

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第3792940号
(P3792940)

(45) 発行日 平成18年7月5日(2006.7.5)

(24) 登録日 平成18年4月14日(2006.4.14)

(51) Int. Cl. F I
 HO4L 12/56 (2006.01) HO4L 12/56 100Z
 HO4M 3/00 (2006.01) HO4L 12/56 260Z
 HO4M 3/00 A

請求項の数 2 (全 23 頁)

(21) 出願番号	特願平11-163463	(73) 特許権者	000005223
(22) 出願日	平成11年6月10日(1999.6.10)		富士通株式会社
(65) 公開番号	特開2000-354063(P2000-354063A)		神奈川県川崎市中原区上小田中4丁目1番1号
(43) 公開日	平成12年12月19日(2000.12.19)	(74) 代理人	100085187
審査請求日	平成16年3月24日(2004.3.24)		弁理士 井島 藤治
		(74) 代理人	100090424
			弁理士 鮫島 信重
		(72) 発明者	今井 祐二
			神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
		(72) 発明者	湯原 雅信
			神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

最終頁に続く

(54) 【発明の名称】 パケットのマルチキャスト配送システム

(57) 【特許請求の範囲】

【請求項1】

少なくとも1台のルータを回線で接続し、各ルータにはノードが接続され、パケットの送受信を行なうシステムにおいて、

転送されるパケットヘッダに複数の宛先アドレスリストとその未配送ビットマップを持つパケットを、ユニキャスト経路に従って中継する手段をルータ又はノードに具備すると共に、

同一パケットを複数の宛先に配送する際に、宛先アドレスのリストと未配送ビットマップをパケットヘッダに格納して送信するトランスミッタを具備し、

前記ルータは、トランスミッタが分岐正則印を添えて送信したパケットを中継する際に、未配送ビットマップで未配送となっている列の両端の2ノード分について経路表検索を行ない、分岐不要の場合には、他のアドレスに関する経路探索を省略し、分岐が必要な場合にのみ全宛先の経路表検索を行なうことを特徴とするパケットのマルチキャスト配送システム。

【請求項2】

パケットを分岐させる際に、配送済みとなったアドレスについてはアドレスとして意味のない値に変更してから配送することを特徴とする請求項1記載のマルチキャスト配送システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は分散パケット交換網におけるマルチキャスト配送システムに関する。

インターネットを代表とする分散パケット交換網においては、パケット配送形式を宛先指定方法によって幾つかに分類している。現在のインターネットで使用されているIPv4や次世代インターネットの標準であるIPv6では、以下のパケット配送が用いられている。

- 1 宛先が唯一のインターフェイスを表わすアドレスであるユニキャスト
- 2 宛先が複数のインターフェイスのグループを表し、その全てへパケットのコピーを配送するマルチキャスト
- 3 宛先が複数のインターフェイスのグループを表し、その中のいずれかが一つにパケットを配送するエニーキャスト

本発明は前記 2 のマルチキャストに関するものである。マルチキャストは、同一の発信内容を複数のノードに効率よく配送できるので、マルチメディアデータの放送や、多点音声動画会議等への応用が試みられている。

【0002】

【従来の技術】

IPv4におけるマルチキャスト実現を例に従来技術を説明する。現在標準化が提案されているIPv6においても実現方法はほぼ同様である。実際にマルチキャストを実行するためには、以下の3つの手順を踏むことになる。

- 1 マルチキャストのアドレスの割り当て
- 2 経路設定の依頼
- 3 パケットの配送

これらの手順について、以下に説明する。

【0003】

- 1 マルチキャストのアドレスの割り当て

IPv4では、4オクテットのIP(Internet Protocol)アドレス空間のうち、16分の1をクラスDと呼ばれるマルチキャストのためのアドレスに割り当てている。クラスDのアドレスは、上位4ビットが1110で始まるように決められている。パケットのマルチキャストを行なう送信者は、マルチキャストするための宛先ノードのグループ毎に一つずつIANA(Internet Assigned Number Authority)、ICANN(Internet Corporation for Assigned Names and Numbers)より、マルチキャストアドレスの割り当てを受ける。

【0004】

- 2 経路設定の依頼

発信者は受信者の一つ一つと連携をとりながら、予め、マルチキャストパケットが配送される経路上の全てのルータに 3 の実際のパケット配送で使用される経路設定を依頼する必要がある。

【0005】

- 3 パケットの配送

IPv6パケットは、ユニキャスト・マルチキャストとも図20に示すヘッダフォーマットを持つ。図のversionは、版数であり、classはビデオ、オーディオ、データの何れであるかを知らせるためのものであり、Flow Labelは、フローに対して何番のフローをつけてやるかを決定するものである。

【0006】

Payload Lengthはデータの長さを示し、Next Headerは何のポートコルであるかを予め知らせるためのものである。Hop Limitは、ネットワーク内でパケットが堂々巡りしないようにパケットの中継回数の上限を示すものである。

【0007】

Source Addressは、発信元アドレス、Destination Addressは、宛先アドレス

s s は、宛先アドレスである。最後に上位プロトコルデータがくる。

【 0 0 0 8 】

送信ノードはこのうち、宛先アドレスに割り当てられたマルチキャストアドレスを格納して送信する。途中のルータは、パケットを正しい方向に中継するために予め用意された経路表を検索する。経路表は、概略図 2 1 のようになっている。図中、ネットワーク列には、インターネット上で、このルータから到達可能なネットワークを列挙してある。

【 0 0 0 9 】

ネットワークは、ネットワークプリフィクスとマスクで表現する。例えば、プリフィクスが 3 F F E : 5 0 1 : 1 0 0 0 : : でマスクが 4 0 (F F F F : F F F F : F F 0 0 : :) ならば、3 F F E : 5 0 1 : 1 0 0 0 : 0 : 0 : 0 : 0 : ~ 3 F F E : 5 0 1 : 1 0 F F : F F F F : F F F F : F F F F : F F F F : F F F F を範囲とするネットワークを表わす。

10

【 0 0 1 0 】

ディスティネーション列には、そのネットワーク宛のパケットを配送するために、このルータ自身が次に配送を依頼するべきルータのアドレス *n e i g h b o r* と、そのためにパケットを送出するインターフェイスを表している。経路表検索は、先ず検索対象となるアドレスと *n e t m a s k* の論理積 (A N D) を計算し、結果がネットワークプリフィクスと等しくなる項目を探す、検索した項目のディスティネーション *n e i g h b o r* に対してパケットをインターフェイスから配送する (この経路表検索には種々の高速化技法があり、特許も多数申請されている。) 。

20

【 0 0 1 1 】

ユニキャストアドレスを中継する際には、宛先 (ディスティネーション) は必ず一つだけ設定されており、指定されたインターフェイスから指定された次のルータへパケットを渡すことになる。発信者から受信者までの全てのルータで必ず一つずつの次のルータが設定されているので、結果としてパケットは 1 本の経路上を配送されることになる。

【 0 0 1 2 】

一方、マルチキャストに関しては、ルータにより宛先 (ディスティネーション) が 1 つの場合と、2 つ以上存在する場合を許す。1 つの場合にはユニキャストと同じ動作を行なうが、2 つ以上の場合には、それぞれにパケットをコピーして配送する。これにより発信時に 1 つだったパケットがネットワーク上で枝分かれしながら複数の受信者へ到達することができる。

30

【 0 0 1 3 】

【 発明が解決しようとする課題 】

既存のマルチキャスト方式には以下の問題がある。

1 アドレス割り当て

マルチキャストを行なう場合、配送するグループ毎に一つずつマルチキャストアドレスを割り付ける必要がある。現行のテレビ・ラジオの代替としての放送型マルチキャストの場合は、恒常的にアドレスを割り付けることができるが、多地点テレビ会議中継等の動的にチャンネルが増える通信の場合にはその都度動的にマルチキャストアドレスの払い出しを行なう必要がある。これは、インターネットで一意になるように行なう必要があり、その整合性をとるために、ある程度の複雑度を持った仕組みが必要になる。

40

【 0 0 1 4 】

2 ルータ設定

マルチキャストアドレスの割り当て後、発信者から宛先クライアント一つ一つに至る経路の途中にある全てのルータの経路表にマルチキャストアドレスの項目の設定を行なう必要がある。インターネットの基幹となるルータでは、この経路表が膨大な数になる。更に、マルチキャストアドレスに対応する宛先ノードグループはメンバが頻繁に入れ替わる。そのたびに経路を再計算し経路表の更新を行なう必要がある。この処理量も膨大な量となる。

【 0 0 1 5 】

50

本発明はこのような課題に鑑みてなされたものであって、アドレス設定とルータ設定を容易にしたマルチキャスト配送システムを提供することを目的としている。

【0016】

【課題を解決するための手段】

(1) 図1は本発明の原理ブロック図である。図に示すシステムは、ルータ1同士が専用線2を介して接続されている。各ルータ1には、ノード4が接続されている。3は専用線上を転送されるパケットである。

【0017】

図では、転送されるパケットヘッダに複数のアドレスリストとその未配送ビットマップを持つパケットをユニキャスト経路に従って、中継するようになっている。この中継手段は、ルータ1又はノード4内に設けられる。なお、ノード4は、以下に説明するトランスミッタ又はクライアントとして機能する。

10

また、同一パケットを複数の宛先に配送する際に、宛先アドレスのリストと未配送ビットマップをパケットヘッダに格納して送信するトランスミッタを具備し、前記ルータは、トランスミッタが分岐正則印を添えて送信したパケットを中継する際に、未配送ビットマップで未配送となっている列の両端の2ノード分について経路表検索を行ない、分岐不要の場合には、他のアドレスに関する経路探索を省略し、分岐が必要な場合にのみ全宛先の経路表検索を行なう。

【0018】

このように構成すれば、マルチキャスト経路情報を持たずに、ユニキャスト経路情報だけでマルチキャスト配送が可能となる。また、マルチキャストパケット配送を行なう際に、検索すべきユニキャストアドレスの数が少なく、パケット配送処理が小さくなる。

20

【0020】

(2) 請求項2では、パケットを分岐させる際に、配送済みとなったアドレスについてはアドレスとして意味のない値に変更してから配送することを特徴とする。

【0021】

このように構成すれば、マルチキャストに参加している他のクライアントに対して、自身が参加していることを秘匿できる。

また、同一パケットを複数の宛先に配送する際に、宛先アドレスのリストと未配送ビットマップをパケットヘッダに格納して送信するトランスミッタを具備するように構成すれば、中継ルータ群に対して、マルチキャスト経路情報設定依頼を行なうことなく、マルチキャストパケットの発信が可能となる。

30

【0022】

また、リストにおけるアドレス出現順位を、パケット配送経路上の任意のパスにおいて、未配送ビットマップの未配送部分が常に連続するように、予め整理され、配送済みであることを示す印を添えて送信するトランスミッタを具備するように構成すれば、分岐正則ルータ(後述)が、効率的にマルチキャストパケット配送を行なうことが可能となる。

【0023】

また、前記トランスミッタが分岐正則印を添えて送信したパケットを中継する際に、未配送ビットマップで未配送となっている列の両端の2ノード分についてのみ経路表検索を行ない、分岐不要の場合には他のアドレスに関する経路探索を省略し、分岐が必要な場合にのみ全宛先の経路表検索を行なうルータを具備するように構成すれば、マルチキャストパケット配送を行なう際に、請求項1のルータに比べて検索すべきユニキャストアドレスの数が少なく、パケット配送処理が小さくなる。

40

【0024】

また、アドレスリストを持つ分岐探索パケットを送信し、得られた探査結果を元に分岐正則リストを作成するトランスミッタを具備するように構成すれば、分岐正則リストの自動的な作成が可能になる。また、管理者等が予め正則なリストを作成する必要がなくなる。

【0025】

また、トランスミッタが送信した経路探査パケットについて、経路表検索を行ない、分岐

50

を行なう際には、自ルータまでの未配送経路リストの最後尾に、未配送ビットマップを追加して探査 packets を中継し、分岐しない場合にはそのまま探査 packets を中継するルータを具備するように構成すれば、分岐正則リストの自動的な作成が可能になる。また、管理者等が予め正則なリストを作成する必要がなくなる。

【0026】

また、前記探査 packets が中継された宛先ノードで、探査 packets の未配送経路リストをそのままトランスミッタに返送するクライアントを具備するように構成すれば、分岐正則リストの自動的な作成が可能となる。また、管理者等が予め正則なリストを作成する必要がなくなる。

【0027】

また、経路表検索の結果、同じルータへ中継する宛先が1つ若しくは2つとなる経路がある場合、2つになる経路を1つまでか、1つになる経路を2つまで経路探索 packets の中継を省略するルータを具備するように構成すれば、経路探索のための packets 数が少なくなり、トランスミッタでの判断の処理が少なくなる。また、ネットワーク流量を節約できる。更に、ルータ/クライアントの中継処理、折り返し処理が節約できる。

【0028】

また、前記ルータによって、経路探索 packets が省略されていることを前提に、分岐木を解析してアドレスリストを分岐正則に整列させるルータを具備するように構成すれば、経路探索のための packets 数が少なくなり、トランスミッタでの判断の処理が少なくなる。また、ネットワーク流量を節約できる。更に、ルータ/クライアントの中継処理、折り返し処理が節約できる。

【0029】

また、一連の packets 送信の途中で、宛先アドレスの追加/削除ができるトランスミッタを具備するように構成すれば、マルチキャスト配送中のチャンネルに、途中から参加/脱退が可能になる。

【0030】

また、追加したアドレスに関する経路探索を行ない、アドレスを分岐正則に保つトランスミッタを具備するように構成すれば、マルチキャスト配送中のチャンネルに、途中から参加/脱退しても分岐正則による配送の効率化が継続する。

【0031】

また、定期的にアドレスリストが分岐正則かをチェックする正則検査 packets を送信し、非正則通知がくると正則化をやり直すトランスミッタを具備するように構成すれば、マルチキャスト中継中に途中経路の変更があっても、一定時間で検出し、適応することが可能になる。

【0032】

また、前記トランスミッタが出す正則検査 packets のアドレスリストを経路表検索し、正則性が崩れていた場合に非正則通知を返送し、それ以外は検査 packets を中継するルータを具備するように構成すれば、マルチキャスト中継中に途中経路の変更があっても、一定時間で検出し、適応することが可能になる。

【0033】

また、前記トランスミッタ及びルータで、 packets の元の宛先アドレスとして、未配送アドレスリストのノードのうち一つを格納するように構成すれば、マルチキャスト配送経路の途中に本発明による配送を介さないルータが存在しても、マルチキャスト中継が可能になる。

【0034】

また、自己が送信し、自己のインタフェースに到着した packets を検査し、自ノードが宛先リストに含まれている時に、これを受理するノードを具備するように構成すれば、マルチキャスト経路情報を持たず、ユニキャスト経路情報だけでマルチキャスト配送が可能となる。

【0035】

10

20

30

40

50

【発明の実施の形態】

本発明では、上記の問題を、従来のパケットに宛先が一つしか指定できなかったのを改め、複数の宛先アドレスのリストとリスト中で未配送なものを示すビットマップをパケットの拡張ヘッダとして添付することで解決するものである。

【0036】

以下、図面を参照して本発明の実施の形態例を詳細に説明する。

マルチキャストでデータを送信するサーバは、図2に示す形式のパケットを作成する。IPヘッダと異なる項目それぞれの意味は以下ようになる。

【0037】

(a) Destination Address (MSC address) このパケットがMSC (Multicast for Small Community) であることを示す特殊なIPアドレス。途中のルータが本発明のマルチキャスト方式でこのパケットを処理すべきであることを判別できるようにする。 10

【0038】

(b) MSC options

以降の処理バリエーションを確保する際にこの項目にオプションを指定する。

(c) # of dest

本パケットに含まれるアドレスのリストの長さ

(d) Destination Address # n

マルチキャストアドレスを配送すべきn個めの宛先アドレス 20

(e) destination bitmap

上記アドレスのリストのうち、本パケットがこれから到達すべき宛先のn番目を“1”、本パケットからは到達させる必要がないものは“0”で表わすビットマップ。最初の発進時は全て“1”とする。

【0039】

宛先アドレスが32を超えた場合には、32個目のアドレスの次に33個目から65個目までのアドレスのビットマップを挟んでアドレスのリストを続けて添付する。

【0040】

本実施の形態例のルータは、リストのアドレスのうちビットマップが“1”になっているものについて、ユニキャストのための経路表を検索する。検索した結果、次のルータ一つ一つについてパケットを転送する。この際、検索結果の次のルータが同じものについては、パケットを一つで済みますので、マルチキャストになる。中継パケットが更に中継されるべき宛先アドレスの位置を示すビットマップを“1”とし、他は“0”とする。 30

【0041】

このように構成すれば、マルチキャスト経路情報を持たずに、ユニキャスト経路情報だけでマルチキャスト配送が可能となる。

以上の中継によって、パケットは最終目的ノードまで到達し、このノードによって、リストが検査され、自己宛のアドレスがある場合は、これを受理し、より上位のプロトコルスタック(例:TCP)処理を行なう。この実施の形態例によれば、自己が送信し、自己のインターフェイスに到着したパケットを検査し、自ノードが宛先リストに含まれている時に、これを受理するノードを具備することにより、マルチキャスト経路情報を持たず、ユニキャスト経路情報だけでマルチキャスト配送が可能となる。 40

【0042】

以降における分岐正則とは、ルータにおける経路表検索を簡略化するための、前処理が施されているアドレスリストの性質である。分岐正則なアドレスリストとは、マルチキャスト経路の全てのルータにおいて、ビットマップが“1”となるノードが必ず連続することが保証されているような順番に整列しているものである。例えば、図3(一実施の形態例を示すブロック図)に示すように分岐するマルチキャストで、[a, b, c, d, e, f, g]というアドレスリストを考える。図1と同一のものは、同一の符号を付して示す。

【0043】

分岐深さは、a, f, g が 2、b, c, d, e が 3 であるものとする。発信者から d までの経路でルータでの分岐を 3 回通る。その間にビットマップは図 4 に示すように変化し、経路のどの点でもビットマップ 1 は連続している。これが全ての終点で成り立つので、[a, b, c, d, e, f, g] は分岐正則である。

【 0 0 4 4 】

一方、リスト [a, g, f, b, c, d, e] は、a に向かう最初の分岐で、ビットマップが [1, 0, 0, 1, 1, 1, 1] となり “ 1 ” が離れてしまう。従って、分岐正則ではない。

【 0 0 4 5 】

分岐正則であることを保証されている場合には、ビットマップで “ 1 ” となっている部分の両端にあるアドレスのみをまず経路表検索し、それが同じルータへ中継することになっていれば、その間にある宛先へも同じルータを経由して配送することが保証できるので、それについては経路表を検索することなしで次の中継先を決定することができる。

【 0 0 4 6 】

本実施の形態例は、同一パケットを複数の宛先に配送する際に、宛先アドレスのリストと未配送ビットマップをパケットヘッダに格納して送信するトランスミッタを具備することにより、中継ルータ群に対して、マルチキャスト経路情報設定依頼を行なうことなく、マルチキャストパケットの発信が可能となる。

【 0 0 4 7 】

このような形式でパケットを発信するトランスミッタと、これにより経路表検索を効率的に行なうルータを記述することができる。この方法を採用してマルチキャストを行なうために、トランスミッタはアドレスリストが分岐正則であることを示すビットを M S C オプションに設定し、ルータはこのビットが立っているかを検査して、経路表検索方法を切り替える。

【 0 0 4 8 】

この実施の形態例によれば、リストにおけるアドレス出現順位を、パケット配送経路上の任意のパスにおいて、未配送ビットマップの未配送部分が常に連続するように、予め整列され、配送済みであることを示す印を添えて送信するトランスミッタを具備することにより、分岐正則ルータが効率的にマルチキャストパケット配送を行なうことが可能となる。

【 0 0 4 9 】

また、前記トランスミッタが分岐正則印を添えて送信したパケットを中継する際に、未配送ビットマップで未配送となっている列の両端の 2 ノード分についてのみ経路表検索を行ない、分岐不要の場合には他のアドレスに関する経路検索を省略し、分岐が必要な場合のみ全宛先の経路表検索を行なうルータを具備することにより、マルチキャスト配送を行なう際に、請求項 1 のルータに比べて検索すべきユニキャストアドレスの数が少なく、パケット配送処理が小さくなる。

【 0 0 5 0 】

上記の分岐正則にリストを整列させるための仕組みについて説明する。トランスミッタは分岐正則としたいアドレスのリストを上位プロトコルデータが空のパケットに添付し、M S C オプションに経路探索依頼である印を添付して発信する。ルータは、請求項 1 の動作をしてパケットをマルチキャストすると同時に、若しパケットを分岐するならば、上位プロトコルデータ部分の最後尾に、どのように分岐を行なったかをビットマップとして追加記録する。

【 0 0 5 1 】

パケットが宛先クライアントに到達すると、トランスミッタからどのように分岐してきたかの履歴が記録されてきたことになる。これをトランスミッタにそのまま返信する。トランスミッタに全てのクライアントから分岐履歴が帰ってきたところで、分岐木を解析しリストを分岐正則に整列させる。

【 0 0 5 2 】

この実施の形態例によれば、アドレスリストを持つ分岐探索パケットを送信し、得られた

10

20

30

40

50

調査結果を元に分岐正則リストを作成するトランスミッタを具備することにより、分岐正則リストの自動的な作成が可能となる。また、管理者等が予め正則なリストを作成する必要がなくなる。

【0053】

また、トランスミッタが送信した経路探索パケットについて、経路表検索を行ない、分岐を行なう際には、自ルータまでの未配送経路リストの最後尾に未配送ビットマップを追加して探査パケットを中継し、分岐しない場合にはそのまま探査パケットを中継するルータを具備することにより、分岐正則リストの自動的な作成が可能になる。また、管理者等が予め正則なリストを作成する必要がなくなる。

【0054】

また、前記探査パケットが中継された宛先ノードで、探査パケットの未配送経路リストをそのままトランスミッタに返送するクライアントを具備することにより、分岐正則リストの自動的な作成が可能となる。また、管理者等が予め正則なリストを作成する必要がなくなる。

【0055】

また、分岐正則探査において、正則が自明となる部分分岐木についてはその先の探査をおこなわなくても分岐正則なリストが作成できることを利用して、探査処理を効率化する発明を記述することができる。

【0056】

探索を行わなくてもよい理由は、以下のアドレスの個数に関する数学的帰納法で示すことができる。

まず、アドレスが1個及び2個の場合には、自明に分岐正則である。

【0057】

次に、 $n - 1$ 個以下リストについて分岐正則なリストが作れるものとする。アドレスが3個以上（仮に n 個とする）である場合を考える。あるルータが n 個の宛先が未配送である探査パケットを受け取ったものとする。これを分岐させると、最小で2グループ、最大 n グループに枝分かれする。それぞれ未配送宛先は、最小で1個、最大で $n - 1$ 個含む。ここで、グループが含む宛先が1個のものとして2個のものの個数で分類すると、図5に示すようなものとなる。

【0058】

（あ）は、宛先が1個及び2個のグループがなく、3以上のグループへしか分岐していないことになる。但し、分岐はしているため、各グループは、 $n - 1$ 個以下になる。この場合は、履歴を追加更新し、全てのグループに探査パケットを中継することで、中継先から帰ってくるパケットで、分岐先の部分リストを分岐正則にすることができる。これを全てのグループについて連結すればよい。

【0059】

（い）は宛先1個のグループが1つあるが、宛先2個のグループがない場合である。宛先1個のグループは、それ自身が分岐正則であることは自明なので、中継しない。他のグループについて中継する。その結果、他のグループについて履歴が帰る。省略した1個のグループには履歴は帰らないが、他のグループの履歴を集めると、分岐は1つあったことがわかるので、これを他のグループの分岐正則と連結すればよい。

【0060】

（う）は同様であるが、省略したのが宛先が1つのグループ1つか、宛先が2個のグループ1つのどちらかであるが、どちらにしろ足りない2つの宛先を連結すれば分岐正則なので、これを他のグループの分岐正則リストに接続する。

【0061】

この実施の形態例によれば、経路表検索の結果、同じルータへ中継する宛先が1つ若しくは2つとなる経路がある場合、2つになる経路を1つまでか、1つになる経路を2つまで経路探査パケットの中継を省略するルータを具備することで、経路探索のためのパケット数が少なくなり、トランスミッタでの判断の処理が少なくなる。また、ネットワーク流量

10

20

30

40

50

を節約できる。更に、ルータ/クライアントの中継処理、折り返し処理が節約できる。

【0062】

また、ルータによって経路探索パケットが省略されていることを前提に、分岐木を解析してアドレスリストを分岐正則に整列させるルータを具備することにより、経路探索のためのパケット数が少なくなり、トランスミッタでの判断の処理が少なくなる。また、ネットワーク流量を節約できる。更に、ルータ/クライアントの中継処理、折り返し処理が節約できる。

【0063】

また、宛先を変更してアドレスを追加/削除することができるトランスミッタを実現することができる。これによれば、一連のパケット送信の途中で宛先アドレスの追加/削除ができるトランスミッタを具備することにより、マルチキャスト配送中のチャンネルに途中から参加/脱退が可能になる。

10

【0064】

また、分岐正則を利用して効率化している状況で、前記トランスミッタ動作をを行なう際に分岐正則化を全てに対してもう一度やり直すトランスミッタである。この実施の形態例によれば、追加したアドレスに関する経路探索を行ない、アドレスを分岐正則に保つトランスミッタを具備することにより、マルチキャスト配送中のチャンネルに、途中から参加/脱退しても分岐正則による配送の効率化が継続する。

【0065】

また、ルータがパケットを分岐する際に、ビットマップを“0”にしたアドレスについては内容をクリアし、途中経路や終点等のホストが他にどこにマルチキャストしたかを隠蔽することができる。この発明によれば、マルチキャストに参加している他のクライアントに対して、自身が参加していることを秘匿できる。

20

【0066】

また、一度構築した正則リストの有効性を検査し、ルーティング環境が変わった際でも適応可能にする仕組みを実現することができる。トランスミッタが、要検査をオプションに設定にすると、経路の各ルータは正則性をチェックする。崩れていなければ履歴と共に発信元に送り返す。トランスミッタはこれを受けて、必要な部分リストについては改めて正則化を行なう。

【0067】

この実施の形態例によれば、定期的にアドレスリストが分岐正則かをチェックする正則検査パケットを送信し、非正則通知がくると正則化をやり直すトランスミッタを具備することにより、マルチキャスト中継中に途中経路の変更があっても、一定時間で検出し、適応することが可能になる。

30

【0068】

また、前記トランスミッタが出す正則検査パケットのアドレスリストを経路表検索し、正則性が崩れていた場合に非正則通知を返送し、それ以外は検査パケットを中継するルータを具備することにより、マルチキャスト中継中に途中経路の変更があっても、一定時間で検出し、適応することが可能になる。

【0069】

また、中継途中にM S Cに対応しないルータがあった場合にもM S Cが通るようにI P v 6ヘッダの宛先をM S Cを表わすアドレスを入れなくて、M S Cリストのうち、未配送のアドレスの一つを入れておくことができる。非M S Cルータは、これを通常のユニキャストと同様に中継するが、途中か終点でM S Cを解釈するルータがこれを分岐できる。

40

【0070】

この実施の形態例によれば、トランスミッタ及びルータで、パケットの元の宛先アドレスとして、未配送アドレスリストのノードのうち一つを格納することにより、マルチキャスト配送経路の途中に本発明による配送を介さないルータが存在しても、マルチキャスト中継が可能になる。

【0071】

50

図6は本発明の他の実施の形態例を示すブロック図である。この図では、ルータ1同志が専用線6を介して接続され、各ルータには、ノードa~hが接続されている。以下、このネットワークを用いて実施の形態例を説明する。このネットワークでは、ホスト8台(a~h)、4本のイーサネット(W~Z)、1本の専用線6、ルータR1、R2の2台からなる。

【0072】

ホストのIPアドレスは、それぞれa~hとし、ルータのインターフェイス各々のIPアドレスは、専用線がr1、r2、イーサネット側がw、x、y、zとする。この時、a、bは図7に示すような経路表を持つ。

【0073】

同様に、c、dはW、wをそれぞれX、xに、e、fはY、yに、g、hはZ、zに変えた経路表となる。

一方、ルータR1、R2の経路表は図8に示すようなものとなる。

【0074】

(基本動作の実施の形態例)

まず、実施の形態例を示す。マルチキャストパケットの発信者をaとし、受信者をb、c、d、e、f、g、hとする。a上のプロセスは、socketインターフェイスを経由してメッセージ送信を依頼する。

【0075】

sendto(socket, msg, length, flag, dest, destlen)
destは本発明のために新たに定義する宛先アドレス指定構造体(struct sock_addr_msc)で以下のように宛先リストを格納する。

【0076】

```
struct sock_addr_msc {
short          smsc_family;          /* AF MSC */
unsigned int   smsc_list_desc;      /* regular list desc */
char           smsc_nnodes;         /* # of scst nodes */
struct inet_addr v6_smc_addrs[MAX_ADDR_LIST] /* dest address list */
/
}
```

ここで、smc_familyはmscのために定義するアドレスファミリの定数値である。

【0077】

smc_list_descは、以降の分岐正則で使用するリスト指示子である。
smc_nnodesは、このリストに含まれる宛先ノードの数である。
smcs_addrsはIPv6アドレスの配列である。

【0078】

ノードaは、これに対して以下の属性を持つパケットを発信する。

```
IPv6 src      =a
IPv6 dst      =MSC
IPv6 opt      =MSC followed
  MSC option   =None
RouteType     =MSC
# of dest     =7
bitmap        =[1,1,1,1,1,1,1]
dest_addr     =[b,c,d,e,f,g,h]
```

パケットは、具体的には図9に示すようなものとなる。

【0079】

aは自身の経路表を使用して[b, c, d, e, f, g, h]を検索し、次の結果を得る。

b: 自身の所属するイーサネットwを通じて直接配送

10

20

30

40

50

c, d, e, f, g, h: w を経由して中継

b宛に直接パケットを送る際、b i t m a p (ビットマップ) を以下のように変更する。

【 0 0 8 0 】

bitmap = [1, 0, 0, 0, 0, 0, 0]

dest addr = [b, c, d, e, f, g, h]

w宛には以下のパケットを送付する。

【 0 0 8 1 】

bitmap = [0, 1, 1, 1, 1, 1, 1]

dest addr = [b, c, d, e, f, g, h]

bはこのパケットを見て、先ずI P v 6ヘッダがM S Cを表わすことを検査し、o p tヘッダを見て、n o o p t i o nであることを確認し、r o u t i n gヘッダのビットマップを検査する。“ 1 ” が立っているアドレスが自分のアドレスなので、これを受理し、上位プロトコルに渡す。 10

【 0 0 8 2 】

ルータR 1はこのパケットを見て、先ずI P v 6ヘッダがM S Cを表わすことを検査し、o p tヘッダを見てn o オプションであることを確認する。次に、r o u t i n gヘッダのビットマップを検査する。“ 1 ” が立っているアドレスそれぞれについて、自分のアドレスでないことを確認した後、経路表を検査して以下の結果を得る。

【 0 0 8 3 】

c, d, g, h: r 2 を経由して配送 20

e, f: イーサネットY 経由で直接配送

これにより、c, d, g, h 用に r 2 に対して

bitmap = [0, 1, 1, 0, 0, 1, 1]

dest addr = [b, c, d, e, f, g, h]

e には、

bitmap = [0, 0, 0, 1, 0, 0, 0]

dest addr = [b, c, d, e, f, g, h]

f には、

bitmap = [0, 0, 0, 0, 1, 0, 0]

dest addr = [b, c, d, e, f, g, h] 30

を送付する。

【 0 0 8 4 】

(分岐正則時の中継動作例)

次に、分岐正則に関する動作として、一実施の形態例を示す。ノード a から宛先 [c, d, g, h] へ送信する場合を考える。請求項 1 の場合では、ルータ R 1 は次のパケットを中継する。

【 0 0 8 5 】

bitmap = [1, 1, 1, 1]

dest addr = [c, d, g, h]

このため、4つの宛先が全て同じ中継先 r 2 に行くにも拘らず c, d, g, h の全てを経路表検索する必要があった。ここでは、c, h のみの検索ですむようにする。 40

【 0 0 8 6 】

先ず、アドレス出現順位を実装したトランスミッタは、宛先アドレスを具備したトランスミッタと同様にパケットを組み立てるが、M S C o p t i o n に、定数 M S C R E G U L A R E D を記す。

【 0 0 8 7 】

分岐が必要な場合のみ全宛先の経路表検索を行なう処理を実装したルータ R 1 は、I P v 6 宛先アドレスが M S C であることを確認した後、M S C オプションに M S C R E G U L A R E D が記されているのを確認する。この場合は、ビットマップの両端 c, h のみを経路検索する。両方共に r 2 経由であるので、d, g に関しては検索せず、r 2 へ中継す 50

る。

【 0 0 8 8 】

一方、分岐が必要な場合のみ全宛先の経路表検索を行なう処理を実装したルータ R 2 は、同様に c , h を検索する。すると、今度は c は X、h は Z のインターフェイスへ中継する必要があることがわかる。従って、d , g も検索を行ない、d は X、h は Z に中継する。

【 0 0 8 9 】

(分岐正則探査の実施の形態例)

上記の分岐正則リストを作成するための例を説明する。全体の流れは以下の通りである。

【 0 0 9 0 】

プログラム：宛先リストを分岐正則にするように依頼。 10
 トランスミッタ：宛先リストを含む探査パケットを発信する。
 ルータ：探査パケットを中継しながら分岐履歴を記録する。

【 0 0 9 1 】

クライアント：到着した探査パケットを分岐履歴毎にトランスミッタに送り返す。
 トランスミッタ：分岐履歴から分岐正則リストを作成する。分岐正則リストに対する識別子を返還。

【 0 0 9 2 】

プログラム：識別子を宛先につけてパケット送信。
 トランスミッタは、以下の情報を持つ探査パケットを作成し、本発明に係る方法で発信する。 20

【 0 0 9 3 】

```
IPv6 src    =a
IPv6 dst    =MSC
IPv6 opt    =MSC followed
MSC option  =MSC branch inquiry
RoutingHdr NextHdr=ICMP
Route Type  =MSC
# of dest   =7
bitmap      =[1,1,1,1,1,1,1]
dest addr   =[b,c,d,e,f,g,h]
ICMP Type   =MSC Branch Inquiry
ICMP Code   =None
ICMP bitmap len=0
```

ICMP Identifier=MSC inquiryをする毎に一意になる I D

この時のパケットは、図 1 0 に示すようなものとなる。

ルータは、請求項 1 と同様な中継動作をしつつ、中継すべきインターフェイスが 2 つ以上あった場合、即ち分岐が発生する場合には、I C M P ヘッダを以下のように更新する。

【 0 0 9 4 】

#of bitmapを 1 加算
 新しい b i t m a p を I C M P ヘッダ末尾に追加 40
 ルータ R 1 で e , f に対して中継する際は、I C M P ヘッダが図 1 1 に示すように変化する。

【 0 0 9 5 】

e はこれを受けとり、a に対して R o u t i n g ヘッダ以下のコピーを以下のように返送する。

```
IPv6 src    =e
IPv6 dst    =a
IPv6 opt    =None
IPv6 NextHdr=ICMP
ICMP Type   =MSC Branch record
```

50

ICMP Code =None

以下、Routingヘッダ以降のコピー

この時のパケットは具体的には、図12に示すようなものとなる。

この場合、同様のパケットが全てのクライアントから帰ってくる。前述のトランスミッタは、Identifierが同一であるパケットを集める。今回の例では、以下の履歴が返信されてくる。

【0096】

[b, c, d, e, f, g, h]

b: [1, 0, 0, 0, 0, 0, 0]

10

c: [0, 1, 1, 1, 1, 1, 1]

[0, 1, 1, 0, 0, 1, 1]

[0, 1, 1, 0, 0, 0, 0]

[0, 1, 0, 0, 0, 0, 0]

d: [0, 1, 1, 1, 1, 1, 1]

20

[0, 1, 1, 0, 0, 1, 1]

[0, 1, 1, 0, 0, 0, 0]

[0, 0, 1, 0, 0, 0, 0]

e: [0, 1, 1, 1, 1, 1, 1]

[0, 0, 0, 1, 1, 0, 0]

[0, 0, 0, 1, 0, 0, 0]

30

f: [0, 1, 1, 1, 1, 1, 1]

[0, 0, 0, 1, 1, 0, 0]

[0, 0, 0, 0, 1, 0, 0]

```

g: [0,1,1,1,1,1,1]
    [0,1,1,0,0,1,1]
    [0,0,0,0,0,1,1]
    [0,0,0,0,0,1,0]

```

```

h: [0,1,1,1,1,1,1]
    [0,1,1,0,0,1,1]
    [0,0,0,0,0,1,1]
    [0,0,0,0,0,0,1]

```

10

これから次の手順で、分岐正則リストを作成する。

【 0 0 9 7 】

```

struct regular list*make regular list()[ /*レギュラーリストを作るには、*/
struct tree tree;

```

```

tree =make tree(); /*分岐木を作成して */

```

```

return make list(tree) /*分岐木をリストにする。*/

```

20

/*分岐木は図 1 3 に示すような形をしている。図において、 は分岐するルータを表わす。ルータの横に (e , f) 等とつくのは、そこで分岐していることはわかるが、その下の分岐構造が未決定のノードのリストである。

*/

```

make tree()[ /* 分岐木を作成する。 */

```

```

分岐木のルートになるルータを作り、

```

```

未決定ノードリストに全てのノードを入れる。

```

【 0 0 9 8 】

```

/*o(b,c,d,e,f,g,h) */

```

```

for(i=0;i<# of node;i++){ /* 全てのノードについて*/

```

30

```

node iから帰ってきた履歴を見る。

```

【 0 0 9 9 】

```

depth=node iの分岐深さ。 /*bだと1、hだと4 */

```

```

for(j=node iがついているルートの深さ。 depth;j>0;j++)

```

```

node iがいるルータからルータを一つ作成して伸ばす。

```

【 0 1 0 0 】

```

新しく作ったルータにnode iを移動。

```

```

]

```

```

今、node iがいるルータにeをぶら下げる。

```

【 0 1 0 1 】

40

```

for(j=# of bitmap ;j>0;j--) /* 全ての履歴について*/

```

```

foreach(j番目とj-1番目の履歴で変化したノード)

```

```

if(ノードは未決定)

```

```

node iまでのルータ系列で深さ j - 1 のルータに未決定ノードとして登録。 ]

```

```

make regular list(tree)[

```

```

treeを深さ優先に探索し、ノードを列挙。

```

```

]

```

上記手続きを例に当てはめると、図 1 4 に示すように分岐木が生成できる。

(効率的分岐正則探索の実施の形態例)

分岐探索で、クライアントの全てから、履歴が帰ってくる必要はない。例えば図 1 5 に示

50

す分岐木の解析途中の経路木 (h) は h の履歴を待たずに g の横に置くことができるのは自明である。そこで、このような場合に h への探査パケット中継を抑制することで探査を効率化する。

【 0 1 0 2 】

次に、他の実施の形態例を以下に説明する。

まず、トランスミッタは、探査パケットを用意するが、M S C オプションとして e f f e c t b r a n c h i n q u i r y を指定する。

【 0 1 0 3 】

また、ルータは、この探査パケットを中継する際には、分岐しない場合にそのまま探査パケットを中継するルータと同様な動作をするが、中継先インターフェイス毎のグループを

10

【 0 1 0 4 】

(あ) の場合には全ての中継を行なう。

(い) の場合で、省略することで中継先がなくならないならば、宛先 1 個の 1 グループについては中継を省略する。

【 0 1 0 5 】

(う) の場合には、省略することで中継先がなくならないならば、宛先 1 個の 2 つのグループ、若しくは、宛先 2 個のグループを 1 つ選択して中継を省略する。

【 0 1 0 6 】

前記トランスミッタでは、経路解析手段が以下のように変わる。

20

```
make tree()[ /* 分岐木を作成する。 */
```

```
分岐木のルートになるルータを作り
```

```
未決定ノードリストに全てのノードを入れる。
```

```
【 0 1 0 7 】
```

```
/* o(b,c,d,e,f,g,h) */
```

```
for(i=0;i<# of node;i++) [ /* 全てのノードについて */
```

```
node i から帰ってきた履歴を見る。
```

```
【 0 1 0 8 】
```

```
depth=node i の分岐深さ。 /* b だと 1、h だと 4 */
```

```
for(j=node i がついているルートの深さ、depth;j>0;j++)
```

30

```
node i がいるルータからルータをルータ 1 つ作って伸ばす。
```

```
【 0 1 0 9 】
```

```
新しく作ったルータに node i を移動。
```

```
]
```

```
今、node i がいるルータに e をぶら下げる。
```

```
【 0 1 1 0 】
```

```
for(j=# of bitmap;j>0;j--) /* 全ての履歴について */
```

```
foreach( j 番目と j-1 番目の履歴で変化したノード)
```

```
if(ノードは未決定)
```

```
node i までのルータ系列で深さ j-1 のルータに未決定ノードとして登録。
```

40

```
foreach(全てのルータについて) /* 前記処理で増えたところ */
```

```
if(未決定ノードが 2 以下なら)
```

```
未決定ノードをルータに直接ぶら下げる。
```

```
]
```

```
[b,c,d,e,f,g,h] では h , c , e , b への中継が省略できる ( 図 1 7 参照 ) 。
```

(宛先変更)

分岐正則をプログラムが指示し、正則化が終わると、識別子が帰る。連続してマルチキャストパケットを流す時には、前の正則リストを流用することで、再度正規化を行なわないことにする。しかしながら、クライアントが受信を中断したり、新しいクライアントが加わると、正則化をやり直す必要がある。

50

【 0 1 1 1 】

また、この際に始めから正規化をやり直すトランスミッタを設けることができる。正規化処理自体は、前述のものと全く同じである。

(分岐先隠蔽)

中継ルータで分岐先ノード以外をクリアし、他のクライアントに自分ノードへの通信先を隠蔽するルータである。請求項1のルータと動作はほぼ同じであるが、b宛に直接パケットを送る際、ビットマップを変更する際にアドレスもクリアする。

【 0 1 1 2 】

bitmap = [1,0,0,0,0,0,0]

dest addr = [b,0,0,0,0,0,0]

10

w宛には以下のパケットを送付する。

【 0 1 1 3 】

bitmap = [0,1,1,1,1,1,1]

dest addr = [0,c,d,e,f,g,h]

(正則検査)

ネットワークのトポロジ変化や障害に対応するために正則性を定期的に検査を行なえるようにする。次に、他のトランスミッタの実施の形態例を以下に説明する。

【 0 1 1 4 】

IPv6 src = a

IPv dst = MSC

20

IPv opt = MSC followed

MSC option = MSC check regular

RoutingHdr NextHdr=ICMP

RoutType = MSC

of dest = 7

bitmap = [1,1,1,1,1,1,1]

dest addr = [b,c,d,e,f,g,h]

ICMP Type = MSC check regular

ICMP Code = None

ICMP bitmap len=0

30

ICMP Identifier=MSC inquiryをする毎に一意になるID

パケットは具体的には、図18に示すようなものになる。

【 0 1 1 5 】

一方、これを中継するルータは、MSCオプションがcheck regularであるので、ビットマップが“1”のノードを全て経路検索する。ここで、全てが同じ経路であるか、若しくは両端のノードが異なる中継先になるなら、通常のMSC中継を行なう。若し、両端が同じ中継先になるにも拘らず、その間に異なる中継先のノードがあれば、以下のパケットを送信先に返信する。

【 0 1 1 6 】

IPv6 src = router addr

40

IPv6 dsc = a

IPv6 NextHdr=ICMP

ICMP Type = MSC not regular

ICMP Code = None

ICMP Identifier=MSC inquiryをする毎に一意になるID

これに異常が起こったビットマップで異常なノードが“0”になるように設定して送り返す。パケットは、具体的には図19に示すようなものとなる。トランスミッタは、これを受けて、正規化を再度行なう。

(非MSCルーティング)

IPv6パケットのdestにMSCルーティングヘッダリストの中でビットマップが“1

50

”である最初のアドレスを入れる。このようにすると、途中でM S C非対応ルータがあっても、宛先ノードのうちの一つに向かって中継が継続される。途中でM S C対応ルータがあるか、宛先ノードがM S C対応ならばここで、更に別の宛先に改めて中継が起こる。

【0117】

以上、説明したように、本実施の形態例によれば、以下の利点を得ることができる。

1 小規模のグループ内での同報通信を、インターネット上のルータにマルチキャストのための特別な経路管理の負担なく行なうことができる。

2 送信者が経路トポロジを探索し、ルータでの経路検索コストを低くするように宛先情報を整理できる。

3 クライアントのマルチキャストグループへの加入/脱退がルータの関与なしに可能になる。その際も 2 の利点を失わない。 10

4 経路途中で本発明の機構を実現しないルータが存在しても、上記の利点を享受することができる。

【0118】

【発明の効果】

以上、説明したように、本発明によれば、以下のような効果が得られる。

(1) 請求項1の発明によれば、転送されるパケットヘッダに複数の宛先アドレスリストとその未配送ビットマップを持つパケットを、ユニキャスト経路に従って中継する手段を有することにより、マルチキャスト経路情報を持たずに、ユニキャスト経路情報だけでマルチキャスト配送が可能となる。 20

また、未配送ビットマップで未配送となっている列の両端の2ノード分について経路表検索を行ない、分岐不要の場合には、他のアドレスに関する経路探索を省略し、分岐が必要な場合のみ全宛先の経路表検索を行なうことにより、マルチキャストパケット配送を行なう際に、検索すべきユニキャストアドレスの数が少なく、パケット配送処理が小さくなる。

(2) 請求項2の発明によれば、パケットを分岐させる際に、配送済みとなったアドレスについてはアドレスとして意味のない値に変更してから配送することにより、マルチキャストに参加している他のクライアントに対して、自身が参加していることを秘匿できる。

【0119】

このように、本発明によれば、アドレス設定とルータ設定を容易にしたマルチキャスト配送システムを提供することができる。 30

【図面の簡単な説明】

【図1】本発明の原理ブロック図である。

【図2】パケットの構成例を示す図である。

【図3】本発明の一実施の形態例を示すブロック図である。

【図4】ビットマップの推移を示す図である。

【図5】グループ分岐の説明図である。

【図6】本発明の他の実施の形態例を示すブロック図である。

【図7】a, bの経路を示す図である。

【図8】R1, R2の経路を示す図である。 40

【図9】パケットの構成例を示す図である。

【図10】パケットの構成例を示す図である。

【図11】ICMPの構成例を示す図である。

【図12】パケットの構成例を示す図である。

【図13】分岐木の構成例を示す図である。

【図14】分岐木の生成を示す図である。

【図15】解析途中の経路木を示す図である。

【図16】中継先インターフェイスごとのグループの場合分けを示す図である。

【図17】h, c, e, bへの中継省略の説明図である。

【図18】パケットの構成例を示す図である。 50

【図19】パケットの構成例を示す図である。

【図20】IPv6のヘッダフォーマットの構成図である。

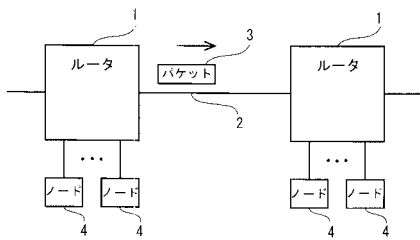
【図21】検索表の構成を示す図である。

【符号の説明】

- 1 ルータ
- 2 専用線
- 3 パケット
- 4 ノード

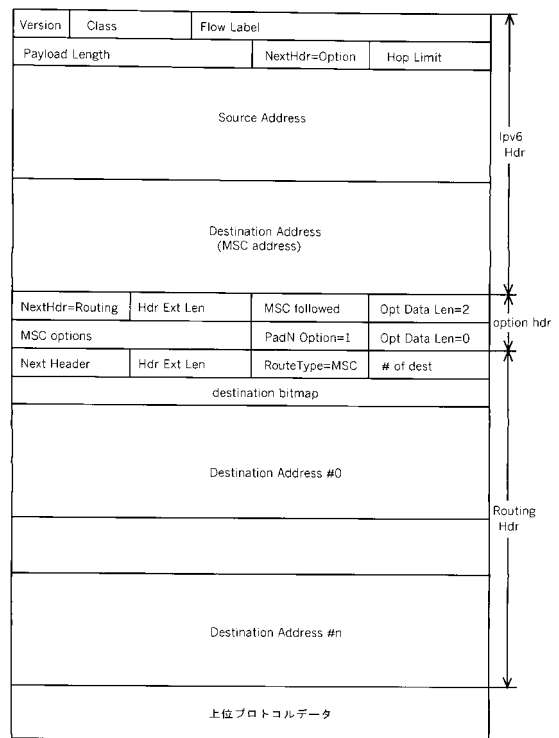
【図1】

本発明の原理ブロック図



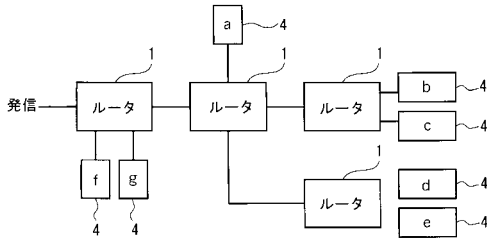
【図2】

パケットの構成例を示す図



【 図 3 】

本発明の一実施の形態例を示すブロック図



【 図 4 】

ビットマップの推移を示す図

```
[ a . b . c . d . e . f . g ]
=====
[ 1 . 1 . 1 . 1 . 1 . 0 . 0 ]
[ 0 . 0 . 0 . 1 . 1 . 0 . 0 ]
[ 0 . 0 . 0 . 0 . 1 . 0 . 0 ]
```

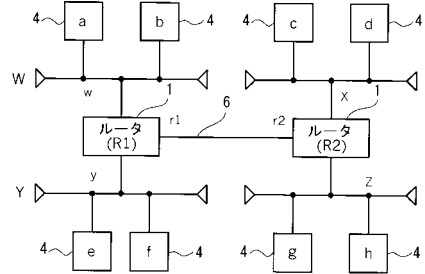
【 図 5 】

グループ分岐の説明図

宛先 2 個 \ 宛先 1 個	なし	1 グループ	2 グループ以上
なし	(あ)	(い)	(う)
1 グループ以上	(う)	(う)	(う)

【 図 6 】

本発明の他の実施の形態例を示すブロック図



【 図 7 】

a, b の経路を示す図

net work	gateway	interface
loopback	-	loopback
W	-	ethernet (W)
default	w	ethernet

【 図 8 】

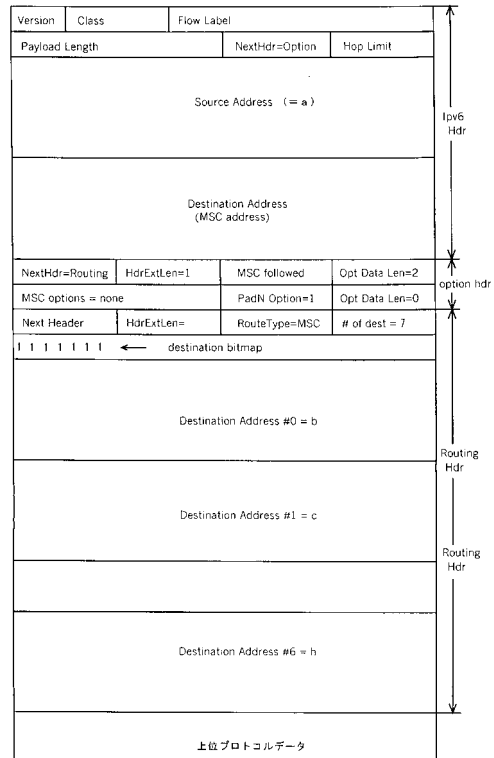
R1, R2 の経路を示す図

[R1]	net work	gateway	interface
loopback	-	loopback	
W	-	ethernet (W)	
X	r2	r1	
Y	-	ethernet (Y)	
Z	r2	r1	

[R2]	net work	gateway	interface
loopback	-	loopback	
W	r1	r2	
X	-	ethernet (X)	
Y	r1	r2	
Z	-	ethernet (Z)	

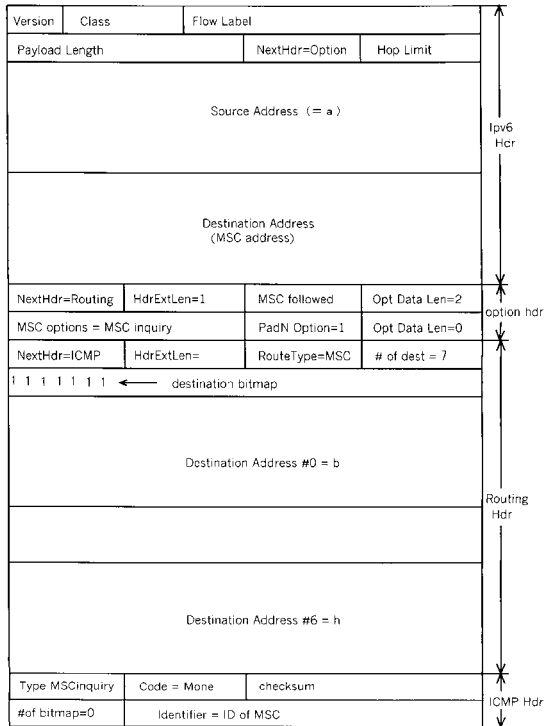
【 図 9 】

パケットの構成例を示す図



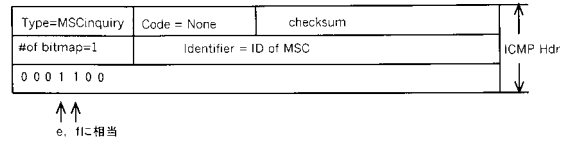
【 図 1 0 】

パケットの構成例を示す図



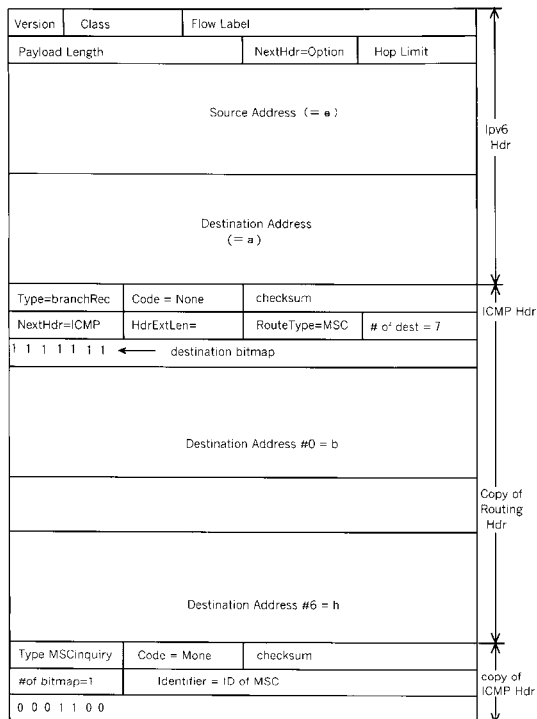
【 図 1 1 】

ICMPの構成例を示す図



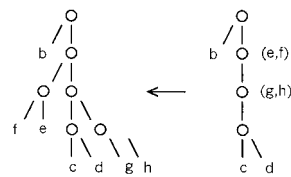
【 図 1 2 】

パケットの構成例を示す図



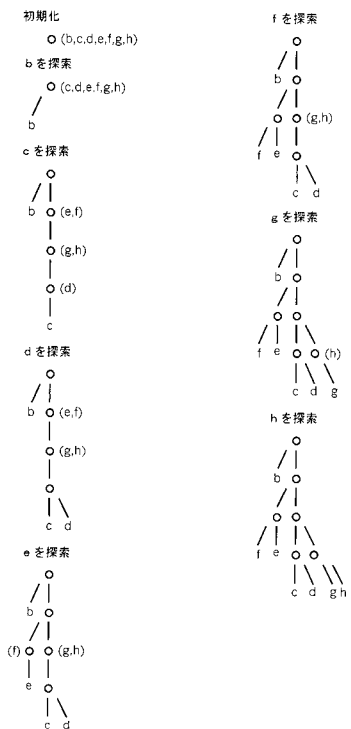
【 図 1 3 】

分岐木の構成例を示す図



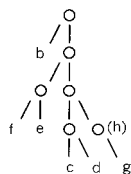
【 図 1 4 】

分岐木の生成を示す図



【 図 1 5 】

解析途中の経路木を示す図



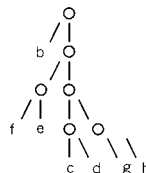
【 図 1 6 】

中継先インタフェースごとのグループの場合分けを示す図

宛先2個\宛先1個	なし	1グループ	2グループ以上
なし	(あ)	(い)	(う)
1グループ以上	(う)	(う)	(う)

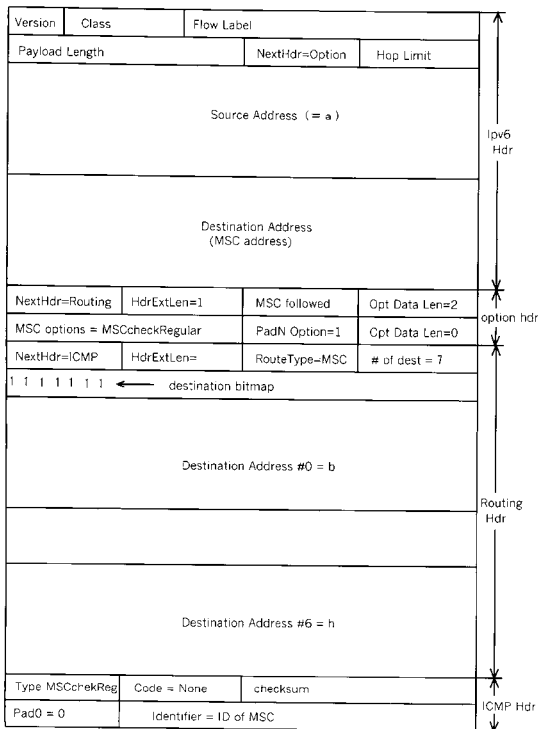
【 図 1 7 】

h, c, e, bへの中継省略の説明図



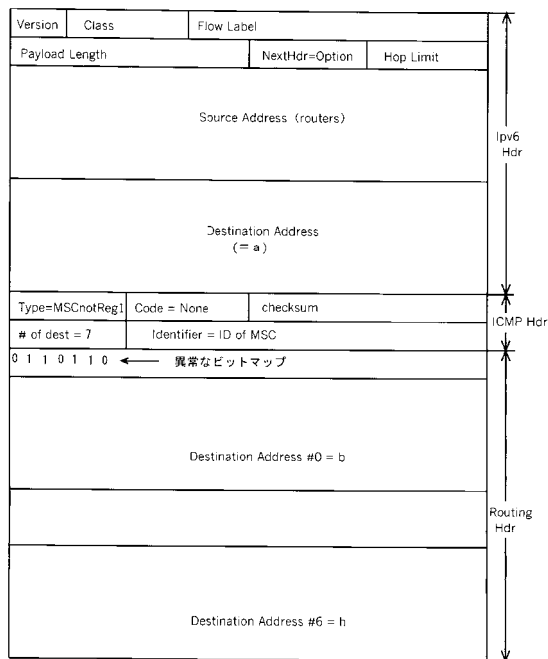
【 図 1 8 】

パケットの構成例を示す図



【 図 1 9 】

パケットの構成例を示す図



【 図 2 0 】

I P V 6 のヘッダフォーマットの構成図

Version	Class	Flow Label	
Payload Length		Next-Header	Hop Limit
Source Address			
Destination Address			
上位プロトコルデータ			

【 図 2 1 】

検路表の構成を示す図

Network		Destination	
network prefix	net mask	neighbor	interface
3FFE:501:1000::	40	12345678	le0
3FFE:501:1100::	40	12345678	ppp0
⋮	⋮	⋮	⋮
3FFE:501:1FFF::	44	12345678	le0
FF02::1	128	12345678 87654321	le0 ppp1
FF02::2	128	12341111 12342222 12343333	le0 ppp1 ppp2
FF02::3	128	12341111	le0

↑ エキスト部
↓ エキスト部

フロントページの続き

審査官 清水 稔

- (56)参考文献 特開平10 - 098464 (JP, A)
特開平06 - 326713 (JP, A)
特開平10 - 210067 (JP, A)

- (58)調査した分野(Int.Cl. , DB名)
H04L 12/56
H04M 3/00