

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2010-3156

(P2010-3156A)

(43) 公開日 平成22年1月7日(2010.1.7)

(51) Int.Cl.

G06F 19/00 (2006.01)

F I

G06F 19/00 110

テーマコード (参考)

審査請求 未請求 請求項の数 8 O L (全 14 頁)

(21) 出願番号 特願2008-162158 (P2008-162158)
(22) 出願日 平成20年6月20日 (2008.6.20)

(71) 出願人 000005496
富士ゼロックス株式会社
東京都港区赤坂九丁目7番3号
(74) 代理人 100104880
弁理士 古部 次郎
(74) 代理人 100118201
弁理士 千田 武
(74) 代理人 100118108
弁理士 久保 洋之
(72) 発明者 世古 丈裕
神奈川県足柄上郡中井町境430 グリー
ンテクノカ い 富士ゼロックス株式会社内
(72) 発明者 渡辺 洋一
神奈川県足柄上郡中井町境430 グリー
ンテクノカ い 富士ゼロックス株式会社内
最終頁に続く

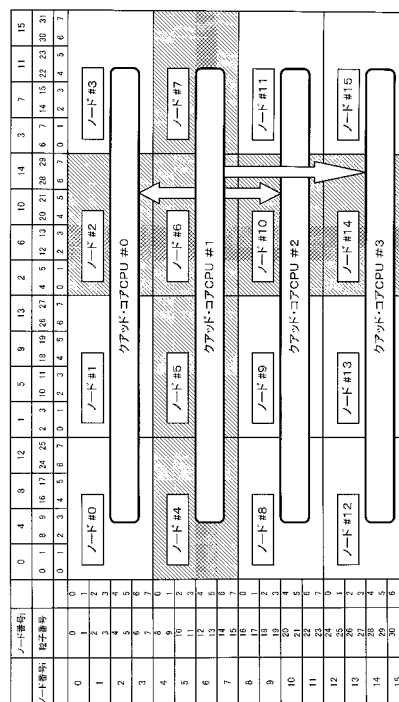
(54) 【発明の名称】 粒子挙動解析装置

(57) 【要約】

【課題】マルチコア・プロセッサを搭載したコンピュータによる粒子の挙動解析の並列分散処理において、力分割法による並列分散処理の実行速度を向上させる。

【解決手段】各々が複数のプロセッサコア（ノード）を持ち、並列分散処理アルゴリズムを用いてプロセッサコアごとに計算対象が割り当てられた複数の計算機を備える。この計算機は、プロセッサコアごとに、粒子間に働く相互作用力を計算する第1計算部と、計算された相互作用力に基づいて個々の粒子の挙動を計算する第2計算部とを備える。個々のプロセッサコアは、並列分散処理アルゴリズムにより、第2計算部の計算対象として、挙動を計算する対象である粒子が割り当てられる。また、第1計算部の計算対象として、相互作用力を計算する対象である粒子の組み合わせであって、同じ計算機上のプロセッサコアの第2計算部に割り当てられている粒子の組み合わせが割り当てられる。

【選択図】図6



【特許請求の範囲】**【請求項 1】**

各々が複数のプロセッサコアを持ち、相互に接続され、並列分散処理アルゴリズムを用いて当該プロセッサコアごとに計算対象が割り当てられた複数の計算機を備え、

前記計算機は、前記プロセッサコアごとに、

粒子間に働く相互作用力を計算する第 1 計算部と、

前記第 1 計算部により計算された前記相互作用力に基づいて個々の粒子の挙動を計算する第 2 計算部とを備え、

個々の前記プロセッサコアは、

前記並列分散処理アルゴリズムにより、前記第 2 計算部の計算対象として、挙動を計算する対象である粒子が割り当てられ、

前記並列分散処理アルゴリズムにより、前記第 1 計算部の計算対象として、相互作用力を計算する対象である粒子の組み合わせであって、同じ前記計算機上の前記プロセッサコアの前記第 2 計算部に割り当てられている粒子の組み合わせが割り当てられることを特徴とする粒子挙動解析装置。

【請求項 2】

前記並列分散処理アルゴリズムは、行および列の一方に相互作用力を受ける粒子を割り当て、他方に当該相互作用力を作用させる粒子を割り当て、各粒子の行と列の交点を含む領域に当該相互作用力を計算する前記プロセッサコアを割り当てた力マトリクスに基づくことを特徴とする請求項 1 に記載の粒子挙動解析装置。

【請求項 3】

前記計算機は、自身の個々の前記プロセッサコアにおける前記第 1 計算部による相互作用力の計算結果を、前記第 2 計算部に対する割り当てにしたがって、各粒子が割り当てられたプロセッサコアに収集し、各当該プロセッサコアにおいて、収集した計算結果の総和値を用いて運動方程式を解き、粒子の挙動を計算することを特徴とする請求項 1 に記載の粒子挙動解析装置。

【請求項 4】

粒子間に働く相互作用力および個々の粒子の挙動を、並列分散処理アルゴリズムによる割り当てにしたがって、並列分散処理により計算する複数の計算ノードを備え、

前記計算ノードは、

複数のプロセッサコアを持つ計算機における個々の当該プロセッサコアであり、

粒子間に働く相互作用力を計算する第 1 計算部と、

前記第 1 計算部により計算された前記相互作用力に基づいて個々の粒子の挙動を計算する第 2 計算部とを備え、

個々の前記計算ノードは、

前記並列分散処理アルゴリズムにより、前記第 2 計算部の計算対象として、挙動を計算する対象である粒子が割り当てられ、

前記並列分散処理アルゴリズムにより、前記第 1 計算部の計算対象として、相互作用力を計算する対象である粒子の組み合わせであって、同じ前記計算機上の前記プロセッサコアである前記計算ノードの前記第 2 計算部に割り当てられている粒子の組み合わせが割り当てられる

ことを特徴とする粒子挙動解析装置。

【請求項 5】

前記並列分散処理アルゴリズムは、行および列の一方に前記相互作用力を受ける粒子を割り当て、他方に当該相互作用力を作用させる粒子を割り当て、各粒子の行と列の交点を含む領域に当該相互作用力を計算する前記プロセッサコアを割り当てた力マトリクスに基づくことを特徴とする請求項 4 に記載の粒子挙動解析装置。

【請求項 6】

前記計算ノードである前記プロセッサコアが設けられる前記計算機は、自身のプロセッサコアである個々の当該計算ノードにおける前記第 1 計算部による前記相互作用力の計算

10

20

30

40

50

結果を、前記第 2 計算部に対する割り当てにしたがって、各粒子が割り当てられた計算ノードに収集し、各当該計算ノードにおいて、収集した計算結果の総和値を用いて運動方程式を解き、粒子の挙動を計算することを特徴とする請求項 4 に記載の粒子挙動解析装置。

【請求項 7】

各々が複数のプロセッサコアを持ち、相互に接続され、並列分散処理アルゴリズムを用いて当該プロセッサコアごとに計算対象が割り当てられた複数の計算機を備え、

前記計算機は、前記プロセッサコアごとに、

粒子間に働く相互作用力を計算する第 1 計算部と、

前記第 1 計算部により計算された前記相互作用力に基づいて個々の粒子の挙動を計算する第 2 計算部とを備え、

10

個々の前記プロセッサコアは、

前記並列分散処理アルゴリズムにより、前記第 2 計算部の計算対象として、挙動を計算する対象である粒子が割り当てられ、

前記並列分散処理アルゴリズムにより、前記第 1 計算部の計算対象として、行および列の一方に前記相互作用力を受ける粒子を割り当て、他方に当該相互作用力を作用させる粒子を割り当て、各粒子の行と列の交点を含む領域に当該相互作用力を計算する前記プロセッサコアを割り当てた力マトリクスに基づいて、当該力マトリクスにおける当該相互作用力を受ける粒子を登録した行または列に沿って、同じ前記計算機上の前記プロセッサコアが並ぶように、当該相互作用力を計算する対象である粒子の組み合わせが割り当てられることを特徴とする粒子挙動解析装置。

20

【請求項 8】

各前記プロセッサコアは、前記相互作用力の計算結果を、前記力マトリクスにおける当該相互作用力を受ける粒子を割り当てた行または列に沿って収集し、各当該プロセッサコアにおいて、収集した計算結果の総和値を用いて運動方程式を解き、粒子の挙動を計算することを特徴とする請求項 7 に記載の粒子挙動解析装置。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、粒子の挙動を解析しシミュレーションする装置に関する。

【背景技術】

30

【0002】

粒子の挙動をシミュレーションする解析処理では、粒子の数が増えると計算量が膨大になる。そのため、一般に、複数台の計算機（コンピュータ）を用いた並列分散処理による解析が行われている。並列分散処理による粒子の挙動解析の手法としては、粒子分割法、領域分割法、力分割法等の種々の手法がある。

【0003】

特許文献 1 に記載された従来技術は、力分割法を用いた並列分散処理による解析手法である。この従来技術は、各粒子の磁気力、静電気力、接触力の各粒子間相互作用について、それぞれ各別の力マトリクスを用いて、特定プロセッサに分散して計算、特定プロセッサ間で通信し分散して計算した相互作用力の和を求め、各粒子の運動方程式を解いて位置座標を計算する。そして、各粒子の位置座標を特定プロセッサに通信し、計算情報を更新する。この処理を、所定の計算ステップに到達するまで繰り返す。

40

【0004】

【特許文献 1】特開 2007 - 193152 号公報

【発明の開示】

【発明が解決しようとする課題】

【0005】

本発明の目的は、マルチコア・プロセッサを搭載したコンピュータによる粒子の挙動解析の並列分散処理において、力分割法による並列分散処理の実行速度を向上させることにある。

50

【課題を解決するための手段】

【0006】

請求項1に記載の発明は、各々が複数のプロセッサコアを持ち、相互に接続され、並列分散処理アルゴリズムを用いて当該プロセッサコアごとに計算対象が割り当てられた複数の計算機を備え、前記計算機は、前記プロセッサコアごとに、粒子間に働く相互作用力を計算する第1計算部と、前記第1計算部により計算された前記相互作用力に基づいて個々の粒子の挙動を計算する第2計算部とを備え、個々の前記プロセッサコアは、前記並列分散処理アルゴリズムにより、前記第2計算部の計算対象として、挙動を計算する対象である粒子が割り当てられ、前記並列分散処理アルゴリズムにより、前記第1計算部の計算対象として、相互作用力を計算する対象である粒子の組み合わせであって、同じ前記計算機上の前記プロセッサコアの前記第2計算部に割り当てられている粒子の組み合わせが割り当てられることを特徴とする粒子挙動解析装置である。

10

請求項2に記載の発明は、前記並列分散処理アルゴリズムは、行および列の一方に相互作用力を受ける粒子を割り当て、他方に当該相互作用力を作用させる粒子を割り当て、各粒子の行と列の交点を含む領域に当該相互作用力を計算する前記プロセッサコアを割り当てた力マトリクスに基づくことを特徴とする請求項1に記載の粒子挙動解析装置である。

請求項3に記載の発明は、前記計算機は、自身の個々の前記プロセッサコアにおける前記第1計算部による相互作用力の計算結果を、前記第2計算部に対する割り当てにしたがって、各粒子が割り当てられたプロセッサコアに収集し、各当該プロセッサコアにおいて、収集した計算結果の総和値を用いて運動方程式を解き、粒子の挙動を計算することを特徴とする請求項1に記載の粒子挙動解析装置である。

20

請求項4に記載の発明は、粒子間に働く相互作用力および個々の粒子の挙動を、並列分散処理アルゴリズムによる割り当てにしたがって、並列分散処理により計算する複数の計算ノードを備え、前記計算ノードは、複数のプロセッサコアを持つ計算機における個々の当該プロセッサコアであり、粒子間に働く相互作用力を計算する第1計算部と、前記第1計算部により計算された前記相互作用力に基づいて個々の粒子の挙動を計算する第2計算部とを備え、個々の前記計算ノードは、前記並列分散処理アルゴリズムにより、前記第2計算部の計算対象として、挙動を計算する対象である粒子が割り当てられ、前記並列分散処理アルゴリズムにより、前記第1計算部の計算対象として、相互作用力を計算する対象である粒子の組み合わせであって、同じ前記計算機上の前記プロセッサコアである前記計算ノードの前記第2計算部に割り当てられている粒子の組み合わせが割り当てられることを特徴とする粒子挙動解析装置である。

30

請求項5に記載の発明は、前記並列分散処理アルゴリズムは、行および列の一方に前記相互作用力を受ける粒子を割り当て、他方に当該相互作用力を作用させる粒子を割り当て、各粒子の行と列の交点を含む領域に当該相互作用力を計算する前記プロセッサコアを割り当てた力マトリクスに基づくことを特徴とする請求項4に記載の粒子挙動解析装置である。

請求項6に記載の発明は、前記計算ノードである前記プロセッサコアが設けられる前記計算機は、自身のプロセッサコアである個々の当該計算ノードにおける前記第1計算部による相互作用力の計算結果を、前記第2計算部に対する割り当てにしたがって、各粒子が割り当てられた当該計算ノードに収集し、各当該計算ノードにおいて、収集した計算結果の総和値を用いて運動方程式を解き、粒子の挙動を計算することを特徴とする請求項4に記載の粒子挙動解析装置である。

40

請求項7に記載の発明は、各々が複数のプロセッサコアを持ち、相互に接続され、並列分散処理アルゴリズムを用いて当該プロセッサコアごとに計算対象が割り当てられた複数の計算機を備え、前記計算機は、前記プロセッサコアごとに、粒子間に働く相互作用力を計算する第1計算部と、前記第1計算部により計算された前記相互作用力に基づいて個々の粒子の挙動を計算する第2計算部とを備え、個々の前記プロセッサコアは、前記並列分散処理アルゴリズムにより、前記第2計算部の計算対象として、挙動を計算する対象である粒子が割り当てられ、前記並列分散処理アルゴリズムにより、前記第1計算部の計算対

50

象として、行および列の一方に前記相互作用力を受ける粒子を割り当て、他方に当該相互作用力を受作用させる粒子を割り当て、各粒子の行と列の交点を含む領域に当該相互作用力を計算する前記プロセッサコアを割り当てた力マトリクスに基づいて、当該力マトリクスにおける当該相互作用力を受ける粒子を登録した行または列に沿って、同じ前記計算機上の前記プロセッサコアが並ぶように、当該相互作用力を計算する対象である粒子の組み合わせが割り当てられることを特徴とする粒子挙動解析装置である。

請求項 8 に記載の発明は、各前記プロセッサコアは、相互作用力の計算結果を、前記力マトリクスにおける相互作用力を受ける粒子を割り当てた行または列に沿って収集し、各当該プロセッサコアにおいて、収集した計算結果の総和値を用いて運動方程式を解き、粒子の挙動を計算することを特徴とする請求項 7 に記載の粒子挙動解析装置である。

10

【発明の効果】

【0007】

請求項 1 の発明によれば、同じ計算機上のプロセッサコア間で相互作用力の計算結果をやり取りすることにより、プロセッサ間通信を削減し、並列分散処理の実行速度を向上させることができる。

請求項 2 の発明によれば、力マトリクスを用いることにより、同じ計算機上のプロセッサコア間で相互作用力の計算結果がやり取りされるように、計算機への割り当てを行うことができる。

請求項 3 の発明によれば、各計算機が自身のプロセッサコア間で相互作用力の計算結果をやり取りすることにより、プロセッサ間通信を削減することができる。

20

請求項 4 の発明によれば、同じ計算機上のプロセッサコアである計算ノード間で相互作用力の計算結果をやり取りすることにより、プロセッサ間通信を削減することができる。

請求項 5 の発明によれば、力マトリクスを用いることにより、同じ計算機上の計算ノード間で相互作用力の計算結果がやり取りされるように、計算機への割り当てを行うことができる。

請求項 6 の発明によれば、各計算機が自身に搭載された計算ノード間で相互作用力の計算結果をやり取りすることにより、プロセッサ間通信を削減することができる。

請求項 7 の発明によれば、力マトリクスを用いて同じ計算機上のプロセッサコアである計算ノード間で相互作用力の計算結果をやり取りするように各計算ノードへの割り当てを行うことにより、プロセッサ間通信を削減することができる。

30

請求項 8 の発明によれば、各計算機が自身に搭載された計算ノード間で相互作用力の計算結果をやり取りすることにより、プロセッサ間通信を削減することができる。

【発明を実施するための最良の形態】

【0008】

以下、添付図面を参照して、本発明の実施形態について詳細に説明する。

< システム構成 >

図 1 は、本実施形態が適用される並列計算システムの全体構成を示す図である。

図 1 に示す本実施形態のシステムは、複数台の計算ノード 110 を備えて構成される。各計算ノード 110 は、ネットワーク接続されており、いわゆるクラスタコンピューティングを実現する。また各ノードは、パーソナルコンピュータ等のコンピュータ（計算機）で実現される。また、特に図示しないが、複数の計算ノード 110 を統括的に制御し、解析を開始させたり、最終的な解析結果を取得して保存し、または出力したりする制御装置が、必要に応じて設けられる。

40

【0009】

図 2 は、計算ノード 110 を実現するコンピュータのハードウェア構成例を示す図である。

図 2 に示すコンピュータ 10 は、演算手段である CPU（Central Processing Unit）10a と、記憶手段である主記憶装置（メインメモリ）10b および外部記憶装置 10c を備える。CPU 10a としては、クアッド・コア（quad-core）やデュアル・コア（dual core）等のように複数のプロセッサコアを集積したマルチコア・プロセッサ（multi-co

50

re processor) が用いられる。外部記憶装置 10c としては、一般に磁気ディスク装置 (HDD: Hard Disk Drive) が用いられるが、より高速にデータの読み書きを行うことができる SSD (Solid State Drive) を用いても良い。また、図 2 のコンピュータ 10 は、ネットワークを介して外部装置に接続するためのネットワーク I/F (インターフェイス) 10d と、ディスプレイ装置へ表示出力を行うための表示機構 10e と、音声出力を行うための音声機構 10f とを備える。さらに、キーボードやマウス等の入力デバイス 10g を備える。CPU 10a と他の構成要素との間には、図示しないチップセットやブリッジ回路が介在している。

【0010】

図 2 において、各構成要素は、システムバスや入出力バス等の各種のバスを介して接続される。例えば、CPU 10a と主記憶装置 10b の間は、システムバスやメモリバスを介して接続される。また、CPU 10a と外部記憶装置 10c、ネットワーク I/F 10d、表示機構 10e、音声機構 10f、入力デバイス 10g 等との間には、PCI (Peripheral Components Interconnect)、PCI Express、シリアル ATA (AT Attachment)、USB (Universal Serial Bus)、AGP (Accelerated Graphics Port) 等の入出力バスを介して接続される。

【0011】

なお、図 2 は、並列計算システムを構成する計算ノード 110 を実現するのに好適なコンピュータのハードウェア構成を例示するに過ぎず、図示の構成に限定されないことは言うまでもない。例えば、計算ノード 110 の補助記憶装置として、外部記憶装置 10c の他に、フレキシブルディスクや光学ディスクをメディアとするドライブを設けたり、USB メモリを設けたりしても良い。USB メモリは、USB を介してブリッジ回路に接続されることとなる。また、音声機構 10f を独立した構成とせず、チップセットの機能として備えても良い。

【0012】

詳しくは後述するが、CPU 10a としてマルチコア・プロセッサを用いる場合、個々のプロセッサコアは、各々、個別の計算ノード 110 として独立に動作する。この場合、図 2 に示すハードウェア構成は、複数の計算ノード 110 を実現するコンピュータの構成として把握される。

【0013】

< ノードの機能 >

図 3 は、計算ノード 110 の機能構成を示す図である。

計算ノード 110 は、自ノードに割り当てられた解析対象に対する個々の処理を実行する装置である。図 3 に示すように、各計算ノード 110 は、それぞれ、並列分散処理アルゴリズムによる制御を行うための並列分散処理制御部 120 と、解析対象のデータや解析結果を保持する保持部 121 と、解析処理のための計算を行う第 1 計算部 122 および第 2 計算部 123 と、他の計算ノード 110 との間でデータの送受信を行うための通信部 124 とを備える。並列分散処理制御部 120、第 1 計算部 122 および第 2 計算部 123 は、主記憶装置 10b に読み込まれたプログラムを CPU 10a が実行することで実現される機能である。保持部 121 は、例えば図 2 に示したコンピュータ 10 において、主記憶装置 10b や外部記憶装置 10c により実現される。通信部 124 は、主記憶装置 10b に読み込まれたプログラムを CPU 10a が実行しネットワーク I/F 10d を制御することで実現される。このように、図 3 に示す計算ノード 110 の各機能ブロックは、ソフトウェアとハードウェア資源とが協働して実現される手段である。

【0014】

並列分散処理制御部 120 は、並列分散処理アルゴリズムにより、分散処理における自ノードの動作を制御する。本実施形態で用いられる並列分散処理アルゴリズムは、カマトリクスと呼ぶデータ構造に基づいて各計算ノード 110 へのデータの割り当てを行う。本実施形態では、相互作用力を計算する対象である粒子の組み合わせが第 1 計算部 122 の計算対象として割り当てられ、挙動を計算する対象である粒子が第 2 計算部 123 の計算

10

20

30

40

50

対象として割り当てられる。並列分散処理制御部 120 は、この並列分散処理アルゴリズムによる割り当てに基づいて、自ノードに割り当てられた計算対象のデータを取得し、保持部 121 に保持させる。また、並列分散処理制御部 120 は、この並列分散処理アルゴリズムによる割り当てに基づいて、自ノードにおける計算結果を他の計算ノード 110 とやり取りする。力マトリクスおよび力マトリクスに基づくデータの割り当て方法の詳細については後述する。

【0015】

保持部 121 は、並列分散処理制御部 120 の制御により、自ノードの計算に必要なデータを取得して保持する。また、第 1 計算部 122 および第 2 計算部 123 による計算結果を保持する。ここで、自ノードの計算に必要なデータとは、並列分散処理アルゴリズムにより自ノードに割り当てられた粒子のデータと、力マトリクスに基づく分散処理により相互作用力を計算するために用いられるデータである。これらのデータは、自ノードに割り当てられた粒子の初期データが外部から取得された後は、計算ノード 110 どうしの間で相互にやり取りされる。

【0016】

第 1 計算部 122 は、粒子間に働く相互作用力を計算する。解析対象の粒子の材質等によって計算すべき相互作用の種類は異なるが、通常は、磁気相互作用、静電相互作用、機械的相互作用（壁などの物体と粒子との間の接触力や粒子間接触の接触力）などが計算される。この相互作用力の計算は、力マトリクスに基づいて複数の計算ノード 110 に分散されて実行される。したがって、第 1 計算部 122 は、この分散によって自ノードが担当する粒子間の相互作用力を計算することとなる。

【0017】

第 2 計算部 123 は、粒子の挙動を計算する。すなわち、第 1 計算部 122 により計算された相互作用力と粒子の現在位置および属性に基づいて、粒子の位置がどのように変化するかを計算する。粒子の属性としては、例えば大きさ、質量、速度、電荷、磁化などが挙げられる。

【0018】

通信部 124 は、並列分散処理制御部 120 の制御により、自ノードに割り当てられた粒子の初期データを受信して保持部 121 に保持させる。また、通信部 124 は、並列分散処理制御部 120 の制御により、他の計算ノード 110 との間で第 1 計算部 122 および第 2 計算部 123 による計算に必要なデータを相互に送受信する。具体的には、自ノードの第 1 計算部 122 および第 2 計算部 123 による計算に必要なデータを他ノードから受信する。そして、保持部 121 に保持されている第 1 計算部 122 および第 2 計算部 123 の計算結果をその計算結果が必要な他ノードに送信する。

【0019】

<力マトリクスに基づく粒子の割り当ておよび計算>

力分割法では、粒子間に働く相互作用力の計算が分散処理により実行される。本実施形態では、この粒子間に働く相互作用力の計算とは別に、各粒子の挙動の計算についても分散処理により実行される。すなわち、本実施形態では、粒子間に働く相互作用力の計算と各粒子の挙動の計算とが個別のアルゴリズムにより分散処理される。したがって、力分割法による並列分散処理では、各計算ノード 110 に対し、挙動計算の対象である粒子が割り当てられ、かつ相互作用力の計算の対象である粒子の組み合わせが割り当てられる。

【0020】

この力分割法では、各計算ノード 110 は、まず、自ノードに割り当てられた粒子の組み合わせに関して、その粒子間に働く相互作用力を計算する。そして、粒子ごとの相互作用力の計算結果が、個々の粒子が割り当てられている各計算ノード 110 に集められる。次に、各計算ノード 110 は、取得した自ノードに割り当てられている粒子に働く相互作用力の総和を計算し、この総和値を用いて、この粒子の挙動を計算する。

【0021】

ここで、各計算ノード 110 に対する、挙動計算の対象である粒子の割り当て、および

10

20

30

40

50

相互作用力の計算の対象である粒子の組み合わせの割り当ては、力マトリクスに基づいて行われる。力マトリクスとは、各計算ノード 1 1 0 と挙動計算の対象である粒子との対応関係（割り当て）、および各計算ノード 1 1 0 と相互作用力の計算の対象である粒子の組み合わせとの対応関係（割り当て）を記述するデータ構造である。

【 0 0 2 2 】

図 4 は、力マトリクスの例を示す図である。

図 4 に示す力マトリクスは、3 2 個の粒子（粒子番号 0 ~ 3 1）を 1 6 の計算ノード 1 1 0（ノード番号 0 ~ 1 5）に割り当てた例を示している。例えば、ノード番号 0 の計算ノード 1 1 0 には粒子番号 0、1 の 2 つの粒子が割り当てられている。力マトリクスでは、行および列の一方に相互作用力を受ける粒子が登録され、他方に当該相互作用力を作用させる粒子が登録され、各粒子の行と列の交点を含む領域に当該相互作用力を計算する計算ノードが登録される。

【 0 0 2 3 】

図 4 の力マトリクスにおいて、縦方向のノード番号 i および粒子番号と横方向のノード番号 j および粒子番号は、相互作用力を計算する際の粒子の組み合わせを示す。より詳細には、ノード番号 i に割り当てられた粒子がノード番号 j に割り当てられた粒子から受ける相互作用力が計算される。そして、縦横ノード 4 個分のマス目は、該当する組み合わせの粒子の相互作用力を計算する計算ノード 1 1 0 を示す。具体的には、例えばノード # 6（ノード番号 6 の計算ノード 1 1 0）は、

- ・ノード番号 4 に割り当てられた 2 個の粒子とノード番号 2、6、1 0、1 4 に割り当てられた 8 個の粒子の相互作用力、
- ・ノード番号 5 に割り当てられた 2 個の粒子とノード番号 2、6、1 0、1 4 に割り当てられた 8 個の粒子の相互作用力、
- ・ノード番号 6 に割り当てられた 2 個の粒子とノード番号 2、6、1 0、1 4 に割り当てられた 8 個の粒子の相互作用力、
- ・ノード番号 7 に割り当てられた 2 個の粒子とノード番号 2、6、1 0、1 4 に割り当てられた 8 個の粒子の相互作用力、

をそれぞれ計算する。このノード # 6 において、ノード番号 4 に割り当てられた粒子番号 8、9 の粒子とノード番号 2 に割り当てられた粒子番号 4、5 の粒子の組み合わせでは、

- ・粒子番号 8 の粒子に対して働く、粒子番号 4 の粒子による相互作用力、
- ・粒子番号 8 の粒子に対して働く、粒子番号 5 の粒子による相互作用力、
- ・粒子番号 9 の粒子に対して働く、粒子番号 4 の粒子による相互作用力、
- ・粒子番号 9 の粒子に対して働く、粒子番号 5 の粒子による相互作用力、

の 4 種類の相互作用力が計算される。したがって、ノード番号の組み合わせごとに 4 種類ずつの相互作用力が計算される。ただし、ノード番号 6 とノード番号 6 の組み合わせは 2 個の粒子が同一なので、粒子番号 1 2 の粒子に対して働く粒子番号 1 3 の粒子による相互作用力と、粒子番号 1 3 の粒子に対して働く粒子番号 1 2 の粒子による相互作用力の 2 つの相互作用力のみが求められる。すなわち、個々の計算ノード 1 1 0 は、6 2（ $= 8 \times 8 - 2$ ）組の粒子の組み合わせについて相互作用力の計算を行うこととなる。

【 0 0 2 4 】

図 5 は、ノード # 6 が相互作用力の計算を担当する粒子の組み合わせを示す。

なお、ノード番号 6 とノード番号 6 の組み合わせにおける 2 つの粒子間に働く相互作用力の大きさは同じ（反対方向）であるから、力の大きさの計算は 1 回だけ行うようにしても良い。

【 0 0 2 5 】

上記のように相互作用力の計算についての分散処理が行われた後、計算ノード 1 1 0 間で通信が行われ、各粒子に関する相互作用力の計算結果がその粒子を割り当てられた計算ノード 1 1 0 に集められる。図 4 の力マトリクスを参照すると、例えば、ノード番号 6 の計算ノード 1 1 0 に割り当てられている粒子番号 1 2、1 3 の粒子に対して働く相互作用力は、ノード # 4 ~ # 7 の 4 つの計算ノード 1 1 0 により計算されている。同様に、ノード

10

20

30

40

50

ド番号4の計算ノード110に割り当てられている粒子番号8、9の粒子に対して働く相互作用力、ノード番号5の計算ノード110に割り当てられている粒子番号10、11の粒子に対して働く相互作用力、ノード番号7の計算ノード110に割り当てられている粒子番号14、15の粒子に対して働く相互作用力は、いずれもノード#4～#7の4つの計算ノード110により計算されている。したがって、図4の力マトリクスにおいて行方向（横方向）に並ぶ4つの計算ノード110（ノード#4～#7）の間で通信を行うことで、各計算ノード110に、自ノードに割り当てられた粒子に関する相互作用力の計算結果が集まることとなる。図4の力マトリクスにおける他の行方向に並ぶ4つの計算ノード110の組（ノード#0～#3、ノード#8～#11、ノード#12～#15）についても同様である。

10

【0026】

各計算ノード110は、上記のようにして自ノードに割り当てられた粒子に関する相互作用力の計算結果を収集し、収集した計算結果の総和値を用いて運動方程式を解き、粒子の挙動（移動後の位置）を計算する。そして、計算結果を、次の相互作用力の計算サイクルにおいて自ノードに割り当てられた粒子の情報を必要とする他の計算ノード110に配信する。例えば、ノード#6で計算された粒子番号12、13の粒子の情報は、自ノード（ノード#6）と共にこれらの粒子に対して働く相互作用力を計算するノード#4、#5、#7と、これらの粒子から他の粒子に対して働く相互作用力を計算するノード#2、#10、#14により必要とされる。したがって、図4の力マトリクスにおいて、ノード#6を基準として、行方向に並ぶ他の計算ノード110（ノード#4、#5、#7）と、列方向（縦方向）に並ぶ他の計算ノード110（ノード#2、#10、#14）に、ノード#6の計算結果が配信される。同様に、各計算ノード110は、図4の力マトリクスにおいて、自ノードを基準として行方向に並ぶ他の計算ノード110および列方向に並ぶ他の計算ノード110に対して、自ノードの計算結果を配信する。

20

【0027】

以上のように、力マトリクスを用いた力分割法による分散処理によれば、各計算ノード110は、自ノードを基準として、力マトリクスの行方向に並ぶ他の計算ノード110と列方向に並ぶ他の計算ノード110との間でのみ通信が行われる。図4に示した力マトリクスでは、各計算ノード110は、行方向に並ぶ3つと列方向に並ぶ3つの合計6つの計算ノード110とのみ通信を行えば良い。そのため、分散処理全体における通信に要するコストの割合が低減する。

30

【0028】

<マルチコア・プロセッサにおける粒子の割り当て>

今日、1つのCPUパッケージ内に複数のプロセッサコアを備えたマルチコア・プロセッサが存在する。マルチコア・プロセッサの各プロセッサコアは独立して動作するため、各々のプロセッサコアを、別個の計算ノード110として用いて良い。このマルチコア・プロセッサでは、多くの場合、複数のプロセッサコアが2次キャッシュ等のキャッシュメモリやメインメモリを共有する構成が採られる。この場合、1つのマルチコア・プロセッサ上のプロセッサコア間でデータ交換する場合、共有するメモリへの読み書きを介してデータの受け渡しが行われ、プロセッサ間通信を行う必要がない。この特徴を利用して、マルチコア・プロセッサにおけるプロセッサコアを力マトリクスのマス目に適切に配置すれば、分散処理における通信コストがさらに低減する。

40

【0029】

具体的には、計算ノード110である各コアプロセッサに、挙動計算の対象である粒子を割り当て、かつ同じマルチコア・プロセッサ上の複数のプロセッサコアに、そのプロセッサコアに割り当てられた粒子に作用する相互作用力の計算対象である粒子の組み合わせを割り当てる。力マトリクスに基づいて表現すれば、相互作用力を受ける粒子を登録した行（または列）に沿って、同じマルチコア・プロセッサ上の複数のプロセッサコアが並ぶように、相互作用力の計算対象である粒子の組み合わせが割り当てられる。以下、4個のコアプロセッサが搭載されたクアッド・コアと2個のコアプロセッサが搭載されたデュア

50

ル・コアを例として、効率の良い力マトリクスのレイアウトについて説明する。

【 0 0 3 0 】

クアッド・コアは、1つのマルチコア・プロセッサ（以下、クアッド・コアCPU）上に4個のプロセッサコアを備える。そこで、この4個のプロセッサコアを、それぞれ1つの計算ノード110に割り当てる。これにより、1つのクアッド・コアCPUを持つコンピュータ（例えば図2のコンピュータ10）が4つの計算ノード110に対応する。例えば、図4に示した力マトリクスでは、16個の計算ノード110が用いられているので、4つのクアッド・コアCPUで16個の計算ノード110全てに対応する。

【 0 0 3 1 】

図6は、クアッド・コアCPUを用いる場合の力マトリクスのレイアウトの例を説明する図である。

図6に示す例では、力マトリクスの行方向に並ぶ4つの計算ノード110に、1つのクアッド・コアCPU上のプロセッサコアが割り当てられている。具体的には、クアッド・コアCPU#0のプロセッサコアがノード#0～#3に割り当てられ、クアッド・コアCPU#1のプロセッサコアがノード#4～#7に割り当てられ、クアッド・コアCPU#2のプロセッサコアがノード#8～#11に割り当てられ、クアッド・コアCPU#3のプロセッサコアがノード#12～#15に割り当てられている。

【 0 0 3 2 】

このように構成すると、相互作用力の計算結果を各粒子が割り当てられた計算ノード110に集める際のデータのやり取りにおいて、プロセッサ間通信を行う必要がなくなる。図4を参照して説明したように、相互作用力の計算結果は、力マトリクスの行方向に並ぶ計算ノード110の間でのみやり取りされるからである。また、各計算ノード110による粒子の挙動計算の結果を他の計算ノード110に配信する際のデータのやり取りにおいても、計算結果を力マトリクスの行方向に並ぶ計算ノード110に渡す場合には、プロセッサ間通信を必要としない。

【 0 0 3 3 】

したがって、図6のようにプロセッサコアが割り当てられたクアッド・コアCPUは、各計算ノード110による粒子の挙動計算の結果を他の計算ノード110に配信する際のデータのやり取りにおいて、計算結果を力マトリクスの列方向に並ぶ計算ノード110に渡す場合にのみ、プロセッサ間通信を行う。このため、分散処理全体における通信に要するコストの割合が一層低減する。

【 0 0 3 4 】

次に、デュアル・コアを用いる場合について説明する。

デュアル・コアは、1つのマルチコア・プロセッサ（以下、デュアル・コアCPU）上に2個のプロセッサコアを備える。そこで、この2個のプロセッサコアを、それぞれ1つの計算ノード110に割り当てる。これにより、1つのクアッド・コアCPUを持つコンピュータ（例えば図2のコンピュータ10）が2つの計算ノード110に対応する。図4に示した力マトリクスでは、16個の計算ノード110が用いられているので、8つのクアッド・コアCPUで16個の計算ノード110全てに対応する。

【 0 0 3 5 】

図7は、デュアル・コアCPUを用いる場合の力マトリクスのレイアウトの例を説明する図である。

図7に示す例では、力マトリクスの行方向に並ぶ2つの計算ノード110に、1つのデュアル・コアCPU上のプロセッサコアが割り当てられている。具体的には、デュアル・コアCPU#0のプロセッサコアがノード#0、#1に割り当てられ、デュアル・コアCPU#1のプロセッサコアがノード#2、#3に割り当てられ、デュアル・コアCPU#2のプロセッサコアがノード#4、#5に割り当てられ、デュアル・コアCPU#3のプロセッサコアがノード#6、#7に割り当てられ、デュアル・コアCPU#4のプロセッサコアがノード#8、#9に割り当てられ、デュアル・コアCPU#5のプロセッサコアがノード#10、#11に割り当てられ、デュアル・コアCPU#6のプロセッサコアが

10

20

30

40

50

ノード# 12、# 13に割り当てられ、デュアル・コアCPU# 7のプロセッサコアがノード# 14、# 15に割り当てられている。

【0036】

このように構成すると、相互作用力の計算結果を各粒子が割り当てられた計算ノード110に集める際のデータのやり取りのうち、図7でペアとなっている2つの計算ノード110の間でのデータのやり取りにおいて、プロセッサ間通信を行う必要がなくなる。また、各計算ノード110による粒子の挙動計算の結果を他の計算ノード110に配信する際のデータのやり取りにおいても、計算結果を力マトリクスの行方向に並ぶ計算ノード110のうち、ペアとなっている2つの計算ノード110の間でのデータのやり取りには、プロセッサ間通信を必要としない。

10

【0037】

したがって、図7のようにプロセッサコアが割り当てられたデュアル・コアCPUは、力マトリクスの行方向に並ぶ計算ノード110のうち、異なるデュアル・コアCPUが割り当てられている計算ノード110の間のデータのやり取りと、計算結果を力マトリクスの列方向に並ぶ計算ノード110の間のデータのやり取りにおいてのみ、プロセッサ間通信を行う。このため、分散処理全体における通信に要するコストの割合が一層低減する。

【0038】

なお、マルチコア・プロセッサの各プロセッサコアに対して、粒子及び粒子の組み合わせを割り当て、並列分散処理を実行させるための(すなわち、個々のプロセッサコアを個別の計算ノード110として動作させるための)具体的な方法としては、例えば、プログラムにおいて、どの計算をどのプロセッサコアで実行するかを指定しておくといった、マルチコア・プロセッサに対するプログラミングの手法として一般的な既存の方法を用いれば良い。

20

【0039】

以上、本実施形態について説明したが、本発明の技術的範囲は上記実施形態に記載の範囲には限定されない。例えば、上記の実施形態では、計算ノード110にクアッド・コアCPUを用いた場合とデュアル・コアCPUを用いた場合について、力マトリクスでのレイアウトを示して説明したが、計算ノード110に6つ以上のプロセッサコアを備えるマルチコア・プロセッサを用いても良い。その他、上記実施形態に、種々の変更または改良を加えたものも、本発明の技術的範囲に含まれることは、特許請求の範囲の記載から明らかである。

30

【図面の簡単な説明】

【0040】

【図1】本実施形態が適用される並列計算システムの全体構成を示す図である。

【図2】本実施形態の計算ノードを実現するコンピュータのハードウェア構成例を示す図である。

【図3】本実施形態の計算ノードの機能構成を示す図である。

【図4】本実施形態で用いられる力マトリクスの例を示す図である。

【図5】図4の力マトリクスにおいて、ノード# 6が相互作用力の計算を担当する粒子の組み合わせを示す図である。

40

【図6】本実施形態において、クアッド・コアCPUを用いる場合の力マトリクスのレイアウトの例を説明する図である。

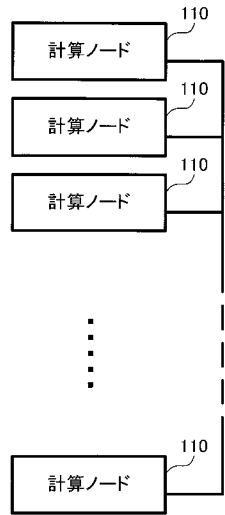
【図7】本実施形態において、デュアル・コアCPUを用いる場合の力マトリクスのレイアウトの例を説明する図である。

【符号の説明】

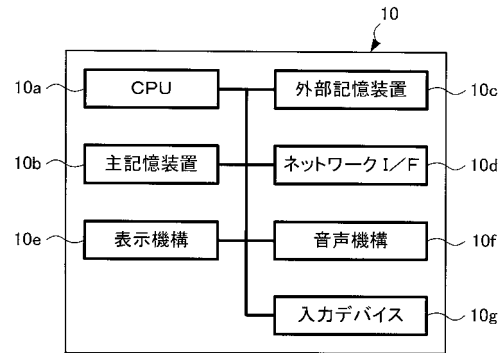
【0041】

110...計算ノード、120...並列分散処理制御部、121...保持部、122...第1計算部、123...第2計算部、124...通信部

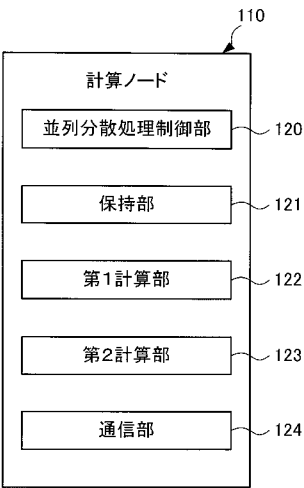
【 図 1 】



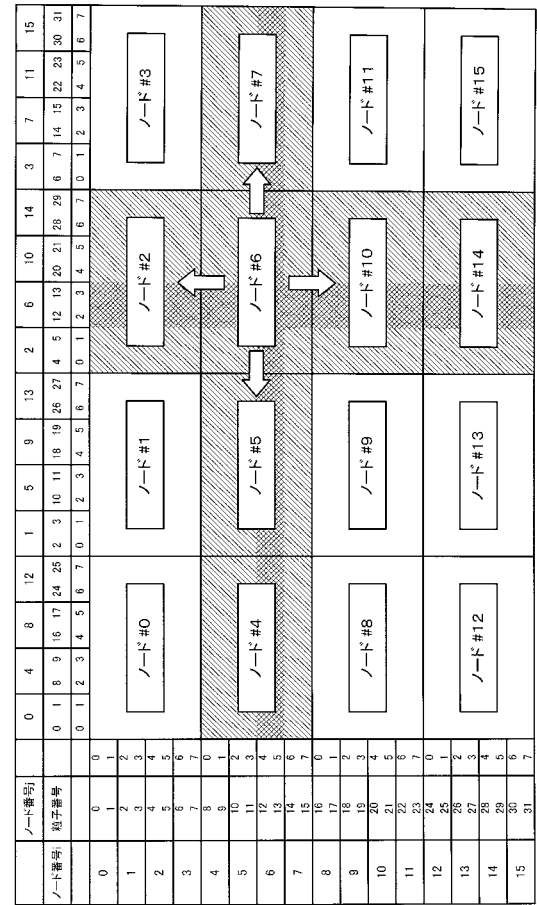
【 図 2 】



【 図 3 】



【 図 4 】



フロントページの続き

- (72)発明者 岡本 英樹
神奈川県足柄上郡中井町境4 3 0 グリーンテクなかい 富士ゼロックス株式会社内
- (72)発明者 廣岡 信行
神奈川県足柄上郡中井町境4 3 0 グリーンテクなかい 富士ゼロックス株式会社内
- (72)発明者 高橋 良輔
神奈川県足柄上郡中井町境4 3 0 グリーンテクなかい 富士ゼロックス株式会社内