

リンクの負荷を考慮したアプリケーションレベル マルチキャストツリーの動的構築方法

村上慶司 佐藤文明†

† 静岡大学 情報学部

概要:近年,TV 会議システムなどの複数のメンバ間で情報のやり取りを行うアプリケーションが開発されておりそのための通信方式としてマルチキャストが利用されている. その中でも,IP マルチキャストのようにマルチキャストルータなどのインフラストラクチャを必要としない利便性から,アプリケーションレベルマルチキャストへの関心が高まっている. アプリケーションレベルマルチキャストでは,フラディングにより配送木の構築を行うことが多いが,IP マルチキャストのように配送経路が最適化されていないため1つのリンクに同一パケットが重複して送られることがある. 本論文では,同一パケットの重複率を削減する配送木の再構築方法を提案し,シミュレーションによって評価した.

A Reconfiguration Method of Application Level Multicast Trees

Keiji Murakami Fumiaki Sato †

† Graduate School of Information, Shizuoka University

Abstract: Many applications which exchange information among two or more members such as the TV conferencing system are developed and the multicast is used as a communication method for that in recent years. Especially, the application level multicast is paid attention because it does not need the special infrastructure such as the multicast router like the IP multicast. However, since the delivery tree is often constructed with Fradding in the application level multicast and the delivery route has not been optimized like the IP multicast, the same packet might be sent to one link more than once. In this paper, we propose the restructuring method of the delivery tree which reduces the duplication of the same packet, and it is evaluated by the simulation.

1 はじめに

近年の高速ネットワークの普及により,比較的小規模のグループによる,グループ通信が盛んに行われるようになってきた. 多人数による情報の共有や,複数の受信者に対する情報の配信など,手軽に行われる環境が整いつつある. 現在,これらの応用は発展途上にあるがすでに,ビデオチャットや,ビデオ会議システムなど映像と音声を同時に配信する技術も開発されてきている. これらの技術の開発について IP マルチキャストはとても有効な手段であり,主なプロトコルとして,DVMRP(Distance Vector Multicast Routing Protocol) や,PIM-SM などがある.(参考文

献 [2]) しかし,これらの IP マルチキャストでは,前提としてマルチキャスト対応ルータなどのインフラストラクチャの準備が必要となるなど,実現に向けて容易でない点があげられる.

このような問題を解決するための手段として,マルチキャストをアプリケーションレベルで実装するための方式がある. この方式では,IP マルチキャストとは異なり,エンド間ユニキャストを仮想リンクとみなしたオーバーレイネットワーク上でマルチキャストを実現するため,現在のインフラストラクチャでの実現が容易に行える. しかしながら,エンド間ユニキャストに基づく,アプリケーションレベルマルチキャストでは,IP マルチキャストのように配送経路が最適

化されていないため1つのリンクに同一パケットが重複して送られることが起こる。本論文では、同一パケットの重複率を削減する配送木の再構築方式を提案し、シミュレーションを評価する。

2 アプリケーションレベルマルチキャスト (ALM)

アプリケーションレベルマルチキャスト (ALM) についての構築方法は、先行研究として様々な方法が考案されている。本研究では、参考文献 [1] の構築方法を用いる。以下にその構築方法を図1と共に示す。

1. 新規参加者は、ALM にすでに参加しているメンバの IP アドレス情報をサーバから取得する。それと同時に新規参加者は、参加時にサーバに自分のアドレス情報を登録することになっている。
2. 新規参加者は、すべてのメンバに対して ping を送信し遅延時間の計測を行う。ノードへの遅延が短い2,3個のノードに対して、オーバーレイネットワーク (配送木の候補経路となる仮想ネットワーク) の接続要求のメッセージを出す。これで、オーバーレイネットワークを構築する。
3. マルチキャストデータの送信者は、オーバーレイネットワークの上に配送木構築のためのメッセージをフラッディングし、SPT の配送木を構築する。また途中参加のノードは、遅延時間に基づいて接木方式で構築していく。

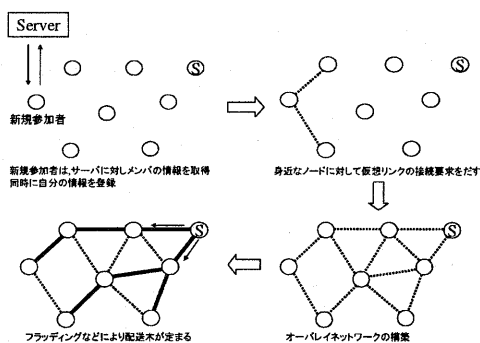


図1: 先行研究での配送木構築一例

以上のような構築手順を踏まえ作成された配送木では、下位層でどのような配送が行われているか考慮できない。つまり、複数の受信者に対して配送経路が重なり、それぞれのノード間のリンクにおいて同一

パケットが重複してしまうという問題が生じやすくなる。

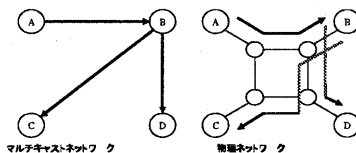
3 ALM 木の動的構築方式

3.1 概要

提案方式による配送木の構築は2段階で行う。初めにSPTを構築し、その配送木において同一パケットが1つのリンクに重複する配送木であると確認できると、このリンク部分の負荷を分散させるような新たな木へと再構築を行う。一つのリンク部分のパケットの重複を調べるために、まず複数の受信者に対して、共通に使われている経路の検出を行わなければならない。その理由として、多くの受信ノードに対してこの共通に使われている経路のリンク部分が長ければ長いほどパケットが重複しているリンクが多い、効率の悪い配送木であるといえる。よって、この共通経路を減らし、重複パケットを削減させるような経路構築方法が必要である。

しかしながら、リンクの利用率を高めるために、共通経路をすべてなくすことは遅延時間の増加につながるため現実的ではない。よって提案方式では、SPTから木を再構築する基準を定める。その基準が、一定の評価に達していない部分について、現在使用している配送木を再構築する。これにより再構築自体の計算量も削減できる。図2では、物理ネットワーク上でのパケット配送の違いを示している。先行研究では、ノードBからノードC,Dに配送されるときパケットが重複しているのがわかる。一方、提案方式では、パケットの重複が少ないことがわかるが、その分送信にかかる遅延時間も増加する。

先行研究



提案方式

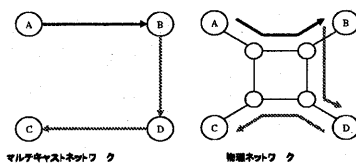


図2: 経路変更により変わる配送木の状態

3.2 経路変更の条件

上記でも述べた共通経路について、この経路の長さ(遅延時間)が、このアルゴリズムでは鍵になる。なぜならば、この長さが短ければ、同一の packets がいくつ流れたとしても、それほど負荷のかかったリンクの数の上昇につながっていかないからである。逆に長ければ長いほどその負荷のかかったリンクの数は増大していくと考えられる。ここでは、その共通経路の長さを使い、経路変更するための条件式、または現在の配木に対する評価基準を、以下に示す。これを満たさないときは、経路の変更を行わないこととする。

$$\alpha * \text{Max} \{ T_1, T_2, T_3 \dots T_n \} \leq \Delta t$$

$$(0 < \alpha < 1)$$

$T_n \rightarrow$ ある中間ノードから、仮想リンクで直接接続されている下位ノード n までの遅延時間

$\Delta t \rightarrow$ ある中間ノードから仮想リンクで直接接続されている n 個の下位ノードにおける共通経路の遅延時間

$\alpha \rightarrow$ ある中間ノードから複数の下位ノードに対しての共通経路の割合

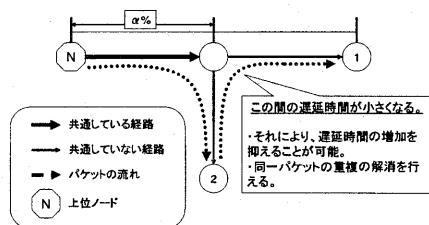


図3: 経路変更条件

上の条件式は、木のある中間ノードから複数の下位ノードのうち最も遅延時間のあるノードまでの時間の $\alpha\%$ 以上を共通に使われている経路として占められている場合に、経路変更をおこなう、という条件式である。 α の値はその共通経路の割合を表し、0%から100%までを考える。

経路を変更する場合、中間ノードから仮想リンクで直接接続しているノードのうち最短の遅延時間をもつノードまで配送し、そのノードから残りのノードに配送する経路に変更する。図3の場合では遅延時間の情報から点線の矢印のようにパケットが配送されることになる。そこからのパケットの配送についても、同様の経路変更手順が適用される。図4では、経路を変更する前と変更した後の遅延時間の増加の様子を点線の矢印で示している。

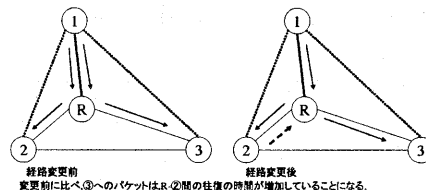


図4: 経路変更に伴う遅延時間の増加

提案方式では、木を再構築していくため、再構築にかかるオーバーヘッドも考慮しなければならないが、現時点において、提案方式では、考慮されていない。木の再構築にかかるオーバーヘッドは今後詳細に検討していく。

3.3 経路変更

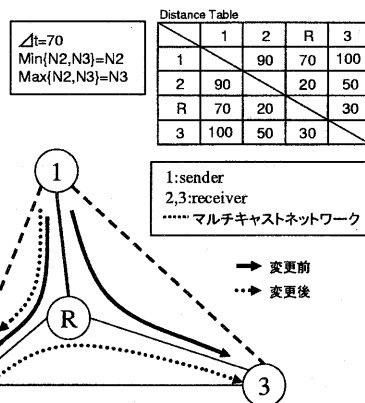


図5: 経路変更を行う木の状態

図5は、経路変更が考えられる最も小さなトポロジの例である。本来であれば、ノード1が送信を行いそれをノード2, 3が受け取る場合、中継ノードRを経て受信ノード2, 3にパケット送信を行うことが一番効率よく見える。しかし、その経路方法では、送信ノード1と中継ノードRとの間の経路において、同一パケットの重複が見られる。そのため、そのリンク部分では負荷が大きくなることがわかる。そこで、そのようなリンクの負荷を解消するため、リンクの負荷が解消されるように矢印のように経路を変更する。これにより、点線の矢印の通りの経路の場合だと、同一パケットの重複が解消され、そのリンクの部分の負荷が軽減されたことがわかる。このような経路の変更をそれぞれのノードの場合において行っていく、リンクの負荷を削減していくというものである。

図5を見たときに、先行研究どおりであれば、送信者1から送られたJoin Queryに対して、受信者2,3はJoin Replyを返さなければならない。そこで、受信者2,3は、サーバに登録されたDistance Tableと経路情報をもとに、Join Replyを返し、配送木を定めることになる。図5では、SPTの場合、Distance Tableをもとに、受信者2と3は、そのままJoin Replyを送信者である1に対して返すことになるが、提案方式のほうでは、Distance Tableと経路情報の二つの情報から共通経路の遅延時間を算出し、式(1)により同一パケットの重複が大きいものと判断したため、受信者3は、ノード2経由で到達したJoin Queryに対して、Join Replyを返すようになる。よって提案方式を考慮して作ると図5のトポロジは矢印のようなパケット配送を行うことになる。

4 シミュレーションと結果

4.1 評価値算出方法

以上の提案を元にそれぞれの木について評価を行うにあたって、リンクの資源の消費量を意味する値を用いることにした。評価値は以下の式で算出し、その値により木の評価を行うことにする。

$$\text{評価値} = \sum \text{Packet}(i) * \text{Link} \quad (1)$$

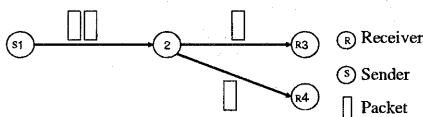


図6: リンクの上を通過するパケット

式(2)で算出される値は、送信者から各受信者に到達するまでの全リンクに送信されているパケット数の総和を求めている。図6では、送信者1が受信者3,4に対してパケットを送出するとき、経路1-2間で同一のパケットが2個、経路2-3,2-4間ではそれぞれ1個のパケットが流れていることがわかる。よって送信者から受信者までの評価の値は式(2)に当てはめると、 $2+1+1=4$ という値が得られる。

シミュレーションプログラムでは、 $100\text{ms} \times 100\text{ms}$ マスの座標の中に、ノードをランダムに配置した。ノードの数は20から100までノード数10毎をとることにする。これは、アプリケーションレベルマルチキャストにおいて想定されるノード数である。それ

ぞれのノード数で、木を1000通り作成しその平均値をデータとして取得していった。受信者数は、3から6まで、ランダムに変化する。

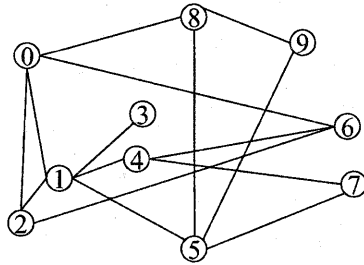


図7: 作成されたトポロジの一例(ノード数10)

シミュレーションプログラムにより評価する対象として、リンクの負荷を見るための式(2)により算出される評価値、それとSPTから変更するために増加してしまう、総遅延時間を経路変更条件10%,30%,50%,70%について取得していく。この二つの値は比較対象の先行研究としてSPTと提案方式とで値を取得し、その値の比較で木の評価を行っている。図7は、シミュレーションプログラムにより計算させたトポロジの一例である。一つのノードの平均リンク数は3に設定しており、遅延時間には、ベクトル距離を用いて計算を行った。

4.2 シミュレーション結果

ここでは、経路変更条件に合致せず、経路変更を行わない場合は先に構築されているSPTによる評価と遅延時間をそのまま使い、経路変更を行える場合は再構築後の木について評価と遅延時間を計算し、平均値を算出した。

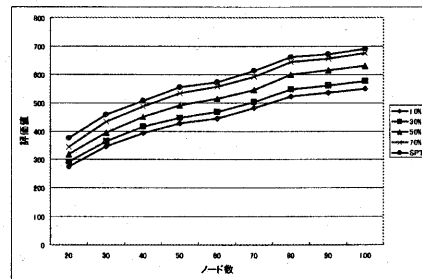


図8: SPTと提案方式による木の評価値の比較

経路変更を行っていた方が、よりよい評価が得られた。経路変更条件が低くなるにつれて、評価値は下

がっているため、よりよい木が作られているといえる。これは、後ほど述べるが、経路変更条件が低いために経路変更発生率が高くなることから、より多くの場合で経路変更がとられていると考えることができる。逆に、経路変更条件は高く設定すればより多くの同一パケットの重複を防ぐことが可能となるが、その条件が高いために、経路変更を行う機会が非常に少ないことがこの評価値の上昇に繋がっているといえる。

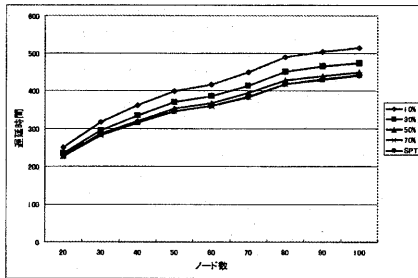


図 9: SPT と提案方式による木の遅延時間の比較

遅延時間は、評価値とは異なり、条件を低く設定すればするほど、大きくなっている。これは、条件設定が低いため、多くの場合で、経路の変更を実行してしまうため、遅延時間が大きくなっていることから容易に考えられる。一方 70% のように経路変更条件を高くしておくと、遅延時間のロスが最小限で抑えられていることがわかる。

4.3 考察

4.3.1 経路変更の発生率

経路変更を行うといっても、変更条件にあう状況が実際どれほど発生するかについても考慮しなければならない。経路変更条件を高く設定すると経路変更を行う状況は減少し、低く設定すると状況は増加する。そのため、無駄に経路変更条件を高く設定するとその経路変更状況が発生しないことも十分に考えられる。そこで、1000 通りの木を作り、経路変更条件に合致する場合がどれほどあるのか、条件毎に計測してみた。その結果は下のグラフである。

この図 10 から分かったとおり、共通経路 70% のかなり高い条件設定にすると経路変更を行う発生状況が格段に落ちることが確認できる。提案方式において、経路変更を行わない場合は、先ほど構築している木 (SPT) を用いるので、共通経路 70% の変更条件では、94% が配送木の再構築をしないままになるため、再構築をするかどうかの検出に意味があるかなどの

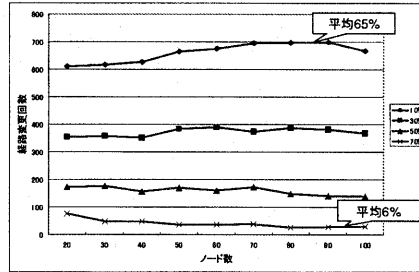


図 10: 経路変更条件に合致する割合

検討も加えていかなければならないと考えられる。

4.3.2 経路変更による評価の考察

経路変更における純粋な影響を考えていくため、経路変更が行える状況のトポロジのみを作成し、経路変更を行った後の配送木について評価値と遅延時間を計測した。

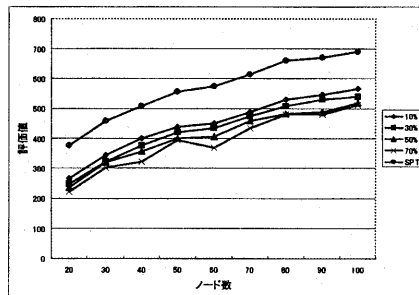


図 11: 経路変更を行えた場合の評価値

図 11 のグラフでは、経路変更を行わない場合に比べ、どのノード数においても、よい評価が得られたといえる。経路変更条件を高くすればするほど、得られる評価値はよくなっている。共通経路が 70% 以上の場合、SPT に比べ、どのノード数においても約 30% 評価値が上昇している。

経路変更条件が高い状態であれば、図 12 を見て分かる通り、遅延時間の上昇も抑えることが可能である。70% の条件では、SPT に近い数値が得られており、経路変更の影響が少ないことが分かる。また、10% の条件では、図 9 で取得した遅延時間と同じように上昇している。これは経路変更の発生率が 10% では高すぎるため、SPT の利点が生かされていない状況が作られている。

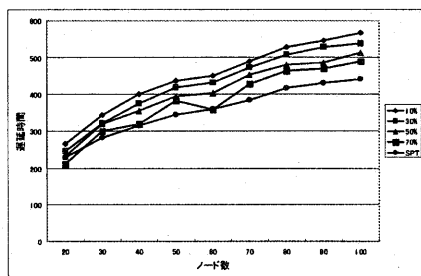


図 12: 経路変更を行えた場合の遅延時間

4.3.3 発生率の操作

以上のことから、経路変更の発生率が高ければ高いほど、経路変更が有効になることがわかった。経路変更条件についてもなるべく高い条件設定をしておけば、遅延時間の増加も最小限に抑えることが可能であった。しかし、実際のネットワーク状況においては、常に経路変更の発生率が高い状況ばかりではない。そこで、この経路変更条件を動的に動かし、経路変更の発生率をあげる方法をとることができる。その動的に動かす基準としては、経路変更発生率が 50% を維持できるように条件を動かすことや、上位ノードでは影響が大きくなることも考え上位ノードでの経路変更条件の高設定、一方下位ノードでは影響が少ないので低設定にすることなどがあげられる。このような方法をとることで発生率を操作する形で評価値、遅延時間ともに良い値が期待できることが考えられる。

5 まとめと今後の課題

本研究では、アプリケーションレベルマルチキャストにおける SPT を利用しての木の動的構築方法について研究してきた。提案方式により、SPT に比べ、同一パケットの重複が解消されていることが分かった。遅延時間は想定通りではあるが、SPT に比べ増加することとなった。また、経路変更条件を変化させることによりまた新たな視点での木の構築が見えてきた。この条件を様々な状況において動的に変化させるなどの工夫を加えればより柔軟な木が構築できると考えられる。今後の課題としては、より詳細な状態でのシミュレーションを行っていくこと (具体的には再構築にかかるオーバーヘッドなどの考慮) や、その他にも、遅延時間と同一パケットの重複の削減との間でのトレードオフを考慮していくことが考えられる。

参考文献

- [1] 中村嘉隆, 山口広純, 廣森聡仁, 安本慶一, 東野輝夫, 谷口健一: ユーザプリファレンスに基づく転送制御を行うアプリケーションレベルマルチキャストの方式, マルチメディア通信と分散処理 (2002,3,28)
- [2] Dave Kosiur: マスタリング TCP/IP -IP マルチキャスト編-, オーム社開発局 (1999,11,30)
- [3] Jurgen Vogel, Jorg Widmer, Dirk Farin, Martin Mauve, et al.: *Application-Aware Distribution Trees for Application-Level Multicast*, University of Mannheim, Germany
- [4] Marvin Theimer, Michael B. Jones: *Scalable Name Service on an Overlay Network*, Microsoft Research, Microsoft Corporation
- [5] Suman Banerjee, Christopher Kommareddy, Koushik Kar, Bobby Bhattacharjee, Samir Khuller: *Construction of an Efficient Overlay Multicast Infrastructure for Real-time Applications*, University of Maryland, College Park