

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第6060094号
(P6060094)

(45) 発行日 平成29年1月11日(2017.1.11)

(24) 登録日 平成28年12月16日(2016.12.16)

(51) Int.Cl.

G06F 7/24 (2006.01)

F 1

G06F 7/24

C

請求項の数 68 (全 22 頁)

(21) 出願番号 特願2013-554591 (P2013-554591)
 (86) (22) 出願日 平成24年2月16日 (2012.2.16)
 (65) 公表番号 特表2014-511526 (P2014-511526A)
 (43) 公表日 平成26年5月15日 (2014.5.15)
 (86) 國際出願番号 PCT/US2012/025375
 (87) 國際公開番号 WO2012/112744
 (87) 國際公開日 平成24年8月23日 (2012.8.23)
 審査請求日 平成27年2月16日 (2015.2.16)
 (31) 優先権主張番号 13/031,056
 (32) 優先日 平成23年2月18日 (2011.2.18)
 (33) 優先権主張国 米国(US)

(73) 特許権者 509123208
 アビニシオ テクノロジー エルエルシー
 アメリカ合衆国 O 2 4 2 1 マサチュー
 セツツ州 レキシントン スプリング ス
 トリート 2 0 1
 (74) 代理人 100079108
 弁理士 稲葉 良幸
 (74) 代理人 100109346
 弁理士 大貫 敏史
 (72) 発明者 スタンフィル, クレイグ ダブリュ.
 アメリカ合衆国, マサチューセツツ州 O
 1 7 7 3, リンカーン, ハックルベリー
 ヒル ロード 4 3

最終頁に続く

(54) 【発明の名称】 ソート

(57) 【特許請求の範囲】

【請求項 1】

複数のデータ要素を受信する工程と、
 前記受信された複数のデータ要素をバッファに格納する工程と、
 前記複数のデータ要素の各個別のデータ要素を受信した後で、前記バッファに格納された前記データ要素に境界条件を適用する工程であって、前記境界条件は、前記バッファに格納された最大データ要素と前記バッファに格納された最小データ要素との許容差を指定する、工程と、

前記境界条件を適用した結果に基づいて、前記バッファから1つまたは複数のデータ要素をイジェクトする工程であって、前記イジェクトされたデータ要素は、既定の順番に従ってソートされたデータ要素の出力ストリームを形成する、工程とを含む、コンピュータ実装方法。

【請求項 2】

前記バッファから前記1つまたは複数のデータ要素をイジェクトする工程は、
 前記バッファから少なくとも前記最小データ要素をイジェクトする工程を含む、請求項1に記載の方法。

【請求項 3】

前記バッファから前記1つまたは複数のデータ要素をイジェクトする工程は、
 前記バッファから少なくとも前記最大データ要素をイジェクトする工程を含む、請求項1に記載の方法。

【請求項 4】

受信された各データ要素は、データ記録生成時間を表す、請求項 1 に記載の方法。

【請求項 5】

前記データ要素の出力ストリームは、新しい順にソートされる、請求項 4 に記載の方法。

【請求項 6】

前記受信された複数のデータ要素の前記データ要素は、部分的にソートされる、請求項 1 に記載の方法。

【請求項 7】

前記受信された複数のデータ要素の前記データ要素は、非同期的に前記バッファに到達する、請求項 1 に記載の方法。 10

【請求項 8】

受信された複数のデータ要素をバッファに格納する工程は、

前記受信された複数のデータ要素のストリームの各個別の要素を、前記バッファのソート順序によって決定された前記バッファの位置に挿入する工程を含む、請求項 1 に記載の方法。

【請求項 9】

前記バッファに格納された前記データ要素に境界条件を適用する工程は、

前記バッファに格納された前記データ要素から前記最小データ要素及び前記最大データ要素を特定する工程と、 20

前記最小データ要素と前記最大データ要素との差が前記許容差を超えているかどうかを決定する工程とを含む、請求項 1 に記載の方法。

【請求項 10】

前記境界条件に基づいて、前記バッファから 1 つまたは複数のデータ要素をイジェクトする工程は、

前記最小データ要素と前記最大データ要素との前記差が前記許容差を超えている場合に、前記バッファから前記 1 つまたは複数のデータ要素をイジェクトする工程を含む、請求項 9 に記載の方法。

【請求項 11】

前記バッファに格納された前記データ要素に境界条件を適用する工程は、 30

前記バッファの最大サイズを超えているかどうかを決定する工程を含む、請求項 1 に記載の方法。

【請求項 12】

前記境界条件に基づいて、前記バッファから 1 つまたは複数のデータ要素をイジェクトする工程は、

前記バッファの前記最大サイズを超えている場合に、前記バッファから前記 1 つまたは複数のデータ要素をイジェクトする工程を含む、請求項 11 に記載の方法。

【請求項 13】

前記バッファに格納された前記データ要素に境界条件を適用する工程は、

新たに受信されたデータ要素がカットオフ値を超えているかどうかを決定する工程を含む、請求項 1 に記載の方法。 40

【請求項 14】

前記新たに受信されたデータ要素が前記カットオフ値を超えていない場合に、前記新たに受信されたデータ要素をバッファリングする工程をさらに含む、請求項 13 に記載の方法。

【請求項 15】

前記境界条件に基づいて、前記バッファから 1 つまたは複数のデータ要素をイジェクトする工程は、

前記新たに受信されたデータ要素が前記カットオフ値を超えていない場合に、前記バッファから生成された前記 1 つまたは複数のデータ要素として、前記新たに受信されたデータ。 50

タ要素をイジェクトする工程を含む、請求項 1 3 に記載の方法。

【請求項 1 6】

前記受信された複数のデータ要素の前記個別のデータ要素の 1 つまたは複数は、グラフベースのコンピューテーションシステムのコンポーネントに入力されるデータ記録から導出される、請求項 1 に記載の方法。

【請求項 1 7】

複数のデータ要素を受信する工程と、

前記複数のデータ要素の各個別のデータ要素を受信した後で、前記受信された前記データ要素に境界条件を適用する工程であって、前記境界条件は、前記受信された個別のデータ要素とバッファに格納された最小又は最大データ要素との許容差を指定する、工程と、

前記受信された個別のデータ要素と前記バッファに格納された前記最小又は最大データ要素との差が前記境界条件によって指定される前記許容差を超えていない場合、前記受信された複数のデータ要素の 1 つまたは複数のデータ要素を前記バッファに格納する工程と、

前記受信された個別のデータ要素と前記バッファに格納された前記最小又は最大データ要素との前記差が前記境界条件によって指定される前記許容差を超えている場合、前記バッファから 1 つまたは複数のデータ要素をイジェクトする工程であって、前記イジェクトされたデータ要素は、既定の順番に従ってソートされたデータ要素の出力ストリームを形成する、工程とを含む、コンピュータ実装方法。

【請求項 1 8】

データ要素をソートするためのコンピュータプログラムを格納するコンピュータ可読記憶装置であって、前記コンピュータプログラムは、コンピューティングシステムに、

複数のデータ要素を受信させるための命令と、

前記受信された複数のデータ要素をバッファに格納させるための命令と、

前記受信されたデータ要素のストリームを処理させるための命令とを含み、前記処理する工程は、

前記複数のデータ要素の各個別のデータ要素を受信した後で、前記バッファに格納された前記データ要素に境界条件を適用する工程であって、前記境界条件は、前記バッファに格納された最大データ要素と前記バッファに格納された最小データ要素との許容差を指定する、工程と、

前記境界条件を適用した結果に基づいて、前記バッファから 1 つまたは複数のデータ要素をイジェクトする工程であって、前記イジェクトされたデータ要素は、既定の順番に従ってソートされたデータ要素の出力ストリームを形成する、工程とを含む、コンピュータ可読記憶装置。

【請求項 1 9】

データ要素をソートするためのコンピューティングシステムであって、

複数のデータ要素を受信するための入力デバイスと、

前記受信された複数のデータ要素をバッファに格納するためのデータ格納システムと、

前記データ格納システムと結合され、前記受信された複数のデータ要素を処理するよう構成された少なくとも 1 つのプロセッサとを含み、前記処理する工程は、

前記複数のデータ要素の各個別のデータ要素を受信した後で、前記バッファに格納された前記データ要素に境界条件を適用する工程であって、前記境界条件は、前記バッファに格納された最大データ要素と前記バッファに格納された最小データ要素との許容差を指定する、工程と、

前記境界条件を適用した結果に基づいて、前記バッファから 1 つまたは複数のデータ要素をイジェクトする工程であって、前記イジェクトされたデータ要素は、既定の順番に従ってソートされたデータ要素の出力ストリームを形成する、工程とを含む、コンピューティングシステム。

【請求項 2 0】

データ要素をソートするためのコンピューティングシステムであって、

10

20

30

40

50

複数のデータ要素を受信するための手段と、
前記受信された複数のデータ要素を格納するための手段と、
前記受信された複数のデータ要素を処理するための手段とを含み、前記処理する工程は
、
前記データ要素のストリームの各個別のデータ要素を受信した後で、バッファに格納された前記データ要素に境界条件を適用する工程であって、前記境界条件は、前記バッファに格納された最大データ要素と前記バッファに格納された最小データ要素との許容差を指定する、工程と、

前記境界条件を適用した結果に基づいて、前記バッファから 1 つまたは複数のデータ要素をイジェクトする工程であって、前記イジェクトされたデータ要素は、既定の順番に従ってソートされたデータ要素の出力ストリームを形成する、工程とを含む、コンピューティングシステム。
10

【請求項 2 1】

前記バッファから前記 1 つまたは複数のデータ要素をイジェクトする工程は、
前記バッファから少なくとも前記最小データ要素をイジェクトする工程を含む、請求項 1 9 に記載のコンピューティングシステム。

【請求項 2 2】

前記バッファから前記 1 つまたは複数のデータ要素をイジェクトする工程は、
前記バッファから少なくとも前記最大データ要素をイジェクトする工程を含む、請求項 1 9 に記載のコンピューティングシステム。
20

【請求項 2 3】

受信された各データ要素は、データ記録生成時間を表す、請求項 1 9 に記載のコンピューティングシステム。

【請求項 2 4】

前記データ要素の出力ストリームは、新しい順にソートされる、請求項 2 3 に記載のコンピューティングシステム。

【請求項 2 5】

前記受信された複数のデータ要素の前記データ要素は、部分的にソートされる、請求項 1 9 に記載のコンピューティングシステム。

【請求項 2 6】

前記受信された複数のデータ要素の前記データ要素は、非同期的に前記バッファに到達する、請求項 1 9 に記載のコンピューティングシステム。
30

【請求項 2 7】

受信された複数のデータ要素をバッファに格納する工程は、
前記受信された複数のデータ要素のストリームの各個別の要素を、前記バッファのソート順序によって決定された前記バッファの位置に挿入する工程を含む、請求項 1 9 に記載のコンピューティングシステム。

【請求項 2 8】

前記バッファに格納された前記データ要素に境界条件を適用する工程は、
前記バッファに格納された前記データ要素から前記最小データ要素及び前記最大データ要素を特定する工程と、
前記最小データ要素と前記最大データ要素との差が前記許容差を超えているかどうかを決定する工程とを含む、請求項 1 9 に記載のコンピューティングシステム。
40

【請求項 2 9】

前記境界条件に基づいて、前記バッファから 1 つまたは複数のデータ要素をイジェクトする工程は、
前記最小データ要素と前記最大データ要素との前記差が前記許容差を超えている場合に、前記バッファから前記 1 つまたは複数のデータ要素をイジェクトする工程を含む、請求項 2 8 に記載のコンピューティングシステム。

【請求項 3 0】

50

前記バッファに格納された前記データ要素に境界条件を適用する工程は、前記バッファの最大サイズを超えていたかどうかを決定する工程を含む、請求項19に記載のコンピューティングシステム。

【請求項31】

前記境界条件に基づいて、前記バッファから1つまたは複数のデータ要素をイジェクトする工程は、

前記バッファの前記最大サイズを超えていた場合に、前記バッファから前記1つまたは複数のデータ要素をイジェクトする工程を含む、請求項30に記載のコンピューティングシステム。

【請求項32】

前記バッファに格納された前記データ要素に境界条件を適用する工程は、

新たに受信されたデータ要素がカットオフ値を超えていたかどうかを決定する工程を含む、請求項19に記載のコンピューティングシステム。

【請求項33】

前記処理する工程は、

前記新たに受信されたデータ要素が前記カットオフ値を超えていない場合に、前記新たに受信されたデータ要素をバッファリングする工程をさらに含む、請求項32に記載のコンピューティングシステム。

【請求項34】

前記境界条件に基づいて、前記バッファから1つまたは複数のデータ要素をイジェクトする工程は、

前記新たに受信されたデータ要素が前記カットオフ値を超えていない場合に、前記バッファから生成された前記1つまたは複数のデータ要素として、前記新たに受信されたデータ要素をイジェクトする工程を含む、請求項32に記載のコンピューティングシステム。

【請求項35】

前記受信された複数のデータ要素の前記個別のデータ要素の1つまたは複数は、グラフベースのコンピューテーションシステムのコンポーネントに入力されるデータ記録から導出される、請求項19に記載のコンピューティングシステム。

【請求項36】

前記バッファから前記1つまたは複数のデータ要素をイジェクトする工程は、

前記バッファから少なくとも前記最小データ要素をイジェクトする工程を含む、請求項20に記載のコンピューティングシステム。

【請求項37】

前記バッファから前記1つまたは複数のデータ要素をイジェクトする工程は、

前記バッファから少なくとも前記最大データ要素をイジェクトする工程を含む、請求項20に記載のコンピューティングシステム。

【請求項38】

受信された各データ要素は、データ記録生成時間を表す、請求項20に記載のコンピューティングシステム。

【請求項39】

前記データ要素の出力ストリームは、新しい順にソートされる、請求項38に記載のコンピューティングシステム。

【請求項40】

前記受信された複数のデータ要素の前記データ要素は、部分的にソートされる、請求項20に記載のコンピューティングシステム。

【請求項41】

前記受信された複数のデータ要素の前記データ要素は、非同期的に前記バッファに到達する、請求項20に記載のコンピューティングシステム。

【請求項42】

受信された複数のデータ要素をバッファに格納する工程は、

10

20

30

40

50

前記受信された複数のデータ要素のストリームの各個別の要素を、前記バッファのソート順序によって決定された前記バッファの位置に挿入する工程を含む、請求項 20 に記載のコンピューティングシステム。

【請求項 4 3】

前記バッファに格納された前記データ要素に境界条件を適用する工程は、

前記バッファに格納された前記データ要素から前記最小データ要素及び前記最大データ要素を特定する工程と、

前記最小データ要素と前記最大データ要素との差が前記許容差を超えているかどうかを決定する工程とを含む、請求項 20 に記載のコンピューティングシステム。

【請求項 4 4】

前記境界条件に基づいて、前記バッファから 1 つまたは複数のデータ要素をイジェクトする工程は、

前記最小データ要素と前記最大データ要素との前記差が前記許容差を超えている場合に、前記バッファから前記 1 つまたは複数のデータ要素をイジェクトする工程を含む、請求項 4 3 に記載のコンピューティングシステム。

【請求項 4 5】

前記バッファに格納された前記データ要素に境界条件を適用する工程は、

前記バッファの最大サイズを超えているかどうかを決定する工程を含む、請求項 20 に記載のコンピューティングシステム。

【請求項 4 6】

前記境界条件に基づいて、前記バッファから 1 つまたは複数のデータ要素をイジェクトする工程は、

前記バッファの前記最大サイズを超えている場合に、前記バッファから前記 1 つまたは複数のデータ要素をイジェクトする工程を含む、請求項 4 5 に記載のコンピューティングシステム。

【請求項 4 7】

前記バッファに格納された前記データ要素に境界条件を適用する工程は、

新たに受信されたデータ要素がカットオフ値を超えているかどうかを決定する工程を含む、請求項 20 に記載のコンピューティングシステム。

【請求項 4 8】

前記処理する工程は、

前記新たに受信されたデータ要素が前記カットオフ値を超えていない場合に、前記新たに受信されたデータ要素をバッファリングする工程をさらに含む、請求項 4 7 に記載のコンピューティングシステム。

【請求項 4 9】

前記境界条件に基づいて、前記バッファから 1 つまたは複数のデータ要素をイジェクトする工程は、

前記新たに受信されたデータ要素が前記カットオフ値を超えていない場合に、前記バッファから生成された前記 1 つまたは複数のデータ要素として、前記新たに受信されたデータ要素をイジェクトする工程を含む、請求項 4 7 に記載のコンピューティングシステム。

【請求項 5 0】

前記受信された複数のデータ要素の前記個別のデータ要素の 1 つまたは複数は、グラフベースのコンピューテーションシステムのコンポーネントに入力されるデータ記録から導出される、請求項 20 に記載のコンピューティングシステム。

【請求項 5 1】

前記バッファから前記 1 つまたは複数のデータ要素をイジェクトする工程は、

前記バッファから少なくとも前記最小データ要素をイジェクトする工程を含む、請求項 18 に記載のコンピュータ可読記憶装置。

【請求項 5 2】

前記バッファから前記 1 つまたは複数のデータ要素をイジェクトする工程は、

10

20

30

40

50

前記バッファから少なくとも前記最大データ要素をイジェクトする工程を含む、請求項 18 に記載のコンピュータ可読記憶装置。

【請求項 5 3】

受信された各データ要素は、データ記録生成時間を表す、請求項 18 に記載のコンピュータ可読記憶装置。

【請求項 5 4】

前記データ要素の出力ストリームは、新しい順にソートされる、請求項 5 3 に記載のコンピュータ可読記憶装置。

【請求項 5 5】

前記受信された複数のデータ要素の前記データ要素は、部分的にソートされる、請求項 18 に記載のコンピュータ可読記憶装置。 10

【請求項 5 6】

前記受信された複数のデータ要素の前記データ要素は、非同期的に前記バッファに到達する、請求項 18 に記載のコンピュータ可読記憶装置。

【請求項 5 7】

受信された複数のデータ要素をバッファに格納する工程は、

前記受信された複数のデータ要素のストリームの各個別の要素を、前記バッファのソート順序によって決定された前記バッファの位置に挿入する工程を含む、請求項 18 に記載のコンピュータ可読記憶装置。 20

【請求項 5 8】

前記バッファに格納された前記データ要素に境界条件を適用する工程は、

前記バッファに格納された前記データ要素から前記最小データ要素及び前記最大データ要素を特定する工程と、

前記最小データ要素と前記最大データ要素との差が前記許容差を超えているかどうかを決定する工程とを含む、請求項 18 に記載のコンピュータ可読記憶装置。 20

【請求項 5 9】

前記境界条件に基づいて、前記バッファから 1 つまたは複数のデータ要素をイジェクトする工程は、

前記最小データ要素と前記最大データ要素との前記差が前記許容差を超えている場合に、前記バッファから前記 1 つまたは複数のデータ要素をイジェクトする工程を含む、請求項 5 8 に記載のコンピュータ可読記憶装置。 30

【請求項 6 0】

前記バッファに格納された前記データ要素に境界条件を適用する工程は、

前記バッファの最大サイズを超えているかどうかを決定する工程を含む、請求項 18 に記載のコンピュータ可読記憶装置。 20

【請求項 6 1】

前記境界条件に基づいて、前記バッファから 1 つまたは複数のデータ要素をイジェクトする工程は、

前記バッファの前記最大サイズを超えている場合に、前記バッファから前記 1 つまたは複数のデータ要素をイジェクトする工程を含む、請求項 6 0 に記載のコンピュータ可読記憶装置。 40

【請求項 6 2】

前記バッファに格納された前記データ要素に境界条件を適用する工程は、

新たに受信されたデータ要素がカットオフ値を超えているかどうかを決定する工程を含む、請求項 18 に記載のコンピュータ可読記憶装置。 20

【請求項 6 3】

前記コンピュータプログラムは、前記コンピュティングシステムに、

前記新たに受信されたデータ要素が前記カットオフ値を超えていない場合に、前記新たに受信されたデータ要素をバッファリングさせるための命令をさらに含む、請求項 6 2 に記載のコンピュータ可読記憶装置。 50

【請求項 6 4】

前記境界条件に基づいて、前記バッファから 1 つまたは複数のデータ要素をイジェクトする工程は、

前記新たに受信されたデータ要素が前記カットオフ値を超えていない場合に、前記バッファから生成された前記 1 つまたは複数のデータ要素として、前記新たに受信されたデータ要素をイジェクトする工程を含む、請求項 6 2 に記載のコンピュータ可読記憶装置。

【請求項 6 5】

前記受信された複数のデータ要素の前記個別のデータ要素の 1 つまたは複数は、グラフベースのコンピューテーションシステムのコンポーネントに入力されるデータ記録から導出される、請求項 1 8 に記載のコンピュータ可読記憶装置。

10

【請求項 6 6】

データ要素をソートするためのコンピューティングシステムであって、

複数のデータ要素を受信するための入力デバイスと、

前記受信された複数のデータ要素を処理するよう構成された少なくとも 1 つのプロセッサとを含み、前記処理する工程は、

前記複数のデータ要素の各個別のデータ要素を受信した後で、前記受信されたデータ要素に境界条件を適用する工程であって、前記境界条件は、前記受信された個別のデータ要素とバッファに格納された最小又は最小データ要素との許容差を指定する、工程と、

前記受信された個別のデータ要素と前記バッファに格納された前記最小又は最大データ要素との差が前記境界条件によって指定される前記許容差を超えていない場合、前記受信された複数のデータ要素の 1 つまたは複数のデータ要素を前記バッファに格納する工程と、

20

前記受信された個別のデータ要素と前記バッファに格納された前記最小又は最大データ要素との前記差が前記境界条件によって指定される前記許容差を超えている場合、前記バッファから 1 つまたは複数のデータ要素をイジェクトする工程であって、前記イジェクトされたデータ要素は、既定の順番に従ってソートされたデータ要素の出力ストリームを形成する、工程とを含む、コンピューティングシステム。

【請求項 6 7】

データ要素をソートするためのコンピューティングシステムであって、

複数のデータ要素を受信するための手段と、

前記複数のデータ要素の各個別のデータ要素を受信した後で、前記受信されたデータ要素に境界条件を適用するための手段であって、前記境界条件は、前記受信された個別のデータ要素とバッファに格納された最小又は最小データ要素との許容差を指定する、手段と、

30

前記受信された個別のデータ要素と前記バッファに格納された前記最小又は最大データ要素との差が前記境界条件によって指定される前記許容差を超えていない場合、前記受信された複数のデータ要素の 1 つまたは複数のデータ要素を前記バッファに格納する手段と、

前記受信された個別のデータ要素と前記バッファに格納された前記最小又は最大データ要素との前記差が前記境界条件によって指定される前記許容差を超えている場合、前記バッファから 1 つまたは複数のデータ要素をイジェクトする手段であって、前記イジェクトされたデータ要素は、既定の順番に従ってソートされたデータ要素の出力ストリームを形成する、手段とを含む、コンピューティングシステム。

40

【請求項 6 8】

データ要素をソートするためのコンピュータプログラムを格納するコンピュータ可読記憶装置であって、前記コンピュータプログラムは、コンピューティングシステムに、

複数のデータ要素を受信させるための命令と、

前記複数のデータ要素の各個別のデータ要素を受信した後で、前記受信されたデータ要素に境界条件を適用させるための命令であって、前記境界条件は、前記受信された個別のデータ要素とバッファに格納された最小又は最小データ要素との許容差を指定する、命令

50

と、

前記受信された個別のデータ要素と前記バッファに格納された前記最小又は最大データ要素との差が前記境界条件によって指定される前記許容差を超えていない場合、前記受信された複数のデータ要素の1つまたは複数のデータ要素を格納させるための命令と、

前記受信された個別のデータ要素と前記バッファに格納された前記最小又は最大データ要素との前記差が前記境界条件によって指定される前記許容差を超えている場合、前記バッファから1つまたは複数のデータ要素をイジェクトさせるための命令であって、前記イジェクトされたデータ要素は、既定の順番に従ってソートされたデータ要素の出力ストリームを形成する、命令とを含む、コンピューティングシステム。

【発明の詳細な説明】

10

【技術分野】

【0001】

関連出願の相互参照

本願は、2011年2月18日に出願された「Sorting」と題する米国特許出願第13/031,056号の優先権を主張し、同特許は、その全内容が参照により本明細書に組み込まれる。

【0002】

本開示は、ソートに関連する。

【背景技術】

【0003】

20

さまざまなデータ処理技術と同様に、ソート技術は、多くのアプリケーションにおいて重要な役割を果たし得る。例えば、データ傾向および相違点は、ソートされたデータリストが提示され次第（1つまたは複数のソートパラメータに基づいて）、ビューアによって迅速に特定され得る。例えば、昇順または降順にリストをソートすることによって、データ境界、範囲および他の数量を視覚的に認識できる可能性がある。ソートアルゴリズムの効率は、ソートされたリストの生成に必要な、実行されるデータ要素比較およびデータ要素の動きの数の関数によってなどの多くの方法論を通じて評価することができる。

【発明の概要】

【課題を解決するための手段】

【0004】

30

一態様では、一般に、コンピュータ実装方法は、受信されたデータ要素のストリームをバッファに格納する工程と、データ要素のストリームの各個別のデータ要素を受信した後で、バッファに格納されたデータ要素に境界条件を適用する工程と、境界条件に基づいて、既定の順序に従ってソートされたデータ要素の出力ストリームとして、バッファから1つまたは複数のデータ要素を生成する工程とを含む。

【0005】

態様は、以下の1つまたは複数を含み得る。

【0006】

バッファから1つまたは複数のデータ要素を生成する工程は、バッファから少なくとも最小データ要素を生成する工程を含み得る。バッファから1つまたは複数のデータ要素を生成する工程は、バッファから少なくとも最大データ要素を生成する工程を含み得る。受信された各データは、データ記録生成時間を表し得る。データ要素の出力ストリームは、新しい順にソートすることができる。受信されたデータ要素のストリームのデータ要素は、部分的にソートすることができる。受信されたデータ要素のストリームのデータ要素は、実質的に非同期的にバッファに到達することができる。受信されたデータ要素のストリームをバッファに格納する工程は、受信されたデータ要素のストリームの各個別の要素を、バッファのソート順序によって決定されたバッファの位置に挿入する工程を含み得る。

【0007】

バッファに格納されたデータ要素に境界条件を適用する工程は、バッファに格納されたデータ要素から最小データ要素を特定する工程と、最小データ要素と受信された個別のデ

40

50

ータ要素との差が閾値を超えていたり、バッファから1つまたは複数のデータ要素を生成する工程は、1つまたは複数のデータ要素が閾値を超えていたり、バッファから1つまたは複数のデータ要素を生成する工程を含み得る。バッファに格納されたデータ要素に境界条件を適用する工程は、バッファの最大サイズを超えていたり、バッファから1つまたは複数のデータ要素を生成する工程は、バッファの最大サイズを超えていたり、バッファから1つまたは複数のデータ要素を生成する工程を含み得る。

【0008】

バッファに格納されたデータ要素に境界条件を適用する工程は、新たに受信されたデータ要素がカットオフ値を超えていたり、バッファから1つまたは複数のデータ要素を生成する工程は、新たに受信されたデータ要素がカットオフ値を超えていたり、バッファに格納されたデータ要素を生成する工程は、新たに受信されたデータ要素が閾値を超えていたり、バッファから生成された1つまたは複数のデータ要素として、新たに受信されたデータ要素を生成する工程を含み得る。バッファは、スキップリストデータ構造によって実装することができる。バッファは、ツリープ(treap)データ構造によって実装することができる。受信されたデータ要素のストリームの個別のデータ要素の1つまたは複数は、グラフベースのコンピューテーションシステムのコンポーネントに入力されるデータ記録から導出することができる。

10

【0009】

別の態様では、一般に、コンピュータ実装方法は、データ要素のストリームの各個別のデータ要素を受信した後で、受信されたデータ要素に境界条件を適用する工程と、境界条件の第1の結果に基づいて、データ要素のストリームの1つまたは複数のデータ要素を格納する工程と、境界条件の第2の結果に基づいて、既定の順序に従ってソートされたデータ要素の出力ストリームとして、1つまたは複数のデータ要素を生成する工程とを含み得る。

20

【0010】

別の態様では、一般に、データ要素をソートするためのコンピュータプログラムを格納するコンピュータ可読記憶媒体は、コンピューティングシステムに、受信されたデータ要素のストリームをバッファに格納させるための命令と、受信されたデータ要素のストリームを処理させるための命令とを含み得る。受信されたデータ要素のストリームを処理する工程は、データ要素のストリームの各個別のデータ要素を受信した後で、バッファに格納されたデータ要素に境界条件を適用する工程と、境界条件に基づいて、既定の順序に従ってソートされたデータ要素の出力ストリームとして、バッファから1つまたは複数のデータ要素を生成する工程とを含み得る。

30

【0011】

別の態様では、一般に、データ要素をソートするためのコンピューティングシステムは、受信されたデータ要素のストリームをバッファに格納するためのデータ格納システムと、受信されたデータ要素のストリームを処理するよう構成された少なくとも1つのプロセッサとを含み得る。処理する工程は、データ要素のストリームの各個別のデータ要素を受信した後で、バッファに格納されたデータ要素に境界条件を適用する工程と、境界条件に基づいて、既定の順序に従ってソートされたデータ要素の出力ストリームとして、バッファから1つまたは複数のデータ要素を生成する工程とを含み得る。

40

【0012】

別の態様では、一般に、データ要素をソートするためのコンピューティングシステムは、受信されたデータ要素のストリームを格納するための手段と、受信されたデータ要素のストリームを処理するための手段とを含み得る。処理する工程は、データ要素のストリームの各個別のデータ要素を受信した後で、バッファに格納されたデータ要素に境界条件を適用する工程と、境界条件に基づいて、既定の順序に従ってソートされたデータ要素の出

50

カストリームとして、バッファから 1 つまたは複数のデータ要素を生成する工程とを含み得る。

【 0 0 1 3 】

態様は、以下の利点の 1 つまたは複数を含み得る。

【 0 0 1 4 】

大部分がソートされる受信されたデータ要素のストリームに関するいくつかの実装形態では、要素の大多数とともに適正な順序にはない一部を除いて、要素は、大部分がソートされた順序にあり得る。従来の技術は、要素をソートするために、かなりのコンピューティングリソースおよび処理時間を消費する恐れがある。いくつかの例では、本明細書に記載される技術は、データ要素の大多数が適切に順序付けられて到達し、従って、コンピュータリソースおよび処理時間を節約することができるという事実を利用することができる。ストリームの各個別のデータ要素を受信した後で、バッファに格納されたデータ要素に境界条件を適用することによって、既定の順序に従ってソートされたデータ要素の出力ストリームを生成することができる。

【 0 0 1 5 】

本発明の他の特徴および利点は、以下の説明および特許請求の範囲から明らかになるであろう。

【 図面の簡単な説明 】

【 0 0 1 6 】

【 図 1 】 ウィンドウソートシステムのブロック図である。

【 図 2 】 一連のウィンドウソートオペレーションを経たソートバッファを示す。

【 図 3 】 グラフベースのコンピューテーションを管理するためのコンピューティングシステムのブロック図である。

【 図 4 】 ウィンドウソートプロセスのフローチャートである。

【 発明を実施するための形態 】

【 0 0 1 7 】

図 1 を参照すると、例示的なソートシステム 100 は、ソートモジュール 108 (この構成では、別のコンピュータシステム 108 によって実行される) にデータ要素またはアイテムのストリーム 106 a ~ n (総称して 106) を提供するコンピュータシステム 104 (例えば、1 つまたは複数のクライアントシステムへのサービスを行うウェブサーバなどのサーバ) 上で実行されるデータソースアプリケーション 102 を含む。要素 106 は、ある一定の時間 (例えば、24 時間) にわたってソースアプリケーション 102 によって生成することができ、生成またはストレージ (例えば、記憶装置) から検索されると送信することができる。要素 106 は、例えば、ソースアプリケーション 102 によって生成されたデータ記録 (図示せず) の 1 つまたは複数の属性に割り当てられた値など、さまざまなタイプの情報を表し得る。いくつかの実装形態では、要素 106 は、ソースアプリケーション 102 によって生成された 1 つまたは複数のデータ記録から演算された値を表し得る。要素 106 によって表され得る他の情報は、例えば、ソースアプリケーション 102 によってデータ記録が作成された際のそれぞれの時間などのタイムスタンプを含み得る。例えば、ソースアプリケーション 102 は、通話データ記録を作成し、特定の時間の間に行われた通話についての情報を記録することができる。そのようなシナリオでは、それぞれのタイムスタンプされたデータ要素は、対応する通話が終了した正確な時間を表し得る。

【 0 0 1 8 】

別の実装形態では、それぞれのタイムスタンプされたデータ要素は、対応する通話が開始された正確な時間を表し得る。通話が開始された時間は、通話が終了した時間から通話の長さを引くことによって計算することができる。そのような計算を容易にするため、通話の長さについての情報は、ソースアプリケーションによって生成される通話記録に格納することができる。

【 0 0 1 9 】

10

20

30

40

50

いくつかの構成では、ストリーム 106 内のデータ要素の個別の配置は、さまざまな形を取ることができる。例えば、データ要素の順序は、完全に無作為の順序または幾つか決定論的な順序（例えば、部分的にソートされる、大部分がまたはほぼソートされる、など）であり得る。要素の大部分がソートされるシナリオでは、比較的ほんのわずかな比率の要素がソートされないかまたは順序に狂いが生じている場合がある。そのため、比較的複数のデータ要素（例えば、数百万）を有するストリームでは、要素のおよそ 0.1 ~ 10 % が順序に狂いが生じている可能性がある。一例として、ソースアプリケーション 102 が通話の終了時間を表すタイムスタンプされたデータ要素を生成するシナリオでは、要素は、実質的に非同期的にソートモジュール 108 に到達することができ、時折、順序に狂いが生じている状態で到達し得る。例えば、第 1 の時間の間にソースアプリケーション 102 によって生成されたデータは、別の時間（第 1 の時間よりも遅い）の間に生成されたデータの後にソートモジュール 108 に到達することができる。10

【0020】

大部分がソートされるストリームでは、要素の大多数とともに適正な順序にはない一部を除いて、要素は、大部分がソートされた順序にあり得る。ストリーム内の適正な位置にそのような外れ値要素を挿入するため、従来のソート技術は、通常、受信されたストリームのすべてのデータ要素を処理する。そのため、これらの技術は、要素をソートするためには、かなりのコンピューティングリソースおよび処理時間を消費する恐れがあり、データ要素の数に比例して増加し得るコンピュータの損失を生み出す恐れがある。しかし、データ要素の大多数が適切に順序付けられて到達するという事実を利用することによって、ソート技術を開発し、コンピュータリソースおよび処理時間を節約することができる。20

【0021】

例示的なソート技術では、ソートモジュール 108 は、ソートバッファに入力要素を一時的に格納し、適切な時間にバッファから要素 118a ~ c（総称して 118）をイジェクトすることによって、ほぼソートされた入力ストリームをソートする。要素イジェクションは、イジェクション優先度順（例えば、昇順、降順など）に従って起こり得る。さらに、イジェクションは、要素に課された条件（例えば、バッファに格納された任意の 2 つの要素間の最大許容差）に違反した場合に起こる。イジェクトされた時点で、イジェクトされた要素 118 を使用して、ソートされた要素の出力ストリームを形成することができる。30

【0022】

いくつかの例では、ソートされた出力ストリームは、順序に狂いが生じている要素をソートバッファに一時的に格納し、それらを適切な時間に出力ストリームに解放する一方で、既にソートされた順序にある入力要素を出力ストリームに直接送ることによって生成することができる。

【0023】

ソートモジュール 108 は、例えば、コンピュータシステム 110 上で実行されるソフトウェアアプリケーションとして実装することができ、ソートバッファ 112（または複数のソートバッファ）を制御するための命令を提供することができ、ソートバッファ 112 は、記憶装置 114（例えば、ハードドライブメモリ、ランダムアクセスメモリなど）に格納することができる。記憶装置は、コンピュータシステム 110 の一部としても、別々のスタンドアロンメモリデバイスとしても実装することができる。40

【0024】

例では、ソートバッファ 112 内の格納要素 116 は、例えば、要素がバッファからイジェクトされる順序などの優先度順に従ってソートされる。優先度順は、要素の既定の特性（例えば、要素の数値）に基づき得る。そのため、数値の昇順に従って要素がソートされる場合は、最初に小さな値の要素がイジェクトされ、その後、より大きな値の要素がイジェクトされ得る。降順に従って要素がソートされる場合は、最初に大きな値の要素がイジェクトされ、その後、より小さな値の要素がイジェクトされ得る。ソートバッファで要素が受信されると、要素は、優先度順に従ってソートされ得る。タイムスタンプされたデ50

ータ要素に関するシナリオでは、要素は、古い順または新しい順にソートされ得る。例えば、通話の終了時間を表す要素は、新しい順に配列することができる。

【0025】

イジェクトされたデータ要素118は、ソートバッファ内の格納要素の最小値（または、好ましい出力に応じて、最大値）要素であり得る。タイムスタンプされた要素を用いると、イジェクトされた要素は、ソートバッファ内の格納要素の最新値（または、好ましい出力に応じて、最古値）要素であり得る。イジェクトされた要素は、別のコンピューティングデバイス（例えば、コンピュータシステム122）上で実行されている1つまたは複数のモジュール、アプリケーション（例えば、受信アプリケーション120）などに提供され得る。受信アプリケーション120は、実質的にソートされた出力ストリームを受信して処理するための任意のソフトウェアアプリケーションであり得る。

10

【0026】

一構成では、ソートバッファの一方の端部は、新たに到達する要素を受信するよう構成することができる。ソートバッファを通じて（この特定の端部から）伝播する間、受信されたデータ要素（他の要素とともに）は、要素がバッファの反対側の端部（そこから要素が最終的にイジェクトされる）に近づくにつれて要素が適切なデータ要素間に配置するよう、ソートモジュール108によって処理することができる。

【0027】

入力要素は大部分がまたはほぼ順序付けられるため、ソートバッファ112への挿入は、バッファの端部で起こる場合が多い。そのような挿入オペレーションは、 $O(1)$ 時間を見なすことができる。ソートされた順序で要素が受信される限り、要素は、ソートバッファの連続記憶場所に漸進的に追加される。しかし、順序に狂いが生じている状態で要素が到達すると、必要な挿入時間は、要素の挿入に適切な記憶場所を見つけるのに必要な時間に対応するため、長くなり得る。例えば、順序に狂いが生じている要素を受信するための位置がソートバッファの端部から「 k 」の距離にある場合、適正な位置の特定には、 $O(\log(k))$ 時間が必要とされ得る。特定された時点で、要素の挿入に必要な時間は、一定と見なすことができる。ソートバッファ内に既に「 n 」個の格納データ要素がある場合、挿入位置は、ソートバッファの端部から「 n 」個の位置を超えないものと見なすことができる。それに応じて、最悪の場合の挿入時間は、 $O(\log(n))$ 時間であり得る。

20

【0028】

ソートバッファ112は、ソートされた順序でデータ要素を受信して格納するよう構成されたデータ構造（例えば、優先度付きキューデータ構造）によって実装することができる。新しいデータ要素が到達すると、ソートモジュール108は、ソートバッファの優勢なソート順序によって決定されるように（優先度付きキューとして実装される）、適切な位置に要素を挿入することができる。さらに、ソートモジュール108は、イジェクション優先度順（例えば、バッファの最小または最大データ要素）に基づいてイジェクトされる1つまたは複数のデータ要素を特定することができる。特定された時点で、ソートモジュールは、ソートバッファからの特定された要素のイジェクションを開始することができる。

30

【0029】

いくつかの実装形態では、ソートバッファ112は、バランスの取れた二分木データ構造（例えば、ヒープ二分木データ構造）であり得る。一般に、ヒープ二分木データ構造（単にヒープと呼ばれる場合が多い）は、2つの条件を満たすと見なされる。第1の条件は、子ノードの優先度が親ノードの優先度と少なくとも同程度であることである。それに応じて、木の最上位のノード（すなわち、根ノード）は、最小優先度を有すると見なすことができる。第2の条件は、根ノードから葉のノードまでの異なる経路が多くとも1つのノードレベルだけ高さが異なることである。

40

【0030】

一実装形態では、ソートバッファは、スキップリストデータ構造によって実装すること

50

ができる。スキップリストデータ構造およびその関連アルゴリズムは、リンクされた線形リストの変形例であり、キーを有するデータ要素の頻繁な挿入を要求する状況におけるソート性能を向上させることができる。スキップリストデータ構造は、データ要素（または、ノード）の順序付けられたリンクされた線形リストを含み得、いくつかの要素は、中間データ要素をスキップする追加のポインタを有し、それにより、データ要素の探索の速度および効率を増加する。

【0031】

スキップリストデータ構造としてソートバッファを実装することによって、例えば、バッファから最小データ要素を検索するのに必要な時間は、最小データ要素の検索に対して $O(1)$ 時間であり得る。データ要素を適切に挿入する（例えば、要素は、ごく最近挿入された要素に隣接して挿入される）ための挿入時間は、 $O(1)$ 時間であり得る。挿入位置がごく最近の要素に隣接しない場合は、挿入時間は $O(\log N)$ 時間であり得る。「 k 」要素（例えば、最大値を有する要素）を読み出すためのイジェクション時間は、 $O(k)$ 時間であり得る。スキップリストデータ構造については、その内容はその全体が参照により本明細書に組み込まれる、William Pughによる「Skip lists: A probabilistic alternative to Balanced trees」(Communications of the ACM. 1990年6月, 668~676頁)と称する文書でさらに詳細に説明されている。

【0032】

いくつかの例では、ソートバッファ 112 は、「ツリープ」データ構造によって実装することができる。ツリープデータ構造は、各ノードが無作為に割り当てられた優先度属性およびキーを有する二分探索木である。ノードは、それらのキーに関して、典型的な二分探索木のような順序にある。すなわち、キーに関して、ノードの左側のサブツリーは、ノードのキー未満のキーを有するノードのみを含み、ノードの右側のサブツリーは、ノードのキー以上のキーを有するノードのみを含む。それに加えて、ノードは、それらの優先度属性に関して、「ヒープ順」にあり、その結果、各子ノードは、親ノードの優先度属性と少なくとも同程度である優先度属性を有する。ツリープデータ構造については、その内容はその全体が参照により本明細書に組み込まれる、Aragon, Cecilia R. および Seidel, Raimund による「Randomized Search Trees」(Proc. 30th Symp. Foundations of Computer Science (FOCS 1989年))と称する文書で説明されている。

【0033】

一実装形態では、ソートモジュール 108 は、ソートバッファ 112 上で境界条件を適用させることができ、境界条件を使用して、ソートバッファから要素をイジェクトするかどうか、および、いつソートバッファから要素をイジェクトするか決定する。境界条件は、ソートバッファに許可される値の「ウインドウ」を定義することができる。例えば、ウインドウは、ソートバッファの任意の 2 つの格納データ要素間の最大許容差を表し得る。そのため、ウインドウの「幅」は、ソートバッファで許可される値の範囲を表す。例えば、ウインドウ幅は、ソートバッファの最小要素と最大要素との差であり得る。

【0034】

境界条件は、少なくとも 2 つの結果を有することができる。例えば、第 1 の結果は、境界条件が満たされていることを示し得、第 2 の異なる結果は、境界条件に違反している（すなわち、満たされていない）ことを示し得る。シナリオでは、提供される入力要素が原因でウインドウ幅を超えることにならない限り、境界条件は満たされると見なすことができる。逆に、入力要素が原因でウインドウ幅を超えることになった場合は、境界条件に違反していると見なすことができる。境界条件への違反があり次第、イジェクション優先度順に従ってバッファの 1 つまたは複数の要素がイジェクトされる。

【0035】

一例として、ソートバッファが格納要素として数値を有する（昇順に）シナリオを想定する。ソートバッファに対する値 15 のウインドウ幅は、バッファの最小格納要素と最大格納要素との差が 15 を超えないことを示す。そのため、バッファの現在の最小要素が 3 であり、新しい入力要素が 19 である場合、最小要素と最大要素との差はここでは 16 で

10

20

30

40

50

あり、これはウィンドウ幅を超えており、境界条件に違反していると見なされる。違反が原因で、ソートバッファの1つまたは複数の要素（例えば、最小または最大データ要素）はイジェクトされる。

【0036】

いくつかの例では、ソートモジュール108は、ウィンドウ幅で決定されるように、ソートバッファ112の最小および最大要素を記録する。入力要素が最大要素を超えると、境界条件に違反していると見なされ、ソートモジュールは、ソートバッファに、例えば最小データ要素をイジェクトさせる。

【0037】

10 ウィンドウ幅は、ユーザによって指定されるものなどさまざまなソースから提供することができる。一実装形態では、ウィンドウ幅は、ソートが必要とされる特定のアプリケーションの一般知識に基づく。状況次第では、ウィンドウ幅は、「順序の狂い」の度合い、例えば、どれほど要素の順序に狂いが生じているかについての度合いに相当する。例えば、タイムスタンプされたデータ要素に関する状況について考慮する。ウィンドウ幅は、バッファ内に現在存在するタイムスタンプされた要素（例えば、最小または最大のタイムスタンプされた要素）と比較して、どのくらい遅くまでタイムスタンプされた要素をソートバッファに提供できるかを示す。例えば、ウィンドウ幅（時間の単位で）が4時間であれば、ソートバッファ112は、ソートバッファの最小のタイムスタンプされた要素と比較して、4時間ウィンドウの範囲内のすべてのタイムスタンプされた要素をバッファリング（すなわち、格納）する。4時間の範囲外で到達するタイムスタンプされた要素は、廃棄される、別々に処理されるなどの場合がある。

【0038】

示される例では単一のバッファが使用されているが、複数のバッファアーキテクチャを利用することもできる。例えば、第2のバッファ（図示せず）を使用して、ある一定の時間の間（例えば、24時間）に到達するタイムスタンプされた要素をバッファリングすることができる。24時間が経過した時点で、タイムスタンプされた要素は、拒否される、廃棄される、別々に処理される可能性がある。一実装形態では、24時間の範囲内に属するとタイムスタンプされた要素が24時間経過後に到達する可能性がある。カットオフタイムスタンプ値を慎重に選択することによって、複数の正当な要素が拒否される可能性を低減することができる。いくつかのシナリオでは、少数の正当な要素の損失は、ウィンドウソート技術がコンピュータリソースの効率的な使用および処理時間の削減を提供するアプリケーションにおいて許容可能である。

【0039】

図2は、ウィンドウソートプロセスのオペレーションの間のソートバッファ（例えば、ソートバッファ112）の状態202～218の図表示である。ソートバッファの個別のコンテンツ、ウィンドウ幅（ラベルWで示される）、最小要素（ラベルSで示される）および現在の出力データ要素についての情報は、各状態202～218に対して提供される。この例では、データ要素は昇順で格納され、入力ストリームは昇順でほぼソートされる。ウィンドウソートプロセスの終結時には、出力ストリームは、昇順で実質的にソートされて提供される。図2は単に8個の要素のストリームに対して動作する例示的なウィンドウソートプロセスを示すが、プロセスは、実質的にそれ以上の要素を含む入力ストリームに対して連続して実行することができる。

【0040】

状態202では、ウィンドウ幅は、値15を有するものとして定義される（すなわち、バッファの最小値要素と最大値要素との差は15以上ではない）。初期化の際、ソートバッファの現在の最小要素は値0（S=0）のものである。さらに、現在の出力は「NILL」（すなわち、出力データ要素無し）で示される。

【0041】

状態204では、第1の値3の入力要素がソートバッファに挿入される。バッファ内の3の位置は、ソートバッファの要素の現在の順序（すなわち、昇順）に基づいて決定され

10

20

30

40

50

る。値 3 の要素は、示されるように、0 に隣接して挿入される。ソートバッファの最小値要素と最大値要素との差はここでは 3 であり、これはウィンドウ幅 15 未満である。それに応じて、境界条件に違反しておらず、データ要素をイジェクトする必要はない。この時点では、出力データ要素は存在しない。

【0042】

状態 206 および 208 では、入力ストリームの値 5 および値 7 の要素がソートバッファの適切な位置に挿入される。状態 208 の終了時のソートバッファの最小要素と最大要素との差は 7 であり、これは依然としてウィンドウ幅 15 未満である。そのため、境界条件に違反しておらず、データ要素をイジェクトする必要はない。それに応じて、依然として出力データ要素は存在しない。

10

【0043】

状態 210 では、値 15 の要素がソートバッファに挿入されると、バッファの最小要素と最大要素との差はここでは 15 であり、これはウィンドウ幅 15 に等しい。境界条件に違反している。それに応じて、ソートバッファ 112 の最小要素である値 0 の要素が出力ストリームの第 1 の要素としてイジェクトされる。それに加えて、ソートバッファの最小要素 S はここでは 3 である。

【0044】

状態 212 では、値 8 の要素がソートバッファの値 15 の要素と値 7 の要素との間に挿入される。バッファの最小値要素と最大値要素との差は 12 であり、これはウィンドウ幅 15 未満である。それに応じて、境界条件に違反しておらず、データ要素はイジェクトされない。

20

【0045】

状態 214 では、値 21 の要素がソートバッファに挿入されると、バッファの最小値要素と最大値要素との差は $21 - 3 = 18$ であり、これはウィンドウ幅 15 を超える。ウィンドウ幅の範囲外の値を有するすべてのデータ要素がイジェクトされる。すなわち、示されるように、値 3、5 および 7 の要素がイジェクトされる。ソートバッファの新しい最小データ要素 S はここでは値 8 の要素である。

【0046】

状態 216 では、値 11 の要素がソートバッファに挿入されると、バッファの最小値要素と最大値要素との差は $21 - 8 = 13$ であり、これはウィンドウ幅 15 未満である。それに応じて、境界条件に違反しておらず、データ要素はイジェクトされない。

30

【0047】

状態 218 では、値 7 の要素がソートバッファに挿入されると、ソートバッファ 112 の最小要素は値 7 のものである。そのため、バッファの最小値要素と最大値要素との差は $21 - 7 = 14$ であり、これはウィンドウ幅 15 未満である。それに応じて、この場合もやはり、境界条件に違反しておらず、データ要素はイジェクトされない。

【0048】

ウィンドウソートプロセスの実装形態は、コンピュータグラフで使用することができる。ここで、図 3 を参照すると、コンピュータグラフ 302 を管理するためのシステム 300 の部品の相互関係が示されている。グラフィック開発環境 (GDE) 304 は、実行可能なコンピュータグラフ 302 を指定して、コンピュータグラフ 302 のウィンドウソートコンポーネント 308 を含む 1 つまたは複数のグラフコンポーネント 306a ~ c のためのパラメータを定義するためのユーザインターフェースを提供する。GDE 304 については、例えば、その内容はその全体が参照により本明細書に組み込まれる、米国特許第 7,164,422 号「PARAMETERIZED GRAPHS WITH CONDITIONAL COMPONENTS」で説明され得る。

40

【0049】

コンピュータグラフ 302 を使用して、複雑なコンピューテーションを表すことができる。そのようなグラフベースのコンピューテーションを実施するシステムについては、その内容はその全体が参照により本明細書に組み込まれる、米国特許第 5,966,072

50

号「EXECUTING COMPUTATIONS EXPRESSED AS GRAPHS」で説明されている。いくつかの実装形態では、コンピュータグラフは、頂点間の有向リンク（作業要素のフローを表す）によって接続された頂点（コンポーネントまたはデータセットを表す）を含む。いくつかの事例では、コンピュータグラフ 302 は、入力データ記録のフローを受信し、データ記録を処理して、コンピュテーションがシャットダウンされるまで無制限にコンポーネント 306 から結果を提供するよう構成される。いくつかの事例では、コンピュータグラフ 302 は、入力データ記録のバッチを受信し、データ記録のバッチを処理して、そのバッチに対する結果を提供し、次いで、バッチが処理された後でシャットダウンするかまたはアイドル状態に戻るよう構成される。

【0050】

10

コンポーネント 306a～c および 308 は、コンポーネント 306a～c および 308 間のデータフローを表す有向弧または線によってともにリンクされる。コンポーネント 306a～c および 308 は、1つまたは複数のデータソースコンポーネント（例えば、コンポーネント 306a）と、データ処理コンポーネント（例えば、コンポーネント 306b）と、データシンクコンポーネント（例えば、306d）とを含み得る。データソースコンポーネント 306a は、例えばデータ記録の形でデータを生成する1つまたは複数のコンピュータシステム上で実施されるソースアプリケーション（図1のソースアプリケーション 102 など）を表し得る。データ処理コンポーネント 306b は、データソースコンポーネント 306a によって生成されたデータ記録を処理する。データシンクコンポーネント 306d は、データ処理コンポーネント 306b の結果を使用する。データ処理コンポーネント 306b からの結果は、大部分がソートされたデータ要素のストリームの形であり得る。そのため、ウィンドウソートコンポーネント 308 を導入して、データ処理コンポーネント 306b からの結果のストリームをソートし、実質的にソートされたデータストリームをシンクコンポーネント 306c に提供することができる。

【0051】

20

例示的なシナリオでは、データソースコンポーネント 306a は、2つの同一のソートされたデータストリーム 310a および 310b を生成し、一方のストリーム（例えば、ストリーム 310b）をコンポーネント 306b に送信することができる。コンポーネント 306b は、ストリーム 310b を処理して、データ記録が格納された記憶場所のアドレスを有するアドレスストリーム 310c を生成する。記憶場所は、連続したもので、次のように、すなわち、 $a_0, a_0 + 1, a_0 + 2, \dots, a_0 + n$ (a_0 は初期の記憶場所のアドレス) と連続的に配列されるものとして知られている。そのため、アドレスストリーム内の個別のアドレスも、連続的に順序付けられるものと予想される。しかし、時折、コンポーネント 306b における処理の遅延により、1つまたは複数のアドレスは、アドレスストリームに遅れて出力され得る。その結果、個別のアドレスはもはやソートされたストリーム 310a 内のデータ記録に適正に対応しない。この状況を改善するため、ソートコンポーネント 308 を使用して、アドレスを元の順序にソートし直すことができる（これは、新しいアドレスストリーム 310d として生成される）。

30

【0052】

40

典型的なアプリケーションでは、アドレスストリーム内の個別のアドレスの位置は、それらの元の位置からそれほど遠くはない。この特性をソートコンポーネントで使用して、アドレスを元の位置に復元し直すことができる。アドレスストリームは、各入力アドレスおよび / または各出力アドレスのイジェクションの後で変更する、動的に再調整されるカットオフアドレスの形で境界条件を有するソートバッファで受信される。最初は、カットオフアドレスは、アドレスストリーム内の最も初期のアドレスに初期化される。その後、入力アドレスがカットオフアドレスより 1 だけ大きければ、カットオフアドレスは入力アドレスに変化する。アドレスがソートコンポーネントに到達すると、カットオフアドレス以下を照合するアドレスは、ソートされた形で直接出力される。他方では、カットオフアドレス値超を照合するアドレスは、ソートバッファに格納される。

【0053】

50

ここで、図4を参照すると、ほぼまたは大部分がソートされたデータ要素のストリームをソートするための例示的なウィンドウソートプロセス400が示されている。一実装形態では、以下に概説される工程は、コンピュータシステム上で実行されるソフトウェアアプリケーションによって行うことができる。例えば、プロセス400は、ソートシステム100(図1に示されるような)によって実現することができる。入力データ要素をバッファリングするためのソートバッファを初期化する(工程402)。初期化は、入力要素を格納するための、記憶域、例えば、複数の連続メモリアドレスを指定する工程を含み得る。初期化の一部として、ソートバッファに対して境界条件を指定することができる。例えば、ソートバッファのウィンドウ幅に相当する値を保持するための変数を定義することができる。ウィンドウ幅は、ソートバッファの任意の2つの格納データ要素値間の最大許容差を表し得る。いくつかの例では、境界条件は、動的に調整されるカットオフ値として表され得る。

【0054】

ソートバッファに対する境界条件が定義されると、ソートバッファがデータ要素の入力ストリームを受信する準備が整う(工程404)。1つまたは複数の技術および方法論を使用して、入力データ要素を受信することができる。一実装形態では、入力要素が受信され、ソートバッファに挿入される。最初は、ソートバッファが空の際は、第1の要素は、事前に定義された特性を有するものとしてデフォルトで選択される(例えば、第1の要素は、最優先のイジェクションを有するものとして選択され得る)。例えば、第1の要素は、ソートバッファの最小要素と見なすことができる。後続のより大きな要素が受信されると、それは、除去またはイジェクション優先度(すなわち、昇順に第1の要素に隣接して)に基づいて適切な位置に挿入される。一実装形態では、要素がバッファに挿入されるたびに、最優先要素(または最小要素)が再計算される。

【0055】

境界条件を適用して、要素が境界条件に違反しているかどうか決定する(工程406)。1つまたは複数の技術は、境界条件適用のために実施することができる。例えば、ソートバッファのウィンドウ幅を超えているかどうかについての決定が下される。境界条件に違反していない(例えば、ソートバッファのウィンドウ幅を超えていない)場合、処理のため、入力ストリームの次のデータ要素が受信される。

【0056】

境界条件に違反している(例えば、ウィンドウ幅を超えている)場合、イジェクション優先度順に従って、1つまたは複数の要素をイジェクトする(工程408)。例えば、イジェクトされる要素は、少なくとも最小要素と、ソートバッファのウィンドウ幅を超えることになる1つまたは複数の他の要素とを含む。さらに、ウィンドウ幅を超えることにはならない新しい最小要素が特定される。そのため、1つまたは複数の技術は、境界条件に基づいて、既定の順序に従ってソートされたデータ要素の出力ストリームとして、バッファから1つまたは複数のデータ要素を生成するために実施することができる。

【0057】

入力要素が存在する限り、プロセス400は、各入力要素に対して繰り返される。

【0058】

一実装形態では、例えば、物理的なメモリサイズを超えることに起因して、ソートバッファが最大容量に到達し得る。そのような状況では、スピルオーバーメカニズムを実装することができる。スピルオーバーメカニズムは、ソートバッファから二次記憶格納域(図示せず)に1つまたは複数のデータ要素を移動する工程を伴う。「主要」要素またはアイテムをソートバッファに挿入して、二次記憶格納域のデータ要素を参照することができる。スピルオーバーメカニズムについては、その内容はその全体が参照により本明細書に組み込まれる、同時係属の米国特許出願第13/031,034号でさらに詳細に説明されている。

【0059】

いくつかの実装形態では、境界条件は、ソートバッファの事前に定義された最大許容サ

10

20

30

40

50

イズ（例えば、バッファに許容される要素の最大数）によって表すことができる。また、一実装形態では、境界条件は、ウィンドウ幅限度と最大サイズ限度の両方によって表すことができる。この実装形態では、ソートモジュールは、バッファのウィンドウ幅または最大サイズのいずれかを超えることになる要素を受信する工程に応じて、要素のイジェクションを引き起こすことができる。

【0060】

いくつかの例では、処理期間の終了時にソートバッファに残存するデータ要素は、処理終了時にソートされた形で出力することができる。いくつかの例では、残存要素は、残りのデータ要素と別々に異なる方法で処理することができる。いくつかの実装形態では、アプリケーションは、例えば、ソートバッファが既定の要素数を超えて格納するかまたは要素を1つでも格納する際のアプリケーションの演算論理のエラーまたは誤動作をユーザに警告することができる。

10

【0061】

一例では、ユーザは、ソートバッファに対する境界条件に違反したデータ要素の存在について警告を受けることができる。例えば、データ記録のリスト（例えば、新しい順の）をユーザに提供する実装形態では、ユーザは、1つまたは複数の順序に狂いが生じているデータ記録の存在について警告を受けることができる。順序に狂いが生じているデータ記録は、記録に対応するデータ要素が境界条件を満たすかどうか、例えば、ユーザによって指定されたウィンドウ幅内にデータ要素があるかどうかを決定する工程に基づいて特定される。例えば、1つまたは複数の順序に狂いが生じているデータ記録をハイライトして、ユーザの注目を集めることができる。いくつかの例では、1つまたは複数の順序に狂いが生じているデータ記録は、別々の処理のために、データ要素のストリームから抽出することができる。

20

【0062】

本明細書に記載される技術は、コンピュータ上で実行するためのソフトウェアを使用して実施することができる。例えば、ソフトウェアは、1つまたは複数のプログラムされたまたはプログラム可能なコンピュータシステム（それは、分散型、クライアント/サーバまたはグリッドなどのさまざまなアーキテクチャのものであってもよい）上で実行する1つまたは複数のコンピュータプログラムの手順を形成し、コンピュータシステムはそれぞれ、少なくとも1つのプロセッサと、少なくとも1つのデータ格納システム（揮発性および不揮発性メモリならびに/または格納要素を含む）と、少なくとも1つの入力デバイスまたはポートと、少なくとも1つの出力デバイスまたはポートとを含む。ソフトウェアは、例えば、コンピューテーショングラフの設計および構成に関連する他のサービスを提供する大規模なプログラムの1つまたは複数のモジュールを形成することができる。グラフのノードおよび要素は、コンピュータ可読媒体に格納されたデータ構造、または、データリポジトリに格納されたデータモデルに適合する他の組織化されたデータとして実装することができる。

30

【0063】

ソフトウェアは、汎用または特殊用途のプログラム可能なコンピュータによる読み取りが可能なCD-ROMなどの記憶媒体上で提供されても、ネットワークの通信媒体上で、ソフトウェアが実行されるコンピュータに送達されてもよい（伝播信号で符号化される）。機能のすべては、特殊用途のコンピュータ上で、または、コプロセッサなどの特殊用途のハードウェアを使用して実施することができる。ソフトウェアは、ソフトウェアによって指定されるコンピューテーションの異なる部分は異なるコンピュータによって実行される分散様式で実施することができる。それぞれのそのようなコンピュータプログラムは、好みしくは、汎用もしくは特殊用途のプログラム可能なコンピュータによる読み取りが可能な記憶媒体もしくはデバイス（例えば、ソリッドステートメモリもしくは媒体、または、磁気もしくは光媒体）上に格納されるか、または、記憶媒体もしくはデバイスにダウンロードされるが、それは、コンピュータシステムで記憶媒体もしくはデバイスを読み取り、本明細書に記載される手順を実行する際にコンピュータを構成および動作するためであ

40

50

る。また、本発明のシステムは、コンピュータ可読記憶媒体として実装され、コンピュータプログラムを用いて構成されるものと考慮され得、そのように構成された記憶媒体は、特定の事前に定義された様式でコンピュータシステムを動作させ、本明細書に記載される機能を実行する。

【 0 0 6 4 】

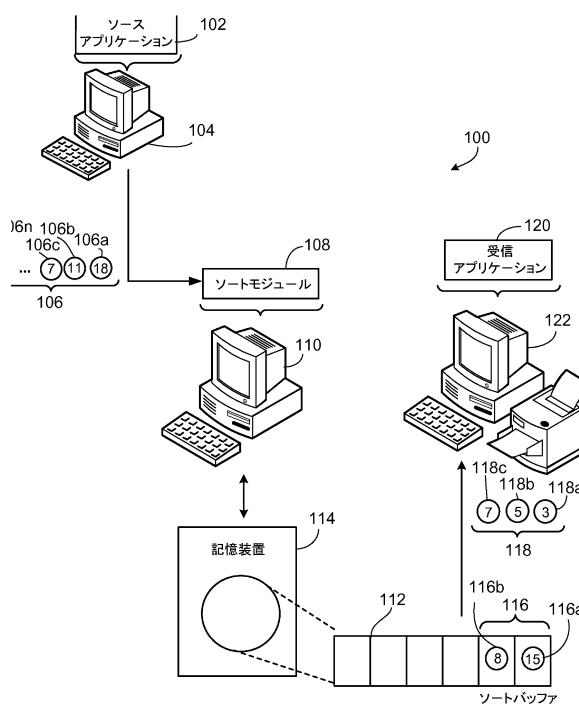
本発明の多くの実施形態について説明してきた。それにもかかわらず、本発明の精神および範囲から逸脱することなく、さまざまな変更を行うことができるところが理解されよう。例えば、上記で説明された工程のいくつかは、順序に依存しないものであり得、従って、説明されている順序とは異なる順序で実行することができる。

【 0 0 6 5 】

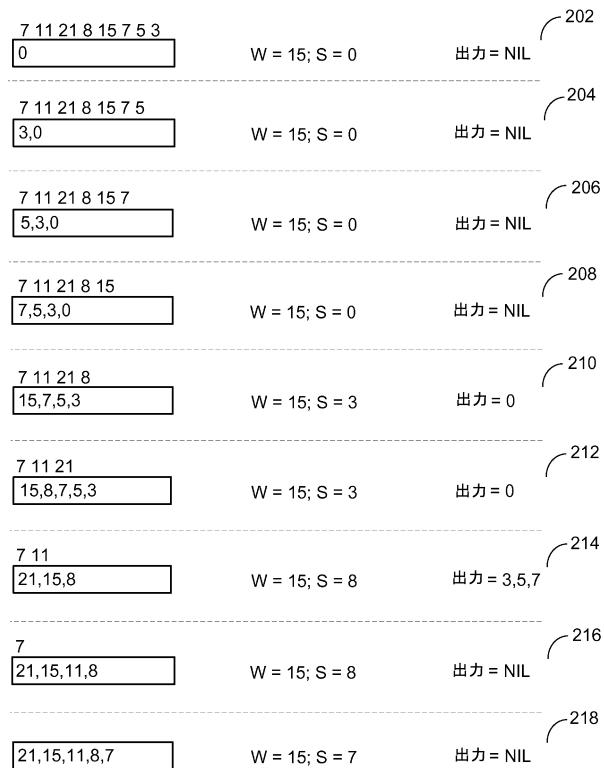
前述の説明は、例示を意図するものであり、本発明の範囲を限定するものではなく、本発明の範囲は、添付の特許請求の範囲によって定義されることを理解されたい。例えば、上記で説明された多くの機能工程は、処理全体に実質的に影響を与えることなく、異なる順序で実行することができる。他の実施形態は、以下の特許請求の範囲内にある。

10

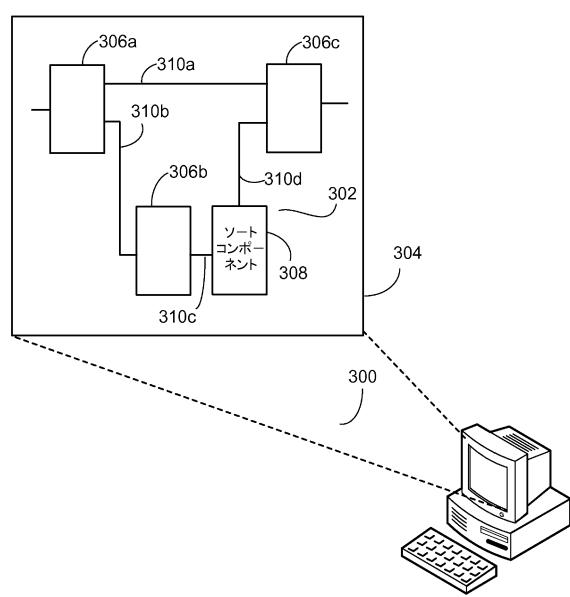
【 図 1 】



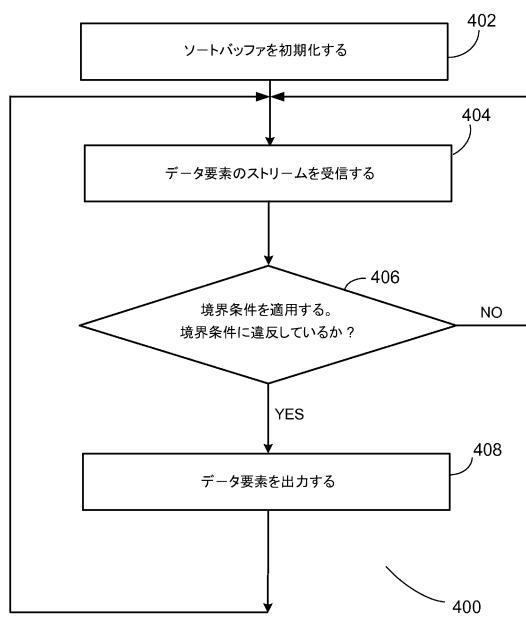
【 図 2 】



【図3】



【図4】



フロントページの続き

(72)発明者 ファインマン, カール リチャード

アメリカ合衆国, マサチューセッツ州 02468, ワパン, モスフィールド ロード 21

審査官 佐賀野 秀一

(56)参考文献 特開2011-009927 (JP, A)

国際公開第2010/110155 (WO, A1)

特開2008-022507 (JP, A)

特開昭64-046830 (JP, A)

特開2007-049540 (JP, A)

特表2008-544423 (JP, A)

特開2006-067158 (JP, A)

特表2007-532053 (JP, A)

特開2007-179183 (JP, A)

米国特許出願公開第2012/0110099 (US, A1)

米国特許出願公開第2008/0052408 (US, A1)

欧州特許出願公開第01879393 (EP, A1)

中国特許出願公開第101106697 (CN, A)

台湾特許出願公開第200826679 (TW, A)

米国特許出願公開第2007/0053607 (US, A1)

米国特許出願公開第2006/0294150 (US, A1)

国際公開第2007/002647 (WO, A1)

欧州特許出願公開第01899873 (EP, A1)

中国特許出願公開第101208695 (CN, A)

韓国公開特許第10-2008-0040641 (KR, A)

カナダ国特許出願公開第02610989 (CA, A1)

豪国特許出願公開第2006261749 (AU, A1)

(58)調査した分野(Int.Cl. , DB名)

G 06 F 7 / 24