



(12) 发明专利申请

(10) 申请公布号 CN 101743534 A

(43) 申请公布日 2010.06.16

(21) 申请号 200880024896.X

(51) Int. Cl.

(22) 申请日 2008.07.16

G06F 9/46 (2006.01)

(30) 优先权数据

11/778,487 2007.07.16 US

(85) PCT申请进入国家阶段日

2010.01.15

(86) PCT申请的申请数据

PCT/US2008/070147 2008.07.16

(87) PCT申请的公布数据

W02009/012296 EN 2009.01.22

(71) 申请人 微软公司

地址 美国华盛顿州

(72) 发明人 J·B·巴纳德 Y·金

(74) 专利代理机构 上海专利商标事务所有限公

司 31100

代理人 顾嘉运 钱静芳

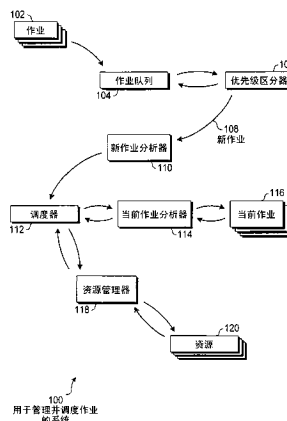
权利要求书 3 页 说明书 7 页 附图 3 页

(54) 发明名称

通过增长和收缩资源分配来进行调度

(57) 摘要

用于计算资源的调度器可周期性地分析正在运行的作业来确定是否可向该作业分配附加资源以帮助该作业更快地完成,并且也可检查是否有开始正在等待的作业的最小量的资源可用。作业可包括可用任务之间的并行或串行关系来定义的多个任务。在执行期间的各个时间点处,可以响应于优先级系统来调整活动作业的资源分配以添加或移除资源。作业能以最小量的资源开始并且这些资源可以在作业的生存周期中增加或减少。



1. 一种方法,包括:

分析操作多个资源的第一作业以确定可被分配给所述第一作业的最大量的资源(306),所述第一作业包括多个任务;

分析多个资源来确定第一资源集合是未在使用的资源并且第二资源集合是分配给所述第一作业的资源(312);以及

在所述第二资源集合小于所述最大资源集合并且所述第一资源集合非空的情况下,将所述第一资源集合的至少一部分分配给所述第一作业(322)。

2. 如权利要求1所述的方法,其特征在于,所述多个任务包括能够并行执行的至少两个任务。

3. 如权利要求1所述的方法,其特征在于,所述多个任务中的一个任务在顺序上依赖于所述多个任务中的另一个任务。

4. 如权利要求1所述的方法,其特征在于,所述资源包括含有下列各项的组中的至少一个:

处理器;

计算机;

存储器;

网络带宽;

网络连接;

输入设备;

输出设备;以及

许可证。

5. 如权利要求1所述的方法,其特征在于,还包括:

分析操作第三资源集合的第二作业来确定可被分配给所述第二作业的最小量的资源;

确定所述第一作业具有比所述第二作业高的优先级;

确定所述最小量的资源小于所述第三资源集合;以及

将所述第三资源集合中的至少一部分分配给所述第一作业。

6. 如权利要求1所述的方法,其特征在于,所述方法是由所述任务中的一个的完成来启动的。

7. 一种包括适用于执行如权利要求1所述的方法的计算机可执行指令的计算机可读介质。

8. 一种系统,包括:

适用于确定第一作业的最大资源需求的当前作业分析器(114),所述第一作业正在部分地执行并且包括多个任务;

适用于确定多个资源的当前状态的资源管理器(118),所述当前状态包括已分配的资源 and 正在使用的资源;以及

适用于基于所述当前状态和所述最大量资源需求来改变分配给所述第一作业的资源调度器(112)。

9. 如权利要求8所述的系统,其特征在于,所述改变资源包括向所述第一作业分配附

加资源。

10. 如权利要求 8 所述的系统,其特征在于,所述改变资源包括向所述第一作业分配更少资源。

11. 如权利要求 8 所述的系统,其特征在于,还包括:

适用于接收多个作业的作业队列,所述多个作业中的每一个包括多个任务;

适用于向所述作业队列中的所述多个作业的每一个分配优先级的优先级系统;以及

适用于确定第二作业的最小资源需求的传入作业分析器;

所述调度器还适用于在可向所述第二作业分配所述最小资源需求时开始所述第二作业。

12. 如权利要求 11 所述的系统,其特征在于,按包括自从作业被提交以来的时间长度的公式确定所述优先级。

13. 如权利要求 11 所述的系统,其特征在于,按包括针对所述作业队列中的所述多个作业的每一个的优先级设置的公式确定所述优先级。

14. 如权利要求 8 所述的系统,其特征在于,所述多个任务包括能够并行执行的至少两个任务。

15. 如权利要求 8 所述的系统,其特征在于,所述多个任务中的一个任务在顺序上依赖于所述多个任务中的另一个任务。

16. 如权利要求 8 所述的系统,其特征在于,所述资源包括含有下列各项的组中的至少一个:

处理器;

计算机;

存储器;

网络带宽;

网络连接;

输入设备;

输出设备;以及

许可证。

17. 一种方法,包括:

分析作业队列来确定所述作业队列中的多个作业的优先级 (328),所述作业中的每一个包括多个任务;

从所述优先级确定第一作业 (330),所述第一作业是要被执行的下一作业;

确定开始执行所述第一作业的最小资源集合 (332);

分析操作多个资源的第二作业来确定可被分配给所述第二作业的最大量的资源 (306);

分析多个资源来确定第一资源集合是未在使用的资源并且第二资源集合是分配给所述第二作业的资源 (312);以及

在所述第二资源集合小于所述最大资源集合并且所述第一资源集合非空的情况下,将所述第一资源集合的至少一部分分配给所述第二作业 (322);以及

在所述最小资源集合可用时开始所述第一作业 (338)。

18. 如权利要求 17 所述的方法,其特征在于,还包括:

分析所述第二作业来确定可被分配给所述第二作业的最小量的资源。

19. 如权利要求 17 所述的方法,其特征在于,所述方法是响应于来所述任务中的一个的完成来开始的。

20. 一种包括适用于执行如权利要求 17 所述的方法的计算机可执行指令的计算机可读介质。

通过增长和收缩资源分配来进行调度

[0001] 背景

[0002] 大型计算环境中的资源通常由调度系统来管理。这些资源可以是计算机或处理器的集群,或可包括其它资源。调度机制可以跨各个资源块来调度大型计算任务。在许多情况下,大型计算资源是极其需要的,如此高效的调度可以更好地利用这些资源。

[0003] 概述

[0004] 用于计算资源的调度器可周期性地分析正在运行的作业来确定是否可向该作业分配附加资源以使该作业更快地完成,并且可检查是否有用于开始正在等待的作业的最小量的资源可用。作业可包括可用任务之间的并行或串行关系来定义的多个任务。在执行期间的各个时间点处,可以响应于优先级系统来调整活动作业的资源分配以添加或移除资源。作业能以最小量的资源开始并且这些资源可以在作业的生存周期中增加或减少。

[0005] 提供本概述是为了以简化的形式介绍将在以下详细描述中进一步描述的一些概念。本概述并不旨在标识出所要求保护的的主题的关键特征或必要特征,也不旨在用于限定所要求保护的的主题的范围。

[0006] 附图简述

[0007] 在附图中:

[0008] 图 1 是示出用于管理和调度作业的系统实施例的图示。

[0009] 图 2 是示出作业和针对该作业的资源加载的实施例的图示。

[0010] 图 3 是示出用于向正在运行和正在等待的作业分配资源的方法的实施例的流程图。

[0011] 详细描述

[0012] 可以使用调度系统来将共享计算资源分配给各个作业。该调度系统可包括新作业的队列,其中优先级系统可以确定接着将开始哪一作业。分析器可以周期性地评估正在执行的作业以标识未被充分使用的资源,并可以确定开始新作业或向现有作业分配附加资源。

[0013] 每一作业可被定义成一系列任务。某些任务能以串行或并行的方式链接,并且每一任务可以使用一定范围的资源。在执行期间的某些时间点处,多个任务可以并行执行,而在其它时间点处,一任务在执行之前等待其它任务完成。在可以执行并行任务的时间段期间,通过应用附加资源可以更快地完成作业,而在其它时间段期间,可以将同样的这些资源分配给其它作业。

[0014] 在某些情况下,新作业可以通过确定可用于开始作业的最小量的资源来开始。在这些资源变得可用时,可以开始新作业。在其它资源变得空闲时,可以将它们分配给该新作业以使该新作业可被快速完成。

[0015] 在对资源分配的周期性分析期间,可以使用一种算法来在正在执行的作业之间分配资源并决定是否要开始新作业。不同的实施例可以使用不同的算法。例如,一个实施例可以偏爱尽可能快地完成现有作业,而另一实施例可以偏爱尽可能快地开始新作业。在某些情况下,在选择动作过程时可以考虑作业和资源之间的各个优先级。

[0016] 使用本发明的具体实施例来示出本发明的具体方面。这些实施例仅作为示例，且易于作出各种修改和替换形式。所附权利要求书旨在覆盖落入由权利要求书所定义的本发明的精神和范围内的所有修改、等效方案、以及替换。

[0017] 本说明书通篇中，在所有附图的描述中，相同的附图标记表示相同的元素。

[0018] 在将元素称为被“连接”或“耦合”时，这些元素可以直接连接或耦合在一起，或者也可以存在一个或多个中间元素。相反，在将元素称为被“直接连接”或“直接耦合”时，不存在中间元素。

[0019] 本发明可被具体化为设备、系统、方法、和 / 或计算机程序产品。因此，本发明的部分或全部能以硬件和 / 或软件（包括固件、常驻软件、微码、状态机、门阵列等）来具体化。此外，本发明可以采用其上包含有供指令执行系统使用或结合其使用的计算机可使用或计算机可读程序代码的计算机可使用或计算机可读存储介质上的计算机程序产品的形式。在本文的上下文中，计算机可使用或计算机可读介质可以是可包含、存储、通信、传播、或传输程序以供指令执行系统、装置或设备使用或结合其使用的任何介质。

[0020] 计算机可使用或计算机可读介质可以是，例如，但不限于，电、磁、光、电磁、红外、或半导体系统、装置、设备或传播介质。作为示例而非限制，计算机可读介质可包括计算机存储介质和通信介质。

[0021] 计算机存储介质包括以用于存储诸如计算机可读指令、数据结构、程序模块或其它数据等信息的任何方法或技术来实现的易失性和非易失性、可移动和不可移动介质。计算机存储介质包括，但不限于，RAM、ROM、EEPROM、闪存或其它存储器技术、CD-ROM、数字多功能盘 (DVD) 或其它光盘存储、磁带盒、磁带、磁盘存储或其它磁性存储设备、或能用于存储所需信息且可以由指令执行系统访问的任何其它介质。注意，计算机可使用或计算机可读介质可以是其上打印有程序的纸张或另一合适的介质，因为程序可以经由例如对纸张或其它介质的光学扫描而电子地捕获，随后如有必要被编译、解释，或以其它合适的方式处理，并随后存储在计算机存储器中。

[0022] 通信介质通常以诸如载波或其它传输机制等已调制数据信号来体现计算机可读指令、数据结构、程序模块或其它数据，并包括任意信息传送介质。术语“已调制数据信号”指的是其一个或多个特征以在信号中编码信息的方式被设定或更改的信号。作为示例而非限制，通信介质包括有线介质，诸如有线网络或直接线连接，以及无线介质，诸如声学、RF、红外线和其它无线介质。上述中任一组合也应包括在计算机可读介质的范围之内。

[0023] 当本发明在计算机可执行指令的一般上下文中具体化时，该实施例可以包括由一个或多个系统、计算机、或其它设备执行的程序模块。一般而言，程序模块包括执行特定任务或实现特定抽象数据类型的例程、程序、对象、组件、数据结构等。通常，程序模块的功能可以在各个实施例中按需进行组合或分布。

[0024] 图 1 是示出用于管理和调度作业的系统实施例 100 的图示。实施例 100 可用于控制可使用多个计算资源的作业。例如，计算机集群可被组织成执行可跨组成该集群的各个计算机来分布的大型计算作业。

[0025] 可以管理许多不同类型的计算机资源。例如，计算机集群中的或多处理器计算机上的各个处理器可被分配组成计算作业的各个任务。在其它情况下，可以分配存储器资源，包括随机存取存储器和数据存储存储器两者。可以分配网络资源，以及软件许可证、计算资

源、以及可能存在争用的其它资源。实施例 100 可用来管理并分配跨多个作业共享的任何计算资源。

[0026] 每一作业 102 可包括多个任务。这些任务可以是可被执行的离散的可执行代码块或功能。每一任务可以具有可由该任务使用的分开定义的资源。例如，在处理器集群上执行作业的典型情况下，单独的任务可以在单个处理器上执行或可以使用单个软件许可证。在某些实施例中，每一任务可以使用多个资源。例如，任务可被定义成使用四个处理器或被定义成最少使用两个处理器以及最多使用八个处理器来运行。

[0027] 因为作业 102 可以用多个任务来定义，所以作业在执行过程期间可以使用不同级别的资源。例如，作业可以具有可并行执行的若干任务。可以通过向单独的处理器分配每一任务来执行作业并使得该作业快速完成。还可以通过在单个处理器上顺序地执行每一任务来执行作业。

[0028] 作业 102 可在被执行之前被置于作业队列 104 中。优先级区分器 106 可以评估作业队列 104 中的作业以确定接着可以执行哪一作业。

[0029] 优先级区分器 106 可以使用不同的机制和算法来确定接着执行哪一作业。在某些实施例中，每一作业可具有笼统的优先级，如低、中、高。某些实施例可以使用自从该作业被提交以来的时间长度作为另一因素。可以使用一种算法来计算每一待决作业的优先级，对队列进行排序，并标识要执行的下一作业。

[0030] 优先级区分器 106 所选择的新作业 108 可由新作业分析器 110 来分析。新作业分析器 110 可以确定可用来开始新作业 108 的最小资源。在该最小量的资源变得可用时，可以开始新作业 108。

[0031] 调度器 112 可以确定资源跨各个作业的分配。在某些情况下，调度器 112 可以向正在执行的作业添加资源，而在其它情况下，调度器 112 可以从作业移除资源。在执行期间，当前作业分析器 114 可以分析当前（即正在运行的）作业 116 以确定是否有作业正在使用其所有分配的资源或是否可向该作业分配附加资源。

[0032] 在许多情况下，作业可具有可并行操作的多个任务，并且在并行执行多个任务期间可以使用附加资源。在其它情况下，作业可具有串行或顺序相关的任务，使得在一个任务完成之后才执行另一个任务。在具有带并行和顺序依赖性的多个任务的作业中，作业在执行期间可以使用不同的资源量。在任务的大量并行执行的时间段期间，可以向该作业分配最大量的资源。一旦该时间段过去并且作业进入多个任务顺序相关的时间段，则该作业可能被分配了超过该作业能使用的资源。

[0033] 在作业的执行期间，当前作业分析器 114 可以确定可以分配给作业的最大量的资源。例如，该最大量的资源可用来确定可以向作业应用多少资源来尽可能快地完成该作业。当前作业分析器 114 可以确定可以分配给同一作业的最小量的资源。例如，该最小量的资源可由调度器 112 用来移除资源，以使较高优先级的作业可以使用该资源。在这种或其它情况下，可以确定最小量的资源，以便执行作业而不会因资源不足引起死锁。

[0034] 资源管理器 118 可以监控并分析资源 120，以确定各资源 120 的当前状态。在许多情况下，所分配的资源可包括集群计算应用程序的处理器。在其它情况下，所分配的资源可包括各个计算机、各种类型的存储器设备、网络连接和带宽、各种输入或输出设备、以及软件许可证或其它计算资源。在本说明书中，对集群计算和分配处理器或计算机资源的任何

参考都是作为示例的并且不是限制。

[0035] 调度器 112 可具有多个不同类型的算法,并且在确定何时开始新作业以及如何跨各个作业来分配或分摊资源时可使用各种因素。每一实施例可以使用不同的逻辑和/或公式来分配资源。在某些实施例中,可以定义着重于使用尽可能多的资源来尽可能快地完成正在执行的作业的逻辑。其它逻辑可以考虑正在执行的作业的优先级,并以对较高优先级作业以优于较低优先级作业的方式来分配资源。又一些实施例可被设计成通过向新作业而非正在执行的作业分配资源来尽可能快地开始作业。

[0036] 在某些实施例中,调度器 112 可以评估正在运行的作业,在正在运行的作业之间分配资源,并随后向任何新作业分配未在使用的资源。其它实施例可以使用各种算法和加权方案来对正在运行的作业和待决作业一起确定优先级。

[0037] 图 2 是示出作业的示例和针对该作业的资源加载的实施例 200 的图示。作业可被定义成可作为并行的、或相关的、或顺序的任务而链接在一起的多个单独的任务。在集群计算应用程序的示例中,每一任务可以是可在单独的处理器的分开的可执行项目。其它实施例可以使用其它准则来将作业划分成各个任务。各任务可具有在某些任务之间共享的并可以是任务之间的关系的基础的特定输入或输出数据。

[0038] 作业的任务序列 202 的图形表示在左侧示出,而资源加载表 204 在右侧示出。

[0039] 任务序列 202 在框 206 处开始。第一任务 208 可用于启动作业。与第一任务 208 相对应,表 204 的第一行指示资源加载具有最大 1 和最小 1。最大和最小资源加载可以是在该时间段内可分配给作业的资源量。因为任务 208 是操作中的一个任务,所以可以分配一个资源。

[0040] 出于简化该示例的目的,每一任务可被定义成使用可以是任何类型的计算资源的单个资源。在其它示例中,每一任务可以使用跨不同的资源类别的多个资源。例如,单个任务可以使用 2 到 16 个处理器、网络访问、以及软件许可证,而另一任务可以使用单个处理器、特定输出设备、以及不使用软件许可证。在图 2 的示例中,为示出概念,每一任务可对应于单个资源。

[0041] 在任务 208 完成之后,任务 210、212、和 214 可并行执行。当任务 210、212、和 214 正在执行时,资源加载具有最大 3 个和最小 1 个资源。当资源加载是 1 时,任务 210、212、和 214 中的每一个可以使用该单个资源顺序地执行。如果向该作业分配了更多资源,则两个或更多个任务可以同时执行。例如,如果向作业 202 分配了 2 个资源,则任务 210 和 212 可以同时执行,并且如果分配了 3 个资源,则任务 210、210、和 214 可以并行执行。

[0042] 作业调度器可以基于优先级来向各个作业分配资源。例如,如果作业 202 具有比正在执行的其它作业高的优先级,则可以分配最高到最大资源加载的附加资源以使作业 202 更快地完成。相反,如果作业 202 具有比其它作业(无论是正在执行的作业还是待决作业)低的优先级,则可将资源分配给其它作业而可为作业 202 保留最小的资源加载,以使作业 202 可被继续执行。

[0043] 任务 216、218、和 220 依赖于任务 210 并且是并行任务。类似地,任务 222 和 224 是并行任务并依赖于任务 214。在该时间段期间,资源加载最大可以是 5,最小是 1。

[0044] 任务 226 依赖于任务 216、218、220、222、224、和 212。可以定义任务 226 对其它任务的依赖性,以便直到所有其它任务完成后才开始任务 226。在任务 226 的时间段期间,最

大资源加载和最小资源加载都可以是 1。尽管任务 226 所依赖的各个任务正在完成,但一个或多个任务可在其余任务之前完成。例如,如果向作业分配了 5 个资源并且 2 个任务在其余任务之前完成,则 3 个资源可由其余任务主动地使用,而 2 个资源可能过量并且没有被作业 202 所使用。一旦使用这些过量资源的任务完成,则这些资源可由作业调度器收回并分配给其它作业。

[0045] 任务 228 可以依赖于任务 226 并可以具有最大和最小资源加载 1。在任务 228 完成后,作业 202 可以在框 230 结束。

[0046] 调度器可以使用各种算法或逻辑来向作业分配资源。例如,在作业 202 的开始处,调度器可以向作业 202 分配最大 5 个资源,以使作业 202 可以快速完成,即使前几个任务可以在不使用最大资源的情况下完成。在其它实施例中,调度器最初可以分配一个资源,并在任务 216 到 224 的执行期间增加到 5 个资源。

[0047] 调度器可以移除分配给某一作业的资源并将这些资源分配给另一作业。例如,如果向作业 202 分配了 5 个资源并且任务 218 和 218 已经完成而任务 220、222 和 224 正在进行,则作业 202 可利用 3 个资源而非所有的 5 个资源。

[0048] 在许多情况下,调度器可以评估可分配给作业的将来资源加载,以确定作业之间的资源分配。例如,调度器可以评估作业 202 并且确定最大资源加载是 5,并向作业 202 分配 5 个资源,即使在任务 216 到 224 被执行之前 5 个资源未被充分利用。

[0049] 在某些实施例中,调度器可以在作业执行的各个阶段分配资源。例如,调度器可以向作业 202 分配单个资源,而作业 202 可以继续使用该资源直到任务 214 完成为止。在该点处,更多资源可变得可用,并且调度器可以分配多达最大 5 个资源的附加资源。

[0050] 在某些情况下,资源可变得可用并且调度器可以将该资源分配给正在在其最大值以下操作的作业。在进行这种分配时,可以评估各种因素,包括作业的优先级、作业的当前资源分配与最大分配之间的差、作业的最大和最小最大量资源加载之间的差、或任何其它因素。

[0051] 在资源可用时,调度器可以评估是否可以开始新作业。每一调度器可以具有不同的逻辑或算法来确定何时开始作业以及如何分配资源。在某些实施例中,调度器可以在尝试向新作业分配资源之前,尝试在正在执行的作业之间分配资源。在其它实施例中,调度器可以使用评估现有作业和待决作业的优先级的算法或逻辑,并可以向新作业分配资源,即使现有作业能够使用这些资源。

[0052] 图 2 的示例示出可以对序列 202 确定资源加载的一种方式。序列 202 可被划分成具有不同资源加载的不同步骤。例如,如果任务 216 和 218 被分组在一起并且任务 220、222 和 224 被分组在一起,则任务加载可具有最大资源加载 3。在某些实施例中,可以通过分析各个任务分组以确定各任务和作业之间的优化资源加载来确定优化的资源加载。

[0053] 图 3 是用于向作业分配资源的方法的示例的实施例 300 的流程图。实施例 300 只是可向正在执行的作业和处于等待执行的作业队列中的那些作业分配资源的算法或逻辑的一个示例。

[0054] 尽管其它实施例可以使用不同的逻辑或方法来分配资源,但实施例 300 被设计成基于正在执行的作业的优先级来向正在执行的作业分配可用资源。一旦向正在执行的作业分配了所有资源,则最高优先级的等待作业可以在该等待作业的最小量资源可用时开始。

[0055] 实施例 300 是一种算法的示例。本领域技术人员可以理解,可以基于具体实现来对实施例 300 进行改变以产生不同的结果。此外,实施例 300 可被用于可由各种任务使用的不同类型的资源的资源分配。在某些实施例中,不同类型的资源可由调度器来管理,并且每一作业可以使用不同量的每一类型的资源。这些实施例可以使用类似或不同的逻辑或算法来在作业之间分配各种类型的资源。

[0056] 在框 302,分析当前资源。在许多实施例中,资源分析器可以确定存在多少资源、分配了哪些资源、哪些资源正被利用、或关于当前资源的其它信息。在某些情况下,资源可偶尔上线和离线,并可能不是永久可用的。

[0057] 在框 304,对于每一正在执行的作业,在框 306 确定最大资源加载,在框 308 确定最小资源加载,并在框 310 确定该作业的优先级。可以针对作业的整体长度或针对作业的一小段(如接下来的若干任务)来确定最大和最小资源加载。

[0058] 在某些情况下,在作业的执行期间,可以多次执行实施例 300 或类似算法,以便在执行的过程期间可以多次分配和从作业移除资源。在这种情况下,可以对较短时间段而非对作业执行的长度评估作业的最大和最小资源加载。例如,在可在周期性地基础上(例如每 10 分钟)分配资源的情况下,可以对执行的下一 10 分钟或 20 分钟来评估作业的最大和最小资源加载。

[0059] 在某些情况下,在作业之间的资源加载显著改变时可以重新计算最大和最小资源加载。例如,如果一个或若干任务完成使得资源加载显著地增加或减少,则可以执行重新计算。这样的实施例可被用来最小化对不显著改变现有加载的资源加载的重新计算。

[0060] 如果在框 312,分配给作业的资源超过该作业的最大值,则在框 314,可将超出的资源标识为未在使用的。在该算法后面的步骤中,超出的资源可被分配给其它作业。

[0061] 在框 316,按优先级来排列正在执行的作业。每一实施例可具有用于确定作业的优先级的不同机制。在某些情况下,用户可以在提交之前确定作业的优先级。在其它情况下,正在执行的最老的作业可被给予比较新作业高的优先级。还有其它情况可以使用其它准则或公式来确定优先级。

[0062] 在框 318,对于按降序优先级顺序的每一作业,在框 320,如果所分配的资源低于最大值,则在框 322,可将未在使用的资源分配给该作业,直至达到该作业的最大值。框 318、320、和 322 的步骤基于优先级将各种未在使用的资源分配给正在执行的作业,以使正在执行的作业可以快速完成。

[0063] 在框 324,分析输入队列。输入队列可包含尚未被执行的作业。在框 326,对于输入队列中的每一作业,在框 328 确定优先级。传入作业的优先级可以使用任何准则或因素来确定,包括输入队列中的时间长度、总体优先级、或其它因素。

[0064] 在框 330,对输入队列进行排序以确定要开始的下一作业。在框 332,确定开始该下一作业的最小资源。

[0065] 在框 334,如果有任何未在使用的资源可用,并且在框 336,有足够的未在使用的资源可用于开始新作业,则在框 338,开始该新作业。在框 340,该过程在继续之前等待有任务完成。

[0066] 在框 334,如果没有资源剩余或在框 336,如果剩余资源不足以开始新作业,则在框 340,该过程等待直到有任务完成。

[0067] 实施例 300 是被设计成在开始新作业之前向现有作业分配可用资源的算法。可以设计其它实施例,以使作业队列中的高优先级作业甚至可以在可向正在运行或执行的作业分配可用资源时开始。

[0068] 实施例 300 被设计成在每次完成任务时执行。这样的实施例在其中任务花费相对长时间来完成的情况下是有用的。在其中任务很短的实施例中,对实施例 300 的分析可消耗大量开销并可变得不便使用。在这种情况下,一实施例可具有在作业完成执行时、在特定数量的任务完成时、或在周期性的时间基础上运行的算法。

[0069] 以上对本发明的描述是出于说明和描述的目的而提出的。它不旨在穷举本主题或将本主题限于所公开的精确形式,且鉴于以上教导其它修改和变型都是可能的。选择并描述实施例来最好地解释本发明的原理及其实际应用,从而使本领域的其他技术人员能够在各种实施例和各种适于所构想的特定用途的修改中最好地利用本技术。所附权利要求书旨在包括除受现有技术所限的范围之外的其它另选的实施例。

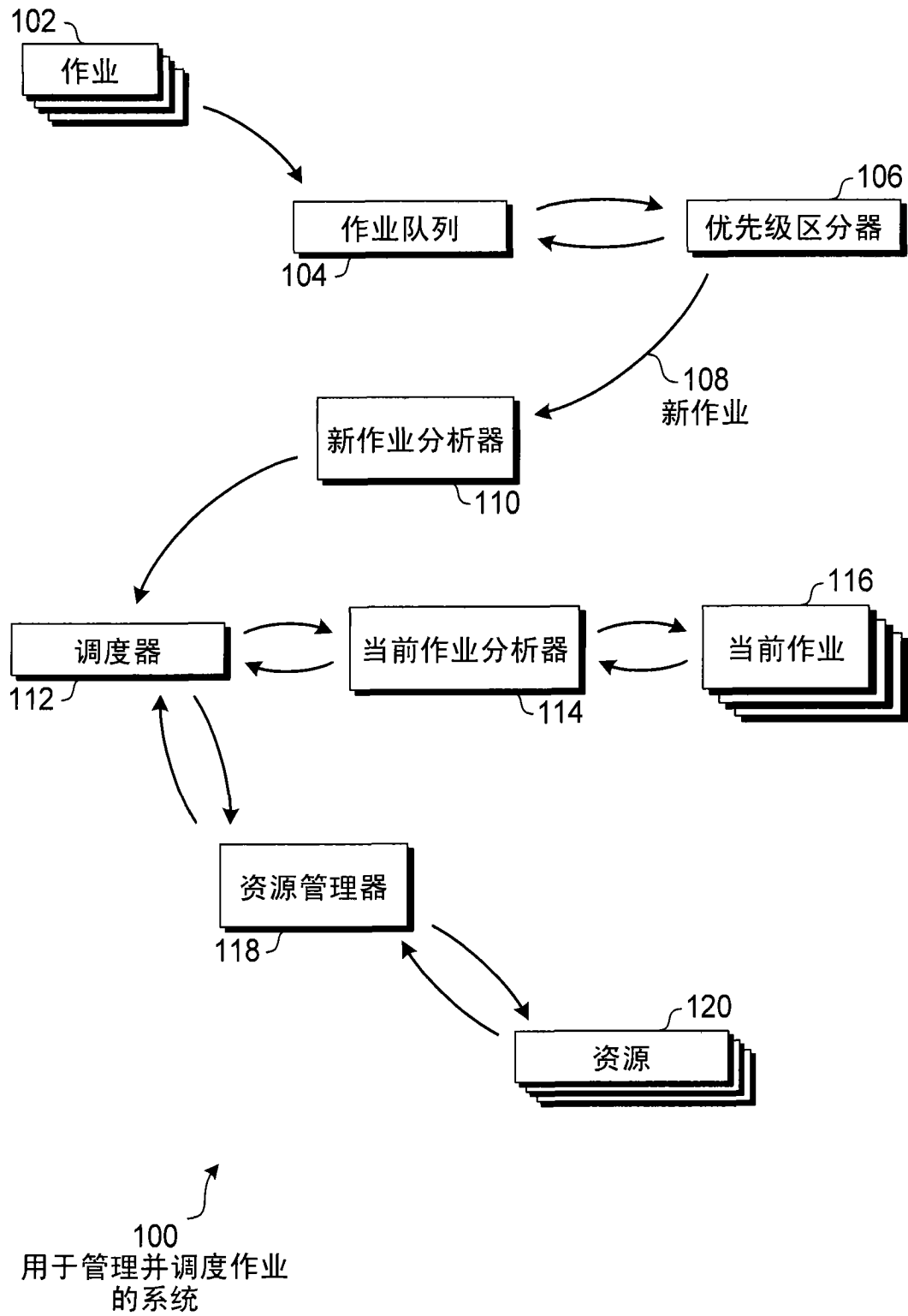


图 1

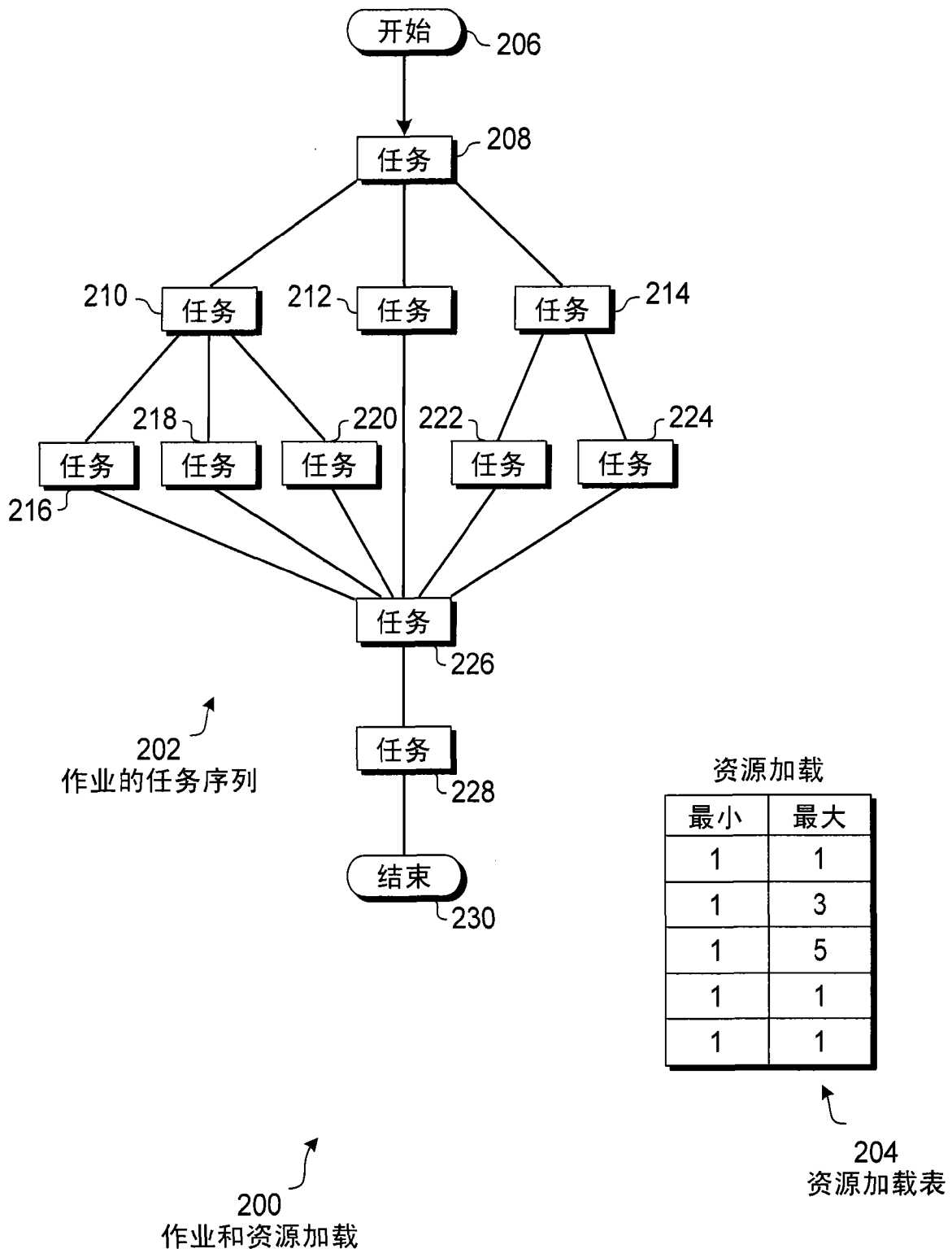


图 2

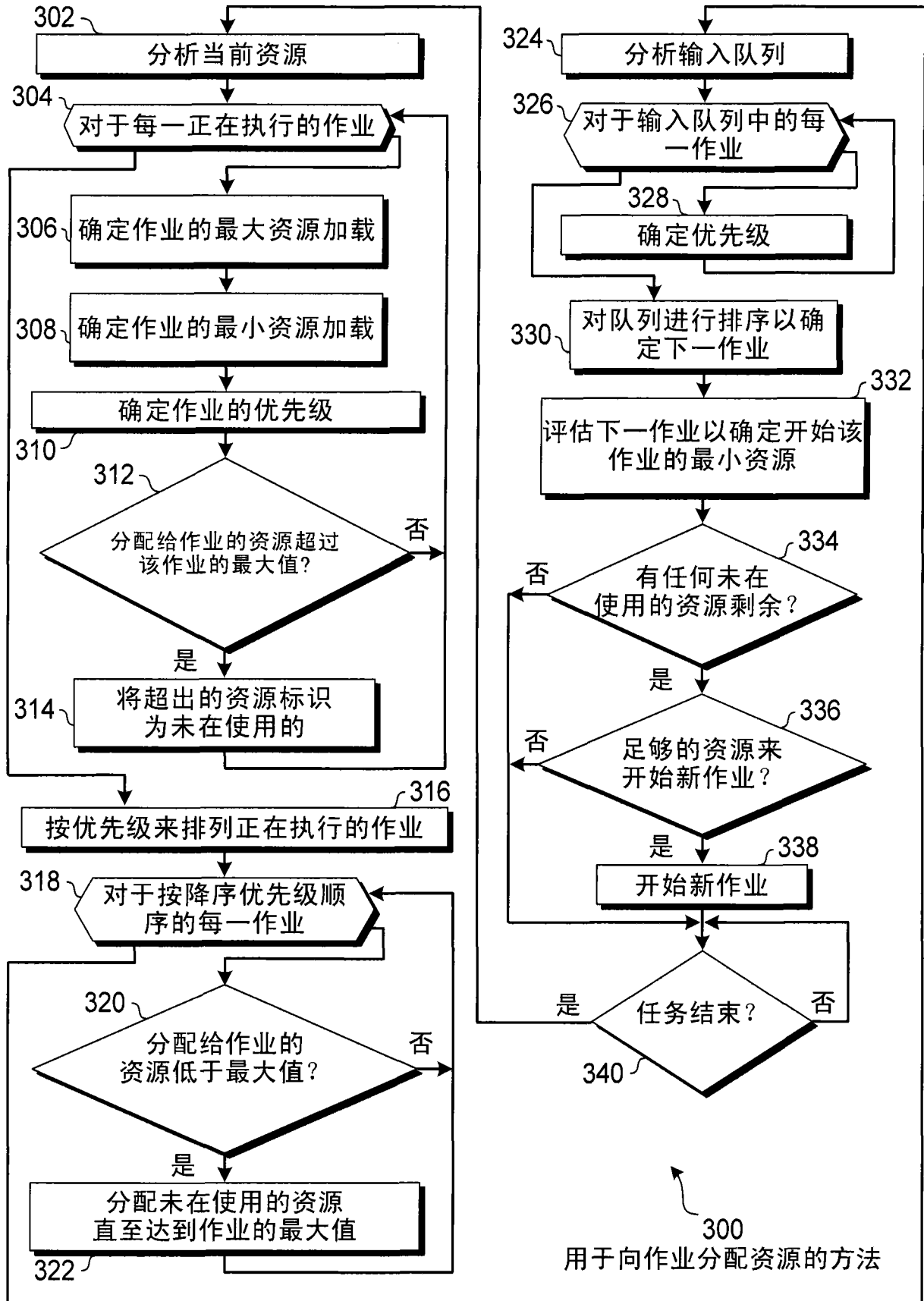


图 3