(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 109684602 A
(43)申请公布日 2019.04.26

(21)申请号 201811647480.5

(22)申请日 2018.12.29

(71)申请人 上海商汤智能科技有限公司
地址 200233 上海市徐汇区桂平路391号3号楼1605A室

(72)发明人 李秀红 梁云 颜深根 贾连成 李英晗

(74)专利代理机构 广州三环专利商标代理有限公司 44202
代理人 郝传鑫 熊永强

(51)Int.Cl.
G06F 17/16(2006.01)
G06F 9/50(2006.01)

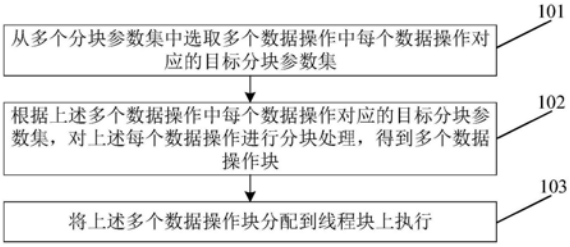
权利要求书2页 说明书19页 附图6页

(54)发明名称

一种批处理方法和装置及计算机可读存储介质

(57)摘要

本申请公开了一种批处理方法及装置。该方法包括:从多个分块参数集中选取多个数据操作中每个数据操作对应的目标分块参数集,其中,所述多个数据操作对应的目标分块参数集对应相同的线程数;根据所述多个数据操作中每个数据操作对应的目标分块参数集,对所述每个数据操作进行分块处理,得到多个数据操作块;将所述多个数据操作块分配到线程块上执行。还公开了相应的装置。在批处理多个数据操作时,通过对多个数据操作进行分块,可提高GPU的资源利用率。



CN 109684602 A

1.一种批处理方法,其特征在于,包括:

从多个分块参数集中选取多个数据操作中每个数据操作对应的目标分块参数集,其中,所述多个数据操作对应的目标分块参数集对应相同的线程数;

根据所述多个数据操作中每个数据操作对应的目标分块参数集,对所述每个数据操作进行分块处理,得到多个数据操作块;

将所述多个数据操作块分配到线程块上执行。

2.根据权利要求1所述的方法,其特征在于,所述从多个分块参数集中选取多个数据操作中每个数据操作对应的目标分块参数集,包括:

从所述多个分块参数集中确定所述多个数据操作中每个数据操作的至少一个可用分块参数集;

从多个数据操作中每个数据操作对应的至少一个可用分块参数集中选取所述每个数据操作的当前分块参数集;

确定所述多个数据操作的当前分块参数集对应的线程数的第一总和;

在所述线程数的第一总和大于第一阈值的情况下,更新所述多个数据操作中的至少一个数据操作的当前分块参数集,直到满足更新截止条件,其中,所述更新截止条件包括所述多个数据操作的当前分块参数集对应的线程数的第一总和小于或等于所述第一阈值,并将满足所述更新截止条件的所述多个数据操作的当前分块参数集所对应的更新前的分块参数集作为所述多个数据操作的目标分块参数集。

3.根据权利要求2所述的方法,其特征在于,所述从多个数据操作中每个数据操作对应的至少一个可用分块参数集中选取所述每个数据操作的当前分块参数集,包括:

从第一数据操作的至少一个可用分块参数集中选取对应的线程数最大且数据操作块尺寸最小的分块参数集作为所述第一数据操作的当前分块参数集,其中,所述多个数据操作包括所述第一数据操作。

4.根据权利要求2或3所述的方法,其特征在于,所述更新所述多个数据操作中的至少一个数据操作的当前分块参数集,包括:

将所述多个数据操作中的第二数据操作的当前分块参数集由第一分块参数集更新为第二分块参数集,其中,所述第二数据操作的至少一个可用分块参数集包括所述第一分块参数集和所述第二分块参数集,所述第二分块参数集对应的数据操作块尺寸大于所述第一分块参数集的数据操作块尺寸或者所述第二分块参数集对应的线程数小于所述第一分块参数集对应的线程数。

5.根据权利要求2至4中任一项所述的方法,其特征在于,在所述从多个数据操作中每个数据操作对应的至少一个可用分块参数集中选取所述每个数据操作的当前分块参数集之前,还包括:

按照分块参数集对应的线程数由大到小且块尺寸由小到大的顺序,对所述多个数据操作中每个数据操作对应的至少一个可用分块参数集进行排序,得到所述每个数据操作的可用分块参数集队列;

所述从多个数据操作中每个数据操作对应的至少一个可用分块参数集中选取所述每个数据操作的当前分块参数集,包括:

从多个数据操作中每个数据操作的可用分块参数集队列中的首个分块参数集作为所

述每个数据操作的当前分块参数集；

所述更新所述多个数据操作中的至少一个数据操作的当前分块参数集，包括：

在所述第一总和大于第一阈值的情况下，将所述至少一个数据操作中每个数据操作的当前分块参数集从所述每个数据操作的可用分块参数集队列中删除，并将所述每个数据操作的当前分块参数集更新为所述每个数据操作的更新后的可用分块参数集队列中的首个分块参数集。

6.根据权利要求2至5中任一项所述的方法，其特征在于，所述更新所述多个数据操作中的至少一个数据操作的当前分块参数集，包括：

响应于所述至少一个数据操作中的每个数据操作的至少一个可用分块参数集包括线程数与所述每个数据操作的当前分块参数集的线程数相同且块尺寸大于所述每个数据操作的当前分块参数集的块尺寸的第一可更新分块参数集，将所述至少一个数据操作中每个数据操作的当前分块参数集更新为所述第一可更新分块参数集；和/或

响应于所述至少一个数据操作中存在第三数据操作，其中，所述第三数据操作的至少一个可用分块参数集不包括所述第一可更新分块参数集，将所述至少一个数据操作中每个数据操作的当前分块参数集更新为线程数小于所述每个数据操作的当前分块参数集的线程数的第二可更新分块参数集。

7.根据权利要求1～6任一项所述的方法，其特征在于，所述将所述多个数据操作块分配到线程块上执行，包括：

为所述多个数据操作块中的至少一个第一数据操作块分配线程块；

确定为当前已分配线程块的所述至少一个第一数据操作块所分配的线程块中包含的总线程数与所述多个数据操作块中当前还未分配线程块的多个第二数据操作块的数量的第二总和；

基于所述第二总和与第二阈值的大小关系，为所述当前还未分配线程块的多个第二数据操作块分配线程块。

8.一种批处理装置，其特征在于，包括：

选取单元，用于从多个分块参数集中选取多个数据操作中每个数据操作对应的目标分块参数集，其中，所述多个数据操作对应的目标分块参数集对应相同的线程数；

处理单元，用于根据所述多个数据操作中每个数据操作对应的目标分块参数集，对所述每个数据操作进行分块处理，得到多个数据操作块；

分配单元，用于将所述多个数据操作块分配到线程块上执行。

9.一种批处理装置，其特征在于，包括：处理器和存储器，所述处理器和所述存储耦合器；其中，所述存储器存储有程序指令，所述程序指令被所述处理器执行时，使所述处理器执行如权利要求1至7任意一项所述的方法。

10.一种计算机可读存储介质，其特征在于，所述计算机可读存储介质中存储有计算机程序，所述计算机程序包括程序指令，所述程序指令当被批处理装置的处理器执行时，使所述处理器执行如权利要求1至7任意一项所述的方法。

一种批处理方法和装置及计算机可读存储介质

技术领域

[0001] 本申请涉及计算机技术领域，尤其涉及一种批处理方法和装置及计算机可读存储介质。

背景技术

[0002] 随着计算机硬件的不断提升，计算机性能越来越强大，高性能计算被广泛应用于深度学习、图像处理和数据信号处理等任务中，其中，高性能计算中包括的矩阵算法等计算密集型任务对于应用的整体性能有着重要影响。然而，在实际应用中，很多数据操作，例如矩阵乘法等密集型操作，其数据量较小，无法充分发挥图像处理单元(GPU)的计算能力。如何提高GPU在矩阵乘法等密集型计算任务中的性能是本领域的研究热点。

发明内容

[0003] 本申请提供一种批处理技术，以实现对数据操作的批处理。

[0004] 第一方面，提供了一种批处理方法，包括：从多个分块参数集中选取多个数据操作中每个数据操作对应的目标分块参数集，其中，所述多个数据操作对应的目标分块参数集对应相同的线程数；根据所述多个数据操作中每个数据操作对应的目标分块参数集，对所述每个数据操作进行分块处理，得到多个数据操作块；将所述多个数据操作块分配到线程块上执行。

[0005] 在一种可能实现的方式中，所述从多个分块参数集中选取多个数据操作中每个数据操作对应的目标分块参数集，包括：从所述多个分块参数集中确定所述多个数据操作中每个数据操作的至少一个可用分块参数集；从多个数据操作中每个数据操作对应的至少一个可用分块参数集中选取所述每个数据操作的当前分块参数集；确定所述多个数据操作的当前分块参数集对应的线程数的第一总和；在所述线程数的第一总和大于第一阈值的情况下，更新所述多个数据操作中的至少一个数据操作的当前分块参数集，直到满足更新截止条件，其中，所述更新截止条件包括所述多个数据操作的当前分块参数集对应的线程数的第一总和小于或等于所述第一阈值，并将满足所述更新截止条件的所述多个数据操作的当前分块参数集所对应的更新前的分块参数集作为所述多个数据操作的目标分块参数集。

[0006] 在另一种可能实现的方式中，所述从多个数据操作中每个数据操作对应的至少一个可用分块参数集中选取所述每个数据操作的当前分块参数集，包括：从第一数据操作的至少一个可用分块参数集中选取对应的线程数最大且数据操作块尺寸最小的分块参数集作为所述第一数据操作的当前分块参数集，其中，所述多个数据操作包括所述第一数据操作。

[0007] 在又一种可能实现的方式中，所述更新所述多个数据操作中的至少一个数据操作的当前分块参数集，包括：将所述多个数据操作中的第二数据操作的当前分块参数集由第一分块参数集更新为第二分块参数集，其中，所述第二数据操作的至少一个可用分块参数集包括所述第一分块参数集和所述第二分块参数集，所述第二分块参数集对应的数据操作

块尺寸大于所述第一分块参数集的数据操作块尺寸或者所述第二分块参数集对应的线程数小于所述第一分块参数集对应的线程数。

[0008] 在又一种可能实现的方式中，在所述从多个数据操作中每个数据操作对应的至少一个可用分块参数集中选取所述每个数据操作的当前分块参数集之前，还包括：按照分块参数集对应的线程数由大到小且块尺寸由小到大的顺序，对所述多个数据操作中每个数据操作对应的至少一个可用分块参数集进行排序，得到所述每个数据操作的可用分块参数集队列；所述从多个数据操作中每个数据操作对应的至少一个可用分块参数集中选取所述每个数据操作的当前分块参数集，包括：从多个数据操作中每个数据操作的可用分块参数集队列中的首个分块参数集作为所述每个数据操作的当前分块参数集；所述更新所述多个数据操作中的至少一个数据操作的当前分块参数集，包括：在所述第一总和大于第一阈值的情况下，将所述至少一个数据操作中每个数据操作的当前分块参数集从所述每个数据操作的可用分块参数集队列中删除，并将所述每个数据操作的当前分块参数集更新为所述每个数据操作的更新后的可用分块参数集队列中的首个分块参数集。

[0009] 在又一种可能实现的方式中，所述更新所述多个数据操作中的至少一个数据操作的当前分块参数集，包括：响应于所述至少一个数据操作中的每个数据操作的至少一个可用分块参数集包括线程数与所述每个数据操作的当前分块参数集的线程数相同且块尺寸大于所述每个数据操作的当前分块参数集的块尺寸的第一可更新分块参数集，将所述至少一个数据操作中每个数据操作的当前分块参数集更新为所述第一可更新分块参数集；和/或响应于所述至少一个数据操作中存在第三数据操作，其中，所述第三数据操作的至少一个可用分块参数集不包括所述第一可更新分块参数集，将所述至少一个数据操作中每个数据操作的当前分块参数集更新为线程数小于所述每个数据操作的当前分块参数集的线程数的第二可更新分块参数集。

[0010] 在又一种可能实现的方式中，所述将所述多个数据操作块分配到线程块上执行，包括：为所述多个数据操作块中的至少一个第一数据操作块分配线程块；确定为当前已分配线程块的所述至少一个第一数据操作块所分配的线程块中包含的总线程数与所述多个数据操作块中当前还未分配线程块的多个第二数据操作块的数量的第二总和；基于所述第二总和与第二阈值的大小关系，为所述当前还未分配线程块的多个第二数据操作块分配线程块。

[0011] 在又一种可能实现的方式中，所述基于所述第二总和与第二阈值的大小关系，为所述当前还未分配线程块的多个第二数据操作块分配线程块，包括：在所述第二总和小于或等于第二阈值的情况下，为所述多个第二数据操作块中的不同数据操作块分配不同的线程块。

[0012] 在又一种可能实现的方式中，所述基于所述第二总和与第二阈值的大小关系，为所述当前还未分配线程块的至少一个第二数据操作块分配线程块，包括：在所述第二总和大于所述第二阈值的情况下，为所述多个第二数据操作块中的至少两个数据操作块分配同一个线程块。

[0013] 在又一种可能实现的方式中，所述至少一个第一数据操作块为多个第一数据操作块，所述为所述多个数据操作块中的至少一个第一数据操作块分配线程块，包括：为所述多个第一数据操作块中的至少两个数据操作块分配同一个线程块。

[0014] 在又一种可能实现的方式中，所述为所述多个数据操作块中的至少一个第一数据操作块分配线程块，包括：为所述多个数据操作块中尺寸参数值最大的数据操作块和所述尺寸参数值最小的数据操作块分配同一个线程块。

[0015] 在又一种可能实现的方式中，所述基于所述第二总和与第二阈值的大小关系，为所述当前还未分配线程块的多个第二数据操作块分配线程块，包括：在所述第二总和大于所述第二阈值的情况下，为当前还未分配线程块的多个第二数据操作块中尺寸参数值最大的数据操作块和尺寸参数值最小的数据操作块分配同一个线程块。

[0016] 在又一种可能实现的方式中，所述为所述多个数据操作块中的至少一个第一数据操作块分配线程块，包括：将所述多个数据操作块中相邻的N个第一数据操作块分配同一个线程块，其中，所述N为大于1的整数，所述N个第一数据操作块的尺寸参数之和大于第三阈值且所述N个第一数据块中的前N-1个第一数据操作块的尺寸参数之和小于或等于所述第三阈值。

[0017] 在又一种可能实现的方式中，所述基于所述第二总和与第二阈值的大小关系，为所述当前还未分配线程块的多个第二数据操作块分配线程块，包括：在所述第二总和大于所述第二阈值的情况下，为当前还未分配线程块的多个第二数据操作块中相邻的M个数据操作块分配同一个线程块，以使得所述M个数据操作块的尺寸参数值之和恰好大于所述第三阈值。

[0018] 在又一种可能实现的方式中，所述数据操作块的尺寸参数为所述数据操作块包括的第一矩阵与第二矩阵的矩阵乘法中的第一矩阵的列数。

[0019] 在又一种可能实现的方式中，所述数据操作为矩阵乘法或卷积运算。

[0020] 第二方面，提供了一种批处理装置，包括：选取单元，用于从多个分块参数集中选取多个数据操作中每个数据操作对应的目标分块参数集，其中，所述多个数据操作对应的目标分块参数集对应相同的线程数；处理单元，用于根据所述多个数据操作中每个数据操作对应的目标分块参数集，对所述每个数据操作进行分块处理，得到多个数据操作块；分配单元，用于将所述多个数据操作块分配到线程块上执行。

[0021] 在一种可能实现的方式中，所述选取单元包括：第一确定子单元，用于从所述多个分块参数集中确定所述多个数据操作中每个数据操作的至少一个可用分块参数集；第一选取子单元，用于从多个数据操作中每个数据操作对应的至少一个可用分块参数集中选取所述每个数据操作的当前分块参数集；第一计算子单元，用于确定所述多个数据操作的当前分块参数集对应的线程数的第一总和；更新子单元，用于在所述线程数的第一总和大于第一阈值的情况下，更新所述多个数据操作中的至少一个数据操作的当前分块参数集，直到满足更新截止条件，其中，所述更新截止条件包括所述多个数据操作的当前分块参数集对应的线程数的第一总和小于或等于所述第一阈值，并将满足所述更新截止条件的所述多个数据操作的当前分块参数集所对应的更新前的分块参数集作为所述多个数据操作的目标分块参数集。

[0022] 在另一种可能实现的方式中，所述第一选取子单元具体用于：从第一数据操作的至少一个可用分块参数集中选取对应的线程数最大且数据操作块尺寸最小的分块参数集作为所述第一数据操作的当前分块参数集，其中，所述多个数据操作包括所述第一数据操作。

[0023] 在又一种可能实现的方式中，所述更新子单元具体用于：将所述多个数据操作中的第二数据操作的当前分块参数集由第一分块参数集更新为第二分块参数集，其中，所述第二数据操作的至少一个可用分块参数集包括所述第一分块参数集和所述第二分块参数集，所述第二分块参数集对应的数据操作块尺寸大于所述第一分块参数集的数据操作块尺寸或者所述第二分块参数集对应的线程数小于所述第一分块参数集对应的线程数。

[0024] 在又一种可能实现的方式中，所述第一选取子单元还用于：按照分块参数集对应的线程数由大到小且块尺寸由小到大的顺序，对所述多个数据操作中每个数据操作对应的至少一个可用分块参数集进行排序，得到所述每个数据操作的可用分块参数集队列；以及所述从多个数据操作中每个数据操作对应的至少一个可用分块参数集中选取所述每个数据操作的当前分块参数集；以及从多个数据操作中每个数据操作的可用分块参数集队列中的首个分块参数集作为所述每个数据操作的当前分块参数集；以及所述更新所述多个数据操作中的至少一个数据操作的当前分块参数集，包括：以及在所述第一总和大于第一阈值的情况下，将所述至少一个数据操作中每个数据操作的当前分块参数集从所述每个数据操作的可用分块参数集队列中删除，并将所述每个数据操作的当前分块参数集更新为所述每个数据操作的更新后的可用分块参数集队列中的首个分块参数集。

[0025] 在又一种可能实现的方式中，所述更新子单元具体用于：响应于所述至少一个数据操作中的每个数据操作的至少一个可用分块参数集包括线程数与所述每个数据操作的当前分块参数集的线程数相同且块尺寸大于所述每个数据操作的当前分块参数集的块尺寸的第一可更新分块参数集，将所述至少一个数据操作中每个数据操作的当前分块参数集更新为所述第一可更新分块参数集；和/或响应于所述至少一个数据操作中存在第三数据操作，其中，所述第三数据操作的至少一个可用分块参数集不包括所述第一可更新分块参数集，将所述至少一个数据操作中每个数据操作的当前分块参数集更新为线程数小于所述每个数据操作的当前分块参数集的线程数的第二可更新分块参数集。

[0026] 在又一种可能实现的方式中，所述分配单元包括：第一分配子单元，用于为所述多个数据操作块中的至少一个第一数据操作块分配线程块；第二确定子单元，用于确定为当前已分配线程块的所述至少一个第一数据操作块所分配的线程块中包含的总线程数与所述多个数据操作块中当前还未分配线程块的多个第二数据操作块的数量的第二总和；第二分配子单元，用于基于所述第二总和与第二阈值的大小关系，为所述当前还未分配线程块的多个第二数据操作块分配线程块。

[0027] 在又一种可能实现的方式中，所述第二分配子单元还用于：在所述第二总和小于或等于第二阈值的情况下，为所述多个第二数据操作块中的不同数据操作块分配不同的线程块。

[0028] 在又一种可能实现的方式中，所述第二分配子单元还用于：在所述第二总和大于所述第二阈值的情况下，为所述多个第二数据操作块中的至少两个数据操作块分配同一个线程块。

[0029] 在又一种可能实现的方式中，所述第二确定子单元还用于：为所述多个第一数据操作块中的至少两个数据操作块分配同一个线程块。

[0030] 在又一种可能实现的方式中，所述第一分配子单元还用于：为所述多个数据操作块中尺寸参数值最大的数据操作块和所述尺寸参数值最小的数据操作块分配同一个线程

块。

[0031] 在又一种可能实现的方式中，所述第二分配子单元还用于：在所述第二总和大于所述第二阈值的情况下，为当前还未分配线程块的多个第二数据操作块中尺寸参数值最大的数据操作块和尺寸参数值最小的数据操作块分配同一个线程块。

[0032] 在又一种可能实现的方式中，所述第一分配子单元还用于：将所述多个数据操作块中相邻的N个第一数据操作块分配同一个线程块，其中，所述N为大于1的整数，所述N个第一数据操作块的尺寸参数之和大于第三阈值且所述N个第一数据块中的前N-1个第一数据操作块的尺寸参数之和小于或等于所述第三阈值。

[0033] 在又一种可能实现的方式中，所述第一分配子单元还用于：在所述第二总和大于所述第二阈值的情况下，为当前还未分配线程块的多个第二数据操作块中相邻的M个数据操作块分配同一个线程块，以使得所述M个数据操作块的尺寸参数值之和恰好大于所述第三阈值。

[0034] 在又一种可能实现的方式中，所述数据操作块的尺寸参数为所述数据操作块包括的第一矩阵与第二矩阵的矩阵乘法中的第一矩阵的列数。

[0035] 在又一种可能实现的方式中，所述数据操作为矩阵乘法或卷积运算。

[0036] 第三方面，本申请提供了一种计算机可读存储介质，所述计算机可读存储介质中存储有计算机程序，所述计算机程序包括程序指令，所述程序指令当被批处理装置的处理器执行时，使所述处理器执行第一方面中任意一项所述的方法。

[0037] 第四方面，本申请提供了一种批处理装置，包括：处理器和存储器，所述处理器和所述存储耦合器；其中，所述存储器存储有程序指令，所述程序指令被所述处理器执行时，使所述处理器执行第一方面中任意一项所述的方法。

[0038] 本申请通过为批处理的多个数据操作中的每个数据操作分别选取合适的分块参数集，其中，多个数据操作对应的分块参数集具有相同的线程数，根据选取的分块参数集将多个数据操作划分成多个数据操作块，并将多个数据操作块分配到线程块上执行，有利于提高批处理的并行度，避免对不同大小的数据操作进行批处理的过程中存在较多空闲线程的情况，从而提高系统的资源利用率和整体数据处理性能。

附图说明

[0039] 为了更清楚地说明本申请实施例或背景技术中的技术方案，下面将对本申请实施例或背景技术中所需要使用的附图进行说明。

[0040] 图1为本申请实施例提供的一种批处理方法的流程示意图；

[0041] 图2为本申请实施例提供的矩阵乘法分块示意图；

[0042] 图3为本申请实施例提供的流多处理器执行矩阵乘法示意图；

[0043] 图4为本申请实施例提供的另一种批处理方法的流程示意图；

[0044] 图5为本申请实施例提供的另一种批处理方法的流程示意图；

[0045] 图6为本申请实施例提供的一种选取目标分块参数集的流程示意图；

[0046] 图7为本申请实施例提供的另一种批处理方法的流程示意图；

[0047] 图8为本申请实施例提供的另一种批处理方法的流程示意图；

[0048] 图9为本申请实施例提供的另一种批处理方法的流程示意图；

[0049] 图10为本申请实施例提供的一种随机森林算法示意图；

[0050] 图11为本申请实施例提供的另一种批处理数据的装置的硬件结构示意图；

[0051] 图12为本申请实施例提供的另一种批处理数据的装置的硬件结构示意图。

具体实施方式

[0052] 为了使本技术领域的人员更好地理解本申请方案，下面将结合本申请实施例中的附图，对本申请实施例中的技术方案进行清楚、完整地描述，显然，所描述的实施例仅仅是本申请一部分实施例，而不是全部的实施例。基于本申请中的实施例，本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例，都属于本申请保护的范围。

[0053] 本申请的说明书和权利要求书及上述附图中的术语“第一”、“第二”等是用于区别不同对象，而不是用于描述特定顺序。此外，术语“包括”和“具有”以及它们任何变形，意图在于覆盖不排他的包含。例如包含了一系列步骤或单元的过程、方法、系统、产品或设备没有限定于已列出的步骤或单元，而是可选地还包括没有列出的步骤或单元，或可选地还包括对于这些过程、方法、产品或设备固有的其他步骤或单元。

[0054] 在本文中提及“实施例”意味着，结合实施例描述的特定特征、结构或特性可以包含在本申请的至少一个实施例中。在说明书中的各个位置出现该短语并不一定均是指相同的实施例，也不是与其它实施例互斥的独立的或备选的实施例。本领域技术人员显式地和隐式地理解的是，本文所描述的实施例可以与其它实施例相结合。

[0055] 为了更清楚地说明本申请实施例或背景技术中的技术方案，下面将对本申请实施例或背景技术中所需要使用的附图进行说明。

[0056] 下面结合本申请实施例中的附图对本申请实施例进行描述。

[0057] 请参阅图1，图1是本申请实施例提供的一种批处理方法的流程示意图。

[0058] 101、从多个分块参数集中选取多个数据操作中每个数据操作对应的目标分块参数集。

[0059] 在本申请实施例中，数据操作包括：矩阵乘法，卷积操作等，在一些实施例中，数据操作对应的数据量较小，可选地，数据操作为计算密集型操作。

[0060] 由于直接执行多个数据操作的运算量较大，处理效率较低，因此，在对数据操作进行处理之前，需要对数据操作进行分块。在一个可能实现的例子中，数据操作可以为矩阵乘法，通过对矩阵乘法进行分块，可使高阶数据的运算转化为低阶数据的运算，将矩阵乘法中的乘法运算转化为加法运算，从而能大大减小矩阵乘法的运算量。

[0061] 在一个可能实现的例子中，给定一个矩阵乘法C＝α×A×B+β×c，其中，A，B均为矩阵，A×B为矩阵乘法，c为更新前的矩阵乘法，C为更新后的矩阵乘法，α和β均为自然数，其分块参数集可以表示为(BY，BX，BK，T)，其中，图形处理器(graphics processing unit，GPU)首先将矩阵乘法A×B进行分块，如图2所示，其中，A、B分别是两个矩阵，M是A的行，N是B的列，A的列和B的行都是K，BY和BX分别为将矩阵乘法A×B分块处理后得到的矩阵块的行和列。对矩阵A进行分块就是将A中的一整行数据(BY*K)和B(BX*K)中的一整列数据分配给矩阵块，这样，对大小为BY*BX的矩阵块来说，它所包含的数据量就由BY和BX决定，也就是说，矩阵块的大小BY*BX直接决定处理该矩阵块所需的线程数量，同时也决定了对矩阵乘法A×B进行分块后得到的矩阵块的数量，即(M*N)/(BY*BX)个，若通过一个线程块来处理一个矩

阵块，则所需的线程块的数量也是(M*N)/(BY*BX)，而每个线程块中的线程数量是确定的，这样，线程块中的线程数量和所需线程块的数量就决定了处理一个矩阵乘法所需的线程数量。此外，由于读取矩阵A中的一整行数据(BY*K)和矩阵B中的一整列数据(BX*K)需要占用较大的存储资源。为了充分利用GPU片上有限的存储资源，如图3所示，在从A、B中读取数据时，需要将A中的一整行数据和B中的一整列数据进行切割，每次只读取矩阵A中大小为BY*BK的一段数据，和矩阵B中大小为BK*BX的一段数据，并将这两段数据组成的矩阵块存储至共享内存，计算单元再通过寄存器从共享内存中读取该矩阵块并完成对该矩阵块的计算。设一个线程块中的线程数目T，则可将由BY、BX、BK、T组成的四元组表示为一个分块参数集。为提高对系统计算资源的利用率，例如对GPU资源的利用率，可以从预先设定的多个分块参数集中为每个数据操作选取合适的分块参数集，其中，对于不同大小的数据操作，选择的目标分块参数集可以相同或不同，与为批处理的多个数据操作统一选取分块参数集(即多个数据操作对应相同的分块参数集)相比，有利于提高每个数据操作的执行效率和资源利用率。可选地，可以为批处理的多个数据操作选择相同的线程数，即多个数据操作的目标分块参数集对应的线程数相同，以使得在批处理多个数据操作的过程中不会出现分配资源的空闲(例如GPU中没有线程处于空闲状态)，从而进一步提高系统资源的利用率。

[0062] 102、根据上述多个数据操作中每个数据操作对应的目标分块参数集，对上述每个数据操作进行分块处理，得到多个数据操作块。

[0063] 目标分块参数集中包含有分块后得到的数据操作块的大小，在数据操作的大小和分块后得到的数据操作块的大小确定的情况下，可将数据操作分成多个数据操作块。在一个可能实现的例子中，数据操作的大小为128*128，目标分块参数集中的BY＝BX＝16，则依据目标分块参数集对数据操作进行分块将得到8个数据操作块。

[0064] 103、将上述多个数据操作块分配到线程块上执行。

[0065] 在对数据操作进行分块后，对数据操作的执行就转化成了对数据操作块的处理，通过线程块来处理分块后得到的多个数据操作块。需要理解的是，一个线程块可能只执行一个数据操作块，也可能同时执行多个数据操作块。

[0066] 本申请实施例中，通过从多个分块参数集中选取合适的目标分块参数集，可实现对数据操作的分块，并提高GPU的线程利用率。

[0067] 请参阅图4，图4是本申请实施例提供的批处理方法中步骤101的一种可能的实现方式的流程示意图。

[0068] 401、从上述多个分块参数集中确定上述多个数据操作中每个数据操作的至少一个可用分块参数集。

[0069] 分块参数集包括分块后得到的数据操作块的大小，因此，通过分块参数集对数据操作进行分块，并得到预设大小的矩阵块。显然，当分块后得到数据操作块的大小大于数据操作的大小时，对该数据操作来说，该分块参数集是不可用的。在一个可选的示例中，以图2中的矩阵乘法A×B为例，当分块参数集中的BY*BX大于M*N或BK大于K时，该矩阵乘法不能通过该分块参数集进行分块。也就是说，若分块参数集中的BY*BX或BK大于数据操作的尺寸参数的大小时，对于该数据操作来说，该分块参数集为不可用分块参数集。因此，将分块参数集中的BY、BX、BK小于或等于数据操作的相应的尺寸参数的分块参数集作为该数据操作集的可用分块参数集。

[0070] 402、从多个数据操作中每个数据操作对应的至少一个可用分块参数集中选取上述每个数据操作的当前分块参数集。

[0071] 对于不同大小的数据操作，往往需要不同的分块参数集，这样就会使得处理不同大小的数据操作分块得到的数据操作块的说所需的线程数不一样，而GPU中的线程块的的线程数都是相同的，这样，就会导致在批处理不同大小的数据操作时，会有部分线程块中的线程处于空闲状态。如：第一个数据操作选取的分块参数集为(BY_1,BX_1,BK_1,T_1)，第二个数据操作选取的分块参数集为(BY_2,BX_2,BK_2,T_2)，其中，T_1>T_2，GPU在同时对这两个数据操作进行处理时，为了满足第一个数据操作的分块参数集的线程数，会选择线程数为T1的线程块来进行批处理，这样，在处理第二个数据操作时，线程块中必然会有(T_1-T_2)个线程处于空闲状态，也就导致GPU的线程利用不充分，降低GPU的线程级并行性。通过将执行所有数据操作块的线程数取为一样，可避免在批处理不同大小的数据操作时出现线程空闲的情况。

[0072] 此外，针对不同大小的数据操作，优先选择线程数大且分块后得到的数据操作块小的分块参数集，保证GPU中的线程数被充分利用，减少甚至避免有线程空闲，保证GPU的线程级并行性。因此，按照分块参数集对应的线程数由大到小且块尺寸由小到大的顺序，对上述多个数据操作中每个数据操作对应的至少一个可用分块参数集进行排序，得到上述每个数据操作的可用分块参数集队列，将每个队列最前面的分块参数集作为当前分块参数集。

[0073] 403、确定上述多个数据操作的当前分块参数集对应的线程数的第一总和。

[0074] GPU中的线程总数是由硬件决定的，也就是说，GPU的线程总数是确定的当批执行数据操作时所需的线程数达到GPU的线程总数时，GPU中的线程是满负荷工作的。因此，设置一个线程总数的第一阈值，在批执行数据操作时所需的线程数达到第一阈值，即说明GPU中没有线程处于空闲。应理解，第一阈值不一定等于GPU的线程总数，可根据具体情况及需求进行调整，可选地，将第一阈值取为GPU的线程总数的80％。需要理解的是，第一阈值的具体大小通过实验和经验确定，本公开实施例中提到的“80％”只是一个举例，第一阈值也可以不以GPU的线程总数为基数进行表示，可以为具体的数值，如：256，对此，本公开实施例将不做具体限定。

[0075] 不同的可用分块参数集分块后得到的数据操作块的数量是不一样的，处理分块后得到的数据操作块所需的线程数也是不一样的。在数据操作的大小一定的情况下，分块后的数据操作块越小，分块得到的数据操作块的数量就越多，相应的，处理分块后得到的数据操作块所需的线程块的数量也就越多，在线程块的线程数大小一定的情况下，处理分块后得到的数据操作块所需的线程数就越多。可选地，通过下式可计算得到处理分块后得到的数据操作块所需的线程数。

[0076] $$TLP = \sum_i^B \frac{M_i \times N_i}{BY_i \times BX_i} \times 256 \cdots \text{公式（1）}$$

[0077] 其中，TLP为处理分块后得到的数据操作块所需的线程数，M、N、BY、BX的具体意义可参见401，0<i≤B，且i为整数，B为数据操作的数量。应理解，公式(1)为计算处理分块后得到的数据操作块所需的线程数的一种方式，具体计算处理分块后得到的数据操作块所需的线程数不局限于公式(1)，对此，本公开实施例不做限定。

[0078] 为保证处理分块后得到的数据块所需的线程总数达到第一阈值，通过公式(1)计

算以当前分块参数集对数据操作进行分块，在对分块后得到的数据操作块进行处理时，所需的线程数的第一总和，并将第一总和与第一阈值进行比较。

[0079] 404、在上述线程数的第一总和大于第一阈值的情况下，更新上述多个数据操作中的至少一个数据操作的当前分块参数集，直到满足更新截止条件。

[0080] 当前分块参数集对应的数据操作块最小，且对应的线程数最大，即当前分块参数集为保证GPU的线程级并行性的最佳分块参数集，但GPU在对数据操作进行处理时，除易出现线程级并行性的问题外，还易因为读取数据的方式不合理，导致GPU的指令级并行性较差。针对不同大小的数据操作，选择分块后得到的数据操作块较大的分块参数集，减小GPU读取数据时的单次读取量，增大GPU读取数据的次数，可提高GPU的指令级并行性。但是，GPU在执行数据操作时，往往不能同时兼顾线程级并行性和指令级并行性，本申请的策略是优先保证线程级并行，因此，将处理分块后得到的数据操作块所需的线程数大于所述第一阈值的所述可选分块参数集作为线程级分块参数集，线程级分块参数集中分块后得到的数据操作块最大的线程级分块参数集则可在满足线程级并行性的同时，尽可能的提高GPU的指令级并行性，因此，从所述线程级分块参数集中确定目标分块参数集，其中，所述目标分块参数集为分块块后得到的数据操作块最大的所述线程级分块参数集。。为了提高选取效率，通过将上述线程数的第一总和与第一阈值进行比较，如果第一总和大于第一阈值，则更新每个数据操作的当前数据操作，直到上述第一总和小于或等于第一阈值，并将满足更新截止条件的多个数据操作的当前分块参数集所对应的更新前的分块参数集作为所述多个数据操作的目标分块参数集，这样，可在不计算所有的可用分块参数集对应的线程总数的前提下，确定目标分块参数集，提高运算速度。

[0081] 本申请实施例中，首先根据可用分块参数集的线程数大小及分块后得到的块的大小确定当前分块参数集，将当前参数集对应的线程数总和与第一阈值进行比较筛选得到目标分块参数集，以在保证GPU线程级并行性的同时，提高GPU的指令级并行性。

[0082] 请参阅图5，图5是本申请实施例提供的批处理方法中步骤404的一种可能的实现方式的流程示意图。

[0083] 501、将上述多个数据操作中的第二数据操作的当前分块参数集由第一分块参数集更新为第二分块参数集。

[0084] 通过将上述第一总和与第一阈值进行比较，若上述线程数的第一总和大于第一阈值，则将当前分块参数集由分块参数集队列中的第一个分块参数集(第一分块参数集)从分块参数集队列中去除，得到更新后的分块参数集队列，并将更新前的分块参数集队列中的第二个分块参数集(第二分块参数集)作为更新后的分块参数集队列中的第一个分块参数集，这样，数据操作的当前分块参数集也就由更新前的分块参数集队列中的第一个分块参数集更新为更新前的分块参数集队列中的第二个分块参数集。当上述第一总和小于或等于第一阈值时，说明此时的当前分块参数集已不能满足线程级并行性，因此，将当前分块参数集队列更新前的当前分块参数集作为目标分块参数集。在一个可能实现的例子中，如图6所示，每个矩阵乘法都有一个对应的可用分块参数集队列，且按分块后得到的矩阵块的大小从从小到大进行排列，其中，图中队列的底部为队首，若第一总和小于或等于第一阈值，则将可用分块参数集队列的当前分块参数集作为目标分块参数集；若第一总和大于第一阈值，则将可用分块参数集队列的当前分块参数集(即队首的分块参数集)从队列中删除，并

更新可用分块参数集队列，若经过多次更新，可用分块参数集中已无T＝256的分块参数集，则将可用分块参数队列更新为T＝128，继续重复前面的计算过程，直到第一总和小于或等于第一阈值。应理解，为了便于说明，图5中可用分块参数集队列中的“小”、“中”、“大”泛指可用分块参数集中BY和BX的相对大小，即在可用分块参数集队列中，BY和BX小的分块参数集排在更靠前的位置。

[0085] 依据上述选择分块参数集的思路，可选地，从可用分块参数集中选取目标参数集的步骤如下：首先，将每一个数据操作的可用分块参数集按分块后得到的数据操作块从小到大进行排列，这样每一个数据操作都将得到一个可用分块参数集队列，将每个队列最前面的分块参数集作为当前分块参数集，并依据当前分块参数集对该数据操作进行分块，得到相应的数据操作块。然后计算处理这些数据操作块所需的线程数，将所需的线程数与第一阈值进行比较，若所需的线程数大于或等于第一阈值，则将每个队列中最前面的分块参数集从队列中去除，这样每一个数据操作都将得到一个更新后的队列，再对每个更新后队列的最前面的分块参数集(即更新后的当前分块参数集)重复前面的计算过程，直到所需的线程数小于第一阈值。应理解，上述步骤是选取目标分块参数集的一种思路，具体选取目标分块参数集的方式或顺序不局限于此，如：可以按线程数的从大到小的顺序，将可用分块参数集进行排列，得到第四队列，再按分块后得到的数据操作块从小到大的顺序，对所述第四队列进行调整，得到第五队列；也可以按分块后得到的数据操作块从小到大的顺序，将可用分块参数集进行排列，得到第六队列，再按线程数的从大到小的顺序，对所述第六队列进行调整，得到第七队列。对于选取目标分块参数集的具体顺序或思路，本公开实施例不做限定。

[0086] 在一个可能实现的例子中，提供了12种预先设定的分块参数集，具体可参见表1。由表1可知，这12种预先设定的分块参数集可分为线程数T＝128和T＝256两大类，每个大类又按分块后得到数据操作块的大小分别细分成了6个分块参数集。如202所述，将每个数据操作的分块参数集的线程数取为相同，可避免在对多个不同大小的数据操作进行批处理时有线程处于空闲状态的情况，再考虑到优先保证GPU的线程级并行性，因此，优先选择线程块的线程数大的分块参数集，即优先选择T＝256的分块参数集，在T＝256的分块参数集不符合要求的前提下，再选择T＝128的分块参数集。应理解，表1中所示的12种分块参数集仅用于举例说明，可选地，预先设定的分块参数集可以包括其他分块参数集，如：线程数为64，对此，本公开实施例不做具体限定。对于三个不同大小的数据操作A、B、C，A的可用分块参数集为：1,2,3,7,8,9；B的可用分块参数集为：1,2,3,4,7,8,9,10；C的可用分块参数集为：1,2,7,8(可用分块参数集的编号对应的分块参数具体参见表1)，则A的可用分块参数集队列为：[7,8,9,1,2,3]；B的可用分块参数集队列为：[7,8,9,10,1,2,3,4]；C的可用分块参数集队列为：[7,8,,1,2]。首先，计算如若用分块参数集7对A、B、C进行分块，处理分块后得到的数据操作块所需的线程数T1，如果T1大于或等于第一阈值，则将分块参数集7从每个队列中去除，这样，A的可用分块参数集队列就更新为：[8,9,1,2,3]，B的可用分块参数集队列就更新为：[8,9,10,1,2,3,4]，C的可用分块参数集队列就更新为：[8,1,2]。再根据公式(1)计算如若用分块参数集8对A、B、C进行分块，处理分块后得到的数据操作块所需的线程数T2，如果T2小于第一阈值，则将分块参数集7作为A、B、C的目标分块参数集。

[0087]

编号	BY	BX	BK	T
1	16	16	16	128
2	32	32	8	128
3	64	64	8	128
4	128	64	8	128
5	64	128	8	128
6	128	128	8	128
7	16	16	16	256
8	32	32	8	256
9	64	64	8	256
10	128	64	8	256
11	64	128	8	256
12	128	128	8	256

[0088] 表1

[0089] 502、响应于上述至少一个数据操作中的每个数据操作的至少一个可用分块参数集包括线程数与上述每个数据操作的当前分块参数集的线程数相同且块尺寸大于上述每个数据操作的当前分块参数集的块尺寸的第一可更新分块参数集，将上述至少一个数据操作中每个数据操作的当前分块参数集更新为上述第一可更新分块参数集。

[0090] 当某个队列中只剩下一种分块参数集时，即使所需的线程数大于或等于第一阈值，也不将最后一个分块参数集去除，而是保留下来，并继续进行下一次计算。在一个具体实现的例子中，继续以501中的A、B、C为例，如果T2大于或等于第一阈值，则将A的可用分块参数集队列和B的可用分块参数集队列中的分块参数集8去除，但不去除C的可用分块参数集队列中的分块参数集8，即A、B为可更新的队列。这样，A的可用分块参数集队列就更新为：[9,1,2,3]，B的可用分块参数集队列就更新为：[9,10,1,2,3,4]，C的可用分块参数集队列仍然为：[8,1,2]。再根据公式(1)计算如若用分块参数集9对A、B进行分块，并用分块参数集8对C进行分块，处理分块后得到的数据操作块所需的线程数T3，如果T3小于第一阈值，则将分块参数集8作为A、B、C的目标分块参数集。

[0091] 503、响应于上述至少一个数据操作中存在第三数据操作。

[0092] 若可用分块参数集队列中只有一个第一类可用分块参数集，确定每一个可用分块参数集队列中第一类可用分块参数集的数量，其中，所述第一类可用分块参数集为线程数为第一预设值的可用分块参数集；若所述每一个可用分块参数集中均只有一个第一类可用分块参数集，将所述可用分块参数集队列中的第一类可用分块参数集删除，得到更新后的可用分块参数集。可选地，若每个队列中都只剩下一种T＝256的分块参数集，且第一总和仍然大于或等于第一阈值，表明对数据操作而言，T＝256的分块参数集不能满足在保证线程级并行性的前提下，提高指令级并行性的要求，则将所有队列中T＝256的分块参数集去除，继续计算T＝128的分块参数集。在一个具体实现的例子中，继续以502中的A、B、C为例，若T3大于或等于第一阈值，则将数据B的队列中的分块参数集9去除，A、B两个队列则不作处理。这样，A的可用分块参数集队列仍然为：[9,1,2,3]，B的可用分块参数集队列就更新为：[10,

1,2,3,4],C的可用分块参数集队列仍然为:[8,1,2]。再根据公式(1)计算如若用分块参数集9对A进行分块,用分块参数集10对B进行分块,用分块参数集8对A进行分块,处理分块后得到的数据操作块所需的线程数T4。如果T4大于或等于第一阈值,由于A、B、C对应的队列中都只有一种T=256的分块参数集,此时,直接将数据操作A的队列中的分块参数集9和数据B的队列中的分块参数集10和数据C的队列中的分块参数集8都去除,这样,A的可用分块参数集队列更新为:[1,2,3],B的可用分块参数集队列就更新为:[1,2,3,4],C的可用分块参数集队列更新为:[1,2]。再根据公式(1)计算如若用分块参数集1对A、B、C进行分块,处理分块后得到的数据操作块所需的线程数T5,如果T5小于第一阈值,则将分块参数集9作为A的目标分块参数集,将分块参数集10作为B的目标分块参数集,将分块参数集8作为C的目标分块参数集;如果T5大于或等于第一阈值,则对剩余的T=128的分块参数集重复前面对T=256的分块参数集的处理,直到为每一组数据确定一种目标分块参数集。

[0093] 本申请实施例通过同时更新每个数据操作的当前分块参数集,使得每个数据操作的目标分块参数集的线程数相同,保证了GPU的线程级并行性;通过对可用分块参数集进行排序确定当前分块参数集,并以更新当前分块参数集的方式,确定目标分块参数集,能高效筛选出在保证线程级并行性的同时能很好的兼顾指令级并行性的分块参数集。

[0094] 请参阅图7,图7是本申请实施例提供的批处理方法中步骤103的一种可能的实现方式的流程示意图。

[0095] 701、为上述多个数据操作块中的至少一个第一数据操作块分配线程块。

[0096] 依据目标分块参数集将数据操作进行分块,得到包括多个不同大小的第一数据操作块的分块结果。GPU将通过线程块来处理分块后得到的多个数据操作块,需要理解的是,一个线程块可能只执行一个数据操作块,也可能同时执行多个数据操作块。在一个可能实现的例子中,102中的8个数据操作块可以由8个不同的线程块分别执行。也可由2个线程块来执行,即每个线程块执行4个数据操作块。

[0097] 702、确定为当前已分配线程块的上述至少一个第一数据操作块所分配的线程块中包含的总线程数与上述多个数据操作块中当前还未分配线程块的多个第二数据操作块的数量的第二总和。

[0098] 由于处理尺寸较小的第一数据操作块所需的线程数较少,若用一个线程块来处理一个尺寸较小的数据操作块,会导致线程块中的部分线程处于空闲状态。因此,为保证每个线程块中线程满负荷工作,需要对数据操作块合理分配至线程块上进行处理,如:将多个尺寸较小的数据操作块分配到同一个线程块上进行处理,将较大的数据操作块单独分配到一个线程块上进行处理。

[0099] 设置第二阈值,使执行数据操作块所需的线程数总和大于第二阈值,以保证GPU的线程级并行性。显然,已分配线程块的线程数是确定的,将未分配线程块的任意一个第二数据操作块的视为由一个线程块单独处理,因此,处理每一个未分配线程块的第二数据操作块所需的线程块的数量就为未分配线程块的第二数据操作块的数量。计算已分配的线程块中包含的总线程数与处理还未分配线程块的多个第二数据操作块所需的线程数的第二总和。

[0100] 703、基于上述第二总和与第二阈值的大小关系,为上述当前还未分配线程块的多个第二数据操作块分配线程块。

[0101] 若第二总和大于或等于第二阈值，说明此时，还可以继续减少处理第一数据操作块所需的线程总数，继续从未分配线程块的多个第二数据操作块中选取至少一个数据操作块分配至同一个线程块。若第二总和小于第二阈值，说明此时处理第一数据操作块所需的线程总数已接近保证线程级并行性的极限，因此，将剩余所有未分配线程块的多个第二数据操作块分配至同一个线程块上进行处理，提高指令级并行性。

[0102] 请参阅图8，图8是本申请实施例提供的批处理方法中步骤701～703的一种具体的实现方式的流程示意图。

[0103] 801、为上述多个数据操作块中尺寸参数值最大的数据操作块和所述尺寸参数值最小的数据操作块分配同一个线程块。

[0104] 将第六队列中的第一个数据操作块和最后一个数据操作块分配至第一线程块，其中，所述第六队列为将所述至少一个第一数据操作块按尺寸参数从小到大的顺序进行排列得到的队列；计算处理未分配线程块的数据操作块所需的线程数和所述第一线程块包含的线程数的总和，得到第一线程数量；若所述第一线程数量小于所述第二阈值，将所述未分配线程块的数据操作块分配至第二线程块。

[0105] 802、在第二总和大于所述第二阈值的情况下，为当前还未分配线程块的多个第二数据操作块中尺寸参数值最大的数据操作块和尺寸参数值最小的数据操作块分配同一个线程块。

[0106] 若第二总和小于或等于第二阈值，将所有未分配线程块的数据操作块分配至同一个线程块。

[0107] 若第二总和大于第二阈值，将未分配线程块的数据操作块按尺寸参数从小到大的顺序进行排列，得到第二队列，再将第二队列中的第一个数据操作块和最后一个数据操作块分配给同一个线程块，再计算已分配的线程块的线程数总和与为分配线程块的数据操作块的总和(即第三总和)，比较第三总和与第二阈值的大小，并根据比较的结果重复801与802中的步骤，直至所有数据操作块均被分配至线程块或已分配的线程块的线程数总和与未分配的线程块的数量的总和小于或等于第二阈值。需要理解的是，在对数据操作块进行分配的过程中，并不会将不同大小的数据操作块放到一起进行分配。

[0108] 本申请实施例中，以将未分配线程块的数据操作块中尺寸最大的数据操作块和尺寸最小的数据操作块分配至同一个线程块的方式，将数据操作块分配线程块上进行处理。

[0109] 请参阅图9，图9是本申请实施例提供的批处理方法中步骤701～703的另一种具体的实现方式的流程示意图。

[0110] 901、将上述多个数据操作块中相邻的N个第一数据操作块分配同一个线程块。

[0111] 当分配到一个线程块里的所有第一数据操作块的K的和大于第三阈值时，即使再往这个线程块中增加第一数据操作块，也不会对提高GPU的指令级并行性有帮助，因此，首先确定满足K的和大于第三阈值所需的第一数据操作块的最少数量，在一个具体实现的例子中：设N为大于或等于1的整数，N个第一数据操作块的K的和大于第三阈值，且N-1个第一数据操作块的K的和小于或等于第三阈值，则大于第三阈值所需的第一数据操作块的数量为N。如：第三阈值为256，第一数据操作块的K的值为60，那么4个第一数据操作块的K的和为240，小于256，且5个第一数据操作块的K的和为300，大于256，这样N就为5。将N个第一数据操作块分配至同一个线程块，并计算已分配的线程块的线程数与还未分配线程块的数据操

作块的数量的和(即第二总和)。

[0112] 902、在所述第二总和大于所述第二阈值的情况下,为当前还未分配线程块的多个第二数据操作块中相邻的M个数据操作块分配同一个线程块,以使得所述M个数据操作块的尺寸参数值之和恰好大于所述第三阈值。

[0113] 若第四总和小于或等于第二阈值,将所有未分配线程块的数据操作块分配至同一个线程块。

[0114] 若第四总和大于第二阈值,从未分配线程块的数据操作块中选取M个数据操作块,并将此次选出的M个数据操作块分配至同一个线程块,计算已分配的线程块的线程数与还未分配线程块的数据操作块的数量的和(即第四总和),比较第五总和与第二阈值的大小,并根据比较的结果重复901与902中的步骤,直至所有数据操作块均被分配至线程块或已分配的线程块的线程数总和与未分配的线程块的数量的总和小于或等于第二阈值,其中,M等于N。需要理解的是,在对数据操作块进行分配的过程中,并不会将不同大小的数据操作块放到一起进行分配,而不同大小的数据操作的目标分块参数集和分块后得到的数据操作块的大小不同,因此,对于不同大小的数据操作,N的取值是不一样的。

[0115] 本申请实施例以数据操作块的尺寸参数及尺寸参数阈值为依据,将数据操作分配至线程块。

[0116] 下例为本申请提供的另一种批处理方法的实施例。

[0117] 给定一个矩阵乘法C＝α×A×B+β×c,GPU上经典的并行方法首先对C进行分块,如图2所示,每一个矩阵块由一个线程块负责执行。每一个矩阵块会读A中的一整行和B中的一整列,因此矩阵块的大小BY×BX直接影响着一个线程块的计算任务,同时也决定了矩阵C可以分成的矩阵块的数目,也就是需要使用的线程块的数目。此外,由于读取A中的一整行BY×K和B中的一整列BX×K需要占用较大的存储资源,为了利用有限的GPU片上存储资源,需要将A中的一整行和B中的一整列进行切割,每次只读取A的一段BY×BK,和B的一段BK×BX,然后进行多次读取。切割的参数BK可以影响一个线程块所使用的片上存储资源的数量,从而影响每个GPU上最多可以同时执行的线程块的数目。另外一个重要的参数是一个线程块中的线程数目T,它和矩阵块大小同时共同决定了完成一个矩阵乘法所需要的线程数目。因此,分块策略指的就是(BY,BX,BK,T),这样一个四元组。

[0118] 传统的分块策略是针对单个矩阵乘法的场景,不同的矩阵分块大小(BY,BX,BK)往往对应不同的线程数目(T)。在批执行矩阵乘法的场景下,不同的矩阵乘法中矩阵大小不一样,往往需要不同的分块策略。如果我们使用传统的分块策略,会使得线程块大小不一致。例如,第一个矩阵乘法选取的分块策略为(BY_1,BX_1,BK_1,T_1),第二个矩阵乘法选取的分块策略为(BY_2,BX_2,BK_2,T_2),其中T_1>T_2。因此,GPU核函数的线程块的大小为两者的较大值T_2。此时执行第二个矩阵乘法的线程块中必然会有(T_1-T_2)的线程处于空闲状态,使得GPU的利用不充分。为此,本公开实施例设计了针对批执行矩阵乘法的分块策略,如表2所示。对于每一种矩阵块大小我们都设计了两种分块策略分别对应于T＝128和T＝256。

[0119] 分块决策算法负责为每一个矩阵乘法选择合适的分块策略。决策算法为每个矩阵乘法选择的分块策略拥有相同的T,即T＝128或者T＝256。

[0120] 决策算法优先考虑提高线程级并行性,也就是我们选择的分块策略会优先提高计算所有的矩阵乘法所需要的线程数。因此我们优先选择T＝256的分块策略,同时优先选择

较小的分块策略，比如Small-128和Small-256。因为，矩阵块越小，矩阵可以划分的矩阵块总数越多，需要的线程块数目越多。保证线程级并行性的前提下，我们倾向于采用较大的分块策略，比如Huge-128和Huge-256提高指令级并行性。

[0121] 基于以上的观点，首先选择优先选择T＝256的分块策略，对于每个矩阵乘法将其可用的分块决策策略按照矩阵块从小到大的优先顺序排序放在各自的可用分块策略队列中。每次从队首获取每个矩阵乘法的分块策略，并计算按照当前分块策略分块之后的总线程数TLP，计算公式为给定B个矩阵乘法：$TLP = \sum_i^B \frac{M_i \times N_i}{BY_i \times BX_i} \times 256$，其中$M_i$和$N_i$为第i个矩阵乘法中C的大小，$BY_i \times BX_i$为分块策略的矩阵块大小。如果TLP大于一个阈值，我们就随机选一个矩阵乘法的分块策略队列，进行删除操作，使用一个较大的分块策略。注意，不会使用只有一个可用分块策略的矩阵乘法进行删除操作。重复以上操作，直到TLP小于该阈值。此时，选择每个优先队列队首的分块策略作为该矩阵乘法的分块策略。如果所有的队列都只剩一个可用分块策略，TLP仍大于该阈值，我们选择T＝128的分块策略，重复以上操作。

[0122] 实验发现将多个K较小的矩阵块分配给一个线程块，让它们顺序执行，有机会提高指令级并行性。因此，首先设计了两种批执行策略，来指导矩阵块分配。然后，设计了一个批执行决策算法，决策使用哪一种批执行策略，对于一个特定的批执行矩阵乘法。

[0123] 该策略优先保证线程级并行性，在此前提下，尝试提高指令级并行性。实验发现当分配到一个线程块里所有矩阵块K的和大于一个阈值(在英伟达Volta 100GPU中该阈值取256)，再增加矩阵块，对于指令级并行性的提高作用不再明显。因此，在进行矩阵块分配之前，首先计算已经分配的线程块中的线程总数，以及未分配的矩阵块所需要的线程总数，如果这两个值的和小于TLP的阈值，为所有未分配的矩阵块各分配一个线程块，该批执行策略结束；如果这两个值的和大于TLP的阈值，那么取一定数目的矩阵块将它们分配到一个线程块中，并且使得这个线程块中的K值之和恰好大于256，然后重复上述操作，直到批执行策略结束。

[0124] 该策略与阈值批执行策略相比，对指令级并行性给了更高的优先级。我们将所有的矩阵块按照K进行排序。每次将有序序列中第一个矩阵块和最后一个矩阵块分配到同一个线程块中，然后用同样的方式检查TLP，如果大于阈值，我们继续进行上述操作；否则，将剩下的矩阵块各自分配一个线程块。

[0125] 前面已经给出了两种将数据块划分为数据块集合的方式，通过这两种方式，即可确定处理多组待处理数据所需的线程块的数量及大小，并确定了不同线程块处理的数据块。以上两种划分方式虽然都能确定线程块的数量及大小，但在具体执行时，两种方式得到的效果存在差异，因此需要在两种方式中选择处理效果更好的一种划分方式。

[0126] 实际应用中，待处理数据可分为离线和在线两种类型。离线的待处理数据的大小都是确定的，这种情况可以比较通过两种方式的效果，择优选择其中一种划分方式。

[0127] 在线的待处理数据，其大小都是不确定的，本申请采用随机森林算法进行选择，如图10所示，随机森林算法由多个决策树组成。决策树中的圆形节点会对输入的特征向量中的某一维的大小与第四阈值的大小进行比较，本申请输入的特征向量为[M,N,K,B]，其中，M,N,K(此处的含义与图2中的含义相同)为所有待处理数据的平均值，B为待处理数据的数量(以组为单位)。在输入特征向量后，相应的方形节点会给出分别选择这上述两种划分方

式的概率值，然后根据对比结果确定下一步路径，最终的方形节点将所有概率值相加，得到两种方式的最终概率值，再选择概率值较大的划分方式即可。需要指出的是，随机森林算法需要训练，第四阈值则是通过训练得到的，具体训练过程此处将不再阐述。

[0128] 上述详细阐述了本申请实施例的方法，下面提供了本申请实施例的装置。

[0129] 请参阅图11，图11为本申请实施例提供的一种批处理数据的装置的结构示意图，该装置1包括：选取单元11、处理单元12及分配单元13。其中：

[0130] 选取单元11，用于从多个分块参数集中选取多个数据操作中每个数据操作对应的目标分块参数集，其中，所述多个数据操作对应的目标分块参数集对应相同的线程数；

[0131] 处理单元12，用于根据所述多个数据操作中每个数据操作对应的目标分块参数集，对所述每个数据操作进行分块处理，得到多个数据操作块；

[0132] 分配单元13，用于将所述多个数据操作块分配到线程块上执行。

[0133] 进一步地，所述选取单元11包括：第一确定子单元111，用于从所述多个分块参数集中确定所述多个数据操作中每个数据操作的至少一个可用分块参数集；第一选取子单元112，用于从多个数据操作中每个数据操作对应的至少一个可用分块参数集中选取所述每个数据操作的当前分块参数集；计算子单元113，用于确定所述多个数据操作的当前分块参数集对应的线程数的第一总和；更新子单元114，用于在所述线程数的第一总和大于第一阈值的情况下，更新所述多个数据操作中的至少一个数据操作的当前分块参数集，直到满足更新截止条件，其中，所述更新截止条件包括所述多个数据操作的当前分块参数集对应的线程数的第一总和小于或等于所述第一阈值，并将满足所述更新截止条件的所述多个数据操作的当前分块参数集所对应的更新前的分块参数集作为所述多个数据操作的目标分块参数集。

[0134] 进一步地，所述第一选取子单元112具体用于：从第一数据操作的至少一个可用分块参数集中选取对应的线程数最大且数据操作块尺寸最小的分块参数集作为所述第一数据操作的当前分块参数集，其中，所述多个数据操作包括所述第一数据操作。

[0135] 进一步地，所述更新子单元114具体用于：将所述多个数据操作中的第二数据操作的当前分块参数集由第一分块参数集更新为第二分块参数集，其中，所述第二数据操作的至少一个可用分块参数集包括所述第一分块参数集和所述第二分块参数集，所述第二分块参数集对应的数据操作块尺寸大于所述第一分块参数集的数据操作块尺寸或者所述第二分块参数集对应的线程数小于所述第一分块参数集对应的线程数。

[0136] 进一步地，所述第一选取子单元114还用于：按照分块参数集对应的线程数由大到小且块尺寸由小到大的顺序，对所述多个数据操作中每个数据操作对应的至少一个可用分块参数集进行排序，得到所述每个数据操作的可用分块参数集队列；以及所述从多个数据操作中每个数据操作对应的至少一个可用分块参数集中选取所述每个数据操作的当前分块参数集；以及从多个数据操作中每个数据操作的可用分块参数集队列中的首个分块参数集作为所述每个数据操作的当前分块参数集；以及所述更新所述多个数据操作中的至少一个数据操作的当前分块参数集，包括：以及在所述第一总和大于第一阈值的情况下，将所述至少一个数据操作中每个数据操作的当前分块参数集从所述每个数据操作的可用分块参数集队列中删除，并将所述每个数据操作的当前分块参数集更新为所述每个数据操作的更新后的可用分块参数集队列中的首个分块参数集。

[0137] 进一步地，所述更新子单元114具体用于：响应于所述至少一个数据操作中的每个数据操作的至少一个可用分块参数集包括线程数与所述每个数据操作的当前分块参数集的线程数相同且块尺寸大于所述每个数据操作的当前分块参数集的块尺寸的第一可更新分块参数集，将所述至少一个数据操作中每个数据操作的当前分块参数集更新为所述第一可更新分块参数集；和/或响应于所述至少一个数据操作中存在第三数据操作，其中，所述第三数据操作的至少一个可用分块参数集不包括所述第一可更新分块参数集，将所述至少一个数据操作中每个数据操作的当前分块参数集更新为线程数小于所述每个数据操作的当前分块参数集的线程数的第二可更新分块参数集。

[0138] 进一步地，所述分配单元13包括：第一分配子单元131，用于为所述多个数据操作块中的至少一个第一数据操作块分配线程块；第二确定子单元132，用于确定为当前已分配线程块的所述至少一个第一数据操作块所分配的线程块中包含的总线程数与所述多个数据操作块中当前还未分配线程块的多个第二数据操作块的数量的第二总和；第二分配子单元133，用于基于所述第二总和与第二阈值的大小关系，为所述当前还未分配线程块的多个第二数据操作块分配线程块。

[0139] 进一步地，所述第二分配子单元133还用于：在所述第二总和小于或等于第二阈值的情况下，为所述多个第二数据操作块中的不同数据操作块分配不同的线程块。

[0140] 进一步地，所述第二分配子单元133还用于：在所述第二总和大于所述第二阈值的情况下，为所述多个第二数据操作块中的至少两个数据操作块分配同一个线程块。

[0141] 进一步地，所述第二确定子单元132还用于：为所述多个第一数据操作块中的至少两个数据操作块分配同一个线程块。

[0142] 进一步地，所述第一分配子单元131还用于：为所述多个数据操作块中尺寸参数值最大的数据操作块和所述尺寸参数值最小的数据操作块分配同一个线程块。

[0143] 进一步地，所述第二分配子单元133还用于：在所述第二总和大于所述第二阈值的情况下，为当前还未分配线程块的多个第二数据操作块中尺寸参数值最大的数据操作块和尺寸参数值最小的数据操作块分配同一个线程块。

[0144] 进一步地，所述第一分配子单元131还用于：将所述多个数据操作块中相邻的N个第一数据操作块分配同一个线程块，其中，所述N为大于1的整数，所述N个第一数据操作块的尺寸参数之和大于第三阈值且所述N个第一数据块中的前N-1个第一数据操作块的尺寸参数之和小于或等于所述第三阈值。

[0145] 进一步地，所述第一分配子单元131还用于：在所述第二总和大于所述第二阈值的情况下，为当前还未分配线程块的多个第二数据操作块中相邻的M个数据操作块分配同一个线程块，以使得所述M个数据操作块的尺寸参数值之和恰好大于所述第三阈值。

[0146] 进一步地，所述数据操作块的尺寸参数为所述数据操作块包括的第一矩阵与第二矩阵的矩阵乘法中的第一矩阵的列数。

[0147] 进一步地，所述数据操作为矩阵乘法或卷积运算。

[0148] 在一些实施例中，本公开实施例提供的装置具有的功能或包含的单元可以用于执行上文方法实施例描述的方法，其具体实现可以参照上文方法实施例的描述，为了简洁，这里不再赘述。

[0149] 图12为本申请实施例提供的一种批处理装置的硬件结构示意图。该处理装置2包

括流多处理器21，还可以包括输入装置22、输出装置23和显卡内存24。该输入装置22、输出装置23、显卡内存24和流多处理器21之间通过总线相互连接。

[0150] 显卡内存包括但不限于是随机存储记忆体(random access memory，RAM)、只读存储器(read-only memory，ROM)、可擦除可编程只读存储器(erasable programmable read only memory，EPROM)、或便携式只读存储器(compact disc read-oNly memory，CD-ROM)，该存储器用于相关指令及数据。

[0151] 输入装置用于输入数据和/或信号，以及输出装置用于输出数据和/或信号。输出装置和输入装置可以是独立的器件，也可以是一个整体的器件。

[0152] 流多处理器可以是一个，也可以是多个，流多处理器可以是单核，也可以是多核包括是一个或多个处理器，例如包括一个或多个中央处理器(ceNtralprocessiNguNit，CPU)，在处理器是一个CPU的情况下，该CPU可以是单核CPU，也可以是多核CPU。

[0153] 流多处理器可以是一个，也可以是多个，流多处理器是众核架构，包含多个计算单元。

[0154] 显卡内存用于存储网络设备的程序代码和数据。

[0155] 流多处理器用于调用该存储器中的程序代码和数据，执行上述方法实施例中的步骤。具体可参见方法实施例中的描述，在此不再赘述。

[0156] 可以理解的是，图12仅仅示出了一种批处理数据的装置的简化设计。在实际应用中，批处理数据的装置还可以分别包含必要的其他元件，包含但不限于任意数量的输入/输出装置、流处理器、控制器、存储器等，而所有可以实现本申请实施例的批处理数据的装置都在本申请的保护范围之内。

[0157] 本领域普通技术人员可以意识到，结合本文中所公开的实施例描述的各示例的单元及算法步骤，能够以电子硬件、或者计算机软件和电子硬件的结合来实现。这些功能究竟以硬件还是软件方式来执行，取决于技术方案的特定应用和设计约束条件。专业技术人员可以对每个特定的应用来使用不同方法来实现所描述的功能，但是这种实现不应认为超出本申请的范围。

[0158] 所属领域的技术人员可以清楚地了解到，为描述的方便和简洁，上述描述的系统、装置和单元的具体工作过程，可以参考前述方法实施例中的对应过程，在此不再赘述。

[0159] 在本申请所提供的几个实施例中，应该理解到，所揭露的系统、装置和方法，可以通过其它的方式实现。例如，以上所描述的装置实施例仅仅是示意性的，例如，所述单元的划分，仅仅为一种逻辑功能划分，实际实现时可以有另外的划分方式，例如多个单元或组件可以结合或者可以集成到另一个系统，或一些特征可以忽略，或不执行。另一点，所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些接口，装置或单元的间接耦合或通信连接，可以是电性，机械或其它的形式。

[0160] 所述作为分离部件说明的单元可以是或者也可以不是物理上分开的，作为单元显示的部件可以是或者也可以不是物理单元，即可以位于一个地方，或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部单元来实现本实施例方案的目的。

[0161] 另外，在本申请各个实施例中的各功能单元可以集成在一个处理单元中，也可以是各个单元单独物理存在，也可以两个或两个以上单元集成在一个单元中。

[0162] 在上述实施例中，可以全部或部分地通过软件、硬件、固件或者其任意组合来实现。当使用软件实现时，可以全部或部分地以计算机程序产品的形式实现。所述计算机程序产品包括一个或多个计算机指令。在计算机上加载和执行所述计算机程序指令时，全部或部分地产生按照本申请实施例所述的流程或功能。所述计算机可以是通用计算机、专用计算机、计算机网络、或者其他可编程装置。所述计算机指令可以存储在计算机可读存储介质中，或者通过所述计算机可读存储介质进行传输。所述计算机指令可以从一个网站站点、计算机、服务器或数据中心通过有线(例如同轴电缆、光纤、数字用户线(digital subscriber line，DSL))或无线(例如红外、无线、微波等)方式向另一个网站站点、计算机、服务器或数据中心进行传输。所述计算机可读存储介质可以是计算机能够存取的任何可用介质或者是包含一个或多个可用介质集成的服务器、数据中心等数据存储设备。所述可用介质可以是磁性介质，(例如，软盘、硬盘、磁带)、光介质(例如，数字通用光盘(digital versatile disc，DVD))、或者半导体介质(例如固态硬盘(solid state disk，SSD))等。

[0163] 本领域普通技术人员可以理解实现上述实施例方法中的全部或部分流程，该流程可以由计算机程序来指令相关的硬件完成，该程序可存储于计算机可读取存储介质中，该程序在执行时，可包括如上述各方法实施例的流程。而前述的存储介质包括：只读存储器(read-only memory，ROM)或随机存储存储器(random access memory，RAM)、磁碟或者光盘等各种可存储程序代码的介质。

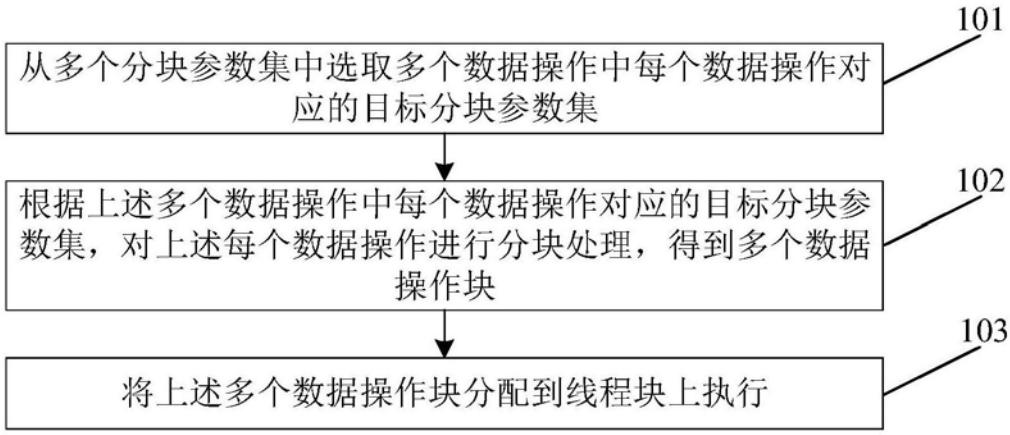

101
从多个分块参数集中选取多个数据操作中每个数据操作对应的目标分块参数集
102
根据上述多个数据操作中每个数据操作对应的目标分块参数集，对上述每个数据操作进行分块处理，得到多个数据操作块
103
将上述多个数据操作块分配到线程块上执行

图1

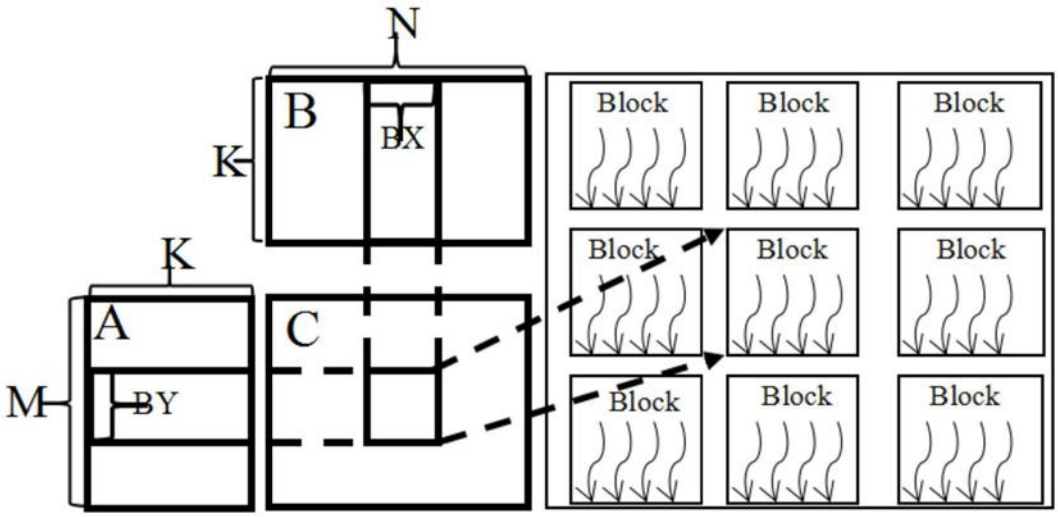

N
B
BX
K
K
A
C
M
BY
Block
Block
Block
Block
Block
Block
Block
Block
Block

图2

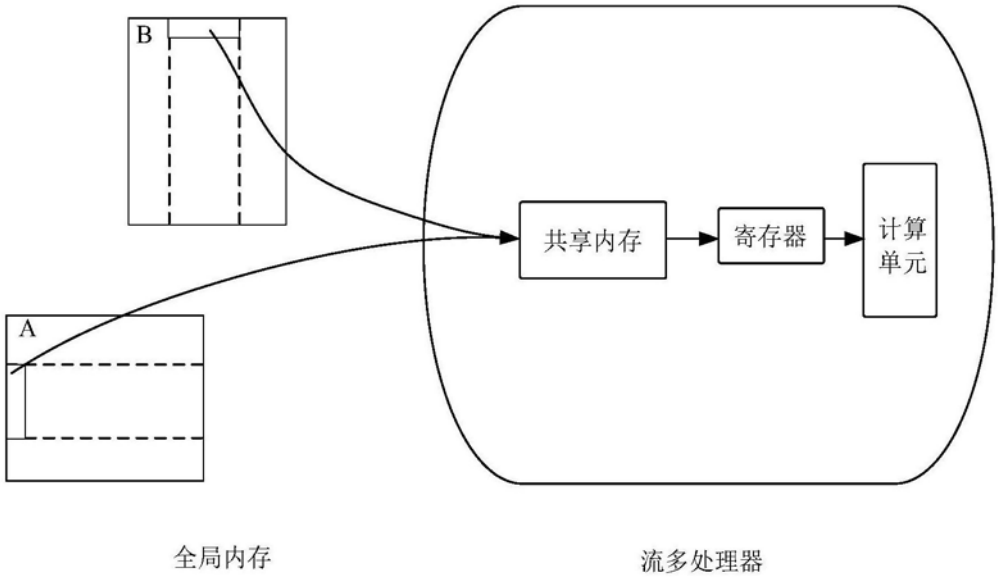

B
A
共享内存
寄存器
计算单元
全局内存
流多处理器

图3

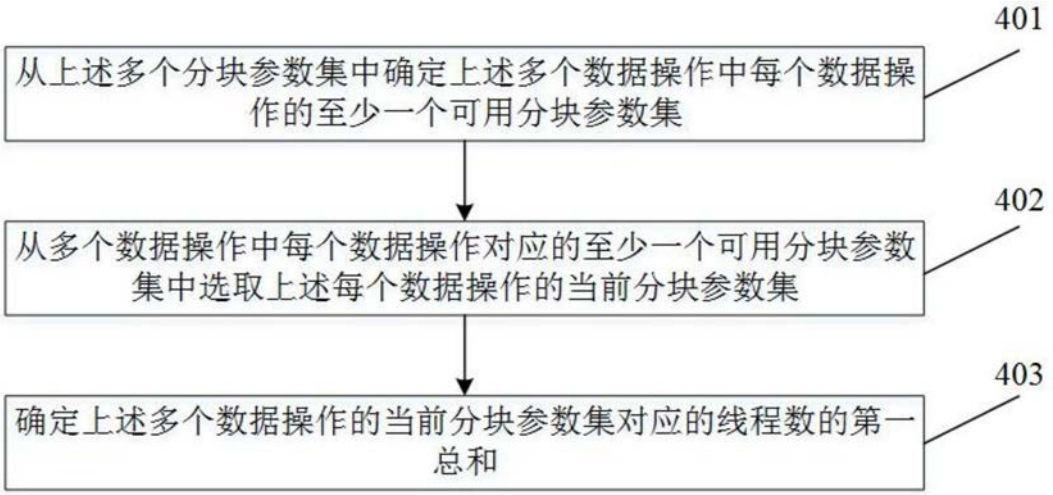

401
从上述多个分块参数集中确定上述多个数据操作中每个数据操作的至少一个可用分块参数集
402
从多个数据操作中每个数据操作对应的至少一个可用分块参数集中选取上述每个数据操作的当前分块参数集
403
确定上述多个数据操作的当前分块参数集对应的线程数的第一总和

图4

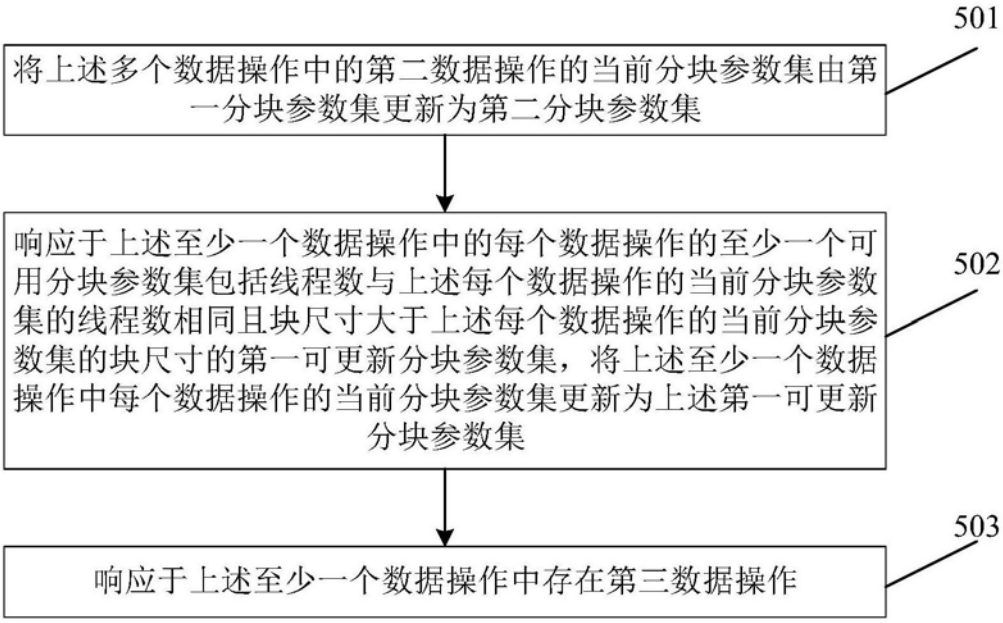

501
将上述多个数据操作中的第二数据操作的当前分块参数集由第一分块参数集更新为第二分块参数集
502
响应于上述至少一个数据操作中的每个数据操作的至少一个可用分块参数集包括线程数与上述每个数据操作的当前分块参数集的线程数相同且块尺寸大于上述每个数据操作的当前分块参数集的块尺寸的第一可更新分块参数集，将上述至少一个数据操作中每个数据操作的当前分块参数集更新为上述第一可更新分块参数集
503
响应于上述至少一个数据操作中存在第三数据操作

图5

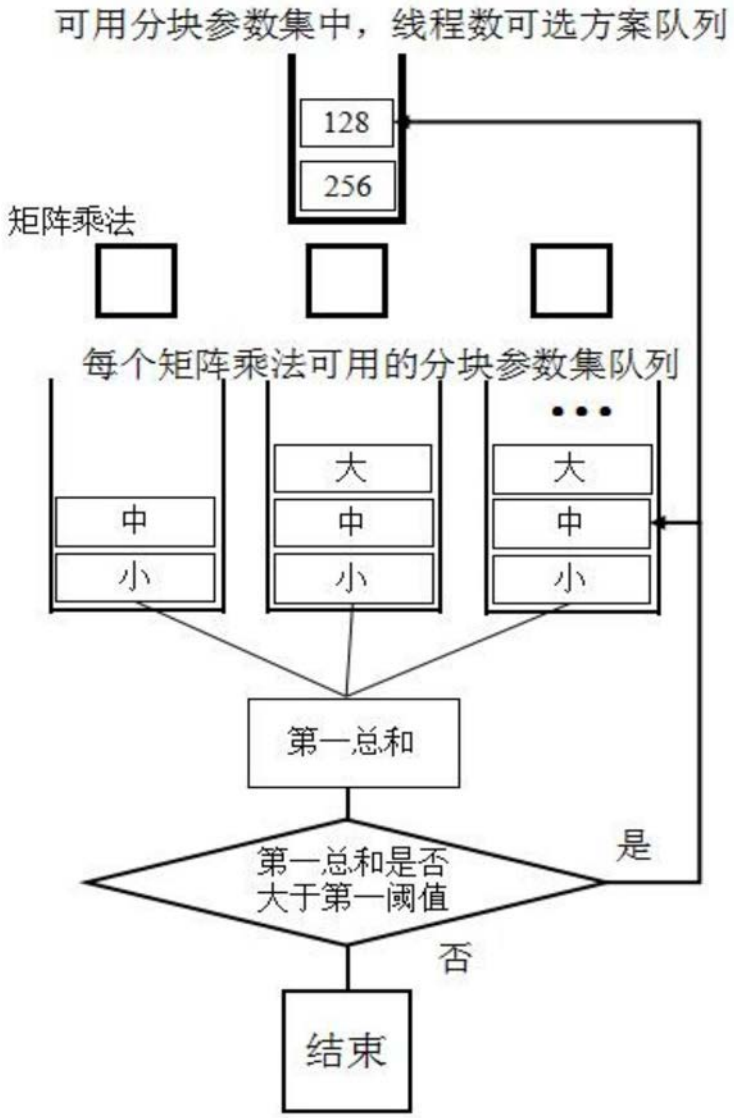

可用分块参数集中，线程数可选方案队列
128
256
矩阵乘法
每个矩阵乘法可用的分块参数集队列
大
中
小
大
中
小
中
小
第一总和
第一总和是否
大于第一阈值
是
否
结束

图6

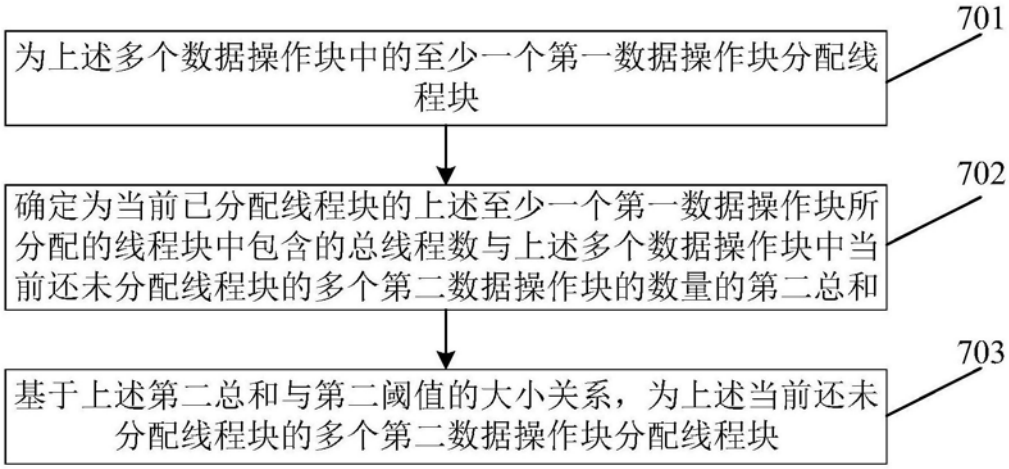

701
为上述多个数据操作块中的至少一个第一数据操作块分配线程块
702
确定为当前已分配线程块的上述至少一个第一数据操作块所分配的线程块中包含的总线程数与上述多个数据操作块中当前还未分配线程块的多个第二数据操作块的数量的第二总和
703
基于上述第二总和与第二阈值的大小关系，为上述当前还未分配线程块的多个第二数据操作块分配线程块

图7

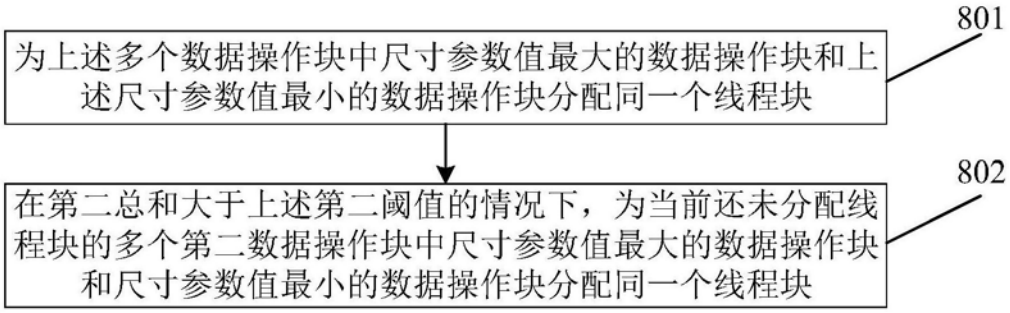

801
为上述多个数据操作块中尺寸参数值最大的数据操作块和上述尺寸参数值最小的数据操作块分配同一个线程块
802
在第二总和大于上述第二阈值的情况下，为当前还未分配线程块的多个第二数据操作块中尺寸参数值最大的数据操作块和尺寸参数值最小的数据操作块分配同一个线程块

图8

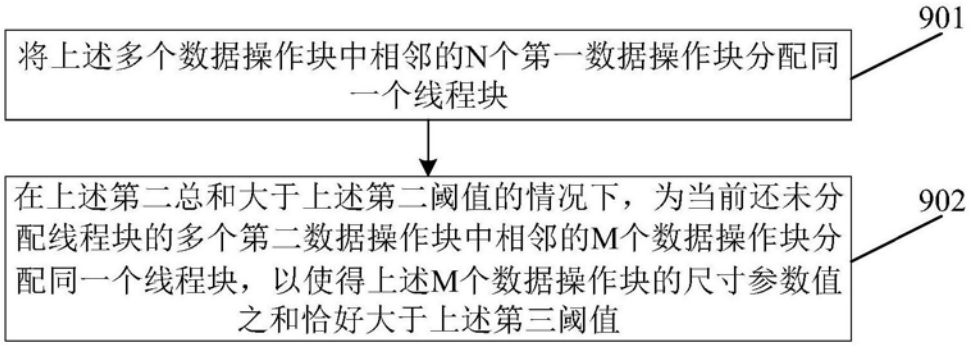

901
将上述多个数据操作块中相邻的N个第一数据操作块分配同一个线程块
902
在上述第二总和大于上述第二阈值的情况下，为当前还未分配线程块的多个第二数据操作块中相邻的M个数据操作块分配同一个线程块，以使得上述M个数据操作块的尺寸参数值之和恰好大于上述第三阈值

图9

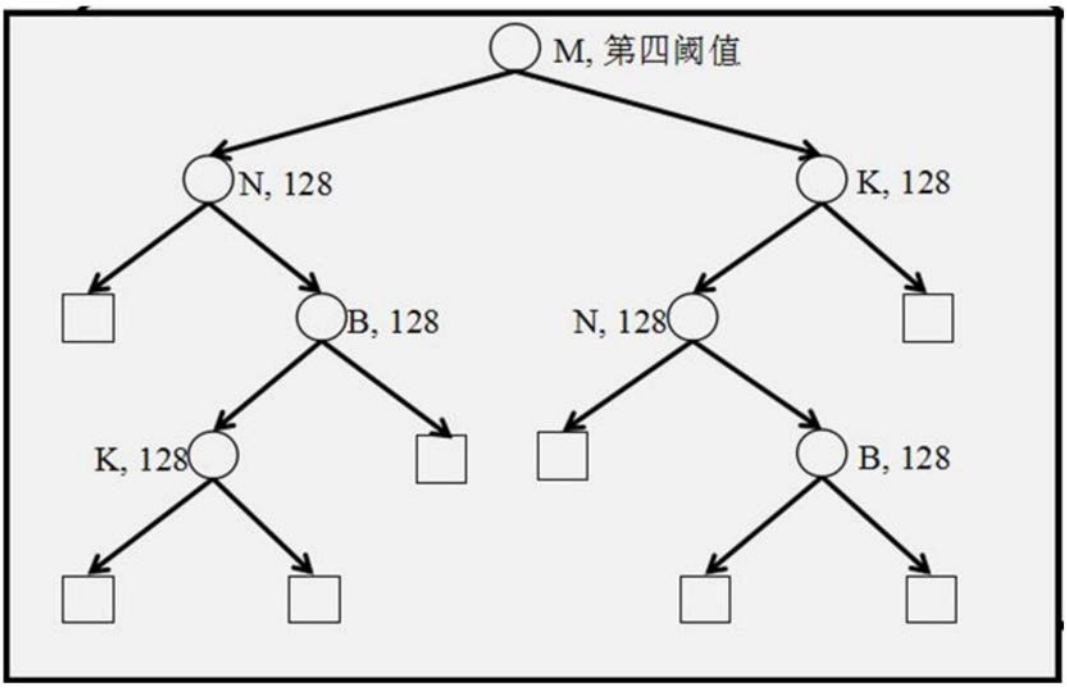

M, 第四阈值
N, 128
K, 128
B, 128
N, 128
K, 128
B, 128

图10

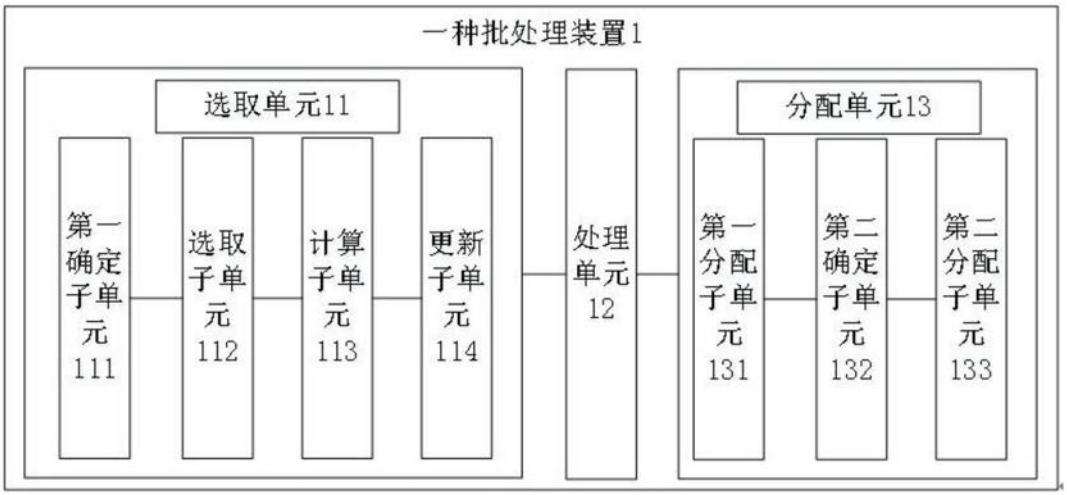

一种批处理装置1
选取单元11
第一确定子单元111
选取子单元112
计算子单元113
更新子单元114
处理单元12
分配单元13
第一分配子单元131
第二确定子单元132
第二分配子单元133

图11

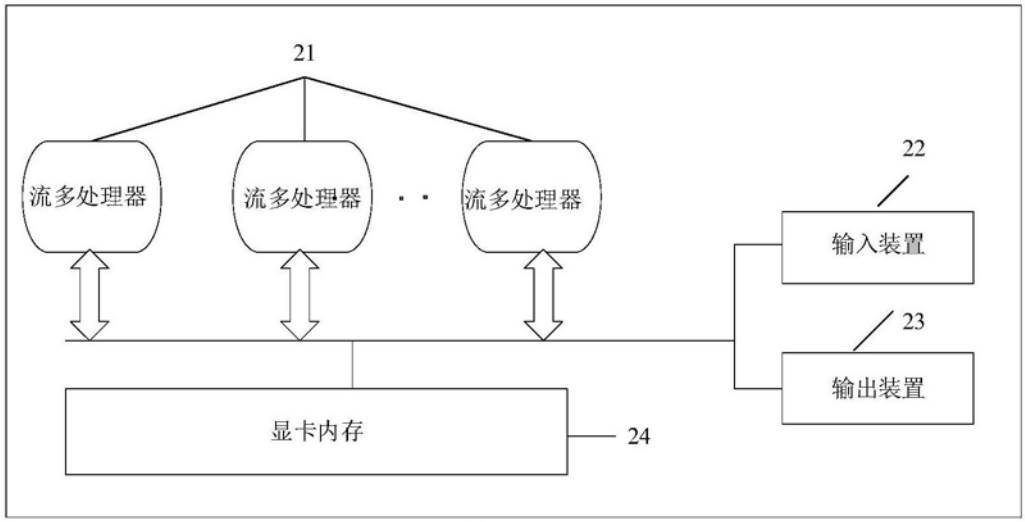

21
流多处理器
流多处理器
流多处理器
22
输入装置
23
输出装置
显卡内存
24

图12