(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第5285690号
(P5285690)
(45)発行日　平成25年9月11日(2013.9.11)　(24)登録日　平成25年6月7日(2013.6.7)

(51)Int.Cl.　　F I
G06F 15/177 (2006.01)　　G06F 15/177 C
G06F 15/173 (2006.01)　　G06F 15/173 640C

請求項の数 17　(全 14 頁)

(21)出願番号　特願2010-503440 (P2010-503440)
(86)(22)出願日　平成20年3月20日 (2008.3.20)
(65)公表番号　特表2010-525433 (P2010-525433A)
(43)公表日　平成22年7月22日 (2010.7.22)
(86)国際出願番号　PCT/EP2008/053377
(87)国際公開番号　WO2008/128836
(87)国際公開日　平成20年10月30日 (2008.10.30)
審査請求日　平成22年10月29日 (2010.10.29)
(31)優先権主張番号　11/736,811
(32)優先日　平成19年4月18日 (2007.4.18)
(33)優先権主張国　米国 (US)

(73)特許権者　390009531
インターナショナル・ビジネス・マシーンズ・コーポレーション
ＩＮＴＥＲＮＡＴＩＯＮＡＬ　ＢＵＳＩＮＥＳＳ　ＭＡＣＨＩＮＥＳ　ＣＯＲＰＯＲＡＴＩＯＮ
アメリカ合衆国１０５０４　ニューヨーク州　アーモンク　ニュー　オーチャード　ロード
(74)代理人　100108501
弁理士　上野　剛史
(74)代理人　100112690
弁理士　太佐　種一
(74)代理人　100091568
弁理士　市位　嘉宏

最終頁に続く

(54)【発明の名称】並列コンピュータ・システム、並列コンピュータ・システム上のノード・トラフィックを動的に再経路指定するためのコンピュータ実装方法、コンピュータ可読記録媒体及びコンピュータ・プ

(57)【特許請求の範囲】
【請求項１】
並列コンピュータ・システムであって、
論理的な３次元デカルト配列内で、各ノードをその６個の最近隣ノード（Ｘ＋、Ｘ－、Ｙ＋、Ｙ－、Ｚ＋及びＺ－）に接続するトーラス・ネットワークによって接続された複数のノードと、
前記並列コンピュータ・システムの前記複数のノード及びネットワーク接続を監視し且つノード及びネットワーク接続の問題リストを作成するノード／ネットワーク監視機構とを備え、
前記複数のノードの少なくともいずれかは、前記問題リスト内のノード及びネットワーク接続を回避するように、各ノードから送信されるデータ・パケット内に６個の最近隣ノード（Ｘ＋、Ｘ－、Ｙ＋、Ｙ－、Ｚ＋及びＺ－）用に設定されたヒント・ビットを使用して、データ・パケットを障害ノード又は障害ネットワーク接続を迂回して１つ以上の前記ネットワーク接続を介して動的に経路指定するものである、並列コンピュータ・システム。
【請求項２】
前記ヒント・ビットが、トーラス・ネットワーク上でトラフィックを送信すべき好ましい方向を指示する複数の２進値であり、６つのビットをもって、ＸＹＺの順序で、Ｘ＋　Ｘ－　Ｙ＋　Ｙ－　Ｚ＋　Ｚ－のように定義され、Ｘ＋又はＸ－の何れか一方を設定することができるが、両方を設定することができないものである、請求項１に記載の並列コン

ピュータ・システム。
【請求項３】
前記ヒント・ビット（５１６）が、前記ネットワーク接続を介して送信されるデータ・パケット（５１０）のヘッダ（５１２）内に含まれる、請求項１又は請求項２に記載の並列コンピュータ・システム。
【請求項４】
前記ノードは、更新された前記問題リストを検出し、前記検出に応じて当該ノード上で実行中のアプリケーションを一時的に停止し、更新された前記問題リストに対応する前記ヒント・ビットを使用してデータ・パケットを動的に経路指定するようにし、前記アプリケーションを一時的に停止された点から再開するものである、請求項１ないし請求項３の何れか１項に記載の並列コンピュータ・システム。
【請求項５】
前記並列コンピュータ・システムが、前記複数のノードを３次元トーラス・ネットワークで相互接続した超並列コンピュータ・システムである、請求項１ないし請求項４の何れか１項に記載の並列コンピュータ・システム。
【請求項６】
ヒント・ビットを使用することにより、並列コンピュータ・システム上で実行中のアプリケーションを再始動することなく、ヒント・ビットを使用して、当該並列コンピュータ・システム内の論理的な３次元デカルト配列内で、各ノードをその６個の最近隣ノード（Ｘ＋、Ｘ－、Ｙ＋、Ｙ－、Ｚ＋及びＺ－）に接続するトーラス・ネットワークによって接続された複数の計算ノード上のノード・トラフィックを動的に再経路指定するためのコンピュータ実装方法であって、
問題について前記複数の計算ノード及びネットワーク接続を監視し且つ障害ノード及び障害ネットワーク接続を問題リスト内で特定するステップと、
更新済みの問題リストを検出するステップと、
アプリケーションを実行中の計算ノードの実行を一時的に停止させるステップと、
前記問題リスト内のノード又はネットワーク接続を分離するように、各ノードから送信されるデータ・パケット内に６個の最近隣ノード（Ｘ＋、Ｘ－、Ｙ＋、Ｙ－、Ｚ＋及びＺ－）用に設定されたヒント・ビットを設定するステップと、
一時的に停止させていた前記アプリケーションの実行を再開するステップとを含む、コンピュータ実装方法。
【請求項７】
前記ヒント・ビットが、トーラス・ネットワーク上でトラフィックを送信すべき好ましい方向を指示する複数の２進値であり、６つのビットをもって、ＸＹＺの順序で、Ｘ＋　Ｘ－　Ｙ＋　Ｙ－　Ｚ＋　Ｚ－のように定義され、Ｘ＋又はＸ－の何れか一方を設定することができるが、両方を設定することができないものである、請求項６に記載のコンピュータ実装方法。
【請求項８】
前記ヒント・ビット（５１６）が、前記ネットワーク接続を介して送信されるデータ・パケット（５１０）のヘッダ（５１２）内に含まれる、請求項６又は請求項７に記載のコンピュータ実装方法。
【請求項９】
前記並列コンピュータ・システムが、前記複数の計算ノードを３次元トーラス・ネットワークで相互接続した超並列コンピュータ・システムである、請求項６ないし請求項８の何れか１項に記載のコンピュータ実装方法。
【請求項１０】
少なくとも１つの障害ノードのプロセスを少なくとも１つのバックアップ・ノードに移動させるステップをさらに含む、請求項６ないし請求項９の何れか１項に記載のコンピュータ実装方法。
【請求項１１】

計算ノードに対する輻輳ネットワークを検出するステップと、
トラフィックを当該輻輳ネットワークを迂回して経路指定するように少なくとも１つのヒント・ビットを設定するステップとをさらに含む、請求項６ないし請求項１０の何れか１項に記載のコンピュータ実装方法。
【請求項１２】
論理的な３次元デカルト配列内で、各ノードをその６個の最近隣ノード（Ｘ＋、Ｘ－、Ｙ＋、Ｙ－、Ｚ＋及びＺ－）に接続するトーラス・ネットワークを構成するネットワーク接続によって接続された複数のノードを有する並列コンピュータ・システムにおいて、
コンピュータを、
前記複数のノード及びネットワーク接続を監視し且つノード及びネットワーク接続の問題リストを作成するノード／ネットワーク監視機構として機能させ、そして
前記問題リスト内のノード及びネットワーク接続を回避するように、各ノードから送信されるデータ・パケット内に６個の最近隣ノード（Ｘ＋、Ｘ－、Ｙ＋、Ｙ－、Ｚ＋及びＺ－）用に設定されたヒント・ビットを使用して、データ・パケットを前記ネットワーク接続を介して動的に経路指定するノード経路指定機構として機能させるためのコンピュータ・プログラム。
【請求項１３】
前記ヒント・ビットが、トーラス・ネットワーク上でトラフィックを送信すべき好ましい方向を指示する複数の２進値であり、６つのビットをもって、ＸＹＺの順序で、Ｘ＋　Ｘ－　Ｙ＋　Ｙ－　Ｚ＋　Ｚ－のように定義され、Ｘ＋又はＸ－の何れか一方を設定することができるが、両方を設定することができないものである、請求項１２に記載のコンピュータ・プログラム。
【請求項１４】
前記ヒント・ビット（５１６）が、前記ネットワーク接続を介して送信されるデータ・パケット（５１０）のヘッダ（５１２）内に含まれる、請求項１２又は請求項１３に記載のコンピュータ・プログラム。
【請求項１５】
前記ノードは、更新された前記問題リストを検出し、前記検出に応じて当該ノード上で実行中のアプリケーションを一時的に停止し、更新された前記問題リストに対応する前記ヒント・ビットを使用してデータ・パケットを動的に経路指定するようにし、前記アプリケーションを一時的に停止された点から再開するものである、請求項１２ないし請求項１４の何れか１項に記載のコンピュータ・プログラム。
【請求項１６】
前記並列コンピュータ・システムが、前記複数のノードを３次元トーラス・ネットワークで相互接続した超並列コンピュータ・システムである、請求項１２ないし請求項１５の何れか１項に記載のコンピュータ・プログラム。
【請求項１７】
請求項６ないし請求項１１の何れか１項に記載の方法の各ステップをコンピュータに実行させるためのコンピュータ・プログラム。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、並列コンピュータ・システム内の障害回復に係り、さらに詳細に説明すれば、超並列コンピュータ・システム上で実行中のアプリケーションを再始動することなく、ヒント・ビットを使用して、超並列コンピュータ・システムの計算ノード上のノード・トラフィックを動的に再経路指定するための装置及び方法に係る。
【背景技術】
【０００２】
高性能のコンピュータ・システムについては、故障時間及び修理コストを減少させるために、効率的な障害回復が重要である。多数の計算ノードを有する並列コンピュータ・シ

ステムでは、単一コンポーネントの障害が生じる結果として、大きな部分又はコンピュータ全体が、修理のためにオフラインに強制されることがある。アプリケーションの再始動は、かかる障害に先行する相当な量の処理時間を浪費することがある。

【０００３】

超並列コンピュータ・システムは、相互接続された多数の計算ノードを有する、１つのタイプの並列コンピュータ・システムである。かかる超並列コンピュータのファミリは、「ＢｌｕｅＧｅｎｅ」という名前の下で、インターナショナル・ビジネス・マシーンズ・コーポレーション（ＩＢＭ）によって開発中である。ＢｌｕｅＧｅｎｅ／Ｌシステムは、スケーラブル・システムであり、計算ノードの現在の最大数は、６５，５３６である。ＢｌｕｅＧｅｎｅ／Ｌシステムのノードは、２つのＣＰＵ及びメモリを有する単一のＡＳＩＣ（特定用途向け集積回路）から成る。コンピュータ全体は、６４個のラック内に収容され、各ラック内には３２個のノード・ボードがある。

【０００４】

Ｂｌｕｅ Ｇｅｎｅ／Ｌシステムは、幾つかの通信ネットワークを介して通信する。６５，５３６個の計算ノードは、１つの論理ツリー・ネットワーク及び１つの３次元トーラス・ネットワークの両方に配列される。論理ツリー・ネットワークは、各計算ノードが１つの親ノード及び１つ又は２つの子ノードと通信するように、計算ノードをツリー構造状に接続する。トーラス・ネットワークは、各計算ノードがコンピュータの１つのセクション内でその６個の最近隣ノードと通信するように、計算ノードを３次元の格子構造状に接続する。計算ノードが、隣接ノードとの通信を必要とするトーラス及びツリー・ネットワーク内で配列されるので、単一ノードのハードウェア障害が生じると、障害ハードウェアを修理することができるまで、システムの大きな部分が停止させられることになる。例えば、単一ノードの障害が生じると、トーラス・ネットワークの完全なセクションが作動不能になることがある。ＢｌｕｅＧｅｎｅ／Ｌシステム内のトーラス・ネットワークの１つのセクションは、ラックの半分又は５１２個のノードである。さらに、障害が修正されるまで、障害を有する区画に割り当てられた全てのハードウェアをオフラインにする必要があろう。

【発明の概要】

【発明が解決しようとする課題】

【０００５】

従来技術の超並列コンピュータ・システムでは、実行中に単一ノードの障害が生じると、ソフトウェア・アプリケーションを最初から又は保存されたチェックポイントから再始動することをしばしば必要とする。障害事象が生じる場合、障害ノードの処理を他のノードに移動できるようにすることが有利であろう。そのようにすれば、アプリケーションを、最小の遅延で、バックアップ・ハードウェア上で再開することが可能となり、その結果、システム全体の効率が向上するからである。障害ノード又は障害を生じる可能性があると指示されたノードから一層効果的に回復する方法がなければ、並列コンピュータ・システムは、潜在的なコンピュータ処理時間を浪費し続け、その結果、運用コストが増大するであろう。

【課題を解決するための手段】

【０００６】

超並列コンピュータ・システム上で実行中のアプリケーションを再始動することなく、障害ノード又は輻輳ネットワークを迂回して経路指定するように、ヒント・ビットを使用して、超並列コンピュータ・システムの計算ノード上のノード・トラフィックを動的に再経路指定するための装置及び方法が開示される。１つのノードが障害を有するか、又は当該ノードが障害を生じる可能性があるという指示が存在する場合、超並列コンピュータ・システム上のアプリケーション・ソフトウェアが中断され、その間に、障害ノード上のデータがバックアップ・ノードに移動される。トーラス・ネットワークのトラフィックは、障害ノードを迂回して経路指定され、そして当該障害ノード用のトラフィックは、バックアップ・ノードに再経路指定される。同様に、ネットワーク・トラフィックは、輻輳ネッ

トワークを迂回して経路指定することができる。
【０００７】
本明細書に開示する複数の例及び説明は、Ｂｌｕｅ Ｇｅｎｅアーキテクチャに向けられているが、これらの例及び説明は、ノード・ハードウェアが他のノードからのカットスルー・トラフィックを処理する、ネットワーク構造状に配列された複数のプロセッサを有する任意の並列コンピュータ・システムまで及ぶ。
【発明の効果】
【０００８】
本発明は、ノード・トラフィックを動的に再経路指定することにより、故障時間の量を著しく減少させることができ、その結果、コンピュータ・システムの効率を向上させることができるという効果を奏する。
【図面の簡単な説明】
【０００９】
【図１】超並列コンピュータ・システムのブロック図である。
【図２】超並列コンピュータ・システム内の計算ノードの入出力接続を示すブロック図である。
【図３】超並列コンピュータ・システム内の計算ノードのブロック図である。
【図４】超並列コンピュータ・システム内の計算ノードのトーラス・ネットワーク・ハードウェアのブロック図である。
【図５】超並列コンピュータ・システム内のトーラス・ネットワーク用データ・パケットのブロック図である。
【図６】ノード・トラフィックを動的に再経路指定する１つの例を示すために、超並列コンピュータ・システムの一部を表すブロック図である。
【図７】ノード・トラフィックを動的に再経路指定する他の例を示すために、超並列コンピュータ・システムの一部を表すブロック図である。
【図８】並列コンピュータ・システム内の問題リストを作成するために、ノード及びネットワークを監視するための方法のフローチャートである。
【図９】並列コンピュータ・システム内のヒント・ビットを使用して、ノード・プロセスを動的に再経路指定するための方法のフローチャートである。
【発明を実施するための形態】
【００１０】
本発明は、超並列コンピュータ・システム上で実行中のアプリケーションを再始動することなく、ヒント・ビットを使用して、超並列コンピュータ・システムの計算ノード上のノード・トラフィックを動的に再経路指定するための装置及び方法に向けられている。１つのノードが障害を有するか、又は当該ノードが障害を生じる可能性があるという指示が存在する場合、超並列コンピュータ・システム上のアプリケーション・ソフトウェアが中断され、その間に、障害ノード上のデータがバックアップ・ノードに移動される。トーラス・ネットワークのトラフィックは、障害ノードを迂回して経路指定され、そして当該障害ノード用のトラフィックは、バックアップ・ノードに再経路指定される。以下、インターナショナル・ビジネス・マシーンズ・コーポレーション（ＩＢＭ）によって開発中のＢｌｕｅ Ｇｅｎｅ／Ｌシステムに関連して、複数の例を説明する。
【００１１】
図１は、Ｂｌｕｅ Ｇｅｎｅ／Ｌシステムのような超並列コンピュータ・システム１００を表すブロック図である。ＢｌｕｅＧｅｎｅ／Ｌシステムは、スケーラブル・システムであり、その計算ノードの最大数は６５，５３６である。各計算ノード１１０は、「Ｂｌｕｅ Ｇｅｎｅ／Ｌ」計算チップとも呼ばれる、１つの特定用途向け集積回路（ＡＳＩＣ）１１２を有する。計算チップ１１２は、２つのプロセッサ又は中央処理装置（ＣＰＵ）を有し、ノード・ドータ・カード１１４に取り付けられる。一般に、計算ノード１１０は、５１２メガバイトのローカル・メモリ（図示せず）を有する。１つのノード・ボード１２０は、３２個のノード・ドータ・カード１１４を収容し、各ノード・ドータ・カード１

１４は、１つの計算ノード１１０を有する。従って、各ノード・ボード１２０は、３２個の計算ノード１１０を有し、各計算ノード１１０は、２つのプロセッサ及び各プロセッサに関連するメモリを有する。ラック１３０は、３２個のノード・ボード１２０を収容するハウジングである。各ノード・ボード１２０は、ミッドプレーン・コネクタ１３４を介して、ミッドプレーン１３２に接続する。ミッドプレーン１３２は、ラックの内部にあり、図１には示されていない。ＢｌｕｅＧｅｎｅ／Ｌシステムの全体は、各々が３２個のノード・ボード１２０を有する、６４個のラック１３０内に収容されるであろう。従って、システムの全体は、６５，５３６個の計算ノード１１０及び１３１，０７２個のＣＰＵ（６４ラック×３２ノード・ボード×３２ノード×２ＣＰＵ）を有するであろう。

【００１２】

Ｂｌｕｅ Ｇｅｎｅ／Ｌシステムの構造は、１つのＩ／Ｏノード面を有する、１つの計算ノード・コアとして記述することができる。すなわち、１，０２４個の計算ノード１１０への通信が、サービス・ノード１４０に接続されたＩ／Ｏプロセッサ１７０を有する、各Ｉ／Ｏノードによって処理される。これらのＩ／Ｏノードは、ローカル・ストレージを有していない。これらのＩ／Ｏノードは、論理ツリー・ネットワークを通して計算ノード１１０に接続され、また機能的なネットワーク（図示せず）を通して機能的な広域ネットワーク能力を有する。この機能的なネットワークは、サービス・ノード１４０から多数の計算ノード１１０への通信を処理するための１つのノード・ボード１２０に位置する、１つのＩ／Ｏプロセッサ（又は「Ｇｅｎｅ／Ｌ」リンク・チップ）１７０に接続される。Ｂｌｕｅ Ｇｅｎｅ／Ｌシステムは、ノード・ボード１２０に接続された１つのＩ／Ｏボード（図示せず）上に、１つ以上のＩ／Ｏプロセッサ１７０を有する。これらのＩ／Ｏプロセッサ１７０は、８個、３２個又は６４個の計算ノード１１０と通信するように構成することができる。複数のＩ／Ｏノードへの接続は、当該Ｉ／Ｏノードがトーラス・ネットワークに接続されないことを除けば、計算ノード１１０への接続に類似する。

【００１３】

コンピュータ・システム１００に含まれるサービス・ノード１４０は、ノードへのソフトウェアのローディングを処理するとともに、システム全体の動作を制御する。一般に、サービス・ノード１４０は、リナックス（登録商標）を実行する、制御コンソール（図示せず）付きの「ＩＢＭ ｐＳｅｒｉｅｓ」サーバのようなミニコンピュータ・システムである。サービス・ノード１４０は、制御システム・ネットワーク１５０を介して、計算ノード１１０のラック１３０に接続される。制御システム・ネットワーク１５０は、ＢｌｕｅＧｅｎｅ／Ｌシステム用の制御、テスト及び立ち上げインフラストラクチャを提供する。制御システム・ネットワーク１５０は、超並列コンピュータ・システムに必要な通信を提供する、種々のネットワーク・インタフェースを含む。これらのネットワーク・インタフェースについては、後述する。

【００１４】

サービス・ノード１４０は、システム管理専用の制御システム・ネットワーク１５０を管理する。制御システム・ネットワーク１５０に含まれる１００Ｍｂｐｓの専用イーサネット（登録商標）・リンクは、サービス・ノード１４０から多数のノードへの通信を処理するノード・ボード１２０上に位置する、Ｉｄｏチップ１８０に接続される。このネットワークがＪＴＡＧネットワークとも呼ばれるのは、これがＪＴＡＧプロトコルを使用して通信するためである。ノード・ボード１２０上にある計算ノード１１０の制御、テスト及び立ち上げの全ては、サービス・ノード１４０と通信するＪＴＡＧポートを通して管理される。さらに、サービス・ノード１４０に含まれるノード／ネットワーク監視機構１４２は、回避すべき障害ノード、障害を生じる可能性があるノード又はネットワーク・リンクを指示する、問題リスト１４４を維持する。ノード／ネットワーク監視機構１４２は、サービス・ノード１４０内のソフトウェアから成るが、システムのノード上で実行中のオペレーティング・システムによって支援されることがある。

【００１５】

Ｂｌｕｅ Ｇｅｎｅ／Ｌシステムは、幾つかの通信ネットワークを介して通信する。図

２は、ＢｌｕｅＧｅｎｅ／Ｌシステム内の１つの計算ノードのＩ／Ｏ接続を示すブロック図である。６５，５３６個の計算ノード１１０及び１，０２４個のＩ／Ｏプロセッサ１７０は、論理ツリー・ネットワーク及び論理的な３次元トーラス・ネットワークの両方に配列される。トーラス・ネットワークは、各計算ノード１１０がその６個の最近隣ノードと通信するように、計算ノードを３次元の格子構造状に接続する。図２では、トーラス・ネットワークは、当該ノードを６個の隣接ノードにそれぞれ接続するネットワーク接続Ｘ＋、Ｘ－、Ｙ＋、Ｙ－、Ｚ＋及びＺ－によって例示される。一方、ツリー・ネットワークは、図２のＴｒｅｅ０、Ｔｒｅｅ１及びＴｒｅｅ２接続によって表される。当該ノードに接続される他の通信ネットワークは、１つのＪＴＡＧネットワーク及び１つのグローバル割り込みネットワークを含む。ＪＴＡＧネットワークは、サービス・ノード１４０から図１の制御システム・ネットワーク１５０を介して、テスト及び制御用の通信を提供する。グローバル割り込みネットワークは、或るタスクの完了時に異なる処理段階に移動するように、複数の計算ノード上の同様のプロセスを同期化するためのソフトウェア・バリアを実装するために使用される。従って、このグローバル割り込みネットワークは、ノードの１つの区画で実行中のアプリケーションを始動、停止又は一時停止させるために使用することができる。さらに、各計算ノード１１０に対するクロック及び電力信号がある。

【００１６】

Ｂｌｕｅ Ｇｅｎｅ／Ｌシステムのトーラス相互接続は、論理的な３次元デカルト配列内で、各ノードをその６個の最近隣ノード（Ｘ＋、Ｘ－、Ｙ＋、Ｙ－、Ｚ＋及びＺ－）に接続する。６個の最近隣ノードへの接続は、ノード・レベル及びミッドプレーン・レベルで行われる。各ミッドプレーンは、８×８×８ノードの配列である。ミッドプレーン内のノード配列の６個の面（Ｘ＋、Ｘ－、Ｙ＋、Ｙ－、Ｚ＋及びＺ－）の各々は、８Ｈ８＝６４ノードのサイズを有する。６個の面の各々にある６４ノードからの各トーラス・ネットワーク信号は、当該ミッドプレーンに接続されたリンク・カード（図示せず）を通して、隣接ミッドプレーン内の対応するノードに通信される。当該ミッドプレーンが、任意の次元において１ミッドプレーンの深さを有する区画内で使用される場合、各面の信号は、対向面上にある同じミッドプレーンの入力に戻されることがある。

【００１７】

図３は、従来技術に従った、Ｂｌｕｅ Ｇｅｎｅ／Ｌシステム内の計算ノード１１０のブロック図である。計算ノード１１０は、ノード計算チップ１１２を有し、当該チップは、２つのプロセッサ３１０Ａ、３１０Ｂを有する。各プロセッサ３１０Ａ、３１０Ｂは、それぞれ処理コア３１２を有する。これらのプロセッサは、レベル３のメモリ・キャッシュ（Ｌ３キャッシュ）３２０及びスタティックＲＡＭ（ＳＲＡＭ）３３０に接続される。Ｌ３キャッシュ３２０からのデータは、ダブル・データ・レート（ＤＤＲ）メモリ・コントローラ３５０を介して、ＤＤＲ同期型ダイナミックＲＡＭ（ＳＤＲＡＭ）３４０にロードされる。

【００１８】

ＳＲＡＭ３３０は、計算チップ１１２からのデータをＩｄｏチップ１８０に通信する、ＪＴＡＧインタフェース３６０に接続される。サービス・ノード（図１）は、Ｉｄｏチップ１８０及び制御システム・ネットワーク１５０（図１）の一部であるイーサネット（登録商標）・リンクを通して、計算ノード１１０と通信する。Ｂｌｕｅ Ｇｅｎｅ／Ｌシステムでは、ノード・ボード１２０当たり１つのＩｄｏチップが存在し、各ミッドプレーン１３２（図１）内のボード上に他のものが存在する。Ｉｄｏチップ１８０は、信頼のおける１００Ｍｂｐｓの専用イーサネット（登録商標）制御ネットワーク上の生のＵＤＰパケットを使用して、サービス・ノードからコマンドを受信する。Ｉｄｏチップは、計算ノードとの通信のために、種々のシリアル・プロトコルをサポートする。ＪＴＡＧプロトコルは、サービス・ノード１４０（図１）から計算 ノード１１０内のＳＲＡＭ ３３０の任意のアドレスを対象とする読み取り及び書き込みを行うために使用され、またシステム初期化及びブート・プロセスのために使用される。

【００１９】

また、図３のノード計算チップ１１２は、ネットワーク・ハードウェア３９０を含む。ネットワーク・ハードウェア３９０は、トーラス・ネットワーク・ハードウェア３９２、ツリー・ネットワーク・ハードウェア３９４及びグローバル割り込みネットワーク・ハードウェア３９６を含む。前述のように、ＢｌｕｅＧｅｎｅ／Ｌシステムのこれらのネットワークは、計算ノード１１０がシステム内の他のノードと通信するために使用される。ネットワーク・ハードウェア３９０は、計算ノード１１０がトーラス・ネットワークを介してデータ・パケットを受信し且つこれを受け渡すことを可能にする。ネットワーク・ハードウェア３９０は、ネットワーク・データのトラフィックを独立的に処理し、その結果、計算ノード１１０のプロセッサ３１０Ａ、３１０Ｂは、トーラス・ネットワーク上を流れるデータの量によって負担を負わされることはない。他のノードを宛先として当該ノードをパススルーするネットワーク・データは、「カットスルー」トラフィックと呼ばれる。

【００２０】

図４は、図３のトーラス・ネットワーク・ハードウェア３９２のブロック図である。トーラス・ネットワーク・ハードウェア３９２は、３つの主要なユニット、すなわちプロセッサ・インタフェース４１０、トーラス・センダ４２０及びトーラス・レシーバ４３０から成る、プロセッサ・インタフェース４１０は、先入れ先出し式のキューとして、プロセッサ注入ＦＩＦＯ４１２及びプロセッサ受信ＦＩＦＯ４１４から成る。これらのＦＩＦＯに対するアクセスは、プロセッサ３１０Ａ、３１０Ｂ（図３）から２つの浮動小数点ユニット（ＦＰＵ）レジスタ（図示せず）を介して行われる。すなわち、１対のＦＰＵレジスタから１２８ビットのメモリ・マップド・ストアを介して、データがこれらのＦＩＦＯにロードされる。一方、これらのＦＩＦＯからデータが読み取られ、１２８ビット単位で、１対のＦＰＵレジスタにロードされる。全部で８個あるプロセッサ注入ＦＩＦＯ４１２は、２グループに編成される。すなわち、一方のグループは、（ノード間オペレーティング・システム・メッセージ用の）２つの高優先順位ＦＩＦＯから成り、他方のグループは、６個の通常優先順位ＦＩＦＯから成る（それらのＦＩＦＯは最近隣ノード接続性のために十分である）。全てのＦＩＦＯ内のパケットは、トーラス・ネットワーク上の任意の方向に送出することができる。プロセッサ受信ＦＩＦＯ４１４にも、ＦＩＦＯの２つのグループがある。各グループは、７個のＦＩＦＯを含み、そのうち１つのＦＩＦＯは高優先順位用のものであり、６個のＦＩＦＯの各々は６入力方向の各々に専用される。具体的には、各レシーバとその対応する受信ＦＩＦＯの間に、１つの専用バスがある。ストレージについては、全てのトーラスＦＩＦＯは、エラー検出及び訂正（ＥＣＣ）によって保護されたＳＲＡＭチップを使用し、全ての内部データ経路はパリティについて検査される。

【００２１】

前述のトーラス・ネットワーク・ハードウェア３９２は、種々のトーラス・ネットワークを横切って、可変サイズのデータ・パケットを送信する。図５は、トーラス・ネットワーク・パケット５１０の１例を示す。Ｂｌｕｅ Ｇｅｎｅ／Ｌシステムでは、各パケット５１０は、ｎ Ｈ ３２バイトである。但し、ｎ＝１～８個の「チャンク」である。例えば、ＭＰＩ（Message Passing Interface）規格に準拠するメッセージは、関連する１つ又は両方のプロセッサ３１０Ａ、３１０Ｂ（図３）上で実行中のソフトウェアによって構築され、送信され且つ受信される、多数のパケットから成ることがある。各パケットの最初の８バイトは、パケット・ヘッダ５１２である。パケット・ヘッダ５１２は、リンク・レベルのプロトコル情報（例えば、シーケンス番号）と、宛先を含む経路指定情報と、仮想チャンネル及びサイズと、伝送中のヘッダ・データの破損を検出する、バイト幅の巡回冗長検査（ＣＲＣ）５１４とを保持する。また、パケット・ヘッダ５１２は、後述するヒント・ビット５１６を保持する。

【００２２】

パケット・ヘッダ５１２の後には、多数のデータ・バイト５１８が保持される。さらに、各パケットには、１バイトの有効性標識５２０とともに、２４ビットのＣＲＣが付加される。有効性標識が必要であるのは、パケットが完全に受信される前にその転送が開始されることがあるためである。このＣＲＣは、各パケットが各リンクを介して送信される際

に、当該各パケットの検査を可能にする。タイムアウト機構は、破損パケットを再送するために使用される。８ビットのパケット・ヘッダＣＲＣを使用することは、パケット・ヘッダ・エラーの早期検出を可能にする最適化である。というのは、ヘッダＣＲＣが、完全なパケットＣＲＣ内に含まれるからである。

【００２３】

前述のように、ヘッダ５１２は、６つの「ヒント」ビット５１６を保持する。ヒント・ビット５１６は、トーラス・ネットワークの３次元内でパケットを経路指定することができる方向を指示する。ヒント・ビット５１６は、ＸＹＺの順序で、Ｘ＋ Ｘ－ Ｙ＋ Ｙ－ Ｚ＋ Ｚ－のように定義される。例えば、１００１００のヒント・ビットは、当該パケットをＸ＋及びＹ－方向に経路指定することができることを意味する。Ｘ＋又はＸ－ヒント・ビットの何れか一方を設定することができるが、両方を設定することはできない。というのは、設定された１つのビットは、その次元内でパケットを送信すべき方向を指示するからである。デフォルトは、パケットを任意の方向に送信できることを指示するように、全てのヒント・ビットを設定解除又はゼロに設定することであろう。

【００２４】

一般に、トーラス・ネットワーク内では、ノード間をデータが流れる次元順序が存在する。本明細書で開示する複数の例では、その次元順序がＸＹＺであると仮定するが、他の順序も使用することができる。次元順序がＸＹＺであるという意味は、最初に、データが１つのノードからＸ次元内で流れ、次に、Ｙ次元内で幾つかのノードを通り、次に、Ｚ次元内で流れるというものである。ＸＹＺヒント・ビットは、ＸＹＺ次元内の経路指定のためにそれぞれ使用される。

【００２５】

各ノードは、トーラス機能（図示せず）を制御する、ソフトウェアで構成可能な１組のレジスタを維持する。例えば、１組のレジスタは、その隣接ノードの座標を保持する。ヒント・ビットがゼロに設定されるのは、１つのパケットが１つのノードを１つの方向に離れ、そして隣接ノード座標レジスタによって決定されるように、当該パケットがその次元内のその宛先に到着するような場合である。ヒント・ビットは、ヘッダ内で早期に現れ、その結果、調停を効率的にパイプライン化することができる。ヒント・ビットは、ソフトウェア又はハードウェアの何れかによって初期化することができる。ハードウェアによって初期化される場合、適切な方向を決定するために、１次元当たり１組２つのレジスタが使用される。これらのレジスタは、最小ホップ経路指定を提供するように構成することができる。経路指定を完全に行うためには、ヒント・ビット及び仮想チャンネルを調べることが必要である。すなわち、経路指定テーブルは、存在しない。パケットは、動的に又は決定論的な次元順序（ＸＹＺ）で、経路指定することができる。すなわち、パケットは、他のトラフィックに基づいて最小輻輳の経路を辿るか、又は固定経路上で経路指定することができる。２地点間パケットのほかに、ヘッダ内の１つのビットを設定することにより、１つのパケットを任意のデカルト次元内でブロードキャストさせ且つこれを各ノードで受信させることができる。後述するように、ソフトウェアは、ヒント・ビットを適切に設定することにより、「デッド」ノード又はリンクを回避することができる。３つまでの非共線型（noncolinear）障害ノードが存在する場合には、完全な接続性を維持することができる。

【００２６】

図６は、ノード・トラフィックを動的に再経路指定する１つの例を示すために、図１の超並列コンピュータ・システムの部分６００を表すブロック図である。この部分６００は、ノード１ ６１０～ノード９ ６１２と表記した９個のノードを例示する。図面の内容を簡潔にするために、図６は、Ｘ次元及びＹ次元内のノードのみを示しているが、当該コンピュータ・システムは、Ｚ次元内に位置するノードも有することを理解されたい。Ｘ次元及びＹ次元は、ＸＹ軸６１４によって示される通りである。この例については、アプリケーションがノード１ ６１０～ノード８ ６２２上で実行中であるものと仮定する。障害又は潜在的な障害がノード５６１８上で検出される場合、アプリケーションが中断又は一時

停止され、そしてＦＩＦＯ内の全てのネットワーク・トラフィックがクリアされるまで待機することにより、ネットワークが休止される。次に、障害ノード５　６１８上のアプリケーションは、スペア・ノード（ノード９　６１２）に移動される。次に、障害ノード５　６１８を通してデータを移動させる必要がある各ノードは、更新済みの問題リスト（図１の１４４）を全てのノード又は少なくとも影響を受けるノードに送信することにより、障害ノード５６１８を回避するように更新される。

【００２７】

これらのノードは、更新済みの問題リストを使用して、データが障害ノード又は障害ネットワークを迂回して経路指定されることを保証する。次に、データが障害ノード５　６１８を迂回して経路指定されるように、各ノードから送信されるデータ・パケット内に適切なヒント・ビットが設定される。図６の例では、ノード２　６２０からのデータ・パケットは、Ｘ－用に設定されたヒント・ビットを有するから、Ｘ－方向に進行してノード８　６２２に至り、その結果、障害ノード５　６１８を回避するであろう。同様に、ノード８　６２２からのデータ・パケットは、Ｘ＋用に設定されたヒント・ビットを有するから、Ｘ＋方向に進行してノード２　６２０に至る。また、ノード４　６２４からのデータ・パケットは、Ｙ＋用に設定されたヒント・ビットを有するから、Ｙ＋方向に進行してノード６　６２６に至り、その結果、障害ノード５　６１８を回避するであろう。これに対し、ノード６　６２６からのデータ・パケットは、Ｙ－用に設定されたヒント・ビットを有するから、Ｙ－方向に進行してノード４　６２４に至る。

【００２８】

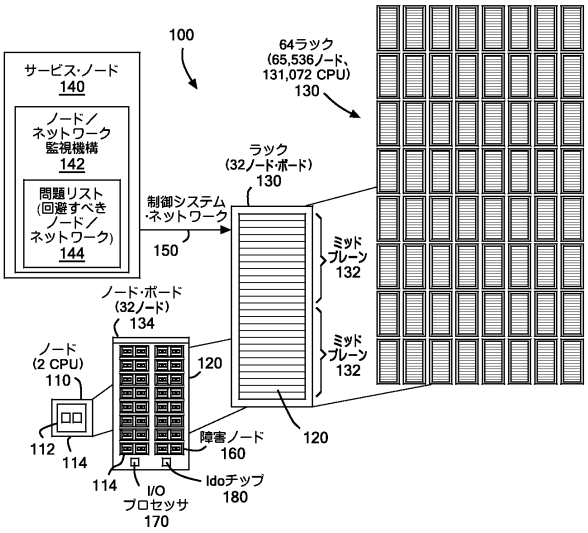
図７は、ノード・トラフィックを動的に再経路指定する他の例を示すために、図１の超並列コンピュータ・システムの部分７００を表すブロック図である。この例は、非隣接ノードについてのヒント・ビットの使用方法を示す。図６の例と同様に、この部分７００は、ノード１　６１０～ノード９　６１２と表記した９個のノードを例示する。この例では、障害又は潜在的な障害がノード８　６２２上で検出される。アプリケーションが一時停止され、ネットワークが休止され、そして障害ノード８６２２上のアプリケーションがスペア・ノード（ノード９６１２）に移動される。次に、図６の例と同様に、障害ノード８　６２２を通してデータを移動させる必要がある各ノードは、更新済みの問題リスト（図１の１４４）を影響を受けるノードに送信することにより、障害ノード８６２２を回避するように更新される。次に、データが障害ノード８　６２２を迂回して経路指定されることを保証するように、ヒント・ビットが設定される。図７の例では、ノード１　６１０からのデータ・パケットは、障害ノード８　６２２がＸ方向に存在しないという理由で、Ｘ方向用に設定されたヒント・ビットを有しないであろう。しかし、ノード１　６１０は、パケットがＹ＋方向に進行するように、設定済みのＹ＋ヒント・ビットを有するであろう。ノード１　６１０からのパケットがノード７６２８に到着し且つＹ次元内の進行を開始する場合、当該パケットは、設定済みのＹ＋ヒント・ビットによって指示されるようにＹ＋方向のノード９６１２に進行し、その結果、障害ノード８６２２を回避するであろう。

【００２９】

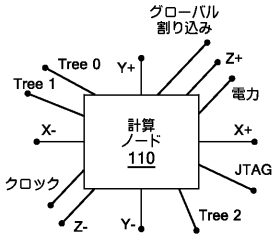
前述のように、ヒント・ビットは、輻輳ネットワークを迂回して動的に経路指定するためにも使用することができる。１例として、図７のノード８　６２２とノード５　６１８との間のネットワーク７１０を検討する。もし、ネットワーク７１０が、ネットワーク監視機構（図１の１４２）によって輻輳ネットワークとして標識付けされるならば、障害ノード８　６２２を迂回して経路指定するためにヒント・ビットを使用する前述の方法と同様に、このネットワークを迂回して動的に経路指定するためにヒント・ビットを使用することができる。代替的に、カットスルー・トラフィックによって、ノードの負担が過度に大きくなることがある。例えば、ネットワーク監視機構が、ノード８　６２２を通るカットスルー・トラフィックに起因して、ノード８　６２２が過負荷であると決定した場合、ノード８　６２２上で実行していたプロセス又はアプリケーションに関するカットスルー・トラフィック負荷を軽減するために、ノード８６２２上のプロセスは、利用可能な交換ノードに動的に再経路指定される。

【００３０】
図８は、並列コンピュータ・システム内の障害ノードのプロセスを動的に再経路指定するために、ノードを監視するための方法８００を示す。この方法は、サービス・ノード上のソフトウェアによって実施されるが、必要とされる情報を集めるためにソフトウェア及びハードウェアの一方又は両方を必要とすることがある。最初に、ネットワークを監視し（ステップ８１０）、ネットワーク・ホットスポットを問題リスト内に記録する（ステップ８２０）。次に、ノードを監視し（ステップ８３０）、障害ノード又は障害を生じる可能性があるノードを問題リスト内に記録する（ステップ８４０）。その後、この方法が終了する。
【００３１】
図９は、並列コンピュータ・システム内の障害ノードのプロセスを動的に再経路指定するための方法９００を示す。好ましくは、この方法は、並列コンピュータ・システムの各ノード上のソフトウェア及びハードウェアの一方又は両方によって実施される。最初に、サービス・ノード上のネットワーク監視機構によって送信され且つ回避すべきノード又はネットワークを保持する、更新済みの問題リストを検出する（ステップ９１０）。次に、障害ノードを有する並列システムの区画で実行中のアプリケーションを一時的に停止する（ステップ９２０）。次に、トーラス・ネットワーク・ハードウェアのＦＩＦＯがそれらのメッセージの送信を完了するまで待機することにより、ネットワークを休止する（ステップ９３０）。次に、ネットワーク用の交換ノード又は代替経路を見つけた後（ステップ９４０）、障害ノードのプロセスを交換ノードに移動させる（ステップ９５０）。次に、障害ノード又は障害ネットワークを通してネットワーク・トラフィックを送信するであろうノードに対し、ヒント・ビットを使用して、当該障害ノードを分離し且つ当該障害ノード又は輻輳ネットワークを迂回してネットワーク・トラフィックを経路指定するように通知する（ステップ９６０）。次に、アプリケーションをその一時的に停止した点から再開することができる（ステップ９７０）。その後、この方法が終了する。
【００３２】
本明細書は、超並列コンピュータ・システム上で実行中のアプリケーションを再始動することなく、ヒント・ビットを使用して、超並列コンピュータ・システムの計算ノード上のノード・トラフィックを動的に再経路指定するための装置及び方法を開示する。ノード・トラフィックを動的に再経路指定すると、故障時間の量を著しく減少させることができ、その結果、コンピュータ・システムの効率を向上させることができる。本方法は、コンピュータ・ソフトウェアの形態で実施することができる。
【００３３】
当業者には明らかなように、本発明の範囲内で多数の変形が可能である。本発明は特定の実施形態を参照して説明されたが、本発明の精神及び範囲から逸脱することなく、形式及び詳細に関する種々の変更を施し得ることは明らかであろう。
【符号の説明】
【００３４】
１００・・・超並列コンピュータ・システム
１１０・・・計算ノード
１４０・・・サービス・ノード
１４２・・・ノード／ネットワーク監視機構
１４４・・・問題リスト
１５０・・・制御システム・ネットワーク
３９２・・・トーラス・ネットワーク・ハードウェア
３９４・・・ツリー・ネットワーク・ハードウェア
３９６・・・グローバル割り込みネットワーク・ハードウェア
５１０・・・トーラス・ネットワーク・パケット
５１２・・・パケット・ヘッダ
５１６・・・ヒント・ビット

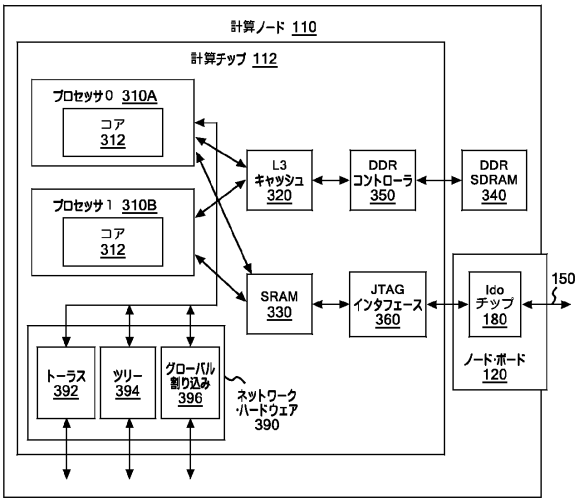
【図１】



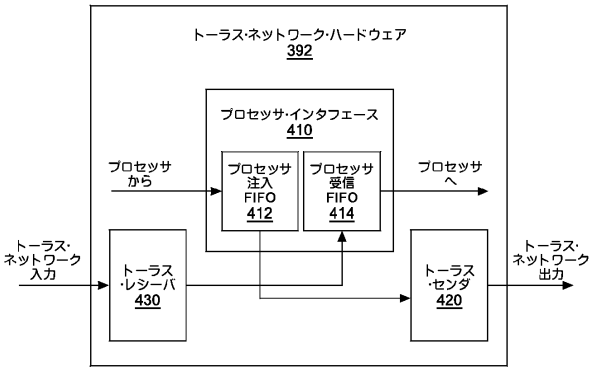
【図２】



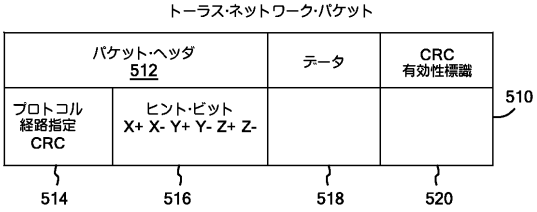
【図３】



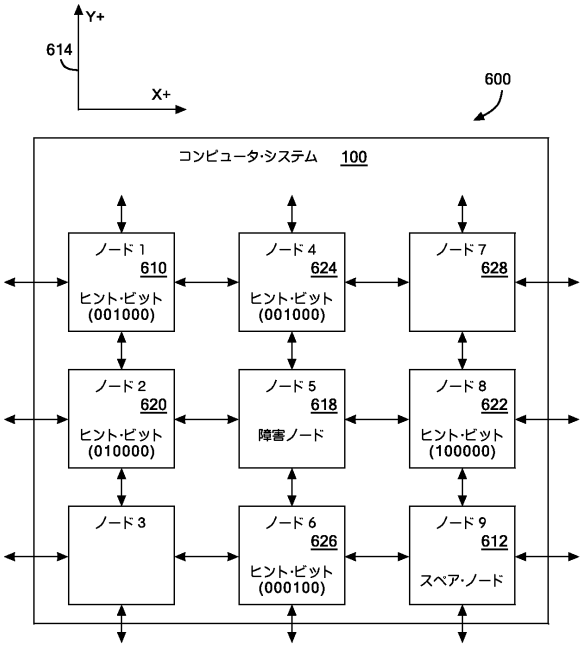
【図４】



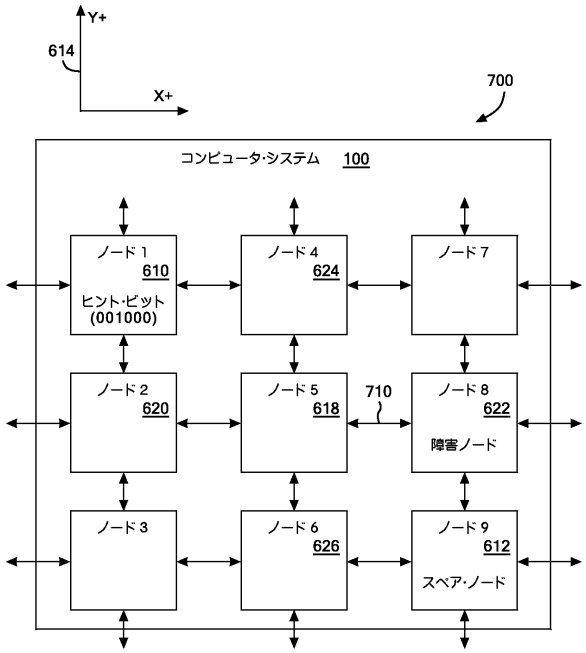
【図５】



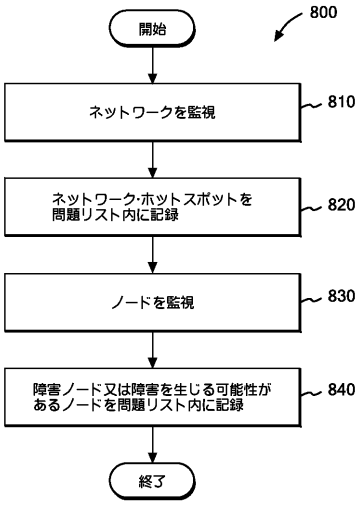
【図６】



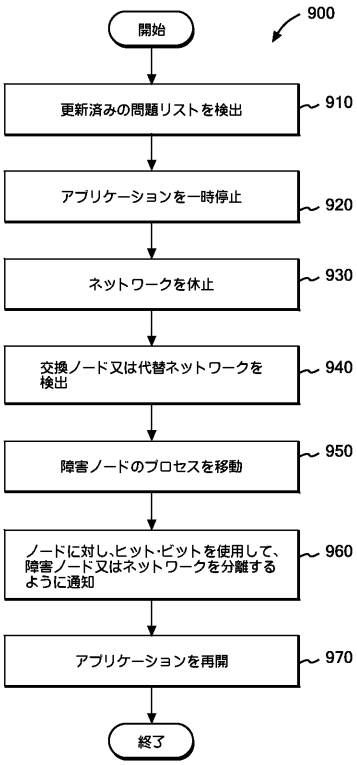
【図７】


Y+
614
X+
700
コンピュータ・システム 100
ノード１ 610
ヒント・ビット (001000)
ノード４ 624
ノード７
ノード２ 620
ノード５ 618
710
ノード８ 622
障害ノード
ノード３
ノード６ 626
ノード９ 612
スペア・ノード

【図８】


開始
800
ネットワークを監視
810
ネットワーク・ホットスポットを問題リスト内に記録
820
ノードを監視
830
障害ノード又は障害を生じる可能性があるノードを問題リスト内に記録
840
終了

【図９】


開始
900
更新済みの問題リストを検出
910
アプリケーションを一時停止
920
ネットワークを休止
930
交換ノード又は代替ネットワークを検出
940
障害ノードのプロセスを移動
950
ノードに対し、ヒット・ビットを使用して、障害ノード又はネットワークを分離するように通知
960
アプリケーションを再開
970
終了

フロントページの続き

(72)発明者 ピーターズ、アマンダ
アメリカ合衆国５５９０１ ミネソタ州 ロチェスター ニクラウス・ドライブ エヌ ダブリュー ５４１９
(72)発明者 シデルニク、アルバート
アメリカ合衆国６１８０１ イリノイ州 アーバナ サウス・リン・ストリート ４０２
(72)発明者 ダーリントン、デーヴィッド
アメリカ合衆国５５９０６ ミネソタ州 ロチェスター センチュリー・ヒルズ・ドライブ エヌ イー ２０４５
(72)発明者 マッカーシー、パトリック、ジョセフ
アメリカ合衆国５５９０１ ミネソタ州 ロチェスター ヴァルキュリヤ・コート エヌ ダブリュー ２１２９
(72)発明者 シュワルツ、ブレント、アラン
アメリカ合衆国５４７２９ ウイスコンシン州 チペワ・フォールズ 第６５アヴェニュー １９１４１
(72)発明者 スミス、ブライアン、エドワード
アメリカ合衆国５５９０１ ミネソタ州 ロチェスター ノル・レーン エヌ ダブリュー ３１２６

審査官 清木 泰

(56)参考文献 特開平０７－２２１７５４（ＪＰ，Ａ）
特開平０７－２３９８３５（ＪＰ，Ａ）
N.R.Adiga et al., Blue Gene/L torus interconnection network, IBM Journal of Research and Development, 米国, IBM, ２００５年 5月, Vol:49, No:2/3, Pages:265-276
Chao Wang, Frank Mueller, Christian Engelmann, Stephen L. Scott, A Job Pause Service under LAM/MPI+BLCR for Transparent Fault Tolerance, Proceedings of IEEE International Parallel and Distributed Processing Symposium 2007(IPDPS2007), IEEE, ２００７年 3月２６日, Pages:1-10
A.Gara et al., Overview of the Blue Gene/L system architecture, IBM Journal of Research and Development, 米国, IBM, ２００５年 5月, Vol:49, No:2/3, Pages:195-212
Jyothish Varma, Chao Wang, Frank Mueller, Christian Engelmann, Stephen L. Scott, Scalable, Fault-Tolerant Membership for MPI Tasks on HPC Systems, Proceedings of the 20th annual International Conference on Supercomputing (ICS2006), ACM, ２００６年 6月２８日, Pages:219-228
Yawei Li, Zhiling Lan, Exploit Failure Prediction for Adaptive Fault-Tolerance in Cluster Computing, Proceedings of Sixth IEEE International Symposium on Cluster Computing and the Grid (CCGRID 06), IEEE, ２００６年 5月１６日, all 8 pages
Daniel A. Reed, Charng-da Lu, Celso L. Mendes, Reliability challenges in large systems, Future Generation Computer Systems, NL, Elsevier Science Publishers B. V., ２００６年 2月, Volume:22, Issue:3, Pages:293-302

(58)調査した分野(Int.Cl., ＤＢ名)
Ｇ０６Ｆ１５／１６－１５／１７７
Ｇ０６Ｆ１５／８０

(54)【発明の名称】並列コンピュータ・システム、並列コンピュータ・システム上のノード・トラフィックを動的に再経路指定するためのコンピュータ実装方法、コンピュータ可読記録媒体及びコンピュータ・プログラム