

利己的なエンドノード間でマルチキャストを実現するための インセンティブ配分法

清水佳範* 中村嘉隆* 山口弘純* 東野輝夫*

エンドノードがマルチキャストルータの役割を果たすことで実現されるオーバーレイマルチキャストでは、データパケットの複製・転送がそのままノード自身の負担となるため、インターネットユーザなど、利己的に振る舞うエンドノード間でマルチキャストを構成する場合には、各エンドノードに対し、コスト負担に見合う対価を支払い、全体のシステム維持に協力させることが望ましい。本稿では、それぞれ利己的な振る舞いをするエンドノード群がマルチキャスト木を構築する際に、低遅延マルチキャスト木の実現のために果たした役割に応じたインセンティブを与えることで、高品質なオーバーレイマルチキャストを実現するインセンティブ配分法、及びその方法を用いたオーバーレイマルチキャスト構築プロトコルの設計について述べる。シミュレーションによる性能評価を行った結果、提案手法により構築されるマルチキャスト木の最大遅延は、最適値に近い最大遅延を実現できる集中型のプロトコルとほぼ同程度に抑えられていることが確認できた。

Design of Incentive Mechanism for Constructing Minimum Delay Multicast among Selfish End Users

Yoshinori Shimizu*, Yoshitaka Nakamura*, Hirozumi Yamaguchi* and Teruo Higashino*

In the overlay multicast where end nodes act as multicast routers, those nodes need to pay costs for replicating and forwarding data packets. Therefore, it is desirable to compensate for the work by each end node in order to make it cooperate in constructing a multicast tree among the other end nodes. In this paper, we propose an incentive mechanism to achieve a high-quality overlay multicast tree by giving each node an incentive corresponding to the role played in constructing the multicast tree. We also propose an overlay multicast tree construction protocol based on the proposed incentive mechanism. Simulation results have shown that our method could achieve a multicast tree which has similar latency the one constructed by a known centralized heuristic protocol.

1 はじめに

近年のネットワークの広帯域化、計算機端末の高性能化に伴い、マルチメディアデータの多人数への配信など、多くのネットワーク資源を消費するアプリケーションの需要が高まってくると考えられる。従来の限られた数のサーバによるユニキャスト通信では、このようなアプリケーションを利用する場合、データの送信元付近で大量のデータパケットが発生する。これに対する解決法としては効率的な伝送経路をルータレベルで提供する IP マルチキャストが代表的であるが、エンドシステムがそれぞれ異なる ISP に所属するインターネットワイドのアプリケーションでは、ISP ごとの可用性など運用面での課題も多い。

一方、オーバーレイマルチキャストはネットワーク上のユーザ同士が仮想的にネットワーク（オーバーレイネットワーク）を構築し、各ユーザがマルチキャストルータの働きをすることによって、マルチキャスト配信を実現する仕組みであり、多くの研究がなされている。オーバーレイマルチキャストではアプリケーション層での仮想リンクを利用してマルチキャスト配信を実現するため、特定のインフラストラクチャを必要としない、制御プロトコルを自由に設計できる、などの利点がある。

しかし、中継ノードがユーザ自身であるオーバーレイマルチキャストでは、転送負荷がそのままユーザにおけるコストとなる。また、特にモバイル環境での利用などを考えた場合には、消費電力等、負担できる転送負荷に大きな制約がある。インターネットにおける自律ユーザ（インターネットを介してアプリケーションを利用する一般的なユーザで、各自の都合、判断でアプリケーションを終了するような行動を取ることができるユーザ、以下単にノードと呼ぶ）は、他ノードのデータ転送に自身のリソースを無償で使われることを嫌う傾向にあるため、それらのノードによる資源提供を期待することは難しい。これに対し、データ転送に貢献したノードに対し報酬（インセンティブ）を与えること

で、全体として安定した通信を実現するための方法が主に P2P やアドホックネットワークの分野で提案されてきている [1, 2, 3, 4, 5, 6, 7]。

本稿では、送信ノードからの最大遅延がなるべく小さい配信木を、(利己的な) ノード間で構築させるためのインセンティブ配分法及び、それに基づくオーバーレイマルチキャスト構築プロトコルの設計について述べる。提案するインセンティブ配分モデルでは、できるだけ多くの出次数を提供し、かつ多くの下流ノードを持つノードに多くのインセンティブを配分する。これにより、各ノードに積極的な計算機資源の提供を促す。また、提供する出次数が大きいノードをより上流に配置することがマルチキャスト木の遅延削減に有効であると考えられる。この考えに基づき、ノードが提供する回数に対しそれを最も有効利用できると予想される木上の位置を提示し、その位置に近いノードにも多くのインセンティブを配分する。ノードにこのインセンティブモデルを開示し、適切な情報を与えた上でインセンティブ増大を目指して移動させることで、マルチキャスト木の最大遅延を徐々に削減する。

提案したインセンティブモデルに基づくシステムおよびプロトコルの設計を行い、簡単なシミュレーションによって性能を評価した。最適値に近い最大遅延のマルチキャスト木を実現できる集中制御型アルゴリズム CT[8] で得られる遅延と比較した結果、比較的少ない木構造の変更で、遜色ない最大遅延を持つマルチキャスト木に収束することが確認できた。

関連研究 既存の遅延最小木構築アルゴリズムとして、集中制御型のアルゴリズム CT[8] や分散制御型のアルゴリズム OMNI[9] などが提案されている。

一方、オーバーレイネットワーク上での通信においてインセンティブモデルを取り入れている研究がいくつか提案されてきている。文献 [1] では、P2P ストリーミングにおいて課税の仕組みを取り入れている。この方式では、例えば帯域を多く持つような潤沢なリソースを持つピアには多くのピアにストリームを供給してもらうという方針を採用している。また、各ピアがどれだけのピアに転送させるかはストリーム

*大阪大学 大学院情報科学研究科
Graduate School of Information Science and Technology, Osaka University

のソースが決定し、各ピアに対してそれぞれのストリームごとに能力に応じた固定の位置を与える。また、文献 [2] では、セッションへ貢献できる資源をスコア化し、それをランク付けすることで P2P ストリーミングでのピア選択を行っている。具体的には、各ピアが貢献した資源量に応じて向上した受信品質と資源提供の差を最大とするようピア選択を行う Rank-Based と呼ばれる方式を提案している。また、文献 [3] の Exchange-Based と呼ばれる方式は、ピア同士が資源をバーター取引するより高速な P2P 通信を実現する。文献 [4, 5] では、P2P システムを Generalized Prisoner's Dilemma (GPD, 一般化された囚人のジレンマ) でモデル化し、利己的なノードを協調させる方法を採用している。文献 [6] では、遅延制約のあるアプリケーション向けに、各ノードでの受信帯域の合計を大きくすることを目標に、マルチキャスト木を構成する各ノードが自身に関する情報を正直に申告することが利益の最大化につながるインセンティブモデルを提案している。

一方、我々の提案手法は、ノードが自身の資源を可能な限り貢献し、かつ送信元から葉ノードまでの最大遅延の低減に貢献できると想定される位置に移動したときに最大となるようにインセンティブを配分するようにしている。各ノードが利己的であり、自らの利益のみを追求する場合にも、システムに貢献した方がノード自身で得られる利益が大きくなるため、結果として各ノードが適切な位置に移動する。このように、自主的にシステムへの貢献度を最大化させるためにインセンティブを用いて遅延を改善させるよう誘導する点で他の研究と異なっている。

2 提案するインセンティブモデル

2.1 前提

提案するインセンティブモデルでは、単一の送信ノードを持つオーバレイマルチキャスト木を対象とする。適当な周期時間 P ごと、各ノードが提供可能な最大次数、各ノードが実際に提供している次数、そのノードの下流ノード数、送信ノードとのリンク遅延、及びそのノードから葉ノードまでの最大遅延に基づき、そのノードに与えるインセンティブ(これを以下、ノードの獲得値とよぶ)を決定する。

提案モデルでは、マルチキャスト木の品質(主に遅延)を改善するような位置に各ノードが移動するよう誘導するための獲得値の配分法を提供する。これにより、より豊富な資源を保持し、インセンティブの獲得を望む自律ユーザに、なるべくマルチキャスト木全体の遅延を改善する可能性の高い位置への移動を自律的に促すことを目的としている。

このようなインセンティブ配分を実現するため、マルチキャスト木全体の構造情報収集およびノードに獲得値を正しく配分するための十分信頼できるノード(例えば第三者もしくは送信ノード)の存在を仮定する。このノードを以下管理ノードと呼ぶ。この管理ノードの役割を以下に述べる。

- 構造情報の管理
各ノードから、そのノードの親ノードおよび子ノードの情報を受け取る。管理ノードとの通信路の暗号化及び電子署名の利用などにより、データ改ざんや他ノードへの成りすましは行われないと仮定する。
- 排他制御
後述するプロトコルにおいて、ノードの移動処理の逐次性を保証する。
- 獲得値の管理
各ノードに獲得値を管理し、インセンティブモデルに合った配分量を決定する。この獲得値はコンテンツ利

用料などの名目で徴収した仮想マネーを原資とする、などが考えられる。

また、各ノードは自身の獲得値を最大とする行動を常にとるものとし、他ノードも同様の行動をとると想定しているものとする。すなわち、このモデルにおいてなるべく多くの獲得値を稼ぐことをノードの行動基準とする。このため、木情報の申告などの行動も想定すべきであるが、ここでは管理ノードにおいて情報の正しさが保証されているものと仮定する。

2.2 インセンティブ配分法の設計

各ノードが獲得値の最大化を目指して行動した結果、マルチキャスト木の最大遅延が改善されるようなインセンティブモデルを提案する。

基本的なアイデアは以下のとおりである。まず、各ノードは以下の 2 つの条件を満たすよう行動すれば獲得値が増加するようにインセンティブモデルを決定する。

1. 自身から送信ノードまでの遅延を短くする
2. 自身を根とする部分木の密度(以下、下流密度とよぶ)とその部分木のノード数(以下、下流ノード数とよぶ)の積を増加させる

上記の条件から、獲得値増加を目指すノードは、より高い位置(送信ノードに近い位置)へ移動しようとする。高い位置にいるノードは、自身への接続を希望するノードの中で、下流ノード数が多く、かつ下流密度の高いノードを子ノードとして選択する。結果として、下流ノード数が多く、かつ下流密度が高いノードが、高い位置にあるノードの子ノードとして選ばれる。多くのノードを含む部分木が現在の位置より高い位置へ移動していくことで木の遅延が部分的に改善され、下流密度の高い部分木が上流に配置されていくことで結果的に木全体の遅延が改善される。

このような行動を実行させるための獲得値 $Get(v)$ を定義する。その前に、以下の表記を導入する。 N で参加ノード数を、 T_v で v を根とする T の部分木を表し、 $|T_v|$ で T_v のノード数を、 $H(T_v)$ で v からの最大遅延を、 $up(v)$ で v から送信ノードまでの T 上での遅延を、 $d_u(v)$ でノード v のその時点での出次数 (T ノード数) を、 $d_{max}(v)$ でノード v の申告出次数を表す。また、下流密度と下流ノード数の積を $down(v)$ で表す。ノード v の下流密度を $density(T_v)$ で表すと、

$$density(T_v) = \frac{|T_v|}{H(T_v)} \quad (1)$$

$$down(v) = density(T_v) \cdot |T_v| = \frac{|T_v|^2}{H(T_v)} \quad (2)$$

で表せる。

なお、各ノードの申告出次数と平均遅延からマルチキャスト木を構築し、この木を最大遅延が削減された理想的な状態であると仮定し、この木を目標木と呼ぶ。与えられた次数に対し、その次数を持つノードが目標木のどのあたりに存在するかにより、申告出次数が $d_{max}(v)$ であるノード v における $up(v)$ 、 $down(v)$ の目標値を決定する。申告出次数が $d_{max}(v)$ であるノード v の $up(v)$ 、 $down(v)$ の目標値を、 $Up(d_{max}(v))$ 、 $Down(d_{max}(v))$ と表す。

このもとで、ノードの獲得値を、送信ノードまでの遅延に基づき決定される値と、下流密度と下流ノード数の積に基づき決定される値の和で表す。具体的には、 $Get(v)$ を以下のように定義する。

$$Get(v) = G_1(v) + G_2(v) \quad (3)$$

$G_1(v)$ は, $up(v) > Up(d_{max}(v))$ のとき,

$$G_1(v) = (1 - \frac{|up(v) - Up(d_{max}(v))|}{Up(d_{max}(v))}) \cdot S_1(d_{max}(v)) \quad (4)$$

$up(v) \leq Up(d_{max}(v))$ のとき,

$$G_1(v) = S_1(d_{max}(v)) \quad (5)$$

とする。また, $G_2(v)$ は, $down(v) < Down(d_{max}(v))$ のとき,

$$G_2(v) = (1 - \frac{|Down(d_{max}(v)) - down(v)|}{Down(d_{max}(v))}) \times S_2(d_{max}(v)) \quad (6)$$

$down(v) \geq Down(d_{max}(v))$ のとき,

$$G_2(v) = S_2(d_{max}(v)) \quad (7)$$

とする。

ここで, $S_1(d_{max}(v))$, $S_2(d_{max}(v))$ は, それぞれ $G_1(v)$, $G_2(v)$ の最大値であり, ノードの申告出次数に応じて決定される。これらの値の決め方については後述する。式 (4), (6) は, v が理想の位置から離れている, すなわち $up(v)$ と $Up(d_{max}(v))$, および $down(v)$ と $Down(d_{max}(v))$ の値の差が大きいくほど, 獲得値は少なくなり, 目標値に近い場所に存在するほど獲得値の値が改善されていくことを示している。よって, $G_1(v)$ の値を大きくするためには, ノード v は $up(v)$ の値を小さくする, つまり, 送信ノードまでの遅延が短くなる位置へ移動すればよく, $G_2(v)$ の値を大きくするためには, ノード v は $down(v)$ の値を大きくする, つまり, (下流密度と下流ノード数の積) の値を大きくすればよい。

また, 式 (5), (7) は, 理想の位置より上位に移動した場合は, それ以上獲得値の改善がなく, 移動するメリットがないことを示している。ここで, 能力の低いノードが理想の位置より上位に位置してしまうと, それ以上移動しようとしないう状況が考えられる。しかしこのときも, そのノードの下流の部分木がそのノードの能力以上の品質を持つことがないため, 上流ノードの要求を満たすことが出来ず, いずれは親ノードから切り離されて他のノードと入れ替わることで適正な木に改善される。

以上より, 前述の条件 1 を満たせば $G_1(v)$ が増加し, 条件 2 を満たせば $G_2(v)$ が増加する。従って, ノード v は獲得値 $Get(v)$ を最大化させるために,

$$\min up(v) \quad (8)$$

$$\max \frac{|T_v|^2}{H(T_v)} \quad (9)$$

を目指して各オペレーションを実行すればよい。

獲得値の割り当て 獲得値の与え方によっては, 次数を過少もしくは過大に申告したり, 申告した出次数を使い切らずに保持しておくノードが生じることも考えられる。各ノードになるべく多くの出次数を申告をさせ, かつそれを最大限利用させるため, 適切な獲得値を与える必要がある。今, 申告次数を m , 使用次数を n とすると, この使用次数での $up(n)$ の最小値, $down(n)$ の最大値は n での目標値 $Up(n)$, $Down(n)$ である。そこで, 使用次数 n のときの獲得値の最大値として, 以下の $E_1(m, n)$, $E_2(m, n)$ を定義する

$$E_1(m, n) = (1 - \frac{|Up(m) - Up(n)|}{Up(m)}) \cdot S_1(m)$$

$$E_2(m, n) = (1 - \frac{|Down(m) - Down(n)|}{Down(m)}) \cdot S_2(m)$$

このとき, 申告次数はなるべく多く, 使用次数も申告次数の範囲内でなるべく多くすることが獲得値を増大させるようにする, すなわち, $0 < \forall i < m, \forall x$ としたとき,

$$E_1(m - i, m - i) > E_1(m, m - i) \quad (10)$$

$$E_1(m, m) > E_1(m - i, x) \quad (11)$$

$$E_2(m - i, m - i) > E_2(m, m - i) \quad (12)$$

$$E_2(m, m) > E_2(m - i, x) \quad (13)$$

の条件を満たすように, $S_1(m)$, $S_2(m)$ を与える。

式 (10), (12) は, 次数を m と申告しながら, $m - i$ の次数しか使用しなかった場合よりも, $m - i$ と申告して $m - i$ の次数を完全に使用した方が獲得値が高くなるための条件を表している。また, 式 (11), (13) は, 次数を $m - i$ としたノードがどのような次数の使い方をしても, 次数を m と申告したノードが次数を使い切った場合の獲得値には及ばないための条件を表している。

これらの条件が満たされるように $S_1(m)$, $S_2(m)$ を設定することで, 自分が提供可能な出次数を偽って多く申告した場合は, 提供可能な出次数の範囲内ですべて提供した場合よりも獲得値が減少するため, 各ノードに実際に提供する予定の出次数と等しい出次数を正直に申告するように促すことが可能となる。また, 申告した出次数はすべて使い切ったときに最も多くの獲得値が得られるので, 各ノードの出次数の使い切りを防ぐこともできる。

アプリケーションに参加する全ノードはデータの受信料として一定額 P を管理ノードに支払う必要があるとする。よって, 管理ノードが受け取る受信料の合計は $N \cdot P$ である。このとき, 関数 $S_1(k)$, $S_2(k)$ は以下の式を満たす。

$$\sum_{i=1}^N S_1(i) + \sum_{j=1}^N S_2(j) = N \cdot P \quad (14)$$

2.3 次数制約付遅延最小木

提案するインセンティブモデルの目標は, 利己的なノードから構成されるオーバレイマルチキャスト木を, 下記の整数線形計画問題の解で決定されるマルチキャスト木 (次数制約付遅延最小木) になるべく近い最大遅延を持つ木に収束させることである。

今, オーバレイネットワーク $G = (V, V \times V)$ およびオーバレイネットワーク上の被覆木 $T = (V, E)$ ($E \subseteq V \times V$), T の送信ノード $s \in V$ が与えられているとする。さらに, $V \times V$ の各リンク (i, j) には遅延 $F(i, j)$ が与えられている。また, 各ノードには供給可能出次数 $D(v)$ が与えられているとする。

このもとの, 提案プロトコルは, かつ各ノードの出次数がそのノードの申請出次数を超えない範囲で s からの最大遅延 $delay$ が最小となるような被覆木 $T' = (V, E')$ を構築する。すなわち, 以下の式を満たすことが必要である。

$$\min delay \quad (15)$$

ノード i の s からの遅延を $up(i)$ で表すと, 定義より

$$\forall i; up(i) \leq delay \quad (16)$$

また, 各リンク $(i, j) \in V \times V$ に対し, $(i, j) \in E'$ である場合に 1, そうでない場合に 0 である 0-1 整数変数 $e(i, j)$ を導入する。 (V, E') が被覆木を構成するためには, 以下の 4 条件が成り立つ必要がある。

$\forall(i, j) \in V \times V (j \neq s);$

$$up(j) \geq up(i) + F(i, j) * e(i, j) - fmax * (1 - e(i, j)) \quad (17)$$

$$\sum_{(i, j) \in V \times V} e(i, j) = |V| - 1 \quad (18)$$

$\forall(i, j) \in E', j \neq s;$

$$1 \geq \sum_{l: (l, i) \in V \times V, i \neq s, l \neq j} e(l, i) \geq e(i, j) \quad (19)$$

$$\forall v \in V; D(v) \geq d(v) \quad (20)$$

ただし、 $F(i, j)$ はリンク (i, j) の遅延を、 $fmax$ は木の最大遅延より十分大きい定数値を表す。また、 $d(v)$ は v において現在使われている出次数を表す。式 (17) の第 3 項は $(i, j) \notin E'$ である場合、この式自体を常に真とするために導入している。また、式 (18) は木となるための、(19) は、 (V, E') が被覆木であるための必要条件を表し、式 (20) は出次数制約が満たされていることを表す。

この問題は、Degree Bounded Minimum Diameter Tree (DBMDT) の構築問題として知られ、NP 困難であることが示されている [10]。

3 インセンティブ改善のためのプロトコル

3.1 初期木

プロトコルの簡単のため、木に参加するノードの処理に関しては比較的簡潔な方法を採用する。木に参加するノードは、管理ノードのアドレスを知っているとす。管理ノードに参加要求を行うと、木上に参加要求がマルチキャスト送信され、出次数の空きがあり受け入れ可能なノードが参加ノードに応答する。その情報を用いてそれらとのリンク遅延を測定し、最もリンク遅延が短く、かつ出次数の空きがあるノードと接続することで木に参加する。

3.2 ノードの動作

オペレーションの実行権を表すメッセージ (トークンメッセージ) を管理ノードから受け取ったノードだけが下記のオペレーションの実行できる。自律ユーザの性質から、通常は自身のインセンティブ獲得値が改善される場合のみこれを実行する。

- 新たな親ノードと接続
- 現在の親ノードと離脱
- 新たな子ノードを接続
- 現在の子ノードを切断

親ノードの選択 この節では各ノードが獲得値を増加させるために各ノードがどのような親ノード、子ノードを選択すればよいかを述べる。

基本的にノード v は送信ノードまでの遅延を現在の値より短くできるノード w 、すなわち、 $F(w, v)$ をリンク (w, v) 間の遅延としたとき、

$$up(w) + F(w, v) < up(v)$$

であるノード w に子ノードとして接続すれば、獲得値を増加させることができる (図 1 に例を示す)。接続後の送信

ノードまでの距離 $f'(v)$ が短いほど獲得値の増加分が大きくなる。ただし、現在の $up(v)$ の値が

$$up(v) \leq Up(d_u(v))$$

の範囲にあるときは、上流へ移動しても獲得値は変わらない (2.2 節の式 (4), (5) 参照)。

オペレーションの実行権を持つノード v は管理ノードを通じて自身より高い位置にいるノードに対して自身の下流情報 (下流ノード数、自身から葉までの最大遅延) とともに接続要求メッセージを送る。ノード v は自身との接続を希望するノード (w とする) があれば、ノード w とのリンク間遅延を計測し、管理ノードに必要な情報 (ノード w から送信ノードまでの遅延) を問い合わせ、接続後の獲得値を計算する。ノード w も v から受け取った情報をもとに獲得値を計算する。両方のノードの獲得値が増加する場合にのみ、ノード v はノード w の子ノードとして接続する。ノード v の接続候補ノードが複数ある場合は、ノード v は接続後、最も獲得値が高くなるような、つまり、接続後、自身と送信ノードまでの距離が最も短くなるようなノードを選び接続する。

子ノードの選択 接続要求メッセージを送信してきたノードを子ノードとして接続するか否かは以下のように判定する。ここでは、ノード w がノード v から接続要求メッセージを受け取った場合を考える。ノード w は自身の獲得値が最も大きくなるような選択をするものとする。ノード w は、結果的に以下の $down(w)$ の値が大きくなるような選択をすることで、獲得値の増加分も大きくすることができる。

$$down(w) = \frac{|T_w|^2}{H(T_w)} = \frac{(w \text{ の下流ノード数})^2}{(w \text{ から葉までの最大遅延})}$$

ただし、現在 $down(w) \geq Down(d_u(w))$ の範囲にあるときは、ノード v を子ノードとして接続しても獲得値が増加することはない (2.2 節の式 (6), (7) 参照)。一方、 $down(w) < Down(d_u(w))$ の範囲にあるときは、ノード w はまず以下の $D1, D2, D3$ を計算する。

- $D1$: 現在の獲得値
- $D2$: 自身に空き次数がある場合、その空き次数を利用してノード v を子ノードとして受け入れるときの獲得値
- $D3$: 自身の現在の子ノードを切断し、ノード v を受け入れる場合の最大の獲得値。そのとき切断する子ノードを j とする。

ノード w は $D1, D2, D3$ の値に応じて以下のような行動をとる。

- $D1$ が最大ならば、ノード v を子ノードとして受け入れない
- $D2$ が最大ならば、出次数の空きを利用してノード v を子ノードとして受け入れる (図 1)
- $D3$ が最大ならば、ノード j を切断し、ノード v を子ノードとして受け入れる (図 2)

空き次数がない場合も、現在接続しているある子ノードを切断して、新たな子ノードを受け入れることによる獲得値の増分が大きければ、その子ノードを切断する。切断されたノードは 3.2 節で述べた条件を満たす親ノードを探す。

オペレーションの実行権を持つノード w は管理ノードを通じて、自身より送信ノードから遠い位置に存在するノードに対して、送信ノードまでの遅延の情報を含んだ、子ノードへ招聘する接続要求メッセージを送る。ノード w は自身との接続を希望するノード (v とする) があれば、そのノードとのリンク間遅延を計測し、管理ノードに必要な情報 (ノ

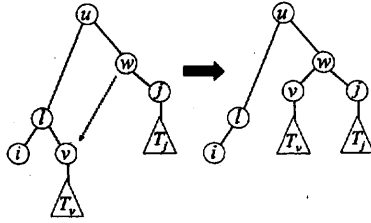


図 1: 空き次数を用いた接続要求の受け入れ

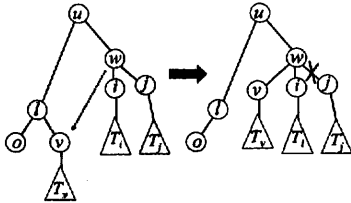


図 2: 既存の子ノードの切断による接続要求の受け入れ

ド v の下流ノード数、ノード v から葉までの最大遅延) を問い合わせ、獲得値を計算する。ノード v も w から受け取った情報をもとに獲得値を計算する。両方のノードの獲得値が増加する場合のみ、ノード v はノード w の子ノードとなる。ノード w との接続を希望するノードが複数ある場合は、ノード w は、最も獲得値の増分を大きくするようなノードを選ぶ。

3.3 木の遅延の改善

ここでは各ノードがインセンティブ増大を目指して 3.2 節で述べたオペレーションを実行していくとき、どのように木の遅延が改善されていくかについて述べる。図 1、図 2 のいずれの場合もノード v がノード w の子ノードとして接続することで、ノード v 、ノード w どちらのノードもともに獲得値が増加するものとする。

図 1 の場合、ノード v がノード w に接続した結果、ノード v から送信ノードまでの距離は移動前より短くなり、ノード v の下流のノードそれぞれについても、送信ノードまでの遅延は移動前より短くなる。ノード v 、ノード w の上流ノードであるノード u から葉までの最大遅延 $H(T_u)$ に関しても現在より短くなり、増大することはない。図 2 の場合も同様に、ノード u の下流のノードに関しては送信ノードまでの遅延が短くなることはあっても増大することはない。切断されたノード j は、移動後、できるだけ送信ノードからの遅延が短くなるようなあるノード x を探し、下流ノードごと移動する。ノード j から送信ノードまでの遅延が短くなる時には、送信ノードから葉までの最大遅延が短くなることもあっても増大することはない。しかし、ノード j から送信ノードまでの遅延が移動後のほうが増大するようなノードと接続した場合、最大遅延が一時的に増大する可能性はある。その場合、ノード j の下流に位置する各ノードから送信ノードまでの距離も長くなり、各ノードの獲得値が減少する。獲得値の減少を不満に感じたノードは、移動後送信ノードからの遅延が短くなるような親ノードを探し、そのノードに接続する。よって図 2 のようにノードの切断を伴う場合も、各ノードが獲得値増大を目指す限り、木全体の遅

延は徐々に改善されると期待できる。

4 シミュレーション実験

4.1 シミュレーション設定

提案プロトコルの簡単なシミュレータを実装し、性能評価を行った。

まず、3.1 節で記述した方法でマルチキャスト木を構築し(以下この木を初期木と呼ぶ)、その後、各ノードが獲得値増加を目指して動くものとする。セッション開始後、セッションへの参加、セッションからの離脱は行わないものとした。各ノードの提供可能出次数は 1~5 の整数値からランダムに選択した。

4.2 実験結果と考察

実験 1 最大遅延の評価 提案アルゴリズムを用いて構築された木の最大遅延、初期木の最大遅延、集中制御型の遅延最小木構築アルゴリズム (CT アルゴリズム [8]) を用いて構築された木の最大遅延について評価した。

各ノード間のリンク遅延が、平均遅延=100[ms]、標準偏差=10[ms]、20[ms]、30[ms] となる正規分布に従うような各場合について評価した。

標準偏差=10[ms] の場合 (図 3)、初期木と比べると最大遅延が大幅に改善し、CT アルゴリズムを用いた場合に構築される木の最大遅延の値と近い値を得ることができた。また、標準偏差=10ms, 20ms としてリンク間遅延の差の度合いを大きくした場合 (図 4, 図 5) でもほぼ同様の結果となった。これにより集中制御型の遅延改善アルゴリズムと同等の性能をもつことが示せたといえる。

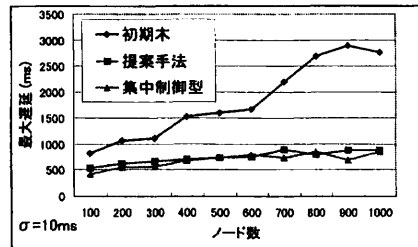


図 3: ノード数に対する最大遅延 ($\sigma=10\text{ms}$)

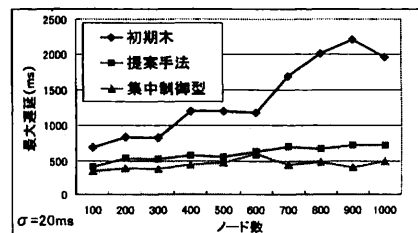


図 4: ノード数に対する最大遅延 ($\sigma=20\text{ms}$)

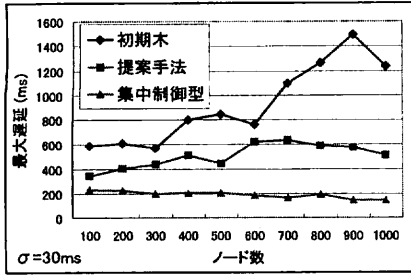


図 5: ノード数に対する最大遅延 ($\sigma=30\text{ms}$)

実験 2 リンクの張り替え数 遅延改善に伴うノード間リンクの張り替え数を評価した。各ノード間のリンク遅延は、平均遅延=100[ms]、標準偏差=20[ms]の正規分布に従うとする。表 1 はノード数 100 ~ 900 の各場合での張り替え数を表している。張り替え数がノード数の 2 倍以下におさえられていることが確認できた。

各ノードはそれぞれの申告出度数に応じて決められた理想的な位置に移動することで獲得値が最大となり、理想の位置に到達したノードはそれ以上移動する要求を持たないため、移動を行わない。結果としてリンクの張り替え数を減らすことができていると考えられる。

ノード数	100	300	500	700	900
張り替え数	38	213	423	1100	1580

表 1: ノード間リンクの張り替え数

実験 3 ノードの振舞いによる最大遅延の変化 獲得値を増やす方向に行動しないノードが存在する場合の最大遅延を評価した。ここでは、参加ノード数は 500 とし、各ノード間のリンク遅延は、平均遅延=100[ms]、標準偏差=20[ms]の正規分布に従うとする。図 6 より獲得値を増やす方向に行動するノードが全体の 70% 以上存在する場合は、最大遅延が CT アルゴリズムを用いた場合の 1.5 倍以下におさえられていることがわかった。

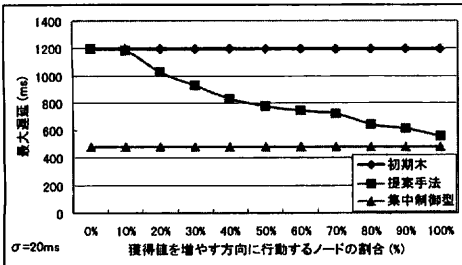


図 6: ノードの振舞いによる最大遅延の変化

5 おわりに

本稿では利己的に振る舞うエンドノードに対し、データの転送コストや送信ノードからの遅延に基づいて、適当な代償をインセンティブとして配分し、各ユーザがインセンティブの改善を目指すことで、効率のよいマルチキャスト木を構築する手法の提案を行った。このプロトコルに対し、シミュレーションによる性能評価を行った結果、集中制御によるマルチキャスト木改善プロトコルと比較しても、遅延改善に伴うノード間リンクの張り替え数を少なく維持したまま、遅延の改善は遜色ないものとなっていることが確認できた。

参考文献

- [1] Y. Chu, J. Chuang, and H. Zhang. A case for taxation in peer-to-peer streaming broadcast. In *Proceedings of the ACM SIGCOMM Workshop on Practice and Theory of Incentives in Networked Systems (ACM SIGCOMM PINS)*, pp. 205–212, August 2004.
- [2] A. Habib and J. Chuang. Incentive mechanism for peer-to-peer media streaming. In *Proceedings of 12th IEEE International Workshop on Quality of Service (IWQoS'04)*, pp. 171–180, June 2004.
- [3] K. G. Anagnostakis and M. B. Greenwald. Exchange-based incentive mechanism for peer-to-peer file sharing. In *Proceedings of the 24th International Conference on Distributed Computing Systems (ICDCS2004)*, pp. 524–533, March 2004.
- [4] M. Feldman, K. Lai, I. Stoica, and J. Chuang. Robust incentive techniques for peer-to-peer networks. In *Proceedings of the 5th ACM Conference on Electronic commerce (EC'04)*, pp. 102–111, 2004.
- [5] M. Afergan and R. Sami. Repeated-game modeling of multicast overlay. In *Proceedings of the Conference on Computer Communications (IEEE INFOCOM 2006)*, April 2006.
- [6] S. Yuen and B. Li. Strategyproof mechanisms for dynamic multicast tree formation in overlay networks. In *Proceedings of the Conference on Computer Communications (IEEE INFOCOM 2005)*, pp. 2135–2146, March 2005.
- [7] W. Z. Wang, X. Y. Li, Z. Sun, and Y. Wang. Design multicast protocols for non-cooperative networks. In *Proceedings of the Conference on Computer Communications (IEEE INFOCOM 2005)*, pp. 1596–1607, March 2005.
- [8] S. Y. Shi, J. S. Turner, and M. Waldvogel. Dimensioning server access bandwidth and multicast routing in overlay networks. In *Proceedings of Network and Operating System Support for Digital Audio and Video (NOSSDAV 2001)*, pp. 83–92, June 2001.
- [9] S. Banerjee, C. Kommareddy, K. Kar, B. Bhattacharjee, and S. Khuller. Construction of an efficient overlay multicast infrastructure for real-time applications. In *Proceedings of the Conference on Computer Communications (IEEE INFOCOM 2003)*, pp. 1521–1531, March 2003.
- [10] S. Y. Shi and J. S. Turner. Multicast routing and bandwidth dimensioning in overlay networks. *IEEE Journal on Selected Areas in Communications*, Vol. 20, No. 8, pp. 1444–1455, 2002.