

マルチキャスト通信におけるアドミッションコントロールに関する一考察

4N-3

横田 英俊 片岸 一起 浅見 徹
 国際電信電話株式会社 研究所

1. はじめに

マルチメディア・マルチキャストセッションでは、音声・画像に代表される種々のリアルタイムストリームが複数のユーザに提供される。複数のユーザが同時に参加する場合、利用可能な帯域幅や処理速度がユーザによって異なるため、受信できるストリームの数も異なる。またある時点において興味のあるストリームがユーザによって異なるため、単一もしくは決められた数のストリームを全ての受信者に配送することは困難な場合が多い。もし、配送されるストリームが受信者の処理能力や興味の高さによって個々に選択できれば、柔軟性が多いが高まる。

アドミッションコントロール (Call Admission Control: CAC) はこのようなマルチキャスト・マルチメディアセッションにおいて、予約要求における競合を解消し、公正で効率的なストリームを供給する役割を持つ。資源予約を目的としたプロトコルである RSVP^[1]では QoS 等の情報を交換する機能を備えているが、競合する予約要求を解消する機構は提供していない。これに対して Shacham 等は複数の予約要求を考慮した CAC に関する手法を提案しているが^[2]、最適な割当を目的としたアルゴリズムであるため膨大な計算量が要求される。本稿では、計算量を削減し単純化した4種類の CAC アルゴリズムを示し、その評価を行う。

2. ネットワークモデル

本稿ではストリームの送信者(ソース)を根とするマルチキャストツリーを前提とし、節に位置する中継ノード (MNODE)、葉に位置するユーザ (ENODE) で構成される。そして ENODE からソースに向かう方向を上流、その逆を下流と定義する。一つのストリームの帯域幅があるリンクのそれを越えない場合、そのストリームはそのリンクにおいて「割り当て可能」と呼ぶ。CAC はさらに以下の条件を常に満たすものとする。

帯域条件 許可されたストリームの帯域幅の合計はそれらが通るどのリンクの帯域幅も越えない。

効率的な分配 同じストリームは2度以上割り当てられない。また要求したユーザまで到達できないストリームは許可されない。

3. 提案する CAC アルゴリズムの概要

ソースはストリームを受信するのに必要な帯域のリストを広報し、各々のユーザは受信したいストリームの要求を bid と共に申告する。bid はユーザにとって要求したストリームの重要度を表す定量的な値であり、もし要求したストリームがそのユーザに配送された場合にソースはその bid と等量の gain を得る。CAC を行うノードは複数の受信者からの bid を評価し、どの要求が

最適かを決定する。そして要求が受け入れられた全ての受信者からの bid はソースが gain として受け取る。したがって、CAC の性能は最終的なソースの gain の合計 (total gain) により比較することができる。全ての状況で gain を最大化する最適な CAC は、マルチキャストツリー上の全てのリンクのリソースを考慮するため計算量が膨大になる。計算量を削減するためには考慮するリンクの情報を絞り、ヒューリスティックなアルゴリズムを用いることが必要となる。本稿では CAC を行うノードや利用可能な情報によって4つの異なる準最適な割当手法を提案する。ただし、どの手法も以下の2つのフェーズを含む。

予約要求フェーズ 各 ENODE は bid を上流、すなわちソース側に送る。そして各々の MNODE は単一割り当て可能な bid のみを上流に送り、それ以外の要求を削除する。

アドミッションフェーズ 決定を下すノードが bid をソートし、最も高い bid のストリームに対して帯域を割り当てる。この後、割り当て不可能な bid を消去し、全ての bid が割り当て済みか、割り当て不可能となるまでこの手続きを繰り返す。bid は評価関数 $Eval()$ でソートされる。本稿では簡単のため以下の関数を用いる。

$$Eval(b, w) = b/w$$

ここで、 b は bid の値、 w は帯域幅とする。

4. 4種類の CAC アルゴリズム

図1に提案するアルゴリズムの基本フローを示す。

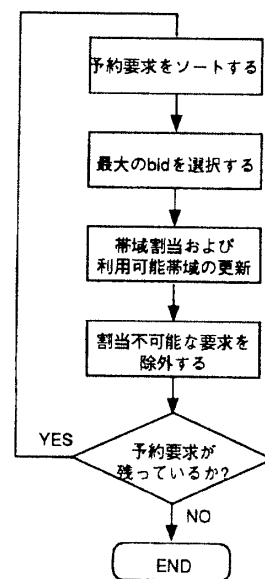


図1: CAC の基本フロー

CAC を行うノードは最初に予約要求をソートする。その後、greedy algorithm に基づき最も高い bid を選択

“Study on Admission Control in Multicast Communications”, Hidetoshi YOKOTA, Kazuki KATAGISHI and Tohru ASAMI, KDD R. & D Laboratories

表 1: 提案手法の比較

手法	決定するノード	割当のフェーズ	割り当てられるストリームの数	配送されるユーザの数
1	ソース	下流フェーズ	1ストリーム/サイクル	複数
2	ソース	下流フェーズ	1ストリーム/サイクル	1ユーザ/サイクル
3	ソース/MNODE	下流フェーズ	可能な限り割り当てる	複数
4	MNODE	上流フェーズ	可能な限り割り当てる	複数

し、それに対応するストリームに帯域を割り当てる。全てのリンクの帯域を更新した後、割り当て不可能な要求があればそれを取り除く。もしまだ要求が残っていたら、最初の段階に戻り、なければ手続きを終了する。

CACは集中的もしくは分散的な方法で行うことができる。集中的な手続きでは、ソースのみが決定を行い、MNODEはbidを集め、ストリームを分配するだけである。しかし、ソースによって選択されたストリームの全てが目的地までの全てのリンクの帯域条件を満たしているとは限らない。これを満たす方法の一つとして、一度に一つのストリームを送ることが考えられる。割り当て可能性は予約要求フェーズにおいて全てチェックされるため、アドミッションフェーズで考慮される個々のストリームは全てのリンクで帯域条件を満たしている。さらに一つのストリームを配送する方法として、選択されたストリームを、(1) 要求した全てのノードに配送する、(2) 最も高いbidを申告したノードに配送する、の2種類が考えられる。本稿では前者を「手法1」、後者を「手法2」と名付ける。

「手法3」では下流フェーズでソースとMNODEが割当を決定する。割当は各々の下流リンクの帯域と親ノードによって許可されたストリームをもとに決定される。この手法ではストリームを要求している全てのユーザに首尾よく配送されるとは限らない。なぜならあるリンクで帯域がなくなりそれ以上先に配送できないことがあるからである。このような途切れたストリームは削除され、開放された帯域は次のサイクルで使われる。

「手法4」では異なるアプローチが取られる。割当はMNODEによって予約要求が上流に上がって行くフェーズで決定される。各々のMNODEは子ノードによって報告されるbidと上流のリンクの容量に基づいて割当を決定する。下流フェーズでは、「手法3」と同様に途切れたストリームを開放し、空いた帯域を次のサイクルで使えるようにする。上記4手法を表1に示す。

5. 性能評価

5.1 各手法の平均的な性能の評価

計算機シミュレーションにより4つの手法の性能評価を行う。シミュレーションではランダムなマルチキャストネットワークを構成し、各手法を適用した時のtotal gainを計算する。ノードの数は10とし、単一のソースと、任意の数の受信者がランダムに選択され、任意の帯域幅(>0)を持ったリンクにより接続される。CACアルゴリズムの適用に先だて、ソースからENODEまでの最大の帯域幅を持つツリーが計算される。gainの平均値を計算するために1,000個のランダムに接続されたネットワークを構成した。評価結果を図2に示す。total gainは各々の場合の最適解で正規化されている。結果として、「手法3」が平均的に最も最適解に近い。「手法

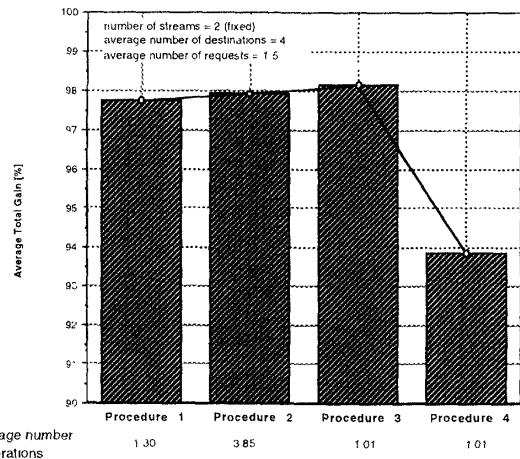


図 2: 各手法の性能の平均値

4」は最適解の90%以上は示しているものの、他の手法に比べ性能が低い。これは「手法4」が各MNODEにおいて局所解に陥っているためである。

5.2 繰り返しによる計算量の評価

4つの手法は全て、考慮する予約要求がなくなるまで図1に示すフローを繰り返す。「手法1」と「手法2」は一つのサイクルに一つのストリームしか割り当てないため、ストリームの数を S とすると「手法1」の繰り返しによる計算量は $O(S)$ となる。さらに「手法2」は一つのサイクルに一つのENODEにしか配送しないため、ENODEの数を N とすると $O(SN)$ となる。一方「手法3」および「手法4」は一度に複数ストリームを複数ユーザに割り当てることができるのでユーザ数およびストリーム数に影響されない。

6. 考察

最適解と比較した平均的な性能および計算量の面から見て、「手法3」が総合的に優れていると言える。また全ての手法において最適解の90%以上を示していることから、ユーザ数やストリーム数に影響されない「手法3」および「手法4」がスケラビリティを備えた現実的な手法と思われる。

7. おわりに

本稿ではマルチキャストネットワーク上でのアドミッションコントロールについて4つのヒューリスティックな手法を提案し、計算機シミュレーションにより最適解との比較評価を行った。最後に、日頃ご指導いただくKDD研究所村上所長に感謝いたします。

参考文献

- [1] R. Braden, et al. Resource ReSerVation Protocol (RSVP) - Version 1 Functional Specification. Internet Draft "draft-ietf-rsvp-spec-14", IETF, November 1996.
- [2] N. Shacham and J. Meditch. An algorithm for optimal multicast of multimedia streams. In *Proceedings of IEEE Infocom '94*, Toronto, Canada, 1994.