(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第3615057号
(P3615057)
(45)発行日　平成17年1月26日(2005.1.26)　(24)登録日　平成16年11月12日(2004.11.12)

(51)Int.Cl.[7]	F I		
H04L 12/56	H04L	11/20	102D
H04L 12/28	H04Q	3/00	
H04Q 3/00	H04L	11/00	310Z
	H04L	11/20	G

請求項の数 21　(全 34 頁)

(21)出願番号　特願平10-203667
(22)出願日　平成10年7月17日(1998.7.17)
(65)公開番号　特開2000-36836(P2000-36836A)
(43)公開日　平成12年2月2日(2000.2.2)
審査請求日　平成14年7月19日(2002.7.19)

(73)特許権者　000003078
株式会社東芝
東京都港区芝浦一丁目１番１号
(74)代理人　100058479
弁理士　鈴江　武彦
(74)代理人　100084618
弁理士　村松　貞男
(74)代理人　100068814
弁理士　坪井　淳
(74)代理人　100092196
弁理士　橋本　良郎
(74)代理人　100091351
弁理士　河野　哲
(74)代理人　100088683
弁理士　中村　誠

最終頁に続く

(54)【発明の名称】ラベルスイッチングパス設定方法及びノード装置

(57)【特許請求の範囲】
【請求項１】
あるパケットフロー用のラベルスイッチングパスの設定に関わるメッセージであって、該パケットフローを識別可能なフロー識別子、及び、該パケットフローに関して処理中であることを示す情報を含むものを受信したノードが、メッセージ処理中として該ノード内に記憶されている情報を対象として、該ラベルスイッチングパスがループを形成するか否かを判断し、
前記メッセージを受信したノードが、前記ラベルスイッチングパスがループを形成しないとは判断できない場合には、前記フロー識別子を、メッセージ処理中として該ノード内に記憶すると共に、前記メッセージを前記パケットフローの経路上のノードへ転送し、
前記メッセージを受信したノードが、前記ラベルスイッチングパスがループを形成しないことを確認できた場合には、前記メッセージに対する返答メッセージを返送し、
この返答メッセージを最新の返答として受信したノードは、該返答メッセージを、前記メッセージを送信／転送しメッセージ処理中となっているノードへ転送し、
前記返答メッセージを最新の返答として受信したノードは、該返答メッセージに対応する、メッセージ処理中として該ノード内に記憶されている情報が、以降は前記判断の対象とならないように、メッセージ処理中としての記憶を削除し、前記ラベルスイッチングパスを利用可能とすることを特徴とするラベルスイッチングパス設定方法。
【請求項２】
前記メッセージを受信したノードが、前記ラベルスイッチングパスがループを形成すると

判断した場合には、前記メッセージの転送は行わず、前記返答メッセージも返送しないことを特徴とする請求項１に記載のラベルスイッチングパス設定方法。

【請求項３】

前記判断においては、受信した前記メッセージに含まれるのと同一のフロー識別子についての情報を、メッセージ処理中として該ノード内に記憶しているか否かに基づいて、前記ラベルスイッチングパスがループを形成するか否かを判断することを特徴とする請求項１に記載のラベルスイッチングパス設定方法。

【請求項４】

前記メッセージには、前記パケットフローに関する処理中のメッセージをネットワークグローバルに識別可能なメッセージ識別子も含まれることを特徴とする請求項１に記載のラベルスイッチングパス設定方法。

【請求項５】

前記判断においては、受信した前記メッセージに含まれるのと同一のフロー識別子及びメッセージ識別子をメッセージ処理中として該ノード内に記憶しているか否かに基づいて、前記ラベルスイッチングパスがループを形成するか否かを判断することを特徴とする請求項４に記載のラベルスイッチングパス設定方法。

【請求項６】

前記メッセージは、設定要求されたラベルスイッチングパスの初段のノードから該メッセージを受信したノードまでのホップ数を求めることが可能な情報を含むものであり、

前記メッセージを受信したノードが、このメッセージに含まれる情報に基づき前記ホップ数が所定値を超えると判断した場合は、前記ラベルスイッチングパスがループを形成すると判断した場合と同様に扱うことをことを特徴とする請求項１または２に記載のラベルスイッチングパス設定方法。

【請求項７】

前記メッセージを受信したノードは、自ノードが設定要求されたラベルスイッチングパスの終段のノードである場合、もしくは、自ノードの下流に対応する確立されたラベルスイッチングパスが存在しこれに設定要求されたパスを接続して良い場合に、前記ラベルスイッチングパスがループを形成しないことを確認できたものとすることを特徴とする請求項１に記載のラベルスイッチングパス設定方法。

【請求項８】

設定しようとするラベルスイッチングパスの初段のノード、もしくは、確立されたラベルスイッチングパスの中段のノードで経路変更が起こった場合のそのノード、もしくは、確立されたラベルスイッチングパスの終段のノードで下流隣接ノードを新たに認識したノードが、前記メッセージを能動的に送信することを特徴とする請求項１に記載のラベルスイッチングパス設定方法。

【請求項９】

前記メッセージ識別子は、該メッセージ識別子を割り当てたノードの識別情報を含むことを特徴とする請求項１に記載のラベルスイッチングパス設定方法。

【請求項１０】

前記返答メッセージの返送の際に、ラベルスイッチングのためのリンク情報に含まれるラベルの割り当てを行うことを特徴とする請求項１に記載のラベルスイッチングパス設定方法。

【請求項１１】

前記メッセージを受信したあるノードで前記ラベルスイッチングパスがループを形成すると判断された場合に、前記メッセージを能動的に送信したノードは、前記ラベルスイッチングパスの設定の必要が無くなった場合に、前記メッセージに対する取消メッセージを送信し、前記メッセージを転送した各ノードは、対応する前記取消メッセージを受信するまでは、メッセージ処理中としての記憶を保持しておくことを特徴とする請求項１に記載のラベルスイッチングパス設定方法。

【請求項１２】

前記メッセージを受信したノードは、設定しようとするラベルスイッチングパスを同じくするメッセージが処理中であることが記憶されており、該ラベルスイッチングパスがループを形成するか否かを判断できない場合には、受信したメッセージと処理中のメッセージとを一つのメッセージとして扱うことを特徴とする請求項１に記載のラベルスイッチングパス設定方法。

【請求項１３】

前記メッセージは、設定要求されたラベルスイッチングパスの初段のノードから該メッセージを受信したノードまでのホップ数を求めることが可能な情報を含むものであり、

前記メッセージを受信したノードは、前記ラベルスイッチングパスがループを形成すると判断した場合に、該ラベルスイッチングパスの初段のノードから自ノードまでのホップ数に関する情報を修正することを特徴とする請求項１に記載のラベルスイッチングパス設定方法。

【請求項１４】

ラベルスイッチングパスを設定するノード装置であって、

あるパケットフロー用のラベルスイッチングパスの設定に関わる要求メッセージであって、該パケットフローを識別可能なフロー識別子、及び、該パケットフローに関して処理中であることを示す情報を含むものを受信した場合に、未確定として記憶手段内に記憶されている情報を対象として、該ラベルスイッチングパスがループを形成するか否かを判断する判断手段と、

前記判断手段により前記ラベルスイッチングパスがループを形成しないことが確認できない場合には、前記フロー識別子を、未確定として前記記憶手段内に記憶すると共に、前記要求メッセージを前記パケットフローの経路上のノードへ転送し、前記判断手段により前記ラベルスイッチングパスがループを形成しないことが確認できた場合には、前記要求メッセージに対する返答メッセージを返送する第１の処理手段と、

前記要求メッセージに対する返答メッセージを最新の返答として受信した場合に、現在は未確定として前記記憶手段内に記憶されている情報が、以降は前記判断の対象とならないように、該情報を確定として記憶するように変更し、該返答メッセージを、前記要求メッセージを転送し現在未確定として情報を記憶している他のノードへ転送して、前記ラベルスイッチングパスを利用可能とする第２の処理手段とを備えたことを特徴とするノード装置。

【請求項１５】

前記判断手段が、前記ラベルスイッチングパスがループを形成すると判断した場合には、前記要求メッセージの転送は行わず、前記返答メッセージも返送しないことを特徴とする請求項１４に記載のノード装置。

【請求項１６】

前記判断手段は、前記要求メッセージに含まれるのと同一のフロー識別子についての情報を、未確定として前記記憶手段内に記憶しているか否かに基づいて、前記ラベルスイッチングパスがループを形成するか否かを判断することを特徴とする請求項１４に記載のノード装置。

【請求項１７】

前記要求メッセージには、前記パケットフローに関する処理中のメッセージをネットワークグローバルに識別可能なメッセージ識別子も含まれ、

前記前記第１の処理手段は、前記メッセージ識別子を前記フロー識別子とともに記憶することを特徴とする請求項１４に記載のノード装置。

【請求項１８】

前記判断手段は、受信した前記要求メッセージに含まれるのと同一のフロー識別子及びメッセージ識別子を未確定として前記記憶手段内に記憶しているか否かに基づいて、前記ラベルスイッチングパスがループを形成するか否かを判断することを特徴とする請求項１４に記載のノード装置。

【請求項１９】

前記判断手段は、自ノードが設定要求されたラベルスイッチングパスの終段のノードである場合、もしくは、自ノードの下流に対応する確立されたラベルスイッチングパスが存在しこれに設定要求されたパスを接続して良い場合に、前記ラベルスイッチングパスがループを形成しないことを確認できたものとすることを特徴とする請求項１４に記載のノード装置。

【請求項２０】

自ノードが、設定しようとするラベルスイッチングパスの初段のノード、もしくは、確立されたラベルスイッチングパスの中段のノードで経路変更が起こった場合のそのノード、もしくは、確立されたラベルスイッチングパスの終段のノードで下流隣接ノードを新たに認識したノードである場合に、要求メッセージを能動的に送信する転送手段をさらに備えたことを特徴とする請求項１４に記載のノード装置。

【請求項２１】

前記第１の処理手段により、設定しようとするラベルスイッチングパスを同じくする他の要求メッセージが現在未確定として記憶されており、前記判断手段により、該ラベルスイッチングパスがループを形成するか否かが判断できない場合には、受信した前記要求メッセージと該他の要求メッセージとを一つのメッセージとして扱うことを特徴とする請求項１４に記載のノード装置。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】

本発明は、ラベルスイッチング機能を有するノード装置およびラベルスイッチングパス設定方法に関する。

【０００２】

【従来の技術】

ＮＢＭＡ（Ｎｏｎ－Ｂｒｏａｄｃａｓｔ　Ｍｕｌｔｉｐｌｅ　Ａｃｃｅｓｓ）ネットワーク上でＩＰ（Ｉｎｔｅｒｎｅｔ　Ｐｒｏｔｏｃｏｌ）ノードが相互接続されるようなネットワークレイヤプロトコルにおいて、パケット転送を高速化するための手法として、ＭＰＬＳ（Ｍｕｌｔｉ－Ｐｒｏｔｏｃｏｌ　Ｌａｂｅｌ　Ｓｗｉｔｃｈｉｎｇ）方式がある。

【０００３】

ＭＰＬＳは、ノード間で特定のパケットフローに特定の「ラベル」を割り当て、各ノードで入力側のラベルと出力側のラベルとを関連付けてラベルスイッチングを行なうことにより、ＩＰ処理を省略し、高速なパケット転送が可能となる。例えば、リンクレイヤがＡＴＭの場合、ＶＰＩ（Ｖｉｒｔｕａｌ　Ｐａｔｈ　Ｉｄｅｎｔｉｆｉｅｒ）／ＶＣＩ（Ｖｉｒｔｕａｌ　Ｃｈａｎｎｅｌ　Ｉｄｅｎｔｉｆｉｅｒ）がラベルとして使用される。パケットフローがラベルスイッチングされる経路をラベルスイッチングパスと言う。なお、以下では、ラベルスイッチングパスを単にパスと呼ぶ。

【０００４】

パスを構成する場合には、ラベル割当てプロトコルが使用される。ラベル割当てプロトコルは、パケットフローの経路上のノード間で、ネットワークグローバルにパケットフローを識別するためのフロー識別子と、リンクローカルにパケットフローを識別するためのラベルとの対応付けを通知するために、制御メッセージ（パス制御メッセージ）をやりとりする。

【０００５】

一般に、パス制御メッセージは、レイヤ３の経路情報に従って転送されるが、レイヤ３の経路がループしている場合、パス制御メッセージがループしたり、あるいは、ループするパスが構成されるおそれがある。

【０００６】

この問題に対し、文献「Ｉｎｔｅｒｎｅｔ－Ｄｒａｆｔ，ｄｒａｆｔ－ｏｈｂａ－ｍｐｌｓ－ｌｏｏｐ－ｐｒｅｖｅｎｔｉｏｎ－００．ｔｘｔ」にて示されているループ検出方法

では、パス制御メッセージに初段ノード情報を含め、各ノードは、パスの入力側のそれぞれの枝に対し、初段ノードのアドレス（初段ノード情報）および初段ノードからのホップ数を記憶し、それらの中で、初段ノードからのホップ数が最も大きい初段ノード情報と、最大のホップ数＋１を下流に通知する。さらに、現在のパスに新たに入力側の枝を追加する場合、またはすでに存在する入力側の枝に対する初段ノード情報または初段ノードからのホップ数を別の値に更新する場合に、同じフローに対し、同じ初段ノード情報が別の入力側の枝にすでに登録している場合には、ループが構成されると判断し、入力側枝の追加／更新を行なわないようにする。

【０００７】

この方法は、パスベクトル方式（パス制御メッセージをフォワードする各ノードが自分のアドレスをラベル割当てメッセージに追加して行き、受信したラベル割当てメッセージに自分のアドレスがすでに書き込んであればループであると判断）に比べ、メッセージ長がホップ数に依存せず固定となるという利点を持つ。

【０００８】

【発明が解決しようとする課題】

しかしながら、文献、「Ｉｎｔｅｒｎｅｔ－Ｄｒａｆｔ，ｄｒａｆｔ－ｏｈｂａ－ｍｐｌｓ－ｌｏｏｐ－ｐｒｅｖｅｎｔｉｏｎ－００．ｔｘｔ」にて示されているループ検出方法には、経路変更時に以下のような問題がある。

【０００９】

例えば、図１９において、レイヤ３の経路上に、ノード装置Ｒ１１→Ｒ１２→Ｒ１３→Ｒ１４→Ｒ１５→Ｒ１６というパスが構成されている場合を考える。

【００１０】

この場合、パス上の各ノード装置は、フロー識別子（ｆｌｏｗｉｄ）、およびパスを識別するための入力リンク情報（入力物理リンク情報（ｉｎｐｕｔ　ｉｎｔｅｒｆａｃｅ）＋入力ラベル（ｉｎｐｕｔ　ｌａｂｅｌ））と一緒に、初段ノード情報（ｉｎｇｒｅｓｓ）として“Ｒ１１”を記憶している。入力リンク情報（ｉｎｐｕｔ　ｌｉｎｋ）は、出力リンク情報（ｏｕｔｐｕｔ　ｌｉｎｋ）と対応付けられてラベルスイッチングに用いられる情報で、リンク識別子から求められるラベルと、メッセージを交換した物理リンクの情報とから求められる。物理リンク情報は、Ｉｉｊ（ｉ：リンクの上流ノードの番号、ｊ：リンクの下流ノードの番号）で表され、リンクは、“物理リンク情報：ラベル”で表される。

【００１１】

図１９のパスが構成されている場合のノード装置Ｒ１５内のフローテーブルの状態を図２０に示す。

【００１２】

ここで、Ｒ１２で経路変更が起こり、Ｆｌｏｗｉｄ＝Ｒ１６への次ホップがＲ１３からＲ１７に変わった場合、Ｒ１２から旧経路上には（ｆｌｏｗｉｄ＝Ｒ１６，ｉｎｇｒｅｓｓ＝Ｒ１１）を指定したラベル割当て解放メッセージ（ＲＥＬＥＡＳＥメッセージ）が送出され、Ｒ１２から新経路上には（ｆｌｏｗｉｄ＝Ｒ１６，ｉｎｇｒｅｓｓ＝Ｒ１１）を指定したラベル割当て要求メッセージ（ＳＥＴＵＰメッセージ）が送出される。

【００１３】

もし、ＳＥＴＵＰメッセージがＲＥＬＥＡＳＥメッセージよりも先にＲ１５に到着すると、Ｒ１５では、入力リンク（Ｉ１７１５：ｘ１７１５）のＳＥＴＵＰメッセージを受信した時に、ｆｌｏｗｉｄ＝Ｒ１６，ｉｎｇｒｅｓｓ＝Ｒ１１であるような別のエントリがすでに入力リンク（Ｉ１５１６：ｘ１５１６）に対応しているため、実際にはループでないにもかかわらずループであると判断される。この結果、Ｒ１５からＲ１７にラベル割当て失敗（ＳＥＴＵＰ＿ＮＡＣＫ）メッセージが返されていた。このとき、ＳＥＴＵＰ＿ＮＡＣＫメッセージを受信したＲ１７は、一定時間後にＳＥＴＵＰメッセージの再送信（リトライ）を行なうが、リトライの間隔が短いと旧経路からＲＥＬＥＡＳＥメッセージが到着する前に何度もリトライを繰り返し、一方、リトライの間隔が長いと、新経路上にパスが

切り替わるまでに時間がかかってしまう。しかしながら、最適なリトライ間隔の値を決めるのは一般に困難であるため、できるだけリトライのないループ回避メカニズムが要求される。

【００１４】

本発明は、上記事情を考慮してなされたもので、効率的かつ早期にラベルスイッチングパスのループを検出可能であるとともに、従来ならば経路変更時にラベルスイッチングパスのループを誤って検出する場合でも従来ならば必要となるループを検出したノード装置の前段のノード装置からのラベル設定のリトライ動作を行なうことなしにラベルスイッチングパスの張り替えを行なうことを可能とするノード装置及びラベルスイッチングパス設定方法を提供することを目的とする。

【００１５】

【課題を解決するための手段】

本発明は、あるパケットフロー用のラベルスイッチングパスの設定に関わるメッセージ（例えば、ＳＥＴＵＰ、ＵＰＤＡＴＥ等）であって、該パケットフローを識別可能なフロー識別子、及び、該パケットフローに関して処理中であることを示す情報（例えば、スレッド等）を含むものを受信したノードが、メッセージ処理中として該ノード内に記憶されている情報を対象として、該ラベルスイッチングパスがループを形成するか否かを判断し、前記メッセージを受信したノードが、前記ラベルスイッチングパスがループを形成しないとは判断できない場合には、前記フロー識別子を、メッセージ処理中として（例えば、確定フラグをオフにして）該ノード内に記憶すると共に、前記メッセージを前記パケットフローの経路上のノードへ転送し、前記メッセージを受信したノードが、前記ラベルスイッチングパスがループを形成しないことを確認できた場合には、前記メッセージに対する返答メッセージを返送し、この返答メッセージを最新の返答として受信したノードは、該返答メッセージを、前記メッセージを送信／転送しメッセージ処理中となっているノードへ転送し、前記返答メッセージを最新の返答として受信したノードは、該返答メッセージに対応する、メッセージ処理中として該ノード内に記憶されている情報が、以降は前記判断の対象とならないように、メッセージ処理中としての記憶を削除し（例えば、確定フラグをオンにし）、前記ラベルスイッチングパスを利用可能とすることを特徴とする。

【００１６】

本発明によれば、ラベルスイッチングパスの設定に関わるメッセージが、経由する各ノードにメッセージ処理中である旨を記憶させながら、パスがループを形成しないことが確認できるノードまで転送され、そこから返答メッセージが返ってくる過程で、順次パスの経路上の各ノードのメッセージ処理中状態が解消され、そこではじめてパスの設定が確立されていくため、ループを形成するか否かの判断は、処理中状態となっているメッセージ（例えば、スレッド）のみを対象とすれば良く、ループ検出が簡単に行える。また、経路変更時も、古いパスは既に確立している（ループ判断の対象外である）ため、新しいパスが古いパスとぶつかった場合にも誤ってループと判断することがなくなる。したがって、新しいパスが確立されるまで古いパスへパケットを流し続けることも可能となり、新しいパスが古いパスと一部経路を同じくする場合にその部分のリンク情報を利用し続けることも可能となる。

【００１７】

好ましくは、前記メッセージを受信したノードが、前記ラベルスイッチングパスがループを形成すると判断した場合には、前記メッセージの転送は行わず、前記返答メッセージも返送しないようにしてもよい。

好ましくは、前記判断においては、受信した前記メッセージに含まれるのと同一のフロー識別子についての情報を、メッセージ処理中として該ノード内に記憶しているか否かに基づいて（新たに受信したメッセージと以前に受信して処理中となっているメッセージとが異なる入力側リンクに関するものか否かをも判断材料としても良い）、前記ラベルスイッチングパスがループを形成するか否かを判断するようにしてもよい。

好ましくは、前記メッセージには、前記パケットフローに関する処理中のメッセージをネ

ットワークグローバルに識別可能なメッセージ識別子（例えばスレッド色）も含まれるようにしてもよい。
好ましくは、前記判断においては、受信した前記メッセージに含まれるのと同一のフロー識別子及びメッセージ識別子をメッセージ処理中として該ノード内に記憶しているか否かに基づいて（新たに受信したメッセージと以前に受信して処理中となっているメッセージとが異なる入力側リンクに関するものか否かをも判断材料としても良い）、前記ラベルスイッチングパスがループを形成するか否かを判断するようにしてもよい。
【００１８】
好ましくは、前記メッセージは、設定要求されたラベルスイッチングパスの初段のノードから該メッセージを受信したノードまでのホップ数を求めることが可能な情報（例えば、スレッド長）を含むものであり、前記メッセージを受信したノードが、このメッセージに含まれる情報に基づき前記ホップ数が所定値を超えると判断した場合は、前記ラベルスイッチングパスがループを形成すると判断した場合と同等に扱うようにしてもよい。
【００１９】
好ましくは、前記メッセージを受信したノードは、自ノードが設定要求されたラベルスイッチングパスの終段のノードである場合、もしくは、自ノードの下流に対応する確立されたラベルスイッチングパスが存在しこれに設定要求されたパスを接続して良い場合（例えば、スレッド長が確立されたラベルスイッチングパスのパス長以下である場合）に、前記ラベルスイッチングパスがループを形成しないことを確認できたものとするようにしてもよい。
【００２０】
好ましくは、設定しようとするラベルスイッチングパスの初段のノード、もしくは、確立されたラベルスイッチングパスの中段のノードで経路変更が起こった場合のそのノード、もしくは、確立されたラベルスイッチングパスの終段のノードで下流隣接ノードを新たに認識したノードが、前記メッセージを能動的に送信するようにしてもよい。
【００２１】
好ましくは、前記メッセージ識別子は、該メッセージ識別子を割り当てたノード（例えば、代理初段ノード）の識別情報を含むようにしてもよい。
【００２２】
好ましくは、前記返答メッセージの返送の際に、ラベルスイッチングのためのリンク情報に含まれるラベルの割り当てを行うようにしてもよい。
【００２３】
これにより、ループ無しのラベルスイッチングパスがエンド－エンドで設定可能なことを確認してから実際のスイッチングに用いるラベルを割り当てることができるため、ラベルリソースの節約になる。
【００２４】
好ましくは、前記メッセージを受信したあるノードで前記ラベルスイッチングパスがループを形成すると判断された場合に、前記メッセージを能動的に送信したノードは、前記ラベルスイッチングパスの設定の必要が無くなった場合（例えば、対応する入力側リンク情報が削除された場合や、経路変更が起こった場合）に、前記メッセージに対する取消メッセージ（例えば、スレッド削除通知）を送信し、前記メッセージを転送した各ノードは、対応する前記取消メッセージを受信するまでは、メッセージ処理中としての記憶を保持しておくようにしてもよい。
【００２５】
これにより、不必要なラベル割当てを防ぐことができるため、ラベルリソースの節約になる。
【００２６】
好ましくは、前記メッセージを受信したノードは、設定しようとするラベルスイッチングパスを同じくするメッセージが処理中であることが記憶されており、該ラベルスイッチングパスがループを形成するか否かを判断できない場合には、受信したメッセージと処理中

のメッセージとを一つのメッセージとして扱うようにしてもよい。

【００２７】

例えば、メッセージに、設定要求されたラベルスイッチングパスの初段のノードから該メッセージを受信したノードまでのホップ数に関する情報が含まれる場合には、このホップ数が大きい方のメッセージの識別子を、前記一つのメッセージの識別子とする。そして、この新たに決めたメッセージ識別子（とホップ数）が、既に転送した（処理中の）メッセージと異なる場合には、この変更をパスの経路上のノードに通知する。これにより、メッセージ数、ひいては処理中の状態数を少なくすることができる。

【００２８】

好ましくは、前記メッセージは、設定要求されたラベルスイッチングパスの初段のノードから該メッセージを受信したノードまでのホップ数を求めることが可能な情報を含むものであり、前記メッセージを受信したノードは、前記ラベルスイッチングパスがループを形成すると判断した場合に、該ラベルスイッチングパスの初段のノードから自ノードまでのホップ数に関する情報を修正するようにしてもよい。

【００２９】

例えば、特別な＜不定＞の値に更新する等して、ループにより処理中のまま保持されるメッセージを後で用いる際にホップ数が誤って加算された状態になることを防ぐことができる。

【００３０】

本発明は、ラベルスイッチングパスを設定するノード装置であって、あるパケットフロー用のラベルスイッチングパスの設定に関わる要求メッセージ（例えば、ＳＥＴＵＰ、ＵＰＤＡＴＥ等）であって、該パケットフローを識別可能なフロー識別子、及び、該パケットフローに関して処理中であることを示す情報（例えば、スレッド等）を含むものを受信した場合に、未確定として記憶手段内に記憶されている情報を対象として、該ラベルスイッチングパスがループを形成するか否かを判断する判断手段と、前記判断手段により前記ラベルスイッチングパスがループを形成しないことが確認できない場合には、前記フロー識別子を、未確定として前記記憶手段内に記憶すると共に、前記要求メッセージを前記パケットフローの経路上のノードへ転送し、前記判断手段により前記ラベルスイッチングパスがループを形成しないことが確認できた場合には、前記要求メッセージに対する返答メッセージを返送する第１の処理手段と、前記要求メッセージに対する返答メッセージを最新の返答として受信した場合に、現在は未確定として前記記憶手段内に記憶されている情報が、以降は前記判断の対象とならないように、該情報を確定として記憶するように変更し、該返答メッセージを、前記要求メッセージを転送し現在未確定として情報を記憶している他のノードへ転送して、前記ラベルスイッチングパスを利用可能とする第２の処理手段とを備えたことを特徴とする。

好ましくは、前記判断手段が、前記ラベルスイッチングパスがループを形成すると判断した場合には、前記要求メッセージの転送は行わず、前記返答メッセージも返送しないようにしてもよい。

好ましくは、前記判断手段は、前記要求メッセージに含まれるのと同一のフロー識別子についての情報を、未確定として前記記憶手段内に記憶しているか否かに基づいて、前記ラベルスイッチングパスがループを形成するか否かを判断するようにしてもよい。

好ましくは、前記要求メッセージには、前記パケットフローに関する処理中のメッセージをネットワークグローバルに識別可能なメッセージ識別子も含まれ、前記前記第１の処理手段は、前記メッセージ識別子を前記フロー識別子とともに記憶するようにしてもよい。

好ましくは、前記判断手段は、受信した前記要求メッセージに含まれるのと同一のフロー識別子及びメッセージ識別子を未確定として前記記憶手段内に記憶しているか否かに基づいて、前記ラベルスイッチングパスがループを形成するか否かを判断するようにしてもよい。

好ましくは、前記判断手段は、自ノードが設定要求されたラベルスイッチングパスの終段のノードである場合、もしくは、自ノードの下流に対応する確立されたラベルスイッチン

グパスが存在しこれに設定要求されたパスを接続して良い場合に、前記ラベルスイッチングパスがループを形成しないことを確認できたものとするようにしてもよい。
好ましくは、自ノードが、設定しようとするラベルスイッチングパスの初段のノード、もしくは、確立されたラベルスイッチングパスの中段のノードで経路変更が起こった場合のそのノード、もしくは、確立されたラベルスイッチングパスの終段のノードで下流隣接ノードを新たに認識したノードである場合に、要求メッセージを能動的に送信する転送手段をさらに備えるようにしてもよい。
好ましくは、前記第１の処理手段により、設定しようとするラベルスイッチングパスを同じくする他の要求メッセージが現在未確定として記憶されており、前記判断手段により、該ラベルスイッチングパスがループを形成するか否かが判断できない場合には、受信した前記要求メッセージと該他の要求メッセージとを一つのメッセージとして扱うようにしてもよい。
また、本発明は、ラベルスイッチングパスを設定するノード装置としてコンピュータを機能させるためのプログラムを記録したコンピュータ読取り可能な記録媒体であって、あるパケットフロー用のラベルスイッチングパスの設定に関わる要求メッセージであって、該パケットフローを識別可能なフロー識別子、及び、該パケットフローに関して処理中であることを示す情報を含むものを受信した場合に、未確定として記憶手段内に記憶されている情報を対象として、該ラベルスイッチングパスがループを形成するか否かを判断する判断機能と、前記判断手段により前記ラベルスイッチングパスがループを形成しないことが確認できない場合には、前記フロー識別子を、未確定として前記記憶手段内に記憶すると共に、前記要求メッセージを前記パケットフローの経路上のノードへ転送し、前記判断手段により前記ラベルスイッチングパスがループを形成しないことが確認できた場合には、前記要求メッセージに対する返答メッセージを返送する第１の処理機能と、前記要求メッセージに対する返答メッセージを最新の返答として受信した場合に、現在は未確定として前記記憶手段内に記憶されている情報が、以降は前記判断の対象とならないように、該情報を確定として記憶するように変更し、該返答メッセージを、前記要求メッセージを転送し現在未確定として情報を記憶している他のノードへ転送して、前記ラベルスイッチングパスを利用可能とする第２の処理機能とを実現させるためのプログラムを記録したコンピュータ読取り可能な記録媒体である。
【００３１】
なお、装置に係る本発明は方法に係る発明としても成立し、方法に係る本発明は装置に係る発明としても成立する。
【００３２】
また、装置または方法に係る本発明は、コンピュータに当該発明に相当する手順を実行させるための（あるいはコンピュータを当該発明に相当する手段として機能させるための、あるいはコンピュータに当該発明に相当する機能を実現させるための）プログラムを記録したコンピュータ読取り可能な記録媒体としても成立する。
【００３３】
【発明の実施の形態】
以下、図面を参照しながら発明の実施の形態を説明する。
【００３４】
最初に、語句や本発明のノード装置の基本的な構成等について説明する。
【００３５】
＜方向の定義＞
各ノードにおいて、ラベルスイッチングパス（以下、パス）の初段ノードに近い側を「入力側」（または上流）、終段ノードに近い側を「出力側」（または下流）と呼ぶ。
【００３６】
各ノードにおいて、パスの入力側の隣接ノード（上流隣接ノード）に対して割当てられるラベルを「入力側ラベル」（以下、入力ラベル）と呼び、出力側の隣接ノード（下流隣接ノード）に対して割当てられるラベルを「出力側ラベル」（以下、出力ラベル）と呼ぶ。

【００３７】
＜隣接ノード＞
隣接ノードは、ラベルの書き換えなしでパケットが到達可能な自ノード以外のノードのことであり、ネットワークプロトコル的な隣接ノードである必要はない。例えば、ラベルがスタックされる場合には、スタックレベルが深くなるとネットワークプロトコル的に２ホップ以上離れたノードが隣接ノードとなりうる。
【００３８】
＜リンク＞
各ノードにおいて、あるパスの個々の入力ラベル、出力ラベルに関連した「状態」を持つ。入力ラベルに関連した状態を「入力リンク」、出力ラベルに関連した状態を「出力リンク」と呼ぶ。リンクを削除する場合には、隣接ノードにリンクを削除する旨を通知する。
【００３９】
＜フロー＞
フロー（パケットフロー）は、パスを流れるパケットの集合である。フローは、フロー識別子により規定される。
【００４０】
＜代理初段ノード＞
代理初段ノードとは、パスの生成を「能動的に」起動するノードのことである。ここで、「能動的に」とは、「上流隣接ノードからのパス設定要求を直接のトリガにすることなく」という意味である。「能動的」の反対の意味で「受動的」という語句が使われる。
【００４１】
代理初段ノードは、（１）初段ノード、（２）入力リンクを有するノードで、経路変更により下流隣接ノードに出力ラベルの割当てを要求するノード、（３）入力リンクを有するノードで下流隣接ノードを新たに認識したノード、のいずれかである。
【００４２】
＜パス長＞
あるノードにおいて、自ノードの上流の初段ノード（複数あり得る）から自ノードまでのホップ数の最大値をパス長と呼ぶ。パス長を下流隣接ノードに通知する場合には、自ノードにおけるパス長を１だけインクリメントした値を通知する。パス長を管理するために、パスの各入力リンク毎に上流隣接ノードが通知するパス長の情報を記憶する必要がある。
【００４３】
経路がループしている場合には、正確なパス長が不定となる。この場合には、パス長として＜不定＞という特別な値を使用してもよい。パス長の値が＜不定＞の場合には、パス長の値が＜不定＞以外の場合よりもパス長が大きいとする。また、値が＜不定＞のパス長を１だけインクリメントしたパス長も＜不定＞となる。
【００４４】
＜スレッド＞
代理初段ノードがあるフローに関して、あるパスに関する出力ラベルの割当てを下流隣接ノードに要求する際、パスのループを検出し回避するために、そのフローに関するＡＣＫ待ちのメッセージをネットワークで一意に識別するための識別子を少なくとも含んだ情報である「スレッド」を生成し、これを下流隣接ノードに通知（フォワード）する。
【００４５】
スレッドは、少なくとも識別子（スレッド色）を持つ。スレッド色が同じスレッドは、同じスレッドであるとみなす。また、スレッドは、パス長に関する情報（スレッド長）を持っていてもよい。このとき、初段ノードは、スレッド長０のスレッドを生成する。
【００４６】
一般に、スレッドは、（スレッド色、スレッド長）で表され、スレッドがスレッド長の情報を含まない場合には、本実施形態では、すべてのスレッドが同じスレッド長＝＜不定＞に設定されるとして扱えばよい。
【００４７】

＜スレッドのフォワード＞
各ノードは、受信したスレッドを受信した順番に処理する。また、隣接ノード間でスレッドの転送順序は保証される必要がある。
【００４８】
スレッドを下流隣接ノードにフォワードする場合には、スレッド長を１　だけインクリメントする。スレッドは、パスがループになるかどうかを判定できるノードまでフォワードされる。
【００４９】
下流隣接ノードにフォワードしたスレッドは、フォワードしたノードのある出力リンクに対応し、上流隣接ノードから受信したスレッドは、そのノードのある入力リンクに対応する。
【００５０】
フォワードしたスレッドは、スレッドがループしないことを確認するＡＣＫが下流隣接ノードから返るまでスレッドをフォワードディング中の状態になる。
【００５１】
例外として、スレッドがループしないことを確認済みのパスに対してパス長の更新のみを通知する場合には、ＡＣＫの必要ないスレッドが定義される。ＡＣＫの必要ないスレッドは、スレッド色＝０で識別される。
【００５２】
＜スレッドの削除＞
代理初段ノードは、出力リンクとスレッドの対応を能動的に削除することができるが、入力リンクとスレッドの対応を能動的に削除することはできない。代理初段ノード以外のノードは、能動的に入力リンクまたは出力リンクとスレッドとの対応を削除することはできない。
【００５３】
あるリンクが削除される場合、そのリンクがスレッドとの対応を持っていればそのスレッドも削除される。
【００５４】
代理初段ノードがある入力リンクとスレッドの対応を受動的に削除するときに、もし、入力リンクが削除されなければその入力リンクに対応する出力リンクのスレッドは削除されない。
【００５５】
代理初段以外のノードがある入力リンクとスレッドの対応を受動的に削除するときに、もし、同じパスの他の入力リンクにループを形成しないスレッドが一つも対応していなければ、その入力リンクに対応する出力リンクのスレッドを削除する。
【００５６】
出力リンクに対応するスレッドを削除する場合には、下流隣接ノードにスレッドの削除が通知される。
【００５７】
＜スレッドのループ＞
スレッドをフォワード中のノードが、そのスレッドを受信した入力リンクとは異なる入力リンクにおいて同じスレッドを受信した場合、または、受信したのと同じスレッドを出力リンクにおいてフォワード中の場合には、スレッドがループを形成するという。
【００５８】
＜スレッドのマージ＞
あるノードがスレッドを受信した時、もし受信スレッドがループしておらず、かつ、同じパスに属する別のスレッドが存在する場合には、スレッドのマージを行なう。逆に、ある入力リンクに対応するスレッドが削除される場合には、スレッドをマージから外す（アンマージ）。もし、削除後も他の入力リンクにループを形成しない別のスレッドが対応している場合には、出力側リンクに対応するスレッドは削除しない。

【００５９】
マージされる前の各々のスレッドのうち、いずれか１つのスレッドをネットワークで統一されたある選択方法により選んで、そのスレッド色／スレッド長をマージ後のスレッドのスレッド色／スレッド長とする。選択方法の１つとして、スレッド長を用いる場合には、マージされたスレッドのスレッド長は、各々のスレッド長の最大値に設定し、マージされたスレッド色は、各々のスレッド長の最大値を与えるスレッドのスレッド色に設定し、もし、スレッド長の最大値を与えるスレッドが複数ある場合には、それらの中で最大のスレッド色（スレッド色を示す数値が最大のもの）に設定する、という方法がある。
【００６０】
＜スレッドの更新＞
ノードが、ある入力リンクに対応するスレッドを受信したとき、その入力リンクに現在対応するスレッドのスレッド色またはスレッド長が受信スレッドと異なる場合には、受信スレッドをその入力リンクに対応させる。スレッドがマージされていない場合、または、スレッドがマージされている場合でマージされたスレッドのスレッド色／スレッド長が変わる場合には、スレッド色／スレッド長の変更を直ちに下流隣接ノードにフォワードする。
【００６１】
さらに、あるノードが、ある出力リンクに対応するスレッドの情報の更新中であるときに、このスレッドに対するさらなる更新が起こっても、前の更新の完了を待つことなく新たな更新を下流隣接ノードに通知するようにする。
【００６２】
スレッドをフォワーディングすることにより、以下のようにループの回避したパス設定を行なう。
【００６３】
＜パスのループ回避アルゴリズム＞
スレッドは、（ｉ）終段ノードに届くか、（ｉｉ）パス長が＜不定＞でない確立されたパスが存在し、スレッド長がパス長以下になるか、（ｉｉｉ）スレッドがループを形成するか、スレッド長が＜不定＞でなくかつスレッド長がしきい値に到達するか、のいずれかになるまでフォワードされる。
【００６４】
（ｉ），（ｉｉ）のいずれかの理由によりスレッドのフォワーディングが完了したときには、スレッドはループを形成しないことが保証される。このとき、ラベルスイッチングパスもループを形成しないことが保証されるため、上流ノードにホップバイホップにスレッドに対するＡＣＫを返す。ＡＣＫを送信／受信したノードは、スレッドがフォワーディング中であるという状態をクリアする。
【００６５】
ＡＣＫがスレッドを生成した代理初段ノードまで返ると、ラベルスイッチングが可能になる。
【００６６】
一方、（ｉｉｉ）の理由によりスレッドのフォワーディングが完了したときには、スレッドがループを形成する場合には、スレッド長＝＜不定＞に設定する。この結果、スレッド長が更新されるなら、更新されたスレッド長を下流ノードに通知する。このとき、ループしたスレッドの入力リンクと出力リンクの対応づけは保つようにする。なお、スレッド長を用いない場合には、スレッド長は常に＜不定＞であるため、スレッド長の更新は下流ノードに通知されない。
【００６７】
もし、代理初段ノードで経路変更が起こった場合には、新たにスレッドを生成し、新経路に生成したスレッドをフォワードする。新経路にスレッドを通知する前にまず、旧経路上の出力リンクにフォワーディング中のスレッドが対応していればこれを削除し、旧経路上にスレッドの削除を通知し、これらの上流側のスレッドと生成したスレッドをつなげる（入出力を関連付ける）。

【００６８】
パス長は、ノードがスレッドを受信した場合に、受信スレッドがループを形成しない場合に、直ちに受信スレッド長に設定してもよいし、また、スレッドに対するＡＣＫを受信したときに設定してもよい。
【００６９】
また、確立されたパスにおいてパス長の更新のみを下流隣接ノードに通知する場合には、スレッド色＝０、スレッド長＝更新後のスレッド長を１だけインクリメントした値、のスレッドが使用される。スレッド色＝０のスレッドは、パス長が更新されなくなるまで下流にフォワードされ、ＡＣＫは必要ない。
【００７０】
なお、本発明は、いかなるフローの定義に対しても使用可能である。また、初段ノードがパス設定を起動する契機は、フローに対するルーチングテーブルエントリが生成された時を契機とするトポロジードリブン、または、実際にそのフローにマッチするパケットが流れ始めた時を契機とするフロードリブン、のいずれでもよい。
【００７１】
同一のフローとする形態には、（１）宛先ＩＰアドレスが同じパケットを同一のフローとする、（２）ネットワークマスクに対応する宛先ＩＰアドレスの部分（すなわち宛先ＩＰアドレスの上位の所定ビット）が同じパケットを同一のフローとする、（３）送信元ＩＰアドレスと宛先ＩＰアドレスの組が同じパケットを同一のフローとする、（４）第１のネットワークマスクに対応する送信元ＩＰアドレスの部分と第２のネットワークマスクに対応する宛先ＩＰアドレスの部分との組が同じパケットを同一のフローとする、（５）あるＩＰアドレスを持つノードを通るパケットを同一のフローとする、（６）上記の（１）～（５）において、送信元および／または宛先のポート番号をも組にしたものが同一であるパケットを同一のフローとする、などが種々のもの考えられる。
【００７２】
ところで、本発明は、（１）初段ノード主導でラベルスイッチングパスを生成する場合（Ｉｎｇｒｅｓｓ　Ｃｏｎｔｒｏｌと呼ぶ）にも、（２）終段ノード主導でラベルスイッチングパスを生成する場合（Ｅｇｒｅｓｓ　Ｃｏｎｔｒｏｌと呼ぶ）にも、適用可能である。
【００７３】
初段となるノード装置がラベル割当て要求メッセージを送信する契機には、例えば、次のような形態がある。
（１）各々のノード装置が、宛先ｄｅｓｔに対するルーティングテーブルが生成されたときを契機に、ラベル割当て要求メッセージを送信する。
（２）各々のノード装置が、宛先ｄｅｓｔに対するルーティングテーブルが生成された後に宛先ｄｅｓｔに対するトラヒック量がしきい値を越えた場合を契機に、ラベル割当て要求メッセージを送信する。
（３）宛先ｄｅｓｔに対するルーティングテーブルが生成された後に、宛先ｄｅｓｔに対するトラヒック量がしきい値を越え、かつ、トラヒック（データグラム）の前段ノード装置が本実施形態に係るラベル割り当てプロトコルをサポートしない、という条件をみたすノード装置のみが、これを契機に自ノードを初段と判断してラベル割当て要求メッセージを送信する（例えばＩＮＴＥＲＮＥＴ－ＤＲＡＦＴ　＜ｄｒａｆｔ－ｉｅｔｆ－ｒｏｌｃ－ｎｈｒｐ－ｌ１．ｔｘｔ＞　ＮＢＭＡ　Ｎｅｘｔ　Ｈｏｐ　Ｒｅｓｏｌｕｔｉｏｎ　Ｐｒｏｔｏｃｏｌ（ＮＨＲＰ）参照）。
【００７４】
さて、以下では、幾つかの具体例を用いて本実施形態についてより詳しく説明していく。
【００７５】
本実施形態に係るノード装置は、ＩＰ（Ｉｎｔｅｒｎｅｔ　Ｐｒｏｔｏｃｏｌ）処理を行う機能を持つとともに、本発明に係るプロトコルに従ってラベルスイッチングを行う機能を持つ。また、末端に位置し中継を行わないノード装置以外のノード装置は、ルータ機能

を持ってもよい（例えば図１においてノード装置Ｒ１～Ｒ４，Ｒ７～Ｒ１１はルータ機能を持ち、ルータ装置Ｒ５，Ｒ６についてはこれらが末端に位置し中継を行わないものである場合にはルータ機能は不要である）。

【００７６】

以下では、宛先ＩＰアドレスが同じパケットを同一のフローであるものとし、特定のフローに着目して説明を行なう。また、以下の例では、マージを行なう場合について説明する（ノード装置は、フロー識別子が等しい複数の入力側ラベルに対して同一の出力リンク情報を対応させることによりパスのマージを行う）。また、以下の例では、パス長は、スレッドに対するＡＣＫが返ってきたときに、そのスレッドのスレッド長に設定するものとする。

【００７７】

パスは、各入力に対して受信スレッドの情報を記憶するために入力リンクを持ち、各出力に対して送信スレッドの情報を記憶するために出力リンクを持つ。また、パスは、パス上でラベルスイッチングされるフローの情報（フローＩＤ）、パス長の情報、自ノードが初段ノードであるか否かを示すフラグ（初段フラグ）、およびパスがスレッドに対するＡＣＫ待ちかどうかを示すフラグ（確定フラグ）を持つ。

【００７８】

スレッドは、（スレッド色、スレッド長）で識別されるが、以下の例では、スレッド色は、“代理初段ノードのアドレス：ローカルＩＤ”で表されるものとする。ここでは、ローカルＩＤは、各代理初段ノードでスレッドを新たに生成する毎に新たに割当てられる識別子である。ただし、以下において、ローカルＩＤ＝０の場合には、ローカルＩＤの記述を省略し、スレッド色を“代理初段ノードのアドレス”で表している。

【００７９】

また、以下の例では、スレッド色の大小は、代理初段ノードアドレスの値で比較して決めるものとする（もし、代理初段ノードアドレスが同じで、シーケンス番号が異なるスレッドがある場合には、シーケンス番号の大小で決めるものとする）。なお、以下の例では、代理初段ノードアドレスとしてＲ１などを用いるが、ｉ＞ｊとした場合に、代理初段ノードアドレスＲｉの大きさ＞代理初段ノードアドレスＲｊの大きさであるものとする。

【００８０】

パスの確定フラグは、スレッドを下流に通知するときにオフになり、スレッドの通知に対するＡＣＫを受けたときにオンになる。マルチキャストの場合には、全出力リンクからスレッドの通知に対するＡＣＫを受けたときにオンになる。

【００８１】

入力リンクは、入力インターフェースとリンク識別子で、出力リンクは、出力インターフェースとリンク識別子で識別される。リンク識別子は、インターフェースでローカルに割当てることができ、リンク識別子として、ラベルを使ってもよいし、その他のＩＤを使ってもよい。

【００８２】

前述のように、本発明は、初段ノード主導でパスを生成する場合にも、終段ノード主導でパスを生成する場合にも適用可能であるが、以下では、初段ノード主導でパスを生成する場合を例にとって説明する。終段ノード主導の場合には、フローの下流隣接ノードから新たにラベルの割当て要求を受信したノードが初段ノードとなってスレッドを生成し下流隣接ノードにフォワードすること以外は初段ノード主導の場合と同じである。

【００８３】

なお、以下の記述で、「ａ＝ｂ」はａの値もしくは内容がｂであることを、「ａ＝＝ｂ」はａがｂと等しい（もしくはａがｂである）ことを、「ａ！＝ｂ」はａがｂと等しくない（もしくはａがｂでない）ことを、「ａ：＝ｂ」はａにｂを代入もしくは設定することを、「ａ＞＝ｂ」はａがｂ以上であることを、「ａ｜｜ｂ」はａかつｂを示し、「ａ＆＆ｂ」はａまたはｂを示すものとする。

【００８４】

（第１の具体例）
まず、本発明の一実施形態に係るノード装置により行われるラベルスイッチングパスの設定の動作について具体例を用いながら説明する。
【００８５】
図１に、本実施形態に係るノード装置を接続して構成されたラベルスイッチングネットワークの一例を示す。なお、図１において、ノード装置Ｒ１～Ｒ１１の全てまたは一部に図示しない他のノード装置が接続されていてもよい。
【００８６】
図１において、隣接ノード間のリンクはそれらのノード間でラベルスイッチングが可能であることを表し、それらのうち矢印付きのリンクは矢印の方向がフローの現在の次ホップを表している。なお、図１には、以降の説明で用いるスレッドの内容の例も併せて示してある。
【００８７】
さて、図１の構成において、ノード装置Ｒ１とノード装置Ｒ６が、ほぼ同時にパスの設定を開始するとする。
【００８８】
ノード装置Ｒ１は、初段フラグをオンにし、ノード装置Ｒ２との間に出力リンクを生成し、スレッド（Ｒ１，０）を生成し、出力リンクにスレッド（Ｒ１，０）を記憶し、ノード装置Ｒ２にスレッド（Ｒ１，１）を通知する。同様に、ノード装置Ｒ６は、初段フラグをオンにし、ノード装置Ｒ７との間に出力リンクを生成し、スレッド（Ｒ６，０）を生成し、出力リンクにスレッド（Ｒ６，０）を記憶し、ノード装置Ｒ７にスレッド（Ｒ６，１）を通知する。このとき、スレッドは、ラベル割当て要求メッセージ（ＳＥＴＵＰ）に入れられる。
【００８９】
ノード装置Ｒ２は、スレッド（Ｒ１，１）を受信すると、ノード装置Ｒ１との間に新たに入力リンクを生成し、スレッド（Ｒ１，１）を記憶する。次に、ノード装置Ｒ３との間に出力リンクを生成し、出力リンクに受信スレッド（Ｒ１，１）を記憶し、受信スレッド（Ｒ１，１）のスレッド長を１だけインクリメントしたスレッド（Ｒ１，２）をノード装置Ｒ３に通知する。
【００９０】
ノード装置Ｒ７は、スレッド（Ｒ６，１）を受信すると、ノード装置Ｒ６との間に新たに入力リンクを生成し、受信スレッド（Ｒ６，１）を記憶する。次に、ノード装置Ｒ８との間に出力リンクを生成し、出力リンクに受信スレッド（Ｒ６，１）を記憶し、受信スレッド（Ｒ６，１）のスレッド長を１だけインクリメントしたスレッド（Ｒ６，２）をノード装置Ｒ８に通知する。
【００９１】
ノード装置Ｒ８は、スレッド（Ｒ６，２）を受信すると、ノード装置Ｒ７との間に新たに入力リンクを生成し、受信スレッド（Ｒ６，２）を記憶する。次に、ノード装置Ｒ３との間に出力リンクを生成し、出力リンクに受信スレッド（Ｒ６，２）を記憶し、受信スレッド（Ｒ６，２）のスレッド長を１だけインクリメントしたスレッド（Ｒ６，３）をノード装置Ｒ３に通知する。
【００９２】
ノード装置Ｒ３は、ノード装置Ｒ１が生成したスレッドをノード装置Ｒ６が生成したスレッドよりも早く受信したとする。
【００９３】
ノード装置Ｒ３は、スレッド（Ｒ１，２）を受信すると、ノード装置Ｒ２との間に新たに入力リンクを生成し、受信スレッド（Ｒ１，２）を記憶する。次に、ノード装置Ｒ４との間に出力リンクを生成し、出力リンクに受信スレッド（Ｒ１，２）を記憶し、受信スレッド（Ｒ１，２）のスレッド長を１だけインクリメントしたスレッド（Ｒ１，３）をノード装置Ｒ４に通知する。

【００９４】
その後、ノード装置Ｒ３は、スレッド（Ｒ６，３）を受信すると、ノード装置Ｒ８との間に新たに入力リンクを生成し、受信スレッド（Ｒ６，３）を記憶する。このとき、ノード装置Ｒ４との間の出力リンクがすでに存在するため、スレッドのマージが行なわれる。いま、受信スレッド（Ｒ６，３）のスレッド長＝３が現在出力リンクに記憶されているスレッド（Ｒ１，２）のスレッド長＝２より大きいため、出力リンクに現在記憶されているスレッド（Ｒ１，２）を受信スレッド（Ｒ６，３）に置き換えるとともに、受信スレッド（Ｒ６，３）のスレッド長を１だけインクリメントしたスレッド（Ｒ６，４）をノード装置Ｒ４に通知する。
【００９５】
ノード装置Ｒ４は、最初に、スレッド（Ｒ１，３）を受信する。このとき、ノード装置Ｒ４は、ノード装置Ｒ３との間に新たに入力リンクを生成し、スレッド（Ｒ１，３）を記憶する。次に、ノード装置Ｒ９との間に出力リンクを生成し、出力リンクに受信スレッド（Ｒ１，３）を記憶し、受信スレッド（Ｒ１，３）のスレッド長を１だけインクリメントしたスレッド（Ｒ１，４）をノード装置Ｒ９に通知する。
【００９６】
ノード装置Ｒ４は、次に、スレッド（Ｒ６，４）を受信する。このとき、すでにノード装置Ｒ３との間に入力リンクが存在するため、入力リンクに現在記憶されているスレッド（Ｒ１，３）を受信スレッド（Ｒ６，４）に置き換える。また、すでにノード装置Ｒ９との間に出力リンクが存在するため、出力リンクに現在記憶されているスレッド（Ｒ１，３）を受信スレッド（Ｒ６，４）に置き換えるとともに、受信スレッド（Ｒ６，４）のスレッド長を１だけインクリメントしたスレッド（Ｒ６，５）をノード装置Ｒ９に通知する。
【００９７】
ノード装置Ｒ９，Ｒ１０も、ノード装置Ｒ４と同様の動作を行なう。
【００９８】
この結果、ノード装置Ｒ３は、ノード装置Ｒ１０からスレッド（Ｒ１，６），スレッド（Ｒ６，７）をこの順番で受信する。
【００９９】
まず、ノード装置Ｒ３がスレッド（Ｒ１，６）を受信すると、ノード装置Ｒ３は、ノード装置Ｒ１０との間に新たに入力リンクを生成し、スレッド（Ｒ１，６）を記憶する。このときすでに同じスレッドがノード装置Ｒ２との間の入力リンクに記憶されているためスレッドのループを検出する。このとき、スレッドのマージは行なわず、ＡＣＫも返さない。また、入力リンクに記憶したスレッド色を０に設定しておく。スレッド色＝０のスレッドはマージしないようにする。さらに、ノード装置Ｒ３は、ノード装置Ｒ４との間の出力リンクに記憶されているスレッド（Ｒ６、３）のスレッド長を＜不定＞に更新し、更新後のスレッド（Ｒ６、＜不定＞）のスレッド長を１だけインクリメントしたスレッド（Ｒ６、＜不定＞）をノード装置Ｒ４に通知する。
【０１００】
次に、ノード装置Ｒ３がスレッド（Ｒ６，７）を受信すると、ノード装置Ｒ３は、ノード装置Ｒ１０との間にすでに存在する入力リンクにスレッド（Ｒ６，７）を記憶する。このときすでに同じスレッドがノード装置Ｒ８との間の入力リンクに記憶されているためスレッドのループを検出する。このとき、スレッドのマージは行なわず、ＡＣＫも返さない。また、入力リンクに記憶したスレッド色を０に設定しておく。スレッド色＝０のスレッドはマージしないようにする。ここで、すでに、ノード装置Ｒ４との間の出力リンクに記憶されているスレッド（Ｒ６、＜不定＞）のスレッド長は＜不定＞のまま更新されないため、ノード装置Ｒ４にはスレッド長の更新を通知しない。
【０１０１】
最後に、ノード装置Ｒ３がスレッド（Ｒ６、＜不定＞）を受信すると、ノード装置Ｒ３は、ノード装置Ｒ１０との間にすでに存在する入力リンクにスレッド（Ｒ６、＜不定＞）を記憶する。このときすでに同じスレッドがノード装置Ｒ８との間の入力リンクに記憶され

ているため、スレッドのループを検出する。このとき、スレッドのマージは行わず、ＡＣＫも返さない。また、入力リンクに記憶したスレッド色を０に設定しておく。スレッド色＝０のスレッドはマージしないようにする。ここで、すでに、ノード装置Ｒ４との間の出力リンクに記憶されているスレッド（Ｒ６、＜不定＞）のスレッド長は＜不定＞のまま更新されないため、ノード装置Ｒ４にはスレッド長の更新を通知しない。

【０１０２】

この結果、図２の点線に示すようなマージされたスレッドが残る。なお、図２において、点線に付した符号はスレッドの各部分におけるスレッド色を表す。

【０１０３】

次に、図２の状態で、ノード装置Ｒ１０において経路変更が起こり、次ホップノードがノード装置Ｒ３からノード装置Ｒ１１に変わった場合について、図３および図４を参照しながら説明する。図３および図４には、通知されるスレッド等を示してある。

【０１０４】

さて、ノード装置Ｒ１０の次ホップノードがノード装置Ｒ３からノード装置Ｒ１１に変わったとすると、このとき、ノード装置Ｒ１０は、まず、旧経路上のノード装置Ｒ３との間の出力リンクを削除し、記憶していたスレッド（Ｒ６，＜不定＞）の削除を通知する（図３参照）。スレッド削除の通知は、スレッド削除メッセージを送信することにより行なう。次に、ノード装置Ｒ１０は、スレッド（Ｒ１０，＜不定＞）を生成し、新経路上のノード装置Ｒ１１との間に出力リンクを生成し、出力リンクにスレッド（Ｒ１０，＜不定＞）を記憶し、生成スレッド（Ｒ１０，＜不定＞）のスレッド長を１だけインクリメントしたスレッド（Ｒ１０，＜不定＞）をノード装置Ｒ１１に通知する（図４参照）。

【０１０５】

ノード装置Ｒ３がスレッド（Ｒ６，＜不定＞）削除の通知を受けると、対応する入力リンクを削除する。

【０１０６】

一方、ノード装置Ｒ１１は、スレッド（Ｒ１０，＜不定＞）を受信すると、ノード装置Ｒ１０との間に新たに入力リンクを生成し、受信スレッド（Ｒ１０，＜不定＞）を記憶する。次に、ノード装置Ｒ１との間に出力リンクを生成し、出力リンクに受信スレッド（Ｒ１０，＜不定＞）を記憶し、受信スレッド（Ｒ１０，＜不定＞）のスレッド長を１だけインクリメントしたスレッド（Ｒ１０，＜不定＞）をノード装置Ｒ１に通知する。

【０１０７】

ノード装置Ｒ１は、スレッド（Ｒ１０，＜不定＞）を受信すると、ノード装置Ｒ１１との間に新たに入力リンクを生成し、受信スレッド（Ｒ１０，＜不定＞）を記憶する。このとき、ノード装置Ｒ２との間の出力リンクがすでに存在するため、スレッドのマージが行なわれる。いま、受信スレッド（Ｒ１０，＜不定＞）のスレッド長＝＜不定＞が現在出力リンクに記憶されているスレッド（Ｒ１，０）のスレッド長＝０より大きいため、出力リンクに現在記憶されているスレッド（Ｒ１，０）を受信スレッド（Ｒ１０，＜不定＞）に置き換えるとともに、受信スレッド（Ｒ１０，＜不定＞）のスレッド長を１だけインクリメントしたスレッド（Ｒ１０，＜不定＞）をノード装置Ｒ２に通知する。

【０１０８】

ノード装置Ｒ２は、スレッド（Ｒ１０，＜不定＞）を受信する。このとき、すでにノード装置Ｒ１との間に入力リンクが存在するため、入力リンクに現在記憶されているスレッド（Ｒ１，１）を受信スレッド（Ｒ１０，＜不定＞）に置き換える。また、すでにノード装置Ｒ３との間に出力リンクが存在するため、出力リンクに現在記憶されているスレッド（Ｒ１，１）を受信スレッド（Ｒ１０，＜不定＞）に置き換えるとともに、受信スレッド（Ｒ１０，＜不定＞）のスレッド長を１だけインクリメントしたスレッド（Ｒ１０，＜不定＞）をノード装置Ｒ３に通知する。

【０１０９】

ノード装置Ｒ３は、スレッド（Ｒ１０，＜不定＞）を受信する。このとき、すでにノード装置Ｒ２との間に入力リンクが存在するため、入力リンクに現在記憶されているスレッド

（Ｒ１，２）を受信スレッド（Ｒ１０，＜不定＞）に置き換える。また、すでにノード装置Ｒ４との間に出力リンクが存在するため、出力リンクに現在記憶されているスレッド（Ｒ６，３）を受信スレッド（Ｒ１０，＜不定＞）に置き換えるとともに、受信スレッド（Ｒ１０，＜不定＞）のスレッド長を１だけインクリメントしたスレッド（Ｒ１０，＜不定＞）をノード装置Ｒ４に通知する。このスレッドは、ノード装置Ｒ１０まで届けられる（図４参照）。このとき、ノード装置Ｒ１０はスレッド（Ｒ１０、＜不定＞）のループを検出するが、すでにスレッド長は＜不定＞であるため、ノード装置Ｒ１１にスレッド長の更新は通知されない。

【０１１０】

この結果、図５の点線に示すようなマージされたスレッドが残る。

【０１１１】

次に、図５の状態で、ノード装置Ｒ４において経路変更が起こり、次ホップノードがノード装置Ｒ９からノード装置Ｒ５に変わった場合について、図６および図７を参照しながら説明する。図６は後述するパターン１に対応するもので、図７は後述するパターン２に対応するものである。図６および図７には、通知されるスレッド等を示してある。

【０１１２】

さて、ノード装置Ｒ４の次ホップノードがノード装置Ｒ９からノード装置Ｒ５に変わったとすると、このとき、ノード装置Ｒ４は、まず、旧経路上のノード装置Ｒ９との間の出力リンクを削除し、記憶していたスレッド（Ｒ１０，＜不定＞）の削除を通知する（図６、図７参照）。スレッド削除の通知は、スレッド削除メッセージを送信することにより行なう。次に、ノード装置Ｒ４は、スレッド（Ｒ４，＜不定＞）を生成し、新経路上のノード装置Ｒ５との間に出力リンクを生成し、出力リンクにスレッド（Ｒ４，＜不定＞）を記憶し、生成スレッド（Ｒ４，＜不定＞）のスレッド長を１だけインクリメントしたスレッド（Ｒ４，＜不定＞）をノード装置Ｒ５に通知する。

【０１１３】

終段ノードであるノード装置Ｒ５は、スレッド（Ｒ４，＜不定＞）を受信すると、ノード装置Ｒ４との間に新たに入力リンクを生成し、パス長を＜不定＞に設定して、受信スレッドに対するＡＣＫを返す。このとき、受信スレッドに対するＡＣＫにはラベル割当て成功メッセージを用いる。このとき、ノード装置Ｒ５は、生成した入力リンクに記憶されるスレッド色を０にする。

【０１１４】

一方、旧経路上のスレッド削除は、各ノードで入力リンクと出力リンクの削除を伴いながら、ノード装置Ｒ１１まで通知されるとする。

【０１１５】

さて、旧経路上にスレッド削除通知がフォワードされ、新経路上にスレッドに対するＡＣＫがフォワードされるが、その到達順によって、動作が相違してくる。以下では、（１）旧経路上にフォワードされるスレッド削除通知が新経路上にフォワードしたスレッドに対するＡＣＫよりも早く初段ノードに到達するケースと、（２）旧経路上にフォワードされるスレッド削除通知が新経路上にフォワードしたスレッドに対するＡＣＫよりも後に初段ノードに到達するケースに分けて説明する。

【０１１６】

（パターン１）

まず、旧経路上にフォワードされるスレッド削除通知が新経路上にフォワードしたスレッドに対するＡＣＫよりも早く初段ノードに到達する場合を考える。

【０１１７】

ノード装置Ｒ９は、ノード装置Ｒ４からのスレッド削除通知を受けると、対応する入力リンクを削除し、対応する出力リンクも削除して、ノード装置Ｒ１０にスレッド削除通知をフォワードする。ノード装置Ｒ１０，Ｒ１１も同様である。

【０１１８】

ノード装置Ｒ１がノード装置Ｒ１１からのスレッド削除通知を受信すると、ノード装置Ｒ

１１との間の入力リンクを削除する。ノード装置Ｒ１は、削除した入力リンクに対応する出力リンクを持つが、初段フラグがオンであるため、この出力リンクは削除しない。このとき、すでに生成したスレッド（Ｒ１，０）を、出力リンクに記憶する。次に、スレッド（Ｒ１，０）のスレッド長を１だけインクリメントしたスレッド（Ｒ１，１）をノード装置Ｒ２に通知する。

【０１１９】

ノード装置Ｒ４は、ノード装置Ｒ５からＡＣＫを受信すると、パス長をＡＣＫに対応するスレッド（Ｒ４，＜不定＞）のスレッド長＝＜不定＞に設定し、上流ノードＲ３にＡＣＫを返し、ＡＣＫを返した上流隣接ノードとの間の入力リンクに記憶されるスレッド色を０にする。ノード装置Ｒ３は、ＡＣＫを受信すると、パス長をＡＣＫに対応するスレッド（Ｒ１０，＜不定＞）のスレッド長＝＜不定＞に設定し、上流ノードＲ２，Ｒ８にＡＣＫを返し、ＡＣＫを返した上流隣接ノードとの間の入力リンクに記憶されるスレッド色を０にする。ノード装置Ｒ８，Ｒ７も同様の動作を行う。

【０１２０】

初段ノードＲ６がＡＣＫを受信すると、パス長をＡＣＫに対応するスレッド（Ｒ６，０）のスレッド長＝０に設定する。

【０１２１】

ノード装置Ｒ２は、ノード装置Ｒ１からのスレッド（Ｒ１，１）を受信する前にノード装置Ｒ３からのＡＣＫを受信したとする。このとき、ノード装置Ｒ２は、パス長をＡＣＫに対応するスレッド（Ｒ１０，＜不定＞）のスレッド長＝＜不定＞に設定し、上流ノードＲ１にＡＣＫを返す。

【０１２２】

その後、ノード装置Ｒ２は、ノード装置Ｒ１からのスレッド（Ｒ１，１）を受信すると、対応する入力リンクのスレッドを受信スレッド（Ｒ１，１）に置き換える。このとき、パス長＝＜不定＞であるため、受信スレッドのスレッド長を１だけインクリメントしたスレッド（Ｒ１，２）をノード装置Ｒ３に通知する。このとき、スレッド（Ｒ１、２）は、更新メッセージ（ＵＰＤＡＴＥ）に入れられる。

【０１２３】

スレッド色＝Ｒ１のスレッドを受信したノード装置Ｒ２，Ｒ３，Ｒ４は、受信リンクに記憶するスレッドを受信スレッドに置き換えた後、スレッド色＝０のスレッドを記憶する入力リンクのスレッド長と受信スレッドのスレッド長のうち、最大の値を新たなパス長とする。パス長が更新されるなら下流ノードに更新後のパス長を１だけインクリメントしたスレッド色＝Ｒ１のスレッドを更新メッセージにより通知する。この通知はノード装置Ｒ５まで届き、最終的にノード装置Ｒ５のパス長は５に変更される。ノード装置Ｒ５は、受信スレッドに対するＡＣＫを返し、これはノード装置Ｒ１まで返される。

【０１２４】

ノード装置Ｒ１は、ノード装置Ｒ２からスレッド（Ｒ１，０）に対するＡＣＫを受信したときに、ラベルスイッチングが可能になる。

【０１２５】

（パターン２）

次に、旧経路上にフォワードされるスレッド削除通知が新経路上にフォワードしたスレッドに対するＡＣＫよりも後に初段ノードに到達する場合を考える。

【０１２６】

さて、ノード装置Ｒ４は、ノード装置Ｒ５からＡＣＫを受信すると、パス長をＡＣＫに対応するスレッド（Ｒ４，＜不定＞）のスレッド長＝＜不定＞に設定し、上流ノードＲ３にＡＣＫを返し、ＡＣＫを返した上流隣接ノードとの間の入力リンクに記憶されるスレッド色を０にする。ノード装置Ｒ３は、ＡＣＫを受信すると、パス長をＡＣＫに対応するスレッド（Ｒ１０，＜不定＞）のスレッド長＝＜不定＞に設定し、上流ノードＲ２，Ｒ８にＡＣＫを返し、ＡＣＫを返した上流隣接ノードとの間の入力リンクに記憶されるスレッド色を０にする。ノード装置Ｒ１，Ｒ２，Ｒ８，Ｒ７も同様の動作を行なう。

【０１２７】
初段ノードＲ６がＡＣＫを受信すると、パス長をＡＣＫに対応するスレッド（Ｒ６，０）のスレッド長＝０に設定する。
【０１２８】
ノード装置Ｒ１１は、ノード装置Ｒ１０からのスレッド削除通知を受信する前にノード装置Ｒ１からのＡＣＫを受信したとする。
【０１２９】
まず、ノード装置Ｒ１１がノード装置Ｒ１からのＡＣＫを受信すると、パス長をＡＣＫに対応するスレッド（Ｒ１０，＜不定＞）のスレッド長＝＜不定＞に設定し、上流ノードＲ１０にＡＣＫを返し、ＡＣＫを返した上流隣接ノードとの間の入力リンクに記憶されるスレッド色を０にする。次に、ノード装置Ｒ１１は、ノード装置Ｒ１０からのスレッド削除通知を受けるが、すでにパスの確定フラグはオンであるため、入力リンクを削除せず、またノード装置Ｒ１にもスレッド削除通知をしない。
【０１３０】
ノード装置Ｒ１０がノード装置Ｒ１１からのＡＣＫを受けると、すでに対応する出力リンクは存在しないため、ノード装置Ｒ１１にラベル解放メッセージ（ＲＥＬＥＡＳＥ）を送信する。ノード装置Ｒ１１がノード装置Ｒ１０からラベル解放メッセージを受けると、ノード装置Ｒ１０との間の入力リンクおよびノード装置Ｒ１との間の出力リンクを削除するとともに、ノード装置Ｒ１にラベル解放メッセージをフォワードする。
【０１３１】
ノード装置Ｒ１はラベル解放メッセージを受信すると、ノード装置Ｒ１１との間の入力リンクを削除する。このとき、初段フラグがオンであるため、入力リンクに対応する出力リンクは削除せず、パス長を＜不定＞から０に設定する。次に、パスの確定フラグはオンであり、パス長が増加しないため、スレッド色＝０とし、スレッド長にパス長を１だけインクリメントした値を入れたスレッド（０，１）をノード装置Ｒ２に通知する。このとき、スレッド（０，１）は、更新メッセージに入れられる。
【０１３２】
スレッド色＝０のスレッドを受信したノード装置Ｒ２，Ｒ３，Ｒ４は、受信リンクに記憶するスレッドを受信スレッドに置き換えた後、スレッド色＝０のスレッドを記憶する入力リンクのスレッド長と受信スレッドのスレッド長のうち、最大の値を新たなパス長とする。パス長が更新されるなら下流ノードに更新後のパス長を１だけインクリメントしたスレッド色＝０のスレッドを更新メッセージにより通知する。この通知はノード装置Ｒ５まで届き、最終的にノード装置Ｒ５のパス長は５に変更される。ノード装置Ｒ５はＡＣＫを返さない。
【０１３３】
この結果、最終的に図８の太線で示すようなパスＲ１→Ｒ２→Ｒ３→Ｒ４→Ｒ５ができる。
【０１３４】
以上の動作により、ループしているパスのどこで経路変更が起こっても、ループが解除されたときには、リトライすることなく正常なパスに復帰することが可能となる。
【０１３５】
ここで、ループしている間は、スレッドを通知中のノードでラベルの割当てが行なわれないため、ラベルリソースの節約が可能である。さらに、ラベル割り当てが行なわれないフローのパケットは代理初段ノードで廃棄するようにすれば、ネットワークリソースの節約も可能である。
【０１３６】
なお、図５に示される状態のとき、もし、ノード装置Ｒ１０で経路変更が起こり、ノード装置Ｒ１０の次ホップノードがノード装置Ｒ１１からノード装置Ｒ３に変わった場合には、ノード装置Ｒ１０は、まず、ノード装置Ｒ１１にスレッド（Ｒ１０、＜不定＞）の削除を通知するとともに、新たなスレッド（Ｒ１０：１、＜不定＞）を生成し、生成したスレ

ッド（Ｒ１０：１、＜不定＞）のスレッド長を１だけインクリメントしたスレッド（Ｒ１０：１、＜不定＞）をノード装置Ｒ３に通知する。このスレッドは、ノード装置Ｒ３、Ｒ４、Ｒ９、Ｒ１０と通知され、ノード装置Ｒ１０は、自分が生成したスレッド（Ｒ１０：１、＜不定＞）を受信する。このとき、スレッド長は＜不定＞のままかわらないため、ノード装置Ｒ３にスレッド長の更新は通知されない。

【０１３７】

（第２の具体例）

次に、本発明の一実施形態に係るノード装置により行われるラベルスイッチングパスの設定の動作についてもう一つの具体例を用いながら説明する。

【０１３８】

図９に、本実施形態に係るノード装置を接続して構成されたラベルスイッチングネットワークの他の例を示す。なお、図９において、ノード装置Ｒ１～Ｒ７の全てまたは一部に図示しない他のノード装置が接続されていてもよい。

【０１３９】

図９において、隣接ノード間のリンクはそれらのノード間でラベルスイッチングが可能であることを表し、それらのうち矢印付きのリンクは矢印の方向がフローの現在の次ホップを表している。なお、図９には、以降の説明で用いるスレッドの内容の例も併せて示してある。

【０１４０】

パス上のノード装置Ｒ１，Ｒ２，Ｒ３，Ｒ４は、パス長として、それぞれ、０，１，２，３を記憶しているものとする。

【０１４１】

さて、図９の構成において、ノード装置Ｒ１がパスの設定を開始するとする。

【０１４２】

ノード装置Ｒ１は、初段フラグをオンにし、ノード装置Ｒ２との間に出力リンクを生成し、スレッド（Ｒ１，０）を生成し、出力リンクにスレッド（Ｒ１，０）を記憶し、ノード装置Ｒ２にスレッド（Ｒ１，１）を通知する。

【０１４３】

ノード装置Ｒ２は、スレッド（Ｒ１，１）を受信すると、ノード装置Ｒ１との間に新たに入力リンクを生成し、スレッド（Ｒ１，１）を記憶する。次に、ノード装置Ｒ３との間に出力リンクを生成し、出力リンクに受信スレッド（Ｒ１，１）を記憶し、受信スレッド（Ｒ１，１）のスレッド長を１だけインクリメントしたスレッド（Ｒ１，２）をノード装置Ｒ３に通知する。ノード装置Ｒ４も同様の動作を行なう。

【０１４４】

終段ノードであるノード装置Ｒ５は、ノード装置Ｒ４からスレッド（Ｒ１，４）を受信すると、ノード装置Ｒ４との間に新たに入力リンクを生成し、スレッド（Ｒ１，４）を記憶する。次に、パス長を４に設定して、受信スレッドに対するＡＣＫを返す。このとき、受信スレッドに対するＡＣＫにはラベル割当て成功メッセージを用いる。

【０１４５】

ノード装置Ｒ４は、ノード装置Ｒ５からＡＣＫを受信すると、パス長をＡＣＫに対応するスレッド（Ｒ１，３）のスレッド長＝３に設定し、上流ノードＲ２にＡＣＫを返す。ノード装置Ｒ２も同様の動作を行なう。

【０１４６】

初段ノードＲ１がＡＣＫを受信すると、パス長をＡＣＫに対応するスレッド（Ｒ１，０）のスレッド長＝０に設定する。

【０１４７】

これにより、図１０の太線に示すようなラベルスイッチングパスＲ１→Ｒ２→Ｒ３→Ｒ４→Ｒ５が設定される。

【０１４８】

次に、図１０の状態で、ノード装置Ｒ２において経路変更が起こり、次ホップノードがノ

ード装置Ｒ３からノード装置Ｒ６に変わった場合について、図１１を参照しながら説明する。図１１には、通知されるスレッド等を示してある。

【０１４９】

さて、ノード装置Ｒ２の次ホップノードがノード装置Ｒ３からノード装置Ｒ６に変わったとすると、このとき、ノード装置Ｒ２は、現在のパス長に等しい長さを持つスレッド（Ｒ２，１）を生成し、新経路上のノード装置Ｒ６との間に出力リンクを生成し、出力リンクにスレッド（Ｒ２，１）を記憶し、生成スレッド（Ｒ２，１）のスレッド長を１だけインクリメントしたスレッド（Ｒ２，２）をノード装置Ｒ６に通知する。

【０１５０】

ノード装置Ｒ６は、スレッド（Ｒ２，２）を受信すると、ノード装置Ｒ２との間に新たに入力リンクを生成し、スレッド（Ｒ２，２）を記憶する。次に、ノード装置Ｒ７との間に出力リンクを生成し、出力リンクに受信スレッド（Ｒ２，２）を記憶し、受信スレッド（Ｒ２，２）のスレッド長を１だけインクリメントしたスレッド（Ｒ２，３）をノード装置Ｒ７に通知する。ノード装置Ｒ７も同様の動作を行なう。

【０１５１】

ノード装置Ｒ４がスレッド（Ｒ２，４）を受信すると、ノード装置Ｒ７との間に新たに入力リンクを生成し、スレッド（Ｒ２，４）を記憶する。次に、すでに同じフローに対してノード装置Ｒ５との間に出力リンクが存在しているため、フローのマージを行なう。このとき、マージ後のパス長が３から４に増加するため、出力リンクに受信スレッド（Ｒ２，４）を記憶し、ノード装置Ｒ５に受信スレッド（Ｒ２，４）のスレッド長を１だけインクリメントしたスレッド（Ｒ２，５）を送る。

【０１５２】

終段ノードＲ５が、スレッド（Ｒ２，５）を受信すると、パス長を受信スレッドのスレッド長（＝５）に設定し、ノード装置Ｒ４にスレッドに対するＡＣＫを返す。

【０１５３】

ノード装置Ｒ４は、ノード装置Ｒ５からＡＣＫを受信すると、パス長をＡＣＫに対応するスレッド（Ｒ２，４）のスレッド長＝４に設定し、上流ノードＲ７にＡＣＫを返す。ノード装置Ｒ７は、ＡＣＫを受信すると、パス長をＡＣＫに対応するスレッド（Ｒ２，３）のスレッド長＝３に設定し、上流ノードＲ６にＡＣＫを返す。ノード装置Ｒ６も同様の動作を行なう。

【０１５４】

ノード装置Ｒ２がノード装置Ｒ６からＡＣＫを受信すると、新経路上にラベルスイッチングパスＲ１→Ｒ２→Ｒ６→Ｒ７→Ｒ４→Ｒ５が設定される。さらに、ノード装置Ｒ２は旧経路上のノード装置Ｒ３との間の出力リンクを削除するとともに、ノード装置Ｒ３にラベル解放メッセージを送信する。

【０１５５】

ノード装置Ｒ３は、ラベル解放メッセージを受信すると、ノード装置Ｒ２との間の入力リンクとノード装置Ｒ４との間の出力リンクを削除するとともに、ノード装置Ｒ４にラベル解放メッセージをフォワードする。

【０１５６】

ノード装置Ｒ４は、ラベル解放メッセージを受信すると、ノード装置Ｒ３との間の入力リンク削除するが、他に入力リンクがあるため、出力側リンクは削除しない。また、入力リンク削除後のパス長は増加しないため、下流ノードにパス長の変更も行なわない。

【０１５７】

この結果、図１２の太線に示すようなパスができる。

【０１５８】

ここで、もし、旧経路上の物理リンクがダウンすることによりラベル解放メッセージをフォワードできない場合でも、物理リンクの下流のノードがダウンに気付いたとき、ノード装置Ｒ４はノード装置Ｒ３からラベル解放メッセージを受信した場合と同じ動作を行なえば、結果は同じである。

【０１５９】
次に、図１２の状態で、ノード装置Ｒ２において再び経路変更が起こり、次ホップノードがノード装置Ｒ６からノード装置Ｒ３に変わった場合について、図１３を参照しながら説明する。図１３には、通知されるスレッド等を示してある。
【０１６０】
さて、ノード装置Ｒ２の次ホップノードがノード装置Ｒ６からノード装置Ｒ３に変わったとすると、このとき、ノード装置Ｒ２は、現在のパス長と同じ長さのスレッド（Ｒ２，１）を生成し、新経路上のノード装置Ｒ３との間に出力リンクを生成し、出力リンクにスレッド（Ｒ２，１）を記憶し、生成スレッド（Ｒ２，１）のスレッド長を１だけインクリメントしたスレッド（Ｒ２，２）をノード装置Ｒ３に通知する。
【０１６１】
ノード装置Ｒ３は、スレッド（Ｒ２，２）を受信すると、ノード装置Ｒ２との間に新たに入力リンクを生成し、スレッド（Ｒ２，２）を記憶する。次に、ノード装置Ｒ４との間に出力リンクを生成し、出力リンクに受信スレッド（Ｒ２，２）を記憶し、受信スレッド（Ｒ２，２）のスレッド長を１だけインクリメントしたスレッド（Ｒ２，３）をノード装置Ｒ４に通知する。
【０１６２】
ノード装置Ｒ４がスレッド（Ｒ２，３）を受信すると、ノード装置Ｒ３との間に新たに入力リンクを生成し、スレッド（Ｒ２，３）を記憶する。次に、すでに同じフローに対してノード装置Ｒ５との間に出力リンクが存在しているため、フローのマージを行なう。このとき、マージ後のパス長は４のまま増加しないため、ただちにノード装置Ｒ３にＡＣＫを返す。
【０１６３】
ノード装置Ｒ３は、ノード装置Ｒ４からＡＣＫを受信すると、パス長をＡＣＫに対応するスレッド（Ｒ２，２）のスレッド長＝２に設定し、上流ノードＲ２にＡＣＫを返す。
【０１６４】
ノード装置Ｒ２がノード装置Ｒ３からＡＣＫを受信すると、新経路上にラベルスイッチングパスＲ１→Ｒ２→Ｒ３→Ｒ４→Ｒ５が設定される。さらに、ノード装置Ｒ２は旧経路上のノード装置Ｒ６との間の出力リンクを削除するとともに、ノード装置Ｒ６にラベル解放メッセージを送信する。
【０１６５】
ノード装置Ｒ６は解放メッセージを受信すると、ノード装置Ｒ４にラベル解放メッセージをフォワードする。ノード装置Ｒ７も同様である。
【０１６６】
ノード装置Ｒ４は、ラベル解放メッセージを受信すると、ノード装置Ｒ７との間の入力リンク削除するが、他に入力リンクがあるため、出力側リンクは削除しない。また、入力リンク削除後のパス長は４から３に減少するため、下流ノードにパス長を３に設定し、ノード装置Ｒ５にパス長を１だけインクリメントしたスレッド（０，４）を送る。
【０１６７】
スレッド色＝０のスレッドを受信したノード装置Ｒ５は、パス長を受信スレッド長に設定するが、ＡＣＫは返さない。
【０１６８】
この結果、図１０の太線に示すようなパスが再び設定される。
【０１６９】
もし、新経路がループになる場合には、ループが解消されるまでスレッドに対するＡＣＫが返ってこない。この場合には、旧経路上のパスを解放せずに使い続けることができる。
【０１７０】
また、経路変更時において、旧経路上のパスを解放してから新経路上にパス設定を行なうようにする場合でも本発明は有効である。
【０１７１】

以下では、本実施形態における、内部データベースの例、スレッド受信時のノード装置の動作手順の例、経路変更が起こったノード装置の動作手順の例、メッセージのフォーマットの例についてそれぞれ説明する。

【０１７２】

まず、本発明のノード装置においてループ検出に使用される内部データベースの構成の一例について説明する。

【０１７３】

図１４、図１５に、内部データベースの構成の一例を示す。

【０１７４】

個々のラベルスイッチングパスの情報は、パスエントリ（ｐａｔｈ＿ｅｎｔｒｙ）に格納され、パスリスト（ｐａｔｈ＿ｌｉｓｔ）（図１４の１０）は、パスエントリのリンクドリストである。

【０１７５】

パスエントリは、入力リンクリスト（ｉｎｐｕｔ＿ｌｉｎｋ＿ｌｉｓｔ）、出力リンクリスト（ｏｕｔｐｕｔ＿ｌｉｎｋ＿ｌｉｓｔ）、フロー識別子（ｆｌｏｗ＿ｉｄ）、パス長（ｌｅｎｇｔｈ）、確定フラグ（ｃｆｌａｇ）、初段フラグ（ｉｎｇｒｅｓｓ＿ｆｌａｇ）を持つ。

【０１７６】

入力リンクリストは、パスの個々の入力側リンクの情報を格納するリンクエントリ（ｌｉｎｋ＿ｅｎｔｒｙ）（図１４の２０）のリンクドリストである。

【０１７７】

出力リンクリストは、パスの個々の出力側リンクの情報を格納するリンクエントリ（ｌｉｎｋ＿ｅｎｔｒｙ）（図１４の３０）のリンクドリストである。マルチキャストの場合には、複数のリンクエントリを持つ。また、ユニキャストの場合においても経路変更時には、旧経路用と新経路用の２つのリンクエントリを持つ場合がある。

【０１７８】

フロー識別子（またはフローＩＤ）は、このパス上でラベルスイッチングされるフローの情報を格納する。

【０１７９】

パス長は、最も遠い初段ノードからのホップ数を記憶する。

【０１８０】

確定フラグは、ＡＣＫ待ちのスレッドがあるかどうかを示すためのフラグである。

【０１８１】

初段フラグは、自分がパスの初段ノードかどうかを示すためのものである。

【０１８２】

リンクエントリは、インターフェース番号（ｉｎｔｅｒｆａｃｅ＿ｎｕｍｂｅｒ）、リンク識別子（ｌｉｎｋ＿ｉｄ），隣接ノードアドレス（ｎｅｉｇｈｂｏｒ＿ａｄｄｒｅｓｓ），スレッド（ｔｈｒｅａｄ）、パスエントリ（ｐａｔｈ＿ｅｎｔｒｙ）の情報を記憶する。

【０１８３】

インターフェース番号は、ノード内で各論理インターフェース毎に一意に割り当てられる番号である。

【０１８４】

リンク識別子は、隣接ノード間でリンクを識別するためのものである。

【０１８５】

隣接ノードアドレスには、リンクの相手先のノードのアドレスを格納する。

【０１８６】

スレッドは、スレッドエントリ（図１４の４０）へのポインタである。スレッドエントリは、スレッド色（ｃｏｌｏｒ）、スレッド長（ｌｅｎｇｔｈ）の属性を持つ。スレッドエントリは、同じパスの入力／出力リンクエントリ間で共有される。

【０１８７】
リンクエントリ中のパスエントリは、そのリンクエントリを入力または出力リンクリストに持つパスエントリを指すポインタである。これにより、各リンクからパスエントリの情報にアクセスできるとともに、入力リンク側から出力リンク側の、また、出力リンク側から入力リンク側への情報にもアクセスできる。
【０１８８】
次に、図１５に、あるパスエントリにおいて、入力／出力リンクエントリ間で共有されるスレッドエントリの一例を示す。図中、２０－１～２０－３は入力側リンクのリンクエントリであり、３０－１は出力側リンクのリンクエントリであり、４０－１～４０－３はスレッドエントリである。
【０１８９】
ここでは、３つのスレッド（５，５），（１０，５），（１０１，２）がマージされている状態を示す。このとき、スレッド（１０，５）が入力リンクエントリ、出力リンクエントリで共有されている。これは、スレッド長が最大（＝５）のスレッド（５，５），（１０，５）のうち、スレッド色が最大のものがマージ後のスレッドとして選ばれていることを表している。
【０１９０】
次に、ノード装置がスレッドを受信したときの処理手順について説明する。
【０１９１】
図１６に、ノード装置がスレッドを受信したときの処理手順を表すフローチャートの一例を示す。
【０１９２】
ここでは、パス長をスレッドのＡＣＫを待たずに更新する場合について説明する。
【０１９３】
まず、Ｆを受信フロー識別子、ｉを受信インターフェース番号、ｊを受信リンク識別子、Ｔを受信スレッド、Ｎをスレッドを送信した隣接ノードとする（ステップＳ１）。
【０１９４】
次に、スレッドの入力処理を行なう（ステップＳ２）。入力処理では、（受信インターフェース番号ｉ、受信リンク識別子ｊ）、で一意に指定される入力リンクＬｉを探し、もし、エントリが見つからなければ新たに生成し、Ｌｉに（Ｎ，Ｔ）を登録する。次に、パスをマージしないなら新たにパスエントリＰを作り、Ｐに、（フロー識別子Ｆ，入力リンクＬｉ）の組を登録する。もし、パスをマージするなら、フロー識別子Ｆを持つすでに存在するパスエントリＰに入力リンクＬｉを登録する。
【０１９５】
次に、スレッドの処理を行なう。
【０１９６】
まず、終段ノード、すなわち、自ノードがパスの終了点であるかどうかをチェックする（ステップＳ３）。もし終段ノードであれば、受信スレッド長をパス長に設定し（Ｐ．ｌｅｎｇｔｈ：＝Ｔ．ｌｅｎｇｔｈ）（ステップＳ７）、ＡＣＫ通知処理を行って（ステップＳ９）、処理を終了する。
【０１９７】
一方、終段ノードでなければ、次に確定フラグがオン（Ｐ．ｃｆｌａｇ＝＝ｏｎ？）であるかどうかをチェックする（ステップＳ４）。もしオンであれば、ＡＣＫ待ちのスレッドがないため、パス長がスレッド長以上であれば（Ｐ．ｌｅｎｇｔｈ＞＝Ｔ．ｌｅｎｇｔｈ？）（ステップＳ８）、ＡＣＫ通知処理を行って（ステップＳ９）、処理を終了し、そうでなければ、ループでないかどうかこのノードでは判断できないため、スレッド通知処理を行なって（ステップＳ６）、処理を終了する。
【０１９８】
一方、ステップＳ４において、確定フラグがオフであれば、もし、スレッドＴがループを形成し、かつ、パス長が＜不定＞でないかまたはスレッド長Ｔ．ｌｅｎｇｔｈがしきい値

Ｌｍａｘ以上であるがＮｏならば（ステップＳ５）、スレッド通知処理を行なって（ステップＳ６）、処理を終了する。

【０１９９】

そうでなく、もし、スレッドＴがループを形成し、かつ、パス長が＜不定＞でないかまたはスレッド長Ｔ．ｌｅｎｇｔｈがしきい値Ｌｍａｘ以上である場合には、パス長が＜不定＞でなければ（ステップＳ１０）、パス長修正処理（ステップＳ１１）を行い、入力リンクＬｉに記憶したスレッドのスレッド色を０に設定して（Ｔ．ｃｏｌｏｒ：＝０）（ステップＳ１２）、処理を終了する。パス長が＜不定＞であれば、入力リンクＬｉに記憶したスレッドのスレッド色を０に設定して（Ｔ．ｃｏｌｏｒ：＝０）（ステップＳ１２）、処理を終了する。

【０２００】

ＡＣＫ通知処理では、パスの確定フラグをオンにし（Ｐ．ｃｆｌａｇ：＝ｏｎ）、隣接ノードＮにＡＣＫを返す。

【０２０１】

スレッド通知処理では、パスの確定フラグをｏｆｆにし（Ｐ．ｃｆｌａｇ：＝ｏｆｆ）、必要であればスレッドのマージを行ない、Ｐの出力リンクＬｏが存在しなければ、新たに生成し、Ｌｏ．ｎｅｉｇｈｂｏｒ＿ａｄｄｒｅｓｓで指定される隣接ノードにスレッドを通知する。なお、マージ対象となるスレッドは、入力リンクに登録されているスレッドのうち、スレッド長が０でないものである。

【０２０２】

スレッド長修正処理では、パス長を＜不定＞とし、スレッド長＝パス長のスレッドをパスＰの出力リンクＬｏのＬｏ．ｎｅｉｇｈｂｏｒ＿ａｄｄｒｅｓｓで指定される下流隣接ノードに通知する。

【０２０３】

次に、経路変更が起こったノード装置の動作手順について説明する。

【０２０４】

図１７に、経路変更が起こったノード装置の処理手順を表すフローチャートの一例を示す。

【０２０５】

ここでは、パス長をスレッドのＡＣＫを待たずに更新する場合について説明する。

【０２０６】

この処理は、経路変更の対象となるフローのフロー識別子を持つパスのそれぞれに対して行なわれる。

【０２０７】

現在、経路変更処理の対象となっているパスをＰとする（ステップＳ２１）。

【０２０８】

まず、このノードが経路変更後に終段ノードとなるかどうかをチェックする（ステップＳ２２）。もし、終段ノードであれば、ただちに処理を終了する。

【０２０９】

終段ノードでなければ、パス長がしきい値Ｌｍａｘ以上かどうかをチェック（Ｐ．ｌｅｎｇｔｈ＞＝Ｌｍａｘ？）（ステップＳ２３）し、もし、そうであれば、ただちに処理を終了する。

【０２１０】

そうでなければ、スレッドＴを新たに生成し、パス長をスレッド長にセットする（Ｔ．ｌｅｎｇｔｈ：＝Ｐ．ｌｅｎｇｔｈ）（ステップＳ２４）。

【０２１１】

次に、パスＰの確定フラグがオフであれば（Ｐ．ｃｆｌａｇ＝＝ｏｆｆ？）（ステップＳ２５）、旧経路上にＡＣＫ待ちのスレッドがあるため、旧経路のスレッド削除処理を行い（ステップＳ２６）、そして、新経路上へのスレッド通知処理を行なう（ステップＳ２７）。

【０２１２】
旧経路のスレッド削除処理では、Ｐの旧経路上の出力リンクＬｏの下流隣接ノードＬｏ．ｎｅｉｇｈｂｏｒ＿ａｄｄｒｅｓｓにスレッドの削除が通知される。
【０２１３】
一方、パスＰの確定フラグがオンであれば（Ｐ．ｃｆｌａｇ＝＝ｏｆｆ？）、旧経路上にＡＣＫ待ちのスレッドがないため、旧経路に対しては何もせずに、新経路上へのスレッド通知処理を行なう（ステップＳ２７）。
【０２１４】
新経路上へのスレッド通知処理では、パスＰの確定フラグをオフにし（Ｐ．ｃｆｌａｇ：＝ｏｆｆ）、次に、新経路上の出力リンクＬｏの下流隣接ノードＬｏ．ｎｅｉｇｈｂｏｒ＿ａｄｄｒｅｓｓにスレッドの通知を行なう。
【０２１５】
次に、ノード装置においてループ検出のためにノード間でやりとりされるラベル配送プロトコルメッセージのフォーマットについて説明する。
【０２１６】
図１８に、メッセージのフォーマットの一例を示す。
【０２１７】
図１８において、隣接ノード識別子は、メッセージの送信ノードに関する情報を表す。隣接ノード識別子として、Ｌ３アドレス、またはＬ３アドレス＋隣接ノードがローカルに割り当てたＩＤ（ｌａｂｅｌ　ｓｐａｃｅ　ｉｄ）が使用可能である。ｌａｂｅｌ　ｓｐａｃｅ　ｉｄは、Ｌ３アドレスがルータＩＤのとき、同じノードの複数のインターフェース間や同じインターフェースにつながる複数の隣接ノードとの間でラベル領域の共有や分割を指定する。
【０２１８】
メッセージタイプには、ラベル配送プロトコルで使用されるメッセージの種類を記入する。メッセージタイプには、ラベル割当て要求メッセージ（ＳＥＴＵＰ）、ラベル解放メッセージ（ＲＥＬＥＡＳＥ）、更新メッセージ（ＵＰＤＡＴＥ）、キャンセルメッセージ、ラベル割当て成功メッセージ（ＡＣＫ）、更新成功メッセージ、トリガメッセージがある。
【０２１９】
メッセージ長には、メッセージ長以降のフィールドの全体の長さを入れる。
【０２２０】
フロー識別子には、フローを識別するための情報で、例えば先に例示した（もしくは前述の文献や特願平１０－５０４４にて示されている）フロー識別子が使用可能である。
【０２２１】
フロー識別子は、フロータイプとタイプ別フロー定義から構成される。
【０２２２】
フロー識別子は、少なくともラベル割当て要求メッセージに含まれる。
【０２２３】
リンク識別子は、隣接ノード間で同じフロー識別子を持つ複数のパスを識別するために使用される、各ノードの各インターフェース内で一意な識別子である。リンク識別子としてラベルそのものを使用してもよい。
【０２２４】
スレッド色、スレッド長には、通知するスレッドの情報を入れる。
【０２２５】
スレッド色は、スレッドを生成したノードのアドレスとノード内でローカルに割り当てたローカルＩＤの組で指定される。ノードがローカルＩＤを割り当てる場合には、同じフローに対して以前割当てた値を使用しないようにする。最も単純な方法は、割当てる度にノードが持つローカルＩＤをインクリメントすることである。
【０２２６】

スレッド長は、初段からのホップカウントを表す。ただし、経路がループしている場合には、第１の具体例で示したように、スレッドを生成したノードでスレッド長の修正が行なわれる。

【０２２７】

ラベルは、ラベルスイッチングされるパケットのヘッダに付けられる値で、ｄｏｗｎｓｔｒｅａｍ ｏｎ ｄｅｍａｎｄ ａｌｌｏｃａｔｉｏｎのときは、ラベル割当て成功メッセージで、ｕｐｓｔｒｅａｍ ａｌｌｏｃａｔｉｏｎ，ｄｏｗｎｓｔｒｅａｍ ａｌｌｏｃａｔｉｏｎのときは、ラベル割当て要求メッセージで隣接ノードに通知される。

【０２２８】

ラベルが未設定のパス上をスレッドが流れる場合で、ｄｏｗｎｓｔｒｅａｍ ｏｎ ｄｅｍａｎｄ ａｌｌｏｃａｔｉｏｎ／ｄｏｗｎｓｔｒｅａｍ ａｌｌｏｃａｔｉｏｎの場合には、ラベル割当て要求メッセージを用いて上流から下流にスレッドが転送される。ｕｐｓｔｒｅａｍ ａｌｌｏｃａｔｉｏｎの場合には、ラベル割当て成功メッセージを用いて上流から下流にスレッドが転送される。

【０２２９】

ラベルが設定済みのパス上をスレッドが流れる場合には、更新メッセージを使用して、上流から下流にスレッドが転送される。

【０２３０】

ラベルを解放する場合には、ラベル解放メッセージが用いられる。

【０２３１】

ラベル解放メッセージには、スレッドのフィールドはなくてもよい。

【０２３２】

スレッドの削除を通知する場合には、キャンセルメッセージを用いる。キャンセルメッセージがラベルが未設定のパス上を流れる場合には、スレッドが削除されるとともに、リンクも削除される。キャンセルメッセージがラベルが設定済みのパス上を流れる場合には、スレッドは削除されるがリンクは削除されない。

【０２３３】

スレッドに対するＡＣＫがラベルが未設定のパス上を流れる場合には、ラベル割当て成功メッセージが用いられる。スレッドに対するＡＣＫがラベルが設定済みのパス上を流れる場合には、更新成功メッセージが使用される。ラベル割当て成功メッセージ、更新成功メッセージにはスレッド情報を入れなくてもよい。

【０２３４】

トリガメッセージは、終段ノード主導でパスを生成する場合に、経路変更ノードにおいて、新経路上の下流隣接ノードにパス設定の要求を通知するために使用する。

【０２３５】

以上により、ループ回避のためのリトライ機構がまったく必要がなく、また、１つのスレッドの更新が個々の更新毎のＡＣＫを待つことなく実行できるため、高速なパス設定が実現できる。

【０２３６】

また、スレッドはマージできるため、ループ回避のために必要なメッセージ数の削減が可能となり、さらに、パスベクトルのようにメッセージ長がパス長が増えるとともに増加せずに一定値で押えられるため、ネットワークリソースの節約が可能となる。

【０２３７】

また、経路変更時に新経路がループしている場合には、旧経路上のパスを使い続けることができる。また、ループしている間は、スレッドを通知中のノードでラベルの割当てが行なわれないため、ラベルリソースの節約が可能である。さらに、ラベル割り当てが行なわれないフローのパケットは代理初段ノードで廃棄するようにすれば、ネットワークリソースの節約も可能である。

【０２３８】

また、スレッド通知中に、途中のノードやリンクの故障によりにスレッドの情報が失われ

ても、ラベル配送プロトコルが隣接ノードとの間のセッションの切断を自動的に検出できれば、スレッド転送のデッドロックやスレッド情報の矛盾が生じないため、ロバストなループ回避メカニズムが提供できる。

【０２３９】

さらに、いったんパスができると、スレッド色は記憶する必要がないため、ネットワークが定常状態の場合に記憶する情報量を減らすことができる。

【０２４０】

これまでの説明では、一旦、ラベルが割り当てられたパスに対しては、ラベル割り当てメッセージは送信しないようなハードステート型のプロトコルを前提にしていたが、本発明は、いったんラベルが割り当てられたパスに対しても、定期的にラベル割り当てメッセージを送信するようなソフトステート型のプロトコルに関しても有効である。ソフトステート型の場合には、一定期間ラベル割り当てメッセージを受信しない場合にはラベルが解放される。

【０２４１】

なお、以上の各機能は、ソフトウェアとしても実現可能である。

【０２４２】

また、本実施形態は、コンピュータに所定の手順を実行させるための（あるいはコンピュータを所定の手段として機能させるための、あるいはコンピュータに所定の機能を実現させるための）プログラムを記録したコンピュータ読取り可能な記録媒体として実施することもできる。

【０２４３】

本発明は、上述した実施の形態に限定されるものではなく、その技術的範囲において種々変形して実施することができる。

【０２４４】

【発明の効果】

本発明によれば、ラベルスイッチングパスの設定に関わるメッセージが、経由する各ノードにメッセージ処理中である旨を記憶させながら、パスがループを形成しないことが確認できるノードまで転送され、そこから返答メッセージが返ってくる過程で、順次パスの経路上の各ノードのメッセージ処理中状態が解消され、そこではじめてパスの設定が確立されていくため、ループを形成するか否かの判断は、処理中状態となっているメッセージ（例えば、スレッド）のみを対象とすれば良く、ループ検出が簡単に行える。また、経路変更時も、古いパスは既に確立している（ループ判断の対象外である）ため、新しいパスが古いパスとぶつかった場合にも誤ってループと判断することがなくなる。したがって、新しいパスが確立されるまで古いパスへパケットを流し続けることも可能となり、新しいパスが古いパスと一部経路を同じくする場合にその部分のリンク情報を利用し続けることも可能となる。

【図面の簡単な説明】

【図１】本発明の一実施形態に係るノード装置の接続状態の一例を示す図

【図２】設定されたラベルスイッチングパスの一例を示す図

【図３】経路の一部に変更が起こったときのノード装置の動作例を説明するための図

【図４】経路の一部に変更が起こったときのノード装置の動作例を説明するための図

【図５】経路変更されたラベルスイッチングパスの一例を示す図

【図６】経路の一部に変更が起こったときのノード装置の動作例を説明するための図

【図７】経路の一部に変更が起こったときのノード装置の動作例を説明するための図

【図８】経路変更されたラベルスイッチングパスの一例を示す図

【図９】本発明の一実施形態に係るノード装置の接続状態の他の例を示す図

【図１０】設定されたラベルスイッチングパスの一例を示す図

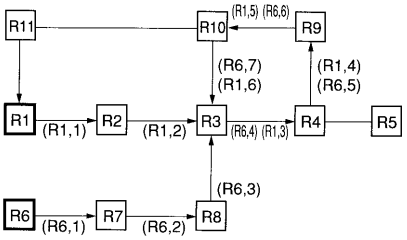
【図１１】経路の一部に変更が起こったときのノード装置の動作例を説明するための図

【図１２】経路変更されたラベルスイッチングパスの一例を示す図

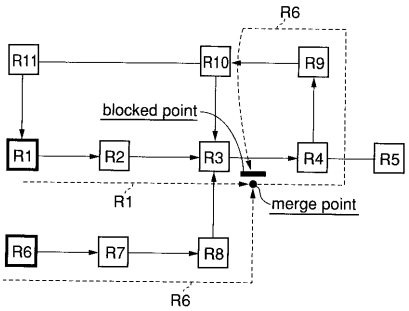
【図１３】経路の一部に変更が起こったときのノード装置の動作例を説明するための図

【図１４】ループ検出に使用される内部データベースの構成例を示す図
【図１５】ループ検出に使用される内部データベースの構成例を示す図
【図１６】スレッド受信時のノード装置の動作の一例を示すフローチャート
【図１７】経路変更が起こったノード装置の動作の一例を示すフローチャート
【図１８】ループ検出のためにノード装置間でやりとりされるメッセージのフォーマットの一例を示す図
【図１９】従来のループ検出方法を説明するための図
【図２０】従来のループ検出方法を説明するための図
【符号の説明】
Ｒ１～Ｒ１１…ノード装置

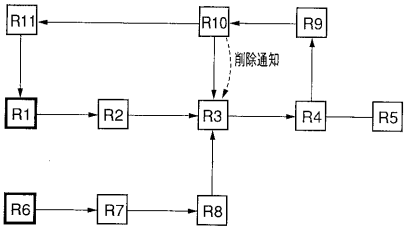
【図１】



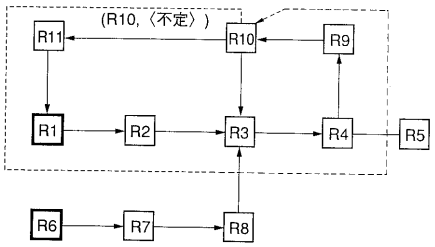
【図２】



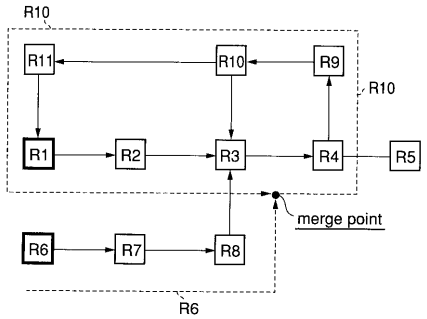
【図３】



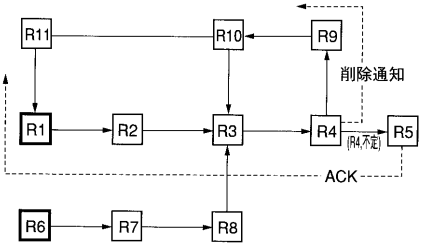
【図４】



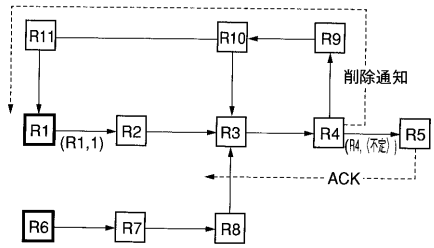
【図５】



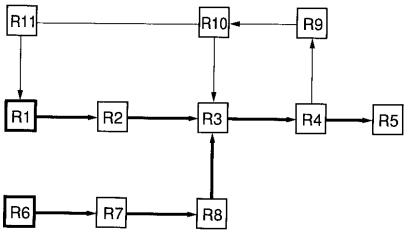
【図７】



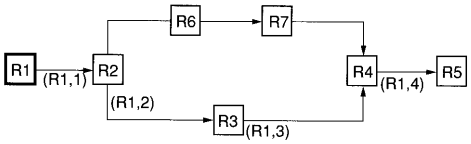
【図６】



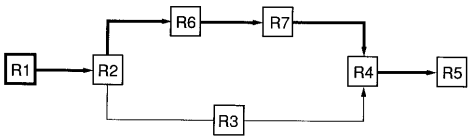
【図８】



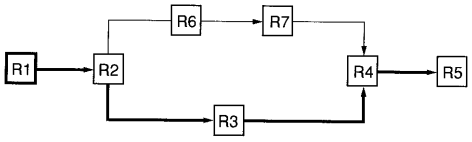
【図９】



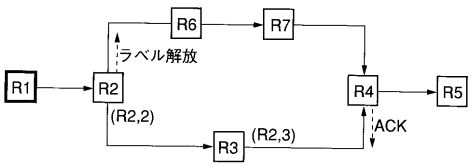
【図１２】



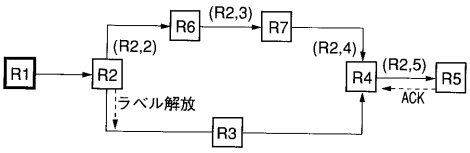
【図１０】



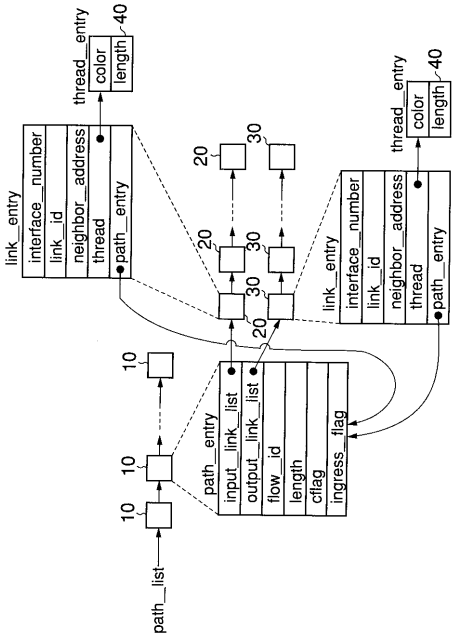
【図１３】



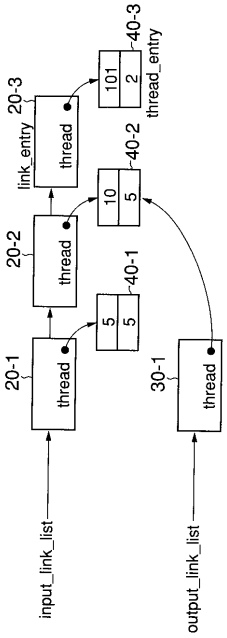
【図１１】



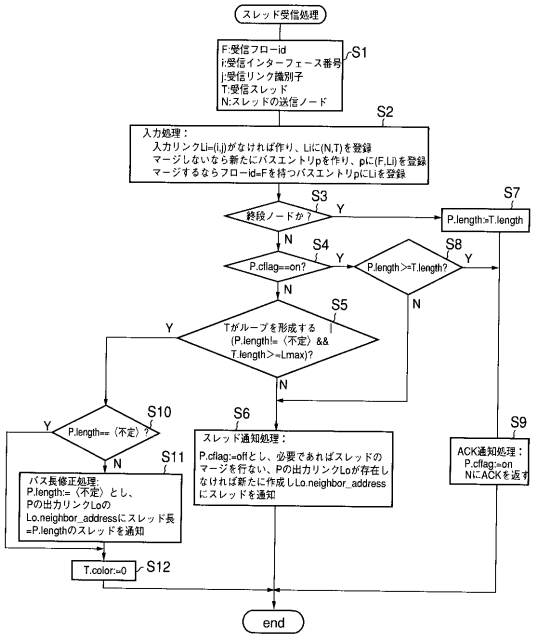
【図１４】



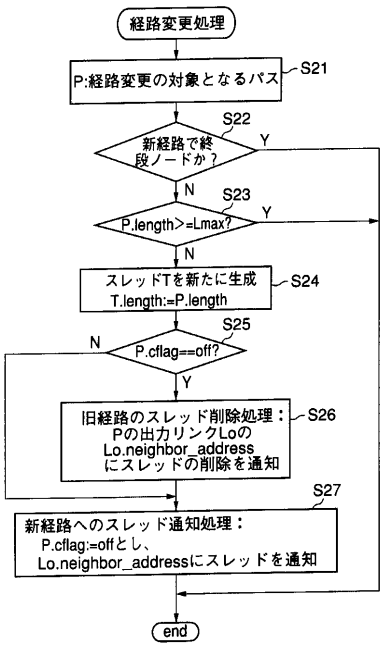
【図１５】



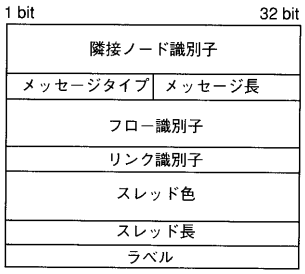
【図１６】



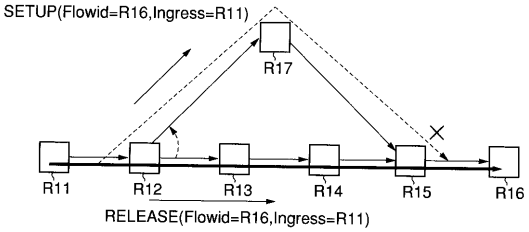
【図１７】



【図１８】



【図１９】



【図２０】

R15's table

input link (i/f:label)	flowid	ingress	output link (i/f:label)
I1415:x1415	R16	R11	I1516:x1516

フロントページの続き

(74)代理人　100070437
弁理士　河井　将次
(72)発明者　大場　義洋
神奈川県川崎市幸区小向東芝町１番地　株式会社東芝研究開発センター内
(72)発明者　勝部　泰弘
神奈川県川崎市幸区小向東芝町１番地　株式会社東芝研究開発センター内

審査官　石井　研一

(56)参考文献　特開２０００－０４９８５４（ＪＰ，Ａ）
特開平１１－２４３４１６（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.[7]，ＤＢ名)
H04L 12/56
H04L 12/28