

アプリケーションレイヤマルチキャストにおける 効率的な配送木の構築に関する研究

加藤 寧 東北大学大学院情報科学研究科教授

1 本研究の目的

最先端なネットワーク技術であるアプリケーションレイヤマルチキャスト (ALM) は、世界各地で活発に研究・実用化が行われている。しかし、ALM による安定したコンテンツ配信システムの実現には、ツリー構築の観点から以下の2点の問題が残されている。1点目は、ユーザの離脱による映像品質の低下である。2点目は、各ユーザの利用帯域の多様性 (不均一性) を考慮しなければならない点である。これらの問題点を解決することは、次世代のコンテンツ配信技術を確立する上で必要不可欠である。しかしながら、現在提案されている ALM システムでは各ノードの利用可能帯域を考慮しつつ、信頼性の高い配信木を構築することは困難である。

そこで、本研究では、これらの問題点を解決するために、多重の配信ツリーを並列に用いるアプローチをとる。更にユーザの離脱による他のツリーに対しての影響を最小限に抑えるため、1つのツリーで節となっているノードが他の全てのツリーでは葉となっている Node-disjoint なツリー構造を用いて、多重の独立したツリーを構築すると同時に、各ユーザの利用可能帯域を考慮した新たなツリーの構築手法の提案を行う。このようなシステムを実現するために、Node-disjoint なツリー構造を構築可能な手法である THAG

(Topology-aware Hierarchical Arrangement Graph) という手法に注目し、THAG を利用可能帯域が不均一な状況でも、十分な性能を達成できるように改良する。更に提案手法をシミュレーションにより評価し、インターネットを想定した利用可能帯域が多様なネットワークにおいて、既存手法よりも提案手法が有効であることを示す。このような配信ツリーを構築することにより、上記の2点の問題点を勘案した、ノード離脱に対する影響を最小限に抑え、かつネットワーク環境の多様性に適応したALM を確立することを本研究の目的とする。

2 アプリケーションレイヤマルチキャスト (ALM)

2-1 ALM 概要

ALM では、IP マルチキャストのようにストリームの複製・転送をルータが行わず、その役割をエンドノードが担うことにより、ネットワーク層に依存せず、アプリケーション層で仮想的なマルチキャスト機能を実現する (図1)。アプリケーション層では、配信サーバを根、ユーザを節または葉とした配信ツリーを構築し、各ユーザはその配信ツリーに沿ってサーバから配信されたストリームを転送または受信する。そのため、ALM ではユニキャストのみでストリームを多地点へ配信可能である。更に、インターネットのような既存のネットワーク上に配信システムを低コストで実現でき、またサーバの配信負荷やネットワークトラフィックを削減できる。このように、ALM は大規模なコンテンツ配信システムを既存のネットワーク上に低コストで構築可能な技術であり、世界各地で活発に研究・実用化が行われている。しかし、ALM による安定したコンテンツ配信システムの実現には、ツリー構築の観点から、依然として下記の3点の問題が残されている。

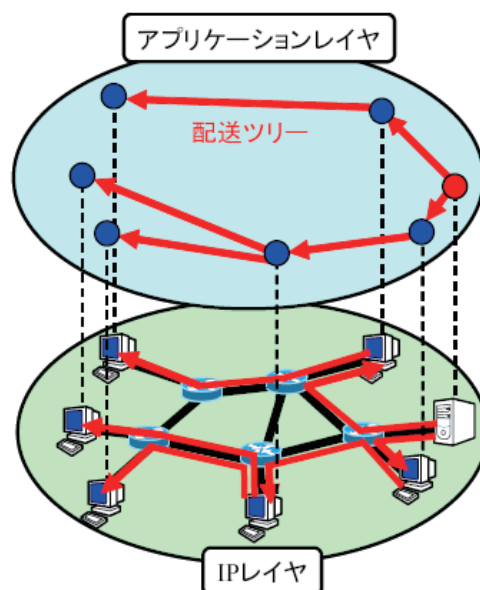


図1: アプリケーションレイヤマルチキャスト。

1 つ目は、ストリームの遅延の増加である。ALM では、マルチキャストツリーを構築しストリームを配信するため、マルチキャストツリーの下位に位置するほど、遅延が増加する。そのため、マルチキャストツリーの高さを小さくしたり、近くのノードからストリームに転送したりする必要がある。

2 つ目は、ユーザの離脱による映像品質の低下である。ALM では、ユーザの参加・離脱が頻繁に起こる。ツリーの上位に位置するユーザの離脱の際に、そのユーザよりもツリーの下位に位置するノードへのストリームの配信が中断し、映像の品質が低下する。そのため、ユーザの離脱の影響をできるかぎり抑える必要がある。

3 つ目に、各ユーザの利用帯域による制約があげられる。現在のインターネットはFTTH, ADSL, 無線ネットワークなどの環境が混在しているため、各ユーザのネットワークへの接続速度は多様である。そのため、各ユーザの利用可能な帯域の制約を考慮した配信ツリーの構築が必要である。

上記の問題点を解決することは、ALM による高品質なストリーム配信を実現する上で必要不可欠である。

2-2 ALM の分類

従来のALM は、配信ツリーの構築方法により、シングルツリーマルチキャスト (図2) とマルチツリーマルチキャスト (図3) に大別できる。

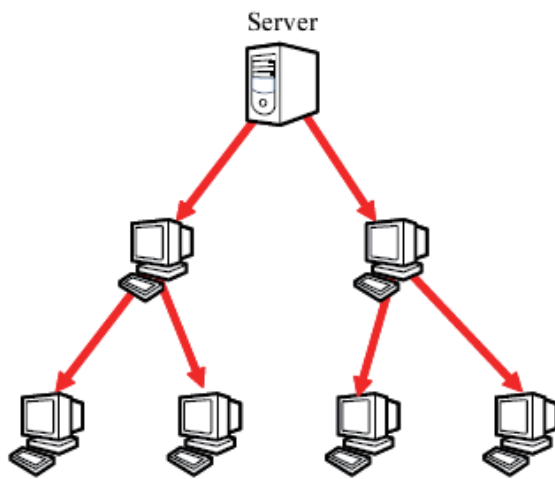


図2: シングルツリーマルチキャスト。

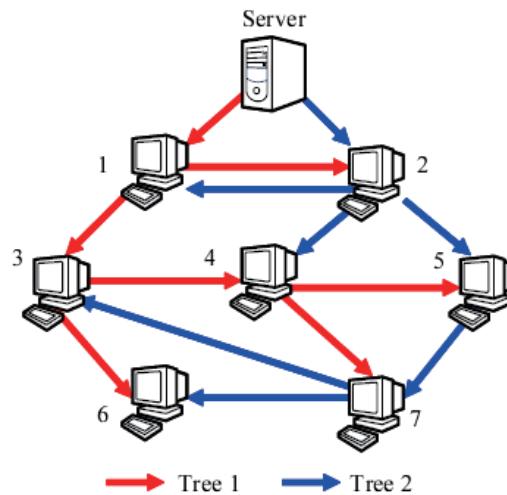


図3: マルチツリーマルチキャスト。

シングルツリーマルチキャストでは、サーバから配信されるストリームを、単一のツリー構造を用いて配信する。しかし、シングルツリーマルチキャストでは、ノードの離脱の際に、ストリームが配信できないノードが多数存在する。そのため、ノードの離脱により、ストリームの受信が途切れる可能性が高い。そこで、ノードの離脱に強い手法として、配信に多重の配信木を用いるマルチツリーマルチキャストが提案されている。マルチツリーマルチキャストでは、ストリームをMDC (Multiple Description Coding) [1], [2]を用いて複数のDescription に分割し、それぞれのDescription に対して、多重の配信木を並列に用いて配信する。MDC は元のストリームを、より小さなサイズを持つ、いくつかのDescription に分割でき、分割されたDescription のどれを受信しても再生可能な仕組みを持っている。更に、多くの分割されたストリームを受信するほど、高い品質で再生可能である。図3 では、配信データを2 つのDescription に分割し、それぞれのDescription に対してツリーを構築している。ALM に参加するノードは全てのツリーに参加することによって、複数のツリーでストリームの配信ができなくなっても、その他のツリーからDescription を受信可能である。そのため、ノードが離脱してもストリームが途切れることなく受信できる。これまでに、マルチツリーマルチキャストとして、CoopNet[9]-[10], SplitStream[8], 及びTHAG (Topology-aware Hierarchical Arrangement Graph) [3]などが提案されている。

特に、THAG はツリーの Node-disjoint 性を保証している。Node-disjointなツリー構造とは、1 つのツリーで親となるノードは、他の全てのツリーでは葉であるようなツリー構造をいう。このようなツリーを構築することにより、ノードの離脱によりDescription が配信できなくなるツリーを1 つのみに抑えることが可能である。図3 のツリーはNode-disjoint なツリー構造であり、例えば、ノード1 はTree 1 では親となっ

ているのに対して、Tree 2 では子となっている。THAG では、このNode-disjoint なツリー構造を、AG (Arrangement Graph) を階層化することにより構築している。そのため、THAG はSplitStream や CoopNet に比べ、ノードの離脱に強い手法である。しかし、THAG ではネットワーク帯域の不均一な環境に適応できないという問題点がある。そこで、本研究ではTHAG に着目し、THAG をノード利用可能帯域が不均一な環境に適応するように改良した。そして、ツリーを分散管理しつつ、各ノードの利用可能帯域を考慮し、更にツリーのNode-disjoint 性を保証する手法の確立を目指す。

3 Topology-aware Hierarchical Arrangement Graph (THAG)

3-1 THAG の概要

THAG ではAG (Arrangement Graph) を階層化することにより、Node-disjoint なツリー構造を構築する。AG の詳細な説明に関しては参考文献[4], [5]に譲り、ここでは省略することとする。サイズ4 のAG を図4 (a)に示し、サイズ4 のAG から構築できる2 本のnode-disjoint なツリーを図4 (b), (c)に示す。サイズs のAG からは、(s-2) 本のnode-disjoint なツリーが構築可能である。そして、それぞれのツリーのルートにDescription を送信することにより、Node-disjoint なツリーに沿ってストリームの配信が可能である。

特にノード21 をエントランスと呼び、エントランスはAG 全体の情報を管理する。

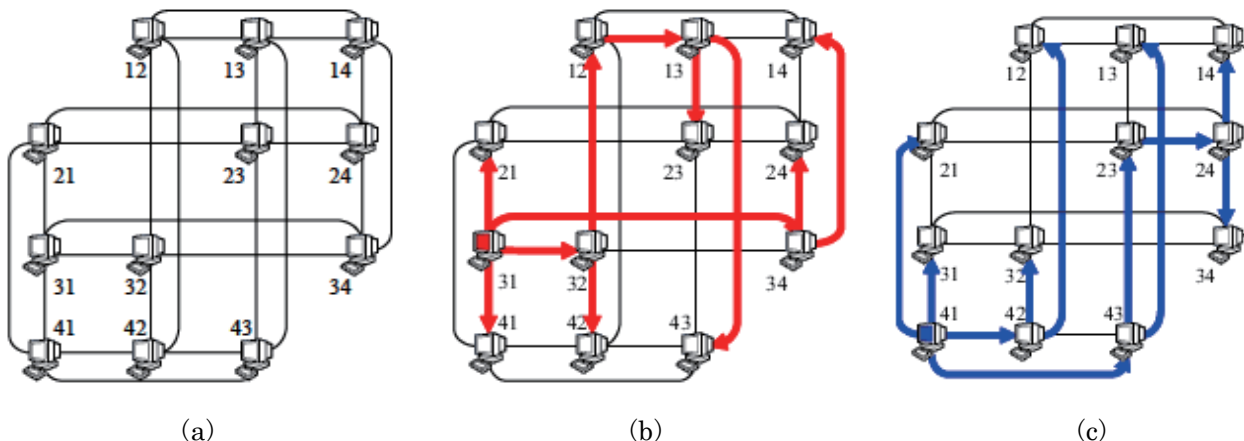


図4: (a) サイズ4 のAG, (b) ノード31 をルートとしたツリー, (c) ノード41 をルートとしたツリー。

1 つのAG のみでは、参加できるノード数が限られているため、AG を階層化することにより、多くのノードをシステムに参加させることが可能である。あるAG の参加可能ノード数が限界に達したとき、そのAG をParent-AG として、新たにChild-AG を作成する。図5 では、Parent-AG におけるノード32, 42 がChild-AG 1 のルートに配信を行い、ノード13, 43がChild-AG 2 のルートに配信を行っている。Child-AG に配信を行うノードの定義などは参考文献[3]に譲り、ここでは省略する。

以上のようなAG の階層化により、配信ネットワークの大規模化が容易に達成できる。

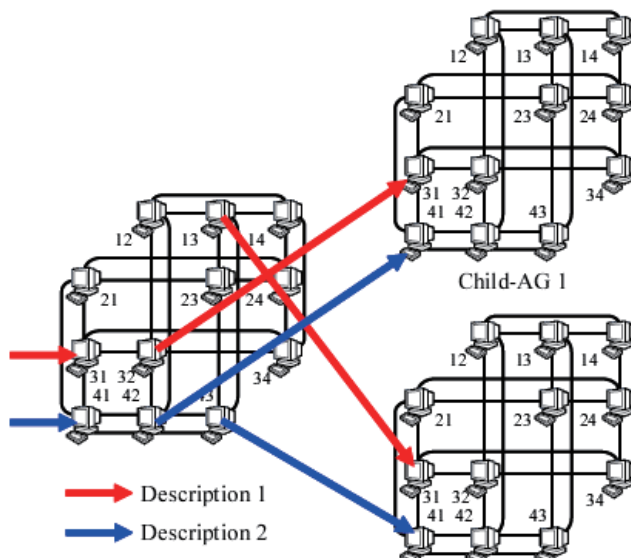


図5: AG の階層化。

3-2 THAG の問題点

THAG は、AG を基にNode-disjoint な多重な配信木を作成可能である。このツリー構造を並列して用いて、Description はそれぞれの配信木を用いて配信される。そのため、複数のDescription が1つのリンクを占有する。現在のインターネットに接続しているノードは上りと下りの回線において接続速度が異なっており、特に有線ノードでは上り帯域の方が下り帯域に比べ狭くなっているのが一般的である。ALMにおいて低遅延でストリームを配信するためには、1つのノードが多数の子ノードを持った方が、ツリーの高さを抑制でき、配信遅延を抑えることが可能である。そのため、ALMでは上り帯域が不足するといった問題点が生じる。THAGも例外ではなく、AGの大きさ s とDescriptionの配信レート r から、最低限の配信に必要なストリームの占有上り帯域が決まる。サイズ s のAGにおいて、AGのルートノードになると最大で $2(s-2)$ の子を持つ。また、どのノードもAGルートになる可能性があるため、サイズ s のAGに参加するためには、 $2(s-2)r$ の上り帯域を持たなければならない。例えば、AGのサイズ $s=8$ 、ストリームのレートが500 kbpsの場合、必要な上り帯域は6 Mbpsである。そのため、この上り必要な帯域よりも小さいリンクに接続しているノードは、配信木に参加しても全てのDescriptionを配信できない。従来のTHAGでは、この必要な帯域に関する不都合は考慮されておらず、全てのDescriptionを配信することを前提とし、全てのAGのサイズが一定としている。そのため、分割されたDescriptionは、全てのリンクで同じだけ帯域を占有する。しかしながら、実際のネットワーク環境では、前章でも述べたように、ユーザの接続するリンクの帯域幅は異なっていると想定するのが自然であり、上記で指摘した不都合が起こることは必至である。

4 提案手法, Network-aware Hierarchical Arrangement Graph (NHAG)

4-1 NHAG の概要

THAGでは、全てのDescriptionを配信することが前提であるため、固定のサイズのAGを基にツリーを作成している。そのため、前章で述べたような問題点が存在する。そこで、本研究では実際のネットワーク環境を想定し、全てのノードに全てのDescriptionを不確実に配信するのではなく、ノードの利用可能帯域に応じて、適切な本数のDescriptionを配信する手法を提案する。そのため、提案手法では、Descriptionのレートとリンクの帯域幅が与えられたときに、AG毎に異なるサイズを用い、更にAGに参加しているノードの状況に合わせて動的にサイズを変更することで、リンクの帯域幅に適応可能である。提案手法ではAG毎にサイズが異なるため、Parent-AGとChild-AGでサイズが異なる状況が発生する。そこで、図6のように、Child-AGのサイズが s の場合、Child-AGは $s-2$ 本のDescriptionを配信する。そのため、上位の階層にサイズの大きなAGが、下位の階層にサイズの小さなAGが位置するような階層構造を構築する方が、配信効率が上昇する。このようなAGの階層構造を実現するために、各ノードは利用可能帯域よりRequested Sizeを計算し、その情報を基にノードの参加・離脱処理を改良する。

4-2 Requested Size の計算方法

ALMへの参加を試みるノードは、まず、リンクの上り帯域BWと1本のDescriptionの配信レート r から、全てのDescriptionを転送可能な最大のAGサイズとして、Requested Size s を計算する。提案手法では、この情報を基にノードの参加・離脱処理を行う。サイズ s のAGに参加するために必要な上り帯域は $2(s-2)r$ である。そのため、Requested Size s_{req} は次式を満たす。

$$2(s_{req} - 2)r = BW \quad \dots (4.1)$$

ここで、BWはノードの利用可能上り帯域である。本研究ではBWをユーザが知るために、利用可能帯域を推測できるIGI (Initial Gap Increasing), SLoPS (Self-Loading Periodic Streams), 及びJitterPathなどの手法を用いる。

式(4.1)を s_{req} について解くことにより、Requested Size

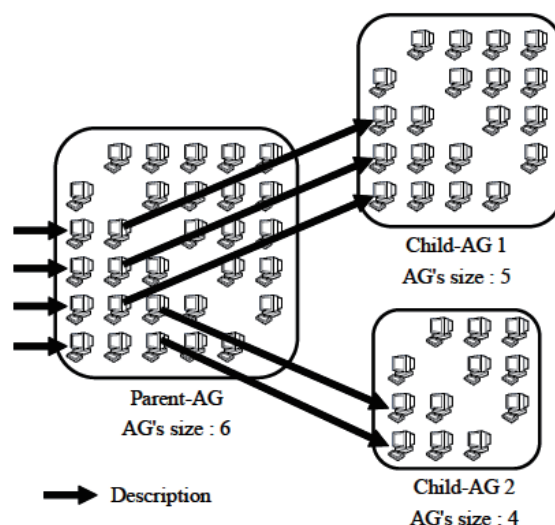


図6: 提案手法におけるAGの階層化。

が次式のように求められる。

$$s_{req} = \left\lceil \frac{BW}{2 \times r} + 2 \right\rceil \dots (4.2)$$

この式を用いて各ノードは、Requested Size の計算を行う。ノードの利用可能帯域はバックグラウンドトラフィックの影響などで、時間とともに変動する可能性がある。そのため、一定時間が経過する毎に各ノードはRequested Size の再計算を行う。もし、利用可能上り帯域が小さく、 $s_{req} = 2$ となってしまった場合、そのノードは1本のDescriptionも受信できない。そのため、 s_{req} が2であるノードのRequested Size は3とする。提案手法では、 s_{req} が2であるノードが存在しても、ノードの上り帯域での輻輳がTHAGよりも軽減できるため、THAGより高い性能が得られることが期待できる。

4-3 ノードの参加手順

提案手法では、参加を試みるノード（新規ノード）のRequested Size s_{join} を基に、参加可能なAGを探索する。参加リクエストを受信したAGのエントランスは、AGのサイズ s_{AG} と s_{join} を比較し、どのように参加させるのかを判断する。

i) $s_{AG} > s_{join}$ である場合のAGの動作

$s_{AG} > s_{join}$ の場合、新たにノードが参加したとしても、全てのDescriptionを転送できない。そこで、新たにChild-AGを作成できる場合、エントランスは大きさ s_{join} のChild-AGを作成して新規ノードに参加させる。一方で、新たにChild-AGを作成できない場合の動作は、既にChild-AGを保持している場合には、エントランスは s_{join} に1番近いサイズのChild-AGを検索し、そのAGに参加するように新規ノードに通知する。その通知を受けた新規ノードは該当するChild-AGに参加リクエストを送る。また、Child-AGを保持していない場合は、AGに参加しているノード数が少数であるため、一時的にノードに参加させる。このノードを“Temporally Joining Node”と呼ぶ。この場合、新規ノードは全てのストリームを転送できない可能性が生ずるが、AGはChild-AGを保持していなく、更に参加しているノード数が少数のため、ストリームが転送できなくなるノードは少数である。

ii) $s_{AG} \leq s_{req}$ である場合のAGの動作

この動作は、THAGの場合とほぼ同じであるが、提案手法では、ノードを置き替える指標として、ノード間の距離を用いずに、Requested Sizeを用いる。THAGでは、既存ノードより新規ノードの方が各AGソースとの距離の和が小さい場合に置き替え処理を行ったが、NHAGでは既存ノードより新規ノードの方がRequested Sizeが大きい場合に置き替え処理を行う。そのため、提案手法では、置き替えの指標として G_{THAG} の代わりに、次式で表される G_{NHAG} を用いる。

$$G_{NHAG}(e) = \frac{RS(e)}{RS(j)} \dots (4.3)$$

ここで、RSはノードのRequested Sizeを示す関数、 e は既存ノード、 j は新規ノードをそれぞれ示す。 $G_{NHAG}(e) < 1$ のとき、既存のノードよりも新規ノードの方が、Requested Sizeが大きい。そのため、全ての既存ノード e に対して G_{NHAG} を計算し、 $G_{THAG} < 1$ となる既存ノードが存在する場合、 G_{NHAG} が最小となる既存ノード e と新しい新規ノード j を置き替える。もし、 G_{NHAG} が最小となるノードが複数存在する場合は、THAGと同様に距離の指標を用いて、置き替えるノードを決定する。これにより、Requested Sizeが大きいノードが上位のAGに参加できる。更に、大きなサイズのAGに一時的に参加しているTemporally Joining Nodeを優先的に置き替えることが可能である。

4-4 ノードの離脱手順

提案手法におけるノードの離脱手順は基本的にTHAGと同じであるが、上位AGでノードが離脱した場合には、そのChild-AGの中で最もRequested Sizeが大きいノードがその代役として供給される。これにより、ノードが離脱しても、Requested Sizeが s_{AG} 以上の十分な帯域を持つノードがChild-AGから供給可能となり、Requested Sizeの大きなノードが上位のAGに参加可能となる。

4-5 AG サイズの更新

ノードの参加や離脱により、AG に参加しているノードが頻繁に入れ替わるため、AGのサイズは、参加しているノードの状況に合わせて動的に更新する必要がある。したがってAG は、ノードの参加や離脱時、ノードの置き替え操作時、及び一定時間経過したときにAG サイズの再計算を行う。ここで、AG サイズの再計算を行う間隔は、各ノードがRequested Size の再計算を行う間隔よりも長く設定する。エントランスはAG サイズを更新するときに、まずAG に参加している N 個のノードのRequested Size の平均 s_{avg} を計算する。

$$s_{avg} = \sum_{i=1}^N s_i \dots (4.4)$$

ここで、 s_i はノード i のRequested Size である。この s_{avg} を基準に、Requested Size の更新をChild-AG を保持していない場合と保持している場合に分けて行う。AG がChild-AG を保持していない場合は、AG サイズ s は基本的に s_{avg} とする。しかし、 s を極端に小さくすると、AG に参加可能なノード数 $s(s-1)$ が減少する。そのため、 s_{avg} のAG が現在のノード数 N に参加できない場合 ($N > s_{avg}(s_{avg}-1)$)、サイズの変更は行わない。

一方、AG がChild-AG を保持している場合は、大きくAG サイズを変更すると、Child-AG へのDescription の配信に支障が出る場合がある。そのため、Child-AG を保持していない場合よりも、 s の増減を緩やかにする。

以上より、提案手法では、ノード毎にRequested Size を計算し、Requested Size を指標にノードの参加、離脱を行う。また、AG のサイズをAG に参加しているノードのRequested Size の平均を基に動的に変更することにより、帯域が不均一な環境においても、利用可能帯域に応じたストリームの配信が可能である。

5 シミュレーションによる性能評価

提案手法の有効性を評価するために、ALM により構築されたツリーによるストリーム配信のシミュレーションをネットワークシミュレータns-2 (Network Simulator version 2.32) [6] を使用して行った。

5-1 シミュレーション環境

それぞれのシミュレーションパラメータの詳細を表1 に示す。バックボーンのネットワークトポロジはTransit-Stub [7] をGT-ITM により生成した。GT-ITM は、Transit-Stub トポロジを生成するツールである。Transit-Stub トポロジは、基幹部分に相当するネットワークであるTransit-Domain と、LAN に相当するネットワークであるStub-Domain からなる (図7)。本シミュレーションでは、1つのTransit-Domain と10のStub-Domain を用いた。シミュレーションでは、Transit-Domain のルータ数を10、各Stub-Domain のルータ数を100とし、合計1010のルータからなるトポロジを生成した。また、輻輳が発生しないように、全てのリンクの帯域を100 Mbps とし、遅延は1 ~ 10 ms の間でランダムに設定した。ノードはStub-Domain にランダムに接続し、ノード数を100 ~ 500の間で50刻みに変化させた。そして、各ノードが2秒毎にALM に順次参加し、全ノード参加後20秒経過した後に、参加順に順次ノードが2秒毎に離脱するようなシナリオを用いた。配信サーバにて、ストリームはMDC により4本のDescription に分割し、各Description の配レートは500 kbps とした。以上のシミュレーション環境で、マルチツリーマルチキャストの従来手法である

SplitStream [8] 及びTHAG [3] と提案手法であるNHAG を比較した。THAG で用いるAG のサイズは6とし、この場合において必要な上り帯域は4 Mbps である。NHAGでは、最大のAG サイズはTHAG と条件を合わせるため、6とした。

この3つの手法を以下の2つのCase において比較する。

i) Case I: 各ノードの上り帯域が4 Mbps で固定THAG において必要な上り帯域は4Mbps ため、第3章で述べたような、帯域が不足するような問題点は起こらない。

ii) Case II: 各ノードの上り帯域が2 ~ 5 Mbps でランダム実際のインターネットを想定した環境であり、THAG において帯域が不足するといった問題が起こり、ノードの上り帯域では輻輳が発生する。

なお、全てのノードの下り帯域は10 Mbps、各リンクの遅延は5 ms とした。

表 1: シミュレーションパラメータ.

Transit-Domain 数	1
Stub-Domain 数	10
ルータ数 / Transit-Domain	10
ルータ数 / Stub-Domain	100
全ルータ数	1010
バックボーンリンク帯域	100 Mbps
ノード数	100~500
ノードの上り帯域	4Mbps(Case I), 2~5Mbps(Case II)
ノードの下り帯域	10Mbps
Description 数	4

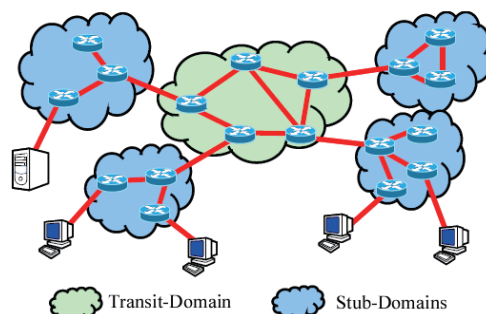


図 7: Transit-Stub トポロジ.

5-2 評価指標

i) Total Throughput

Total Throughput は、受信した各 Description の Throughput の合計であり、次式で定義される.

$$Total\ Throughput = \sum_{i=1}^M Throughput(i)$$

ここで、Throughput(i) は i 番目の Description の Throughput であり、 M は Description の本数である。そのため、本シミュレーション環境では、各 Description の配信レートが 500 Mbps、 $M=4$ であるので、Total Throughput は最大で 2 Mbps となる。

ii) BSR (Bandwidth Satisfaction Rate)

BSR は、各ノードが要求するストリームレートに対し、実際に配信されたストリームレートの割合であり、次式で定義される。

$$BSR = \frac{\sum_{i=1}^M Throughput(i)}{rnd \times r}$$

ここで、 rnd は要求する Description 数であり、 r は Description の配信レートである。 rnd は SplitStream と THAG では 4 となり、提案手法では $s_{req} - 2$ となる。 s_{req} は各ノードの Requested Size である。

iii) RDP (Relative Delay Penalty)

RDP は、各 Description の ALM で配信した場合の遅延とユニキャストで配信した場合の遅延の比の平均であり、次式で定義される。

$$RDP = \frac{1}{M} \sum_{i=1}^M \frac{mDelay(i)}{uDelay}$$

ここで、 $mDelay(i)$ は i 番目の Description の ALM での配信遅延であり、 $uDelay$ はユニキャストでの配信遅延である。RDP は配信遅延が小さいほど 1 に近づき、QoS が高い。

iv) RDV (Relative Delay Variation)

RDV は、各 Description の遅延差を表す指標であり、次式で定義される。

$$RDV = \frac{D_{max} - D_{min}}{D_{min}}$$

ここで、 D_{max} と D_{min} は、各 Description の最大遅延と最小遅延をそれぞれ表している。Description の遅延差が大きいと、ユーザは全ての Description を受信するまで映像再生ができないため、QoS は低下する。RDV は Description の遅延差が小さいほど 0 に近づき、高い QoS を実現している。

i) Total Throughput

図8に、Total Throughputの平均値を示す。Total Throughputの平均値では、THAGにおいて、輻輳が起こらない理想な環境であるCase Iの場合、提案手法とTHAGは2 Mbpsに近い高いTotal Throughputを達成している。これより、THAGとNHAGでは、全てのノードが配信されたストリームのほぼ全てを受信できている。

一方で、Case IIの場合は、THAGでは、上りリンクで輻輳が発生し大幅に平均Total Throughputが低下しているにもかかわらず、提案手法では、利用可能帯域の多様性に対応したSplitStreamと同等のTotal Throughputを維持している。

SplitStreamはCase IとCase IIで、ほぼ変わらない平均Total Throughputを実現している。Case Iにおいて、THAGとNHAGよりもTotal Throughputが低いのは、SplitStreamがNode-disjointなツリー構造を構築していないためである。Case IIにおいて、SplitStreamのTotal Throughputが落ちないのは、SplitStreamは帯域が不均一な環境に適応しているためである。SplitStreamでは、保持している子ノード数に余裕のあるノードでSpare Capacity Groupというグループを作っている。もし、親となるべきノードに空き帯域がない場合、このグループより適切な親ノードを検索する。そのため、SplitStreamは帯域が不均一な環境においても、輻輳が起きないツリーの構築が可能である。しかし、Total Throughputの最小値は、NHAGよりも両Caseにおいて大幅に低く、そのため、各ノードが受信できるストリームにばらつきがある。

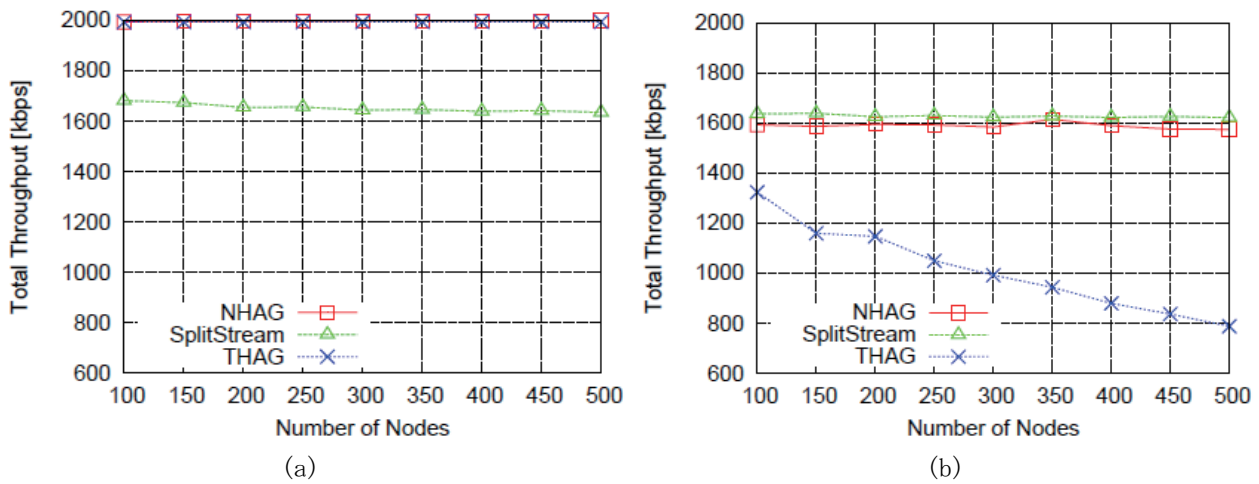


図8: Total Throughputの平均値 (a) Case I, (b) Case II.

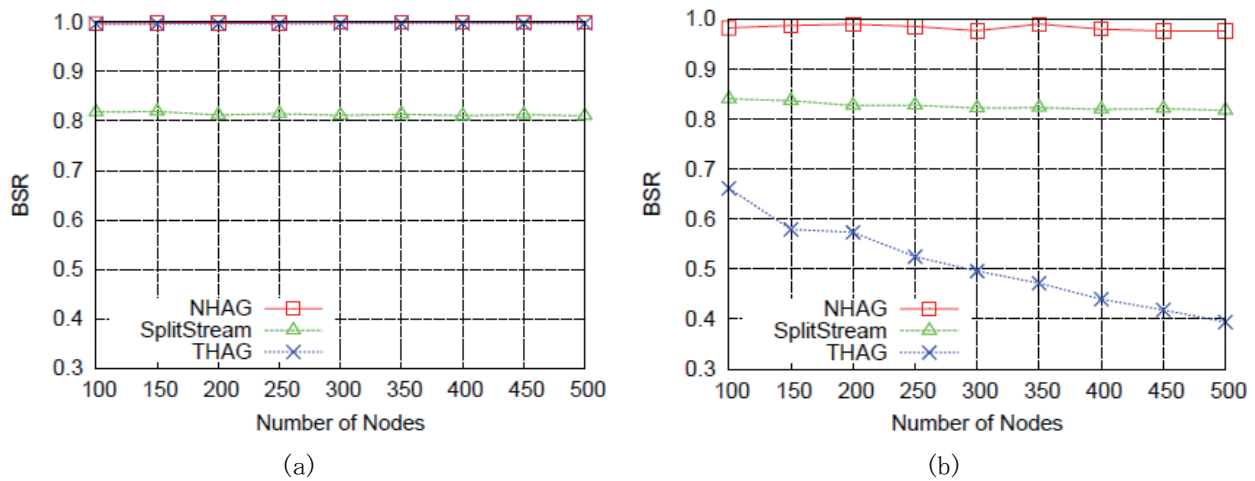


図9: BSRの平均値 (a) Case I, (b) Case II.

ii) BSR (Bandwidth Satisfaction Rate)

各手法におけるBSRの平均値を図9に示す。Case Iでは、提案手法、THAG共にBSRの平均値はほぼ1であり、各ノードが希望したストリームが全て配信されている。Case IIでは、THAGは要求するストリームレートよりも受信レートが減少するため、BSRはノード数が増加するに従って小さくなっていく。特に、ノード数が500の場合には、平均で要求しているストリームの4割程度しか受信できない。一方で、提案手法のBSRは理想的な場合に近く、ノード数によらず、平均値がほぼ1である。SplitStreamはどちらの場合でも、全てのDescriptionを受信していないため、平均値は1より小さく、0.8程度である。

この結果より、提案手法はノードの利用可能な上り帯域に合った適切な本数のDescriptionの配信を実現していることがわかる。

iii) RDP (Relative Delay Penalty)

RDPの結果の平均値を図10に示す。Case Iでは、提案手法、THAG共に平均で4程度の低いRDPを実現している。Case IIでは、THAGは輻輳によるキューイング遅延が増加するため、配信遅延が増加する。そのため、RDPはCase Iに比べると高くなっており、平均で8程度、最大で30程度に増加している。それに対し、NHAGでは、AGサイズを動的に変化させ、輻輳を回避しているため、RDPの平均値はCase Iとほぼ変わらず、最大値でも多少は増加するものの15程度となっている。一方で、SplitStreamは、Case I, Case II どちらの場合においても、ノード数の増加により、RDPが大幅に増加している。これより、SplitStreamの配信遅延はとて大きいことがわかる。

この結果より、提案手法はノードの利用可能帯域が不均一な状況においても、ノード数によらない低い配信遅延を実現していることがわかる。

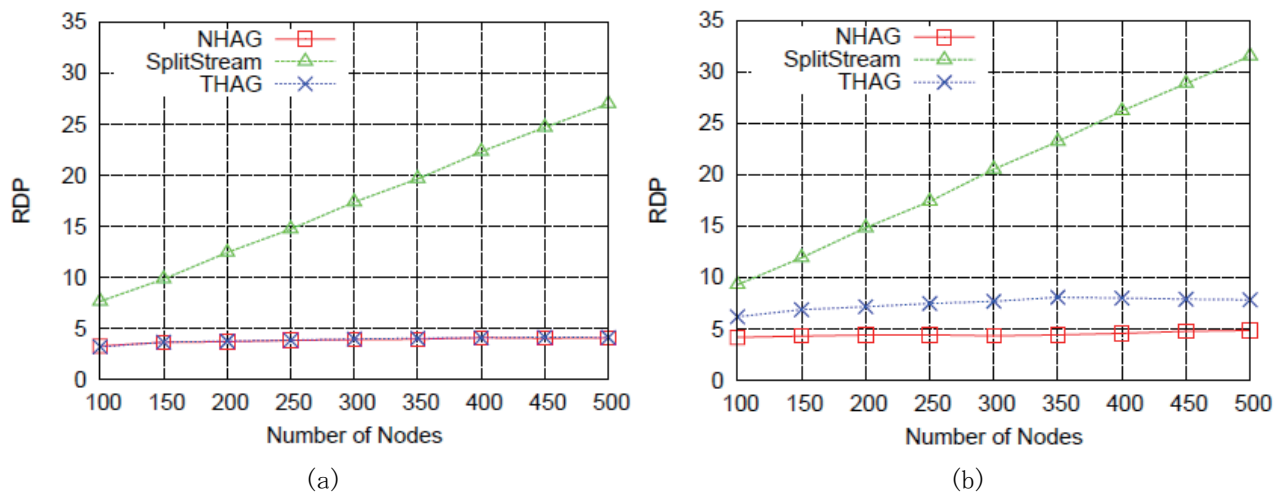


図 10: RDP の平均値 (a) Case I, (b) Case II.

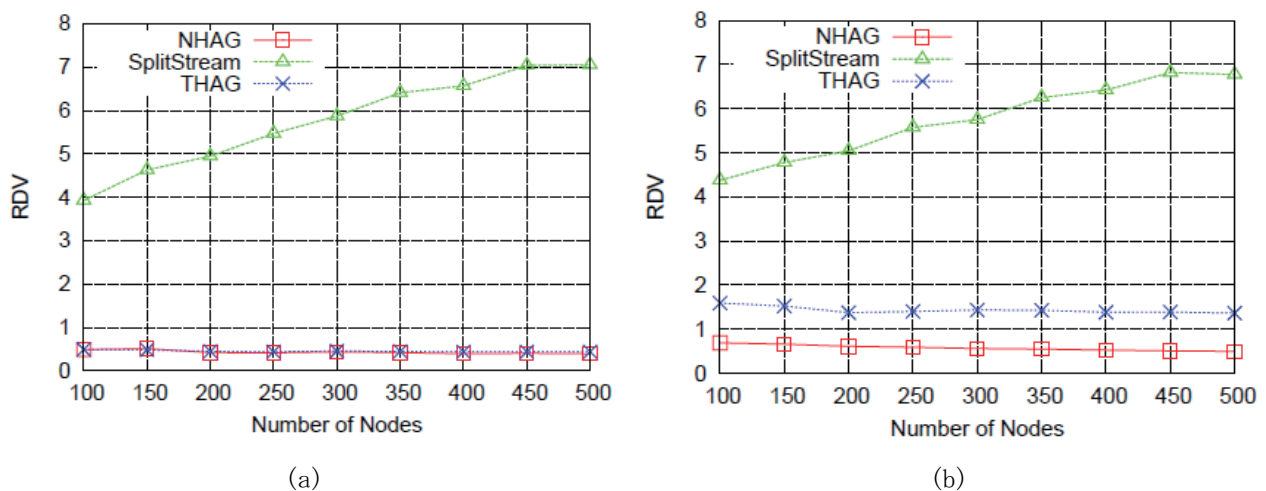


図 11: RDV の平均値 (a) Case I, (b) Case II.

iv) RDV (Relative Delay Variation)

RDV の結果の平均値を図11 に示す. Case I では, 提案手法, THAG 共に平均で0.5 程度の低いRDV を実現している. そのため, 提案手法, THAG はDescription 間の遅延差は小さい. Case II では, THAG の各Description の配信遅延は, 輻輳により不均一になるため, RDV は平均値で1.5 程度に上昇している. 一方で, NHAG では, RDP と同様にCase I と同等の低いRDV を実現しており, 平均値で0.8 程度である. SplitStream のRDV は両Case において, 提案手法, THAG に比べると圧倒的に大きい. このため, Description 間の遅延差は大きく, QoS は低い. この結果より, 提案手法はDescription の遅延差が少なく, 高いQoS を実現していることがわかる.

以上の結果より, THAG は, Case I では, 高いTotal Throughput とQoS を実現しているが, Case II では, 輻輳により受信できるストリームが減少するため, 性能が大幅に低下している. また, SplitStream は, ノードの利用可能帯域の多様性に適応しているが, 各Description の配信遅延と遅延差が大きいため, QoS は低くなっている. そして, 提案手法は, Case I においては, THAG と同等の性能を発揮し, Case II においても, SplitStream と同等のTotal Throughput とCase I のTHAG と同等の低いRDP, RDV を実現している. これは, NHAG では, 上り帯域に合わせて参加させるAG を適切に選択し, 理想的な場合とほぼ同程度のストリームを受信できているためである. すなわち, 提案手法は, 利用可能な帯域に応じてストリームの配信が可能であり, 従来手法よりも帯域に適応したストリームの配信が実現できたといえる.

6 結論

本研究では, ノード離脱に対する影響を最小限に抑え, ネットワーク環境の多様性に適応したALM を実現するために, THAG という手法に注目した. THAG は, AG を階層化することによりNode-disjoint な多重の配信木を構築可能であり, ノードの離脱にロバストな手法である. しかし, THAG では, 全てのAG のサイズが同一なため, 全てのノードが全ての配信ツリーに参加する. そのため, 各ノードの利用可能帯域の多様性に対応することは困難であった. 本研究ではTHAG をユーザの利用可能帯域に適応させた手法を提案した. 本提案では, THAG のように不確実に全てのストリームを配信するのではなく, ユーザの帯域に合わせた適切な本数のストリームを配信する. 提案手法では, このようなストリームの配信を実現するために, 各AG のサイズを参加しているノードの利用可能帯域を考慮し動的に変化させた. また, 各ノードで計算したRequested Size を指標に, ノードが利用可能帯域にあった適切なAG に参加できるように, ノードの参加・離脱アルゴリズムを改良した.

本研究の成果は, ALM におけるマルチメディアコンテンツのストリーミング配信の品質向上に寄与するものであると考える.

【参考文献】

- [1] M. Alasti, K. Sayrafian-Pour, A. phremides, and N. Farvardin, "Multiple description coding in networks with congestion problem," *IEEE Trans. Inf. Theory*, vol.47, no.3, pp.891–902, Mar. 2001.
- [2] V.K. Goyal, "Multiple Description Coding: Compression meets the network," *IEEE Signal Process. Mag.*, vol.18, no.5, pp.74–93, Sep. 2001.
- [3] R. Tian, Q. Zhang, Z. Xiang, Y. Xiong, X. Li, and W. Zhu, "Robust and efficient path diversity in application-layer multicast for video streaming," *IEEE Trans. Circuits Syst. Video Technol.*, vol.15, no.8, pp.961–972, Aug. 2005.
- [4] K. Day, and A. Tripathi, "Characterization of node disjoint paths in arrangement graphs," Technical Report TR91–43, Computer Science Dept., Univ. Minnesota, 1991.
- [5] Y.S. Chen, T.Y. Juang, and E.H. Tseng, "Congestion-free embedding of $2(n-k)$ spanning trees in an arrangement graph," *J. Syst. Archit.*, vol.47, no.1, pp.73–86, Jan. 2001.
- [6] "The Network Simulator – ns-2," <http://www.isi.edu/nsnam/ns/>.
- [7] E.W. Zegura, K.L. Calvert, and S. Bhattacharjee, "How to model an Internetwork," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, pp.594–602, Mar. 1996.
- [8] M. Castro, P. Druschel, A.M. Kermarrec, A. Nandi, A. Rowstron, and A. Singh, "SplitStream: High-bandwidth content distribution in cooperative environments," in *Proc. Int. Workshop on Peer-to-Peer Syst. (IPTPS)*, pp.298–313, Oct. 2003.
- [9] V.N. Padmanabhan, H.J. Wang, P.A. Chou, and K. Sripanidkulchai, "Distributing streaming media content using cooperative networking," in *Proc. ACM Int. Workshop on Network and Operating Syst. Support for Digital Audio & Video (NOSSDAV)*, pp.177–186, May 2002.
- [10] V.N. Padmanabhan, H.J. Wang, and P.A. Chou, "Resilient peer-to-peer streaming," in *Proc. IEEE Int. Conf. Network Protocol (ICNP)*, pp.16–27, Nov. 2003.

〈発表資料〉

題名	掲載誌・学会名等	発表年月
Reliable Application Layer Multicast over Combined Wired and Wireless Networks	IEEE Transactions on Multimedia	2009年12月