

5T-01 Diffserv ネットワーク上における リアルタイムデータ転送用の トランスポート層プロトコルの研究*

野田 陽子 坂井 達彦 古賀 佑匠 重野 寛 松下 温 †
慶應義塾大学理工学部‡

1 背景

近年のインターネットの爆発的な普及に伴い、音声や映像のようなリアルタイムデータ通信への需要が高まってきている。しかし、現在のインターネットは Best Effort であるため、リアルタイム通信に必要な QoS(Quality of Service) を実現できない。そこでそのような QoS を提供する技術として Diffserv(Differentiated Services)[1][2] が登場した。Diffserv ネットワーク上では多様な QoS を柔軟に実現できるため、リアルタイムデータ通信に必要な QoS を提供することが可能であると考えられる。

そこで本文では既存のトランスポート層プロトコルを用いてコンピュータシミュレーションによる評価を行い、Diffserv を用いたネットワーク上でリアルタイムデータ転送に適したトランスポート層プロトコルについて研究を行った。

2 背景技術

Diffserv はトラヒックに対して相対的な QoS の差別化を実現する仕組みとして登場した。Diffserv ネットワークでは、入り口ノードにおいてパケットを SLA(Service Level Agreement) で定められたクラスに分類し、ネットワーク内部では同一クラスのパケットに対して Per Hop Behavior (PHB) と呼ばれる同一の転送処理を行い、ルータで WFQ(Weighted Fair Queuing) や RED(Random Early Detection) によってクラス別に優先制御を行うことによって相対的に QoS を実現する。

3 Diffserv 上のリアルタイム通信

現在通常のデータ通信のトランスポート層プロトコルとしては信頼性を保証する TCP が、リアルタイム

データ通信には信頼性を保証しない UDP が用いられている。しかし、リアルタイムデータへの需要が高まる今日ではリアルタイムデータ通信においても信頼性を向上させる機能も必要であると考えられる。

そこでリアルタイムデータ通信を多様な QoS を実現する Diffserv を用いて行う。既存のトランスポート層プロトコルはベストエフォートなネットワークでの使用を前提としているため、Diffserv ネットワーク上で既存のトランスポート層プロトコルである UDP と TCP、TCP の SACK オプションを用いた場合の性能を評価し、Diffserv に最適なトランスポート層プロトコルについての研究を行う。

4 シミュレーション評価

4.1 シミュレーション環境

今回のシミュレーションではリアルタイムデータ通信アプリケーションを想定した。送信側アプリケーションで毎秒 N フレーム画像フレームを生成・送信し、トランスポート層で画像フレームをパケットに分割して送信し、受信側で再び画像フレームに組み立て、受信側アプリケーションで一定間隔で再生した。

シミュレーションモデルは図 1 のような単一ボトルネックモデルを用いた。Diffserv ネットワーク内部では、リアルタイムデータに対して 3Mbps の帯域を保証するものとする。

以上のような環境でトランスポート層プロトコルとして UDP と New Reno TCP を用いた場合と TCP の SACK オプションを用いた場合について性能評価を行った。

評価値としては以下のように画像フレーム再生成功率を定義し、ネットワークへのトラヒック流入量を変化させて評価を行った。

$$\text{再生成功率} = \frac{\text{再生成功フレーム数}}{\text{総再生フレーム数}}$$

ただし、再生成功の条件としては受信側アプリケー

* A summary of transport layer protocol for transmitting real-time data over Diffserv network

† Yoko Noda, Tatsuhiko Sakai, Yuzo Koga, Hiroshi Shigeno, Yutaka Matsushita

‡ Faculty of Science and Technology, Keio University

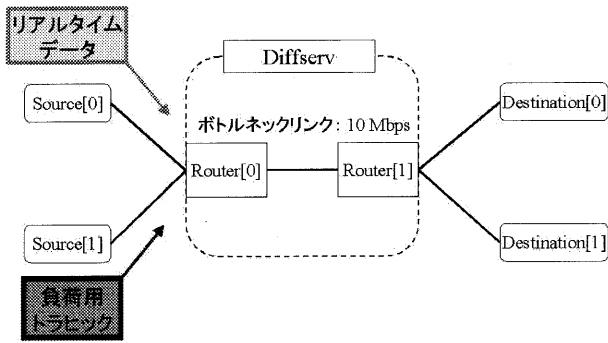


図 1: シミュレーションモデル

シミュレーションでの一定間隔の再生までに再生画像フレームを構成するパケットが全て到着することとする。

4.2 予想される挙動

UDP は輻輳に対してフロー制御や、損失に対して再送を行わないため、トラフィック流入量が保証レートを上回ると輻輳によるパケット損失が生じ、性能が低下すると考えられる。

TCP はウィンドウ制御によってトラフィック流入量を決定し、一時的にトラフィック流入量が生成レートを上回るようなバースト的に送信がおこり、それが保証レートを上回った場合、バースト的な損失が発生すると予想される。TCP は損失パケットを再送することで損失回復を行い、再送中も送信側が許せば損失パケット以降のデータを送信するが、損失回復には時間がかかるという問題が生じると予想される。また損失回復後のフロー制御によって送信側での遅延が生じ、遅延が累積し、再生に間にあわなくなるパケットが増加するという問題が生じると予想される。

TCP の SACK オプションを用いるとバースト的な損失を効率的に回復することができるため、性能が向上すると予想される。

5 シミュレーション結果

トラフィック流入量と再生成功率の関係を図 2 に示す。

トラフィック流入量が Diffserv によって保証されたレートを下回る場合、いずれも 100%送信されるが、保証レートを上回る場合とパケット損失が発生する。UDP の場合、損失に対して反応しないため性能が悪化する。TCP の場合、損失に対して再送処理とフロー制御処理を行い、それらの処理が Diffserv によって保証された帯域によって効果的に働くために UDP よりも良い性能を示すが、前述した問題のため性能に限界があるこ

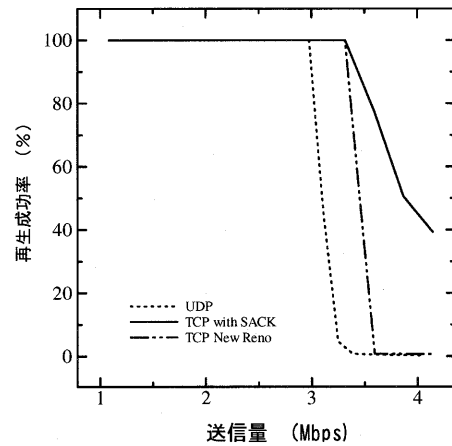


図 2: トラフィック流入量と再生成功率の関係

とがわかる。TCP の SACK オプションを用いた場合は、SACK オプションによってバースト的な損失を効率良く速やかに回復することができるため、性能が向上することがわかる。

6 まとめ

本文では、Diffserv を用いて帯域を保証するネットワークにおける、既存のトランスポート層プロトコルを用いたリアルタイムデータの転送の性能評価を行った。保証帯域以上のデータ転送を行った場合は、信頼性を保証する TCP の方が UDP よりもリアルタイムデータ転送に適していることがわかった。しかし TCP はバースト損失や、送信側フロー制御による遅延の問題によって、性能の向上には限界があることが分かった。バースト損失については TCP の SACK オプションを用いて改善が可能であることがわかった。以上の結果をふまえて、前述した問題を解決し、Diffserv 上でのリアルタイムデータ転送により適したトランスポート層プロトコルを研究していくのが今後の課題である。

参考文献

- [1] IETF."Definition of the Differentiated Service Field (DS Field) in the IPv4 and IPv6 Headers", RFC-2474, Dec,1998
- [2] IETF."An Architecture for Differentiated Service", RFC-2475, Dec,1999