

動的最適化多重経路通信機構による適応的ルーティングの提案

越智大介，北形元，菅沼拓夫，木下哲男，白鳥則郎

東北大学電気通信研究所/情報科学研究科

本論文では，ネットワーク上の通信において，多重経路選択アルゴリズムにより複数の経路を生成し，流量調整機構によりそれらの経路を適応的に使用する，動的最適化多重経路通信機構（DO-MRT 機構）を提案する．従来の IP 通信においては，各通信ノードが保持する半固定的なルーティングテーブルをもとに単一経路を選択し，他に複数の転送効率の良い経路がある場合でも 1 つの最短経路にトラフィックが集中する傾向がある．本論文で提案する機構を用いることで，それらの問題点を解決し，ネットワークのトラフィック負荷を均等に分散させることが可能となる．本稿では，DO-MRT 機構の主要機能要素である，多重経路選択アルゴリズムと流量調整機構の設計と実装について述べる．

DO-MRT: Proposal of Adaptive Routing

Using Dynamic Optimizing Multiple Route Transport Mechanism

Daisuke Ochi, Gen Kitagata, Takuo Suganuma, Tetsuo Kinoshita and Norio Shiratori

Research Institute of Electrical Communication / Graduate School of Information Sciences

Tohoku University

In this paper, we propose a mechanism of adaptive routing, Dynamic Optimizing Multiple Route Transport Mechanism (DO-MRT). This mechanism consists of multiple route selection algorithm and adaptive flow control mechanism to optimize multiple flows selected. In traditional IP routing, each node has semi-fixed routing table, and they configure single route from source to destination based on the entries of each routing table. This causes a problem of traffic centralization to a specific single route. By using proposed mechanism, network traffic is decentralized, and effective use of network is realized. In this paper, we focus on the design and implementation of multiple route selection algorithm and adaptive flow control mechanism.

1. はじめに

インターネットの普及およびその規模の拡大に伴い、通信パケットの効率的なルーティングが重要な課題となっている。現在の IP 通信においては、各通信ノードが保持するルーティングテーブルに基づき経路が選択される。その経路は半固定的であり、かつ送信元から送信先まで単一の経路を用いている。このため他に複数の転送効率の良い経路がある場合でも 1 つの最短経路にトラフィックが集中する傾向があり、ネットワーク全体で見たときの転送効率が悪い場合がある。本論文では、これらの問題を解決するための手法として動的最適化多重経路通信機構による適応的ルーティングを提案する。本機構の特徴は以下の 3 点である。

- (1) 複数の経路にトラフィックを分割して送信する、多重経路通信を行う。
- (2) 多重経路は、通信要求が発生した時に動的に選択、決定される。
- (3) 通信時の負荷状況に応じて、複数経路に送出するトラフィックの流量を調整する。

本論文では、動的最適化多重経路通信機構の主要機能要素である、多重経路選択アルゴリズムと流量調整機構の設計と実装について述べる。

本論文ではまず 2 章において我々の着目する問題点を述べたあと、3 章において本機構の提案を行う。4 章で実装に関して述べ、5 章でプロトタイプシステムを用いて行った実験について述べる。最後に 6 章でまとめる。

2. 現状の IP ルーティングの問題点

現在の IP 通信では、各ノードの保持するルーティングテーブルに基づき単一の経路が選択される。ルーティングテーブルは、RIP や OSPF などを用いてベクトル距離やリンク状態が最適になるよう設定されることが多いが、しかしその経路は半固定的であり、かつ単一であるため、次のような問題点が発生している。

- (P1) 他に転送効率の良い経路があった場合でもそれを利用することができない。
- (P2) 各経路の利用状況に応じて流量を調整できず、トラフィック変動に対処できない。

以上の問題により、ネットワーク上のある一部の部分ネットワークにトラフィックの負荷が偏り、それによってネットワーク全体の転送効率が悪化させられている場合がある。

(P1)を解決するためには、より転送効率の良い経路を含められるよう複数の経路を使用し、通信パケットを分配してトラフィックを負荷分散することが有効である。また、複数の経路を使用することで、流量の調整に関する柔軟性・適応性が高まり、(P2)の解決に寄与することが期待できる。

従来の研究と比較すると、例えば OSPF など一部のプロトコルでは、同一のホップカウント数であるときのみ複数の経路を使用して負荷分散を行っているが、それは限定的であり、常に複数の経路で負荷分散を動的に行うことはない。過去にも複数の経路を生成してトラフィックの負荷分散を図る研究はなされてきたが、代替経路として使用できる経路は多くなく、ランダムに代替経路が生成されるのみであるため、効率の良い経路が生成されるとは限らなかった。また、ネットワークトラフィック全体の負荷分散を考える上で効果的ではなかった[1]。また別の研究では、効率的な経路を複数用いて負荷分散を可能としているが、ルーティングテーブルにすべてのノード間の経路を多重化して常時保持するため、ネットワーク規模が大きくなるとルーティングテーブルが破綻する可能性がある[2]。

本論文では、以上の問題点を考慮し、動的に複数の経路を生成し、それらの流量を調整する枠組みとして、動的最適化多重経路通信機構(以下、DO-MRT 機構 : Dynamic Optimizing Multiple Route Transport Mechanism)を提案する。本機構では、ルーティングテーブルを常時保持せずに、実行時に利用可能なあらゆる経路を生成し、それらの間でトラフィックを分散させることで効果的なルーティングを実現する。これにより(P1)を解決する。また、各経路の利用頻度を経路の転送効率によって自律的に調整する。これにより(P2)を解決する。

次章より DO-MRT 機構について述べる。

3. DO-MRT 機構

DO-MRT 機構の主要部分は大きく分けて 2 つの要素から構成される。1 つは多重経路選択アルゴリズム

△ (以下, MRS アルゴリズム : Multiple Route Selection Algorithm) であり, 利用可能な複数の経路を生成するアルゴリズムである。もう1つは**流量調整機構** (以下, FC 機構 : Flow Control Mechanism) であり, MRS アルゴリズムで生成された各経路へ適応的にネットワークトラフィックを分配し, 負荷分散を行う機構である。これら2つの構成要素を中心に, DO-MRT 機構は適応的なルーティングおよびデータ転送を実現する。

次節より, MRS アルゴリズム, 経路情報の最適化, および FC 機構について述べる。

3.1. MRS アルゴリズム

3.1.1. MRS アルゴリズムの概要

DO-MRT 機構で求められる多重経路通信を実現するためには, 通信要求が発生した時に動的に多重経路を生成するためのアルゴリズムが必要である。ここでは, 本論文で提案する MRS アルゴリズムについて述べる。MRS アルゴリズムの目的は, 不要な経路をすべて取り除き, 利用可能な経路を抽出することである。不要な経路を許可するならば, 最も簡単な多重経路通信の方法は, あるノードにおけるすべての隣接ノードのうち一つをランダムに次段の経路として選択し, また次段のノードも同じようにすべての隣接ノードからランダムに次々段の経路を選択し, これをすべてのノードが同様にを行いパケットを送信することである。ただし, ここでは簡略化のために, 各ノードはパケットを受け取ったノードへはパケットを返さないものとする。しかし, この方法では袋小路や複数ノード間でのループが発生し (図1), これにより多数のパケットが不達となる可能性がある。すなわち, 「袋小路」および「ループ」が不要な経路に相当する。また, これら不要な経路を排除できれば, 送出されたパケットは送信先へ届くことが保証される。

MRS アルゴリズムは, 袋小路やループとなる不要な経路を排除し, それ以外の経路群を抽出するアルゴリズムである。経路群を抽出する過程では, 既存の IP におけるルーティングテーブルを積極的に利用する。MRS アルゴリズムの動作手順を以下より説明する。

送信元 Source から送信先 Destination への通信セッ

ションが開始されるとする。まず, 初期条件として Source を親ノード P とする。 P では, P に隣接する子ノード群 $C = \{C_1, \dots, C_n\}$ ($n = P$ の隣接ノード数) を調査するため, 以下に説明する CheckChildren 部を実行する。MRS アルゴリズムでは, あるノードが CheckChildren 部の実行中に隣接ノードへ CheckChildren 部実行の依頼メッセージを送る。それに従って, Source から始まり, 各ノードは再帰的に CheckChildren 部を実行する。ここで, あるノードが CheckChildren 部を実行するまでに経由した一連のノード群を祖先ノード群 $A = \{A_1, \dots, A_m\}$ ($m =$ 祖先ノード数) と呼ぶ。なお, $A_1 = \text{Source}$ となる。

Source が CheckChildren 部の実行を終了すると, MRS アルゴリズムは完了する。

以下に CheckChildren 部について述べる。

3.1.2. CheckChildren 部

CheckChildren 部のフローチャートを図2に示す。親ノードは Source から始まる一連の祖先ノード群 A を知っており, CheckChildren 部ではまず A に親ノード P 自身を追加する。

$$A = \{A_1, \dots, A_m, P\}$$

次に親ノードに隣接する子ノードをすべて抽出し子ノード群 C を作成する。また, それぞれの子ノード C_x ($1 \leq x \leq n$) の標準経路 $\text{Path}(C_x)$ を調べる。ここで, ルーティングテーブルによる最小ホップ数の観点から見た, x から Destination への次段ノードを「 x の標準経路」と呼ぶこととし, $\text{Path}(x)$ と表す。これは既存の IP におけるルーティングテーブルを参照して求める。

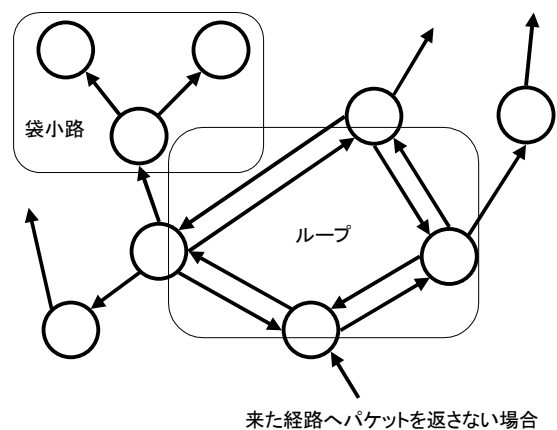
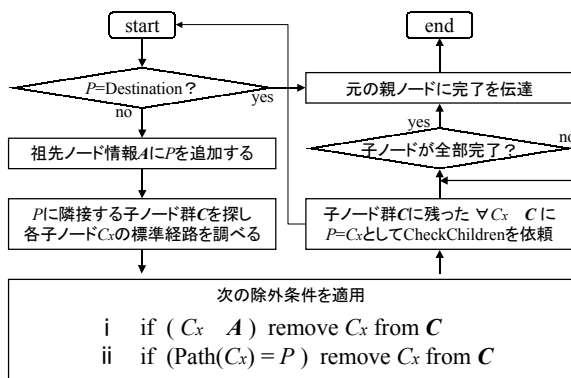


図1 最も簡単な多重経路通信の例



ルーティングテーブルによる最小ホップ数の観点から見た x から Destination への次段ノードを「 x の標準経路」と呼ぶ

図2 MRS アルゴリズムの CheckChildren 部

次におのおの子ノード C_x について、次の2つの除外条件を適用する。

祖先ノード群 A に属する子ノード C_x は、子ノード群 C から除外

if ($C_x \in A$) remove C_x from C

子ノード C_x の標準経路 $Path(C_x)$ に親ノード P が指定される場合、 C_x を子ノード群 C から除外

if ($Path(C_x) = P$) remove C_x from C

除外条件により、ループになる経路を除外できる。また、袋小路となる場合、子ノードの標準経路 $Path(C_x)$ には親ノード P が指定されるため、除外条件により袋小路になる経路が除外できる。

以上の除外条件を子ノード群 C 内のすべての子ノードに対して適用すると、ある Source と Destination が与えられたときに親ノード P が利用できる次段経路が特定できる。

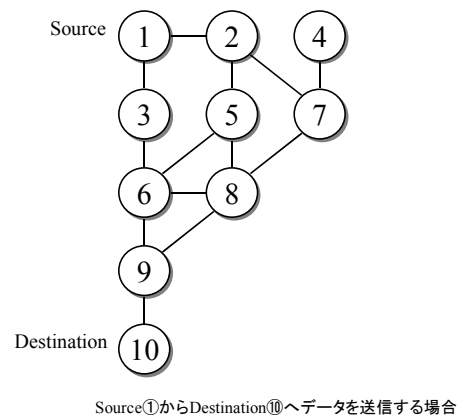
子ノード群 C に子ノードが残った場合は、祖先ノード情報 A を渡し、その子ノードを親として再帰的に CheckChildren 部の実行を依頼する。

if ($C \neq \phi$) $C_x \in C$,

CheckChildren(Source, Destination, A) to C_x

この依頼は、Destination 自身が親ノードとして CheckChildren 部を開始するところまで続けられる。元の親ノード P は、子ノードが全部調査を完了するのを待ち、自分の親ノード A_m へ自分の完了を告げ、CheckChildren 部を完了する。

以上が CheckChildren 部である。CheckChildren 部



Source①からDestination⑩へデータを送信する場合

図3 ネットワーク構成の例

の完了は再帰的にその親ノードへ反映され、最終的に Source (=A1) が CheckChildren 部を完了したところで MRS アルゴリズムは終了する。その時点において、各ノードはある Source から Destination への使用可能な次段経路を保持している。ここで作成した経路を保持するテーブルを多重経路表と呼ぶ。

以上が MRS アルゴリズムの流れである。簡単のために、図3を例にして MRS アルゴリズムの動作を追う。図3は Source を、Destination をとしたネットワーク構成図であり、までのホップ数が少ないノードほど下へ配置されている。このような描き方をすることで、各ノードは同段もしくは上下1段以内のノードと隣接し、標準経路として自分より下段の隣接ノードのうちいずれかを指定することが分かる。例えば、またはを標準経路として指定する。

まずを親ノード P として開始する。の子ノードとしてとがあるので、が除外条件に当てはまるかどうかをチェックする。祖先ノード情報が $A = \{ \}$ なので、共に条件に当たらない。また、 $Path() =$ 、 $Path() =$ であるから共に条件にも当たらない。すなわち Destination がの場合、は $C = \{ , \}$ を子ノードとして持つことができ、またパケットをまで届けるためにはかへパケットを渡せば良いことがわかる。そしては、に祖先ノード情報 A を渡し、それぞれに CheckChildren 部の実行を依頼する。は、から CheckChildren 部の完了が報告されるまで待機する。

次にについてその動作を追う。まずは自分を祖先ノード情報 A へ追加し、隣接する子ノード群

$C = \{ \dots, \dots \}$ について標準経路を求める。ここで

A なので、条件より は子ノード群 C から除外される。 は標準経路として または のどちらかを選択するが、 を選択しないので条件、とも当てはまらない。同様に も除外されないので、 はパケットが と来た場合、経路として、 を持ち、 は経路とならない。そして は、 に CheckChildren 部の実行を求める。以上のような作業は Destination へ到達するまで再帰的に繰り返される。

以外に に到達するパターンは、

があり、これは、 と が条件 に当てはまらないとき、すなわち、 $Path(\dots) = \dots$ かつ、 $Path(\dots) = \dots$ のときに、生成される可能性がある。これ以外の条件においては MRS アルゴリズムではこの経路は生成されない。 の順で CheckChildren の依頼が来た場合、 は再び CheckChildren 部を実行し、経路 が生成される。これ以外に へ来る経路は存在しないので、最終的に が CheckChildren 部を完了した時点で の持つ多重経路表は次のようになる。

```
From    to    {
1:      , (    )
2:      (    )
}
```

ここで最初の行は、 から へのパケットについて {} 内の経路情報を用いることを示す。() 内はパケットの経由した履歴を表す。2行目は祖先の履歴がずらりとならんでいるが、 をとる経路は1つしか存在しないため、経路を最適化して多重経路表は次のようになる。

```
From    to    {
1:      , (    )
2:      (    )
}
```

この最適化については次節で述べる。

3.2. 多重経路表の最適化

ここでは、多重経路表を最適化するアルゴリズムについて述べる。

多重経路表の冗長性を低減するために、以下の条

件に基づいて多重経路表を修正する。

1. 次段経路が異なる場合、履歴情報を唯一と判断できるところでうち切る。
2. 次段経路が同一の場合、履歴情報が共通である範囲で1つにまとめる。

これらの条件により、多重経路表を最適化できる。例として、図3のノードにおける多重経路表の最適化の手順を説明する。 は MRS アルゴリズムにより、次の多重経路表を保持している。

```
From    to    {
1:      , (    )
2:      (    )
3:      (    )
4:      (    )
5:      , (    )
6:      (    )
}
```

最適化条件1により不要な履歴情報を省くことで、これらは次のように修正される。

```
From    to    {
1:      , (    )
2:      (    )
3:      (    )
4:      (    )
5:      , (    )
6:      (    )
}
```

また、最適化条件2により3行目と4行目はまとめられ、

```
(    )
```

となる。以上の結果、最適化された の多重経路表は、次のようになる。

```
From    to    {
1:      , (    )
2:      (    )
3:      (    )
4:      (    )
5:      , (    )
}
```

このようにして多重経路表を最適化する。

Source ノードは MRS アルゴリズムを完了すると、多重経路表のすべての次段経路に対し最適化を要請する。その後、各次段ノードも同様に自分の保持する多重経路表の全経路に対し最適化を要請し、要請は連鎖的に Destination まで伝達する。これらの要請に基づき各ノードは多重経路表の最適化を実施する。

3.3. FC 機構

ここでは、MRS アルゴリズムで生成された各次段経路へ効果的にネットワークトラフィックを分配し、負荷分散を行う FC 機構について述べる。

FC 機構では、多重経路表をもとに次段経路をある確率でランダムに選択し、パケットをルーティングする。その際、実際に送出したデータを用いて次段経路ごとの転送効率を計測し、それに基づいて各次段経路の流量を調整する。なお、ここで扱うパケットは IP パケットと異なり、ヘッダに経路情報を付加でき、かつ転送経路を指定可能なパケットを想定する。

データを転送する通信セッションが開始されると、まず MRS アルゴリズムにより各ノードに多重経路表が生成される。すでに生成されている場合はそれを利用する。各ノードは最初、生成された n 個の次段経路 $R_1 \sim R_n$ に対しルーティングする確率 $P(R_1) \sim P(R_n)$ を決める。例えば 3 個の次段経路があり、標準経路 R_1 を 70% の確率で利用し、残りの経路群を 30% で利用するときは、

$$P(R_1) = 0.7$$

$$P(R_2) = 0.15$$

$$P(R_3) = 0.15$$

のように決める。過去の履歴がある場合はそれを使用する。そして Source ノードからその確率に応じて次段経路を選択し、パケットを送出する。中継ノードも同様にしてルーティングする。その際に中継ノードは、パケットのヘッダにある経路情報に自分のノード情報を付加し、使用した経路が Destination ノード側で識別できるようにする。

Destination ノードではパケットを受信するごとにそのパケットに記録されている経路情報を読みとり、Source ノードから Destination ノードまでの一連経路ごとに受信データ量を加算して保存する。そして一

定時間ごとに応答パケットを、各一連経路の逆向きに Source まで返信する。応答パケットにはその一連経路から単位時間に受け取ったデータ量を書き込む。各中継ノードは、複数の応答パケット内の情報を読みとり、自分の持つ次段経路ごとに単位時間あたりの Destination への到着データ量 $D(R_1) \sim D(R_n)$ および、その合計到達量 D を計算する。それに基づいて次段経路の利用確率を

$$P(R_i) = D(R_i) / D \quad (1 \leq i \leq n)$$

として更新する。

各ノードがこれらの操作を行うことで、空いている経路を効果的に利用し、混雑している経路は徐々に利用しなくなるといった適応的なルーティングが可能となる。

以上により、FC 機構はネットワークトラフィックの負荷を分散し、Destination ノードへ効率よくデータを転送することができる。

3.4. DO-MRT 機構の有効性

本論文で提案した DO-MRT 機構では、Source ノードから Destination ノードへデータを送信する際に、MRS アルゴリズムにより不要となる経路を排除することで複数の経路を利用することができる。そして FC 機構によりそれらの経路へ効率的に負荷分散する事が可能となり、ネットワークトラフィック全体の負荷の平均化が実現できる。

また通信の途中で一部の経路が切断された場合、FC 機構は自動的にその経路を使用しなくなるため、切断の検知、代替経路検出の手間が必要ない。また通信中に新たなノードが加わった場合、新規参入ノードがネットワーク構成に変化が起きた旨を各隣接ノードへ伝達し、隣接ノードもそれを伝搬することでネットワーク全体に通知する。そして MRS アルゴリズムを再度実行し多重経路表を書き換えることにより、そのノードを利用することが可能となる。

多重経路表は、IP のルーティングテーブルとは異なり、必要に応じて生成するためすべてのエントリを常時保持する必要が無く、また RIP のように周期的にブロードキャストする必要もない。通信セッションが終了するたびに多重経路表を破棄することも可能だが、一度作成した多重経路表をキャッシュしておくことで利用効率を向上することができる。

また MRS アルゴリズムにおいて CheckChildren の最大再帰数を制限することで、多重経路表の作成にかかる時間を短縮でき、かつ多重経路表のサイズを縮小することが可能である。

関連研究である Scout アルゴリズム[2]と比較すると、Scout アルゴリズムが Destination ノード側からの定期的な要求により多重経路を探索するのに対し、MRS アルゴリズムは Source ノード側からの実行時の要求により探索される。そのため Scout アルゴリズムは1つのノードがすべてのノードに対する多重経路表を常に保持している必要があるのに対し、MRS アルゴリズムはその必要がない。多重経路を常時保持することは、その情報量の大きさとスケーラビリティから考えて好ましくない。

DO-MRT 機構を実装するルータが持つ情報量は、基本的に IP のルーティングテーブルの情報量と同等である。多重経路表をキャッシュする量は任意に設定可能なのでこの情報量は定数のオーダーと見なせる。

多重経路表を常時保持しないために MRS アルゴリズムの不利点と思われる多重経路表作成時間のオーバーヘッドについては、多重経路表が完成するまでの間、各ノード N_x が Destination ノードへの標準経路 $Path(N_x)$ を利用してパケットを転送できるよう実装することで、オーバーヘッドを軽減することが可能であると考えられる。

DO-MRT 機構の不利点は、パケットのヘッダ情報をいくつか照合、追加および削除する必要があることである。既存 IP のルータと比べ、パケットを処理する負荷が高い。しかし各経路の利用確率が固定的で FC 機構が必要ない場合は、各ノードの多重経路表を Source ノードが収集し、Source ノードがあらかじめすべての経路を決定して経路指定送信することで中継ノードの負荷を軽減できる。またコンピュータの処理能力の向上により、徐々にこの影響は減少すると考えられる。

4. DO-MRT 機構の実装

実際のネットワーク上で適用可能な DO-MRT 機構を実現し評価実験を行うために、プロトタイプを実装した。

DO-MRT 機構は既存のルーティングテーブルを利

用しつつ、IP ルータを拡張して既存ルーティング機能に置き換わるものとして提案している。しかし今回は以下の理由により、TCP/IP の上にアプリケーションルータ（以下、AR: Application Router）を構築し、AR 内で DO-MRT 機構を実装した。

- ・ ルータを改良することなく、既存の広域ネットワーク環境上で動作実験を行えるため。
- ・ 実現形態として、実用面から見るとスケーラビリティの観点でより現実的であるため。

AR は Java 1.3 により実装した。AR は起動時に隣接ノードを調べ、隣接ノードで起動されている別の AR と接続する。そして接続された AR 群は、それらの中で独自のネットワークを構築し、その上で仮想的にトランスポート層の機能を提供する。隣接ノード情報および標準経路情報は、DO-MRT 機構を IP ルータのレベルで実現するときには IP のルーティングテーブルより取得するが、AR で行うことを考慮し、ファイルなどに手動で指定し、それを読み込ませることも可能とした。これにより、ネットワーク内の代表的なノードのみに AR を起動させ、そのルータを経由して多重経路通信が可能となる。（図 4）

AR を利用したサーバアプリケーションおよびクライアントアプリケーションは、Java で ServerSocket クラス / Socket クラスの代わりに ServerSocketAR クラス / SocketAR クラスを用いることで容易に作成できる。例としてファイルサーバおよびクライアントを AR 用に作成した。

AR がルーティングするパケットは、サーバアプ

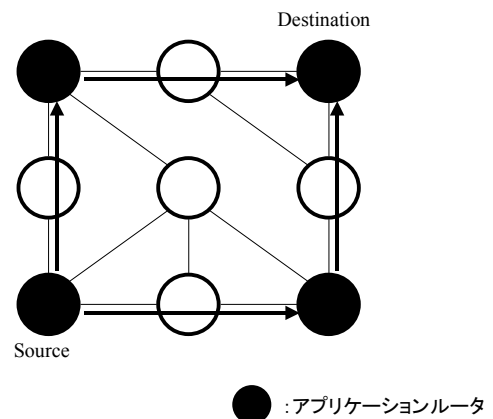


図 4 アプリケーションルータの配置例

リケーションが送出する送信データのストリームを分割してパケット化し、独自のパケットとして生成する。このパケットはヘッダに経由ノードの情報を追加でき、また転送経路をあらかじめ指定可能である。送信方法としては次の3種類の方法が指定可能である。

1. 標準送信
2. 多重経路送信
3. 経路指定送信

標準送信を指定する場合は、既存のIPルーティングと同じ標準経路を利用してパケットを転送する。多重経路送信はDO-MRT機構によって多重経路を用いて送信する方法である。なお、多重経路を選択・作成している間は自動的に標準送信される。経路指定送信は、あらかじめヘッダに埋め込まれた予定経路情報に従ってパケットを送信する方法で、主に応答パケットに利用される。

ARはパケットをルーティングする際にヘッダの経由情報に自分のノードを追加し、Destinationノード側で到達経路が識別できるようにする。それに基づきDestinationノードでは定期的に応答パケットを経路指定送信する。

Destinationノード上のARが受け取った受信データは、クライアントアプリケーションへ渡される。多重経路で受信した場合、パケットの到着順が不連続なため並べ替えが必要であるが、クライアントアプリケーションが直接並び替えるか、InputStreamARクラスを用いて連続したデータを受け取ることが可能となる。

各ARが持つトラフィック情報は外部から獲得できる。またFC機構による流量制御機能は停止することができ、次段経路の利用確率 $P(R_1) \sim P(R_n)$ を外部から任意に設定可能である。これによって、FC機構以外の別の流量制御機構からARの流量を制御可能とした。

このようなARを用いて次章で述べる評価実験を行った。

5. 評価実験

本機構の有用性を検証するため、次のような3種類の実験を行った。実験1と実験2はMRSアルゴリズムの動作検証実験であり、実験3は多重経路を

利用することによる転送効率向上の評価実験である。

実験1 図5のようにネットワークを構成し、袋小路を回避する様子を検証した。

実験2 図6のようにネットワークを構成し、ループを回避する様子を検証した。

実験3 図7のようにネットワークを構成し、標準送信した場合と多重経路送信した場合との転送効率を比較した。

なお、各図のSはデータの送信元、Dは送信先、N1、N2は中継ノードをさし、また各矢印はそれぞれの持つDへの標準経路を示す。

各ノードのネットワークインタフェースのIPアドレスは図中の数値の通りで、プレフィクス長はすべて24ビットである。

5.1. 実験1：袋小路回避の検証

図5においてMRSアルゴリズムを適用したところ、SとN1に次のような多重経路表が生成された。なお、DおよびN2には多重経路表は生成されなかった。

Sの多重経路表

```
From 10.70.0.1 to 10.70.2.1 {  
    10.70.0.2  
}
```

N1の多重経路表

```
From 10.70.0.1 to 10.70.2.1 {  
    10.70.2.1  
}
```

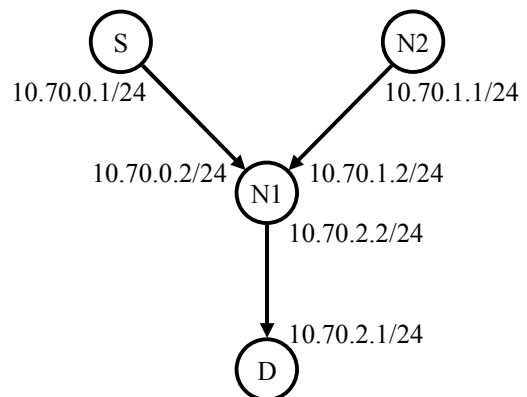


図5 袋小路を回避する実験の構成図

この多重経路表の生成結果から、パケットは N2 へ転送されることはなく、袋小路を回避したことが検証された。

5.2. 実験 2：ループ回避の検証

実験 2 では、同様に図 6 において MRS アルゴリズムを適用した。D 以外の各ノードに次のような多重経路表が生成された。

```
S の多重経路表
From 10.70.0.1 to 10.70.2.1 {
    10.70.0.2 , 10.70.1.2
}
```

```
N1 の多重経路表
From 10.70.0.1 to 10.70.2.1 {
    10.70.2.1 , 10.70.4.2 ( 10.70.0.1 )
    10.70.2.1 ( 10.70.4.2 10.70.1.1 )
}
```

```
N2 の多重経路表
From 10.70.0.1 to 10.70.2.1 {
    10.70.3.1 , 10.70.4.1 ( 10.70.1.1 )
    10.70.3.1 ( 10.70.4.1 10.70.0.1 )
}
```

この多重経路表の生成結果から、S、N1、N2 間で起こる可能性のあるループを回避したことが検証された。

5.3. 実験 3：転送効率の比較

既存 IP の単一経路による通信と、DO-MRT 機構に

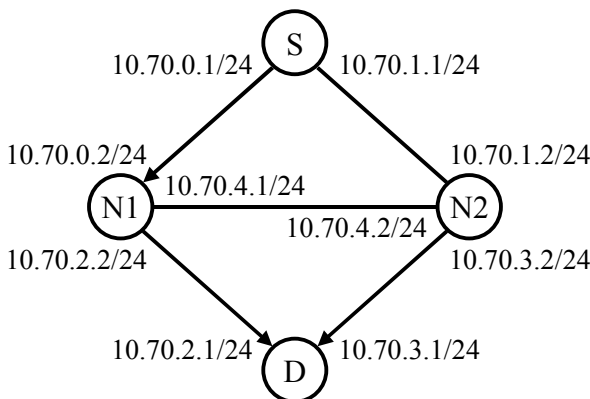


図 6 ループを回避する実験の構成図

よる多重経路通信を、転送効率において比較するのが本来の目的であるが、DO-MRT 機構の実装が AR であり、AR 上での仮想的トランスポート層に対して既存 TCP/IP のトランスポート層を比較対象とすることは適当でない。そこで、AR 上にて既存の IP ルーティングと同じ標準経路を利用してパケットを転送する標準送信を模倣的に利用し、標準送信と多重経路送信の転送効率を比較することで代替とする。ただこの実験では、パケットのヘッダを操作する際のオーバーヘッドについては検証できない。

なお、将来的には DO-MRT 機構を実際の IP ルータに実装し、既存 IP の転送効率と比較する予定である。

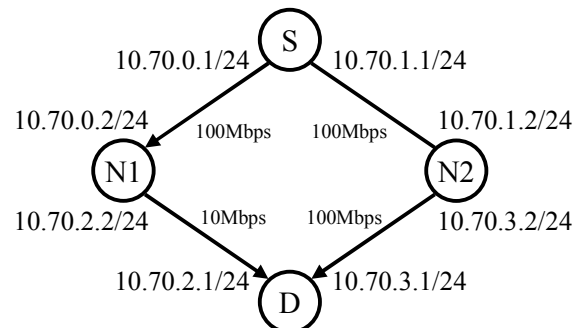
実験 3 では、S から D へ 2 本の送信経路を利用してデータを多重経路送信する。なお、N2 はコンピュータの性能が低く、N1 から D へのネットワーク回線は他の回線より細い。また AR の分割するパケットはヘッダが約 20~50 バイトの可変長で、データは 8 キロバイト以下に設定してある。事前の予備実験として S、N2、D の一連経路で計測したところ、転送効率は 2.28Mbps であった。

標準送信した場合の単一経路は S、N1、D となり、この転送効率を計測した結果、5.04Mbps であった。次に多重経路送信でデータを転送したところ、N1 と N2 の利用確率は

$$P(N1) = 0.7, P(N2) = 0.3$$

であり、転送効率は 6.50Mbps であった。これにより、DO-MRT 機構を用いた多重経路通信のほうが単一経路通信より転送効率が良いことがわかった。

ただ、多重経路を利用した理想的な転送効率は、S



S, N1, D: Pentium III 500MHz, 64MB, Windows98
 N2: MMX Pentium 150MHz, 32MB, Windows98
 NICは全て100Mbps, N1-D間は10MbpsのHUBを使用し、
 他はクロスケーブルで直接接続

図 7 多重経路通信の実験の構成図

N1 D と S N2 D の転送効率を合計した 7.33Mbps となる。しかし実際は 6.50Mbps となり 0.82Mbps ほど少ないが、この原因として次のような実装上の問題が考えられる。

- ・ MRS アルゴリズムが完了するまで単一経路で転送されるため、複数経路を利用できない時間が存在する。
- ・ FC 機構による利用確率に揺らぎがある。
- ・ パケットの送受信にキューを利用しているため、N1 の送信するパケットがすべて D まで到着し終えた後も、N2 がキューに残ったデータを送信し続ける状況が発生する。

これらは実装方法を改良することでより理想値に近づけることができるであろう。

なお、利用確率の理想値は、

$$P_{ideal}(N1) \quad 5.04 / 7.33 \quad 0.688$$

$$P_{ideal}(N2) \quad 2.28 / 7.33 \quad 0.312$$

であるので、実際の値は理想的であったと言える。

以上より、DO-MRT 機構の有用性を検証した。

6. むすび

本論文では、現状の IP における問題点を指摘し、その解決法として動的最適化多重経路通信機構による適応的なルーティングを提案した。また評価実験により、この提案の有用性を検証した。

今後は、本機構を IP ルータに実装し、本機構の IP レベルでの性能評価を行っていく予定である。

それと同時に、動的ネットワークアーキテクチャの FN 層内[3]で本機構を実現することで、広域異種ネットワーク環境上で本機構を利用可能とするための種々の検討を進めていく。

参考文献

- [1] 棟朝雅晴, 高井昌彰, 佐藤義治 : 遺伝的アルゴリズムによる負荷分散を有する適応的ルーティング, 情報処理学会論文誌 Vol.39, No.2, pp.219-227 (1997)
- [2] Johnny Chen and Peter Druschel and Devika Subramanian : A Simple, Practical Distributed Multi-Path Routing Algorithm, Rice Computer Science Technical Report TR98-320 (1998)
- [3] Suganuma T., Kinoshita T. and Shiratori N. : Flexible

Network Layer in Dynamic Networking Architecture, Proc of The 1st International Workshop on Flexible Networking and Cooperative Distributed Agents (FNCDA2000), pp.473-478 (2000)