(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国 际 局

(43) 国际公布日
2018 年 7 月 5 日 (05.07.2018)

WIPO | PCT

(10) 国际公布号
WO 2018/120993 A1

(51) 国际专利分类号:
G06F 9/50 (2006.01)

(21) 国际申请号: PCT/CN2017/106110

(22) 国际申请日: 2017 年 10 月 13 日 (13.10.2017)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:
201611261962.8 2016年12月30日 (30.12.2016) CN

(71) 申请人: 华 为 技 术 有 限 公 司 (HUAWEI TECHNOLOGIES CO., LTD.) [CN/CN]; 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。

(72) 发明人: 曾 艳 (ZENG, Yan); 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。 林宗芳 (LIN, Zongfang); 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。 朱冠宇 (ZHU, Guanyu); 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。

(81) 指定国 (除另有指明, 要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW。

(84) 指定国 (除另有指明, 要求每一种可提供的地区保护): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT,

(54) **Title:** METHOD AND DEVICE FOR ALLOCATING DISTRIBUTED SYSTEM TASK

(54) **发明名称**: 一种分布式系统任务分配的方法和装置

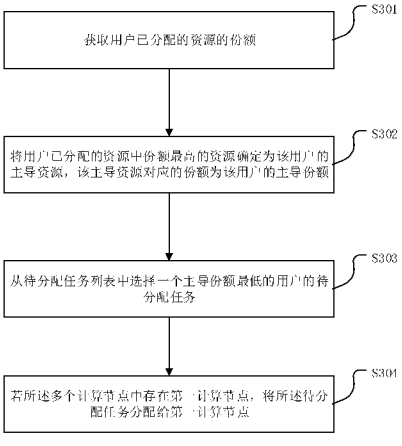

图 3

S301 ACQUIRING A SHARE OF A RESOURCE ALLOCATED TO A USER
S302 DETERMINING, FROM AMONG RESOURCES ALLOCATED TO THE USER, A RESOURCE WITH THE LARGEST SHARE TO BE A DOMINANT RESOURCE OF THE USER, THE SHARE CORRESPONDING TO THE DOMINANT RESOURCE BEING THE DOMINANT SHARE OF THE USER
S303 SELECTING, FROM A LIST OF TASKS TO BE ALLOCATED, A TASK TO BE ALLOCATED OF A USER WITH THE LOWEST DOMINANT SHARE
S304 IF THERE IS A FIRST COMPUTING NODE IN THE PLURALITY OF THE COMPUTING NODES, ALLOCATING THE TASK TO BE ALLOCATED TO THE FIRST COMPUTING NODE

(57) Abstract: The present application relates to the field of distributed systems, in particular to resource scheduling technology for a distributed system. In a method for allocating tasks, a share of a resource allocated to a user is acquired, a task to be allocated is selected from a list of tasks to be allocated, and based on the highest threshold value, the task to be allocated is allocated to a first computing node, the remaining resources thereof being capable of satisfying the task to be allocated, and after the task to be allocated is allocated to the first computing node, there is, in the first computing node, at least one type of computing node, the remaining amount of monitored resources thereof being greater than or equal to the highest threshold value corresponding to the monitored resource. The solution provided in the present application may reduce the generation of resource fragments in a distributed system, thereby improving the resource utilization rate of a distributed system.

(57) 摘要: 本申请涉及分布式系统领域，尤其涉及分布式系统中的资源调度技术。在一种任务分配的方法中，获取用户的已分配资源的份额，从待分配任务列表中选择一个待分配任务，并基于最高阈值，将待分配任务分配到剩余资源能够满足所述待分配任务的第一计算节点，且所述待分配任务分配到所述第一计算节点后，所述第一计算节点中存在至少一种被监测资源的剩余量大于或等于与所述一种被监测资源相对应的最高阈值的计算节点中。通过本申请提供的方案，可以减少分布式系统中资源碎片的产生，提高分布式系统的资源利用率。

WO 2018/120993 A1

[见续页]

RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

本国际公布:

— 包括国际检索报告(条约第21条(3))。

一种分布式系统任务分配的方法和装置

本申请要求在2016年12月30日提交中国专利局、申请号为201611261962.8、发明名称为“一种分布式系统任务分配的方法和装置”的中国专利申请的优先权，其全部内容通过引用结合在本申请中。

技术领域

本申请涉及分布式系统领域，尤其涉及分布式系统中的资源调度技术。

背景技术

在各种分布式计算框架（如 hadoop、spark 等）以及分布式资源统一管理与调度平台（如 mesos 和 yarn）中，细粒度、多维度的资源调度是分布式计算框架和分布式资源统一管理与调度平台的一个核心问题。而在资源调度时，如何实现资源的公平分配和资源利用率的提高是一个关键问题，也是当前分布式资源管理与调度技术领域中的热门话题。

目前一些主流的分布式资源管理与调度框架如 Mesos、Yarn 等都采用了 DRF(Dominant Resource Fairness，主导资源公平)算法。该算法的主要思想是在多维度资源环境下，一个用户的资源分配应该由用户的 dominant share（主导份额）决定，dominant share 是在所有已经分配给用户的多种资源中，占据总资源的最大值，该值对应的资源为主导资源。DRF 算法的主旨是试图最大化所有用户中最小的 dominant share，或者尽可能使不同用户的主导资源相等。

DRF 算法虽然保证了用户资源公平性，但对任务的分配存在资源碎片问题。即通过 DRF 算法进行资源调度后，可能出现每个节点的剩余资源都不足以满足某一任务的资源需求，但是从分布式系统整体来看，各个节点上该种剩余资源的总和却又大于该任务的资源需求，从而造成了资源碎片。资源碎片问题会导致资源利用率降低，并且由于资源碎片不能被应用，导致一些任务执行延迟，时间性能降低。

发明内容

本文描述了一种分布式系统任务分配方法，装置及系统，以减少分布式系统中的资源碎片，提高系统资源利用率和任务执行效率。

一方面，本申请的实施例提供一种基于 DRF 算法的分布式系统任务分配的方法。方法包括当基于 DRF 算法进行分配时，若要分配的计算节点在分配了待分配任务后的一种被监测资源的剩余量小于与所述一种被监测资源相对应的最高阈值，则不将该待分配任务分配到该计算节点中。由此，分后待分配任务后的计算节点仍然有一定的剩余量用以执行其他任务，从而降低了资源碎片的产生。

被检测资源是指在分布式系统中的各类资源中，需要对碎片的产生进行监控的资源。在具体的实施例中，被监测资源可以通过人工进行预设或者指定，在另一些具体的实施例中，也可以通过管理系统动态的确定和调整被监控资源。例如，通过对资源碎片产生的监控，将产生资源碎片多的资源作为别监控资源。

在一种实现方式中，方法包括获取用户的已分配的资源的份额，份额是指用户的已分配的一种资源占资源与分布式系统中的可分配总量的比值，用户的已分配的资源中份额最高的资源为用户的主导资源，主导资源对应的份额为用户的主导份额；从待分配任务列表中选择一个待分配任务，待分配任务为在多个用户中主导份额最低的用户的任务；若所述多个计算节点中存在第一计算节点，将所述待分配任务分配给第一计算节点，其中，所述第一计算节点为剩余资源量能够满足所述待分配任务对资源的需求量的计算节点，且所述待分配任务分配到所述第一计算节点后，所述第一计算节点中存在至少一种被监测资源，所述一种被监测资源的剩余量大于或等于与该被监测资源对应的最高阈值。通过本方法，可以在保证用户资源均衡的前提下，实现前述降低资源碎片的目的，提高资源利用率。

在该种实现的一种可能的实施方式中，方法还包括还包括，若所述多个计算节点中不存在第一计算节点，且存在第二计算节点，将所述待分配任务分配给第二计算节点，其中，所述第二计算节点为剩余资源量能够满足所述待分配任务对资源的需求量的计算节点，且所述待分配任务分配到所述第二节点后，所述第二节点种至少存在一种被监测资源，所述一种被监测资源的剩余量小于或等于与所述被检测资源相对应的最低阈值，所述最低阈值小于所述最高阈值。

在另一种实现方式中，方法包括获取用户的已分配的资源的份额，份额是指用户的已分配的一种资源占资源与分布式系统中的可分配总量的比值，用户的已分配的资源中份额最高的资源为用户的主导资源，主导资源对应的份额为用户的主导份额；从待分配任务列表中选择一个待分配任务，待分配任务为在多个用户中主导份额最低的用户的任务；若所述多个计算节点中存在第一计算节点，将所述待分配任务分配给第一计算节点，其中，所述第一计算节点为剩余资源量能够满足所述待分配任务对资源的需求量的计算节点，且所述待分配任务分配到所述第一计算节点后，所述第一计算节点中存在至少一种被监测资源，所述一种被监测资源的剩余量大于或等于与该被监测资源对应的最高阈值或者剩余量小于或等于与该被监测资源对应的最低阈值，其中最低阈值低于最高阈值。通过本方法，可以在保证用户资源均衡的前提下，实现前述降低资源碎片的目的，提高资源利用率，且产生低于最低阈值的可容忍碎片时仍然会进行分配，从而提高了算法的任务分配效率。

在另一种实现方式中，在另一种实现方式中，方法包括获取用户的已分配的资源的份额，份额是指用户的已分配的一种资源占资源与分布式系统中的可分配总量的比值，用户的已分配的资源中份额最高的资源为用户的主导资源，主导资源对应的份额为用户的主导份额；从待分配任务列表中选择一个待分配任务，待分配任务为在多个用户中主导份额最低的用户的任务；若所述多个计算节点中存在第一计算节点，将所述待分配任务分配给第一计算节点，其中，所述第一计算节点为剩余资源量能够满足所述待分配任务对资源的需求量的计算节点，且所述待分配任务分配到所述第一计算节点后，所述第一计算节点中存在至少一种被监测资源，所述一种被监测资源的剩余量小于或等于与该被监测资源对应的最低阈值，其中最低阈值低于最高阈值。或者，若所述多个计算节点中不存在第一计算节点，且存在第二计算节点，将所述待分配任务分配给第二计算节点，其中，所述第二计算节点为剩余资源量能够满足所述待分配任务对资源的需求量的计算节点，且所述待分配任务分配到所述第二节点后，所述第二节点种至少存在一种被

监测资源，所述一种被监测资源的剩余量大于或等于与所述被检测资源相对应的最高阈值，所述最低阈值小于所述最高阈值。通过本方法，可以在保证用户资源均衡的前提下，实现前述降低资源碎片的目的，提高资源利用率，且产生低于最低阈值的可容忍碎片时仍然会进行分配，从而提高了算法的任务分配效率。

在一种实现方式中，最高阈值大于或等于所述待分配任务列表中至少一个待分配任务对所述被监测资源的需求量。由此，能够保证每次分配后的剩余资源量能够至少之星一个待分配任务，从而减少碎片的产生。

在一种实现方式中，最高阈值大于或等于所述待分配任务列表中一种被监测资源的需求量最小的N个未分配任务中每个任务的一种被监测资源的需求量的最大值，其中，N为大于或等于1且小于等于所述待分配任务列表中未分配任务总数的整数。所谓最小的N个未分配任务是指将未分配任务的被检测资源的需求量从小到大排列，取其中前N个未分配任务。由此，最高阈值的取值能够保证计算节点的被监测资源的剩余资源量能够至少执行一个未分配的任务，且能够使最大阈值的取值尽可能的小，从而提高分配效率。

在一种是实现方式中，需要第一计算节点中任意一种被监测资源的剩余量均大于或等于与所述任意一种被监测资源相对应的最高阈值。由此，可以使得第一计算节点中每种被监测资源的剩余量均能够大于最高阈值，从而减少所有被监测资源的资源碎片的产生。

在一种实现方式中，最高阈值大于或等于至少一组任务中每组任务的所述一种被监测资源的最大需求量中的最大值，所述最大需求量为一组任务中每个任务的所述一种被监测资源的需求量的最大值，所述一组任务为所述待分配任务列表中的N个未分配任务，N为大于或等于1的整数。由此，使得最高阈值的取值大于一组中被监测资源需求量的最大值，避免了当存在多种被监测资源时，每种被监测资源的阈值的取值过小而使得没有计算节点同时满足所有的被监测资源相对应的阈值的情况，提高了算法是适应性和效率。

在该种实现方式的一种具体的方案中，一组任务是指待分配任务列表中任意一种被监测资源需求量最小的N个未分配任务。由于以被监测资源需求量最小的N个未分配任务为一组，可以使得最高阈值的在满足前述的要求下，尽可能的小，进而提高算法的效率。

在一种实现方式中，方法还包括：获取采样任务数据，所述采样任务数据包含多个任务的资源需求信息，以及根据所述采样任务数据，确定至少一种被监测资源相对应的最高阈值。由此，通过采样任务数据来确定最高阈值，可以灵活的选择不同的采样数据，提高算法的适用性。

在该种实现方式下的一种具体实现中，最高阈值大于或等于至少一组任务中每组任务的所述一种被监测资源的最大需求量中的最大值，所述最大需求量为一组任务中每个任务的所述一种被监测资源的需求量的最大值，所述一组任务为采样任务中的N个未分配任务，N为大于或等于1的整数。在一种具体的实现中，一组任务为采样任务中任意一种被监测资源需求量最小的N个未分配任务

在该种实现方式下的另一种具体实现中，确定被监测资源X对应的最小任务集合的被监测资源Y的最大需求量为所述被监测资源Y相对应最高阈值，其中，被监测资源X为任意一种被监测资源，被监测资源Y为所要确定相对应的最高阈值的被监测资源，所述被

监测资源X对应的最小任务集合为所述采样任务数据中对所述被监测资源X的需求量最小的M个任务，所述最小任务集中每个任务对被监测资源Y的需求量的最大值为所述最小任务集的被监测资源Y的最大需求量，M为大于或等于1的正整数；或者，确定多种被监测资源对应的多个最小任务集合的被监测资源Y的最大需求量的最大值为所述被监测资源Y相对应最高阈值。由此，按照最小任务集合中的M个任务来确定最高阈值，能够保证各类被监测资源的剩余量大于或等于最高阈值的情况下能够执行M个任务。

在一种实现方式中，方法还包括获取至少一个更新采样任务数据，更新采样任务数据包括预设的时间段内执行的任务的资源需求信息；根据所述更新采样任务数据，更新至少一种资源相对应的最高阈值。由此，可以对采样数据进行更新，从而更新最高阈值，从而提高任务分配的准确性。

在一种实现方式中，方法还包括若不存在第一计算节点时，减小最高阈值的取值。

在一种实现方式中，方法还包括若不存在第一计算节点，且不存在第二节点时，减小最高阈值的取值。

在一种实现方式中，方法还包括从待分配任务列表中选择一个待分配任务之后，若所述待分配任务列表中没有除所述选择的待分配任务外的其他待分配任务，则所述多个计算节点中剩余资源能够满足所述选择的待分配任务的计算节点为可分配所述选择的待分配任务的计算节点。由此，若待分配的任务是待分配任务列表的最后一个任务，则不再考虑资源碎片的问题，提高了分配效率。

另一方面，本发明实施例提供了一种管理节点，该节点具有实现上述方法的功能。所述功能可以通过硬件实现，也可以通过硬件执行相应的软件实现。所述硬件或软件包括一个或多个与上述功能相对应的模块。

在一种可能的实现方式中，管理节点的结构中包括处理器、存储器和网络接口，所述处理器被配置为支持管理节点执行上述方法中相应的功能。所述网络接口用于与用户或者计算节点进行通信，从而获取或者上述方法中的外部信息。所述存储器用于与处理器耦合，其保存基站必要的程序指令和数据。

又一方面，本发明实施例提供了一种分布式系统，该系统包括上述方面所述管理节点以及用于为多个用户的待分配任务提供所需的资源以执行所述待分配任务的计算节点。

再一方面，本发明实施例提供了一种计算机存储介质，用于储存为上述管理节点所用的计算机软件指令，其包含用于执行上述方面所设计的程序。

相较于现有技术，本发明提供的实施例可以减少分布式系统任务分配中的资源碎片的产生，从而提高资源利用率和系统执行效率。

附图说明

为了更清楚地说明本发明实施例或现有技术中的技术方案，下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍。显而易见地，下面附图中反映的仅仅是本发明的一部分实施例，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本发明的其他实施方式。而所有这些实施例或实施方式都在本发明的保护范围之内。

图1A为本发明的一种可能的应用场景示意图；

图 1B 为本发明的一种可能的应用场景的系统架构图；

图 1C 为本发明的另一种可能的应用场景的系统架构图；

图 1D 为本发明的又一种可能的应用场景的系统架构图；

图 1E 为本发明的又一种可能的应用场景的系统架构图；

图 2 为 DRF 算法的流程示意图；

图 3 为本发明的一种实施例的方法流程图；

图 4 为本发明的又一种实施例的方法流程图；

图 5 为本发明的又一种实施例的方法流程图；

图 6 为本发明的又一种实施例的方法流程图；

图 7 为本发明实施例的一种及管理节点的逻辑结构图；

图 8 为本发明实施例的一种管理节点的硬件结构图。

具体实施方式

为使本发明的目的、技术方案和优点更加清楚，下面将结合附图对本发明实施方式作进一步地详细描述。

结合图 1A，是本发明实施例的一种可能的实现场景的系统示意图。图 1A 为包含一个或多个互连计算设备的分布式系统 100 的简化图示。显示了三种类型的设备：计算设备，管理设备以及客户机和服务器。然而，设备的数量和类型仅仅用于例证目的，并且对本发明来说并不是至关重要的。

示出了诸如主机 1011、1012 和 1013 之类的主机。主机可以是计算设备，除了其它功能之外，所述计算设备向其他计算设备提供一个或多个服务。举例来说，主机可以是企业网络中的服务器，数据库，或者可以是向其他计算设备提供数据和/或服务的其他任何设备。

图 1A 还示出了系统 100 中的多个管理设备。为了简单起见，示出了两个管理设备，即管理设备 1021 和管理设备 1022。管理设备可以执行不同的功能，包括但不局限于控制用户任务由那一个计算设备提供资源从而执行。

在图 1A 的基础上，下面通过图 1B、图 1C 和图 1D 和图 1E 示例性的说明本发明实施例应用的几类常见的分布式系统的系统架构。

如图 1B 所示的是一种一层架构的中心化分布式系统架构示意图。在该架构下，用户(user1、user2、……)将任务提交到主节点（master）上，主节点通过调度器（scheduler）对任务进行调度分配，为任务分配所需要的资源，从而把任务分配带满足任务所需资源的从节点（slave1、slave2、……slaveN）上。在该架构下，本发明实施例所述的执行主体管理节点指的是该架构下的主节点，具体由调度器执行本发明实施例所述的方法。

如图 1C 所示的是一种二层架构场景下的中心化分布式系统架构图。在该架构下，主节点分配资源给多个框架（framework）框架。框架用于去解决或者处理复杂的问题，比如大数据处理框架 hadoop、批处理框架 spark 等。在该架构下的一种可能的调度方式中，framework 向主节点告知自己所需多个任务所需的总资源，由主节点分配相应资源给 framework，framework 再进行二级调度(将资源分配给各个任务)。该场景下，本发明实施例所述的执行主体管理节点指的是该架构下的主节点，具体由主节点中的调度器（Scheduler）执行本发明实施例所述的方法。在该构架下的另一种可能的调度方式中，主

节点以写资源空闲列表的方式将空闲资源告知 framework，由 framework 挑选资源，并将资源分配给任务。该场景下，本发明的执行主体在 framework 的调度器(Scheduler)中。

如图 1D 所示是一种去中心化的分布式系统架构。在去中心化的架构下，分布式系统中包含多个节点（Node1、Node2、Node3……），每个节点都具有高度自治的特征。节点之间彼此可以自由连接，形成新的连接单元。任何一个节点都可能成为阶段性的中心，但不具备强制性的中心控制功能。因此，当一个节点在某个阶段具备调度功能，可以将用户（user）提交的任务进行分配时，本发明实施例所述的执行主体管理节点指的是该节点，具体由节点的调度器（Scheduler）执行本发明实施例所述的方法。可以理解的是，在图 1D 中，用户仅示意性的描述了一个，但在实际架构中，可以有多个用户同时以类似的方式接入分布式系统进行任务的调度和执行。

图 1E 所示的是一种集群联邦架构下的分布式系统。在该架构下，存在多层的主节点，L1 的主节点（L1:master）可以将用户（user1、user2、user3、user4）的任务调度给 L2 的主节点（L2:master），则对于 L1 的主节点而言，计算节点是以每个 L2 的主节点下的子节点的资源总量的集合。L2 的主节点可以对分配的任务进行再次调度，将任务分配到具体的子节点（slave1、slave2、……slaveN）中，对于 L2 的主节点而言，计算节点是每个子节点的资源集合，或者每个子节点中的多个资源集合（参考后述的计算节点的划分原理）。用户也可以直接将任务直接通过 L2 的主节点进行调度。在该种架构下，本发明实施例所述的执行主体管理节点指的是该各层的主节点，具体由节点的调度器（Scheduler）执行本发明实施例所述的方法。

在前述的多种架构中，调度器可以是集成在相关节点上的硬件设备，也可以通过节点的通用硬件通过软件实现。在本发明实施例中，不限制管理节点实施所述调度器功能的具体方式。此外，在本发明实施例中，用户可以指一个客户端，或者一个

本发明实施例中所述的资源调度，是指将分布式中的计算节点的资源分配给用户的待分配任务。应当注意的是，在本发明的各类实施例中，计算节点指的是分布式系统中作为资源调度单位的一个资源集合，一般而言，计算节点是以一台服务器或者一个物理计算机为单位的。但是，在一些场景下，针对不同的资源类型，可以按照不同的划分单位来划分计算节点。例如，当以 CPU 作为被监测资源时，可以按照处理器或者处理核作为划分单位，分布式系统中每个处理器或者处理核作为一个计算节点，一个服务器中可能包含多个计算节点。又例如，当仅以存储资源作为被监测资源时，可以按照数据分片作为计算节点的划分单位，则分布式数据库中一个数据分片作为一个计算节点，一个服务器中可能包含多个数据分片，即包含多个计算节点。

本发明实施例基于现有的 DRF 算法进行改进，因此，除特殊说明外，现有 DRF 算法中的方法以及概念可以适用于本发明实施例中。结合图 2，是对现有 DRF 算法的算法流程的简要介绍。

如图 2 所示，DRF 算法进行资源分配的具体步骤:

（1）计算每个用户的每一种已经分配给用户的资源的份额(share)，份额为该种资源占分布式系统总资源的比值。基于一个用户的各个资源的份额，选择选择份额中的最大值为用户的主导份额（dominant share）S_i ：

$$s_i = \max_{j=1}^{m}\{u_{ij} / r_j\}$$

其中，u_{ij} 表示用户 i 对资源 j 的占用量，r_j 表示资源 j 的总量, m 表示资源总类型。

（2）每次从任务列表中选择主导份额最低的的用户的一个任务准备运行，若系统中有足够的可用资源执行该任务，则启动该任务进行执行。

（3）重复第一步和第二步，直到不存在可用资源或不存在需要执行的任务。

DRF 算法的主旨是试图最大化所有用户中最小的主导份额或尽可能使不同用户的主导份额相等。举个例子，假如用户 A 运行 CPU 密集的任务而用户 B 运行内存密集的任务，DRF 会试图均衡用户 A 的 CPU 资源份额和用户 B 的内存资源份额。

本发明实施例对 DRF 算法进行了改进，在确定了每次从任务列表中选择主导份额最低的的用户的一个任务后，不会直接将其分配到任意一个剩余资源满足该任务的节点中。而是通过引入最高阈值，通过分配了该任务后节点剩余资源与最高阈值的关系，从而确定该节点是否为该任务的可分配节点。仅将任务分配到可分配的节点中，从而减少了任务分配后该节点产生碎片的可能性。

结合图 3，是本发明一种实施例的方法流程图。所述方法包括:

S301、获取用户的已分配的资源的份额。

份额是指所述用户的已分配的一种资源占所述资源与所述分布式系统中的总量的比值。已分配的资源是指该用户已经分配的任务中所占用的各类资源。资源是指分布式系统中的各类系统资源，包括但不限于处理器资源、内存资源、存储资源、网络带宽资源、GPU 加速资源等。

在具体的实施方式中，可以获取用户所占用的的每种资源的份额，也可以只确定特定的几种资源的份额。在确定某种资源的份额时，需要知道该用户的已分配的任务该种资源的占用量，以及分布式系统中该种资源的总量。其中，在一种可能的实施方式中，分布式中的每个服务器或者计算节点上报各自的可分配资源总量以及该用户已经分配的资源量，从而使得控制设备能够根据上报数据统计各个用户的已分配的任务的该种资源占用量以及分布式系统中该种资源的总量；在另一种可能的实施方式中，控制节点中保存有分布式系统的可用资源总量数据，并在每进行任务分配后记录各个用户所分配的资源量，从而可以直接存储了出各个用户的已分配的任务的该种资源占用量以及分布式系统中该种资源的总量信息。

S302、将用户的已分配的资源中份额最高的资源确定为该用户的主导资源，该主导资源对应的份额为该用户的主导份额。

用户的已分配的资源是对应前一步骤中获取了份额的已分配资源。即，在具体实施例中，若获取的是用户所占用的每种资源的份额，则主导资源为所分配的中资源中份额最高的资源；若获取的是特定几种资源的份额，则主导资源为所述特定几种资源中份额最高的资源。

在具体实施方式中，S301 步骤和 S302 步骤可以多次实施，从而确定多个用户的主导份额。可以理解的，对于多个用户而言，由于每个用户的主导资源不同，因此多个用户的主导份额可能分别对应的是不同的资源。

S303、从待分配任务列表中选择一个主导份额最低的用户的待分配任务。

待分配任务列表是存储有需要在分布式系统中执行的任务的信息的数据结构。每个任务的信息包含了该任务所需要的各类资源的需求量。所谓待分配任务，是指需要在分布式系统中执行，但尚未分配到分布式系统的计算节点中的任务，即，尚未分配资源的

任务。在一种实现方式中，待分配任务列表可以按照不同用户分为不同的子表，每个子表中包含了该用户的所有需要在分布式系统中执行的任务；在另一种实现方式中，待分配任务列表中的每个任务信息中还包含了该任务所属的用户信息，从而可以基于用户信息区分不同用户的任务。此外，待分配任务列表中可以只存储尚未分配的任务，当任务进行分配后，则将其从列表中移除；待分配任务列表也可以对于已分配或者未分配的任务通过进行标记的方式进行区分，从而不需要将已分配的任务移除。

从待分配任务列表中，选择主导份额最低的用户的任务。结合前述，确定了用户的主导份额，根据所确定的主导份额，可以确定主导份额最高的用户。该主导份额最低的用户的任务即为所要选择的待分配任务。

一般而言，同一用户在待分配任务列表中可能有多个任务。在本实施例中，并不限制具体选择该用户的哪一个任务。在可能的实施方式中，可以按照不同的预设规则进行选择。例如，可以选择该用户多个任务中优先级最高的任务，或者按照时间先后顺序或者入队顺序依次选择最先进入列表的任务，或者选择资源占用最大或者最小的任务等。

S304、若所述多个计算节点中存在第一计算节点，将所述待分配任务分配给第一计算节点，其中，所述第一计算节点为剩余资源量能够满足所述待分配任务对资源的需求量的计算节点，且所述待分配任务分配到所述第一计算节点后，所述第一计算节点中存在至少一种被监测资源，所述一种被监测资源的剩余量大于或等于与该被监测资源对应的最高阈值。

第一计算节点为分布式系统中全部或者部分计算节点中，满足上述条件的计算节点。在可能的实现方式中，可以根据当前用户或者当前任务，或者根据其他的因素，先确定分布式系统全部节点中的部分计算节点，再从部分节点中确定满足前述条件的计算节点作为第一计算节点。

计算节点的剩余资源，是指该节点当前各类资源的可分配量，即该节点各类资源的可分配总量减去该类资源的已分配量后的剩余量。而分配了待分配任务后资源的剩余量，是指该种资源的剩余资源减去待分配任务该种资源的需求量后的剩余量。

计算节点的剩余资源能够满足所述待分配任务，是指节点当前各类资源的可分配量均大于待分配任务对各类资源的需求量。计算节点分配了待分配任务后，有至少一种被监测资源的剩余量大于或等于相应的最高阈值，是指计算节点当前的可分配的各类资源中，一种或几种特定的资源的当前可分配量减去待分配任务对该一种或者几种特定的资源的需求量之后，其剩余量大于或等于相应的阈值。被监测资源即指的特定的一种或者几种资源，被监测资源可以通过预设方式，由人工指定特定的一种或者几种资源，从而保证任务分配过程中被监测资源的剩余量在阈值之上，从而控制被监测资源的碎片化程度。

在一种可能的实现中，被监测资源也可以由分布式系统的管理节点或者具有类似功能的节点根据分布式系统中各个节点的资源碎片情况或者资源负载情况动态的调整被监测资源的范围。例如，当系统中某类资源在较多的节点中均存在资源碎片，则将该类资源调整为被监测资源；又如，当某类资源在任务列表中多数任务均需要分配，则该类资源由于频繁调度产生碎片的几率较高，则将该类资源调整为被监测资源。

在一种可能的实现方式中，最高阈值可以通过人工预先配置得到。根据历史数据以及用户任务需求等因素，对被监测资源所对应的最高阈值进行配置，使得剩余量高于最

高阈值的被监测资源有较大的可能性满足后续任务的需求，从而减小碎片的产生。在另一些可能的实现方式中，最高阈值可以通过历史数据或者未分配任务的资源需求进行计算得到，具体方式可参考后续的实施例。

在本实施例中，需要至少一种被监测资源的剩余量满足前述条件。即，在前述属于被监测资源的资源中，在一些具体的实现方式中，可以只需要该节点的特定的一种或者几种被监测资源满足前述条件，该节点即为第一计算节点；或者，只需要该节点满足前述条件的被监测资源的种数达到预设的数量（一种或者几种），该节点即为可分配的计算节点；或者，该节点中所有被监测资源均满足前述条件，该节点才可为可分配的计算节点。

将待分配任务分配给第一计算节点，可以在每个待分配任务确定了与之对应的第一计算节点后，即将该任务分配到该计算节点中，也可以当多个待分配任务均确定待分与之对应的第一计算节点后，再将多个任务分别分配到与之对应的第一计算节点中。当与之对应的第一计算节点有多个时，从多个所述计算节点中选择一个节点进行分配，在选择时可以考虑节点负载等其他因素，本实施例中不再赘述。

在本实施例中，通过前述 S301-S304 步骤，可以完成对一个待分配任务的分配。在具体的实现方式中，基于该实施例的思想，可以重复重复执行全部步骤或者部分步骤，从而将对待分配任务列表中的任务分配到计算节点中。

例如，如图 4 所示的，在一种具体的实现方式中，基于输入的待分配任务列表，以及计算节点空闲资源队列，按照如下方式对待分配任务列表中的任务进行分配:

S401、输入待分配任务列表以及空闲资源列表，并确定各个被监测资源相对应的阈值，执行 S402。

空闲资源列表用于记录当前各个节点的剩余资源量。在一种可能的实现中，列表中每个元素的数据结构中存储着一个计算节点的信息以及该计算节点的各类资源的剩余资源量。

S402、参考步骤 S301、S302 以及 S303，确定主导份额最小的用户，并从待分配任务列表确定一个该用户的待分配任务，并执行 S403。

S403、参考步骤 S304，判断空闲资源列表中当前元素中的被监测资源是否满足判断条件，若满足，执行 S404，若不满足，执行 S405。

S404、将空闲资源列表中的当前元素中的资源分配给待分配任务，即，将待分配任务分配到当前元素所对应的计算节点中。并执行 S407。

当执行 S404 时，将一个任务进行分配后，会造成用户的已分配资源发送变化，从而导致主导份额变化，因此需要对变化的用户的主导份额进行更新。

S405、当前空闲资源列表是否全部遍历，即，通过 S403 以及 S407 步骤的循环，是否遍历了空闲资源列表的全部元素而没有发现满足条件的元素。若是，执行 S407；若否，执行 S406。

S406、选取资源队列的下一个元素，并执行 S403。

S407、判断待分配任务列表中的任务是否全部已经分配，或者已经遍历后且进行任何分配。即，如果任务列表中的任务是否全部已经分配；或者经过 S402、S403、S404 步骤的多次循环后，遍历了待分配任务列表中的全部任务且所有任务均无法再进行分配时，则执行 S408 结束分配，否则执行 S402，继续进行任务分配。

S408、完成任务分配。

根据本实施例，再分配待分配任务时，将待分配任务分配到分配待分配任务后的被监测资源的剩余量大于或等于最高阈值的计算节点中，从而使得分配后的计算节点仍然有一定的资源量用以分配其他任务，减少了由于分配任务后资源的剩余量过小而造成的资源碎片化问题，提高了资源利用率。

结合图 5，是本发明又一种实施例的方法流程图，该实施例基于图 3 所示的实施例基础上进行改进，在 S304 的判断条件的基础上，增加了其他的判断条件。因此，本实施例可以结合图 3 所示实施例进行理解。

S501、S502、S503 步骤与 S301、S302、S303 步骤相同，在本实施例中不再赘述。

S504、若计算节点的剩余资源能够满足所述待分配任务，且计算节点分配了待分配任务后，有至少一种被监测资源的剩余量大于或等于相应的最高阈值，则该计算节点为可分配待分配任务的计算节点。

S505、若计算节点的剩余资源能够满足所述待分配任务，且计算节点分配了待分配任务后，有至少一种被监测资源的剩余量小于或等于相应的最低阈值，则该计算节点为可分配待分配任务的计算节点。其中，最低阈值的

S506、若计算节点的剩余资源能够满足所述待分配任务，且待分配任务列表中没有除该待分配任务外的其他任务，则该计算节点为可分配待分配任务的计算节点。

S504、S505、S506 为判断计算节点是否为可分配待分配任务的节点的三个判断条件，该三个判断条件为并列关系，一个计算节点满足三个判断条件中的任意一个，即为可分配待分配任务的计算节点。

在具体的实现方式中，一个节点对三个条件的判断的具体顺序以及相应的任务分配步骤可以是多种多样的。例如，可以先判断 S504,若当计算节点不满足 S504 的条件后，再判断 S505，若 S504、S505 均不满足时，再判断 S506。也可以按照其他顺序进行判断。此外，对于分布式系统中的所有计算节点而言，可以分别对每个节点进行三个条件的判断，也可以根据一个条件轮询全部节点，然后再根据另一个条件轮询为满足前一条件的剩余节点。

在具体的实现方式中，可以三个条件均进行判断，也可以只判断 S504、S505 或者只判断 S504、S506。

S504 为与 S304 步骤中相一致的判断条件，可以参考 S304 的描述进行理解。

S505 中，第二资源可以是与被监测资源相同资源，也可以是不同的资源。最低阈值可以通过预设的方式进行配置，使得最低阈值的取值小于或等于当前分布式系统中对于第二资源而言可以容忍的资源碎片大小。

S506 中，待分配任务列表中没有除该待分配任务外的其他任务，即当前的待分配任务是待分配任务列表中的最后一个任务。此时，由于之后没有任务再需要分配，因此在分配该任务时，无需在考虑资源碎片的问题。

S507、将待分配任务分配到 S505、S506、S507 中所确定的可分配所述可分配所述待分配任务的计算节点中。

应当注意的是，“可分配所述待分配任务的计算节点”仅仅是在本实施例中方便说明而提出的概念。在具体的实现方式中，可以存在将节点确定为“可分配所述待分配任务

的计算节点”这一步骤，然后将任务分配到这些节点中；也可以仅在逻辑上存在“可分配所述待分配任务的计算节点”，而没有实际将节点定义为该概念的步骤。例如，在每次执行完前述 S505、S506、S507 中的的判断步骤后，可以直接将符合判断条件的节点进行任务分配。

在本实施例中，在能够实现上一实施例所述的技术效果外，在确定可分配待分配任务的计算节点是，还可以分配到将有至少一种第二资源的剩余量小于或等于相应的最低阈值的计算节点，从而可以使得在分配任务时可以产生小于等于最低阈值的可容忍的资源碎片，提高了任务分配方法的适应性和分配效率。此外，当待分配任务为任务列表中的最后一个任务时，直接分配到资源满足的计算节点中，而不再考虑是否产生资源碎片，从而提高了分配效率。

下面介绍本发明提供的又一实施例，在本实施例中，基于前两个实施例，对最高阈值的取值设计了自动生成和动态调整的方法。因此，在本实施例中，主要介绍最高阈值的取值以及判断的相关方法。可结合前两个实施例，对完整的任务分配方法进行理解。

如前所述，最高阈值的作用，是为了保证计算节点在分配了待分配任务后，其一种或几种被监测资源的剩余量大于或等于最高阈值，从而能够尽量满足后续任务的分配，避免资源碎片的产生。因此，最高阈值的取值，应当尽可能的满足后续任务中至少一个任务的被监测资源的需求量。

基于此，在一种实施方式中，最高阈值的取值大于或者等于为待分配任务列表中，与之对应的被监测资源的需求量最小的N个未分配任务的被监测资源的需求量的最大值。即，最高阈值的取值能够满足任务列表中被监测资源需求量最小的N个任务的被监测资源需求。其中，待分配任务列表中的未分配任务为在当前进行任务分配时尚没有进行分配的任务。例如，待分配任务列表是一个任务队列，新任务加入则该任务执行入队操作，任务分配到计算节点则该任务执行出队操作，则未分配任务为当前队列中的任务。

N 的取值可以根据具体的场景进行设定，从而使得基于最高阈值进行分配后计算节点的被监测资源的剩余量能够至少满足N个待分配任务。其中，在一种可能的实现方式中，N 可以通过预设方式进行人工配置；在另一种可能的实现方式中，N 也可以自动取待分配任务列表中未分配任务总数的 n%后取整的值，其中 n 可以根据资源碎片控制的需求进行配置。当 N 取固定值时，若 N 的取值大于了待分配任务列表中未分配任务的总数，则 N 的取值调整为未分配任务总数。

在该种实施方式中，通过最高阈值的取值，使得确定的计算节点中被监测资源的剩余量能够满足待分配任务列表中至少的N个任务的被监测资源的需求量，从而避免的资源碎片的产生。

在另一种可能的实施方式中，结合前述 S304 步骤，在判断可分配待分配任务的计算节点时，计算节点分配了待分配任务后，计算节点中任意一种被监测资源的剩余量均大于或等于与该种被监测资源相对应的最高阈值。即，在该种实施方式下，需要判断计算节点中所有的被监测资源的剩余量进行判断。

与本实施例上一种实施方式相区别的，本实施方式要求对所有被监测资源的剩余量是否大于或等于相应的最高阈值进行判断，从而保证了选择的计算节点在分配的当前待分配任务后，剩余的所有被监测资源均能够满足部分未分配任务的资源需求。

在本实施例中的一种具体的实现方式中，最高阈值的确定要使得计算节点的所有被监测资源的剩余量能够满足N个未分配任务中任何一个任务的所有被监测资源的需求量。N个未分配任务可以基于多种方式确定，例如，N个未分配任务可以基于一种被监测资源的需求量进行确定，选择一种主导资源需求量最小的N个未分配任务。则：

A、 最高阈值只需大于等于一种被监测资源需求量最小的 N 个未分配任务的剩余资源量。这种情况下，最高阈值需要大于或等于该N个任务中每个任务的相对应的被监测资源的需求量的最大值。

B、 最高阈值需要同时大于等于几种被监测资源需求量最小的 N 个任务的剩余资源量。在这种情况下，将一种资源的需求量最小的N个未分配任务视为一组，每组中的N个未分配任务的被监测资源需求量的最大值为该组任务的被监测资源的最大需求量，则最高阈值需要大于多组未分配任务的对应的被监测资源的最大需求量的最大值。

与上一实施方式类似的，N 的取值可以预设，也可以自动生成和调整，在此不再赘述。

在该种实施方式中，被监测资源的取值能够满足一种或者多种被监测资源需求量最小的N个未分配任务中任意一个任务的被监测资源的需求量。与上一实施例相比，该实施例能够避免在上一实施例中由于一个任务需要多种被监测资源而造成的不可分配问题。

例如，当前存在 CPU 和内存两种被监测资源，待分配列表中存在三个任务，按照“任务名(CPU需求量，内存需求量)”的格式表示分别为：A(5,5)、B(10，1)、C(5,2)、D(2，5)、E(1,10)。则当A任务为待分配任务时，按照上一实施例，N取2，最高阈值的取值要大于至少相应的被监测资源需求量最小的两个任务的被监测资源的需求量的最大值，则 CPU 对应的最高阈值为任务 D、E 中的最大值 2，内存对应的最高阈值为任务 B、C 中的最大值 2，但若一个计算节点剩余资源为（2,2），则无法满足 B、C、D、E 中任何一个任务。

而在本实施例中，N 取 2，以 CPU 和内存为被监测资源，当取 2 个 CPU 需求量最小的任务，则为任务D、E,当取2个内存需求量最小的任务，则为任务A、B。若最高阈值只需大于等于一种被监测资源需求量最小的 N 个未分配任务的剩余资源量，以 CPU 为例，则 CPU 和内存的最高阈值分别为任务 D、E 中 CPU 和内存的需求量的最大值 2 和 10。此时，一个计算节点剩余资源为(2,10)时，可以满足 D、E 中任意一个任务 CPU 以及内存需求。

类似的，若最高阈值需要同时大于等于几种被监测资源需求量最小的N个任务的相对应的被监测资源的剩余资源量，例如在确定 CPU 对应的最高阈值时，CPU 和内存同时为被监测资源，CPU 的需求量最小的任务D、E 的 CPU 需求量最大值 2，而内存为的需求量最小的任务 B、C 的 CPU 需求量最大值 10，则 CPU 的最高阈值为两者的最大值 10，同理得到内存的对应的最高阈值为 10。一个计算节点剩余资源为(10,10)时，可以满足 B、C、D、E 中任意一个任务的 CPU 以及内存需求

本实施例所介绍的最高阈值的确定方法，结合前述两个实施例中所述方法中被监测资源相对应的最高阈值的确定和更新。在具体的实施方式中，最高阈值的确定可以在确定可分配待分配任务的计算节点之间的各个阶段并行或者独立的进行。在每进行一次任务分配后，可以根据分配后待分配任务列表的变化对最高阈值的取值进行更新。例如，在图 4 的示例中，可以在 S401 步骤时确定最高阈值的取值。在执行 S404 后，对进行分配

后的资源的最高阈值进行更新。

在一种实施方式中，对最高阈值的更新可以在每进行一次任务分配后进行更新，具体的，可以仅对任务分配时进行分配了的被监测资源相对应的最高阈值进行更新。在另一种实现方式中，也可以在遍历了待分配任务队列后，没有可以进行分配的任务时，对被监测资源的阈值进行更新。

在本实施例中，结合前述实施例的方法，除了能达到前述效果外，由于根据待分配任务列表中的任务资源需求确定和更新被监测资源对应的最高阈值，从而保证根据最高阈值所判断的可分配待分配任务的计算节点的被监测资源的剩余量能够满足待分配任务列表中的任务，从而使得最高阈值的取值能加准确，提高算法的分配效率。

下面结合图 6 介绍本发明提供的又一实施例，与上一实施例目的类似，本实施例提供了又一种最高阈值的自动生成和动态调整的方法。因此，在本实施例中，可结合前述图 3 或图 5 所对应的两个实施例，对完整的任务分配方法进行理解。本实施例还包括:

S601、获取采样任务数据，采样任务数据包含了采样任务的资源需求信息。资源需求信息为每个任务对各类资源的需求量。

采样任务是指用以确定最高阈值的任务样本集合。采样任务是一个历史样本，可以包含多个历史任务。历史任务可以包含预先存储的历史任务，也可以包括在任务分配过程中，已经分配了的任务。采样任务数据中包括任务的用户信息，以及任务各个资源的的资源需求量或者资源实际消耗量。若采样任务数据中包含资源的实际消耗量，可以将资源的实际消耗量作为任务的资源需求信息。

S602、根据所述采样任务数据，确定至少一种资源相对应的最高阈值。

在确定最高阈值时，可以结合前一实施例中的相关描述，在前一实施例中基于待分配任务列表中的任务确定被监测资源相对应的最高阈值，在本实施例中，可以基于相类似的原理根据替采样任务可以确定被监测资源相对应的最高阈值。

在一种可能的实施方式中，所述采样任务中一种资源的需求量最小的 M 个任务作为一个最小任务集合，最小任务集中每个任务的一种被监测资源的需求量的最大值为该种被监测资源相对应的最高阈值。

在另一种可能的实施方式中，以所述采样任务中多种资源的需求量最小的 M 个任务分别作为多个最小任务集合，将多个任务集中每个任务集中每个任务的一种被监测资源的需求量的最大值作为该种被监测资源相对应的最高阈值。

本实施例所介绍的最高阈值的确定方法，结合前述两个实施例中所述方法中被监测资源相对应的最高阈值的确定和更新。在具体的实施方式中，最高阈值的确定可以在确定可分配待分配任务的计算节点之间的各个阶段并行或者独立的进行。在每进行一次任务分配后，可以根据分配后待分配任务列表的变化对最高阈值的取值进行更新。例如，在图 4 的示例中，可以在 S401 步骤时确定最高阈值的取值。在执行 S404 后，对进行分配后的资源的最高阈值进行更新。

在一种实施方式中，对最高阈值的更新可以在每进行一次任务分配后进行更新，具体的，可以仅对任务分配时进行分配了的被监测资源相对应的最高阈值进行更新。在另一种实现方式中，也可以在遍历了待分配任务队列后，没有可以进行分配的任务时，对被监测资源的阈值进行更新。

在本实施例中，结合前述实施例的方法，除了能达到前述图 3 和图 5 所对应实施例的效果外，由于根据采样任务数据确定和更新被监测资源对应的最高阈值，从而保证根据最高阈值所判断的可分配待分配任务的计算节点的被监测资源的剩余量能够满足待分配任务列表中的任务，从而使得最高阈值的取值能加准确，提高算法的分配效率。

结合图 7，是本发明实施例提供的一种管理节点 700 的逻辑结构图。本实施例基于与前述几种方法实施例的发明构思，提供了包含能够实现前述方法的功能模块的管理节点。该管理节点包括:

获取模块 701，用于获取用户的已分配的资源的份额。获取模块可用于执行前述实施例中 S301。

处理模块 702，用于从待分配任务列表中选择一个待分配任务，以及确定可分配待分配任务的计算节点。处理模块可用于执行前述实施例中 S302、S303 步骤以及 S304 中确定可分配所述待分配任务的计算节点的步骤，还可以执行 S504、S505、S506 步骤，以及前述生成和调整最高阈值的两个实施例中的方法。

分配模块 703，用于将所述待分配任务分配给一个所述可分配所述待分配任务的计算节点。分配模块可用于执行前述实施例中 S304 中分配待分配任务到计算节点的步骤以及 S507 步骤。

在一些实施方式中，该管理节点还包括采集模块 704，用于获取采样任务数据。当处理模块执行基于采样任务数据前述生成和调整最高阈值的实施例中的方法时，采集模块可用于执行 S601 步骤。

本申请实施例中对模块的划分是示意性的，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，另外，在本申请各个实施例中的各功能模块可以集成在一个处理器中，也可以是单独物理存在，也可以两个或两个以上模块集成在一个模块中。上述集成的模块既可以采用硬件的形式实现，也可以采用软件功能模块的形式实现。

通过该实施例所提供的管理节点，当通过该管理节点对分布式系统中的计算节点进行任务分配时，可以实现前述各方法实施例所述的技术效果。

图 8 说明了适用于本发明实施例的一种管理节点的系统实例。基于该实施例中的系统环境可以实现上一实施例中管理节点的各个逻辑模块的功能。该实施例只是一个适用于本发明的实例，并不试图建议对本发明所提供的管理节点的功能和结构构成任何限制。

本发明实施例以一种通用计算机系统环境作为示例来对管理节点进行说明。众所周知的，可适用于该管理节点还可以采用其他的硬件架构来实现类似的功能。包括并不限制于，个人计算机，服务计算机，多处理器系统，基于微处理器的系统，可编程消费电器，网路 PC，小型计算机，大型计算机，包括任何上述系统或设备的分布式计算环境，等等。

参照图 8，实现本发明所举例的系统包括管理节点 800 形式的通用计算设备。结合前述图 1 中所述的系统场景及架构，本实施例所述的管理节点可以为前述场景及架构中所说明的本发明实施例的执行主体。例如，可以为主节点、管理节点或者去中心化架构中的任一节点。

管理节点 800 的元件可以包括，但并不限制于，处理单元 820，系统存储器 830，和

系统总线 810。系统总线将包括系统存储器的各种系统元件与处理单元 820 相耦合。系统总线 810 可以是几种类型总线结构中的任意一种总线，这些总线可以包括存储器总线或存储器控制器，外围总线，和使用一种总线结构的局部总线。总线结构可以包括工业标准结构(ISA)总线，微通道结构(MCA)总线，扩展 ISA(EISA)总线，视频电子标准协会(VESA)局域总线，以及外围器件互联(PCI)总线。

管理节点 800 一般包括多种管理节点可读媒介。管理节点可读媒介可以是任何管理节点 800 可有效访问的媒介，并包括易失性或非易失性媒介，以及可拆卸或非拆卸的媒介。例如，但并不限制于，管理节点可读媒介可以包括管理节点存储媒介和通讯媒介。管理节点存储媒介包括易失性和非易失性，可拆卸和非拆卸媒介，这些媒介可以采用存储诸如管理节点可读指令，数据结构，程序模块或其他数据的信息的任何方法或技术来实现。管理节点存储媒介包括，但并不限制于，RAM， ROM，EEPROM，闪存存储器或其他存储器技术，或者硬盘存储、固态硬盘存储、光盘存储，磁盘盒，磁盘存储或其它存储设备，或任何其它可以存储所要求信息和能够被管理节点 800 访问的媒介。通讯媒介一般包括嵌入的计 算机可读指令，数据结构，程序模块或在模块化数据信号(例如，载波或其他传输机制)中的其他数据，并且还包括任何信息传递的媒介。术语“模块化数据信号”是指具有一个或多个信号特征组或采用对信号中的信息进行编码的方式来改变的信号。例如，但并不限制，通讯媒介包括诸如有线网络或直接有线连接的有线媒介，和诸如声，RF 红外和其它无线媒介的无线媒介。上述任何组合也应该包括在管理节点可读媒介的范围内。

系统存储器 830 包括管理节点存储媒介，它可以是易失性和非易失性存储器， 例如，只读存储器(ROM)831 和随即存取存储器(RAM)832。基本输入/输出 系统 833(BIOS)一般存储于 ROM831 中，包含着基本的例行程序，它有助于在管理节点 810 中各元件之间的信息传输。RAM 832 一般包含着数据和/或程序模块，它可以被处理单元 820 即时访问和/或立即操作。例如，但并不限制于，图 8 说明了操作系统 834，应用程序 835，其他程序模块 836 和程序数据 837。

管理节点 800 也可以包括其他可拆卸/非拆卸，易失性/非易失性的管理节点存储媒介。仅仅是一个实例，图 8 说明了硬盘存储器 841，它可以是非拆卸和非易失性的可读写磁媒介；外部存储器 851，它可以是可拆卸和非易失性的各类外部存储器，例如光盘、磁盘、闪存或者移动硬盘等；硬盘存储器 81 一般是通过非拆卸存储接口(例如，接口 840)与系统总线 810 相连接，外部存储器一般通过可拆卸存储接口(例如，接口 860) 与系统总线 810 相连接。

上述所讨论的以及图 8 所示的驱动器和它相关的管理节点存储媒介提供了管理节点可读指令，数据结构，程序模块和管理节点 800 的其它数据的存储。例如，硬盘驱动器 841 说明了用于存储操作系统 842，应用程序 843，其它程序模块 844 以及程序数据 845。值得注意的是，这些元件可以与操作系统 834，应用程序 835，其他程序模块 836，以及程序数据 837 是相同的或者是不同的。

在本实施例中，前述实施例中的方法或者上一实施例中逻辑模块的功能可以通过存储在管理节点存储媒介中的代码或者可读指令，并由处理单元 820 读取所述的代码或者可读指令从而执行所述方法。

用户可以通过各类输入设备 861 管理节点 800 输入命令和信息。各种输入设备经常都

是通过用户输入接口 860 与处理单元 820 相连接，用户输入接口 860 与系统总线相耦合，但也可以通过其他接口和总线结构相连接，例如，并行接口，或通用串行接口(USB)。显示设备 890 也可以通过接口(例如，视频接口 890)与系统总线 810 相连接。此外，诸如计算设备 800 也可以包括各类外围输出设备 820，输出设备可以通过输出接口 880 等来连接。

管理节点 800 可以在使用逻辑连接着一个或多个计算设备，例如，远程计算机 870。远程计算节点包括管理节点，计算节点，服务器， 路由器，网络 PC，等同的设备或其它通用的网络结点，并且一般包括许多或所有与管理节点 800 有关的上述所讨论的元件。结合前述图 1 所描述的架构中，远程计算节点可以是从节点、计算节点或者其他管理节点。在图 8 中所说明的逻辑连接包括局域网(LAN)和广域网 (WAN)，也可以包括其它网络。通过逻辑连接，管理节点可以与其他节点实现本发明中与其他主题之间的交互。例如，可以通过与用户的逻辑链接进行任务信息和数据的传输，从而获取用户的待分配任务；通过和计算节点的逻辑链接进行资源数据的传输以及任务分配命令的传输，从而实现各个节点的资源信息的获取以及任务的分配。

本领域技术人员应该可以意识到，在上述示例中，本发明所描述的功能可以用硬件、软件、固件或它们的任意组合来实现。当使用软件实现时，可以将这些功能存储在计算机可读介质中或者作为计算机可读介质上的一个或多个指令或代码进行传输。计算机可读介质包括计算机存储介质和通信介质，其中通信介质包括便于从一个地方向另一个地方传送计算机程序的任何介质。存储介质可以是通用或专用计算机能够存取的任何可用介质。

以上所述的具体实施方式，对本发明的目的、技术方案和有益效果进行了进一步详细说明，所应理解的是，以上所述仅为本发明的具体实施方式而已，并不用于限定本发明的保护范围，凡在本发明的技术方案的基础之上，所做的任何修改、等同替换、改进等，均应包括在本发明的保护范围之内。

权 利 要 求

1、一种分布式系统任务分配的方法，其特征在于，所述方法用于将多个用户的待分配任务分配到分布式系统中的多个计算节点中，所述方法包括:

获取用户的已分配资源的份额，所述份额为已分配给所述用户的一种资源的数量与所述资源在所述分布式系统中的可分配总量的比值，所述用户的已分配的资源中份额最高的资源为所述用户的主导资源，所述主导资源对应的份额为所述用户的主导份额;

从任务列表中选择一个待分配任务，所述待分配任务为在所述多个用户中主导份额最低的用户的任务;

若所述多个计算节点中存在第一计算节点，将所述待分配任务分配给第一计算节点，其中，所述第一计算节点为剩余资源量能够满足所述待分配任务对资源的需求量的计算节点，且所述待分配任务分配到所述第一计算节点后，所述第一计算节点中存在至少一种被监测资源，所述一种被监测资源的剩余量大于或等于与该被监测资源对应的最高阈值。

2、根据权利要求 1 所述方法，其特征在于，所述方法还包括，若所述多个计算节点中不存在第一计算节点，且存在第二计算节点，将所述待分配任务分配给第二计算节点，其中，所述第二计算节点为剩余资源量能够满足所述待分配任务对资源的需求量的计算节点，且所述待分配任务分配到所述第二节点后，所述第二节点种至少存在一种被监测资源，所述一种被监测资源的剩余量小于或等于与所述被检测资源相对应的最低阈值，所述最低阈值小于所述最高阈值。

3、根据权利要求 1 或 2 所述方法，其特征在于，所述最高阈值大于或等于所述待分配任务列表中至少一个待分配任务对所述被监测资源的需求量。

4、根据权利要求 3 所述方法，其特征在于， 所述最高阈值大于或等于所述待分配任务列表中对所述被监测资源的需求量最小的 N 个未分配任务中每个任务对所述被监测资源的需求量的最大值，其中，N 为大于或等于 1 且小于等于所述待分配任务列表中未分配任务总数的整数。

5、根据权利要求 1 或 2 所述方法，其特征在于，所述第一计算节点中任意一种所述被监测资源的剩余量均大于或等于与所述任意一种被监测资源对应的最高阈值。

6、根据权利要求 5 所述方法，其特征在于，

所述最高阈值大于或等于至少一组任务中每组任务的所述一种被监测资源的最大需求量中的最大值，所述最大需求量为一组任务中每个任务的所述一种被监测资源的需求量的最大值，所述一组任务为所述待分配任务列表中的 N 个未分配任务，N 为大于或等于 1 的整数。

7、根据权利要求 6 所述方法，其特征在于，所述一组任务具体为:

所述待分配任务列表中任意一种被监测资源需求量最小的 N 个未分配任务。

8、根据权利要求 1 或 2 所述方法，其特征在于，所述方法还包括:

获取采样任务数据，所述采样任务数据包含多个任务的对被监测资源的需求信息;

根据所述采样任务数据，确定所述至少一种被监测资源相对应的最高阈值。

9、根据权利要求 8 所述方法，其特征在于，所述根据所述采样任务数据，确定所述至少一种被监测资源相对应的最高阈值包括:

确定被监测资源 X 对应的最小任务集合的被监测资源 Y 的最大需求量为所述被监测资源 Y 相对应最高阈值，其中，被监测资源 X 为任意一种被监测资源，被监测资源 Y 为所要确定相对应的最高阈值的被监测资源，所述被监测资源 X 对应的最小任务集合为所述采样任务数据中对所述被监测资源 X 的需求量最小的 M 个任务，所述最小任务集中每个任务对被监测资源 Y 的需求量的最大值为所述最小任务集的被监测资源 Y 的最大需求量，M 为大于或等于 1 的正整数; 或者，

确定多种被监测资源对应的多个最小任务集合的被监测资源 Y 的最大需求量的最大值为所述被监测资源 Y 相对应最高阈值。

10、 根据权利要求 8 或 9 所述方法，其特征在于，所述方法还包括:

获取至少一个更新采样任务数据，所述更新采样任务数据包括预设的时间段内执行的任务的资源需求信息;

根据所述更新采样任务数据，更新至少一种资源相对应的最高阈值。

11、 一种管理节点，其特征在于，所述管理节点用于将多个用户的待分配任务分配到分布式系统中的多个计算节点中，所述管理节点包括:

获取模块，用于获取用户的已分配资源的份额，所述份额为已分配给所述用户的一种资源的数量与所述资源在所述分布式系统中的可分配总量的比值，所述用户的已分配的资源中份额最高的资源为所述用户的主导资源，所述主导资源对应的份额为所述用户的主导份额;

处理模块，用于从任务列表中选择一个待分配任务，所述待分配任务为在所述多个用户中主导份额最低的用户的任务;

分配模块，用于若所述多个计算节点中存在第一计算节点，将所述待分配任务分配给第一计算节点，其中，所述第一计算节点为剩余资源量能够满足所述待分配任务对资源的需求量的计算节点，且所述待分配任务分配到所述第一计算节点后，所述第一计算节点中存在至少一种被监测资源，所述一种被监测资源的剩余量大于或等于与该被监测资源对应的最高阈值。

12、 根据权利要求 11 所述管理节点，其特征在于，所述分配模块还用于，若所述多个计算节点中不存在第一计算节点，且存在第二计算节点，将所述待分配任务分配给第二计算节点，其中，所述第二计算节点为剩余资源量能够满足所述待分配任务对资源的需求量的计算节点，且所述待分配任务分配到所述第二节点后，所述第二节点种至少存在一种被监测资源，所述一种被监测资源的剩余量小于或等于与所述被检测资源相对应的最低阈值，所述最低阈值小于所述最高阈值。

13、 根据权力要求 11 或 12 所述管理节点，其特征在于，所述处理模块还用于确定

所述一种被检测资源相对应的最高阈值，所述最高阈值大于或等于所述待分配任务列表中至少一个待分配任务对所述被监测资源的需求量。

14、 根据权利要求 13 所述管理节点，其特征在于，所述最高阈值大于或等于所述待分配任务列表中对所述被监测资源的需求量最小的 N 个未分配任务中每个任务对所述被监测资源的需求量的最大值，其中，N 为大于或等于 1 且小于等于所述待分配任务列表中未分配任务总数的整数。

15、 根据权利要求 11 或 12 所述管理节点，其特征在于，所述所述第一计算节点具体为，第一计算节点中任意一种所述被监测资源的剩余量均大于或等于与所述任意一种被监测资源对应的最高阈值。

16、 根据权利要求 15 所述管理节点，其特征在于，所述处理模块还用于确定被监测资源相对应的最高阈值，所述最高阈值大于或等于至少一组任务中每组任务的所述一种被监测资源的最大需求量中的最大值，所述最大需求量为一组任务中每个任务的所述一种被监测资源的需求量的最大值，所述一组任务为所述待分配任务列表中的 N 个未分配任务，N 为大于或等于 1 的整数。

17、 根据权利要求 16 所述管理节点，其特征碍于，所述一组任务具体为：

所述待分配任务列表中任意一种被监测资源需求量最小的 N 个未分配任务。

18、 根据权利要求 11 或 12 所述管理节点，其特征在于，所述管理节点还包括：

采样模块，用于获取采样任务数据，所述采样任务数据包含多个任务的对被监测资源的需求信息；

所述处理模块还用于，根据所述采样任务数据，确定所述至少一种被监测资源相对应的最高阈值。

19、 根据权利要求 18 所述方法，其特征在于，所述处理模块用于根据所述采样任务数据，确定至少一种被监测资源相对应的最高阈值时具体用于，

确定被监测资源 X 对应的最小任务集合的被监测资源 Y 的最大需求量为所述被监测资源 Y 相对应最高阈值，其中，被监测资源 X 为任意一种被监测资源，被监测资源 Y 为所要确定相对应的最高阈值的被监测资源，所述被监测资源 X 对应的最小任务集合为所述采样任务数据中对所述被监测资源 X 的需求量最小的 M 个任务，所述最小任务集中每个任务对被监测资源 Y 的需求量的最大值为所述最小任务集的被监测资源 Y 的最大需求量，M 为大于或等于 1 的正整数；或者，

确定多种被监测资源对应的多个最小任务集合的被监测资源 Y 的最大需求量的最大值为所述被监测资源 Y 相对应最高阈值。

20、 根据权利要求 18 或 19 所述计算节点，其特征在于，

所述采集模块还用于，获取至少一个更新采样任务数据，所述更新采样任务数据包括预设的时间段内执行的任务的资源需求信息；

所述处理模块还用于，根据所述更新采样任务数据，更新至少一种资源相对应的最高阈值。

21、 一种分布式系统，其特征在于，所述分布式系统包括多个计算节点，所述计算节点为多个用户的待分配任务提供所需的资源以执行所述待分配任务，以及，所述分布式系统还包括：

如权利要求 11-20 中任意一项权利要求所述的管理节点。

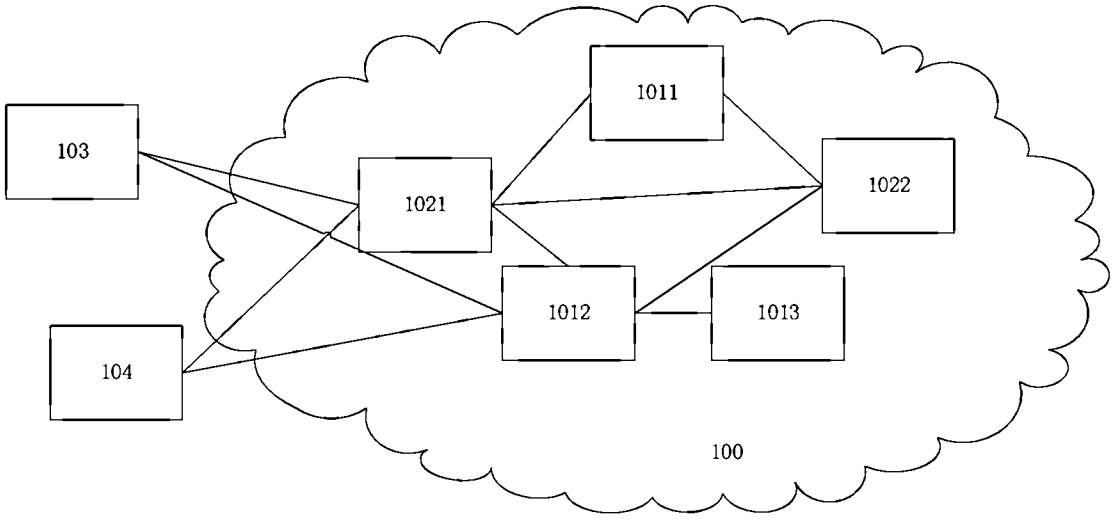
1011
103
1021
1022
1012
1013
104
100

图 1A

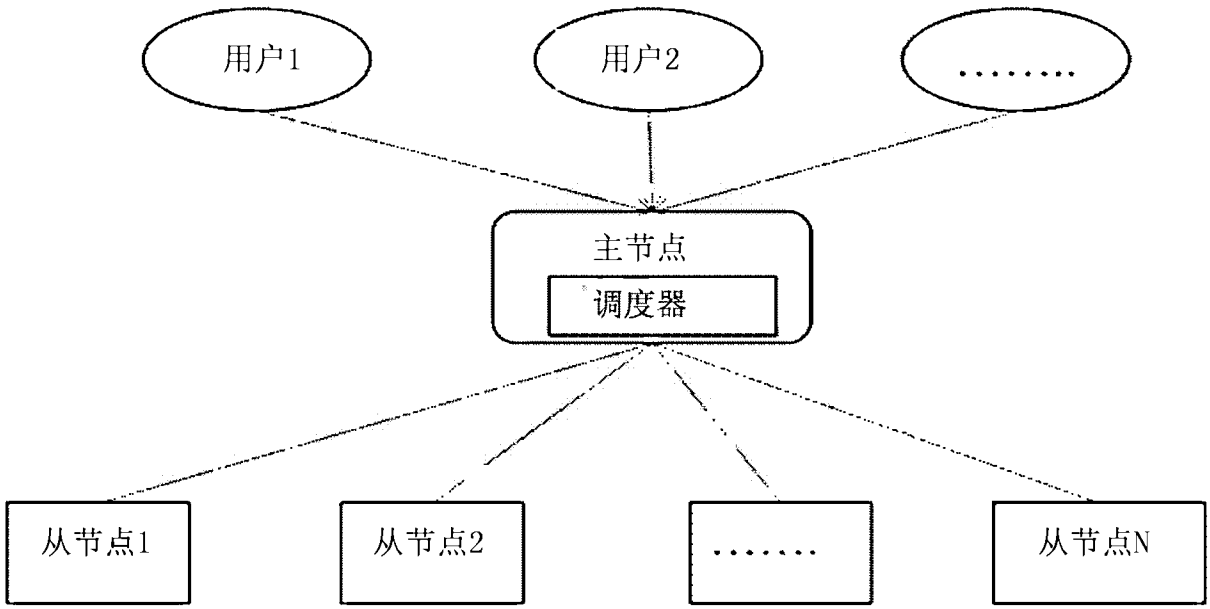
用户1
用户2
.....
主节点
调度器
从节点1
从节点2
.....
从节点N

图 1B

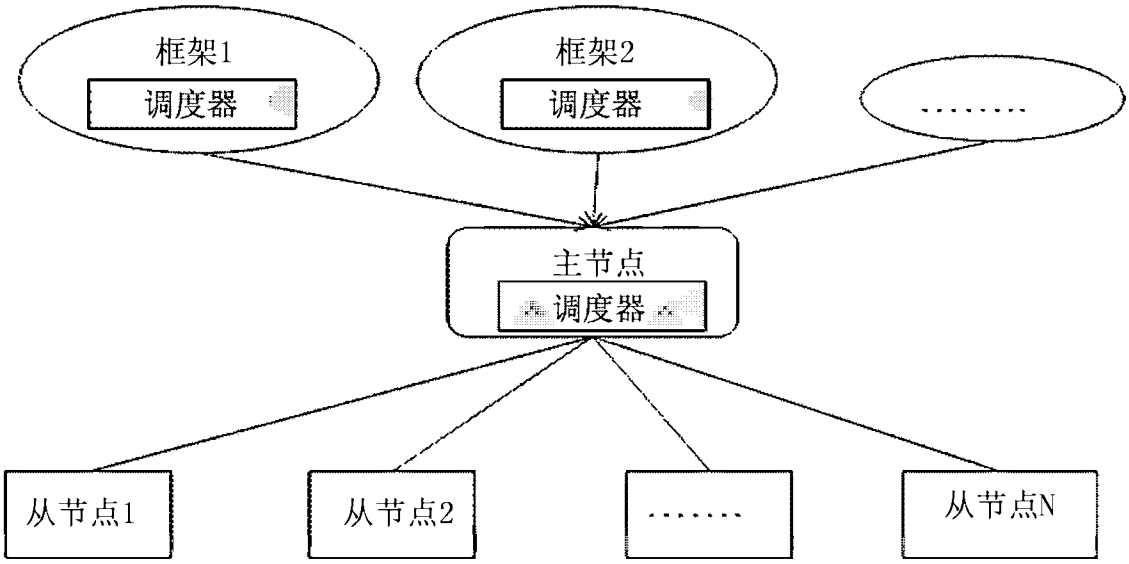
框架1
调度器
框架2
调度器
.....
主节点
调度器
从节点1
从节点2
.....
从节点N

图 1C

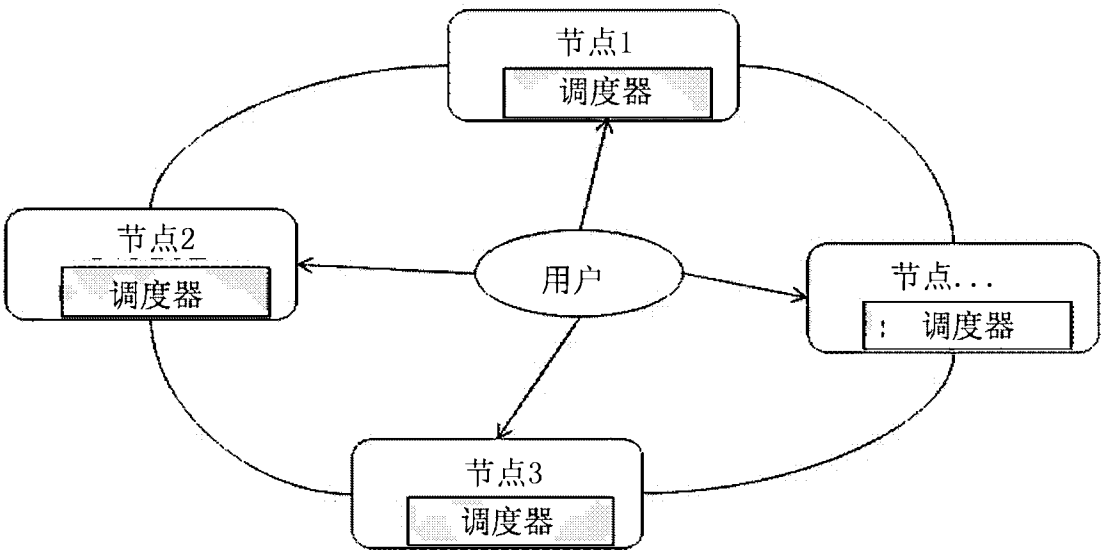
节点1
调度器
节点2
调度器
用户
节点...
调度器
节点3
调度器

图 1D

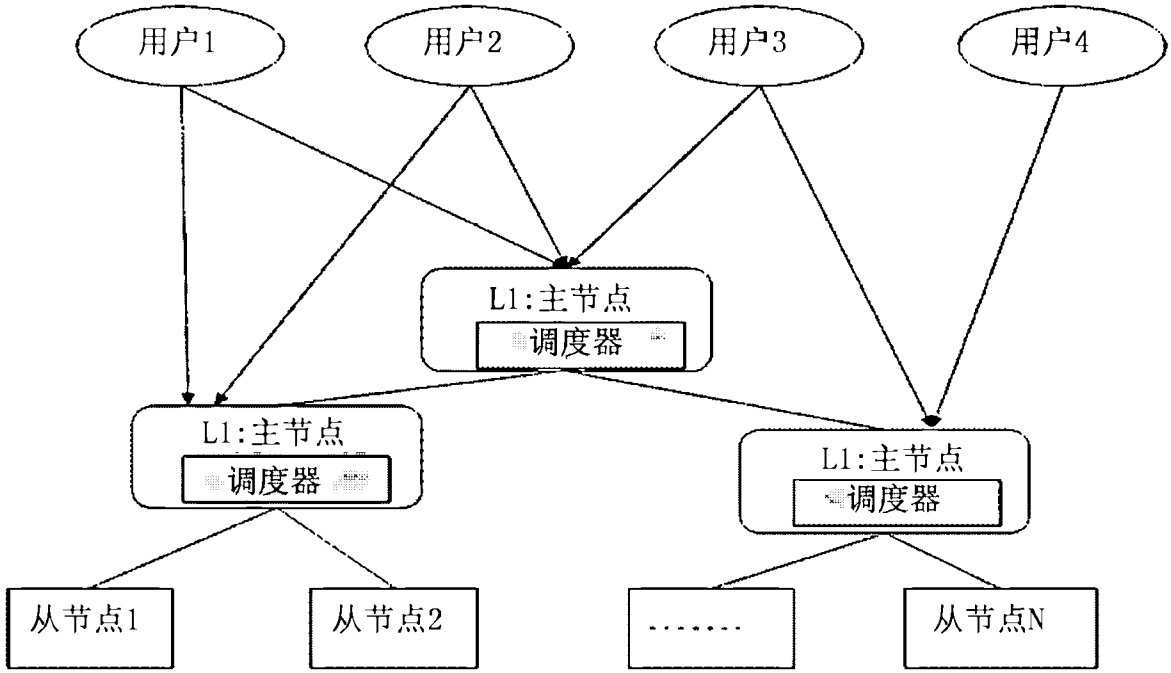
用户1
用户2
用户3
用户4
L1:主节点
调度器
L1:主节点
调度器
L1:主节点
调度器
从节点1
从节点2
.....
从节点N

图 1E

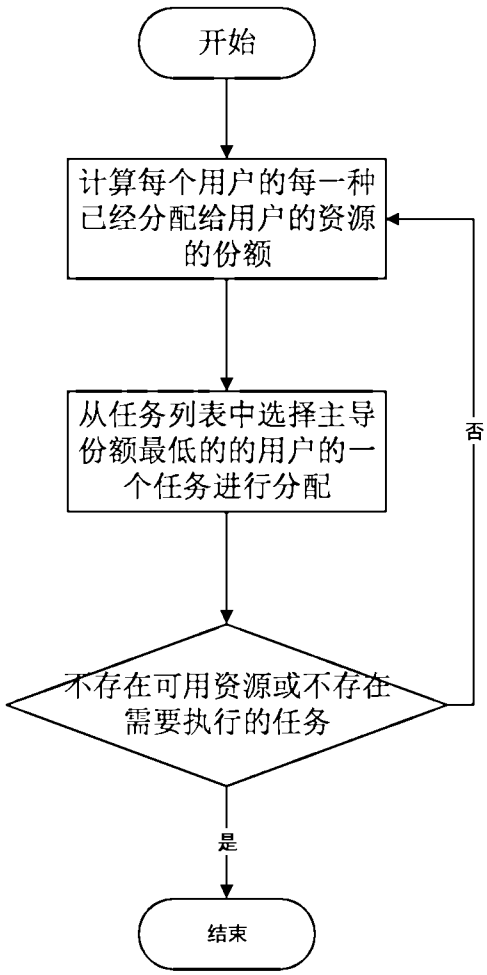
开始
计算每个用户的每一种已经分配给用户的资源的份额
从任务列表中选择主导份额最低的的用户的一个任务进行分配
不存在可用资源或不存在需要执行的任务
否
是
结束

图 2

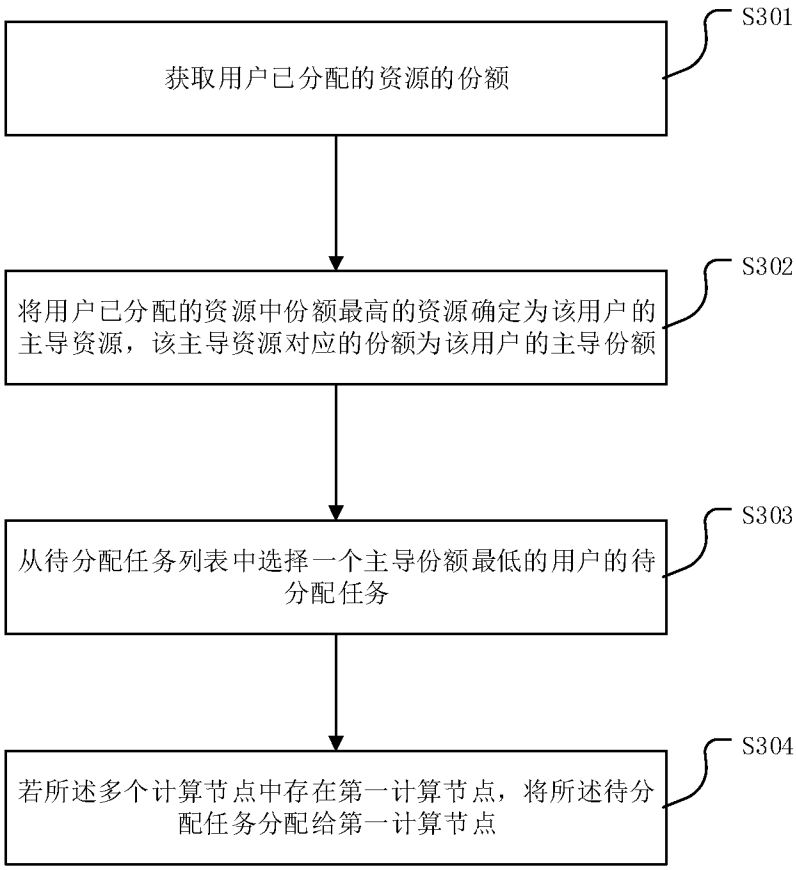
S301
获取用户已分配的资源的份额
S302
将用户已分配的资源中份额最高的资源确定为该用户的主导资源，该主导资源对应的份额为该用户的主导份额
S303
从待分配任务列表中选择一个主导份额最低的用户的待分配任务
S304
若所述多个计算节点中存在第一计算节点，将所述待分配任务分配给第一计算节点

图 3

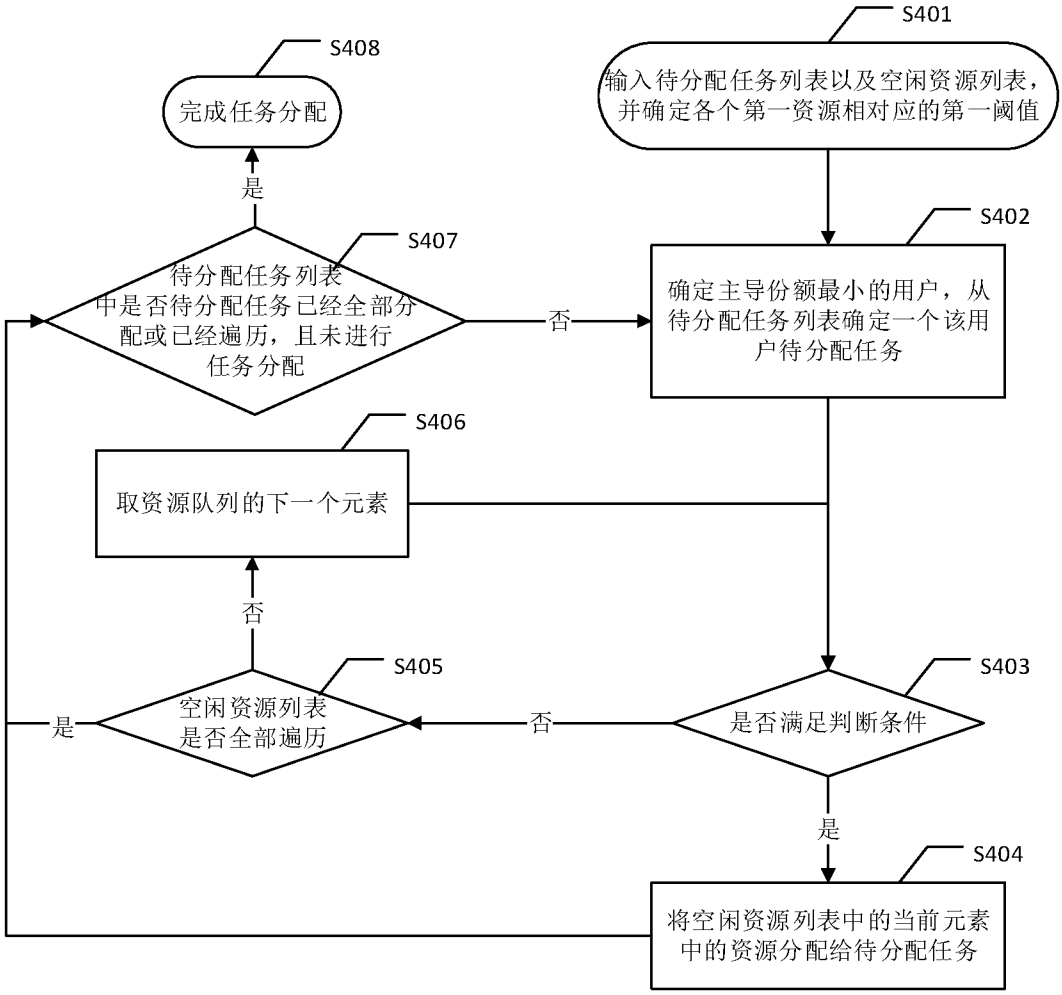
S401
输入待分配任务列表以及空闲资源列表，并确定各个第一资源相对应的第一阈值
S402
确定主导份额最小的用户，从待分配任务列表确定一个该用户待分配任务
S403
是否满足判断条件
否
S404
将空闲资源列表中的当前元素中的资源分配给待分配任务
是
S405
空闲资源列表是否全部遍历
否
S406
取资源队列的下一个元素
否
S407
待分配任务列表中是否待分配任务已经全部分配或已经遍历，且未进行任务分配
是
S408
完成任务分配
是

图 4

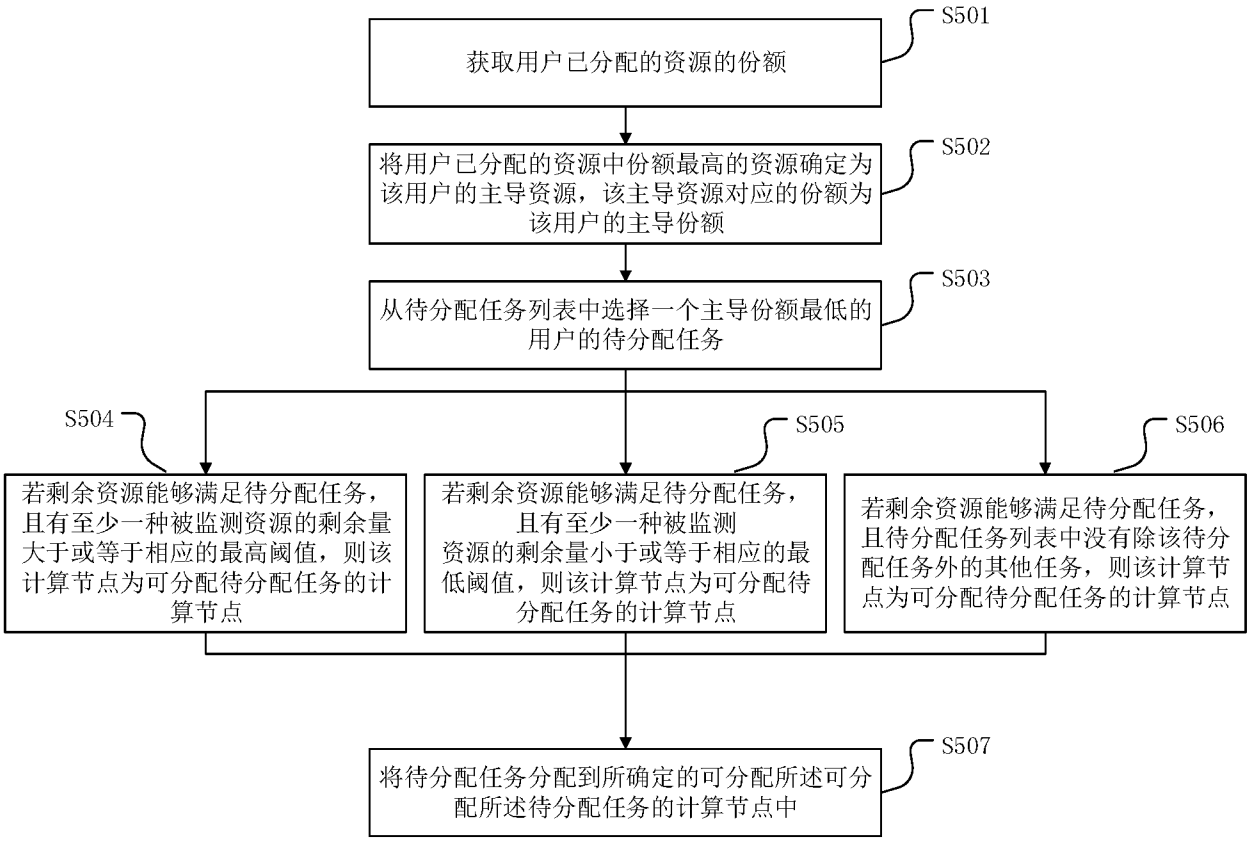
获取用户已分配的资源的份额
S501
将用户已分配的资源中份额最高的资源确定为该用户的主导资源，该主导资源对应的份额为该用户的主导份额
S502
从待分配任务列表中选择一个主导份额最低的用户的待分配任务
S503
S504
若剩余资源能够满足待分配任务，且有至少一种被监测资源的剩余量大于或等于相应的最高阈值，则该计算节点为可分配待分配任务的计算节点
S505
若剩余资源能够满足待分配任务，且有至少一种被监测资源的剩余量小于或等于相应的最低阈值，则该计算节点为可分配待分配任务的计算节点
S506
若剩余资源能够满足待分配任务，且待分配任务列表中没有除该待分配任务外的其他任务，则该计算节点为可分配待分配任务的计算节点
将待分配任务分配到所确定的可分配所述可分配所述待分配任务的计算节点中
S507

图 5

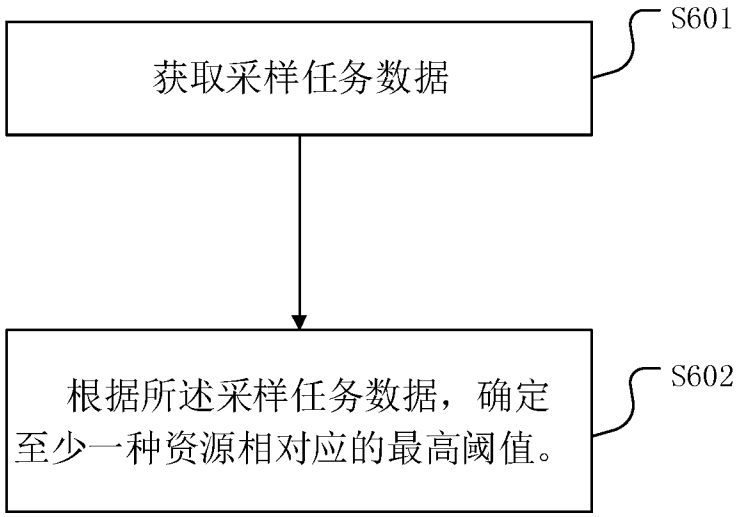
获取采样任务数据
S601
根据所述采样任务数据，确定至少一种资源相对应的最高阈值。
S602

图 6

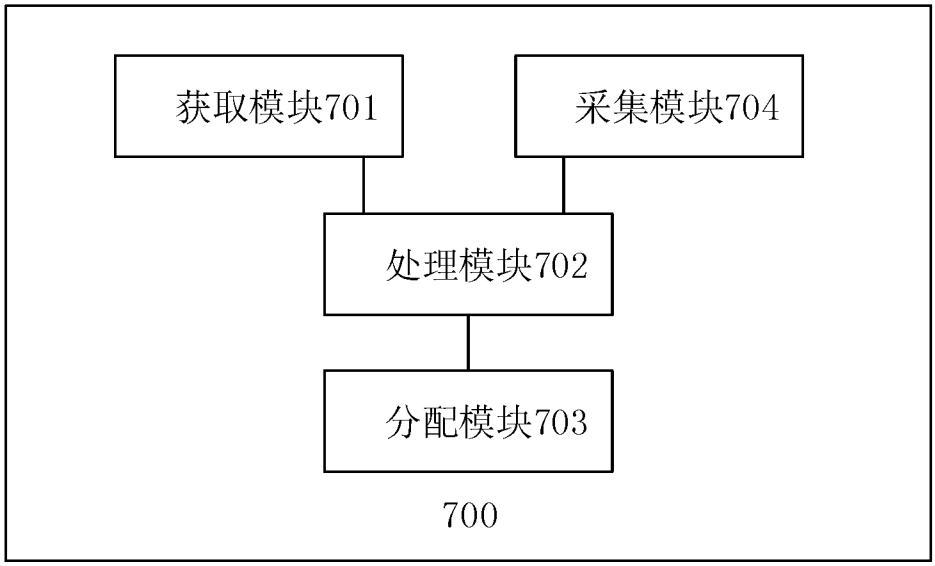
获取模块701
采集模块704
处理模块702
分配模块703
700

图 7

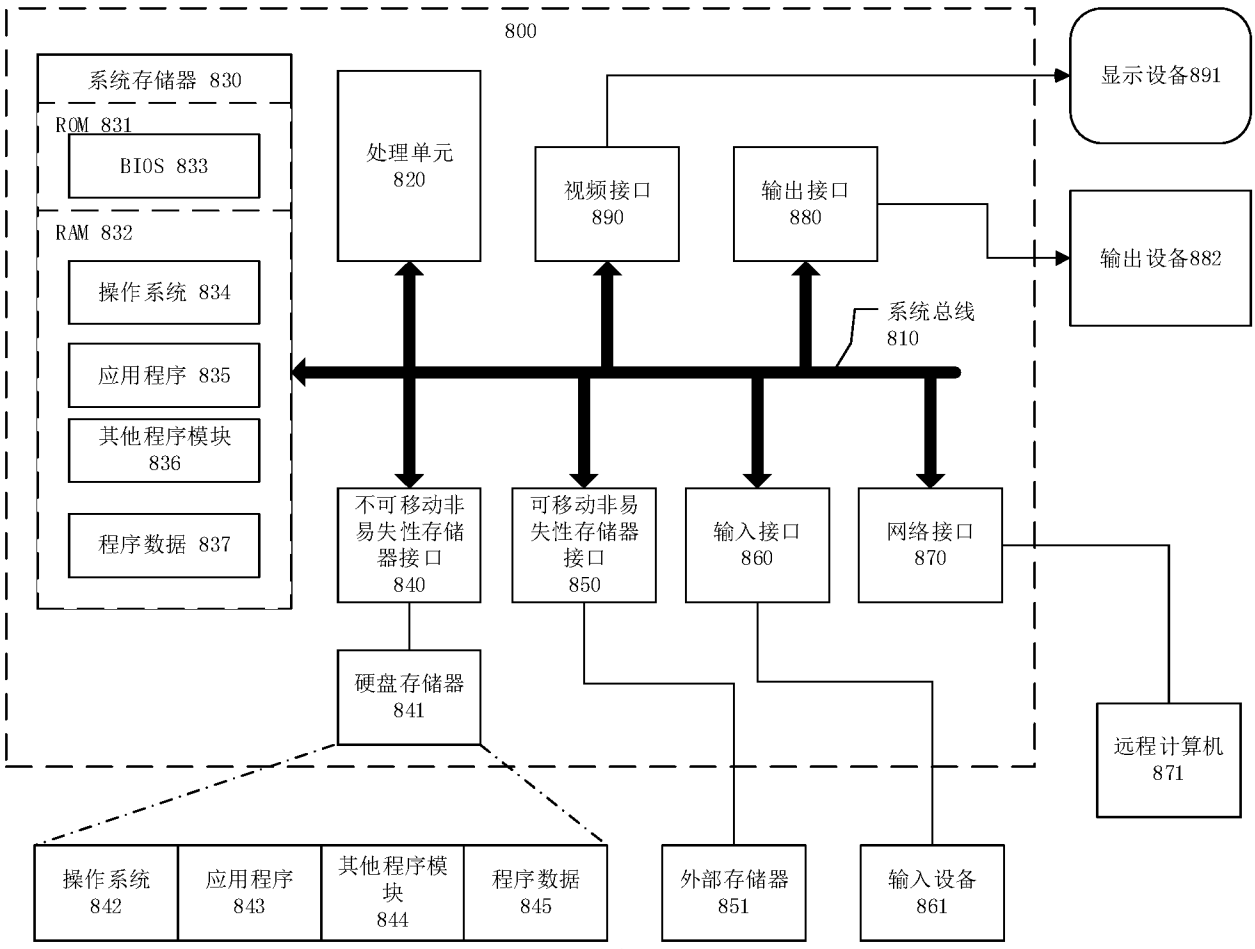
800
系统存储器 830
ROM 831
BIOS 833
RAM 832
操作系统 834
应用程序 835
其他程序模块 836
程序数据 837
处理单元 820
视频接口 890
输出接口 880
显示设备891
输出设备882
系统总线 810
不可移动非易失性存储器接口 840
可移动非易失性存储器接口 850
输入接口 860
网络接口 870
硬盘存储器 841
远程计算机 871
操作系统 842
应用程序 843
其他程序模块 844
程序数据 845
外部存储器 851
输入设备 861

图 8

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2017/106110

A. CLASSIFICATION OF SUBJECT MATTER

G06F 9/50 (2006.01) i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F 9/-

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

CNTXT; CNABS; DWPI; CKNI: 华为, 林宗芳, 朱冠宇, 曾艳, 分布式, 分派, 分配, 主导, 主要, 主, 资源, 份额, 作业, 任务, 剩余, 可用, 所剩, 碎片, 阈值, 大于, hadoop, yarn, spark, mesos, distribut+, resource, main, dominant, DRF, threshold, ratio, task, remain, more

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	CN 104881322 A (INSTITUTE OF COMPUTING TECHNOLOGY, CHINESE ACADEMY OF SCIENCES), 02 September 2015 (02.09.2015), entire document	1-21
A	US 2016350377 A1 (IBM), 01 December 2016 (01.12.2016), entire document	1-21

☐ Further documents are listed in the continuation of Box C. ☒ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
08 December 2017	22 December 2017
Name and mailing address of the ISA State Intellectual Property Office of the P. R. China No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing 100088, China Facsimile No. (86-10) 62019451	Authorized officer SUN, Lei Telephone No. (86-10) 62089291

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2017/106110

Patent Documents referred in the Report	Publication Date	Patent Family	Publication Date
CN 104881322 A	02 September 2015	None	
US 2016350377 A1	01 December 2016	WO 2016193851 A1	08 December 2016
		US 2016350376 A1	01 December 2016

国际检索报告

国际申请号
PCT/CN2017/106110

A. 主题的分类

G06F 9/50(2006.01)i

按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类

B. 检索领域

检索的最低限度文献(标明分类系统和分类号)

G06F9/-

包含在检索领域中的除最低限度文献以外的检索文献

在国际检索时查阅的电子数据库(数据库的名称，和使用的检索词（如使用）)

CNTXT;CNABS;DWPI;CKNI: 华为，林宗芳，朱冠宇，曾艳，分布式，分派，分配，主导，主要，主，资源，份额，作业，任务，剩余，可用，所剩，碎片，阈值，大于，hadoop，yarn，spark，mesos，distribut+，resource，main，dominant，DRF，threshold，ratio，task，remain，more

C. 相关文件

类 型*	引用文件，必要时，指明相关段落	相关的权利要求
A	CN 104881322 A (中国科学院计算技术研究所) 2015年 9月 2日 (2015-09-02) 全文	1-21
A	US 2016350377 A1 (IBM) 2016年 12月 1日 (2016-12-01) 全文	1-21

☐ 其余文件在C栏的续页中列出。

☑ 见同族专利附件。

* 引用文件的具体类型：

“A” 认为不特别相关的表示了现有技术一般状态的文件

“E” 在国际申请日的当天或之后公布的在先申请或专利

“L” 可能对优先权要求构成怀疑的文件，或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件（如具体说明的）

“O” 涉及口头公开、使用、展览或其他方式公开的文件

“P” 公布日先于国际申请日但迟于所要求的优先权日的文件

“T” 在申请日或优先权日之后公布，与申请不相抵触，但为了理解发明之理论或原理的在后文件

“X” 特别相关的文件，单独考虑该文件，认定要求保护的发明不是新颖的或不具有创造性

“Y” 特别相关的文件，当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时，要求保护的发明不具有创造性

“&” 同族专利的文件

国际检索实际完成的日期	国际检索报告邮寄日期
2017年 12月 8日	2017年 12月 22日

ISA/CN的名称和邮寄地址	受权官员
中华人民共和国国家知识产权局(ISA/CN) 中国北京市海淀区蓟门桥西土城路6号 100088	孙蕾
传真号 (86-10)62019451	电话号码 (86-10)62089291

国际检索报告
关于同族专利的信息

国际申请号
PCT/CN2017/106110

检索报告引用的专利文件			公布日（年/月/日）	同族专利			公布日（年/月/日）
CN	104881322	A	2015年 9月 2日	无			
US	2016350377	A1	2016年 12月 1日	WO	2016193851	A1	2016年 12月 8日
				US	2016350376	A1	2016年 12月 1日