

# アプリケーションレベルマルチキャスト Emma の 性能向上に関する検討

中村嘉隆\* 廣森聡仁† 山口弘純\* 安本慶一‡ 東野輝夫\*

本稿では、電子会議などエンドホスト間で実時間ビデオを交換するグループアプリケーション向けのアプリケーションレベルマルチキャストプロトコル Emma の性能向上、特に安定性の向上に関する検討及び性能評価を行う。Emma は、エンド間ユニキャストからなるオーバーレイネットワーク上にメディア配信木を構築し、エンドホストの帯域制約のもとで、ユーザ満足度がなるべく満たされるよう、オーバーレイリンク上のビデオ転送制御を動的に完全分散制御で行う。この Emma の安定性向上のために、エンドホストの離脱に対し既存のビデオ配信をなるべく維持するための機能を Emma の分散制御を損わない形で追加した。シミュレーション実験を行った結果、エンドホストの参加、離脱が繰り返された後でも、ユーザ満足度を維持できていることがわかった。

## Improving Robustness of Application Layer Multicast Protocol Emma

Yoshitaka Nakamura\*, Akihito Hiromori†, Hirozumi Yamaguchi\*,  
Keiichi Yasumoto‡ and Teruo Higashino\*

In this paper, we add a new functionality to an application level multicast protocol called Emma, designed for audio/video multi-party applications, in order to improve the robustness. In Emma, source-based trees for media distribution are constructed on an overlay network where the end-hosts have direct connections between them. Video packet forwarding is dynamically controlled in a distributed manner, so that user preference can be satisfied as much as possible. The functionality is to maintain existing media distribution when an end-host leaves the overlay network. The experimental results have shown that high satisfaction of users (user preference) could be achieved after iterative joining and leaving of end hosts.

### 1 はじめに

高速ネットワークや計算機の普及は、近い将来、比較的小規模（数人から数十人程度）の多数のグループによるビデオチャットのようなグループ通信をもたらすと予想される。しかし、そのようなグループ通信におけるデータ交換を限られた数のサーバが処理することは、ネットワーク資源及びコネクション数の制限の観点から現実的ではない。

グループ通信は主にグループ内ユーザへの同報通信からなるため、特定のサーバを必要とせず、ネットワーク資源の利用効率も高いマルチキャストが有用な通信技術の一つであると考えられる。しかし、特に広域網（バックボーン）でのインフラストラクチャの整備の問題から、現状での利用は難しい。一方、すべての2ユーザ間のユニキャストでそのような同報通信を実現することは、グループサイズのスケールの点で非効率的である。

このような問題に対する現実解として、マルチキャストをアプリケーション層で実現する通信形態（アプリケーションレベルマルチキャスト, ALM）が注目を集めており、近年になり多くの研究がなされている [4, 5, 6, 9]。ALM はエンドホスト間ユニキャストトネリングによるオーバーレイネットワーク上で、それら

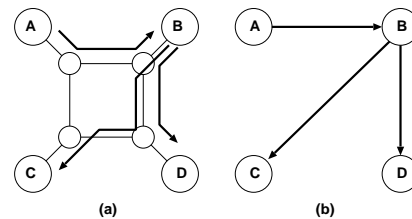


図 1: アプリケーションレベルマルチキャスト

エンドホストがマルチキャスト配信木を管理し、トネリング間のパケット複製及び転送を行なうことで実現される。例えば、図 1(a) ではホスト B は A からのデータを複製し、C, D に転送している。また、図 1(b) はマルチキャスト配信木を表す。

我々は文献 [1] において電子ビデオ会議などのグループアプリケーション向けの ALM プロトコルである Emma (End-user Multicast for Multi-party Applications) を提案している。Emma は、(1) エンドホストがそれぞれ同時並行かつ継続的に他のエンドホストにビデオを送出し、(2) 各エンドホストが扱える（受信または転送できる）ビデオ数は、処理能力及びエンドホスト付近の帯域を考慮した場合には一定の制限があり、(3) 各エンドホストが受信したいビデオには偏りがある（プリファレンスがある）、といった特徴を持つユーザアプリケーション（主にビデオ会議など）を対象とし、オーバーレイネットワークの自律的構築とプリファレンスに応じたビデオの動的転送制御を完全な分散制御で行なう。文献 [1] では、Emma の基本機能

\* 大阪大学 大学院情報科学研究科  
Graduate School of Information Science and Technology, Osaka University  
† 大阪大学 大学院基礎工学研究科  
Graduate School of Engineering Science, Osaka University  
‡ 奈良先端技術大学院大学 情報科学研究科  
Graduate School of Information Science, Nara Institute of Science and Technology

を実現するためのプロトコル設計概論を述べているが、Emma をより堅牢かつ高性能なプロトコルとするためには、アプリケーションレベルマルチキャストの特徴の一つであるエンドホストの流動性（セッション途中での参加や離脱が比較的しやすい傾向）に対しても、既存セッションをできる限り維持することが望まれる。

本稿では、Emma の性能向上のため、エンドホストの離脱に対しても、既存のビデオ配信をなるべく維持するための機能を、Emma の基本的な設計概念（完全分散制御）を損なわない形で追加し、その性能を評価した。シミュレーション実験の結果、エンドホストが参加、離脱を繰り返した時も、オーバーレイネットワークの効率、ユーザ満足度が維持できていることがわかった。

## 2 プロトコル Emma の概要

### 2.1 Emma によるビデオ配信アーキテクチャ

Emma は複数のエンドホスト（以下ノードと呼ぶ）とそれらの IP アドレスを管理する一つのロビーサーバからなる。各エンドホストは潜在的なビデオの送信者であり、他のいくつかのノード間にユニキャストコネクションを確立する。これをオーバーレイリンクと呼ぶ。どのノードとオーバーレイリンクを確立するかは、参加時にロビーサーバから入手したノードリスト上のノード間遅延時間から決定する。簡単のため、各ノードが送信するビデオが利用する帯域は同じであるとする。

各ノードは自身のオーバーレイリンクに何本のビデオを同時に配信できるかをあらかじめ指定しておく。

各ノードは定期的なフラッディングにより最短経路木を構成し、その経路木を表すオーバーレイネットワーク上のルーティングテーブルを維持する。他ノードからのビデオデータパケットをどのノードに転送するかを、このルーティングテーブルをもとに決定する。また、各ノードは複数のノードが発信するビデオを受信するが、それらをどの程度受信したいかを表す値（プリファレンス値）を設定しているとする。

図2にオーバーレイネットワークの例を示す。このオーバーレイネットワークでは二本のビデオ  $V_A, V_F$  を配信している。ノード  $A$  は自身のビデオ  $V_A$  をノード  $B, C, D, E, F$  に図のような経路で配信しており、それらのノードは  $V_A$  に対するプリファレンス値を、それぞれ 3, 6, 4, 1, 4 のように与えている。また、ノード  $F$  は  $V_F$  をノード  $A, B, C, E$  に図のような経路で配信しており、それらのノードは  $V_F$  に対するプリファレンス値を、それぞれ 3, 3, 2, 5 のように与えている。

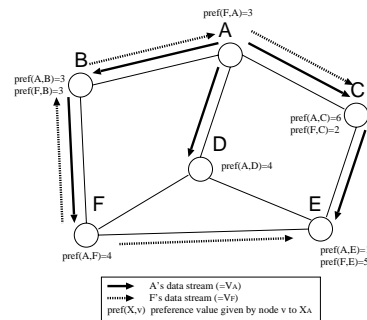


図 2: Emma によるビデオ配信とプリファレンス値の例

### 2.2 Emma によるビデオ配信制御

ビデオの配信においてはそれぞれのビデオがある程度の帯域を恒常的に消費するため、オーバーレイリンク上に空き帯域を十分に確保できない場合も多い。Emma ではその場合、プリファレンス値に基づき、既存のビデオの配信停止によるプリファレンス値の損失をなるべく小さくしながら空き帯域を確保できる可能性のあるリンクを選択する。

**プリファレンス値集約** ノード  $v$  のビデオ  $V_s$  に対するプリファレンス値を  $pref(s, v)$  で表す。各ノード  $v$  は受信中のビデオ  $V_s$  に対し、 $v$  を根とする  $V_s$  配信木上のノードのプリファレンス総和  $Pref(s, v)$  を含むメッセージ  $Media/Keep$  を、適当な時間周期間隔ごとに上流ノードに送出する。 $Pref(s, v)$  は  $v$  の子ノード  $w$  からの  $Pref(s, w)$  の総和に自身の  $pref(s, v)$  を加えた値として得られる。

なお、 $v$  が同じ親ノードから複数のビデオを受信している場合は、それらに関する  $Media/Keep$  メッセージは一つにまとめて送出する。これにより、1 オーバレイリンク上に一周時間あたりに送出されるメッセージを一つにできる。

**空き帯域情報とロス値の集約** ノード  $v$  はノード  $s$  からのビデオ  $V_s$  の受信要求を行う場合、ルーティングテーブルに基づき  $v$  の  $V_s$  についての上流ノード  $u$  に対しビデオ配信要求メッセージ ( $Media/JoinREQ$ ) を送信する。

$V_s$  に関する  $Media/JoinREQ$  メッセージは  $Media/Keep$  メッセージと同様、集約されて上流に転送される。この転送は  $V_s$  が既に配信されているノードに  $Media/JoinREQ$  メッセージが到達するまで繰り返される。

ただし、 $Media/JoinREQ$  メッセージが転送されてきた各リンク上（これらのリンクで構成される木を  $V_s$  の要求木とよぶ）に少なくともビデオ一本分の空き帯域を確保しなくてはならない。Emma では既存のビデオの配信停止方法のうち、最もプリファレンス値損失

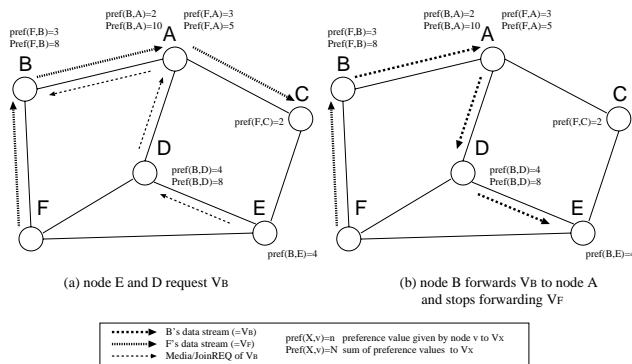


図 3: ビデオ配信の変更例

が小さいような配信停止方法を計算し、その損失が要求受け入れによるプリファレンス値増加を下回る場合は、それらのビデオの配信を停止し、要求を受け入れることで全体として満足されるプリファレンス値を増加させる（計算法の詳細は文献 [1] 参照）。

図 3 に例を挙げる．簡単のために各リンク上のスロットの数は 1、つまり一つのオーバーレイリンクが流すことのできるビデオは一種類とする．ノード  $E$  が  $B$  のビデオに対して Media/JoinREQ メッセージを送ったとき、図 3 (a) のように転送されるとする．オーバーレイリンク  $B-A$  間では既にビデオ  $V_F$  が流れているので、配信停止をするかどうかを決定するために、各ビデオのノード  $A$  までのプリファレンス総和  $Pref(B, A)$ ,  $Pref(F, A)$  (= 損失) を比較する．この例では、 $Pref(B, A) = 10$ ,  $Pref(F, A) = 5$  であるので、図 3 (b) のように、 $B-A$  での  $V_F$  の配信を停止し、 $V_B$  の配信を開始する．

なお、受信要求の受け入れ手続きは以下で述べる．

受信要求の受け入れ Media/JoinREQ メッセージを受け取ったノード  $v$  上に  $V_s$  がすでに配信されている場合（ $v = s$  の場合も含む）、 $v$  は Media/JoinREQ メッセージを受け取った各子ノード  $w$  について、 $w$  に配信しているビデオ  $V_i$  のうち、 $V_i$  の配信停止によるプリファレンス値損失の最小値が  $Pref(s, w)$  より小さいなら、 $V_i$  から  $V_s$  へのビデオ変更命令メッセージを、そうでなければ変更不可通知メッセージを  $w$  に向けて送信する．

### 3 ノード離脱時における木の再構築

中継ノードとなっていたあるノードが離脱した場合には、ノード間の相互接続性、及びユーザ満足度（プリファレンス）をなるべく維持するため、離脱したノードを介してビデオを受信していた隣接ノードはそれぞれそのビデオの配信木上のいずれかのノードに再接続し、ビデオ配信木の維持を試みる．Emma は完全な分

散制御をその設計目標の一つとしているため、再接続先ノード情報もその枠組を維持しながら収集できることが望ましい．ただし、ノード離脱のためのメッセージを新たに定義することはプロトコルのオーバーヘッドや複雑性を増すために望ましくない．また、再接続後の配信木においては、ビデオのソースノードから各受信ノードへの遅延（ホップ数）がなるべく小さく、かつ各ノードの空き帯域がバランス良く残されていることが、安定したビデオ配信には望ましいといえる．

以上の要求を満たすように我々は以下のような機能を Emma に追加した．各ノード  $v$  は、各ビデオ  $V_i$  についての MEDIA/Keep メッセージに対し、一定以上の空き帯域がありかつソースノード  $i$  からのホップ数が小さい順に、 $v$  を根とする  $V_i$  の配信木上のノードから定数個を選び、そのアドレスリスト（ノードリスト）を含める．MEDIA/Keep メッセージを受け取ったノードは、自身の子ノードを根とする  $V_i$  の配信木についての上記ノードリストが得られるため、それらと自身の空き帯域情報から、自身を根とする  $V_i$  の配信木に関するノードリストが計算できる．

このもとで、あるノードが離脱した場合、そのノードを介してビデオ  $V_i$  を受信していた隣接ノード  $v$  は、自らに流されているビデオ  $V_i$  の配信木において離脱ノードの親ノードに、前述のノードリストを問い合わせる．ノードリスト内のあるノードに対してオーバーレイリンクが存在しなければ構築し、そこにいつなぎ変える．これにより、 $v$  を根とする  $V_i$  の配信木はビデオ配信を維持できる．また、ノード  $v$  の動作は以下ようになる．

- ビデオ  $V_i$  について、 $v$  の子ノード  $w$  から、ノードリスト  $Empty_w$  を受け取った場合、
  - 自らの空き帯域があらかじめ定められた閾値以上であるならば、
    - \*  $\bigcup_w Empty_w$  から、ホップ数が小さい順に数個を選び  $Empty_v$  として Media/Keep メッセージに付加する．
- $V_i$  についての親ノード  $u$  から離脱メッセージが届いた場合、もしくは  $u$  とのオーバーレイリンクの切断が検知された場合、
  - $u$  を介して受信していた配信木のビデオのうち、 $u$  を根とする配信木のプリファレンス値総和が大きい順（それぞれ  $V_j$  とする）に再接続を試みる．
  - $v$  は  $V_j$  について  $u$  の親ノード  $u'$  に  $u'$  の各子ノード  $t$  の  $Empty_t$  ( $t \neq u$ ) を問い合わせ、 $Empty_t$  内のノードのいずれかに要求を受け入れられるまで順次接続要求メッセージを送る．
  - 要求を受け入れられた場合、

- \* オーバレイリンクが存在しなければオーバレイリンクを確立し、配信木を更新する。

図4に再構築の例を示す。簡単のために、配信されているビデオは  $V_A$  のみとし、各ノードの空き帯域は3とする。既に Media/Keep メッセージによりプリファレンス総和  $Pref(A, v)$ , 空き帯域を持つノードリスト  $Empty_v$  が、ノード  $A$  まで集約されているとする。ここで、図4(a)のようにノード  $B$  が離脱する場合、離脱時に、子ノード  $E, F$  に向かって離脱メッセージを送出する\*。離脱メッセージを受け取った  $E$  と  $F$  は、 $B$  の親ノードであった  $A$  にメッセージを送り、 $Empty_A$  を得る。 $B$  が離脱したために  $A$  には空き帯域1ができるので、 $B$  の子ノードのうちいずれか（ここでは  $E$ ）と再接続する。 $A$  と再接続できなかったノード（この場合  $F$ ）は、受け取った  $Empty_A$  の中から  $A$  からのホップ数が最も小さいもの（ここでは  $C$ ）を選び、このノードと再接続する。この結果、図4(b)のように配信木が再構築される。

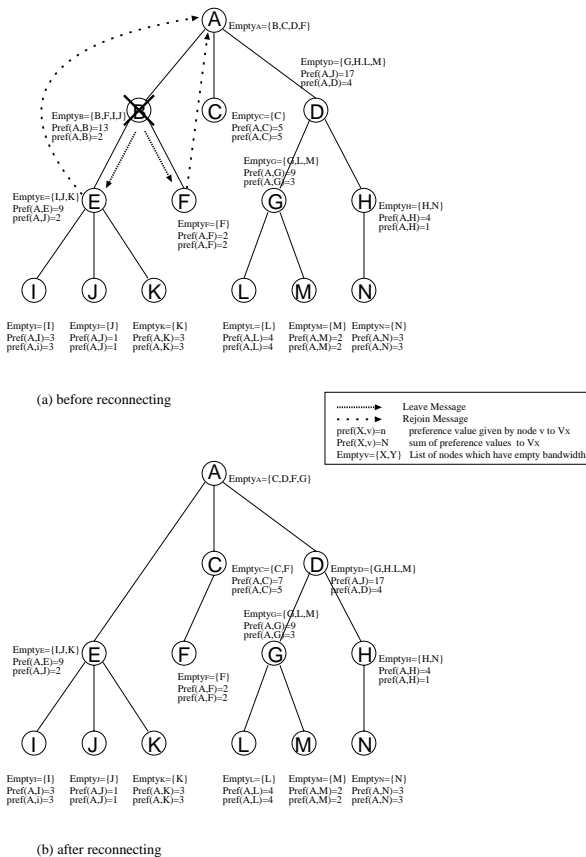


図4: ノード離脱後の再構築例

\* 予告のない離脱については各子ノードが親ノードからのトラフィックをモニタすることで検出する。

## 4 シミュレーションによる性能評価

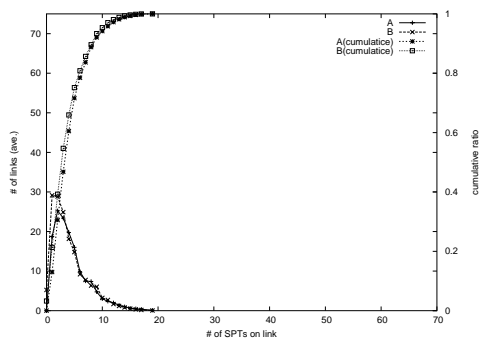
Emma の性能評価のため、我々はスクリプト型言語 Ruby[3] により Emma を実装し、分散イベント型シミュレーションにより性能評価を行った。

### 4.1 ネットワークモデルとシミュレーションシナリオ

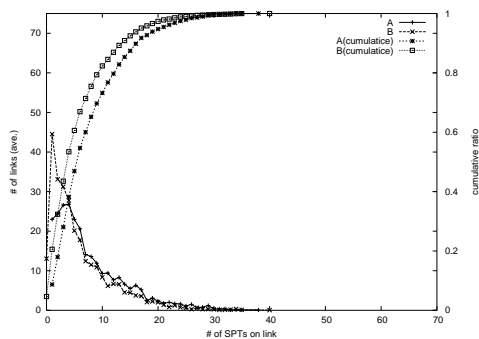
ネットワークは LAN, MAN 及び WAN より構成される階層型トポロジを tiers モデル [2] に基づき生成し、LAN に属するノードの約 50% をエンドホスト（セッションに参加するユーザ）としている。また、LAN, MAN, WAN の帯域はそれぞれ平均で約 2Mbps, 4Mbps 及び 10Mbps となるようランダムに決定し、各ユーザのデータストリーム（映像）は 200kbps であるとした。各ノードはセッションの参加時に 3 つのノードとオーバレイリンクを確立し、一ノードのオーバレイリンクの最大数は 6、各オーバレイリンクの容量は 3, 4, 5 のいずれかとした。

また、下記のシナリオで実験を行った。

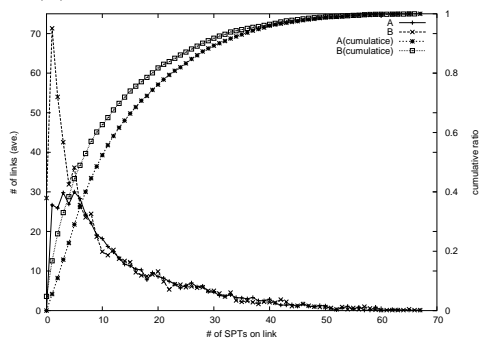
- 全ユーザについて、ユーザ  $i$  ( $i = 1, \dots, N$ ) のビデオに対するプリファレンスを zipf モデルに基づき  $2N/i$  に設定する。これにより、プリファレンスの偏向性を表現する。例えば会議において、議長や会議場の映像は多くの人が受信しようとするためプリファレンス値は一般に高い、などである。
- ユーザは順次セッションに参加し、オーバレイネットワークを構築する。
- 全ユーザの参加後、各ユーザはランダムに選択した4つのビデオの受信要求を順次行う。これによりユーザごと与えるプリファレンスの違いを表現する。その後は、未受信のビデオのうち、受信中のビデオより高いプリファレンスを指定したビデオをランダムに選択し、受信要求を行う動作を、全体のプリファレンス値総和が充分高くなるまで繰り返す。繰り返した後の状態を状態 A とよぶ。
- その後は、ランダムにユーザを三名選択して離脱させ、新しいユーザを参加させる動作を、離脱ユーザ数がもとの全ユーザ数の 10% にあたるまで繰り返す。繰り返した後の状態を状態 B とよぶ。なお、ユーザは LAN に属するノードであるため、あるユーザに関連しないすべての 2 ノード間の実ネットワーク上の経路はそのユーザの離脱に影響されない（すなわち、そのユーザに関連しないオーバレイリンクは影響を受けない）。また新しいユーザは、LAN に属する既存のノードから選択する。



(a) 56 ノード (ユーザ数平均は 36)

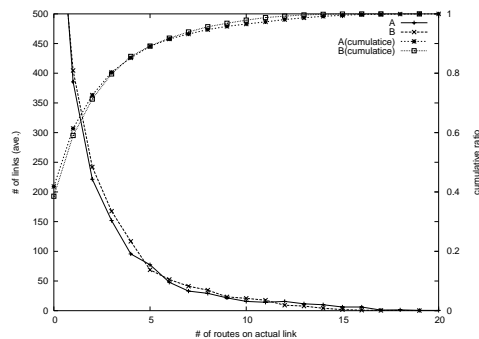


(b) 100 ノード (ユーザ数平均は 66)

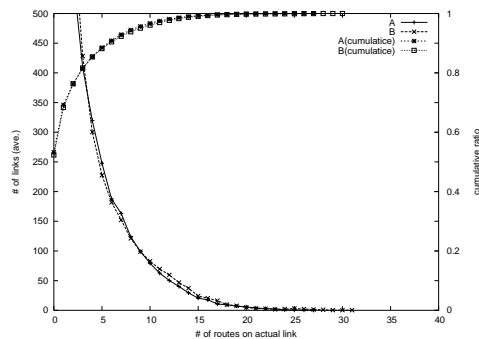


(c) 151 ノード (ユーザ数平均は 120)

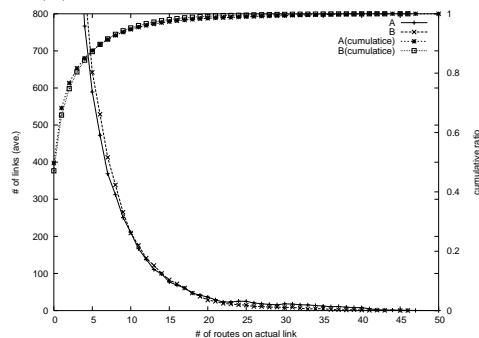
図 5: オーバレイリンクあたりの SPT 数の分布



(a) 56 ノード (ユーザ数平均は 26)



(b) 100 ノード (ユーザ数平均は 66)



(c) 151 ノード (ユーザ数平均は 120)

図 6: 実リンクあたりの単一 SPT の重複度の分布

## 4.2 実験結果

**オーバーレイリンクあたりの木の分布** Emma ではオーバーレイリンク上での SPT 同士の重複度が少ない方がより効率的であると考えられる。したがってオーバーレイリンクあたりいくつの SPT が存在するかを、状態 A (全ユーザが参加して安定した後の状態) 及び状態 B (参加/離脱を繰り返した後の状態) について測定した。その分布を図 5 に示す。図の X 軸は SPT 数を、Y 軸は全オーバーレイリンク中その SPT 数が存在するリンク総数をそれぞれ表す。

図 5 より、状態 A と比較すると、状態 B の方がわずかに重複度は小さくなっているもののほぼ差異がないことが読み取れる。

**実リンクあたりの経路重複度** 前節と同じ実験において、各実リンクが単一の SPT に何回含まれるかを状態 A 及び状態 B について測定した。その分布を図 6 に示す。X 軸は単一の SPT に含まれる回数を、Y 軸が全実リンク中その回数含まれる実リンクの総数を表す。なお、図 6 (a), (b), (c) の X 軸の目盛単位はそれぞれ異なることに注意されたい。図 5 同様、参加/離脱を繰り返したあとも大きな変化はないことがわかる。このために、経路重複度の高いボトルネックリンクが存在しないという ALM の利点は維持できているといえる。

**ユーザ満足度変化** 参加/離脱を繰り返した後の状態 (状態 B) において、状態 A での満足度をどの程度

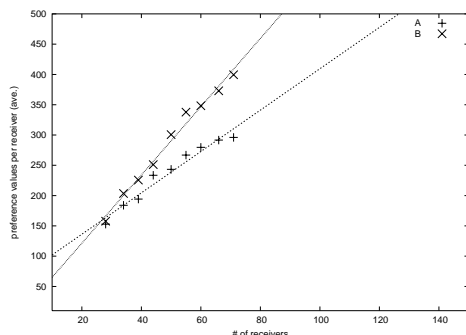


図 7: ユーザあたりの平均プリファレンス値の変化

維持できているかを，ノード数の異なるネットワークにおいて状態 A 及び状態 B でのプリファレンス値総和を測定することで調べた．図 7 にその結果を示す．X 軸はユーザ数，Y 軸はユーザあたりの平均プリファレンス値を表す．いずれもノード数（ユーザ数）の増加に比例して増加するが，状態 B では状態 A の 100% ~ 140% を達成している．なお，状態 B での値が状態 A よりも大きいことについては，再構築時及びノードの新規参加時に新たなオーバレイリンクを追加している場合があり，このときに受信要求を出されているながら配信されていなかったビデオの配信が可能になることが影響しているものと考えられる．

## 5 まとめ

本稿では，ユーザプリファレンスを考慮したアプリケーションレベルマルチキャスト Emma に対し，エンドホストの参加・離脱時における安定性の向上に関する機能を追加し，シミュレーションによる性能評価を行った．シミュレーション実験の結果，機能追加時にも Emma の性能低下は認められなかった．

今後の課題は，Emma によるメディア配信時のジッタやパケットロスの計測等，より詳細なシミュレーション実験による性能評価，Emma のワイヤレスアドホックマルチキャストへの応用などがあげられる．

## 参考文献

[1] 中村嘉隆, 山口弘純, 廣森聡仁, 安本慶一, 東野輝夫, 谷口健一, “映像による複数人のコミュニケーションシステム向けのアプリケーションレベルマルチキャスト Emma の性能評価,” マルチメディア, 分散, 協調とモバイル ( DICOMO2002 ) シンポジウム論文集, pp253–256, 2002.

[2] K. L. Calvert, M. B. Doar and E. W. Zegura, “Modeling Internet Topology,” *IEEE Communications Magazine*, pp. 160–163, 1997.

[3] “Ruby Home Page,” <http://www.ruby-lang.org>

[4] Y.-H. Chu, S. G. Rao, S. Seshan and H. Zhang, “Enabling Conferencing Applications on the Internet using an Overlay Multicast Architecture,” *Proc. of ACM SIGCOMM*, 2001.

[5] Y.-H. Chu, S. G. Rao and H. Zhang, “A Case for End System Multicast,” *Proc. of ACM SIGMETRICS*, 2000.

[6] D. Pendarakis, S. Shi, D. Verma and M. Waldvogel, “ALMI: An Application Level Multicast Infrastructure,” *Proc. of 3rd Usenix Symp. on Internet Technologies & Systems*, 2001.

[7] J. Jannotti, D. Gifford, K. Johnson, M. Kaashoek and J. O’Toole, “Overcast: Reliable Multicasting with an Overlay Networks,” *Proc. of 4th Usenix Symp. on Operating Systems Design and Implementation (OSDI)*, 2000.

[8] V. Roca, A. El-Sayed, “A Host-Based Multicast (HBM) Solution for Group Communications,” *Proc. of 1st IEEE Int. Conf. on Networking (ICN’01)*, 2001.

[9] R. Cohen and G. Kaempfer, “A Unicast-based Approach for Streaming Multicast,” *Proc. of IEEE INFOCOM 2001*, 2001.

[10] Y. Chawathe, S. McCanne and E. A. Brewer, “RMX: Reliable Multicast for Heterogeneous Networks,” *Proc. of IEEE Infocom 2000*, 2000.

[11] P. Francis, “Yallcast: Extending the Internet Multicast Architecture,” <http://www.yallcast.com>.

[12] F. Baccelli, D. Kofman and J. L. Rougier, “Self Organizing Hierarchical Multicast Trees and Their Optimization,” *Proc. of IEEE INFOCOM 2001*, 2001.