

分布式大数据存储的复杂任务调度策略

孙云¹ 倪伟¹

SUN Yun NI Wei

摘要

为进一步提升系统性能与用户服务质量,首先分析了复杂任务的定义、特点,并阐述常见任务调度策略;其次,对复杂任务调度策略进行设计,涉及任务树逻辑设计、任务树调度器设计和任务控制块的设计与实现;再次,在此基础上,提出了结合遗传算法与模拟退火算法的调度策略;最后,通过实验验证了所提调度策略的有效性,可为分布式大数据存储中的复杂任务调度提供有益参考与解决思路。

关键词

分布式; 大数据存储; 复杂任务调度

doi: 10.3969/j.issn.1672-9528.2024.09.027

0 引言

大数据时代正面临着前所未有的挑战与机遇。随着信息技术的迅猛发展,多元的用户需求与海量的信息资源如潮水般涌现,这对存储系统提出了更高、更复杂的要求。传统的集中式存储系统在面对如此庞大的数据处理需求时显得力不从心,其扩展性、灵活性和响应速度均难以满足现代应用的需求。因此,分布式存储系统因其卓越的高可扩展性、高灵活性以及良好的容错性,逐渐成为解决大数据存储和管理问题的有效方案。

然而,分布式存储系统并非万能钥匙,其也面临着诸多挑战。如何更迅速、更准确地响应用户的短时、突发需求任务,如何高效地管理、调度和分配海量的信息资源,以确保系统的整体性能和稳定性,成为亟待解决的问题。在这种背景下,复杂任务调度策略应运而生。该策略通过将复杂的任务智能地分割成若干子任务,并根据各虚拟机节点的资源状况、负载情况以及处理能力,动态、智能地将子任务分配到最合适的节点上执行。这样一方面有助于显著降低任务的完成时间,提高系统的整体处理效率;另一方面,也能够更好地满足用户的多样化需求,显著提升用户的服务质量和体验。因此,对分布式大数据存储的复杂任务调度策略进行深入研究和探讨,具有重要的理论意义和实际应用价值。

1 复杂任务调度分析

1.1 复杂任务的定义与特点

复杂任务指的是那些无法由单一机器节点独立完成,需要多个子任务协同执行的任务。这类任务通常具备执行时间长、资源需求多样、逻辑复杂、规模庞大等显著特点。处理

复杂任务的关键在于将复杂任务适当分解成若干易于管理且独立的子任务。与此同时,分解过程不仅要求确保各子任务的独立性,以有效减少协调与通信的开销,还需要保证子任务之间的组合性与层次性,以方便大数据存储系统按照任务逻辑完成灵活调度。此外,为确保分布式大数据存储系统的整体性能,也需要注意子任务规模的均匀性,以避免个别任务因执行时间过长而造成性能瓶颈。

分布式环境下,子任务之间的依赖关系也不容忽视,其中,数据依赖与控制依赖是两类主要的依赖类型,它们用于定义任务执行的数据流动方向与先后顺序^[1]。在分布式大数据存储系统中,运用恰当的复杂任务调度策略对于提升资源利用率与系统性能具有重要作用。为有效提升大数据存储系统性能、资源利用率,精准管理依赖关系,本研究使用复杂任务树这一数据结构来抽象描述复杂任务,并利用任务树调度器完成智能调度,以更好地实现复杂任务的高效执行与资源的最优利用。此外,还引入遗传算法与模拟退火算法,进一步优化任务调度策略,以此显著提升存储系统的整体性能。

1.2 常见任务调度策略

在分布式大数据存储环境中,复杂任务可构建成任务树,其中,边表示任务之间的依赖关系,节点代表子任务。而任务树调度器则用来高效地分配与执行这些任务。常用的任务调度策略包括静态调度策略、动态调度策略以及启发式调度策略。静态调度策略在编译时明确任务执行顺序,特别适用于执行时间可预测且任务依赖关系明确的场景;动态调度策略能够根据大数据存储系统具体状态来灵活调整任务执行,具有较高的灵活性;启发式调度策略很好地结合了静态调度策略与动态调度策略的优点,不仅考虑任务间的依赖关系,还考虑实时系统状态,以此实现最佳的

1. 江苏省扬州技师学院 江苏扬州 225003

调度效果^[2]。此外，为实现高效调度，综合运用遗传算法与模拟退火算法，通过模拟自然选择与物理退火过程来寻找最佳的任务执行顺序，进而进一步提高分布式大数据存储系统性能与调度效率。

2 复杂任务调度策略设计

2.1 复杂任务树逻辑设计

为提高分布式大数据存储的效率与稳定性，本研究对复杂任务树逻辑进行设计，通过将逻辑分解复杂的任务，形成树状结构，以此更高效地完成任务调度。复杂任务树主要包括串行任务、并行任务、备选任务、多择一任务等。其中，串行任务要求各子任务按顺序执行，并且每个子任务成功执行是后续任务执行的前提；并行任务允许多个子任务同时执行，以提高处理速度，然而全部子任务都成功才能完成整个复杂任务；备选任务提供了执行策略的灵活性，至少有一个子任务成功，整个复杂任务即算完成；多择一任务则更为宽松，只要同时执行的子任务中有一个任务成功，任务就算完成^[3]。

上述四种任务类型构成了复杂任务树的基础，便于分布式大数据存储系统可按照实际需求来灵活组合与调整任务执行策略。例如，在大数据存储过程中，可将数据写入操作设计成并行任务，以有效提高写入速度；将数据验证与备份设计成备选任务，以保障数据的完整性与安全性。此外，本文设计的复杂任务树逻辑具备良好的可扩展性，伴随业务场景的不断变化，能够轻松地调整任务之间的关系或是增加新的任务类型，其灵活性使得分布式大数据存储系统可以更好地应对持续变化的业务需求。

在分布式大数据存储中，将复杂任务视为一棵树的构造过程。该过程从根任务开始，将其拆解为若干子任务，同时分析子任务之间的控制或数据依赖关系，明确它们是串行执行、并行处理，还是存在备选或多选一的关系。如果子任务仍显复杂，会继续递归分解，直至每个任务都简化到可单独执行的程度，此时复杂任务树即构造完成。此树状结构包含三种节点：叶子节点代表可单独执行的简单任务；非叶节点则描绘复杂任务及其子任务间的关系，不直接执行任务；根节点标志着整个任务的起点，它可以是简单任务，也可以是包含多个子任务的复杂任务。通过这种方式，可清晰映射并处理极为复杂的任务，优化分布式大数据存储环境中的任务调度。

2.2 任务树调度器设计

任务树调度器用于处理复杂任务执行过程中的各种复杂情况，如任务的串行与并行执行、任务类型嵌套、数据依赖及可扩展性需求等。任务树调度器设计过程中面临的主要挑

战包括多层次复杂结构、动态生成的任务树、处理串行与并行任务的混合执行、未来可能出现的任务类型扩展^[4]。为处理这些问题，本研究设计的任务树调度器采用一种高效且灵活的策略。复杂任务调度流程示意图如图1所示。

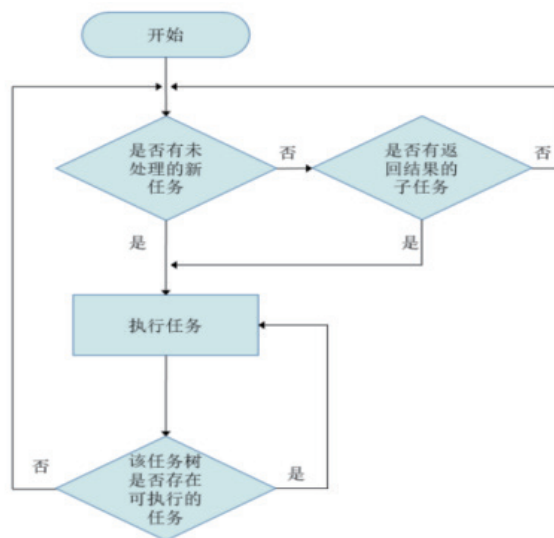


图1 复杂任务调度流程示意图

如图1所示，本研究采用了一种高效的任务处理机制。系统启动后，首先会进入一个主循环，该循环会不断地从两个队列中选取可执行的任务进行处理。这两个队列分别是新任务队列与已完成子任务的队列。值得注意的是，为优化系统响应时间，新任务队列在处理顺序上享有更高的优先级。每当一个任务执行完成后，系统会进入任务树调度器的内部循环。在此环节，任务树调度器会以刚完成的任务为输入，在其所属的任务树中寻找下一个可执行的任务。该过程会持续进行，直到任务树中没有可执行的任务为止。任务树调度器的输入来源涉及尚未处理的新任务、子任务返回的结果以及刚刚执行完成的任务。

复杂任务调度流程为，首先任务树调度器从返回待处理任务队列与新任务请求队列中获取可执行任务，可能是已完成任务的子任务，也可能是未经处理的新任务。任务树调度器按照任务的类型与状态，自动选择下一个可执行的任务。对于备选任务与串行任务，任务树调度器将依次执行子任务，确保前一个任务成功完成后才开始执行当前任务；而对于多选一任务与并行任务，任务树调度器会同时启动多个子任务，同时等待其中一个或若干任务完成^[5]。此外，任务树调度器充分考虑子任务之间的数据依赖关系，可以动态地构建任务树。如果存在某个任务的结果是后续任务的输入，那么此时调度器将等待该任务完成后，再执行依赖它的任务。该设计一方面显著提高了分布式大数据存储系统的可扩展性与灵活性，另一方面保障了并行处理的效率与任务的顺序执行。通过设计并优化任务树调度器，分布式大数据存储系统能够更

好地适应复杂的任务调度需求，进一步提升系统的稳定性与整体性能。

2.3 任务控制块设计与实现

任务控制块（TCB）作为标注与管理任务属性的重要数据结构，对于推进任务执行与维护任务状态至关重要。TCB中涵盖很多关键成员变量，如子任务状态与数量、是否处于并行分支、是否根任务、任务状态、类型等这些变量一同构成了复杂任务调度的决策基础。利用有限状态机理论，TCB将任务划分成未初始化、等待、成功与错误四种状态，实现了精细管理任务状态^[6]。任务控制块核心成员变量表如表1所示。

表1 任务控制块核心成员变量表

成员变量名	类型	功能描述
taskID	uint64_t	唯一标识本任务的ID号
rootTaskID	uint64_t	根任务的唯一ID号
state	enum	标注任务当前的状态
type	enum	定义任务的类型
is_root_task	bool	表明该任务是否作为整个任务的起始点
child_task_num	uint64_t	统计该任务直接子任务的总数量
init_child_task_num	uint64_t	记录当前处于INIT状态的子任务数量
complete_child_task_num	uint64_t	统计当前处于SUCCESS状态的子任务数量
error_child_task_num	uint64_t	统计当前处于ERROR状态的子任务数量
is_parallel_branch	bool	指示该任务是否处于某个并行任务分支之中
return_value	自定义类型	存储任务执行后的返回值，用于后续处理或传递给其他任务
argument_list	自定义类型	存储执行任务所需的参数列表

在实现方面，本研究采用组合模式，使得组合任务与单个任务在接口上保持一致，大幅简化了应用层代码，并显著提高了分布式大数据存储系统的灵活性与可扩展性。其中，TaskBase类作为基类，用于定义任务的状态转换逻辑与基本操作；NoLeafTask类是复杂任务，管理子任务的移除与添加，同时定义复杂任务的失败与成功条件；LeafTask类是简单任务，主要定义任务的具体行为^[7]。如此，系统应用层能够一致地处理复杂任务与简单任务，而无需区分任务类型，进而有效简化了代码逻辑。与此同时，新定义的任务类型能够轻松地与原有代码和结构兼容，显著增强了系统的可扩展性。

本研究所采用的任务树调度方法通过TaskBase类的静态函数NextAvailTask实现，以当前任务为起点，快速定位到下一个可执行的任务。该函数针对不同类型的任务，如串

行、并行、备选等，采取了差异化的处理策略，确保了任务调度的准确性与高效性。同时，该策略还具备良好的扩展性，可轻松应对新增任务类型。在算法设计中，特别考虑了并行任务的管理，通过向上回溯机制，确保了所有并行任务的成功执行。该调度策略不仅时间复杂度低，大部分情况下能达到 $O(1)$ ，而且在管理大量并行任务时，仍能保持良好的性能，以此有效支撑了分布式大数据存储系统中复杂任务的高效调度。

3 基于遗传算法与模拟退火算法的调度策略

3.1 遗传算法在任务调度中的应用

复杂任务调度策略的关键是将 N 个用户任务合理分配至 M 个存储节点上，以实现复杂任务的高效处理。存储节点资源与用户任务集合之间存在着映射关系，并使用矩阵来表示这种映射，以清晰直观地反映出节点与任务之间的分配情况。遗传算法在分布式大数据存储的任务调度中借助编码个体来表示不同的任务分配方案。完成编码后，通过解码来获取最优的任务调度方案，计算出各存储节点的任务执行时间，据此能进一步得到用户任务的平均完成时间，用以客观评价调度方案的优劣，相关公式为：

$$\text{TotalTime}(p_i) = \sum_{j=1}^N \text{Time}(i, j), i \in [1, M] \quad (1)$$

式中： M 是指可供使用的虚拟机资源节点； N 表示执行的用户子任务的总数目； $\text{Time}(i, j)$ 表示虚拟机 V_i 执行第 j 个子任务所用的时间。为缩短任务完成时间，本研究将子任务的最短完成时间当作个体的适应度函数，以在迭代过程中持续优化任务分配方案，从而寻找最短的任务完成时间。遗传操作主要包括选择、交叉与变异，在复杂任务调度中，这些操作被用于生成新的任务分配方案。通过科学设定变异概率与交叉概率，遗传算法可以在保证优良基因的同时，增加新的变异因素，以扩大找到全局最优解的可能性。通过持续优化任务分配方案，使得复杂任务调度策略可以显著提高任务完成效率，从而提升整体服务质量。

3.2 模拟退火算法在复杂任务调度中的优化

由于任务调度的复杂性会伴随数据规模的扩大而增加，为有效提升任务调度的效率，本研究还引入模拟退火算法来优化任务调度策略。该算法与遗传算法有效融合形成GASA算法，能够更好地适应分布式大数据存储背景下短时需求任务调度^[8]。模拟退火算法具备独特的寻优方式，在避免陷入局部最优解方面具备明显优势，它通过模拟物理退火过程，可在解空间中完成广泛而深入的搜索，据此找到全局最优解。在分布式大数据存储的复杂任务调度中，模拟退火算法主要被用于在遗传算法寻找到的最优解附近完成精细搜索，以寻

求进一步的优化。

GASA 算法有效融合了遗传算法的全局搜索能力与模拟退火算法的局部寻优能力。具体而言,在每一次迭代中,遗传算法先生成一组候选解,结合使用模拟退火算法来局部优化这些解,通过持续迭代与优化,逐步逼近最佳的任务调度方案^[9]。在设计与实施过程中,本研究定义领域函数与冷却进度表来有效控制算法的温度下降速度与搜索过程。其中,领域函数负责生成新的候选解,而冷却进度表负责搜索过程中的“温度”变化,以此影响接收新解的概率。此外,还运用 Metropolis 准则来决定是否使用新的任务调度方案,通过当前“温度”与新解和当前解的差异来计算接受新解的概率,最大限度地避免了算法陷入局部最优解,为分布式大数据存储环境中的复杂任务调度提供了一种实用可行的解决方案^[10]。

4 实验与结果分析

为验证本文设计的复杂任务调度策略的有效性,本研究设计了一系列涉及分布式大数据处理的实验。实验注重模拟真实的大数据应用场景,如多维数据分析、数据整合以及模式识别等复杂任务。本研究选用了 6 台配置相同的阿里云 ECS 服务器作为实验环境,服务器之间使用千兆网络连接。其中,1 台用于管理元数据,4 台作为数据服务器,剩余 1 台则负责模拟用户端的复杂任务请求。服务器的配置包括 40 GB 硬盘、1024 MB DDR3 内存以及 1 核心 Intel Xeon CPU,以保证实验环境的一致性。在复杂任务处理方面,本文设计的复杂任务调度策略展现出了出色的性能。无论是多维数据分析、数据整合还是模式识别,该策略都能有效地优化资源分配,提高任务执行效率。实验数据表明,伴随数据服务器的增加,任务处理能力呈现出线性增长的趋势,证明了该策略具有良好的横向扩展能力。

5 结语

本文从复杂任务的定义与特点出发,深入探讨了分布式大数据存储环境下复杂任务调度的挑战与需求。针对这些挑战,设计了一种基于任务树的调度策略,该策略能够将复杂的任务有效地分解为多个子任务,并通过合理的资源分配和任务调度,实现任务的高效执行。为了进一步提升调度策略的性能,我们综合运用了遗传算法与模拟退火算法进行优化与实现,这两种算法的结合使得调度策略能够更好地适应动态变化的存储环境,并在任务执行效率和资源利用率方面取得了显著的提升。

实验结果表明,本文所提出的复杂任务调度策略具有可行性和有效性,能够在分布式大数据存储系统中实现任务的快速响应和高效执行,对于提升用户满意度和系统性

能具有重要现实意义。同时,我们也认识到,随着大数据应用的不断深入和用户需求的日益多样化,复杂任务调度策略将面临更多的挑战与机遇。未来,我们将继续深入探索更多先进的调度技术与算法,如深度学习、强化学习等,以期在分布式大数据存储系统的创新与发展中提供更多、更有效的技术支持,推动大数据技术的不断进步与应用拓展。

参考文献:

- [1] 李杰,陈润丰,彭婷.面向网络化无人机集群的计算换通信分布式任务调度方法[J].国防科技大学学报,2023,45(4):45-54.
- [2] 包圣瑞,张在房.面向无人机物流的分布式任务调度[J].计量与测试技术,2024,51(2):87-91.
- [3] 翁锦阳,朱铁兵,柏志安.基于密度划分的分布式数据容错存储算法研究[J].吉林大学学报(信息科学版),2024,42(1):67-73.
- [4] 曾月阳,戎阳枫,王佳佳,等.分布式架构下的电力作业定时任务调度优化方法[J].信息技术,2023,47(9):148-153+159.
- [5] 赵小凡,杜舒明,刘超.基于电网分布式存储构建业务分析模型的探索和研究[J].自动化技术与应用,2024,43(3):124-127.
- [6] 刘深,王利朋,刘光享.一种可信的分布式异构存储系统[J].电脑编程技巧与维护,2024(2):95-98+172.
- [7] 赵军富,杜海渊,靳永胜,等.分布式 3D 打印服务的实时多任务调度研究[J].制造技术与机床,2024(4):188-195.
- [8] 李悦艺,张丰,杜震洪,等.基于 LSM-OCtree 的时空流分布式调度和存储方案[J].浙江大学学报(理学版),2023,50(2):204-212.
- [9] 张宇.基于云计算架构的分布式数据存储系统设计[J].信息与电脑(理论版),2023,35(16):184-186.
- [10] 洪俊望.面向大数据存储系统的分布式复杂任务树调度器的研究[D].广州:华南理工大学,2016.

【作者简介】

孙云(1987—),女,江苏扬州人,本科,讲师,研究方向:计算机应用技术、计算机网络。

倪伟(1989—),男,江苏扬州人,本科,一级实习指导教师,研究方向:计算机软硬件、计算机网络、物联网、云计算。

(收稿日期:2024-06-04)