

# 遅延を考慮した適応 Diffserv スケジューラと バッファ管理アルゴリズムの提案

後藤 渉<sup>†</sup> 保岡 裕<sup>†</sup> 甲藤 二郎<sup>†</sup>

<sup>†</sup>早稲田大学大学院理工学研究科 〒169-8555 東京都新宿区大久保 3-4-1

E-mail: <sup>†</sup>{goto,yasuoka,katto}@katto.comm.waseda.ac.jp

**あらまし** Diffserv ルータでは、IP ヘッダのクラス情報に基づくスケジューリングを行い、サービスの階層化を実現している。しかし、特定のクラスにフローが集中した場合、既存の単純に棄却するアルゴリズムでは、アプリケーションの要求を十分に満足しているとは言い難い。本稿では、WRR 方式と WRR の重みを制御する方式を組み合わせ、両者を輻輳状況に応じて切り替えるスケジューリングアルゴリズム、ならびにバッファ管理アルゴリズムを提案し、シミュレーションによって提案方式の有効性を明らかにする。

**キーワード** Diffserv スケジューリング WRR バッファ管理

## A Proposal on an Adaptive Diffserv Scheduler with Delay Constraints and its Buffer Management Algorithm

Wataru GOTO<sup>†</sup> Yutaka YASUOKA<sup>†</sup> Jiro KATTO<sup>†</sup>

<sup>†</sup> Graduate School of Science and Engineering, Waseda University

3-4-1 Okubo, Shinjyuku-ku, Tokyo, 169-8555 Japan

E-mail: <sup>†</sup>{goto,yasuoka,katto}@katto.comm.waseda.ac.jp

**Abstract** Diffserv routers carry out packet classification and scheduling according to IP header information and provide service differentiation. However, when a flow concentrates on a specific class, current algorithms simply drop packets of the overloaded class, and application demands would not be satisfied. In this paper, we therefore propose a hybrid packet scheduling algorithm, in which WRR and weight-controlled WRR are switched according to congestion condition. We also propose buffer management algorithm, and validity of our proposal is clarified by computer simulations.

**Keyword** Diffserv Scheduling WRR Buffer Management

### 1. はじめに

近年、VoIP や映像配信などリアルタイム、準リアルタイム IP アプリケーションが企業網や地域網で数多く使用されるようになってきた。しかし、様々なトラフィックが混在するベストエフォート網においては、遅延やジッタ、パケット廃棄などによる音声や映像の品質劣化が生じる可能性がある。そのような背景の中、インターネット上を流れる様々なデータの種類に応じたサービスを提供するために、ネットワーク QoS(Quality of Service)を保証する必要が出てきた。そこで、IETF(Internet Engineering Task Force)において、QoS を保証するための技術として Diffserv(Differentiated Services)[1]技術が標準化されている。

Diffserv は、フローが流入するエッジルータにおいて、EF(Expedited Forwarding)[2]、AF(Assured

Forwarding)1-4[3]、BE(Best Effort)の 6 つのサービスクラスにフローを集約し、クラス毎の優先制御を行なうことでスケラビリティを確保し、QoS をクラス単位で保証するサービスである。これにより、バックボーンなどのフローが集中するところでも、あらかじめ決められた QoS を提供することが可能になる。また、AF クラスには 4 つのクラス (AF1 ~ AF4)、3 段階の廃棄優先度 (Green、Yellow、Red) を設定することができる。これにより、主に Class-based の相対的な優先制御と最低帯域保証を実現することが可能となる。

しかしながら、ある特定のクラスにフローが集中してしまうと、従来のスケジューリング方式、バッファ管理アルゴリズムでは、そのクラスの要求を十分に満たしているとは言い難く、十分なサービス品質を満たすことが出来なくなってしまう。

そこで、本稿では、AF クラスにおいて、遅延を考慮

したスケジューリング方式と、バッファ管理アルゴリズムを提案し、シミュレーションを行い、その結果についての考察を行なった。

## 2. Diffserv における優先制御

Diffserv での優先制御(図 1)において、クラス単位のスケジューリングを可能とする方式として WRR(Weighted Round Robin)[4]、WFQ(Weighted Fair Queuing)[5]等が一般的に用いられている。また、キュー内の滞在時間を用いたスケジューリング方式として WTP(Waiting Time Priority)[6]という方式も提案されている。また、バッファ管理として一般的に用いられるものとして、Drop Tail、RED(Random Early Detection)[7]等が知られている。

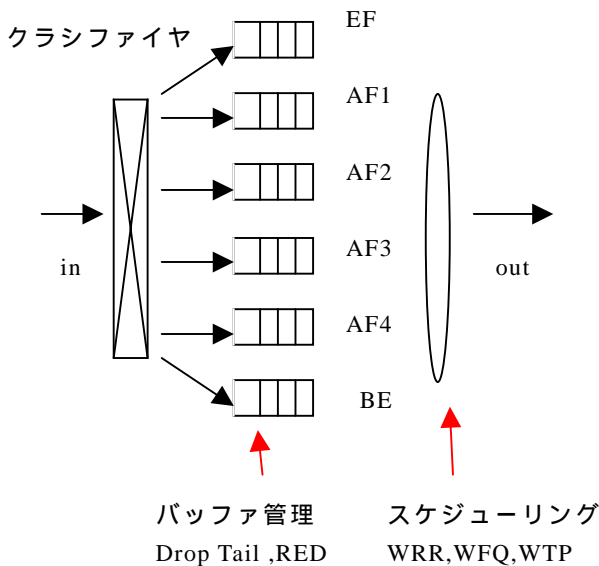


図 1 Diffserv における優先制御

従来方式の問題点として、WRR や WFQ では、ある一つのキューにパケットが偏ると、優先順位が逆転してしまうことがある。遅延においても厳密に保証されていない。また、WTP においてもネットワーク状況に優先度が左右されすぎて、意図した優先度で送出できないという問題点がある。

そこで、WRR とは別に遅延に関する優先度を設定し、最大遅延値の保証として、各キュー内に滞在した時間が閾値を上回った場合、優先的にパケットの送信を行なうスケジューリング方式を提案した[8]。これにより、遅延を考慮したスケジューリングが可能になる。

しかし、この方式のみでは、あるクラスの遅延品質は向上するが、その結果として、他のクラスのサービス品質劣化への影響が大きくなってしまおうという問題が生じる。

そこで、他のクラスに対しての品質劣化の影響を緩和するため、このスケジューリング方式と併用して、キュー内の平均滞在時間を基にしたバッファ管理アルゴリズムの提案を行う。以下に、提案するスケジューリング方式、バッファ管理アルゴリズムについての詳細を述べる。

## 3. 提案方式

### 3.1 スケジューリング方式

提案方式では、通常は WRR であらかじめ定めた重み付けによるスケジューリングを行なうが、閾値を越えた場合はキュー内の平均滞在時間を基にして WRR の重みを動的に制御するスケジューリング方式に切り替わる。

パケットが到着した時、以下のとおりに優先制御を行なう。

STEP1 各キューに対し、平均滞在時間  $avg\_time$  を求める。

この  $avg\_time$  はキューに入った時間からキューの先頭にくるまでの待ち時間を滞在時間とし、ローパスフィルタにより平均時間を見積もる。このフィルタの時定数は  $weight\ factor\ w$  によって制限させ、以下の式で表される。

$$avg\_time = (1-w) * avg\_time + w * current\_time$$

STEP2 各クラスに設定された閾値  $max\_th\_sch$ 、 $min\_th\_sch$  を  $avg\_time$  と比較する。このときあるクラスにおいて、以下の条件分岐に従ってクラスの重みを変化させる。

if ( $avg\_time < min\_th\_sch$ )

WRR によって定められた重みに従ってパケットの転送を行なう。

else if ( $min\_th\_sch < avg\_time < max\_th\_sch$ )

そのクラスの重みを増やす。

else ( $avg\_time > max\_th\_sch$ )

限界まで重みを増やす。

ここで、WRR が各クラスにサービスを行なう周期を 1 ラウンドとし、計測間隔は 1 ラウンドごとに行なうこととする。また、高優先クラスのみが連続にサービスされないようにサービスされるまでの時間をラウンド間隔時間とし、その間隔が長くなった場合、優先的

に当該クラスをサービスする。

### 3.2 バッファ管理

次に、提案するバッファ管理アルゴリズムについて述べる。

Diffserv においては、Green、Yellow、Red にマーキングされたパケットに対して、それぞれ異なるパラメータをもつ RED を実装することが多いが、本稿では 3.1 項のスケジューリング方式と併用することを考慮し、RED における平均キュー長の最小閾値、最大閾値の代わりに、Green、Yellow、Red にマーキングされたパケットに対し、キュー内の平均滞在時間の最小閾値、最大閾値、廃棄確率を設定し、廃棄を行なうバッファ管理方式を提案する。3.1 項における STEP 2 において、スケジューリングが提案方式に切り替わったとき、以下のようにしてバッファ管理を行なう。

STEP1 各キューに対し、平均滞在時間 `avg_time` を求める。

STEP2 各クラスに設定された閾値 `max_th_buf`、`min_th_buf` を `avg_time` と比較する。このときあるクラスにおいて、以下の条件分岐に従ってパケットのエンキュー、廃棄を決定する。

```
if (avg_time < min_th_buf)
    パケットをエンキューする。
else if (min_th_buf < avg_time < max_th_buf)
    そのクラスに設定された廃棄確率に従いパケットを廃棄する。
else (avg_time > max_th_buf)
    パケットを廃棄する。
```

上記のようにして、各キュー内のパケットの平均滞在時間を計測し、ある閾値を越えたら優先的なスケジューリングを行い、それと同時に、遅延のジッタを抑えるため、キュー内の平均滞在時間に基づいた廃棄を行う。この方式によりフローが集中したクラスに対してだけではなく、他のクラスに対しても遅延のジッタを抑えることが可能となる。

### 4. シミュレーション評価

本稿では ns-2 (network simulator ver.2)[9]を用いたシミュレーションによる定量評価を行った。トラフィックのクラスとして Diffserv の DSCP(Diffserv Code Point)を利用し、最優先クラス(AF1)、中優先クラス

(AF2)、低優先クラス(AF3)の3つのクラスを定義する。各ノードから UDP パケットを送信し、各ノードからのパケットがエッジルータに到着したとき、最優先クラス(AF1)にパケットが偏った場合を想定したシミュレーションを行なった。シミュレーショントポロジは図 2 に示す。

また、提案方式との比較を行なうため、WRR でスケジューリングを行い、バッファ管理として RED を用いたときのシミュレーションも行なった。

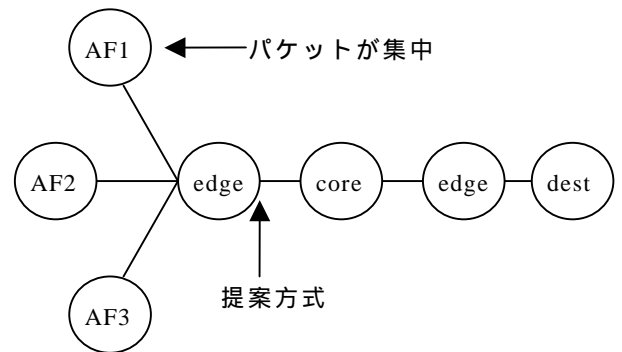


図 2 シミュレーショントポロジー

### 5. シミュレーション結果

従来方式(WRR+RED)、提案方式の各々について、トラフィックが集中する入力側のエッジルータにおいて、各クラスのキュー内平均滞在時間、パケット廃棄率を求めた。各クラスのキュー内平均滞在時間を図 3、4 に、パケット廃棄率を図 5、6 に示す。

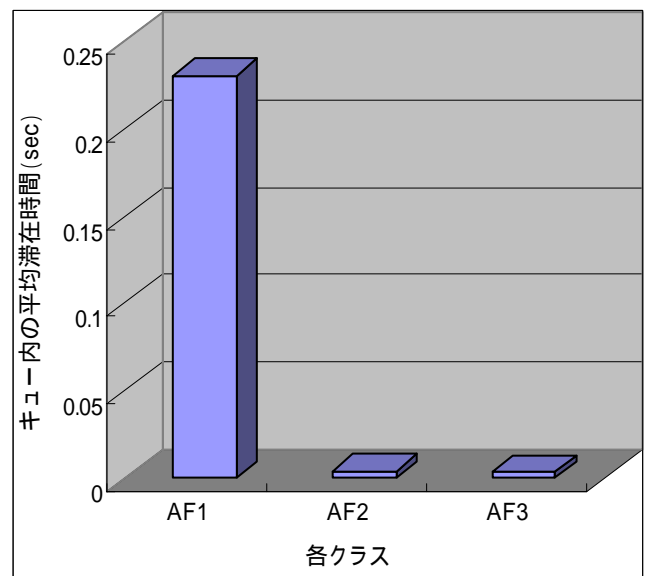


図 3 従来方式

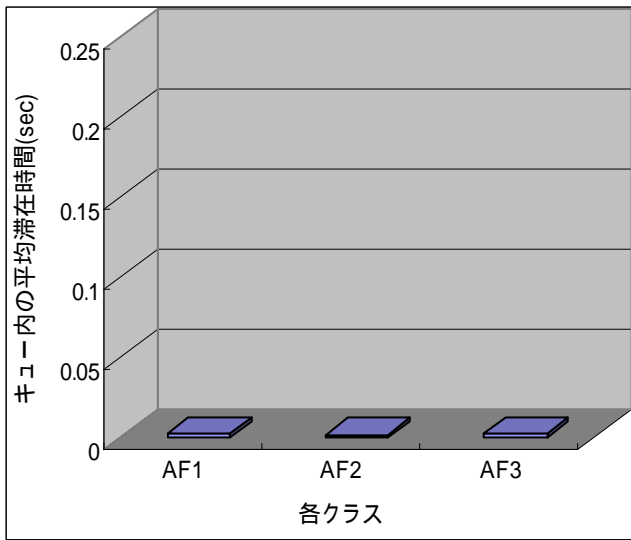


図4 提案方式

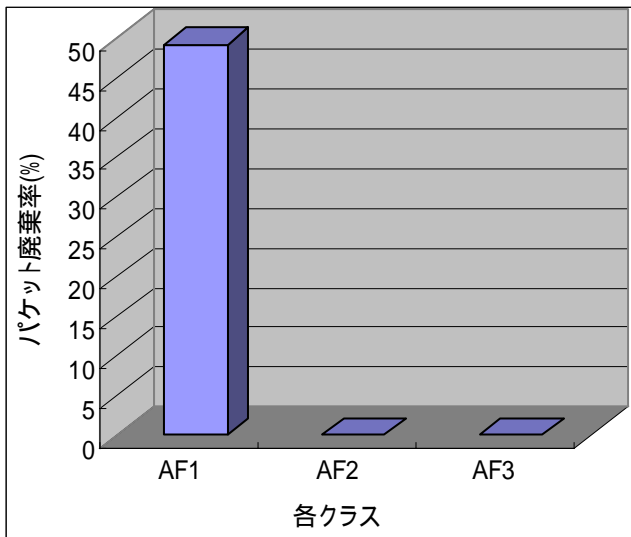


図5 従来方式

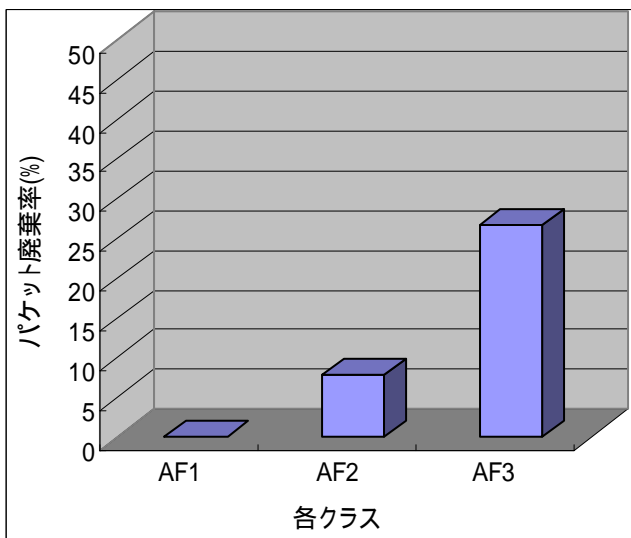


図6 提案方式

シミュレーション結果より、従来方式においては、トラフィックが集中しているAF1クラスにおいてキュー内の平均滞在時間が増加しているのに対し、提案方式では、AF1クラスにおいて平均滞在時間が抑えられ、他のクラスに対してもキュー内の平均滞在時間が小さくなり、ジッタが抑えられていることがわかる。これは、バッファ管理により、キュー内において滞在時間が大きいパケットが廃棄されるためであると考えられる。

パケット廃棄率に関しては、従来方式においては、トラフィックが集中しているAF1クラスのみが突出して増加してしまっているが、提案方式においては、各クラスのパケット廃棄率の相対的な関係が保たれていることがわかる。

## 6. まとめ

本稿において、遅延を考慮した Diffserv スケジューラ、またそれと併用して、ジッタを抑えるためのバッファ管理アルゴリズムの提案を行い、従来方式との比較検討を行なった。結果から、ルータにおいて遅延、ジッタを制御することができ、提案方式の有効性を示すことができた。

## 文 献

- [1] S.Blake, D.Black, M.Carlson, E.Davies, Z.Wang, W.Weiss, An Architecture for Differentiated Services, RFC2475, Dec.1998.
- [2] V.Jacobson, K.Nichols, K.Poduri, An Expedited Forwarding PHB, RFC2598, June. 1999.
- [3] J.Heinonen, F.Baker, W.Weiss, J.Wroclawski, Assured Forwarding PHB Group, RFC2597, June.1999.
- [4] M. Katevenis, S. Sidiropoulos, C.Courcoubetis, Weighted round-robin cell multiplexing in a general-purpose ATM switch chip, IEEE Journal on Selected Areas in Communications, October 1991.
- [5] S. J. Golestani, A Self-Clocked Fair Queueing Scheme for Broadband Applications, Proc. IEEE INFOCOM, pp. 636--646, 1994.
- [6] C.Dovrolis, P.Bhagwat, A Case for Relative Differentiated Services and the Propotional Differentiation Model, IEEE Network, Sep.1999.
- [7] S.Floyd, V.Jacobson, Random Early Detection gateways for Congestion Avoidance, IEEE/ACM Transactions on Networking, August 1993.
- [8] 後藤 涉, 甲藤 二郎, 遅延を考慮した適応型 Diffserv 型スケジューラの検討, 2002 年電子情報通信学会ソサイエティ大会.B-7-78.
- [9] Network Simulator-ns (ver.2), <http://www.isi.edu/nanam/ns>, UCB/LBN/VINT.