

モバイル・マルチキャストルーティング問題に対する 最適パス選択解法の提案

常村 和史[†], 田島 滋人[†], 藤井 雅和^{†††}, 中田 明夫[†],
横平 徳美^{††}, 船曳 信生^{††}, 東野 輝夫[†]

[†] 大阪大学 大学院 基礎工学研究科 情報数理系専攻

^{††} 岡山大学 工学部 通信ネットワーク工学科

^{†††} 九州松下電器株式会社

本稿では、モバイル・マルチキャストルーティング問題に対する最適パス選択解法 OPSAM (Optimal Path Selection Algorithm for Multicast routing) を提案する。ネットワークポロジを表すグラフ、情報の送信元であるソースノード、受信元であるディスティネーションノードの集合、遅延時間の上限値が与えられた時に、まず、各ディスティネーションノードまでの経路上の遅延時間が上限値以下で、辺の総コストが最小となるマルチキャスト木の探索を行う。次に、動的に移動するユーザの集合が与えられ、各ユーザから最も近いノードをディスティネーションノードとしてマルチキャスト木を逐次修正する。この時、各状態でのマルチキャスト木の総コストと、修正前後でのマルチキャスト木の変化分との重み付き線形和が最小となるマルチキャスト木集合の探索を行う。OPSAM では、遅延制約を充足する経路候補をあらかじめ複数個同時に抽出し、その中の経路選択を行うことにより、高速に高精度の解を出力する。ランダムグラフと、現実のネットワークに近い Tiers モデルに対するシミュレーションを行い、高性能の解を短時間で出力することを示す。

キーワード: マルチキャストルーティング, モバイル・マルチキャスト, 通信ネットワーク, NP 完全, 近似解法

A Proposal of an Optimal Path Selection Algorithm for Mobile Multicast Routing Problems

Kazufumi Tsunemura[†], Shigeto Tajima[†], Masakazu Fujii^{†††}, Akio Nakata[†],
Tokumi Yokohira^{††}, Nobuo Funabiki^{††} and Teruo Higashino[†]

[†] Department of Informatics and Mathematical Science, Osaka University

^{††} Department of Communication Network Engineering, Okayama University

^{†††} Kyushu Matsushita Electric Co. Ltd.

In this paper, we present an approximation algorithm called OPSAM (Optimal Path Selection Algorithm for Multicast) for the newly defined mobile multicast routing problem. Given a graph with the cost and delay associated to each edge, a data source, a destination set, and the delay time limit, this problem requires finding a multicast tree such that the total edge cost is not only minimized, but also the delay of any path from the source to a destination does not exceed its limit. The problem requires finding a sequence of multicast trees which should be modified to follow changes of mobile user locations, such that the sum of the total edge costs and the difference between two consecutive trees is minimized. OPSAM first extracts plural path candidates for each destination to satisfy the delay constraint. Then, it repeatedly selects a better path among candidates so as to find a low-cost tree within a short computation time. The performance of OPSAM is verified through simulations in the random graph and Tiers model. The simulation results show the performance of OPSAM is better than BSMA.

Keywords: multicast routing, mobile multicast, communication network, NP-complete, approximate algorithm

1 はじめに

近年、ビデオオンデマンド (VoD) やビデオ会議システムといったマルチキャスト・アプリケーションが盛んに研究されている。マルチキャストはネットワーク上での通信形態の1つで、1つのホスト (ソース) から複数のホスト (ディスティネーション) へ、同時に同一のデータを送信する方法である。

ネットワーク上でこのようなマルチキャストセッションを多数実行するためには、各セッションでのコストを最小化することが重要である。同時に、QoS (Quality Of Service) 要求を満たすために、一定の遅延時間内に情報を伝送することが要

求されることがある。セッション毎に、コストの最小化、あるいは上記の2つの条件を同時に満たす経路選択を表すマルチキャスト木の探索が重要であり、マルチキャストルーティング問題と呼ばれている。本問題は NP 完全問題である [1]。

セッションを開始してから終了するまでディスティネーションノードの追加・削除が行われない静的マルチキャストルーティング問題に対してすでにいくつかの解法が提案されている [1]-[5]。これらの中で、文献 [3] の BSMA (Bounded Shortest Multicast Algorithm) は、遅延制約有り、ネットワーク形状が非対称な問題に対し、最も精度の高い解を得られることが知られている [4][10]。しかしながら BSMA は時間計算量が大き

く、解を得るのに多くの時間を必要とする。

一方、動的マルチキャストルーティング問題に対しては様々な形での定式化が行われているが、本研究では、各ユーザが動的に移動するモバイル環境における動的マルチキャストルーティング問題を考え、モバイル・マルチキャストルーティング問題と呼ぶこととする。

本稿では、モバイル・マルチキャストルーティング問題に対する新しい近似アルゴリズムである最適パス選択解法 OPSAM (Optimal Path Selection Algorithm for Multicast-tree) を提案する [6]。

OPSAM の性能評価は、ランダムグラフ生成法の 1 つである Waxman 法、および、Tiers モデルに基づく方法で生成した 2 種類のネットワークポロジを用いたシミュレーションにより行う。モバイル・マルチキャストルーティング用に改良した BSMA との比較を通して、OPSAM が BSMA よりも高性能の解を高速に出力することを示す。

2 モバイル・マルチキャストルーティング問題

本稿で対象とするモバイル・ネットワークは、各モバイルホストと直接、無線によりデータ通信を行う無線基地局と、同じ地域の複数の無線基地局を管理する中継局、中継局間のデータ通信を実現する有線のネットワークから構成されている。中継局間の各リンクにはコスト・遅延時間が与えられる。ここでコストとは、そのリンクで使用されている帯域幅などのネットワーク資源、遅延時間とは情報を伝送する際の伝送遅延を表す。そして、移動通信を行うユーザの集合が入力として与えられる。各ユーザの現在の位置はタイムスロットと呼ばれる一定の時間間隔毎に報告される。そしてユーザには、タイムスロット毎の位置に最も近い基地局から、無線でデータが送信されるものとする。モバイル・マルチキャストルーティング問題とは、タイムスロット毎に、ユーザ位置から最も近い基地局を管理する中継局をディスティネーションノードとして、マルチキャスト木の修正を繰り返す問題である。本問題では、中継局間のネットワークを対象として、マルチキャスト木を求めることが要求される。したがって、モバイル・マルチキャストルーティング問題は以下のように定式化される。

- 入力：有向連結グラフ $G = (V, E)$ 、リンクコスト $C(l)$ 、遅延時間 $D(l)$ 、ソースノード s 、ユーザ集合 U 、総タイムスロット数 $tslot$ 、ユーザ位置の系列集合 $u_{it}(1 \leq t \leq tslot, 1 \leq i \leq |U|)$ 、遅延時間の上限 $\Delta_t(1 \leq t \leq tslot)$
- 出力：マルチキャスト木の集合 $T_t = (V_{T_t}, E_{T_t})(1 \leq t \leq tslot)$ 。ここで、 D_t を t 番目のタイムスロットでのユーザ位置から得られるディスティネーションノードの集合とすると、 $\{s\} \cup D_t \subseteq V_{T_t} \subseteq V, E_{T_t} \subseteq E$ である。
- 制約条件：ソースノードから各ディスティネーションノードへの経路を構成するリンクの遅延時間の総和が Δ_t 以下
- 目的関数：マルチキャスト木を構成するリンクのコストと修正前後での木の変化分 (使用されなくなったリンクの遅延時間) の総和であり、式 (1) で表される。

$$\sum_t \left\{ \sum_{l \in E_{T_t}} C(l) + K \times \sum_{\substack{l \in E_{T_t-1} \\ l \notin E_{T_t}}} D(l) \right\} \rightarrow \min \quad (1)$$

式 (1) において、 E_{T_t} は t 番目のタイムスロットでのマルチキャスト木を構成するリンクの集合を表す。また、 K は

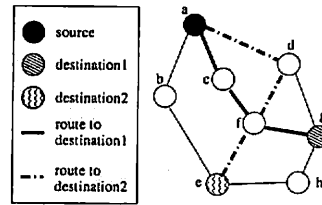


図 1: 閉路の発生

定数項を表す値であり、本論文のシミュレーションではその値を 3 としている。これは、一例題 ($n = 100, |D| = 50$) を対象として K の値を 1 から 10 まで変化させた所、 $K = 3$ までは木の変化量が急激に小さくなったのに対し、それ以上では大きな変化が見られなくなり、木全体のリンクコストを最小化するために K を 3 と設定した。

3 提案する最適パス選択解法 OPSAM

3.1 OPSAM の狙いと構成

モバイル・マルチキャストルーティング問題に対して、最適パス選択解法 OPSAM を提案する。本アルゴリズムでは、計算速度の高速化と解の高精度化を狙って、以下の 3 段階で構成する。

1. 経路候補の生成
2. 解の探索
3. マルチキャスト木の修正

提案アルゴリズム OPSAM では、経路探索に要する時間を短縮するために、第 1 段階でソースから各ディスティネーションに対し、一度に複数個の経路探索を行い、その結果を経路候補として記憶する。第 2 段階で経路候補の置換を繰り返すことにより、最適な経路を探索する。そして第 3 段階でマルチキャスト木の修正を行う。

ここで、各ディスティネーションノードへの経路候補は、基本的に他のディスティネーションノードへの経路候補とは独立に生成される。そのため、第 2 段階で経路候補を置換した際に、異なるディスティネーションノードへの経路候補同士が図 1 のような閉路を作り、マルチキャスト木のコストが大きくなる可能性がある。明らかに、閉路を構成する部分経路 $\langle a, c, f \rangle, \langle a, d, f \rangle$ の一方は不要である。そこで第 1 段階において、異なるディスティネーションノードへの経路候補同士が閉路を作るかどうかを調べ、その結果を記憶する。そして第 2 段階で閉路が発生した場合には、図 2 のように閉路を作るディスティネーションノードへの経路候補を、閉路を作らない経路候補に強制的に置換することによりコストの低下 (解の高精度化) を図る。

ユーザの移動に伴いディスティネーションノードに変更が加わった場合、第 1 段階、第 2 段階の手法を用いて、マルチキャスト木の修正を行う。以降では、各段階での動作について説明を行う。

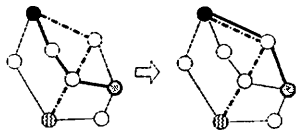


図 2: 閉路の除去

3.2 第1段階：経路候補生成段階

経路候補生成段階では、 k -経路候補生成法 [11] に基づき、ソースノードと各ディスティネーションノード間に対して、遅延時間制約を充足し、後述する選択関数 F の値の小さいものから、後述する関数 K_{d_i} で与えられる個数の経路を生成し、それらを経路候補として記憶する。そして経路候補の中から、順次、マルチキャスト木のコストの増加が最小となるように経路を1つ選択し、マルチキャスト木に加えていくことで初期解を生成する。さらに、異なるディスティネーションノード間の経路候補が閉路を作るかどうかを調べ、その結果を記憶する。以下にそのアルゴリズムを示す。

1. ユーザ位置からディスティネーションノードを決定する。
2. Dijkstra アルゴリズム [12] により、ソースノードから各ディスティネーションノードへの遅延最小経路とその遅延時間を求める。
3. 各ディスティネーションノード $d_i \in D$ について、2. で得られた遅延時間の降順に、
 - (a) 制約条件を充足する経路について、 k -経路候補生成法に従い、 K_{d_i} (式 (2)) 個の経路を、選択関数 F (式 (3)) の値の小さいものから順次生成し、経路候補として記憶する。
 - (b) 経路候補の中から、マルチキャスト木のコストの増加が最小となる経路を1つ選択し、木に加える。
4. 生成したマルチキャスト木 T とそのコストを記憶する。ここで、以下の変数を用いる。
 - $T_{best} = T$: 生成した最良のマルチキャスト木
 - $C_T = \sum_{e \in T} C(e)$: マルチキャスト木 T の総コスト
 - $C_{best} = C_{T_{best}}$: 最良のマルチキャスト木 T_{best} の総コスト
5. ディスティネーションノードが異なる、任意の2つの経路候補間で、閉路を作るかどうかを調べ記憶する。

3.2.1 経路候補数関数 K_{d_i} の設定

経路候補数は、計算時間短縮のためには少ない方が望ましいが、解精度の向上のためにはある程度数を用意し、経路候補に多様性を持たせる必要がある。以下の3つの要因を考慮した関数 K_{d_i} で与えることとした。

要因 (1) あるディスティネーションノードへの経路の多様性は、ネットワーク上の全ノード数に依存して増加する。

要因 (2) 異なるディスティネーションノードへの経路の組合せによる閉路の発生を防止するためには、ディスティネーション数に依存して経路候補を多様化する必要がある。

要因 (3) あるディスティネーションノードへの経路の多様性は、それとソースノードとの距離に依存して増加する。

ここで、厳密に多様な経路数を算出することは不可能であり、また、問題毎に経路候補数が極端に変化することを避けるために、本論文では、経路候補数関数 K_{d_i} を単純に、上記の3要因に比例する項の線形和で構成することとした。但し、 A は定係数である。

$$K_{d_i} = A \times ((\text{ノード数}) + (\text{ディスティネーションノード数}) + (\text{ソースノードと } d_i \text{ の距離})) \quad (2)$$

3.2.2 選択関数 F の設定

コストの小さいマルチキャスト木を求めるためには、ソースノードと各ディスティネーションノードの間でコストの小さい経路を求めるだけでなく、各ディスティネーション間でできるだけリンクを共有する必要がある。そこで選択関数 F を以下のように定義した。 α は定係数である。

$$F = (\text{経路候補のリンクのコストの和}) - \alpha \times (\text{他のディスティネーションノードに対する経路候補と共有するリンクのコストの和}) \quad (3)$$

3.3 第2段階：解探索段階

解探索段階では、ディスティネーションノードを1つ選択し、そのディスティネーションノードに対する、現在の選択経路以外の経路でコスト最小となるものを経路候補の中から選択し置換する。置換した結果閉路が発生した場合には、閉路を作るディスティネーションノードへの経路候補を、閉路を作らない経路候補に強制的に置換する。閉路を作らない経路候補が存在しない場合には、そのノードを経路未割りリストに保存し、次回以降で経路割当を行う。この操作を繰返し行うことで解の探索を行う。以下にアルゴリズムを示す。

1. 経路未割りリストの中からディスティネーションノード d_i を1つランダムに選択する。経路未割りリストが空の場合は、ディスティネーションノードを1つランダムに選択する。
2. d_i に対し、現在の経路以外の候補の中で、 C_T が最小となる候補を選択し、新たに T とする。
3. 経路未割りリストが空であり、 $C_T < C_{best}$ なら、 $T_{best} := T$, $C_{best} := C_T$ とする。
4. 今回選択された候補により閉路が作られるかどうかを調べ、閉路が存在する場合には以下の操作を行う。
 - (a) 2. で選択された経路以外の閉路を作る経路を消去する。
5. 一定回数 (L とする) 繰り返したら 6. へ進み、そうでなければ 1. に戻る。
6. 経路未割りリストが空であれば終了し、そうでなければ、以下を実行した後に終了する。
 - (a) 経路未割りリストの中からディスティネーションノードを順次選択し、そのノードに対して C_T が最小となる候補を選択し、 T とする。
 - (b) $C_T < C_{best}$ なら $T_{best} := T$, $C_{best} := C_T$ とする。

3.4 第3段階：マルチキャスト木の修正

マルチキャスト木を修正する段階では以下の点を考慮する必要がある。

- ユーザの移動に伴い、修正前にはディスティネーションノードではなかったノードが、新たにディスティネーションノードとなる。
- ユーザの移動に伴い、修正前にはディスティネーションノードであったノードが、新たにディスティネーションノードではなくなる。

この2点を考慮し、以下にそのアルゴリズムを示す。

1. ユーザ位置を取得する。
2. ユーザ位置からディスティネーションノードを決定する。
3. もし、修正前にディスティネーションノードであったノードがディスティネーションノードではなくなる場合には、
 - (a) 記憶している経路候補を削除する。
 - (b) 閉路発生に関する情報を削除する。
4. もし、修正前にディスティネーションノードではなかったノードがディスティネーションノードとなる場合には、
 - (a) 第1段階:経路候補生成段階の方法で、経路候補を生成し記憶する。
 - (b) 経路候補の中から、目的関数値の増加が最小となる経路を1つ選択し木に加える。
 - (c) 他のディスティネーションノードとの間で、その任意の2つの経路候補間で閉路を作るかどうかを調べ、その結果を記憶する。
5. 第2段階:解探索段階の方法で解を改善する。
6. タイムスロット数 $tslot$ 回繰り返したら7.へ進み、そうでなければ1.に戻る。
7. 解を出力して終了する。

ここで、OPSAMでは、異なるディスティネーションノードへの経路間でリンクを共有する経路候補の抽出が、その精度向上に不可欠である。モバイル・マルチキャストルーティング問題では、そのためにまず、3.においてディスティネーションノードではなくなったノードへの経路候補をすべて削除した上で、4.において新たなディスティネーションノードへの経路候補を、既存の経路候補とリンクを極力共有するように抽出し直すことにより、計算時間を抑制しながら、現在のディスティネーションノード分布に適した経路候補の抽出を実現している。

4 モバイル・マルチキャストルーティング問題に対する性能評価

提案する OPSAM を C 言語によって実装し、Pentium III 866MHz 上でシミュレーションを行った。比較手法としては、BSMA[3]を改良したものを用い、ランダムグラフ生成法の1つとして広く使用されている Waxman 法 [7]、Tiers モデル [8][9]によるグラフポロジを対象として、両者の解精度・実行時間を比較した。

4.1 シミュレーション例題の生成

4.1.1 Waxman 法による例題生成

例題の生成は、文献 [3]における Waxman 法 [7]を用いた手法により行った。

- ネットワークポロジ: ノードの平均次数が5程度になるように生成した。
- リンクコスト: 2ノード間のマンハッタン距離とした。
- 遅延時間: (リンクコスト) \times (0~1のランダムな実数)+1とした。
- ソースノード: ランダムに決定した。
- 遅延上限 Δ : d_{max} を、各ディスティネーションノードへの遅延時間が最小となるマルチキャスト木の中で、遅延時間が最大となるディスティネーションノードまでの遅延時間とし、全例題について、 $\Delta = d_{max}$, $d_{max} \times 1.2$, $d_{max} \times 1.5$ の3通りでシミュレーションを実行した。
- 例題のサイズ: ノード数は、網目状に接続されたネットワークの現実的なサイズを考慮して、25, 50, 75, 100の4種類とした。また、ディスティネーションノード数は、ノード数の半分までの4~50の18種類とした。
- 例題数: 乱数の偏りを防ぐために、各サイズにつき、異なる乱数を用いて100例題を生成した。
- BSMAの superedge 置換の際に探索する経路候補数 M は、ノード数の2分の1とした。

また、ユーザ位置 u_{ti} については以下のように設定した。

- 初期位置 u_{1i} ($1 \leq i \leq |U|$): グラフが生成される $n \times n$ (n はノード数)座標平面の格子点上に、重複を許してランダムに配置した。また、ユーザ毎に移動速度 v_i ($1 \leq i \leq |U|$)を $1 \leq v_i \leq 5$ の範囲の整数値で、ユーザの進行方向を表す x 軸との角度 θ_i ($1 \leq i \leq |U|$)を、 $0 \leq \theta_i \leq 360$ の範囲の整数値で与える。

- t ($1 < t \leq tslot$)番目のタイムスロットでのユーザ位置 u_{ti} : u_{ti} の x 座標を $x_{u_{ti}}$, y 座標を $y_{u_{ti}}$ と置く

$$\begin{aligned}x_{u_{ti}} &= x_{u_{(t-1)i}} + v_i \times \cos\left(\theta_i \times \frac{\pi}{180}\right), \\y_{u_{ti}} &= y_{u_{(t-1)i}} + v_i \times \sin\left(\theta_i \times \frac{\pi}{180}\right)\end{aligned}\quad (4)$$

与えられる。ただし、 u_{ti} が $n \times n$ 座標平面の外に出る場合には、 θ_i に $90 \sim 180$ の範囲のランダムな整数を加えて再度 u_{ti} を求める。また、 θ_i には $-10 \sim 10$ の範囲のランダムな整数を加える。

4.1.2 パラメータの適正化

最大サイズの例題 ($n = 100, |D| = 50$)を対象として、各パラメータ値(経路候補関数 K_d の係数 A , 選択関数 F の係数 α)を変化させた場合のコスト、計算時間の変化結果 [15]より、 $A = 0.2$, $\alpha = 15$, $L = 200n$ とした。マルチキャスト木の修正段階では、修正される経路が少ないことから、 L の値はその5分の1 ($L = 40n$)とした。

4.1.3 Tiers モデルによる例題生成

例題は以下のように生成した。

- ネットワークトポロジー: ネットワークトポロジーの生成には Tiers モデル生成プログラム [13] を使用した。
- リンクコスト: 本プログラムで出力される帯域幅の逆数に比例する値を使用した。これは、得られるマルチキャスト木の各リンクの未使用帯域幅を最大化するためである。
- 遅延時間: 本プログラムで出力される遅延時間を使用した。
- ソースノード: LAN に属するノードの中からランダムに決定した。これは、ソースやディスティネーションとなるサーバやユーザホストは、LAN に接続されていると考えられるからである。
- 遅延上限 Δ : Waxman 法の場合と同様に設定した。
- 例題のサイズ: ノード数は、130, 195, 260 の 3 種類とした。これは、Tiers モデルの WAN, MAN, LAN の 3 階層それぞれにある程度の数のノードを持たせるためである。また、ディスティネーションノード数は、LAN 階層上のノード数に応じて、4~38 の 15 種類とした。
- 例題数: 各サイズにつき異なる乱数を用いて Waxman 法と同様に 100 例題を生成した。
- BSMA の *superedge* 置換の際に探索する経路候補数 M は、ノード数の 2 分の 1 とした。

また、ユーザ位置 u_{ii} については以下のように設定した。

- 初期位置 u_{1i} ($1 \leq i \leq |U|$): LAN 中心からある一定の範囲に、重複を許してランダムに配置した。また、ユーザ毎の移動速度 v_i ($1 \leq i \leq |U|$) を $50 \leq v_i \leq 200$ の範囲の整数値で、ユーザの進行方向は Waxman 法と同様に設定した。
- t ($1 < t \leq tslot$) 番目のタイムスロットでのユーザ位置 u_{ti} : Waxman 法と同様に式 (4) で与えられる。また、 u_{ti} が LAN の中心からある一定の範囲外に出る場合も、同様にして u_{ti} を求める。

4.1.4 パラメータの適正化

Waxman 法による例題の場合と同様に、最大サイズの例題 ($n = 260, |D| = 38$) を対象として、各パラメータ値を変化させた場合のコスト、計算時間の変化結果 [15] より、 $A = 0.04$, $\alpha = 75$, $L = 120n$ とした。マルチキャスト木の修正段階では Waxman 法と同様に、 L の値はその 5 分の 1 ($L = 24n$) とした。このように、トポロジーにより適正なパラメータ値に違いが生じたのは、Waxman は 1 階層ランダムモデルであり、Tiers が 3 階層モデルであるためと考えられる。すなわち前者では、取り得る経路候補数が多く A および L の値を大きくする必要があり、同時に異なるディスティネーションに対する経路間で共有するリンクが相対的に少なくなるために、 α の値を小さくする必要が生じたと考えられる。後者では、異なる層間を接続するリンクが制限され、取り得る候補数は相対的に少なく A および L の値を小さくでき、同時に異なるディスティネーションに対する経路間で共有するリンクが多くなるために、 α の値を大きくする必要が生じたと考えられる。

4.2 シミュレーション結果と考察

シミュレーション結果を表 1, 2 に示す。表中の木変化量は、目的関数の第二項である木の変化分を表す。

これらの結果より、式 (1) の目的関数の値は、Waxman 法の $\Delta = 1.5d_{max}$ の場合を除き、OPSAM が BSMA よりも 0.7~2.5% 優れている。コスト、木変化量を個別に評価した場合、Waxman 法では全般的にコストは BSMA が良く、木変化量は OPSAM が良いという結果が得られた。これは、BSMA では木の変化を大きくしても各木のコストを最小化する能力が高いことを示している。また、Tiers モデルでは、コストは OPSAM が優れ、木変化量はほぼ同等の結果が得られた。すなわち、OPSAM は木の変化は同じながら、コストの小さな木を探索できることを示している。一方、計算時間は全ての例題において、OPSAM は BSMA よりも Waxman 法で 3~10 倍、Tiers モデルで 12~22 倍高速であった。BSMA の問題点は、ディスティネーションノードの追加に伴うマルチキャスト木修正時の全ディスティネーションノードに対しての経路探索が必要な点である。モバイル・マルチキャストルーティング問題では、木の修正を頻繁に行う必要がある。ノード数が多い場合には、経路探索を行う回数はさらに増大し、計算時間も増大する。これに対し、OPSAM では、ディスティネーションノードの追加時においても、そのノードのみに対する経路探索を行うだけで済むため、計算時間を抑えることができる。上記の結果はこのような理由からであると考えられる。

5 まとめ

本稿では、モバイル・マルチキャストルーティング問題に対する最適パス選択解法 OPSAM の提案を行った。Waxman 法、Tiers モデルによるトポロジーを用いたシミュレーションにより、OPSAM が高速に高精度の解を探索することを示した。

今後の課題としては、まず、OPSAM の各パラメータ設定の自動化、および大規模サイズの Waxman 法に対する OPSAM の精度改善のために、経路候補数関数の見直しや解探索段階での経路候補の再抽出が挙げられる。また、ハンドオーバー時のデータ損失を防ぐプロトコルの研究と、それに関連したモバイル・マルチキャストルーティング問題の定式化の見直し、並びにそのためのアルゴリズムの改良が挙げられる。更に、サーバとしてミラーサイトが複数設置された場合に対応する、複数ソースノードからのマルチキャスト木 (森) 問題に取り組むことが重要と考えられる。

参考文献

- [1] V. P. Kompella, J. C. Pasquale and G. C. Polyzos, "Multicast routing for multimedia communication", IEEE/ACM Trans. Networking, Vol. 1, No. 3, pp.286-292(1993).
- [2] R. Sriram, G. Manimaran and C. Siva Ram Murthy, "Algorithms for delay-constrained low-cost multicast tree construction", Computer Commun., Vol. 21, pp.1693-1706(1998).
- [3] M. Parsa, Q. Zhu and J. J. Garcia-Luna-Aceves, "An iterative algorithm for delay-constrained minimum-cost multicasting", IEEE/ACM Trans. Networking, Vol. 6, No. 4, pp.461-474(1998).
- [4] X. Jia, "A distributed algorithm of delay-bounded multicast routing for multimedia applications in wide area

表 1: Waxman 法による例題の結果

		OPSAM				BSMA			
Δ	$ D $	コスト	遅延時間	木変化量	計算時間	コスト	遅延時間	木変化量	計算時間
d_{max}	4~25	2841.3	158.4	3316.5	0.631	2818.8	178.3	3353.7	6.617
d_{max}	38,50	8182.7	342.6	9210.3	5.585	8215.4	408.8	9441.9	28.886
$1.2d_{max}$	4~25	2673.6	149.5	3122.1	0.927	2658.4	162.4	3145.6	6.764
$1.2d_{max}$	38,50	7900.5	330.6	8892.3	7.471	7855.8	378.6	8991.7	29.209
$1.5d_{max}$	4~25	2606.7	143.6	3037.6	1.265	2565.2	153.2	3024.8	6.800
$1.5d_{max}$	38,50	7797.8	320.2	8758.5	9.218	7663.5	355.6	8730.4	29.531

表 2: Tiers モデルによる例題の結果

		OPSAM				BSMA			
Δ	$ D $	コスト	遅延時間	木変化量	計算時間	コスト	遅延時間	木変化量	計算時間
d_{max}	4~25	9242.4	2353.5	16303.0	1.853	9326.0	2356.8	16396.5	41.122
d_{max}	38	14883.2	5380.0	31023.1	4.547	15088.7	5399.0	31285.9	62.088
$1.2d_{max}$	4~25	8639.6	2340.0	15659.5	2.037	8787.0	2334.3	15790.0	43.968
$1.2d_{max}$	38	14353.6	5368.9	30460.2	4.834	14686.5	5367.4	30788.7	65.476
$1.5d_{max}$	4~25	8626.8	2338.6	15642.8	2.231	8770.2	2333.7	15771.2	44.104
$1.5d_{max}$	38	14347.3	5367.5	30450.0	5.343	14677.4	5366.4	30776.7	65.995

- networks*", IEEE/ACM Trans. Networking, Vol. 6, No. 6, pp.828-837(1998).
- [5] Q. Zhang and Yiu-Wing Leung, "An orthogonal genetic algorithm for multimedia multicast routing", IEEE Trans. Evolutionary Computation, Vol. 3, No. 1, pp.53-62(1999).
- [6] 藤井 雅和, 船曳 信生, 中田 明夫, 北道 淳司, 東野 輝夫, "静的マルチキャストルーティング問題に対する最適パス選択解法の提案", 信学技報, Vol. 100, No. 568, pp.39-46(2001).
- [7] B. M. Waxman, "Routing of multipoint connections", IEEE J. Select. Areas. Commun., Vol. 6, pp.1617-1622(1988).
- [8] K. L. Calvert, M. B. Doar and E. W. Zegura, "Modeling Internet Topology", IEEE Commun. Magazine, June, pp.160-163(1997).
- [9] M. B. Doar, "A Better Model for Generating Test Networks", Proc. GLOBECOM '96 (1996).
- [10] H. F. Salama, D. S. Reeves and Y. Viniotis, "Evaluation of multicast routing algorithms for real-time communication on high-speed networks", IEEE J. Select. Areas. Commun., Vol. 15, No. 3, pp.332-345(1997).
- [11] 馬場 孝之, 船曳 信生, 西川 清史, "無線通信網の通信経路割当て問題を対象としたグリーディニューラルネットワーク解法の提案", 電子情報通信学会論文誌, Vol. J81-D-I, No. 6, pp700-707(1998).
- [12] T. H. Cormen, C. H. Leiserson and R. L. Rivest, "Introduction to algorithms", MIT press(1990).
- [13] M. B. Doar, "A Random Network Topology Generator", <http://www.pobox.com/~doar/>.
- [14] C.-C. Tseng, K.-H. Chi, and T.-L. Huang, "A new locality-based IP multicasting scheme for mobile hosts", Computer Commun., vol. 24, pp.486-495 (2001).
- [15] "OPSAM における各パラメータ値とコスト, 計算時間の変化", <http://www-ocf.ics.es.osaka-u.ac.jp/multicast/opsam.html>.