

(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2020-201659
(P2020-201659A)

(43) 公開日 令和2年12月17日(2020. 12. 17)

(51) Int. Cl. F 1 テーマコード (参考)
G 0 6 F 1 7 / 1 6 (2 0 0 6 . 0 1) G O 6 F 1 7 / 1 6 F 5 B O 5 6

審査請求 未請求 請求項の数 9 O L (全 19 頁)

(21) 出願番号 特願2019-107283 (P2019-107283)
(22) 出願日 令和1年6月7日 (2019.6.7)

(特許庁注：以下のものは登録商標)

1. SMALL TALK

(71) 出願人 503359821
国立研究開発法人理化学研究所
埼玉県和光市広沢2番1号
(74) 代理人 110000877
龍華国際特許業務法人
(72) 発明者 牧野 淳一郎
埼玉県和光市広沢2番1号 国立研究開発
法人理化学研究所内
(72) 発明者 戎崎 俊一
埼玉県和光市広沢2番1号 国立研究開発
法人理化学研究所内
Fターム(参考) 5B056 AA04 EE07

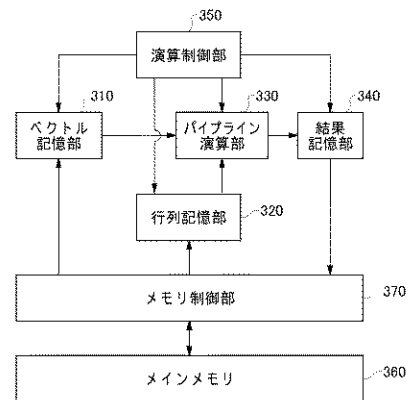
(54) 【発明の名称】 演算装置、演算方法、および演算プログラム

(57) 【要約】 (修正有)

【課題】 行列演算を効率良く行う演算装置を提供する。

【解決手段】 演算装置であって、第1ベクトルを分割した第1の複数の部分ベクトルのうち、第1部分ベクトルを少なくとも記憶するベクトル記憶部と、第1ベクトルに乘じる第1行列を行方向および列方向に分割した第1の複数の部分行列のうち、第1部分ベクトルに乘じるべき第1部分行列を少なくとも記憶する行列記憶部とパイプライン演算により、行列記憶部に記憶された部分行列とベクトル記憶部に記憶された部分ベクトルとの行列ベクトル積に、中間ベクトルを加える演算を実行可能なパイプライン演算部と、パイプライン演算部が、第1部分行列および第1部分ベクトルの行列ベクトル積のパイプライン演算中に、第1部分ベクトルまたは第1部分行列を用いた他の行列ベクトル積の演算の実行をパイプライン演算部に指示する演算制御部とを備える。

【選択図】 図3



300

【特許請求の範囲】**【請求項 1】**

第 1 ベクトルを分割した第 1 の複数の部分ベクトルのうち、第 1 部分ベクトルを少なくとも記憶するベクトル記憶部と、

前記第 1 ベクトルに乗じる第 1 行列を行方向および列方向に分割した第 1 の複数の部分行列のうち、前記第 1 部分ベクトルに乗じるべき第 1 部分行列を少なくとも記憶する行列記憶部と、

パイプライン演算により、前記行列記憶部に記憶された部分行列と前記ベクトル記憶部に記憶された部分ベクトルとの行列ベクトル積に、中間ベクトルを加える演算を実行可能なパイプライン演算部と、

前記パイプライン演算部が、前記第 1 部分行列および前記第 1 部分ベクトルの行列ベクトル積のパイプライン演算中に、前記第 1 部分ベクトルまたは前記第 1 部分行列を用いた他の行列ベクトル積の演算の実行を前記パイプライン演算部に指示する演算制御部とを備える演算装置。

【請求項 2】

前記ベクトル記憶部は、前記第 1 の複数の部分ベクトルのうち、第 2 部分ベクトルを更に記憶し、

前記行列記憶部は、前記第 1 の複数の部分行列のうち、前記第 2 部分ベクトルに乗じるべき第 2 部分行列を更に記憶し、

前記演算制御部は、前記第 1 部分行列および前記第 1 部分ベクトルの行列ベクトル積の演算結果が遅延なく利用可能となるサイクル以降に、前記第 2 部分行列および前記第 2 部分ベクトルの行列ベクトル積を、前記第 1 部分行列および前記第 1 部分ベクトルの行列ベクトル積の演算結果に加える演算の実行を前記パイプライン演算部に指示する

請求項 1 に記載の演算装置。

【請求項 3】

前記ベクトル記憶部は、前記第 1 行列に乗じるべき第 2 ベクトルを分割した第 2 の複数の部分ベクトルのうち、前記第 1 部分行列に乗じるべき第 3 部分ベクトルを更に記憶し、

前記演算制御部は、前記第 1 部分行列および前記第 1 部分ベクトルの行列ベクトル積のパイプライン演算中に、前記他の行列ベクトル積の演算として、前記第 1 部分行列および前記第 3 部分ベクトルの行列ベクトル積の演算の実行を前記パイプライン演算部に指示する

請求項 1 または 2 に記載の演算装置。

【請求項 4】

前記第 1 ベクトルおよび前記第 2 ベクトルは、前記第 1 行列に乗じるべき第 2 行列に含まれる列ベクトルである請求項 3 に記載の演算装置。

【請求項 5】

前記ベクトル記憶部は、前記第 2 行列に含まれる複数の前記第 2 ベクトルを記憶し、

前記演算制御部は、前記第 1 部分行列および前記第 1 部分ベクトルの行列ベクトル積のパイプライン演算の開始後から演算結果が遅滞なく利用可能となる前までの間の各サイクルを、前記第 1 部分行列および前記複数の第 2 ベクトルのそれぞれからの前記第 3 部分ベクトルの行列ベクトル積の演算で充填する

請求項 4 に記載の演算装置。

【請求項 6】

前記行列記憶部は、前記第 1 の複数の部分行列のうち、前記第 1 部分ベクトルに乗じるべき第 3 部分行列を更に記憶し、

前記演算制御部は、前記第 1 部分行列および前記第 1 部分ベクトルの行列ベクトル積のパイプライン演算中に、前記他の行列ベクトル積の演算として、前記第 3 部分行列および前記第 1 部分ベクトルの行列ベクトル積の演算の実行を前記パイプライン演算部に指示する

請求項 1 または 2 に記載の演算装置。

10

20

30

40

50

【請求項 7】

前記行列記憶部は、複数の前記第 3 部分行列を記憶し、

前記演算制御部は、前記第 1 部分行列および前記第 1 部分ベクトルの行列ベクトル積のパイプライン演算の開始後から演算結果が遅滞なく利用可能となる前までの間の各サイクルを、前記複数の第 3 部分行列のそれぞれおよび前記第 1 部分ベクトルの行列ベクトル積の演算で充填する

請求項 6 に記載の演算装置。

【請求項 8】

ベクトル記憶部が、第 1 ベクトルを分割した第 1 の複数の部分ベクトルのうち、第 1 部分ベクトルを少なくとも記憶し、

行列記憶部が、前記第 1 ベクトルに乗じる第 1 行列を行方向および列方向に分割した第 1 の複数の部分行列のうち、前記第 1 部分ベクトルに乗じるべき第 1 部分行列を少なくとも記憶し、

パイプライン演算により、前記行列記憶部に記憶された部分行列と前記ベクトル記憶部に記憶された部分ベクトルとの行列ベクトル積に、中間ベクトルを加える演算を実行可能なパイプライン演算部が、前記第 1 部分行列および前記第 1 部分ベクトルの行列ベクトル積のパイプライン演算中に、前記第 1 部分ベクトルまたは前記第 1 部分行列を用いた他の行列ベクトル積の演算の実行を開始する

演算方法。

【請求項 9】

演算装置によって実行される演算プログラムであって、

前記演算装置は、

第 1 ベクトルを分割した第 1 の複数の部分ベクトルのうち、第 1 部分ベクトルを少なくとも記憶するベクトル記憶部と、

前記第 1 ベクトルに乗じる第 1 行列を行方向および列方向に分割した第 1 の複数の部分行列のうち、前記第 1 部分ベクトルに乗じるべき第 1 部分行列を少なくとも行列記憶部と

パイプライン演算により、前記行列記憶部に記憶された部分行列と前記ベクトル記憶部に記憶された部分ベクトルとの行列ベクトル積に、中間ベクトルを加える演算を実行可能なパイプライン演算部と

を備え、

当該演算プログラムは、前記演算装置に、前記第 1 部分行列および前記第 1 部分ベクトルの行列ベクトル積のパイプライン演算中に、前記第 1 部分ベクトルまたは前記第 1 部分行列を用いた他の行列ベクトル積の演算の実行を開始させるためのものである

演算プログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、演算装置、演算方法、および演算プログラムに関する。

【背景技術】

【0002】

例えば数値計算および深層学習といった種々の応用において、行列行列積（以下、「行列積」と示す。）および行列ベクトル積は、計算量の大部分を占める。このため、このような行列演算を効率良く実行する演算装置および演算方法が開発されている（特許文献 1～3 参照）。また、行列演算を実行可能なプロセッサも開発されている。

[先行技術文献]

[特許文献]

[特許文献 1] 国際公開第 2018/207926 号

[特許文献 2] 特開 2018-139045 号公報

[特許文献 3] 特開 2018-197906 号公報

10

20

30

40

50

【発明の概要】

【発明が解決しようとする課題】

【0003】

n次元正方行列およびn次元ベクトルの行列ベクトル積は、 n^2 の乗算および約 n^2 の加算を含み、約 $2n^2$ の演算量となる。このため、n次元正方行列が固定である場合、行列ベクトル積の演算量は、n次元ベクトルの入力に対して n^2 オーダーとなる。したがって、行列サイズを大きくして行列演算器を大きくすれば、演算量に対するデータのロード量の比率を小さくすることができる。しかし、行列演算器を大きくすると、レジスタファイル等のロード/ストア能力が相対的に低くなり、サイズが小さい行列の演算および行列以外の演算の処理性能が相対的に低くなってしまふ。

10

【課題を解決するための手段】

【0004】

上記課題を解決するために、本発明の第1の態様においては、演算装置を提供する。演算装置は、第1ベクトルを分割した第1の複数の部分ベクトルのうち、第1部分ベクトルを少なくとも記憶するベクトル記憶部を備えてよい。演算装置は、第1ベクトルに乘じる第1行列を行方向および列方向に分割した第1の複数の部分行列のうち、第1部分ベクトルに乘じるべき第1部分行列を少なくとも記憶する行列記憶部を備えてよい。演算装置は、パイプライン演算により、行列記憶部に記憶された部分行列とベクトル記憶部に記憶された部分ベクトルとの行列ベクトル積に、中間ベクトルを加える演算を実行可能なパイプライン演算部を備えてよい。演算装置は、パイプライン演算部が、第1部分行列および第1部分ベクトルの行列ベクトル積のパイプライン演算中に、第1部分ベクトルまたは第1部分行列を用いた他の行列ベクトル積の演算の実行をパイプライン演算部に指示する演算制御部を備えてよい。

20

【0005】

ベクトル記憶部は、第1の複数の部分ベクトルのうち、第2部分ベクトルを更に記憶してよい。行列記憶部は、第1の複数の部分行列のうち、第2部分ベクトルに乘じるべき第2部分行列を更に記憶してよい。演算制御部は、第1部分行列および第1部分ベクトルの行列ベクトル積の演算結果が遅延なく利用可能となるサイクル以降に、第2部分行列および第2部分ベクトルの行列ベクトル積を、第1部分行列および第1部分ベクトルの行列ベクトル積の演算結果に加える演算の実行をパイプライン演算部に指示してよい。

30

【0006】

ベクトル記憶部は、第1行列に乘じるべき第2ベクトルを分割した第2の複数の部分ベクトルのうち、第1部分行列に乘じるべき第3部分ベクトルを更に記憶してよい。演算制御部は、第1部分行列および第1部分ベクトルの行列ベクトル積のパイプライン演算中に、他の行列ベクトル積の演算として、第1部分行列および第3部分ベクトルの行列ベクトル積の演算の実行をパイプライン演算部に指示してよい。

【0007】

第1ベクトルおよび第2ベクトルは、第1行列に乘じるべき第2行列に含まれる列ベクトルであってよい。

【0008】

ベクトル記憶部は、第2行列に含まれる複数の第2ベクトルを記憶してよい。演算制御部は、第1部分行列および第1部分ベクトルの行列ベクトル積のパイプライン演算の開始後から演算結果が遅滞なく利用可能となる前までの間の各サイクルを、第1部分行列および複数の第2ベクトルのそれぞれからの第3部分ベクトルの行列ベクトル積の演算で充填してよい。

40

【0009】

行列記憶部は、第1の複数の部分行列のうち、第1部分ベクトルに乘じるべき第3部分行列を更に記憶してよい。演算制御部は、第1部分行列および第1部分ベクトルの行列ベクトル積のパイプライン演算中に、他の行列ベクトル積の演算として、第3部分行列および第1部分ベクトルの行列ベクトル積の演算の実行をパイプライン演算部に指示してよい

50

。

【 0 0 1 0 】

行列記憶部は、複数の第 3 部分行列を記憶してよい。演算制御部は、第 1 部分行列および第 1 部分ベクトルの行列ベクトル積のパイプライン演算の開始後から演算結果が遅滞なく利用可能となる前までの間の各サイクルを、複数の第 3 部分行列のそれぞれおよび第 1 部分ベクトルの行列ベクトル積の演算で充填してよい。

【 0 0 1 1 】

本発明の第 2 の態様においては、演算方法を提供する。演算方法は、ベクトル記憶部が、第 1 ベクトルを分割した第 1 の複数の部分ベクトルのうち、第 1 部分ベクトルを少なくとも記憶することを備えてよい。演算方法は、行列記憶部が、第 1 ベクトルに乘じる第 1 10
 行列を行方向および列方向に分割した第 1 の複数の部分行列のうち、第 1 部分ベクトルに乘じるべき第 1 部分行列を少なくとも記憶することを備えてよい。演算方法は、パイプライン演算により、行列記憶部に記憶された部分行列とベクトル記憶部に記憶された部分ベクトルとの行列ベクトル積に、中間ベクトルを加える演算を実行可能なパイプライン演算部が、第 1 部分行列および第 1 部分ベクトルの行列ベクトル積のパイプライン演算中に、第 1 部分ベクトルまたは第 1 部分行列を用いた他の行列ベクトル積の演算の実行を開始することを備えてよい。

【 0 0 1 2 】

本発明の第 3 の態様においては、演算装置によって実行される演算プログラムを提供する。演算装置は、第 1 ベクトルを分割した第 1 の複数の部分ベクトルのうち、第 1 部分ベ 20
 クトルを少なくとも記憶するベクトル記憶部を備えてよい。演算装置は、第 1 ベクトルに乘じる第 1 行列を行方向および列方向に分割した第 1 の複数の部分行列のうち、第 1 部分ベクトルに乘じるべき第 1 部分行列を少なくとも行列記憶部を備えてよい。演算装置は、パイプライン演算により、行列記憶部に記憶された部分行列とベクトル記憶部に記憶された部分ベクトルとの行列ベクトル積に、中間ベクトルを加える演算を実行可能なパイプライン演算部を備えてよい。演算プログラムは、演算装置に、第 1 部分行列および第 1 部分ベクトルの行列ベクトル積のパイプライン演算中に、第 1 部分ベクトルまたは第 1 部分行列を用いた他の行列ベクトル積の演算の実行を開始させるためのものであってよい。

【 0 0 1 3 】

なお、上記の発明の概要は、本発明の必要な特徴の全てを列挙したものではない。また、30
 これらの特徴群のサブコンビネーションもまた、発明となりうる。

【 図面の簡単な説明 】

【 0 0 1 4 】

【 図 1 】 本実施形態に係る行列演算の一例を示す。

【 図 2 】 本実施形態に係る行列演算を、部分行列および部分ベクトルの行列ベクトル積に分解した計算式の一例を示す。

【 図 3 】 本実施形態に係る演算装置 3 0 0 の構成を示す。

【 図 4 】 本実施形態に係る演算装置 3 0 0 によるパイプライン処理の第 1 例を示す。

【 図 5 】 本実施形態に係る演算装置 3 0 0 によるパイプライン処理の第 2 例を示す。

【 図 6 】 本実施形態に係る演算装置 3 0 0 によるパイプライン処理の第 3 例を示す。 40

【 図 7 】 本実施形態に係る演算装置 3 0 0 によるパイプライン処理の第 4 例を示す。

【 図 8 】 図 8 は、本発明の複数の態様が全体的または部分的に具現化されてよいコンピュータ 2 2 0 0 の例を示す。

【 発明を実施するための形態 】

【 0 0 1 5 】

以下、発明の実施の形態を通じて本発明を説明するが、以下の実施形態は特許請求の範囲にかかる発明を限定するものではない。また、実施形態の中で説明されている特徴の組み合わせの全てが発明の解決手段に必須であるとは限らない。

【 0 0 1 6 】

図 1 は、本実施形態に係る行列演算の一例を示す。本図は、行列 A および行列 B の行列 50

積を計算し、行列 C に代入する行列演算 $C = A \times B$ を示す。行列 A、B、および C は、8 行 8 列の正方行列である。

【0017】

a_{ij} ($i = 1, 2, \dots, 8, j = 1, 2, \dots, 8$) は、行列 A の成分 (「要素」とも示す。) である。 b_{ij} ($i = 1, 2, \dots, 8, j = 1, 2, \dots, 8$) は、行列 B の成分である。 c_{ij} ($i = 1, 2, \dots, 8, j = 1, 2, \dots, 8$) は、行列 C の成分である。 $j = 3$ の範囲について、 b_{ij} および c_{ij} の各成分の図示は省略している。

【0018】

また、行列 B の j 列の列ベクトル、すなわち行列 B の j 列の各成分 b_{ij} ($i = 1, 2, \dots, 8$) を成分とするベクトルをベクトル v_{bj} 、行列 C の j 列の列ベクトルをベクトル v_{cj} と示す。すなわち、ベクトル $v_{bj} = (b_{1j}, b_{2j}, \dots, b_{8j})^T$ 、ベクトル $v_{cj} = (c_{1j}, c_{2j}, \dots, c_{8j})^T$ と示す。このとき、ベクトル v_{cj} は、行列 A およびベクトル v_{bj} の行列ベクトル積 $v_{cj} = A \times v_{bj}$ によって計算できる。

10

【0019】

ここで、例えば 4 行 4 列の行列と 4 要素のベクトルとの行列ベクトル積を 1 単位の演算として実行可能な演算装置を用いる場合、行列演算 $C = A \times B$ を、演算装置が一度に演算できる単位に分割して行う。本図において、行列 A を、行列 A を行方向および列方向にそれぞれ 2 分割して得られる部分行列を A_{11} 、 A_{21} 、 A_{12} 、および A_{22} と示す。部分行列 A_{mn} ($m = 1, 2, n = 1, 2$) は、行列 A における、行方向に分割した m 番目の行範囲、および列方向に分割した n 番目の列範囲の成分を、その部分行列の成分とする。また、ベクトル v_{bj} を、行方向に 2 分割して得られる部分ベクトルを v_{b1j} および v_{b2j} と示す。 v_{bmj} ($m = 1, 2$) は、ベクトル v_{bj} における、行方向に分割した m 番目の行範囲の成分を、その部分ベクトルの成分とする。また、ベクトル v_{cj} を、行方向に 2 分割して得られる部分ベクトルを v_{c1j} および v_{c2j} と示す。 v_{cmj} ($m = 1, 2$) は、ベクトル v_{cj} における、行方向に分割した m 番目の行範囲の成分を、その部分ベクトルの成分とする。

20

【0020】

なお、本図においては、行列演算の一例として行列積を示し、行列ベクトル積については行列積の一部に含まれるものとして説明した。行列積の演算に含まれない行列ベクトル積については、行列 B および行列 C の第 1 列に関する行列ベクトル積 $v_{c1} = A \times v_{b1}$ 等と同様である。また、本実施形態においては、行列 A、B、および C は、行方向および列方向に 2 のべき乗個の要素を有し、行列 A が行方向および列方向において 2 のべき乗個に分割される場合について例示する。これに代えて、行列 A、B、および C は、行方向または列方向の少なくとも 1 つについて 2 のべき乗個以外の数の要素を有してもよく、行列 A が行方向または列方向の少なくとも 1 つについて 2 のべき乗個以外の数に分割されてもよい (例えば 3×3 、 5×5 、 9×9 、 3×5 、 5×9 等)。

30

【0021】

図 2 は、本実施形態に係る行列演算を、部分行列および部分ベクトルの行列ベクトル積に分解した計算式の一例を示す。行列 A およびベクトル v_{bj} の行列ベクトル積 $v_{cj} = A \times v_{bj}$ は、部分ベクトル v_{c1j} を計算する $v_{c1j} = (A_{11} \ A_{12}) \times v_{bj} = A_{11} \times v_{b1j} + A_{12} \times v_{b2j}$ と、部分ベクトル v_{c2j} を計算する $v_{c2j} = (A_{21} \ A_{22}) \times v_{bj} = A_{21} \times v_{b1j} + A_{22} \times v_{b2j}$ とに分けることができる。すなわち、 $j = 1$ の場合、 $v_{c11} = A_{11} \times v_{b11} + A_{12} \times v_{b21}$ 、 $v_{c21} = A_{21} \times v_{b11} + A_{22} \times v_{b21}$ となる。また、 $j = 2$ の場合、 $v_{c12} = A_{11} \times v_{b12} + A_{12} \times v_{b22}$ 、 $v_{c22} = A_{21} \times v_{b12} + A_{22} \times v_{b22}$ となる。以下、 $j = 3, \dots, 8$ も同様である。

40

【0022】

このように、行列 A を行方向および列方向にそれぞれ d 個に分割し、ベクトル v_{bj} を d 個に分割すると、行列 A およびベクトル v_{bj} の行列ベクトル積は、部分行列および部分ベクトルの行列ベクトル積を $d \times d$ 個含むものとなる。演算装置が単一の部分行列を格納可

50

能なレジスタしか有しない場合、演算装置は、部分行列をメモリからレジスタに順次ロードしながら図2に示した行列演算を行うこととなり、処理性能が低下してしまう。

【0023】

図3は、本実施形態に係る演算装置300の構成を示す。演算装置300は、仕様上定められた行数および列数までの行列と、仕様上定められた行数までのベクトルとの行列ベクトル積を1単位の演算としてパイプライン演算により実行可能である。演算装置300は、1単位の演算で処理可能なサイズよりも大きい行列およびベクトルの行列ベクトル積を、1単位の演算で処理可能な部分行列および部分ベクトルの行列ベクトル積の複数組に分割して計算する。

【0024】

演算装置300は、ベクトル記憶部310と、行列記憶部320と、パイプライン演算部330と、結果記憶部340と、演算制御部350と、メインメモリ360と、メモリ制御部370とを備える。ベクトル記憶部310は、第1ベクトルを分割した第1の複数の部分ベクトルのうち、第1部分ベクトルを少なくとも記憶する。本実施形態において、ベクトル記憶部310は、一例としてレジスタである。これに代えて、ベクトル記憶部310は、キャッシュメモリ等の、パイプライン的に部分ベクトルをパイプライン演算部330に供給できる他の記憶装置であってもよい。

【0025】

ここで、第1ベクトルは、少なくとも1つの部分行列が行列記憶部320に記憶された1行列を乗じる対象となる対象ベクトルである。第1ベクトルは、演算装置300が1単位の演算で処理可能なサイズよりも大きい。第1の複数の部分ベクトルは、第1ベクトルを、1単位の演算で処理可能な大きさに分割したものである。図1の行列演算において、第1ベクトルは、ベクトル v_{bj} のいずれか（例えば v_{b1} ）に相当する。第1の複数の部分ベクトルは、第1ベクトル v_{bj} を分割して得られる部分ベクトル v_{bij} （ $i = 1, 2$ ）に相当する。第1ベクトルがさらに大きい場合、第1ベクトルは、3以上の部分ベクトルに分割されてもよい。

【0026】

また、ベクトル記憶部310は、第1の複数の部分ベクトルのうち、第2部分ベクトル、およびその他の部分ベクトルを更に記憶するべく、十分な記憶領域を有してもよい。例えば、図1の行列演算において、ベクトル記憶部310は、部分ベクトル v_{b1j} および部分ベクトル v_{b2j} を記憶してもよい。

【0027】

行列記憶部320は、第1ベクトルに乘じる第1行列を行方向および列方向に分割した第1の複数の部分行列のうち、第1部分ベクトルに乘じるべき第1部分行列を少なくとも記憶する。本実施形態において、行列記憶部320は、一例としてレジスタである。これに代えて、行列記憶部320は、キャッシュメモリ等の、パイプライン的に部分行列をパイプライン演算部330に供給できる他の記憶装置であってもよい。

【0028】

ここで、第1行列は、少なくとも1つの部分ベクトルがベクトル記憶部310に記憶された第1ベクトルに乘じる対象となる対象行列である。第1行列は、演算装置300が1単位の演算で処理可能なサイズよりも大きい。第1の複数の部分行列は、第1行列を、演算装置300が1単位の演算で処理可能な大きさに分割したものである。図1の行列演算において、第1行列は、行列Aに相当する。第1の複数の部分行列は、第1行列Aを行方向および列方向に分割して得られる部分行列 A_{ij} （ $i = 1, 2, j = 1, 2$ ）に相当する。第1行列がさらに大きい場合、第1行列は、行方向および列方向のそれぞれにおいて3以上に分割されてもよい。

【0029】

また、行列記憶部320は、第1の複数の部分行列のうち、第2部分ベクトルに乘じるべき第2部分行列、およびその他の部分行列を更に記憶するべく、十分な記憶領域を有してもよい。例えば、図1の行列演算において、行列記憶部320は、部分ベクトル v_{b1}

10

20

30

40

50

j に乗じるべき部分行列 A_{11} と、部分ベクトル v_{b2j} に乗じるべき部分行列 A_{12} を記憶してもよい。ここで、第 1 部分ベクトルおよび第 2 部分ベクトルは、第 1 ベクトルにおける異なる行範囲に位置する。このため、第 1 部分行列および第 2 部分行列は、対象行列における異なる列範囲に位置する。なお、第 1 部分行列および第 2 部分行列は、対象行列における同じ行範囲に位置してよい。

【 0 0 3 0 】

パイプライン演算部 330 は、ベクトル記憶部 310 および行列記憶部 320 に接続され、ベクトル記憶部 310 に記憶された演算対象の部分ベクトルをベクトル記憶部 310 から受け取り、行列記憶部 320 に記憶された演算対象の部分行列を行列記憶部 320 から受け取る。パイプライン演算部 330 は、パイプライン演算により、演算対象の部分行列および部分ベクトルの行列ベクトル積に、中間ベクトルを加える演算を実行可能である。本実施形態において、パイプライン演算部 330 は、4 行 4 列の部分行列と 4 行の部分ベクトルとの行列ベクトル積を算出し、この行列ベクトル積に 4 行の中間ベクトルを加えて演算結果となる部分ベクトル（「結果ベクトル」とも示す。）を算出する演算を 1 単位の演算として実行可能である。

10

【 0 0 3 1 】

ここで、1 単位の演算として実行可能とは、パイプライン演算部 330 が、例えば外部からの指示または命令の実行等の要求に応じて、演算対象の部分行列および部分ベクトルの行列ベクトル積に、中間ベクトルを加える演算をまとめて実行し、その結果を出力することを意味する。パイプライン演算部 330 は、この演算に含まれる全ての基本演算（例えば、値同士の乗算、加算）を別個の演算器で行うべく多数の演算器を有してもよく、これに代えて一部の演算を同じ演算器で行ってもよい。

20

【 0 0 3 2 】

また、パイプライン演算部 330 がパイプライン演算を行うとは、パイプライン演算部 330 が演算の開始後複数のステージにおける処理を経て結果を出力するところ、各ステージは並列に動作可能であることを意味する。すなわち、パイプライン演算部 330 は、ある演算の開始後結果を出力するまでの各サイクルにおいて、特に実行上の障害がなければ順次他の演算を開始することができる。

【 0 0 3 3 】

例えば、パイプライン演算部 330 は、1 サイクル目に、部分行列および部分ベクトルを入力し、2 サイクル目に、部分行列および部分ベクトルの対応する要素同士を乗算し、3 サイクル目に、結果ベクトルに含まれるべき各要素について 2 サイクル目に計算した積を合計し、4 サイクル目に、演算結果の部分ベクトルを出力してもよい。パイプライン演算部 330 は、必要に応じて任意の段数のパイプライン構造をとることができる。

30

【 0 0 3 4 】

結果記憶部 340 は、パイプライン演算部 330 に接続される。結果記憶部 340 は、パイプライン演算部 330 が出力する結果ベクトルを受け取って、格納する。結果ベクトルは、例えば図 2 における v_{c11} および v_{c21} 等である。本実施形態において、結果記憶部 340 は、一例としてレジスタである。これに代えて、結果記憶部 340 は、キャッシュメモリ等の、パイプライン的にパイプライン演算部 330 からの部分ベクトルを格納できる他の記憶装置であってもよい。なお、ベクトル記憶部 310、行列記憶部 320、および結果記憶部 340 は、同一の記憶装置として実装されてもよい。

40

【 0 0 3 5 】

演算制御部 350 は、ベクトル記憶部 310、行列記憶部 320、パイプライン演算部 330、および結果記憶部 340 に接続される。演算制御部 350 は、例えば演算装置 300 の外部からの指示を受けたこと、または演算装置 300 におけるプログラム実行中に行列演算命令をデコードしたこと等のような行列演算の実行要求に応じて、要求された行列演算を実行するべくベクトル記憶部 310、行列記憶部 320、パイプライン演算部 330、および結果記憶部 340 を制御する。

【 0 0 3 6 】

50

メインメモリ 360 は、行列演算の対象となる行列および演算結果を格納する。メモリ制御部 370 は、ベクトル記憶部 310、行列記憶部 320、および結果記憶部 340 と、メインメモリ 360 との間に接続される。メモリ制御部 370 は、外部からの指示、または演算装置 300 におけるプログラム実行中のメモリアクセス命令に応じて、ベクトル記憶部 310、行列記憶部 320、および結果記憶部 340 と、メインメモリ 360 との間のデータ転送を行う。

【0037】

例えば、メモリ制御部 370 は、メインメモリ 360 からベクトル記憶部 310 へのベクトルロードが要求されたことに応じて、メインメモリ 360 に記憶された部分ベクトルのうちベクトルロードによって指定された部分ベクトルをメインメモリ 360 から読み出して、ベクトル記憶部 310 へと格納する。また、メモリ制御部 370 は、メインメモリ 360 から行列記憶部 320 への行列ロードが要求されたことに応じて、メインメモリ 360 に記憶された部分行列のうち行列ロードによって指定された部分行列をメインメモリ 360 から読み出して、行列記憶部 320 へと格納する。また、メモリ制御部 370 は、結果記憶部 340 からメインメモリ 360 への行列またはベクトルストアが要求されたことに応じて、結果記憶部 340 に記憶された、演算結果の行列またはベクトルを読み出して、メインメモリ 360 へと格納する。なお、演算装置 300 の設計によっては、ベクトル記憶部 310 および行列記憶部 320 に加えてメインメモリ 360 を設けず、ベクトル記憶部 310 および行列記憶部 320 として機能する比較的大きいメモリを設けて、当該メモリから直接パイプライン的にパイプライン演算部 330 に部分ベクトルおよび部分行列を供給できるようにしてもよい。

【0038】

以上に示した構成において、パイプライン演算部 330 は、パイプライン処理により、行列ベクトル積の演算を実行する。例えば図 2 に示した $v_{c11} = A_{11} \times v_{b11} + A_{12} \times v_{b21}$ の演算を行う場合には、パイプライン演算部 330 は、第 1 部分行列 A_{11} および第 1 部分ベクトル v_{b11} の演算を開始した後に演算結果を得るまでに、複数サイクルを要する。このため、パイプライン演算部 330 は、第 1 部分行列 A_{11} および第 1 部分ベクトル v_{b11} の行列ベクトル積を計算する第 1 演算を開始したサイクルの次のサイクルに、第 2 部分行列 A_{12} および第 2 部分ベクトル v_{b21} の行列ベクトル積を第 1 演算の結果に加える第 2 演算が投入されたとしても、第 2 演算の実行に障害が生じ（パイプラインハザード）、第 1 演算の演算結果が利用可能となるまで第 2 演算の処理を待たせる必要が生じてしまう。

【0039】

なお、パイプラインの設計によっては、第 1 演算の結果をレジスタに書き込むのを待たずに第 2 演算へと供給する（バイパスまたはフォワーディング）等により、第 2 演算の処理待ちをある程度は削減することができる。しかし、第 1 演算および第 2 演算の間に依存関係がある以上、パイプラインハザードによってパイプライン演算部 330 のパイプラインに生じる空きを完全になくすことは難しい。

【0040】

そこで、演算制御部 350 は、パイプライン演算部 330 が、第 1 部分行列および第 1 部分ベクトルの行列ベクトル積（例えば $A_{11} \times v_{b11}$ ）のパイプライン演算中に、第 1 部分ベクトルまたは第 1 部分行列を用いた他の行列ベクトル積の演算の実行をパイプライン演算部 330 に指示する。ここで「他の行列ベクトル積」は、第 1 部分行列および第 1 部分ベクトルの行列ベクトル積の演算結果を使用しない演算であり、行列ベクトル積を含む演算、すなわち例えば行列ベクトル積に第 1 部分行列および第 1 部分ベクトルの行列ベクトル積以外の演算結果を加えるような演算であってもよい。これにより、演算制御部 350 は、パイプライン演算部 330 が第 1 演算の演算結果を待ってから第 2 演算を実行開始するまでの間に、第 1 演算の演算結果に依存しない 1 または複数の他の行列ベクトル積をパイプライン演算部 330 へと投入し、これによってパイプライン演算部 330 の利用効率を高めることができる。

【 0 0 4 1 】

さらに、演算制御部 3 5 0 は、第 1 部分行列および第 1 部分ベクトルの行列ベクトル積の演算結果が遅延なく利用可能となるサイクル以降に、第 2 部分行列および第 2 部分ベクトルの行列ベクトル積を、第 1 部分行列および第 1 部分ベクトルの行列ベクトル積の演算結果に加える演算の実行をパイプライン演算部 3 3 0 に指示してもよい。これにより、演算制御部 3 5 0 は、第 2 演算にパイプラインハザードが生じるのを防ぐことができ、第 1 演算および第 2 演算の間に他の行列ベクトル積の演算を投入可能とすることができる。

【 0 0 4 2 】

図 4 は、本実施形態に係る演算装置 3 0 0 によるパイプライン処理の第 1 例を示す。サイクル 0 と示した演算において、演算制御部 3 5 0 は、第 1 部分ベクトルの一例である v_{b11} の読み出しをベクトル記憶部 3 1 0 に指示し、第 1 部分行列の一例である A_{11} の読み出しを行列記憶部 3 2 0 に指示するとともに、第 1 部分行列 A_{11} および第 1 部分ベクトル v_{b11} の行列ベクトル積を計算し、計算途中の中間ベクトルとしてパイプライン演算部 3 3 0 が有する中間レジスタ（テンポラリレジスタ） v_{ctmp1} に格納する演算の実行をパイプライン演算部 3 3 0 に指示する。

【 0 0 4 3 】

サイクル 1 の実行開始までに、ベクトル記憶部 3 1 0 は、第 1 部分ベクトルの一例である v_{b11} および第 2 部分ベクトルの一例である v_{b21} に加えて、第 1 行列 A を乗じるべき第 2 ベクトル（一例として v_{b2} ）を分割した第 2 の複数の部分ベクトル v_{bi2} のうち、第 1 部分行列 A_{11} を乗じるべき第 3 部分ベクトル（一例として v_{b12} ）を更に記憶する。本例において、第 1 ベクトルおよび第 2 ベクトルは、第 1 行列 A に乗じるべき第 2 行列 B に含まれる列ベクトルであり、例えば第 1 ベクトルは v_{b1} 、第 2 ベクトルは v_{b2} である。第 3 部分ベクトルは、第 2 ベクトル v_{b2} を分割した第 2 の複数の部分ベクトル v_{bi2} のうち第 1 部分行列 A_{11} を乗じるべき v_{b12} である。これに代えて、第 1 ベクトルおよび第 2 ベクトルは、それぞれ行列 A を乗じるべき別個のベクトルであってもよい。

【 0 0 4 4 】

サイクル 1 と示した演算において、第 1 部分行列および第 1 部分ベクトルの行列ベクトル積のパイプライン演算中に、演算制御部 3 5 0 は、第 3 部分ベクトル v_{b12} の読み出しをベクトル記憶部 3 1 0 に指示し、第 1 部分行列 A_{11} の読み出しを行列記憶部 3 2 0 に指示するとともに、パイプラインハザードを生じさせない他の行列ベクトル積の演算として、第 1 部分行列および第 3 部分ベクトルの行列ベクトル積の演算の実行をパイプライン演算部 3 3 0 に指示する。これを受けて、パイプライン演算部 3 3 0 は、第 1 部分行列および第 3 部分ベクトルの行列ベクトル積を、計算途中の中間ベクトルとしてパイプライン演算部 3 3 0 が有する中間レジスタ v_{ctmp2} に格納する演算を実行する。この演算は、図 2 の第 3 行目における 1 つ目の行列ベクトル積の演算に相当し、サイクル 0 および 1 の行列ベクトル積は、互いに異なる結果ベクトル v_{c11} および v_{c12} に反映されるものである。したがって、これらの演算の間に依存関係はないから、パイプライン演算部 3 3 0 は、これらの演算を、パイプラインハザードを発生させることなく実行することができる。

【 0 0 4 5 】

サイクル 2 と示した演算において、演算制御部 3 5 0 は、第 2 部分ベクトルの一例である v_{b21} の読み出しをベクトル記憶部 3 1 0 に指示し、第 2 部分行列の一例である A_{12} の読み出しを行列記憶部 3 2 0 に指示するとともに、第 2 部分行列 A_{12} および第 2 部分ベクトル v_{b21} の行列ベクトル積を計算し、サイクル 0 の演算の演算結果 v_{ctmp1} を加える演算の実行をパイプライン演算部 3 3 0 に指示し、演算の結果得られる部分ベクトル v_{c11} を格納することを結果記憶部 3 5 0 に指示する。ここで、サイクル 2 の演算はサイクル 0 の演算に依存するところ、演算制御部 3 5 0 は、サイクル 0 の演算に依存しないサイクル 1 の演算をサイクル 0 およびサイクル 2 の演算の間に挿入することで、パイプライン演算部 3 3 0 のパイプラインの利用効率を上げることができる。

10

20

30

40

50

【 0 0 4 6 】

サイクル3の実行開始までに、ベクトル記憶部310は、第2の複数の部分ベクトルのうち、第2部分行列A12を乗じるべき第4部分ベクトルを更に記憶してよい。サイクル3と示した演算において、演算制御部350は、第4部分ベクトルの一例であるvb22の読み出しをベクトル記憶部310に指示し、第2部分行列の一例であるA12の読み出しを行列記憶部320に指示するとともに、第2部分行列A12および第4部分ベクトルvb22の行列ベクトル積を計算し、サイクル1の演算の演算結果vc t m p 2を加える演算の実行をパイプライン演算部330に指示し、演算の結果得られる部分ベクトルvc12を格納することをメインメモリ360に指示する。ここで、サイクル3の演算はサイクル1の演算に依存するところ、演算制御部350は、サイクル1の演算に依存しないサイ

10

【 0 0 4 7 】

本図の例では、サイクル0~3において行列Cの複数の列ベクトル(v c 1、v c 2)における第1行範囲(第1~4行)の部分ベクトル(v c 1 1、v c 1 2)を計算し、サイクル4~7において行列Cの複数の列ベクトル(v c 1、v c 2)における第2行範囲(第5~8行)の部分ベクトル(v c 2 1、v c 2 2)を計算する。サイクル4~7の演算は、部分行列A11、A12に代えて部分行列A21、A22を用い、部分ベクトルv c 1 1、v c 1 2に代えて部分ベクトルv c 2 1、v c 2 2を用いる他は同様であるので説明を省略する。

20

【 0 0 4 8 】

本例において、演算制御部350は、第1部分行列および第1部分ベクトルの行列ベクトル積の第1演算と、その演算結果を利用する第2演算との間に、第1部分行列を用いた他の行列ベクトル積の演算、すなわち本例においては第1部分行列および第3部分ベクトルの行列ベクトル積の演算、を挿入する。これによって、演算制御部350は、第1演算および第2演算の間に必要となる空きサイクルを1つ利用することができる。

【 0 0 4 9 】

第1演算および第2演算の間に複数の空きサイクルが生じる場合、演算制御部350は、第1部分行列および複数の第3部分ベクトルのそれぞれの行列ベクトル積を第1演算および第2演算の間に挿入してよい。例えば、ベクトル記憶部310は、第2行列Bに含まれる複数の第2ベクトルvb2、vb3、を更に記憶しておく。演算制御部350は、第1部分行列A11および第1部分ベクトルvb11の行列ベクトル積のパイプライン演算の開始後から演算結果が遅滞なく利用可能となる前までの間の各サイクルを、第1部分行列A11および複数の第2ベクトルvb2、vb3、のそれぞれからの第3部分ベクトルvb12、vb13、の行列ベクトル積A11×vb12、A11×vb13、の演算で充填する。なお、第1ベクトルおよび複数の第2ベクトルは、第2行列の列順または列順の逆順に並んでいてもよく、また第2行列の列順に並んでおらず、それぞれ任意の列の列ベクトルであってよい。

30

【 0 0 5 0 】

図5は、本実施形態に係る演算装置300によるパイプライン処理の第2例を示す。パイプライン演算部330がより多くの中間レジスタを有する場合、または演算結果を一旦メインメモリ360に格納した後に利用可能となる場合等においては、演算制御部350は、図4におけるサイクル4~5の演算を、サイクル2~3の演算の前に行うように制御してもよい。この場合、演算制御部350は、第1部分行列A11および第1部分ベクトルvb11の行列ベクトル積を演算する第1演算のパイプライン演算の実行中に、第1部分行列を用いた他の行列ベクトル積の演算である図5中のサイクル1の演算と、第1部分ベクトルを用いた他の行列ベクトル積の演算である図5中のサイクル2の演算とをパイプライン演算部330に実行させる。また、演算制御部350は、サイクル2の演算に用いた第2部分行列A21と、サイクル1の演算に用いた第2部分ベクトルvb12との行列ベクトル積の演算であるサイクル3の演算を、第1演算および第2演算の間に実行させて

40

50

よい。これにより、演算制御部 350 は、第 1 演算および第 2 演算の間の空きサイクルを更に充填することが可能となる。なお、サイクル 0 ~ 3 の演算同士の実行順序は任意であってよく、サイクル 4 ~ 7 の演算同士の実行順序はサイクル 0 ~ 3 における対応する演算の実行順序に応じて決定されてよい。

【0051】

図 6 は、本実施形態に係る演算装置 300 によるパイプライン処理の第 3 例を示す。サイクル 0 と示した演算において、演算制御部 350 は、図 4 のサイクル 0 と同様の制御を行う。

【0052】

サイクル 1 の実行開始までに、行列記憶部 320 は、第 1 部分行列の一例である A 1 1 および第 2 部分行列の一例である A 1 2 に加えて、第 1 行列 A を行方向および列方向に分割した第 1 の複数の部分行列 A $i j$ のうち、第 1 部分ベクトル $v b 1 1$ に乗じるべき第 3 部分行列（一例として A 2 1）を更に記憶する。これに代えて、第 3 部分行列は、第 1 行列 A 以外の行列に含まれる部分行列であってもよい。

【0053】

サイクル 1 と示した演算において、第 1 部分行列および第 1 部分ベクトルの行列ベクトル積のパイプライン演算中に、演算制御部 350 は、第 1 部分ベクトル $v b 1 1$ の読み出しをベクトル記憶部 310 に指示し、第 3 部分行列 A 2 1 の読み出しを行列記憶部 320 に指示するとともに、パイプラインハザードを生じさせない他の行列ベクトル積の演算として、第 3 部分行列 A 2 1 および第 1 部分ベクトル $v b 1 1$ の行列ベクトル積の演算の実行をパイプライン演算部 330 に指示する。これを受けて、パイプライン演算部 330 は、第 3 部分行列 A 2 1 および第 1 部分ベクトル $v b 1 1$ の行列ベクトル積を、計算途中の中間ベクトルとしてパイプライン演算部 330 が有する中間レジスタ $v c t m p 2$ に格納する演算を実行する。この演算は、図 2 の第 2 行目における 1 つ目の行列ベクトル積の演算に相当し、サイクル 0 および 1 の行列ベクトル積は、互いに異なる結果ベクトル $v c 1 1$ および $v c 2 1$ に反映されるものである。したがって、これらの演算の間に依存関係はないから、パイプライン演算部 330 は、これらの演算を、パイプラインハザードを発生させることなく実行することができる。

【0054】

サイクル 2 と示した演算において、演算制御部 350 は、図 4 のサイクル 2 と同様の制御を行う。ここで、サイクル 2 の演算はサイクル 0 の演算に依存するところ、演算制御部 350 は、サイクル 0 の演算に依存しないサイクル 1 の演算をサイクル 0 およびサイクル 2 の演算の間に挿入することで、パイプライン演算部 330 のパイプラインの利用効率を上げることができる。

【0055】

サイクル 3 の実行開始までに、ベクトル記憶部 310 は、第 1 の複数の部分行列 A $i j$ のうち、第 2 部分ベクトル $v b 2 1$ に乗じるべき第 4 部分行列を更に記憶してよい。サイクル 3 と示した演算において、演算制御部 350 は、第 2 部分ベクトルの一例である $v b 2 1$ の読み出しをベクトル記憶部 310 に指示し、第 4 部分行列の一例である A 2 2 の読み出しを行列記憶部 320 に指示するとともに、第 4 部分行列 A 2 2 および第 2 部分ベクトル $v b 2 1$ の行列ベクトル積を計算し、サイクル 1 の演算の演算結果 $v c t m p 2$ を加える演算の実行をパイプライン演算部 330 に指示し、演算の結果得られる部分ベクトル $v c 2 1$ を格納することをメインメモリ 360 に指示する。ここで、サイクル 3 の演算はサイクル 1 の演算に依存するところ、演算制御部 350 は、サイクル 1 の演算に依存しないサイクル 2 の演算をサイクル 1 およびサイクル 3 の演算の間に挿入することで、パイプライン演算部 330 のパイプラインの利用効率を上げることができる。

【0056】

本図の例では、サイクル 0 ~ 3 において行列 C の 1 つの列ベクトル $v c 1$ に含まれる 2 つの部分ベクトル $v c 1 1$ 、 $v c 2 1$ を計算し、サイクル 4 ~ 7 において行列 C の別の列ベクトル $v c 2$ に含まれる 2 つの部分ベクトル $v c 1 2$ 、 $v c 2 2$ を計算する。サイクル

10

20

30

40

50

4～7の演算は、部分ベクトル v_{b11} 、 v_{b21} に代えて部分ベクトル v_{b12} 、 v_{b22} を用い、部分ベクトル v_{c11} 、 v_{c21} に代えて部分ベクトル v_{c12} 、 v_{c22} を用いる他は同様であるので説明を省略する。

【0057】

本例において、演算制御部350は、第1部分行列および第1部分ベクトルの行列ベクトル積の第1演算と、その演算結果を利用する第2演算との間に、第1部分ベクトルを用いた他の行列ベクトル積の演算、すなわち本例においては第3部分行列および第1部分ベクトルの行列ベクトル積の演算、を挿入する。これによって、演算制御部350は、第1演算および第2演算の間に必要となる空きサイクルを1つ利用することができる。

【0058】

第1演算および第2演算の間に複数の空きサイクルが生じる場合、演算制御部350は、複数の第3部分行列のそれぞれおよび第1部分ベクトルの行列ベクトル積を第1演算および第2演算の間に挿入してよい。例えば、行列記憶部320は、第1行列Aに含まれる、第1部分ベクトルに乘じるべき複数の第3部分行列 A_{21} 、 A_{31} 、を記憶しておく。演算制御部350は、第1部分行列 A_{11} および第1部分ベクトル v_{b11} の行列ベクトル積のパイプライン演算の開始後から演算結果が遅滞なく利用可能となる前までの間の各サイクルを、複数の第3部分行列 A_{21} 、 A_{31} 、のそれぞれおよび第1部分ベクトル v_{b11} の行列ベクトル積の演算で充填する。なお、第1部分行列および複数の第3部分行列は、第1行列の同一の行範囲において列順または列順の逆順に並んでいてもよく、また第2行列の列順に並んでおらず、それぞれ任意の列範囲の部分行列であってよい。

【0059】

図7は、本実施形態に係る演算装置300によるパイプライン処理の第4例を示す。パイプライン演算部330がより多くの中間レジスタを有する場合、または演算結果を一旦メインメモリ360に格納した後に利用可能となる場合等においては、演算制御部350は、図6におけるサイクル4～5の演算を、サイクル2～3の演算の前に行うように制御してもよい。この場合、演算制御部350は、第1部分行列 A_{11} および第1部分ベクトル v_{b11} の行列ベクトル積を演算する第1演算のパイプライン演算の実行中に、第1部分ベクトルを用いた他の行列ベクトル積の演算である図7中のサイクル1の演算と、第1部分行列を用いた他の行列ベクトル積の演算である図7中のサイクル2の演算とをパイプライン演算部330に実行させる。また、演算制御部350は、サイクル1の演算に用いた第3部分行列 A_{21} と、サイクル2の演算に用いた第2部分ベクトル v_{b12} との行列ベクトル積の演算であるサイクル3の演算を、第1演算および第2演算の間に実行させてよい。これにより、演算制御部350は、第1演算および第2演算の間の空きサイクルを更に充填することが可能となる。なお、サイクル0～3の演算同士の実行順序は任意であってよく、サイクル4～7の演算同士の実行順序はサイクル0～3における対応する演算の実行順序に応じて決定されてよい。ここで、図7のパイプライン処理は、図5のパイプライン処理におけるサイクル1および2の演算を入れ換え、サイクル5および6の演算を入れ換えたものと実質的に同一である。

【0060】

以上に示した第1例から第4例を含む任意のパイプライン処理において、演算制御部350は、パイプライン演算部330が使用する部分ベクトルおよび部分行列を、パイプライン演算部330が必要とするよりも前にメインメモリ360からベクトル記憶部310および行列記憶部320へと転送するようにメモリ制御部370に指示してよい。例えば、図4の例において、メインメモリ360は、サイクル0の前に、部分ベクトル v_{b11} 、 v_{b12} 、 v_{b21} 、 v_{b22} をベクトル記憶部310へと転送し、部分行列 A_{11} および A_{12} を行列記憶部320へと転送してもよい。これに代えて、メインメモリ360は、サイクル0の前に、部分ベクトル v_{b11} 、 v_{b12} をベクトル記憶部310へと転送し、部分行列 A_{11} を行列記憶部320へと転送し、サイクル2の前に、部分ベクトル v_{b21} 、 v_{b22} をベクトル記憶部310へと転送し、部分行列 A_{12} を行列記憶部320へと転送してもよい。

10

20

30

40

50

【 0 0 6 1 】

第 1 例および第 2 例に示したパイプライン処理の場合、パイプライン演算部 3 3 0 は、サイクル毎に異なる部分ベクトル $v b 1 1$ 、 $v b 1 2$ 、 $v b 2 1$ 、 $v b 2 2$ を使用するが、部分行列 $A 1 1$ 、 $A 1 2$ 、 $A 2 1$ 、 $A 2 2$ は 2 サイクルに 1 つずつ使用する。このため、行列記憶部 3 2 0 は、2 サイクルに 1 つずつ部分行列を出力できるスループットを有すればよく、行列記憶部 3 2 0 の消費電力および回路規模を低減することができる。

【 0 0 6 2 】

第 3 例および第 4 例に示したパイプライン処理の場合、パイプライン演算部 3 3 0 は、サイクル毎に異なる部分行列 $A 1 1$ 、 $A 1 2$ 、 $A 2 1$ 、 $A 2 2$ を使用するが、部分ベクトル $v b 1 1$ 、 $v b 1 2$ 、 $v b 2 1$ 、 $v b 2 2$ は 2 サイクルに 1 つずつ使用する。このため、行列記憶部 3 2 0 は、2 サイクルに 1 つずつ部分ベクトルを出力できるスループットを有すればよく、ベクトル記憶部 3 1 0 の消費電力および回路規模を低減することができる。

【 0 0 6 3 】

演算装置 3 0 0 の設計者または演算装置 3 0 0 を使用するユーザは、演算装置 3 0 0 の回路規模をより小さくできるように、または、演算装置 3 0 0 の消費電力をより小さくできるように、パイプライン処理の実行順序を選択してよい。

【 0 0 6 4 】

本発明の様々な実施形態は、フローチャートおよびブロック図を参照して記載されてよく、ここにおいてブロックは、(1) 操作が実行されるプロセスの段階または (2) 操作を実行する役割を持つ装置のセクションを表わしてよい。特定の段階およびセクションが、専用回路、コンピュータ可読媒体上に格納されるコンピュータ可読命令と共に供給されるプログラマブル回路、およびコンピュータ可読媒体上に格納されるコンピュータ可読命令と共に供給されるプロセッサのいずれかによって実装されてよい。専用回路は、デジタルおよびアナログのいずれかのハードウェア回路を含んでよく、集積回路 (I C) およびディスクリート回路の何れかを含んでよい。プログラマブル回路は、論理 A N D 、論理 O R 、論理 X O R 、論理 N A N D 、論理 N O R 、および他の論理操作、フリップフロップ、レジスタ、フィールドプログラマブルゲートアレイ (F P G A) 、プログラマブルロジックアレイ (P L A) 等のようなメモリ要素等を含む、再構成可能なハードウェア回路を含んでよい。

【 0 0 6 5 】

コンピュータ可読媒体は、適切なデバイスによって実行される命令を格納可能な任意の有形なデバイスを含んでよく、その結果、そこに格納される命令を有するコンピュータ可読媒体は、フローチャートまたはブロック図で指定された操作を実行するための手段を作成すべく実行され得る命令を含む、製品を備えることになる。コンピュータ可読媒体の例としては、電子記憶媒体、磁気記憶媒体、光記憶媒体、電磁記憶媒体、半導体記憶媒体等が含まれてよい。コンピュータ可読媒体のより具体的な例としては、フロッピー (登録商標) ディスク、ディスクット、ハードディスク、ランダムアクセスメモリ (R A M) 、リードオンリメモリ (R O M) 、消去可能プログラマブルリードオンリメモリ (E P R O M またはフラッシュメモリ) 、電氣的消去可能プログラマブルリードオンリメモリ (E E P R O M) 、静的ランダムアクセスメモリ (S R A M) 、コンパクトディスクリートオンリメモリ (C D R O M) 、デジタル多用途ディスク (D V D) 、ブルーレイ (R T M) ディスク、メモリスティック、集積回路カード等が含まれてよい。

【 0 0 6 6 】

コンピュータ可読命令は、アセンブラ命令、命令セットアーキテクチャ (I S A) 命令、マシン命令、マシン依存命令、マイクロコード、ファームウェア命令、状態設定データ、または S m a l l t a l k 、 J A V A (登録商標) 、 C + + 等のようなオブジェクト指向プログラミング言語、および「 C 」プログラミング言語または同様のプログラミング言語のような従来の手続型プログラミング言語を含む、1 または複数のプログラミング言語の任意の組み合わせで記述されたソースコードまたはオブジェクトコードのいずれかを含

んでよい。

【0067】

コンピュータ可読命令は、汎用コンピュータ、特殊目的のコンピュータ、若しくは他のプログラム可能なデータ処理装置のプロセッサまたはプログラマブル回路に対し、ローカルにまたはローカルエリアネットワーク（LAN）、インターネット等のようなワイドエリアネットワーク（WAN）を介して提供され、フローチャートまたはブロック図で指定された操作を実行するための手段を作成すべく、コンピュータ可読命令を実行してよい。プロセッサの例としては、コンピュータプロセッサ、処理ユニット、マイクロプロセッサ、デジタル信号プロセッサ、コントローラ、マイクロコントローラ等を含む。

【0068】

図8は、本発明の複数の態様が全体的または部分的に具現化されてよいコンピュータ2200の例を示す。コンピュータ2200にインストールされたプログラムは、コンピュータ2200に、本発明の実施形態に係る装置に関連付けられる操作または当該装置の1または複数のセクションとして機能させることができてもよいし、または当該操作または当該1または複数のセクションを実行させることができてもよいし、コンピュータ2200に、本発明の実施形態に係るプロセスまたは当該プロセスの段階を実行させることができてもよい。そのようなプログラムは、コンピュータ2200に、本明細書に記載のフローチャートおよびブロック図のブロックのうちのいくつかまたはすべてに関連付けられた特定の操作を実行させるべく、CPU2212によって実行されてよい。

【0069】

本実施形態によるコンピュータ2200は、CPU2212、RAM2214、グラフィックコントローラ2216、およびディスプレイデバイス2218を含み、それらはホストコントローラ2210によって相互に接続されている。コンピュータ2200はまた、通信インターフェイス2222、ハードディスクドライブ2224、DVD-ROMドライブ2226、およびICカードドライブのような入出力ユニットを含み、それらは入出力コントローラ2220を介してホストコントローラ2210に接続されている。コンピュータはまた、ROM2230およびキーボード2242のようなレガシの入出力ユニットを含み、それらは入出力チップ2240を介して入出力コントローラ2220に接続されている。

【0070】

CPU2212は、ROM2230およびRAM2214内に格納されたプログラムに従い動作し、それにより各ユニットを制御する。グラフィックコントローラ2216は、RAM2214内に提供されるフレームバッファ等またはそれ自体の中にCPU2212によって生成されたイメージデータを取得し、イメージデータがディスプレイデバイス2218上に表示されるようにする。

【0071】

通信インターフェイス2222は、ネットワークを介して他の電子デバイスと通信する。ハードディスクドライブ2224は、コンピュータ2200内のCPU2212によって使用されるプログラムおよびデータを格納する。DVD-ROMドライブ2226は、プログラムまたはデータをDVD-ROM2201から読み取り、ハードディスクドライブ2224にRAM2214を介してプログラムまたはデータを提供する。ICカードドライブは、プログラムおよびデータをICカードから読み取り、プログラムおよびデータをICカードに書き込む。

【0072】

ROM2230はその中に、アクティブ化時にコンピュータ2200によって実行されるブートプログラム等、およびコンピュータ2200のハードウェアに依存するプログラムのいずれかを格納する。入出力チップ2240はまた、様々な入出力ユニットをパラレルポート、シリアルポート、キーボードポート、マウスポート等を介して、入出力コントローラ2220に接続してよい。

【0073】

10

20

30

40

50

プログラムが、DVD-ROM 2201またはICカードのようなコンピュータ可読媒体によって提供される。プログラムは、コンピュータ可読媒体から読み取られ、コンピュータ可読媒体の例でもあるハードディスクドライブ 2224、RAM 2214、またはROM 2230にインストールされ、CPU 2212によって実行される。これらのプログラム内に記述される情報処理は、コンピュータ 2200に読み取られ、プログラムと、上記様々なタイプのハードウェアリソースとの間の連携をもたらす。装置または方法が、コンピュータ 2200の使用に従い情報の操作または処理を実現することによって構成されてよい。

【0074】

例えば、通信がコンピュータ 2200および外部デバイス間で実行される場合、CPU 2212は、RAM 2214にロードされた通信プログラムを実行し、通信プログラムに記述された処理に基づいて、通信インターフェイス 2222に対し、通信処理を命令してよい。通信インターフェイス 2222は、CPU 2212の制御下、RAM 2214、ハードディスクドライブ 2224、DVD-ROM 2201、またはICカードのような記録媒体内に提供される送信バッファ処理領域に格納された送信データを読み取り、読み取られた送信データをネットワークに送信し、またはネットワークから受信された受信データを記録媒体上に提供される受信バッファ処理領域等へ書き込む。

10

【0075】

また、CPU 2212は、ハードディスクドライブ 2224、DVD-ROMドライブ 2226 (DVD-ROM 2201)、ICカード等のような外部記録媒体に格納されたファイルまたはデータベースの全部または必要な部分がRAM 2214に読み取られるようにし、RAM 2214上のデータに対し様々なタイプの処理を実行してよい。CPU 2212は次に、処理されたデータを外部記録媒体にライトバックする。

20

【0076】

様々なタイプのプログラム、データ、テーブル、およびデータベースのような様々なタイプの情報が記録媒体に格納され、情報処理を受けてよい。CPU 2212は、RAM 2214から読み取られたデータに対し、本開示の随所に記載され、プログラムの命令シーケンスによって指定される様々なタイプの操作、情報処理、条件判断、条件分岐、無条件分岐、情報の検索および置換等のいずれかを含む、様々なタイプの処理を実行してよく、結果をRAM 2214に対しライトバックする。また、CPU 2212は、記録媒体内のファイル、データベース等における情報を検索してよい。例えば、各々が第2の属性の属性値に関連付けられた第1の属性の属性値を有する複数のエントリが記録媒体内に格納される場合、CPU 2212は、第1の属性の属性値が指定される、条件に一致するエントリを当該複数のエントリの中から検索し、当該エントリ内に格納された第2の属性の属性値を読み取り、それにより予め定められた条件を満たす第1の属性に関連付けられた第2の属性の属性値を取得してよい。

30

【0077】

上で説明したプログラムまたはソフトウェアモジュールは、コンピュータ 2200上またはコンピュータ 2200近傍のコンピュータ可読媒体に格納されてよい。また、専用通信ネットワークまたはインターネットに接続されたサーバーシステム内に提供されるハードディスクまたはRAMのような記録媒体が、コンピュータ可読媒体として使用可能であり、それによりプログラムを、ネットワークを介してコンピュータ 2200に提供する。

40

【0078】

以上、本発明を実施の形態を用いて説明したが、本発明の技術的範囲は上記実施の形態に記載の範囲には限定されない。上記実施の形態に、多様な変更または改良を加えることが可能であることが当業者に明らかである。その様な変更または改良を加えた形態も本発明の技術的範囲に含まれ得ることが、特許請求の範囲の記載から明らかである。

【0079】

特許請求の範囲、明細書、および図面中において示した装置、システム、プログラム、および方法における動作、手順、ステップ、および段階等の各処理の実行順序は、特段「

50

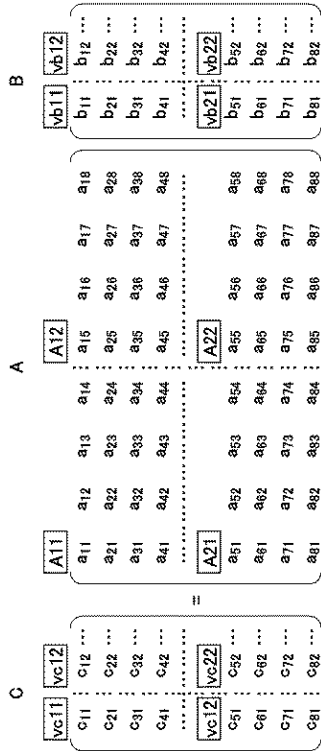
より前に」、「先立って」等と明示しておらず、また、前の処理の出力を後の処理で用いるのでない限り、任意の順序で実現しうることに留意すべきである。特許請求の範囲、明細書、および図面中の動作フローに関して、便宜上「まず」、「次に、」等を用いて説明したとしても、この順で実施することが必須であることを意味するものではない。

【符号の説明】

【 0 0 8 0 】

3 0 0	演算装置	
3 1 0	ベクトル記憶部	
3 2 0	行列記憶部	
3 3 0	パイプライン演算部	10
3 4 0	結果記憶部	
3 5 0	演算制御部	
3 6 0	メインメモリ	
3 7 0	メモリ制御部	
2 2 0 0	コンピュータ	
2 2 0 1	D V D - R O M	
2 2 1 0	ホストコントローラ	
2 2 1 2	C P U	
2 2 1 4	R A M	
2 2 1 6	グラフィックコントローラ	20
2 2 1 8	ディスプレイデバイス	
2 2 2 0	入出力コントローラ	
2 2 2 2	通信インターフェイス	
2 2 2 4	ハードディスクドライブ	
2 2 2 6	D V D - R O Mドライブ	
2 2 3 0	R O M	
2 2 4 0	入出力チップ	
2 2 4 2	キーボード	

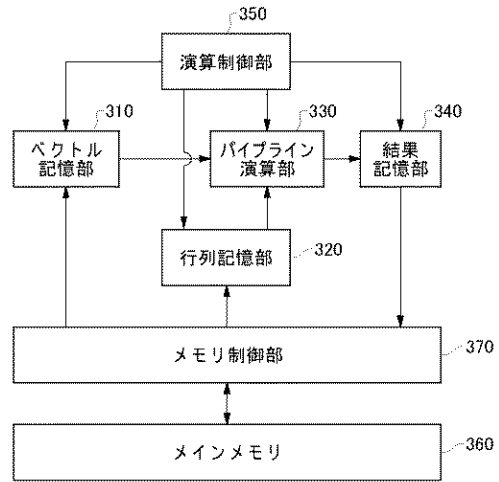
【図1】



【図2】

$$\begin{cases} vc11 = A11 \times vb11 + A12 \times vb21 \\ vc21 = A21 \times vb11 + A22 \times vb21 \\ \vdots \\ vc12 = A11 \times vb12 + A12 \times vb22 \\ vc22 = A21 \times vb12 + A22 \times vb22 \\ \vdots \end{cases}$$

【図3】



300

【図4】

サイクル	演算
0	vctmp1 = A11 × vb11
1	vctmp2 = A11 × vb12
2	vc11 = A12 × vb21 + vctmp1
3	vc12 = A12 × vb22 + vctmp2
4	vctmp1 = A21 × vb11
5	vctmp2 = A21 × vb12
6	vc21 = A22 × vb21 + vctmp1
7	vc22 = A22 × vb22 + vctmp2

【図6】

サイクル	演算
0	vctmp1 = A11 × vb11
1	vctmp2 = A21 × vb11
2	vc11 = A12 × vb21 + vctmp1
3	vc21 = A22 × vb21 + vctmp2
4	vctmp1 = A11 × vb12
5	vctmp2 = A21 × vb12
6	vc12 = A12 × vb22 + vctmp1
7	vc22 = A22 × vb22 + vctmp2

【図5】

サイクル	演算
0	vctmp1 = A11 × vb11
1	vctmp2 = A11 × vb12
2	vctmp3 = A21 × vb11
3	vctmp4 = A21 × vb12
4	vc11 = A12 × vb21 + vctmp1
5	vc12 = A12 × vb22 + vctmp2
6	vc21 = A22 × vb21 + vctmp3
7	vc22 = A22 × vb22 + vctmp4

【図7】

サイクル	演算
0	vctmp1 = A11 × vb11
1	vctmp2 = A21 × vb11
2	vctmp3 = A11 × vb12
3	vctmp4 = A21 × vb12
4	vc11 = A12 × vb21 + vctmp1
5	vc21 = A22 × vb21 + vctmp2
6	vc12 = A12 × vb22 + vctmp3
7	vc22 = A22 × vb22 + vctmp4

【 図 8 】

