(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 102722417 A
(43)申请公布日 2012.10.10

(21)申请号 201210185787.4

(22)申请日 2012.06.07

(71)申请人 腾讯科技(深圳)有限公司
地址 518044 广东省深圳市福田区振兴路赛格科技园2栋东403室

(72)发明人 董涛 付立群 李瑞 金海峰

(74)专利代理机构 北京德琦知识产权代理有限公司 11018
代理人 张驰 宋志强

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

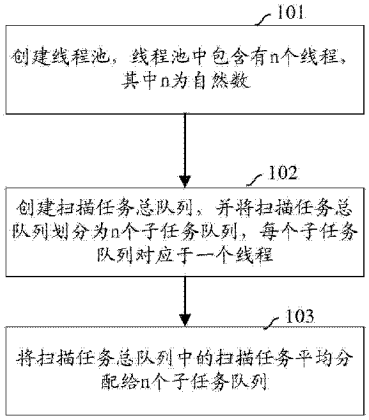
权利要求书 3 页 说明书 7 页 附图 2 页

(54)发明名称

一种扫描任务的分配方法和装置

(57)摘要

本发明实施方式提出一种扫描任务的分配方法和装置。方法包括:创建线程池,所述线程池包含有n个线程,其中n为自然数;创建扫描任务总队列,并将扫描任务总队列划分为n个子任务队列,每个子任务队列对应于一个线程;将所述扫描任务总队列中的扫描任务平均分配给所述n个子任务队列。本发明实施方式通过应用线程池技术来平均分配扫描任务,避免了多线程访问同一子任务队列时的线程同步开销,提高了扫描效率。



CN 102722417 A

1. 一种扫描任务的分配方法，其特征在于，包括：

创建线程池，所述线程池包含有 n 个线程，其中 n 为自然数；

创建扫描任务总队列，并将扫描任务总队列划分为 n 个子任务队列，每个子任务队列对应于一个线程；

将所述扫描任务总队列中的扫描任务平均分配给所述 n 个子任务队列。

2. 根据权利要求 1 所述的扫描任务的分配方法，其特征在于，所述创建线程池步骤包括：创建线程池，其中线程池中线程数为中央处理单元（CPU）的逻辑内核数。

3. 根据权利要求 1 所述的扫描任务的分配方法，其特征在于，该方法还包括：线程池中的线程从对应的子任务队列中获取扫描任务并执行获取的扫描任务。

4. 根据权利要求 1 所述的扫描任务的分配方法，其特征在于，所述扫描任务包括：垃圾文件扫描任务、临时文件扫描任务、历史访问痕迹扫描任务或病毒扫描任务。

5. 根据权利要求 1 所述的扫描任务的分配方法，其特征在于，所述将扫描任务总队列中的扫描任务平均分配给所述 n 个子任务队列步骤之前还包括：将所有扫描任务放入扫描任务总队列。

6. 根据权利要求 5 所述的扫描任务的分配方法，其特征在于，在所述将所有扫描任务放入扫描任务总队列步骤之后还包括：对扫描任务总队列中的各个扫描任务按照各自的扫描任务历史执行平均耗时进行从大到小排序。

7. 根据权利要求 6 所述的扫描任务的分配方法，其特征在于，所述将扫描任务总队列中的扫描任务平均分配给所述 n 个子任务队列包括：

当扫描任务总队列中的扫描任务总数小于或等于子任务队列总数 n 时，将所述扫描任务总队列中的扫描任务逐一分配给所述子任务队列。

8. 根据权利要求 6 所述的扫描任务的分配方法，其特征在于，所述将扫描任务总队列中的扫描任务平均分配给所述 n 个子任务队列包括：

任务分配步骤：当扫描任务总队列中的扫描任务总数大于子任务队列总数 n 时，从排序后的扫描任务总队列中依次取出 n 个扫描任务，放入所述 n 个子任务队列中；

排序步骤：对所述 n 个子任务队列按照当前所含任务执行耗时进行从小到大排序，并将扫描任务总队列中剩余的扫描任务依次平均分配给所述排序后的 n 个子任务队列。

9. 根据权利要求 8 所述的扫描任务的分配方法，其特征在于，所述将扫描任务总队列中剩余的扫描任务依次平均分配给排序后的 n 个子任务队列包括：

判断步骤：判断扫描任务总队列中剩余的扫描任务总数是否依然大于子任务队列总数 n；

如果大于，则重复所述任务分配步骤、排序步骤和判断步骤循环执行，直到扫描任务总队列中的扫描任务总数小于子任务队列总数；

如果不大于，则将扫描任务总队列中剩余的扫描任务依次分配给所述排序后的子任务队列。

10. 根据权利要求 6 至 9 中任意一项所述的扫描任务的分配方法，其特征在于，所述对各个扫描任务按照各自的扫描任务历史执行平均耗时进行从大到小排序包括：

对于每个扫描任务，分别计算其扫描任务历史执行平均耗时 TaverageTime，其中扫描任务的历史执行次数 k，本次扫描耗时为 currentTime，本次扫描之前的历史执行平均耗时

为 averageTime ;其中 :

TaverageTime=(averageTime*k+currentTime)/(k+1) ;

对于各个扫描任务,按照各自计算的 TerageTime 进行从大到小排序。

11. 根据权利要求 6 至 9 中任意一项所述的扫描任务的分配方法,其特征在于,所述对各个扫描任务按照扫描任务历史执行平均耗时进行从大到小排序包括 :

对于各个扫描任务,按照各自的上次扫描耗时进行从大到小排序。

12. 一种扫描任务的分配装置,其特征在于,该装置包括线程池创建单元和扫描任务总队列分配单元,其中 :

线程池创建单元,用于创建线程池,所述线程池包含有 n 个线程,其中 n 为自然数 ;

扫描任务总队列分配单元,用于创建扫描任务总队列,将扫描任务总队列划分为 n 个子任务队列,每个子任务队列对应于一个线程,并将所述扫描任务总队列中的扫描任务平均分配给所述 n 个子任务队列。

13. 根据权利要求 12 的扫描任务的分配装置,其特征在于,

线程池创建单元,用于创建线程池,其中线程池中线程数为中央处理单元(CPU)的逻辑内核数。

14. 根据权利要求 12 的扫描任务的分配装置,其特征在于,所述扫描任务包括 :垃圾文件扫描任务、临时文件扫描任务、历史访问痕迹扫描任务或病毒扫描任务。

15. 根据权利要求 12 的扫描任务的分配装置,其特征在于,

扫描任务总队列分配单元,还用于在将所述扫描任务总队列中的扫描任务平均分配给所述 n 个子任务队列之前,将所有扫描任务放入扫描任务总队列。

16. 根据权利要求 15 的扫描任务的分配装置,其特征在于,

扫描任务总队列分配单元,还用于在将所有扫描任务放入扫描任务总队列之后,对扫描任务总队列中的各个扫描任务按照各自的扫描任务历史执行平均耗时进行从大到小排序。

17. 根据权利要求 16 的扫描任务的分配装置,其特征在于,

扫描任务总队列分配单元,用于当扫描任务总队列中的扫描任务总数小于或等于子任务队列总数 n 时,将所述扫描任务总队列中的扫描任务逐一分配给所述子任务队列。

18. 根据权利要求 16 的扫描任务的分配装置,其特征在于,

扫描任务总队列分配单元,用于执行 :

任务分配步骤 :当扫描任务总队列中的扫描任务总数大于子任务队列总数 n 时,从排序后的扫描任务总队列中依次取出 n 个扫描任务,放入 n 个子任务队列中 ;

排序步骤 :对所述 n 个子任务队列按照当前所含任务执行耗时进行从小到大排序,并将扫描任务总队列中剩余的扫描任务依次平均分配给所述排序后的 n 个子任务队列。

19. 根据权利要求 18 的扫描任务的分配装置,其特征在于,

扫描任务总队列分配单元,用于执行 :

判断步骤 :判断扫描任务总队列中剩余的扫描任务总数是否依然大于子任务队列总数 n ;

如果大于,则重复所述任务分配步骤、排序步骤和判断步骤循环执行,直到扫描任务总队列中的扫描任务总数小于子任务队列总数 ;

如果不大于，则将扫描任务总队列中剩余的扫描任务依次分配给所述排序后的子任务队列。

20. 根据权利要求 16-19 中任意一项所述的扫描任务的分配装置，其特征在于，

扫描任务总队列分配单元，用于对于每个扫描任务，分别计算其扫描任务历史执行平均耗时 TaverageTime，其中扫描任务的历史执行次数 k，本次扫描耗时为 currentTime，本次扫描之前的历史执行平均耗时为 averageTime ；其中 ：

TaverageTime=(averageTime*k+currentTime)/(k+1) ；

并对于各个扫描任务，按照各自计算的 TaverageTime 进行从大到小排序。

21. 根据权利要求 16-19 中任意一项所述的扫描任务的分配装置，其特征在于，

扫描任务总队列分配单元，用于对于各个扫描任务，按照各自的上次扫描耗时进行从大到小排序。

一种扫描任务的分配方法和装置

技术领域

[0001] 本发明实施方式涉及计算机管理技术领域，更具体地，涉及一种扫描任务的分配方法和装置。

背景技术

[0002] 随着电脑硬件和软件系统的不断升级，当今计算机科学发展趋势向微型化、网络化和智能化等多方面发展。

[0003] 在计算机应用中，通常会涉及到对各种扫描任务的分配处理。目前计算机处理扫描任务的分配方式主要包括单线程方式和多线程方式。在单线程方式中，计算机以单线程的方式逐个对单个扫描点进行扫描处理；在多线程方式中，计算机以多线程的方式同时对多个扫描点进行扫描处理。

[0004] 然而，对于单线程扫描方式，在同一时刻只能进行一个扫描任务，不能充分利用多核中央处理单元（CPU）的优势，因此扫描速度较慢，扫描效率低下。对于多线程扫描方式，虽然采用了多线程的方式，但是并没有做很好的任务分配，这样可能会导致其中某些线程执行了较多或者耗时较长的任务，而其它线程早早结束了任务，实际上没有充分利用多线程的优势，因此扫描效率也并不令人满意。

发明内容

[0005] 本发明实施方式提出一种扫描任务的分配方法，以提高扫描效率。

[0006] 本发明实施方式提出一种扫描任务的分配装置，以提高扫描效率。

[0007] 本发明实施方式的具体方案如下：

[0008] 一种扫描任务的分配方法，包括：

[0009] 创建线程池，所述线程池包含有 n 个线程，其中 n 为自然数；

[0010] 创建扫描任务总队列，并将扫描任务总队列划分为 n 个子任务队列，每个子任务队列对应于一个线程；

[0011] 将所述扫描任务总队列中的扫描任务平均分配给所述 n 个子任务队列。

[0012] 一种扫描任务的分配装置，该装置包括线程池创建单元和扫描任务总队列分配单元，其中：

[0013] 线程池创建单元，用于创建线程池，所述线程池包含有 n 个线程，其中 n 为自然数；

[0014] 扫描任务总队列分配单元，用于创建扫描任务总队列，将扫描任务总队列划分为 n 个子任务队列，每个子任务队列对应于一个线程，并将所述扫描任务总队列中的扫描任务平均分配给所述 n 个子任务队列。

[0015] 从上述技术方案可以看出，在本发明实施方式中，创建线程池，线程池包含有 n 个线程，其中 n 为自然数；创建扫描任务总队列，将扫描任务总队列划分为 n 个子任务队列，每个子任务队列对应于一个线程；将所述扫描任务总队列中的扫描任务平均分配给所述 n 个

子任务队列。由此可见，本发明实施方式通过应用线程池技术来平均分配扫描任务总队列，从而避免了多线程访问同一子任务队列时的线程同步开销，因此提高了扫描效率。

附图说明

[0016] 图 1 为根据本发明实施方式的扫描任务的分配方法流程图；

[0017] 图 2 为根据本发明实施方式的扫描任务的分配方法示范性流程图；

[0018] 图 3 为根据本发明实施方式的对扫描任务按照任务队列进行分配的示范性流程图；

[0019] 图 4 为根据本发明实施方式的扫描任务的分配装置结构图。

具体实施方式

[0020] 为使本发明的目的、技术方案和优点更加清楚，下面结合附图对本发明作进一步的详细描述。

[0021] 图 1 为根据本发明实施方式的扫描任务的分配方法流程图。

[0022] 如图 1 所示，该方法包括：

[0023] 步骤 101：创建线程池，线程池包含有 n 个线程，其中 n 为自然数。

[0024] 在这里，首先创建线程池，其中在线程池中优选具有多个(至少两个)线程。线程池是一种多线程处理形式，处理过程中将任务添加到任务队列，在创建线程后，首先启动线程并由线程具体执行任务队列中的任务。

[0025] 优选地，线程池中线程数与中央处理单元(CPU)的逻辑内核数保持一致，从而充分优化利用计算机的处理资源。

[0026] 步骤 102：创建扫描任务总队列，将扫描任务总队列划分为 n 个子任务队列，每个子任务队列对应于一个线程。

[0027] 在这里，扫描任务总队列中包含有扫描任务，而且可以基于具体应用需要创建多种类的扫描任务。

[0028] 比如，扫描任务具体可以包括：垃圾文件扫描任务、临时文件扫描任务、历史访问痕迹扫描任务或病毒扫描任务，等等。

[0029] 以上虽然通过罗列的方式详细列出了扫描任务的一些示范性具体种类，本领域技术人员可以意识到，这种罗列仅仅是示范性的，并不用于对本发明实施方式的保护范围进行限定。

[0030] 步骤 103：将所述扫描任务总队列中的扫描任务平均分配给所述 n 个子任务队列。

[0031] 创建完扫描任务总队列之后，需要将扫描任务总队列中的扫描任务平均分配给线程池中的线程，从而缩短扫描处理时间，优化扫描效率。

[0032] 此处涉及平均任务分配问题，为了便于说明，首先假定每个线程执行同样的扫描任务时，其耗时相同。在这种情况下，只需要关注每个扫描任务耗时即可，只要保证分配给每个线程的扫描任务，其总执行耗时大致相等即可，具体分配算法优选可以基于现有的贪心算法原理(Greedy algorithm)。

[0033] 贪心算法是一种对某些求最优解问题的更简单、更迅速的设计技术。用贪心算法设计算法的特点是一步一步地进行，以当前情况为基础根据某个优化测度作最优选择，

而不考虑各种可能的整体情况，它省去了为找最优解要穷尽所有可能而必须耗费的大量时间。

[0034] 具体地，在本发明实施方式中，首先准备好扫描任务总队列 Q。将所有扫描任务放入扫描任务总队列 Q 中，并且按照扫描任务历史执行平均耗时从大到小进行排序。

[0035] 优选地，判断扫描任务总队列 Q 中扫描任务数与线程池线程总数的大小关系，其中：

[0036] 如果扫描任务总队列 Q 中扫描任务数 m 小于线程池线程总数 n，则从 Q 中取出全部任务（Q 中同时清除掉这些任务），并依次放入子任务队列 q0、q1…q(m-1)。

[0037] 如果扫描任务总队列 Q 中的扫描任务总数 m 大与线程池线程总数 n，则依次执行：

[0038] 任务分配步骤：从排序后的扫描任务总队列中依次取出 n 个扫描任务，放入 n 个子任务队列中；

[0039] 排序步骤：对所述 n 个子任务队列按照当前所含任务执行耗时进行从小到大排序，并将扫描任务总队列中剩余的扫描任务依次平均分配给所述排序后的 n 个子任务队列。

[0040] 具体地，排序步骤中将扫描任务总队列中剩余的扫描任务依次平均分配给排序后的 n 个子任务队列可以包括：

[0041] 判断步骤：判断扫描任务总队列中剩余的扫描任务总数是否依然大于子任务队列总数 n；

[0042] 如果大于，则重复所述任务分配步骤、排序步骤和判断步骤循环执行，直到扫描任务总队列中的扫描任务总数小于子任务队列总数；

[0043] 如果不大于，则将扫描任务总队列中剩余的扫描任务依次分配给所述排序后的子任务队列。

[0044] 实际上，当扫描任务总队列 Q 中扫描任务数大与线程池线程总数时，可以从 Q 中按照排序结果取出 n 个任务，并放入子任务队列 q0、q1…q(n-1) 中，对子任务队列按其当前所含任务总执行耗时从小到达排列；若 Q 中还剩余任务，则返回到判断 Q 中扫描任务总数与线程池线程总数大小的步骤。

[0045] 下面通过举例方式对上述算法进行更加详细的说明。

[0046] 在一个实施方式中，当扫描任务总队列中的扫描任务总数小于或等于线程池中线程总数时，可以将扫描任务总队列中的扫描任务逐一分配给线程池中的线程，其中每个线程处理单一的扫描任务。

[0047] 比如：当扫描任务总队列中的扫描任务总数为 3 时，可以将扫描任务分别命名为扫描任务 1、扫描任务 2 和扫描任务 3。如果线程池中线程总数为 5，可以分别命名为线程 1、线程 2、线程 3、线程 4 和线程 5。假设线程 1 用于处理子任务队列 q0、线程 2 用于处理子任务队列 q1、线程 3 用于处理子任务队列 q2、线程 4 用于处理子任务队列 q5。此时，扫描任务总队列中的扫描任务总数（3）小于线程池中线程总数（5），因此可以将这 3 个扫描任务随机分配给 3 个线程，或者按照命名顺序将扫描任务 1 分配给线程 1（即 q0）、扫描任务 2 分配给线程 2（即 q1），以及将扫描任务 3 分配给线程 3（即 q2），等等。

[0048] 在一个实施方式中，当扫描任务总队列中的扫描任务总数大于线程池中线程总数 n 时，对各个扫描任务按照扫描任务历史执行平均耗时进行从大到小排序；再从排序后的

扫描任务总队列中依次取出 n 个扫描任务，放入 n 个子任务队列中；然后对 n 个子任务队列按照当前所含任务执行耗时进行从小到大排序，并继续判断扫描任务总队列中的扫描任务总数是否依然大于线程池中线程总数，如果依然大于，则从排序后扫描任务总队列中依次取出 n 个扫描任务放入 n 个子任务队列中之后，重复上述流程，直到扫描任务总队列中的扫描任务总数小于线程池中线程总数，然后将扫描任务总队列中剩余的扫描任务依次分配给所述排序后的 n 个子任务队列；再将 n 个子任务队列分配给所述线程池中的线程，其中每个线程处理单一的子任务队列。

[0049] 该方法还可以包括：线程池中的线程从对应的子任务队列中获取扫描任务并执行获取的扫描任务。

[0050] 下面举一个具体的例子，对扫描任务总队列中的扫描任务总数大于线程池中线程总数的情形，进行更加详细的说明。

[0051] 比如：假定当前线程池中有 3 个线程，分别为线程 1’、线程 2’和线程 3’；扫描任务总队列 Q 中有 6 个扫描任务，分别为：扫描任务 1’（扫描回收站文件）：其历史执行平均耗时为 1；扫描任务 2’（扫描系统临时文件）：其历史执行平均耗时为 3；扫描任务 3’（扫描 IE 临时文件）：其历史执行平均耗时为 5；扫描任务 4’（扫描无效快捷方式）：其历史执行平均耗时为 2；扫描任务 5’（扫描内存转储文件）：其历史执行平均耗时为 4；扫描任务 6’（扫描缩略图缓存）：其历史执行平均耗时为 7。

[0052] 对各个扫描任务按照扫描任务历史执行平均耗时进行从大到小排序，此时从大到小排序结果为：

[0053] 扫描任务 6’、扫描任务 3’、扫描任务 5’，扫描任务 2’、扫描任务 4’和扫描任务 1’。

[0054] 由于线程数为 3，因此可以将扫描任务总队列 Q 划分为 3 个子任务队列，分别为 q1，q2，q3。

[0055] 接着，从排序后的扫描任务总队列中依次取出 n 个扫描任务，放入 n 个子任务队列中。

[0056] 此时，经过第一轮划分后：

[0057] q0 为：扫描任务 6’（总耗时 7）

[0058] q1 为：扫描任务 3’（总耗时 5）

[0059] q2 为：扫描任务 5’（总耗时 4）。

[0060] 接着，再对这 3 个子任务队列按照当前所含任务执行耗时进行从小到大排序，并将扫描任务总队列中剩余的扫描任务依次分配给这排序后的 3 个子任务队列。

[0061] 此时，按照当前所含任务执行耗时从小到达排序应为排序应为：q2，q1，q0，然后再将扫描任务总队列中剩余的扫描任务依次分配给这排序后的 3 个子任务队列。

[0062] 具体为：

[0063] 经过第二轮划分后：

[0064] q2 为：扫描任务 5’和扫描任务 2’（总耗时 7）；

[0065] q1 为：扫描任务 3’和扫描任务 4’（总耗时 7）

[0066] q0 为：扫描任务 6’和扫描任务 1’（总耗时 8）

[0067] 此时，按照总耗时从小到达排序应为排序应为：q2，q1，q0。而且扫描任务总队列中已没有待划分的任务，可以看到 q2，q1，q0 的耗时大致相同，总的执行时间最少。

[0068] 假定每个线程具有相同的执行资源，然后，再将这 3 个子任务队列（q2,q1,q0）分配给线程池中的 3 个线程，其中每个线程处理单一的子任务队列。

[0069] 比如：可以将这三个扫描任务随机分配给 3 个线程，或者按照命名顺序将子任务队列 1’（q0）分配给线程 1’、子任务队列 2’（q1）分配给线程 2’，以及将子任务队列 3’（q2）分配给线程 3’，等等。

[0070] 在上述实施方式中，对各个扫描任务按照扫描任务历史执行平均耗时进行从大到小排序。

[0071] 实际上，扫描任务的耗时是一个平均值，可以根据需要从配置文件中读取，并且每次扫描完成后都要重新生成扫描任务历史平均耗时，而且可以根据海量用户关于该项扫描任务耗时的上报统计出平均值。

[0072] 具体地，对于每个扫描任务，分别计算该扫描任务的历史执行平均耗时 TaverageTime，其中扫描任务的历史执行次数 k，本次扫描耗时为 currentTime，本次扫描之前的历史执行平均耗时为 averageTime；其中：

[0073] TaverageTime=(averageTime*k+currentTime)/(k+1)；

[0074] 对于各个扫描任务，按照各自计算的 TaverageTime 进行从大到小排序，然后再新计算出 averageTime 以及扫描次数 k+1，并写入配置文件以供下次扫描时读取。

[0075] 在另一个实施方式中，对于各个扫描任务，也可以仅基于各自的上次扫描耗时来对扫描任务进行排序，从而节省计算量。

[0076] 在本发明实施方式中，可以首先划分扫描任务总队列，再创建线程池；也可以首先创建线程池，再创建扫描任务总队列。

[0077] 图 2 为根据本发明实施方式的扫描任务的分配方法示范性流程图。

[0078] 如图 2 所示，该方法包括：

[0079] 步骤 201：将等待执行的扫描任务放入扫描任务总队列中。

[0080] 步骤 202：获取 CPU 的逻辑内核个数 n，并对扫描任务总队列进行划分，划分为 n 个子任务队列。

[0081] 步骤 203：创建含有线程个数为 n 的线程池，每个线程池对应于一个子任务队列。

[0082] 步骤 204：线程池中的线程持续地从指定的子任务队列中获取任务并执行该扫描任务。

[0083] 图 3 为根据本发明实施方式的对扫描任务按照任务队列进行分配的示范性流程图。

[0084] 如图 3 所示，该方法包括：

[0085] 步骤 301：准备扫描任务总队列 Q 以及 n 个子任务队列 q0、q1…q(n-1)，其中 n 为线程池中线程个数。

[0086] 步骤 302：将所有扫描任务放入扫描任务总队列 Q。

[0087] 步骤 303：对扫描任务总队列 Q 中的扫描任务，按照扫描任务历史执行平均耗时从大到小进行排序。

[0088] 步骤 304：判断扫描任务总队列 Q 中的扫描任务是否小于等于线程池中线程数 n，如果是则执行步骤 305 并结束本流程，如果不是则执行步骤 306 及其后续步骤。

[0089] 步骤 305：按照在步骤 303 中确定的历史执行平均耗时排序，从 Q 中取出全部扫描

任务(Q中同时清除掉全部扫描任务),并依次放入子任务队列中,其中如果全部扫描任务的总数为m,则依次放入子任务队列q0、q1…q(m-1)。

[0090] 步骤306:按照按步骤303中确定的历史执行平均耗时排序,从Q中取出n个任务(Q中同时清除掉这n个任务),并放入子任务队列中q1…q(n-1)。

[0091] 步骤307:对子任务队列q1…q(n-1)按其当前所含任务总执行耗时从小到达排列。

[0092] 步骤308:判断扫描任务总队列Q是否还有任务,是则返回步骤304继续运行本流程,没有则结束本流程。

[0093] 在本发明实施方式中,利用线程池的技术来分配执行各个扫描任务,而且将扫描任务总队列划分成个数与线程池中线程个数相同的子任务队列,各子任务队列中总执行耗时基本相同,并且各线程只从指定的子任务队列中获取任务执行,从而避免多线程访问同一子任务队列时的线程同步开销。

[0094] 基于上述详细分析,本发明实施方式还提出了一种扫描任务的分配装置。

[0095] 图4为根据本发明实施方式的该装置扫描任务的分配装置结构图。

[0096] 如图4所示,该装置包括线程池创建单元401和扫描任务总队列分配单元402,其中:

[0097] 线程池创建单元401,用于创建线程池,线程池包含有n个线程,其中n为自然数;

[0098] 扫描任务总队列分配单元402,用于创建扫描任务总队列,将扫描任务总队列划分为n个子任务队列,每个子任务队列对应于一个线程,并将所述扫描任务总队列中的扫描任务平均分配给所述n个子任务队列。

[0099] 在一个实施方式中,线程池创建单元401,用于创建线程池,其中线程池中线程数为中央处理单元(CPU)的逻辑内核数。

[0100] 在一个实施方式中,扫描任务总队列分配单元402,还用于在将所述扫描任务总队列中的扫描任务平均分配给所述n个子任务队列步骤之前,将所有扫描任务放入扫描任务总队列。

[0101] 优选地,扫描任务总队列分配单元402,还用于在将所有扫描任务放入扫描任务总队列之后,对扫描任务总队列中的各个扫描任务按照各自的扫描任务历史执行平均耗时进行从大到小排序。

[0102] 具体地,扫描任务可以包括:垃圾文件扫描任务、临时文件扫描任务、历史访问痕迹扫描任务或病毒扫描任务,等等。

[0103] 以上虽然通过罗列的方式详细列出了扫描任务的一些示范性具体种类,本领域技术人员可以意识到,这种罗列仅仅是示范性的,并不用于对本发明实施方式的保护范围进行限定。

[0104] 在一个实施方式中,扫描任务总队列分配单元402,用于当扫描任务总队列中的扫描任务总数小于或等于子任务队列总数n时,将所述扫描任务总队列中的扫描任务逐一分配给所述子任务队列。

[0105] 在一个实施方式中,扫描任务总队列分配单元402,用于执行:

[0106] 任务分配步骤:当扫描任务总队列中的扫描任务总数大于子任务队列总数n时,从排序后的扫描任务总队列中依次取出n个扫描任务,放入n个子任务队列中;

[0107] 排序步骤：对所述 n 个子任务队列按照当前所含任务执行耗时进行从小到大排序，并将扫描任务总队列中剩余的扫描任务依次平均分配给所述排序后的 n 个子任务队列。

[0108] 在一个实施方式中，扫描任务总队列分配单元 402 再执行将扫描任务总队列中剩余的扫描任务依次平均分配给所述排序后的 n 个子任务队列时，具体执行：

[0109] 判断步骤：判断扫描任务总队列中剩余的扫描任务总数是否依然大于子任务队列总数 n；

[0110] 如果大于，则重复所述任务分配步骤、排序步骤和判断步骤循环执行，直到扫描任务总队列中的扫描任务总数小于子任务队列总数；

[0111] 如果不大于，则将扫描任务总队列中剩余的扫描任务依次分配给所述排序后的子任务队列。

[0112] 具体地，扫描任务总队列分配单元 402，用于当扫描任务总队列中的扫描任务总数大于线程池中线程总数 n 时，对各个扫描任务按照各自的扫描任务历史执行平均耗时进行从大到小排序；从排序后的扫描任务总队列中依次取出 n 个扫描任务，放入 n 个子任务队列中；对 n 个子任务队列按照当前所含任务执行耗时进行从小到大排序，并将扫描任务总队列中剩余的扫描任务依次平均分配给排序后的 n 个子任务队列；并将 n 个子任务队列分配给线程池中的线程，其中每个线程处理单一的子任务队列。

[0113] 在一个实施方式中，扫描任务总队列分配单元 402，用于对于每个扫描任务，分别计算其扫描任务历史执行平均耗时 TaverageTime，其中扫描任务的历史执行次数 k，本次扫描耗时为 currentTime，本次扫描之前的历史执行平均耗时为 averageTime；其中：

[0114] TaverageTime=(averageTime*k+currentTime)/(k+1)；

[0115] 并对于各个扫描任务，按照各自计算的 TaverageTime 进行从大到小排序。

[0116] 可选地，扫描任务总队列分配单元 402，用于对于各个扫描任务，按照各自的上次扫描耗时进行从大到小排序。

[0117] 其中，线程池创建单元 401，可以用于创建线程池，其中线程池中线程数为中央处理单元（CPU）的逻辑内核数。

[0118] 综上所述，在本发明实施方式中，创建线程池，线程池包含 n 个线程，其中 n 为自然数；创建扫描任务总队列，并将扫描任务总队列划分为 n 个子任务队列，每个子任务队列对应于一个线程；将所述扫描任务总队列中的扫描任务平均分配给所述 n 个子任务队列。由此可见，本发明实施方式通过应用线程池技术来平均分配扫描任务总队列，从而避免了多线程访问同一子任务队列时的线程同步开销，因此提高了扫描效率。

[0119] 另外，在本发明实施方式中，优选将任务队列划分成个数与线程池中线程个数相同的子任务队列，而且各子任务队列中总执行耗时相同，从而进一步提高了扫描效率。

[0120] 以上所述，仅为本发明的较佳实施例而已，并非用于限定本发明的保护范围。凡在本发明的精神和原则之内，所作的任何修改、等同替换、改进等，均应包含在本发明的保护范围之内。

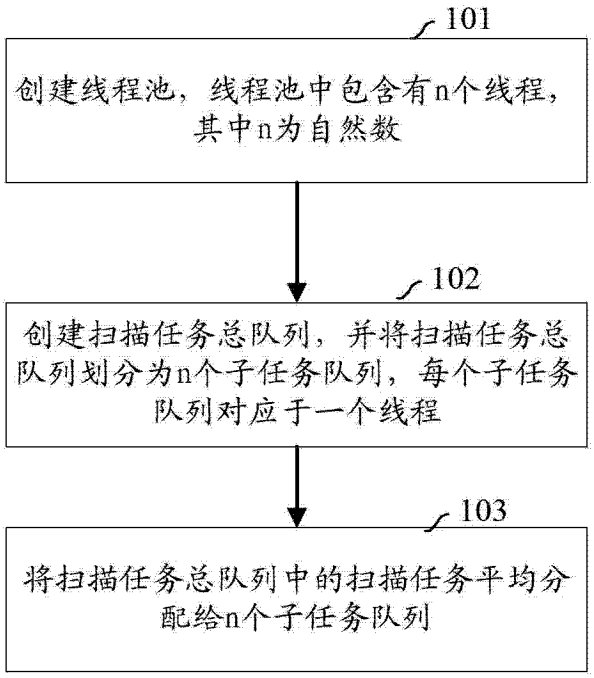

101
创建线程池，线程池中包含有n个线程，其中n为自然数
102
创建扫描任务总队列，并将扫描任务总队列划分为n个子任务队列，每个子任务队列对应于一个线程
103
将扫描任务总队列中的扫描任务平均分配给n个子任务队列

图 1

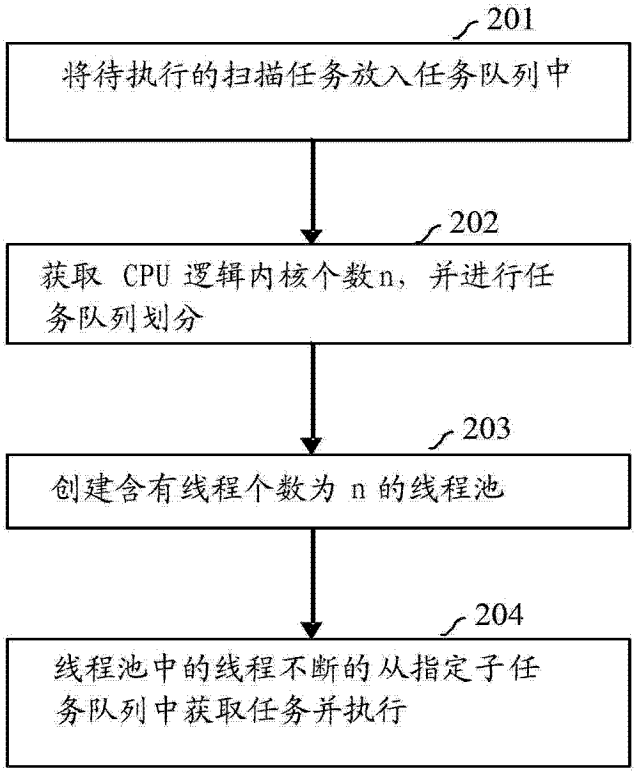

201
将待执行的扫描任务放入任务队列中
202
获取 CPU 逻辑内核个数n，并进行任务队列划分
203
创建含有线程个数为 n 的线程池
204
线程池中的线程不断的从指定子任务队列中获取任务并执行

图 2

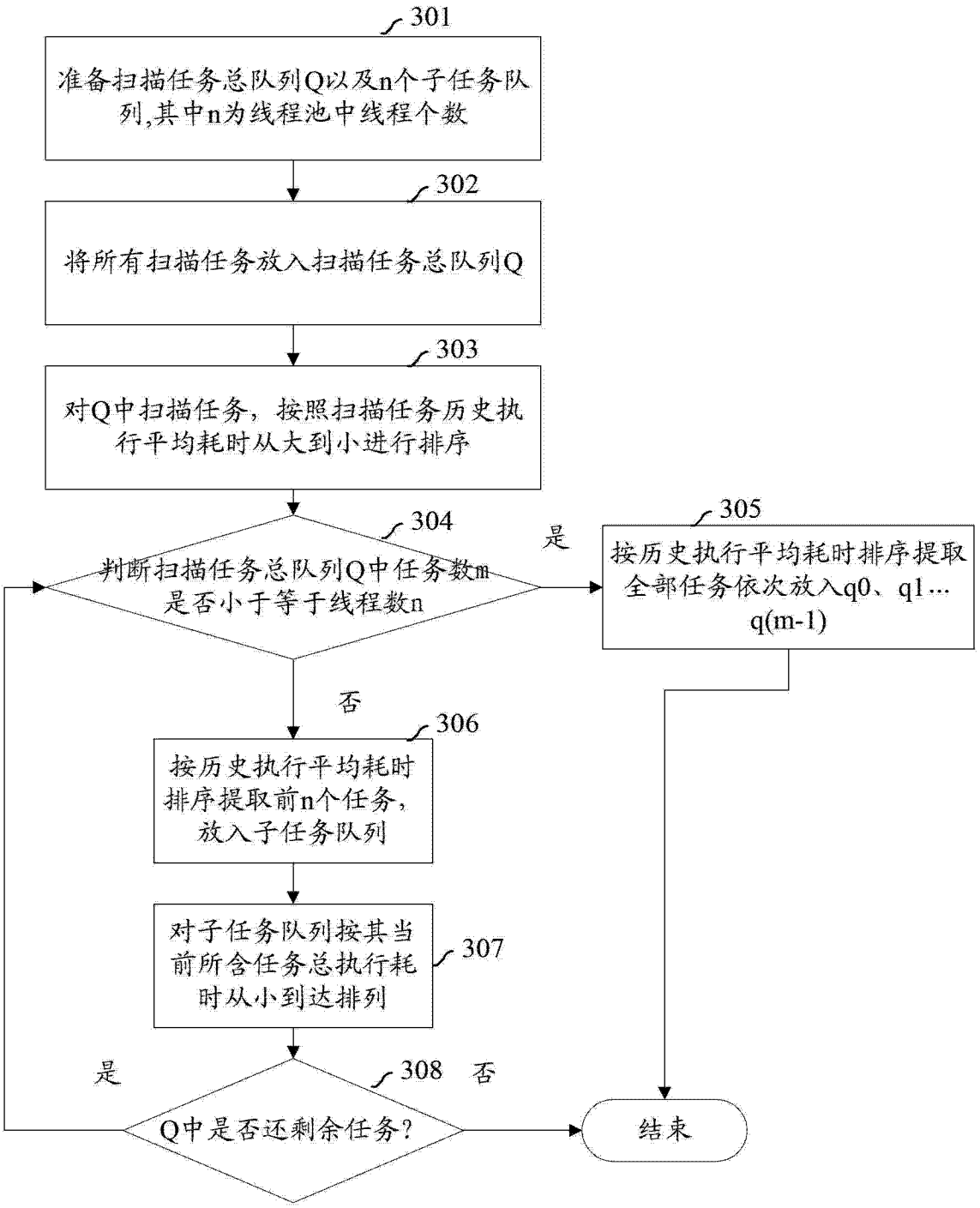

301
准备扫描任务总队列Q以及n个子任务队列,其中n为线程池中线程个数
302
将所有扫描任务放入扫描任务总队列Q
303
对Q中扫描任务，按照扫描任务历史执行平均耗时从大到小进行排序
304
判断扫描任务总队列Q中任务数m是否小于等于线程数n
是
305
按历史执行平均耗时排序提取全部任务依次放入q0、q1…q(m-1)
否
306
按历史执行平均耗时排序提取前n个任务，放入子任务队列
307
对子任务队列按其当前所含任务总执行耗时从小到达排列
是
308
否
Q中是否还剩余任务?
结束

图 3

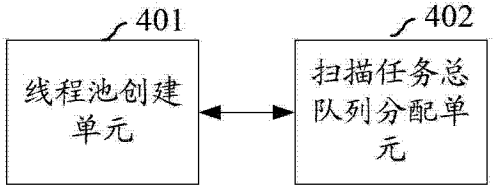

401
线程池创建单元
402
扫描任务总队列分配单元

图 4