

求解任务可拆分多项目协同调度问题的启发式算法

王 磊¹, 聂兰顺², 战德臣^{2†}, 王弼陡¹, 罗刚银¹

(1. 中国科学院 苏州生物医学工程技术研究所, 江苏 苏州 215163;

2. 哈尔滨工业大学 计算机科学与技术学院, 哈尔滨 150001)

摘 要: 生产项目计划与调度过程中任务可以被拆分为更小粒度的子任务分批次执行, 实现缩短项目总工期的优化目标. 针对抢占式任务可拆分多项目调度问题, 从协同优化角度探讨任务拆分与重组方式, 提出一个长工期任务优先拆分、长工期项目优先拆分和高资源利用率项目优先拆分 3 种任务拆分优先级判断规则, 设计一种求解任务可拆分多项目协同调度问题的启发式算法. 最后通过数值实例和仿真分析验证了所提出方法在多项目调度总工期的优化效果和求解效率.

关键词: 多项目调度; 协同优化; 任务可拆分; 抢占式; 启发式规则

中图分类号: TP391

文献标志码: A

Heuristic algorithm for solving multi-project collaborative scheduling problem with activity splitting

WANG Lei¹, NIE Lan-shun², ZHAN De-chen^{2†}, WANG Bi-dou¹, LUO Gang-yin¹

(1. Suzhou Institute of Biomedical Engineering and Technology, Chinese Academy of Sciences, Suzhou 215163, China;

2. School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China)

Abstract: The activity in project can be split into multiple sub-activities in batches of smaller granularity execution in the real production project planning and scheduling environment to shorten the total projects makespan. For the preemptive multi-project scheduling problem with activity splitting, the patterns of activity splitting and recombination are discussed from the perspective of collaborative optimization, three heuristic rules for the activity splitting judgment are proposed, such as long activity duration(LAD), long project makespan(LPM) and high resource rate(HRR), and a heuristic algorithm for solving multi-project collaborative scheduling problem with activity splitting is designed. Simulation results show that the method can shorten the total makespan of the multi-project scheduling problem by activity splitting.

Keywords: multi-project scheduling; collaborative optimization; activity splitting; preemptive; heuristic rule

0 引 言

传统的资源受限项目调度问题(RCPSP)一般将任务视为不可拆分的,即任务只能被一次执行,在执行过程中任务不可中断、资源不能被抢占^[1].然而,在现实的生产项目计划与调度过程中,任务往往可以被拆分为更小粒度的多个子任务分批执行,多个并行项目之间的任务通过拆分与重组的方式可以实现分专业、分批次执行,减小不同类型任务之间切换的准备时间,从而提高生产效率,实现缩短多项目总工期的优化目标.例如船舶制造企业中,船舶建造任务具有面向资源的动态可拆分性,各项工作在船舶生产设计阶段可视情况按阶段、产品、资源、组织、类别、工种、

场地灵活地进行分解,可依据资源的能力或可用性将任务动态分解或组成不同粒度、不同层次的任务^[2].

任务可拆分资源受限项目调度问题是一类扩展的多模式资源受限项目调度问题,属于NP-hard问题^[3].项目一般包含多项可拆分的任务,并且每个可拆分任务的拆分模式不唯一,多个并行项目的任务拆分方式将形成组合爆炸.一方面,采用任务拆分和重组方式在一定程度上能够达到缩短项目工期等优化目标;另一方面,该优化问题的搜索空间过于庞大,导致难以找到适合的精确解法,而近似解法的求解时间往往较长. Buddhakulsomsiri 等^[4]研究了资源紧缺情况下通过项目执行过程中拆分任务的方式解决资源不

收稿日期: 2016-04-08; 修回日期: 2016-09-12.

基金项目: 国家自然科学基金项目(61273038); 江苏省政策引导类计划(产学研合作)——前瞻性联合研究项目(BY2016049-01).

作者简介: 王磊(1982—),男,副研究员,博士,从事智能算法、项目计划与控制方法等研究; 聂兰顺(1979—),男,副教授,博士,从事协同资源管理、软件重构与复用等研究.

†通讯作者. E-mail: dechen@hit.edu.cn

足的难题,提出了分支定界方法求解,并根据对比实验结果验证了可以通过适当地拆分任务实现缩短项目工期的优化目标. 进一步, Buddhakulsomsiri 等^[5]又对求解方法进行深入探讨,提出了一种基于优先级的启发式算法,引入移动资源的概念判断项目执行过程中何时进行任务拆分更加合理. 实验结果表明,该方法对于资源更加紧缺、项目规模更大的 MRCPSP 具有更好的求解效果. 雒兴刚等^[6]针对任务可拆分的项目调度问题建立了项目工期最短的数学模型,并提出一种结合邻域搜索的混合遗传算法,通过算例验证了算法的有效性,同时对比分析了任务不可拆分和可拆分两种情况下的寻优效果. 邓林义等^[7]提出了一种求解任务可拆分项目调度问题的带有局部搜索的粒子群算法 LPSO,采用基于任务排列的粒子表示方法将遗传算法中的定位交叉引入粒子的更新过程中,并采用局部搜索技术对更新后的粒子进行改进,实验结果表明算法具有较快的收敛速度.

以上对于任务可拆分项目调度问题的研究没有对任务中断方式(即抢占式和非抢占式)进行区分和探讨. Cheng 等^[8]对该特性进行了深入探讨,明确区分了任务不可拆分 RCPSP、非抢占式任务可拆分 RCPSP 和抢占式 RCPSP 三类问题,对比提出了三类问题的形式化数学模型,并对非抢占式任务可拆分多模式 RCPSP 的求解方法进行了研究. Peteghem 等^[9]提出了一种求解抢占式多模式 RCPSP 的双种群遗传算法,采用两个独立种群并扩展顺序调度生成策略的方式提高遗传算法的收敛速度,仿真结果表明,采用抢占式任务拆分方式可以有效缩短项目工期. Shou 等^[10]针对抢占式 RCPSP 提出了一种杂交粒子群优化算法,通过引入顶点交叉算子提出了粒子更新机制,实验结果验证了该方法能够有效缩短抢占式 RCPSP 的项目工期.

本文针对抢占式任务可拆分多项目调度问题开展工作,从协同优化角度探讨如何通过任务拆分和重组的方式实现并行多项目总工期最短的优化目标. 针对项目内任务之间、多项目之间的任务拆分优先级判断难题,提出长工期任务优先拆分、长工期项目优先拆分和高资源利用率项目优先拆分 3 种启发式规则,给出求解任务可拆分多项目协同调度问题的启发式算法,并通过对比实验验证了所提出方法的有效性和求解效率.

1 任务可拆分多项目协同调度问题

经典资源受限项目调度问题和多模式项目调度问题都假设任务在执行过程中不可中断,但在生产实

际中,某些任务可进一步拆分成更小粒度的子任务分批完成. 对于任务的拆分可分为主动和被动两种.

主动拆分一般发生在项目计划和调度阶段,由决策者根据项目任务和资源情况对某些任务进行拆分. 例如,船舶建造过程的分段制造任务,由于加工机械、吊装设备、船台资源、人力资源等因素影响,对于分段大小的划分具有一定的调整范围,计划决策者可以将较大的分段(总段)划分为更小的分段,更精细地组织分段建造、系统预安装(舾装)以及分段吊装. 一般在能够缩短项目工期或者提高资源利用率等前提下,决策者会主动进行任务拆分.

被动拆分往往是由于资源、天气等外界因素影响导致的被迫将任务拆分为两个以上子任务分批执行. 例如,船舶分段在建造过程中需要完成大量的系统预安装(舾装)任务,这些任务的周期一般较长. 由于工作日历有固定的休息日和节假日,工人在此其间不能工作,从而导致已经开始的系统预安装任务需要暂停,等待下一个工作日继续执行. 再如,船舶建造一般是在大型水域旁的露天作业,受天气影响可能导致某些作业不得不暂停等待天气好转后继续,从而任务被迫拆分.

以上两种任务拆分方式必然导致项目工期和资源利用率的变化,如何更合理地利用任务可拆分特性缩短项目总工期、提高资源利用率是计划决策者面临的问题和挑战. 下文举例说明如何利用任务可拆分特性实现单项目和多项目调度的优化目标.

1.1 任务可拆分的项目调度

采用 AON(activity on node)网络图表示一个项目,节点表示其中的任务,节点间的有向弧表示任务之间的时序依赖关系. 假设项目执行需要一种可更新资源 R ,每个节点用 $j/r_j/d_j$ 表示,其中 j 表示任务, r_j 表示任务 j 执行所需要的资源量, d_j 表示任务 j 的工期.

图1为项目调度问题实例,项目 P_1 含有7项任务,开始任务0和结束任务6为两个虚任务. 如果任务不可拆分,则以项目工期最短为优化目标,项目 P_1 的调度结果如图1(b)所示,项目工期 $D = 8$. 如果任务可拆分,则通过将任务2拆分为子任务2a和2b分批次执行,最优调度结果如图1(c)所示,此时的项目工期由8缩短为7.

通过以上包含一个项目的 RCPSP 任务不可拆分和可拆分调度结果可以看出,采用任务主动拆分方式可以在某些环境下有效缩短项目工期,实现优化目标.

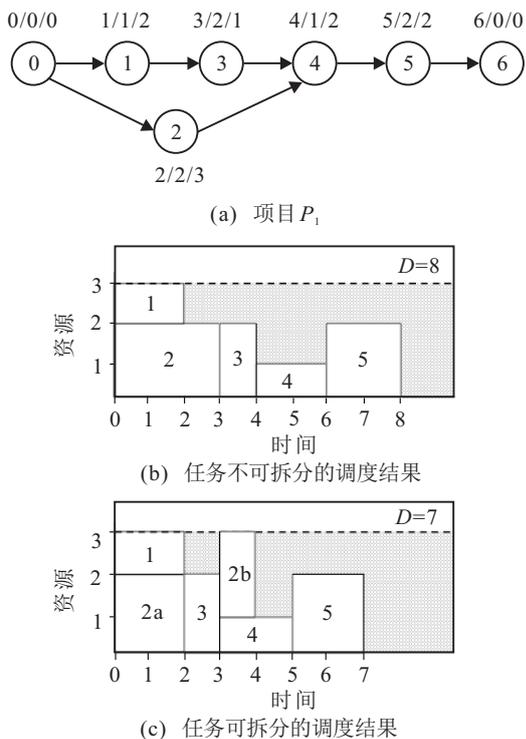


图1 项目调度问题实例

1.2 任务可拆分的多项目协同调度

同一时段并行的多个项目之间任务不存在时序依赖关系,而是通过共享资源发生联系. 在资源争用冲突情况下,决策者需要决定哪个项目、哪个任务优先占用资源得到执行,从而实现多个项目总工期最短或者其他优化目标. 单项目可以通过任务拆分实现缩短项目工期,在多项目环境下仍可采用这种方式.

如图2(a)所示,项目 P_2 在任务不可拆分情况下工期为 D^a . 假设采用任务拆分方式可以一定程度缩短项目工期,图2(b)项目工期调整为 D^b ,且有 $D^b(\#2) < D^a(\#2)$,这是一种理想情况下的调度结果. 在多项目并行环境下,由于争用共享资源可能

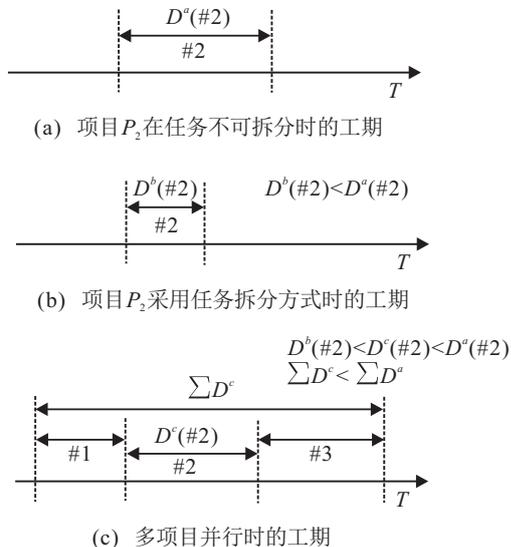


图2 任务可拆分多项目调度问题

导致其中的单一项目难以达到理想调度结果,实际调度结果是项目工期为 D^c , 一般有 $D^b < D^c < D^a$, $\sum D^c < \sum D^a$,如图2(c)所示,说明在资源争用平衡前提下,多项目全局调度最优难以保证单项目调度最优. 如何实现多项目全局调度最优情况下单项目调度也接近最优,可以考虑任务的可拆分特性.

每个可拆分任务的拆分模式不唯一,多个并行项目发生资源冲突时,多个并行项目的任务拆分方式将形成组合爆炸. 如何确定优先拆分哪个项目、哪个任务,如何设计更加合理有效的求解方法,是值得深入探讨的问题.

2 符号描述

在项目的AON网络图中,节点表示任务,节点间的有向弧表示任务之间的时序依赖关系,项目 i 包含两个虚拟任务,即虚拟起始任务0和虚拟终止任务 n_i . 任务 j_i 有 $|\text{Mode}_{j_i}|$ 种可能的拆分方式. 假设多项目调度问题中包含一种可更新资源 R ,在时段 t 的最大可用量为 U_{Rt} .

调度问题包含的多个项目组成集合 $Z = \{1, 2, \dots, z\}$,其中项目 $i (i \in Z)$ 的工期为 D_i ,多项目总工期为 $\sum_{i \in Z} D_i$. 项目 i 的所有任务组成集合 J_i , $J_i = \{0, 1, \dots, n_i\}$. 任务 j_i 的计划最早开始时间为 ES_{j_i} ,计划最晚完成时间为 LF_{j_i} ,实际开始时间为 s_{j_i} ,实际完成时间为 f_{j_i} ,前序和后序任务集合分别为 pred_{j_i} 和 succ_{j_i} .

任务 j_i 的拆分模式集合 Mode_{j_i} ,在拆分模式 m 下的工期为 $d_{j_i}^m$,所需资源量为 $r_{j_i}^m$,其中 $m \in \text{Mode}_{j_i}$, U_{Rt} 为可更新资源 R 在时段 t 的最大可用量.

任务可拆分多项目调度问题的主要决策变量包括: 1) 任务拆分模式 $y_{j_i}^m$,即如果任务 j_i 以模式 m 执行,则决策变量 $y_{j_i}^m = 1$,否则 $y_{j_i}^m = 0$,其中 $m \in \text{Mode}_{j_i}$; 2) 任务具体时段的拆分模式 $x_{j_i t}^m$,即如果任务 j_i 在时段 t 以模式 m 执行,则决策变量 $x_{j_i t}^m = 1$,否则 $x_{j_i t}^m = 0$,其中 $m \in \text{Mode}_{j_i}, t \in T$. 通过任务执行模式和执行时间的调整,实现缩短项目工期 D_i 以及项目总工期 $\sum_{i \in Z} D_i$ 的决策目标.

3 启发式算法

前面介绍了任务可拆分多项目协同调度问题,本节提出该问题的一种启发式求解方法,通过采用不同的启发式规则,寻求近似最优的任务拆分方案和多项目调度结果,以更加接近理想情况下的最优调度.

3.1 长工期任务优先拆分

在一个项目中,工期较长的任务因长时间占用资源,往往导致其他任务不能尽早开始,故该启发式规

则选择一个项目中的长工期任务进行拆分,称为长工期任务优先规则(LAD).图3给出了采用LAD规则进行单一项目任务拆分项目调度的实例.项目 P_2 包含7项任务,按照任务不可拆分进行任务时序安排,项目工期 $D_2^0 = 10$.分别拆分任务4和任务5并重新进行任务时序安排,能够得到不同的调度结果.

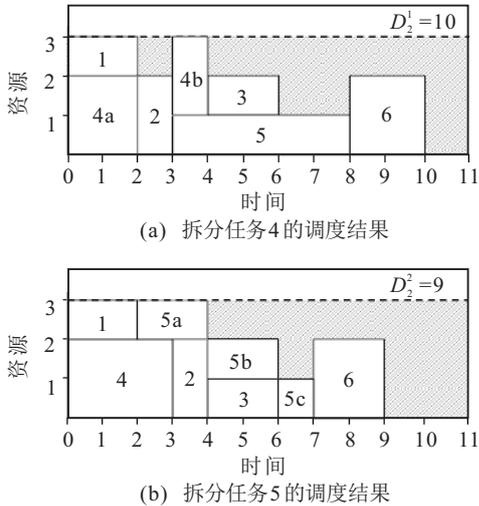


图3 LAD规则项目调度实例

图3(a)所示为拆分工期较短的任务4之后得到的调度结果,项目工期没有变化, $D_2^1 = 10$;图3(b)为拆分工期较长的任务5之后的调度结果,项目工期由原来的10缩短为9.

3.2 长工期项目优先拆分

并行的多个项目通过共享资源发生联系.当某时段的可用资源总量不足时,需要判断确定优先拆分哪个项目的任务.相对而言,项目工期较长的项目占用资源时间长,可以优先拆分该项目的任务,从而提高多项目之间的并行度.该启发式规则在多个项目中选择一个项目工期较长的项目进行任务拆分,称为长工期项目优先规则(LPM).多项目调度的实例中项

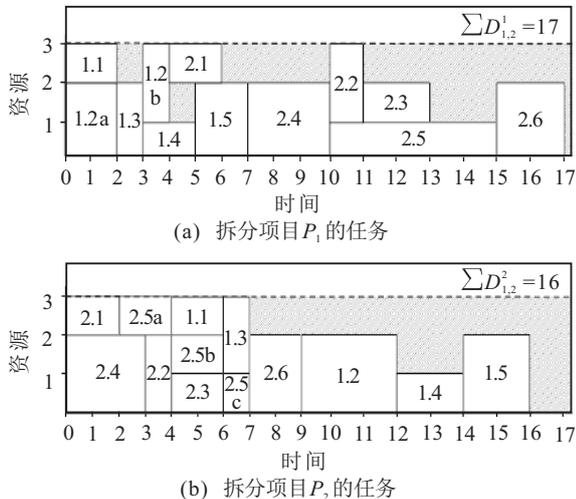


图4 LPM规则项目调度实例

目 P_1 的初始项目工期 $D_1^0 = 8$,项目 P_2 的初始项目工期 $D_2^0 = 10$,初始项目总工期 $\sum D_{1,2}^0 = 18$.

相对而言,项目 P_1 工期较短而 P_2 工期较长.分别拆分两个项目的任务,对任务重新安排时序,得到新的调度结果如图4所示.

图4(a)为拆分项目 P_1 的任务1.2之后的调度结果,项目总工期有所缩短, $\sum D_{1,2}^1 = 17$;图4(b)为拆分项目 P_2 的任务2.5之后的调度结果,项目总工期缩短为 $\sum D_{1,2}^2 = 16$.由此可见,拆分项目工期较长的项目任务能够更加有效地缩短项目总工期,得到更好的调度结果.

3.3 高资源利用率项目优先拆分

当多个项目争用共享资源发生冲突时,任务资源利用率更高的项目将更牢固地占有资源,因此,可以优先拆分该项目的任务从而缓解共享资源被个别项目单独占用的情况,从而提高多项目的资源共享程度.该启发式规则在多个项目中选择一个资源利用率更高的项目进行任务拆分,称为高资源利用率项目优先规则(HRR).在采用HRR规则进行任务拆分多项目调度的实例中,项目 P_2 的资源利用率 $RR_2 = 70\%$,项目 P_3 的资源利用率 $RR_3 = 55.6\%$,项目总工期 $\sum D_{2,3}^0 = 20$.

对比两种任务拆分方式:对资源利用率较高的项目 P_2 中的任务2.5进行拆分后,调整后的调度结果如图5(a)所示,项目总工期缩短为 $\sum D_{2,3}^1 = 16$;对资源利用率较低的项目 P_3 中的任务3.2进行拆分后,调整后的调度结果如图5(b)所示,项目总工期缩短为 $\sum D_{2,3}^2 = 17$.由此可见,并行多项目环境下,当发生共享资源争用冲突时,优先拆分资源利用率较高的项目任务可以得到更优的调度结果.

以上提出了针对单一项目任务拆分优先级判断

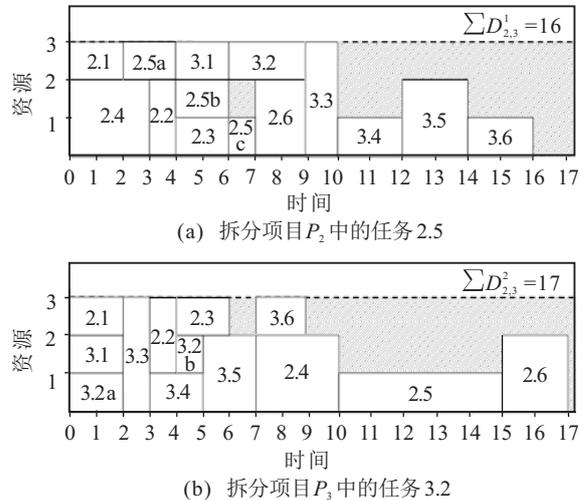


图5 HRR规则项目调度实例

的长工期任务优先规则,以及针对并行多项目环境下任务拆分优先级判断的长工期项目优先和高资源利用率优先规则,并通过实例验证启发式规则的正确性.在多项目协同调度问题求解过程中,一般需要将启发式规则组合运用,即首先判断多项目的“项目优先级”,再判断具体项目“任务优先级”,将该优先级判断规则应用于算法流程中,迭代式调整每个时段的任务拆分方案以及任务时序安排与资源分配.

3.4 算法整体流程

项目计划展望期为 T ,调度算法迭代式地为每个计划时段 t 安排资源分配方案,确定所有项目的任务开始和结束时间.当发生共享资源争用冲突时,需要判断确定任务拆分方案,调整项目计划与资源分配.算法主体部分与RCPSP求解方法相似,可以参考相关文献^[11].当资源不能满足需求时,需要通过任务拆分的方式将原有任务拆分为粒度更小的若干子任务,重新计算调度方案.

对于任务拆分优先级的判断,提出包括单一项目内的拆分规则和多项目之间的拆分规则如下:在单一项目中,长工期任务优先拆分;在多项目环境下,长工期项目优先拆分,高资源利用率项目优先拆分.

4 数值实例实验与分析

为了验证本文提出的求解任务可拆分多项目协同调度问题的启发式算法,设计生成了项目调度测试问题集.通过将启发式规则进行组合并应用于求解算法中,与现有的多项目调度解法在测试问题集上进行仿真实验与比较,从任务拆分对调度优化效果的影响、任务拆分对计算时间的影响两方面进行对比分析,从而验证所提出算法的正确性和有效性,探讨算法的优势与局限性.

4.1 测试问题集

项目调度问题一般采用Patterson测试集中的问题实例进行测试与比较^[12].由于多模式多项目调度问题以及任务可拆分特性,本文采用项目调度问题测试集生成工具ProGen并进行扩展,设计生成了8组测试问题.所生成的测试问题根据项目数(PN)、网络复杂度(NC)、资源强度(RS)进行区别^[12].项目数指多项目调度问题中所包含的项目数量,按照 $PN = 2$ 和 $PN = 5$ 生成测试问题.网络复杂度指包括虚任务在内的任务节点平均有向弧数量,按照 $NC = 1.4$ 和 $NC = 2.5$ 生成测试问题.资源强度用于描述资源的“稀缺程度”,取值范围 $[0, 1]$.当 $RS = 0$ 时表示最紧张的资源强度,资源水平刚好满足任务需要,当 $RS = 1$ 时表示最宽松的资源强度,所有任务都可以按照最早

开始时间安排.按照 $RS = 0.3$ 和 $RS = 0.7$ 生成测试问题.另外,假设项目中含有可拆分任务的比例为常数30%.计算过程采用Visual Studio 2012 C++编程实现,CPU为Intel Core 2,主频2.5 GHz,4 GB内存,操作系统为Windows 7.

4.2 仿真结果分析

按照上述参数设定,利用ProGen测试问题集生成工具生成8组测试问题,并分别采用本文所提出的启发式算法和分支定界法进行仿真运算.由于本文启发式规则有不同种组合方式,采用多项目根据长工期项目优先+单项目根据长工期任务优先(LPM+LAD)的方式,以及多项目根据高资源利用率项目优先+单项目根据长工期任务优先(HRR+LAD)两种方式构造启发式算法,分别从任务拆分对调度优化效果的影响、任务拆分对计算时间的影响两个角度对仿真结果进行分析.

4.2.1 任务拆分对调度优化效果的影响

前文举例说明了任务拆分在适当的情况下可以缩短项目工期.针对生成的8组测试算例,分别采用任务不拆分的方式以及LPM+LAD、HRR+LAD两种启发式算法进行任务拆分的多项目调度,计算项目总工期如表1所示.

表1 优化效果对比

编号	问题属性			项目总工期(天)		
	PN	NP	RS	不拆分	LPM+LAD	HRR+LAD
1	2	1.4	0.3	14	13	14
2	2	1.4	0.7	19	17	16
3	2	2.5	0.3	17	14	17
4	2	2.5	0.7	23	17	19
5	5	1.4	0.3	49	43	41
6	5	1.4	0.7	58	52	48
7	5	2.5	0.3	64	47	48
8	5	2.5	0.7	82	62	59

从表1所示结果可以看出:在项目数量较少、资源强度较低的情况下,任务不拆分与任务拆分两类调度方式的项目总工期区别不大;随着项目数量增加、网络复杂度以及资源强度增大,任务拆分方式对于项目总工期的缩短效果明显.可以认为,由于多项目环境的复杂程度增加,通过任务拆分,一定程度上可以提高多项目多任务的并行程度,更加高效地利用共享资源,最终达到缩短多项目总工期的优化目标.同时,HRR+LAD启发式算法相比于LPM+LAD算法在优化效果方面更优.

4.2.2 任务拆分对计算时间的影响

任务拆分重组将引起计算复杂性提高,导致计算时间的延长.为了测试本文所提出启发式算法在计算时间方面能否满足实际需要,将LPM+LAD、

HRR+LAD两种启发式算法与分支定界法在求解测试实例过程中的表现进行比较,所得结果如表2所示。

表2 计算时间对比

编号	计算时间			CPU时间/s		
	PN	NP	RS	LPM +LAD	HRR +LAD	分支定界法
1	2	1.4	0.3	0.352	0.426	1.107
2	2	1.4	0.7	0.473	0.518	1.228
3	2	2.5	0.3	1.264	1.302	2.737
4	2	2.5	0.7	3.085	4.920	6.375
5	5	1.4	0.3	6.085	9.804	23.085
6	5	1.4	0.7	9.707	13.599	31.014
7	5	2.5	0.3	12.653	18.984	47.483
8	5	2.5	0.7	15.075	21.375	66.842

结果显示,随着项目数量增加、网络复杂度和资源强度增大,分支定界算法的时间明显延长;而启发式算法虽然在延长比例上也有明显增大,但是绝对时间能够控制在可接受范围内。相对而言,HRR+LAD启发式算法比LPM+LAD算法在计算时间方面一般更长,这是由于HRR+LAD算法在计算资源利用率和资源冲突消解过程的耗时更长。

由仿真实验和结果的分析可以看出:从任务拆分对调度优化效果的影响、任务拆分对计算时间的影响两方面,本文提出的任务可拆分多项目协同调度启发式算法在多项目总工期优化方面有较为明显的效果;另一方面,启发式算法的计算时间相比于精确算法更短,有利于算法在并行多项目环境下的应用。

5 结 论

针对抢占式任务可拆分多项目调度问题,本文提出了基于项目和任务拆分优先级判断的启发式算法。任务拆分多项目调度问题的核心是判断项目与任务的拆分优先级,本文提出了长工期任务优先拆分、长工期项目优先拆分和高资源利用率项目优先拆分3种启发式规则。通过数值实例实验与分析,验证了该算法在多项目协同调度的项目总工期优化方面具有良好表现,通过将任务拆分为更小粒度的多个子任务,多个并行项目之间的任务通过拆分与重组的方式实现分批次执行,可以减小不同类型任务之间切换的准备时间,最终实现缩短多项目总工期的优化目标。在未来工作中,有必要解决算法收敛速度难题,提高启发式算法在生产项目计划与调度领域的应用性。

参考文献(References)

[1] Weglarz J, Jozefowska J, Mika M, et al. Project scheduling with finite or infinite number of activity processing modes — A survey[J]. *European J of Operational*

Research, 2011, 208(3): 177-205.

- [2] 战德臣, 喻小光, 聂兰顺, 等. 船舶建造分层生产计划体系[J]. *计算机集成制造系统*, 2012, 18(3): 497-506. (Zhan D C, Yu X G, Nie L S, et al. Shipbuilding layered production plan system[J]. *Computer Integrated Manufacturing Systems*, 2012, 18(3): 497-506.)
- [3] Hartmann S, Briskorn D. A survey of variants and extensions of the resource-constrained project scheduling problem[J]. *European J of Operational Research*, 2010, 207(1): 1-14.
- [4] Buddhakulsomsiri J, Kim D S. Properties of multi-mode resource-constrained project scheduling problems with resource vacations and activity splitting[J]. *European J of Operational Research*, 2006, 175(1): 279-295.
- [5] Buddhakulsomsiri J, Kim D S. Priority rule-based heuristic for multi-mode resource-constrained project scheduling problems with resource vacations and activity splitting[J]. *European J of Operational Research*, 2007, 178(2): 374-390.
- [6] 雒兴刚, 汪定伟, 唐加福. 任务可拆分项目调度问题[J]. *东北大学学报: 自然科学版*, 2006, 27(9): 961-964. (Luo X G, Wang D W, Tang J F. Project scheduling problem involving time-splittable tasks[J]. *J of Northeastern University: Natural Science*, 2006, 27(9): 961-964.)
- [7] 邓林义, 林焰. 粒子群算法求解任务可拆分项目调度问题[J]. *控制与决策*. 2008, 23(6): 681-688. (Deng L Y, Lin Y. Particle swarm optimization for resource-constrained project scheduling problems with activity splitting[J]. *Control and Decision*. 2008, 23(6): 681-688.)
- [8] Cheng J, Fowler J, Kempf K, et al. Multi-mode resource-constrained project scheduling problems with non-preemptive activity splitting[J]. *Computers & Operations Research*, 2015, 53: 275-287.
- [9] Peteghem V, Vanhoucke M. A genetic algorithm for the preemptive and non-preemptive multi-mode resource-constrained project scheduling problem[J]. *European J of Operational Research*, 2010, 201(2): 409-418.
- [10] Shou Y, Li Y, Lai C. Hybrid particle swarm optimization for preemptive resource-constrained project scheduling[J]. *Neurocomputing*, 2015, 148(19): 122-128.
- [11] Herroelen W, Demeulemeester E, De Reyck B. A note on the paper resource-constrained project scheduling: notation, classification, models and methods[J]. *European J of Operational Research*, 2001, 128(3): 679-688.
- [12] Patterson J H. Comparison of exact approaches for solving the multiple constrained resource project scheduling problem[J]. *Management Science*, 1984, 30(7): 854-867.

(责任编辑: 李君玲)