズは  $x(t)\Delta t$  [bits] である。この映像データの映像ビットレートは  $r(t)$  [bps] であるため、  $x(t)\Delta t$  [bits] 分の映像データ

の再生時間は  $(x(t)\Delta t)/r(t)[\text{sec}]$  となる. 一方, 同時に再生部で再生した映像データの再生時間は  $p(t)\Delta t[\text{sec}]$  となる. したがって, そのときの再生バッファにおける残余再生時間の増分を  $\Delta T_p[\text{sec}]$  とすると,  $\Delta T_p = (x\Delta t)/r - p\Delta t$  と表される. ここで,  $\Delta t \rightarrow 0$  とすると, 次の微分方程式が得られる.

$$\frac{dT_p}{dt} = \frac{x}{r} - p \tag{8}$$

この微分方程式は, 視聴端末が配信サーバから送出された映像データがクライアントに到着してから再生されるまでの流れを支配している.

### 3.3 確率的拡散予測を用いたレート制御

前節で導出した映像データの流れの支配方程式と TCP スループットの確率的拡散予測を用いたレート制御を提案する. 提案方式の基本的な考え方は, TCP スループットの確率的拡散予測として式 (2) が既に得られているが, このうち  $x^-(t)$  を将来の TCP スループット予測として採用することで, TCP スループット変動に対して安全側の対策を講じることである. 例えば,  $x^-(t)$  において  $\alpha = 3$  とすることで,  $[-3\sigma, \infty]$  の範囲の変動をカバーできることになるが, これは確率で表すと 99.87% の範囲である.

将来の TCP スループットが  $x^-(t)$  であるという仮定の下では,  $T[\text{sec}]$  後の残余再生時間  $T_p(T)$  は微分方程式 (8) を次のように解くことで得られる.

$$T_p(T) = T_p(0) + \int_0^T \left( \frac{x^-}{r} - p \right) dt \tag{9}$$

ここで,  $r$  と  $p$  が  $[0, T]$  の範囲で一定であるならば<sup>(注3)</sup>, 以下の解析解が得られる.

$$T_p(T) = T_p(0) + \left( \frac{x_0}{r} - p \right) T + \frac{\mu}{2r} T^2 - \frac{2\alpha\sigma}{3r} T^{\frac{3}{2}} \tag{10}$$

提案するレート制御は,  $T[\text{sec}]$  後の残余再生時間  $T_p(T)$  を所定の目標値  $T_r[\text{sec}]$  に近づけるように制御する. 式 (11) に示すように, 目標値との誤差に所定のフィードバックゲイン  $K_P$  を乗じて加えることで映像ビットレートを更新する.

$$r \leftarrow r + K_P(T_r - T_p(T)) \tag{11}$$

この操作を制御インターバルごとに実施することで, 常に  $T[\text{sec}]$  後の将来の残余再生時間が目標値  $T_r[\text{sec}]$  以上になるように制御される.

## 4. 適応配信の評価実験

提案した適応配信方式を実装した配信サーバを開発し, 評価実験を行った. 以下で, 実験方法および実験結果について説

(注3): レート制御において  $r$  を  $[0, T]$  の範囲で一定として式 (10) の解を得たが, これは同範囲において最適な定数  $r^*$  を一つ選出するためである. 当然, より最適な  $t$  についての関数  $r^*(t) (t \in [0, T])$  が存在する可能性があるが, これを求めるのは困難であるため, 今回  $[0, T]$  の範囲で一定と近似した.



図7 蓄積映像ファイル (fragmented 3GP) 構成

表1 レート制御パラメータ

記号	値	意味
$\alpha$	2	確率的拡散の予測範囲を定める係数
$n$	30	$\sigma$ を算出するためのデータ数
$m$	10	$\mu$ を算出するためのデータ数
$T$	10	$T[\text{sec}]$ 先までを予測する
$T_r$	3	残余再生時間の目標値 [sec]
$K_P$	0.5	レート制御のフィードバックゲイン

明する.

### 4.1 実験方法

配信サーバは HTTP Servlet (Java) で実装され, クライアントに対して HTTP/TCP で蓄積映像 (フラグメント 3GP ファイル) を配信する. このとき, TCP スループットの確率的拡散予測 (2章) を実施し, その予測結果に基づいたレート制御 (3章) を行い, 該当する映像を配信する<sup>(注4)</sup>.

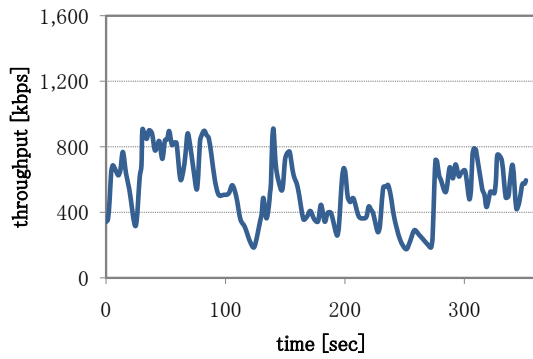
配信する蓄積映像は, 250, 350, 500, 750, 1000, 1250, 1500[kbps] の7種類のフラグメント 3GP ファイルである. 各フラグメント 3GP ファイルは 2[sec] ごとにフラグメンテーションされており, 150個のフラグメントから構成されている (図7). 配信サーバは, 一つのフラグメントを配信するごとに上述のレート制御を実施し, 次に配信するフラグメントを決定する. レート制御に必要なパラメータは表1に示した.

クライアントはフラグメント 3GP をストリーミング再生またはプログレッシブダウンロード再生できる QuickTime Player (Apple) とした. 提案する適応配信方式はクライアントに特別な処理を要求しないため, フラグメント 3GP をストリーミング再生またはプログレッシブダウンロード再生できるプレイヤーであれば任意のプレイヤーで構わない.

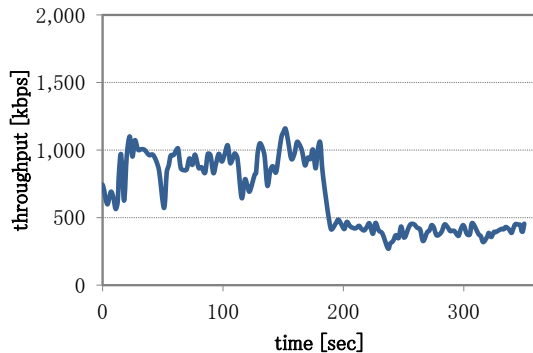
配信サーバとクライアントは有線ネットワークで接続したが, その間に TCP スループットを制御できる機能を設けた. この TCP スループット制御機能は, 任意の TCP スループットの時系列データを読み込んで, 映像配信の TCP スループットを当該時系列データに従って制限する. 本実験では, 2.3節で採集した TCP スループットデータから二つ選択して用いた (図8). どちらの TCP スループットデータもモバイル網経由で採取したデータであるため, 変動が激しくなっている.

以上の条件で, 配信サーバからクライアントへ蓄積映像を

(注4): 実際に算出されるビットレート  $r$  は連続値であるのに対し, 用意した蓄積映像のビットレートは離散値である. そのため,  $r$  を超えない最大のビットレートの映像を配信する



(a) TCP スループットデータ 1



(b) TCP スループットデータ 2

図 8 評価実験で用いた TCP スループットデータ

配信し、配信した映像のビットレート、残余再生時間を出力した。また、クライアントで再生停止した時間を計測した。

対照実験として、TCP スループット予測を行わない適応配信方式(予測なし方式と呼ぶ)と、確率的拡散予測ではなく確定的な予測を用いた適応配信方式(確定的予測方式と呼ぶ)の2方式でも同様の実験を行った。予測なし方式は、現時点でのTCP スループットを参照して適応配信方式を実施する方式である。具体的には、式(9)において  $x^-$  の代わりに  $x_0$  を代入する。一方、確定的予測方式は、将来のTCP スループットを唯一確定的に予測する。予測に用いる確率過程モデルはこの場合も Wiener 過程を採用する。Wiener 過程では平均値  $x_0 + \mu t$  で最も確率が高くなるため、この平均値を確定的予測値として採用する。すなわち、確定的予測方式では、式(9)において  $x^-$  の代わりに  $x_0 + \mu t$  を代入することになる。言い換えれば、確定的予測方式はドリフトのみを考慮して確定的に予測した方式である。

#### 4.2 実験結果と考察

##### a) 動特性の結果と考察

予測なし方式、確定的予測方式および確率的拡散予測方式(提案方式)の3方式の適応配信の結果として、TCP スループット、映像ビットレートおよび残余再生時間を図9～図11にプロットした。TCP スループットは青破線、映像ビットレートは赤実線で左軸に[kbps]で示している。残余再生時間は緑二

重線で右軸に[sec]で示している。また、左側の図(a)がTCP スループットデータ1における配信結果であり、右側の図(b)がTCP スループットデータ2のそれである。

まず、予測なし方式の結果(図9)を考察する。予測なし方式は現状のTCP スループット  $x_0$  を参照して映像ビットレートを制御するため、映像ビットレートはTCP スループットの変動に追従して変化しているように見える。しかしながら、制御間隔がフラグメント3GPファイルの1フラグメント(=2[sec]分)配信完了毎となっているため、TCP スループットの変動に追従できない。結果として、残余再生時間が0、すなわち再生停止が頻発している。当然、1フラグメントのサイズを小さくしていくと追従性能は高まることになるが、フラグメント数増加に伴う映像データのオーバーヘッド増加(注5)や映像ビットレート切り替え間隔が狭まる(=単位時間あたりの切り替え回数が増加することによる映像のちらつきが問題となる。後者の切り替え回数増加の問題については、レート制御に平滑化処理など切り替え回数が小さくなるような仕組みを付与することで解決することもできるが、別途検討が必要である。

次に、確定的予測方式の結果(図10)を分析する。確定的予測方式はドリフトを考慮した予測となっているため、TCP スループットが増加傾向もしくは減少傾向のときに予測なし方式と比較してレート制御の追従性が高くなる。その様子は図10でいくつか読み取れる。例えば、(a)の140[sec]付近のレート増加や(b)の200[sec]付近のレート減少では、予測なし方式よりも素早く増減させていることが分かる。しかし、ドリフト自体の急変には対応できず、(a)の150[sec]付近や(b)の30[sec]付近ではオーバーシュート(行き過ぎ)が発生している。その結果、再生停止回避能力については予測なし方式と大差なく、再生停止が頻発している。

最後に、確率的拡散予測方式の結果(図11)を考察する。確率的拡散予測方式は、拡散するTCP スループットの予測値として安全側の  $x^-(t)$  を採用してレート制御しているため、実験に用いられた二つのTCP スループットデータのいずれも、残余再生時間が0にならない、すなわち再生停止が発生していない。そして、再生停止が生じない範囲で、できるだけ高いレートを実現できている。

TCP スループットの変動幅が大きくなった箇所では、拡散幅  $\sigma\sqrt{t}$  が大きくなるため、反対に  $x^-(t)$  は小さくなる。その結果、レート制御は映像ビットレートを減少させて再生停止回避のため残余再生時間を増加させようとする。(a)の150～250[sec]付近、(b)の200[sec]付近などTCP スループットが大きく変動する箇所で残余再生時間が増加しているのはその

(注5)：各フラグメント先頭はIフレーム(Intra-coded Frame)でなければ映像ビットレートの切り替え時に正常に再生できないため、情報量が多いIフレームが必然的に多くなるのがデータサイズ増加に大きな影響を及ぼす。他にフラグメントのヘッダ情報が増加することも要因の一つであるが、影響としては前者の方が大きい。

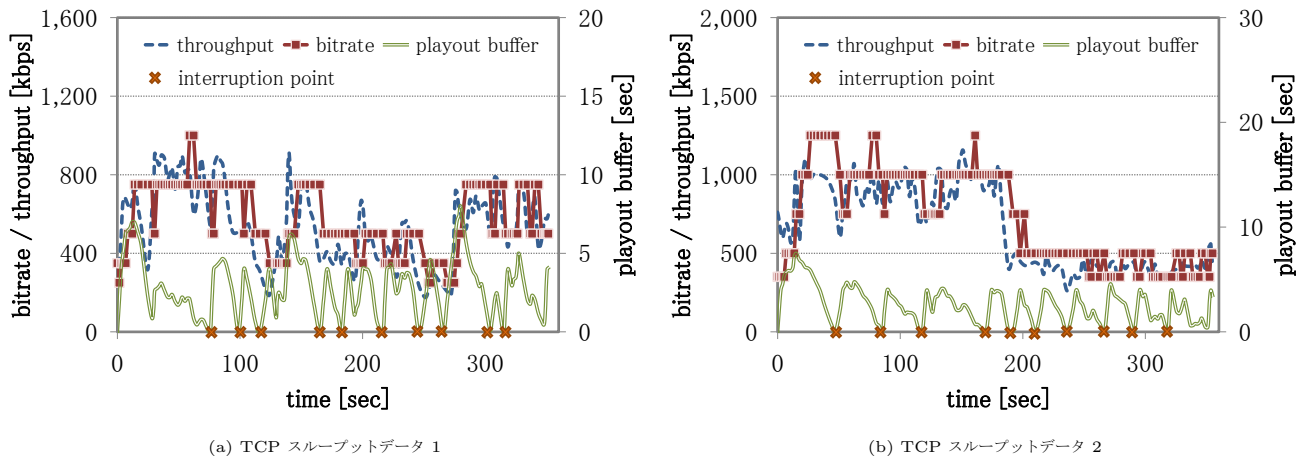


図 9 予測なし方式

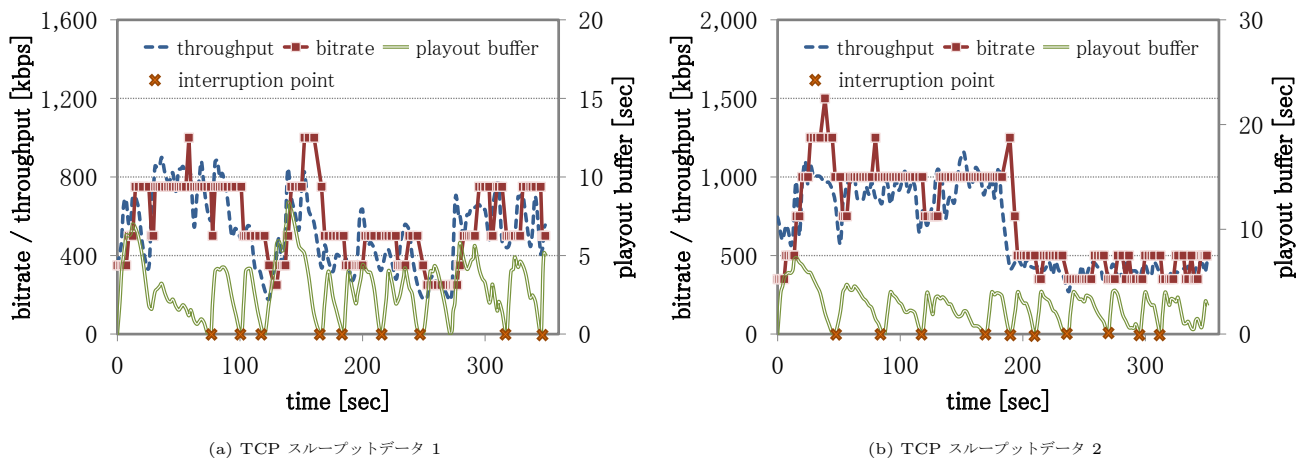


図 10 確定的予測方式

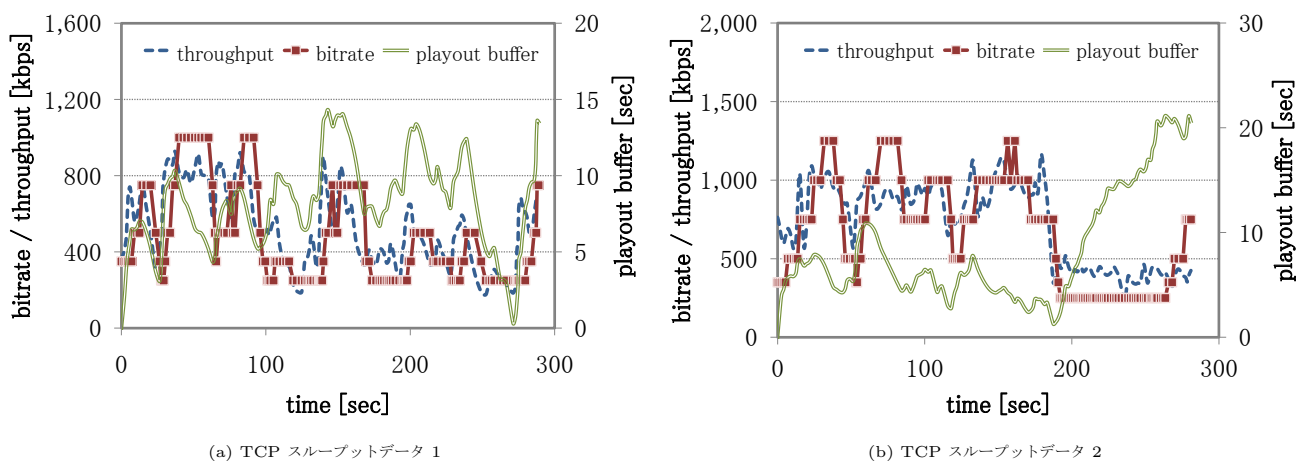


図 11 確率的拡散予測方式 (提案方式)



表 2a 適応配信結果まとめ (TCP スループットデータ 1)

	予測なし方式	確定的予測方式	確率的拡散予測方式
平均 TCP スループット	522.0 [kbps]	515.3 [kbps]	<b>510.9</b> [kbps]
平均映像ビットレート	559.2 [kbps]	547.5 [kbps]	<b>422.6</b> [kbps]
総再生停止時間	59.0 [sec]	57.5 [sec]	<b>0.0</b> [sec]

表 2b 適応配信結果まとめ (TCP スループットデータ 2)

	予測なし方式	確定的予測方式	確率的拡散予測方式
平均 TCP スループット	657.5 [kbps]	661.1 [kbps]	<b>720.3</b> [kbps]
平均映像ビットレート	717.1 [kbps]	711.4 [kbps]	<b>618.9</b> [kbps]
総再生停止時間	56.6 [sec]	51.5 [sec]	<b>0.0</b> [sec]

ためである。特に、(b) の 200[sec] 付近での TCP スループットの大幅な低下の影響は大きく、残余再生時間を増加させる方向に働いている。以上のようなレート制御の働きにより、確率的拡散予測方式は極めて高い再生停止回避能力を有する。

#### b) 大局的な結果と考察

この適応配信の結果から、平均 TCP スループット、平均映像ビットレート、総再生停止時間を計算し、表 2a, 表 2b にまとめた。各平均値は、映像を配信開始してから配信完了するまでの時間での平均である。

平均 TCP スループットが若干異なるのは、同一の TCP スループットデータを用いているにもかかわらず、後述するように映像を配信完了するまでに要した時間が異なるからである。特に、TCP スループットデータ 2 では後半に TCP スループットが低くなっているため、途中で再生停止が発生して配信完了が長引いた予測なし方式と確定的予測方式では確率的拡散予測方式よりも平均 TCP スループットが低くなっている。

平均映像ビットレートは、確率的拡散予測方式が他の 2 方式と比較して 100[kbps] 程度低くなった。これは、確率的拡散予測を用いたレート制御では、確率的拡散の下側である  $x^-(t)$  を TCP スループットの予測値として用いることで激しい変動にも対応できるようにしているためである。予測なし方式および確定的予測方式におけるレート制御は現状の TCP スループットおよびドリフトしか考慮しないため、TCP スループット変動に対応できず、TCP スループットよりも大きな映像ビットレートを選択することがあった。そのため、この 2 方式では平均映像ビットレートが高くなったが、1 分程度も再生停止が発生した。

総再生停止時間は再生停止回避能力の差を明らかにしている。確率的拡散予測方式におけるレート制御では再生停止が全く発生しなかったのに対し、他の 2 方式のレート制御は 5 分間の映像に対して、約 10 回延べ 1 分程度もの間、再生が停止している。このことは、確率的拡散予測方式のグラフ (図 11(a)) は映像配信が 300[sec] 未満で完了している (300[sec] 未満でプロットが終了している) が、他の 2 方式のグラフ (図 9(a), 図 10(a)) は約 360[sec] まで配信している (360[sec] までプロットが続いている) ことから見て取れる。

以上の結果から、今回の実験では確率的拡散予測方式は映像ビットレートを 100[kbps] 程度犠牲にすることで、再生停止を回避したということになる。確率的拡散予測方式は確率的拡散を参照してレート制御しているため、もっと安定した TCP スループットデータであれば映像ビットレート減少幅はもっと小さくなったであろうし、これ以上に変動が激しいデータならばさらに映像ビットレートを減少させて残余再生時間を確保する必要があったであろう。重要なことは、本方式はこれまでの TCP スループットの変動を鑑みて、どの程度映像ビットレートを減少させて残余再生時間を確保するかを明らかにしているということである。具体的には、式 (10) 右辺第 4 項  $\frac{2\alpha\sigma}{3r}T^{\frac{3}{2}}$  だけ残余再生時間を目標値よりも大きくとるようにレート制御することが最適であると示している。

## 5. まとめ

本稿では TCP スループットの確率的拡散を予測して適応配信に活用することで、安定した視聴を実現する適応配信システムを提案した。TCP スループットの変動を Wiener 過程でモデリングすることで、TCP スループットの確率的拡散を 80%以上という高い精度で予測できることを示した。さらに、この確率的拡散予測結果を用いることで、TCP スループット変動が激しい環境でも高い再生停止回避能力を実現できることを示した。

## 文 献

- [1] A. Zambelli, "IIS Smooth Streaming Technical Overview," <http://www.microsoft.com/downloads/>, 2009.
- [2] B. Wang, J. Kurose, P. Shenoy, and D. Towsley, "Multimedia Streaming via TCP: An Analytic Performance Study," ACM Transactions on Multimedia Computing, Communications, and Applications, Vol. 4, Issue 2, May 2008.
- [3] Adobe Systems Inc., "HTTP Dynamic Streaming on the Adobe Flash Platform," <http://www.adobe.com/products/httpdynamicstreaming/>, 2010.
- [4] Apple Inc., "HTTP Live Streaming Overview," <http://developer.apple.com/>, 2011
- [5] 吉田 裕志, 里田 浩三, "映像パケットの流体モデルに基づく映像ストリーミング制御," 信学技報, IN2008-201, Vol. 108, No. 458, pp. 411-416, 2009 年 3 月.
- [6] L. De Cicco, S. Mascolo, and V. Palmisano, "Feedback Control for Adaptive Live Video Streaming," ACM Multi-

- media Systems '11, Feb. 2011.
- [7] X. Qiu, H. Liu, D. Li, S. Zhang, D. Ghosal, and B. Mukherjee, "Optimizing HTTP-based Adaptive Video Streaming for Wireless Access Networks," IEEE International Conference on Broadband Network and Multimedia Technology 2010, pp. 838-845, Oct. 2010.
  - [8] S. Akhshabi, A. C. Begen, and C. Dovrolis, "An Experimental Evaluation of Rate-Adaptation Algorithms in Adaptive Streaming over HTTP," ACM Multimedia Systems '11, Feb. 2011.
  - [9] Q. He, C. Dovrolis, and M. Ammar, "On the Predictability of Large Transfer TCP Throughput," ACM SIGCOMM '05, Vol. 35, Issue 4, pp. 145-156, Oct. 2005.
  - [10] Q. He, C. Dovrolis, and M. Ammar, "Prediction of TCP throughput: formula-based and history-based methods," ACM SIGMETRICS '05, Vol. 33, Issue 1, pp. 388-389, June 2005.
  - [11] A. Sang and S. Li, "A Predictability Analysis of Network Traffic," INFOCOM 2000, Vol. 1, pp. 342-351, Mar. 2000.
  - [12] M. Mirza, J. Sommers, P. Barford, and X. Zhu, "A Machine Learning Approach to TCP Throughput Prediction," ACM SIGMETRICS '07, Vol. 35, Issue 1, pp. 97-108, June 2007.
  - [13] M. Mathis, J. Semke, J. Mahdavi, and T. Ott, "The Macroscopic Behavior of the TCP Congestion Avoidance Algorithm," ACM Computer Communications Review, Vol. 27, Issue 3, pp. 67-82, July 1997.
  - [14] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP throughput: A Simple Model and its Empirical Validation," ACM SIGCOMM '98, Vol. 28, Issue 4, pp. 303-314, Oct. 1998.
  - [15] T. Zhang, E. van den Berg, J. Chennikara, P. Agrawal, J. Chen, and T. Kodama, "Local Predictive Resource Reservation for Handoff in Multimedia Wireless IP Networks," IEEE Journal on Selected Areas in Communications, Vol. 19, pp. 1931-1941, Oct. 2001.