

適応型キャッシュを用いた リライアブルマルチキャストプロトコル

田島 佳武 森川 博之 青山 友紀

東京大学工学系研究科

〒113-8656 東京都文京区本郷7丁目3番1号

Tel: 03-3812-2111 (ext.6710) Fax: 03-3818-5706 E-mail: tajima@mlab.t.u-tokyo.ac.jp

あらまし: 本稿では適応型キャッシュを用いたリライアブルマルチキャスト RMAC (Reliable Multicast using Adaptive Caching) のシステムならびにプロトコルの詳細を示し, その性能評価について述べる. リライアブルマルチキャストを効率的に支援するためには, ネットワーク内での機能的な処理が必須であるとの考えに基づき, RMAC では機能なルータを導入する. RMAC によれば, ルータでパケットのキャッシング・再送を行うことで再送効率の向上を図ることが可能である. また, キャッシングをネットワークの状態に応じて動的, 分散的に行うことで, 必要となるキャッシュ容量を低減することも可能である. シミュレーションによる性能評価により, RMAC は従来の方式に比べ, 受信ホスト数が多い場合も分散キャッシングにより再送遅延を低減させ, スケーラビリティを向上させることがわかった.

キーワード: リライアブルマルチキャスト, アクティブネットワーク, 適応型キャッシュ, ネットワークサービス, インターネット

Reliable Multicast using Adaptive Caching

Yoshitake TAJIMA, Hiroyuki MORIKAWA, and Tomonori AOYAMA

School of Engineering, The University of Tokyo

7-3-1, Hongo, Bunkyo-ku, Tokyo 113-8656, JAPAN

Tel: +81-3-3812-2111 (ext.6710) Fax: +81-3-3818-5706 E-mail: tajima@mlab.t.u-tokyo.ac.jp

Abstract: This paper presents the system, protocol and performance of RMAC (Reliable Multicast using Adaptive Caching). RMAC provides router-assisted reliable multicast transport. Our approach is motivated by studies that computations within the network are required to make reliable multicast effective. RMAC improves retransmission efficiency by caching and retransmission at routers within the network, and reduces router cache size by caching data packets distributedly over RMAC routers and adaptively to conditions of network. Simulation results show RMAC decreases recovery latency, and improves cache utilization as well as scalability compared with the conventional schemes for reliable multicast.

Key Words: Reliable Multicast, Active Network, Adaptive Caching, Network Service, Internet

1 まえがき

インターネットの発展にともない、多数の受信者に対して同一情報を送信することのできるマルチキャスト通信への期待が高まっている。文書やプログラムの配信といったパケット損失が許されないアプリケーションに対しては、エンド・エンド間でパケット損失を回復して信頼性を保証する SRM [1] 等のリライアブルマルチキャストが研究されている。

効果的にリライアブルマルチキャストを実現するためには、一般的に以下の課題がある。

- 複数の受信ホストまでのネットワーク状態を送信ホストが把握することが難しい
- 最適な再送元、再送範囲を決定・制御することが難しく、再送時の負荷の分散、帯域消費の低減が難しい
- 受信ホストが多数存在することから、受信ホストから送信されるパケットがネットワーク内のノードに集中する
- ネットワークの一部で発生した輻輳がリライアブルマルチキャスト全体のスループットの低下を招く

これらの問題点を解決するために、ネットワークを構成するルータがリライアブルマルチキャストを支援する手法が研究されている。PGM[2]ではルータで重複するパケット、転送不要なパケットを削減するが、データパケットをキャッシングする処理はなされていないため、再送元を最適にすることができない。ARM[3]ではPGMの機能に加えてルータでデータパケットのキャッシングを行うことで再送を効率化しているが、キャッシングを行う時点・位置についての制御がなされておらず、また、各ルータごとに独立にキャッシュを行っており、キャッシングの効率化が図られていない。

そこで本稿では、ルータにおいてデータパケットのキャッシング・再送を行うとともに、キャッシングをネットワークの状態に応じて動的・分散的に行う、適応型キャッシュを用いたリライアブルマルチキャスト RMAC (Reliable Multicast using Adaptive Caching) のシステムならびにプロトコルを示す。

本稿では、まず、リライアブルマルチキャストのネットワークでのサポートという観点からルータで支援すべき処理を明らかにする。次に、RMAC のシステムならびにプロトコルの詳細を記述する。さらに、シミュレーションにより SRM, ARM [3] との性能比較を行う。

2 リライアブルマルチキャストのためのネットワーク支援

マルチキャストにおいて損失パケットを回復するためには、次の二つの処理が必要になる。(1) パケットの損失地点より下流の受信ホストが損失地点より上流に再送

要求パケットを送信する。(2) 要求を受け取ったノードがパケットの損失地点より下流の受信ホストに再送をマルチキャストする。

この時、次の条件が満たされた場合、最適な再送であると考えられる。(a) 送信される再送要求/再送パケットが共に一つずつである。(b) 再送パケットはパケットの損失地点より下流だけに転送される。(c) 再送/再送要求パケットが転送される距離は可能な限り短い。これらの条件を満たすためには損失地点とトポロジーの情報をもとに再送を行う必要がある。

そこで、本稿ではルータにキャッシュと再送の機能を組み込み、可能な限り損失地点に近いルータでデータパケットのキャッシングを行い、再送を最適化することを目的とする。

本稿ではネットワーク内のルータに機能的な処理を行う機構を組み込み、リライアブルマルチキャストを支援するために利用する。そのためには、以下に挙げる要求を満たすことが望ましい。

- ルータ内の共有資源の利用は必要な機能を実現する最低限のものとして、共有資源の効率的利用を図ること。
- リライアブルマルチキャストを支援する機能を具備しないルータとの相互運用性を確保し、ネットワーク上の一部のルータから導入を進めることが可能であること。

ネットワーク内においてリライアブルマルチキャストを支援する機能を利用し、再送を効率化するプロトコルとしてこれまでに PGM [2], ARM [3] が提案されている。PGM はルータで重複するパケット、転送不要なパケットを削減する方式である。しかし、ネットワーク内でデータパケットのキャッシングを行わないことから、再送元が最適化されない。また、ARM は PGM の機能に加え、ルータでパケットのキャッシング・再送を行う方式であるが、キャッシングを行う時点・位置についての制御がなされておらず、最適な位置でのキャッシングを行うことができない。また、各ルータが独立にキャッシングを行うことからキャッシュ利用の効率化が図られず、先に述べたルータに機能を組み込む際の共有資源に対する要求を十分に満たしているとは言えない。

そこで、本研究では新たに以下の機能をルータに導入する。

- 出力キュー長を監視し、輻輳を検知する
- ルータ間で協調し、パケットを分散的にキャッシングする

輻輳を検知することでキャッシング・再送を行う最適な位置を決定する。また、分散キャッシングにより共有資源であるキャッシュの効率的利用を図る。これらの機

能を用いて、輻輳によりパケット損失が発生した地点より上流で、再送が必要なデータパケットをキャッシング・再送する。これにより、再送を最適化し、効率的なキャッシングを行うことが可能である。

また、機能的ルータの仮想的ツリー構造を形成することで、リライアブルマルチキャストのネットワーク支援をネットワーク内の一部から導入することができ、システムを迅速に展開することが可能である。

3 適応型キャッシュを用いたリライアブルマルチキャスト

3.1 システムの概要

本方式では全てのルータがIPマルチキャスト転送を行うことを前提とする。また、本方式に対応したルータをネットワーク内に導入する。本方式に対応したルータを以下ではRMACルータと呼ぶ。ネットワーク内に散在するRMACルータがマルチキャストツリー上で互いに通信することで、RMACルータの仮想的ツリーを形成する。RMACルータでは、再送要求/再送パケットの通過を記録するためのテーブルを保持し、重複パケット、不要なパケットを削減する。また、輻輳を検出してパケットを適応的にキャッシングし、廃棄されたデータパケットの回復遅延を低減する。隣接するRMACルータ間でキャッシング要求を交換することで再送が必要なパケットを分散的にキャッシングする。

RMACルータでパケットの廃棄やキャッシング等の付加的な処理を行うことは、IP Router Alert Option[4, 5]を利用することで実現可能である。ルーティング処理とは別のプロセッサを設けて付加的な処理を行うことで、他のプロトコルへの影響を小さくすることができる。

RMACプロトコルでは、パルクデータの転送を主な対象とし、全パケットに対して完全な信頼性を提供する。パケットの到着順は保証しない。図1にRMACプロトコルによる典型的な転送の例を示す。受信ホストがパケット損失を検出すると、再送要求パケット(NACK)が送信され、RMACルータの仮想ツリーに沿って転送される。RMACルータでは通過する再送要求を記録する(NACK Table)。再送要求が当該データパケットをキャッシュに保持するルータまで到達すると、再送パケットが下流に対してマルチキャストされる。RMACルータでは再送パケットに対応する再送要求の通過が記録されている場合だけ再送パケットを転送し、記録されていない場合は、不要な再送となるため再送パケットを廃棄する。

図2に適応的キャッシングの動作を示す。RMACルータは輻輳によりキャッシュを開始するが、キャッシュの残容量が少なくなった場合に上流のRMACルータに対し

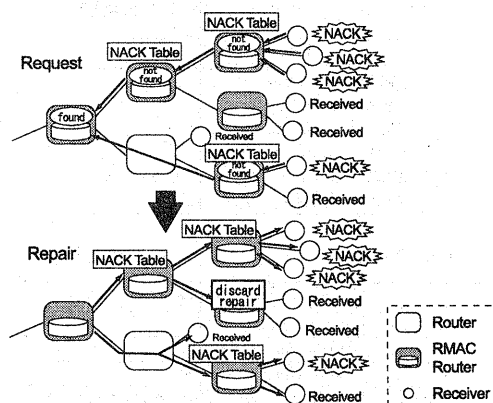


図 1: RMAC での転送

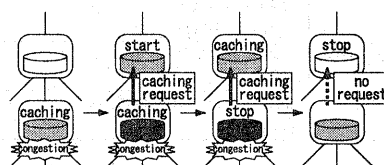


図 2: RMAC ルータでの適応的キャッシングの動作

てキャッシング要求を出す。キャッシング要求を受けたRMACルータではキャッシングを開始する。輻輳が解消し、キャッシュの残容量が確保されるとキャッシング要求を止める。キャッシングを行っている上流のRMACルータでは、キャッシング要求がなくなった場合キャッシングを停止する。

キャッシングはルータ上の一時的な記憶領域で行われ、その記憶領域の容量は有限である。これはネットワーク上の資源という観点から非常に重要な制限要素である。また、キャッシュは複数のマルチキャストセッションによって共用される。従って一つのマルチキャストセッションで利用できるキャッシュ容量はコスト、マルチキャストセッション数によって制限される。本方式ではルータ間で協調して分散的にキャッシュを行うことでこの問題に対処している。

3.2 RMACプロトコル

本プロトコルは送信ホスト、受信ホスト、及びRMACルータで処理される。本方式で用いるパケットの種類を表1に示す。また、各パケットのヘッダフィールドを表2に示す。PktType, MGroup, Source, PrevIDは各パケットに共通のフィールドである。共通のフィールドの他に、DATAパケットはSeqNo, NACKパケットとREPRパケットはSeqNoとNC, CCTLパケットはSeqNoとCacheReq, SESSパケットはPktNoのフィールドを持つ。全てのパケットでIP Router Alert Optionの

表 1: 本方式で用いるパケットの種類

Packet Type	
DATA	data packet
NACK	NACK packet
CCTL	cache control packet
REPR	repair packet
SESS	session information packet
SACK	ACK to SESS packet

表 2: RMAC パケットヘッダに記述する情報

Fields of Packet Header	
<i>PktType</i>	packet type
<i>MGroup</i>	multicast address
<i>Source</i>	data source address
<i>PrevID</i>	address of previous adaptive router
<i>SeqNo</i>	sequence number of DATA (DATA, NACK, CCTL, REPR)
<i>NC</i>	NACK count (NACK, REPR)
<i>CacheReq</i>	cache request (CCTL)
<i>PktNo</i>	number of data packets (SESS)

ビットがセットされる。

RMAC ルータで保持する情報を表 3 に示す。RMAC ルータでは重複する NACK、不要な REPR を廃棄するために、マルチキャストセッションごとに NACK、REPR の通過情報を保持する。また、キャッシュ制御のための CCTL の受信情報を保持する。

セッションの確立

送信ホストはデータの送信に先立って SESS をマルチキャストし、セッション情報を通知すると共に、RMAC の仮想ツリーを形成する。仮想ツリーの形成は次の手順で行われる。SESS がマルチキャストされ RMAC ルータにより転送される時、PrevID に当該 RMAC ルータのアドレスが書き込まれる。直下の RMAC ルータでは受信した SESS の PrevID を直上の RMAC ルータアドレスとして保持する。SESS が受信ホストに到達すると、SACK を直上の RMAC ルータ宛に返送する。このとき PrevID に自アドレスが書き込まれる。通過する RMAC ルータでは PrevID の内容を直下の RMAC ルータのアドレスとして保持する。これにより、RMAC ルータは上流、下流の RMAC ルータのアドレスを取得し、RMAC

表 3: RMAC ルータで保持する情報

NACK / Repair Table Entry	
<i>SeqNo</i>	sequence number of requested DATA
<i>NC</i>	NACK count
CCTL Table Entry	
<i>SeqNo</i>	sequence number of DATA
<i>DescIDs</i>	descendants' CCTLs receipt

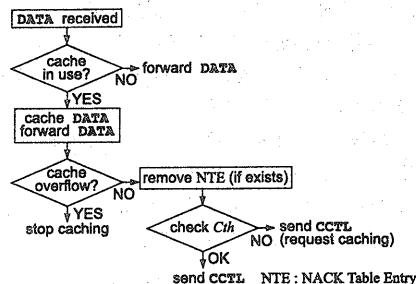


図 3: RMAC ルータでの DATA 処理

ルータの仮想ツリーが形成される。

本方式では NACK、CCTL は受信ホストからユニキャストで送信される。通常、受信ホストから送信ホストまでのユニキャストルーティングのパスとマルチキャストツリーのパスが一致しているとは限らない。しかし、仮想ツリーの形成によって、マルチキャストツリーに沿って上流方向への RMAC ルータ間での通信が可能となる。

データ転送

送信ホストは送信するデータを一定の大きさに分割し、DATA としてマルチキャストアドレス宛に一定レートで転送する。

図 3 に RMAC ルータでの DATA 処理とそれに伴うキャッシュ制御のための CCTL 送信処理を示す。RMAC ルータはキャッシュを行う状態と行わない状態との二つの状態を持ち、これらの状態は後述するキャッシュ制御によって決定される。キャッシュを行わない状態で DATA を受信した場合、パケットの転送のみを行う。キャッシュを行う状態では、パケットをキャッシュにコピーしてパケットの転送を行い、キャッシュ制御のための CCTL 送信処理を行う。この CCTL 送信処理については後述する。

最後の DATA が損失した場合に、受信ホストがパケット損失を早期検出できるように、送信ホストは最後の DATA を一定レートで繰り返し送信する。

再送制御

受信ホストは DATA の SeqNo によってパケット損失を検知し、NACK を直上のルータ宛に送信する。また、タイマーをセットし、タイムアウトまでに要求した REPR を受信しなければ、タイムアウトすることにより NC を 1 ずつ増やして NACK を再送する。

図 4 に RMAC ルータ及び送信ホストで NACK を受信した場合の処理を示す。ルータは NACK を受信すると同一 SeqNo の DATA がキャッシュ中に存在するか調べる。DATA がキャッシュに存在する場合、Repair Table に当該 DATA のエントリがなければ、これを作成し、REPR を送信する。エントリがある場合、エントリ中の NC が NACK の NC より小さければエントリ中の NC を NACK の NC にセットして REPR を送信し、エントリ中の NC

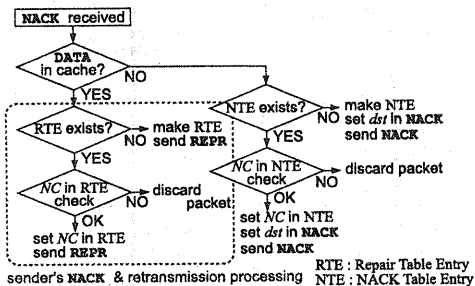


図 4: RMAC ルータと送信ホストでの NACK 処理と再送

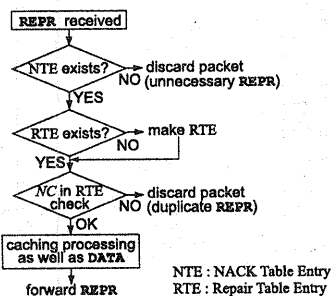


図 5: RMAC ルータでの REPR 処理と再送

が NACK の NC 以上の値であれば、以前の再送と重複するので REPR の送信は行わない。

DATA がキャッシュに存在しない場合、NACK Table に当該 DATA のエントリがなければ、これを作成し、NACK の宛先アドレスを上流の RMAC ルータのアドレスにセットし、転送する。エントリがある場合、エントリ中の NC が NACK の NC より小さければエントリ中の NC を NACK の NC にセットして NACK を転送し、エントリ中の NC が NACK の NC 以上の値であれば、以前の NACK の転送と重複するので NACK の転送を行わない。

送信ホストにおいては要求された DATA は必ず存在するため、図 4 中の破線で囲んだ部分の処理だけを行う。

また、RMAC ルータにおいて REPR が転送される場合の処理を図 5 に示す。RMAC ルータは REPR を受信すると、NACK Table に同一の SeqNo のエントリを参照する。エントリが存在しない場合、下流から再送要求が通過していないことを意味し、転送不要な再送パケットであるため REPR を廃棄する。

NACK Table にエントリが存在する場合、Repair Table の同一 SeqNo のエントリを、なければ作成して参照し、エントリ中の NC が REPR の NC 以上の場合、以前の再送と重複するので REPR を廃棄する。エントリ中の NC が REPR の NC より小さい場合、DATA を受信する場合と同様のキャッシング処理を行い、REPR を転送する。

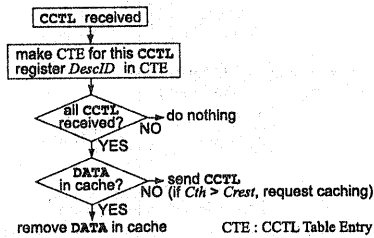


図 6: RMAC ルータでの CCTL 処理

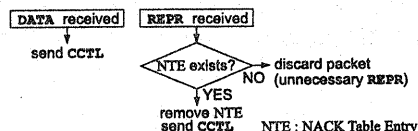


図 7: 受信ホストの処理

キャッシュ制御

図 6 に RMAC ルータで CCTL を受信した場合の処理を示す。RMAC ルータが CCTL を受信した場合、同一 SeqNo の CCTL Table のエントリを参照し、CCTL を送信した直下の RMAC ルータを DescIDs に記録する。直下の全ての RMAC ルータから CCTL を受信した際には、直下の全ての RMAC ルータより下流への DATA の転送が保証されたことになるので、キャッシュに同一 SeqNo の DATA が存在する場合これを削除し、存在しない場合 CCTL を転送する。

CCTL ではデータ受信の報告に加え、キャッシング要求も行う。図 3 にキャッシング要求の送信処理を示す。RMAC ルータは DATA または転送の必要な REPR を受信し、パケットをキャッシュにコピーする時に利用可能なキャッシュの残容量 Crest を確認する。残容量がない場合はキャッシュを停止する。コピーできる場合、以後当該 DATA に対する NACK を上流に転送しないので、NACK Table のエントリがあればこれを削除する。Crest がキャッシング要求閾値 Cth 以下の場合、CCTL による上流の RMAC ルータへのキャッシング要求が行われる。Crest がキャッシング要求閾値 Cth より大きい場合は CCTL でのキャッシング要求は行われない。また、図 7 に示すように、受信ホストが RMAC ルータに CCTL を送信する場合も、キャッシング要求は行わない。

RMAC ルータの状態遷移

RMAC ルータはキャッシュ制御に関して次の状態を持つ。

- キャッシングを行う／行わない
- 上流の RMAC ルータにキャッシング要求を行う／行わない

図 8 に RMAC ルータでのキャッシングの状態遷移を示す。RMAC ルータは、出力バッファのキュー長、CCTL

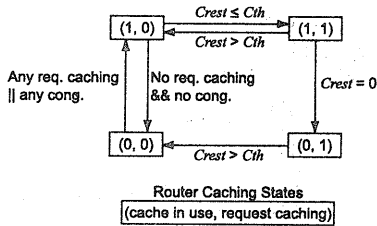


図 8: RMAC ルータでのキャッシングの状態遷移

によるキャッシュ要求, キャッシュ残容量 $Crest$, キャッシュ要求閾値 Cth の値によってキャッシングの状態を変える。

4 シミュレーション

RMAC の性能評価を行うために, シミュレーションによって SRM[1], ARM[3] との比較を行った。シミュレーションには文献 [1, 3] で用いられているネットワークシミュレータ *ns*[6] を用いた。

ネットワークモデルは 1 送信ホストを根とする 256 端末のツリートポロジである。受信ホストの接続するルータでランダムにパケットを発生させて輻輳状態を作り, ランダムなパケット損失を与えた。1KB のパケットを 1000 パケット転送し, 全てのパケット損失に対し, 正規化再送遅延時間を測定した。ここで, 正規化再送遅延時間とは受信ホストがパケット損失を検出してから, 最初の再送パケットを受信するまでの時間を, 送信ホストとの間の RTT で割った値である。

4.1 スケーラビリティ

10 ~ 100 の端末をランダムに受信ホストとして選んだ場合の正規化再送遅延時間を図 9 に示す。ARM 及び RMAC では, 各ルータ上のキャッシュ容量は 200 パケット分としている。

送信ホスト, 受信ホストからの再送によりパケット損失を回復する SRM では, 定常的な輻輳が発生した場合, 受信ホスト数が増えると正規化再送時間が増大する。RMAC では受信ホスト数によらず, 正規化再送遅延時間をほぼ一定に保つことができ, ARM より正規化再送遅延時間を短縮できることがわかった。

これはキャッシュ制御により, 受信ホストの近くで損失パケットが分散的にキャッシングされ, 効果的に再送されていることを示している。マルチキャスト通信において受信ホストの近くから再送を行うことは, 不要な再送パケットを削減するという観点からも非常に重要である。

4.2 キャッシュの有効利用

図 10 はルータ上のキャッシュを利用する ARM と RMAC について, 受信ホスト数を 100 に固定し, ルータ上で利用可能なキャッシュ容量を 100 ~ 1000 パケット分に変化させた場合の正規化再送遅延時間である。

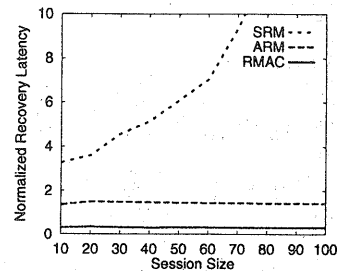


図 9: 受信ホスト数と正規化再送遅延時間

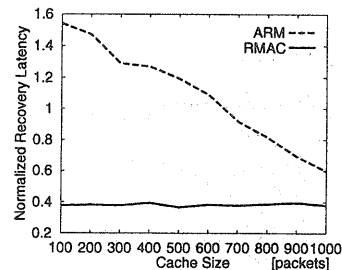


図 10: キャッシュ容量と正規化再送遅延時間

RMAC では, 輻輳に応じて, 再送に必要なパケットを, 効果的に分散してキャッシュすることができ, 再送遅延, 消費帯域の削減を図ることができる。

5 むすび

本稿では, リラiableマルチキャストをネットワークで支援するアーキテクチャに関して検討し, 適応型キャッシュを用いたリラiableマルチキャスト (RMAC) の詳細を示した。RMAC の性能評価をシミュレーションによって行い, ネットワーク内で分散的にキャッシュして再送することの有効性を示した。現在, 更なるシミュレーションによって RMAC の詳細な性能評価を進めており, ネットワークに展開する指針についても検討を進めている。

参考文献

- [1] Sally Floyd, et al. A Reliable Multicast Framework for Light-weight Session and Application Level Framing. In *Proceedings of ACM SIGCOMM '95*, pp. 342-356, October 1995.
- [2] Tony Speakman. PGM Reliable Transport Protocol Specification. *draft-speakman-pgm-spec-02.txt*, August 1998.
- [3] Li-wei H. Lehman, et al. Active Reliable Multicast. In *Proceedings of IEEE Infocom98*, April 1998.
- [4] D. Katz. IP Router Alert Option. *RFC2113 IETF*, February 1997.
- [5] D. Katz. IPv6 Router Alert Option. *draft-ietf-ipngwg-ipv6router-alert-04.txt*, February 1998.
- [6] UCB/LBNL/VINT project. *Network Simulator - ns (version 2)*. <http://www.mash.cs.berkeley.edu/ns/>.