

マルチメディア情報伝送中のサーバリダイレクション に関する一考察

西村 弘毅 山本 潮 小野里 好邦

群馬大学大学院工学研究科

〒 376-8515 群馬県桐生市天神町 1-5-1

あらまし インターネット上での情報伝送量の増加やインターネット中継などのリアルタイム情報伝送による長時間伝送から、サーバへの負荷の集中が問題になっている。サーバへの負荷を分散させる方法としてミラーサーバの設置や、それらに自動的にリダイレクションするための方式が研究・開発されてきているが、これまで提案されている方式においてはリダイレクションは最初にしか行なわれない。しかし、ネットワークの状況は常に変化しているため、最初に選択したサーバが適切であり続ける保証はない。そこで、本稿では情報伝送中にサーバリダイレクションを行ない、ネットワーク状況の変化に対応した動的なサーバ選択を行なう方式について検討する。また、インターネット上で実験を行ない、本方式の有効性を検討する。

キーワード リダイレクション, 負荷分散, ミラーリング, マルチメディア

A Study on Server Redirection Method under Transmitting Multimedia Information

Hiroki Nishimura, Ushio Yamamoto, and Yoshikuni Onozato

Graduate School of Engineering, Gunma University

Abstract Some load balancing techniques such as mirroring and automatic redirection have been developed to disperse the load of information server and network. In the case of long-term transmission by large-scale information or realtime Internet broadcasting, however, these techniques may not work effectively. One of the reasons is that the server to send information is selected at only the first access of the client. In this paper, we propose a new redirection method in which the servers are changed/selected during information transmission, according to the status of the network and the server. We have some experiments based on our approach and show the results of the effectiveness of our method.

keywords Redirection, Load Balancing, Mirroring, Multimedia

1 はじめに

近年のインターネットの普及は目覚しく、そのつながりは全世界規模のネットワークを形成し、現在も成長を続けている。これには、操作性や利便性の向上と、それに伴う一般ユーザへの浸透が理由として挙げられる。また、回線の高速化から扱うデータ量も大きくなっており、音声、画像、動画等のマルチメディアコンテンツもネットワーク上でやりとりされるようになりつつある。しかし、インターネット人口の増大や扱うコンテンツサイズの増大はサーバ回線へ過大な負荷を与えることになり、クライアントアクセス時間の長時間化などの問題を引き起こす原因となっている。

アクセスの長時間化や、サーバへの負荷の集中という問題に対する一つの解決策として、ミラーサーバの設置が現在主に行なわれている。同一コンテンツを複数のサーバに置き、ネットワーク上の異なる場所に設置することで、クライアントのアクセスを分散させることを目的としている。クライアントのアクセスが分散されれば、サーバへの負荷も分散されることになり、アクセス時間の長時間化を未然に防ぐことが期待されている。

しかし、クライアントがネットワークの状況を推測してミラーサーバの選択を行なう場合、どのサーバが適切であるのかをクライアントが判断することは難しい。また、DNSのラウンドロビン機能を利用してシステム側でサーバの自動選択を行なう方法 [1] も提案されているが、ネットワークの状況を考慮していないため、選択されたミラーサーバが適切である保証はない。このように、ミラーサーバ設置により負荷の分散が効果的になされているとは必ずしもいえない。また、ストリーミング配信においても RealNetworks 社が提供している RealServer [2] のスプリッティング技術による、一種のミラーサーバの設置が行なわれている。クライアントが自分により近いサーバを選択することで高品質なコンテンツを提供する試みであるが、適切なサーバ選択という面では同様の問題が発生する可能性がある。

適切なミラーサーバの選択については、ラウンドトリップタイムを観測し、ネットワークの状況を推測してサーバの選択を行なう TASS 方式 [3] や、DNSのラウンドロビン機能を利用した RR-TASS [4] 方式が提案されている。これらはネットワーク状況に対応した、より適切なサーバを選択するための試みである。

これらの方式においては一度サーバが決定されると、以降はそのサーバのみからの情報伝送となる。しかし、マルチメディアコンテンツを扱うことが求

められてきている現在、クライアントのアクセス時間はさらに長時間化する傾向にあり、情報の伝送中にネットワークの状況が変化する可能性は大きい。インターネット中継などを行なう場合はなおさらである。よって、最初に選択したサーバがある時間経過した後も適切であるとは限らない。

そこで、本稿では情報伝送中のネットワーク状況変化への対応という問題に対し、情報伝送中のネットワーク状況を観測し、現在のクライアント-サーバ間の回線が悪化した場合に、回線状況の良い他のサーバへのリダイレクションを行なう方法を検討する。また、情報の伝送を完全に切替えるのではなく、複数のサーバに対して送信データを分割して割り当て、クライアント側で並行してデータを受信することで、データ伝送の効率化を図る。この方式によるクライアントアクセス時間の短縮と、サーバへの負荷の分散に対する効果を検討する。

2 関連研究

ここでは、ネットワーク状況に適応した適切なサーバ選択を行なう方式として提案されている TASS 方式と、リアルタイム情報伝送においてミラーリングの技術が用いられている RealNetworks 社の RealServer について紹介する。

2.1 TASS 方式

TASS (Traffic Adaptive Server Selection) 方式は、ICMP を使用したパケットによるラウンドトリップタイムの観測でネットワーク状況を推測し、適切なサーバの選択を行なう方式である。まず、アクセスを受けたプライマリサーバは、Web サーバ/クライアント間、Web サーバ/キャッシングサーバ間のラウンドトリップタイムを観測する。ラウンドトリップタイムを評価した結果、プライマリサーバが伝送するのに適切でないと推測した場合、プライマリサーバは各ミラーサーバに対してミラーサーバ/キャッシングサーバ間のラウンドトリップタイムの観測要求を行なう。各ミラーサーバはラウンドトリップタイムを観測し、その結果をプライマリサーバに返す。プライマリサーバは最小値のラウンドトリップを返したミラーサーバに転送するようクライアントにメッセージを送り、そのサーバがクライアントに Web ページデータを伝送する。この方式により、ネットワーク状況に対応した適切なサーバの選択が行なわれる。

2.2 RealServer

RealNetworks 社が提供している RealServer とは、収録済みのメディアやライブメディアイベントをネットワーク上にストリーミング配信するソフトウェアである。クライアントはメディアをリアルタイムで受信することができ、ダウンロード完了を待つ必要はない。

RealServer においてスプリッティングという技術が用いられている。スプリッティングとは、情報を発信するソースサーバがクライアントではなく、他の RealServer (スプリッタと呼ばれる) にメディアを配信する。このスプリッタはミラーサーバのような働きを持ち、ソースサーバからメディアを受信したスプリッタがクライアントに対し、メディアを再度配信する。情報発信源であるソースサーバから遠く離れたクライアントも、自分の近くに設置されているスプリッタからメディアを受信することで、高品質のコンテンツを受けとることができ、帯域幅の利用も押えられる。

この RealServer のように、リアルタイム性が要求されるメディアの配信においても、ミラーリング技術を用いることでクライアントのアクセスの分散とそれによる伝送品質保持の試みがなされている。

2.3 問題点

最初に適切なサーバを行なった場合でも、依然として次のような問題が残っている。マルチメディア情報等大容量のコンテンツを扱う場合、クライアントアクセス時間は長時間に及ぶ。アクセス時間が長くなればそれだけネットワークの状況が変化する可能性が高いため、最初に選択した適切なサーバが、ある時間経過後も適切であるとは限らない。サーバに要求を送信した時点で回線速度が良好であったサーバが、ある時間経過後にクライアントからのアクセスが増化した場合、回線速度は当然遅くなる。しかし、これまでの方式では一度要求を送信するサーバを決定したら、それ以降回線速度が悪化した場合でもサーバの切替えは行なわれていない。つまり、情報伝送中のネットワーク状況変化への対応が考慮されていないということである。

3 提案方式

3.1 概要

前節で述べた情報伝送中のネットワーク状況変化への対応という問題に対し、情報を伝送中に現在の

サーバとの回線状況が悪化した場合に、他のサーバへのリダイレクションを行なう方法を提案する。ここで、クライアントが最初にアクセスするサーバをプライマリサーバ、リダイレクションを行なうサーバをミラーサーバとする。プライマリサーバから情報の受信を開始し、ある時間間隔毎に回線速度を計算する。この回線速度をネットワークが混雑していると判断する閾値と比較し、閾値を下回った場合は、同一コンテンツの置いてあるミラーサーバへ接続要求を送る。ここで、クライアント側でサーバに対して送信データの割り当てを行ない、新たに接続したミラーサーバに対しては情報送信を始める場所を伝え、データ受信を開始する。プライマリサーバに対しては送信を終了する場所を伝える。

情報伝送中にサーバを切替え、送信データを分割して割り当てることで、ネットワーク状況の変化への対応と効率的なデータ伝送が実現できると期待できる。また、本方式においてはクライアント側で全ての制御を行なう。サーバ側で制御を行なう場合は、クライアントの再接続やデータ分割の管理などの処理からサーバ側に負荷がかかることになる。サーバ側で制御を行なうことで、個々のクライアントアプリケーションがリダイレクションを意識する必要がないという利点もあるが、今回はサーバへの負荷の軽減という観点からクライアント側で制御を行なうことにした。

3.2 ネットワーク状況の観測

ネットワーク状況の観測を行なう方法としては、unix の ping コマンドを用いてクライアント-サーバ間のラウンドトリップタイムを測る方法と、受信したデータ量と経過時間から回線速度を推測する方法が考えられる。ping を用いる場合は、コマンドを実行した瞬間の回線状況が計算されるが、ネットワーク状況の変化は激しく、次の瞬間に回線速度が大きく変化する可能性もある。つまり、ping によるネットワーク状況の観測はあくまで一時的なものにすぎない。よって、本稿ではネットワーク状況の観測を行なう方法として、ある一定時間毎に受信したデータ量を計算し、(受信データ量/経過時間) から回線速度を推測する方法を採った。ある一定の時間内での平均回線速度となるため、現在の回線状況により近い数値であると判断したためである。

3.3 Client-Server 間の動作

以下に、本方式におけるクライアント-サーバ間の動作を説明する。

Step.1 プライマリサーバに伝送要求を送り、データ受信開始(図1中の(1)).

Step.2 回線速度 C_r (Kbyte/sec) を計算し(3.2節参照), ネットワークが混雑していると判断する閾値 C_i (Kbyte/sec) と比較する.

1. $C_r > C_i$ の場合 (ネットワーク状況が良好) プライマリサーバからデータ受信を続行.
2. $C_r \leq C_i$ の場合 (ネットワークが混雑)
 - 2.1 ミラーサーバへ接続要求を送信(図1中の(2)).
 - 2.2 プライマリサーバ, ミラーサーバに対し, 送信データの割り当てを行なう(3.4節参照)
 - 2.3 ミラーサーバに送信要求と, 送信データの先頭アドレスを送り, ミラーサーバからのデータ受信を開始(図1中の(3)).

Step.3 いずれかのサーバがデータの送信を完了した場合は, 未送信データ D (Kbyte) の全送信データに対する割合 R_D (判断する閾値 L_D (%) と比較する.

1. $R_D < L_D$ の場合
未送信データの再分割は行なわない.
2. $R_D \geq L_D$ の場合
 - 2.1 未送信データの再分割を行なう.(3.4節参照)
 - 2.2 先に送信を完了したサーバに送信データの先頭アドレスを送り, データの受信を再開.

Step.4 Step.3を繰り返し, 全てのデータを受信したらデータを結合し終了.

3.4 送信データの分割

プライマリサーバ, ミラーサーバに送信データを割り当てる方法を説明する.

3.4.1 回線速度比による分割

リアルタイム性が要求されないデータファイルなどの場合は, 全てのデータ受信が完了した後で分割されたデータを結合して復元すればよいため, 未送信データをデータサイズのみを考慮し分割する. 分

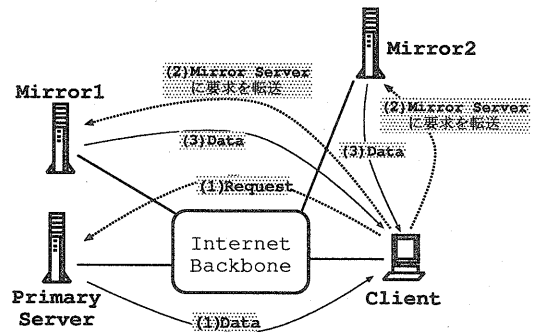


図1: クライアント-サーバ間の動作

割, 結合の処理が単純なため, ファイル処理によるオーバーヘッドも少ない. ここでは, プライマリクライアント間, ミラークライアント間の回線速度比を計算し, それをもとに割り当てるデータのサイズを決定する.

以下にデータ分割の手順を説明する.

Step.1 (受信データ量/経過時間) から, プライマリサーバの回線速度: $P.C_r$ を計算. ミラーサーバの回線速度 $M.C_r$ は, 1回目は ping で得られるラウンドトリップタイムをもとに計算する(2回目以降はプライマリサーバと同じ).

Step.2 送信データ量 D に対し, プライマリサーバが送信するデータ量を D_P , ミラーサーバが送信するデータ量 D_M とし, 以下のように計算する.

$$D_P = D \times \frac{P.C_r}{P.C_r + M.C_r}$$

$$D_M = D \times \frac{M.C_r}{P.C_r + M.C_r}$$

Step.3 ミラーサーバには(2)のアドレスを送信し, プライマリサーバは(2)で送信を終了する(図2参照).

3.4.2 一定サイズにスライスした割り当て

インターネット中継などのようにリアルタイム性が要求される場合は, 送信データを一定サイズにスライスし, 先頭から順に各サーバに割り当てる. クライアントは全てのデータ受信を待つ必要はなく, 受信したデータから順に結合し再生する.

リアルタイム性が重視される場合は, いかにかに速く伝送するか, というよりも, ある一定の伝送速

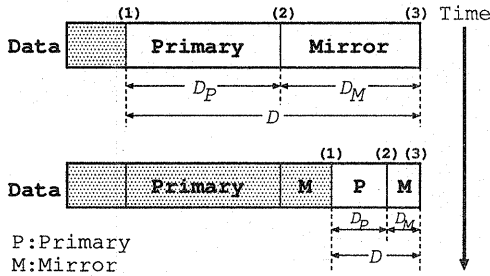


図 2: 回線速度比による分割

度を確保することが重要となる。例えば画像品質が観賞に耐えうる程度の回線速度が確保されれば、それ以上の速度を求める必要はない。ここで、必要最低限の回線速度を $L.C_r$ とする。

以下にデータ分割の手順を説明する。

Step.1 送信データ D (Kbyte) を、 D_s (Kbyte) 毎にスライスする。

Step.2 プライマリサーバ、ミラーサーバに割り当てる数をそれぞれ M, N とすると、

$$D = D_s \times M + D_s \times N$$

M, N は 3.4.1 節と同様に、それぞれの回線速度から計算する。

Step.3 クライアント側でスライスしたデータを交互に受信し、受信データ量と経過時間から回線速度を計算。

1. $L.C_r \leq P.C_r + M.C_r$ の場合
スライスの割り当て数を変更しない。
2. $L.C_r > P.C_r + M.C_r$ の場合
 $L.C_r \leq P.C_r + M.C_r$ となるまで回線速度の早いサーバのスライス割り当て数を増やす。

4 実験

本方式を評価するため、FTP クライアントを作成し、ネットワーク状況の観測と送信データの分割、結合、さらにサーバリダイレクションの機能を付加し、インターネット上で実験を行なった。今回はインターネット上の FTP サーバを利用し、クライアントが最初にダウンロードを開始したサーバ

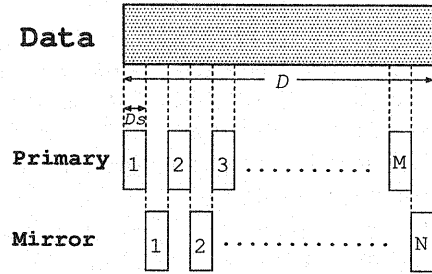


図 3: 一定サイズにスライスした割り当て

をプライマリサーバとして、クライアントがデータを受信しながら回線速度の推測を行なう。回線速度がネットワークが混雑していると判断される閾値と比較し、閾値を下回った場合はミラーサーバに接続し、2つもしくは3つのサーバに送信データを割り当て、並行してダウンロードを行なった。

4.1 実験結果

図 4 に一つのサーバのみからファイルをダウンロードした場合と、情報伝送中に回線速度が悪化した場合にミラーサーバへのリダイレクションを行ない、2つ、もしくは3つのサーバから並行してダウンロードを行なった結果を示す。ファイルの分割は 3.4 節で示した回線速度比による分割を行なっている。分割回数は 1 回、ネットワークが混雑していると判断する閾値は 64(Kbyte/sec) とした。

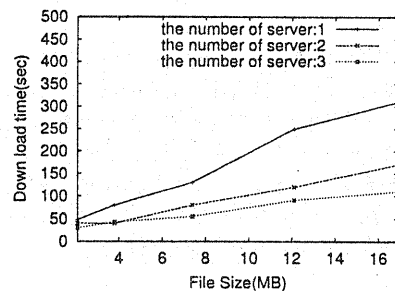


図 4: リダイレクションによるダウンロード時間の変化

縦軸にダウンロード時間、横軸にダウンロードするファイルのサイズとする。情報伝送中にサーバのリダイレクションを行なった結果、ダウンロード時間の短縮が確認できた。また、ファイルサイズが大

きい場合に、その効果が大きく現われている。ファイルサイズが小さい場合にダウンロード時間の差が少ないのは、ミラーサーバに接続する時間やファイルの分割、結合などの処理時間の占める割合が、全体の処理時間に対して大きいことが考えられる。

図5に送信データを分割する回数が0,1,2回の場合の比較を示す。分割したデータを割り当てるサーバ数は2とした(0回の場合はサーバ数1)。分割を行なう場合と行なわない場合の差は大きい。1回と2回の場合の差はあまり感じられない。2回目の分割を行なう時点で、全送信データに対する未送信データの割合が少ない場合には、分割を2回以上行なうことの効果が少ないと考えられる。つまり、プライマリサーバとミラーサーバとの回線速度に大きな差がない場合は、送信データの分割回数を増やしてもさほど変化はない。逆に、プライマリサーバとミラーサーバの回線速度差が大きい場合は、ファイルの分割を2回以上行なうことによるダウンロード時間の短縮効果は大きいと考えられる。

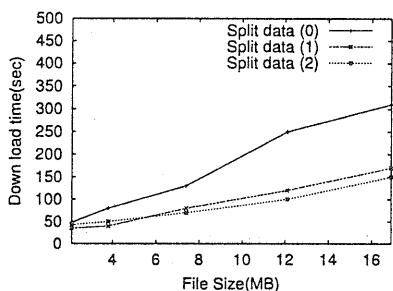


図5: 分割回数によるダウンロード時間の変化

また、図6にネットワークが混雑していると判断する閾値を32,64,128(KByte/sec)とした場合のダウンロード時間の比較を示す。サーバ数は2、分割回数は1回とする。閾値が32Kのときにダウンロード時間が長くなっている。これは閾値を低く設定するとそれだけプライマリサーバのみからの情報伝送時間が長くなるためと考えられる。

5 まとめ

本稿ではネットワーク状況の変化に対応するために、情報伝送中にサーバのリダイレクションを行なう方式を提案した。また、インターネット上のFTPサーバを利用した実験を行ない、本方式を用いた場合と一つのサーバのみからダウンロードする場合の、ダウンロード時間の比較を行なった。

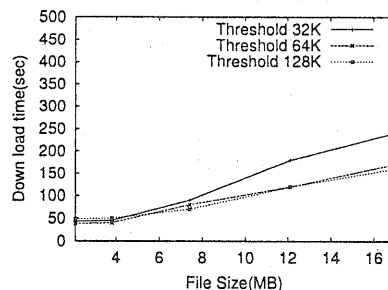


図6: 閾値によるダウンロード時間の変化

その結果、ダウンロードするファイルのサイズが大きい場合に、ダウンロード時間の短縮効果が顕著にみられた。また、クライアント-プライマリサーバ間とクライアント-ミラーサーバ間の回線速度差が大きい場合に、送信データを再分割することの効果も大きいことが確認できた。本方式を用いても効果が得られにくいのは、ファイルサイズが小さい場合や、クライアントに近い場所にボトルネックが存在する場合であると考えられる。サーバに近い場所にボトルネックが存在する場合はリダイレクションを行なうことで回避できるが、ユーザに近い場所に存在する場合、必ずその場所で回線速度が下がってしまうためである。逆に、サーバに近い側にボトルネックが存在する場合に本方式が有効であるといえる。

今後、リアルタイム性が要求されるデータ伝送において、本方式の有効性を確かめることが課題である。

参考文献

- [1] A. Paul and L. Cricket. *DNS and BIND*. O'Reilly & Associates, Inc, 1992
- [2] URL=<http://service.jp.real.com/> RealServer 管理ガイド.
- [3] 竹内, 小野里, 山本, 清水, 木村. インターネットにおけるトラフィック適応型サーバ選択方式. 情報処理学会第55回全国大会, 5U-1,1997.
- [4] 富沢, 山本, 小野里. WWWにおけるラウンドロビン-トラフィック適応型サーバ選択方式. マルチメディア通信と分散処理ワークショップ, P181-186,1998.