

(19) 日本国特許庁(JP)

(12) 公表特許公報(A)

(11) 特許出願公表番号

特表2012-519424
(P2012-519424A)

(43) 公表日 平成24年8月23日(2012.8.23)

(51) Int.Cl.		F I				テーマコード (参考)
HO4W 40/38	(2009.01)	HO4Q	7/00	372		5K067
HO4W 84/18	(2009.01)	HO4Q	7/00	633		
HO4W 40/32	(2009.01)	HO4Q	7/00	364		

審査請求 有 予備審査請求 未請求 (全 44 頁)

(21) 出願番号 特願2011-552204 (P2011-552204)
 (86) (22) 出願日 平成22年2月26日 (2010.2.26)
 (85) 翻訳文提出日 平成23年10月26日 (2011.10.26)
 (86) 国際出願番号 PCT/US2010/025665
 (87) 国際公開番号 W02010/099492
 (87) 国際公開日 平成22年9月2日 (2010.9.2)
 (31) 優先権主張番号 61/155,868
 (32) 優先日 平成21年2月26日 (2009.2.26)
 (33) 優先権主張国 米国 (US)
 (31) 優先権主張番号 61/185,535
 (32) 優先日 平成21年6月9日 (2009.6.9)
 (33) 優先権主張国 米国 (US)
 (31) 優先権主張番号 12/712,983
 (32) 優先日 平成22年2月25日 (2010.2.25)
 (33) 優先権主張国 米国 (US)

(71) 出願人 595020643
 クォアルコム・インコーポレイテッド
 QUALCOMM INCORPORATED
 アメリカ合衆国、カリフォルニア州 92
 121-1714、サン・ディエゴ、モア
 ハウス・ドライブ 5775
 (74) 代理人 100108855
 弁理士 蔵田 昌俊
 (74) 代理人 100159651
 弁理士 高倉 成男
 (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】 拡張オーバーレイ状態維持のための方法および装置

(57) 【要約】

ピアツーピアオーバーレイネットワーク中の拡張オーバーレイ状態維持のための方法および装置。第1の方法は、第1のノードがオーバーレイネットワークを離れていることを推測することと、サイズカウンタ値をデクリメントさせるメッセージを送信することを含む。第2の方法は、オーバーレイネットワークの第1のノードに関するノードのセットを識別することと、ノードのセットの各ノードに関するセグメント長を取得することと、ノードのセット中のノードの総数を、セグメント長の合計で割ることによりオーバーレイネットワークのサイズを決定することを含む。第3の方法は、オーバーレイネットワークの第1のノードに関するノードのセットを識別することと、第1のノードと、ノードのセットの各ノードとに関するサイズ推定を取得することと、サイズ推定を平均することにより、オーバーレイネットワークのサイズを決定することを含む。

【選択図】 図5

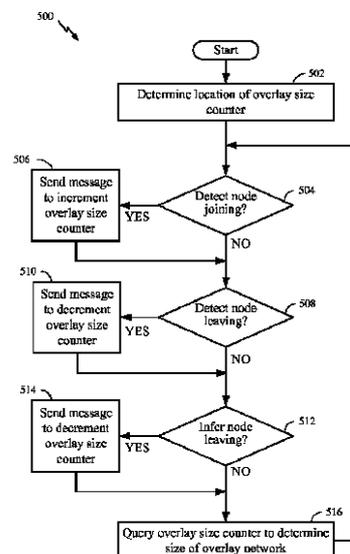


FIG. 5

【特許請求の範囲】**【請求項 1】**

ピアツーピアオーバーレイネットワークのサイズを決定するための方法において、第 1 のノードが前記オーバーレイネットワークを離れていることを推測することと、サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信することを含む方法。

【請求項 2】

前記推測することは、前記第 1 のノードとの通信が失敗したことを決定することを含む請求項 1 記載の方法。

【請求項 3】

前記推測することは、前記第 1 のノードが前記オーバーレイネットワークを離れているという表示を第三者から受信することを含む請求項 1 記載の方法。

【請求項 4】

第 2 のノードが前記オーバーレイネットワークに加わっていることを決定することと、前記サイズカウンタ値をインクリメントさせるインクリメントメッセージを送信することとをさらに含む請求項 1 記載の方法。

【請求項 5】

第 3 のノードが前記オーバーレイネットワークを離れていることを決定することと、前記サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信することとをさらに含む請求項 1 記載の方法。

【請求項 6】

前記サイズカウンタ値を問い合わせ、前記オーバーレイネットワークのサイズを決定することをさらに含む請求項 1 記載の方法。

【請求項 7】

前記サイズカウンタ値を問い合わせることは、よく知られているストリングをハッシングして、前記サイズカウンタ値を導出することを含む請求項 6 記載の方法。

【請求項 8】

ピアツーピアオーバーレイネットワークのサイズを決定するための装置において、第 1 のノードが前記オーバーレイネットワークを離れていることを推測するように構成されているプロセッサと、前記プロセッサに結合されており、サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信するように構成されている送信機とを具備する装置。

【請求項 9】

前記プロセッサは、前記第 1 のノードとの通信が失敗したことを決定することにより、推測するように構成されている請求項 8 記載の装置。

【請求項 10】

前記プロセッサは、前記第 1 のノードが前記オーバーレイネットワークを離れているという表示を第三者から受信することにより、推測するように構成されている請求項 8 記載の装置。

【請求項 11】

前記プロセッサは、第 2 のノードが前記オーバーレイネットワークに加わっていることを決定するように構成され、前記送信機は、前記サイズカウンタ値をインクリメントさせるインクリメントメッセージを送信するように構成されている請求項 8 記載の装置。

【請求項 12】

前記プロセッサは、第 3 のノードが前記オーバーレイネットワークを離れていることを決定するように構成され、前記送信機は、前記サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信するように構成されている請求項 8 記載の装置。

【請求項 13】

10

20

30

40

50

前記プロセッサは、前記サイズカウンタ値を問い合わせ、前記オーバーレイネットワークのサイズを決定するように構成されている請求項 8 記載の装置。

【請求項 14】

前記プロセッサは、よく知られているストリングをハッシングして、前記サイズカウンタ値を導出することにより、前記サイズカウンタ値を問い合わせるように構成されている請求項 13 記載の装置。

【請求項 15】

ピアツーピアオーバーレイネットワークのサイズを決定するための装置において、第 1 のノードが前記オーバーレイネットワークを離れていることを推測する手段と、サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信する手段とを具備する装置。

10

【請求項 16】

前記推測する手段は、前記第 1 のノードとの通信が失敗したことを決定する手段を備える請求項 15 記載の装置。

【請求項 17】

前記推測する手段は、前記第 1 のノードが前記オーバーレイネットワークを離れているという表示を第三者から受信する手段を備える請求項 15 記載の装置。

【請求項 18】

第 2 のノードが前記オーバーレイネットワークに加わっていることを決定する手段と、前記サイズカウンタ値をインクリメントさせるインクリメントメッセージを送信する手段とをさらに具備する請求項 15 記載の装置。

20

【請求項 19】

第 3 のノードが前記オーバーレイネットワークを離れていることを決定する手段と、前記サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信する手段とをさらに具備する請求項 15 記載の装置。

【請求項 20】

前記サイズカウンタ値を問い合わせ、前記オーバーレイネットワークのサイズを決定する手段をさらに具備する請求項 15 記載の装置。

【請求項 21】

前記サイズカウンタ値を問い合わせる手段は、よく知られているストリングをハッシングして、前記サイズカウンタ値を導出する手段を備える請求項 20 記載の装置。

30

【請求項 22】

ピアツーピアオーバーレイネットワークのサイズを決定するためのコンピュータプログラムプロダクトにおいて、

第 1 のノードが前記オーバーレイネットワークを離れていることを推測するための、プロセッサにより実行可能なコードと、

サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信するための、前記プロセッサにより実行可能なコードとを具現化するコンピュータ読取可能媒体を含むコンピュータプログラムプロダクト。

【請求項 23】

前記コードは、前記プロセッサに、前記第 1 のノードとの通信が失敗したことを決定させることにより推測させるように構成されている請求項 22 記載のコンピュータ読取可能媒体。

40

【請求項 24】

前記コードは、前記プロセッサに、前記第 1 のノードが前記オーバーレイネットワークを離れているという表示を第三者から受信させることにより推測させるように構成されている請求項 22 記載のコンピュータ読取可能媒体。

【請求項 25】

前記コードは、

前記プロセッサに、第 2 のノードが前記オーバーレイネットワークに加わっていること

50

を決定させるように構成され、

前記プロセッサに、前記サイズカウンタ値をインクリメントさせるインクリメントメッセージを送信させるように構成されている請求項 2 2 記載のコンピュータ読取可能媒体。

【請求項 2 6】

前記コードは、

前記プロセッサに、第 3 のノードが前記オーバーレイネットワークを離れていることを決定させるように構成され、

前記プロセッサに、前記サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信させるように構成されている請求項 2 2 記載のコンピュータ読取可能媒体。

【請求項 2 7】

前記コードは、前記プロセッサに、前記サイズカウンタ値を問い合わせ、前記オーバーレイネットワークのサイズを決定させるように構成されている請求項 2 2 記載のコンピュータ読取可能媒体。

【請求項 2 8】

前記コードは、前記プロセッサに、よく知られているストリングをハッシングさせて、前記サイズカウンタ値を導出させることにより、前記サイズカウンタ値を問い合わせさせるように構成されている請求項 2 7 記載のコンピュータ読取可能媒体。

【請求項 2 9】

ピアツーピアオーバーレイネットワークのサイズを決定するための方法において、オーバーレイネットワークの第 1 のノードに関するノードのセットを識別することと

、前記ノードのセットのそれぞれのノードに関するセグメント長を取得することと、

前記ノードのセット中のノードの総数を、前記セグメント長の合計で割ることにより、前記オーバーレイネットワークのサイズを決定することを含む方法。

【請求項 3 0】

前記ノードのセットは、前記第 1 のノードに関するフィンガーを含む請求項 2 9 記載の方法。

【請求項 3 1】

前記ノードのセットは、前記第 1 のノードに関する隣接しているものを含む請求項 2 9 記載の方法。

【請求項 3 2】

前記ノードのセットは、前記第 1 のノードと通信中の 1 つ以上のノードを含む請求項 2 9 記載の方法。

【請求項 3 3】

前記取得することは、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせ、前記セグメント長の少なくとも一部分を取得することを含む請求項 2 9 記載の方法。

【請求項 3 4】

前記取得することは、それぞれ、前記ノードのセットの少なくとも一部分とのビジーバック通信中に、前記セグメント長の少なくとも一部分を取得することを含む請求項 2 9 記載の方法。

【請求項 3 5】

ピアツーピアオーバーレイネットワークのサイズを決定するための装置において、オーバーレイネットワークの第 1 のノードに関するノードのセットを識別するように構成されているプロセッサと、

前記プロセッサに結合されており、前記ノードのセットのそれぞれのノードに関するセグメント長を取得するように構成されているトランシーバと、

前記ノードのセット中のノードの総数を、前記セグメント長の合計で割ることにより、前記オーバーレイネットワークのサイズを決定するように構成されている前記プロセッサとを具備する装置。

【請求項 3 6】

10

20

30

40

50

前記ノードのセットは、前記第1のノードに係するフィンガーを含む請求項35記載の装置。

【請求項37】

前記ノードのセットは、前記第1のノードに係する隣接しているものを含む請求項35記載の装置。

【請求項38】

前記ノードのセットは、前記第1のノードと通信中の1つ以上のノードを含む請求項35記載の装置。

【請求項39】

前記ランシーバは、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせ、前記セグメント長の少なくとも一部分を取得するように構成されている請求項35記載の装置。

10

【請求項40】

前記ランシーバは、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記セグメント長の少なくとも一部分を取得するように構成されている請求項35記載の装置。

【請求項41】

ピアツーピアオーバーレイネットワークのサイズを決定するための装置において、オーバーレイネットワークの第1のノードに係するノードのセットを識別する手段と

20

、前記ノードのセットのそれぞれのノードに係するセグメント長を取得する手段と、前記ノードのセット中のノードの総数を、前記セグメント長の合計で割ることにより、前記オーバーレイネットワークのサイズを決定する手段とを具備する装置。

【請求項42】

前記ノードのセットは、前記第1のノードに係するフィンガーを含む請求項41記載の装置。

【請求項43】

前記ノードのセットは、前記第1のノードに係する隣接しているものを含む請求項41記載の装置。

【請求項44】

30

前記ノードのセットは、前記第1のノードと通信中の1つ以上のノードを含む請求項41記載の装置。

【請求項45】

前記取得する手段は、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせ、前記セグメント長の少なくとも一部分を取得する手段を備える請求項41記載の装置。

【請求項46】

前記取得する手段は、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記セグメント長の少なくとも一部分を取得する手段を備える請求項41記載の装置。

40

【請求項47】

ピアツーピアオーバーレイネットワークのサイズを決定するためのコンピュータプログラムプロダクトにおいて、

オーバーレイネットワークの第1のノードに係するノードのセットを識別するためのプロセッサにより実行可能なコードと、

前記ノードのセットのそれぞれのノードに係するセグメント長を取得するための前記プロセッサにより実行可能なコードと、

前記ノードのセット中のノードの総数を、前記セグメント長の合計で割ることにより、前記オーバーレイネットワークのサイズを決定するための、前記プロセッサにより実行可能なコードとを具現化するコンピュータ読取可能媒体を含むコンピュータプログラムプロ

50

ダクト。

【請求項 48】

前記ノードのセットは、前記第 1 のノードに係するフィンガーを含む請求項 47 記載のコンピュータ読取可能媒体。

【請求項 49】

前記ノードのセットは、前記第 1 のノードに係する隣接しているものを含む請求項 47 記載のコンピュータ読取可能媒体。

【請求項 50】

前記ノードのセットは、前記第 1 のノードと通信中の 1 つ以上のノードを含む請求項 47 記載のコンピュータ読取可能媒体。

10

【請求項 51】

前記コードは、前記プロセッサに、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせさせて、前記セグメント長の少なくとも一部分を取得させるように構成されている請求項 47 記載のコンピュータ読取可能媒体。

【請求項 52】

前記コードは、前記プロセッサに、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記セグメント長の少なくとも一部分を取得させるように構成されている請求項 47 記載のコンピュータ読取可能媒体。

【請求項 53】

ピアツーピアオーバーレイネットワークのサイズを決定するための方法において、
オーバーレイネットワークの第 1 のノードに係するノードのセットを識別することと

20

、
前記第 1 のノードに係するサイズ推定と、前記ノードのセットのそれぞれのノードに係するサイズ推定とを取得することと、

前記サイズ推定を平均することにより、前記オーバーレイネットワークのサイズを決定することを含む方法。

【請求項 54】

前記ノードのセットは、前記第 1 のノードに係するフィンガーを含む請求項 53 記載の方法。

【請求項 55】

前記ノードのセットは、前記第 1 のノードに係する隣接しているものを含む請求項 53 記載の方法。

30

【請求項 56】

前記ノードのセットは、前記第 1 のノードと通信中の 1 つ以上のノードを含む請求項 53 記載の方法。

【請求項 57】

前記取得することは、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせ、前記サイズ推定の少なくとも一部分を取得することを含む請求項 53 記載の方法。

【請求項 58】

前記取得することは、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記サイズ推定の少なくとも一部分を取得することを含む請求項 53 記載の方法。

40

【請求項 59】

ピアツーピアオーバーレイネットワークのサイズを決定するための装置において、
オーバーレイネットワークの第 1 のノードに係するノードのセットを識別し、前記第 1 のノードに係するサイズ推定を取得するように構成されているプロセッサと、

前記プロセッサに結合されており、前記ノードのセットのそれぞれのノードに係するサイズ推定を取得するように構成されているトランシーバと、

前記サイズ推定を平均することにより、前記オーバーレイネットワークのサイズを決定するように構成されている前記プロセッサとを具備する装置。

50

【請求項 6 0】

前記ノードのセットは、前記第 1 のノードに関係するフィンガーを含む請求項 5 9 記載の装置。

【請求項 6 1】

前記ノードのセットは、前記第 1 のノードに関係する隣接しているものを含む請求項 5 9 記載の装置。

【請求項 6 2】

前記ノードのセットは、前記第 1 のノードと通信中の 1 つ以上のノードを含む請求項 5 9 記載の装置。

【請求項 6 3】

前記トランシーバは、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせ、前記サイズ推定の少なくとも一部分を取得するように構成されている請求項 5 9 記載の装置。

【請求項 6 4】

前記トランシーバは、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記サイズ推定の少なくとも一部分を取得するように構成されている請求項 5 9 記載の装置。

【請求項 6 5】

ピアツーピアオーバーレイネットワークのサイズを決定するための装置において、オーバーレイネットワークの第 1 のノードに関係するノードのセットを識別する手段と

、前記第 1 のノードに関係するサイズ推定と、前記ノードのセットのそれぞれのノードに関係するサイズ推定とを取得する手段と、

前記サイズ推定を平均することにより、前記オーバーレイネットワークのサイズを決定する手段とを具備する装置。

【請求項 6 6】

前記ノードのセットは、前記第 1 のノードに関係するフィンガーを含む請求項 6 5 記載の装置。

【請求項 6 7】

前記ノードのセットは、前記第 1 のノードに関係する隣接しているものを含む請求項 6 5 記載の装置。

【請求項 6 8】

前記ノードのセットは、前記第 1 のノードと通信中の 1 つ以上のノードを含む請求項 6 5 記載の装置。

【請求項 6 9】

前記取得する手段は、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせ、前記サイズ推定の少なくとも一部分を取得する手段を備える請求項 6 5 記載の装置。

【請求項 7 0】

前記取得する手段は、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記サイズ推定の少なくとも一部分を取得する手段を備える請求項 6 5 記載の装置。

【請求項 7 1】

ピアツーピアオーバーレイネットワークのサイズを決定するためのコンピュータプログラムプロダクトにおいて、

オーバーレイネットワークの第 1 のノードに関係するノードのセットを識別するための、プロセッサにより実行可能なコードと、

前記第 1 のノードに関係するサイズ推定を取得するための、前記プロセッサにより実行可能なコードと、

前記ノードのセットのそれぞれのノードに関係するサイズ推定を取得するための、前記プロセッサにより実行可能なコードと、

10

20

30

40

50

前記サイズ推定を平均することにより、前記オーバーレイネットワークのサイズを決定するための、前記プロセッサにより実行可能なコードとを具現化するコンピュータ読取可能媒体を含むコンピュータプログラムプロダクト。

【請求項 7 2】

前記ノードのセットは、前記第 1 のノードに関係するフィンガーを含む請求項 7 1 記載のコンピュータ読取可能媒体。

【請求項 7 3】

前記ノードのセットは、前記第 1 のノードに関係する隣接しているものを含む請求項 7 1 記載のコンピュータ読取可能媒体。

【請求項 7 4】

前記ノードのセットは、前記第 1 のノードと通信中の 1 つ以上のノードを含む請求項 7 1 記載のコンピュータ読取可能媒体。

【請求項 7 5】

前記コードは、前記プロセッサに、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせさせて、前記サイズ推定の少なくとも一部分を取得させるように構成されている請求項 7 1 記載のコンピュータ読取可能媒体。

【請求項 7 6】

前記コードは、前記プロセッサに、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記サイズ推定の少なくとも一部分を取得させるように構成されている請求項 7 1 記載のコンピュータ読取可能媒体。

【発明の詳細な説明】

【米国特許法第 1 1 9 条の下での優先権主張】

【0 0 0 1】

本特許出願は、双方が、本発明の譲受人に譲渡され、ここでの参照によりここに明確に組み込まれている、“ピアツーピアオーバーレイネットワークのサイズ推定のための方法および装置”と題し、2009年2月26日に出願された仮出願第61/155,868号、および、“オーバーレイルーティング安定化およびサイズ推定に対する、適応自己同調可能アプローチのための方法および装置”と題し、2009年6月9日に出願された仮出願第61/185,535号に対して優先権を主張する。

【分野】

【0 0 0 2】

本出願は、一般的に、オーバーレイネットワークの動作に関連し、さらに詳細には、ピアツーピアオーバーレイネットワーク中の拡張オーバーレイ状態維持のための方法および装置に関連している。

【背景】

【0 0 0 3】

サーバベースのインフラストラクチャなしで、メンバーノードがサービスを取得するネットワークは、“ピアツーピア”オーバーレイネットワークとして呼ばれる。ピアツーピアオーバーレイでは、ピアノードは、互いに協働して、サービスを提供し、ネットワークを維持する。ピアツーピアオーバーレイネットワークは、インターネットプロトコル(IP)を利用するネットワークのような基礎となるネットワークのトップ上に築くことができる。

【0 0 0 4】

ピアツーピアオーバーレイネットワークでは、それぞれのノードは、オーバーレイ中に参加している 1 個以上のピアの知識を有する。オーバーレイネットワーク上で発信元ノードから宛先ノードにデータをルーティングするための、シンプルだが非効率的なアプローチは、宛先ノードに到達するまで、次のホップへ、または、後継ノード(すなわち、識別スペース中の発信元ノードの論理的に隣のノード)へデータを継続的にパスすることである。しかしながら、このアプローチは、オーバーレイネットワークのサイズ、および、ホップの数が増加するにつれて、過大な待ち時間を招く。それゆえ、ルーティング最適化の

10

20

30

40

50

ために、それぞれのノードは、それ自身から、1個のノード、2個のノード、4個のノード、または、 $2^{(m-1)}$ 個までのノード分離れているフィンガーのリストを維持する。ここで、 m は、それぞれのノード識別子に割り当てられたビットの数である。これは、ノード $O(n)$ からノード $O(\log(n))$ にデータをルーティングするために必要とされるホップの数を最小化するのを助ける。ここで、 n は、平均的なケースに対するノードの数である。

【0005】

しかしながら、ノードは、いつでも来たり行ったりし、結果として、オーバーレイネットワークコンフィギュレーションが変化し、この変化は、特定のノードに知られているフィンガーに影響するかもしれない。この可変性を補うために、それぞれのノードは、そのノードが知るフィンガーを再計算するフィンガー安定化アルゴリズムを再ランする。フィンガー安定化アルゴリズムの結果は、フィンガーテーブルに記憶させる。残念ながら、それぞれのノードは、それ自身のフィンガーテーブルを維持していることから、それぞれのノードがそのフィンガー安定化アルゴリズムをランさせる必要がある頻度を決定することは問題になるかもしれない。例えば、ノードがそのフィンガー安定化アルゴリズムを十分頻繁にランしない場合、そのフィンガーテーブルは古くなり、結果として、非効率的で遅れたパケットルーティングが生じるかもしれない。ノードがそのフィンガー安定化アルゴリズムを頻繁にランしすぎる場合、これは、結果として、オーバーレイ帯域幅を無駄にすること、および/または、オーバーレイネットワーク上の他のノード上に負担をかけることになるかもしれない。例えば、安定化アルゴリズムをランするには、電力を必要とし、安定化アルゴリズムの過大な実行は、バッテリーで動作するノードにおける電力を無駄にするかもしれない。

10

20

【0006】

さらに、構造化ピアツーピアオーバーレイネットワークは、本質的に広く分散されていることから、参加するノードは、完全なルーティングテーブルを有しておらず、したがって、それらが動作しているオーバーレイネットワークのサイズを知らない。しかしながら、オーバーレイネットワークのサイズの知識は、オーバーレイをマージするとき、負荷バランシングを実施するとき、または、ストラテジーをキャッシングするときとともに、ルーティングプロトコル特殊化のような、いくつかの目的に対して有用であることがある。

【0007】

残念ながら、従来のシステムは、正確なオーバーレイネットワークサイズを提供することに失敗することがある。例えば、ノードは、それらの隣接するものに進んで知らせることなく、オーバーレイを離れるかもしれない、結果として、オーバーレイネットワークサイズは、正確に維持されないかもしれない。

30

【0008】

それゆえ、ノードが、フィンガー安定化アルゴリズムを適応的にスケジューリングし、オーバーレイネットワークのサイズを決定できるようにして、上述した問題を克服するように動作する、シンプルでコスト効率のよいメカニズムを有することが望ましいだろう。

【概要】

【0009】

1つ以上の態様では、ノードが、フィンガー安定化アルゴリズムの実行間の時間間隔を決定できるようにし、それによって、アルゴリズムの実行を適応的にスケジューリングするように適応的に動作する、方法および装置を備える適応スケジューリング(AS)システムを提供する。システムはまた、ノードが、その上にこれらが参加するオーバーレイネットワークのサイズを決定でき、それによって、マージング、負荷バランシング、およびキャッシングのようなネットワーク機能を促進するように動作する。

40

【0010】

ある態様では、ピアツーピアオーバーレイネットワークのサイズを決定するための方法を提供する。方法は、第1のノードがオーバーレイネットワークを離れていることを推測することと、サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信する

50

こととを含む。

【0011】

ある態様では、ピアツーピアオーバーレイネットワークのサイズを決定するための装置を提供する。装置は、第1のノードがオーバーレイネットワークを離れていることを推測するように構成されているプロセッサと、プロセッサに結合されており、サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信するように構成されている送信機とを具備する。

【0012】

ある態様では、ピアツーピアオーバーレイネットワークのサイズを決定するための方法を提供する。方法は、オーバーレイネットワークの第1のノードに関するノードのセットを識別することと、ノードのセットのそれぞれのノードに関するセグメント長を取得することと、ノードのセット中のノードの総数を、セグメント長の合計で割ることにより、オーバーレイネットワークのサイズを決定することを含む。

10

【0013】

ある態様では、ピアツーピアオーバーレイネットワークのサイズを決定するための装置を提供する。装置は、オーバーレイネットワークの第1のノードに関するノードのセットを識別するように構成されているプロセッサと、プロセッサに結合されており、ノードのセットのそれぞれのノードに関するセグメント長を取得するように構成されているトランシーバとを具備し、プロセッサは、ノードのセット中のノードの総数を、セグメント長の合計で割ることにより、オーバーレイネットワークのサイズを決定するように構成されている。

20

【0014】

ある態様では、ピアツーピアオーバーレイネットワークのサイズを決定するための方法を提供する。方法は、オーバーレイネットワークの第1のノードに関するノードのセットを識別することと、第1のノードに関するサイズ推定と、ノードのセットのそれぞれのノードに関するサイズ推定とを取得することと、サイズ推定を平均することにより、オーバーレイネットワークのサイズを決定することを含む。

【0015】

ある態様は、ピアツーピアオーバーレイネットワークのサイズを決定するための装置を提供する。装置は、オーバーレイネットワークの第1のノードに関するノードのセットを識別し、第1のノードに関するサイズ推定を取得するように構成されているプロセッサと、プロセッサに結合されており、ノードのセットのそれぞれのノードに関するサイズ推定を取得するように構成されているトランシーバとを具備し、プロセッサは、サイズ推定を平均することにより、オーバーレイネットワークのサイズを決定するように構成されている。

30

【0016】

これから述べる図面の簡単な説明、説明、および特許請求の範囲を参照した後、他の態様が明らかになるだろう。

【図面の簡単な説明】

【0017】

ここに記述した前述の態様は、付随する図面を参照する際の下記の記述への参照により、さらに容易に明らかになるだろう。

40

【図1】図1は、ASシステムの動作を図示するネットワークを示している。

【図2】図2は、ASシステムにしたがって構築された例示的なAS装置を示している。

【図3】図3は、ASシステムにしたがって、フィンガー安定化アルゴリズムを適応的にスケジューリングするための例示的な方法を示している。

【図4】図4は、ASシステムにしたがって構築された例示的なAS装置を示している。

【図5】図5は、ASシステムにしたがって、オーバーレイネットワークサイズを決定するための第1の方法を示している。

【図6】図6は、ASシステムにしたがって、オーバーレイネットワークサイズを決定す

50

るための第 2 の方法を示している。

【図 7】図 7 は、A S システムにしたがって、オーバーレイネットワークサイズを決定するための第 1 の改善方法を示している。

【図 8】図 8 は、A S システムにしたがって、オーバーレイネットワークサイズを決定するための第 2 の改善方法を示している。

【図 9】図 9 は、A S システムにしたがって構築された例示的な A S 装置を示している。

【図 10】図 10 は、A S システムにしたがって構築された例示的な A S 装置を示している。

【図 11】図 11 は、A S システムにしたがって構築された例示的な A S 装置を示している。

【説明】

【0018】

下記の説明は、フィンガー安定化アルゴリズムの実行を適応的にスケジューリングし、オーバーレイネットワークのサイズを決定する、A S システムの態様およびインプリメンテーションを記述している。

【0019】

図 1 は、A S システムの動作を図示するネットワーク 100 を示している。ネットワーク 100 は、インターネットプロトコルネットワークのような何らかのタイプのネットワークを含む基礎となるネットワーク 102 を備える。基礎となるネットワーク 102 を、単一のエンティティとして示しているが、基礎となるネットワークは、WAN、LAN、ワイヤレスネットワーク、および/または他の何らかのタイプのネットワークのような、任意の数のネットワークまたは任意のタイプのネットワークを含んでいてもよい。

【0020】

ピアツーピアオーバーレイネットワーク 104 は、基礎となるネットワーク 102 のノードのサブセットを備え、基礎となるネットワーク 102 のサービスを利用して、それらのノードが通信できるようにするように動作する。ピアツーピアオーバーレイネットワーク 104 では、ノードは、通信リンク 106 により接続され、円形ルーティングパスを形成する。通信リンク 106 は、基礎となるネットワーク 102 により提供される安全なトンネルであってもよい。ピアツーピアオーバーレイネットワーク 104 は、基礎となるネットワーク 102 のものとは異なる、許可および対話のセットとともに動作する。ピアツーピアオーバーレイネットワーク 104 は、何らかのルーティングパターンを可能にする何らかのトポロジーまたはアーキテクチャを有するかもしれないが、この何らかのルーティングパターンは、図 1 中に示されているルーティングに限定されないことにも留意すべきである。

【0021】

ピアツーピアオーバーレイネットワーク 104 中のノードのそれぞれは、ノード識別子を確立する。説明の簡略化および簡潔のために、ピアツーピアオーバーレイネットワーク 104 のノードのノード識別子は、(1、4、7、10、13、16、19、および 22) である。実際には、オーバーレイネットワークは、非常に大多数のノードを備え、より多くのノード識別子を利用することに留意すべきである。動作中に、トラフィックは、ピアツーピアオーバーレイネットワーク 104 の回りをいずれの方向にも流れることができる。

【0022】

トラフィックルーティングを促進するために、ネットワーク 104 のそれぞれのノードは、オーバーレイネットワーク 104 全体に及ぶルーティングフィンガーを識別するフィンガーテーブルを計算して維持する。例えば、それぞれのノードは、フィンガー安定化アルゴリズムをランして、オーバーレイネットワークの現在のコンフィギュレーションに基づいてルーティングフィンガーを識別する。したがって、ノードがオーバーレイに加わるおよび離れるとき、フィンガー安定化アルゴリズムは、それぞれのノードに関係するルーティングフィンガーを識別するだろう。例えば、ノード 4 においてランしているフィンガ

10

20

30

40

50

安定化アルゴリズムは、ノード13に対するフィンガー108と、ノード19に対するフィンガー110とを識別している。フィンガーの使用は、オーバーレイネットワーク104にわたって、より効率的なパケットルーティングを提供する。例えば、ノード16にルーティングされることになっている、ノード4におけるパケットは、第1のホップ中で、フィンガー108をわたってルーティングすることができ、その後、ルーティングパス114により図示されているように、第2のホップ中で、ノード16にルーティングすることができる。

【0023】

オーバーレイネットワーク104のノードのそれぞれにおいて、AS装置を提供する。簡略化のために、オーバーレイネットワーク104のノード4においてAS装置112を示している。しかしながら、類似したAS装置が、オーバーレイネットワーク104のそれぞれのノードに位置していてもよい。AS装置112は、ノード4において実施するフィンガー安定化アルゴリズムの実行を適応的にスケジューリングするように動作する。AS装置112は、何らかのタイプのフィンガー安定化アルゴリズムとの使用に適しており、その動作の簡単な説明を下記に提供する。

10

【0024】

AS装置112は、フィンガー安定化アルゴリズムの実行間の初期の時間間隔またはベースの時間間隔（すなわち t 秒）で開始する。安定化アルゴリズムをランするとき、ノード4は、 N 個のフィンガーを発見する。AS装置112は、 t 秒間、時間間隔を測定して、フィンガー安定化アルゴリズムを再度実行するようにノード4をトリガする。2個のフィンガー結果の間の差分（すなわち、決定されたフィンガーの、数および/またはタイプ）が、第1の基準を満たす場合には、AS装置112は、フィンガー安定化アルゴリズムの実行間の時間間隔を増加させるように動作する。例えば、1つのインプリメンテーションでは、時間間隔を、2倍だけ増加させる。しかしながら、差分が第2の基準を満たす場合には、AS装置112は、フィンガー安定化アルゴリズムの実行間の時間間隔を減少させるように動作する。例えば、1つのインプリメンテーションでは、時間間隔を、 $1/2$ に減少させる。第1および第2の基準は、任意の適切な性能目標を満たすように規定することができる。例えば、2個のフィンガー結果の間に差分がない場合に、第1の基準を満たしてもよい。2個のフィンガー結果の間に何らかの差分が見つかる場合に、第2の基準を満たしてもよい。したがって、特定のノードのフィンガーテーブル中の変化に関する条件の何らかのセットをテストするために、第1および第2の基準を規定することも可能である。

20

30

【0025】

それゆえ、この適応自己同調可能アプローチを通して、AS装置112は、それぞれのノードが、フィンガー安定化アルゴリズムの実行間の安定状態間隔に、独立的、かつ、適応的に、達することができるようにする。さらに、システムは、それぞれのノードが、オーバーレイのサイズを独立的に近似できるようにする。例えば、特定のノードが、時間 t において、オーバーレイネットワーク中に N 個のフィンガーを識別した場合には、その時間において、オーバーレイネットワーク中に多くて 2^N 個のノードがある。したがって、特定のノードが、3個の一意的なフィンガーを識別した場合に、ノードは、オーバーレイネットワークのサイズを $8(2^3)$ 個のノードに近似する。しかしながら、この近似は、時間とともに変化することになり、オーバーレイネットワークの近似サイズ範囲をノードに提供するように機能する。

40

【0026】

ASシステムは、フィンガー安定化アルゴリズムをスケジューリングするために、ノード間の関連、または、システムのグローバルな知識を必要とせず、このことは、ピアツーピア分散アプリケーションに対して有益である。AS装置112の動作のさらに詳細な説明を下記に提供する。

【0027】

図2は、ASシステムにしたがって構築された例示的なAS装置200を示している。

50

例えば、AS装置200は、図1中に示したAS装置112としての使用に適している。AS装置200は、データバス210を使用して通信するようにすべて結合されている、プロセッサ202と、メモリ204と、タイマー206と、トランシーバ208とを備えている。AS装置200は、1つのインプリメンテーションに過ぎないことと、他のインプリメンテーションも可能性があることに留意すべきである。

【0028】

ピアツーピアオーバーレイネットワーク上で、AS装置200が、データまたは他の情報を複数のノードと通信できるようにするように動作する、ハードウェア、および/または、ハードウェア実行ソフトウェアを、トランシーバ208は備えている。トランシーバ208は、フィンガー安定化アルゴリズムまたはネットワークサイズ推定を実施する目的として、ピアツーピアオーバーレイネットワークのノードとの1つ以上の通信リンク212を確立するように動作可能である。例えば、通信リンク212は、基礎となるIPネットワークのサービスを利用して形成される安全なトンネルであってもよい。

10

【0029】

メモリ204は、ASシステムの動作中に情報の記憶および取出ができるようにするように動作可能な何らかの適切な記憶デバイスを含んでいる。メモリ204は、オーバーレイネットワークパラメータ114を記憶するように動作する。オーバーレイネットワークパラメータ114は、ノード識別子、基礎となるネットワーク識別子、サービス識別子、および他の何らかのパラメータを含む、オーバーレイネットワークについての情報、あるいは、オーバーレイの動作または使用に関連する情報を含む。オーバーレイネットワークパラメータ114はまた、ASシステムの動作の間に使用する、第1および第2のセットの基準を含む。例えば、基準のセットを、プロセッサ202によりメモリ204中に記憶させる。プロセッサ202はまた、基準のセットの更新、変更、または、他の修正を行うように動作可能である。フィンガー安定化アルゴリズム218の実行間の時間間隔をいつ増加または減少させるかを決定するために、ASシステムの動作中に基準のセットを使用する。

20

【0030】

メモリ204はまた、ピアツーピアオーバーレイネットワークの1つ以上のノードに関するフィンガー情報を含むフィンガーデータベース216を記憶するように動作可能である。フィンガーデータベース216は、ノードに対して利用可能なフィンガーの数およびタイプについての情報を記憶させるために使用する。例えば、フィンガーデータベース216は、フィンガーの数と、フィンガーのタイプと、フィンガーエンドノードのような情報、および、オーバーレイネットワークのフィンガーに関連する他の何らかの情報を含む。フィンガーデータベース216中の情報を、プロセッサ202によるフィンガー安定化アルゴリズム218の実行から決定する。

30

【0031】

タイマー206は、プロセッサ202から受信した時間パラメータに基づいて、時間間隔を測定するように動作する、ハードウェア、および/または、ハードウェア実行ソフトウェアを含む。例えば、時間パラメータは、カウンタを初期化するために使用するカウントダウン値を含む。カウントダウン値は、タイマー206により測定されることになる特定の時間間隔に対応する。したがって、プロセッサ202は、タイマー206を設定して、任意の望ましい時間間隔を測定してもよい。時間間隔を測定したとき、タイマー206は、プロセッサ202に対してタイマー満了を示す。例えば、タイマー206は特定の時間間隔を測定し、その終わりにおいて、プロセッサ202は、その終わりを知らされて、その後フィンガー安定化アルゴリズム218を実行するように動作する。フィンガー安定化アルゴリズム218は、特定のノードに関するフィンガーについての情報を決定するように動作する。

40

【0032】

プロセッサ202は、少なくとも1つの、CPU、プロセッサ、ゲートアレイ、ハードウェア論理、メモリエlement、および/またはハードウェア実行ソフトウェアを含む。

50

プロセッサ 202 は、オーバーレイネットワーク中の特定のノードに対して利用可能なフィンガーを決定するように動作する。例えば、プロセッサ 202 は、フィンガー安定化アルゴリズム 218 を実行し、決定したフィンガーについての情報をフィンガーデータベース 216 中に記憶させる。プロセッサ 202 はまた、フィンガー安定化アルゴリズムにより決定したフィンガー決定を比較して、フィンガー安定化アルゴリズムを実施する前に、時間間隔を増加させるか、または、減少させるかを決定するように動作する。プロセッサ 202 は、タイマー 206 を制御して、この時間間隔を測定する。プロセッサ 202 はまた、オーバーレイサイズ推定のための 1 つ以上の方法を実施するように動作する。

【0033】

適応時間間隔決定

AS システムの動作中に、タイマー 206 は時間間隔を測定するように動作し、その後、プロセッサ 202 はフィンガー安定化アルゴリズム 218 を実行する。例えば、タイマー 206 は、時間間隔が終了したまたは満期になったことを、プロセッサ 202 にシグナリングする。プロセッサ 202 は、オーバーレイネットワークパラメータ 214 を利用して、フィンガー安定化アルゴリズム 218 を実行するために使用する情報を決定するように動作する。結果として生じるフィンガー決定を、フィンガーデータベース 216 中に記憶させる。

【0034】

プロセッサ 202 は、その後、フィンガー安定化アルゴリズムの次の実行の前に、測定することになる時間間隔を決定する。プロセッサ 202 は、フィンガー安定化アルゴリズムの 2 個のフィンガー結果を比較することにより、次の時間間隔を決定する。例えば、プロセッサ 202 は、2 個の最も最近のフィンガー決定を比較してもよいし、平均されたフィンガー結果を比較してもよいし、比較する何らかの特定のフィンガー結果を選択してもよい。2 個のフィンガー結果の間の差分が、第 1 のセットの基準（すなわち、同じ数、タイプ、エンドノード等）を満たす場合には、プロセッサ 202 は、時間パラメータを増加させ、これに対応して、時間間隔 (TI) を増加させる。例えば、1 つのインプリメンテーションでは、時間間隔を、下記のように、選択された最大 TI_{max} まで増加させ、このことは、アルゴリズムが、 $1 / TI_{max}$ の最低頻度において実施されることを保証する。

$$TI_{new} = TI_{old} * 2$$

【0035】

2 個のフィンガー結果の間の差分が第 2 のセットの基準を満たす場合には、プロセッサ 202 は、時間パラメータを減少させ、これに対応して、時間間隔 (TI) を減少させる。例えば、1 つのインプリメンテーションでは、時間間隔を、下記のように、選択された最低 TI_{min} までダウンして減少させ、このことは、アルゴリズムが、 $1 / TI_{min}$ の最高頻度において実施されることを保証する。

$$TI_{new} = TI_{old} / 2$$

【0036】

いったん新しい時間パラメータを決定すると、プロセッサ 202 は、時間パラメータをタイマー 206 に提供し、新しい時間間隔を測定できるようにする。時間間隔の終わりにおいて、プロセッサ 202 は、フィンガー安定化アルゴリズムを再び実行する。上記で提供した、時間間隔を増加させるおよび減少させるための技術は、1 つのインプリメンテーションに過ぎないことと、他の技術を使用してもよいことに留意すべきである。例えば、時間間隔は、上述したものよりも速い、もしくは、上述したものよりも遅いレートに、増加させてもよく、および/または、減少させてもよい。実質的には、時間間隔を増加および/または減少させるための何からの技術を使用してもよい。

【0037】

プロセッサ 202 は、第 1 および第 2 のセットの基準を規定する何らかのセットのパラメータを発生させて、AS システムの選択された性能を得てもよいことにも留意すべきである。例えば、2 個のフィンガー決定の間に差分がない場合に、または、2 個のフィンガ

10

20

30

40

50

ー決定の間に小さい差分がある場合のみに、これらの基準を満たすように、第1のセットの基準を規定してもよい。さらに、2個のフィンガー決定の間に何らかの差分がある場合に、または、2個のフィンガー決定の間に非常に多くの差分がある場合のみに、これらの基準を満たすように、第2のセットの基準を規定してもよい。したがって、何らかのフィンガーダイナミック（すなわち、変化がない、小さい変化、大きい変化、特定の変化等）を実質的に検出し、検出したフィンガーダイナミックに基づいて、フィンガー安定化アルゴリズムの実行間の時間間隔を調節するために、基準のセットは、プロセッサ202によって設定することができる。

【0038】

1つのインプリメンテーションでは、ASシステムは、1つ以上のプログラム命令（“命令”）、あるいは、コンピュータ読取可能媒体上に記憶されている、または、コンピュータ読取可能媒体上に具現化されている“コード”のセットを有する、コンピュータプログラムプロダクトを含む。少なくとも1つのプロセッサ、例えば、プロセッサ202によりコードが実行されるとき、これらの実行は、AS装置200に、ここで記述するASシステムの機能を提供させる。例えば、コンピュータ読取可能媒体は、フロッピー（登録商標）ディスク、CDROM、メモ리카ード、FLASHメモリデバイス、RAM、ROM、あるいは、AS装置200にインターフェースする他の何らかのタイプのメモリデバイスまたはコンピュータ読取可能媒体を含む。別の態様では、外部デバイスまたは通信ネットワークリソースから、AS装置200中にコードのセットをダウンロードしてもよい。コードのセットは、実行されるとき、ここで記述するような、ASシステムの態様を提供するように動作する。

10

20

【0039】

図3は、ASシステムにしたがって、フィンガー安定化アルゴリズムを適応的にスケジューリングするための例示的な方法300を示している。明確にするために、図2中に示したAS装置200を参照して、方法300を下記に記述する。1つのインプリメンテーションでは、プロセッサ202は、AS装置200を制御するための1つ以上のセットのコードを実行して、下記に記述している機能を実施する。

【0040】

ブロック302において、オーバーレイネットワークパラメータのデータベースを維持する。1つのインプリメンテーションでは、プロセッサ202は、パラメータデータベース214をメモリ204中に維持する。例えば、パラメータデータベース214は、これらには限定されないが、IPアドレス、ノード識別子、および/または、ピアツーピアオーバーレイネットワーク上で動作する1つ以上のノードに関連する他の何らかのパラメータを含む。

30

【0041】

ブロック304において、フィンガー安定化アルゴリズムを実施する。1つのインプリメンテーションでは、プロセッサ202は、安定化アルゴリズム218を実行し、トランシーバ208および通信リンク212を使用してオーバーレイネットワークの他のノードと通信して、オーバーレイネットワークのフィンガーを決定する。プロセッサ202は、オーバーレイネットワークに関係するフィンガーを決定するための何らかの適切な安定化アルゴリズムを実施する。

40

【0042】

ブロック306において、フィンガー安定化アルゴリズムの結果を、メモリ中に記憶させる。例えば、プロセッサ202は、フィンガー安定化アルゴリズムの結果を、フィンガーデータベース216の一部として、メモリ204中に記憶させる。

【0043】

ブロック308において、時間間隔を初期化する。例えば、プロセッサ202は、タイマー206が初期時間間隔を測定できるように、初期時間パラメータをタイマー206中に入力する。

【0044】

50

ブロック 3 1 0 において、方法は、現在の時間間隔が測定されるのを待つ。例えば、タイマー 2 0 6 は、現在の時間間隔を測定して、時間間隔が満期になったとき、プロセッサ 2 0 2 に表示を提供する。

【 0 0 4 5 】

ブロック 3 1 2 において、フィンガー安定化アルゴリズムを実施する。1つのインプリメンテーションでは、タイマー 2 0 6 満期に 응답して、プロセッサ 2 0 2 は、安定化アルゴリズム 2 1 8 を実行し、トランシーバ 2 0 8 および通信リンク 2 1 2 を使用してオーバーレイネットワークの他のノードと通信して、オーバーレイネットワークのフィンガーを決定する。プロセッサ 2 0 2 は、オーバーレイネットワークに関係するフィンガーを決定するための何らかの適切な安定化アルゴリズムを実施する。

10

【 0 0 4 6 】

ブロック 3 1 4 において、フィンガー安定化アルゴリズムの結果を、メモリ中に記憶させる。例えば、プロセッサ 2 0 2 は、フィンガー安定化アルゴリズムの結果を、フィンガーデータベース 2 1 6 の一部として、メモリ 2 0 4 中に記憶させる。

【 0 0 4 7 】

ブロック 3 1 6 において、2つのフィンガー決定の間の差分が第1のセットの基準を満たすか否かに関して、決定を行う。例えば、プロセッサ 2 0 2 は、フィンガーデータベース 2 1 6 から、現在のフィンガー結果および以前のフィンガー結果を取り出して、それらと比較し、それらの間の差分が第1のセットの基準を満たすか否かを決定する。例えば、第1および第2のフィンガー決定の間に差分がない場合に、または、第1および第2のフィンガー決定の間に小さい差分がある場合のみに、第1のセットの基準を満たしてもよい。第1のセットの基準を満たす場合、方法はブロック 3 1 8 に進む。第1のセットの基準を満たさない場合、方法はブロック 3 2 0 に進む。

20

【 0 0 4 8 】

ブロック 3 1 8 において、現在の時間間隔を増加させる。例えば、プロセッサ 2 0 2 は、2倍だけ時間間隔を増加させ、タイマー 2 0 6 が、更新された時間間隔を測定できるように、新たな時間パラメータをタイマー 2 0 6 中に入力する。その後、方法は、ブロック 3 1 0 に進み、タイマー 2 0 6 の満期を待つ。プロセッサ 2 0 2 は、何らかの適切な技術またはアルゴリズムを利用して、時間間隔を増加させることができることに留意すべきである。

30

【 0 0 4 9 】

ブロック 3 2 0 において、2個のフィンガー決定の間の差分が第2のセットの基準を満たすか否かに関して、決定を行う。例えば、プロセッサ 2 0 2 は、フィンガーデータベース 2 1 6 から、現在のフィンガー結果および以前のフィンガー結果を取り出して、それらと比較し、それらの間の差分が第2のセットの基準を満たすか否かを決定する。例えば、第1および第2のフィンガー決定の間に差分がある場合に、または、第1および第2のフィンガー決定の間に大きい差分がある場合のみに、第2のセットの基準を満たしてもよい。第2のセットの基準を満たす場合、方法はブロック 3 2 2 に進む。第2のセットの基準を満たさない場合、方法はブロック 3 1 0 に進み、時間間隔を変更しない。

【 0 0 5 0 】

ブロック 3 2 2 において、時間間隔を減少させる。例えば、プロセッサ 2 0 2 は、1/2に時間間隔を減少させ、タイマー 2 0 6 が、更新された時間間隔を測定できるように、新たな時間パラメータをタイマー 2 0 6 中に入力する。その後、方法は、ブロック 3 1 0 に進み、タイマー 2 0 6 の満期を待つ。プロセッサ 2 0 2 は、何らかの適切な技術またはアルゴリズムを利用して、現在の時間間隔を減少させることができることに留意すべきである。1つのインプリメンテーションでは、時間間隔は、時間間隔を増加させるレートよりも速いレートにおいて、減少させる。

40

【 0 0 5 1 】

それゆえ、方法 3 0 0 は、オーバーレイネットワークに関係するフィンガーがいつ変化したかを決定し、ASシステムにしたがって、フィンガー安定化アルゴリズムの実行間の

50

時間間隔を適応的に調節するために、ノードにおいて動作可能である。方法 300 は 1 つのインプリメンテーションに過ぎないことと、方法 300 の動作は、再構成することができ、または、そうでなければ、さまざまなインプリメンテーションの範囲内で修正できるとに留意すべきである。したがって、他のインプリメンテーションは可能性がある。

【0052】

図 4 は、AS システムにしたがって構築された例示的な AS 装置 400 を示している。例えば、AS 装置 400 は、図 2 中で示した AS 装置 200 としての使用に適している。ある態様では、ここで記述したような AS システムの態様を提供するように構成されている 1 つ以上のモジュールを備える少なくとも 1 つの集積回路により、AS 装置 400 は実現される。例えば、1 つのインプリメンテーションでは、それぞれのモジュールは、ハードウェア、および/または、ハードウェア実行ソフトウェアを備える。

10

【0053】

AS 装置 400 は、ノードに係する第 1 および第 2 のフィンガーを比較する手段 (402) を含む第 1 のモジュールを備える。この第 1 のモジュールは、ある態様ではプロセッサ 202 を含む。AS 装置 400 はまた、第 1 および第 2 のフィンガー決定の間の差分が第 1 の基準を満たす場合に、フィンガー安定化アルゴリズムの実行間の時間間隔を増加させる手段 (404) を含む第 2 のモジュールを備える。この第 2 のモジュールは、ある態様ではタイマー 206 を含む。AS 装置 400 はまた、第 1 および第 2 のフィンガー決定の間の差分が第 2 の基準を満たす場合に、フィンガー安定化アルゴリズムの実行間の時間間隔を減少させる手段 (406) を含む第 3 のモジュールを備える。この第 3 のモジュールは、ある態様ではタイマー 206 を含む。

20

【0054】

オーバーレイサイズ決定

AS システムはまた、オーバーレイネットワークサイズ推定を決定するように動作する。1 つのインプリメンテーションでは、AS 装置 200 は、ピアツーピアオーバーレイネットワークのサイズ推定のための 2 つの方法を実施するように動作する。第 1 の方法では、中心カウンタを利用し、ノードダイナミックの推論と中心カウンタを組み合わせ、オーバーレイネットワークのサイズを推定する。第 2 の方法では、ピア間のピギーバック通信とともに、1 つ以上のヒューリスティックを使用して、オーバーレイネットワークのサイズを推定する分散推定プロセスを実施する。

30

【0055】

サイズ決定 方法 1

図 5 は、AS システムにしたがって、オーバーレイネットワークサイズを決定するための第 1 の方法 500 を示している。サイズ決定の第 1 の方法 500 は、関係するデータキーを任意のデータアイテムが有する、オーバーレイネットワークの分散ハッシュテーブルのインプリメンテーションに基づいている。数値キーに対して衝突耐性のあるハッシュ関数を使用して、データキースtring をハッシングすることより、データキーを取得する。ノードはまた、同じキースペース中のノード識別子を受信する。そのノードキーが、数値距離でデータアイテムのキー (数値キー) に最も近いノードにおいて、データアイテムを記憶する。

40

【0056】

明確にするために、図 2 中に示した AS 装置 200 を参照して、方法 500 を下記に記述する。1 つのインプリメンテーションでは、プロセッサ 202 は、AS 装置 200 を制御するための 1 つ以上のセットのコードを実行して、下記に記述する機能を実施する。

【0057】

ブロック 502 において、オーバーレイサイズカウンタが維持され、オーバーレイネットワークの現在のサイズを示す。オーバーレイサイズカウンタは、オーバーレイネットワーク中に参加している任意のノードによりアクセスできる。オーバーレイサイズカウンタの説明、および、オーバーレイサイズカウンタをどのように維持し、更新するのかを下記に提供する。

50

【 0 0 5 8 】

ブロック 5 0 4 において、加わるノードが検出されるか否かに関して、決定を行う。1つのインプリメンテーションでは、オーバーレイネットワークに加わる何らかのノードは、最も近い隣接しているものに追加される。この最も近い隣接しているものは、数値キースペース中の後継点である。すなわち、この最も近い隣接しているものは、そのノードキーが、時計回り方向において、加わるノードのキーに最も近い。例えば、図 1 中に示した A S 装置 1 1 2 を備えるノード 4 は、時計回り方向において、加わるノードに最も近いと仮定する。したがって、ノード 4 は、加わるノードに対する“ジョインアナウンサー”として指定される。

【 0 0 5 9 】

ジョインアナウンサーは、ノードの到着をアナウンスすることと、オーバーレイサイズカウンタをインクリメントさせることを担当している。異なる加わるノードに対して、ジョインアナウンサーが異なってもよいことに留意すべきである。1つのインプリメンテーションでは、オーバーレイネットワークに加わりたいことを示している加わるノードからの通信を受信することにより、加わるノードが検出されるか否かに関して、プロセッサ 2 0 2 は決定を行う。加わるノードが検出された場合、方法はブロック 5 0 6 に進む。加わるノードが検出されなかった場合、方法はブロック 5 0 8 に進む。

【 0 0 6 0 】

ブロック 5 0 6 において、オーバーレイサイズカウンタをインクリメントさせるために、メッセージを送る。オーバーレイサイズカウンタは、分散ハッシュテーブル中に維持されている。以前に同意された一意的なカウンタ変数を表す“オーバーレイサイズ”のような、よく知られているストリングをハッシングすることにより取得するデータキーにおいて記憶されることになるメッセージを、ジョインアナウンサーが送るときに、オーバーレイサイズカウンタがインクリメントされる。ジョインアナウンサーは、インクリメントを規定するメッセージをこのオーバーレイサイズカウンタのキーに送る。これらのメッセージに基づいて、カウンタを記憶するノード上で、すなわち、そのノードキーが、カウンタキー、例えば、ストリング“オーバーレイサイズ”のハッシュに最も近いノード上で、オーバーレイサイズカウンタをインクリメントさせる。1つのインプリメンテーションでは、プロセッサ 2 0 2 は、トランシーバ 2 0 8 を制御して、オーバーレイサイズカウンタをインクリメントさせるメッセージを送信する。

【 0 0 6 1 】

ブロック 5 0 8 において、離れるノードが検出されるか否かに関して、決定を行う。1つのインプリメンテーションでは、オーバーレイを離れる何らかのノードは、数値キースペース中でその隣接しているものに、進んでリーブメッセージを送る。そのノードキーが、時計回り方向において、離れるノードのキーに最も近いノードは、離れるノードに対する“リーブアナウンサー”として指定される。例えば、図 1 中に示した A S 装置 1 1 2 を備えるノード 4 が、時計回り方向において、離れるノードに最も近いと仮定する。したがって、ノード 4 は、リーブアナウンサーとして指定され、オーバーレイサイズカウンタをデクリメントさせることを担当する。1つのインプリメンテーションでは、オーバーレイネットワークを離れたいことを示す離れるノードからの通信を受信することにより、離れるノードが検出されるか否かに関して、プロセッサ 2 0 2 は決定を行う。離れるノードが検出された場合、方法はブロック 5 1 0 に進む。離れるノードが検出されなかった場合、方法はブロック 5 1 2 に進む。

【 0 0 6 2 】

ブロック 5 1 0 において、オーバーレイサイズカウンタをデクリメントさせるために、メッセージを送る。オーバーレイサイズカウンタは、分散ハッシュテーブル中に維持されている。以前に同意された一意的なカウンタ変数を表す“オーバーレイサイズ”のような、よく知られているストリングをハッシングすることにより取得するデータキーにおいて記憶されることになるメッセージを、リーブアナウンサーが送るときに、オーバーレイサイズカウンタがデクリメントされる。リーブアナウンサーは、デクリメントを規定するメ

10

20

30

40

50

ッセージをこのオーバーレイサイズカウンタのキーに送る。これらのメッセージに基づいて、カウンタを記憶するノード上で、すなわち、そのノードキーが、カウンタキー、例えば、ストリング“オーバーレイサイズ”のハッシュに最も近いノード上で、オーバーレイサイズカウンタをデクリメントさせる。1つのインプリメンテーションでは、プロセッサ202は、トランシーバ208を制御して、オーバーレイサイズカウンタをデクリメントさせるメッセージを送信する。

【0063】

ブロック512において、離れるノードが推測できるか否かに関して、決定を行う。ノードは、いつもオーバーレイネットワークに進んで加わるが、オーバーレイを進んで離れはしないかもしれないので、リーブアナウンサーは、ノードがオーバーレイを離れるときを推論により決定するように動作する。1つのインプリメンテーションでは、リーブアナウンサーは、ノードがオーバーレイを離れたと推測するとき、オーバーレイサイズカウンタを更新する。例えば、何らかの特定のノードに対するリーブアナウンサーは、その特定のノードに対する接続を維持して、リングのルーティング完全性を維持する必要がある。リーブアナウンサーは、接続の失敗または他の表示により、その特定のノードが離れたことを推測し、続いて、オーバーレイサイズカウンタを更新することができる。例えば、図1中に示したAS装置112を備えるノード4は、特定のノードに対する接続を維持し、リングのルーティング完全性を維持することを仮定する。1つのインプリメンテーションでは、トランシーバ208は、特定のノードに関係する接続の失敗を検出する。したがって、ノード4は、リーブアナウンサーとして指定され、プロセッサ202は、進んだものでない離脱が起こったとしても、特定のノードがオーバーレイを離れたことを推測し、オーバーレイサイズカウンタを更新するように動作する。

10

20

【0064】

1つのインプリメンテーションでは、接続の失敗を検出することにより、または、ノードがオーバーレイネットワークを離れたと示している第三者通信を受信することにより、ノードがオーバーレイを離れたことを推測するか否かに関して、プロセッサ202は決定を行う。第三者通信は、オーバーレイネットワーク中の別のノードから発信されてもよい。離れるノードが推測された場合、方法はブロック514に進む。離れるノードが推測されなかった場合、方法はブロック516に進む。

【0065】

ブロック514において、オーバーレイサイズカウンタをデクリメントさせるために、メッセージを送る。オーバーレイサイズカウンタは、分散ハッシュテーブル中に維持されている。以前に同意された一意的なカウンタ変数を表す“オーバーレイサイズ”のような、よく知られているストリングをハッシングすることにより取得されるデータキーにおいて記憶されることになるメッセージを、リーブアナウンサーが送るときに、オーバーレイサイズカウンタがデクリメントされる。リーブアナウンサーは、デクリメントを規定するメッセージをこのオーバーレイサイズカウンタのキーに送る。これらのメッセージに基づいて、カウンタを記憶するノード上で、すなわち、そのノードキーが、カウンタキー、例えば、ストリング“オーバーレイサイズ”のハッシュに最も近いノード上で、オーバーレイサイズカウンタをデクリメントさせる。1つのインプリメンテーションでは、プロセッサ202は、トランシーバ208を制御して、オーバーレイサイズカウンタをデクリメントさせるメッセージを送信する。

30

40

【0066】

ブロック516において、現在のネットワークサイズを決定するために、オーバーレイネットワークサイズカウンタに問い合わせる。オーバーレイサイズカウンタの現在の値を取り出すために、“オーバーレイサイズ”のような、よく知られているストリングをハッシングすることによる同じカウンタキーに問い合わせることにより、ノードは、ネットワークサイズについて知ることができる。オーバーレイサイズカウンタの現在の値は、オーバーレイネットワークサイズの現在の推定である。

【0067】

50

AS装置200を備える何らかのノードにおいて、上記の動作を実施できることに留意すべきである。したがって、ジョインアナウンサーまたはリーブアナウンサーとしてアクトする何らかのノードにより、オーバーレイサイズカウンタをインクリメントまたはデクリメントさせる分散方法で、動作を実施する。それゆえ、ASシステムのインプリメンテーションは、進んだものでない離脱のケースにおいて、オーバーレイサイズカウンタを正確に更新できるように、ノードが離れていることを推測するように動作する。方法500は1つのインプリメンテーションに過ぎないことと、方法500の動作は、再構成することができ、または、そうでなければ、さまざまなインプリメンテーションの範囲内で修正できることにも留意すべきである。したがって、他のインプリメンテーションも可能性がある。

10

【0068】

サイズ決定 方法2

サイズ決定の第2の方法は、上記の方法500中のような加わることと離れることとを示す何らかの特有なメッセージを送り出すことを必要とせず、分散ハッシュテーブル中に維持されるよく知られている同意されたカウンタでもない。第2の方法は、さらなる精度のために適用できる2つの改善を含む推定技術を利用する。

【0069】

図6は、ASシステムにしたがってオーバーレイネットワークサイズを決定するための第2の方法600を示している。方法600では、Sは、何らかのセットのk個のノードにより管理されるセグメント長(数値IDスペースの長さ)の合計を示す。そして、オーバーレイサイズの推定E、すなわち、オーバーレイ中のノードの数は、 k/S から決定できる。

20

【0070】

明確にするために、図2中に示したAS装置200を参照して、方法600を下記に記述する。1つのインプリメンテーションでは、プロセッサ202は、AS装置200を制御するための1つ以上のセットのコードを実行して、下記に記述する機能を実施する。オーバーレイネットワーク中のノードにおいてAS装置が位置していることと、AS装置200は、下記の機能を実施して、オーバーレイネットワークのサイズを決定することとを仮定する。

【0071】

ブロック602において、オーバーレイネットワーク中で、ノードのセットを識別する。セットは、オーバーレイネットワーク中のノードのうちいくつかまたはすべてを含んでいてもよい。例えば、ノードは、フィンガーを、隣接するものを、または、AS装置を備えるノードと通信中であるノードを含んでいてもよい。1つのインプリメンテーションでは、プロセッサ202は、セットの一部であるノードを識別するように動作する。

30

【0072】

ブロック604において、ノードのセットに関係するセグメント長を取得する。1つのインプリメンテーションでは、これらの関係するセグメント長を取得するために、ノードのセットに問い合わせる。セグメント長は、所定のノードとその時計回りの後継点との間のIDスペースの一部である。セグメント長が減少するにつれて、ネットワーク中のノードの数がより多くなる。

40

【0073】

オーバーレイネットワーク中のすべてのノードをサンプリングして、それらの関係するセグメント長を導出することは実用的ではないかもしれないので、1つのインプリメンテーションでは、AS装置を備えるノードが、そのノードが既にルーティング関係を維持するノードに、それらのセグメント長を尋ねる。例えば、AS装置は、セット中のノードのそれぞれに対して、それらが管理するセグメント長について問い合わせることが可能である。別のインプリメンテーションでは、セグメント長情報は、AS装置を備えるノードと、そのそれぞれのフィンガーまたは隣接するものとの間の通常のメッセージのトップ上にピギーバックされている。したがって、プロセッサ202は、トランシーバ208を制御

50

して、通常通信のトップ上に情報をピギーバックさせることにより、ノードのセットから、セグメント情報のうちのいくつかまたはすべてを取得する。

【0074】

ブロック606において、それらのセグメント長の合計で割ったセット中のノードの数に基づいて、オーバーレイネットワークのサイズを決定する。例えば、IDスペースの長さは、0と1との間になるように区分されていると仮定する。オーバーレイネットワーク中に1つのノードのみがある場合には、 $E = 1 / 1 = 1$ であり、このケースにおいて正しい推定である。2つのノードと、これらが、第1のノードが0.3を所有し、第2のノードが0.7を所有するように、IDスペースを区分することを考える。そのとき、 $E = 2 / (0.3 + 0.7) = 2$ である。1つのインプリメンテーションでは、プロセッサ202は、セット中のノードの総数を、それらの関係するセグメント長の合計で割ることにより、オーバーレイネットワークの推定されるサイズを決定するように動作する。

10

【0075】

それゆえ、方法600は、ノードの識別されたセットと、それらの関係するセグメント長とに基づいて、オーバーレイネットワークのサイズを決定するように動作する。方法600は1つのインプリメンテーションに過ぎないことと、方法600の動作は、再構成することができ、または、そうでなければ、さまざまなインプリメンテーションの範囲内で修正できることとに留意すべきである。したがって、他のインプリメンテーションも可能性がある。

【0076】

20

方法2に対する改善

図7は、ASシステムにしたがって、オーバーレイネットワークサイズを決定する第1の改善方法700を示している。例えば、方法700では、1つ以上ノードからの推定を平均することにより、より正確なおよび/またはよりロバーストなオーバーレイサイズ推定を決定するように動作する。明確にするために、図2中に示されたAS装置200を参照して、方法700を下記に記述する。1つのインプリメンテーションでは、プロセッサ202は、AS装置200を制御するための1つ以上のセットのコードを実行して、下記に記述する機能を実施する。AS装置200は、オーバーレイネットワーク中のノードにおいて位置していることと、AS装置200は、下記の機能を実施して、オーバーレイネットワークのサイズを決定することとを仮定する。

30

【0077】

ブロック702において、第1のノードにおいて、サイズ推定を決定する。1つのインプリメンテーションでは、サイズ推定は、方法600により決定する。例えば、AS装置200は、第1のノードにおいて、方法600を実施して、オーバーレイネットワークのサイズ推定を決定するように動作する。

【0078】

ブロック704において、他のノードからのサイズ推定を取得する。例えば、AS装置を備える第1のノードは、他のノードからサイズ推定を取得するように動作する。1つのインプリメンテーションでは、フィンガーから、隣接するものから、および、第1のノードと通信中である他のノードから、サイズ推定を取得する。例えば、プロセッサ202は、トランシーバ208を制御して、他のノードにそれらのそれぞれのサイズ推定を問い合わせることにより、他のノードからのサイズ推定を取得するように動作する。

40

【0079】

ブロック706において、取得したサイズ推定に基づいて、オーバーレイサイズを決定する。1つのインプリメンテーションでは、第1のノードは、第1のノードがそのフィンガーから、隣接するものから、または他のノードから知ったすべてのオーバーレイサイズ推定の平均を取ることにより、そのサイズ推定をさらに改善することができる。したがって、最終サイズ推定 $E = (E_0 + E_1 + E_2 + \dots + E_n) / n$ である。ここで、 E_1 から E_n は、 n 個の他のノードから取得したサイズ推定であり、 E_0 は、第1のノードのサイズ推定である。1つのインプリメンテーションでは、プロセッサ202は、サイズ推定

50

を平均して、より正確な平均されたサイズ推定を取得するように動作する。

【 0 0 8 0 】

それゆえ、方法 7 0 0 は、より正確な平均されたサイズ推定を取得するために、1つ以上のノードからのサイズ推定を平均することにより、第 1 の方法 6 0 0 を改善するように動作する。方法 7 0 0 は 1 つのインプリメンテーションに過ぎないことと、方法 7 0 0 の動作は、再構成することができ、または、そうでなければ、さまざまなインプリメンテーションの範囲内で修正できることに留意すべきである。したがって、他のインプリメンテーションも可能性がある。

【 0 0 8 1 】

図 8 は、AS システムにしたがって、オーバーレイネットワークサイズを決定するための第 2 の改善方法 8 0 0 を示している。例えば、方法 8 0 0 は、オーバーレイネットワーク中の何らかのノードからの推定を平均することにより、より正確なおよび/またはロバーストなオーバーレイサイズ推定を決定するように動作する。明確にするために、図 2 中に示した AS 装置 2 0 0 を参照して、方法 8 0 0 を下記に記述する。1 つのインプリメンテーションでは、プロセッサ 2 0 2 は、AS 装置 2 0 0 を制御するための 1 つ以上のセットのコードを実行して、下記に記述する機能を実施する。AS 装置 2 0 0 は、オーバーレイネットワーク中の第 1 のノードにおいて位置していることと、AS 装置 2 0 0 は、下記の機能を実施して、オーバーレイネットワークのサイズを決定することとを仮定する。

【 0 0 8 2 】

ブロック 8 0 2 において、第 1 のノードにおいて、サイズ推定を決定する。1 つのインプリメンテーションでは、サイズ推定は、方法 6 0 0 により決定する。例えば、AS 装置 2 0 0 は、第 1 のノードにおいて、方法 6 0 0 を実施して、オーバーレイネットワークのサイズ推定を決定するように動作する。

【 0 0 8 3 】

ブロック 8 0 4 において、他のノードからのサイズ推定を取得する。例えば、上記の方法 7 0 0 を使用して、他のノードからサイズ推定のサンプリングが ID スペース中で均でない可能性がある。これに取り組むために、何らかのノードとの何らかのケット交換上にサイズ推定情報をピギーバックさせる。これは、そのノードから個々のサイズ推定が取得されるノードの数を増加させることから、ネットワークのサイズ推定のよりロバーストなビューを提供する。ノードの隣接するもののみからのこれらの推定を取得するよりむしろ、この改善は、特定のノードを通してケットをルーティングする何らかのノードからこの情報を得ることを可能にし、または、ルーティングから離れて特定のノードとのトランザクションを行うことを可能にする。したがって、第 1 のノードは、フィンガーから、隣接しているものから、および、第 1 のノードと対話する他の何らかのノードからサイズ推定を取得することができる。1 つのインプリメンテーションでは、プロセッサ 2 0 2 は、トランシーバ 2 0 8 を制御して、第 1 のノードとの何らかのケット交換上にピギーバックされたサイズ推定を取得するように動作する。

【 0 0 8 4 】

ブロック 8 0 6 において、取得したサイズ推定に基づいて、オーバーレイサイズを決定する。1 つのインプリメンテーションでは、第 1 のノードは、そのフィンガーから、隣接するものから、または、第 1 のノードと通信中である他の何らかのノードから第 1 のノードが知ったすべてのオーバーレイサイズ推定の平均を取ることににより、そのサイズ推定をさらに改善できる。したがって、最終サイズ推定 $E = (E_0 + E_1 + E_2 + \dots + E_n) / n$ である。ここで、 E_1 から E_n は、 n 個の他のノードから取得したサイズ推定であり、 E_0 は、第 1 のノードのサイズ推定である。1 つのインプリメンテーションでは、プロセッサ 2 0 2 は、サイズ推定を平均して、より正確なサイズ推定を取得するように動作する。

【 0 0 8 5 】

それゆえ、方法 8 0 0 は、何らかのケット送信上にピギーバックされていて取得したサイズ推定を提供して、よりロバーストな平均を提供することにより、第 1 の方法 6 0 0

10

20

30

40

50

を改善するように動作する。方法 800 は 1 つのインプリメンテーションに過ぎないことと、方法 800 の動作は、再構成することができ、または、そうでなければ、さまざまなインプリメンテーションの範囲内で修正できることに留意すべきである。したがって、他のインプリメンテーションも可能性がある。

【0086】

図 9 は、AS システムにしたがって構築された例示的な AS 装置 900 を示している。例えば、AS 装置 900 は、図 2 中に示した AS 装置 200 としての使用に適している。ある態様では、ここで記述したような AS システムの態様を提供するように構成されている 1 つ以上のモジュールを備える少なくとも 1 つの集積回路により、AS 装置 900 を実現する。例えば、1 つのインプリメンテーションでは、それぞれのモジュールは、ハードウェア、および/または、ハードウェア実行ソフトウェアを備える。

10

【0087】

AS 装置 900 は、第 1 のノードがオーバーレイネットワークを離れていることを推測する手段 (902) を含む第 1 のモジュールを備える。この第 1 のノードは、ある態様ではプロセッサ 202 を含む。AS 装置 900 はまた、サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信する手段 (904) を含む第 2 のモジュールを備える。この第 2 のモジュールは、ある態様ではトランシーバ 208 を含む。

【0088】

図 10 は、AS システムにしたがって構築された例示的な AS 装置 1000 を示している。例えば、AS 装置 1000 は、図 2 中で示した AS 装置 200 としての使用に適している。ある態様では、ここで記述したような AS システムの態様を提供するように構成されている 1 つ以上のモジュールを備える少なくとも 1 つの集積回路により、AS 装置 1000 を実現する。例えば、1 つのインプリメンテーションでは、それぞれのモジュールは、ハードウェア、および/または、ハードウェア実行ソフトウェアを備える。

20

【0089】

AS 装置 1000 は、オーバーレイの第 1 のノードに関するノードのセットを識別する手段 (1002) を含む第 1 のモジュールを備える。この第 1 のモジュールは、ある態様ではプロセッサ 202 を含む。AS 装置 1000 はまた、ノードのセットのそれぞれのノードに関するセグメント長を取得する手段 (1004) を含む第 2 のモジュールを備える。この第 2 のモジュールは、ある態様ではトランシーバ 208 を含む。AS 装置 1000 はまた、ノードのセット中のノードの総数を、セグメント長の合計で割ることにより、オーバーレイネットワークのサイズを決定する手段 (1006) を含む第 3 のモジュールを備える。この第 3 のモジュールは、ある態様ではプロセッサ 202 を含む。

30

【0090】

図 11 は、AS システムにしたがって構築された例示的な AS 装置 1100 を示している。例えば、AS 装置 1100 は、図 2 中で示した AS 装置 200 としての使用に適している。ある態様では、ここで記述したような AS システムの態様を提供するように構成されている 1 つ以上のモジュールを備える少なくとも 1 つの集積回路により、AS 装置 1100 を実現する。例えば、1 つのインプリメンテーションでは、それぞれのモジュールは、ハードウェア、および/または、ハードウェア実行ソフトウェアを備える。

40

【0091】

AS 装置 1100 は、オーバーレイネットワークの第 1 のノードに関するノードのセットを識別する手段 (1102) を含む第 1 のモジュールを備える。この第 1 のモジュールは、ある態様ではプロセッサ 202 を含む。AS 装置 1100 はまた、第 1 のノードに関するサイズ推定と、ノードのセットのそれぞれのノードに関するサイズ推定とを取得する手段 (1104) を含む第 2 のモジュールを備える。この第 2 のモジュールは、ある態様ではトランシーバ 208 を含む。AS 装置 1100 はまた、サイズ推定を平均することにより、オーバーレイネットワークのサイズを決定する手段 (1106) を含む第 3 のモジュールを備える。この第 3 のモジュールは、ある態様ではプロセッサ 202 を含む。

50

【 0 0 9 2 】

ここに開示した態様と関連して説明したさまざまな例示的な論理、論理ブロック、モジュールおよび回路は、汎用プロセッサ、デジタル信号プロセッサ（DSP）、特定用途向け集積回路（ASIC）、現場プログラム可能ゲートアレイ（FPGA）または他のプログラム可能論理デバイス、ディスクリートゲートまたはトランジスタ論理、ディスクリートハードウェアコンポーネント、あるいは、ここに記述した機能を実行するように設計されているこれらの任意の組み合わせたものにより、実現または実行してもよい。汎用プロセッサはマイクロプロセッサであってもよいが、代替実施形態では、プロセッサは、任意の従来のプロセッサ、制御装置、マイクロ制御装置、または状態機械であってもよい。プロセッサはまた、例えば、DSPおよびマイクロプロセッサを組み合わせたものや、複数のマイクロプロセッサや、DSPコアを伴う1つ以上のマイクロプロセッサや、任意のその他のこのようなコンフィギュレーションのような、コンピューティングデバイスを組み合わせたものとして実現してもよい。

10

【 0 0 9 3 】

ここに開示した態様と関連して記述した方法またはアルゴリズムのステップは、ハードウェアで、プロセッサにより実行されるソフトウェアモジュールで、あるいは、双方を組み合わせたもので直接的に具現化してもよい。ソフトウェアモジュールは、RAMメモリ、フラッシュメモリ、ROMメモリ、EPROMメモリ、EEPROMメモリ、レジスタ、ハードディスク、リムーバブルディスク、CD-ROM、または、技術的に知られている他の何らかの形態の記憶媒体に存在していてもよい。例示的な記憶媒体は、プロセッサが記憶媒体から情報を読み取り、記憶媒体に情報を書き込むことができるようにプロセッサに結合されていてもよい。代替実施形態では、記憶媒体は、プロセッサと一体化されていてもよい。プロセッサおよび記憶媒体は、ASICに存在していてもよい。ASICは、ワイヤレス通信デバイスに存在していてもよい。代替実施形態では、プロセッサおよび記憶媒体は、ディスクリートコンポーネントとしてワイヤレス通信デバイスに存在していてもよい。

20

【 0 0 9 4 】

開示した態様の記述は、当業者が本発明を製作または使用できるように提供した。これらの態様に対するさまざま改良は当業者に容易に明らかになり、ここに規定した一般的な原理は、例えば、インスタントメッセージングサービス中、または、任意の一般的なワイヤレスデータ通信アプリケーション中で、本開示の精神または範囲から逸脱することなく、他の態様に適用してもよい。したがって、本開示は、ここで示した態様に限定されることを意図しているものではなく、ここで開示した原理および新しい特徴と一致した最も広い範囲に一致させるべきである。“例示的な”という用語は、“例として、インスタンスとして、あるいは例証として機能すること”を意味するように、ここでは排他的に使用している。“例示的な”としてここで記述した何らかの態様は、必ずしも、他の態様より好ましい、または、効果的であるとして解釈すべきではない。

30

【 0 0 9 5 】

したがって、適応スケジューリングシステムの態様をここに図示し、記述してきたが、それらの精神または不可欠な特性から逸脱することなく、態様に対してさまざまな変更を行うことができることを正しく認識するだろう。それゆえ、ここでの開示および記述は、これには限定されないが、下記の特許請求の範囲において述べる本発明の範囲を例示するものであることを意図している。

40

【 図 1 】

図 1

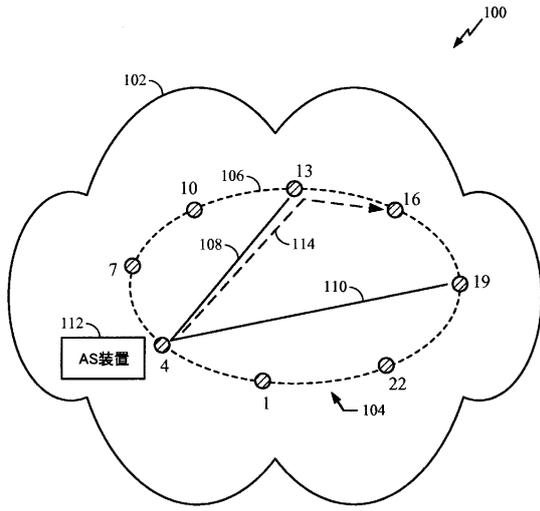


FIG. 1

【 図 2 】

図 2

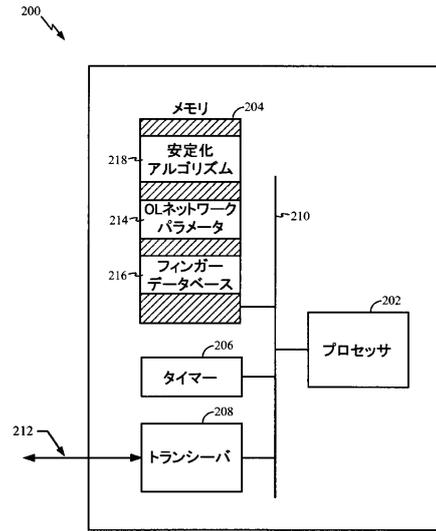


FIG. 2

【 図 3 】

図 3

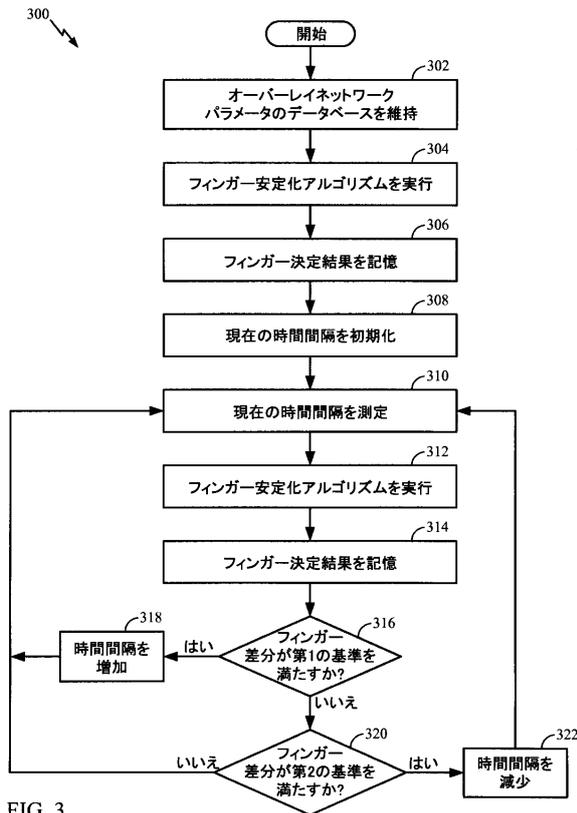


FIG. 3

【 図 4 】

図 4

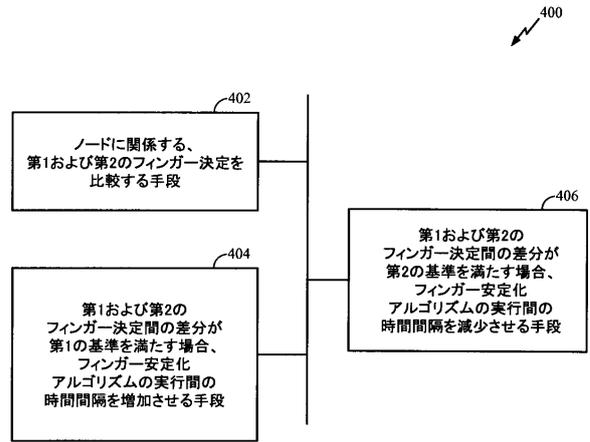


FIG. 4

【 図 5 】

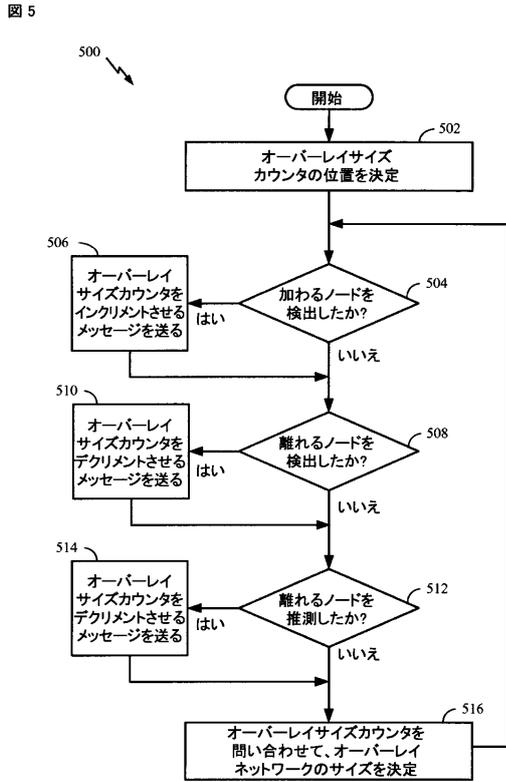


FIG. 5

【 図 6 】

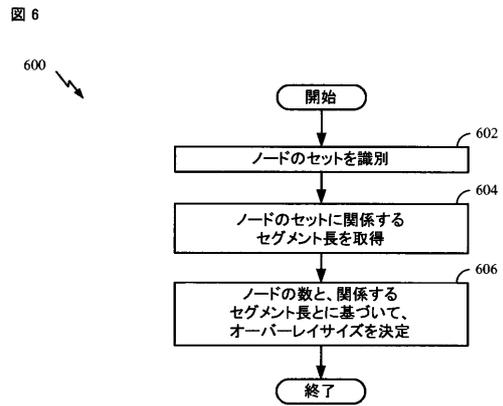


FIG. 6

【 図 7 】

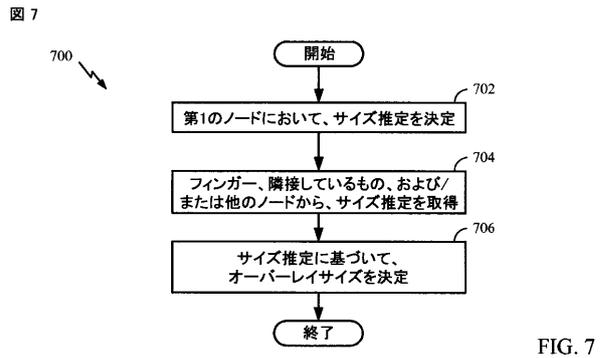


FIG. 7

【 図 8 】

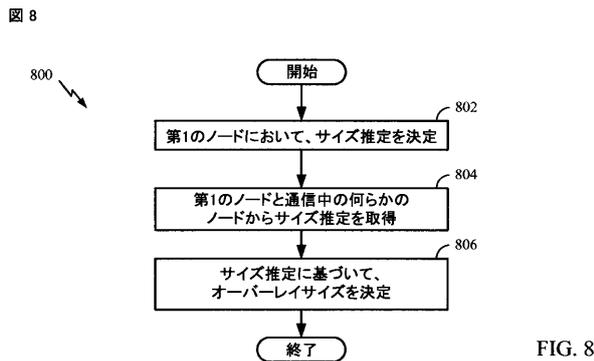


FIG. 8

【 図 10 】

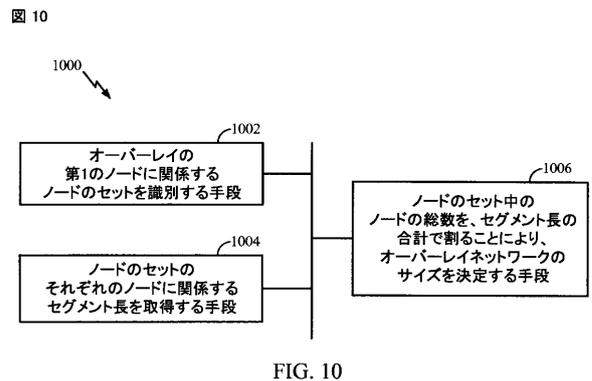


FIG. 10

【 図 9 】

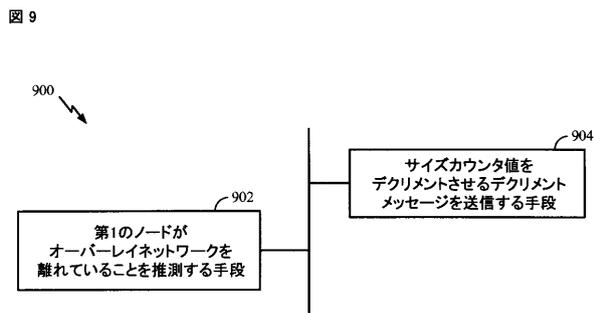


FIG. 9

【 図 11 】

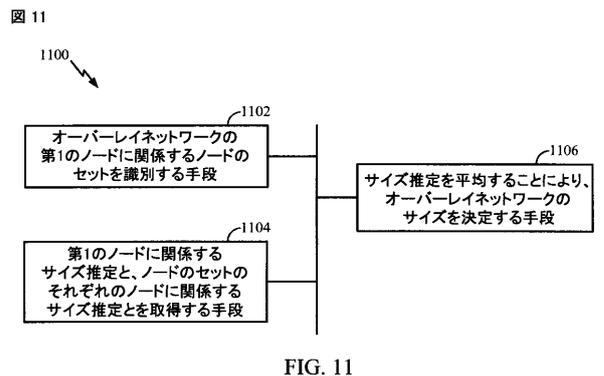


FIG. 11

【手続補正書】

【提出日】平成23年12月22日(2011.12.22)

【手続補正1】

【補正対象書類名】特許請求の範囲

【補正対象項目名】全文

【補正方法】変更

【補正の内容】

【特許請求の範囲】**【請求項1】**

ピアツーピアオーバーレイネットワークのサイズを決定するための方法において、
第1のノードが前記オーバーレイネットワークを離れていることを推測することと、
サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信することを含む方法。

【請求項2】

前記推測することは、前記第1のノードとの通信が失敗したことを決定することを含む請求項1記載の方法。

【請求項3】

前記推測することは、前記第1のノードが前記オーバーレイネットワークを離れているという表示を第三者から受信することを含む請求項1記載の方法。

【請求項4】

第2のノードが前記オーバーレイネットワークに加わっていることを決定することと、
前記サイズカウンタ値をインクリメントさせるインクリメントメッセージを送信することとをさらに含む請求項1記載の方法。

【請求項5】

第3のノードが前記オーバーレイネットワークを離れていることを決定することと、
前記サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信することとをさらに含む請求項1記載の方法。

【請求項6】

前記サイズカウンタ値を問い合わせ、前記オーバーレイネットワークのサイズを決定することをさらに含む請求項1記載の方法。

【請求項7】

前記サイズカウンタ値を問い合わせることは、よく知られているストリングをハッシュングして、前記サイズカウンタ値を導出することを含む請求項6記載の方法。

【請求項8】

ピアツーピアオーバーレイネットワークのサイズを決定するための装置において、
第1のノードが前記オーバーレイネットワークを離れていることを推測するように構成されているプロセッサと、
前記プロセッサに結合されており、サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信するように構成されている送信機とを具備する装置。

【請求項9】

前記プロセッサは、前記第1のノードとの通信が失敗したことを決定することにより、推測するように構成されている請求項8記載の装置。

【請求項10】

前記プロセッサは、前記第1のノードが前記オーバーレイネットワークを離れているという表示を第三者から受信することにより、推測するように構成されている請求項8記載の装置。

【請求項11】

前記プロセッサは、第2のノードが前記オーバーレイネットワークに加わっていることを決定するように構成され、
前記送信機は、前記サイズカウンタ値をインクリメントさせるインクリメントメッセー

ジを送信するように構成されている請求項 8 記載の装置。

【請求項 1 2】

前記プロセッサは、第 3 のノードが前記オーバーレイネットワークを離れていることを決定するように構成され、

前記送信機は、前記サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信するように構成されている請求項 8 記載の装置。

【請求項 1 3】

前記プロセッサは、前記サイズカウンタ値を問い合わせ、前記オーバーレイネットワークのサイズを決定するように構成されている請求項 8 記載の装置。

【請求項 1 4】

前記プロセッサは、よく知られているストリングをハッシングして、前記サイズカウンタ値を導出することにより、前記サイズカウンタ値を問い合わせるように構成されている請求項 1 3 記載の装置。

【請求項 1 5】

ピアツーピアオーバーレイネットワークのサイズを決定するための装置において、第 1 のノードが前記オーバーレイネットワークを離れていることを推測する手段と、サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信する手段とを具備する装置。

【請求項 1 6】

前記推測する手段は、前記第 1 のノードとの通信が失敗したことを決定する手段を備える請求項 1 5 記載の装置。

【請求項 1 7】

前記推測する手段は、前記第 1 のノードが前記オーバーレイネットワークを離れているという表示を第三者から受信する手段を備える請求項 1 5 記載の装置。

【請求項 1 8】

第 2 のノードが前記オーバーレイネットワークに加わっていることを決定する手段と、前記サイズカウンタ値をインクリメントさせるインクリメントメッセージを送信する手段とをさらに具備する請求項 1 5 記載の装置。

【請求項 1 9】

第 3 のノードが前記オーバーレイネットワークを離れていることを決定する手段と、前記サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信する手段とをさらに具備する請求項 1 5 記載の装置。

【請求項 2 0】

前記サイズカウンタ値を問い合わせ、前記オーバーレイネットワークのサイズを決定する手段をさらに具備する請求項 1 5 記載の装置。

【請求項 2 1】

前記サイズカウンタ値を問い合わせる手段は、よく知られているストリングをハッシングして、前記サイズカウンタ値を導出する手段を備える請求項 2 0 記載の装置。

【請求項 2 2】

ピアツーピアオーバーレイネットワークのサイズを決定するためのコンピュータ読取可能記憶媒体において、

第 1 のノードが前記オーバーレイネットワークを離れていることを推測するための、プロセッサにより実行可能なコードと、

サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信するための、前記プロセッサにより実行可能なコードとを含むコンピュータ読取可能記憶媒体。

【請求項 2 3】

前記コードは、前記プロセッサに、前記第 1 のノードとの通信が失敗したことを決定させることにより推測させるように構成されている請求項 2 2 記載のコンピュータ読取可能記憶媒体。

【請求項 2 4】

前記コードは、前記プロセッサに、前記第 1 のノードが前記オーバーレイネットワークを離れているという表示を第三者から受信させることにより推測させるように構成されている請求項 2 2 記載のコンピュータ読取可能記憶媒体。

【請求項 2 5】

前記コードは、

前記プロセッサに、第 2 のノードが前記オーバーレイネットワークに加わっていることを決定させるように構成され、

前記プロセッサに、前記サイズカウンタ値をインクリメントさせるインクリメントメッセージを送信させるように構成されている請求項 2 2 記載のコンピュータ読取可能記憶媒体。

【請求項 2 6】

前記コードは、

前記プロセッサに、第 3 のノードが前記オーバーレイネットワークを離れていることを決定させるように構成され、

前記プロセッサに、前記サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信させるように構成されている請求項 2 2 記載のコンピュータ読取可能記憶媒体。

【請求項 2 7】

前記コードは、前記プロセッサに、前記サイズカウンタ値を問い合わせ、前記オーバーレイネットワークのサイズを決定させるように構成されている請求項 2 2 記載のコンピュータ読取可能記憶媒体。

【請求項 2 8】

前記コードは、前記プロセッサに、よく知られているストリングをハッシングさせて、前記サイズカウンタ値を導出させることにより、前記サイズカウンタ値を問い合わせさせるように構成されている請求項 2 7 記載のコンピュータ読取可能記憶媒体。

【請求項 2 9】

ピアツーピアオーバーレイネットワークのサイズを決定するための方法において、オーバーレイネットワークの第 1 のノードに関するノードのセットを識別することと、

前記ノードのセットのそれぞれのノードに関するセグメント長を取得することと、前記ノードのセット中のノードの総数を、前記セグメント長の合計で割ることにより、前記オーバーレイネットワークのサイズを決定することを含む方法。

【請求項 3 0】

前記ノードのセットは、前記第 1 のノードに関するフィンガーを含む請求項 2 9 記載の方法。

【請求項 3 1】

前記ノードのセットは、前記第 1 のノードに関する隣接ノードを含む請求項 2 9 記載の方法。

【請求項 3 2】

前記ノードのセットは、前記第 1 のノードと通信中の 1 つ以上のノードを含む請求項 2 9 記載の方法。

【請求項 3 3】

前記取得することは、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせ、前記セグメント長の少なくとも一部分を取得することを含む請求項 2 9 記載の方法。

【請求項 3 4】

前記取得することは、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記セグメント長の少なくとも一部分を取得することを含む請求項 2 9 記載の方法。

【請求項 3 5】

ピアツーピアオーバーレイネットワークのサイズを決定するための装置において、オーバーレイネットワークの第 1 のノードに関するノードのセットを識別するように

構成されているプロセッサと、

前記プロセッサに結合されており、前記ノードのセットのそれぞれのノードに関するセグメント長を取得するように構成されているトランシーバとを具備し、

前記プロセッサは、前記ノードのセット中のノードの総数を、前記セグメント長の合計で割ることにより、前記オーバーレイネットワークのサイズを決定するようにさらに構成されている装置。

【請求項 36】

前記ノードのセットは、前記第 1 のノードに関するフィンガーを含む請求項 35 記載の装置。

【請求項 37】

前記ノードのセットは、前記第 1 のノードに関する隣接ノードを含む請求項 35 記載の装置。

【請求項 38】

前記ノードのセットは、前記第 1 のノードと通信中の 1 つ以上のノードを含む請求項 35 記載の装置。

【請求項 39】

前記トランシーバは、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせ、前記セグメント長の少なくとも一部分を取得するように構成されている請求項 35 記載の装置。

【請求項 40】

前記トランシーバは、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記セグメント長の少なくとも一部分を取得するように構成されている請求項 35 記載の装置。

【請求項 41】

ピアツーピアオーバーレイネットワークのサイズを決定するための装置において、オーバーレイネットワークの第 1 のノードに関するノードのセットを識別する手段と

、
前記ノードのセットのそれぞれのノードに関するセグメント長を取得する手段と、
前記ノードのセット中のノードの総数を、前記セグメント長の合計で割ることにより、前記オーバーレイネットワークのサイズを決定する手段とを具備する装置。

【請求項 42】

前記ノードのセットは、前記第 1 のノードに関するフィンガーを含む請求項 41 記載の装置。

【請求項 43】

前記ノードのセットは、前記第 1 のノードに関する隣接ノードを含む請求項 41 記載の装置。

【請求項 44】

前記ノードのセットは、前記第 1 のノードと通信中の 1 つ以上のノードを含む請求項 41 記載の装置。

【請求項 45】

前記取得する手段は、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせ、前記セグメント長の少なくとも一部分を取得する手段を備える請求項 41 記載の装置。

【請求項 46】

前記取得する手段は、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記セグメント長の少なくとも一部分を取得する手段を備える請求項 41 記載の装置。

【請求項 47】

ピアツーピアオーバーレイネットワークのサイズを決定するためのコンピュータ読取可能記憶媒体において、

オーバーレイネットワークの第1のノードに係するノードのセットを識別するためのプロセッサにより実行可能なコードと、

前記ノードのセットのそれぞれのノードに係するセグメント長を取得するための前記プロセッサにより実行可能なコードと、

前記ノードのセット中のノードの総数を、前記セグメント長の合計で割ることにより、前記オーバーレイネットワークのサイズを決定するための、前記プロセッサにより実行可能なコードとを含むコンピュータ読取可能記憶媒体。

【請求項48】

前記ノードのセットは、前記第1のノードに係するフィンガーを含む請求項47記載のコンピュータ読取可能記憶媒体。

【請求項49】

前記ノードのセットは、前記第1のノードに係する隣接ノードを含む請求項47記載のコンピュータ読取可能記憶媒体。

【請求項50】

前記ノードのセットは、前記第1のノードと通信中の1つ以上のノードを含む請求項47記載のコンピュータ読取可能記憶媒体。

【請求項51】

前記コードは、前記プロセッサに、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせさせて、前記セグメント長の少なくとも一部分を取得させるように構成されている請求項47記載のコンピュータ読取可能記憶媒体。

【請求項52】

前記コードは、前記プロセッサに、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記セグメント長の少なくとも一部分を取得させるように構成されている請求項47記載のコンピュータ読取可能記憶媒体。

【請求項53】

ピアツーピアオーバーレイネットワークのサイズを決定するための方法において、オーバーレイネットワークの第1のノードに係するノードのセットを識別することと、

前記第1のノードに係するサイズ推定と、前記ノードのセットのそれぞれのノードに係するサイズ推定とを取得することと、

前記サイズ推定を平均することにより、前記オーバーレイネットワークのサイズを決定することを含む方法。

【請求項54】

前記ノードのセットは、前記第1のノードに係するフィンガーを含む請求項53記載の方法。

【請求項55】

前記ノードのセットは、前記第1のノードに係する隣接ノードを含む請求項53記載の方法。

【請求項56】

前記ノードのセットは、前記第1のノードと通信中の1つ以上のノードを含む請求項53記載の方法。

【請求項57】

前記取得することは、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせ、前記サイズ推定の少なくとも一部分を取得することを含む請求項53記載の方法。

【請求項58】

前記取得することは、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記サイズ推定の少なくとも一部分を取得することを含む請求項53記載の方法。

【請求項59】

ピアツーピアオーバーレイネットワークのサイズを決定するための装置において、

オーバーレイネットワークの第1のノードに係するノードのセットを識別し、前記第1のノードに係するサイズ推定を取得するように構成されているプロセッサと、前記プロセッサに結合されており、前記ノードのセットのそれぞれのノードに係するサイズ推定を取得するように構成されているトランシーバとを具備し、前記プロセッサは、前記サイズ推定を平均することにより、前記オーバーレイネットワークのサイズを決定するようにさらに構成されている装置。

【請求項60】

前記ノードのセットは、前記第1のノードに係するフィンガーを含む請求項59記載の装置。

【請求項61】

前記ノードのセットは、前記第1のノードに係する隣接ノードを含む請求項59記載の装置。

【請求項62】

前記ノードのセットは、前記第1のノードと通信中の1つ以上のノードを含む請求項59記載の装置。

【請求項63】

前記トランシーバは、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせ、前記サイズ推定の少なくとも一部分を取得するように構成されている請求項59記載の装置。

【請求項64】

前記トランシーバは、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記サイズ推定の少なくとも一部分を取得するように構成されている請求項59記載の装置。

【請求項65】

ピアツーピアオーバーレイネットワークのサイズを決定するための装置において、
オーバーレイネットワークの第1のノードに係するノードのセットを識別する手段と

、
前記第1のノードに係するサイズ推定と、前記ノードのセットのそれぞれのノードに係するサイズ推定とを取得する手段と、

前記サイズ推定を平均することにより、前記オーバーレイネットワークのサイズを決定する手段とを具備する装置。

【請求項66】

前記ノードのセットは、前記第1のノードに係するフィンガーを含む請求項65記載の装置。

【請求項67】

前記ノードのセットは、前記第1のノードに係する隣接ノードを含む請求項65記載の装置。

【請求項68】

前記ノードのセットは、前記第1のノードと通信中の1つ以上のノードを含む請求項65記載の装置。

【請求項69】

前記取得する手段は、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせ、前記サイズ推定の少なくとも一部分を取得する手段を備える請求項65記載の装置。

【請求項70】

前記取得する手段は、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記サイズ推定の少なくとも一部分を取得する手段を備える請求項65記載の装置。

【請求項71】

ピアツーピアオーバーレイネットワークのサイズを決定するためのコンピュータ読取可能記憶媒体において、

オーバーレイネットワークの第 1 のノードに係するノードのセットを識別するための、プロセッサにより実行可能なコードと、

前記第 1 のノードに係するサイズ推定を取得するための、前記プロセッサにより実行可能なコードと、

前記ノードのセットのそれぞれのノードに係するサイズ推定を取得するための、前記プロセッサにより実行可能なコードと、

前記サイズ推定を平均することにより、前記オーバーレイネットワークのサイズを決定するための、前記プロセッサにより実行可能なコードとを含むコンピュータ読取可能記憶媒体。

【請求項 7 2】

前記ノードのセットは、前記第 1 のノードに係するフィンガーを含む請求項 7 1 記載のコンピュータ読取可能記憶媒体。

【請求項 7 3】

前記ノードのセットは、前記第 1 のノードに係する隣接ノードを含む請求項 7 1 記載のコンピュータ読取可能記憶媒体。

【請求項 7 4】

前記ノードのセットは、前記第 1 のノードと通信中の 1 つ以上のノードを含む請求項 7 1 記載のコンピュータ読取可能記憶媒体。

【請求項 7 5】

前記コードは、前記プロセッサに、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせさせて、前記サイズ推定の少なくとも一部分を取得させるように構成されている請求項 7 1 記載のコンピュータ読取可能記憶媒体。

【請求項 7 6】

前記コードは、前記プロセッサに、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記サイズ推定の少なくとも一部分を取得させるように構成されている請求項 7 1 記載のコンピュータ読取可能記憶媒体。

【手続補正 2】

【補正対象書類名】明細書

【補正対象項目名】0095

【補正方法】変更

【補正の内容】

【0095】

したがって、適応スケジューリングシステムの態様をここに図示し、記述してきたが、それらの精神または不可欠な特性から逸脱することなく、態様に対してさまざまな変更を行うことができることを正しく認識するだろう。それゆえ、ここでの開示および記述は、これには限定されないが、下記の特許請求の範囲において述べる本発明の範囲を例示するものであることを意図している。

以下に、本願出願時の特許請求の範囲に記載された発明を付記する。

[1] ピアツーピアオーバーレイネットワークのサイズを決定するための方法において

第 1 のノードが前記オーバーレイネットワークを離れていることを推測することと、
サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信することを含む方法。

[2] 前記推測することは、前記第 1 のノードとの通信が失敗したことを決定することを含む [1] 記載の方法。

[3] 前記推測することは、前記第 1 のノードが前記オーバーレイネットワークを離れているという表示を第三者から受信することを含む [1] 記載の方法。

[4] 第 2 のノードが前記オーバーレイネットワークに加わっていることを決定することと、

前記サイズカウンタ値をインクリメントさせるインクリメントメッセージを送信するこ

ととをさらに含む [1] 記載の方法。

[5] 第 3 のノードが前記オーバーレイネットワークを離れていることを決定することと、

前記サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信することとをさらに含む [1] 記載の方法。

[6] 前記サイズカウンタ値を問い合わせ、前記オーバーレイネットワークのサイズを決定することをさらに含む [1] 記載の方法。

[7] 前記サイズカウンタ値を問い合わせることは、よく知られているストリングをハッシュして、前記サイズカウンタ値を導出することを含む [6] 記載の方法。

[8] ピアツーピアオーバーレイネットワークのサイズを決定するための装置において、

第 1 のノードが前記オーバーレイネットワークを離れていることを推測するように構成されているプロセッサと、

前記プロセッサに結合されており、サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信するように構成されている送信機とを具備する装置。

[9] 前記プロセッサは、前記第 1 のノードとの通信が失敗したことを決定することにより、推測するように構成されている [8] 記載の装置。

[10] 前記プロセッサは、前記第 1 のノードが前記オーバーレイネットワークを離れているという表示を第三者から受信することにより、推測するように構成されている [8] 記載の装置。

[11] 前記プロセッサは、第 2 のノードが前記オーバーレイネットワークに加わっていることを決定するように構成され、

前記送信機は、前記サイズカウンタ値をインクリメントさせるインクリメントメッセージを送信するように構成されている [8] 記載の装置。

[12] 前記プロセッサは、第 3 のノードが前記オーバーレイネットワークを離れていることを決定するように構成され、

前記送信機は、前記サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信するように構成されている [8] 記載の装置。

[13] 前記プロセッサは、前記サイズカウンタ値を問い合わせ、前記オーバーレイネットワークのサイズを決定するように構成されている [8] 記載の装置。

[14] 前記プロセッサは、よく知られているストリングをハッシュして、前記サイズカウンタ値を導出することにより、前記サイズカウンタ値を問い合わせるように構成されている [13] 記載の装置。

[15] ピアツーピアオーバーレイネットワークのサイズを決定するための装置において、

第 1 のノードが前記オーバーレイネットワークを離れていることを推測する手段と、サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信する手段とを具備する装置。

[16] 前記推測する手段は、前記第 1 のノードとの通信が失敗したことを決定する手段を備える [15] 記載の装置。

[17] 前記推測する手段は、前記第 1 のノードが前記オーバーレイネットワークを離れているという表示を第三者から受信する手段を備える [15] 記載の装置。

[18] 第 2 のノードが前記オーバーレイネットワークに加わっていることを決定する手段と、

前記サイズカウンタ値をインクリメントさせるインクリメントメッセージを送信する手段とをさらに具備する [15] 記載の装置。

[19] 第 3 のノードが前記オーバーレイネットワークを離れていることを決定する手段と、

前記サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信する手段とをさらに具備する [15] 記載の装置。

[2 0] 前記サイズカウンタ値を問い合わせ、前記オーバーレイネットワークのサイズを決定する手段をさらに具備する [1 5] 記載の装置。

[2 1] 前記サイズカウンタ値を問い合わせる手段は、よく知られているストリングをハッシングして、前記サイズカウンタ値を導出する手段を備える [2 0] 記載の装置。

[2 2] ピアツーピアオーバーレイネットワークのサイズを決定するためのコンピュータプログラムプロダクトにおいて、

第 1 のノードが前記オーバーレイネットワークを離れていることを推測するための、プロセッサにより実行可能なコードと、

サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信するための、前記プロセッサにより実行可能なコードとを具現化するコンピュータ読取可能媒体を含むコンピュータプログラムプロダクト。

[2 3] 前記コードは、前記プロセッサに、前記第 1 のノードとの通信が失敗したことを決定させることにより推測させるように構成されている [2 2] 記載のコンピュータ読取可能媒体。

[2 4] 前記コードは、前記プロセッサに、前記第 1 のノードが前記オーバーレイネットワークを離れているという表示を第三者から受信させることにより推測させるように構成されている [2 2] 記載のコンピュータ読取可能媒体。

[2 5] 前記コードは、

前記プロセッサに、第 2 のノードが前記オーバーレイネットワークに加わっていることを決定させるように構成され、

前記プロセッサに、前記サイズカウンタ値をインクリメントさせるインクリメントメッセージを送信させるように構成されている [2 2] 記載のコンピュータ読取可能媒体。

[2 6] 前記コードは、

前記プロセッサに、第 3 のノードが前記オーバーレイネットワークを離れていることを決定させるように構成され、

前記プロセッサに、前記サイズカウンタ値をデクリメントさせるデクリメントメッセージを送信させるように構成されている [2 2] 記載のコンピュータ読取可能媒体。

[2 7] 前記コードは、前記プロセッサに、前記サイズカウンタ値を問い合わせ、前記オーバーレイネットワークのサイズを決定させるように構成されている [2 2] 記載のコンピュータ読取可能媒体。

[2 8] 前記コードは、前記プロセッサに、よく知られているストリングをハッシングさせて、前記サイズカウンタ値を導出させることにより、前記サイズカウンタ値を問い合わせさせるように構成されている [2 7] 記載のコンピュータ読取可能媒体。

[2 9] ピアツーピアオーバーレイネットワークのサイズを決定するための方法において、

オーバーレイネットワークの第 1 のノードに関するノードのセットを識別することと

、

前記ノードのセットのそれぞれのノードに関するセグメント長を取得することと、

前記ノードのセット中のノードの総数を、前記セグメント長の合計で割ることにより、

前記オーバーレイネットワークのサイズを決定することを含む方法。

[3 0] 前記ノードのセットは、前記第 1 のノードに関するフィンガーを含む [2 9] 記載の方法。

[3 1] 前記ノードのセットは、前記第 1 のノードに関する隣接しているものを含む [2 9] 記載の方法。

[3 2] 前記ノードのセットは、前記第 1 のノードと通信中の 1 つ以上のノードを含む [2 9] 記載の方法。

[3 3] 前記取得することは、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせ、前記セグメント長の少なくとも一部分を取得することを含む [2 9] 記載の方法。

[3 4] 前記取得することは、それぞれ、前記ノードのセットの少なくとも一部分との

ピギーバック通信中に、前記セグメント長の少なくとも一部分を取得することを含む [2 9] 記載の方法。

[3 5] ピアツーピアオーバーレイネットワークのサイズを決定するための装置において、

オーバーレイネットワークの第 1 のノードに係するノードのセットを識別するように構成されているプロセッサと、

前記プロセッサに結合されており、前記ノードのセットのそれぞれのノードに係するセグメント長を取得するように構成されているランシーバと、

前記ノードのセット中のノードの総数を、前記セグメント長の合計で割ることにより、前記オーバーレイネットワークのサイズを決定するように構成されている前記プロセッサとを具備する装置。

[3 6] 前記ノードのセットは、前記第 1 のノードに係するフィンガーを含む [3 5] 記載の装置。

[3 7] 前記ノードのセットは、前記第 1 のノードに係する隣接しているものを含む [3 5] 記載の装置。

[3 8] 前記ノードのセットは、前記第 1 のノードと通信中の 1 つ以上のノードを含む [3 5] 記載の装置。

[3 9] 前記ランシーバは、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせ、前記セグメント長の少なくとも一部分を取得するように構成されている [3 5] 記載の装置。

[4 0] 前記ランシーバは、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記セグメント長の少なくとも一部分を取得するように構成されている [3 5] 記載の装置。

[4 1] ピアツーピアオーバーレイネットワークのサイズを決定するための装置において、

オーバーレイネットワークの第 1 のノードに係するノードのセットを識別する手段と、

前記ノードのセットのそれぞれのノードに係するセグメント長を取得する手段と、

前記ノードのセット中のノードの総数を、前記セグメント長の合計で割ることにより、前記オーバーレイネットワークのサイズを決定する手段とを具備する装置。

[4 2] 前記ノードのセットは、前記第 1 のノードに係するフィンガーを含む [4 1] 記載の装置。

[4 3] 前記ノードのセットは、前記第 1 のノードに係する隣接しているものを含む [4 1] 記載の装置。

[4 4] 前記ノードのセットは、前記第 1 のノードと通信中の 1 つ以上のノードを含む [4 1] 記載の装置。

[4 5] 前記取得する手段は、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせ、前記セグメント長の少なくとも一部分を取得する手段を備える [4 1] 記載の装置。

[4 6] 前記取得する手段は、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記セグメント長の少なくとも一部分を取得する手段を備える [4 1] 記載の装置。

[4 7] ピアツーピアオーバーレイネットワークのサイズを決定するためのコンピュータプログラムプロダクトにおいて、

オーバーレイネットワークの第 1 のノードに係するノードのセットを識別するためのプロセッサにより実行可能なコードと、

前記ノードのセットのそれぞれのノードに係するセグメント長を取得するための前記プロセッサにより実行可能なコードと、

前記ノードのセット中のノードの総数を、前記セグメント長の合計で割ることにより、前記オーバーレイネットワークのサイズを決定するための、前記プロセッサにより実行可

能なコードとを具現化するコンピュータ読取可能媒体を含むコンピュータプログラムプロダクト。

[4 8] 前記ノードのセットは、前記第 1 のノードに関係するフィンガーを含む [4 7] 記載のコンピュータ読取可能媒体。

[4 9] 前記ノードのセットは、前記第 1 のノードに関係する隣接しているものを含む [4 7] 記載のコンピュータ読取可能媒体。

[5 0] 前記ノードのセットは、前記第 1 のノードと通信中の 1 つ以上のノードを含む [4 7] 記載のコンピュータ読取可能媒体。

[5 1] 前記コードは、前記プロセッサに、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせさせて、前記セグメント長の少なくとも一部分を取得させるように構成されている [4 7] 記載のコンピュータ読取可能媒体。

[5 2] 前記コードは、前記プロセッサに、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記セグメント長の少なくとも一部分を取得させるように構成されている [4 7] 記載のコンピュータ読取可能媒体。

[5 3] ピアツーピアオーバーレイネットワークのサイズを決定するための方法において、

オーバーレイネットワークの第 1 のノードに関係するノードのセットを識別することと

前記第 1 のノードに関係するサイズ推定と、前記ノードのセットのそれぞれのノードに関係するサイズ推定とを取得することと、

前記サイズ推定を平均することにより、前記オーバーレイネットワークのサイズを決定することを含む方法。

[5 4] 前記ノードのセットは、前記第 1 のノードに関係するフィンガーを含む [5 3] 記載の方法。

[5 5] 前記ノードのセットは、前記第 1 のノードに関係する隣接しているものを含む [5 3] 記載の方法。

[5 6] 前記ノードのセットは、前記第 1 のノードと通信中の 1 つ以上のノードを含む [5 3] 記載の方法。

[5 7] 前記取得することは、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせ、前記サイズ推定の少なくとも一部分を取得することを含む [5 3] 記載の方法。

[5 8] 前記取得することは、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記サイズ推定の少なくとも一部分を取得することを含む [5 3] 記載の方法。

[5 9] ピアツーピアオーバーレイネットワークのサイズを決定するための装置において、

オーバーレイネットワークの第 1 のノードに関係するノードのセットを識別し、前記第 1 のノードに関係するサイズ推定を取得するように構成されているプロセッサと、

前記プロセッサに結合されており、前記ノードのセットのそれぞれのノードに関係するサイズ推定を取得するように構成されているトランシーバと、

前記サイズ推定を平均することにより、前記オーバーレイネットワークのサイズを決定するように構成されている前記プロセッサとを具備する装置。

[6 0] 前記ノードのセットは、前記第 1 のノードに関係するフィンガーを含む [5 9] 記載の装置。

[6 1] 前記ノードのセットは、前記第 1 のノードに関係する隣接しているものを含む [5 9] 記載の装置。

[6 2] 前記ノードのセットは、前記第 1 のノードと通信中の 1 つ以上のノードを含む [5 9] 記載の装置。

[6 3] 前記トランシーバは、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせ、前記サイズ推定の少なくとも一部分を取得するように構成されている [5 9]

] 記載の装置。

[6 4] 前記トランシーバは、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記サイズ推定の少なくとも一部分を取得するように構成されている [5 9] 記載の装置。

[6 5] ピアツーピアオーバーレイネットワークのサイズを決定するための装置において、

オーバーレイネットワークの第 1 のノードに関係するノードのセットを識別する手段と、

前記第 1 のノードに関係するサイズ推定と、前記ノードのセットのそれぞれのノードに関係するサイズ推定とを取得する手段と、

前記サイズ推定を平均することにより、前記オーバーレイネットワークのサイズを決定する手段とを具備する装置。

[6 6] 前記ノードのセットは、前記第 1 のノードに関係するフィンガーを含む [6 5] 記載の装置。

[6 7] 前記ノードのセットは、前記第 1 のノードに関係する隣接しているものを含む [6 5] 記載の装置。

[6 8] 前記ノードのセットは、前記第 1 のノードと通信中の 1 つ以上のノードを含む [6 5] 記載の装置。

[6 9] 前記取得する手段は、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせ、前記サイズ推定の少なくとも一部分を取得する手段を備える [6 5] 記載の装置。

[7 0] 前記取得する手段は、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記サイズ推定の少なくとも一部分を取得する手段を備える [6 5] 記載の装置。

[7 1] ピアツーピアオーバーレイネットワークのサイズを決定するためのコンピュータプログラムプロダクトにおいて、

オーバーレイネットワークの第 1 のノードに関係するノードのセットを識別するための、プロセッサにより実行可能なコードと、

前記第 1 のノードに関係するサイズ推定を取得するための、前記プロセッサにより実行可能なコードと、

前記ノードのセットのそれぞれのノードに関係するサイズ推定を取得するための、前記プロセッサにより実行可能なコードと、

前記サイズ推定を平均することにより、前記オーバーレイネットワークのサイズを決定するための、前記プロセッサにより実行可能なコードとを具現化するコンピュータ読取可能媒体を含むコンピュータプログラムプロダクト。

[7 2] 前記ノードのセットは、前記第 1 のノードに関係するフィンガーを含む [7 1] 記載のコンピュータ読取可能媒体。

[7 3] 前記ノードのセットは、前記第 1 のノードに関係する隣接しているものを含む [7 1] 記載のコンピュータ読取可能媒体。

[7 4] 前記ノードのセットは、前記第 1 のノードと通信中の 1 つ以上のノードを含む [7 1] 記載のコンピュータ読取可能媒体。

[7 5] 前記コードは、前記プロセッサに、それぞれ、前記ノードのセットの少なくとも一部分に問い合わせさせて、前記サイズ推定の少なくとも一部分を取得させるように構成されている [7 1] 記載のコンピュータ読取可能媒体。

[7 6] 前記コードは、前記プロセッサに、それぞれ、前記ノードのセットの少なくとも一部分とのピギーバック通信中に、前記サイズ推定の少なくとも一部分を取得させるように構成されている [7 1] 記載のコンピュータ読取可能媒体。

【 国際調査報告 】

INTERNATIONAL SEARCH REPORT

International application No PCT/US2010/025665

A. CLASSIFICATION OF SUBJECT MATTER INV. H04L29/06 H04L12/24 H04L29/08 ADD.		
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED Minimum documentation searched (classification system followed by classification symbols) H04L		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practical, search terms used) EPO-Internal, PAJ, WPI Data		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WAI CHEN ET AL: "Dynamic Local Peer Group Organizations for Vehicle Communications" MOBILE AND UBIQUITOUS SYSTEMS - WORKSHOPS, 2006. 3RD ANNUAL INTERNATIONAL CONFERENCE ON, IEEE, PI, 1 July 2006 (2006-07-01), pages 1-8, XP031089403 ISBN: 978-0-7803-9791-0 page 3, right-hand column, line 6 - line 52 ----- -/--	1-28
<input checked="" type="checkbox"/> Further documents are listed in the continuation of Box C. <input type="checkbox"/> See patent family annex.		
* Special categories of cited documents : *A* document defining the general state of the art which is not considered to be of particular relevance *E* earlier document but published on or after the international filing date *L* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) *O* document referring to an oral disclosure, use, exhibition or other means *P* document published prior to the international filing date but later than the priority date claimed *T* later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention *X* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone *Y* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art. *&* document member of the same patent family		
Date of the actual completion of the international search 31 August 2010		Date of mailing of the international search report 06/09/2010
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016		Authorized officer Siebel, Christian

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2010/025665

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	<p>STOICA I ET AL: "MIT-LCS-TR-819 - Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications" INTERNET CITATION 23 March 2001 (2001-03-23), XP002328538 Retrieved from the Internet: URL: http://www.lcs.mit.edu/publications/pubs/ps/MIT-LCS-TR-819.ps [retrieved on 2005-05-17] paragraph [01.0] - paragraph [04.3] paragraph [07.3]</p>	1-28
X	<p>GANESH A J ET AL: "Peer counting and sampling in overlay networks based on random walks" DISTRIBUTED COMPUTING, SPRINGER, BERLIN, DE LNKD- DOI:10.1007/S00446-007-0027-Z, vol. 20, no. 4, 5 June 2007 (2007-06-05), pages 267-278, XP019564156 ISSN: 1432-0452 * abstract paragraph [001.] - paragraph [02.2]</p>	1-28
X	<p>LE MERRER E ET AL: "Peer to peer size estimation in large and dynamic networks: A comparative study" 2006 15TH IEEE INTERNATIONAL CONFERENCE ON HIGH PERFORMANCE DISTRIBUTED COMPUTING PARIS, FRANCE JUNE 19-23 2006, PISCATAWAY, NJ, USA, IEEE, 19 June 2006 (2006-06-19), pages 7-17, XP010926036 ISBN: 978-1-4244-0307-3 * abstract page 8, left-hand column, line 5 - line 15 paragraph [II.]</p>	53-76
A	<p>MAENPAA G CAMARILLO J HAUTAKORPI ERICSSON J: "A Self-tuning Distributed Hash Table (DHT) for Resource Location And Discovery (RELOAD); draft-maenpaa-p2psip-self-tuning-00.txt" A SELF-TUNING DISTRIBUTED HASH TABLE (DHT) FOR RESOURCE LOCATION AND DISCOVERY (RELOAD); DRAFT-MAENPAA-P2PSIP-SELF-TUNING-00.TXT, INTERNET ENGINEERING TASK FORCE, IETF; STANDARDWORKINGDRAFT, INTERNET SOCIETY (ISOC) 4, RUE DES FALAISES CH- 1205 GENEVA, 16 February 2009 (2009-02-16), XP015061541 paragraph [05.1]</p>	29-52

-/--

INTERNATIONAL SEARCH REPORT

International application No PCT/US2010/025665

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	<p>GHINITA G ET AL: "An Adaptive Stabilization Framework for Distributed Hash Tables" PARALLEL AND DISTRIBUTED PROCESSING SYMPOSIUM, 2006. IPDPS 2006. 20TH INTERNATIONAL RHODES ISLAND, GREECE 25-29 APRIL 2006, PISCATAWAY, NJ, USA, IEEE LNKD-DOI:10.1109/IPDPS.2006.1639269, 25 April 2006 (2006-04-25), pages 1-10, XP010920239 ISBN: 978-1-4244-0054-6 page 3, left-hand column, line 3 - line 17 page 6, left-hand column, line 29 - line 36</p> <p style="text-align: center;">-----</p>	29-52

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2010/025665**Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)**

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. Claims Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:

3. Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

see additional sheet

1. As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.

2. As all searchable claims could be searched without effort justifying an additional fees, this Authority did not invite payment of additional fees.

3. As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:

4. No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

Remark on Protest

- The additional search fees were accompanied by the applicant's protest and, where applicable, the payment of a protest fee.
- The additional search fees were accompanied by the applicant's protest but the applicable protest fee was not paid within the time limit specified in the invitation.
- No protest accompanied the payment of additional search fees.

International Application No. PCT/US2010 /025665

FURTHER INFORMATION CONTINUED FROM PCT/ISA/ 210

This International Searching Authority found multiple (groups of) inventions in this international application, as follows:

1. claims: 1-28

Method to inform a distant counter about a change in the size of a peer-to peer network

2. claims: 29-52

Method to estimate the size of a peer-to peer network using heuristics and segment lengths

3. claims: 53-76

Method to ameliorate the estimation of the size of a peer-to peer network based on averaging size estimates

フロントページの続き

(81) 指定国 AP(BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), EA(AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), EP(AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, SE, SI, SK, SM, TR), OA(BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG), AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW

(74) 代理人 100109830

弁理士 福原 淑弘

(74) 代理人 100075672

弁理士 峰 隆司

(74) 代理人 100095441

弁理士 白根 俊郎

(74) 代理人 100084618

弁理士 村松 貞男

(74) 代理人 100103034

弁理士 野河 信久

(74) 代理人 100119976

弁理士 幸長 保次郎

(74) 代理人 100153051

弁理士 河野 直樹

(74) 代理人 100140176

弁理士 砂川 克

(74) 代理人 100158805

弁理士 井関 守三

(74) 代理人 100124394

弁理士 佐藤 立志

(74) 代理人 100112807

弁理士 岡田 貴志

(74) 代理人 100111073

弁理士 堀内 美保子

(74) 代理人 100134290

弁理士 竹内 将訓

(72) 発明者 ダス、サウミトラ・モーハン

アメリカ合衆国、カリフォルニア州 9 2 1 2 1、サン・ディエゴ、モアハウス・ドライブ 5 7 7 5

(72) 発明者 ナラヤナン、ビドヤ

アメリカ合衆国、カリフォルニア州 9 2 1 2 1、サン・ディエゴ、モアハウス・ドライブ 5 7 7 5

(72) 発明者 ドンデティ、ラクシュミナス・レディー

アメリカ合衆国、カリフォルニア州 9 2 1 2 1、サン・ディエゴ、モアハウス・ドライブ 5 7 7 5

(72) 発明者 ジャヤラム、ランジス・エス .

アメリカ合衆国、カリフォルニア州 9 2 1 2 1、サン・ディエゴ、モアハウス・ドライブ 5 7 7 5

F ターム(参考) 5K067 AA22 AA43 BB21 CC21 DD17 EE02 EE25 HH22 HH23