

◎理论研究、研发设计◎

基于赋权有向超图的云计算依赖任务调度研究

孙凌宇¹, 冷明^{1,2}, 朱平¹, 李金忠¹SUN Lingyu¹, LENG Ming^{1,2}, ZHU Ping¹, LI Jinzhong¹

1. 井冈山大学 计算机科学系, 江西 吉安 343009

2. 加州大学洛杉矶分校 电子工程系, 美国 加利福尼亚州 90095

1. Department of Computer Science, Jinggangshan University, Ji'an, Jiangxi 343009, China

2. Department of Electrical Engineering, University of California, Los Angeles, CA 90095, USA

SUN Lingyu, LENG Ming, ZHU Ping, et al. Research of dependent task schedule of cloud computing based on weighted directed hypergraph. Computer Engineering and Applications, 2015, 51(24): 34-39.

Abstract: How to schedule dependent task efficiently is the key issue in cloud computing environment. The formal description of dependent task scheduler in cloud computing is presented. This paper adopts the weighted directed acyclic hypergraph as the mathematical model of the dependent task scheduling problem in cloud computing, whose vertex can be considered as the dependent task and directed hyperedge can be represented as the priority dependency among the tasks. Furthermore, it transforms the dependent task scheduling problem to the hypergraph partition problem and proposes the task partitioning algorithm based on the multilevel method and the weighted directed hypergraph. It also designs and implements the prototype system of the cloud computing dependent tasks schedule based on the multilevel method. It carries out the comparative experiments among the Min-Min algorithm, Max-Min algorithm and the proposed algorithm based on CloudSim simulation platform of cloud computing. The experiment and analysis show the proposed algorithm has better performance in terms of decreasing the task completing time and the improvement of resource load balancing.

Key words: cloud computing; task scheduling; weighted directed hypergraph; multi-level method; optimization algorithm

摘要: 如何对依赖任务进行高效合理的调度是云计算急需解决的关键问题之一。对云计算环境下的依赖任务调度系统进行了形式化描述。采用赋权有向无环超图来构造依赖任务调度问题的数学模型, 结点对应于依赖任务, 有向超边对应于任务之间的执行先后依赖关系。将云计算依赖任务调度问题转换为赋权有向超图的优化划分问题, 提出了基于多水平方法和赋权有向超图的依赖任务划分优化算法。设计并实现了基于多水平方法的云计算依赖任务调度原型系统。在 CloudSim 云计算仿真实验平台下, 与 Min-Min 算法、Max-Min 算法进行了对比实验, 实验数据对比表明该算法在减少依赖任务执行时间的同时, 优化了资源负载均衡性能。

关键词: 云计算; 任务调度; 赋权有向超图; 多水平方法; 优化算法

文献标志码: A **中图分类号:** TP391 **doi:** 10.3778/j.issn.1002-8331.1501-0076

1 引言

云计算的核心思想是利用分布在各地闲散异构的

大规模廉价物理资源, 整合形成巨大的虚拟资源池, 再通过网络将用户提交的计算和存储任务调度到不同的

基金项目: 国家自然科学基金(No.61363014, No.61163062, No.61063007); 江西省青年科学家培养对象计划(No.20153BCB23003); 江西省科技厅支撑项目(No.20132BBE50048); 江西省自然科学基金(No.20132BAB201035); 江西省教育厅科学技术研究项目(No.GJJ13540, No.GJJ12474); 流域生态与地理环境监测国家测绘地理信息局重点实验室资助课题。

作者简介: 孙凌宇(1976—), 女, 副教授, 研究方向为云计算, 任务调度等; 冷明(1975—), 通讯作者, 男, 博士, 副教授, 研究方向为算法分析与设计, 组合分析与优化等; 朱平(1955—), 男, 教授, 研究方向为算法分析与设计; 李金忠(1976—), 男, 博士, 副教授, 研究方向为组合分析与优化等。E-mail: lengming@jgsu.edu.cn

收稿日期: 2015-01-07 **修回日期:** 2015-03-12 **文章编号:** 1002-8331(2015)24-0034-06

CNKI 网络优先出版: 2015-07-24, <http://www.cnki.net/kcms/detail/11.2127.TP.20150724.1549.008.html>

虚拟机上,使得人们能够以极低的成本投入来提升计算能力及存储容量,获得较高的服务质量^[1]。云计算环境下任务调度的实质就是将 n 个任务按照特定的约束条件分配到 m 个闲散异构的物理资源上,使得总任务完成时间最小并且可用资源得到充分利用^[2]。任务调度作为云计算平台的重要组成部分,其效率直接影响到云计算平台的整体性能和服务质量。

任务调度问题已经被证明是一个 NP 完全问题,在 mn 个可能任务调度的解空间寻找近似最优解,使得总任务的执行时间和负载均衡度最小,其中执行时间最小是为了满足用户的服务质量,负载均衡度最小是为了保证云环境的稳定性^[3]。目前云计算的任务调度机制还未形成统一的标准和规范^[3],但由于该问题的重要性,国内外研究者提出了大量的云计算任务调度算法来计算任务调度的近似最优解,既有传统的顺序任务调度^[1]、Min-Min^[4]、Max-Min^[5]、动态规划^[6]等启发式调度算法,也有基于免疫算法^[7]、遗传算法^[8]、粒子群算法^[9]、蚁群算法^[10]和差分进化算法^[11]等智能调度算法。传统启发式调度算法以最早完成时间为目标进行调度,有着较好的负载均衡性能,但总任务的实际执行时间并非最少^[3,6]。智能调度算法通过对任务调度方案的编码,并依据遗传算法、粒子群算法、蚁群算法、免疫算法和差分进化算法等智能算法思想,在 mn 大小的解空间多样性搜索和集中性搜索之间建立平衡,最终有效地降低了任务的执行时间。

然而,大部分云计算任务调度算法仅为元任务的调度算法,既没有考虑依赖任务间的数据关联与执行先后依赖关系,也没有考虑任务的需求和虚拟机之间的差别,只是部分地解决了资源异构性和可用性问题的,缺乏普遍适用性。此外,智能调度算法在进行海量任务调度过程中,易陷入局部最优解,在收敛速度和负载均衡方面的效果有待提高。

本文提出了基于多水平方法和赋权有向超图的依赖任务划分优化算法,解决云计算环境下任务调度中执行时间和负载均衡的优化问题。首先,给出了云计算环境下的依赖任务调度系统的形式化描述;接着,借助赋权有向无环超图的多对多关系构造云计算环境下依赖任务调度问题的数学模型,结点对应于分解后的依赖任务,有向超边对应于任务之间的执行先后依赖关系,任意超边的尾端结点所对应任务的全部前驱任务都包含在该超边的源端子集中。进而将云计算依赖任务调度问题转换为赋权有向超图的优化划分问题,即每个任务子集的总指令长度与所在虚拟机的指令执行速度成比例,对应于多约束优化划分问题的平衡约束条件,任务子集之间的依赖关系数目达到最小,对应于多目标优化划分问题的最小化总割切。提出了基于多水平方法和

赋权有向超图的依赖任务划分优化算法,设计并实现了基于多水平方法的云计算依赖任务调度原型系统。最后,在 CloudSim 云计算仿真实验平台下,进行了 Min-Min 算法、Max-Min 算法和本文算法的对比实验,实验数据对比表明,本文算法在减少依赖任务执行时间的同时,优化了资源负载均衡性能。

2 云计算环境下的依赖任务调度系统的形式化描述

假设云计算环境下的依赖任务调度系统中,云资源通过虚拟化技术以虚拟机资源方式提供给用户;每个任务在任一虚拟机上的执行时间已知;具有依赖关系的任务间数据传输的通信时间已知。

定义 1 云计算环境下,用户提交作业分解成 n 个依赖任务集合,且任务之间存在着数据关联与执行先后依赖关系; m 个虚拟资源的集合参加任务调度,且虚拟资源通过虚拟机方式提供,即虚拟资源为云计算集群中的虚拟机;任务数量 n 不小于虚拟机资源数量 $m(n \geq m)$,每个任务只能分配给一个虚拟机执行,且在某一时间段一个虚拟机只能执行一个任务,不能同时执行多个任务;定义该依赖任务调度系统为 $\mathfrak{R} = \{T, \prec, D, V, C, \mathfrak{S}\}$,其中:

(1) $T = \{t_1, \dots, t_i, \dots, t_n\}$ 是 n 个存在依赖关系的任务集合,其中 t_i 为第 i 个任务 ($i = 1, 2, \dots, n$), n 为分解后的任务数量,且 t_i 的指令长度为 MI_i 。

(2) \prec 是任务集合 T 上的一个偏序关系,描述依赖任务之间的执行先后依赖关系,即如果 $t_i \prec t_j$,则意味着任务 t_i 必须在任务 t_j 之前开始执行。

(3) D 是一个 $n \times n$ 的通信矩阵,其中 d_{ij} 表示第 i 个任务 t_i 传输给第 j 个任务 t_j 数据的通信时间。

(4) $V = \{v_1, \dots, v_j, \dots, v_m\}$ 是虚拟机集合,其中 v_j 为第 j 个虚拟机资源 ($j = 1, 2, \dots, m$), m 为虚拟机数量,且 v_j 的指令执行速度(每秒执行指令条数)为 $MIPS_j$ 。

(5) C 是一个 $n \times m$ 的时间矩阵,其中 c_{ij} 表示第 i 个任务 t_i 在第 j 个虚拟机 v_j 上执行的时间且 $c_{ij} = MI_i / MIPS_j$,即预期执行时间 c_{ij} 等于任务 t_i 的指令长度 MI_i 除以虚拟机 v_j 的每秒执行指令条数 $MIPS_j$ 。

(6) \mathfrak{S} 是 n 个不同依赖任务 $T = \{t_1, \dots, t_i, \dots, t_n\}$ 调度到 m 个不同虚拟机 $V = \{v_1, \dots, v_j, \dots, v_m\}$ 上所有可能的任务分配方案集,其中 X 代表任务分配方案集 \mathfrak{S} 中的一种分配方案,即一个 $n \times m$ 的矩阵。其中, x_{ij} 表示为任务 t_i 与虚拟机 v_j 的分配关系,且 $x_{ij} \in \{0, 1\}$, $\sum_{j=1}^m x_{ij} = 1$, $i \in \{1, 2, \dots, n\}$, $j \in \{1, 2, \dots, m\}$ 。即如果任务 t_i 分配在

虚拟机 v_j 上执行, 则 $x_{ij} = 1$, 否则 $x_{ij} = 0$ 。

$$X = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ \vdots & \vdots & & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nm} \end{pmatrix}$$

定义 2 对于某任务分配方案 X , 定义虚拟机 v_j 的完成时间 VT_j 为分配给虚拟机 v_j 所有任务的预期完成时间、分配至虚拟机 v_j 的任务与分配至其他虚拟机的任务之间通信时间两者之和, 即 $VT_j = \sum_{i=1}^n x_{ij} \times c_{ij} + \sum_{i=1}^n d_{ij} \times (1 - c_{ij})$ 。如果某两个任务分配至同一个虚拟机, 则它们的通信时间视为零。

定义 3 对于 n 个不同依赖任务 $T = \{t_1, \dots, t_i, \dots, t_n\}$ 调度到 m 个不同虚拟机 $V = \{v_1, \dots, v_j, \dots, v_m\}$ 的任务调度问题是寻找分配方案 X , 使得该分配方案中虚拟机的任务最迟完成时间最早 $\min_{X \in N} \left(\max_{j=1}^m (VT_j) \right)$, 或者说各虚拟机的最长完成时间最短 $\min_{X \in N} \left(\max_{j=1}^m \left(\sum_{i=1}^n x_{ij} \times c_{ij} + \sum_{i=1}^n d_{ij} \times (1 - c_{ij}) \right) \right)$ 。

显然, 云计算环境下依赖任务调度的关键环节是将大量分散的任务按照特定的约束条件划分为多个规模较小的任务子集, 使划分后的各个任务子集计算规模与所在虚拟机的指令执行速度成比例, 且任务子集之间的通信时间达到最少, 从而最大程度地实现云计算平台的负载均衡和缩短整个任务完成的时间跨度。任务的划分结果对整个云计算环境的运行效率有着重要的影响, 可有效地减少资源空闲时间, 提高资源的利用效益。

3 云计算环境下基于赋权有向无环超图的依赖任务调度模型

由于云计算环境中依赖任务之间存在着数据关联与执行先后依赖关系, 因此本文采用赋权有向无环超图 $H = (U, E)$ 来构造依赖任务调度的数学模型, 具体映射如下:

(1) 结点集合 U 表示云计算任务集合 T , 即任务 t_i 表示为结点 u_i 。

(2) 结点 u_i 的权值表示任务 t_i 的指令长度 MI_i 。

(3) 有向超边集合 E 表示任务间的执行先后依赖关系 $<$, 其中每条超边可以连接两个以上的结点, 超边尾端结点的所有直接前驱结点都包含在该超边的源端子集中, 即一个任务必须接收到它的所有前驱任务数据后才能开始执行。

(4) 超边 e_j 的权值表示该超边尾端结点对应任务所需的通信时间, 等于该超边源端子集中对应的每个任务

传递给该超边尾端结点对应任务的通信时间累加和。

相比有向无环图而言, 赋权有向超图为云计算环境下依赖任务的调度提供了更为精确的模型。该模型的多对多关系提供了描述依赖任务的手段, 结点对应于分解后的依赖任务, 有向超边对应于任务之间的执行先后依赖关系, 任意超边的尾端结点所对应任务的全部前驱任务都包含在该超边的源端子集中。赋权有向超图能全面地表示云计算环境的异构性、分布性、广域性等特点, 从而提高依赖任务调度的准确性和执行效率。

4 云计算环境下的依赖任务划分优化算法

伴随着云计算将计算能力作为服务供用户购买的运营目标的实现, 云计算的规模将非常巨大。同时, 区别于网格计算的独占式资源分配方式, 整个云计算环境中的资源会被所有用户同时共享, 以保证对延迟敏感的任务在云上也能够很好地运行, 这意味着云计算中用户提交的任务会被分解为进程甚至线程的粒度级别。所以, 云计算环境下任务调度问题的规模不断地增长, 对划分的速度、准确性和能力有了更高的要求。

4.1 基于多水平方法和赋权有向超图的依赖任务划分优化算法

Karypis 针对结点规模达到几百万的划分问题, 提出了多水平划分的概念, 在相对较短的时间可以得到高质量的划分, 并实现了相应的超图优化划分算法 hMETIS^[12-13]。多水平划分包含粗化、初始划分和迁移优化三个阶段。在粗化阶段, 将某些结点合并在一起从而减小超图的规模, 重复此过程来构建多水平层的粗化超图, 直到粗化超图足够小为止。然后进入初始划分阶段, 对最小粗化超图进行初始划分。最后进入迁移优化阶段, 将初始划分从最小粗化超图投影回初始超图, 在每一水平层的粗化超图中对划分进行优化。

本文将云计算环境下依赖任务调度问题转换为赋权有向超图的优化划分问题, 并提出了基于多水平方法和赋权有向超图的依赖任务划分优化算法。调度问题转换为优化划分问题的映射关系如下:

(1) 赋权有向超图的划分要求每个任务子集的总指令长度与所在虚拟机的指令执行速度成比例, 对应于多资源约束条件下的平衡约束条件。

(2) 赋权有向超图的划分结果使得任务子集之间的通信时间达到最小, 对应于多目标优化划分问题的最小化总割切。

4.2 多水平方法的计算效率分析

设任务调度的任务数为 n , 虚拟机数为 m , 且通常情况下任务调度测试基准满足 $n > m$ 。由于任务调度问题已经被证明是一个 NP 完全问题, 典型的任务调度算法最坏情况下时间复杂度为 $O(n^m)$ ^[9,14]。

Karypis 在文献[12-13]中,已经在理论上解释了多水平方法的可行性和计算效率。首先,在多水平方法的粗化阶段,由于最小粗化超图有效地减少了初始超图中超边的条数和结点的数目,因此对最小粗化超图所做的任何一种划分,在初始超图上的划分投影质量都要好于在相同时间内直接对初始超图所做的划分。其次,在多水平方法的迁移优化阶段,粗化超图中迁移一个边界结点相当于移动细化超图中一组结点。因此通过移动粗化超图中的多结点,可以有效地解决迁移细化超图中结点组的问题。这不仅加快结点的迁移速度,还通过移动一组结点增强了迁移优化算法的逃离局部最优能力。因此,相比典型调度算法的 $O(n^m)$ 时间复杂度^[9,14],多水平方法的时间复杂度得到了较大的改进。

5 云计算依赖任务调度系统的设计与实现

本文设计并实现了基于多水平方法的云计算依赖任务调度原型系统,提高云计算的任务调度性能和资源利用率。该系统已申请发明专利并处于公示过程中(公示号:CN103885839A),其创新之处体现在将基于多水平方法的多目标赋权有向超图优化划分应用在云计算环境下的依赖任务调度问题中,并通过依赖任务到赋权有向超图的转换,避免直接在任务上划分,最终提高云计算任务调度的效率。该系统采取的技术路线如图1所示,具体步骤如下:

(1)输入云计算环境下用户提交的任务,并对其进行应用类型和类度的分析,确定任务的并行化程度及特点。

(2)根据云计算的资源共享分配方式等独特性质,对用户任务按照进程粒度级别进行分解,并对分解后的任务进行资源特性分析。

(3)对分解后的任务建立描述其资源需求及依赖关系的赋权有向超图模型,并按照压缩存储格式进行保存。

(4)调用多目标赋权有向超图优化划分子系统,经过多水平方法的粗化阶段、初始划分阶段和投影优化阶段,对赋权有向超图进行多目标划分,将结果存储在超图优化划分文件中。

(5)检测到划分完成后,读取相应的划分结果,根据划分信息构造任务子集,并借助有向无环超图的拓扑排序算法对任务子集进行调度。

5.1 多水平粗化阶段

在多水平粗化阶段,对赋权有向超图 $H_0=(U_0, E_0)$ 中的结点进行匹配,将它们结合在一起得到下一级粗化赋权有向超图,重复此过程得到一系列粗化赋权有向超图 $H_1, H_2, \dots, H_k, \dots, H_l$, 其中 $|U_1|>|U_2|>\dots>|U_k|>\dots>|U_l|$, 直至粗化赋权有向超图足够小为止。

在粗化阶段,两两结点选择结合的过程从本质上说就是结点的匹配过程,该调度系统除提供RM、SHEM匹配策略外,还提出了一种基于核排序的重边匹配策略,借助核值的全局信息,改进以往仅利用结点的度等局部信息进行匹配的粗化策略^[15]。

5.2 多水平初始划分阶段

在多水平初始划分阶段,对粗化阶段获得的最小赋权有向超图 $H_l=(U_l, E_l)$ 进行划分得到一个初始划分。由于最小赋权有向超图有效地减少了初始超图超边的

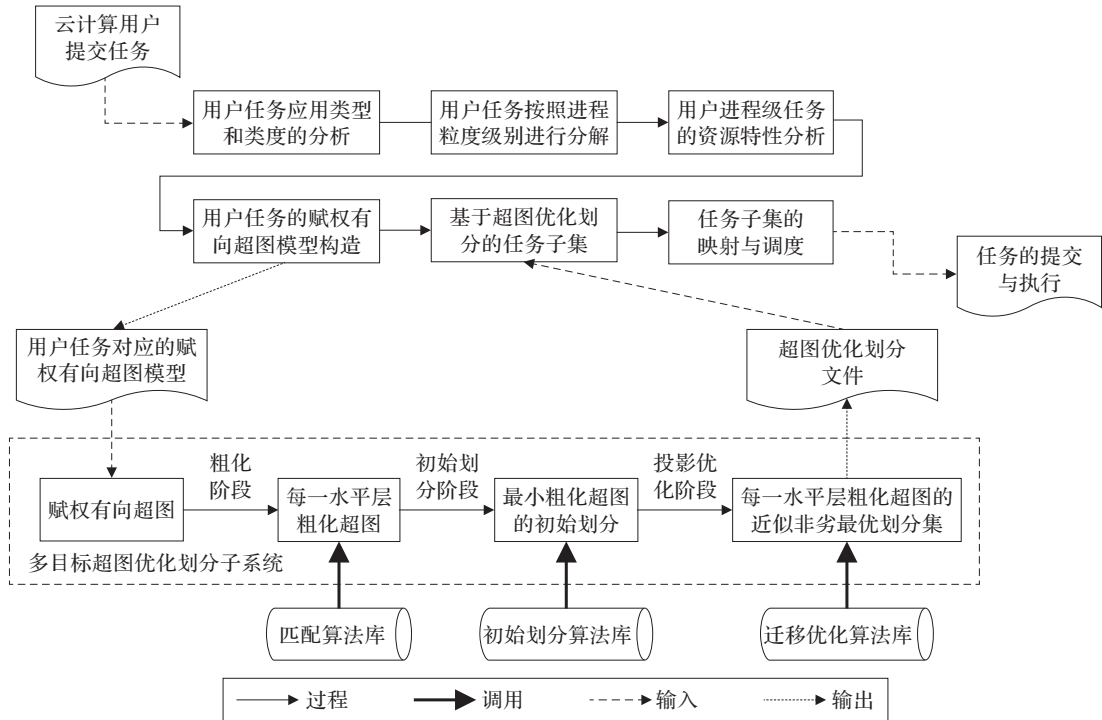


图1 基于多水平方法的云计算依赖任务调度系统的技术路线图

条数和结点的数目,对最小赋权有向超图所做的任何一种划分,在初始超图上的划分投影质量都要好于在相同时间内直接对初始超图所做的划分。

由于最小粗化超图结点数一般为20~40左右,所以其相应的一致超图的Laplace矩阵次小特征值对应的特征向量计算量相当有限。该调度系统采用谱方法对最小粗化超图进行划分,基于其特征向量刻画结点间相对距离的思想,为优化阶段提供更好的初始划分。

5.3 多水平投影优化阶段

在投影优化阶段,将在初始划分阶段得到的最小赋权有向超图 $H_l=(U_l, E_l)$ 的划分投影到上一级细化赋权有向超图 $H_{l-1}=(U_{l-1}, E_{l-1})$,即赋权有向超图 H_{l-1} 的划分由赋权有向超图 H_l 的划分投影得到。依此类推,直至得到原始输入赋权有向超图 H_0 的划分。在优化阶段还需对每一水平层粗化图 H_k 的划分 P_k 进行迁移优化,即在划分 $P_k=(U_k^1, U_k^2)$ 中挑选两个结点子集 $U_k^{1'} \subset U_k^1$ 和 $U_k^{2'} \subset U_k^2$,使得划分 $P_k'=((U_k^1 - U_k^{1'}) \cup U_k^{2'}, (U_k^2 - U_k^{2'}) \cup U_k^{1'})$ 相比划分 P_k 割切更小。

该调度系统采用多目标的离散群智能思想,每个个体所处 $|U|$ 维空间位置对应于一个划分方案,即个体在每个维度空间的位置,代表该维度空间对应结点所处划分的结点子集;伴随着优化阶段的细化过程,每个个体解决方案代表的划分投影到下一水平层的粗化超图上,且个体的自由度随着空间维度 $|U|$ 的增加而增大;个体之间利用它们聚集协同表现出的智能行为,有效地搜索多约束条件下、多目标组合的 Pareto 有效解,使非劣迁移优化解向着 Pareto-最优面逼近,增强迁移优化算法的多目标搜索能力。

6 仿真对比实验与结果分析

6.1 对比实验模型

本文选用墨尔本大学网络实验室的 Cloudsim 云计算仿真平台^[16],具体的仿真对比实验步骤如下:

步骤1 初始化 CloudSim 包:对 CloudSim 的用户数量、日期和跟踪标志等参数进行初始化。

步骤2 创建数据中心:创建主机列表且设置每个主机的 ID、内存等配置参数,创建 PE 列表且设置每个 PE 的 ID、MIPs 等配置参数,创建数据中心对象等。

步骤3 创建数据中心代理:通过对 DataCenterBroker 代理类扩展,在其内部分别实现 Min-Min 算法、Max-Min 算法、基于多水平方法和赋权有向超图的依赖任务调度算法。

步骤4 创建虚拟机:对虚拟机的 ID、MIPS、CPU 数量等参数进行设置并提交任务代理。

步骤5 创建云任务:创建指定参数的云任务,设定

任务的 ID 并提交任务代理。

步骤6 调用任务调度算法:借助自定义的任务调度策略,分配任务并绑定到虚拟机。

步骤7 启动仿真。

步骤8 收集仿真实验数据。

为了公平地和 Min-Min 和 Max-Min 元任务调度算法进行实验对比,保证实验结果的可重复性和公正性,本文假设忽略因任务执行先后依赖关系导致非关键路径上的时间余量,且对比实验中的参数设置如下:固定选用的随机数种子为 1 000;实验中设定虚拟机数量为 5,虚拟机指令执行速度在 [100, 300] MIPS 随机产生;五个对比实验分别设定任务数为 500、1 000、1 500、2 000 和 2 500,每个任务指令长度在 [1 000, 5 000] MI 随机产生;随机生成有向无环超图来描述任务之间的执行先后依赖关系,且每个任务存在依赖关系的概率为 50%,对应的通信时间在 [1, 5] s 随机产生。通过分析仿真实验数据,对比各算法的性能,验证本文算法在任务执行时间减少、资源负载相对均衡等方面的优点。

6.2 仿真对比实验的结果与分析

定义4 定义 n 个不同任务调度到 m 个不同虚拟机上的平均负载,等于 n 个任务的总指令长度除以 m 个虚拟机指令执行速度累加和、两两任务之间通信时间累加和除以虚拟机个数 m 两者之和,即总任务最优完成时间

$$\overline{VT} = \frac{\sum_{i=1}^n MI_i}{\sum_{j=1}^m MIPS_j + \sum_{i=1}^n \sum_{j=1}^n d_{ij}} / m。$$

表1为不同数量任务下算法的最迟完成时间对比表,表明 Cloudsim 中内嵌的顺序调度算法是一种简单的轮询调度,算法简单,因此所耗费时间远长于 Min-Min 算法、Max-Min 算法和本文算法;本文算法相比优先选取时间较长任务执行的 Max-Min 算法和优先选取时间较短任务执行 Min-Min 算法,所需完成时间最短并接近于任务最优完成时间 \overline{VT} 。

表1 不同数量任务下算法的最迟完成时间对比表_s

任务数	顺序调度	Min-Min	Max-Min	本文算法	\overline{VT}
500	3 140.85	1 606.25	1 595.79	1 593.99	1 593.59
1 000	6 221.23	3 173.03	3 166.87	3 165.22	3 164.78
1 500	9 319.95	4 754.39	4 746.24	4 745.48	4 744.89
2 000	12 408.25	6 332.76	6 321.93	6 320.62	6 320.07
2 500	15 478.90	7 901.58	7 888.73	7 886.48	7 886.31

定义5 对于 n 个不同任务到 m 个不同虚拟机的某分配方案 X ,定义该分配方案 X 的资源负载平衡因子 LB_X 为不同虚拟机任务完成时间的方差值,即 $LB_X =$

$$\sqrt{\frac{\sum_{j=1}^m (VT_j - \overline{VT})^2}{m}}。$$

表2为不同数量任务的算法资源负载平衡因子对比表,表明 Cloudsim 顺序调度算法由于是轮询调度,没

有考虑计算资源的负载均衡,因此其资源负载平衡最差;Max-Min 算法的资源负载平衡能力略好于的 Min-Min 算法,但都劣于本文算法的资源负载平衡能力。

表2 不同数量任务的算法资源负载平衡因子对比表

任务数	顺序调度	Min-Min	Max-Min	本文算法
500	596.66	12.87	1.74	0.121
1 000	1 190.42	12.44	1.41	0.013
1 500	1 787.39	12.68	1.46	0.030
2 000	2 382.77	13.08	1.83	0.033
2 500	2 974.37	13.07	2.09	0.005

7 结束语

云计算环境下面向数据密集型应用的任务调度问题大多为依赖任务的调度问题。例如,在天气预报、暴力破解密码等云计算应用中,计算任务之间往往存在执行先后依赖关系,要求一个计算任务必须接收到它的所有前驱任务消息后才能开始执行。相比有向无环图而言,赋权有向超图为依赖任务的调度提供了更为精确的模型。因此,本文借助赋权有向超图理论融合云计算任务调度特征,研究依赖任务的调度优化问题,提高任务调度的准确性和执行效率。

本文阐述了当前国内外主流的云计算任务调度算法,其中包含了传统启发式调度算法与智能调度算法。本文给出了云计算环境下的依赖任务调度系统的形式化描述;借助赋权有向无环超图的多对多关系构造云计算环境下依赖任务调度问题的数学模型;将云计算依赖任务调度问题转换为赋权有向超图的优化划分问题,提出了基于多水平方法和赋权有向超图的依赖任务划分优化算法;设计并实现了基于多水平方法的云计算依赖任务调度原型系统;在 CloudSim 云计算仿真实验平台下,进行了 Min-Min 算法、Max-Min 算法和本文算法的对比实验,实验数据对比表明,本文算法在减少依赖任务执行时间的同时,优化了资源负载均衡性能。

参考文献:

[1] 刘鹏.云计算[M].2版.北京:电子工业出版社,2013.
[2] 史恒亮.云计算任务调度研究[D].南京:南京理工大学,2012.
[3] 刘少伟,孔令梅,任开军.云环境下优化科学 workflows 执行性

能的两阶段数据放置与任务调度策略[J].计算机学报,2011,34(11):2121-2130.
[4] Braun T D,Siegel H J,Beck N.A comparison of eleven static heuristics for mapping a class of independent tasks onto heterogenous distributed computing systems[J].Journal of Parallel and Distributed Computing,2001,61(1):810-837.
[5] Zheng Q,Veeravalli B,Tham C K.On the design of fault-tolerant strategies using primary-backup approach for computational grids with replication costs[J].IEEE Transactions on Computers,2009,58(3):380-393.
[6] 史少锋,刘宴兵.基于动态规划的云计算任务调度研究[J].重庆邮电大学学报:自然科学版,2012,24(6):687-692.
[7] 何丽,饶俊,赵富强.一种基于能耗优化的云计算系统任务调度方法[J].计算机工程与应用,2013,49(20):19-23.
[8] 李建锋,彭舰.云计算环境下基于改进遗传算法的任务调度算法[J].计算机应用,2011,31(1):184-187.
[9] 张陶,于炯,杨兴耀,等.基于改进粒子群算法的云计算任务调度算法[J].计算机工程与应用,2013,49(19):68-72.
[10] 张雨,李芳,周涛.云计算环境下基于遗传蚁群算法的任务调度研究[J].计算机工程与应用,2014,50(6):51-55.
[11] 董丽丽,黄贲,介军.云计算中基于差分进化算法的任务调度研究[J].计算机工程与应用,2014,50(5):90-95.
[12] Karypis G,Aggarwal R,Kumar V,et al.Multilevel hypergraph partitioning: applications in vlsi domain[C]//Proc 34th Design Automation Conference,1998:526-529.
[13] Karypis G,Kumar V.HMetis 1.5.3:a hypergraph partitioning package[R/OL].Department of Computer Science, University of Minnesota,1998.http://www.cs.umn.edu/Metis.
[14] 查英华,杨静丽.改进蚁群算法在云计算任务调度中的应用[J].计算机工程与设计,2013,34(5):1716-1816.
[15] 冷明,孙凌宇,边计年,等.一种时间复杂度为 $O(n)$ 的无向超图核值求解算法[J].小型微型计算机系统,2013,34(11):2568-2573.
[16] Reuters T.CloudSim:a framework for modeling and simulation of cloud computing infrastructures and services introduction[EB/OL].[2015-01-04].http://www.buyya.com/gridbus/cloudsim/.
[17] 朱宗斌,杜中军.基于改进 GA 的云计算任务调度算法[J].计算机工程与应用,2013,49(5):77-80.