オーディオビジュアル複合情報処理　35－25
(2001. 12. 14)

ポリシング付き優先度キュー(PPQ)方式によるQoS保証可能なルータ装置

松古　典夫† 武藤　大† 福富　昌司† 藤川　賢治‡
藤本　義人‡ 岡　敏生* 太田　昌孝**

†古河電気工業株式会社
〒254-0016 平塚市東八幡 5-1-9
‡京都大学大学院情報学研究科
〒606-0016 京都市左京区吉田本町
*東京大学大学院新領域創成科学研究科
**東京工業大学理工学研究科
〒152-8552 目黒区大岡山 2-12-1

あらまし 本稿ではインターネット上でフローごとのQoS保証を実現するルータ上でのポリシング付き優先度キュー(Policed Priority Queuing, PPQ)を提案し、動画伝送などリアルタイム性を要求されるアプリケーションの品質を確保するために有効であることをルータへの実装、検証結果と共に報告する。

キーワード QoS保証、ポリシング、優先度キューイング、リアルタイムアプリケーション

A router providing QoS guarantees using policed priority queueing (PPQ)

Norio MATSUFURU†, Dai MUTOU‡, Shouji FUKUTOMI, Kenji FUJIKAWA‡
Yoshito HUJIMOTO‡, Toshio OKA*, Masataka OHTA**

†Furukawa Electric Co. Ltd.,
Higashi-Yahata 5-1-9, Hiratsuka-shi, Kanagawa, 254-0016 Japan
‡Graduate School of Informatics, Kyoto University
Yoshidahonmachi, Sakyouku, Kyoto,606--8501 Japan
*Graduate School of Frontier Science, University of Tokyo
**Graduate School of Information Science and Engineering, Tokyo Institute of Technology

Abstract In this paper, we propose a queueing method, called policed priority queueing (PPQ), which provides QoS guarantees for each flow on the Internet. We implemented PPQ in a router and evaluate its performance. Our results show that PPQ is quite effective to ensure the QoS of real-time applications, such as voice and TV conferences.

Key words QoS guarantees, policing, priority queueing, real-time application

1. はじめに

近年のインターネットでは VoIP、TV 会議等のリアルタイムアプリケーションが普及してきている。
これらのアプリケーションを快適に用いるために、遅延時間や帯域などの Quality of Service (QoS) を保証したデータ転送サービスへの要求が高まっている。
現在のインターネットではベストエフォト型と呼ばれる QoS 保証のない通信方式が一般的だが、最近では QoS 保証を行なう通信サービスも提供されるようになってきている。QoS 保証型通信サービスを提供するには、ユーザが QoS 要求を行なうためのしくみに加えて、要求があった時に網側でそれを受け付けるかどうかを判断する制御、QoS 要求を受理したデータフローに対してネットワーク資源を割り当て、要求された QoS を満たすようにパケットのフォワーディングを行なうしくみ等が必要になる。
本稿では、これらの中で特にユーザからの QoS 要求を満たすようにパケットをフォワーディングするしくみについて検討を行ない、筆者らの提案している Policed Priority Queueing (PPQ) と呼ばれる方式について、その有効性とルータへの実装について報告する。

2. 従来の技術と問題点

要求された QoS を保証しつつフォワーディングを行うしくみはバッファアクセス制御とスケジューリングにより実現される。典型的な方法では、あるパケットフロー(例えば同一の送信元、宛先アドレスを持つパケットの集合)に対して固定長のバッファ容量を持つキューを作成し、各キューから一定の割合でパケット送信する。
特に各キューから一定割合でパケット送信する機能はパケットスケジューリングと呼ばれ、Weighted Fair Queueing (WFQ) を始めとして非常に多くの方式が提案されている[4][5][6][7][8]。
これら既存の方式のほとんどは以下の処理ステップを踏む。

1. 装置(ルータ)へのパケット到着時に、各方式固有のあるアルゴリズムに従ってタイムスタンプを計算し、パケットに付けてキューに入力する。
2. パケット送信時には各キューの先頭パケットについているタイムスタンプを見て,それが最小なパケットを送信する (図 1)。

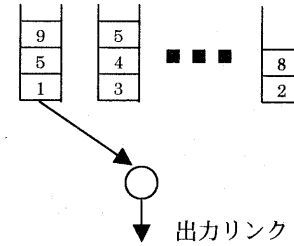

図 1. タイムスタンプによるスケジューリング

ここでタイムスタンプは原則として 1 パケット到着することに 1/r_i づつ増加していく(r_i はそのパケットが属するフロー i に割り当てられた帯域)。このような方式により、各フローに対してパケットバッファと帯域を割り当てる。本稿ではこれらのタイムスタンプベースのスケジューリング方式をフェアキューイングと呼ぶこととする。
フェアキューイングを用いると以下のような QoS を保証することが可能であることが示されている[10]。
パケットフロー i のトラフィック特性が平均帯域 r_i, バケツ深さ σ_i のリーキーバケツに従う時、エンドツーエンドの最大遅延時間 D_i は以下の式(1)で表される。

$$D_i = (\sigma_i + (k-1) L_i)/r_i + k L_{max}/C \quad 式(1)$$

$$K_i = \sigma_i + nL_i \quad 式(2)$$

また、式(2) はｎホップ目のルータにおいてフロー i のキューがあふれることによるパケット損失をなくすために必要なバッファ容量である。
ここで、k は経路のホップ数を表し, L_i, L_max はそれぞれフロー i の最大パケット長、リンクの最大パケット長であり、C はリンク速度を表す。
上記の 2 式よりフェアキューイングにより QoS 保証を実現する際の特性として以下のことが挙げられる。

1. 保証できる最大遅延時間は各フロー i に割り当てられた帯域 r_i にほぼ反比例し、リンク速度にはほとんど依存しない。
2. キューあふれをゼロにするためには各フロー毎に σ_i + ホップ数程度のバッファ容量が経路上の各ルータで必要となる。

現実のリアルタイムアプリケーションには、例えば VoIP のように必要帯域は小さいが厳しい遅延時間を要求するものや、必要帯域は大きいが遅延要求は

ゆるいものなど様々なものが存在するが、上記 1. の特性は、フェアキューイングではこれらの要求に対して効率的な解を与えることが困難であることを示している。
また、上記 2. で述べたような、フロー数に比例したバッファ容量を各ルータ上で確保することは事実上困難である(例えばフロー当たり 16 k バイトとしてもわずか 1000 フローで 16 M バイト)。
また、フェアキューイングを実行するためにはタイムスタンプのソーティングが不可欠である。一般にパケット到着時にソーティングで発生する処理量は、フローの数を N とすると log N となる。
従って、処理コストに関しても大量のフローをサポートするのは難しい面がある。これらの問題を解決するために近年多くの研究が行われているが、根本的な解決策は得られていない。
これに対して、筆者の一人である太田らのグループによりフェアキューイングとは全く異なったポリシーに基づく方式 Policed Priority Queueing(PPQ)が提案されている[1][2][3]。

3. Policed Priority Queueing

PPQ 方式と PPQ に基づいて構成されるサービスである Confortable Service (CS) の特徴を以下に示す[1][2][3].

a. ユーザに提供する通信サービスはベストエフォートと QoS 保証の 2 種類。
b. QoS 保証サービスでは全てのパケットフローに対して一律のパケットロス率、遅延時間を提供する.(個々のフロー帯域はユーザが設定できる)。
c. ルータに到着するパケットフローのトラフィック特性を Poisson 型と仮定し、それを満たすようにポリシングをかける。
d. QoS 保証サービスに属するパケットは単一のキューに入力され、ベストエフォートサービスに属するトラフィックよりも常に先に出力される。

次に PPQ 方式の動作を図 2 を用いて説明する。

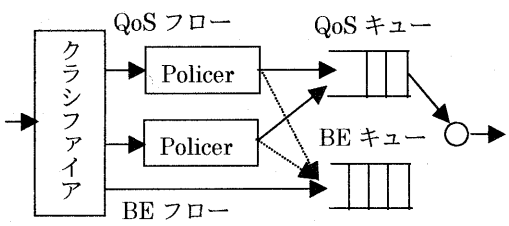

図 2 PPQ 方式

1. ルータの出力キューに到着したパケットは、クラシファイアにより QoS サービスを要求するフロー(QoS フロー)とベストエフォトサービスを要求するフロー(ベストエフォトフロー)に分類される。
2. ルータには各出力ポート毎に QoS キューと Best Effort (BE)キューの 2 つのキューが用意されており、ベストエフォトフローに関しては無条件に BE キューに入力される。
3. 一方、QoS フローは図の Policer によってポリシングされ、要求されるトラフィックパターン(Poisson 到着)を満たすパケットだけが QoS キューに入力され、残りは Best Effort キューに収容される。
4. パケット送信に際しては QoS キューにパケットがたまっている時は常にそちらを先に出力し、QoS キューにパケットがない時だけ BE キューからパケットが送信される。

以上の動作より、PPQ では以下の特性が得られる。

1) パケット到着時に必要な処理はフローの数 N に無関係な一定値である。
2) ルータで必要とされるバッファ容量もフローの数 N に無関係な一定値である。
3) QoS 保証サービスの最大遅延時間は QoS キューの最大キュー長(bit)を出力リンク速度(bps)で割った値で得られる.。
   各フロー i が予約した帯域 r_i とは無関係である。
4) 出力リンク上に予約できる QoS 保証サービス全体のレートをリンク帯域よりも少し小さな値に制御することで、各フローが互いに独立である条件の元に、パケットがキューあふれで落ちる確率がエラーで落ちる確率程度に小さくできることが M/D/1/K 待ち行列の解析等から示されている[2]。

これらの特性は従来のフェアキューイングでは得られないものであり、フェアキューイングがかかえる問題が PPQ では発生しないことが分かる。

4. ルータへの実装

前節で示した PPQ の有効性を確かめるため、PPQ のルータへの実装を行った。本節では実装の詳細について述べ、次節でそのパフォーマンス測定結果を示す。実装に用いたルータのハードウェア構成の概略図を図 3 に示す。

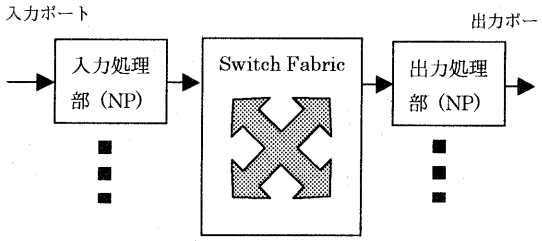

図3 ルータアーキテクチャ

実装を行ったルータは Network Processor(NP)による高速プロトコル処理が可能であり、スイッチファブリックはハードウェアスケジューリング機構をもっている.。

ルータで PPQ を実現するモジュールは、1)パケットクラシファイア、2)ポリサー、3)スケジューラの 3 つの機能ブロックに分けられる.

4.1 パケットクラシファイア

ギガビットを越える高速リンクを収容するため、パケットクラシファイアでの処理には高速性が要求される。これを満たすため、クラシファイアは高速 CAM を用いて実現した。

CAM 内の各エントリには QoS 保証フローに対応する送信元 IP アドレス、宛先 IP アドレス、送信元ポート番号、宛先ポート番号、プロトコル番号等の組が登録されている。

ルータに到着したパケットのヘッダ情報から送信元 IP アドレス情報等を抽出し、CAM で検索をかける。ヒットするエントリが CAM に存在すれば、それは QoS フローに属するパケットであると判断し、検索の結果得られたアドレスからポリシングに必要な情報を取得する (図 4)。

クラシファイアは図 3 の入力処理部で実装した。

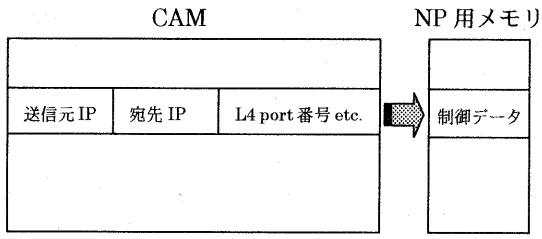

図4. クラシファイアのアーキテクチャ

4.2 ポリサー

ポリサーにおいては各 QoS フローの到着パターンが、要求した平均帯域を持つ Poisson 型トラフィックであるかどうかの検定を行う必要がある。

実際には厳密な検定を行うことは実装の観点から困難であるため、文献 [1] ではパラメータの異なるリーキーバケツを多段に組み合わせる方法が提案されている。

われわれの実装ではリーキーバケツを 2 段組み合わせてポリサーを構成した。

高速リンクをサポートするためにはポリサーにおいても高速処理が必要とされる。PPQ ではルータにおけるパケットフォワーディングにかかる処理コストが、利用帯域(bps)よりもむしろ単位当たりの処理パケット数(pps)に依存することに着目し、単位時間あたりの許容パケット数(pps)によりポリシングを行うことが提案されている。

この場合、パケット長は固定とみなすことができ、ATM スイッチの実装においてよく知られた Virtual Scheduling Algorithm [11](図 5) が利用できる。

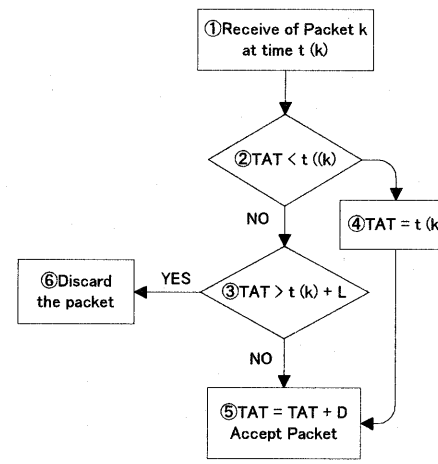

図 5. Virtual Scheduling Algorithm

図で, L , D はリーキーバケツパラメータより計算される

Virtual Scheduling Algorithm を使えば, 各パケット到着時に Theoritical Arrival Time (TAT) と呼ばれる変数を更新していくだけで容易にリーキーバケツを構成できる。2 段のリーキーバケツを構成する場合も、各フロー毎に L と D のパラメータを 2 つづつと TAT を 2 つ保持するメモリ領域を用意するだけで良く、数千規模のフローにも容易に対応できる。

ポリサーのルータへの実装は、ルータのフォワーディングエンジンに含まれるネットワークプロセッサ(NP)にマイクロコードと呼ばれるアセンブラコードを作成することで行った。
ポリシングにかかった処理ステップ数は約 30 命令となった。ポリサーは図 3 の入力処理部で実装した。

4.3 スケジューラ

PPQ 方式では各ポート毎にキューを 2 つしか用意する必要がなく、送信パケット選択も固定優先であるため、キューイング部分については特別な工夫を必要とすることなくハードウェアにより容易に実現できる。QoS フローの最大遅延時間を決めるキュー長についてはルータの設定により可変にした。

5. 性能評価

本節では実装したルータの性能評価を行う。まず最初に具体的なアプリケーションとして 30Mbps の帯域を持つ動画像データの配送を行い、PPQ 処理なしとありの場合のパケットロス率の差を測定し,PPQ 方式により QoS 保証が実現できていることを示す。次に実装したルータにおいて PPQ 処理ありの場合となしの場合でスループットを測定し、PPQ の実装コストに関する評価を行う.

5.1 動画像配信

PPQ の QoS 保証性能を評価するため、図 6 の環境を構築して動画像配信を行った。

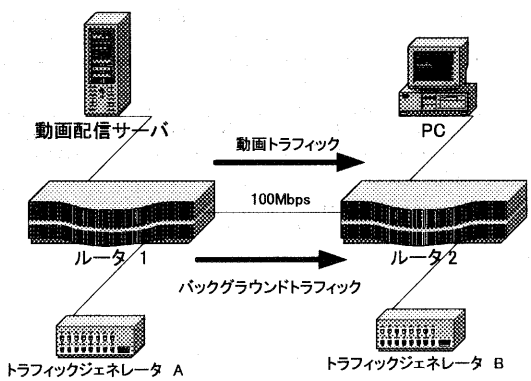

図 6. 実験環境

図 6 において、全てのノードは 100Mbps の Fast Ethernet で結ばれており,ルータ 1 とルータ 2 には PPQ が実装されている。 ここで、動画像サーバから PC へ向けて 30Mbps (パケット長 538 bytes, 7520 pps) の動画トラフィックが配信される。
一方、トラフィックジェネレータ A から B に向けてバックグラウンドトラフィック(パケット長 128bytes)をベストエフォトで送信する。
従って、ルータ 1 とルータ 2 を結ぶ 100M のリンクは動画とバックグラウンドのトラフィックにより共有されることになる。
図のような環境で、トラフィックジェネレータから送信されるバックグラウンドトラフィックをリンク速度の 100%近くまで増加させた場合に, 動画トラフィックがベストエフォトサービスで転送される場合と、PPQ を用いて QoS 保証サービスで転送される場合のパケットロス率の比較を図 7 に示す。
QoS 保証サービスの場合の PPQ のポリサーのパラメータは 1 段目のリーキーバケツレートを 7600pps、バケツ深さを 100 (パケット)、2 段目についてはそれぞれ 8000 pps と 10 に設定した。
図から分かるように、ベストエフォトサービス(QoS なし)の場合はバックグラウンドトラフィックが増加するにつれて動画データはほとんど転送されなくなるのに対して、PPQ を用いて QoS 保証を行った場合では、バックグラウンドトラフィックが 100%の帯域でデータを流そうとした時でも動画トラフィックが優先して送信され、パケットロスが発生しないことが分かる.

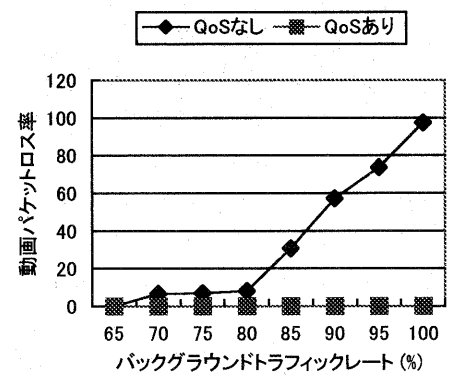

図 7 動画像トラフィックパケットロス率

5.2 実装コストの評価

次に、PPQ 方式の処理コストを評価するため、100Mbps の Fast Ethernet リンクと 1000Base

SX の Gigabit Ethernet リンクに対して、PPQ 処理を行わずにパケット中継を行った場合と PPQ 処理を行いながらパケット中継を行った場合のスループットを測定した。

測定結果を図 8 と 図 9 にそれぞれ示す。

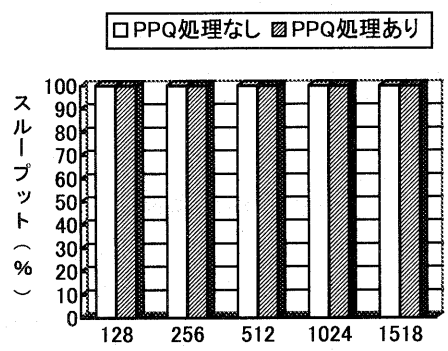

図8. FastEthernet(100Mbps)のスループット比較

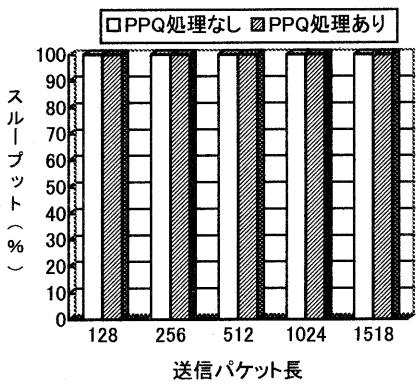

図9. GigaEthernet(1Gbps)のスループット比較

図 8. から分かるように Fast Ethernet (100Mbps) のリンク上において PPQ 処理を行った場合でも行わなかった場合と同じく 100%のスループットを達成できた。

また、図 9 では Gigabit Ethernet (1Gbs)リンク上で同様な測定結果が示されており、Gigabit クラスの高速リンクにおいても PPQ を用いた QoS 保証をワイヤレートで実現できたことが分かる。

6. まとめ

本稿ではインターネット上で QoS 保証された通信を実現するための PPQ 方式について、主に実装の観点から評価を行った。PPQ 方式はリアルタイムアプリケーションに対して QoS 保証通信を提供でき、ルータにおいて低コストで実装できることを示した.

文　献

[1] 岡 敏生, 藤川 賢治, 岡部 寿男, 池田 克夫, ``優先度付きキューを用いた確率的 QoS 保証におけるポリーシング法,'' DICOMO2001, pp.531--536, June, 2001.

[2] 藤川 賢治, 藤本 義人, 岡部 寿男, 太田 昌孝, 池田 克夫, ``Policed Priority Queuing (PPQ) を用いたフローごとの QoS 保証のための Comfortable Service (CS) の提案,'' DICOMO2001, pp.537--542, June, 2001

[3] 岡 敏生, 藤川 賢治, 池田 克夫, ``QoS 保証のための PPQ における多段トークンバケットを用いたポリーシング手法,'' 第 60 回情報処理学会全国大会, March 2010

[4] Abhay K. Parekh, and Robert G. Gallager, ``A generalized processor sharing approach to flow control in integrated services networks: The single-node case,'' IEEE/ACM Trans. Networking, vol.1, no.3, pp.344--357, 1993.

[5] Abhay K. Parekh, and Robert G. Gallager, ``A generalized processor sharing approach to flow control in integrated services networks: The multiple node case,'' IEEE/ACM Trans. Networking, vol.2, no.2, pp.137--150, 1994.

[6] L. Zhang, ``VirtualClock: A new traffic control algorithm for packet switching networks,'' ACM. Trans. Comput. Syst., vol.9, pp.101--124, 1991.

[7] S. Jamaloddin Golestani, ``A self-clocked fair queueing scheme for broadband applications,'' in Proc.IEEE INFOCOM'94, pp.636--646, 1994.

[8] Jon C. R. Bennett and Hui Zhang, ``Hierarchical packet fair queueing algorithms,'' IEEE/ACM Trans. Networking, vol.5, no.5, pp.675--689, 1997.

[9] Dimitrios Stiliadis and Anujan Varma, ``Efficient fair queueing algorithms for packet-switched networks'' IEEE/ACM Trans. Networking, vol.6, no.2, pp.175--185, 1998.

[10] Dimitrios Stiliadis and Anujan Varma, ``Latency-rate servers: A general model for analysis of traffic scheduling algorithms,'' in Proc. IEEE INFOCOM'96, pp.111--119, 1996.

[11] The ATM Forum Technical Comittee, ``Traffic management specification version 4.0'', April 1996.

盛光印刷所