# マルチホームネットワークにおけるアプリケーションプロトコルの性質を考慮した動的トラフィック分散

岡　山　聖　彦†　山　井　成　良††
久　保　武　志†††　宮　下　卓　也††

自組織のネットワークを複数のバックボーンネットワークと接続したマルチホームネットワークを効率良く利用するためには，ネットワークの状態に応じてバックボーンを選択する動的トラフィック分散が重要である．しかし，従来の手法ではすべてのフローに対して同一のバックボーンの選択方法を適用していたためアプリケーションプロトコルの特性を考慮できない，あるいは同一セッション中に依存関係を持つ複数のフローが存在する場合に対処できないなどの問題点があった．そこで，本論文ではアプリケーションプロトコルに応じてバックボーン選択方法を切り替え，また同一セッションに属する複数のフローを同じバックボーンに割り当てる手法を提案する．

# A Dynamic Traffic Balancing on Multihomed Networks Considering Application Protocol Properties

Kiyohiko Okayama,† Nariyoshi Yamai,†† Takeshi Kubo†††
and Takuya Miyashita††

To achieve efficient use of multihomed network, which is the network connected to the Internet with two or more backbones, dynamic traffic balancing, which is a method to select appropriate backbone according to the status of backbones, is important. However, existing methods have two problems: (1) they do not care of characteristics of application protocols bacause they apply the same backbone selection mechanism for all communication flows, (2) they can not correctly handle applications which have two or more communication flows depening each other in the same session. In this paper, we propose a dynamic traffic balanging method, which can select appropriate backbone selection mechanism according to characteristics of application protocols, and can allocate communication flows in the same session to the same backbone.

## 1. はじめに

インターネットの利用は年々増加の一途をたどっており，これにともない WWW などのネットワークアプリケーションにおける応答時間の劣化が顕著になってきている．これに対処する 1 つの方法として，自組織のネットワークを複数のバックボーンネットワーク（以下，単にバックボーンと呼ぶ）と接続し，通信先に応じて利用するバックボーンを使い分けることにより応答時間の改善を図るマルチホームネットワークが最近注目されるようになってきた．

しかし，マルチホームネットワークでトラフィックを分散する場合，従来の経路制御方法ではバックボーンから入手した経路情報と通信先アドレスのみで利用するバックボーンが一意に定まるため，通信先に偏りが生じると効率的なトラフィック分散が行われず，特定のバックボーンにトラフィックが集中する危険性がある．また，一般にマルチホームネットワークでは接続先バックボーンから経路情報を入手できるようにバックボーン管理者と協調して設定作業を行う必要があり，導入や管理にかなりの技術レベルと管理コストが要求される点も問題である．このような理由から，マルチホームネットワークの導入は特に中小規模の組織では困難であった．

この問題に対処するため，我々の研究グループでは，複数のバックボーンと自組織のネットワークとの接続

† 岡山大学工学部
Faculty of Engineering, Okayama University
†† 岡山大学総合情報基盤センター
Information Technology Center, Okayama University
††† 株式会社日立製作所
Hitachi Ltd.

を受け持つルータにおいて，たとえばRTT（Round Trip Time）や利用可能帯域など，各バックボーンの状態を自らが判断して，フロー☆単位で適切なバックボーンを選択する方法を提案してきた[1)~3)]（以下，まとめて従来手法という）．しかし，従来手法では，すべてのフローに対して同一のバックボーン選択方法を適用していたため，たとえばTELNET[4)]などの対話型プロトコルとHTTP[5)]のようなバルク転送型プロトコルが混在して用いられる場合でもアプリケーションプロトコルの特性を考慮することができず，非効率的なトラフィック分散を行ってしまう危険性があった．また，FTP[6)]における制御用コネクションとデータ用コネクションのように，同一セッション中に依存関係を持つ複数のフローが存在する場合，従来手法ではこれらをうまく扱うことができず，アプリケーションが正常に動作しない危険性があった．

そこで，本論文ではアプリケーションプロトコルの性質を考慮して適切なバックボーンを選択できる動的トラフィック分散手法を提案する．これにより，TELNETに対してはRTTが小さいバックボーンを選択し，同時にHTTPに対しては利用可能帯域が広いバックボーンを選択するなど，より効果的なバックボーン選択が可能となる．さらに，FTPなどのように複数のフローを扱うアプリケーションに対しても，同一セッション内のすべてのフロー同じバックボーンに割り当てることにより，セッション単位で動的トラフィック分散を適用することが可能となる．

以下，2章では従来手法とその問題点を考察する．3章では本論文で提案するアプリケーションプロトコルの性質を考慮した動的トラフィック分散手法について述べ，4章では提案手法の有効性を確かめるための実験について述べる．最後に，5章では結論と今後の課題について述べる．

なお，本論文で用いる「動的トラフィック分散」という用語は，通信先が同一であっても，フロー発生時のネットワーク状況に応じて複数のバックボーンを（フロー単位で）動的に選択できることを意味しており，フロー開始後にバックボーンを切り替えるものではないことに注意する．

## 2. 従来の動的トラフィック分散方法とその問題点

マルチホームネットワークは，1つのネットワーク

☆ 5つ組（トランスポート層プロトコル，送信元IPアドレス，送信元ポート番号，送信先IPアドレス，送信先ポート番号）が同一の通信．

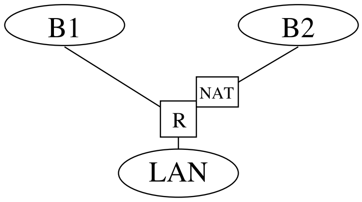



**図1** マルチホームネットワーク
Fig. 1 A multihomed network.

を複数のバックボーンによりインターネットに接続する形態で，トラフィック分散による応答性の改善や耐故障性の向上などを図る方法として注目されている．これまでに知られているマルチホームネットワークの構成方法としては，AS（Autonomous System）番号を取得する方法（方法1）[7)]，NAT（Network Address Translation）[8),9)]を用いる方法（方法2）[10)]，アプリケーションゲートウェイ（ALG: Application Level Gateway）を用いる方法（方法3）[11)]などがあげられる．

これらのうち，方法1および2ではBGP4[12)]を用いて経路情報の交換を行う必要があるので，導入・管理にかなりの技術レベルと管理コストが要求される点が問題となる．さらに，これらの方式では，経路制御は送信先アドレスのみに依存して行われ，現在のトラフィック量などバックボーンの利用状況が反映されないため，通信先に偏りがあった場合に特定バックボーンにトラフィックが集中する可能性も残されている．

一方，方法3では適用できるアプリケーションがWWWなどALGに対応した一部のものに限られ，また，ALGが透過型プロクシとして実現されていない場合は，ユーザがALGの存在を意識する必要がある点が問題となる．

これらの問題を解決するため，我々の研究グループでは独自の経路制御方法に基づいてこれらの問題を解決する動的トラフィック分散方法を提案した．以下ではその概要を説明する．

### 2.1 ネットワーク構成

以下の説明ではインターネットの末端に位置する組織のネットワークを対象とし，**図1**に示すように，自組織のネットワーク（LAN）を1つのルータRにより2つのバックボーン（B1およびB2）に接続した構成をとるものとする．また，自組織のネットワークにはB1から与えられたアドレスが割り当てられており，B2とはNATを経由してアクセスするように設定されているものとする．

このような構成のネットワークにおいて，本方法では内部のクライアントと外部のサーバとの間のフロー

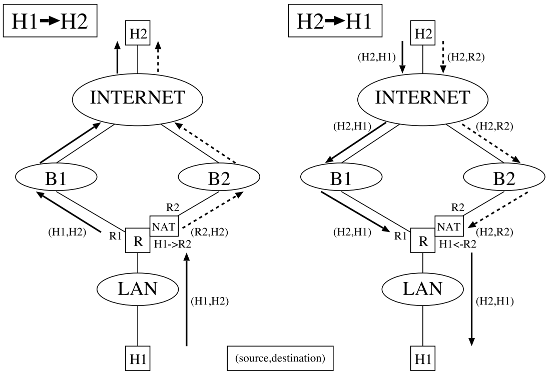



**図 2** 往路および復路でのパケットの流れ
Fig. 2 Routes of incoming and outgoing packets.

を対象とし，新たなフローの検出時にルータが適切なバックボーンを選択する．なお，外部のクライアントと内部のサーバとの間のフローについては，DNS ラウンドロビン[13] などの方法が知られており，本論文では対象としない．

### 2.2 往路・復路でのトラフィック分散

次に，往路と復路それぞれにおけるトラフィック分散について述べる．

往路において，ルータはそれぞれのバックボーンの状態を監視し，内部から外部へ新たなフローに属するパケットが来ると，その時点での各バックボーンの状態から適切なバックボーンを選択する．一度フローを検出すると，同一フローに属するパケットは同一のバックボーンを利用する．

復路のトラフィック分散は，NAT を用いることにより行う．この様子を，**図 2** において H1 が H2 と通信をする場合を例にとり説明する．

まず，ルータが往路で左図の実線矢印で示すように B1 を選択した場合を考える．この場合，ルータ R ではアドレス変換されずに H2 にそのまま届き，復路では H2 は H1 宛にパケットを送り返す．ここで，H1 は B1 から割り当てられたアドレスを用いているため，このパケットは右図の実線矢印で示すように往路と同じ B1 を経由して H1 に届く．

一方，ルータが往路で B2 を選択した場合を考えると，ルータは NAT を用いて通信元アドレスを H1 から R2（B2 から割り当てられたアドレス）に変換するため，復路では H2 は R2 宛にパケットを送り返す．ここで，R2 は B2 から割り当てられたアドレスであるため，このパケットは右図の破線矢印で示すように往路と同じ B2 を経由してルータ R に届き，発信先アドレスが R2 から H1 に変換されて最終的に H1 に届く．

以上のように，NAT を用いることにより往路と復路は同一のバックボーンを経由することになるため，往路でトラフィック分散を行うと復路でも自動的にトラフィック分散が行われることになる．

なお，ルータはバックボーン B2 を経由するフローに対してアドレス変換を行うための表を管理している．これは，NAT におけるアドレス変換表とほぼ同様であるが，3 つ以上のバックボーンに対応するために，アドレス変換表の各エントリに対して，使用するバックボーンの情報（ルータが各バックボーンに接続するネットワークインタフェース名）が追加されている点が異なる．

### 2.3 バックボーン選択方法

我々はこれまでにいくつかのバックボーン選択方法を提案してきた[1)~3)] が，本論文ではそのうち提案手法に関係の深い代表的な方法として，TCP 通信を対象とした，RTT に基づく方法，利用可能帯域に基づく方法，および両者を併用する方法を示す．

RTT に基づく方法[1)]（以下，RTT 方式という）では，ルータは内部ネットワークから SYN フラグ付きパケット（SYN パケット）を受け取るとこれをすべてのバックボーンに対して同時に送出して通信先とのコネクションの確立を試み，このうち最も早く SYN フラグおよび ACK フラグ付きパケット（SYN+ACK パケット）を受け取ったバックボーンを選択して ACK フラグ付きパケット（ACK パケット）を送出し，それ以外のバックボーンには SYN+ACK パケットを受け取るとただちに RST フラグ付きパケット（RST パケット）を送出してコネクションを破棄する（**図 3** 参照）．

一方，利用可能帯域に基づく方法[3)]（以下，帯域方式という）では，基本的には RTT 方式と同じであるが，ルータは内部ネットワークから SYN パケットを受け取ると，すべてのバックボーンに対してそれぞれ 2 つの SYN パケットを連続して送出し，これらに対する SYN+ACK パケットの到着間隔の短いバックボーンを選択する点が異なる．この方法は，利用可能帯域が狭いバックボーンではパケットの送出間隔が大きくなることから，SYN+ACK パケットの到着間隔と利用可能帯域との間には強い相関関係が見られる性質を利用している．

また，両者を併用する方法[3)]（以下，併用方式という）も帯域方式とほぼ同じであるが，SYN+ACK パケットの到着間隔ではなく，2 つ目の SYN パケットに対する応答を早く返したバックボーンを選択する点が異なる．これは 1 つ目の SYN パケットに対する応答時間と SYN+ACK パケットの到着間隔の両方が 2

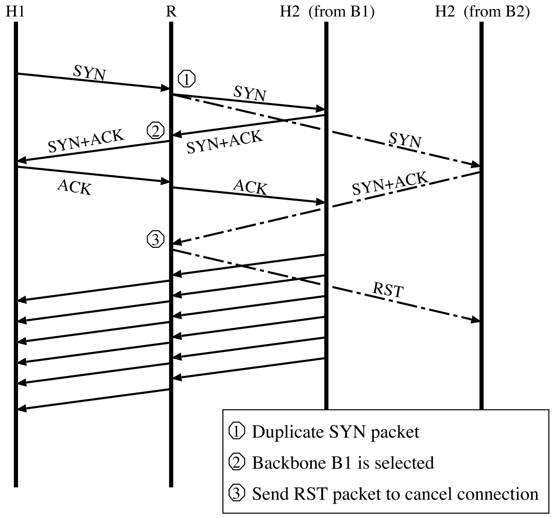



**図 3** バックボーン選択時のパケットの流れ
Fig. 3 Routes in backbone selection.

つ目の SYN パケットに対する応答時間に含まれている点に着目したものである．

なお，UDP 通信に対するバックボーン選択手法についても文献 2) で提案しているが，本論文では触れない．

### 2.4 従来の動的トラフィック分散方法の問題点

前節で述べた従来手法では，同一の通信先であっても，使用するバックボーンをフロー発生時のネットワーク状況に応じて動的に選択することができるが，以下の 2 つの問題がある．

**(1)** 同一のトランスポート層プロトコルを用いるすべてのフローに対して同一のバックボーン選択方法しか適用されないこと

**(2)** 各フローに対して個別にバックボーンが選択されること

（1）は，たとえば TELNET など利用可能帯域よりも RTT が重視される対話型プロトコルと，HTTP など RTT よりも利用可能帯域が重視されるバルク転送型プロトコルが混在して用いられる場合でも，このような特性を考慮することができないことを意味する．このため，対話型プロトコルを用いるフローに対して帯域方式が適用されたり，あるいはバルク転送型プロトコルを用いるフローに対して RTT 方式が適用されたりするなど，非効率的なトラフィック分散を行ってしまう危険性がある．

（2）は，同一セッション中に依存関係を持つ複数のフローが存在する場合に問題となる．たとえば，FTP では 1 つのセッション中で 1 つの制御用コネクションと 1 つ以上のデータ転送用コネクションが用いられるが，従来の動的トラフィック分散方法ではこれらを独立したものとして扱うため，これらのフローに異なるバックボーンが割り当てられてしまうと，正常な通信を行えない危険性がある．

## 3. アプリケーションプロトコルに依存した動的トラフィック分散

2.4 節で述べた問題は，バックボーン選択の際に，アプリケーションプロトコルの特性や，フロー間の依存関係を考慮することにより，解決できると考えられる．

本章では，まず，比較的単純なアプリケーションプロトコル，すなわち，他のフローと依存関係のないフローを持つアプリケーションプロトコルについて，その特性を考慮したトラフィック分散を実現するための方法について述べ，次に，同一セッション内に依存関係のある複数のフローを持つプロトコルを正しく扱うための方法について述べる．

### 3.1 プロトコル特性を考慮したトラフィック分散

アプリケーションプロトコルの特性を考慮してトラフィック分散を行うためには，トラフィック分散を行うルータにおいて，以下の 2 つの機能が必要である．

**(1)** フローが属するアプリケーションプロトコルの種類を判別する機能

**(2)** アプリケーションプロトコルごとにバックボーン選択手法を選択する機能

（1）を実現するための方法としては，TCP および UDP のポート番号を用いる方法のほか，Snort [14] などの不正侵入検知システムやトラフィック制御装置におけるトラフィック解析で用いられる，トラフィックシグネチャを利用する方法が考えられる．前者は，アプリケーションクライアントがサーバにアクセスする際の宛先ポート番号（いわゆる well-known ポート）に注目するものである．ポート番号はパケットヘッダに含まれるため容易に判別できるが，最近増加しつつある P2P アプリケーションなどのように，ポート番号のみではその種類が判別できないようなアプリケーションには対応できないという欠点がある．これに対し，後者はパケットのデータ部分に注目し，そこに現れる特定のパターン（メッセージの文字列など）をデータベースと照合することによって不正侵入を検知するものである．前者に比して複雑な処理を要するものの，これを応用すれば，ポート番号のみでは種類が判別できないようなアプリケーションにも対応できる可能性がある．

しかし，提案手法のベースとなる RTT 方式，帯域方式，併用方式はいずれも，フロー開始時（TCP 通

信であればコネクション確立前）にバックボーンの選択を行うため，フロー開始後の通信内容を解析するような方法は原理上適用できない．また，フロー開始時にいずれかのバックボーンを割り当てておき，その後の通信データ解析に基づいて適切なバックボーンを再選択する方法も考えられるが，その結果バックボーンの切替えが発生するとサーバからみたクライアントのIP アドレスが変化するので，通信が継続できない．したがって，提案手法ではアプリケーションの種類の判別に，パケットに含まれる宛先ポート番号を利用することにする．このとき，P2P などのアプリケーションの多くはファイル共有などのバルク転送を指向しているため，たとえばポート番号でアプリケーションの種類が判別できないフローは一括して帯域方式を適用するようにすれば，実用上大きな問題は生じないと考えられる．

一方，(2) の機能については，アプリケーションの種類を示すポート番号と，それに対して適用するバックボーン選択手法との対応表を導入し，ルータはこの表を参照してバックボーン選択を行うことにより実現する．たとえば，TELNET（ポート番号 23）およびWWW（ポート番号 80）について，前者は RTT 方式，後者は帯域方式を適用すると表に記述したとする．ルータが新しいフローに属するパケットを受信すると，その宛先ポートに基づいて表を検索し，ポート番号が23 の場合は RTT 方式，80 の場合は帯域方式という結果を得る．あとは，従来のバックボーン選択手法を用いてバックボーンを選択すれば，アプリケーションプロトコルの特性を考慮した効果的なトラフィック分散が期待できる．

なお，ポート番号とバックボーン選択手法の対応表は，ルータの管理者が用意する必要がある．しかし，基本的に，アプリケーションの特性は組織によって変わるものではないため，対応表をインターネット上で公開するなどして共有すれば，管理者は対応表を新たに作成する必要はない．また，対応表に明記されていないポート番号を使用するアプリケーションについては，デフォルトで使用するバックボーン選択手法をあらかじめ設定することにより，すべてのアプリケーションに対して動的トラヒック分散を行うことができる．このとき，アプリケーションによっては適切なバックボーン選択手法が適用されない可能性もあるが，一般に，RTT と利用可能帯域にはある程度の相関関係があるので，動的トラヒック分散を行わない場合に比べれば，効率的な通信が実現できると考えられる．

### 3.2 依存関係を持つフローを考慮したトラフィック分散

FTP やストリーム系のプロトコルなど，一部のアプリケーションプロトコルでは，制御用フローとデータ用フローなど依存関係を持つ複数のフローが同一セッション中に存在する．

従来手法をこのようなプロトコルにそのまま適用すると，各フローに対して独立にバックボーン選択を行うので，フローごとに異なるバックボーンが割り当てられる可能性がある．これは，図 2 の H2 からみると，フローごとに H1 の IP アドレスが異なることを意味するため，H2 はこれらのフローの通信相手が同一の計算機であるとは判断できず，正常に動作しない可能性が大きい．また，このようなプロトコルでは，2 本目以降のフローはインバウンドフロー（図 2 におけるH2 側から H1 側に向けて発生するフロー）であることが多く，このフローがバックボーン B2 を経由すると，ルータにはアドレス変換に関する情報がないため破棄されることになる．

この問題に対処するためには，ルータがアプリケーションプロトコルを解析して，複数のフローを同一のバックボーンに割り当てたり，インバウンドフロー用にアドレス変換表を作成したりする必要がある．しかし，その解析方法はアプリケーションプロトコルによって異なるので，本論文では，代表的なアプリケーションとして，ファイル転送のためのアプリケーションである FTP と，ストリーミング通信のためのアプリケーションである RealPlayer8 を扱う．

なお，以下では，図 2 の環境においてクライアントH1 とサーバ H2 との間で通信するものとする．

#### 3.2.1 FTP の場合

FTP では，制御コネクションとデータコネクションの 2 種類のフローが存在し，さらに，データコネクションはその確立方法によって，アクティブモード（インバウンドフロー）とパッシブモード（アウトバンドフロー）に分類される．

H1 が H2 に対して FTP セッションを開始する際，制御コネクションについてはバックボーン B2 が選択されたものとする．すると，ルータのアドレス変換表には**表 1** の上段に示すようなエントリが作成され，制御コネクションのパケットはアドレス変換処理を受けながら B2 を経由して H1〜H2 間を流れることになる．ここまでの手順は従来手法と同様であるが，提案手法ではさらに，ルータが制御コネクションの通信内容を監視して，アクティブモード時の PORT コマンドあるいはパッシブモード時の PASV コマンドを検

**表 1** FTP 用のアドレス変換ルール
Table 1 Address translation rules for FTP.

| source | alias | dest | proto | I/F |
|---|---|---|---|---|
| H1:1234 | R2:1234 | H2:21 | TCP | B2 |
| H1:h1 | R2:r2 | H2:20 | TCP | B2 |

**表 2** RealPlayer8 用のアドレス変換ルール
Table 2 Address translation rules for RealPlayer8.

| source | alias | dest | proto | I/F |
|---|---|---|---|---|
| H1:1234 | R2:1234 | H2:554 | TCP | B2 |
| H1:h1 | R2:r2 | H2:h2 | UDP | B2 |

出し，それぞれの場合に応じた処理を行う．

まず，H1 がデータコネクションの受信ポートを H2 に通知する PORT コマンドを検出すると，これに含まれる H1 の受信ポート番号（h1）と，その時点でのルータの空きポート番号（r2）を用いて，表 1 の下段に示すようなエントリを追加するとともに，PORT コマンドに含まれる IP アドレスとポート番号をそれぞれ R2 および r2 に書き換えて H2 に送信する．その後 H2 から開始されるデータコネクションのパケットは追加されたエントリに基づいてアドレス変換が行われるので，ルータで破棄されることなく，しかも，制御コネクションと同じバックボーンを経由してデータコネクションも確立される．

一方，H1 が H2 に対して送信する PASV コマンドを検出した場合，H2 はデータコネクションを受信するための IP アドレスおよびポート番号を返すので，ルータはこれを記録し，その後 H1 から H2 へのデータコネクション確立用の SYN パケットを受信した際に，制御コネクションと同じバックボーンを選択すればよい．

#### 3.2.2 RealPlayer8 の場合

RealPlayer8 によるストリーミング通信の場合，H1 は RTSP（Real Time Streaming Protocol）[15] を用いて H2 との間に制御コネクションを確立する．H1 と H2 はこれを利用してコンテンツ名などの情報を交換し，その後，H2 は H1 からの再生コマンドによってコンテンツ配信（データフロー）を開始するが，交換される情報には，データフローのために用いるトランスポート層プロトコルや，H1 の受信ポート，H2 の送信ポートが含まれる．したがって，ルータでは FTP の場合と同様に，制御コネクションの監視により得られたトランスポート層プロトコルと送信および受信ポート番号に基づき，制御コネクションと同じバックボーンを経由するようにデータフローのための変換ルールをアドレス変換テーブルに追加すれば，H2 から送信されるデータフローのパケットはルータで破棄されることなく H1 に中継される．

たとえば，制御コネクションが B2 を経由している場合，ルータのアドレス変換表には**表 2** 上段のようなエントリが追加されている．そして，H1 が UDP のポート番号 h1 でデータフローを受信することを H2 に通知し，H2 はポート番号 h2 で送信することを H1 に通知したとする．ルータはこれらの情報が含まれるメッセージを検出すると，メッセージ中のポート h1 をルータでの空き番号である r2 に書き換えて B2 に送出すると同時に，アドレス変換表に表 2 下段のようなデータフローのためのエントリを追加する．

一方，H2 は制御コネクションがルータ（R2）との間で確立されているように見えるので，再生コマンドを受信すると，送信ポート番号 h2 を用いて R2 のポート番号 r2 に向けてデータフローを開始する．ルータにはデータフローを変換するためのエントリがすでに追加されているので，これに従って H1 に転送することにより，正常に通信を行うことができる．

なお，FTP や RealPlayer8 以外のアプリケーションプロトコルでも同様の処理は可能と思われるが，仮にこれが不可能であれば，一部のよく利用されるアプリケーションプロトコルのみをトラフィック分散の対象とし，それ以外は B1 を選択するようにすればよい．

## 4. 実装と評価

### 4.1 提案手法の実装

提案手法の有効性を検証するため，3 章で述べた動的トラフィック分散機能をルータ（FreeBSD4.1 搭載機，CPU: Celeron566 MHz）に実装し，アプリケーションの特性に応じた適切なバックボーン選択が行われていることを確認するための実験を行った．

従来手法では，デバッグの容易さなどを考慮して，ルータにおける動的トラフィック分散機能をユーザ空間で動作するプログラム（dlbd）として実装している．ルータ内部の構成は**図 4** に示すようになっており，各ネットワークインタフェース（NIC）から到着するすべてのパケットを，FreeBSD の divert 機能を利用してカーネルが経路制御を行う前に dlbd に渡す．そして，dlbd ではあらかじめ指定された方式（RTT/帯域/併用）に基づいて送出先となるネットワークインタフェースを決定し，bpf（Berkeley Packet Filter）を用いてそのネットワークインタフェースにパケットを直接書き出すことにより，カーネルの経路制御に依存しないトラフィック分散を実現している．したがって，ユーザ空間のプログラムがネットワークインタフェースを直接的に操作できる機能を備えた OS であ

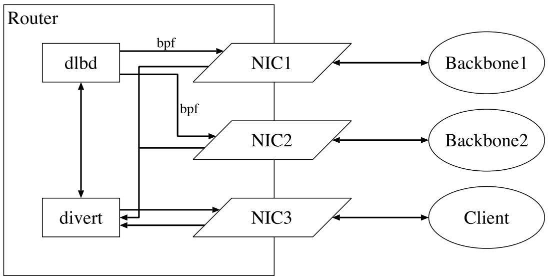



**図 4**　ルータ内部の構成
Fig. 4　The internal structure of the router.

れば，比較的容易に実装できると考えられる．

本研究では，従来手法の dlbd に対して，3.1 節で述べたポート番号とバックボーン選択手法の対応表を追加するとともに，この対応表の検索結果に応じて RTT 方式，帯域方式，併用方式のいずれかを適用する機能を追加した．さらに，FTP（ポート番号 20）および RealPlayer8（ポート番号 554）の制御コネクションを検知した場合には，3.2 節で述べたように，データフローを制御コネクションと同一のバックボーンに割り当てる機能も追加した．

なお，文献 2) の帯域方式においては，帯域を測定するために送出する連続した 2 つの SYN パケットの大きさが問題になることが判明した．本来であれば，帯域の狭いバックボーンを 2 つの SYN パケットが通過した際には，通信先からの 2 つの SYN+ACK パケットがルータに到着する間隔が大きくなるので，帯域が狭いということを判別できる．しかし，SYN パケットは IP ヘッダおよび TCP ヘッダのみから構成され，合計 40 バイトであるので，この程度の小さなパケットでは，極端に帯域の狭いバックボーンでない限り到着間隔が大きくならず，他のバックボーンと比較した結果帯域の狭いバックボーンを誤って選択してしまう可能性がある．

そこで，帯域方式選択時にルータが送出する SYN パケットに対して IP の No Operation オプションを複数個挿入し，SYN パケットを意図的に大きくして帯域を測定するように変更した．具体的には，フラグメンテーションが発生して SYN パケットが分割されないよう，TCP の最大セグメントサイズの保証値である 512 バイトに設定した．このような変更を施して数回実験を行ったところ，誤って帯域の狭いバックボーンを選択する割合が減少した．なお，この問題は帯域方式特有のものであり，RTT 方式では各バックボーンの RTT 自体を比較するのみであるので，このような変更を行う必要はない．

**表 3**　実験におけるバックボーン特性
Table 3　Characteristic of each backbone in the experiment.

| B1 | | B2 | |
|---|---|---|---|
| bandwidth | delay | bandwidth | delay |
| 100 Kbit/s | 0 ms | 1 Mbit/s | 500 ms |

### 4.2　実験と考察

提案手法の評価方法については，インターネットのような実環境における実験と，外部から隔離された実験ネットワークにおける実験が考えられる．前者は実環境における提案手法の有効性を直接的に評価できるという利点があるが，正確な評価を行うためには，ルータから組織外のサーバに至るすべての経路において，経路上にある各ルータの性能やリンク特性をはじめ，実験時には実験以外のトラフィックを正確に把握する必要があるため，非常に困難である．これらの情報を正確に把握できない場合には，長期的かつ継続的に実験を実施して，提案手法を用いる場合と用いない場合の，各アプリケーションごとのスループットや応答時間などの差を検証する必要がある．このため，実環境における評価は今後の課題とし，本研究では外部から隔離された実験ネットワークを構築して実験を行った．

実験環境は図 2 とほぼ同様の構成であり，各バックボーンの特性は FreeBSD の dummynet 機能を用いて，**表 3** に示すように，B1 は相対的に狭帯域かつ遅延小，B2 は広帯域かつ遅延大となるように設定した．この実験環境において，TELNET，WWW，RealPlayer8，FTP の 4 種類のアプリケーションを同時に起動し，各アプリケーションプロトコルごとのバックボーン選択率を測定した．さらに，比較の対象として，従来のバックボーン選択手法（RTT 方式，帯域方式，併用方式）を単独で適用した場合についても同様に測定を行った．なお，RealPlayer8 は動画フローに TCP と UDP のいずれかを選択できるが，本実験では TCP と UDP のフローを混在させるため，UDP を用いるように設定した．

また，実験で使用した各アプリケーションは，**表 4** のように設定した．TELNET は対話型のアプリケーションであるため RTT 方式，FTP および RealPlayer8 は制御用とデータ用の 2 つのフローを持つが，データ送信時のスループットを重視して帯域方式を選択するようにした．TELNET および FTP の制御コネクションは対話的に操作を行う必要があるので，expect [16] を使用することにより，アプリケーションの実行を自動化している．また，WWW については，テキストファ

**表 4** 実験における各アプリケーションの設定
Table 4 Configuration of each application in the experiment.

| アプリケーション | 同時起動数 | データ量 (KB) | 選択手法 |
|---|---|---|---|
| TELNET | 1 | – | RTT |
| WWW | 2 | 800, 200, 12 | 併用 |
| RealPlayer8 | 3 | 250 | 帯域 |
| FTP (Active) | 1 | 800 | 帯域 |
| FTP (Passive) | 1 | 800 | 帯域 |

イルや画像ファイルなど，大きさの異なるコンテンツが存在することを考慮して，WWW サーバに 3 種類の大きさのファイルを用意した．このとき，テキストファイルなどの小さなファイルにアクセスする際は応答時間を優先し，画像などの大きなファイルにアクセスする際は転送時間を優先するため，WWW については併用方式を選択するようにした．なお，各アプリケーションの同時起動数が多過ぎると，両方のバックボーンで輻輳が発生してしまい，有意な結果が得られない．そこで予備実験を数回行い，過度の輻輳が発生しないように同時起動数を決定した．

実験の結果として，それぞれの場合におけるバックボーン B1 の選択率を**図 5** に示す．従来手法では，アプリケーションの特性を無視してバックボーン選択を行うので，RTT，帯域，併用方式のいずれにおいても，広帯域である B2 の選択率が大きくなっている．RTT 方式においても B1 の選択率が低いのは，実験開始時は遅延の小さい B1 を選択するが，次第にフローが集中して輻輳が発生し，その結果として B1 の RTT が B2 よりも大きくなったためと考えられる．これに対し，提案手法では，TELNET に対しては低遅延の B1 を，他のプロトコルに対しては広帯域の B2 を優先して選択しており，アプリケーションの特性に応じて適切なバックボーンを選択できていることが確認できる．なお，FTP および RealPlayer8 については，実験時の dlbd のログデータを解析することにより，制御コネクション開始時に選択されたバックボーンをデータフローにも割り当てていることを確認している．

一方，インターネットのような実環境では，ルータの経路キャッシュ機能（同じ宛先の 2 番目以降の IP パケットは経路表の参照を行うことなく高速に転送する機能）などがバックボーン選択に与える影響を考慮する必要がある．たとえば，一方のバックボーンを経由する経路上のルータ群にはキャッシュがあり，もう一方のルータ群にはない場合，RTT 方式適用時に誤って遅延の大きいバックボーンを選択したり，帯域方式適用時に利用可能帯域の狭いバックボーンを選択する可能性が生じるが，実環境において経路上のすべての

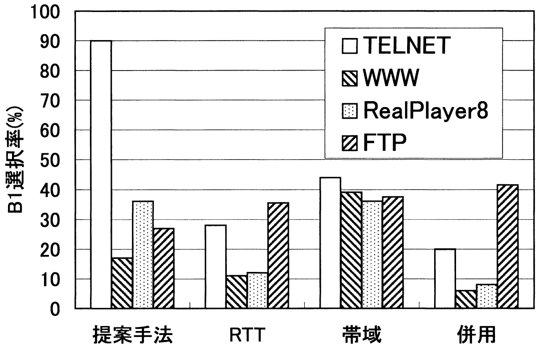


**図 5** バックボーン B1 の選択率
Fig. 5 The rate of selection for B1.

**表 5** 誤選択率を変化させた場合のバックボーン B1 の選択率
Table 5 The rate of selection for B1 with selection mistake.

| 誤選択率（%） | | 0 | 10 | 20 | 30 | 40 |
|---|---|---|---|---|---|---|
| B1 の選択率（%） | TELNET | 90 | 70 | 65 | 65 | 50 |
| | WWW | 17 | 30 | 32 | 39 | 46 |
| | RealPlayer8 | 36 | 43 | 45 | 51 | 50 |
| | FTP | 27 | 31 | 43 | 46 | 55 |

ルータの状態を把握することは非常に困難である．ただし，ルータの経路キャッシュが提案手法に与える影響はバックボーン選択時のみであり，バックボーン選択後の通信時には影響しない（バックボーン選択時のパケットによって各ルータにキャッシュが作成される）と考えられるので，経路キャッシュなどの要因を間接的に評価する手段として，バックボーン誤選択の割合が提案手法のバックボーン選択率に与える影響を測定した．

実験環境やバックボーン特性，各アプリケーションの設定は前出の実験と同様であり，提案手法の dlbd に対して，指定した割合でバックボーンを誤選択，すなわち，各バックボーン選択手法を適用した結果とは異なるバックボーンを選択するように変更し，誤選択率を 0%から 40%まで 10%刻みで変化させた場合のそれぞれについて実験を行った．

この実験におけるバックボーン B1 の選択率を**表 5** に示す．多少の変動はあるものの，誤選択率の増加に従って，TELNET は B1 の選択率が減少し，それ以外のアプリケーションは B1 の選択率が増加していることが分かる．特に，誤選択率が 40%になると，すべてのアプリケーションにおける B1 の選択率が 50%に近くなり，2 つのバックボーンを交互に選択する場合と変わらなくなる．したがって，今回の実験結果を見る限りでは，誤選択率が 30%程度までであれば，提案方式による動的トラヒック分散の効果が得られるとい

える．

## 5. おわりに

本論文では，依存関係にある複数のフローを扱うことができ，さらにアプリケーションプロトコルの性質を考慮するような動的トラフィック分散手法を提案し，その有効性を確認した．今後の課題として，インターネット上での実験や，提案方式のルータに関する詳細な性能評価（ルータが同時に維持可能なセッション数やスループットなど）があげられる．

## 参考文献

1) 岡山聖彦，山井成良，島本裕志，宮下卓也，岡本卓爾：マルチホームネットワークにおける透過的な動的トラヒック分散，情報処理学会論文誌，Vol.41, No.12, pp.3255–3264 (2000).
2) 岡山聖彦，山井成良，宮下卓也：マルチホームネットワークにおける帯域を考慮したバックボーン選択手法，情報処理学会研究報告，2002-DSM-27, pp.1–6 (2002).
3) 山井成良，久保武志，岡山聖彦，宮下卓也：マルチホームネットワークにおける UDP 通信の透過的トラヒック分散，情報処理学会分散システム/インターネット運用技術シンポジウム 2002 論文集，pp.33–38 (2002).
4) Postel, J. and Reynolds, J.K.: Telnet Protocol specifications, RFC854 (1983).
5) Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P. and Berners-Lee, T.: Hypertext Transfer Protocol — HTTP/1.1, RFC2616 (1999).
6) Postel, J. and Reynolds, J.K.: File Transfer Protocol, RFC959 (1985).
7) Hawkinson, J. and Bates, T.: Guidelines for creation, selection and registration of an Autonomous System (AS), RFC1930 (1996).
8) Egevang, K. and Francis, P.: The IP Network Address Translator (NAT), RFC1631 (1994).
9) Srisuresh, P. and Holdrege, M.: The IP Network Address Translator (NAT) Terminology and Considerations, RFC2663 (1999).
10) 梶田将司，結縁祥治：NAT によるプライベートネットワークの準マルチホーム化技法，情報処理学会分散システム/インターネット運用技術研究報告，No.16, pp.73–78 (1999).
11) 中川郁夫，上谷 一，鍋島公章，樋地正浩，今野幸典：マルチホーム環境におけるアプリケーションルーティング技術の提案，情報処理学会分散システム運用技術研究報告，No.12, pp.37–42 (1998).
12) Rekhter, Y. and Li, T.: A Border Gateway Protocol 4, RFC1771 (1995).
13) Brisco, T.: DNS Support for Load Balancing, RFC1794 (1995).
14) Caswell, B. and Roesch, M.: Snort — The Open Source Network Intrusion Detection System. http://www.snort.org/
15) Schulzrinne, H., Rao, A. and Lanphier, R.: Real Time Streaming Protocol (RTSP), RFC 2326 (1998).
16) Chaffee, G. and Savoye, R.: Expect — programmed dialoge with interactive programs, Version 5. http://expect.nist.gov/

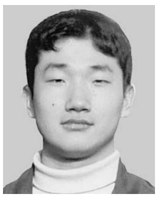
(平成 16 年 7 月 5 日受付)
(平成 17 年 2 月 1 日採録)

**岡山 聖彦**（正会員）

平成 2 年大阪大学基礎工学部情報工学科卒業．平成 4 年同大学大学院基礎工学研究科博士前期課程修了．同年同大学院基礎工学研究科博士後期課程を退学し，同大学工学部助手．平成 6 年奈良先端科学技術大学院大学情報科学研究科助手．平成 10 年岡山大学工学部助手．博士（工学）．インタネットアーキテクチャ，ネットワーク管理，ネットワークセキュリティの研究に従事．電子情報通信学会会員．

**山井 成良**（正会員）

昭和 59 年大阪大学工学部電子工学科卒業．昭和 61 年同大学大学院博士前期課程修了．昭和 63 年同大学院基礎工学研究科（物理系専攻情報工学分野）博士後期課程退学．同年奈良工業高等専門学校情報工学科助手．同講師，大阪大学情報処理教育センター助手，同大学大型計算機センター講師，岡山大学総合情報処理センター助教授を経て，現在岡山大学総合情報基盤センター助教授．分散システム，マルチメディアシステム，マルチメディアネットワークの研究に従事．IEEE，電子情報通信学会各会員．博士（工学）．



**久保 武志**

平成 13 年岡山大学工学部情報工学科卒業．平成 15 年同大学大学院自然科学研究科博士前期課程修了．同年株式会社日立製作所入社．広域分散システム，マルチメディアシステム，高速ネットワーク等に興味を持つ．

**宮下　卓也**

平成 3 年岡山大学工学部電気電子工学科卒業．平成 5 年同大学大学院工学研究科（電気電子工学専攻）修了．平成 8 年同大学院自然科学研究科（知能開発科学専攻）修了．平成 9 年東京農工大学ベンチャービジネスラボラトリー博士研究員．平成 10 年岡山大学総合情報処理センター助手．平成 16 年岡山大学総合情報基盤センター助手．主にディジタル機器からの放射電磁雑音の計算機シミュレーションの研究に従事．情報処理教育，マルチメディア，高速ネットワーク等に興味を持つ．博士（工学）．IEEE，電子情報通信学会，エレクトロニクス実装学会各会員．