

高効率大容量ファイル転送プロトコル NBT on XCAST

住田智雄† 伊東克能†

本稿では、ピュアな IP 網への適用を考慮した、蓄積型大容量ファイルを効率的に転送するプロトコル NBT について述べる。従来のファイル転送はユニキャストが基本であり、同一の蓄積型大容量ファイルを同時に複数個送信することは帯域の無駄遣いになるという欠点がある。この欠点を除こうとして先に提案した NBT (Non-ordered Block Transfer) では、マルチキャストに ATM を採用しており、使用領域が制限されるという問題点があった。その解決策として、新しいマルチキャストプロトコルとして提案されている Explicit Multicast (XCAST) を採用することにより、汎用性のある NBT の実現を目標とする。提案方式は、大容量のファイルを高頻度で転送する場合に特に有効であると考えられる。

A Highly Efficient Transport Protocol for Large Capacity Files: Non-ordered Block Transfer on XCAST

Tomoo SUMIDA† Katsuyoshi ITO†

This paper describes on a protocol named NBT (NBT: Non-ordered Block Transfer) which efficiently transfers large-capacity stored files, and is applicable in a pure IP network. Traditionally, file transfer is performed on unicast basis. However, the unicast wastes bandwidth by sending the same large-capacity file to each request, when plural download-requests are simultaneously made. NBT has been proposed to alleviate this problem. Since the original NBT adopts ATM as a multicast scheme, the network domain in which the NBT can be used is restricted. As the solution, adoption of Explicit Multicast (XCAST) is proposed as a multicast scheme, resulting in a flexible NBT. The proposed system is especially effective when the server frequently transfers large capacity files.

1 はじめに

ここ数年の間にインターネット関連の技術は著しく発展してきた。それに伴って、インターネットを利用するユーザ数も爆発的に増加している。しかしながら、このように著しい成長を遂げるインターネットにも様々な問題が生じてきていることも事実である。その一つとして、大容量ファイルを配信するために必要となる帯域の増大、デ

ータ転送方式の不適合などが挙げられる。MPEG-2 により圧縮された 2 時間前後の映画は約 5GByte のデータ量になるが、この大容量ファイルを現在最も広帯域な ADSL を用いた場合、約 15 分でダウンロードできる。高速・広帯域なインターネット環境が整い始めた今では、上述のように大容量データの配信も可能になってきている。しかしながら、同一の GByte クラスのデータファイルを複数のクライアントが同時に要求すると、現在利用されているユニキャスト方式では、サーバでクライアントの数だけデータを作成し、各クライアント

† 東洋大学大学院工学研究科

Graduate School of Engineering, Toyo University

に向けて送信しなければならない。またクライアントまでの途中経路が重なる部分においても、同一内容の packets が複数個存在することになる。つまり、ユニキャスト方式では、サーバに対する余分な負荷、帯域の無駄遣いといった問題が生じる。この問題の解決は、インターネットサービスにとって重要な課題の一つであるといえる。

本稿では、

- ・ 蓄積型大容量ファイルの転送
- ・ 複数のクライアントを対象とした同一ファイルの効率的な配信
- ・ 転送方式にマルチキャストの一つである XCAST を採用

を前提として、大容量ファイル転送を効率的に行う方式として先に提案した順序非保存ブロック転送方式 (NBT: Non-ordered Block Transfer) [1] の改良、有効性の検討、再送方法の考察と基本的動作の確認について述べる。

以下、2 章では、順序非保存ブロック転送方式 NBT の概要とその有効性、本稿で提案するピュアな IP 網への適用を考慮した NBT について述べる。3 章では、基本的な動作の確認を行うための構成と再送方法について述べ、4 章で部分的動作実験の結果について述べる。5 章は結論である。

2 順序非保存ブロック転送方式

順序非保存ブロック転送方式 NBT の概要、その有効性に関する考察、新たに提案した Explicit Multicast (XCAST) [2] を用いる NBT について述べる [3]~[5]。

2.1 概要

ファイルサーバにおいて、ファイルの転送時間内の異なる時刻に複数のクライアントから転送要求が生じた場合に、個別に対処するのではなく、マルチキャストにより経路上のノードにおいてコピーを作成することにより、サーバにおけるファイルの転送処理の軽減と、ネットワークの帯域の消費量の最小化を図ることが NBT の目的である。

図 1 において、サーバとクライアント A の間でデータファイルを送信中に、クライアント B が、ついでクライアント C が同一内容のデータファイルを要求した場合を想定する。データファイルを

A1+A2 とする。クライアント B へは、クライアント A 宛てに送信中のデータファイル A2 のコピーを適切な転送経路の途中のノードで作成し、それをクライアント B に送信し、後にサーバからデータファイル A1 を送信する。クライアント C についても同様な手順でデータファイルを送信する。

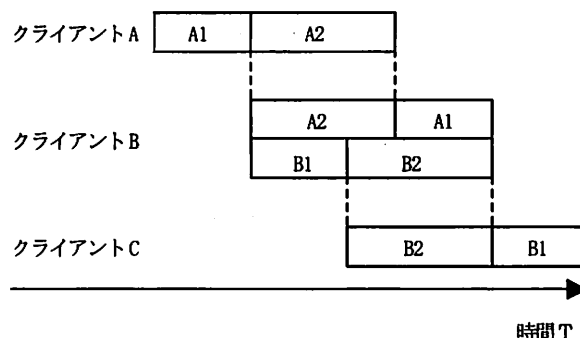


図 1 NBT の基本的動作

このような方法で大容量ファイルを効率的に配信することを目的として提案された NBT であるが、先に提案した NBT では、ATM によるマルチキャストを想定していた。そのため使用領域が制限される可能性があった。

2.2 NBT on XCAST

上述の問題点の解決策として、ピュアな IP 網への適用とサーバ側主導の制御を考慮して、新しいマルチキャストプロトコルとして提案されている XCAST を採用した。NBT を実装する上で XCAST を用いる利点として、

- ・ データリンク層に依存しないため使用領域に制限がない
- ・ 送信側制御であるため、課金情報を含み、各クライアント毎に異なる情報を把握しやすい
- ・ マルチキャストアドレス、マルチキャストルーティングプロトコルが不要なため、実装が容易

などが挙げられる。この XCAST を用いた NBT を、NBT on XCAST と呼ぶことにする。

2.3 NBT の有効性

NBT の採用により帯域の有効利用の程度を図 2 に示す単純化したネットワークモデルで検討する。ここで、各クライアントにファイルを転送するのに必要な帯域 B 、時間 T 、伝送距離 L の積の和で

ある帯域距離消費量 A をもってネットワーク利用効率を示す指標とする。

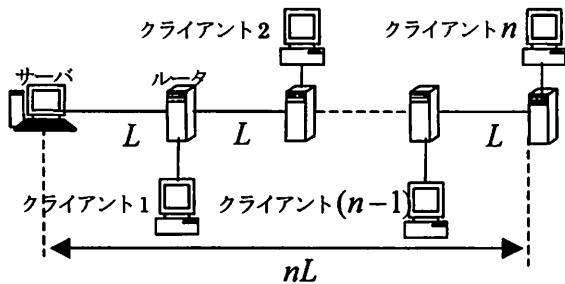


図 2 NBT の有効性を検討のため簡易化したネットワークモデル

(1) ユニキャストの場合

ユニキャスト方式の帯域距離消費量 A_u は

$$A_u = \sum_{i=1}^n iBTL = \frac{BTL}{2} n(n+1)$$

で与えられ、クライアント数 n の 2 乗に比例し、スケラブルではないといえる。ユニキャスト方式では、同一内容のファイルを要求してきたクライアントに対して、個別に送信するため、中継ネットワーク上に複数の同一内容の packets が存在してしまうためである。

(2) NBT の場合

NBT における帯域距離消費量 A_n は、

$$A_n = \frac{BTL}{n} \sum_{i=1}^{n-1} i + BT \cdot nL = \frac{BTL}{2} (3n-1)$$

で与えられ、送信順序に無関係に、クライアント数 n に比例する。従って、スケラブルであり、NBT の有効性が期待できる。NBT では、複数のクライアントに対して途中までの経路が重なる部分に関しては、単一の packets が伝送されるからである。

2.4 NBT on XCAST の転送能力

ファイルサイズ G 、帯域幅 B とするとファイルの転送時間 T は次の式で与えられる。

$$T = G/B$$

そのファイルの転送を同時に要求するクライアント数を n 、サーバが同時に m 個のファイルを処理可能とすると、単位時間当たりの総

転送数 N_h は次の式で与えられる。

$$N_h = mn/T = mnB/G$$

$m = 1, n = 64, B = 1\text{Gbit/s}, G = 5\text{GB}$ とすると、1日当たりに処理できるクライアント数 N は、

$$N = 1.4 \times 10^5$$

となる。これは、1Gbit/s のアクセス網が利用可能という条件下ではあるが、実用性がある数字であると考えられる。

3 NBT on XCAST の構成

NBT を実装する際のシステム構成について検討する。

3.1 システム構成

図 3 に想定する NBT on XCAST のシステム構成を示す。

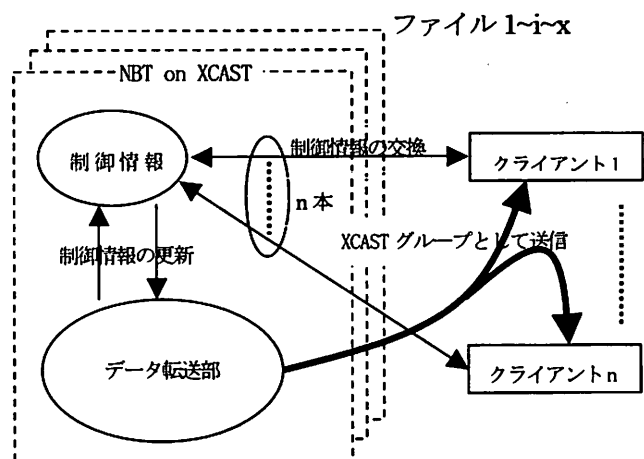


図 3 NBT on XCAST のシステム構成

ファイルの最初の要求者であるクライアント 1 の転送要求は、まず NBT on XCAST の制御情報部に接続される。制御情報部では、NBT on XCAST に関する必要な制御情報を作成し、それをクライアント 1 に送信する。つぎに、制御情報部はデータ転送部をスレッドとして起動する。データ転送部はクライアントが要求したファイルを、制御情報に基づいて送信する。NBT on XCAST に必要な制御情報は刻々と変化するため、制御情報部とデータ転送部は常に制御情報の更新と交換を行う。以下に、更新

する必要がある制御情報を示す。

- ・ クライアント番号
- ・ 各クライアントへの送信開始データブロック番号
- ・ 各クライアントへの送信終了データブロック番号
- ・ 現在送信中のデータブロック番号
- ・ 次に送信するデータブロック番号
- ・ 送信済みデータサイズ

2 番目以降のクライアントは、最初のクライアントと同様に制御情報部に接続され、最新の制御情報を取得した後、XCAST グループに参加して、既存のデータ転送部から要求したデータの受信を開始する。クライアント毎に異なるデータ受信の終了条件は、サーバとクライアント間の制御情報に基づいて決定される。

また、複数のクライアントからの異なるファイルの転送要求に対しては、各クライアントからの要求を、要求受信時に識別することによって対応する。同じコンテンツを要求した複数のクライアントを一つの XCAST グループにまとめ、異なる XCAST グループを複数作成することによって、複数の異なるファイル要求に対応する。

3.2 パス構成

図 4 に NBT on XCAST のパス構成を示す。

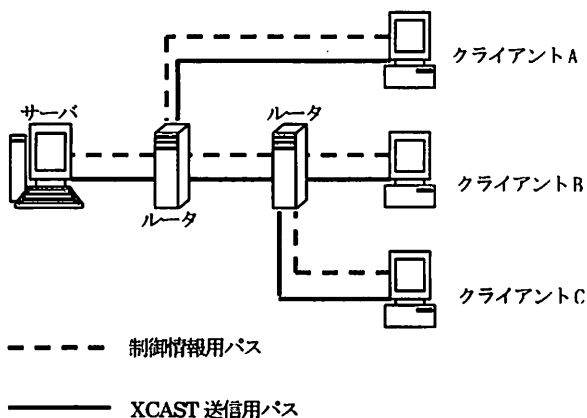


図 4 NBT on XCAST のパス構成

NBT on XCAST では、NBT 転送データと制御情報データのパスは分離した。つまり、データ転送用パス (XCAST/UDP) と制御情報用パス (TCP) の 2 つの

パスを設定した。このようにパスを 2 つに分けて管理することで、全体の設計と制御が容易になる [4]。

ファイル転送を要求する各クライアントは、制御情報用パスを用いてサーバにデータを要求する。サーバは、現在のクライアント状況やデータ配信状況など NBT を利用する上で必要な各種情報を、制御情報用パスを用いてクライアントに送信する。次にサーバは、クライアント毎に作成した制御情報管理テーブルを参照しながら、データ転送用パスを用いて各クライアントへデータを送信する。

3.3 再送方式

輻輳などの原因によってパケット損失が発生した場合にその部分のデータを再送する方式について考察する [5]。

NBT on XCAST の再送には、それぞれ 2 通りの再送方法および再送タイミングが考えられる。

再送方法

- クライアント毎に損失したパケットをユニキャストで再送する
- 同じデータを要求したクライアントが属する XCAST グループに対して、損失したパケットを XCAST で再送する

再送タイミング

- サーバが各クライアントからパケットの再送要求を受けた時点で個別に再送する
- サーバが、一通りのファイルを送信し終えた後、再送要求のあったパケットを一括して再送する

NBT on XCAST の再送は、この再送方法と再送タイミングの $2 \times 2 = 4$ 通りの組み合わせで考えることができる。表 1 に、NBT on XCAST における再送方法と再送タイミングの組み合わせを示す。

表 1 NBT on XCAST における再送方法とタイミングの組み合わせ

再送方法	再送タイミング	
	再送要求受信時	一括
ユニキャスト	i, iii	i, iv
XCAST	ii, iii	ii, iv

以下、表 1 に示した組合せについて考察する。

1) クライアント毎に再送する場合

クライアント毎に再送を行う場合、全てのクライアントのデータ受信状況を個別に管理しなければならない。また、同じ内容の packets を複数のクライアントに送信しなければならない可能性があり、効率的ではない。サーバに必要以上の負荷がかかる可能性もある。この方式を採用する場合は、再送はタイミグ iii で制御パスを通じて転送するのが妥当であろう。

2) XCAST グループに対して再送する場合

XCAST グループに一括して再送を行う場合、クライアントが再送要求を行ってから、実際に再送データを受信するまでに時間がかかる可能性があるが、再送する際に作成される packets は一つだけなので、サーバの負荷が軽減される。再送はタイミグ iv で XCAST パスを通じて行われる。

以上の考察から、NBT on XCAST における損失データの再送については、ファイル転送終了時に XCAST グループに対して、XCAST パスで再送する方法が優れていると考える。

4 部分的動作実験

前章で述べた NBT on XCAST の構成に基づいて部分的な実装を行った結果について述べる。ネットワークポロジを図 5 に示す。

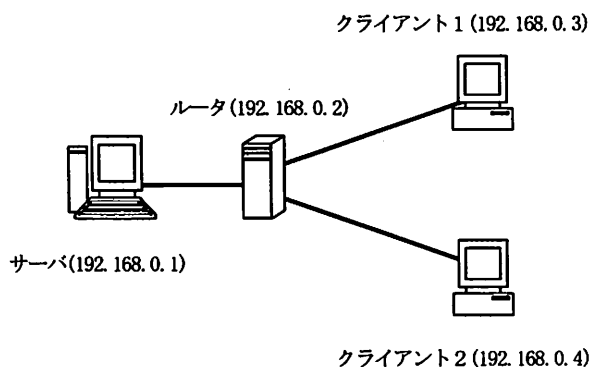


図 5 ネットワークポロジ

2 つのクライアントがファイルの転送時間内の異なる時刻に、サーバ側の大容量ファイルの転送を要求するとする。今回は、部分的動作の確認であるため、サーバ内のファイルは 1 つである。

4.1 サーバ

図 5 のネットワークポロジに基づいて、サーバの動作状況を示すログファイルを図 6、図 7 に示す。

図 6 は、サーバが XCAST ルーティングデーモンを用いて経路更新を行っている間にクライアント 1 がファイルを要求し、NBT on XCAST が開始されたことを示す。

「00」「01」行は、隣接する XCAST ルータへの経路更新確認とその応答である。「02」行は XCAST パケットを受信したことを表す。「03」行は受信した XCAST パケットのヘッダを解析した結果、その XCAST パケットのクライアント数を示す。まだクライアント数が 1 つであるため、「04」行においてデータを要求したクライアントに対して、XCAST のルールに従ってユニキャストパケットで送信する。以下、同様の処理を繰り返していく。

00	15:47:19 Sending route update to 192.168.0.2
01	15:47:25 Received route update from 192.168.0.2
02	15:47:29 Received XCAST packet from 192.168.0.1
03	15:47:29 Num destinations = 1
04	15:47:29 Unicast packet sent to 192.168.0.3/port 1058
05	15:47:30 Received XCAST packet from 192.168.0.1
06	15:47:30 Num destinations = 1
07	15:47:30 Unicast packet sent to 192.168.0.3/port 1058
08	15:47:31 Received XCAST packet from 192.168.0.1
09	15:47:31 Num destinations = 1
10	15:47:31 Unicast packet sent to 192.168.0.3/port 1058
11	15:47:32 Received XCAST packet from 192.168.0.1
12	15:47:32 Num destinations = 1
13	15:47:32 Unicast packet sent to 192.168.0.3/port 1058

図 6 クライアント 1 の転送要求に対するサーバの動作

次に図 7 は、クライアント 1 にデータを送信している間にクライアント 2 が同一データを要求した場合のサーバの動作状況を示すログファイルである。「00」～「08」行はクライアントが 192.168.0.3 一つの場合を示す。「09」行以降から 2 番目のクライアント 192.168.0.4 が最初のクライアントが受信しているファイルを要求し、XCAST グループに参加してきたことを示す。「10」行は、受信した XCAST パケットのヘッダを解析し、クライアント数が 2 つであることを示す。「11」行は、受信した XCAST パケットのアドレスリストに含まれるクライアントのアドレスを示す。これは、XCAST の特徴の一つである XCAST ヘッダ内

に複数のクライアントアドレスを明示的に示していることを表す。「12」行では、隣接する XCAST 対応ルータに対して XCAST パケットを作成して送信していることがわかる。

00	15:47:38 Received XCAST packet from 192.168.0.1
01	15:47:38 Nua destinations = 1
02	15:47:38 Unicast packet sent to 192.168.0.3/port 1058
03	15:47:39 Received XCAST packet from 192.168.0.1
04	15:47:39 Nua destinations = 1
05	15:47:39 Unicast packet sent to 192.168.0.3/port 1058
06	15:47:40 Received XCAST packet from 192.168.0.1
07	15:47:40 Nua destinations = 1
08	15:47:40 Unicast packet sent to 192.168.0.3/port 1058
09	15:47:40 Received XCAST packet from 192.168.0.1
10	15:47:40 Nua destinations = 2
11	15:47:40 Dests: 192.168.0.3/1058 192.168.0.4/1058
12	15:47:40 XCAST packet sent to 192.168.0.2
13	15:47:41 Received XCAST packet from 192.168.0.1
14	15:47:41 Nua destinations = 2
15	15:47:41 Dests: 192.168.0.3/1058 192.168.0.4/1058
16	15:47:41 XCAST packet sent to 192.168.0.2
17	15:47:41 Received XCAST packet from 192.168.0.1
18	15:47:41 Nua destinations = 2
19	15:47:41 Dests: 192.168.0.3/1058 192.168.0.4/1058
20	15:47:41 XCAST packet sent to 192.168.0.2
21	15:47:42 Received XCAST packet from 192.168.0.1
22	15:47:42 Nua destinations = 2
23	15:47:42 Dests: 192.168.0.3/1058 192.168.0.4/1058
24	15:47:42 XCAST packet sent to 192.168.0.2

図7 クライアント2の転送要求に対するサーバの動作状況の変化

4.2 ルータ

図8は、中継ルータの動作状況を示すログファイルである。

前のルータ（本実験ではサーバ）から XCAST パケットを受信した最終中継ルータは、XCAST ヘッダ内の宛先を解析し、各クライアントに対してそれぞれパケットの複製を行った後、それぞれのクライアントへユニキャストパケットを転送していることが分かる。

5 むすび

本稿では、蓄積型大容量ファイルを複数のクライアントに対し効率的に配信することを目的に、提案した順序非保存ブロック転送方式（NBT: Non-ordered Block Transfer）を紹介した。つぎにその汎用性を拡大するため、ピュアな IP 網への

00	16:14:14 Received route update from 192.168.0.1
01	16:14:20 Sending route update to 192.168.0.1
02	16:14:35 Received XCAST packet from 192.168.0.1
03	16:14:35 Nua destinations = 2
04	16:14:35 Unicast packet sent to 192.168.0.3/port 1058
05	16:14:35 Unicast packet sent to 192.168.0.4/port 1058
06	16:14:36 Received XCAST packet from 192.168.0.1
07	16:14:36 Nua destinations = 2
08	16:14:36 Unicast packet sent to 192.168.0.3/port 1058
09	16:14:36 Unicast packet sent to 192.168.0.4/port 1058
10	16:14:36 Received XCAST packet from 192.168.0.1
11	16:14:36 Nua destinations = 2
12	16:14:36 Unicast packet sent to 192.168.0.3/port 1058
13	16:14:36 Unicast packet sent to 192.168.0.4/port 1058
14	16:14:37 Received XCAST packet from 192.168.0.1
15	16:14:37 Nua destinations = 2
16	16:14:37 Unicast packet sent to 192.168.0.3/port 1058
17	16:14:37 Unicast packet sent to 192.168.0.4/port 1058
18	16:14:37 Received XCAST packet from 192.168.0.1
19	16:14:37 Nua destinations = 2
20	16:14:37 Unicast packet sent to 192.168.0.3/port 1058
21	16:14:37 Unicast packet sent to 192.168.0.4/port 1058

図8 ルータの動作状況

適用を考慮した、Explicit Multicast (XCAST) を用いた順序非保存ブロック転送方式 (NBT on XCAST) の詳細について述べた。また、提案した NBT on XCAST の基本的動作の確認を行った。

今後、パケット損失が発生した場合の再送、複数コンテンツへの対応など、より詳細な実装を行う予定である。

参考文献

- [1] 高取, 伊東, 高崎, "順序非保存ブロック転送方式によるネットワーク利用効率の向上", 情報処理学会第59回全国大会 4U-09 (1999)
- [2] R. Boivie, Y. Imai, W. Livens, D. Ooms, and O. Paradaens, "Explicit Multicast: Basic specification", Internet Draft, draft-ooms-xcast-spec-06.txt, June 2004
- [3] 住田, 伊東, "XCAST を用いた高能率大容量ファイル転送方式の有効性", 電子情報通信学会ソサイエティ大会 B-7-3 (2003)
- [4] 住田, 伊東, "XCAST を用いた高能率大容量ファイル転送方式の実装方式", 電子情報通信学会総合大会 B-7-40 (2004)
- [5] 住田, 伊東, "高能率大容量ファイル転送プロトコル NBT の再送方式", 2004 電子情報通信学会ソサイエティ大会 B-7-4 (2004)