

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4323524号
(P4323524)

(45) 発行日 平成21年9月2日(2009.9.2)

(24) 登録日 平成21年6月12日(2009.6.12)

(51) Int. Cl. F I
H04L 12/56 (2006.01) H04L 12/56 400Z

請求項の数 15 (全 21 頁)

(21) 出願番号	特願2006-545273 (P2006-545273)	(73) 特許権者	598036300
(86) (22) 出願日	平成16年11月9日 (2004.11.9)		テレフオンアクチーボラゲット エル エム エリクソン (パブル)
(65) 公表番号	特表2007-515125 (P2007-515125A)		スウェーデン国 ストックホルム エスー
(43) 公表日	平成19年6月7日 (2007.6.7)		164 83
(86) 国際出願番号	PCT/SE2004/001637	(74) 代理人	100076428
(87) 国際公開番号	W02005/060162		弁理士 大塚 康德
(87) 国際公開日	平成17年6月30日 (2005.6.30)	(74) 代理人	100112508
審査請求日	平成18年7月24日 (2006.7.24)		弁理士 高柳 司郎
(31) 優先権主張番号	10/742, 731	(74) 代理人	100115071
(32) 優先日	平成15年12月19日 (2003.12.19)		弁理士 大塚 康弘
(33) 優先権主張国	米国 (US)	(74) 代理人	100116894
			弁理士 木村 秀二
		(74) 代理人	100130409
			弁理士 下山 治

最終頁に続く

(54) 【発明の名称】 インターネットプロトコル (IP) ベースネットワークにおけるリンクスコープタイプ管理オブジェクトの集中コンフィグレーション

(57) 【特許請求の範囲】

【請求項1】

少なくとも1つの管理局と、ネットワークノードのセット及び前記ネットワークノード間と前記管理局と前記ネットワークノード間の複数の通信リンクを有するインターネットプロトコル (IP) ベースのネットワークのリンク - スコープコンフィグレーションを行う方法であって、

前記ネットワークのトポロジグラフを準備するステップと、

コンフィグレーション対象のターゲットリンクのセットを識別するステップと、

前記ターゲットリンクをN個の分離したサブセット $T_1 - T_N$ に分類するステップと、

各サブセットからリンクを同時にコンフィグレーションし、サブセット T_1 から開始して、連続的に各サブセットを1つずつ処理するステップと

を備えることを特徴とする方法。

【請求項2】

前記トポロジグラフを準備するステップは、開放型最小経路優先 (OSPF) ルーティングプロトコルに基づくトポロジグラフを準備するステップを含み、

前記ターゲットリンクを分類するステップは、

前記OSPFトポロジグラフから、コンフィグレーション対象でない非ターゲットリンクを除去するステップと、

前記OSPFトポロジグラフに残っている前記リンク間の依存関係を判定するステップと、

10

20

前記リンク間の前記依存関係に基づいて、前記ターゲットリンクを前記サブセットに分類するステップと

を含むことを特徴とする請求項 1 に記載の方法。

【請求項 3】

前記リンク間の依存関係を判定するステップは、リンクグラフを構築するステップを含み、

前記リンクグラフを構築するステップは、

前記 O S P F トポロジグラフ内の各ターゲットリンクに対して、前記リンクグラフに新規ノードを配置するステップと、

前記リンクグラフに配置される各ノードに対して、前記 O S P F トポロジグラフから全網（フルメッシュ）の近隣ノードを生成するステップと、

前記リンクグラフに、前記 O S P F トポロジグラフ内の前記管理局を示すノードを追加するステップと、

前記管理局を示す前記ノードを、前記 O S P F トポロジグラフ内の前記管理局を起点とする前記リンクに接続するステップと

を備えることを特徴とする請求項 2 に記載の方法。

【請求項 4】

前記リンク間の前記依存関係に基づいて、前記ターゲットリンクを前記サブセットに分類するステップは、前記リンクグラフからリンクツリーを構築するステップを含み、

前記リンクツリーを構築するステップは、

(a) 前記管理局を示す前記ノードを第 1 開始点として指定するステップと、

(b) 前記リンクグラフに、前記第 1 開始点に隣接するノードと前記第 1 開始点とを接続するリンクのすべてを追加するステップと、

(c) 前記第 1 開始点に隣接するノードを選択するステップと、

当該隣接するノードを選択するステップは、

(c) (1) 他の隣接ノード以外の複数の近隣ノードを有する隣接ノードが存在する場合、前記リンクツリーにまだ追加されていない最大数の近隣ノードを有する隣接ノードを、第 2 開始点として選択するステップと、

(c) (2) 複数の隣接ノードが、前記リンクツリーにまだ追加されていない最大数の近隣ノードを有する場合、前記第 2 開始点として、前記最大数の近隣ノードを有する前記隣接ノード群から隣接ノードを任意に選択するステップとを備え、

(d) 前記リンクツリーに、前記リンクツリーに既に存在するリンク以外の、前記第 2 開始点を起点とするリンクのすべてを追加するステップと、

(e) 第 3 開始点として、前記リンクツリー内のノードを選択するステップと、

当該リンクツリーのノードを選択するステップは、

(e) (1) 他のノード以外の複数の近隣ノードを有する前記リンクツリー内にノードが存在する場合、前記リンクツリーにまだ追加されていない最大数の近隣ノードを有する前記リンクツリー内のノードを選択するステップと、

(e) (2) 複数のノードが、前記リンクツリーにまだ追加されていない前記最大数の近隣ノードを有している場合、前記第 3 開始点として、前記第 1 開始点から最も遠い位置にある前記最大数の近隣ノードを有する前記ノード群からノードを選択するステップと

(e) (3) 複数のノードが、前記リンクツリーにまだ追加されていない前記最大数の近隣ノードを有していて、かつ前記最大数の近隣ノードを有する前記ノードのすべてが前記第 1 開始点から同一距離にある場合、前記第 3 開始点として、前記最大数の近隣ノードを有する前記ノード群からノードを任意に選択するステップとを備え、

(f) 前記リンクツリーに、前記リンクツリー内に既に存在しているリンク以外の、前記第 3 開始点を起点とする前記リンクのすべてを追加するステップと、

(g) 前記リンクグラフの前記ノードのすべてが前記リンクツリーに追加されているかを判定するステップと、

10

20

30

40

50

(h) 前記リンクグラフの前記ノードのすべてが前記リンクツリーに追加されている場合、前記リンクツリー内の前記リンクのすべてを、分離したサブセット T_i に分類するステップと

を備えることを特徴とする請求項 3 に記載の方法。

【請求項 5】

前記リンクツリーに追加されていないターゲットリンクが存在するかを判定するステップと、

前記リンクツリーに追加されていないターゲットリンクが存在する場合、前記リンクツリーに追加されていない前記ターゲットリンクを有するリンクサブグラフを生成するステップと、

前記 (a) ステップから (h) ステップを繰り返して、分離したサブセット T_{i+1} を生成するステップと

を更に備えることを特徴とする請求項 4 に記載の方法。

【請求項 6】

前記各サブセットから前記リンクを同時にコンフィグレーションするステップは、

前記 OSPF トポロジーグラフに、コンフィグレーションされていない前記非ターゲットリンクを有するスケルトンを構築するステップと、

最後にコンフィグレーションされる ノードが前記スケルトンに存在するように、同一レベルの前記ターゲットリンクのすべてに対する前記ノードを同時にコンフィグレーションするステップと

を備えることを特徴とする請求項 5 に記載の方法。

【請求項 7】

ネットワークノードのセット及び前記ネットワークノード間と当該管理局と前記ネットワークノード間の複数の通信リンクを有するインターネットプロトコル (IP) ベースのネットワークのリンク・スコープコンフィグレーションを行う管理局であって、

前記ネットワークのトポロジーグラフを準備する手段と、

コンフィグレーション対象のターゲットリンクのセットを識別する手段と、

前記ターゲットリンクを N 個の分離したサブセット $T_1 - T_N$ に分類する手段と、

各サブセットから前記リンクを同時にコンフィグレーションし、サブセット T_1 から開始して、連続的に各サブセットを 1 つずつ処理する手段と

を備えることを特徴とする管理局。

【請求項 8】

前記トポロジーグラフを準備する手段は、開放型最小経路優先 (OSPF) ルーティングプロトコルに基づくトポロジーグラフを準備する手段を含んでいる

ことを特徴とする請求項 7 に記載の管理局。

【請求項 9】

前記ターゲットリンクを分類する手段は、

前記 OSPF トポロジーグラフから、コンフィグレーション対象でない非ターゲットリンクを除去する手段と、

前記 OSPF トポロジーグラフに残っている前記リンク間の依存関係を判定する手段と、

前記リンク間の前記依存関係に基づいて、前記ターゲットリンクを前記サブセットに分類する手段と

を備えることを特徴とする請求項 8 に記載の管理局。

【請求項 10】

ネットワークノードのセット及び前記ネットワークノード間と管理局と前記ネットワークノード間の複数の通信リンクを有するインターネットプロトコル (IP) ベースのネットワークの前記管理局のプロセッサ上で動作する、媒体に記憶されているコンピュータソフトウェアプログラムであって、

前記ネットワークのトポロジーグラフを準備するステップと、

10

20

30

40

50

コンフィグレーション対象のターゲットリンクのセットを識別するステップと、
 前記ターゲットリンクをN個の分離したサブセット $T_1 - T_N$ に分類するステップと、
 各サブセットから前記リンクを同時にコンフィグレーションし、サブセット T_1 から開始して、連続的に各サブセットを1つずつ処理するステップと
 を前記管理局に実行させることによって、前記IPネットワークのリンクスコープコンフィグレーションを行うように動作可能なコンピュータソフトウェアプログラム。

【請求項11】

当該コンピュータソフトウェアプログラムは、前記管理局に、開放型最小経路優先(OSPF)ルーティングプロトコルに基づくトポロジグラフを準備させ、かつ、当該コンピュータソフトウェアプログラムは、

前記OSPFトポロジグラフから、コンフィグレーション対象でない非ターゲットリンクを除去するステップと、

前記OSPFトポロジグラフに残っている前記リンク間の依存関係を判定するステップと、

前記リンク間の前記依存関係に基づいて、前記ターゲットリンクを前記サブセットに分類するステップと

を前記管理局に実行させることによって、前記ターゲットリンクを分類することを特徴とする請求項10に記載のコンピュータソフトウェアプログラム。

【請求項12】

当該コンピュータソフトウェアプログラムは、

前記OSPFトポロジグラフ内の各ターゲットリンクに対して、前記リンクグラフに新規ノードを配置するステップと、

前記リンクグラフに配置される各ノードに対して、前記OSPFトポロジグラフから、全網(フルメッシュ)の近隣ノードを生成するステップと、

前記リンクグラフに、前記OSPFトポロジグラフ内の前記管理局を示すノードを追加するステップと、

前記管理局を示す前記ノードを、前記OSPFトポロジグラフ内の前記管理局を起点とする前記リンクに接続するステップと

を実行することによって、前記管理局に、リンクグラフを構築させることによって、前記リンク間の依存関係を判定させる

ことを特徴とする請求項11に記載のコンピュータソフトウェアプログラム。

【請求項13】

当該コンピュータソフトウェアプログラムは、

(a) 前記管理局を示す前記ノードを第1開始点として指定するステップと、

(b) 前記リンクグラフに、前記第1開始点に隣接するノードと前記第1開始点とを接続するリンクのすべてを追加するステップと、

(c) 前記第1開始点に隣接するノードを選択するステップと、

当該隣接するノードを選択するステップは、

(c)(1) 他の隣接ノード以外の複数の近隣ノードを有する隣接ノードが存在する場合、前記リンクツリーにまだ追加されていない最大数の近隣ノードを有する隣接ノードを、第2開始点として選択するステップと、

(c)(2) 複数の隣接ノードが、前記リンクツリーにまだ追加されていない最大数の近隣ノードを有する場合、前記第2開始点として、前記最大数の近隣ノードを有する前記隣接ノード群から隣接ノードを選択するステップとを備え、

(d) 前記リンクツリーに、前記リンクツリーに既に存在するリンク以外の、前記第2開始点を起点とするリンクのすべてを追加するステップと、

(e) 第3開始点として、前記リンクツリー内のノードを選択するステップと、

当該リンクツリーのノードを選択するステップは、

(e)(1) 他のノード以外の複数の近隣ノードを有する前記リンクツリー内にノードが存在する場合、前記リンクツリーにまだ追加されていない最大数の近隣ノードを有す

10

20

30

40

50

る前記リンクツリー内のノードを選択するステップと、

(e)(2) 複数のノードが、前記リンクツリーにまだ追加されていない前記最大数の近隣ノードを有している場合、前記第3開始点として、前記第1開始点から最も遠い位置にある前記最大数の近隣ノードを有する前記ノード群からノードを選択するステップと、

(e)(3) 複数のノードが、前記リンクツリーにまだ追加されていない前記最大数の近隣ノードを有していて、かつ前記最大数の近隣ノードを有する前記ノードのすべてが前記第1開始点から同一距離にある場合、前記第3開始点として、前記最大数の近隣ノードを有する前記ノード群からノードを任意に選択するステップとを備え、

(f) 前記リンクツリーに、前記リンクツリー内に既に存在しているリンク以外の、前記第3開始点を起点とする前記リンクのすべてを追加するステップと、

(g) 前記リンクグラフの前記ノードのすべてが前記リンクツリーに追加されているかを判定するステップと、

(h) 前記リンクグラフの前記ノードのすべてが前記リンクツリーに追加されている場合、前記リンクツリー内の前記リンクのすべてを、分離したサブセット T_i に分類するステップと

を執行することによって、前記管理局に、前記リンクグラフからリンクツリーを構築させることによって、前記ターゲットリンクを前記サブセットに分類させる

ことを特徴とする請求項12に記載のコンピュータソフトウェアプログラム。

【請求項14】

当該コンピュータソフトウェアプログラムは、

前記リンクツリーに追加されていないターゲットリンクが存在するかを判定するステップと、

前記リンクツリーに追加されていないターゲットリンクが存在する場合、前記リンクツリーに追加されていない前記ターゲットリンクを有するリンクサブグラフを生成するステップと、

前記(a)ステップから(h)ステップを繰り返して、分離したサブセット T_{i+1} を生成するステップと

を前記管理局に実行させる

ことを特徴とする請求項13に記載のコンピュータソフトウェアプログラム。

【請求項15】

当該コンピュータソフトウェアプログラムは、

前記OSPFトポロジーグラフに、コンフィグレーションされていない前記非ターゲットリンクを有するスケルトンを構築するステップと、

最後にコンフィグレーションされるノードが前記スケルトン内に存在するようにように、同一レベルの前記ターゲットリンクのすべてに対する前記ノードを同時にコンフィグレーションするステップと

を前記管理局に実行させることによって、前記各サブセットから前記リンクを同時にコンフィグレーションさせる

ことを特徴とする請求項14に記載のコンピュータソフトウェアプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明の分野

本発明は、インターネットプロトコル(IP)通信ネットワークに関するものである。より詳しくは、これに制限されるものではなく、本発明は、集中化管理ノードから、IPベースネットワーク内のリンク-スコープ-タイプ管理オブジェクト(link-scope-type-managed object)をコンフィグレーション(構成: configuration)する方法に向けたものである。

【0002】

10

20

30

40

50

背景

インターネットプロトコル（IP）は、ホストの物理的な接続とは独立して、ホストを接続する通信プロトコルである。一般的には、IPホストは、IPプロトコルスタック及びアプリケーションを含むコンピュータである。ホストのセット（集合）は直接接続される。即ち、それらは、同一のケーブル上にあり、それによって、互いに直接通信することができる。この構成は、IPネットワークあるいはサブネットワーク（あるいは、単に、IPサブネット）と呼ばれる。IPホストが直接接続されない場合、つまり、物理的に分けられた複数のリンクが存在する場合、物理的に分けられたIPサブネット上のホスト間でのIP接続性を提供するためには、ルータが必要となる。ルータは、異なるIPサブネットを接続するものである。IPに基づく最大のコンピュータネットワークはインターネットであり、これは、ルータによって接続されている多数のIPサブネットを構成している。ルータが少なくとも2つのサブネットを接続している場合、これらのサブネット上のホストは、互いにルータを介して通信することができる。もちろん、それぞれのサブネット上に直接接続されているホストは、互いに直接通信することができるが、1つのサブネット上のホストが、別のサブネット上のホストと通信したい場合は、通信はルータを経由する。ルータ自身は、ホストによって送信された受信IPパケットを転送するための専用ハードウェア及びソフトウェアを有するコンピュータである。

【0003】

ルータは多くの機能を実装していて、これらの機能は、ルータに様々なプロトコル及びサービスをサポートすることを可能にし、かつ他の機能を実行することを可能にする。ルータの機能は、可変パラメータによって制御される。これらのパラメータの所与の値のセットは、コンフィグレーション（構成）として参照される。単一のルータでのコンフィグレーション管理は、エレメントコンフィグレーション管理として参照され、一方で、ネットワーク内の複数のルータとホストの現在のコンフィグレーションは、ネットワークコンフィグレーションとして参照される。ネットワークコンフィグレーション管理は、ネットワークの動作機能のプランニング（計画）及びセッティングを含んでいる。これらの機能は、ルーティングプロトコル、転送ポリシー、仮想プライベートネットワーク、いくつかのサービス品質（QoS）特徴及びその類を含んでいる。加えて、リンク関連コンフィグレーション、例えば、IPオーバーPPP（IP over PPP）接続が存在する。各ルータは、ネットワークコンフィグレーションの自身の個別の部分を実行し、これには、レイヤ1の物理接続の専用属性、レイヤ2のデータリンクレベルインタフェース、ソフトウェアコンフィグレーション、エレメントセキュリティ及びその類がある。

【0004】

与えられるルータに対して、これらのパラメータの値のセットは、ルータのコンフィグレーションである。同様に、ネットワーク内でのルータコンフィグレーションのこれらのセットの結合も、ネットワークコンフィグレーションである。しかしながら、これらのセットは、分離しない。ルータコンフィグレーションは、ネットワーク内の他のルータに依存するいくつかの変数と、他のルータとは独立しているいくつかの変数を持っている。この可変パラメータは、以下のように分類することができる。

【0005】

・ルータ - スコープパラメータ：これらのパラメータは、ルータ自身のみに関連する。ルータスコープパラメータの変更は、他のルータの動作に直接影響を与えない（例えば、ルータのホスト名、アクセスパスワード、ルーティング処理id及びその類の変更）。

【0006】

・リンクスコープパラメータ：これらのパラメータは、リンクによって接続されている複数のルータに関連する（このリンクは、PPPオーバーシリアル接続のような物理リンクであっても良いし、あるいは、例えば、開放型最小経路優先（Open Shortest Path First：OSPF）隣接性（adjacency）接続のような論理リンクであっても良い）。リンク - スコープパラメータは、構成されるリンクによって接続されるルータ内で均一な値を

10

20

30

40

50

持たなければならない（例えば、OSPFハローインターバル（HelloInterval）は、正しい隣接性を構築するために隣接（adjacent：アジェセント）ルータに対して同一にしなければならない）。

【0007】

・エリア - スコープパラメータ：これらのパラメータは、論理ドメインあるいはエリアに属するルータのグループに対して関連している（例えば、ASに対して同一の自律システム（AS）番号、あるいはOSPFエリアに対して同一のOSPFエリアID）。

【0008】

上述のように、ルータの機能は、基本的には、受信IPパケットをどこに転送するかを判定し、かつそのパケットを自身の宛先に転送することである。転送情報は、静的あるいは動的ルーティングを使用してルータに提供されても良い。静的ルーティングを用いる場合、ネットワーク管理者は、手動でルータを各ルータのルーティングテーブルに設定する。動的ルーティングを用いる場合、ルータは、ルーティングプロトコルを使用して、ネットワーク内の既存のルートを判定する。ルータは、ルーティングプロトコルによって判定される情報を使用して、自身のルーティングテーブルを維持する。

【0009】

ネットワーク内のルータの数が一定数以上である場合（例えば、4あるいは5）、動的な方法が好ましい。これは、ルーティングプロトコルが各ルータで開始され、かつルータが適切に動作するようにコンフィグレーション（構成）されることを意味する。IPネットワーク内で共通して使用されるルーティングプロトコルの1つには、OSPFプロトコルがある。このプロトコルのコンフィグレーションは、可変パラメータ、即ち、ルータ - スコープ、リンク - スコープ及びエリア - スコープと同一の方法で分類することができる。ルータ - スコープは、ルータ内のOSPF処理をコンフィグレーション（構成）する処理を参照する。リンク - スコープは、OSPFリンク（隣接性）を構成する処理を参照する。エリア - スコープは、OSPFエリアを構成する処理を参照する。

【0010】

現在のIPネットワーク管理では、ネットワーク管理者は、ほとんどのコンフィグレーション操作を、以下のエレメント管理方法の1つを使用して手動で実行している。

【0011】

・コマンドラインインタフェース（CLI）：これは、ルータコンフィグレーション管理用に最も広く使用されている方法である。CLIを用いることで、テルネットあるいはコンソール接続を介してルータにアクセスし、任意のコマンドセットを使用して、ネットワーク管理者はコマンドを入力して、ルータから情報を検索し、パラメータを設定する。CLIコマンドセットは、かなり大規模なものになり得る。CLIの最適な例には、CiscoCLIがあり、これは、デファクトスタンダードとなっている。

【0012】

・コンフィグレーションファイル編集：この方法は、特殊用途のCLI方法である。この場合、ネットワーク管理者は、CLIコマンドのシーケンスを含むコンフィグレーションファイルを編集する。次に、このコンフィグレーションファイルは、ファイル転送プロトコル（FTP）あるいは単純ファイル転送プロトコル（TFTP）を使用して、関連するルータにダウンロードされる。この処理は、いくつかのCLI対話を必要としていて、これは、例えば、ルータがFTPあるいはTFTPクライアントのみを持っている場合にはルータからダウンロードを開始する。

【0013】

・メニューベースのエレメントマネージャ：CLIインタフェースは持たないが、テルネットあるいはコンソール接続を介してアクセス可能なメニュー駆動型システムを持っているルータがいくつか存在する。ネットワーク管理者は、これらのインタフェースを使用して、ルータコンフィグレーションを確認あるいは設定することができる。この方法は、かなりまれに使用される。

【0014】

10

20

30

40

50

・単純ネットワーク管理プロトコル (SNMP) : SNMPは、インターネットエンジニアリングタスクフォース (IETF) 標準プロトコルであり、これは、標準的なエレメント監視及びコンフィグレーション方法を提供している。管理情報ベース (MIB) は、管理されるオブジェクト及びそれらの属性を定義する。SNMPは、これらの属性の値を取得し、設定するプロトコルである。いくつかのアプリケーションは、SNMPを使用している。与えられるMIBを単にブラウズし、かつ1つのターゲットルータ上の自身の属性を取得するあるいは設定する、いわゆるMIBブラウザが存在する。1つ以上のルータを取り扱うことができるアプリケーションが存在する。しかしながら、実際には、SNMPは、典型的には、コンフィグレーション管理ではなく、統計値の監視、収集、管理及び故障管理用に使用される。この理由の1つには、MIBには、統計値を提供する読出専用属性が多く含まれていることがある。別の理由には、標準的なMIBは、あらゆる事象をカバーするのではないので、多くのタイプのルータは、標準的なものではない独自のMIBによって管理されることが好ましい。

【0015】

・HTTPベースのエレメント管理 : いくつかのタイプのルータは、ウェブブラウザサービスを持っている。ハイパーテキスト転送プロトコル (HTTP) ブラウザは、コンフィグレーション及び他の情報にアクセスすることができる。ユーザは、普及しているハイパーテキストマークアップ言語 (HTML) ページ上のパラメータを検索する、あるいは設定することができる。

【0016】

これらのエレメント管理方法の中では、SNMPが、アプリケーション用に使用するのに最も適している。CLIは、手動コンフィグレーション管理用に設計されているが、アプリケーションで使用するように変更することができる。コンフィグレーションファイルの編集は、アプリケーションによって支援することができる。メニューベース及びウェブサーバベースの方法は、アプリケーションで使用するようには設計されていない。これらの方法は、手動コンフィグレーション管理用に限って有効である。

【0017】

加えて、ネットワーク管理のあるレベルを提供するアプリケーションが存在する。これらのプログラムは、2つの基本セットに分けることができる。1つは、ルータベンダによって提供されるアプリケーション (例えば、シスコ社のCiscoWorks) であり、もう1つは、他のソフトウェア開発企業によって提供されるアプリケーションである (例えば、HPオープンビュー)。

【0018】

リモート管理については、テルネットプロトコルが、ルータをアクセスするために広く使用されている。テルネットは、ターゲットルータへ直接実行することができる。あるいは、ターゲットの近隣 (neighbor: ネイバー) ルータへテルネットを行い、かつその近隣からターゲットルータにテルネットを行うことによって間接的に、ターゲットルータへテルネットを実行することができる。しかしながら、この種の管理は、コンフィグレーション属性のスコープ (範囲) を完全には気付いていない。スコープ気付コンフィグレーションの周知の記事のみが存在していて、この記事は、文献の中で、ピー・クリシュナン等による、「IPネットワーク用コンフィグレーション操作の順序付け」という見出しがついている。これは、LISA 2000の第14システム管理会議の議事録である (以下では、クリシュナンと呼ぶ)。しかしながら、以下で説明されるように、クリシュナンによる方法は、十分なソリューションとは言えない。

【0019】

従来技術のIPコンフィグレーション管理方法の最も重要な特性は、各ターゲットルータが、互いに独立して1つずつ構成されることである。ネットワーク管理者は、動作を「考慮して」設計し、関連するルータを1つずつ構成することによってその動作を実現する。最初のステップは、管理者が変更対象のパラメータと、設定対象の値を定義することである。これらの変更は、次に、関連するルータについてなされる。最初の部分は論理的な

10

20

30

40

50

ものであり、次の部分は具体的なものである。つまり、最初のステップは、管理者の頭の中あるいは用紙上で実行される。次に、管理者は、関連するルータについて必要なエレメント管理操作を実行する。この方法によるリンク - スコープ O S P F パラメータは、コンフィグレーション費用の問題、順序付けの問題、長期の操作時間及び、キャンセル及びエラー処理に伴う問題を招くことになる。これらの各問題の部分を、以下に説明する。

【 0 0 2 0 】

コンフィグレーション費用の問題：O S P F リンクは、リンクスコープ属性を持っている。これらの属性はルータに記憶され、また、適切な O S P F 隣接性について均一な値を持たなければならない。O S P F の論理的なコンフィグレーションは、これらのリンク - スコープ属性の値を定義する必要がある。しかしながら、物理的なコンフィグレーションは、構成された O S P F リンクによって接続されているそれぞれのルータにこれらの値を設定する必要がある。ポイントツーポイントリンクの場合、これは、2つのルータであることを意味する。しかしながら、ブロードキャストあるいは非ブロードキャストマルチアクセス (N B M A) のような他の場合、2つ以上のルータが存在し得る。また、複数の O S P F リンクがターゲットである場合、いくつかのターゲットルータが増やされる。2つの重要なリンク - スコープ O S P F パラメータである、ハロー (Hello -) とデッドインターバル (DeadInterval) を考慮すると、管理者は、各ターゲットリンクに対して新規の値を定義しなければならない。変更のための論理的な2つのパラメータが存在するが、管理者は、いくつかのルータにアクセスして、これらの2つの値を各ルータに設定しなければならない。理論的に必要とされるコンフィグレーション操作 (本例では、2つのパラメータを設定) と、実際のコンフィグレーション操作 (同一の値で、いくつかのルータに2つのパラメータを設定) との間の違いは、かなり煩わしくなる。加えて、ネットワーク管理者は、いくつかのルータに数回同一のことを実行しなければならない。これは、ネットワークコンフィグレーションにおける、ヒューマンエラーの可能性が増加する。ネットワーク管理者の作業負担を軽減し、かつネットワークコンフィグレーションで発生するヒューマンエラーの可能性を軽減するコンフィグレーション方法を実現することは有効である。

【 0 0 2 1 】

順序付け問題：別の問題及びより重大な問題には、順序付け (sequencing) 問題がある。大規模の I P ネットワークの管理は、分散されるのではなく、集中管理されることが多い。一般的には、いくつかのネットワーク操作センタだけがネットワークについて関与しているので、コンフィグレーション変更 (エレメントコンフィグレーション管理) は、これらのセンタから実行される。その結果、操作中に、各ターゲットルータに I P 接続性を維持することはかなり重要である。小規模ネットワークでは、ターゲットルータの数は少ないので、これについては、大きな問題とならない。しかしながら、ルータの数が数百の桁である場合、エレメントコンフィグレーションのシーケンス (順序) は重要である。この問題を理解するためには、どのようにして O S P F プロトコルがリンク (即ち、隣接性) を扱うかを思い起こす必要がある。

【 0 0 2 2 】

近隣にある O S P F ルータ群は隣接性を構築する。通信、既知の経路の広告、及び O S P F データベースを同期するためにチャネルが使用される。適切な通信がなされないと、いくつかの接続は、O S P F によって使用することができず、ある経路は、ルーティング計算について利用することができない。それゆえ、これらのルータは、トラフィック用に利用することができない。つまり、いくつかのルータ、ホストあるいはサブネットワークは、ネットワークの任意の位置からアクセス不可能となり得る。O S P F 隣接性は、適切な O S P F ルーティングにとって重要である。O S P F 隣接性の確立は、リンク - スコープ属性に基づいている。一般的なルールとして、これらのパラメータが均一の値を持たなければならないということがある。ポイントツーポイントリンクでは、近隣ルータ同士は、O S P F リンクに対して、均一の O S P F リンク - スコープ属性を持たなければならない。ブロードキャスト、N B M A あるいはポイントツーマルチポイント接続では、隣接性

10

20

30

40

50

は、近隣で広告する均一なリンク - スコープ値の間で確立される。関係するルータが他のルータとは異なる値を広告する場合、その他のルータは、その関係するルータとの隣接性を確立しないし、また、その関係するルータは、その他のルータとの隣接性を確立しない。

【 0 0 2 3 】

OSPFリンク - スコープパラメータがどのようにしてOSPFリンク上で変更されるかについても理解されるべきである。ネットワーク管理者が動作中のOSPFリンク上のリンク - スコープ属性を変更し、かつそのリンクのエンドポイントにアクセスしたい場合、その管理者は、リンク - スコープ属性が変更される場合には、他の機器（群）が均一な値を持つまでOSPFリンクが損失され得ることを考慮しなければならない。重要な要素はリンクコンフィグレーション時間であり、これは、リンク上の最初のルータへのアクセスから、リンク上の最後のルータで最後のパラメータを設定する間の時間である。リンク損失の可能性は、発信元のハロー（Hello-）とデッドインターバル（DeadInterval）、及びこのコンフィグレーション時間に依存する。近隣ルータ間の転送レートは取るに足りない。いくつかの環境では、リンクは、操作中に原型を保っているが、他の環境では、新規のものが新規のリンク - スコープ値で確立されるまで一時的に損失する可能性がある。この動作は、ネットワークのツリー部分においてリンクが構成される場合に重要である。

【 0 0 2 4 】

図1は、従来技術で存在する順序付け問題を示すネットワーク図である。図示の場合では、最初のアクセスが管理局11から最も近いルータR-1 12に対して行われる。ここで、リンク - スコープ変更が実行される。しかしながら、終端にアクセスする前にリンクが損失する場合、管理局は、最も遠いルータR-3 14あるいはおそらくは、中間のルータR-2 13にも到達することができない。それゆえ、管理局は、新規のリンク - スコープ値を最も遠いルータR-3に設定することができず、新規のリンクを確立することができない。つまり、ランダムなコンフィグレーション順序は、永続的なルータの損失を容易に招き得る。ネットワークを構成する場合には、順序付け問題を解決し、かつルータ損失を回避するコンフィグレーション方法を実現することは有効である。

【 0 0 2 5 】

長い操作時間：多くのリンクがリンク - スコープ値の変更のターゲットであり、かつそれに伴い、多くのルータがコンフィグレーション対象である大規模環境を考慮すると、操作時間は重要である。このコンフィグレーション操作中には、他のコンフィグレーション操作を開始することは勧められない。これは、その操作がルーティングに影響を与えるからである。コンフィグレーション操作中に、いくつかのターゲットリンクが一時的に損失する場合に、一時的なルーティング変更が発生する可能性がある。伝統的な方法（1つずつ順番に）でのターゲットルータの構成は、長い操作時間を招き得る。ネットワークコンフィグレーションに関連する操作時間を削減するコンフィグレーション方法を実現することは有効である。

【 0 0 2 6 】

キャンセル及びエラー処理：ネットワークオペレータが自身の考えを変更する場合、あるいは、自身が誤ったコンフィグレーション操作を開始したことに気付いた場合、オペレータはそれをキャンセルしたい可能性がある。最も安全なソリューションは、もちろん、キャンセルしないで、その操作を終了させることである。しかしながら、この場合、オペレータは、従前の状態に戻した上で、大量の追加のコンフィグレーションを実行しなければならない可能性がある。つまり、コンフィグレーション操作をキャンセルする機能は有用なことであるが、その実現は容易なことではない。キャンセルに伴う問題は、コンフィグレーション操作がキャンセルできない場合に発生する。リンクの終端が構成された後に操作がキャンセルされる場合、他の終端（群）が構成される前では、リンクは損失することになる。これは、多くのリンクが同時に構成される場合により生じ易い。つまり、キャンセルが開始されると、コンフィグレーションが開始しているリンクは終了されなければならないが、追加のリンクのコンフィグレーションは開始すべきでない。伝統的な

10

20

30

40

50

方法での適切なキャンセルは主要な問題ではないが、ソフトウェアベースのソリューションにおいて、特に同時に実行する場合には、このキャンセルは、慎重に検討すべきである。

【 0 0 2 7 】

別の重要な検討事項は、いくつかのエレメント管理操作は、いくつかの理由により失敗する可能性があることである。この種のエラーが発生する場合、その操作は、キャンセルと同一の方法で停止すべきである。しかしながら、単なる停止は十分でない。エラーが生じる状況は集中管理局から取り扱われることも重要である。しかしながら、従来のソリューションでは、このことは常に可能ではない。ランダムコンフィグレーションを用いた場合は、集中管理局がエラーを解決するために何かを実行することができず、また、技術者は、故障のあるルータを手動でエラーを訂正しなければならないという状況が容易に生じ得る。

10

【 0 0 2 8 】

クリシュナンの記事（上述を参照）で提案されるソリューションは、現在のルーティングに基づいてリンク及びルータシーケンスを計算する。管理局とターゲットルータ間で前方及び後方経路を持たせると、これは、シーケンスを定義するツリーを構築する。このツリーは、リーフからルートへ渡っている。クリシュナンのソリューションの重要な特徴と制限は、（１）対称性のあるルーティングのみが考慮されること、（２）ルーティング情報がルータ自身から取得されること、（３）キャンセルが考慮されていないこと、（４）エラー処理が考慮されていないことである。

20

【 0 0 2 9 】

上述の問題を解決するIPコンフィグレーション方法を実現することは有効である。本発明は、このような方法を提供する。

【 0 0 3 0 】

本発明の要約

本発明は、IPネットワークの集中化管理ノードから、開放型最小経路優先（OSPF）リンクのリンクスコープコンフィグレーションを行う方法である。この方法は、（１）順序付け問題に対するソリューションと、（２）同時処理の動作時間の高速化、（３）適切なキャンセル、（４）好適な誤り処理、（５）従来技術のソリューションよりも単純なシーケンス計算を提供する。

30

【 0 0 3 1 】

つまり、１つの構成では、本発明は、少なくとも１つの管理局と、ネットワークノードのセット及び前記ネットワークノード間と前記管理局と前記ネットワークノード間の複数の通信リンクを有するインターネットプロトコル（IP）ベースのネットワークのリンク-スコープコンフィグレーションを行う方法に向けられたものである。この方法は、前記ネットワークのOSPFトポロジグラフを準備するステップと、コンフィグレーション対象のターゲットリンクのセットを識別するステップと、前記ターゲットリンクをN個の分離したサブセット $T_1 - T_N$ に分類するステップと、各サブセットから前記リンクを同時にコンフィグレーションし、サブセット T_1 から開始して、連続的に各サブセットを１つずつ処理するステップとを備える。このターゲットリンクは、前記OSPFトポロジグラフから、コンフィグレーション対象でない非ターゲットリンクを除去し、前記OSPFトポロジグラフに残っているリンク間の依存関係を判定し、及び前記リンク間の前記依存関係に基づいて、前記ターゲットリンクを前記サブセットに分類することによって、分類されても良い。

40

【 0 0 3 2 】

リンク間の依存関係は、リンクグラフを構築することによって判定されても良い。このリンクグラフは、前記OSPFトポロジグラフ内の各ターゲットリンクに対して、前記リンクグラフに新規ノードを配置することによって構築されても良い。リンクグラフに配置される各ノードに対しては、全網（full mesh：フルメッシュ）の近隣ノードがOSPFトポロジグラフから生成される。これは、前記リンクグラフに、前記OSPFトポロ

50

ジグラフ内の前記管理局を示すノードを追加し、かつ前記管理局を示す前記ノードを、前記OSPFトポロジグラフ内の前記管理局を起点とする前記リンクに接続することによってなされる。

【0033】

ターゲットリンクは、リンクグラフからリンクツリーを構築することによって、リンク間の依存関係に基づいてサブセットに分類されても良い。このリンクツリーは、前記管理局を示す前記ノードを第1開始点として指定することによって構築されても良い。次に、前記第1開始点に隣接するノードと前記第1開始点を接続するリンクのすべてが、前記リンクグラフに追加される。次に、前記第1開始点に隣接するノードが第2開始点として選択される。この選択は、前記リンクツリーにまだ追加されていない最大数の近隣ノードを有する隣接(adjacent)ノードを選択することによってなされても良い(他の隣接ノード以外の複数の近隣ノードを有する隣接ノードが存在する場合)。複数の隣接ノードが、前記リンクツリーにまだ追加されていない最大数の近隣ノードを有する場合、前記第2開始点は、前記最大数の近隣ノードを有する前記隣接ノード群から任意に選択される。

10

【0034】

前記リンクツリーは、前記リンクツリーに既に存在するリンク以外の、前記第2開始点と起点とするリンクのすべてを追加し、かつ第3開始点として、前記リンクツリー内のノードを選択することによって構築され続けられる。この第3開始点は、前記リンクツリーにまだ追加されていない最大数の近隣ノードを有する前記リンクツリー内のノードを選択することによって選択されても良い(他のノード以外の複数の近隣ノードを有する前記リンクツリー内にノードが存在する場合)。複数のノードが、前記リンクツリーにまだ追加されていない前記最大数の近隣ノードを有している場合、前記第1開始点から最も遠い位置にある前記最大数の近隣ノードを有する前記ノードが前記第3開始点として選択される。複数のノードが、前記リンクツリーにまだ追加されていない前記最大数の近隣ノードを有していて、かつ前記最大数の近隣ノードを有する前記ノードのすべてが前記第1開始点から同一距離にある場合、前記第3開始点は、前記最大数の近隣ノードを有する前記ノード群から任意に選択される。

20

【0035】

リンクツリーは、前記リンクツリー内に既に存在しているリンク以外の、前記第3開始点を起点とする前記リンクのすべてを追加することによって構築され続けられる。前記リンクグラフの前記ノードのすべてが前記リンクツリーに追加されている場合、前記リンクツリー内の前記リンクのすべては、分離したサブセット T_i に分類される。次に、リンクツリーに追加されていないターゲットリンクが存在するかが判定される。そうである場合、前記リンクツリーに追加されていない前記ターゲットリンクを有するリンクサブグラフが生成され、上述のステップが繰り返されて、分離したサブセット T_{i+1} を生成する。すべてのサブセットが生成される場合、前記OSPFトポロジグラフに、コンフィグレーションされていない前記非ターゲットリンクを有するスケルトンを構築し、前記スケルトンに最後にコンフィグレーションされるように、同一レベルの前記ターゲットリンクのすべてに対する前記ノードを同時にコンフィグレーションすることによって、前記各サブセットからのリンクが同時にコンフィグレーションされても良い。

30

40

【0036】

実施形態の詳細説明

本発明は、集中管理ノードから、IPベースネットワーク内のリンクスコープタイプ管理オブジェクトを構成する方法を改善したものを提供する。本実施形態は、開放型最小経路優先(Open Shortest Path First: OSPF)で説明される。これは、OSPFは、リンク-スコープパラメータを構成する問題を説明するための、かなり明快なリンク-スコープ、ルータ-スコープ及びエリア-スコープで管理されるオブジェクトを持っているからである。コンフィグレーションのためにルータへアクセスすることは、直接接続あるいはリモート(遠隔操作)によって実行することができる。直接接続については、ネットワーク管理者の端末、コンソールあるいはワークステーションがルータへの直接接続を行う

50

。この接続は、例えば、シリアルコンソール接続を使用することによって、管理されているIPインフラストラクチャからは独立してなされる。リモートコンフィグレーションについては、ネットワーク管理者は、例えば、ルータにログインするテルネットを使用して、管理されているIPネットワークを介して、ルータに接続されている機器からルータへアクセスする。本発明の実施形態は、リモートコンフィグレーションを実行する方法を提供する。

【0037】

今日の複雑で、大規模のIPネットワークでは、ネットワークコンフィグレーションは、通常、ネットワーク機能のコンフィグレーションを含んでいる。このネットワーク機能は、論理的なエンティティ、例えば、サービス、経路、OSPFのようなプロトコルあるいはこれらのサブセットとして見なすことができるものである。つまり、大量のコンフィグレーション操作が、多くのルータをコンフィグレーション（構成）する場合に実行される。そのため、ネットワークオペレータがネットワークコンフィグレーション時に変更を行いたい場合は、そのオペレータは、いくつかのエLEMENTコンフィグレーション操作を実行しなければならない。1つ以上のルータに関係するコンフィグレーション操作は、マルチターゲット操作と呼ばれる。この例には、ネットワークオペレータが、ネットワーク内のいくつかのOSPFリンクにおけるOSPFハローインターバル設定を変更したい場合がある。本発明の実施形態は、マルチターゲット操作を実行する方法を提供する。

10

【0038】

本発明は、ソフトウェアで実現されても良い。この管理ソフトウェアは、ネットワーク管理者用のOSPFリンク-スコープ操作を提供する。その結果、管理者は、ターゲットリンクと、リンク-スコープパラメータの新規の値を定義できれば良く、残りはそのソフトウェアが実行する。それゆえ、本実施形態において、本発明は、リモートで、マルチターゲットリンク-スコープOSPFコンフィグレーションを行うための実装可能ソフトウェアソリューションを提供する。

20

【0039】

本発明は、任意のトポロジーと任意のルーティング（対称及び非対称）で動作する。本発明は、リンク間でトポロジー的な依存関係がない場合でさえも、大規模な複雑なネットワークにおいて、同時にコンフィグレーションを行うことができるターゲットリンクの最大数を検出することによって、コンフィグレーション操作を加速する。

30

【0040】

図2は、本発明の実施形態の全体アルゴリズムのステップ群を示すフローチャートである。本発明は、選択されたターゲットリンクを同時にコンフィグレーションすることによって、できる限り少ないステップで、それらのリンクをコンフィグレーションするアルゴリズムを実現する。このアルゴリズムは、グリーディグラフ（greedy graph）アルゴリズムであり、ステップ21で、入力として、G（ネットワークのOSPFトポロジーグラフ）とT（ターゲットリンクのセット）を取得する。ステップ22で、このアルゴリズムは、ターゲットリンクをN個の分離したサブセット $T_1 - T_N$ に分類する。分類後は、このアルゴリズムは、ステップ23で、このサブセットを T_1 から開始して1つずつ取得し、これらのELEMENTを同時にコンフィグレーションする。

40

【0041】

図3は、本発明の方法の実施形態における、分類アルゴリズム22のステップ群を示すフローチャートである。分類アルゴリズムは再帰アルゴリズムであり、アルゴリズム全体の重要な部分である。ステップ31で、分類アルゴリズムは、トポロジーグラフG内の非ターゲットリンクに接続されているルータノードを削除することによって、そのトポロジーグラフGから非ターゲットリンクを削除する。ステップ32で、いわゆるリンクグラフ（Link Graph）が、リンク間の依存関係を発見するために構築される。新規のノードが、各ターゲットリンクに対するグラフに配置され、また、オリジナルグラフ内のルータノードの全網（full mesh）の近隣ノードが各ルータに対して生成される（以下の図6 - 図10参照）。そして、ステップ33で、アルゴリズムは、リンクグラフ（Link

50

Graph) を使用して、図 4 A - 図 4 B に示されかつ記載されるリンクツリー (Link Tree) を構築する。

【 0 0 4 2 】

図 4 A - 図 4 B は、リンクグラフからリンクツリーを構築するためのアルゴリズム 3 3 のステップ群を示すフローチャートの部分である。ステップ 4 1 で、カウンタ I が 1 に設定される。ステップ 4 2 で、管理局 M を示すノードが開始点して識別され、リンクツリー内に配置される。ステップ 4 3 で、M を起点とするエッジのすべてがリンクツリーに配置される。ステップ 4 4 で、リンクツリー内に既に存在していない 1 より多いリーフ (ノード) が、最大数の近隣ノードを持っているかが判定される。そうでない場合、つまり、単一のリーフが最大数の近隣ノードを持っている場合、ステップ 4 5 で、最大数の近隣ノードを持っているリーフが変数 S として選択される。1 より多いリーフが、同一の最大数の近隣ノードを持っている場合、本方法は、ステップ 4 4 からステップ 4 6 へ進み、ここで、最大数の近隣ノードを持っている任意のリーフがリンクグラフ内で 2 親等 (degree) より大きいかが判定される。そうである場合、本方法は、ステップ 4 7 へ進み、そこで、M から最も遠い位置にある 2 親等より大きい単一のリーフが存在するか、あるいは M から最も遠い位置にある 2 親等より大きい複数のリーフが存在するかが判定される。M から最も遠い位置にある 2 親等より大きい単一のリーフが存在する場合、本方法は、ステップ 4 8 へ進み、そこで、M から最も遠い位置にある 2 親等より大きい単一のリーフは、変数 S として選択される。一方、M から最も遠い位置にある 2 親等より大きいリーフが複数存在する場合、本方法は、ステップ 4 9 へ進み、そこで、複数のリーフから条件に合う任意のリーフが変数 S として選択される。

10

20

【 0 0 4 3 】

ステップ 4 6 に戻り、2 親等より大きい最大数の近隣ノードを持っている任意のリーフがリンクグラフ内に存在しないと判定されると (即ち、2 親等のリーフがいくつか存在する)、本方法は、ステップ 5 1 へ進み、そこで、M に最も近い位置にある 2 親等の単一のリーフが存在するか、あるいは M に最も近い位置にある 2 親等の複数のリーフが存在するかが判定される。M に最も近い位置にある 2 親等の単一のリーフが存在する場合、本方法は、ステップ 5 2 へ進み、そこで、M に最も近い位置にある 2 親等の単一のリーフが変数 S として選択される。一方、M に最も近い位置にある 2 親等の複数のリーフが存在する場合、本方法は、ステップ 5 3 に進み、そこで、複数のリーフから条件に合う任意のリーフが変数 S として選択される。つまり、選択されたリーフを持つことで、ステップ 4 5、4 8、4 9、5 2 あるいは 5 3 のいずれかで、本方法は、ステップ 5 4 へ進み、そこで、S が選択されたリーフとして設定される。次に、本方法は、図 4 B に進む。

30

【 0 0 4 4 】

ステップ 5 5 で、本方法は、リンクツリーに戻らない、S を起点とするエッジのすべてをリンクツリーに配置する。ステップ 5 6 で、リンクグラフ内のすべてのノードがリンクツリーに配置されているかが判定される。そうである場合、本方法はステップ 5 7 へ進み、そこで、リンクツリーのすべてのリーフがサブセット T_i に配置される。一方、リンクグラフ内のすべてのノードがリンクツリーに配置されない場合、本方法は、ステップ 5 8 へ進み、リンクツリーに配置されているノードをオリジナルのリンクグラフから取り去ることによって、新規のリンクグラフを構築する。ステップ 5 9 で、ステップカウンタ (i) が 1 つインクリメントされ、本方法は、ステップ 4 2 へ戻り、処理を繰り返す。

40

【 0 0 4 5 】

オリジナルの O S P F トポロジーグラフが、M である 1 つのノードだけを構成している場合、サブセット T_i が定義され、かつコンフィギュレーションアルゴリズムの処理対象として用意される。

【 0 0 4 6 】

図 5 は、本発明で使用するのに適している O S P F トポロジーグラフの例である。図示の例では、リンクは 1 ~ 1 0 の番号が付けられている。ネットワーク内の各リンクはターゲットリンクであり、そのため、グラフは、非ターゲットリンクの終端を削除することに

50

よって縮小することはできない。図 4 A - 図 4 B に示される処理に従えば、最初のステップはリンクグラフを構築してリンク間の依存関係を発見する。得られるリンクグラフが図 6 に示される。このリンクグラフは、上述のルールに従うリンクツリーを構築するために利用される。

【 0 0 4 7 】

図 7 は図 4 A - 図 4 B の処理に従って、図 6 のリンクグラフからリンクツリーを構築する処理の例を示している。まず、ノード M が選択され、エッジ (M , 1)、(M , 2) 及び (M , 4) がリンクツリーに追加される。図 7 では、これらのリンクには「 a 」のラベルが付加され、これらがリンクツリーに最初に追加されていることが示される。次のステップは、実際のリンクツリーのリーフが非リンクツリー近隣ノードをいくつか持っているかを調査する。この場合、ノード 1 は、2 つの近隣ノード (ノード 3 及びノード 5) を持っていて、ノード 2 は、2 つの近隣ノード (ノード 6 及びノード 8) を持っていて、ノード 4 は 2 つの近隣ノード (ノード 3 及びノード 7) を持っている。このようにして、ツリーノード (ノード 1、2 及び 4) 群の内のどれが S に設定されるべきかについての決定がなされなければならない。すべてのツリーノードは、リンクツリー内では 1 段階深く (即ち、2 親等)、かつ M に最も近い単一のノードはないので、図 4 A のステップ 5 3 に従ってノードが任意に選択される。

【 0 0 4 8 】

図示される例では、ノード 4 が選択されて S に設定される。つまり、リンクツリーに追加される次のエッジは、(4 , 3) 及び (4 , 7) である。図 7 では、これらのリンクには「 b 」のラベルが付加され、これらがリンクツリーに 2 番目に追加されていることが示される。次のステップでは、実際のリンクツリーのリーフ群が再度検証され、どのノードが最大数の非リンクツリー近隣ノードを持っているかを判定する。この時点では、ノード 1 とノード 3 の両方は、1 つの非リンクツリー近隣ノードを持っていると判定され、一方、ノード 7 及びノード 2 の両方は 2 つの非リンクツリー近隣ノードを持っていると判定される。但し、この場合、ノード 7 は、リンクツリーでは 2 段階深く (即ち、2 親等以上)、ノード 2 は 1 段階だけ深い (即ち、2 親等)。それゆえ、図 4 A のステップ 4 8 に従えば、ノード 7 が選択されて S に設定され、エッジ (7 , 8) と (7 , 10) がリンクツリーに追加される。図 7 では、これらのリンクには「 c 」のラベルが付加され、これらが 3 番目にリンクツリーに追加されていることが示される。

【 0 0 4 9 】

同様の処理の後に、実際のリンクツリーのリーフが再度検証され、どのノードが最大数の非リンクツリー近隣ノードを持っているかを判定する。この時点では、ノード 8 とノード 10 の両方が 1 つの非リンクツリー近隣ノードを持っていると判定される。両方のノードは、リンクツリーでは 3 段階深い (即ち、2 親等以上) のので、図 4 A のステップ 4 9 に従ってノードが任意に選択される。図示される例では、ノード 8 が選択されて S に設定され、かつエッジ (8 , 6) がリンクツリーに追加される。図 7 では、エッジ (8 , 6) には「 d 」のラベルが付加され、これがリンクツリーに 4 番目に追加されていることが示される。

【 0 0 5 0 】

次のステップでは、実際のリンクツリーのリーフ群が再度検証され、どのノードが最大数の非リンクツリー近隣ノードを持っているかを判定する。この時点では、ノード 6 が最大数の非リンクツリー近隣ノードを持っていると判定され、また、図 4 A のステップ 4 5 に従って S として選択される。次に、エッジ (6 , 9) と (6 , 5) がリンクツリーに追加され、図 7 では、これらのリンクには「 e 」のラベルが付加され、これらが 5 番目にリンクツリーに追加されていることが示される。これで、図 7 に示されるようなリンクツリーの構築が完了する。

【 0 0 5 1 】

リンクツリーの構築後、同時に構成可能なリンクのセットがリンクツリー内のリーフ群を検出することによって判定される。つまり、 T_1 (即ち、第 1 ステップで構成されるリ

10

20

30

40

50

リンクのセット)は{1, 2, 3, 5, 9, 10}である。これらのリンクは、図8で示される。これらのリンクの近隣ノードは削除される。その結果のグラフが図9に示され、また、それ自身のリンクグラフが図10に示される。次のステップで、リンクグラフが更新され、その結果として得られるリンクツリーが図11で示される。つまり、セット T_2 は{4, 6, 7, 8}である。

【0052】

エレメント管理動作は、リンクグラフによって管理されるシーケンスで実行される。同一レベルのリンク群が同時に構成される。実際のターゲットリンクで接続されるルータ間のシーケンスは、以下の方法で、グラフGから導出される。まず、スケルトン、即ち、非ターゲットリンクとコンフィグレーション(構成)対象でないリンクを構築するサブグラフが、オリジナルのOSPFトポロジーグラフで構築される。その結果、接続されたグラフが得られる。リンクのコンフィグレーション(構成)中では、ルータ変更のシーケンスの唯一の制約は、最後にコンフィグレーション(構成)されるルータがスケルトン内に存在しなければならないことである。他のルータのシーケンスは任意であり、:これらのコンフィグレーションは、同時に実行することができる。最後のルータは、リンクに接続されている他のルータのすべてが正常に構成された後のみ変更することができる。各ターゲットリンクが同一のエリアに属している場合にこのアルゴリズムが常に正確であることが、テストを行うことによって示される。

【0053】

以下の実行ルールを適用することによってキャンセルが行われる。このアルゴリズムの実際の段階では、いくつかのターゲットリンクが同時にコンフィグレーション(構成)される。上述のように、ルータがアクセスされると、実際のリンク内の各リンクが正常に構成されないまで、アルゴリズムは、次の段階(T_{i+1})に進まない。キャンセルが開始される場合、既にルータをコンフィグレーション(構成)している実際のリンクは完全にコンフィグレーション(構成)されていなければならない。

【0054】

図12は、本発明の技術に従うIPネットワーク62を構成するリンク-スコープコンフィグレーション(構成)用の管理局61のブロック図である。IPネットワークはネットワークノードのセットと、ネットワークノード間と、管理局とネットワークノード間の複数の通信リンクを含んでいる。トポロジーグラフビルダー63はIPネットワーク用のコンフィグレーション情報を、例えば、コンフィグレーションデータベース64から取得する。このデータベースは、管理局の内部あるいは外部に存在し得る。トポロジーグラフはターゲットリンク識別器65に送信され、これは、コンフィグレーション(構成)対象のトポロジーグラフからターゲットリンクのセットを識別する。識別されたターゲットリンクのセットはターゲットリンク分類器66に送信され、これは、ターゲットリンクをN個の分離されたサブセット $T_1 - T_N$ に分類する。ターゲットリンク分類器内において、非ターゲットリンクリムーバ(remover)67は、ターゲットリンクとして識別されないリンクをトポロジーグラフから除去する。リンクが除去されたトポロジーグラフは、次に、リンクグラフビルダー68へ渡され、ここで、リンク間の依存関係を判定するためのリンクグラフを構築する。次に、この依存関係はリンクツリービルダー69に送信され、ここで、リンク間の依存関係に基づいてターゲットリンクをN個の分離されたサブセット($T_1 - T_N$)に分類するためにリンクツリーを構築する。追加のターゲットリンク識別器70は、リンクツリー内に配置されていない追加のターゲットリンクが存在するかどうかを判定し、そうである場合、リンクツリービルダーは、残りのターゲットリンクをサブセットに分類するために別のリンクツリーを構築する。

【0055】

ターゲットリンクのすべてがサブセットに分類されると、リンクツリービルダー69は、リンクサブセットパラレルコンフィグレーションユニット71に対するサブセット $T_1 - T_N$ を識別し、ここで、各サブセットから同時にIPネットワークリンク群をコンフィグレーション(構成)し、サブセット T_1 で開始し、サブセット T_N に渡って1つずつ各サ

10

20

30

40

50

ブセットを連続的に処理する。コンフィギュレーションが完了すると、そのコンフィギュレーションユニットは、コンフィギュレーションデータベース64を更新する。

【0056】

従来技術によって認識されるように、本願で説明される革新的な概念は、広い範囲の用途に渡って修正し、かつ変更することができる。従って、発明特定事項の範囲は上述の特有な例示の技術に制限されるべきものではないどころか、請求項によって定義されるものである。

【図面の簡単な説明】

【0057】

【図1】従来技術で存在する順序付け問題を示すネットワーク図である。

10

【図2】本発明の方法の実施形態におけるアルゴリズム全体のステップ群を示すフローチャートである。

【図3】本発明の方法の実施形態における分類アルゴリズムのステップ群を示すフローチャートである。

【図4A】リンクグラフからリンクツリーを構築するためのアルゴリズムのステップ群を示すフローチャートの一部を示す図である。

【図4B】リンクグラフからリンクツリーを構築するためのアルゴリズムのステップ群を示すフローチャートの一部を示す図である。

【図5】本発明で使用するのに適切なOSPFトポロジーグラフの例を示す図である。

【図6】OSPFトポロジーグラフから導出され、かつリンク間の依存関係を発見するために利用されるリンクグラフの例を示す図である。

20

【図7】図6のリンクグラフの例からリンクツリーを構築するプロセスを示す図である。

【図8】第1サブセット T_1 に分類されるリンク群を示す図7のリンクツリーを示す図である。

【図9】サブセット T_1 内のリンクの近隣ノードが除去される場合に得られるグラフを示す図である。

【図10】図9のグラフのリンクグラフを示す図である。

【図11】図10のリンクグラフから構築されるリンクツリーを示す図である。

【図12】本発明の技術に従うIPネットワークのリンク・スコープコンフィギュレーションを行う管理局のブロック図である。

30

【図1】

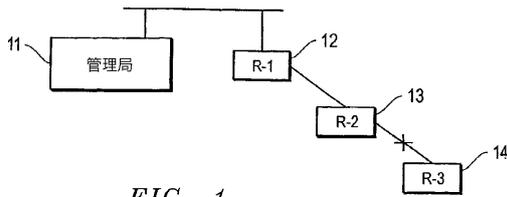


FIG. 1
(Prior Art)

【図2】

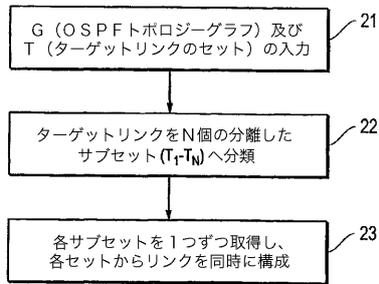


FIG. 2

【図3】

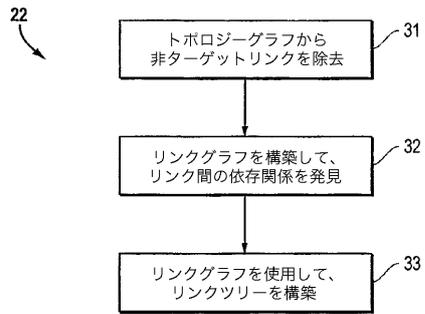


FIG. 3

【図4A】

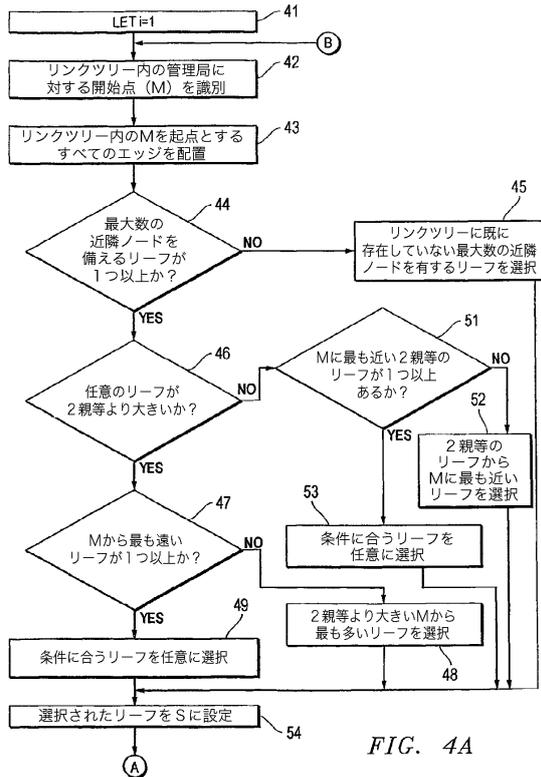


FIG. 4A

【図4B】

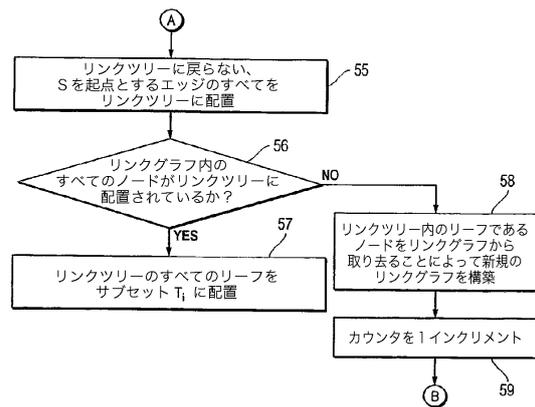


FIG. 4B

【図5】

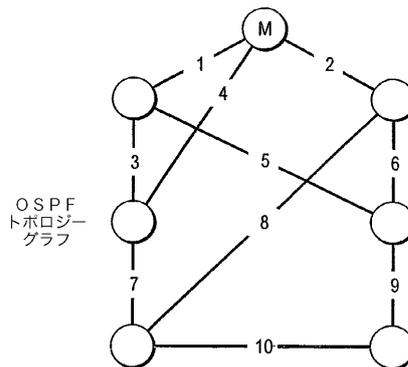


FIG. 5

【図 6】

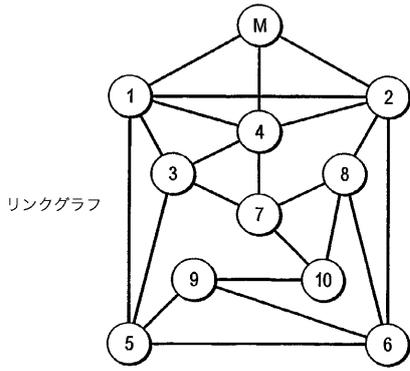


FIG. 6

【図 7】

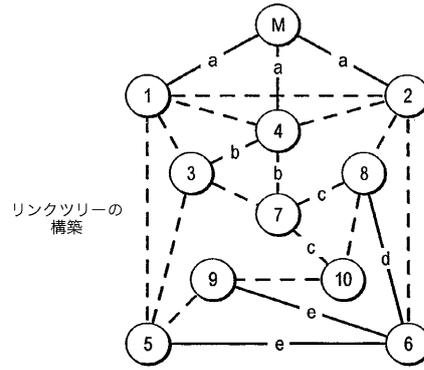


FIG. 7

【図 8】

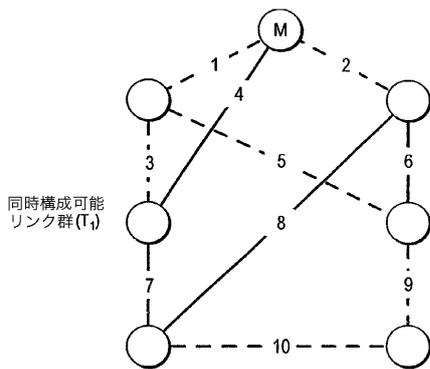


FIG. 8

【図 10】

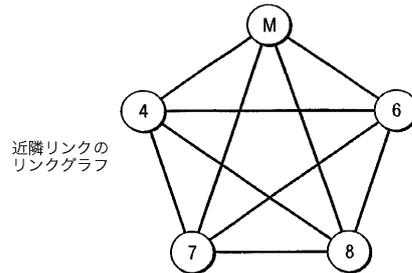


FIG. 10

【図 9】

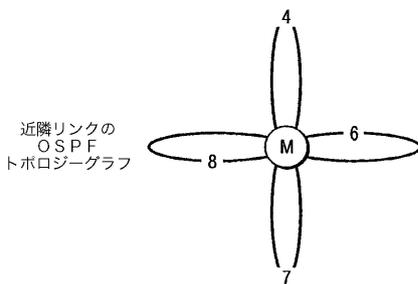


FIG. 9

【図 11】

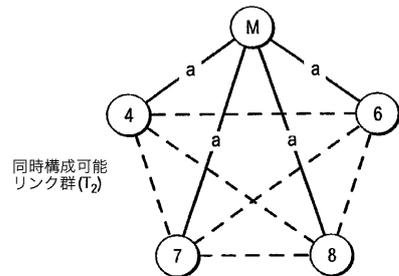


FIG. 11

【図 12】

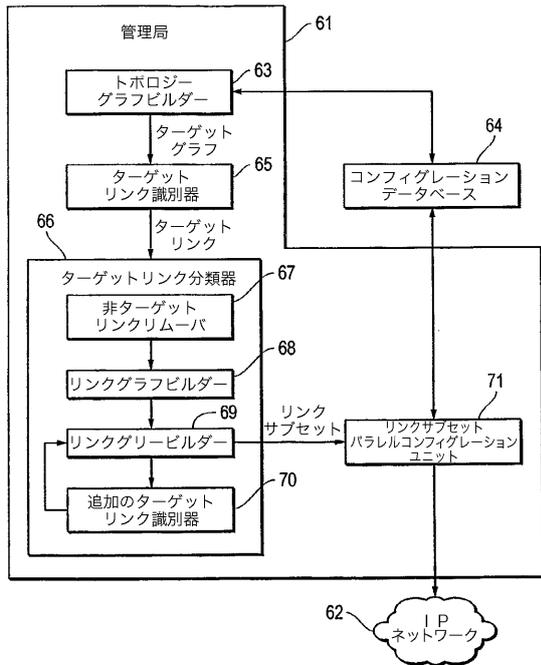


FIG. 12

フロントページの続き

- (72)発明者 モルナール, ゲルゲリー
ハンガリー共和国 ブダペスト エイチ - 1085 ペル. ユ. . 6 III / 37 / エー
- (72)発明者 トス, ガボール
ハンガリー共和国 スツィゲツェントミクロス エイチ - 2310, パプタグ. ユ. . 33
- (72)発明者 ゲーレ, バラツ, ペーター
ハンガリー共和国 ペクス エイチ - 7633, スタツィオン ユ. . 3 / エー
- (72)発明者 ノール, アティラ, ラユント
ハンガリー共和国 ブダペスト エイチ - 1039, III ズリンイ ユ. . 43

審査官 矢頭 尚之

- (56)参考文献 米国特許出願公開第2003/0043820 (US, A1)
国際公開第03/049342 (WO, A2)

- (58)調査した分野(Int.Cl., DB名)
H04L 12/56