

P2P ストリーミング放送システムの改良

2007MI122 牧 祐希
指導教員

2007MI151 長江 翔
後藤 邦夫

1 はじめに

近年、コンピュータ性能やネットワーク回線の向上に伴い、デジタルコンテンツ技術が注目されている。その中でもストリーミング形式によるコンテンツの配信が増加している。今までのストリーミング配信ではクライアントサーバ配信を利用するため、ユーザからの大量アクセスによる負荷に耐えることができる大規模サーバや広帯域回線が必要であり、多大なコストがかかった。

現在では P2P 技術の発達により、PeerCast や ShareCast など誰でも配信者(ストリーマ)となる P2P ストリーミングが実現された。P2P ストリーミングでは、各ノードがストリームデータを受信すると同時にデータを複製し、下流のノードにデータを受け渡す方法を用いる。そのためサーバとなるノードの負担を減らすことができる。

しかし、P2P ストリーミングではストリームデータを中継しているノードが接続を中断した場合にその下流のノードはデータを受け取れなくなってしまう。また他ノードへの再接続までに時間が必要となり、データが一時的に再生できない問題がある。

本研究は、過去の卒論(発行 2009 年)[2]の引き継ぎであり、システムの改良を目的としている。[2]では、再接続までの時間短縮のためにルーティング方式 [1]を改良しているが、プログラム実行中にエラーが生じる。またノードが接続を中断した場合に再接続し、ストリーミングを再開することを目的としているがプログラムが未実装である。

そこで本研究では、過去プログラムを参考に、問題点を解消したプログラムを再実装する。安定したプログラム動作を確保するため、新しいルーティング方式を採用し、ノードの離脱に対する処理を完成する。またネットワークエミュレータである Goto's IP Network Emulator(以下 GINE)[3]を使って仮想ネットワークを模倣し、複数のノードでプログラムを動作し、改良後のシステムを評価する。

2 システムの概要

旧システムには動作上の問題と仕様上の問題がある。本研究ではこれらの問題を解消したシステムを実装する。この節では旧システムの問題点と新システムのルーティング方式について説明する。

2.1 旧システムのルーティング方式

旧システムのルーティング方式はノード離脱時の信頼性を考慮したルーティング方式であり、接続が切れた場合の再接続までの時間短縮を目的とする。データを配送する主ノードと、データを配送しないが主ノードの接続

が切れた場合に主ノードへと切り替わる副ノードをあらかじめ繋いでおく方式である。この方式により下流ノードがデータを受け取れない時間を軽減することができる。また各ノード間で定期的にノードの接続状況を確認し、主ノードの接続が確認できない場合は副ノードを主ノードに切替えて、新たに副ノードを決定する。図 1 は接続が切れた場合のシステムの流れを示す。

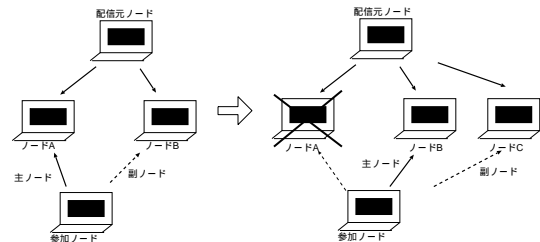


図 1 主ノードと副ノードの切替え

● 問題点

- 実装が不完全であり、動作が不安定である。
- ストリーミングに参加していた番組から脱退する時データを中継されている下流ノードが考慮されていない
- システム全体のノード間の関係が適切に更新されない

2.2 改良点

新システムは旧システムの構成を参考に実装する。旧システムから新システムへ変更した箇所を次に述べる。

・マップの採用

新システムでは C++ 標準テンプレートライブラリであるマップを採用する。マップとは、要素をキーと値の組合せによって管理する連想コンテナであり、要素の挿入や削除を容易に処理することができる。旧システムでは、番組情報や経路情報の追加や削除を双方向リストにより操作している。これをマップに変更することで複雑なリスト操作をする必要がなくなり、リスト操作によるバグの発生を抑える。また隣接ノードの情報を配列で管理していたのを同様にマップで管理する。これにより旧システムより詳しい情報を管理することが可能となる。

・ルーティング方式の変更

複雑なルーティング方法から変更することで、ルーティング部分を除くシステムの安定性を確認する。また新しいルーティング方式により、ノード間は適切な隣接関係を築く。

・仕様の追加

旧システムには 2 つの仕様を追加し、システムの安定

性を高める。

1. ストリーマの設定削除

ストリーマがデータ配信を中止する際、旧システムではシステム全体に処理が行き届かない状態になっている。これにより存在しない番組への参加がシステム動作を不安定にしていた。新システムではストリーマによる配信中止をシステム全体に送信し、処理をする。

2. 中継ノードによる番組からの脱退

旧システムには下流ノードを考慮した処理がされていない。番組の脱退処理を追加し、下流ノードのストリーミングデータを確保する。

2.3 新システムのルーティング方式

本研究では、システムの安定を確認するため旧システムより簡略化したルーティング方式を採用する。ルーティング方式は後に別のルーティング方式が採用できる構成になっている。ここでは現在採用しているルーティング方式について説明する。ルーティング実装のために考慮する点は以下の3つである。

- 1つのノードに対する下流ノードの数を制限し、利用可能帯域を超えないようにする
- ストリーミングデータの中継回数を減らしデータの品質の劣化を防ぐ
- IPネットワークにおけるホップ数を減らしネットワークへのトラフィックを軽減する

1つのノードに対する下流ノードの数はストリーマの設定によって決定する。ストリーマはデータの配送を設定する際、データ送信に必要なバンド幅を入力する。この情報からその番組を中継するノードが1度に配送できるノードの数(以下、配送人数)を決定する。1つのノードがデータを配送するのに使用できる最大帯域は10Mbpsし、配送できる最大人数は10人とする。図2はストリーマが5Mbpsのデータを配送し、1つのノードの配送人数が2の場合のルーティングの流れ図である。

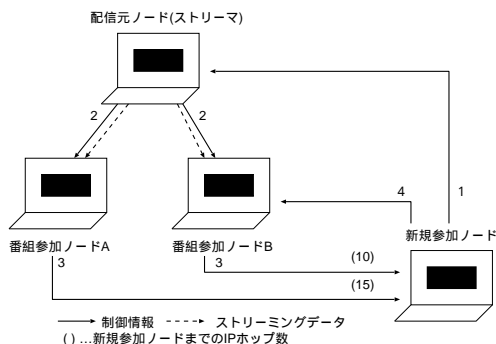


図2 ルーティングの流れ

1. 新規参加ノードは既知ノードから番組表を受信し、その番組表から視聴したい番組を配信してい

るストリーマに番組への参加要請を送信する。

2. 参加要請を受信したストリーマは、配送人数を満たしていないノードの深さまで要請を中継する。
3. 参加要請を受信したノードAとノードBは、配送人数を満たしていないので、新規参加ノードにストリーミングデータを中継可能であることを送信する。
4. 新規参加ノードは中継可能の情報を受信した際のttlからIPホップ数の少ないノードに配信を要請し、ストリーミングデータを中継してもらう。

2.4 ノードの離脱に対する処理

新システムではシステムの安定のために、ノード離脱に対する処理をする。ノード離脱とは2つの場合があり、次に説明する。

- ノードが番組から脱退する場合

ストリーマはストリーミングをしている木の深さとストリーミングに参加しているノードの情報を持っている。番組から脱退を希望しているノードの木の深さからそのノードが木の先端にあるノードでなかった場合、直前に番組に参加したノードをその離脱箇所に移動させ、下流ノードを引き継ぐ処理をする。これにより下流ノードがストリーミングデータを受信できない状態を防ぐ。

- ノードの接続が確認できない場合

各ノードは一定時間ごとにノードの生存を確認する。ノードの接続が確認できない場合、周辺のノードの情報を取得し、接続が確認できないノードの状態を調べる。状態に別で適した処理をし、システムの安定性を高める。接続が確認できないノードの状態別の処理を次に述べる。

1. ストリーマの場合

システム全体にストリーマが離脱し番組を中止することを伝える。システム内の番組表から番組を削除し、その番組を視聴していたノードの視聴状態を中止する。

2. 番組を中継していた場合

前に述べた番組からの脱退と同様の処理をする。

3. 上記以外の場合

ノードの離脱を確認したノードがそのノードの周辺ノードを引き継ぐ。

3 システムの実現

この節ではシステムの実現の詳細について述べる。提案するシステムは3つのプログラムによって構成される。

- Streamer : UDPでストリームを送信するプログラム
- P2Pnode : 送られてきたデータを制御するプログラム
- Listener : データを受け取るプログラム

Streamer と Listener は動画や音声を送受信する外部プログラムである。

3.1 P2Pnode について

P2Pnode の概要について説明する。P2Pnode は次のクラスで構成される。

- 中継スレッド:データの送受信をする
- 制御スレッド:制御情報の送受信をする
- 生存確認スレッド:一定時間ごとに隣接したノードの接続を確認する
- 経路表:制御スレッドによって決定された経路情報を操作する
- 番組表:システム内の番組情報を操作する

図 3 は P2P ノードの簡略図である。

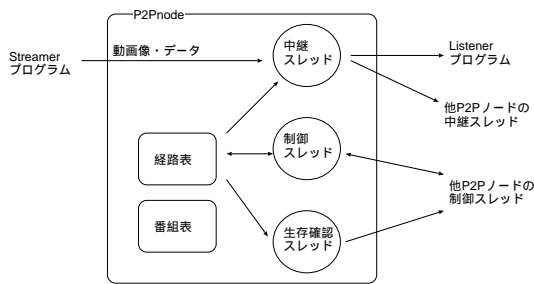


図 3 P2P ノード

このシステムはソケットを 4 つ使用し、2 つのポートを bind する。ソケットは中継スレッドの送受信用、制御スレッドの送信用と受信用、生存確認スレッドの送受信用で使用される。P2P プログラム用のポートをプログラム起動時に指定し、そのポート番号が制御スレッドの受信用ソケットに割り当てられ、そのポート番号 +1 が中継スレッドのソケットに割り当てられる。2 つソケットを持つことで、ストリーミング用と制御用の通信を確保することが可能となる。

また制御スレッド用のデータ構造と中継スレッド用のデータ構造をもつクラスを用意し、P2P ストリーミングシステムに適した通信を可能とした。

- 制御スレッド用データ構造
データ構造には処理してほしい制御情報とパスワードが含まれている。パスワードが一致しないノードからの要求は受け付けられないものとする。パスワードはプログラム起動時に指定する必要がある。
- 中継スレッド用データ構造
このデータ構造はパケットに似せた構成になっている。中継スレッドはストリーマから送られてきたストリーミングデータに送信元と宛先のアドレスをつける。宛先アドレスにはそのストリーミングデータのマルチキャストアドレスが入っている。このデータ構造を受け取ったノードは宛先ア

ドレスから経路情報を検索し、配送先にデータを中継する仕組みになっている。

3.2 ルーティングの概要

この節ではルーティングの詳細を説明する。ルーティングを実装するために使用する変数を次に述べる。またこの変数は経路情報のクラス内に存在する。

- stream: ノードが配送可能な人数を表す
- limit: ストリーマ専用の変数であり、木の深さに対する配送可能な残りのノード数を表す
- treesize: 木の深さを表す。ストリーマはストリーミングツリーのサイズを表し、番組を中継しているノードはそのノードの位置を表す。

図 4 は配送人数 2 の時のストリーミングツリーの図である。

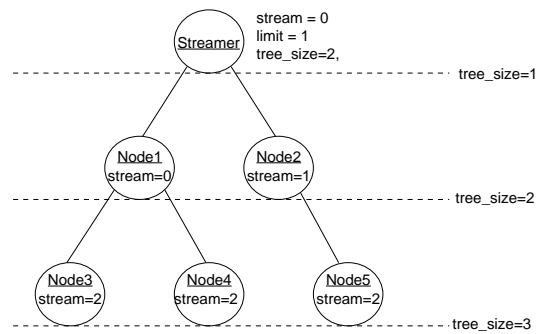


図 4 ツリー実装図

limit はストリーマの配送人数と木の深さから決定する。配送人数と木の深さの関係は表 1 に示す。

表 1 木の深さに対するノード数

	配送人数 2	配送人数 3	配送人数 m
treesize1	2	3	m
treesize2	4	9	m^2
treesizen	2^n	3^n	m^n

図 4 は treesize が 2 なので中継ノードとなる候補ノードは Node1 と Node2 である。また 2 つのノードからデータを中継してもらえるノードの数 (limit) は 4 ノードである。limit はノードの参加が決定する度に 1 減っていき、Node5 の配信が決定した時点で図のような値となる。新規参加ノードは、候補ノードの中で配送人数を満たしていないノードの中から IP ホップの少ないノードを中継ノードに決定する。limit が 0 になると treesize は 1 増え、新しい limit が計算される。この仕組みにより深さが均等な木を構成する。

4 実験と評価

我々が提案するシステムは、100 名程度が P2P ストリーミング放送を利用して、500kbps 程度の低レートビ

デオや音楽データを送受信する小規模システムと想定している。我々は仮想ネットワークを模倣し実験する。またノードの離脱に対する処理の評価は実機を用いて実験する。

4.1 GINE を用いた実験

・ネットワークモデル

ノード数 100 名のネットワークを GINE を利用し模倣する。OS は VineLinux4.0, カーネルは 2.6.16 を使用する。GINE とは、多数のルータやリンクで構成される広域ネットワークを模倣することができるネットワークエミュレータである。ネットワークスタックのみを仮想化する機能を搭載しているため、P2P プログラムを仮想ネットワーク上で容易に動作させることができる。GINE 上での P2P プログラムの実行は expect スクリプトを用いる。expect スクリプトとは定型的な作業を自動実行できるプログラムである。図 5 はネットワークモデルであり、NS1 から NS100 が P2P ノードである。

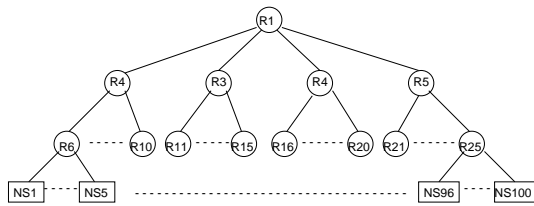


図 5 仮想ネットワーク

・実験と結果

実験ではストリーミングツリーを構築し、その先端ノードでストリーミングデータが中継されているか確認する。リスナー数は 90 ノードとし、音声データは 320kbps 程度の MP3, 1.4Mbps 程度のオーディオ用 CD を使用する。

音楽データの再生を確認できた最大通信速度と最大中継回数を表 2 に示す。

また、音楽データの再生が成功した木の関係を表 3 に示す。

表 2 ストリーミングの中継結果

最大通信速度	1.4Mbps
最大中継回数	90node

表 3 ストリーミングツリーの構成

配送人数	2	3	5	10
木の深さ	7	5	4	3

4.2 実機を用いた実験

ノード離脱時のストリーミングの安定性を確認する。この実験はローカルエリアネットワーク内での実験であり、音楽データは 1.4Mbps, ノード数は 5 ノードである。上流ノードは番組から離脱する場合、2.4 節で述

べた方法で新しい上流ノードへと切り替わる。下流ノードで tcpdump コマンドを実行し、ストリーミングデータを受信する間隔を計測する。また離脱ノードから受信した最後のデータと新しい上流ノードから受信した最初のデータとの時間間隔から離脱時の処理時間を計測する。各値は 5 回計測し、平均値の結果を表 4 に示す。tcpdump とはネットワーク上のパケットをモニタリングするツールである。

表 4 平均間隔

受信間隔	処理時間	処理後の受信間隔
7.941msec	1007.51msec	7.968msec

この実験で音楽データが中断されることなく再生できることを確認した。また結果から離脱処理による影響は少ないといえる。

5 おわりに

本研究では、プログラムの安定性を目的とした P2P プログラムを再実装し、その性能を評価した。P2P ノード間の接続と基本操作は完成し、1.4Mbps 程度の音楽データのストリーミングの動作が確認されている。またノード間の離脱に対する処理を追加し、下流ノードがストリーミングデータを受け取れない問題を解消した。しかし、番組から脱退するノードの位置によっては音楽データに雑音が入り、音質の劣化が発生した。またノードの再接続は 5 ノードを手動で操作した場合成功したが、GINE を用いた 100 ノードでの自動実行では通信状況にエラーが生じた。今後の課題として以下の 4 点が挙げられる。

1. 新たなルーティングの提案
2. 動画像のストリーミング実験
3. ストリーミングデータの品質確保
4. 再接続処理の改良

より詳しい実験とシステムの向上が必要だと考える。

参考文献

- [1] 後藤祐輝, 田中達也: P2P ストリーミングのためのルーティング提案とそのエミュレーション, 卒業論文, 南山大学 数理情報学部 情報通信学科 (2008).
- [2] 柴田美沙登, 渡邊藍子: P2P ストリーム放送のためのルーティング実装とその性能評価, 卒業論文, 南山大学 数理情報学部 情報通信学科 (2009).
- [3] Sugiyama, Y. and Goto, K.: Design and Implementation of a Network Emulator using Virtual Network Stack, *Proc. of the Seventh International Symposium on Operations Research and Its Applications (ISORA2008), Lecture Notes in Operations Research, Vol.8*, pp. pp.351-358 (2008).