(19)国家知识产权局

(12)发明专利

(10)授权公告号 CN 111026552 B
(45)授权公告日 2023.03.03

(21)申请号 201911251093.4
(22)申请日 2019.12.09
(65)同一申请的已公布的文献号
申请公布号 CN 111026552 A
(43)申请公布日 2020.04.17
(73)专利权人 腾讯科技(深圳)有限公司
地址 518000 广东省深圳市南山区高新区科技中一路腾讯大厦35层
(72)发明人 刘翊 张文杰 刘刚
(74)专利代理机构 北京市立方律师事务所 11330
专利代理师 张筱宁
(51)Int.Cl.
G06F 9/50(2006.01)
(56)对比文件
CN 108090225 A,2018.05.29
CN 104853338 A,2015.08.19
CN 101868037 A,2010.10.20
JP 2009211386 A,2009.09.17
CN 110297701 A,2019.10.01
CN 110162398 A,2019.08.23
CN 107343023 A,2017.11.10
CN 110362407 A,2019.10.22
杨勇等.支持大规模云服务平台的敏捷弹性伸缩技术.《华中科技大学学报(自然科学版)》.2014,第41卷
郝亮.面向能耗优化的云计算资源调度算法研究.《中国博士学位论文全文数据库信息科技辑》.2016,
腾讯技术工程.一篇文章搞懂腾讯云AI平台的人工智能IDE:TI-ONE.《https://zhuanlan.zhihu.com/p/37400202》.2018,
腾讯大数据.腾讯大数据之新一代资源管理与调度平台.《https://cloud.tencent.com/developer/article/1030007》.2018,
Mingming Lu等.Deep Learning Based Urban Post-Accidental Congestion Prediction.《2018 IEEE 15th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)》.2018,
审查员 王迪明

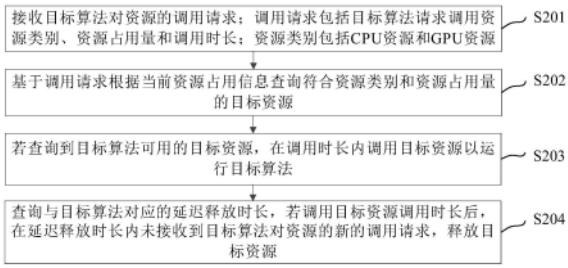
权利要求书2页 说明书11页 附图4页

(54)发明名称

资源的调度方法、装置、电子设备和计算机可读存储介质

(57)摘要

本申请涉及计算机操作系统技术领域,公开了一种资源的调度方法、装置和电子设备,资源的调度方法包括:接收目标算法对资源的调用请求;调用请求包括目标算法请求调用的资源类别、资源占用量和调用时长;资源类别包括CPU资源和GPU资源;基于调用请求根据当前资源占用信息查询符合资源类别和资源占用量的目标资源;若查询到目标算法可用的目标资源,在调用时长内调用目标资源以运行目标算法;查询与目标算法对应的延迟释放时长,若调用目标资源调用时长后,在延迟释放时长内未接收到目标算法对资源的新的调用请求,释放目标资源。本申请提供的资源的调度方法可以有效减少资源的浪费。



CN 111026552 B

1.一种资源的调度方法,其特征在于,包括:

接收目标算法对资源的调用请求;所述调用请求包括目标算法请求调用的资源类别、资源占用量和调用时长;所述资源类别包括CPU资源和GPU资源;

基于所述调用请求根据当前资源占用信息查询符合所述资源类别和所述资源占用量的目标资源;

若查询到所述目标算法可用的所述目标资源,在所述调用时长内调用所述目标资源以运行所述目标算法;

确定目标算法的调用类别;调用类别包括占用不释放类别和占用后释放类别;

若目标算法的调用类别为占用后释放类别,查询与所述目标算法对应的延迟释放时长,若调用所述目标资源所述调用时长后,在延迟释放时长内未接收到所述目标算法对资源的新的调用请求,释放所述目标资源;

若所述目标算法的调用类别为占用不释放类别,调用所述目标资源以运行所述目标算法,并持续调用所述目标资源以等待接收所述目标算法的新的调用请求。

2.根据权利要求1所述的资源的调度方法,其特征在于,还包括:

若未查询到所述目标算法可用的所述目标资源,将所述调用请求存储于请求队列中;

针对于所述请求队列中的每一个调用请求,执行所述基于所述调用请求在当前资源占用信息中查询符合所述资源类别和所述资源占用量的目标资源。

3.根据权利要求2所述的资源的调度方法,其特征在于,所述针对于所述请求队列中的每一个调用请求,执行所述基于所述调用请求在当前资源占用信息中查询符合所述资源类别和所述资源占用量的目标资源,包括:

根据每一个调用请求的接收时间,对请求队列中的至少一个调用请求进行排序;

基于对调用请求的排序,依次执行基于所述调用请求在当前资源占用信息中查询符合所述资源类别和所述资源占用量的目标资源。

4.根据权利要求2所述的资源的调度方法,其特征在于,所述针对于所述请求队列中的每一个调用请求,执行所述基于所述调用请求在当前资源占用信息中查询符合所述资源类别和所述资源占用量的目标资源,包括:

查询每一个调用请求对应的目标算法的紧急调用级别,依据所述紧急调用级别对请求队列中的至少一个调用请求排序;

基于对调用请求的排序,依次执行基于调用请求在当前资源占用信息中查询符合所述资源类别和所述资源占用量的目标资源。

5.根据权利要求1-4任一项所述的资源的调度方法,其特征在于,所述在所述调用时长内调用所述目标资源以运行所述目标算法之后,还包括:

根据所述目标资源的资源类别、资源占用量和调用时长,对所述当前资源占用信息进行更新。

6.一种资源的调度装置,其特征在于,包括:

接收模块,用于接收目标算法对资源的调用请求;所述调用请求包括目标算法请求调用的资源类别、资源占用量和调用时长;所述资源类别包括CPU资源和GPU资源;

查询模块,用于基于所述调用请求在当前资源占用信息中查询符合所述资源类别和所述资源占用量的目标资源;

第一调用模块，用于若查询到所述目标算法可用的所述目标资源，在所述调用时长内调用所述目标资源以运行所述目标算法；

释放模块，用于确定目标算法的调用类别；调用类别包括占用不释放类别和占用后释放类别；若目标算法的调用类别为占用后释放类别，查询与所述目标算法对应的延迟释放时长，若调用所述目标资源所述调用时长后，在延迟释放时长内未接收到所述目标算法对资源的新的调用请求，释放所述目标资源；若所述目标算法的调用类别为占用不释放类别，调用所述目标资源以运行所述目标算法，并持续调用所述目标资源以等待接收所述目标算法的新的调用请求。

7.一种电子设备，包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序，其特征在于，所述处理器执行所述程序时实现权利要求1-5任一项所述的资源的调度方法。

8.一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上存储有计算机程序，该程序被处理器执行时实现权利要求1-5任一项所述的资源的调度方法。

资源的调度方法、装置、电子设备和计算机可读存储介质

技术领域

[0001] 本申请涉及计算机操作系统技术领域，具体而言，本申请涉及一种资源的调度方法、装置、电子设备及计算机可读存储介质。

背景技术

[0002] 在计算机操作系统中，CPU(中央处理器，central processing unit)和GPU(图像处理器，Graphics Processing Unit)往往需要被多种任务共享。

[0003] 通常，需要根据可能被用到的算法或模型规划布局大量的CPU和GPU资源，但只有少量的算法或服务需要长期占用资源，例如，针对AI(人工智能，Artificial Intelligence)推理系统(模型部署平台，为训练好的AI模型提供部署平台)、ADC(集中托管式数据应用中心，Application data center)应用系统或AI训练系统等多种算法，其中只有部分算法需要长期占用资源，另一些算法调用资源的时间是随机不确定的，这使得用户必须提供充足的资源准备随时被调用，但是这会造成资源的浪费。

[0004] 因此，有必要提供一种资源调度的方法，可以减少资源的浪费。

发明内容

[0005] 本申请的目的旨在至少能解决上述的技术缺陷之一，特提出以下技术方案：

[0006] 第一方面，提供了一种资源的调度方法，包括：

[0007] 接收目标算法对资源的调用请求；调用请求包括目标算法请求调用的资源类别、资源占用量和调用时长；资源类别包括CPU资源和GPU资源；

[0008] 基于调用请求根据当前资源占用信息查询符合资源类别和资源占用量的目标资源；

[0009] 若查询到目标算法可用的目标资源，在调用时长内调用目标资源以运行目标算法；

[0010] 查询与目标算法对应的延迟释放时长，若调用目标资源调用时长后，在延迟释放时长内未接收到目标算法对资源的新的调用请求，释放目标资源。

[0011] 在第一方面的可选实施例中，资源的调度方法还包括：

[0012] 若未查询到目标算法可用的目标资源，将调用请求存储于请求队列中；

[0013] 针对于请求队列中的每一个调用请求，执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源。

[0014] 在第一方面的可选实施例中，针对于请求队列中的每一个调用请求，执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源，包括：

[0015] 根据每一个调用请求的接收时间，对请求队列中的至少一个调用请求进行排序；

[0016] 基于对调用请求的排序，依次执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源。

[0017] 在第一方面的可选实施例中，针对于请求队列中的每一个调用请求，执行基于调

用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源，包括：

[0018] 查询每一个调用请求对应的目标算法的紧急调用级别，依据紧急调用级别对请求队列中的至少一个调用请求排序；

[0019] 基于对调用请求的排序，依次执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源。

[0020] 在第一方面的可选实施例中，查询与目标算法对应的延迟释放时长之前，还包括：

[0021] 确定目标算法的调用类别；调用类别包括占用不释放类别和占用后释放类别；

[0022] 查询与目标算法对应的延迟释放时长，包括：

[0023] 若目标算法的调用类别为占用后释放类别，查询与目标算法对应的延迟释放时长。

[0024] 在第一方面的可选实施例中，资源的调度方法还包括：

[0025] 若目标算法的调用类别为占用不释放类别，调用目标资源以运行目标算法，并持续调用目标资源以等待接收目标算法的新的调用请求。

[0026] 在第一方面的可选实施例中，在调用时长内调用目标资源以运行目标算法之后，还包括：

[0027] 根据目标资源的资源类别、资源占用量和调用时长，对当前资源占用信息进行更新。

[0028] 第二方面，提供了一种资源的调度装置，包括：

[0029] 接收模块，用于接收目标算法对资源的调用请求；调用请求包括目标算法请求调用的资源类别、资源占用量和调用时长；资源类别包括CPU资源和GPU资源；

[0030] 查询模块，用于基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源；

[0031] 第一调用模块，用于若查询到目标算法可用的目标资源，在调用时长内调用目标资源以运行目标算法；

[0032] 释放模块，用于查询与目标算法对应的延迟释放时长，若调用目标资源调用时长后，在延迟释放时长内未接收到目标算法对资源的新的调用请求，释放目标资源。

[0033] 在第二方面的可选实施例中，资源的调度装置还包括存储模块，存储模块用于：

[0034] 若未查询到目标算法可用的目标资源，将调用请求存储于请求队列中；

[0035] 针对于请求队列中的每一个调用请求，执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源。

[0036] 在第二方面的可选实施例中，存储模块在针对于请求队列中的每一个调用请求，执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源时，具体用于：

[0037] 根据每一个调用请求的接收时间，对请求队列中的至少一个调用请求进行排序；

[0038] 基于对调用请求的排序，依次执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源。

[0039] 在第二方面的可选实施例中，存储模块在针对于请求队列中的每一个调用请求，执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源时，具体用于：

[0040] 查询每一个调用请求对应的目标算法的紧急调用级别，依据紧急调用级别对请求队列中的至少一个调用请求排序；

[0041] 基于对调用请求的排序，依次执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源。

[0042] 在第二方面的可选实施例中，资源的调度装置还包括确定模块，确定模块用于：

[0043] 确定目标算法的调用类别；调用类别包括占用不释放类别和占用后释放类别；

[0044] 释放模块在查询与目标算法对应的延迟释放时长时，具体用于：

[0045] 若目标算法的调用类别为占用后释放类别，查询与目标算法对应的延迟释放时长。

[0046] 在第二方面的可选实施例中，资源的调度装置还包括第二调用模块，第二调用模块用于：

[0047] 若目标算法的调用类别为占用不释放类别，调用目标资源以运行目标算法，并持续调用目标资源以等待接收目标算法的新的调用请求。

[0048] 在第二方面的可选实施例中，资源的调度装置还包括更新模块，更新模块用于：

[0049] 根据目标资源的资源类别、资源占用量和调用时长，对当前资源占用信息进行更新。

[0050] 第三方面，提供了一种电子设备，包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序，处理器执行程序时实现本申请第一方面所示的资源的调度方法。

[0051] 第四方面，提供了一种计算机可读存储介质，计算机可读存储介质上存储有计算机程序，该程序被处理器执行时实现本申请第一方面所示的资源的调度方法。

[0052] 本申请提供的技术方案带来的有益效果是：

[0053] 当接收到目标算法对资源的调用请求时，调用目标资源一定的调用时长以运行目标算法，并查询目标算法的延迟释放时长，若调用目标资源调用时长后，在延迟释放时长内未接收到目标算法对资源的新的调用请求，释放目标资源，可以在满足目标算法对目标资源的使用，并且在调用目标资源后，及时释放目标资源，以提供给其他算法调用，有效减少资源的浪费。

[0054] 进一步的，可以先确定目标算法的调用类别，对于占用不释放类别的目标算法，即便根据调用时长运行完，也不用释放目标算法，可以满足这类别的目标算法的长期调用目标资源的要求。

[0055] 本申请附加的方面和优点将在下面的描述中部分给出，这些将从下面的描述中变得明显，或通过本申请的实践了解到。

附图说明

[0056] 本申请上述的和/或附加的方面和优点从下面结合附图对实施例的描述中将变得明显和容易理解，其中：

[0057] 图1为本申请实施例提供的一种资源的调度方法的应用环境图；

[0058] 图2为本申请实施例提供的一种资源的调度方法的流程示意图；

[0059] 图3为本申请实施例提供的一种资源的调度方法的流程示意图；

[0060] 图4为本申请实施例提供的一种请求队列的示意图；

[0061] 图5为本申请一个示例中提供的一种资源的调度方法的流程示意图；

[0062] 图6为本申请实施例提供的一种资源的调度装置的结构示意图；

[0063] 图7为本申请实施例提供的一种资源的调度的电子设备的结构示意图。

具体实施方式

[0064] 下面详细描述本申请的实施例，实施例的示例在附图中示出，其中自始至终相同或类似的标号表示相同或类似的元件或具有相同或类似功能的元件。下面通过参考附图描述的实施例是示例性的，仅用于解释本申请，而不能解释为对本申请的限制。

[0065] 本技术领域技术人员可以理解，除非特意声明，这里使用的单数形式“一”、“一个”、“所述”和“该”也可包括复数形式。应该进一步理解的是，本申请的说明书中使用的措辞“包括”是指存在特征、整数、步骤、操作、元件和/或组件，但是并不排除存在或添加一个或多个其他特征、整数、步骤、操作、元件、组件和/或它们的组。应该理解，当我们称元件被“连接”或“耦接”到另一元件时，它可以直接连接或耦接到其他元件，或者也可以存在中间元件。此外，这里使用的“连接”或“耦接”可以包括无线连接或无线耦接。这里使用的措辞“和/或”包括一个或更多个相关联的列出项的全部或任一单元和全部组合。

[0066] 为使本申请的目的、技术方案和优点更加清楚，下面将结合附图对本申请实施方式作进一步地详细描述。

[0067] 本申请提供的资源的调度方法、装置、电子设备及计算机可读存储介质，旨在解决现有技术的如上技术问题。

[0068] 下面以具体地实施例对本申请的技术方案以及本申请的技术方案如何解决上述技术问题进行详细说明。下面这几个具体的实施例可以相互结合，对于相同或相似的概念或过程可能在某些实施例中不再赘述。下面将结合附图，对本申请的实施例进行描述。

[0069] 本申请提供的资源的调度方法，可以应用于如图1所示的应用环境中。具体的，服务器或终端中设置有资源的调度系统，资源类别包括CPU资源和GPU资源；当资源的调度系统接收到目标算法的调用请求时，查询可用的目标资源，若查询到符合调用请求的目标资源，在调用时长内调用目标资源以运行目标算法；若调用目标资源调用时长后，在延迟释放时长内未接收到目标算法对资源的新的调用请求，释放目标资源。

[0070] 本申请实施例中提供了一种可能的实现方式，如图2所示，提供了一种资源的调度方法，可以应用于图1所示的资源的调度系统中，可以包括以下步骤：

[0071] 步骤S201，接收目标算法对资源的调用请求；调用请求包括目标算法请求调用的资源类别、资源占用量和调用时长；资源类别包括CPU资源和GPU资源。

[0072] 其中，目标算法可以包括多种需要调用CPU资源或GPU资源的算法，也可以包括需要调用资源的模型、服务等等。

[0073] 例如，目标算法可以包括多种AI推理模型、ADC应用模型或AI训练模型。

[0074] 具体的，资源占用量是指需要调用的资源容量，调用时长是指调用资源的时长。

[0075] 在具体实施过程中，资源的调度系统通过统一的访问地址接收调用请求，例如采用HTTP协议，固定统一的访问地址，将统一的访问地址注册于各个不同算法或模型的系统中，便于各个不同的算法或模型请求调用资源。

[0076] 步骤S202,基于调用请求根据当前资源占用信息查询符合资源类别和资源占用量的目标资源。

[0077] 具体的,资源的调度系统中实时对当前资源占用信息进行监测,当前资源占用信息可以包括不同类型的资源的正在调用量、剩余调用时长、可用资源量等等。

[0078] 例如,当前资源占用信息可以包括总共资源有3块CPU资源,其中2块正在被调用,被调用的2块CPU中1块还需要被调用5分钟,另一块还需要被调用10分钟,3块CPU中还有一块是可用资源。

[0079] 步骤S203,若查询到目标算法可用的目标资源,在调用时长内调用目标资源以运行目标算法。

[0080] 具体的,可以根据当前资源占用信息查询可用资源量,判断目标算法请求调用的资源类型中可用资源量是否不小于请求调用的资源占用量;若请求调用的资源类型中可用资源量满足目标算法请求调用的资源占用量,则查询到目标算法可用的目标资源。

[0081] 在具体实施过程中,若资源的调度系统查询到目标算法可用的目标资源,可以是资源的调度系统调用目标资源,然后将目标资源供应与目标算法运行;也可以是资源的调度系统查询到目标算法可用的目标资源,目标算法直接调用目标资源。

[0082] 步骤S204,查询与目标算法对应的延迟释放时长,若调用目标资源调用时长后,在延迟释放时长内未接收到目标算法对资源的新的调用请求,释放目标资源。

[0083] 其中,释放的目标资源包括所调用的资源,还可以包括所占用的内存。

[0084] 具体的,延迟释放时长是指目标算法在调用目标资源运行调用时长后,继续保持调用目标资源的时间,设置延迟释放时长,可以避免需要频繁调用资源的目标算法,在运行完后直接释放资源,但又需要重新调用目标资源,导致操作繁琐。

[0085] 若调用目标资源调用时长后,在延迟释放时长内接收到目标算法对资源的新的调用请求,则重复调用目标资源调用时长,然后继续等待延迟释放时长,直至在调用目标资源调用时长后,在延迟释放时长内未接收到目标算法对资源的新的调用请求,释放目标资源。

[0086] 在具体实施过程中,资源的调用系统可以直接释放目标资源,也可以控制目标算法释放目标资源,具体针对目标资源的释放方式在此不作限制。

[0087] 上述的资源的调度方法,当接收到目标算法对资源的调用请求时,调用目标资源一定的调用时长以运行目标算法,并查询目标算法的延迟释放时长,若调用目标资源调用时长后,在延迟释放时长内未接收到目标算法对资源的新的调用请求,释放目标资源,可以在满足目标算法对目标资源的使用,并且在调用目标资源后,及时释放目标资源,以提供给其他算法调用,有效减少资源的浪费。

[0088] 本申请实施例中提供了一种可能的实现方式,如图3所示,资源的调度方法还可以包括:

[0089] 步骤S205,若未查询到目标算法可用的目标资源,将调用请求存储于请求队列中。

[0090] 具体的,可以根据当前资源占用信息查询可用资源量,判断目标算法请求调用的资源类型中可用资源量是否不小于请求调用的资源占用量;若请求调用的资源类型中可用资源量不满足目标算法请求调用的资源占用量,则未查询到目标算法可用的目标资源。

[0091] 步骤S206,针对于请求队列中的每一个调用请求,执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源。

[0092] 具体的，请求队列中可以包括至少一个调用请求，当请求队列中包括至少两个调用请求时，可以按照对调用请求的接收时间对多个调用请求进行排序，也可以按照调用请求的紧急程度对调用请求进行排序，对请求队列中的至少两个调用请求，依次执行查询可调用的目标资源，并且可以循环执行查询任务。

[0093] 如图4所示，请求队列中存储有N个算法发送的调用请求，其中，N为大于2的自然数，资源的调度系统可以按照队列中调用请求的顺序，依次查询与调用请求1、调用请求2……直至调用请求N对应的资源，可以在查询完整个请求队列后，循环从调用请求1开始继续查询，查询到符合任一个调用请求的资源，即调用该资源供对应的算法运行。

[0094] 在一种实施过程中，步骤S206的针对于请求队列中的每一个调用请求，执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源，可以包括：

[0095] (1)根据每一个调用请求的接收时间，对请求队列中的至少一个调用请求进行排序；

[0096] (2)基于对调用请求的排序，依次执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源。

[0097] 具体的，可以根据请求队列中的每一个调用请求的接收时间，将请求队列中所包括的所有调用请求进行排序。

[0098] 例如，先接收到的调用请求，先执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源。

[0099] 在另一种实施过程中，步骤S206的针对于请求队列中的每一个调用请求，执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源，可以包括：

[0100] (1)查询每一个调用请求对应的目标算法的紧急调用级别，依据紧急调用级别对请求队列中的至少一个调用请求排序；

[0101] (2)基于对调用请求的排序，依次执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源。

[0102] 具体的，也可以根据目标算法的紧急调用级别对请求队列中的所有调用请求进行排序，紧急调用级别越高，对应的目标算法越重要、越紧急，可以先执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源。

[0103] 本申请实施例中提供了一种可能的实现方式，步骤S204的查询与目标算法对应的延迟释放时长之前，还可以包括：确定目标算法的调用类别；调用类别包括占用不释放类别和占用后释放类别。

[0104] 其中，占用不释放类别是指目标算法调用目标资源后，即便根据调用时长运行完，也不用释放目标算法，这种类别的算法需要调用目标资源较频繁，或者目标算法本身需要持续调用目标资源不能间断。

[0105] 本申请实施例中提供了一种可能的实现方式，步骤S204的查询与目标算法对应的延迟释放时长，可以包括：若目标算法的调用类别为占用后释放类别，查询与目标算法对应的延迟释放时长。

[0106] 具体的，占用后释放类别是指目标算法调用目标资源运行调用时长后，需要释放目标资源，这类目标算法对于目标资源的调用不太频繁，不需要持续调用目标资源不间断，

因此调用完后，如果在延迟释放时长内未接受到目标算法的新的调用申请，资源的调用系统可以释放目标资源。

[0107] 本申请实施例中提供了一种可能的实现方式，资源的调度方法还可以包括：

[0108] 若目标算法的调用类别为占用不释放类别，调用目标资源以运行目标算法，并持续调用目标资源以等待接收目标算法的新的调用请求。

[0109] 具体的，若目标算法的调用类别为占用不释放类别，即算法需要调用目标资源较频繁，或者目标算法本身需要持续调用目标资源不能间断，在调用时长内调用目标资源以运行目标算法后，保持持续调用目标资源，等待接收目标算法的新的调用请求，并重复依据新的调用请求运行目标算法。

[0110] 本申请实施例中提供了一种可能的实现方式，步骤S03的在调用时长内调用目标资源以运行目标算法之后，还可以包括：

[0111] 根据目标资源的资源类别、资源占用量和调用时长，对当前资源占用信息进行更新。

[0112] 具体的，若查询到目标资源，需要对当前资源占用信息进行更新，以便根据下一个目标算法的调用请求查询更新后的资源占用信息。

[0113] 可以理解的是，当调用目标资源调用时长后，在延迟释放时长内未接收到目标算法对资源的新的调用请求，释放目标资源后，同样需要根据所释放的目标资源的资源类别、资源占用量和调用时长，对更新后的资源占用信息再次更新。

[0114] 本申请实施例中提供了一种可能的实现方式，若请求队列中的调用请求的数量超过预设阈值，或者请求队列中的多个调用请求持续保持等待的时间超过预设时间，还可以发送超负荷提醒到用户终端，提醒用户终端进行资源的扩增。

[0115] 上述的资源的调度方法，当接收到目标算法对资源的调用请求时，调用目标资源一定的调用时长以运行目标算法，并查询目标算法的延迟释放时长，若调用目标资源调用时长后，在延迟释放时长内未接收到目标算法对资源的新的调用请求，释放目标资源，可以在满足目标算法对目标资源的使用，并且在调用目标资源后，及时释放目标资源，以提供给其他算法调用，有效减少资源的浪费。

[0116] 进一步的，可以先确定目标算法的调用类别，对于占用不释放类别的目标算法，即便根据调用时长运行完，也不用释放目标算法，可以满足这类别的目标算法的长期调用目标资源的要求。

[0117] 为了便于理解，以下将结合具体示例详细阐述本发明的资源的调度方法：

[0118] 在一个示例中，如图5所示，本申请提供的资源的调度方法，包括如下步骤：

[0119] 步骤S501，接收目标算法对资源的调用请求；

[0120] 步骤S502，根据当前资源占用信息查询是否存在目标算法可用的目标资源；若存在，执行步骤S503；若不存在，执行步骤S508；

[0121] 步骤S503，在调用时长内调用目标资源以运行目标算法；

[0122] 步骤S504，确定目标算法的调用类别；若调用类别为占用后释放类别，执行步骤S505；若调用类别为占用不释放类别，执行步骤S507；

[0123] 步骤S505，查询与目标算法对应的延迟释放时长；

[0124] 步骤S506，若调用目标资源调用时长后，在延迟释放时长内未接收到目标算法对

资源的新的调用请求，释放目标资源；

[0125] 步骤S507，调用目标资源以运行目标算法，并持续调用目标资源以等待接收目标算法的新的调用请求；

[0126] 步骤S508，将调用请求存储于请求队列中；

[0127] 步骤S509，针对请求队列中的每一个调用请求，执行步骤S502。

[0128] 上述示例中，当接收目标算法对资源的调用请求时，根据当前资源占用信息查询是否存在目标算法可用的目标资源；若存在，调用目标资源一定的调用时长以运行目标算法；确定目标算法的调用类别；若调用类别为占用后释放类别，查询与目标算法对应的延迟释放时长；调用目标资源调用时长后，在延迟释放时长内未接收到目标算法对资源的新的调用请求，释放目标资源，可以及时释放目标资源，以提供给其他算法调用，有效减少资源的浪费。

[0129] 本申请实施例中提供了一种可能的实现方式，如图6所示，提供了一种资源的调度装置60，包括接收模块601、查询模块602、第一调用模块603和释放模块604，其中，

[0130] 接收模块601，用于接收目标算法对资源的调用请求；调用请求包括目标算法请求调用的资源类别、资源占用量和调用时长；资源类别包括CPU资源和GPU资源；

[0131] 查询模块602，用于基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源；

[0132] 第一调用模块603，用于若查询到目标算法可用的目标资源，在调用时长内调用目标资源以运行目标算法；

[0133] 释放模块604，用于查询与目标算法对应的延迟释放时长，若调用目标资源调用时长后，在延迟释放时长内未接收到目标算法对资源的新的调用请求，释放目标资源。

[0134] 本申请实施例中提供了一种可能的实现方式，资源的调度装置60还包括存储模块，存储模块用于：

[0135] 若未查询到目标算法可用的目标资源，将调用请求存储于请求队列中；

[0136] 针对于请求队列中的每一个调用请求，执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源。

[0137] 在第二方面的可选实施例中，存储模块在针对于请求队列中的每一个调用请求，执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源时，具体用于：

[0138] 根据每一个调用请求的接收时间，对请求队列中的至少一个调用请求进行排序；

[0139] 基于对调用请求的排序，依次执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源。

[0140] 在第二方面的可选实施例中，存储模块在针对于请求队列中的每一个调用请求，执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源时，具体用于：

[0141] 查询每一个调用请求对应的目标算法的紧急调用级别，依据紧急调用级别对请求队列中的至少一个调用请求排序；

[0142] 基于对调用请求的排序，依次执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源。

[0143] 在第二方面的可选实施例中，资源的调度装置60还包括确定模块，确定模块用于：

[0144] 确定目标算法的调用类别；调用类别包括占用不释放类别和占用后释放类别；

[0145] 释放模块在查询与目标算法对应的延迟释放时长时，具体用于：

[0146] 若目标算法的调用类别为占用后释放类别，查询与目标算法对应的延迟释放时长。

[0147] 在第二方面的可选实施例中，资源的调度装置60还包括第二调用模块，第二调用模块用于：

[0148] 若目标算法的调用类别为占用不释放类别，调用目标资源以运行目标算法，并持续调用目标资源以等待接收目标算法的新的调用请求。

[0149] 在第二方面的可选实施例中，资源的调度装置60还包括更新模块，更新模块用于：

[0150] 根据目标资源的资源类别、资源占用量和调用时长，对当前资源占用信息进行更新。

[0151] 上述的资源的调度装置，当接收到目标算法对资源的调用请求时，调用目标资源一定的调用时长以运行目标算法，并查询目标算法的延迟释放时长，若调用目标资源调用时长后，在延迟释放时长内未接收到目标算法对资源的新的调用请求，释放目标资源，可以在满足目标算法对目标资源的使用，并且在调用目标资源后，及时释放目标资源，以提供给其他算法调用，有效减少资源的浪费。

[0152] 进一步的，可以先确定目标算法的调用类别，对于占用不释放类别的目标算法，即便根据调用时长运行完，也不用释放目标算法，可以满足这类别的目标算法的长期调用目标资源的要求。

[0153] 本公开实施例的图片的资源的调度装置可执行本公开的实施例所提供的一种图片的资源的调度方法，其实现原理相类似，本公开各实施例中的图片的资源的调度装置中的各模块所执行的动作是与本公开各实施例中的图片的资源的调度方法中的步骤相对应的，对于图片的资源的调度装置的各模块的详细功能描述具体可以参见前文中所示的对应的图片的资源的调度方法中的描述，此处不再赘述。

[0154] 基于与本公开的实施例中所示的方法相同的原理，本公开的实施例中还提供了一种电子设备，该电子设备可以包括但不限于：处理器和存储器；存储器，用于存储计算机操作指令；处理器，用于通过调用计算机操作指令执行实施例所示的资源的调度方法。与现有技术相比，本申请中的资源的调度方法可以在满足目标算法对目标资源的使用，并且在调用目标资源后，及时释放目标资源，以提供给其他算法调用，有效减少资源的浪费。

[0155] 在一个可选实施例中提供了一种电子设备，如图7所示，图7所示的电子设备4000包括：处理器4001和存储器4003。其中，处理器4001和存储器4003相连，如通过总线4002相连。可选地，电子设备4000还可以包括收发器4004。需要说明的是，实际应用中收发器4004不限于一个，该电子设备4000的结构并不构成对本申请实施例的限定。

[0156] 处理器4001可以是CPU(Central Processing Unit，中央处理器)，通用处理器，DSP(Digital Signal Processor，数据信号处理器)，ASIC(Application Specific Integrated Circuit，专用集成电路)，FPGA(Field Programmable Gate Array，现场可编程门阵列)或者其他可编程逻辑器件、晶体管逻辑器件、硬件部件或者其任意组合。其可以实现或执行结合本申请公开内容所描述的各种示例性的逻辑方框，模块和电路。处理器

4001也可以是实现计算功能的组合,例如包含一个或多个微处理器组合,DSP和微处理器的组合等。

[0157] 总线4002可包括一通路,在上述组件之间传送信息。总线4002可以是PCI(Peripheral Component Interconnect,外设部件互连标准)总线或EISA(Extended Industry Standard Architecture,扩展工业标准结构)总线等。总线4002可以分为地址总线、数据总线、控制总线等。为便于表示,图7中仅用一条粗线表示,但并不表示仅有一根总线或一种类型的总线。

[0158] 存储器4003可以是ROM(Read Only Memory,只读存储器)或可存储静态信息和指令的其他类型的静态存储设备,RAM(Random Access Memory,随机存取存储器)或者可存储信息和指令的其他类型的动态存储设备,也可以是EEPROM(Electrically Erasable Programmable Read Only Memory,电可擦可编程只读存储器)、CD-ROM(Compact Disc Read Only Memory,只读光盘)或其他光盘存储、光碟存储(包括压缩光碟、激光碟、光碟、数字通用光碟、蓝光光碟等)、磁盘存储介质或者其他磁存储设备、或者能够用于携带或存储具有指令或数据结构形式的期望的程序代码并能够由计算机存取的任何其他介质,但不限于此。

[0159] 存储器4003用于存储执行本申请方案的应用程序代码,并由处理器4001来控制执行。处理器4001用于执行存储器4003中存储的应用程序代码,以实现前述方法实施例所示的内容。

[0160] 其中,电子设备包括但不限于:移动电话、笔记本电脑、数字广播接收器、PDA(个人数字助理)、PAD(平板电脑)、PMP(便携式多媒体播放器)、车载终端(例如车载导航终端)等等的移动终端以及诸如数字TV、台式计算机等等的固定终端。图7示出的电子设备仅仅是一个示例,不应对本公开实施例的功能和使用范围带来任何限制。

[0161] 本申请实施例提供了一种计算机可读存储介质,该计算机可读存储介质上存储有计算机程序,当其在计算机上运行时,使得计算机可以执行前述方法实施例中相应内容。与现有技术相比,本申请中的资源的调度方法可以在满足目标算法对目标资源的使用,并且在调用目标资源后,及时释放目标资源,以提供给其他算法调用,有效减少资源的浪费。

[0162] 应该理解的是,虽然附图的流程图中的各个步骤按照箭头的指示依次显示,但是这些步骤并不是必然按照箭头指示的顺序依次执行。除非本文中有明确的说明,这些步骤的执行并没有严格的顺序限制,其可以以其他的顺序执行。而且,附图的流程图中的至少一部分步骤可以包括多个子步骤或者多个阶段,这些子步骤或者阶段并不必然是在同一时刻执行完成,而是可以在不同的时刻执行,其执行顺序也不必然是依次进行,而是可以与其他步骤或者其他步骤的子步骤或者阶段的至少一部分轮流或者交替地执行。

[0163] 需要说明的是,本公开上述的计算机可读介质可以是计算机可读信号介质或者计算机可读存储介质或者是上述两者的任意组合。计算机可读存储介质例如可以是——但不限于——电、磁、光、电磁、红外线、或半导体的系统、装置或器件,或者任意以上的组合。计算机可读存储介质的更具体的例子可以包括但不限于:具有一个或多个导线的电连接、便携式计算机磁盘、硬盘、随机访问存储器(RAM)、只读存储器(ROM)、可擦式可编程只读存储器(EPROM或闪存)、光纤、便携式紧凑磁盘只读存储器(CD-ROM)、光存储器件、磁存储器件、或者上述的任意合适的组合。在本公开中,计算机可读存储介质可以是任何包含或存储程

序的有形介质，该程序可以被指令执行系统、装置或者器件使用或者与其结合使用。而在本公开中，计算机可读信号介质可以包括在基带中或者作为载波一部分传播的数据信号，其中承载了计算机可读的程序代码。这种传播的数据信号可以采用多种形式，包括但不限于电磁信号、光信号或上述的任意合适的组合。计算机可读信号介质还可以是计算机可读存储介质以外的任何计算机可读介质，该计算机可读信号介质可以发送、传播或者传输用于由指令执行系统、装置或者器件使用或者与其结合使用的程序。计算机可读介质上包含的程序代码可以用任何适当的介质传输，包括但不限于：电线、光缆、RF(射频)等等，或者上述的任意合适的组合。

[0164] 上述计算机可读介质可以是上述电子设备中所包含的；也可以是单独存在，而未装配入该电子设备中。

[0165] 上述计算机可读介质承载有一个或者多个程序，当上述一个或者多个程序被该电子设备执行时，使得该电子设备执行上述实施例所示的方法。

[0166] 可以以一种或多种程序设计语言或其组合来编写用于执行本公开的操作的计算机程序代码，上述程序设计语言包括面向对象的程序设计语言—诸如Java、Smalltalk、C++，还包括常规的过程式程序设计语言—诸如“C”语言或类似的程序设计语言。程序代码可以完全地在用户计算机上执行、部分地在用户计算机上执行、作为一个独立的软件包执行、部分在用户计算机上部分在远程计算机上执行、或者完全在远程计算机或服务器上执行。在涉及远程计算机的情形中，远程计算机可以通过任意种类的网络——包括局域网(LAN)或广域网(WAN)—连接到用户计算机，或者，可以连接到外部计算机(例如利用因特网服务提供商来通过因特网连接)。

[0167] 附图中的流程图和框图，图示了按照本公开各种实施例的系统、方法和计算机程序产品的可能实现的体系架构、功能和操作。在这点上，流程图或框图中的每个方框可以代表一个模块、程序段、或代码的一部分，该模块、程序段、或代码的一部分包含一个或多个用于实现规定的逻辑功能的可执行指令。也应当注意，在有些作为替换的实现中，方框中所标注的功能也可以以不同于附图中所标注的顺序发生。例如，两个接连地表示的方框实际上可以基本并行地执行，它们有时也可以按相反的顺序执行，这依所涉及的功能而定。也要注意的是，框图和/或流程图中的每个方框、以及框图和/或流程图中的方框的组合，可以用执行规定的功能或操作的专用的基于硬件的系统来实现，或者可以用专用硬件与计算机指令的组合来实现。

[0168] 描述于本公开实施例中所涉及到的模块可以通过软件的方式实现，也可以通过硬件的方式来实现。其中，模块的名称在某种情况下并不构成对该模块本身的限定，例如，接收模块还可以被描述为“用于接收调用请求的模块”。

[0169] 以上描述仅为本公开的较佳实施例以及对所运用技术原理的说明。本领域技术人员应当理解，本公开中所涉及的公开范围，并不限于上述技术特征的特定组合而成的技术方案，同时也应涵盖在不脱离上述公开构思的情况下，由上述技术特征或其等同特征进行任意组合而形成的其它技术方案。例如上述特征与本公开中公开的(但不限于)具有类似功能的技术特征进行互相替换而形成的技术方案。

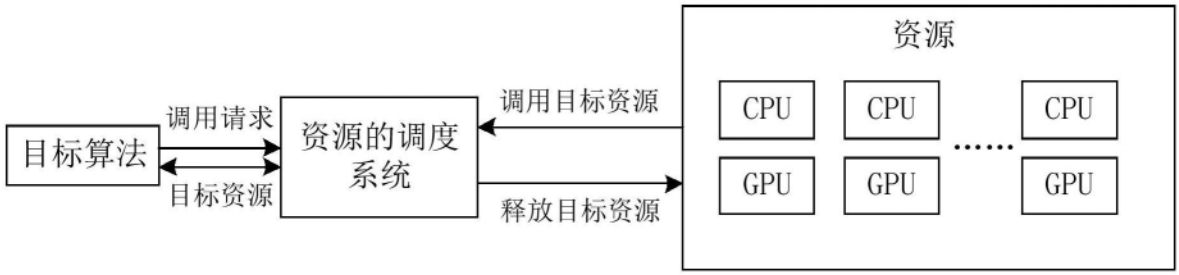

资源
目标算法
调用请求
目标资源
资源的调度系统
调用目标资源
释放目标资源
CPU
CPU
CPU
GPU
GPU
GPU

图1

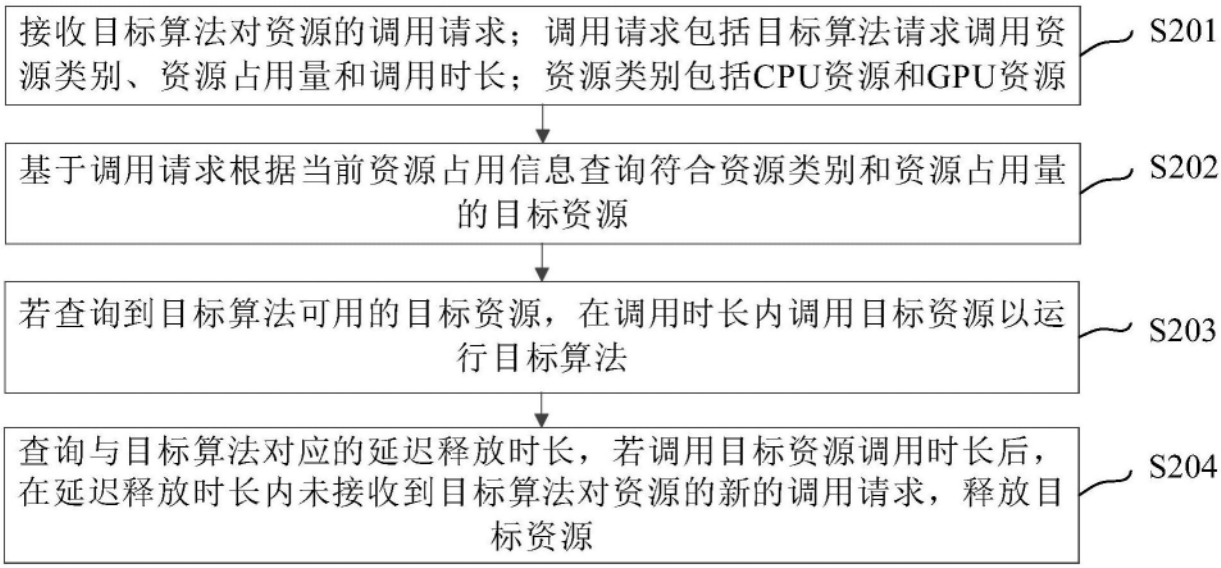

接收目标算法对资源的调用请求；调用请求包括目标算法请求调用资源类别、资源占用量和调用时长；资源类别包括CPU资源和GPU资源
S201
基于调用请求根据当前资源占用信息查询符合资源类别和资源占用量的目标资源
S202
若查询到目标算法可用的目标资源，在调用时长内调用目标资源以运行目标算法
S203
查询与目标算法对应的延迟释放时长，若调用目标资源调用时长后，在延迟释放时长内未接收到目标算法对资源的新的调用请求，释放目标资源
S204

图2

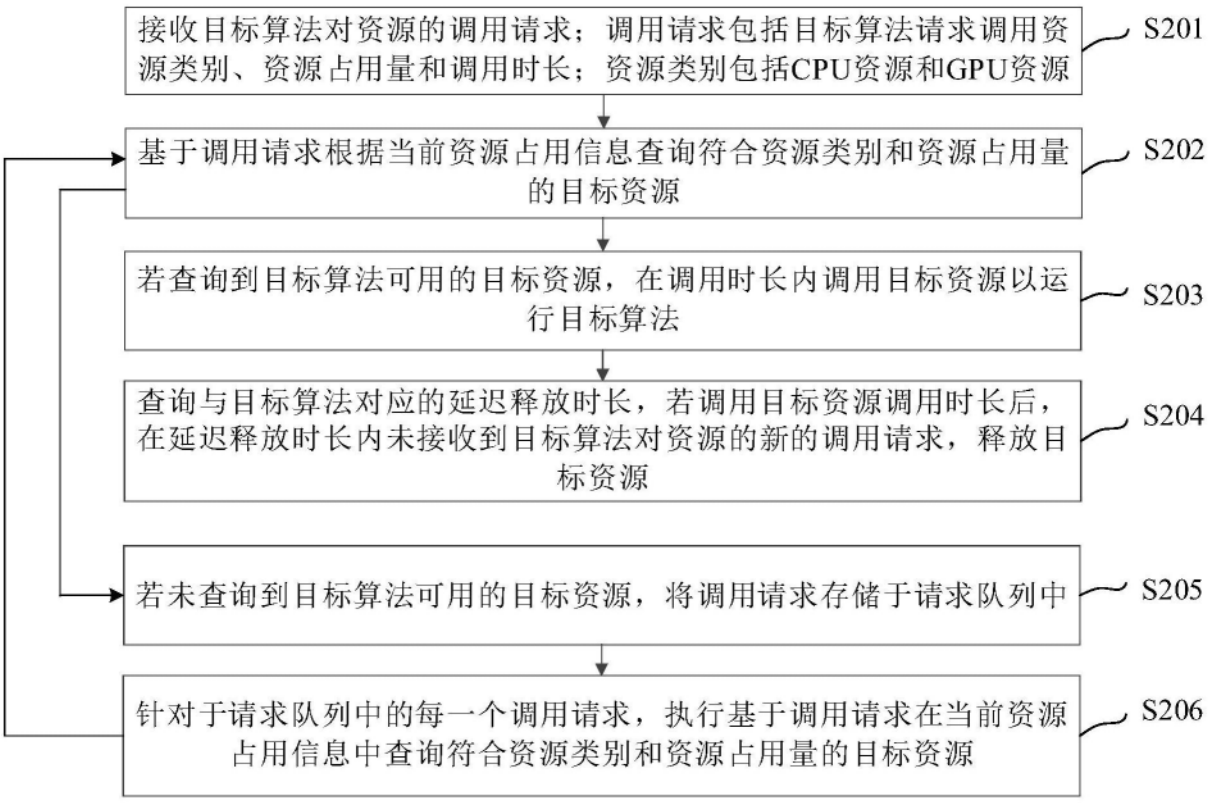

接收目标算法对资源的调用请求；调用请求包括目标算法请求调用资源类别、资源占用量和调用时长；资源类别包括CPU资源和GPU资源
S201
基于调用请求根据当前资源占用信息查询符合资源类别和资源占用量的目标资源
S202
若查询到目标算法可用的目标资源，在调用时长内调用目标资源以运行目标算法
S203
查询与目标算法对应的延迟释放时长，若调用目标资源调用时长后，在延迟释放时长内未接收到目标算法对资源的新的调用请求，释放目标资源
S204
若未查询到目标算法可用的目标资源，将调用请求存储于请求队列中
S205
针对于请求队列中的每一个调用请求，执行基于调用请求在当前资源占用信息中查询符合资源类别和资源占用量的目标资源
S206

图3

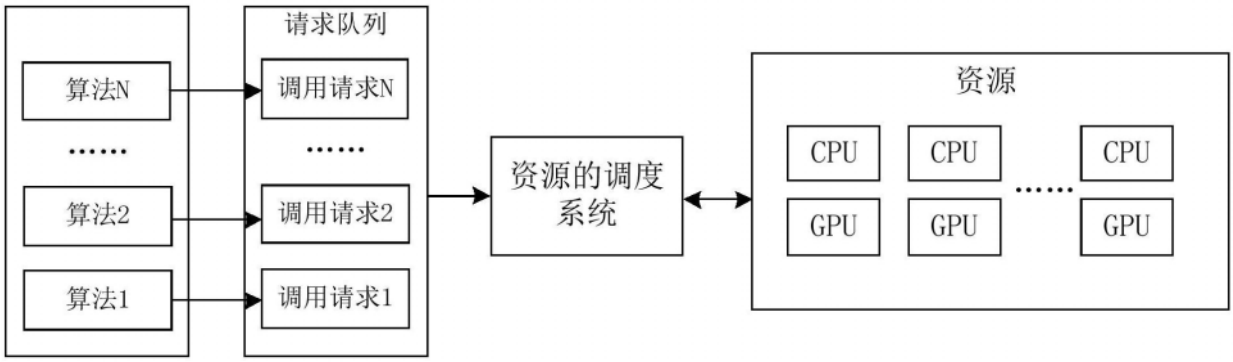

请求队列
算法N
算法2
算法1
调用请求N
调用请求2
调用请求1
资源的调度系统
资源
CPU
CPU
CPU
GPU
GPU
GPU

图4

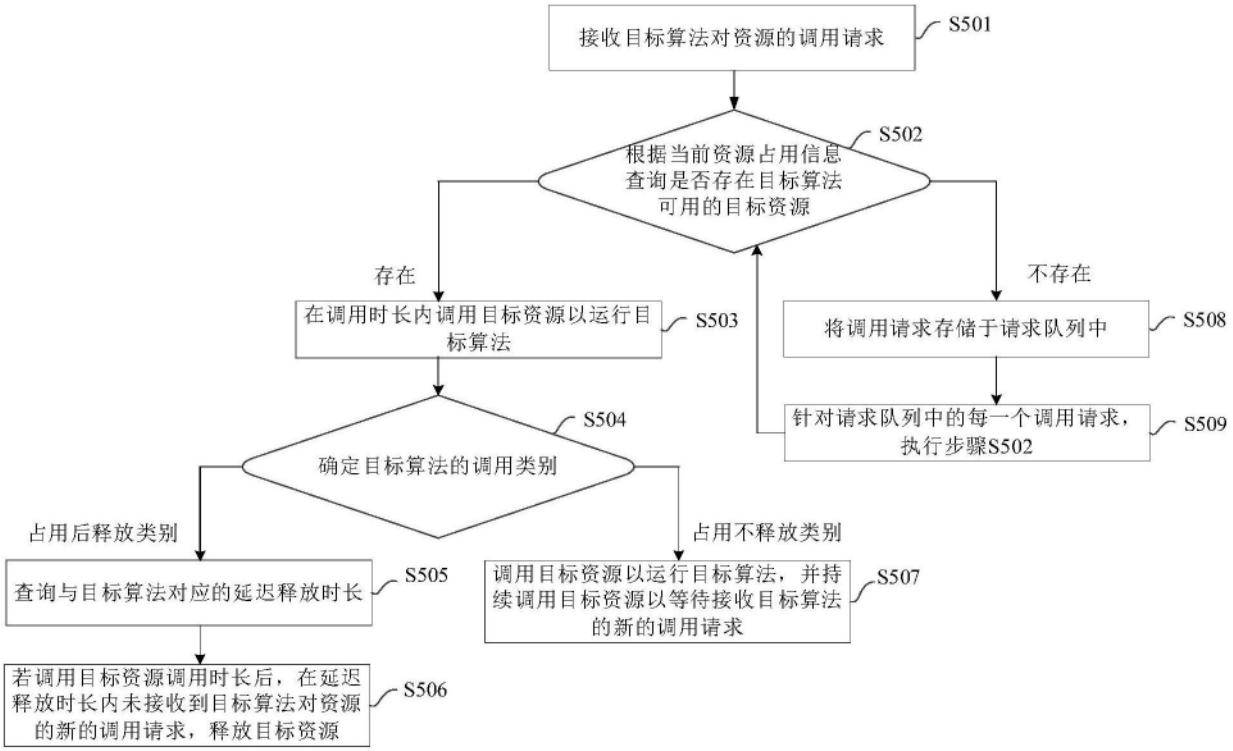

接收目标算法对资源的调用请求
S501
根据当前资源占用信息查询是否存在目标算法可用的目标资源
S502
存在
不存在
在调用时长内调用目标资源以运行目标算法
S503
将调用请求存储于请求队列中
S508
针对请求队列中的每一个调用请求，执行步骤S502
S509
确定目标算法的调用类别
S504
占用后释放类别
占用不释放类别
查询与目标算法对应的延迟释放时长
S505
调用目标资源以运行目标算法，并持续调用目标资源以等待接收目标算法的新的调用请求
S507
若调用目标资源调用时长后，在延迟释放时长内未接收到目标算法对资源的新的调用请求，释放目标资源
S506

图5

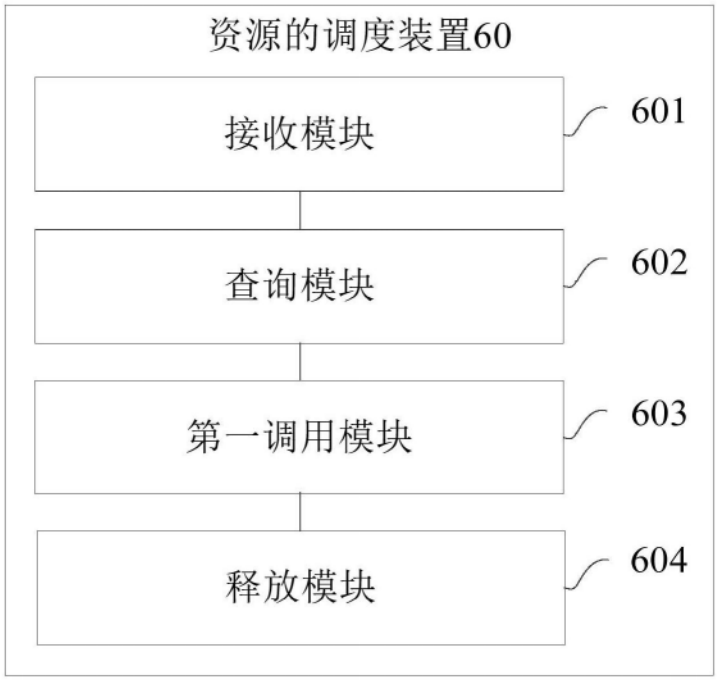

资源的调度装置60
接收模块
601
查询模块
602
第一调用模块
603
释放模块
604

图6

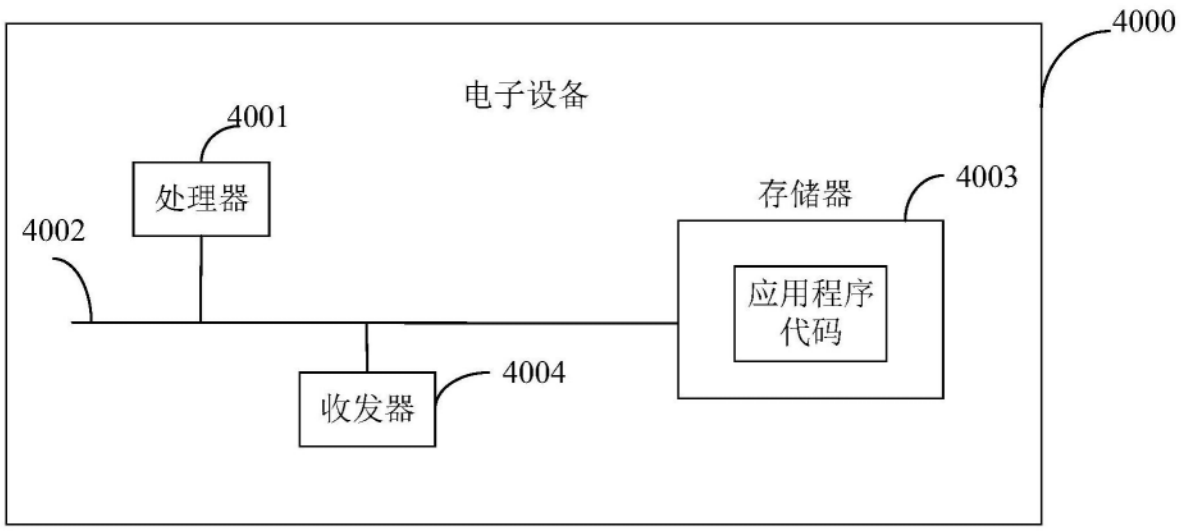


图7