



# (12) 发明专利申请

(10) 申请公布号 CN 115774736 A

(43) 申请公布日 2023. 03. 10

(21) 申请号 202310095934.7

G06F 16/22 (2019.01)

(22) 申请日 2023.02.10

G06F 16/901 (2019.01)

(71) 申请人 之江实验室

G06F 16/903 (2019.01)

地址 311121 浙江省杭州市余杭区中泰街  
道之江实验室南湖总部

G06F 9/50 (2019.01)

(72) 发明人 程永利 陈光 曾令仿 程宏才  
陈兰香 李勇 朱健 张云云  
张丽颖

(74) 专利代理机构 杭州求是专利事务所有限公  
司 33200

专利代理师 邱启旺

(51) Int. Cl.

G06F 16/2455 (2019.01)

G06F 16/245 (2019.01)

G06F 16/2453 (2006.01)

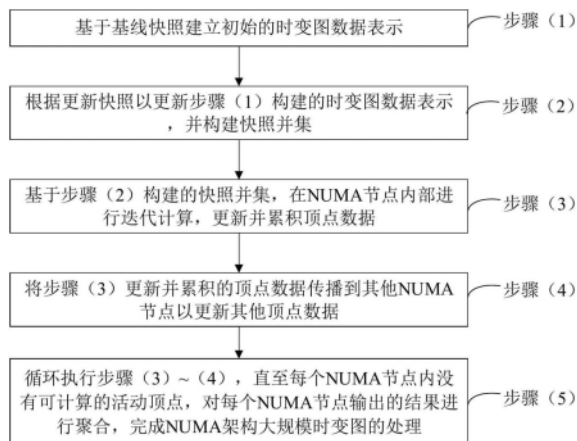
权利要求书2页 说明书7页 附图3页

## (54) 发明名称

一种数据延迟发送的NUMA架构时变图处理方法与装置

## (57) 摘要

本发明公开了一种数据延迟发送的NUMA架构时变图处理方法与装置, 首选基于基线快照建立初始的时变图数据表示; 根据更新快照以更新时变图数据表示, 并构建快照并集; 基于快照并集, 在NUMA节点内部进行迭代计算, 更新并累积顶点数据; 将累积的顶点数据传播到其他NUMA节点以更新其他顶点数据; 循环上述步骤, 直至每个NUMA节点内没有可计算的活动顶点, 对每个NUMA节点输出的结果进行聚合, 完成NUMA架构时变图的处理。本发明关注了服务器的NUMA结构特征, 实现了数据的合理分配以及数据包的灵活传输, 降低了NUMA节点间的通信频率, 提高计算资源的利用率, 使时变图的计算效率得到显著提高。



1. 一种数据延迟发送的NUMA架构时变图处理方法,其特征在于,所述方法包括以下步骤:

- (1) 基于基线快照建立初始的时变图数据表示;
- (2) 根据更新快照以更新步骤(1)构建的时变图数据表示,并构建快照并集;
- (3) 基于步骤(2)构建的快照并集,在NUMA节点内部进行迭代计算,更新并累积顶点数据;
- (4) 将步骤(3)更新并累积的顶点数据传播到其他NUMA节点以更新其他顶点数据;
- (5) 循环执行步骤(3)~(4),直至每个NUMA节点内没有可计算的活动顶点,对每个NUMA节点输出的结果进行聚合,完成NUMA架构时变图的处理。

2. 根据权利要求1所述的数据延迟发送的NUMA架构时变图处理方法,其特征在于,所述步骤(1)具体包括以下子步骤:

(1.1) 创建一个线程池,所述线程池的容量为服务器中的CPU数量,并将线程池中的各个线程平均分配并绑定到对应的NUMA节点中;

(1.2) 把图分成互不相交的图分区,所述图分区的数量为服务器中的NUMA节点数量;读取基线快照文件,根据依次读取到的边的源顶点ID,通过对其进行求余操作计算出此源顶点所属的图分区,并将此边添加到这个图分区对应的NUMA节点中任务数量较少的线程对应的任务队列中;

(1.3) 读取完基线快照文件后,线程池中的所有线程开始执行自己任务队列中的任务,在各个NUMA节点中构建对应的图分区,得到初始的时变图数据表示。

3. 根据权利要求1所述的数据延迟发送的NUMA架构时变图处理方法,其特征在于,所述步骤(2)具体包括以下子步骤:

(2.1) 读取后续的更新快照,根据依次读取到的边的源顶点ID,通过对其进行求余操作计算出此顶点所属的图分区,并将此边添加到这个图分区对应的NUMA节点中任务数量较少的线程的任务队列;

(2.2) 循环步骤(2.1),读取全部的更新快照,启动线程池中的线程,每条线程执行自己任务队列中的任务以更新各个图分区以构建快照并集。

4. 根据权利要求1或3所述的数据延迟发送的NUMA架构时变图处理方法,其特征在于,所述快照并集中包含了所有在时变图多快照中出现过的顶点和边,并且每个顶点或边都只存储一次,不会重复存储。

5. 根据权利要求1所述的数据延迟发送的NUMA架构时变图处理方法,其特征在于,所述步骤(3)具体为:

基于步骤(2)构建的快照并集,在每个NUMA节点内迭代计算的过程中,每个NUMA节点使用计数器统计下一次在各自NUMA节点内要迭代计算的活动顶点的数量,并且在当前活动的顶点在计算完后更新并累积顶点数据。

6. 根据权利要求2所述的数据延迟发送的NUMA架构时变图处理方法,其特征在于,所述步骤(4)包括同一分区内的顶点数据传播过程和不同分区的顶点数据传播过程;

其中,同一分区内的顶点数据传播过程包括:顶点在一轮计算结束后将更新值通过边传播给相邻顶点,相邻顶点接收到更新值后进行更新;

不同分区的顶点数据传播过程包括:将每个分区计算结束后的更新值存储至消息数组

中,延迟分区间的消息传播,降低分区间的通信频率,将当前NUMA节点所持有的消息数组打包,发送到其他NUMA节点中以更新其他图分区内的顶点数据,且启动其他NUMA节点进行新一轮的迭代计算。

7. 根据权利要求6所述的数据延迟发送的NUMA架构时变图处理方法,其特征在于,不同分区的顶点数据传播过程具体包括以下步骤:

在每个NUMA节点内迭代计算活动顶点的数量的过程中都会统计下一次该NUMA节点内的活动顶点数量;若在统计过程中,当前NUMA节点内的活动顶点数量大于下一次该NUMA节点内的活动顶点数量时,就把当前NUMA节点内的传播阈值设置为下一次该NUMA节点内的活动顶点数量;并且统计过程中,下一次该NUMA节点内的活动顶点数量还需要与传播阈值进行比较;迭代计算结束后,将当前NUMA节点所持有的累积消息数组打包发送到其他NUMA节点中;只有当下次活动顶点数量小于或等于传播阈值时,才进行NUMA节点间的消息传播;

其他NUMA节点接收到数据包后,开始更新分区内的顶点数据,并且重新统计下一次活动的顶点数量。

8. 根据权利要求7所述的数据延迟发送的NUMA架构时变图处理方法,其特征在于,不同分区的顶点数据传播过程还包括:

将活动顶点数量大于传播阈值的NUMA节点作为频繁发送数据包的NUMA节点,对频繁发送数据包的NUMA节点的传播阈值施加惩罚机制,所述惩罚机制用于修改传播阈值以降低数据包的发送频率。

9. 一种数据延迟发送的NUMA架构时变图处理装置,包括存储器和处理器,其特征在于,所述存储器与所述处理器耦接;其中,所述存储器用于存储程序数据,所述处理器用于执行所述程序数据以实现上述权利要求1-8任一项所述的数据延迟发送的NUMA架构时变图处理方法。

10. 一种计算机可读存储介质,其上存储有计算机程序,其特征在于,所述程序被处理器执行时实现如权利要求1-8中任一所述的数据延迟发送的NUMA架构时变图处理方法。

## 一种数据延迟发送的NUMA架构时变图处理方法与装置

### 技术领域

[0001] 本发明属于时变图处理技术领域,具体涉及一种数据延迟发送的NUMA架构时变图处理方法与装置。

### 背景技术

[0002] 图作为一种有效描述大数据的数据结构,在互联网分析、社交网络分析、推荐网络分析等领域都起着巨大的作用,现实中许多复杂的计算问题都可以转化为一个基于图的问题,并且使用图的相关算法就可以轻松解决。但是,现实世界无时无刻都在发生改变,所以单纯地只处理静态图并不能很好地满足社会的需要,还需要做到能够对时变图进行快速的分析。所谓时变图,也称时序图,由多张快照组成,并且这些快照在时间上是连续的,每一张快照都表示为原始图在演变过程中某个时刻的图结构状态。通过快速分析对时变图快照间的内在联系,就能够帮助人们预测现实世界未来的发展趋势,为电商、社交等不同领域提供决策支持。

[0003] NUMA(Non-uniform memory access)架构,是指计算机的系统架构由多个节点组成,而每个节点内部拥有多个CPU,节点内部的CPU使用共有的内存控制器,节点之间通过互联模块进行连接和信息交互。因此节点内的所有内存对于本节点所有的CPU都是等同的,但对于其他节点中的所有CPU都不同。也就是说,每个CPU可以访问整个系统内存,但是访问本地节点的内存速度最快,访问非本地节点的内存速度稍慢,即CPU访问内存的速度与节点的距离有关。尽管这种特性可能会对图分析的效率产生重大影响,但是现有的图处理系统在很大程度上是NUMA无关的,如graphchi、Ligra、X-stream等,它们专注于其他方面,如改进外存访问、支持复杂的任务调度程序、减少边的随机访问等。

[0004] 尽管也存在少量系统也有关注NUMA架构,如polymer,HyGN等。polymer改善了节点的访问方式,将大量的远程访问转成本地访问,将大量的随机访问转化成顺序访问,优化了数据访问的局部性,提高了计算效率;HyGN利用同步和异步两种处理模式的特点,将在同一个图计算任务中结合同步和异步处理,能够根据算法、执行阶段和图拓扑的不同,系统自行根据情况切换计算模式,支持复杂的任务调度程序,提高计算效率。但是这些系统只关注静态图的计算,而无法支持时变图的计算。要计算处理时变图,它们需要在多快照上分别执行静态图算法,因此算法执行时间往往与快照数量成正比,导致算法执行时间过长。

[0005] 针对上述大多数图处理系统都忽视了NUMA架构的影响和缺乏针对在NUMA架构下的时变图的计算方法的问题,亟需一种基于NUMA架构的大规模时变图处理方法。

### 发明内容

[0006] 针对现有技术不足,本发明的目的在于提供一种数据延迟发送的NUMA架构大规模时变图处理方法、装置。

[0007] 为实现上述目的,本发明采用的技术方案是:本发明实施例第一方面提供了一种数据延迟发送的NUMA架构时变图处理方法,所述方法包括以下步骤:

- [0008] (1) 基于基线快照建立初始的时变图数据表示；
- [0009] (2) 根据更新快照以更新步骤(1)构建的时变图数据表示,并构建快照并集；
- [0010] (3) 基于步骤(2)构建的快照并集,在NUMA节点内部进行迭代计算,更新并累积顶点数据；
- [0011] (4) 将步骤(3)更新并累积的顶点数据传播到其他NUMA节点以更新其他顶点数据；
- [0012] (5) 循环执行步骤(3)~(4),直至每个NUMA节点内没有可计算的活动顶点,对每个NUMA节点输出的结果进行聚合,完成NUMA架构时变图的处理。
- [0013] 进一步地,所述步骤(1)具体包括以下子步骤：
- [0014] (1.1) 创建一个线程池,所述线程池的容量为服务器中的CPU数量,并将线程池中的各个线程平均分配并绑定到对应的NUMA节点中；
- [0015] (1.2) 把图分成互不相交的图分区,所述图分区的数量为服务器中的NUMA节点数量;读取基线快照文件,根据依次读取到的边的源顶点ID,通过对其进行求余操作计算出此源顶点所属的图分区,并将此边添加到这个图分区对应的NUMA节点中任务数量较少的线程对应的任务队列中；
- [0016] (1.3) 读取完基线快照文件后,线程池中的所有线程开始执行自己任务队列中的任务,在各个NUMA节点中构建对应的图分区,得到初始的时变图数据表示。
- [0017] 进一步地,所述步骤(2)具体包括以下子步骤：
- [0018] (2.1) 读取后续的更新快照,根据依次读取到的边的源顶点ID,通过对其进行求余操作计算出此顶点所属的图分区,并将此边添加到这个图分区对应的NUMA节点中任务数量较少的线程的任务队列；
- [0019] (2.2) 循环步骤(2.1),读取全部的更新快照,启动线程池中的线程,每条线程执行自己任务队列中的任务以更新各个图分区以构建快照并集。
- [0020] 进一步地,所述快照并集中包含了所有在时变图多快照中出现过的顶点和边,并且每个顶点或边都只存储一次,不会重复存储。
- [0021] 进一步地,所述步骤(3)具体为：
- [0022] 基于步骤(2)构建的快照并集,在每个NUMA节点内迭代计算的过程中,每个NUMA节点使用计数器统计下一次在各自NUMA节点内要迭代计算的活动顶点的数量,并且在当前活动的顶点在计算完后更新并累积顶点数据。
- [0023] 进一步地,所述步骤(4)包括同一分区内的顶点数据传播过程和不同分区的顶点数据传播过程；
- [0024] 其中,同一分区内的顶点数据传播过程包括:顶点在一轮计算结束后将更新值通过边传播给相邻顶点,相邻顶点接收到更新值后进行更新；
- [0025] 不同分区的顶点数据传播过程包括:将每个分区计算结束后的更新值存储至消息数组中,延迟分区间的消息传播,降低分区间的通信频率,将当前NUMA节点所持有的消息数组打包,发送到其他NUMA节点中以更新其他图分区内的顶点数据,且启动其他NUMA节点进行新一轮的迭代计算。
- [0026] 进一步地,不同分区的顶点数据传播过程具体包括以下步骤：
- [0027] 在每个NUMA节点内迭代计算活动顶点的数量的过程中都会统计下一次该NUMA节点内的活动顶点数量;若在统计过程中,当前NUMA节点内的活动顶点数量大于下一次该

NUMA节点内的活动顶点数量时,就把当前NUMA节点内的传播阈值设置为下一次该NUMA节点内的活动顶点数量;并且统计过程中,下一次该NUMA节点内的活动顶点数量还需要与传播阈值进行比较;迭代计算结束后,将当前NUMA节点所持有的累积消息数组打包发送到其他NUMA节点中;只有当下次活动顶点数量小于或等于传播阈值时,才进行NUMA节点间的消息传播;

[0028] 其他NUMA节点接收到数据包后,开始更新分区内的顶点数据,并且重新统计下一次活动的顶点数量。

[0029] 进一步地,不同分区的顶点数据传播过程还包括:

[0030] 将活动顶点数量大于传播阈值的NUMA节点作为频繁发送数据包的NUMA节点,对频繁发送数据包的NUMA节点的传播阈值施加惩罚机制,所述惩罚机制用于修改传播阈值以降低数据包的发送频率。

[0031] 本发明实施例第二方面提供了一种数据延迟发送的NUMA架构时变图处理装置,包括存储器和处理器,所述存储器与所述处理器耦接;其中,所述存储器用于存储程序数据,所述处理器用于执行所述程序数据以实现上述的数据延迟发送的NUMA架构时变图处理方法。

[0032] 本发明实施例第三方面提供了一种计算机可读存储介质,其上存储有计算机程序,所述程序被处理器执行时实现上述的数据延迟发送的NUMA架构时变图处理方法。

[0033] 与现有技术相比,本发明的有益效果为:

[0034] 本发明提出的一种数据延迟发送的NUMA架构时变图处理方法着重关注了服务器的NUMA结构特征,通过设置惩罚机制实现了数据的合理分配以及数据包的灵活传输,降低了NUMA节点间的通信频率,提高计算资源的利用率,其实现方法简便,手段灵活,使时变图算法计算效率得到显著提高。

## 附图说明

[0035] 为了更清楚地说明本发明实施例中的技术方案,下面将对实施例描述中所需要使用的附图作简单地介绍,显而易见地,下面描述中的附图仅仅是本发明的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动性的前提下,还可以根据这些附图获得其他的附图。

[0036] 图1为本发明提供的一种数据延迟发送的NUMA架构大规模时变图处理方法的流程图;

[0037] 图2为本发明提供的一种数据延迟发送的NUMA架构时变图处理系统的架构图;

[0038] 图3为本发明实施例提供的时变载入子系统的示意图;

[0039] 图4为本发明实施例提供的时变计算子系统的示意图;

[0040] 图5为本发明提供的一种数据延迟发送的NUMA架构时变图处理装置的示意图。

## 具体实施方式

[0041] 这里将详细地对示例性实施例进行说明,其示例表示在附图中。下面的描述涉及附图时,除非另有表示,不同附图中的相同数字表示相同或相似的要素。以下示例性实施例中所描述的实施方式并不代表与本发明相一致的所有实施方式。相反,它们仅是与如所附

权利要求书中所详述的、本发明的一些方面相一致的装置和方法的例子。

[0042] 在本发明使用的术语是仅仅出于描述特定实施例的目的，而非旨在限制本发明。在本发明和所附权利要求书中所使用的单数形式的“一种”、“所述”和“该”也旨在包括多数形式，除非上下文清楚地表示其他含义。还应当理解，本文中使用的术语“和/或”是指并包含一个或多个相关联的列出项目的任何或所有可能组合。

[0043] 应当理解，尽管在本发明可能采用术语第一、第二、第三等来描述各种信息，但这些信息不应限于这些术语。这些术语仅用来将同一类型的信息彼此区分开。例如，在不脱离本发明范围的情况下，第一信息也可以被称为第二信息，类似地，第二信息也可以被称为第一信息。取决于语境，如在此所使用的词语“如果”可以被解释成为“在……时”或“当……时”或“响应于确定”。

[0044] 下面结合附图，对本发明进行详细说明。在不冲突的情况下，下述的实施例及实施方式中的特征可以相互组合。

[0045] 如图1和图2所示，本发明实施例提出了一种数据延迟发送的NUMA架构时变图处理方法，该方法及系统不仅关注服务器的内部结构，提高计算资源的利用率，还有利于减少NUMA节点间的通信频率，提高时变图的计算效率；所述方法包括以下步骤：

[0046] (1) 基于基线快照建立初始的时变图数据表示；

[0047] 所述步骤(1)具体包括以下子步骤：

[0048] (1.1) 创建一个线程池，所述线程池的容量为服务器中的CPU数量，并将线程池中的各个线程平均分配并绑定到对应的NUMA节点中；

[0049] (1.2) 把图分成互不相交的图分区，所述图分区的数量为服务器中的NUMA节点数量。读取基线快照文件，根据依次读取到的边的源顶点ID，通过对其进行求余操作计算出此源顶点所属的图分区，并将此边添加到这个图分区对应的NUMA节点中任务数量较少的线程对应的任务队列中；

[0050] (1.3) 读取完基线快照文件后，线程池中的所有线程开始执行自己任务队列中的任务，在各个NUMA节点中构建对应的图分区，得到初始的时变图数据表示，在此过程中能够对顶点或边加入用户自定义的数据。

[0051] (2) 根据更新快照以更新步骤(1)构建的时变图数据表示，并构建快照并集。

[0052] 所述步骤(2)具体包括以下子步骤：

[0053] (2.1) 读取后续的更新快照，根据依次读取到的边的源顶点ID，通过其进行求余操作计算出此源顶点所属的图分区，并将此边添加到这个图分区对应的NUMA节点中任务数量较少的线程的任务队列；

[0054] (2.2) 循环步骤(2.1)，读取全部的更新快照，启动线程池中的线程，每条线程执行自己任务队列中的任务以更新各个图分区以构建快照并集。在此过程中能够对顶点或边加入或修改用户自定义的数据。

[0055] 如图3所示，时变图由若干快照组成，每张快照代表时变图在某个时间点的一个状态，快照并集中包含了所有在时变图多快照中出现过的顶点和边，并且每个对象，即每个顶点或边都只存储一次，不会重复存储。

[0056] (3) 基于步骤(2)构建的快照并集，在每个NUMA节点内进行迭代计算，更新并累积顶点数据；

[0057] 基于步骤(2)构建的快照并集,在每个NUMA节点内迭代计算的过程中,每个NUMA节点使用各自独立的计数器统计下一次参与迭代计算的活动顶点的数量,并且在当前活动的顶点计算完成后,得到更新的顶点数据。

[0058] (4)将步骤(3)更新并累积的顶点数据传播到其他NUMA节点以更新其他顶点数据。

[0059] 由于在步骤(1)中把图分成多个互不相交的图分区,因此,顶点数据可以分为分区内的顶点数据以及分区外的顶点数据。在同一分区内的顶点,顶点在一轮计算结束后,能够立刻把更新值通过边传播给相邻顶点,相邻顶点接收到更新值后会被立刻更新。对于在不同分区的顶点数据,尽管它们之间也存在连接关系,但是每个分区计算结束后的更新值不会被立刻传播给其他分区的顶点,而是把更新值存储到消息数组中,延迟分区间的消息传播,降低分区间的通信频率。分区间的消息传播需要通过自适应的数据包传播算法把当前NUMA节点所持有的消息数组打包,并将消息数据包发送到其他NUMA节点中以更新其他图分区内的顶点数据且启动其他NUMA节点进行新一轮的迭代计算。

[0060] 在本实例中,通过自适应的数据包传播算法将步骤(3)累积的顶点数据发送到传播到其他NUMA节点以更新其他分区的顶点数据;具体步骤如下:

[0061] (4.1)在每个NUMA节点内迭代计算活动顶点的数量的过程中都会统计下一次该NUMA节点内的活动顶点数量,若在统计过程中,当前的活动顶点数量大于下一次该NUMA节点内的活动顶点数量时,就把该NUMA节点内的传播阈值设置为下一次该NUMA节点内的活动顶点数量,后面统计过程中下一次该NUMA节点内的活动顶点数量还需要与阈值进行比较。此次迭代执行结束后,将累积下来的消息打包发送到其他NUMA节点中;只有当下次活动顶点数量小于或等于阈值时,才进行NUMA节点间的消息传播;

[0062] (4.2)其他NUMA节点接收到数据包后,开始更新分区内的顶点数据,并且重新统计下一次活动的顶点数量;

[0063] 特别地,对于频繁发送数据包的NUMA节点,即活动顶点的数量经常超过阈值,就会对该NUMA节点的阈值设置施加惩罚机制,在本实例中设置惩罚机制为频繁发送数据包的NUMA节点的自适应数据包传播算法中的传播阈值的一半,使其降低数据包的发送频率,减少数据包的频繁发送。

[0064] (5)循环执行步骤(3)~(4),直至每个NUMA节点都已经计算收敛(即没有可计算的活动顶点),对每个NUMA节点输出的收敛结果进行聚合,完成NUMA架构时变图的处理。

[0065] 相应地,本发明提出了一种数据延迟发送的NUMA架构时变图处理系统,用于实现上述的数据延迟发送的NUMA架构时变图处理方法,所述系统包括时变图载入子系统和时变图计算子系统。所述时变图载入子系统用于将图的拓扑结构、用户定义的数据以及运行时状态分配到服务器中的各个NUMA节点中;所述时变图计算子系统用于在时变图计算时,以控制NUMA节点间的通信频率并且在触发惩罚机制时使NUMA节点间通过数据包的形式来传输消息;最后将各个NUMA节点中已经收敛的计算结果聚合起来输出。

[0066] 实施例1:基于上述的一种数据延迟发送的NUMA架构时变图处理系统,本实施例1进行详细说明,如图4所示,在本实例中,假定该台计算机有2个NUMA节点,因此,第一基线快照就被分成两个图分区,其中第一分区包含第一顶点V1和第二顶点V2,第一分区包含第三顶点V3和第四顶点V4。然后读取更新第二基线快照和第三基线快照,把两张快照的边依照源顶点id所在的分区添加对应的分区上,构建出快照并集。然后在快照并集上进行迭代计



算,第一顶点V1和第二顶点V2在第一NUMA节点中各自计算,第三顶点V3和第四顶点V4在第二NUMA节点中各自计算,在计算过程中,第一NUMA节点和第二NUMA节点会统计T1时刻和T2时刻的活动顶点数量,第一NUMA节点发现T1和T2时刻的活动顶点数量都是2,而第二NUMA节点在T1和T2时刻的活动顶点数量分别是2和0。由于第一NUMA节点在T1时刻和T2时刻的活动顶点数量相等,所以不会发送消息给第二NUMA节点,因此在T1时刻,在第一NUMA节点中,第一顶点V1和第二顶点V2会将第三顶点V3的更新值存到消息数组中,第二顶点V2也会对第一顶点V1进行直接更新,而在第二NUMA节点中,第三顶点V3直接更新第四顶点V4,由于在这个时刻没有要传给第一NUMA节点的消息,因此并不会发送数据包;接着到T2时刻,第一顶点V1和第二顶点V2会将第四顶点V4的更新值存到消息数组中,并且发现T3时刻的时候活动顶点的数量为0,就在T2时刻结束后,立刻把存储了T1和T2时刻更新值的消息数组打包,发送到第二NUMA节点更新顶点第三顶点V3和第四顶点V4。这时第二NUMA节点重新统计活动顶点数量,发现T3时刻的活动顶点是2,而T4时刻的活动顶点数量为0,因此在T3时刻结束后,立刻把存储了第三顶点V3对第二顶点V2的更新值和第四顶点V4对第一顶点V1的更新值的消息数组打包,发送给第一NUMA节点更新第一顶点V1和第二顶点V2。

[0067] 与前述数据延迟发送的NUMA架构时变图处理方法的实施例相对应,本发明还提供了数据延迟发送的NUMA架构时变图处理装置的实施例。

[0068] 参见图5,本发明实施例提供的一种数据延迟发送的NUMA架构时变图处理装置,包括一个或多个处理器,用于实现上述实施例中的数据延迟发送的NUMA架构时变图处理方法。

[0069] 本发明数据延迟发送的NUMA架构时变图处理装置的实施例可以应用在任意具备数据处理能力的设备上,该任意具备数据处理能力的设备可以为诸如计算机等设备或装置。装置实施例可以通过软件实现,也可以通过硬件或者软硬件结合的方式实现。以软件实现为例,作为一个逻辑意义上的装置,是通过其所在任意具备数据处理能力的设备的处理器将非易失性存储器中对应的计算机程序指令读取到内存中运行形成的。从硬件层面而言,如图5所示,为本发明数据延迟发送的NUMA架构时变图处理装置所在任意具备数据处理能力的设备的一种硬件结构图,除了图5所示的处理器、内存、网络接口、以及非易失性存储器之外,实施例中装置所在的任意具备数据处理能力的设备通常根据该任意具备数据处理能力的设备的实际功能,还可以包括其他硬件,对此不再赘述。

[0070] 上述装置中各个单元的功能和作用的实现过程具体详见上述方法中对应步骤的实现过程,在此不再赘述。

[0071] 对于装置实施例而言,由于其基本对应于方法实施例,所以相关之处参见方法实施例的部分说明即可。以上所描述的装置实施例仅仅是示意性的,其中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模块来实现本发明方案的目的。本领域普通技术人员在不付出创造性劳动的情况下,即可以理解并实施。

[0072] 本发明实施例还提供一种计算机可读存储介质,其上存储有程序,该程序被处理器执行时,实现上述实施例中的数据延迟发送的NUMA架构时变图处理方法。

[0073] 所述计算机可读存储介质可以是前述任一实施例所述的任意具备数据处理能力

的设备的内部存储单元,例如硬盘或内存。所述计算机可读存储介质也可以是任意具备数据处理能力的设备,例如所述设备上配备的插接式硬盘、智能存储卡(Smart Media Card, SMC)、SD卡、闪存卡(Flash Card)等。进一步的,所述计算机可读存储介质还可以既包括任意具备数据处理能力的设备的内部存储单元也包括外部存储设备。所述计算机可读存储介质用于存储所述计算机程序以及所述任意具备数据处理能力的设备所需的其他程序和数据,还可以用于暂时地存储已经输出或者将要输出的数据。

[0074] 本领域技术人员在考虑说明书及实践这里公开的内容后,将容易想到本申请的其它实施方案。本申请旨在涵盖本申请的任何变型、用途或者适应性变化,这些变型、用途或者适应性变化遵循本申请的一般性原理并包括本申请未公开的本技术领域中的公知常识或惯用技术手段。说明书和实施例仅被视为示例性的。

[0075] 应当理解的是,本申请并不局限于上面已经描述并在附图中示出的精确结构,并且可以在不脱离其范围进行各种修改和改变。

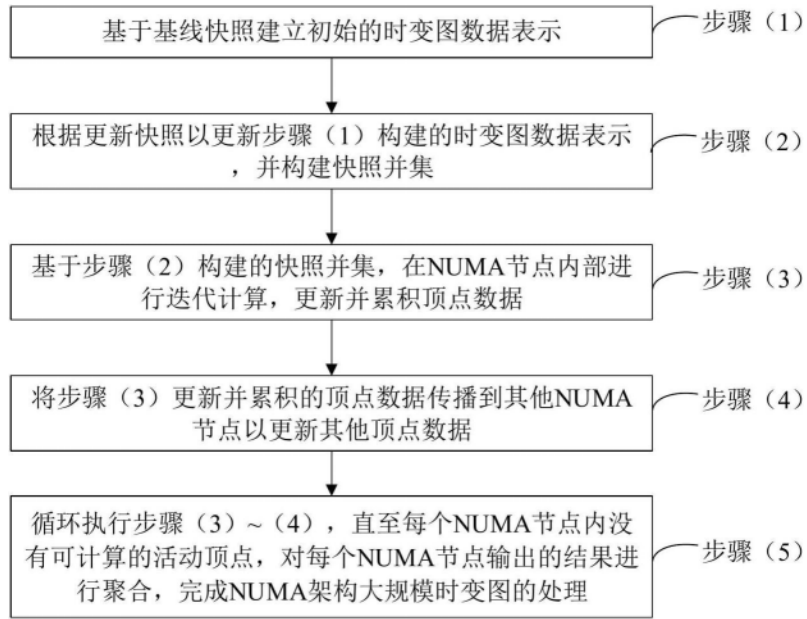


图 1

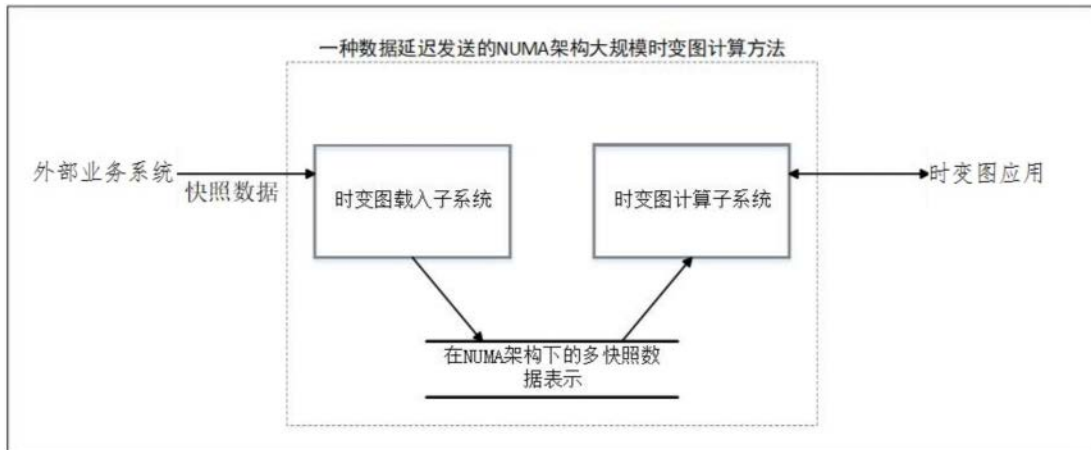


图 2

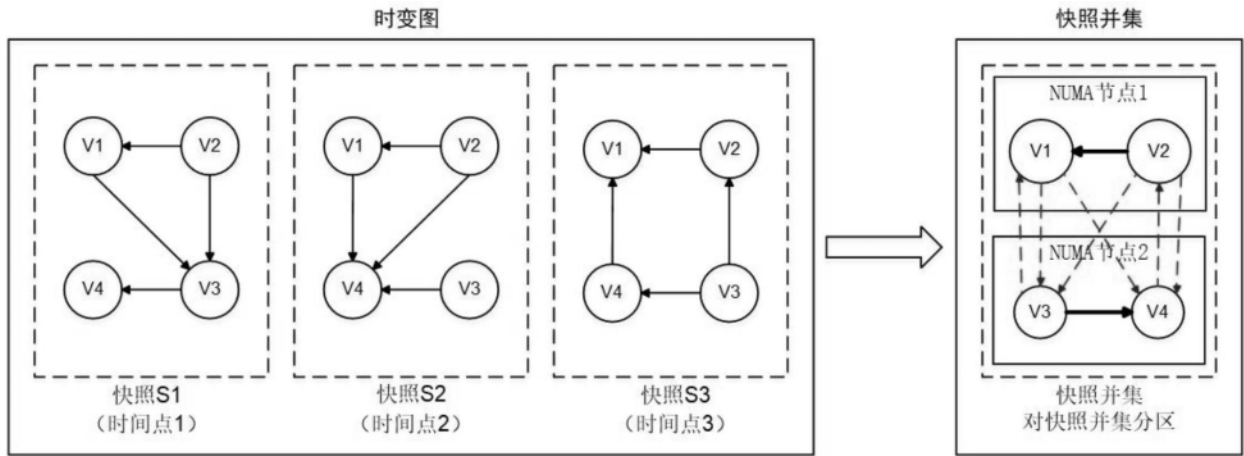


图 3

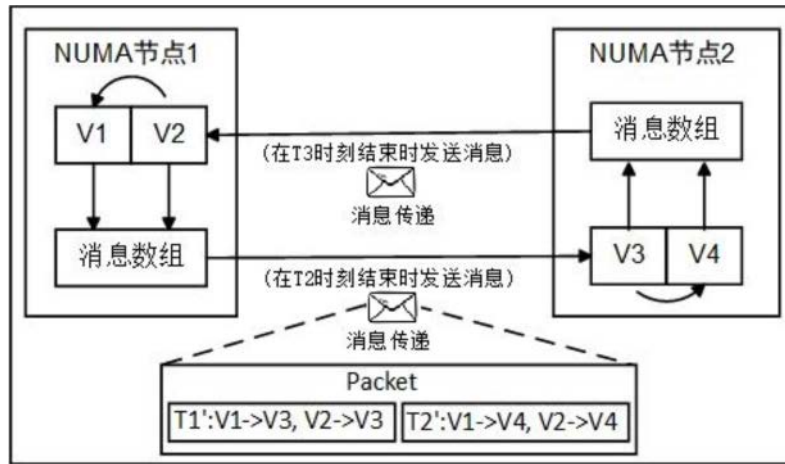


图 4

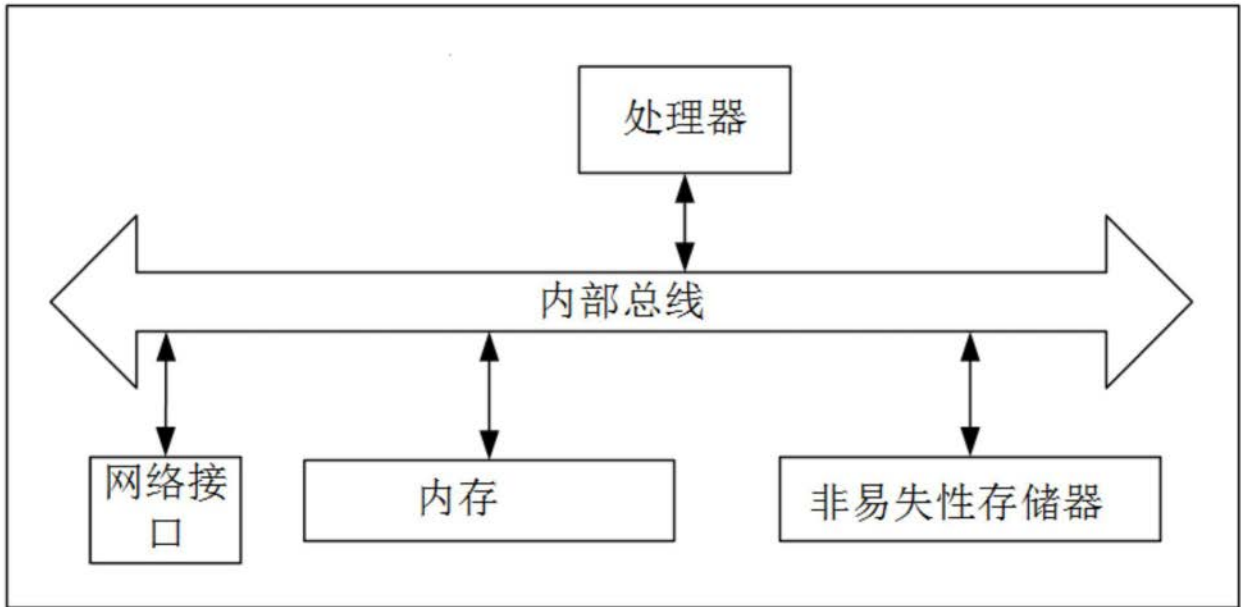


图 5