

放送型ネットワークにおける通信エラーの対処方式の評価

鈴木 健 アフマドルリィ 岡田 謙一
慶應義塾大学理工学部

辻 順一郎
通信・放送機構 新川崎リサーチセンター

我々は、通信と放送を融合したネットワークにおけるマルチメディアシステムの構築技術に関する研究を行っている。地上波テレビ放送、衛星放送等のデジタル化に伴い、これまでのテレビ放送とコンピュータの融合が進展することが予想される。そのような状況のもとでは放送ネットワークにおける通信エラーはこれまで以上に大きな問題となっていく。現在の放送ネットワークにおける通信エラーへの対処としては同じデータを繰り返し放送する手法が主流であるが、今後の放送データの増加に伴い新たな通信エラーへの対処方式の開発が必要である。本稿では、基本的に一方向の通信である放送ネットワークとは別個に双方向通信路を設け、通信エラーを検出したクライアント端末が能動的に通信エラーを起こしたデータを獲得に行くような通信エラー対処方式について紹介し、実験とシミュレーションにより評価する。

Evaluation of Error Correction Mechanism Using Proxy on Broadcasting Network

Takeshi Suzuki Achmad Rully Kenichi Okada
Faculty of Science and Technology,
Keio University

Junichiro Tsuji
Telecommunications Advancement Organization
of Japan Shin-Kawasaki Research Center

We are now investigating the technology for constructing multimedia system on the integrated communication/broadcasting network. Recently, with the progress of Digital Broadcasting Network, integration of broadcasting network and computer network will become more popular. In such environment, transfer error recovering will become more important. In this report, we will review new method for transfer error recovering on broadcasting network, in which terminal of broadcasting network will recognize the transfer error, and it will require the correct data to server through computer network. And we will evaluate this system using experiment and simulation.

1. はじめに

近年、衛星デジタル放送や地上波テレビ放送における垂直帰線期間(VBI)を用いたデータ放送など、テレビ放送のデジタル化が進み、地上波デジタルデータ放送の近い将来の実現が確実視されてきている。

現在、放送ネットワークにおける通信エラーへの対処方式としては、同じデータを周期的に繰り返し放送するという方法があるが^{1),2)}、この方法では、放送ネットワークにおけるデータ転送速度は各データを繰り返し放送する回数に反比例して低下する。

また、他の方法として、放送ネットワークとは別個に設けた双方向通信路を介してクライアント端末が通信エラーを発生したデータの識別子を双方向通信路を介して

放送サーバに送り、放送サーバは同時に通信エラーの報告が多数通知されたデータを優先的に再放送するという手法がある³⁾。しかし、この方法では、一部のクライアントのみに対して通信エラーが発生したデータに関しては再放送されないという問題点がある。

以上の問題点を解決する方法として、放送ネットワークとは別個に設けた双方向通信路を利用して通信エラーを検出したクライアントが能動的に双方向通信路を介して正しいデータを獲得するという方式を、通信・放送機構新川崎リサーチセンターにおいて提案している⁴⁾。そこで本稿では、通信・放送機構において提案された方式の評価を行なう。実際のプロキシシステムを用いた実験による評価とシミュレーションによる評価を行い、この方式の有効性を確かめる。

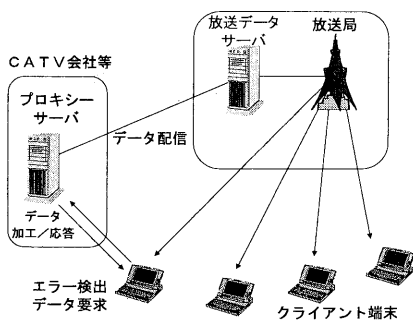


図1 概念図

2. 放送型ネットワークにおける通信エラーの対処方式

図1に、本方式の構成概念図を示す。中央の放送局には放送サーバが設置され、順次放送ネットワークを介して多数のクライアント端末にデータを配布する。また、同様のデータを一定のクライアント毎に設置されたプロキシサーバに送付し蓄積する。このプロキシサーバは、例えば、地域毎に設置されているCATV会社等に設置されるものであり、その地域の各クライアントと双方向通信路を介して接続される。この双方向通信路は、家庭等の固定されたクライアント環境においては一般の公衆電話回線やCATV会社のケーブルネットワーク等を用い、モバイル環境においては、PHSや携帯電話等の移動体通信を用いる。

各クライアント端末は、何らかの外的要因により、通信エラーが発生すると、その通信エラーを検知して、双方向通信路を介してプロキシサーバに該当するデータの獲得要求を発行する。プロキシサーバでは、データ獲得要求を受信すると、あらかじめ蓄積されている該当データを双方向通信路を介して、クライアント端末に回答として転送する。この際、プロキシサーバにおいて、クライアント端末とプロキシサーバの間のデータの転送を高速化するために、放送データを圧縮処理あるいは、メディア変換を行って、データ量を削減した形で蓄積する。この圧縮及びメディア変換に関しては、WEBアクセスにおいて、モバイル環境におけるマルチメディアアクセスを高速化するための手法として昨今盛んに研究されている方式^{5),6)}を利用する。

本方式を用いると、通信エラーが起った時、プロキシサーバに対して、多数のクライアントから同じデータに対する要求が同時に発行される。そのため、プロキシサーバからクライアントへの応答が遅れる可能性がある。そこで、プロキシサーバに対して多数のクライアントから同一のデータに対する要求が集中したときのプロキ

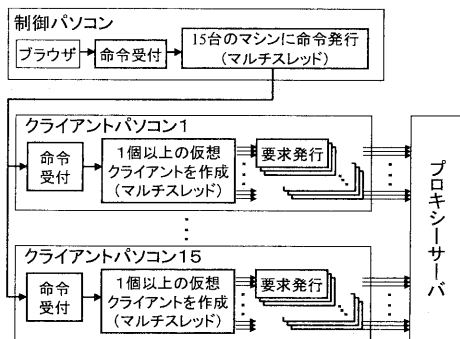


図2 評価システム構成

シーサーバの応答性能の特性を実際のプロキシを用いた実験、およびシミュレーションを用いて評価する。

3. 実際のプロキシを用いた評価

単発のエラーに対するプロキシサーバの応答特性の評価のために実際のプロキシサーバを用いた評価を行った。

3.1 評価システムの構成

実際のプロキシシステムを用いて実験を行うための評価システムを構築するにあたって、数百台のクライアント端末を用意して実験を行なうことは不可能であるので、プロキシサーバに対して擬似的に数百台分の要求を出せるような評価システムを開発した。図2に評価システムの構成を示す。図2において、ブラウザは既存のインターネットエクスプローラ5.0を用いた。また、ブラウザ以外の部分はC++にて作成した。

評価システムはブラウザを含む制御パソコン1台と、クライアントパソコン15台とプロキシサーバ1台から構成される。制御パソコン上のブラウザから命令を発行する。ブラウザからの命令をマルチスレッド機能を用いて15台のクライアントパソコンに送信する。各クライアントパソコンでは制御パソコンからの命令を受け付け、マルチスレッド機能を用いて1個以上の仮想クライアントを作成する。作成された仮想クライアントがそれぞれプロキシサーバに対して要求を発行する。仮想クライアントには、プロキシに要求を発行してから要求が帰ってくるまでの時間を測定する機能を付与した。各クライアントパソコンでの仮想クライアントの作成する数を変化させることにより、プロキシサーバへの同時要求の数を変化させ、プロキシサーバの応答特性を評価するようにした。

また、プロキシサーバは、キャッシュデータとしてWEBサーバからの応答を蓄積する機能と静止画情報の品質を低下させてデータ量を削減するメディア変換機能を備えている。プロキシサーバは、要求された画像に

対して、そのままオリジナル画像を返信、要求毎にメディア変換をして容量を小さくして返信、メディア変換した画像をキャッシュしておきそれを返信、の3種類の応答方法を持つ。

3.2 実験結果

図3にオリジナル画像を返信する場合の実験結果を示す。30Kバイトから180Kバイトの各種サイズを持つ5つのJPEGファイルについて、それぞれ30個から120個のクライアントから同時に要求する場合の応答時間を測定した。それぞれ、10回測定を行ないその平均を応答時間とした。各サイズとも台数の増加に比例して応答時間が増加している。クライアント数120、177Kバイトの時、約3秒強の応答時間である。

各要求毎にメディア変換して返信する場合の実験を行った。メディア変換を行うことにより、データ量を95%削減して元の画像の5%となる。クライアント数120、177Kバイトの時、約60秒強の応答時間を示すことを実験により確認した。これは、データの送信にかかる時間の減少以上に各要求毎にメディア変換を行うことに時間がかかるからである。よって、オリジナル画像取得する場合に比べ応答時間が大きいので、本システムにおいて、各要求毎にメディア変換するのは不適切といえる。

しかし、本システムにおけるプロキシサーバは、放送にさきがけて、データが配送されるので、その時点でメディア変換を行い、そのデータをキャッシュ上に保存する。各クライアントに対してはキャッシュ上に保存されたメディア変換後のデータを返信することにより、応答時間を削減することが可能となると予想される。

図4にキャッシュにあるメディア変換済の画像を返信する場合の実験結果を示す。各サイズとも台数の増加に比例して応答時間が増加していることから、プロキシの動作としてはデータサイズの小さなオリジナル画像の返信するのと同じであるといえる。クライアント数120、177Kバイトの時、約2秒弱の応答時間となっているので、図3に比べて応答時間が約半分弱になっているといえる。これは、データサイズが元の5%になることにより、実質上約8kバイトのオリジナル画像を取得することになるため、データの送信にかかる時間が減少したからである。

3.3 考察

実験結果より、一台のプロキシサーバにおいて100台規模のクライアントからの同時データ要求に対して、平均として3秒強の範囲内で応答が可能であることを確認した。また、100K以下のサイズのデータに対しては100台規模の同時要求に対しても、平均として2秒強での応答が確認できており、100K以上のデータサイズを持つデータに対しては100K以内のサイズに収まるようなデータのブロック化を施すことにより、さらに高速な応答が可能であると考えられる。

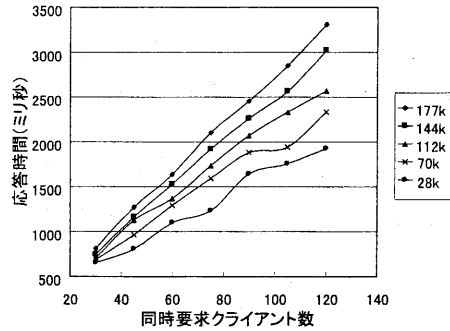


図3 オリジナル画像取得時の応答時間

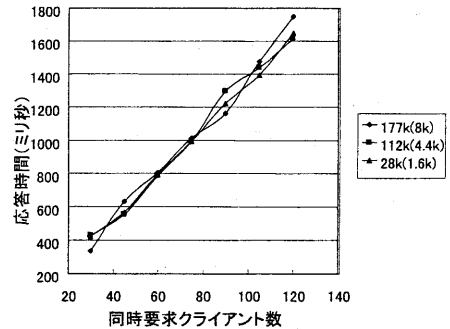


図4 メディア変換済の画像取得時の応答時間

また、メディア変換処理を本システムにおけるプロキシサーバに適用し、放送サーバから配信されるマルチメディア情報にメディア変換処理を行った結果としてのデータをキャッシュに蓄えることにより、応答時間を大幅に短縮することが可能であることを確認した。メディア変換処理を用いることにより、より効果的な放送ネットワークにおける通信エラーへの対処を実現することができる。

4. シミュレーションによる評価

前節では、100台規模のクライアントからの同時要求に対するプロキシの応答特性に関して、実際のプロキシを用いた評価を行った。このとき、単一のエラーに対する各クライアントからの要求に対するプロキシの応答時間を計測した。

しかし、地域毎にCATV会社等の拠点にプロキシサーバを配置した場合、地域内におけるクライアント数はかなりの数になることが予想される。平成9年度末現在で、自主放送を行うCATV会社は720社でその加入契約数は672万であり、平均して一社あたり約10000の契約数となっている⁷⁾。データ放送の視聴率を考慮して

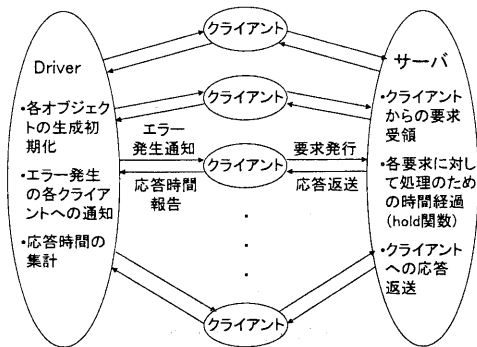


図5 オブジェクト構成

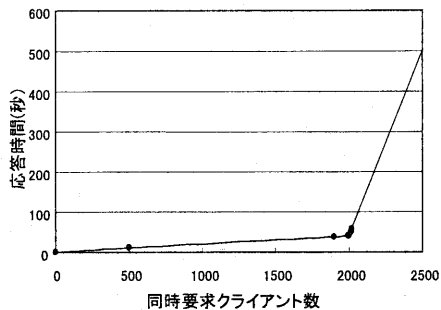


図6 エラー発生間隔 80 秒における応答時間

も、数百から数千のクライアントからの同時要求に対処する必要があると思われる。そこで、大規模な放送ネットワークにおける本システムの評価を行うためにシミュレーションシステムを構築して評価を行うこととした。

また、実際の放送システムでは、無線通信路の信頼性が有線通信路に比較して格段に劣ることから、通信エラーは間欠的に発生し、プロキシの応答特性がより大規模な放送システムにおけるクライアント数に対して充分でない場合には、前のエラーに対する応答処理の遅延が、次のエラーに対する応答処理に影響することが考えられる。そこで、シミュレーションによる大規模システムの評価においては、無線通信路のビット誤り率を勘案して、周期的にエラーが発生するような状況を仮定した評価を行うこととした。

シミュレーションシステムの構築にあたっては、カリフォルニア大学ロサンゼルス校 (UCLA) で開発された、オブジェクト指向に基づくシミュレーションシステムである Parsec システム⁸⁾ を用い、数千台規模までのクライアントからの同時要求に対するプロキシサーバの応答特性を評価した。

4.1 シミュレーションシステムの構成

図5に、本評価におけるシミュレーションシステムのオブジェクト構成を示す。

Driver オブジェクトは、ひとつのサーバオブジェクトと、同時要求を発行するだけの数のクライアントオブジェクトを生成し、クライアントとサーバの通信路の設定を行う。また、無線通信路におけるエラー率に基づいて設定した各エラーの発生時刻に従って、各クライアントオブジェクトエラーの発生を通知するメッセージを送付する。また、各クライアントからの応答時間の報告メッセージを受け取って、その平均値を記録する。

サーバオブジェクトはプロキシサーバを表すオブジェクトであり、各クライアントからの要求メッセージを受け取り、hold 関数を用いて各要求に対する処理時間の後、各クライアントに対して応答メッセージを返送する。各

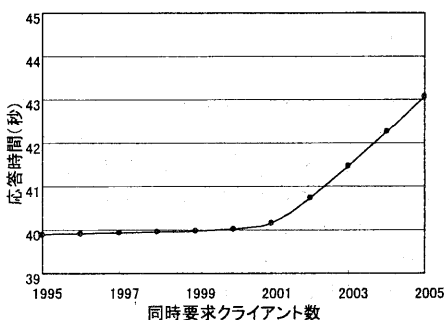


図7 エラー発生間隔 80 秒における変曲点付近の応答時間

要求に対するサーバにおける処理時間は、実験による評価において計測した同時要求の際の平均応答時間をもとに設定し、乱数を用いて各要求毎に前後 10% ずつの揺らぎを持たせることとした。

各クライアントオブジェクトは、ひとつの仮想的なユーザ毎に作成し、Driver オブジェクトからのエラー発生通知に従ってサーバオブジェクトに要求メッセージを送付する。サーバからの応答メッセージを受け取ることにより、要求発行から応答受領までの応答時間を測定し、その結果を Driver オブジェクトに通知する。

4.2 シミュレーション結果

シミュレーションにおいて、放送ネットワークで放送されるデータのサイズは 100K バイト単位とし、それらのデータをクライアントから要求した際のサーバにおける処理時間を平均 40ms とし、乱数を用いて前後 10% の揺らぎを与えることとした。実際には、応答時間にはネットワークにおける伝送遅延等が加算されるため、プロキシサーバにおける一要求あたりの処理時間はそれより低い値が予想される。また、プロキシサーバにおける処理時間の揺らぎに関しては、プロキシサーバの負荷などの要因で各要求に対する処理時間に揺らぎが出るこ

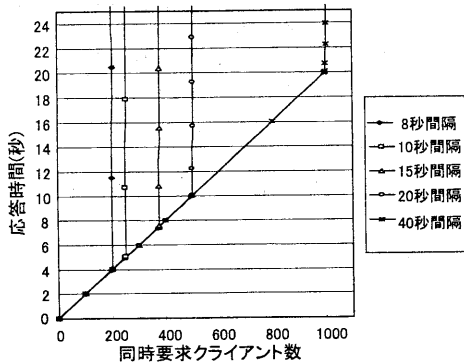


図8 エラー発生間隔8秒から40秒における応答時間

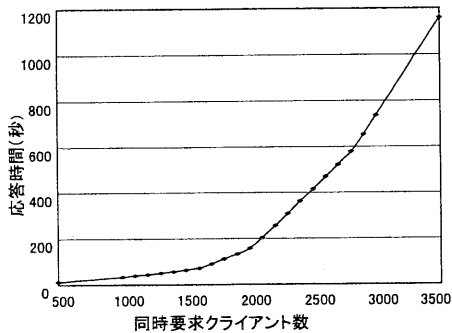


図9 エラー発生間隔平均80秒(指数分布)の応答時間

とを想定したものであり、前節における実験における複数回の計測の中での応答時間の揺らぎを基に前後10%とした。

また、シミュレーション上での経過時間は1時間とし、平均エラー発生間隔を8秒～80秒としてその平均応答時間を求めた。

図6にエラー発生間隔を80秒とした場合の同時要求クライアント数に対する平均応答時間を示す。また、図6のグラフ中の変曲点付近を詳細を図7に示す。また、図8に、エラー間隔を8秒、10秒、15秒、20秒、40秒としたときの結果を示し、図9に、エラー間隔を平均80秒の指数分布とした時の結果を示す。

4.3 考察

図6、及び図7に示されるように、エラー間隔80秒の時約2000台を境として、同時要求クライアント数の増加に対して、平均応答時間が急激に増加している。プロキシにおいてクライアントからのデータ要求に対する応答処理を行う上で、クライアント数が増加するにつれ、単一のエラーに対する全ての処理を終了するまでの時間が増加していくのであるが、その単一のエラーに対する応答処理が終了する前に次のエラーが発生する状況が起

り、そのために次のエラーのための応答処理の開始が遅れ、その遅延が次々と蓄積していくことにより、応答時間が急激に増加してしまうことを表している。

これらの閾値付近における平均応答時間は約40秒程度となっている。また、図8より、エラー間隔が8秒の場合応答時間が4秒程度、エラー間隔が20秒の場合応答時間が10秒程度である事がわかる。また、エラー間隔が80秒の場合に、同時要求クライアント数が500台において10秒弱、1000台において20秒弱の平均応答時間を示している。一般にユーザインタフェースにおいては、各種の応答時間として2秒が限界であるといわれている⁹⁾。しかし、本システムにおいて、通信エラーに対する正しいデータの取得は、クライアント端末がエラーを認識することによって自動的に行われることを考えるため、要求のプロキシサーバへの発行の時刻をユーザが意識するわけではなく、本システムにおける応答時間としては数秒から10秒程度が限界と考えられる。従って、図8より約500台が同時要求クライアント数の限界と考えることができる。

このことから、放送ネットワークにおける通信エラーへの対処として一つのプロキシにおいて処理し得る同時要求クライアント数の限度は、エラーが頻繁に発生するような状況においてはエラーの発生間隔が問題となり、例えば8秒間隔程度のばあいには200台程度となる。また、エラーが余り起こらない状況においては、一つのエラーに対するプロキシの応答処理の負荷が問題となって約500台が同時要求クライアント数の限度と考えられる。

従って、広域放送ネットワークにおいて、提案する通信エラー対処方式を適用する場合には、実際のエラーの平均発生間隔と一つのエラーに対するプロキシサーバの応答性能を考慮して、複数のプロキシサーバを適切に配置して、プロキシサーバの負荷分散を図ることが必要となる。あるいは、前節で応答処理の高速化に有効であることが確認できたメディア変換処理を適用する必要があると思われる。

以上の結果は、エラー間隔と平均応答時間の関連を明確にするため、エラー間隔を固定として周期的にエラーが発生するような状況を仮定したシミュレーションを行った結果である。しかし現実的にはエラーは偶発的に発生するため、その間隔は一定となることはない。そこで、より現実的な状況を見るために、エラー間隔を2秒から350秒の間で指数分布に従うようにエラーを発生させ、平均エラー間隔が80秒となるような状況でのシミュレーションを行った。その結果、エラー間隔を80秒に固定した場合の図6のように2000台付近で急激に応答時間が悪化するのではなく、図9に示すように、1500台程度から応答時間が悪化し始め、2000台付近で更に急激に悪化することがわかる。また、1500台以下においても、エラー間隔を固定した場合と比べて、その応答時間は若干悪くなっ

ている。このように、より現実的な状況では、エラー間隔を固定した時に比べて、その応答性能は悪くなる傾向があるので、実際に本システムを運用する場合には、エラーの発生状況を考慮して、プロキシの配置等について検討することが必要であると思われる。

5. ま と め

本稿では、放送ネットワークにおけるプロキシサーバを用いた通信エラーへの対処方式について紹介し、実際のプロキシを用いた実験とシミュレーションにより評価を行った。本方式におけるプロキシサーバは、クライアントから要求されるデータが、放送ネットワークにおいて配布されたデータに限定されることから、従来のインターネットにおけるWEBアクセスのためのプロキシサーバ等とは異なり、キャッシュのヒット率は100%に近く、適切なプロキシサーバの配置により、各クライアントに対して十分なエラー対処機能を提供できることを確認した。

さらに、マルチメディア変換機能をプロキシサーバに適用し、あらかじめ放送データがプロキシサーバに配信された時点でメディア変換処理を実施し、プロキシサーバのキャッシュ上にメディア変換後のデータを格納しておくことにより、多数のクライアントからの同時要求に対してさらに高速な応答が可能であることを確認した。

以上の評価により、本システムが有効であることが確認できた。

参 考 文 献

- 1) T. Imielinski and S. Viswanathan. *Wireless Publishing: Issues and Solutions*, chapter 11, pp. 299-330. Kluwer Academic Publishers, Norwell, Massachusetts, 1996.
- 2) S. Acharya, R. Alonso, M. Franklin, and S. Zdonik. *Broadcast Disks: Data Management for Asymmetric Communication environments*, chapter 12, pp. 331-362. Kluwer Academic Publishers, Norwell, Massachusetts, 1996.
- 3) 河村真一, 久野琢磨. 情報伝送方式及びシステムと、該システムに用いるホスト装置及び端末装置, 1998. 公開特許公報 (特開平 10-243372) .
- 4) 辻順一郎, 水野忠則, 佐藤文明. 放送型ネットワークにおける通信エラーの対処方式の提案. 情報処理学会研究報告 (モバイルコンピューティング研究会), Vol. 99, No. 13, pp. 71-77, 1999.
- 5) 辻順一郎, 小津浩直, 三浦敦史, 滝沢直樹, 水野忠則. モバイルプロキシサーバシステムの試作. 情報処理学会研究報告 (モバイルコンピューティング研究会), Vol. 97, No. 72, pp. 67-72, 1997.
- 6) 伊東直子, 立川恒央, 中川好介, 水越康博. 携帯端末向けの WWW ページ変換 - SmartProxy, 1997. (<http://pochi.nwsl.mesh.ad.jp/nwsl/papers/09732/>).
- 7) 郵政省. 平成 11 年版通信白書, 1999.
- 8) R. Bagrodia, R. Meyer, M. Takai, Y. Chen, X. Zeng, J. Martin, B. Park, and H. Song. Parsec: A Parallel Simulation Environment for Complex Systems. *Computer*, Vol. 31, No. 10, pp. 77-85, October 1998.
- 9) Ben Shneiderman. ユーザ・インタフェースの設計 - 使いやすい対話型システムへの指針 -. 日経マクロウヒル社, ISBN 4-8222-7053-X, December 1987.