

## 遺伝的アルゴリズムによる負荷分散機構を有する 適応型ルーティング

棟 朝 雅 晴<sup>†</sup> 高 井 昌 彰<sup>††</sup> 佐 藤 義 治<sup>†</sup>

本論文では、遺伝的アルゴリズムにより代替経路のリストを生成するとともに、それらの間で通信パケットを確率的に分配することで負荷の分散をはかる適応型ルーティングアルゴリズムを提案する。RIP や SPF など従来用いられてきたルーティングアルゴリズムではルーティングテーブルやリンク状態をネットワーク全体にブロードキャストするため、ネットワークが大規模化した場合に多くの通信コストを要し、ネットワーク全体の性能を低下させる。また、複数の良い代替経路がある場合でも1つの最短経路を集中して使用してしまう。本論文で提案するルーティングアルゴリズムは、実際に多数のパケットが使用している経路に関してのみ代替経路の生成およびその通信遅延時間の評価を行うため、ルーティングの情報交換に必要な通信コストを大きく削減することが可能となる。また、本アルゴリズムでは代替経路間でパケットを確率的に分配することで負荷の分散をはかる。離散事象シミュレーションに基づくネットワークシミュレータを構築し、比較実験を行った。その結果、従来手法に比べ少ない通信コストにより効果的なルーティングがなされていることが示された。

### An Adaptive Routing Algorithm with Load Balancing by a Genetic Algorithm

MASAHARU MUNETOMO,<sup>†</sup> YOSHIKI TAKAI<sup>††</sup> and YOSHIHARU SATO<sup>†</sup>

This paper presents an adaptive routing algorithm which has a load balancing mechanism among alternative paths by a genetic algorithm. Conventional routing algorithms such as RIP and SPF broadcast information on routing tables or link status in a network, which yields much communication overhead and degrades total performance when the network becomes large. Conventional routing algorithms only generate the shortest path to send a packet, even if some good alternative paths are available. Our routing algorithm generates alternative paths and observes communication latency only for paths frequently used. This mechanism greatly reduces communication cost for information exchanging of the routing. Moreover this algorithm realizes load balancing on the alternative paths by distributing packets among them. We perform simulation experiments using a discrete event simulator of network communication. The result of the experiments shows that our algorithm achieves effective routing with less communication overhead.

#### 1. はじめに

計算機ネットワークが大規模なものとなるにつれて、通信パケットのルーティングが重要な課題となりつつある。インターネットではホップカウント距離に基づくベクトル距離ルーティングやリンク状態ルーティングが一般に使用されている。ベクトル距離ルーティングではルーティングテーブルを、リンク状態ルーティングではリンク状態をそれぞれブロードキャストするために、ネットワークが大規模になった場合、多くの通

信コストを要する。この問題を解決するため、サブネット内ではデフォルト経路などを使用することでゲートウェイ外のノードへの経路をまとめてルーティングすることは可能であるが、ゲートウェイ間でのルーティングでは根本的な解決策とはならない。

ネットワークの負荷状態を観測して最適な経路を導出するためには、それぞれのリンクの負荷情報を交換する必要があると考えられる。ブロードキャストにより定期的に負荷情報を集めるという方法を採用した場合、情報の更新間隔を大きくすると負荷情報の精度が低下し、逆に小さくして頻繁に情報を集めると負荷情報の収集のための通信がネットワークの負荷を増大させてしまう。また、従来の手法においては、パケットを送出するときに最適な経路を1つだけ選択しその経

<sup>†</sup> 北海道大学大学院工学研究科

Graduate School of Engineering, Hokkaido University

<sup>††</sup> 北海道大学大型計算機センター

Computing Center, Hokkaido University

路のみを用いて通信を行っている。そのため、他に代替経路がある場合でも1つの経路にパケットが集中するおそれがある。同一のホップカウント距離を有するときのみ負荷分散を行っているプロトコルも存在するものの、代替経路間での負荷分散を動的に行うという観点は基本的には存在しない。

以上の背景から、ネットワークサイズが大きくなった場合でも少ない通信コストで効果的に負荷情報を集め、適切なルーティングを行うことが必要である。また、代替経路間で通信パケットを配分して通信回線に関する負荷分散を行うことも重要である。

本論文では、遺伝的アルゴリズム<sup>5)</sup>を適用して代替経路を生成し、それらの中で負荷を分散させることで効果的なルーティングを実現するGLBR (Genetic Load Balancing Routing) アルゴリズムを提案する。Coxら<sup>6)</sup>は遺伝的アルゴリズムをネットワークルーティングへ適用したが、そこではネットワークの帯域幅の割当て問題を組合せ最適化問題として解いており、パケット単位でのルーティングを実現してはいない。また、遺伝的アルゴリズムによるFCS (Fuzzy Classifier System) を用いてパケットを分配する研究例<sup>1)</sup>も存在するが、そこでは小規模ネットワークを対象に、直接接続している経路を選ぶか迂回路を選ぶかという2つの選択枝を学習するのみであり、現実のルーティングアルゴリズムに対して直接適用はできない。

GLBR アルゴリズムは、パケットの送出元ですべての経路を決定する始点制御ルーティングにおいて、ルーティングテーブルを構成する経路の集合に対して遺伝的操作を適用することで代替経路を生成する。経路を評価するためにそれぞれの経路の遅延時間に関する情報を集め、その情報を用いて動的にルーティングテーブルを更新する。

本論文の構成は以下のとおりとなる。まず従来用いられてきたルーティングアルゴリズムについて述べ、提案するアルゴリズムであるGLBR アルゴリズムについて述べる。次にルーティングに要する通信量に関して理論的考察を行った後、ネットワークシミュレータを用いた評価実験の結果を示す。

## 2. ルーティングアルゴリズム

互いに自律したノード (ゲートウェイ) およびリンク (通信回線) からネットワークは構成される。それぞれのノード間でパケットの交換により通信が実現される。あるノードから別のノードへパケットを送出するとき、その経路制御 (ルーティング) が必要となる。ベクトル距離ルーティングは、経路制御テーブルを直

接ブロードキャストすることで、距離情報を互いに交換する方式である。距離の基準としてRIP (Routing Information Protocol)<sup>4)</sup>などで用いられているのはネットワークのホップカウント距離 (hop count metric)<sup>2)</sup>である。これは通過したゲートウェイの数として定義される。ベクトル距離制御ルーティングは、経路制御テーブルの内容をすべてブロードキャストするため、分散システムの規模が大きくなった場合に、多くの通信量が必要になり、全体としてネットワークの性能を低下させるという問題点が存在する。

リンク状態ルーティングでは、各ノードがリンク状態 (リンクの接続状態、コストなど) をブロードキャストすることで他のノードへネットワークのトポロジー情報を伝える。通信経路はDijkstraのアルゴリズムを用いて最短経路を計算することで求められる。SPF (Shortest Path First) プロトコル、OSPF (Open Shortest Path First) プロトコル<sup>7)</sup>がリンク状態ルーティングの代表例である。リンク状態ルーティングでは、それぞれのリンク状態の情報のみがブロードキャストされるため、ベクトル距離ルーティングと比較した場合、互いに転送すべき情報量を削減することができる。しかしながら、実際には使用されないリンクについてもその状態がブロードキャストされるなど、システムが大規模化した場合には無駄な情報が通信量を増大させるおそれがある。

本論文で提案するGLBR アルゴリズムでは、実際に通信が行われている経路についてのみ、その経路の通信遅延時間の評価を行うための評価パケットを送出する。これにより、あまり使用されていない通信リンクに関する情報は交換されず、全体としてルーティングの情報収集のために使用される通信コストを大きく削減できる。

## 3. GLBR アルゴリズム

ここでは、本論文で提案するGLBR アルゴリズムについて述べる。GLBR アルゴリズムは始点制御ルーティング (source routing) に基づくルーティングアルゴリズムである。始点制御ルーティングでは、通信パケットの送出元ですべての経路を決定する。すなわち、あるパケットを送信する場合に、目的ノードへ到達するまでに経由するすべてのノードのリストが経路情報として与えられる。インターネットプロトコル (IP) においても、始点制御オプションが存在し、これを指定することで始点制御によるルーティングを行うことができる。

表1 ルーティングテーブル  
Table 1 A routing table.

dest.	route	delay	weight
2	(1 3 2)*	50	0.7
	(1 3 4 2)	60	0.2
	(1 3 4 5 2)	70	0.1
6	(1 8 6)*	100	0.4
	(1 10 11 6)	105	0.6
8	(1 8)*	40	0.9
	(1 7 8)	40	0.1

### 3.1 ルーティングテーブル

本手法で用いられるルーティングテーブルを表1に示す。それぞれの目的ノード (dest) ごとに経路リストがあり、それぞれの経路 (route) について、それを選択する確率 (weight), その経路を選択したパケットの通信遅延 (delay) が与えられる。表で、\* のついた経路はホップカウント距離の観点から見た最短経路である (以下では、default route と呼ぶものとする)。

初期状態では、ルーティングテーブルは空である。ルーティングテーブルに存在しない目的ノードへのルーティングを行う必要が生じた場合、トポロジー情報に基づき Dijkstra のアルゴリズムにより default route を求める。この場合、その目的ノードへの経路は default route のみであるため、weight の初期値は 1.0 となる。一定数のパケットを処理するごとに遅延時間の評価パケットを送出し、経路の遅延時間を計測する。また、一定回数遅延評価パケットを送出するごとに遺伝的操作を適用し、代替経路を生成する。

代替経路の生成によりルーティングテーブルのサイズが拡大するが、メモリ量の制限および検索速度を維持するため、テーブルのサイズを制限する。そのために、天野ら<sup>9)</sup>による手法を採用する。これは第一に、テーブル全体のサイズに関して制限を設け、それを超えた場合には使用頻度の低い目的ノードに関するルーティングテーブルを除去する。第二に、1つの目的ノードに関して代替経路数に制限を設け、それを超えた場合には重みの値 (適応度値) が小さい経路をルーティングテーブルから除去する。これは遺伝的アルゴリズムにおける selection<sup>3)</sup>に相当する。

以上の仕組みにより、送出パケット数が少ない目的ノードに関しては SPF と同様にホップカウント距離に基づく最短経路が用いられ、パケットの送出数がある一定数を超えた目的ノードに関してのみ、経路の遅延時間を計測するとともに遺伝的操作による代替経路の生成を行う。

### 3.2 代替経路の生成

代替経路の生成法として天野ら<sup>9)</sup>による手法では、

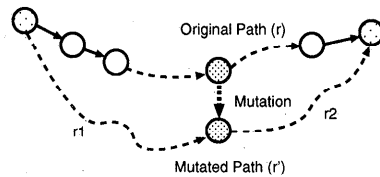


図1 経路に関する mutation  
Fig. 1 Mutation for a path.

始点となるノードに隣接するノードの中から1つのノードをランダムに選び (これを mutation node と呼ぶ), そのノードからの最短経路を生成することで、代替経路を求める。この場合、生成される代替経路が始点の隣接ノードに関するものに限られてしまう。また、村井ら<sup>8)</sup>による手法では、経路を構成するノードリストの中から、1つのノードをランダムに選び、そのノードを別のものに変化させるとともに、経路の整合性を確保するために、その前後に新たな経路を強制的に補う。この場合には、mutation により元の経路と大きく異なる経路が生成されるおそれがあり、その場合には長く無駄な経路が生成されることが予想される。

本論文では、元の経路に近く、かつ多様な代替経路を生成することが可能な mutation の手法を提案する。本手法では、ある経路リストに存在するノードで始点と終点を除いたものからランダムに1つのノードを選び、それに隣接するノードを再びランダムに1つ選択する。その選択されたノードを通過するという条件を満たす最短経路を求めることにより代替経路を生成する。

これを図示したものが図1である。たとえば、 $r = (0\ 3\ 5\ 6\ 7\ 10\ 12\ 15)$  なる経路に対して mutation を適用する場合を考える。はじめに経路からランダムに1つのノード、たとえばノード7を選択する。mutation のためにノード7に隣接するノード8が選ばれたとすると、始点とノード8を結ぶ経路  $r_1 = (0\ 2\ 4\ 8)$  と、ノード8と終点を結ぶ経路  $r_2 = (8\ 10\ 12\ 15)$  をそれぞれ生成する。最後に  $r_1$  と  $r_2$  を合成して、結果として  $r' = (0\ 2\ 4\ \boxed{8}\ 10\ 12\ 15)$  なる代替経路が生成される。

$r_1$  と  $r_2$  に重複が生じた場合、たとえば  $r_1 = (0\ 2\ 5\ 6\ 7\ 8)$ ,  $r_2 = (8\ 7\ 10\ 12\ 15)$  というように (7 8) のリンクを重複して通る経路となった場合には、結果として生成される経路が冗長なものとなるため mutation を行わない。

また、crossover に関しては、村井ら<sup>8)</sup>の手法を基本とし、改良したものを用いる。従来手法では、すべてのノードからランダムに交叉点となるノードを決定し、

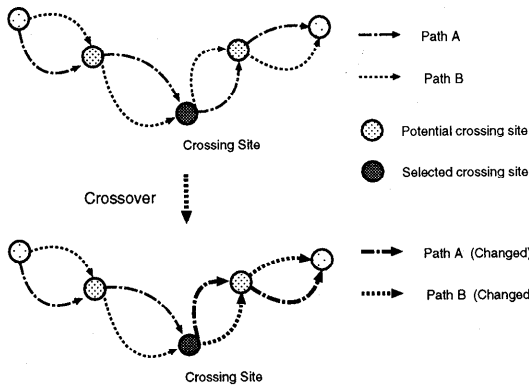


図2 経路に関する crossover  
Fig. 2 Crossover for two paths.

それと同じノードが crossover の相手となる経路に存在する場合に crossover を行う。これに対し本手法では、crossover を行う 2 つの経路で共通に存在するノードの中からランダムに 1 つのノードを選択し、この選択された交差点に関して、その後の経路を互いに交換する。これを図示したものが図 2 である。Path A, Path B を crossover する場合、両方の経路がともに通過するノードの中から交差点をランダムに選択し、その交差点となるノード以降の経路を互いに交換する。

ここで、例として以下に示す 2 つの経路に関して crossover を行う場合を考える。

$$r_1 = (0 \ 2 \ 3 \ \boxed{7} \ 9 \ \boxed{11} \ 12 \ \boxed{15} \ 17 \ 18 \ 20),$$

$$r_2 = (0 \ 4 \ 5 \ \boxed{7} \ 10 \ \boxed{11} \ 13 \ \boxed{15} \ 16 \ 19 \ 20).$$

共通のノードであるノード 7, 11, 15 の中からランダムに交差点を選択する。ここで、ノード 11 が選択されたものとする、ノード 11 の後の部分経路を交換することで crossover が以下のようにして実行される。

$$r_1 = (0 \ 2 \ 3 \ 7 \ 9 \ \boxed{11} \ \mathbf{13 \ 15 \ 16 \ 19 \ 20}),$$

$$r_2 = (0 \ 4 \ 5 \ 7 \ 10 \ \boxed{11} \ \mathbf{12 \ 15 \ 17 \ 18 \ 20}).$$

経路間で共通のノードが存在しない場合には crossover を行わない。

### 3.3 負荷分散

代替経路が複数存在する場合に、OSPF においては、ホップカウント距離が等しい経路間では負荷分散を行うが、それ以外の場合には負荷分散を行わない。

GLBR では、ルーティングテーブルにおいて、 $i$  番目の経路を用いて通信パケットを送信する確率重み  $w_i$  の値を以下の式により求め、その重みに応じた確率でパケットを分配することで、代替経路間での負荷分散を実現する<sup>10)</sup>。

$$w_i = \frac{1/\eta_i}{\sum_{j \in S} 1/\eta_j}, \quad \eta_i = \frac{d_i}{\sum_{j \in S} d_j}. \quad (1)$$

ここで、 $d_i$  は  $i$  番目の経路の観測された通信遅延時間、 $S$  は同じ目的ノードを持つ経路の集合である。この重み  $w_i$  の値を遺伝的操作における適応度値 (fitness value) として用いる。

遺伝的アルゴリズムによる天野ら<sup>9)</sup>の手法では、遅延時間の逆数を適応度値とし、それに比例した確率で通信パケットを分配していた。この場合、遅延時間の単位のとり方によって重みの値が変化するという問題点が生じるため、一度遅延時間を正規化してから逆数をとるという方法を採用した。

### 3.4 実行手順

GLBR ルーティングアルゴリズムは以下の手順で実行される。

#### 1) 初期化

起動時にルーティングテーブルを空にする。

#### 2) パケットの送信

通信パケットがある目的ノードへ送信する必要がある場合に、ルーティングテーブル内の目的ノードに対する経路群から確率重みに従って 1 つの経路を選択し、それに基づいてパケットを送出する。ルーティングテーブルに、その目的ノードに関する経路が存在しない場合には、default route を Dijkstra のアルゴリズムにより生成し、その経路をルーティングテーブルに登録する。

#### 3) 遅延評価パケットの送信

ある経路により一定数のパケットを送出するごとに、通信遅延時間評価のためのパケットをその経路を介して送出する。遅延評価パケットが目的ノードに到着すると、通信遅延時間を評価し、その結果が送信元へ送り返される。評価パケットを受け取った後、式 (1) により同じ目的ノードを有するすべての経路に関する適応度値 (確率重み) が計算される。リンクの過負荷などにより評価パケットがある一定時間以上経過しても戻ってこない場合には、その経路は使用しないものとしてルーティングテーブルから除去される。

#### 4) 遺伝的操作による代替経路の生成

ある目的ノードに関して通信遅延時間の評価が一定回数行われるごとに、遺伝的操作である mutation と crossover がその目的ノードに関する経路群に対して適用され、新たな代替経路が生成される。新たに生成された経路に対する通信遅延時間の初期値はそれを生成した元の経路から継承される。crossover の場合には、その経路の生成元である 2 つの経路に関する値の平均値を継承する。継承後、適応度値  $w_i$  の再計算を

行う。

新たな目的ノードの登録や代替経路の生成により、ルーティングテーブルのサイズがあらかじめ定められた閾値を超えた場合に、使用頻度の少ない目的ノードに関するルーティングテーブルを削除する。また、ある目的ノードにおいて代替経路数がある一定数を超えた場合には、適応度値の最も小さな経路を削除する。この場合にも、適応度値  $w_i$  の再計算を行う。

#### 4. 通信量の比較

ネットワークのノード数を  $n$  とした場合に、ルーティングを行うために必要なメッセージの数とそのサイズ、全体としての通信量について比較検討を行う。

RIP は一定時間間隔でルーティングテーブルをブロードキャストすることから、メッセージ数は  $O(n^2)$ 、通信量は  $O(kn^2)$  ( $k$ : ルーティングテーブルのサイズ) となる。ここで、 $k \sim O(n)$  であることから、結局、メッセージ数は  $O(n^3)$  となる。すなわち大規模なネットワークではメッセージ通信量が膨大なものとなることが予想される。

SPF は一定時間間隔でリンク状態をブロードキャストすることから、メッセージ数は  $O(n^2)$ 、通信量は  $O(ln^2)$  ( $l$ : ノードの次数) となる。ここでは  $l$  については、ネットワークが 2 次元格子など次数一定のグラフの場合に  $l \sim O(1)$ 、ハイパキューブの場合で  $l \sim O(\log n)$ 、完全グラフの場合で  $l \sim O(n)$  となるため、全体の通信量は  $O(n^2) \sim O(n^3)$  となる。ただし、完全グラフのネットワークにおいてはルーティングを行う必要が生じないので、実際には考慮しなくてもよい。

GLBR の場合、それぞれのノードにおいてパケットが一定数送出されるごとに評価パケットが送出され、ネットワークのノード数に関係しないため、ネットワーク全体としてのメッセージ数は  $O(n)$  となる。また、メッセージのサイズは一定であるため、全体としての通信量は  $O(n)$  となる。

以上の結果をまとめて表 2 に示す。ネットワークのサイズが大きくなった場合、ルーティングテーブルのために必要なメッセージ通信量が GLBR で最も少な

くなり、次いで、SPF、RIP の順となる。すなわち、提案手法を用いることにより大規模なネットワークにおいてルーティングに必要なメッセージ通信量を大きく削減することが可能となる。

#### 5. 評価実験

離散事象シミュレーション (Discrete Event Simulation) の考え方を基本としたシミュレータを構築し、それを使用して評価実験を行った。

以下のアルゴリズムについて、比較実験を行った。

- RIP
- SPF (ホップカウント距離に基づく)
- 適応型 SPF (通信遅延を考慮した SPF)
- GLBR (本論文での提案手法)

RIP、SPF に関しては、30 (s) の間隔でルーティングテーブルまたはリンク状態をブロードキャストするものとする。リンクの負荷状態を考慮した適応型 SPF は、それぞれのリンクの遅延時間を調べて一定時間間隔でブロードキャストし、遅延時間最小の経路を Dijkstra のアルゴリズムにより求める。ここではリンクの遅延時間をブロードキャストする間隔を 30 (s)、60 (s) として実験を行った。

実験には図 3 に示すネットワーク例を使用した。図において、太い線は 450 Kbps の通信リンクを示し、細い線は 150 Kbps の通信リンクを示す。また、それぞれの丸印はノードを示し、すべてのノードにおいて同じ発生間隔でパケットが生成される。パケットの目的ノードは、ノード 0、3、7、11、12、17 に限られるものとする。ここで、ノード 11 においては他のノードの 2 倍、ノード 7 においては他のノードの 3 倍の確率でパケットの目的ノードとして選択されるものと

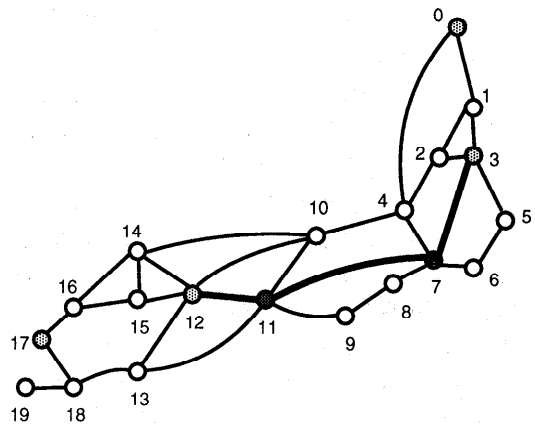


図 3 実験に使用したネットワーク例  
Fig. 3 An example of network for the experiments.

表 2 通信量の比較

Table 2 Comparison of communication overhead.

	メッセージ数	メッセージサイズ	通信量
RIP	$O(n^2)$	$O(n)$	$O(n^3)$
SPF	$O(n^2)$	$O(1) \sim O(n)$	$O(n^2) \sim O(n^3)$
GLBR	$O(n)$	$O(1)$	$O(n)$

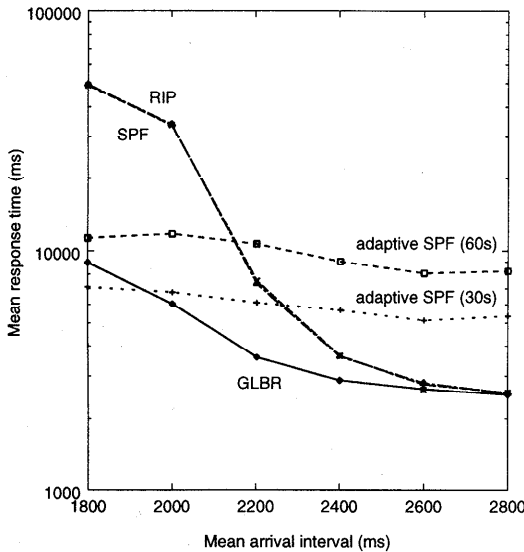


図4 平均通信遅延時間の比較

Fig. 4 A comparison of mean delay.

する。

パケットサイズは指数分布に従うものとし、その平均値を10 KBytesとした。シミュレーションを3000秒間実行し、終了後のネットワーク状態を観測し、以下に示す結果を得た。

### 5.1 平均通信遅延時間の比較

全体としてのルーティングの効果を比較するため、図4にパケットが発生してからそれが目的地に到着するまでの通信遅延時間の平均値に関する比較実験の結果を示す。図の横軸は、パケットの発生間隔（指数分布に従う）の平均値を示し、縦軸は平均通信遅延時間（対数表示）を示している。適応型SPFに関しては、リンク状態のブロードキャスト間隔が30(s)および60(s)の場合に関して結果を示している。

この結果によると、パケットの発生間隔が短く負荷が重い場合には、RIP、SPFに比べ、提案手法であるGLBRにより平均通信遅延時間が大きく改善されていることが分かる。たとえば発生間隔が2000(ms)の場合にはRIP、SPFの1/5の通信遅延時間となっている。また、適応型のSPFを用いることにより、負荷の重い場合には通信遅延時間がGLBRと同程度まで改善されているが、負荷が軽い場合であっても通信遅延時間が短くはならず、他の手法に比べて長い通信遅延時間となっている。たとえば、発生間隔が2800(ms)の場合、GLBRなどに比べ、ブロードキャスト間隔が30(s)の適応型SPFで2倍、60(s)で3倍の通信遅延時間となっている。

また、ルーティングを行うために使用したパケット

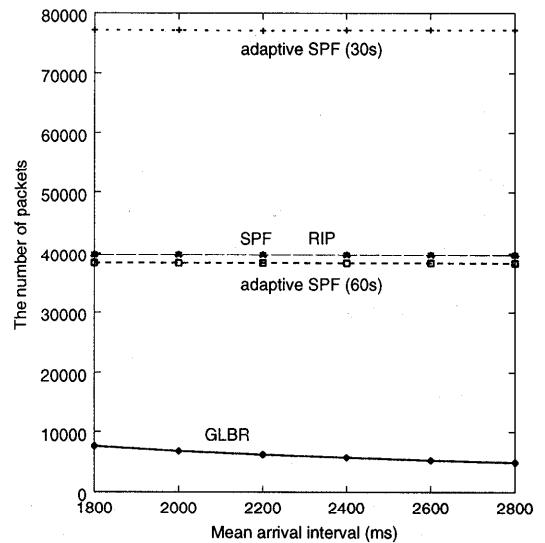


図5 ルーティングに要するパケット数の比較

Fig. 5 A comparison of the number of packets for routing.

の送出数に関する比較結果を図5に示す。図の横軸は、パケットの平均発生間隔（指数分布に従う）を示し、縦軸は送出されたパケット数を示す。

RIP、SPFおよび適応型SPFでは定期的にルーティングテーブルやリンクの遅延をブロードキャストする必要があるため、多くのパケットを送出する結果となっている。適応型SPFではブロードキャストの間隔を短くすることでリンク遅延時間の変化に速やかに対応することが期待できるが、そのためにはより多くのパケットを送出する必要があるが、ブロードキャスト間隔をいくらでも短くすることはできない。GLBRではパケットが一定数送出された場合に遅延時間評価のパケットを送出するため、全体としてのパケット送出数が大幅に少なくなっているのみならず、パケットの送出間隔が大きく全体としてパケットの送出数が少ない場合には、ルーティングに使用されるパケット数もさらに減少している。

### 5.2 負荷分散の効果について

上で述べた負荷分散の効果について、それを裏付ける実験結果を示す。パケットの平均発生間隔が2000(ms)の場合において、それぞれの通信リンクの混み具合（平均待ち行列長）を図6、7、8、9に示す。ここで、リンクの太さは平均待ち行列長に対数をとったものに比例している。図6の下部にリンクの太さとその平均待ち行列長との関係を示す。他の図7~9においても共通である。したがって、リンクが太くなるにつれ、急激に待ち行列長が増加していることとなる。

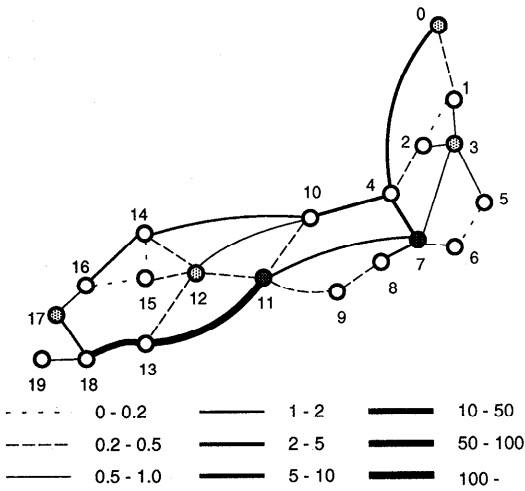


図6 ネットワークの負荷状態 (RIP)  
Fig. 6 Load status of links (RIP).

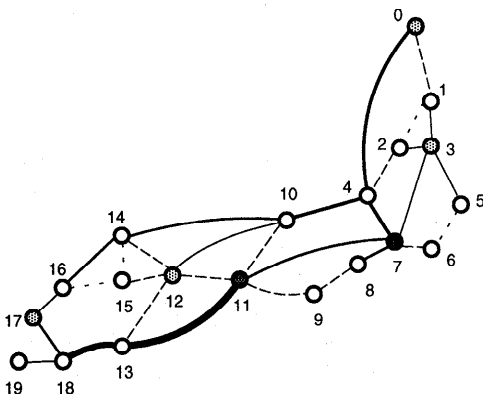


図7 ネットワークの負荷状態 (SPF)  
Fig. 7 Load status of links (SPF).

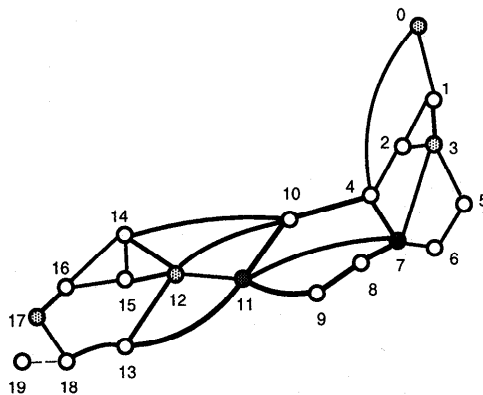


図8 ネットワークの負荷状態 (adaptive SPF)  
Fig. 8 Load status of links (adaptive SPF).

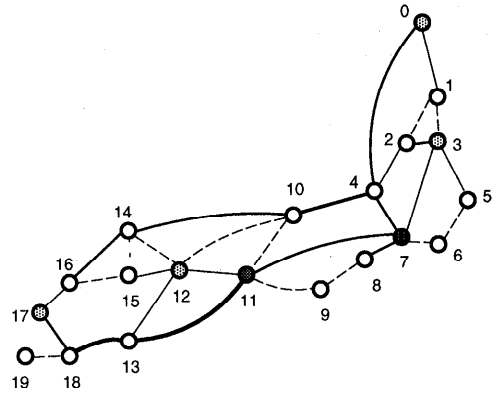


図9 ネットワークの負荷状態 (GLBR)  
Fig. 9 Load status of links (GLBR).

表3 ノード0において生成されたルーティングテーブル  
Table 3 The routing table generated in node No.0.

dest.	route	delay	weight
3	(0 1 3)*	554	0.802636
	(0 4 2 3)	2253	0.197364
7	(0 4 7)*	5052	1.000000
11	(0 4 7 11)*	4423	0.533488
	(0 1 3 7 11)	5058	0.466512
12	(0 4 7 11 12)	2941	0.564116
	(0 4 10 12)*	6210	0.267160
	(0 1 2 4 10 12)	9833	0.168724
17	(0 4 10 14 16 17)*	2859	1.000000

ここで得られた結果を見ると、RIP、SPFにおいては、(11 13 18)の経路が著しく混雑している一方、(11 13)の代替経路である(11 12 13)や、(11 13 18)の代替経路である(11 12 15 16 17 18)がほとんど使用されていない。適応型 SPF においては、負荷状態の観測を行っているために負荷の偏り自体は少なくなっているが、負荷状態観測の packets が多数送出されるため、全体的にリンクの負荷が増加している。以上の3つとは対照的に、提案手法である GLBR では負荷の著しい偏りが生じておらず、どのリンクもある程度平均的に使用されると同時に、ルーティングのために使用される packets 数も少ないため、それぞれのリンクの負荷も少ない。結果として GLBR アルゴリズムによりネットワークの負荷分散が実現できていることが示された。

### 5.3 生成されたルーティングテーブル

ここでは、どのようなルーティングテーブルが実際に生成されたかについてシミュレーションの結果を示す。

表3に、ノード0で生成されたルーティングテーブルを示す。ノード12が目的地の場合、最も遅延が

表4 ノード13において生成されたルーティングテーブル  
Table 4 The routing table in node No.13 generated after the simulation.

dest.	route	delay	weight
0	(13 11 7 4 0)	7429	1.000000
3	(13 11 7 3)	10415	1.000000
7	(13 12 11 7)	2464	0.844551
	(13 11 7)	12756	0.155449
11	(13 12 11)	309	0.992808
	(13 11)	42655	0.007192
12	(13 11 12)	2725	0.006562
	(13 12)	18	0.993438
17	(13 18 17)	212	1.000000

短く選択される確率の大きい経路は(0 4 7 11 12)となっている。従来の手法で用いられているホップカウント距離による最短経路は(0 4 10 12)である。しかし、(7 11), (11 12)のリンクは他のリンクに比べて帯域幅が大きいので、遅延時間に関して(0 4 10 12)は最短経路とはいえない。GLBR アルゴリズムにより自動的に遅延時間の短い経路が選択されており、さらにはいくつかの代替経路を用いて通信パケットが送出されていることがこのルーティングテーブルより分かる。たとえば、ノード12が目的地の場合には3つの経路が同時に使用されている。

表4にノード13において生成されたルーティングテーブルを示す。

ここで、顕著な傾向としては、(13 11)のリンクがほとんど使用されていないということがあげられる。たとえば、目的地がノード11の場合に、直接つながっている(13 11)の経路はほとんど使用されず、代わりに(13 12 11)なる経路が使用されている。また、目的ノードがノード7の場合に、ホップカウント距離に関する最短経路(13 11 7)ではなく、(13 12 11 7)が使用される結果となっている。

以上の結果により、GLBR アルゴリズムにおいては、直接リンクの状態を観測せず、実際に使用されている経路の通信遅延の情報を用いて間接的に遅延時間の少ない経路を発見していることが分かる。

## 6. 結 論

本論文では、代替経路間での負荷分散を実現するルーティングアルゴリズムを提案し、ルーティングに要する通信量の比較検討、およびシミュレーションによる評価実験を行った。その結果、提案手法は少ない通信量でルーティングが行うことが可能であり、またネットワークにおける負荷の分散がはかられており、結果としてパケットの平均通信遅延時間が大きく改善

されることが確かめられた。

今後の課題としては、始点制御以外のルーティングアルゴリズムに対する本手法の応用や、現実で使用されているルーティングプロトコルへの実装があげられる。

## 参 考 文 献

- 1) Carse, B., Fogarty, T.C. and Munro, A.: Adaptive Distributed Routing Using Evolutionary Fuzzy Control, *Proc. 6th International Conference on Genetic Algorithms*, Eshelman, L.J. (Ed.), pp.389-396, Morgan Kaufmann Publishers (1995).
- 2) Comer, D.E.: *Internetworking with TCP/IP*, Vol.I, Second Edition, Prentice Hall (1991).
- 3) Goldberg, D.E.: *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison Wesley (1989).
- 4) Hedrick, C.: Routing Information Protocol, Network Working Group, RFC-1058 (1988).
- 5) Holland, J.H.: *Adaptation in Natural and Artificial Systems*, University of Michigan Press (1975).
- 6) Cox Jr., L.A., Davis, L. and Qiu, Y.: Dynamic Anticipatory Routing in Circuit-Switched Telecommunications Networks, *Handbook of Genetic Algorithms*, Davis, L. (Ed.), Van Nostrand Reinhold (1991).
- 7) Moy, J.: The OSPF Specification, Network Working Group, RFC-1131 (1989).
- 8) 村井康紀, 棟朝雅晴, 高井昌彰: 遺伝的アルゴリズムによる始点制御ルーティング, 情報処理学会研究報告, Vol.96, No.108, pp.43-48 (1996).
- 9) 天野美樹, 高井昌彰, 佐藤義治: 遺伝的操作用による分散協調型ルーティングとその評価, 情報処理学会研究報告, Vol.94, No.105, pp.25-30 (1994).
- 10) 棟朝雅晴, 高井昌彰, 佐藤義治: 負荷分散機構を有する適応型ルーティングの一手法, 情報処理学会研究報告, Vol.97, No.13, pp.205-210 (1997).

(平成9年5月9日受付)

(平成9年9月10日採録)



棟朝 雅晴 (正会員)

昭和43年生。平成3年北海道大学工学部電気工学科卒業。平成8年同大学大学院工学研究科情報工学専攻博士課程修了。同年同大学工学部助手(情報解析学分野)。現在に至る。分散処理、遺伝的アルゴリズムに関する研究に従事。IEEE, ACM各会員。





高井 昌彰 (正会員)

昭和35年生。昭和58年東北大学工学部電子工学科卒業。昭和63年同大学大学院工学研究科情報工学専攻博士課程修了。工学博士。同年東京大学理学部情報科学科助手。平成元年北海道大学工学部講師。平成4年同助教授。平成7年同大学大型計算機センター助教授。現在に至る。高度並列・分散処理システム、計算機アーキテクチャ、コンピュータグラフィックスの研究に従事。著書「WWWオーサリング技法」(森北出版)など。電子情報通信学会、IEEE各会員。



佐藤 義治 (正会員)

昭和21年生。昭和50年北海道大学大学院工学研究科情報工学専攻修士課程修了。同年同大学工学部助手。同助教授を経て現在、同大学工学部教授(情報解析学分野)。工学博士。多次元データ解析、多変量解析、ニューラルネットワーク理論、ファジィデータ解析の研究に従事。日本統計学会、日本応用統計学会、日本計算機統計学会、日本行動計量学会、日本ファジィ学会、国際計算機統計学会、国際分類学会、アメリカ統計学会各会員。