

TCP親和性を持つ受信者駆動型階層化 マルチキャストとレート制御に関する一検討

山口 一郎 本間 健一 甲藤 二郎

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: {yamaguchi, homma, katto}@katto.comm.waseda.ac.jp

あらまし 本研究では、TCP親和性のある受信者駆動型階層化マルチキャストの実現を目的として、最悪受信環境下にある代表受信者を選び、送信側と定期的に情報を交換することによって適切かつ安定した送信レートを階層ごとに決定するという手法により、受信側の利用可能帯域に対して柔軟に対応し、TCP親和性のあるデータ配信を提案する。さらに、実験によって、提案方式の有効性を定量的に明らかにする。

キーワード マルチキャスト 階層化マルチキャスト RLM TCPフレンドリ、Network Simulator

A Study on Receiver-Driven Layered Multicast with TCP-friendly rate control

Ichiro YAMAGUCHI Kenichi HOMMA Jiro KATTO

Graduate School of Science and Engineering, Waseda University 3-4-1 Ohkubo, Shinjuku-ku, Tokyo, 169-8555 Japan

E-mail: {yamaguchi, homma, katto}@katto.comm.waseda.ac.jp

Abstract In this paper, we propose a novel Receiver-Driven Layered Multicast approach with TCP-friendly rate control. A sender adaptively selects a receiver that is assumed to be the worst receiving environment and determines an appropriate and smooth sending rate for every layer according to feedback information from the receiver. The sending rate is updated with a newly developed TCP-friendly rate control method, which avoids link underutilization. Finally, validity of our proposal is quantitatively clarified by computer simulations.

Keyword Multicast, Layered Multicast, RLM, TCP-Friendly, Network Simulator

1. はじめに

近年、インターネットにおいて、音声や映像のようなマルチメディアのデータをタイムリーに、かつ確実にどのユーザにも配信するという研究が行われている。そしてユーザはそのようなストリームデータを手軽に利用できるようになり、その利用は増加の一途をたどっている。また、リアルタイムのデータ通信の必要性に加えてアプリケーションも急速に発展している。そ

のため、これまでの1対1の通信だけでなく、1対Nの通信の需要が今後ますます高まると考えられる。

このような同報的な情報の配信に関しては、ネットワーク資源の有効利用の点からもマルチキャストと呼ばれる技術が重用視されている。マルチキャストでは、配信途中のルータでパケットをコピーする仕組みによって、受信者が複数存在する場合でも送信者が送るパケットの数は1つで済み、帯域を有効利用することが可

能となる配信技術である。

しかし通常のマルチキャストでは、そのマルチキャストグループに参加しているどの受信者に対しても送信者の送信レートは一定であるために、利用可能な帯域幅や端末の処理速度の違いなど、受信者の異質性に対応できないといった問題が生じる。このネットワークの多様性に対応するためのひとつのアプローチとして、各受信者が各自の利用可能帯域に応じて受信レートを決定することのできる RLM (Receiver-driven Layered Multicast) [1]と呼ばれる受信者駆動型の階層化マルチキャスト方式が提案されている。

しかし、受信者駆動型階層化マルチキャスト通信の実用化のためには、インターネットトラフィックの約9割を占めるTCP通信と親和性のあるTCPフレンドリな輻輳制御を実現することが重要である[2]。本研究では、TCP親和性のある受信者駆動型階層化マルチキャストの実現を目的とする。

以下、2章では受信者駆動型階層化マルチキャストとTCPのスループットモデル化についての研究背景を説明する。3章では、提案方式について説明し、4章においてその有効性をシミュレーションにより検討する。最後に5章でまとめを行う。

2. 研究背景

2.1 受信者駆動型階層化マルチキャスト

一般にマルチキャスト通信において、各受信者のネットワーク環境が必ずしも均一である保証はない。例えば電話回線とモデムを経由した数十 kbps 程度の回線を通じてマルチキャストグループに参加しているユーザもいれば、高速なLAN経由で数百 Mbps を利用できる環境にあるユーザもいると考えられる。また、ユーザ数の多いネットワーク内においては、ネットワークで輻輳が生じることによってマルチキャストデータを受け取るために十分な帯域を利用できない場合も考えられる。

そこで、受信者の異質性に対応して効率的なマルチキャスト通信を行うための方式として、階層化マルチキャストが提案されている。

階層化マルチキャストでは、送信者は階層符号化を用いて送信するストリームデータを基本階層と複数の拡張階層に階層的に符号化する。階層符号化では基本階層とより上位の拡張階層を合わせて復号するほど高い品質が得られる。送信者はこれらの階層化されたデータに異なるマルチキャストアドレスを割り当てて送信する。受信者は、各自で推定した利用可能帯域に合わせて基本階層といくつかの拡張階層を受信し、復号する。こうして各受信者の異質性に対応したマルチキャスト配信を行うことが可能となる。

ここで、階層化マルチキャストにはネットワーク支援型と受信者駆動型の手法があるが、実装のしやすさを考慮して、後者の手法が用いられることが多く、本稿ではこちらの手法を用いる。

受信者駆動型の階層化マルチキャスト方式であるRLMプロトコルにおいて、送信側は動的な役割をせず、単純にマルチキャストグループの各レイヤを別々に送信するだけである。それに対して各受信者は、まだ使用可能な帯域がある場合は受信するレイヤを増やすためにグループにjoinし、ボトルネックで輻輳が生じた場合には受信するレイヤを落とすためにleaveを行う。こうして各受信者はそれぞれ最適な受信レベルを探していく。

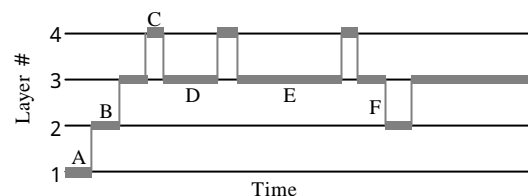


図1: RLMでの受信レート制御

この受信レートの決定方法について詳しく説明する。図1に、最高4つのレイヤを受信する場合でのRLMプロトコルの受信レート制御の様子を示す。

まず始めに、受信者は基本階層であるレイヤ1に登録を行いjoin-timerをセットする(A)。このjoin-timerは、動的に変化する使用帯域を調査するためのタイマーである。そして、join-timerの期限が切れたらすぐに受信者はレイヤ2を受信し、また別のjoin-timerをセットする(B)。このように自発的に受信するレイヤ数を増やすことによって最適な受信レートを決定する方法をjoin-experimentと呼ぶ。

こうしてこのjoin-experimentを繰り返し、レイヤ4の受信を行った時点で輻輳が生じたと想定する(C)。このとき、パケットロスが生じると、受信者はこれにより輻輳を検知する。そして直ちにレイヤ4の受信を行わないようにし、レイヤ4のjoin-timerの値を指数的に増加させる。そして暫くして、レイヤ4のjoin-timerの期限が切れると(D)、再びレイヤ4の受信を行う。しかし、またそこで輻輳が生じた場合、join-timerの値は更に増加される(E)。その後リンクに別の輻輳が生じて受信するレイヤが3から2に落ちたとすると(F)、その時はレイヤ3のjoin-timerの値はまだ小さく設定されているので、すぐに復帰してレイヤ3の受信を行うことができる。

join-experimentとその結果を正確にリンクさせるために、join-experimentによるレイヤ数の変更がネットワーク内で確立されるまでにどれくらいの時間が必要か、受信者がjoin-experimentの結果を得るのにどれく

らの期間が必要か、ということを決める必要がある。この期間を detection-time と呼ぶ。もし detection-time の間に輻輳が起らなかった場合、join-experiment が成功したとみなし、逆にもしこの detection-time の間に輻輳が検出された場合は、join-experiment が失敗したとみなしてそのレイヤの join-timer を増加させる。

この動作を繰り返すことによって RLM プロトコルは受信者の最適な受信レートを決定する。

2.2 TCPスループットのモデル化

TCPのレート制御では、ウィンドウフロー制御が用いられていて、ウィンドウサイズをネットワークの負荷に応じて変更する。以下にTCP Renoのウィンドウサイズ更新アルゴリズムを示す。

下記のようにウィンドウサイズが変更され、送信側がパケットロスを検知するとウィンドウサイズを半分にする。パケットロス検知の方法としては、2通りあって、1つ目は、Triple Duplicate ACKによるもので、2つ目はタイムアウトによるものである。以下で、Triple Duplicate ACKのみによるパケットロス検知と、Triple Duplicate ACKとタイムアウトによるパケットロス検知の2通りの場合でのスループットのモデル化について説明する。これより以下では、Triple Duplicate ACKをTDA、タイムアウトをTOとする[5]。

ACK 到着毎に		
<u>Slow Start Phase</u>		
$wnd = wnd + 1 (ssth > wnd)$		
<u>Congestion Avoidance Phase</u>		
$wnd = wnd + 1/wnd (ssth < wnd)$		
<u>Detect Packet Loss</u>		
$wnd = wnd / 2$		
この時、 $ssth = ssth / 2$ と変更する		
wnd : 輻輳ウィンドウ		
$ssth$: Slow Start Threshold		

TDAのみによるロス検知

TCP Renoのウィンドウサイズの変化は、上記に示している通り、スロースタートフェーズでは、ACK受信毎にウィンドウサイズを1増加させ、輻輳回避フェーズでは、1/cwndずつ増加させる。

スループットのモデル式は、廃棄が発生する間に送られるパケット数の平均値を求めることにより、スループットのモデル式が算出され、

$$B = \frac{\frac{1-p}{p} + \frac{2+b}{3b} + \sqrt{\frac{8(1-p)}{3bp} + \left(\frac{2+b}{3b}\right)^2}}{RTT \left(\frac{2+b}{6} + \sqrt{\frac{2b(1-p)}{3p} + \left(\frac{2+b}{6}\right)^2} + 1 \right)} \approx \frac{1}{RTT} \sqrt{\frac{3}{2bp}} + o(1/\sqrt{p}) \quad (1)$$

となる。ただし、Bはスループット、pはロス率、RTTはラウンドトリップタイム、bは1回のACKを送信側に返す時に確認するデータパケット数である。

TDAおよびTOによるロス検知

TOが発生すると、まず、輻輳ウィンドウを1に戻して再送を行う。その間に別のパケットに対するタイムアウトが発生すると、最初のタイムアウト時間をTOとすると、2TOとなり、さらにタイムアウトが発生すると、4TOとなる。このように指数関数的にタイムアウト時間は増加していき、64TOまで増加する。タイムアウトは、同一ウィンドウ内で複数のパケットロスが発生すると起こる。この場合のスループットモデル式は以下のようになる。

$$B \approx \frac{1}{RTT \sqrt{\frac{2bp}{3}} + TO \min \left(1, 3 \sqrt{\frac{3bp}{8}} \right) p (1 + 32p^2)} \quad (2)$$

実際のTCP-Friendly Rate Control Protocolでは、上式(2)が使われている[6]。

3. TCP親和性を持つ受信者駆動型階層化マルチキャスト

3.1 従来手法における問題点

複数の受信者に確実にデータを配信することを目的とした信頼性マルチキャストプロトコル (Reliable Multicast) では、TCPフレンドリな輻輳制御の実現を目的とした提案方式がいくつか発表されている[3][4]。これらの提案方式は、複数の受信者と送信者の間で測定されたRTT、ロス率からスループットを算出して『最も受信状態の悪い受信者』を選択し、その最悪受信者の状態変化に追従してTCPフレンドリな輻輳制御を行う。対して受信者駆動型階層化マルチキャストでは、送信するストリームデータを基本階層と複数の拡張階層に階層的に符号化して送信し、受信者はネットワークの状況に適した階層数を動的に選択し、受信することで輻輳制御を行っているが、このメカニズムは先のTCPフレンドリな信頼性マルチキャストプロトコルで説明したRTT測定は用いられておらず、TCPと親和性のあるプロトコルとはいえない。

本研究では、各受信者が利用可能な帯域のうち、実際に利用されていない帯域の全受信者での総和を最小

に保つことを目的とする。

3.2 提案方式

本提案方式では、TCPフレンドリな受信者駆動型階層化マルチキャストの実現のために、送信者側でレート制御を行う。送信レートの制御は送信者と、受信者の中でも最悪環境下にある代表受信者との間で定期的に情報を交換することによって決定する。また、この代表受信者は、階層毎に決定し、階層別にレート制御を行う。

3.2.1 最悪環境下にある受信者の決定方法

最悪環境下にある受信者の決定方法について、図2を用いて説明する。

まず、受信者がパケットロスを検知すると、受信者は受信者のアドレス、受信階層数等の情報を送信者に伝え、RTT測定を開始する。

そして送信者はその情報を受け取ると、その受信者アドレスに対して測定パケットを送り返す。このとき、送信者が持っている現在の最悪受信者の情報、G値を付加する。このG値は、 $RTT\sqrt{p}$ とする。

この情報を受け取った受信者は、測定したRTT値とロス率により、自分のG値を先程と同様に計算する。そして、もし送信者から送られたG値よりも、その値が大きければ、代表受信者変更要求メッセージを送信者に送る。送信者は、この代表受信者と定期的に測定パケットを交換しあうことで、現在のG値を更新し、同時にレート制御を行う。G値は送信者と代表受信者との間で一定時間に移動平均を用いて更新する。

しかしスケラビリティの面から、パケットロス検知の度に送信者に対して通知メッセージを送るのは送信者側の負荷が大きくなり過ぎてしまうため、受信者は一定間隔にランダム時間待った後、通知メッセージを送信するものとする。

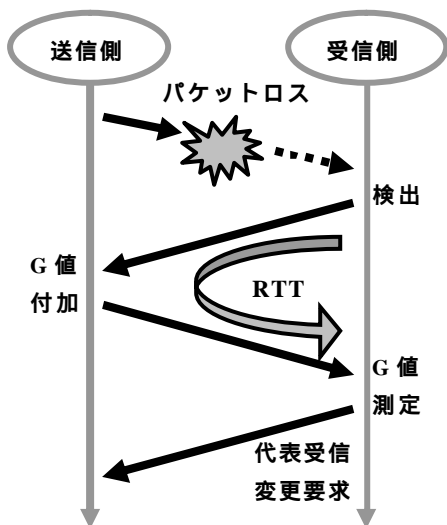


図2: 最悪環境下にある受信者の決定方法

3.2.2 レート制御アルゴリズム

定常状態におけるTCPのスループットでは、標準的なTCPであるRenoバージョンのウィンドウサイズの変化を対象とし、モデル化を行う。モデル化では、高速再送メカニズムの振る舞いとタイムアウトによる影響が考慮されている。

送信レート制御は送信者と階層毎に設定された代表受信者との間で行われる情報交換によって行う。このとき、レート制御には以下のようなアルゴリズムを用いる。

$$\begin{aligned}
 & \text{if } (p = 0) \\
 & \quad Rate_n = Rate_c + packetsize / RTT; \\
 & \text{if } (p > 0) \\
 & \quad Rate_n = \frac{packetsize}{RTT \sqrt{\frac{2bp}{3}} + T_o \min\left(1, 3\sqrt{\frac{3bp}{8}}\right) p(1 + 32p^2)};
 \end{aligned}$$

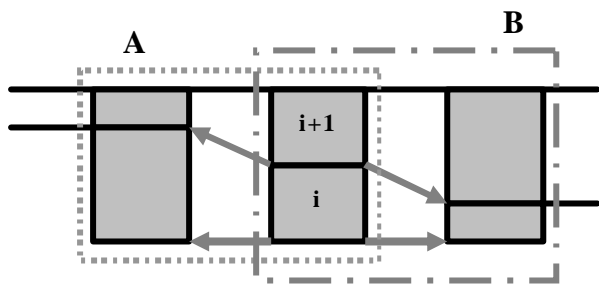
p : ロス率
 $Rate_c$: 現在の送信レート
 $Rate_n$: 新しい送信レート
 T_o : タイムアウト時間

受信者は、パケットロスを検知すると送信側にロス率 p と RTT をレポートとしてフィードバックする。送信者は受け取ったロス率 p と RTT の移動平均を求めパラメータとして使う。レート更新を行う点は、従来のTCP-Friendly方式と同方式であるが、パケットロスが発生していない場合は、 $p=0$ となり、利用可能帯域が十分にあると判断して送信レートを1RTT毎に1パケット増加するようにしている。

パケットロスが発生していない、つまりロス率 $p=0$ となった場合、従来のユニキャストを前提としたTCP-Friendly方式では、TCP Renoのスロースタートフェーズと同様に、受信レポートが返ってくる毎に送信レートを2倍にする[7]。しかし、この方式をそのまま本提案に適用した場合、送信レートが急激に増大してしまいロスが発生しやすくなる。従って、受信レイヤ数が減少する可能性が高くなる。そこで、前者に比べて送信レートを緩やかに増加させる手法として、TCP Renoの輻輳回避フェーズと同等の制御を行うこととした。

各レイヤ間でのレート制御方法について図3を用いて説明する。レイヤ i が前述のレート制御アルゴリズムによって送信レートを増加させた場合(図3(A))、レイヤ $i+1$ の送信レートを、減少させる。同様に、レイヤ i が送信レートを減少させた場合(図3(B))、レイヤ $i+1$ の送信レートを、減少させる。こうして、下位レイヤでのレート変動が上位レイヤの送信レートに影響を与えてしまうのを防ぎ、各レイヤで独立して

レート制御を行う。



: レイヤ $i+1$ までの送信レート
 : レイヤ i までの送信レート

図 3: 各レイヤ間でのレート制御

4. シミュレーション評価

ns(network simulator)[7]を用いてシミュレーションを行い、従来方式と提案方式の性能を比較する。

4.1 シミュレーション条件

以下の図 4 に示すように、3 つの受信環境を設定してシミュレーションを行っている。4 階層は、64,128,192,256kbps としている。送信開始時は、4 階層とも 64kbps である。シミュレーション時間は 1000sec で、TCP との親和性を比較する実験では、500~1000sec において送信側と 150kb のリンクの受信者側の間で TCP コネクションをはる。

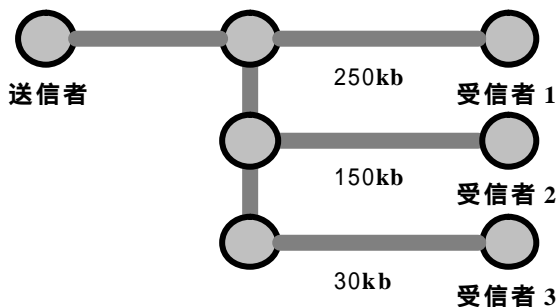


図 4: ネットワークトポロジ

4.2 リンク速度への適応性

以下の図 5、図 6 にはそれぞれ従来方式と提案方式のグッドプットを示す。従来方式では、レート制御を行っていないので 64kbps を受信できる環境にない 30kb のリンクと接続している受信者は、64kbps の送信情報を受信できない。それに対して、提案方式ではその問題が解消され、受信者のリンク速度に適応できている。インターネットのような様々な受信環境が想定されるネットワークにおいて柔軟な対応ができることがわかり、有効性が認められる。

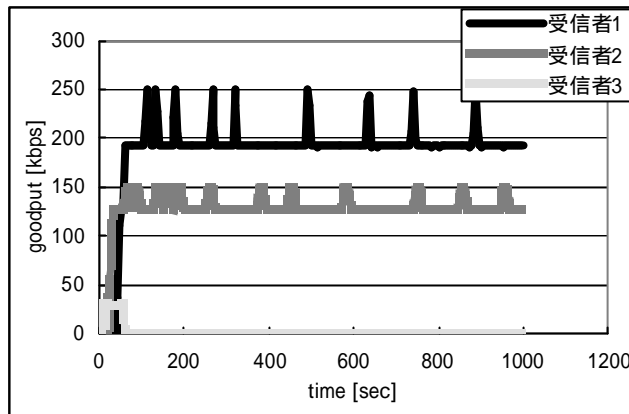


図 5: 従来手法

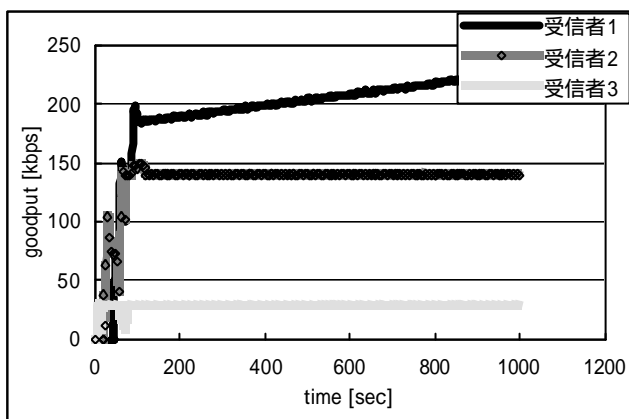


図 6: 提案手法

4.3 TCP 親和性

以下の図 7、図 8 にはそれぞれ TCP フローを流したときの従来手法と提案手法のグッドプットを示す。従来方式では、レート制御を行っていないので、帯域は UDP によって支配されることになり、TCP との親和性が認められない。それに対して、提案手法では、TCP のスループットとほぼ一致していて TCP と親和性があることが認められる。これによって UDP と競合したときに TCP トラフィックのスループット低下という問題が解消されることとなる。

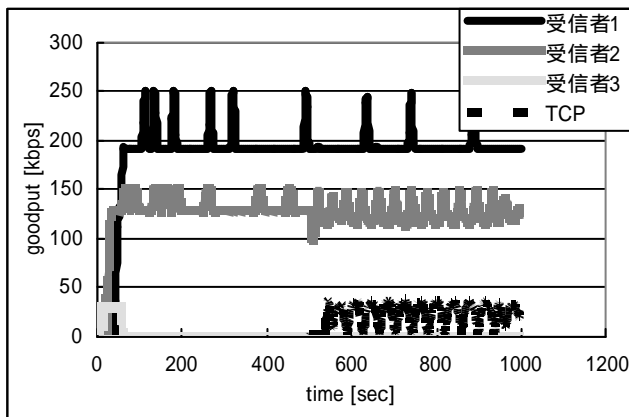


図 7: 従来手法

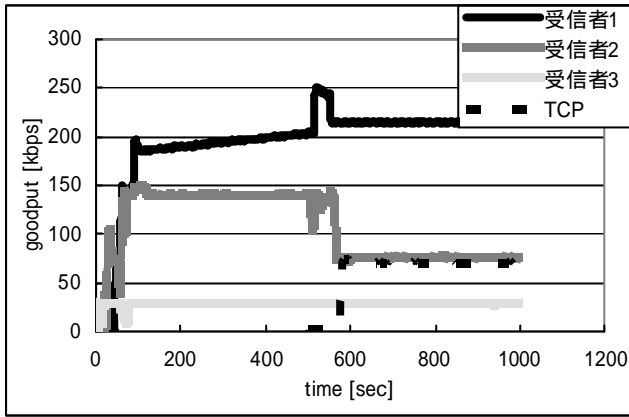


図 8: 提案手法

4.4 フィードバックレポートの送信間隔とオーバーヘッド

送信者と代表受信者との情報交換のタイミングとして、ロス検知を行うたびにフィードバックレポートを返す手法、受信状況を観測して定期的にレポートを返す場合等が考えられる。ここでは、受信者がロスを検知してから5秒間受信状況を確認してフィードバックレポートを返す場合について実験を行った。以下の図8、図9にそれぞれ提案手法と送信者のオーバーヘッドの結果を示す。

フィードバックレポートを返す頻度を少なく抑えることにより、スケラビリティを実現し、送信側の処理を軽減することが可能となる。図8と図9の対比として、図9ではレートの変動が大きい部分も見られるが、スループットの著しい低下も見られることなく、TCPとの親和性が保たれているので有効性が認められる。

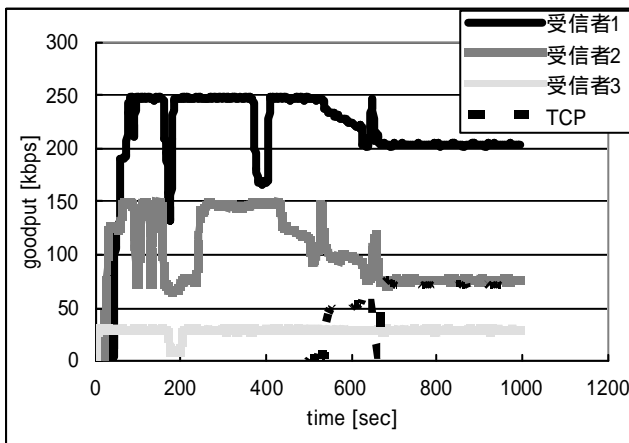
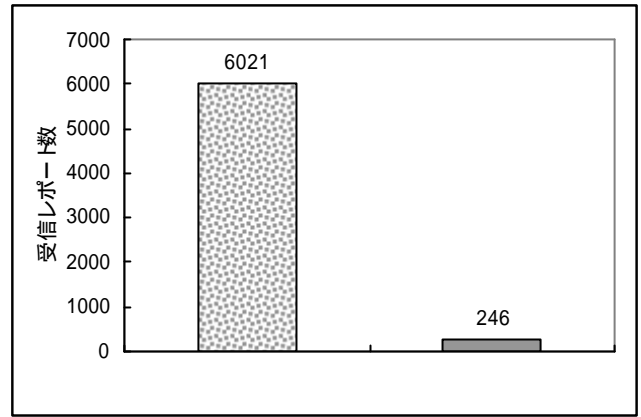


図 9: 提案手法 2(ロスを検知して5、秒間受信状況を観測してフィードバックを送信する場合)



ロス検知毎にレポートを送信する場合
ロス検知をして5秒間受信状況を観測してレポートを送信する場合

図 10: 送信者のオーバーヘッド

5. まとめ

本稿では RLM プロトコルを改良し、最悪受信環境下にある代表受信者を選び、送信側と定期的に情報を交換することによって適切かつ安定した送信レートを階層ごとに決定するという手法を用いることにより、受信側の利用可能帯域に対して柔軟に対応し、TCP 親和性のあるデータ配信を提案した。そしてシミュレーションによる従来方式との評価によって、提案方式の有効性を示すことができた。

参考文献

- [1] S.McCanne, V.Jacobson and M.Vetterli, "Receiver-driven Layered Multicast", ACM SIGCOMM'96
- [2] L. Vicisano, L. Rizzo, J. Crowcroft, "TCP-like congestion control for layered multicast data transfer", INFOCOMM'98
- [3] 山本和徳、山本幹, "TCP 親和性を持つ信頼性マルチキャスト輻輳制御方式", 信学技報 IN, 2001.2
- [4] 森谷優貴、渥美幸雄, "信頼性マルチキャストにおける輻輳制御の検討", 信学技報 IN, 2001.7
- [5] J.Padhye, V.Firoiu, D.Towsley, J.Kurose, "Modeling TCP Throughput: A Simple Model and its Empirical Validation", SIGCOMM98
- [6] S.Floyd, M.Handley, J.Padhye, J.Widmer, "Equation-Based Congestion Control for Unicast Applications", SIGCOMM2000
- [7] M.Handlley, J.Padhye, S.Floyd, "INTERNET-DRAFT draft-ietf-tsvwg-tfrc-04.txt", 27.April 2002
- [8] "Network Simulator - ns (version 2)", <http://www.isi.edu/nsnam/ns,UCB/LBNL/VINT>