



(12) 发明专利申请

(10) 申请公布号 CN 117546145 A

(43) 申请公布日 2024. 02. 09

(21) 申请号 202280044269.2

(22) 申请日 2022.04.27

(30) 优先权数据

17/302,657 2021.05.10 US

(85) PCT国际申请进入国家阶段日

2023.12.21

(86) PCT国际申请的申请数据

PCT/US2022/026449 2022.04.27

(87) PCT国际申请的公布数据

W02022/240584 EN 2022.11.17

(71) 申请人 甲骨文国际公司

地址 美国加利福尼亚

(72) 发明人 M·P·库里纳 J·D·格里芬

陈高

(74) 专利代理机构 中国贸促会专利商标事务所

有限公司 11038

专利代理师 林小枫

(51) Int.Cl.

G06F 9/50 (2006.01)

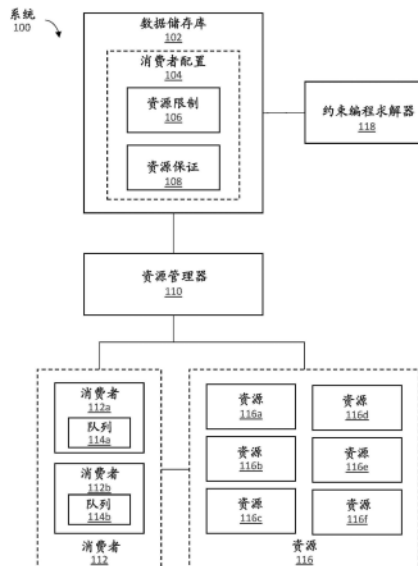
权利要求书5页 说明书22页 附图10页

(54) 发明名称

使用约束编程来设定资源分配限制以便向消费者分配资源

(57) 摘要

资源分配限制包括资源限制和资源保证。如果使用比保证的资源更多的资源,则消费者易受到其他消费者中断的影响。资源是基于资源限制和资源保证来指定和/或分配给消费者的。约束编程(CP)求解器基于资源使用数据确定使弱点和/或弱点成本最小化的资源限制和资源保证。CP数据模型包括限制元素、保证元素和弱点元素。CP数据模型还包括保证-弱点约束,保证-弱点约束依赖于根据消费者的资源使用数据生成的超越分布。CP数据模型利用约束来声明性地表达了问题的组合性质。CP是声明式编程的一种形式。



1. 一种或多种非临时性机器可读介质, 存储指令, 所述指令当由一个或多个处理器执行时导致:

确定计算系统中的一组消费者和一组资源;

获得特定时间段内分别对应于所述一组消费者的一组资源使用数据集;

其中特定资源使用数据集对应于特定消费者;

生成分别对应于所述一组消费者的一组超越分布, 其中生成对应于所述特定消费者的特定超越分布包括:

确定所述特定资源使用数据集中的数据点的相应计数, 其中所述数据点指示所述特定消费者的资源使用超过一组临界值中的每一个;

生成约束编程数据模型, 包括:

指定一组限制元素, 所述一组限制元素分别表示关于所述一组消费者的资源限制;

其中关于所述特定消费者的资源限制是允许所述特定消费者在给定时间使用的资源数量的上限;

指定一组保证元素, 所述一组保证元素分别表示关于所述一组消费者的资源保证;

其中关于所述特定消费者的资源保证是所述特定消费者在给定时间能够访问的资源数量的下限;

其中所述一组保证元素中的特定保证元素对应于所述特定消费者;

基于所述一组资源使用数据集指定一组弱点元素, 所述一组弱点元素分别表示所述一组消费者的弱点计数;

其中所述特定消费者的弱点计数是指示弱点的数据点的计数, 其中所述指示弱点的的数据点的计数基于所述特定消费者在所述特定时间段内的资源使用数据集;

其中所述一组弱点元素中的特定弱点元素对应于所述特定消费者;

指定与所述一组保证元素和所述一组弱点元素关联的一组保证-弱点约束;

其中对应于所述特定消费者的保证-弱点约束要求所述特定弱点元素被赋予计数值, 所述计数值对应于基于所述特定超越分布而被赋值给所述特定保证元素的保证值;

生成约束编程搜索指令, 包括:

指定与所述一组弱点元素关联的最小化目标函数;

将所述约束编程数据模型和所述约束编程搜索指令应用于约束编程求解器, 以获得关于所述一组消费者的所述资源限制和关于所述一组消费者的所述资源保证中的至少一者。

2. 按照权利要求1所述的一种或多种介质, 其中:

所述一组限制元素中的特定限制元素对应于所述特定消费者;

生成约束编程数据模型还包括:

指定与所述一组限制元素和所述一组保证元素关联的一组限制-保证约束;

其中对应于所述特定消费者的限制-保证约束要求所述特定保证元素等于所述特定限制元素除以超额预订率;

其中所述超额预订率等于所述一组限制元素之和除以所述资源的总数。

3. 按照权利要求1所述的一种或多种介质, 其中:

所述一组限制元素中的特定限制元素对应于所述特定消费者;

生成约束编程数据模型还包括:

指定与所述一组保证元素和所述资源的总数关联的总资源约束；
其中所述总资源约束要求所述一组保证元素之和小于或等于所述资源的总数。

4. 按照权利要求1所述的一种或多种介质,其中所述最小化目标函数使以下中的一个最小化:

所述一组弱点元素的最大值;

所述一组弱点元素之和;和

所述一组弱点元素的平均值。

5. 按照权利要求1所述的一种或多种介质,还存储使得执行以下操作的指令:

获得分别对应于所述一组消费者的一组弱点因子,其中特定弱点因子对应于所述特定消费者;

其中生成约束编程数据模型还包括:

指定一组每消费者成本元素,所述一组每消费者成本元素分别表示在所述特定时间段内由弱点引起的消费者的成本,其中特定的每消费者成本元素对应于所述特定消费者;

指定与所述一组弱点因子和所述一组弱点元素关联的一组每消费者成本约束,其中对应于所述特定消费者的每消费者成本约束要求所述特定的每消费者成本元素等于(a)所述特定弱点因子和(b)所述特定弱点元素的乘积;

其中所述最小化目标函数还与所述一组每消费者成本元素关联。

6. 按照权利要求5所述的一种或多种介质,其中所述最小化目标函数使以下中的一个最小化:

所述一组每消费者成本元素的最大值;

所述一组每消费者成本元素之和;和

所述一组每消费者成本元素的平均值。

7. 按照权利要求1所述的一种或多种介质,还存储使得执行以下操作的指令:

获得分别对应于所述一组消费者的一组弱点因子,其中特定弱点因子对应于所述特定消费者;

其中生成约束编程数据模型还包括:

指定总成本元素,所述总成本元素表示在所述特定时间段内由弱点引起的所述消费者的总成本;

指定与所述一组弱点因子和所述一组弱点元素关联的总成本约束,其中所述总成本约束要求所述总成本元素等于(a)所述一组弱点因子和(b)所述一组弱点元素的点积;

其中所述最小化目标函数还与所述总成本元素关联。

8. 按照权利要求1所述的一种或多种介质,还存储使得执行以下操作的指令:

获得分别对应于所述一组消费者的一组弱点因子,其中特定弱点因子对应于所述特定消费者;

其中生成约束编程数据模型还包括:

指定一组每消费者成本元素,所述一组每消费者成本元素分别表示在所述特定时间段内由弱点引起的所述消费者的成本,其中特定的每消费者成本元素对应于所述特定消费者;

指定总成本元素,所述总成本元素表示在所述特定时间段内由弱点引起的所述消费者

的总成本；

指定与所述一组弱点因子和所述一组弱点元素关联的一组每消费者成本约束,其中对应于所述特定消费者的每消费者成本约束要求所述特定的每消费者成本元素等于(a)所述特定弱点因子和(b)所述特定弱点元素的乘积；

指定与所述一组弱点因子和所述一组弱点元素关联的总成本约束,其中所述总成本约束要求所述总成本元素等于所述一组每消费者成本元素之和；

其中所述最小化目标函数还与(a)所述一组每消费者成本元素和(b)所述总成本元素中的至少一者关联。

9.按照权利要求1所述的一种或多种介质,其中:

所述保证-弱点约束被表示为元素约束；

所述特定保证元素充当所述元素约束的输入索引元素；

所述特定超越分布充当所述元素约束的输入值序列；和

所述特定弱点元素充当所述元素约束的输出。

10.按照权利要求1所述的一种或多种介质,其中所述特定时间段内的所述一组资源使用数据集包括测量的历史时间段内的资源使用数据。

11.按照权利要求1所述的一种或多种介质,其中所述特定时间段内的所述一组资源使用数据集包括预测的未来时间段内的资源使用数据。

12.按照权利要求1所述的一种或多种介质,还存储使得执行以下操作的指令:

向所述特定消费者指定所述一组资源的数量,所述数量等于由所述约束编程求解器针对所述特定消费者确定的资源保证。

13.按照权利要求12所述的一种或多种介质,其中为所述特定消费者指定的资源不能被所述一组消费者中的其他消费者使用。

14.按照权利要求1所述的一种或多种介质,还存储使得执行以下操作的指令:

判定所述特定消费者做出的待决进程请求的数量是否大于由所述约束编程求解器针对所述特定消费者确定的资源限制；

响应于判定待决进程请求的数量大于所述资源限制:将所述特定消费者做出的新的进程请求添加到队列,而不是向新的进程分配资源。

15.按照权利要求14所述的一个或多个介质,其中没有资源被分配给所述队列中的任何进程请求。

16.按照权利要求1所述的一种或多种介质,还存储使得执行以下操作的指令:

获得第二时间段内分别对应于所述一组消费者的第二组资源使用数据集；

基于所述第二组资源使用数据集生成分别对应于所述一组消费者的第二组超越分布；

基于所述第二组超越分布生成第二约束编程数据模型；

至少将所述第二约束编程数据模型应用于所述约束编程求解器,以获得关于所述一组消费者的更新的资源限制和关于所述一组消费者的更新的资源保证中的至少一者。

17.按照权利要求1所述的一种或多种介质,其中所述约束编程求解器基于更新的资源使用数据集,定期更新关于所述一组消费者的资源限制和关于所述一组消费者的资源保证中的至少一者。

18.按照权利要求1所述的一种或多种介质,还存储使得执行以下操作的指令:

获得分别对应于所述一组消费者的一组弱点因子,其中特定弱点因子对应于所述特定消费者;

其中所述一组限制元素中的特定限制元素对应于所述特定消费者;

其中生成约束编程数据模型还包括:

指定与所述一组限制元素和所述一组保证元素关联的一组限制-保证约束;

其中对应于所述特定消费者的限制-保证约束要求所述特定保证元素等于所述特定限制元素除以超额预订率;

其中所述超额预订率等于所述一组限制元素之和除以所述资源的总数;

指定与所述一组保证元素和所述资源的总数相关联的总资源约束;

其中所述总资源约束要求所述一组保证元素之和小于或等于所述资源的总数。

指定一组每消费者成本元素,所述一组每消费者成本元素分别表示在所述特定时间段内由弱点引起的所述消费者的成本,其中特定的每消费者成本元素对应于所述特定消费者;

指定总成本元素,所述总成本元素表示在所述特定时间段内由弱点引起的所述消费者的总成本;

指定与所述一组弱点因子和所述一组弱点元素关联的一组每消费者成本约束,其中对应于所述特定消费者的每消费者成本约束要求所述特定的每消费者成本元素等于 (a) 所述特定弱点因子和 (b) 所述特定弱点元素的乘积;

指定与所述一组弱点因子和所述一组弱点元素关联的总成本约束,其中所述总成本约束要求以下中的至少一者:

(i) 所述总成本元素等于 (a) 所述一组弱点因子和 (b) 所述一组弱点元素的点积;和

(ii) 所述总成本元素等于所述一组每消费者成本元素之和;

其中所述最小化目标函数使以下中的至少一者最小化:

所述一组弱点元素的最大值;

所述一组弱点元素之和;和

所述一组弱点元素的平均值;

所述一组每消费者成本元素的最大值;

所述一组每消费者成本元素之和;

所述一组每消费者成本元素的平均值;和

总成本元素;

其中:

所述保证-弱点约束被表示为元素约束;

所述特定保证元素充当所述元素约束的输入索引元素;

所述特定超越分布充当所述元素约束的输入值序列;和

所述特定弱点元素充当所述元素约束的输出;

其中所述特定时间段内的所述一组资源使用数据集包括 (a) 测量的历史时间段内的资源使用数据,和 (b) 预测的未来时间段内的资源使用数据中的至少一者;

向所述特定消费者指定所述一组资源的数量,所述数量等于由所述约束编程求解器针对所述特定消费者确定的资源保证;

- 其中为所述特定消费者指定的资源不能被所述一组消费者中的其他消费者使用；
- 判定所述特定消费者做出的待决进程请求的数量是否大于由所述约束编程求解器针对所述特定消费者确定的资源限制；
- 响应于判定待决进程请求的数量大于所述资源限制：将所述特定消费者做出的新的进程请求添加到队列，而不是向新的进程分配资源；
- 其中没有资源被分配给所述队列中的任何进程请求；
- 获得第二时间段内分别对应于所述一组消费者的第二组资源使用数据集；
- 基于所述第二组资源使用数据集生成分别对应于所述一组消费者的第二组超越分布；
- 基于所述第二组超越分布生成第二约束编程数据模型；
- 至少将所述第二约束编程数据模型应用于所述约束编程求解器，以获得关于所述一组消费者的更新的资源限制和关于所述一组消费者的更新的资源保证中的至少一者；
- 其中所述约束编程求解器基于更新的资源使用数据集，定期更新关于所述一组消费者的资源限制和关于所述一组消费者的资源保证中的至少一者。
19. 一种方法，包括按照权利要求1-18任意之一所述的操作。
20. 一种系统，所述系统包括硬件处理器，并被配置为进行按照权利要求1-18任意之一所述的操作。
21. 一种系统，所述系统包括用于进行按照权利要求1-18任意之一所述的操作的装置。

使用约束编程来设定资源分配限制以便向消费者分配资源

[0001] 相关申请;引用并入

[0002] 本申请涉及于2021年5月10日提交的美国非临时专利申请No.17/302,657,该申请通过引用并入本文中。

技术领域

[0003] 本公开涉及约束编程的使用。特别地,本公开涉及使用约束编程来设定资源分配限制,以便向计算系统中的消费者分配资源。

背景技术

[0004] 计算系统包括在一组消费者(例如,数据库实例)之间共享的一组资源(例如,中央处理单元(CPU))。当一个消费者对共享的一组资源提出大量需求时,该消费者可能会对其他消费者的性能产生负面影响。

[0005] 在共享有限的一组资源的消费者之间减少负面影响的一种方式是对消费者施行资源分配限制。然而,消费者与不同的进程类型、优先级和/或其他属性关联。此外,消费者的进程负载是动态变化的。在不考虑消费者的静态属性和/或动态负载的情况下设定的资源分配限制可能导致资源的低效分配。例如,一些消费者可能会超出限制,而一些资源却没有得到充分利用。

[0006] 记载在本节中的方法是可以执行的方法,但未必是以前构思或执行过的方法。于是,除非另有说明,否则不应认为记载在本节中的任何方法仅仅由于包含在本节中而成为现有技术。

附图说明

[0007] 在附图的各个图中作为例子而非限制地图解说明了各个实施例。应当注意的是,本公开中对“实施例”或“一个实施例”的引用不一定指的是同一个实施例,而是意味着至少一个实施例。附图中:

[0008] 图1图解说明按照一个或多个实施例的管理计算系统的资源管理器;

[0009] 图2A-图2B图解说明按照一个或多个实施例的约束编程数据模型和搜索指令生成系统;

[0010] 图3图解说明按照一个或多个实施例的用于基于给定历史时间段内的资源使用情况来更新关于计算系统中的消费者的资源限制和/或资源保证的一组示例操作;

[0011] 图4图解说明按照一个或多个实施例的用于生成约束编程数据模型的一组示例操作;

[0012] 图5A-图5C图解说明按照一个或多个实施例的用于生成约束编程搜索指令的多组示例操作;

[0013] 图6图解说明按照一个或多个实施例的用于将约束编程数据模型和约束编程搜索指令应用于约束编程求解器的一组示例操作;

[0014] 图7图解说明按照一个或多个实施例的用于基于由约束编程求解器更新的资源保证向消费者指定资源的一组示例操作；

[0015] 图8图解说明按照一个或多个实施例的用于对超过由约束编程求解器更新的资源限制的进程请求进行排队的一组示例操作；

[0016] 图9示出了图解说明按照一个或多个实施例的计算机系统的框图。

具体实施方式

[0017] 在下面的说明中,为了便于解释,陈述了众多的具体细节以便透彻地理解本发明。可以在没有这些具体细节的情况下实践一个或多个实施例。在一个实施例中说明的特征可以与在不同实施例中说明的特征结合。在一些例子中,参考框图形式说明了公知的结构和设备,以避免不必要地模糊本发明。

[0018] 1. 总体概述

[0019] 2. 计算系统架构

[0020] 3. 约束编程数据模型和搜索指令生成系统架构

[0021] 4. 确定和更新资源分配限制

[0022] 5. 生成约束编程数据模型和约束编程搜索指令并将其应用于约束编程求解器

[0023] 6. 施行资源保证和资源限制

[0024] 7. 硬件概述

[0025] 8. 杂项;扩展

[0026] 1. 总体概述

[0027] 一个或多个实施例包括设定资源分配限制,以便向计算系统中的消费者分配资源。资源分配限制包括资源限制和资源保证。资源限制是允许特定消费者在给定时间使用的资源数量的上限。资源保证是保证特定消费者在给定时间可以访问的资源数量的下限。如果其他消费者的资源需求突然增加,那么使用超过其保证资源的消费者就变得易受损于(vulnerable)中断。本文中使用的术语“弱点”(“vulnerability”)是指消费者的资源使用超过该消费者的资源保证。

[0028] 一个或多个实施例包括使用约束编程(CP)来设定资源分配限制以便向计算系统中的消费者分配资源。CP是声明式编程的一种形式。CP基于CP数据模型的规范以及可选的CP搜索指令来获得现实世界问题的解。相反,命令式编程是基于一系列步骤的规范。

[0029] CP数据模型包括一组数据模型元素;可以赋值给每个元素的可能值的域;以及定义被允许赋值给所述元素的值的组合的一个或多个约束。基于CP数据模型,确定满足所有约束的所述一组元素的一组值。满足所有约束的所述一组元素的一组值在本文中可以被称为“CP解”。

[0030] CP搜索指令指导一组值向一组数据模型元素的赋值,所述赋值满足由CP数据模型指定的所有约束。CP搜索指令导致产生某些序列,在这些序列中尝试将值赋值给一个或多个数据模型元素。另外或可替代地,CP搜索指令对于一个或多个数据模型元素使某些值的赋值优先于其他值。对于同一CP数据模型的不同CP搜索指令可能导致不同的CP解。另外或可替代地,对于同一CP数据模型232的不同CP搜索指令234可能导致对于获得CP解的不同效率水平和/或运行时间。

[0031] 一个或多个实施例包括生成CP数据模型,用于确定资源分配限制,以便向计算系统中的消费者分配资源。CP数据模型包括若干组数据模型元素。一组限制元素表示关于消费者的资源限制。一组保证元素表示关于消费者的资源保证。一组弱点元素表示指示弱点的数据点的计数(也称为“弱点计数”)。特定弱点元素表示:指示对于给定资源保证,特定消费者易受损的数据点的计数。此类数据点的计数基于所述特定消费者在给定历史时间段内的资源使用数据。

[0032] CP数据模型还包括若干约束。限制-保证约束将限制元素和保证元素联系起来。限制-保证约束要求(特定消费者的)保证元素等于(特定消费者的)限制元素除以超额预订率,其中超额预订率等于限制元素之和除以资源的总数。总资源约束将保证元素和计算系统中的资源的总数联系起来。总资源约束要求保证元素之和小于或等于资源的总数。保证-弱点约束将保证元素和弱点元素联系起来。保证-弱点约束要求(特定消费者的)弱点元素等于弱点计数,其中该弱点计数与基于历史时间段内的资源使用情况而被赋值(或要被赋值)给(特定消费者的)保证元素的保证值对应。

[0033] 在实施例中,保证-弱点约束被表示为元素约束。元素约束接受(a)索引元素和(b)值序列作为输入。索引元素用于索引值序列,使得元素约束从值序列中返回与被赋值(或要被赋值)给索引元素的值对应的特定值。对于保证-弱点约束,索引元素是(特定消费者的)保证元素,值序列是(关于特定消费者的)超越分布。超越分布表明指示特定消费者的资源使用超过临界值的数据点的计数,其中临界值的范围从0到计算系统中的资源的总数。此类数据点的计数基于特定消费者在给定历史时间段内的资源使用数据。

[0034] 一个或多个实施例包括生成CP搜索指令,用于确定资源分配限制,以便向计算系统中的消费者分配资源。在实施例中,CP搜索指令包括与弱点计数关联的最小化目标函数。在实施例中,CP搜索指令使一组弱点元素的最大值最小化。在再一个实施例中,CP搜索指令使一组弱点元素的平均值最小化。在又一个实施例中,CP搜索指令使弱点元素之和最小化(即,使计算系统的总弱点计数最小化)。在又一个实施例中,CP搜索指令包括与弱点计数和弱点成本关联的最小化目标函数,其中特定消费者的每消费者弱点成本是关于特定消费者的弱点因子乘以关于特定消费者的弱点元素。不同的消费者可能与不同的弱点因子关联。在实施例中,CP搜索指令使每消费者弱点成本的最大值最小化。在再一个实施例中,CP搜索指令使每消费者弱点成本的平均值最小化。在又一个实施例中,CP搜索指令使每消费者弱点成本之和最小化(即,使计算系统的总弱点成本最小化)。

[0035] 一个或多个实施例包括将CP数据模型和CP搜索指令应用于CP求解器以获得资源分配限制。资源管理器从CP求解器获得更新的资源限制和/或资源保证。基于关于特定消费者的资源保证,资源管理器向特定消费者指定资源;指定的资源的数量等于资源保证。资源管理器不允许其他消费者使用指定的资源。此外,基于关于特定消费者的资源保证和/或资源限制,资源管理器对特定消费者做出的新进程请求进行排队或分配。如果特定消费者的待决进程请求的数量等于或低于资源保证,则可以将新的进程请求优先分配给指定的资源。如果待决进程请求的数量超过资源限制,则将新的进程请求添加到特定消费者的队列;因此,资源管理器确保被分配给资源的进程请求在资源限制内。如果待决进程请求的数量高于资源保证并且等于或低于资源限制,则从特定消费者的队列中移除的进程请求(或者,如果没有队列,则新的进程请求本身)可以被分配给特定资源。资源管理器可以中断特定资

源上正在进行的由其他消费者做出的进程请求的执行,以便将该特定资源分配给特定消费者的进程请求。

[0036] 选择使用超越分布对资源分配问题进行建模使CP求解器能够有效地确定优化解。包含在历史资源使用日志中的大量数据被压缩为超越分布,通过元素约束,所述超越分布在逻辑上与保证限制完全对应。此外,超越分布和相关元素约束的使用允许CP求解器在不要求目标函数的形状、进行目标函数的导数和/或提供对CP解的初始估计的情况下寻求优化解。

[0037] CP数据模型和CP搜索指令允许CP求解器基于最近的资源使用数据有效地确定更新的资源分配限制。对资源分配限制的持续更新允许计算系统快速适应不断变化的资源使用模式。最佳资源分配限制降低了消费者出现弱点和/或资源未得到充分利用的可能性。

[0038] 在本说明书中描述和/或在权利要求中列举的一个或多个实施例可能不包括在本总体概述部分中。

[0039] 2. 计算系统架构

[0040] 图1图解说明按照一个或多个实施例的管理计算系统的资源管理器。如图1中图解所示,系统100包括资源管理器110、数据储存库102、一组消费者112、一组资源116和约束编程(CP)求解器118。在一个或多个实施例中,系统100可以包括比图1中图解所示的组件更多或更少的组件。图1中图解所示的组件可以是彼此本地的或者彼此远离的。图1中图解所示的组件可以用软件和/或硬件实现。每个组件可以分布在多个应用和/或机器上。多个组件可以组合成一个应用和/或机器。关于一个组件描述的操作可以改为由其他组件进行。

[0041] 在一个或多个实施例中,消费者112指的是为资源116产生工作负载的硬件和/或软件。消费者112的例子包括数据库实例、虚拟机和/或硬件服务器。由消费者112产生的工作负载可以用各种单位来测量,比如进程请求的数量、要传送和/或存储的数据量和/或处理时间的持续时间。

[0042] 在一个或多个实施例中,资源116指的是服务于和/或处理由消费者112产生的工作负载的硬件和/或软件。资源116的例子包括中央处理单元(CPU)、磁盘存储器和/或通信带宽。

[0043] 在一个或多个实施例中,资源管理器110指的是被配置为向消费者112分配资源116的硬件和/或软件。分配指的是将资源116分配给消费者112的请求,使得被分配给请求的资源服务于和/或处理该请求。另外或可替代地,分配指的是将资源116指定给消费者112,使得对于消费者指定的资源不能服务其他消费者的请求。

[0044] 在实施例中,资源管理器110基于包括资源限制106和/或资源保证108的消费者配置104进行分配决策。资源限制106是在任何给定时间可以分配给消费者112的资源116的最大数量。资源保证108是在任何给定时间对于消费者112保证的资源116的最小数量。应用资源限制106和/或资源保证108的操作的例子在下面参考图7-图8来进一步描述。

[0045] 不同的消费者配置104可以对应于不同的消费者112。例如,一组消费者配置可以对应于消费者112a,而不同的一组消费者配置可以对应于消费者112b。

[0046] 在一个或多个实施例中,数据储存库102是用于存储数据的任何类型的存储单元和/或设备(例如,文件系统、数据库、表的集合或任何其他存储机制)。此外,数据储存库102可以包括多个不同的存储单元和/或设备。多个不同的存储单元和/或设备可以是或者可以

不是同一类型或者可以位于或者可以不位于同一物理站点。此外,数据储存库102可以在与资源管理器110相同的计算系统上实现或执行。可替代地或另外地,数据储存库102可以在与资源管理器110分离的计算系统上实现或执行。数据储存库102可以经由直接连接或经由网络在通信上耦接到资源管理器110。描述消费者配置104的信息可以跨系统100内的任何组件实现。然而,出于清楚和解释的目的,在数据储存库102内示出了该信息。

[0047] 在一个或多个实施例中,CP求解器118指的是被配置为在给定CP数据模型232和CP搜索指令234的情况下确定CP解的硬件和/或软件。CP求解器118的进一步实施例和/或例子在下面参考图2A-图2B来描述。

[0048] 在一个或多个实施例中,图1的系统100和图2A的系统200可以组合成被配置为对资源116向消费者112的分配进行管理的单个系统。图1的CP求解器118等同于图2A的CP求解器218。(由资源管理器110使用的)图1的资源限制106和资源保证108分别等同于(由CP求解器218输出的)图2A的资源限制206和资源保证208。

[0049] 3. 约束编程数据模型和搜索指令生成系统架构

[0050] 图2A-2B图解说明按照一个或多个实施例的约束编程数据模型和搜索指令生成系统。如图2A中图解所示,系统200包括超越分布生成器222、模型和指令生成器224、数据储存库202、CP求解器218、资源限制206和资源保证208。在一个或多个实施例中,系统200可以包括比图2A中图解所示的组件更多或更少的组件。图2A中图解所示的组件可以是彼此本地的或者彼此远离的。图2A中图解所示的组件可以用软件和/或硬件实现。每个组件可以分布在多个应用和/或机器上。多个组件可以组合成一个应用和/或机器。关于一个组件描述的操作可以改为由其他组件进行。如上所述,图1的系统100和图2A的系统200可以组合成配置为向消费者分配资源的单个系统。

[0051] 在一个或多个实施例中,数据储存库202是用于存储数据的任何类型的存储单元和/或设备(例如,文件系统、数据库、表的集合或任何其他存储机制)。此外,数据储存库202可以包括多个不同的存储单元和/或设备。多个不同的存储单元和/或设备可以是或者可以不是同一类型或者可以位于或者可以不位于同一物理站点。此外,数据储存库202可以在与超越分布生成器222、模型和指令生成器224和/或CP求解器218相同的计算系统上实现或执行。可替代地或另外地,数据储存库202可以在与超越分布生成器222、模型和指令生成器224和/或CP求解器218分离的计算系统上实现或执行。数据储存库202可以经由直接连接或经由网络在通信上耦接到超越分布生成器222、模型和指令生成器224和/或CP求解器218。

[0052] 描述资源使用数据集226、超越分布228、弱点因子230、CP数据模型232和/或CP搜索指令234的信息可以跨系统200内的任何组件实现。然而,出于清楚和解释的目的,在数据储存库202内示出了该信息。

[0053] 在一个或多个实施例中,特定消费者的资源使用数据集226包括指示由特定消费者在一段时间内使用的资源量的数据点。资源使用数据集226可以以日志、电子表格和/或任何其他格式的形式存储。资源使用可以利用计数(比如所使用的CPU的计数)、容量(比如所使用的存储空间的容量)、时间(比如使用通信信道的持续时间)和/或其他单位来测量。资源使用数据集226可以包括按照某些时间间隔的数据点,所述时间间隔可以是固定的和/或变化的。每个数据点都是实际测量结果和/或聚合测量结果。实际测量结果是在给定时间点的实际测量的资源使用情况。聚合测量结果指的是在时间窗口内的实际测量结果的某种

操作(比如求和或平均)的结果。例如,监测代理可以检测每秒正在使用的CPU的实际数量。从时间=0秒开始并在时间=9秒结束,CPU使用情况的实际测量结果可以是2、3、4、4、4、2、3、4、5、6。监测代理可以确定每3秒正在使用的CPU的数量的移动平均值。在时间=2时,基于时间=0、时间=1和时间=2时的实际测量结果来计算平均值。因此,从时间=2开始并在时间=9秒结束,平均CPU使用情况可以是3.00、3.67、4.00、3.33、3.00、3.00、4.00、5:00。监测代理可以将平均CPU使用情况报告为“聚合测量结果”

[0054] 在一个或多个实施例中,超越分布228表明指示特定消费者的资源使用超过临界值的数据点的计数,其中临界值的范围从0到计算系统中的资源的总数。此类数据点的计数基于特定消费者在给定历史时间段内的资源使用数据集226。

[0055] 例如,资源使用数据集可以包括以下数据点:

[0056] 2021年1月1日上午10:00:2个CPU;

[0057] 2021年1月1日上午10:01:3个CPU;

[0058] 2021年1月1日上午10:02:4个CPU;

[0059] 2021年1月1日上午10:03:4个CPU;

[0060] 2021年1月1日上午10:04:4个CPU;

[0061] 2021年1月1日上午10:05:2个CPU;

[0062] 2021年1月1日上午10:06:3个CPU;

[0063] 2021年1月1日上午10:07:4个CPU;

[0064] 2021年1月1日上午10:08:5个CPU;

[0065] 2021年1月1日上午10:09:6个CPU。

[0066] 因此,2021年1月1日上午10:00至上午10:09的资源使用数据集的超越分布将表明:

[0067] 临界值0:10个数据点超过临界值;

[0068] 临界值1:10个数据点超过临界值;

[0069] 临界值2:8个数据点超过临界值;

[0070] 临界值3:6个数据点超过临界值;

[0071] 临界值4:2个数据点超过临界值;

[0072] 临界值5:1个数据点超过临界值;

[0073] 临界值6:0个数据点超过临界值。

[0074] 超越分布将忽略大于与无超越相关联的最小临界值(即,6)的信息。此类信息的省略是可接受的,因为0个数据点超过临界值6必然意味着0个数据点超过临界值7、8、9、10。另外或可替代地,为了确保资源保证等于或小于与无超越相关联的最小临界值,需要此类信息的省略。

[0075] 在一个或多个实施例中,弱点因子230指示与特定消费者的弱点数据点相关联的成本因子。例如,和与可中断工作负载关联的消费者相比,与不可中断工作负载关联的消费者可以与更高的弱点因子关联。和与较低优先级关联的消费者相比,与较高优先级(和/或具有较高优先级的进程)关联的消费者可以与更高的弱点因子关联。和与较低的收益关联的消费者相比,与较高的收益关联的消费者可以与更高的弱点因子关联。在假设所有消费者的弱点因子230相同的情况下,所有消费者的弱点因子230可以表示为“1”。

[0076] 在一个或多个实施例中,CP数据模型232指的是信息的特定组织、结构和/或表示。CP数据模型232利用约束来声明性地表达问题的组合性质。CP数据模型232可以被实现为软件数据结构。软件数据结构可由机器读取。CP数据模型232可以包括可由包括硬件处理器的一个或多个设备读取的一组数据和/或指令。CP数据模型232可以充当诸如CP求解器218之类的硬件和/或软件组件的输入参数。

[0077] 参见图2B,CP数据模型232包括一组数据模型元素;以及可以被赋值给每个元素的可能值的域。数据模型元素不能被赋予不在数据模型元素的域内的值。数据模型元素可以被实现为数组、向量、链表、表、软件变量、常量和/或其他软件数据结构或数据对象。

[0078] 在实施例中,CP数据模型232包括若干组数据模型元素。一组限制元素252表示关于消费者的资源限制。每个限制元素252的域的范围从0到计算系统中的资源的总数。

[0079] 一组保证元素254表示关于消费者的资源保证。每个保证元素254的域的范围从0到计算系统中的资源的总数。

[0080] 一组弱点元素256表示基于给定历史时间段内的资源使用情况的消费者的弱点计数。每个弱点元素256的域的范围从0到给定历史时间段内对应消费者的数据点的总数。

[0081] 例如,计算系统可以包括8个CPU和4个数据库实例。每个数据库实例的资源使用日志可以指示每分钟一个数据点。感兴趣的时间段可能是2021年1月1日上午10:00至上午10:09。因此,在感兴趣的时间段内,每个消费者有10个数据点。

[0082] 基于上述设置,CP数据模型可以具有以下数据模型元素和对应的域:

[0083] $L[0]: \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$

[0084] $L[1]: \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$

[0085] $L[2]: \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$

[0086] $L[3]: \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$

[0087] $G[0]: \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$

[0088] $G[1]: \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$

[0089] $G[2]: \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$

[0090] $G[3]: \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$

[0091] $V[0]: \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

[0092] $V[1]: \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

[0093] $V[2]: \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

[0094] $V[3]: \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

[0095] 其中 $L[i]$ 表示限制元素, $G[i]$ 表示保证元素, $V[i]$ 表示弱点元素,并且 i 对应于数据库实例。

[0096] 可选地,CP数据模型232包括一组每消费者成本元素258。一组每消费者成本元素258表示在给定的历史时间段内由弱点引起的消费者的相应成本。特定消费者的每消费者成本元素258表示(a)特定消费者的弱点因子和(b)给定历史时间段内特定消费者的弱点计数的乘积。特定消费者的每消费者成本元素258的域包括(a)特定消费者的弱点因子和(b)特定消费者的弱点元素的域中的每个值的每个乘积。

[0097] 例如,计算系统可以包括8个CPU和4个数据库实例。每个数据库实例的资源使用日志可以包括每分钟一个数据点。感兴趣的时间段可能是2021年1月1日上午10:00至上午10:

09。4个数据库实例的弱点因子可能分别为0.9、0.6、0.7和0.8。

[0098] 如上所述,CP数据模型可以包括以下弱点元素:

[0099] $V[0]: \{0,1,2,3,4,5,6,7,8,9,10\}$

[0100] $V[1]: \{0,1,2,3,4,5,6,7,8,9,10\}$

[0101] $V[2]: \{0,1,2,3,4,5,6,7,8,9,10\}$

[0102] $V[3]: \{0,1,2,3,4,5,6,7,8,9,10\}$ 。

[0103] 第一数据库实例的每消费者成本元素的域CC[0]可以包括0.9与V[0]的域中的每个值的每个乘积。因此,CP数据模型可以包括以下每消费者成本元素:

[0104] $CC[0]: \{0.0,0.9,1.8,3.6,4.5,5.4,6.3,7.2,8.1,9.0\}$

[0105] $CC[0]: \{0.0,0.6,1.2,1.8,2.4,3.0,3.6,4.2,4.8,5.4,6.0\}$

[0106] $CC[0]: \{0.0,0.7,1.4,2.1,2.8,3.5,4.2,4.9,5.6,6.3,7.0\}$

[0107] $CC[0]: \{0.0,0.8,1.6,2.4,3.2,4.0,4.8,5.6,6.4,7.2,8.0\}$ 。

[0108] 可选地,CP数据模型232包括总成本元素260。总成本元素260表示在给定历史时间段内由弱点引起的计算系统中的消费者的总成本。总成本元素260的域指示计算系统中的消费者的可能的总成本值。域中的最小值为0。域中的最大值是(a)每个消费者的每个弱点因子和(b)给定历史时间段内每个消费者的数据点的总数的每个乘积之和。

[0109] 例如,计算系统可以包括8个CPU和4个数据库实例。每个数据库实例的资源使用日志可以指示每个消费者每分钟一个数据点。感兴趣的时间段可能是2021年1月1日上午10:00至上午10:09。4个数据库实例的弱点因子可能分别为0.9、0.6、0.7和0.8。因此,总成本元素的域中的最大值将是 $0.9 \times 10 + 0.6 \times 10 + 0.7 \times 10 + 0.8 \times 10 = 30$ 。

[0110] 在实施例中,CP数据模型232包括若干约束。约束定义被允许赋值给一组数据模型元素(比如上面所述的限制元素252、保证元素254和弱点元素256)的值的组合。

[0111] 一组限制-保证约束262将限制元素252和保证元素254联系起来。限制-保证约束262要求(特定消费者的)保证元素254等于(特定消费者的)限制元素252除以超额预订率,其中超额预订率等于限制元素252之和除以资源的总数。限制-保证约束262可以表示为:

$$[0112] \quad G[i] = \frac{L[i]}{\left(\sum_{c \in \text{Consumers}} L[c]\right) / (\text{资源的总数})}$$

[0113] 其中L[i]表示限制元素252,G[i]表示保证元素254,L[c]表示该组限制元素,并且i对应于计算系统中的特定消费者。

[0114] 总资源约束264将保证元素254与计算系统中的资源的总数联系起来。总资源约束264要求保证元素254之和小于或等于资源的总数。总资源约束264可以表示为:

$$[0115] \quad \sum_{c \in \text{消费者}} G[c] \leq \text{资源的总数}$$

[0116] 其中G[c]表示一组保证元素254。

[0117] 一组保证-弱点约束266将保证元素254和弱点元素256联系起来。保证-弱点约束266要求(特定消费者的)弱点元素256等于如下这样的弱点计数,其中该弱点计数对应于基于给定历史时间段内的资源使用情况而被赋值(或要被赋值)给(特定消费者的)保证元素

254的保证值。

[0118] 在实施例中,保证-弱点约束266被表示为元素约束。元素约束接受(a)索引元素和(b)值序列作为输入。索引元素输入用于索引值序列,使得元素约束从值序列中返回与被赋值(或要被赋值)给索引元素的值对应的特定值。对于保证-弱点约束266,索引元素是保证元素254,并且值序列是超越分布228。此外,索引元素输出到弱点元素256。

[0119] 例如,CP数据模型可以包括以下数据模型元素:

[0120] $G[0]: \{0,1,2,3,4,5,6,7,8\}$

[0121] $V[0]: \{0,1,2,3,4,5,6,7,8,9,10\}$

[0122] 其中 $G[0]$ 表示对应于特定消费者的保证元素, $V[0]$ 表示对应于同一特定消费者的弱点元素。

[0123] 超越分布可以是: $\{10,8,7,6,4,1,1,0\}$ 。

[0124] 保证-弱点约束可以表示为元素约束,把(a) $G[0]$ 作为输入索引元素,(b)超越分布作为输入值序列,以及(c) $V[0]$ 作为输出元素。基于元素约束,赋值给 $G[0]$ 的值“0”映射到特定值序列的“10”;因此返回值“10”。赋值给 $G[0]$ 的值“1”映射到特定值序列的“8”;因此返回值“8”。赋值给 $G[0]$ 的值“2”映射到特定值序列的“7”;因此返回值“7”。返回的值必须在 $V[0]$ 的域内。

[0125] 可选地,一组每消费者成本约束268每个消费者地将弱点因子和弱点元素256联系起来。(关于特定消费者的)每消费者成本约束268是(a)(关于特定消费者的)弱点因子和(b)(关于特定消费者的)弱点元素的乘积。每消费者成本约束268可以表示为:

[0126] $CC[i] = VF[i] \times V[i]$

[0127] 其中 $CC[i]$ 表示每消费者成本元素, $VF[i]$ 表示弱点因子, $V[i]$ 表示弱点元素,并且*i*对应于计算系统中的消费者。

[0128] 可选地,总成本约束270跨计算系统中的消费者地将弱点因子和弱点元素256联系起来。总成本约束270是(a)弱点因子和(b)一组弱点元素的点积。总成本约束270可以表示为:

[0129] $TC = VF \cdot V$

[0130] 其中TC表示总成本元素,VF表示消费者的弱点因子,V表示弱点元素。

[0131] 或者,在使用每消费者成本元素的情况下,总成本约束270可以表示为:

[0132] $TC = \sum cc[c]$

[0133] 其中TC表示总成本元素, $CC[c]$ 表示一组每消费者成本元素。

[0134] 返回参见图2A,CP搜索指令234是在给定特定CP数据模型232的情况下确定最佳CP解的过程中,指导CP求解器218的指令。CP搜索指令234指导一组值向一组数据模型元素的赋值,所述赋值满足由CP数据模型120指定的所有约束。CP搜索指令234对于一个或多个数据模型元素使某些值的赋值优先于其他值。对于同一CP数据模型232的不同CP搜索指令234可能导致不同的CP解。另外或可替代地,对于同一CP数据模型232的不同CP搜索指令234可能导致对于获得CP解的不同效率水平和/或运行时间。CP搜索指令234可以被实现为软件数据结构。CP搜索指令234可以包括一组计算机可读指令。CP搜索指令234可以充当诸如CP求解器218之类的硬件和/或软件组件的输入参数。

[0135] 在实施例中,CP搜索指令234指定目标函数。目标函数是与计算系统中的弱点计数

和/或弱点成本关联的最小化函数。

[0136] CP搜索指令234可以包括使弱点元素256的最大值最小化。另外或可替代地,CP搜索指令234可以包括使弱点元素256之和最小化。另外或可替代地,CP搜索指令234可以包括使弱点元素256的平均值(例如,均值、中位数或众数)最小化。

[0137] 在考虑弱点成本的情况下,CP搜索指令234可以包括使每消费者成本元素258的最大值最小化。另外或可替代地,CP搜索指令234可以包括使每消费者成本元素258(或总成本元素260)之和最小化。另外或可替代地,CP搜索指令234可以包括使每消费者成本元素258的平均值(例如,均值、中位数或众数)最小化。可以使用附加的和/或备选的最小化函数。

[0138] 在一个或多个实施例中,模型和指令生成器224指的是被配置为生成CP数据模型232和/或CP搜索指令234的硬件和/或软件。生成CP数据模型232的操作的例子在下面参考图4来描述。生成CP搜索指令234的操作的例子在下面参考图5A-图5C来描述。

[0139] 在一个或多个实施例中,模型和指令生成器224在一个或者多个数字设备上实现。术语“数字设备”通常指的是包括处理器的任何硬件设备。数字设备可以指的是执行应用或虚拟机的物理设备。数字设备的例子包括计算机、平板电脑、膝上型计算机、桌上型计算机、上网本、服务器、web服务器、网络策略服务器、代理服务器、通用机器、功能特定的硬件设备、大型机、电视机、内容接收器、机顶盒、打印机、移动手持终端、智能电话机、个人数字助理(PDA)。

[0140] 在一个或多个实施例中,CP求解器218指的是被配置为在给定CP数据模型232和CP搜索指令234的情况下,使用约束编程技术来确定CP解的硬件和/或软件。CP解指定资源限制206和/或资源保证208。约束编程技术例如包括回溯算法、前向检查算法和约束传播。将CP数据模型232和CP搜索指令234应用于CP求解器218的操作的例子在下面参考图6来描述。

[0141] 4. 确定和更新资源分配限制

[0142] 图3中图解所示的一个或多个操作可以一起被修改、重新排列和/或省略。因而,图3中图解所示的特定操作序列不应被解释为限制一个或多个实施例的范围。

[0143] 图3图解说明按照一个或多个实施例的用于基于给定历史时间段内的资源使用情况来更新关于计算系统中的消费者的资源限制和/或资源保证的一组示例操作。

[0144] 一个或多个实施例包括判定是否已经发生用于更新资源限制和/或资源保证的触发(操作302)。模型和指令生成器(比如图2A的模型和指令生成器224)判定是否发生了用于更新资源限制和/或资源保证的触发。触发可以基于时间表。例如,时间表可以指定应每三小时更新资源限制。另外或可替代地,触发可以基于计算系统中的一个或多个消费者的工作负载。例如,触发可以是消费者的工作负载超过阈值。另一个触发可以是消费者的总工作负载超过另一个阈值。可以使用附加和/或备选的触发。

[0145] 一个或多个实施例包括确定计算系统中的消费者和资源(操作304)。跟踪系统(可以包括在图1的资源管理器110和/或模型和指令生成器中)经由通知系统、心跳系统和/或其他方法来跟踪计算系统中的消费者和资源。跟踪系统将消费者和资源的身份存储在数据储存库中。模型和指令生成器从数据储存库获得计算系统中的消费者和资源的身份。

[0146] 一个或多个实施例包括获得特定时间段内消费者的资源使用数据集(操作306)。监测系统(可以但并不必然包括在模型和指令生成器中)监测消费者的资源使用情况。监测系统以规律的时间间隔,比如每秒一次,检测、聚合、测量和/或以其他方式获得每个消费者

的资源使用情况。监测系统在每个消费者的相应资源使用数据集中针对每个资源使用测量结果创建一个数据点。监测系统将资源使用数据集存储在数据储存库中。模型和指令生成器从数据储存库中获得资源使用数据集。

[0147] 在实施例中,在用于获得更新的资源限制和/或资源保证的每次迭代中,模型和指令生成器使用来自最近历史时间段的资源使用数据集。因此,用于确定资源限制和/或资源保证的CP数据模型是基于最近的历史使用数据生成的。于是,资源限制和/或资源保证基于计算系统的消费者的不断变化的资源使用情况而自适应地更新。

[0148] 例如,可以每小时更新计算系统的资源限制和资源保证。在2021年1月1日上午11:00的第一次迭代中,模型和指令生成器可以基于从2021年1月1日上午10:00至上午10:59的资源使用日志生成CP数据模型。因此,可以基于从上午10:00至上午10:59的资源使用数据来确定资源限制和/或资源保证。在2021年1月1日中午12:00的第二次迭代中,模型和指令生成器可以基于从2021年1月1日上午11:00至上午11:59的资源使用日志生成新的CP数据模型。因此,可以基于从上午11:00至上午11:59的资源使用数据来确定资源限制和/或资源保证。于是,资源限制和/或资源保证可以基于最近的资源使用情况而被不断更新为最佳的。

[0149] 在实施例中,在用于获得更新的资源限制和/或资源保证的每次迭代中,模型和指令生成器对即将到来的时间段使用资源使用情况的预测。资源使用情况预测可以基于(a)来自最近历史时间段的资源使用数据集,(b)来自与即将到来的时间段共享特性的历史时间段的资源使用数据集,和/或(c)其他因素。与即将到来的时间段共享特性的历史时间段例如可以是与即将到来时间段相同的一天中的时间、一周中的某一天和/或一个月中的某一天相关联的历史时间段。于是,资源限制和/或资源保证基于计算系统的消费者的资源使用情况的预测而自适应地更新。

[0150] 一个或多个实施例包括确定消费者的弱点因子(操作308)。用户和/或应用指定消费者的弱点因子。弱点因子存储在数据储存库中。模型和指令生成器从数据储存库中获得弱点因子。

[0151] 一个或多个实施例包括基于资源使用数据集生成超越分布(操作310)。超越分布生成器(比如图2A的超越分布生成器222,它可以但并不必然包括在模型和指令生成器中)基于资源使用数据集生成超越分布。

[0152] 首先,超越分布生成器确定给定历史时间段内特定消费者的资源使用数据集中的数据点的计数。超越分布生成器继续进行迭代过程,其中给定历史时间段内的数据点的计数是迭代次数的上限。超越分布生成器从为0的临界值开始。超越分布生成器对指示高于临界值的资源使用的数据点的数量进行计数。超越分布生成器与临界值关联地记录计数。超越分布递增临界值并迭代上述过程。超越分布生成器迭代上述过程,直到临界值大于给定历史时间段内的数据点的计数。因此,超越分布生成器获得多对值,其中每一对包括临界值和指示高于该临界值的资源使用情况的数据点的计数。所述多对值形成特定消费者的超越分布。超越分布生成器还针对每个消费者迭代上述过程,以获得每个消费者的超越分布。

[0153] 超越分布生成器将消费者的超越分布存储在数据储存库中。模型和指令生成器从数据储存库中获得消费者的超越分布。

[0154] 一个或多个实施例包括生成CP数据模型(操作312)。模型和指令生成器生成CP数

据模型。用于生成CP数据模型的操作的例子在下面参考图4进一步讨论。

[0155] 一个或多个实施例包括生成CP搜索指令(操作314)。模型和指令生成器生成CP搜索指令。用于生成CP搜索指令的操作的例子在下面参考图5A-图5C进一步讨论。图5A-图5C分别图解说明用于生成不同的备选CP搜索指令的操作。或者,可以使用未参考图5A-图5C描述的其他CP搜索指令。

[0156] 一个或多个实施例包括将CP数据模型和CP搜索指令输入到CP求解器中,以获得资源限制和/或资源保证(操作316)。模型和指令生成器将CP数据模型和CP搜索指令输入到CP求解器(比如图2A的CP求解器218)中。用于将CP数据模型和CP搜索指令应用于CP求解器的操作的例子在下面参考图6进一步讨论。

[0157] 一个或多个实施例包括将资源限制和/或资源保证应用于计算系统(操作318)。CP求解器输出关于消费者的资源限制和/或资源保证。资源管理器(比如图1的资源管理器110)使用资源限制和/或资源保证来管理计算系统。用于基于资源保证向消费者指定资源的操作的例子在下面参考图7来描述。用于对超过资源限制的进程请求进行排队的操作的例子在下面参考图8来描述。

[0158] 5.生成约束编程数据模型和约束编程搜索指令并将其应用于约束编程求解器

[0159] 图4-图6中图解所示的一个或多个操作可以一起被修改、重新排列和/或省略。因而,图4-图6中图解所示的特定操作序列不应被解释为限制一个或多个实施例的范围。

[0160] 图4图解说明按照一个或多个实施例的用于生成约束编程数据模型的一组示例操作。

[0161] 一个或多个实施例包括指定一组限制元素,每个限制元素表示关于相应消费者的资源限制(操作402)。模型和指令生成器(比如图2A的模型和指令生成器224)指定一组限制元素。该组限制元素可被表示为向量、数组和/或任何其他数据结构。每个限制元素表示关于计算系统中的相应消费者的资源限制。

[0162] 另外,模型和指令生成器指定限制元素的域。每个域可被表示为向量、数组和/或任何其他数据结构。每个限制元素的域的范围从0到计算系统中的资源的总数。

[0163] 一个或多个实施例包括指定一组保证元素,每个保证元素表示关于相应消费者的资源保证(操作404)。模型和指令生成器指定一组保证元素。该组保证元素可被表示为向量、数组和/或任何其他数据结构。每个保证元素表示关于计算系统中的相应消费者的资源保证。

[0164] 另外,模型和指令生成器指定保证元素的域。每个域可被表示为向量、数组和/或任何其他数据结构。每个保证元素的域的范围从0到计算系统中的资源的总数。

[0165] 一个或多个实施例包括指定一组弱点元素,每个弱点元素表示关于相应消费者的弱点计数(操作406)。模型和指令生成器指定一组弱点元素。该组弱点元素可被表示为向量、数组和/或任何其他数据结构。每个弱点元素表示关于计算系统中的相应消费者的弱点计数。

[0166] 另外,模型和指令生成器指定弱点元素的域。每个域可被表示为向量、数组和/或任何其他数据结构。每个弱点元素的域的范围从0到给定历史时间段内对应消费者的数据点的总数。

[0167] 在实施例中,用户和/或应用可以指定计算系统中的某些特权消费者。用户和/或

应用可以为特权消费者的弱点计数指定某些最大值。模型和指令生成器可以按照指定的最大值设定特权消费者的弱点元素的域。另外或可替代地,用户和/或应用可以为特权消费者的资源保证指定某些最小值。模型和指令生成器可以按照指定的最小值设定特权消费者的保证元素的域。

[0168] 例如,计算系统可以包括8个CPU和4个数据库实例。每个数据库实例的资源使用日志可以指示每分钟一个数据点。感兴趣的时间段可能是2021年1月1日上午10:00至上午10:09。

[0169] 在没有任何外部指定的弱点计数的最大值的情况下,一组弱点元素可能是:

[0170] $V[0]: \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

[0171] $V[1]: \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

[0172] $V[2]: \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

[0173] $V[3]: \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ 。

[0174] 然而,如果用户已经指定第一数据库不能有任何弱点,则对于第一个数据库允许的唯一弱点计数为0。从第一数据库的弱点元素的域中移除所有大于0的值。因此,该组弱点元素可能变成:

[0175] $V[0]: \{0\}$

[0176] $V[1]: \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

[0177] $V[2]: \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

[0178] $V[3]: \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ 。

[0179] 作为另一个例子,计算系统可以包括8个CPU和4个数据库实例。每个数据库实例的资源使用日志可以指示每分钟一个数据点。感兴趣的时间段可能是2021年1月1日上午10:00至上午10:09。

[0180] 在没有任何外部指定的资源保证的最小值的情况下,一组保证元素可能是:

[0181] $G[0]: \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$

[0182] $G[1]: \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$

[0183] $G[2]: \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$

[0184] $G[3]: \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$ 。

[0185] 然而,如果用户指定第一数据库的资源保证必须大于或等于4,则0、1、2、3都不是资源保证的允许值。因此,从第一数据库的保证元素的域中移除0、1、2、3。因此,该组保证元素可能变成:

[0186] $G[0]: \{4, 5, 6, 7, 8\}$

[0187] $G[1]: \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$

[0188] $G[2]: \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$

[0189] $G[3]: \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$ 。

[0190] 作为另一个例子,可以组合上述两个例子,使得保证元素和弱点元素都可以通过外部指定的要求进行修改。组合以上两个例子,CP数据模型可以包括以下保证元素和弱点元素:

[0191] $G[0]: \{4, 5, 6, 7, 8\}$

[0192] $G[1]: \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$

[0193] $G[2]: \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$

[0194] $G[3]: \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$

[0195] $V[0]: \{0\}$

[0196] $V[1]: \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

[0197] $V[2]: \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

[0198] $V[3]: \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ 。

[0199] 一个或多个实施例包括指定一组每消费者成本元素和/或总成本元素(操作408)。

[0200] 可选地,模型和指令生成器指定一组每消费者成本元素。该组每消费者成本元素可被表示为向量、数组和/或任何其他数据结构。每个每消费者成本元素表示关于计算系统中的相应消费者的弱点成本。另外,模型和指令生成器指定每消费者成本元素的域。每个域可被表示为向量、数组和/或任何其他数据结构。特定消费者的每消费者成本元素的域包括(a)特定消费者的弱点因子和(b)特定消费者的弱点元素的域中的每个值的每个乘积。

[0201] 可选地,模型和指令生成器指定总成本元素。总成本元素表示计算系统的总弱点成本。另外,模型和指令生成器指定总成本元素的域。域可被表示为向量、数组和/或任何其他数据结构。总成本元素的域指示计算系统中的消费者的可能的总成本值。域中的最小值为0。域中的最大值是(a)每个消费者的每个弱点因子和(b)给定历史时间段内每个消费者的数据点的总数的每个乘积之和。

[0202] 一个或多个实施例包括基于资源的总数指定将限制元素和保证元素联系起来的约束(操作410)。

[0203] 模型和指令生成器生成一组限制-保证约束。限制-保证约束262要求(特定消费者的)保证元素等于(特定消费者的)限制元素除以超额预订率,其中超额预订率等于限制元素252之和除以资源的总数。

[0204] 另外,模型和指令生成器生成总资源约束。总资源约束264要求保证元素254之和小于或等于资源的总数。

[0205] 一个或多个实施例包括指定要求每个弱点元素被赋予如下这样的弱点计数的约束(操作412),其中所述弱点计数对应于被赋值给保证元素的保证值。模型和指令生成器生成要求每个弱点元素被赋予如下这样的弱点计数的约束,其中所述弱点计数对应于基于给定历史时间段内的资源使用情况而被赋值给保证元素的保证值。

[0206] 模型和指令生成器生成元素约束。模型和指令生成器将特定消费者的保证元素作为索引元素输入到元素约束。模型和指令生成器将特定消费者的超越分布(例如,在图3的操作310确定的超越分布)作为值序列输入到元素约束。模型和指令生成器将特定消费者的弱点元素设定为元素约束的输出元素。

[0207] 模型和指令生成器针对每个消费者迭代上述过程,以生成一组保证-弱点约束。

[0208] 一个或多个实施例包括指定每个消费者地和/或跨消费者地将弱点因子和弱点元素联系起来的一个或多个约束(操作414)。

[0209] 可选地,模型和指令生成器生成每个消费者地将弱点因子和弱点元素联系起来的约束。模型和指令生成器设定要求特定消费者的每消费者弱点成本元素等于(a)特定消费者的弱点因子和(b)特定消费者的弱点元素的乘积的约束。模型和指令生成器针对每个消费者迭代上述过程,以获得一组每消费者成本约束。

[0210] 可选地,模型和指令生成器生成跨计算系统的消费者地将弱点因子和弱点元素联系起来的约束。模型和指令生成器设定要求总弱点成本元素等于 (a) 消费者的弱点因子和 (b) 消费者的一组弱点元素的点积的约束。或者,在使用每消费者成本元素的情况下,模型和指令生成器设定要求总弱点成本元素等于消费者的每消费者成本元素之和的约束。

[0211] 一个或多个实施例包括生成包括数据模型元素和约束的CP数据模型(操作416)。模型和指令生成器生成CP数据模型。CP数据模型包括限制元素、保证元素、弱点元素,以及可选的每消费者成本元素和/或总成本元素。CP数据模型还包括限制-保证约束、总资源约束、保证-弱点约束,以及可选的每消费者成本约束和/或总成本约束。

[0212] CP数据模型可被存储为软件数据结构。CP数据模型可以包括一个或多个数组、向量、链表、表、软件变量、常量和/或数据对象。CP数据模型可以包括可由包括硬件处理器的一个或多个设备读取的一组数据和/或指令。

[0213] 图5A-图5C图解说明按照一个或多个实施例的用于生成约束编程搜索指令的多组示例操作。图5A-图5C分别图解说明用于生成不同的备选CP搜索指令的操作。

[0214] 参见图5A,一个或多个实施例包括生成目标函数,该目标函数使一组每消费者成本元素的最大值最小化(操作502)。模型和指令生成器(比如图2A的模型和指令生成器224)生成使一组每消费者成本元素的最大值最小化的目标函数。目标函数可以表示为:

[0215] $\min(\max(CC))$

[0216] 其中CC表示一组每消费者成本元素。

[0217] 一个或多个实施例包括生成包括目标函数的CP搜索指令(操作504)。模型和指令生成器生成CP搜索指令。CP搜索指令包括目标函数。CP搜索指令可被存储为软件数据结构。CP搜索指令可以包括一个或多个数组、向量、链表、表、软件变量、常量和/或数据对象。CP搜索指令可以包括可由包括硬件处理器的一个或多个设备执行的一组指令。

[0218] 参见图5B,一个或多个实施例包括生成目标函数,该目标函数使一组每消费者成本元素之和最小化(操作506)。模型和指令生成器生成使一组每消费者成本元素之和最小化的目标函数。目标函数可以表示为:

[0219] $\min(\sum CC[i])$

[0220] 其中CC[i]表示每消费者成本元素,并且i对应于计算系统中的消费者。

[0221] 或者,目标函数可以表示为:

[0222] $\min(TC)$

[0223] 其中TC表示总成本元素。

[0224] 一个或多个实施例包括生成包括目标函数的CP搜索指令(操作508)。上面参考操作504描述了用于生成CP搜索指令的操作的例子。

[0225] 参见图5C,一个或多个实施例包括生成目标函数,该目标函数使一组每消费者成本元素的平均值最小化(操作510)。模型和指令生成器生成使一组每消费者成本元素的平均值最小化的目标函数。目标函数可以表示为:

[0226] $\min(\text{ave}(CC))$

[0227] 其中CC表示一组每消费者成本元素。

[0228] 一个或多个实施例包括生成包括目标函数的CP搜索指令(操作512)。上面参考操作504描述了用于生成CP搜索指令的操作的例子。

[0229] 图6图解说明按照一个或多个实施例的用于将约束编程数据模型和约束编程搜索指令应用于约束编程求解器的一组示例操作。图6的操作假设CP搜索指令指定计算系统的总成本的最小化。然而,可以使用修改后的用于实现不同的CP搜索指令的操作。

[0230] 一个或多个实施例包括由CP求解器接受CP数据模型和CP搜索指令作为输入参数(操作602)。CP求解器(比如图2A的CP求解器218)接受CP数据模型和CP搜索指令作为输入参数。上面参考图4描述了用于生成CP数据模型的操作的例子。上面参考图5A-图5C描述了用于生成CP搜索指令的操作的例子。

[0231] 一个或多个实施例包括判定CP求解器是否可以基于CP数据模型和CP搜索指令返回CP解(操作604)。CP求解器在CP搜索指令的指导下,将一种或多种约束编程技术应用于CP数据模型。CP求解器132基于CP数据模型和CP搜索指令确定CP解,或者判定不存在满足CP数据模型的所有约束的CP解。

[0232] 约束编程技术例如包括约束传播、回溯搜索算法和/或前向检查算法。约束传播涉及从CP数据模型中的数据模型元素的域中消除不一致的值。回溯搜索算法涉及逐步构建CP解的候选者,包括一旦CP求解器确定候选者不可能被完成以提供有效的CP解就放弃该候选者。前向检查算法涉及尝试预见选择CP解的一个候选者而不是其他候选者的效果,或者确定尝试CP解的候选者的序列。在实施例中,当CP求解器遍历数据模型元素并将值赋值给数据模型元素时,CP求解器更新数据模型元素的域。考虑到迄今为止所做的初步分配,CP求解器从域中移除不再可能的值。CP求解器从域中移除将违反CP数据模型中的任何约束的值。

[0233] 例如,CP求解器可以遍历由CP数据模型指定的一组限制元素中的每一个,以确定CP解。CP求解器可以从特定的限制元素开始。来自特定限制元素的域的值被初步赋值给特定限制元素。CP求解器然后从CP数据模型的其他数据模型元素的域中消除与特定限制元素的初步赋予的值不一致的值。如果该初步赋值违反了由CP数据模型指定的约束(例如,另一个数据模型元素的域被完全消除),则将来自特定限制元素的域的另一个值初步赋值给特定限制元素。如果尝试了来自特定限制元素的域的所有值,并且这些所有值都不满足约束,则进行先前遍历的一个或多个限制元素的重新赋值。特别地,来自先前遍历的限制元素的域的值被初步分配给先前遍历的限制元素。基于重新赋值的需要,来自先前遍历的限制元素的域的另一个值现在被初步赋值给先前遍历的限制元素。然后,重新尝试特定的限制元素的赋值。

[0234] 此外,CP求解器可以由CP搜索指令指定的目标函数来引导。CP求解器可以基于CP搜索指令来确定尝试将值赋值给一个或多个限制元素和/或保证元素的序列。如果在值B之前尝试将值A赋值给特定限制元素,并且找到了有效的CP解,则最终确定值A的赋值,而不管值B的赋值是否会得到有效的CP解。因此,值A的赋值优先于值B。

[0235] 因此,CP求解器可以遍历每个限制元素,直到在不违反约束的情况下,每个限制元素被赋予来自相应域的限制值为止。CP求解器可以将初步赋值确定为最终赋值。CP解可以包括限制值到限制元素和/或保证值到保证元素的最终赋值。

[0236] 或者,CP求解器可以遍历每个限制元素,直到尝试了将每个限制值赋值给限制元素,但是没有任何一个赋值满足约束为止。然后,CP求解器可以确定对于CP数据模型不存在有效的CP解。CP求解器可以返回指示不存在CP解的消息。

[0237] 如果没有返回CP解,则一个或多个实施例包括使用最后确定的CP解来返回资源限

制和/或资源保证(操作614)。由于在当前迭代中没有返回CP解,因此CP求解器检索在上次迭代中返回的CP解。CP求解器将来自上次迭代的CP解识别为最后确定的CP解。CP求解器还将来自上次迭代的CP解识别为满足CP搜索指令的目标函数的CP解。在目标函数使总成本最小化的情况下,来自上次迭代的CP解是与最低总成本关联的CP解(与满足CP数据模型的约束的其他CP解相比)。该CP解指定了被赋值给限制元素的限制值和/或被赋值给保证元素的保证值。限制值用作关于计算系统中的消费者的资源限制。保证值用作关于计算系统中的消费者的资源保证。

[0238] 如果不存在最后确定的CP解(即,CP求解器不能在操作602的第一次迭代中返回CP解),则CP解生成指示对于所提供的CP数据模型不存在有效CP解的消息。

[0239] 相反,如果返回CP解,则一个或多个实施例包括判定是否已经接收到对迭代过程的中断(操作606)。如图所示,操作602-612形成用于找到与最低总成本关联的CP解的迭代过程。用户和/或应用可以中断迭代过程。例如,用户可以经由用户接口指示用户期望迄今为止确定的最佳CP解,而无需等待迭代过程完成。

[0240] 如果接收到中断,则一个或多个实施例包括使用最后确定的CP解来返回资源限制和/或资源保证(操作614)。由于在当前迭代中返回了CP解,因此CP求解器将来自当前迭代的CP解识别为最后确定的CP解。在操作614的以上描述中提供了用于基于CP解确定资源限制和/或资源保证的操作的例子。

[0241] 在实施例中,在CP求解器处于确定CP解的过程中时(在操作602-604),可能会接收到中断。中断是在当前迭代中返回CP解之前接收到的。因此,CP求解器将来自上次迭代的CP解识别为最后确定的CP解。

[0242] 基于中断的最后确定的CP解不一定是与最低总成本关联的CP解。可以接收恢复用于确定与最低总成本关联的CP解的迭代过程的请求。响应于恢复请求,可以在操作608继续进行迭代过程。

[0243] 如果没有接收到中断,则一个或多个实施例包括将与CP解关联的总成本识别为“当前最小总成本”(操作608)。CP求解器识别被赋值给与在当前迭代中在操作604获得的CP解相关联的总成本元素的总成本。该总成本被识别为“当前最小总成本”

[0244] 一个或多个实施例包括从总成本元素的域中移除大于或等于当前最小总成本的任何值(操作610)。CP求解器使用当前最小总成本作为在下次迭代期间要被赋值给总成本元素的可能总成本值的上限。CP求解器从总成本元素的域中移除任何大于或等于当前最小总成本的值。

[0245] 例如,在当前迭代期间,CP求解器可以返回当前CP解。当前CP解可能指示总成本元素被赋予总成本值“9”。CP求解器可以将“9”确定为当前最小总成本。CP求解器可以确定总成本元素的当前域是{0, 4, 5, 8, 9, 11, 12}。CP求解器可以从总成本元素的域中移除大于或等于9的任何总成本值。因此,CP求解器可以将总成本元素的域修改为{0, 4, 5, 8}。

[0246] 一个或多个实施例包括修改CP数据模型(操作612)。CP求解器修改CP数据模型以包括具有缩减的域的总成本元素。基于操作610,总成本元素的域只包括低于当前最小总成本的总成本值。

[0247] CP求解器针对修改后的CP数据模型迭代操作602-612。在操作602,CP求解器接受已经被修改的CP数据模型作为输入参数。在操作604-608,假设CP求解器基于修改后的CP数

据模型确定新的CP解,CP求解器更新当前最小总成本。CP求解器将被赋值给与新的CP解关联的总成本元素的总成本值识别为当前最小总成本。在操作610,CP求解器从总成本元素的域中移除大于或等于当前最小总成本值的任何值。在操作612,CP求解器再次修改CP数据模型。CP求解器继续迭代过程,直到CP求解器在操作604无法找到CP解,或者在操作606接收到中断为止。如上参考操作614所述,当CP求解器在操作604无法找到CP解,或者在操作606接收到中断时,CP求解器使用最后确定的CP解来返回资源限制和/或资源保证。如果CP求解器在没有中断的情况下完成迭代,则最后确定的CP解就是最优解,即,不存在成本更低的其他解。

[0248] 在一个或多个实施例中,基于不同的搜索技术,可以进行用于将约束编程数据模型和搜索指令应用于约束编程求解器的附加和/或备选操作。下面描述不同搜索技术的例子。

[0249] 在实施例中,在操作602将CP数据模型首次应用于CP求解器之前,修改CP数据模型。在总成本元素的域中,在可能的总成本值的范围内的特定值被确定为初始截止值。初始截止值例如可以是可能的总成本值的中位数。从总成本元素的域中移除所有高于初始截止值的值。因此,需要CP求解器的第一次运行来生成具有低于初始截止值的总成本值的CP解。通过设定初始截止值,如果存在有效的CP解,则CP求解器可以更有效地获得具有更低总成本的CP解。

[0250] 在实施例中,CP求解器迭代,直到找到针对给定CP数据模型的所有可能的CP解。CP求解器不基于先前的迭代来修改总成本元素的域。因此,CP求解器可能找到与先前的CP解相比,具有更高总成本值的CP解。在确定所有可能的CP解之后,CP求解器比较与每个可能的CP解关联的总成本值。CP求解器然后识别具有最低总成本值的CP解。

[0251] 在实施例中,多个CP解可以与相同的最低总成本关联。可以修改迭代过程以确定这样的CP解。在操作610,从总成本元素的域中移除任何大于当前最小总成本的值(但保持任何等于当前最小总成本的值)。在操作604,CP求解器尝试寻找尚未找到的CP解。如果不能找到另外的CP解,则在操作614,识别具有相同总成本的一组最后确定的CP解。从具有相同总成本的一组最后确定的CP解中选择一个CP解。可以使用任何选择标准。例如,CP求解器可以从具有相同总成本的一组最后确定的CP解中选择与最均匀地分布在消费者之间的资源限制关联的CP解。

[0252] 在一个或多个实施例中,可以基于不同的CP搜索指令,进行用于将约束编程数据模型和搜索指令应用于约束编程求解器的附加和/或备选的操作。下面描述不同的CP搜索指令的例子。

[0253] 在实施例中,CP搜索指令指定使一组每消费者成本元素的最大值最小化。操作602-606和612-614保持不变。然而,操作608被修改,以将一组每消费者成本元素的最大值识别为“当前最小的最大每消费者成本”。操作610被修改,以从每个每消费者成本元素的域中移除任何大于或等于当前最小的最大每消费者成本的值。

[0254] 在实施例中,CP搜索指令指定使一组每消费者成本元素的平均值最小化。CP数据模型包括附加的数据模型元素:平均成本元素。平均成本元素具有包括计算系统中的每消费者的可能平均成本的域。要求平均成本元素等于每消费者成本元素之和除以消费者的总数。操作602-606和612-614保持不变。然而,操作608被修改,以将平均成本元素的值识别为

“当前最小平均成本”。操作610被修改,以从平均成本元素的域中移除任何大于或等于当前最小平均成本的值。

[0255] 6. 施行资源保证和资源限制

[0256] 图7-图8中图解所示的一个或多个操作可以一起被修改、重新排列和/或省略。因而,图7-图8中图解所示的特定操作序列不应被解释为限制一个或多个实施例的范围。

[0257] 图7图解说明按照一个或多个实施例的用于基于由约束编程求解器更新的资源保证向消费者指定资源的一组示例操作。

[0258] 一个或多个实施例包括获得关于计算系统中的消费者的资源限制和/或资源保证(操作702)。资源管理器(比如图1的资源管理器110)基于由CP求解器(比如图2A的CP求解器218)确定的CP解,获得关于计算系统中的消费者的资源限制和/或资源保证。上面参考图6,特别是操作614描述了用于确定CP解的操作的例子。

[0259] 一个或多个实施例包括基于资源保证向消费者指定资源(操作704)。资源管理器基于资源保证向消费者指定资源。在实施例中,资源管理器在没有接收用户对资源保证的确认的情况下,基于由CP求解器确定的资源保证向消费者指定资源。

[0260] 例如,计算系统可以包括3个数据库实例,DB1、DB2和DB3,以及8个CPU,CPU1、CPU2、CPU3、CPU4、CPU5、CPU6、CPU7和CPU8。CP解可以指定以下资源保证:

[0261] 对于DB1,2个CPU;

[0262] 对于DB2,3个CPU;

[0263] 对于DB3,1个CPU。

[0264] 因此,资源管理器可以向DB1指定两个CPU;向DB2指定三个CPU;以及向DB3指定一个CPU。具体地,资源管理器可以向DB1指定CPU1和CPU3;向DB2指定CPU4、CPU6和CPU8;以及向DB3指定CPU2。指定的资源被保留以供对应的消费者使用,而不能被计算系统中的其他消费者使用。于是,使用指定资源的消费者不易受到由计算系统中的其他消费者的需求引起的中断的影响。同时,CPU5和CPU7仍然是未指定的,可用于在数据库实例之间共享。

[0265] 一个或多个实施例包括判定资源限制和/或资源保证是否已被更新(操作706)。资源管理器判定资源限制和/或资源保证是否已被更新。如果有更新,则在操作704,资源管理器重新确定资源指定。

[0266] 图8图解说明按照一个或多个实施例的用于对超过由约束编程求解器更新的资源限制的进程请求进行排队的一组示例操作。

[0267] 一个或多个实施例包括确定计算系统中的特定消费者已生成需要计算系统中的资源的新的进程请求(操作802)。资源管理器(比如图1的资源管理器110)确定计算系统中的特定消费者已生成需要计算系统中的资源的新的进程请求。特定消费者可能已将新的进程请求发送到资源管理器以进行调度。另外或可替代地,特定消费者可能已经向资源管理器发送通知消息,以向资源管理器通知新的进程请求。

[0268] 一个或多个实施例包括确定由特定消费者做出的待决进程请求的数量(操作804)。资源管理器确定(a)已由特定消费者做出并且(b)尚未完全得到服务的进程请求(包括新的进程请求)的数量。

[0269] 一个或多个实施例包括判定待决进程请求的数量是否等于或低于关于特定消费者的资源保证(操作806)。资源管理器判定待决进程请求的数量是否等于或低于关于特定

消费者的资源保证。在实施例 中,资源管理器在没有接收用户对由 CP 求解器确定的资源保证的确认的情况下,使用由 CP 求解器确定的资源保证作为阈值。

[0270] 如果待决进程请求的数量等于或低于资源保证,则一个或多个实施例包括将指定的资源分配给新的进程请求(操作 808)。资源管理器向新的进程请求分配资源。在实施例中,资源管理器被设定为只要有可能就分配指定的资源。由于待决进程请求的数量等于或低于资源保证,因此至少一个指定的资源可用。于是,资源管理器将可用的专用资源分配给新的进程请求。

[0271] 如果待决进程请求的数量高于资源保证,则一个或多个实施例包括判定待决进程请求的数量是否大于关于特定消费者的资源限制(操作 810)。资源管理器判定待决进程请求的数量是否大于关于特定消费者的资源限制。在实施例中,资源管理器在没有接收用户对由 CP 求解器确定的资源限制的确认的情况下,使用由 CP 求解器确定的资源限制作为阈值。

[0272] 如果待决进程请求的数量大于资源限制,则一个或多个实施例包括对新的进程请求进行排队(操作 814)。资源管理器对新的进程请求进行排队。计算系统中的资源不会被分配给队列中的进程请求,直到进程请求从队列中移除为止。

[0273] 当正被服务的特定消费者的进程请求的数量低于资源限制时,资源管理器可以选择从特定消费者的队列中移除进程请求。如果另一资源可用,则资源管理器可以将该可用资源分配给从队列中移除的进程请求。如果没有资源可用,则资源管理器可以可选择地中断特定资源的正在进行的由另一个消费者做出的进程请求的执行。资源管理器将特定资源分配给从特定消费者的队列中移除的进程请求。或者,如果没有资源可用,则资源管理器可以避免中断任何正在进行的执行。是否中断可取决于各种因素,其例子在下面参考操作 812 来描述。

[0274] 如果待决进程请求的数量 (a) 大于资源保证并且 (b) 小于或等于资源限制,则一个或多个实施例包括将资源分配给新的进程请求或者使进程请求排队(操作 812)。如果至少一个资源可用,并且特定消费者的队列为空,则资源管理器将可用资源分配给新的进程请求。如果至少一个资源可用,并且特定消费者的队列不为空,则资源管理器从队列的前面移除进程请求,并将可用资源分配给该进程请求。资源管理器将新的进程请求添加到队列的后面。

[0275] 如果没有资源可用,则资源管理器可以评估各种因素,以判定是否中断正在进行的由另一消费者做出的进程请求的执行。因素的例子包括与特定消费者做出的新的进程请求和正在执行的其他进程请求关联的优先级;与特定消费者和其他消费者关联的优先级;其他进程请求已被执行的时间的时长;预测的用于完成其他进程请求的执行的剩余时间;预测的用于完成由特定消费者做出的新的进程请求的执行的时间。基于各种因素,资源管理器可以中断特定资源的正在进行的进程请求的执行。资源管理器可以将特定资源分配给从特定消费者的队列中移除的进程请求(或者,如果队列为空,则分配给由特定消费者做出的新的进程请求)。或者,基于各种因素,资源管理器可以对新的进程请求进行排队,并在将资源分配给由特定消费者做出的任何进程请求之前继续等待。

[0276] 7. 硬件概述

[0277] 按照一个实施例,本文中所述的技术由一个或多个专用计算设备实现。所述专用

计算设备可以是硬连线的以实现所述技术,或者可以包括被永久编程以实现所述技术的数字电子设备,比如一个或多个专用集成电路(ASIC)、现场可编程门阵列(FPGA)或网络处理单元(NPU),或者可以包括被编程为按照固件、存储器、其他存储装置或它们的组合中的程序指令实现所述技术的一个或多个通用硬件处理器。这样的专用计算设备还可以将定制的硬接线逻辑、ASIC、FPGA或NPU与定制的编程相结合,以实现所述技术。所述专用计算设备可以是桌上型计算机系统、便携式计算机系统、手持设备、连网设备或包含硬接线和/或程序逻辑以实现所述技术的任何其他设备。

[0278] 例如,图9是图解说明可在其上实现本发明的实施例的计算机系统900的框图。计算机系统900包括用于传送信息的总线902或其他通信机构,以及与总线902耦接用于处理信息的硬件处理器904。硬件处理器904例如可以是通用微处理器。

[0279] 计算机系统900还包括耦接到总线902,用于存储信息和要由处理器904执行的指令的主存储器906,比如随机存取存储器(RAM)或其他动态存储设备。主存储器906还可以用于在由处理器904执行的指令的执行期间存储临时变量或其他中间信息。当存储在处理器904可访问的非临时性存储介质中时,这样的指令使计算机系统900变成为进行在指令中指定的操作而定制的专用机器。

[0280] 计算机系统900还包括耦接到总线902,用于为处理器904存储静态信息和指令的只读存储器(ROM)908或其他静态存储设备。设置诸如磁盘或光盘之类的存储设备910并耦接到总线902,用于存储信息和指令。

[0281] 计算机系统900可以经由总线902耦接到显示器912,比如阴极射线管(CRT),以便向计算机用户显示信息。包括字母数字键和其他键的输入设备914耦接到总线902,以便将信息和命令选择传送给处理器904。另一种用户输入设备是光标控制器916,比如鼠标、跟踪球或光标方向键,用于向处理器904传送方向信息和命令选择,以及用于控制显示器912上的光标移动。该输入设备通常在两个轴(即,第一轴(例如,x)和第二轴(例如,y))上具有两个自由度,这使该设备可以指定平面中的位置。

[0282] 计算机系统900可以使用与计算机系统结合,使计算机系统900成为或者把计算机系统900编程为专用机器的定制硬连线逻辑、一个或多个ASIC或FPGA、固件和/或程序逻辑来实现本文中所述的技术。按照一个实施例,响应于处理器904执行包含在主存储器906中的一个或多个指令的一个或多个序列,计算机系统900进行本文中的技术。这样的指令可以从其他存储介质,比如存储设备910读取到主存储器906中。包含在主存储器906中的指令序列的执行使处理器904进行本文中所述的处理步骤。在备选实施例中,代替软件指令或者与软件指令结合,可以使用硬连线电路。

[0283] 本文中使用的术语“存储介质”指的是存储使机器以特定方式操作的数据和/或指令的任何非临时性介质。这样的存储介质可以包含非易失性介质和/或易失性介质。非易失性介质例如包括光盘或磁盘,例如存储设备910。易失性介质包括动态存储器,比如主存储器906。存储介质的常见形式例如包括软盘、柔性软盘、硬盘、固态驱动器、磁带、或任何其他磁性数据存储介质、CD-ROM、任何其他光学数据存储介质、任何具有各种小孔图案的物理介质、RAM、PROM、EPROM、FLASH-EPROM、NVRAM、任何其他存储芯片或盒式存储器、内容可寻址存储器(CAM)、以及三态内容可寻址存储器(TCAM)。

[0284] 存储介质不同于传输介质,但是可以与传输介质结合使用。传输介质参与在存储

介质之间传送信息。例如,传输介质包括同轴电缆、铜线和光纤,包括构成总线902的导线。传输介质还可以采取声波或光波的形式,比如在无线电波和红外数据通信期间产生的声波或光波。

[0285] 在将一个或多个指令的一个或多个序列传送到处理器904以便执行时,可能涉及到各种形式的介质。例如,指令最初可以被携带在远程计算机的磁盘或固态驱动器上。远程计算机可以将指令载入其动态存储器中,然后使用调制解调器通过电话线路发送所述指令。计算机系统900本地的调制解调器可以在电话线路上接收所述数据,并使用红外发射器将数据转换成红外信号。红外检测器可以接收携带在红外信号中的数据,并且适当的电路可以将所述数据放置在总线902上。总线902将所述数据传送到主存储器906,处理器904从主存储器906检索并执行指令。主存储器906接收的指令可以在由处理器904执行之前或之后,视情况存储在存储设备910上。

[0286] 计算机系统900还包括耦接到总线902的通信接口918。通信接口918提供耦接到网络链路920的双向数据通信,网络链路920连接到本地网络922。例如,通信接口918可以是综合业务数字网络(ISDN)卡、线缆调制解调器、卫星调制解调器、或提供与对应类型的电话线路的数据通信连接的调制解调器。作为另一个例子,通信接口918可以是提供与兼容LAN的数据通信连接的局域网(LAN)卡。还可以实现无线链路。在任何这样的实现中,通信接口918发送和接收携带表示各种信息的数字数据流的电信号、电磁信号或光信号。

[0287] 网络链路920通常提供通过一个或多个网络到其他数据设备的数据通信。例如,网络链路920可以提供通过本地网络922到主计算机924或者到由因特网服务提供商(ISP)926操作的数据设备的连接。ISP 926又通过现在通常称为“因特网”928的全球分组数据通信网络来提供数据通信服务。本地网络922和因特网928都使用携带数字数据流的电信号、电磁信号或光信号。通过各种网络的信号,以及在网络链路920上并通过通信接口918的往来于计算机系统900传送数字数据的信号是传输介质的示例形式。

[0288] 计算机系统900可以通过网络、网络链路920和通信接口918发送消息和接收数据,包括程序代码。在因特网例子中,服务器930可以通过因特网928、ISP 926、本地网络922和通信接口918传送所请求的应用程序的代码。

[0289] 接收的代码可以在接收时由处理器904执行,和/或存储在存储设备910或其他非易失性存储装置中以便稍后执行。

[0290] 8. 杂项;扩展

[0291] 实施例涉及一种具有一个或多个设备的系统,所述设备包括硬件处理器,并被配置为进行在本文中说明和/或在以下任何权利要求中列举的任何操作。

[0292] 在实施例中,非临时性计算机可读存储介质包括指令,所述指令当由一个或多个硬件处理器执行时,导致执行在本文说明和/或在任何权利要求中列举的任何操作。

[0293] 按照一个或多个实施例,可以使用在本文中说明的特征和功能的任何组合。在上面的说明书中,参考了可能因实现而异的众多具体细节说明了实施例。因而,说明书和附图应被认为是例证性的而不是限制性的。本发明的范围以及申请人拟作为本发明的范围的内容的唯一且排他的指示物是从本申请提出的一组权利要求在提出这些权利要求的具体形式(包括任何后续修正)方面的字面上的等同范围。

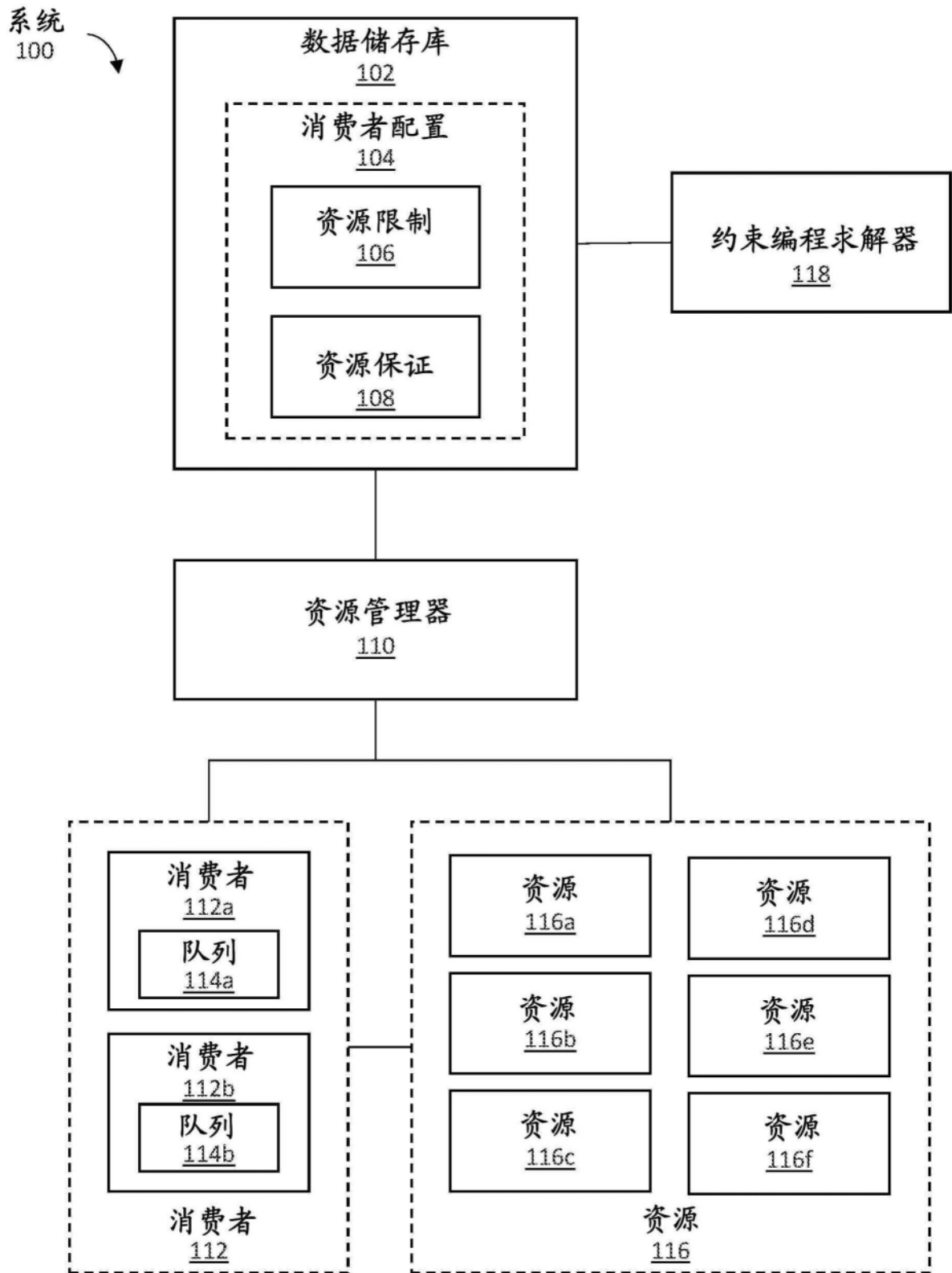


图1

系统
200

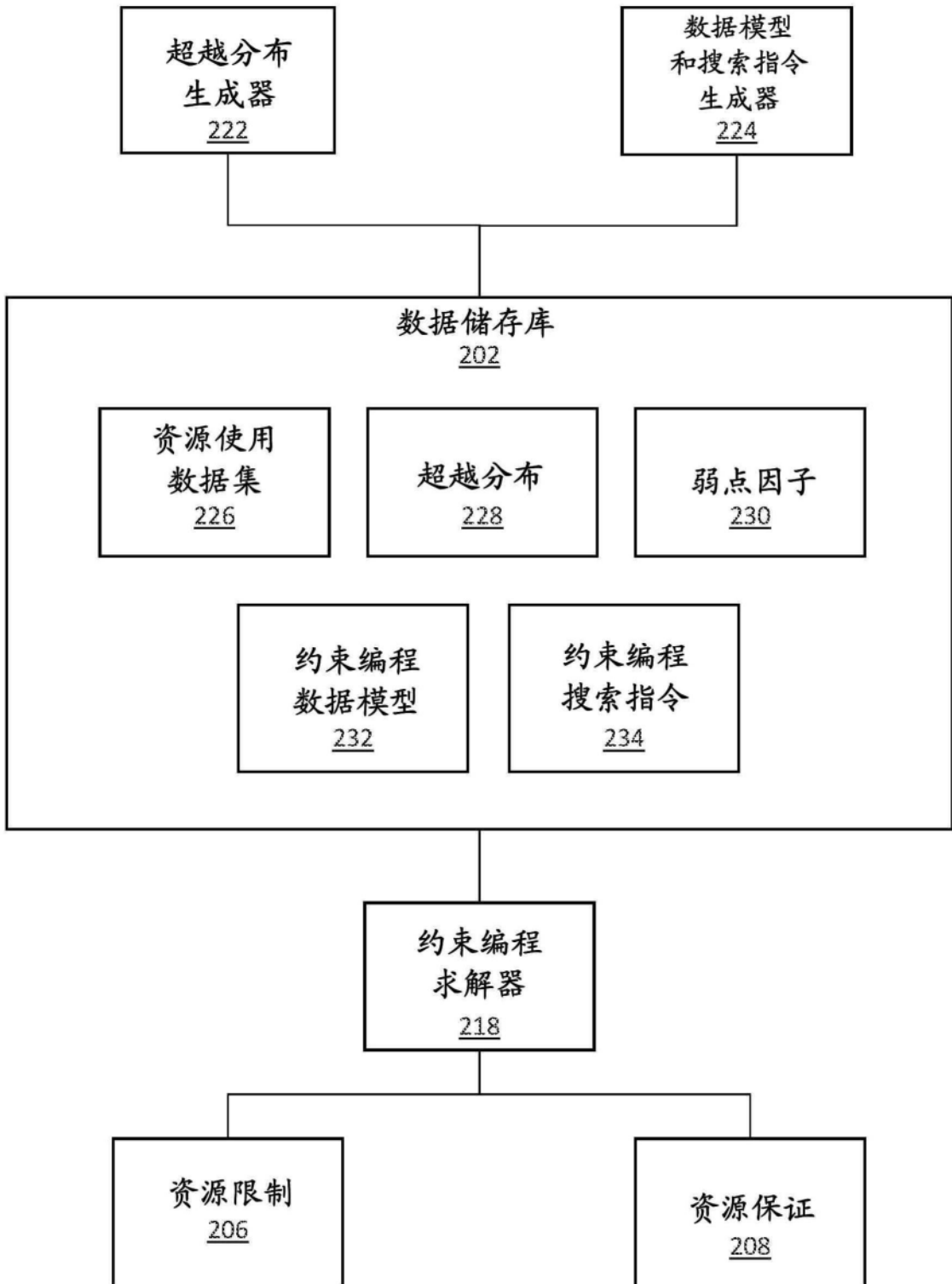


图2A

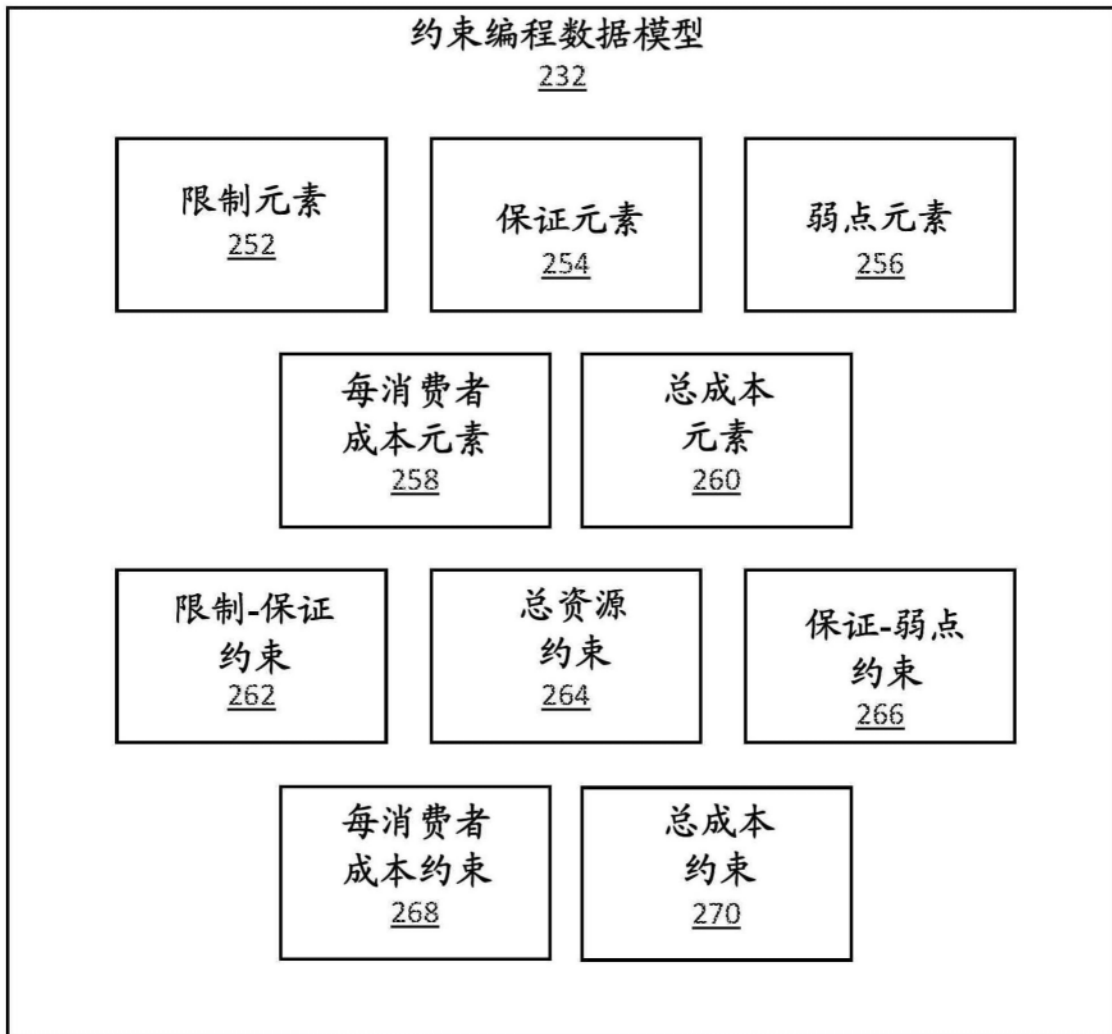


图2B

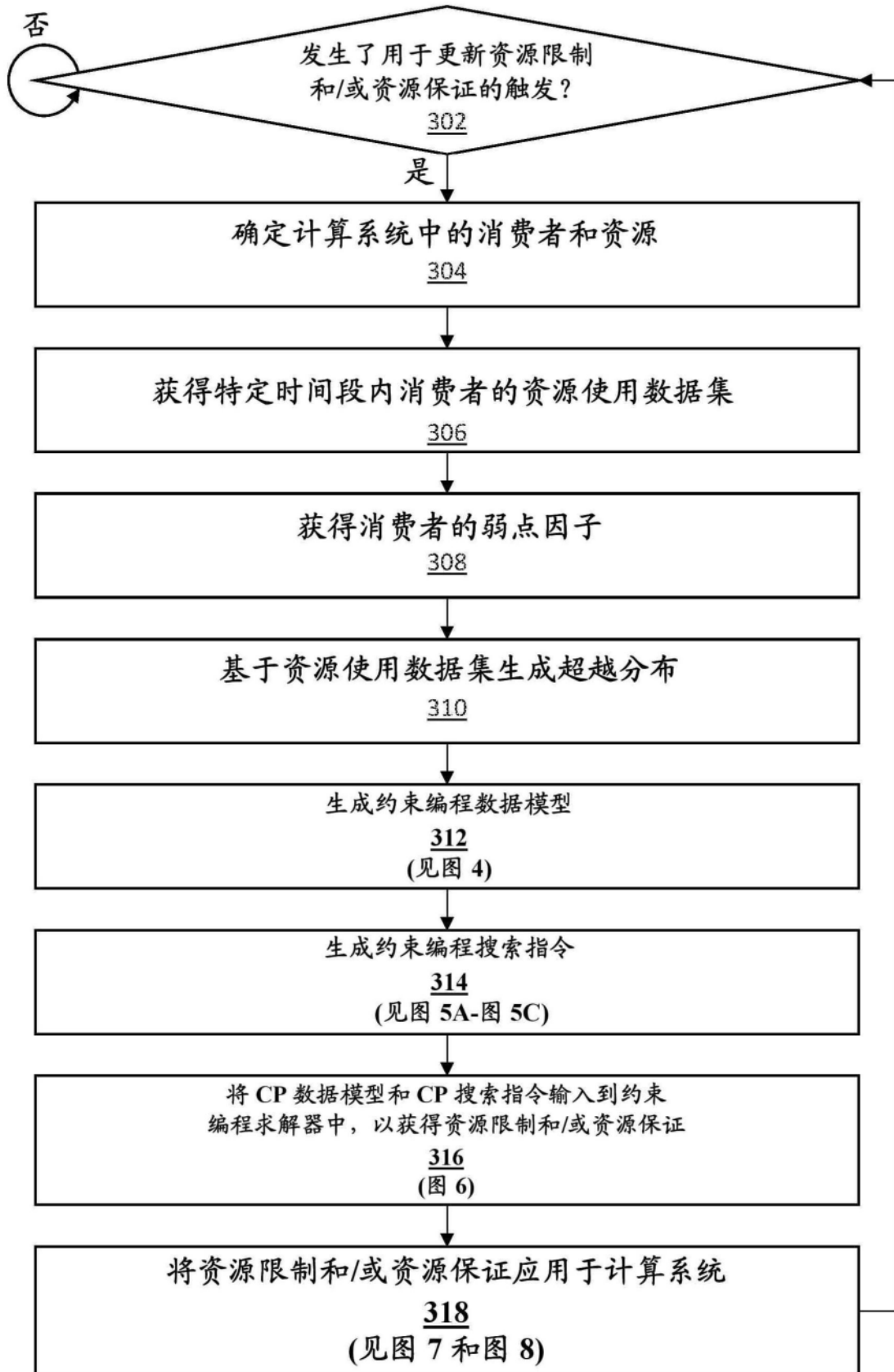


图3

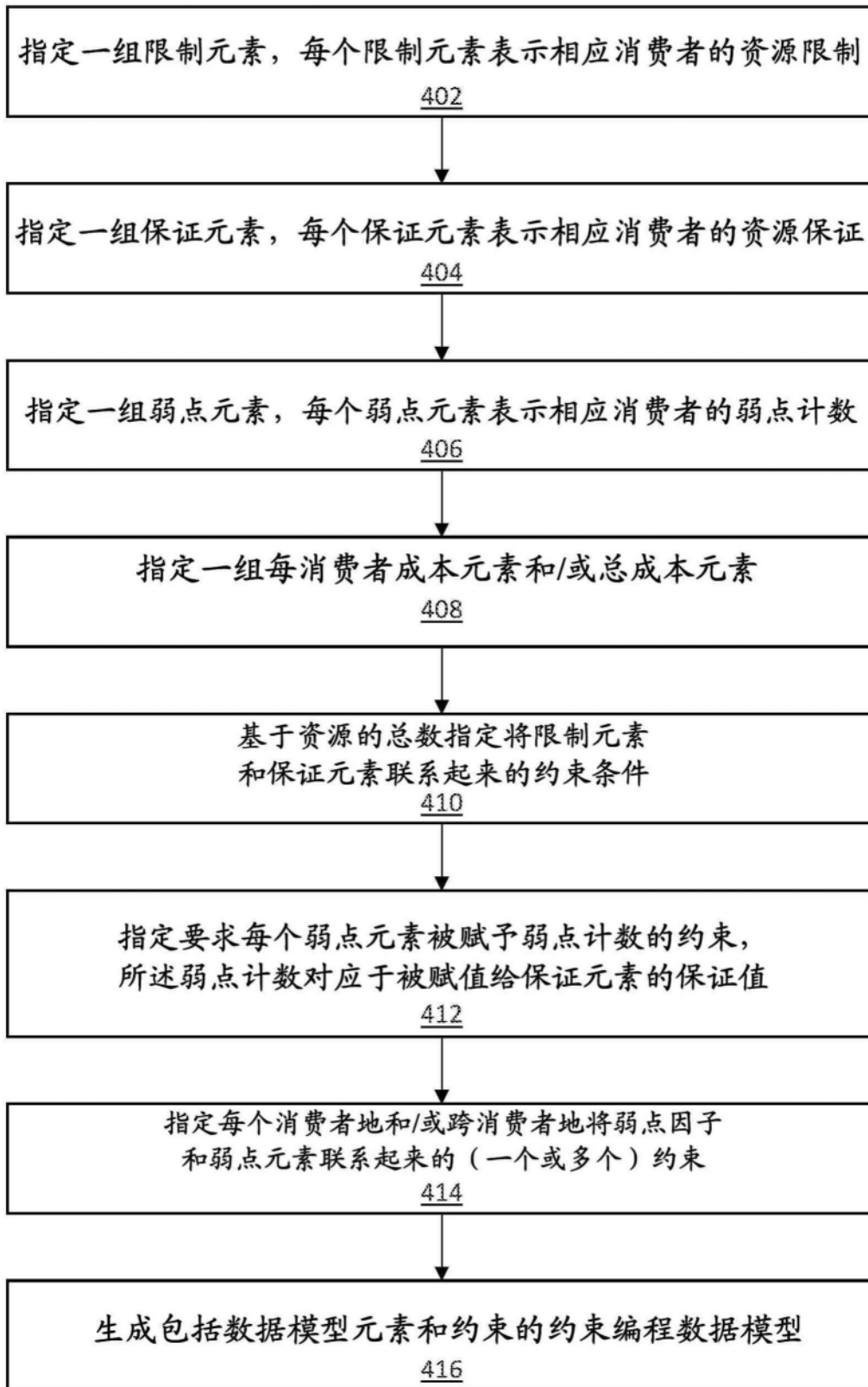


图4

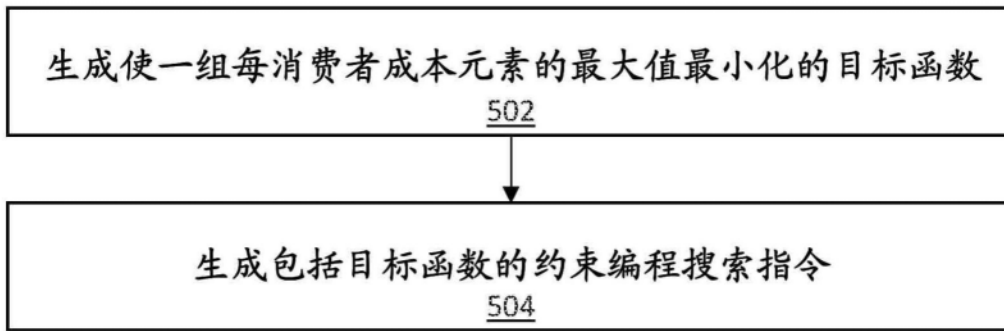


图5A

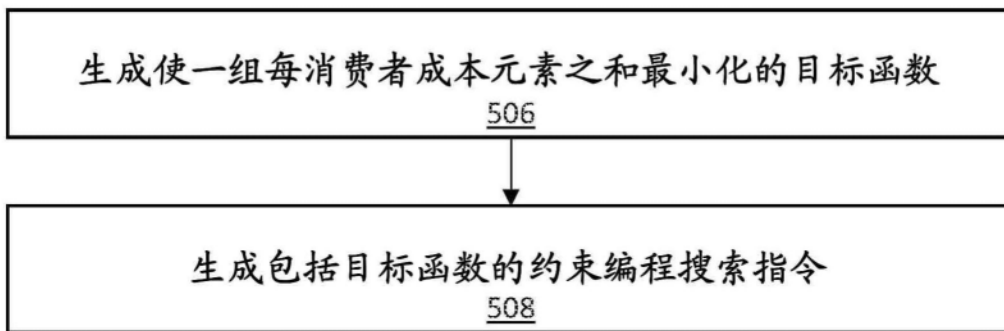


图5B

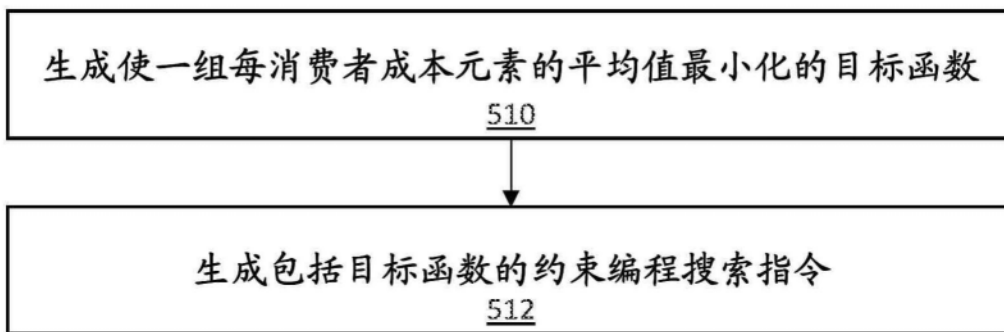


图5C

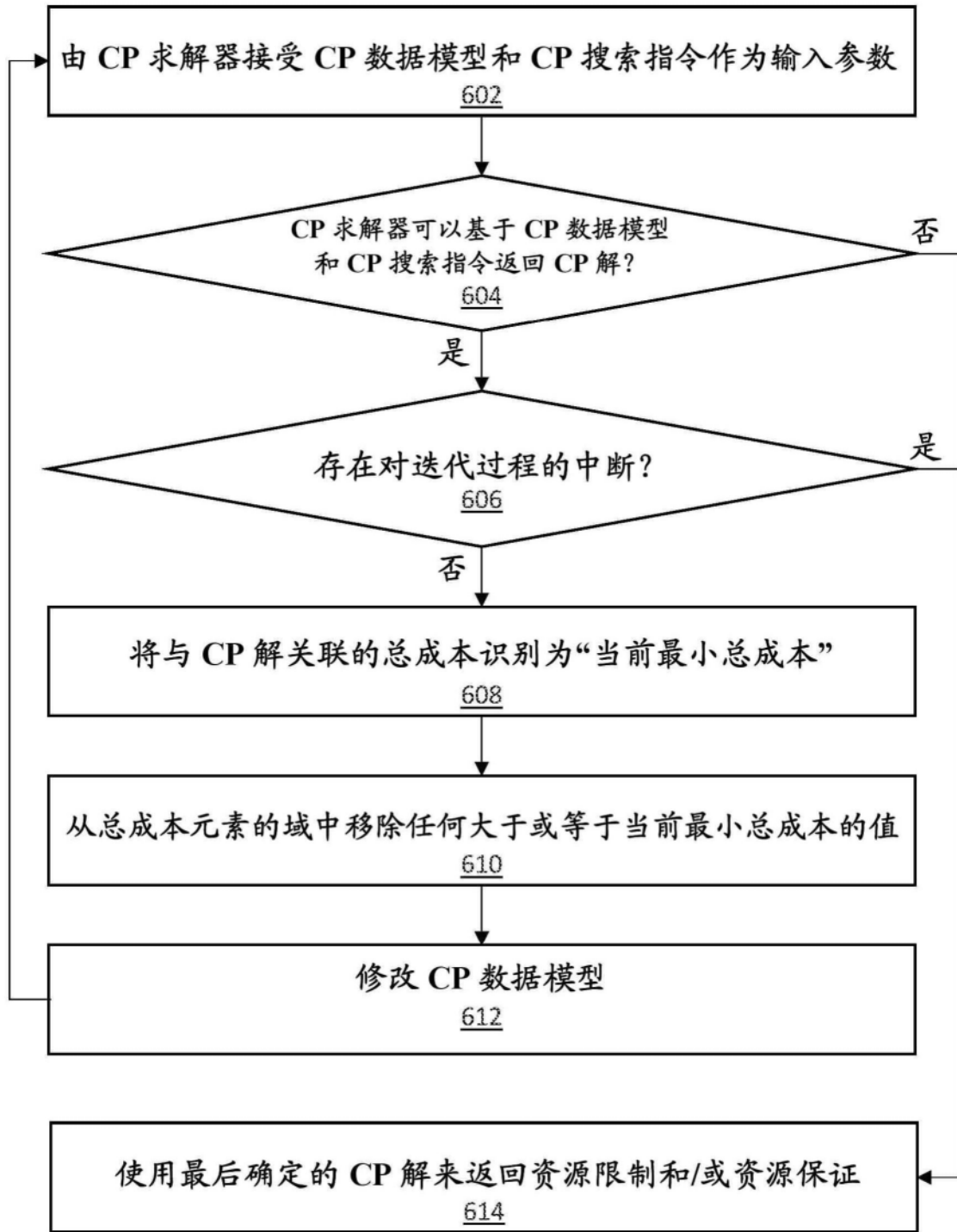


图6

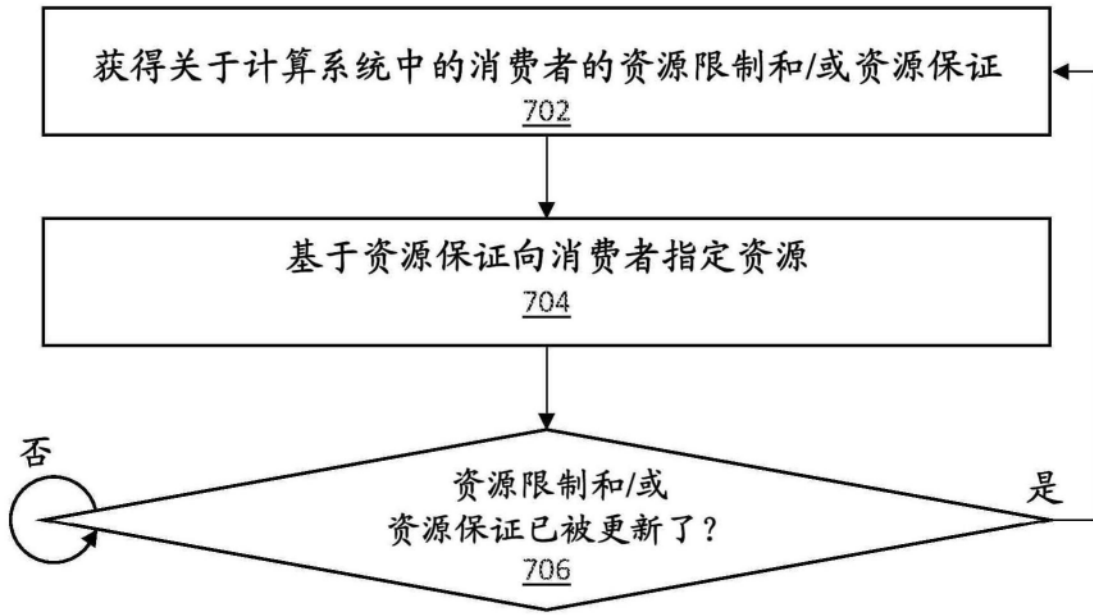


图7

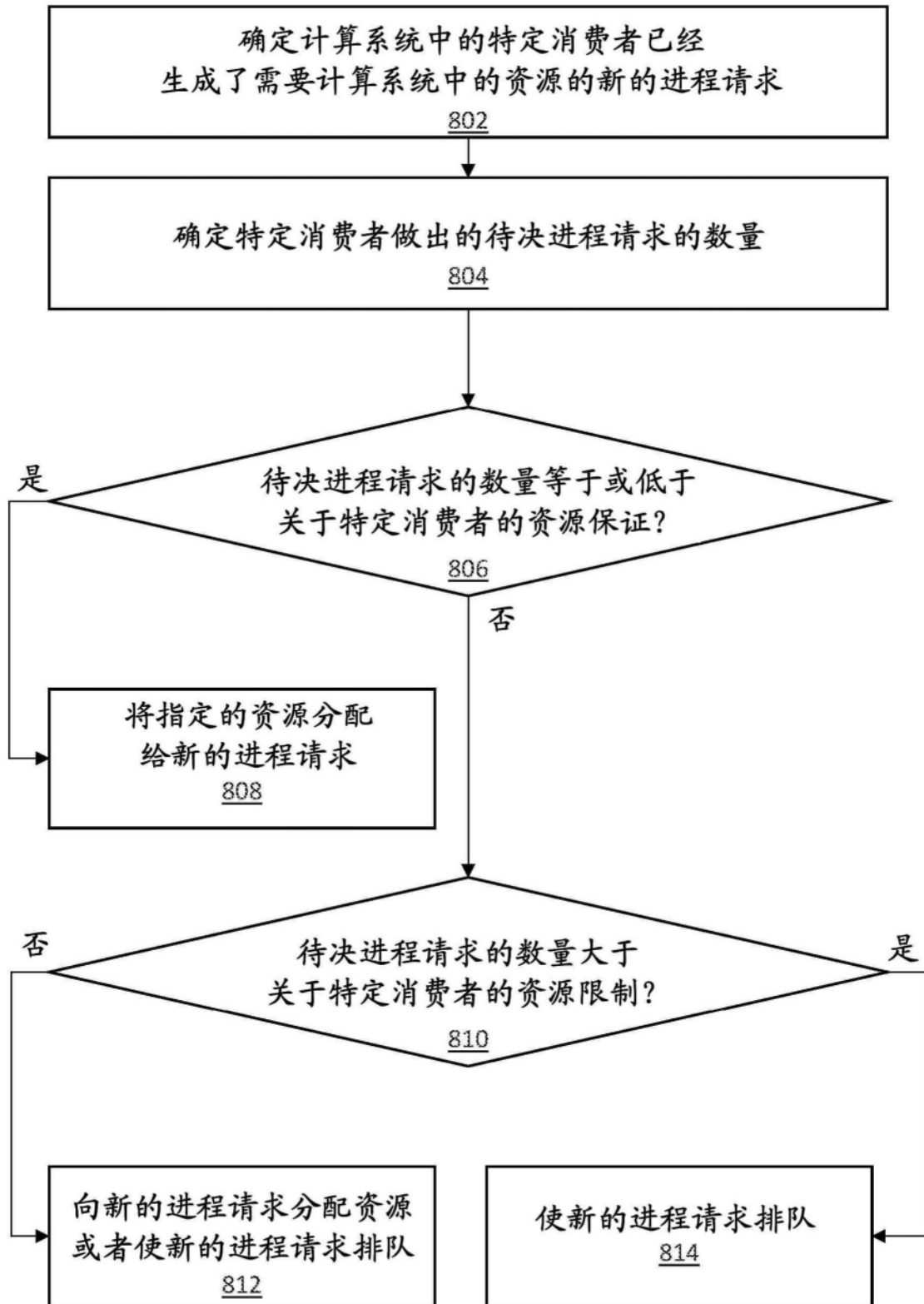


图8

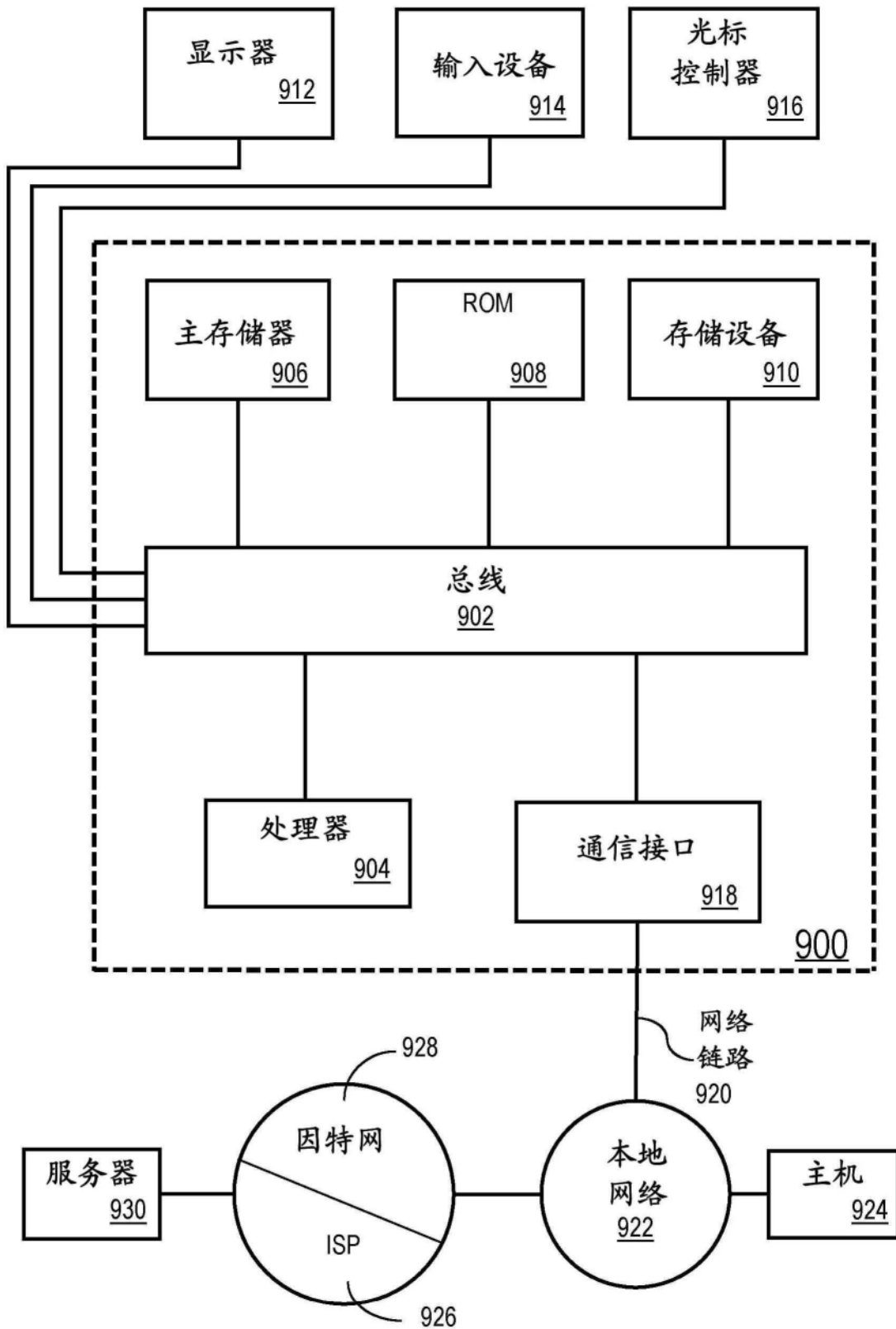


图9