

活动重叠模式与资源约束下的项目调度优化^①

于静¹, 徐哲^{2*}, 谢芳²

(1. 天津理工大学管理学院, 天津 300383; 2. 北京航空航天大学经济管理学院, 北京 100191)

摘要: 采取活动重叠模式通常是加速研发的有效手段, 带有活动重叠的资源受限项目调度问题是经典资源受限项目调度问题的扩展. 首先, 深入分析了活动重叠对于项目调度的影响, 对活动重叠及其不确定进行详细描述与建模, 提出了活动重叠导致下游活动返工时间的二项分布概率模型; 其次, 构建了以最小化研发项目期望工期为目标的优化调度模型, 设计了基于串行进度生成机制的遗传算法对大规模问题进行优化求解; 最后, 基于 PSPLIB J60 问题库中 480 个算例分析了该算法的计算结果, 并考察了网络参数、资源参数和重叠参数变化时, 采用活动重叠模式对缩短项目工期的影响. 研究表明: 活动对资源的需求强度越小或资源稀缺程度越低, 可重叠活动对数量就会增加, 项目工期缩短得越明显; 网络复杂度的变化对缩短项目工期的影响不大; 项目中重叠活动对越多, 重叠导致的下游活动返工的概率越小, 项目工期缩短的越明显.

关键词: 项目调度; 遗传算法; 活动重叠; 资源; 返工

中图分类号: N945 **文献标识码:** A **文章编号:** 1007-9807(2017)09-0036-10

0 引言

复杂产品研发过程中采用的新技术、新工艺、新材料是伴随着研发需要而同期研究的, 通常面临技术难度大、创新点多等挑战, 并伴随着多种风险与不确定性, 最终可能导致研发工期延长^[1]. 而研发工期决定了企业对竞争对手和技术进步做出反应的速度, 尽可能地缩短项目工期是编制项目计划和执行项目过程中首要考虑的因素. 采用活动重叠(overlapping)执行模式是缩短项目工期的有效手段. 许多活动具有技术上的优先约束关系, 导致这些活动无法完全并行执行, 但是通过重叠执行可实现一定程度上的赶工, 即在具有优先约束关系的活动中, 下游活动在其上游活动完成之前便提前开始执行, 当上游活动完全完工后, 下游活动可能需要对自身的工作内容进行修正,

导致其执行时间延长, 即产生额外的活动返工时间. 通常情况下, 活动重叠带来的项目工期减少量是高于活动修正导致的额外延长量的. 因此, 虽然活动重叠带来的项目工期缩短是以返工成本增加为代价的^[2], 但活动重叠仍然是值得的. 经典的资源受限项目调度问题(resources constrained project scheduling problem, RCPSP)的研究多是在确定性环境和完全信息条件下, 通过优化某一调度目标(工期、成本、资源等), 获取满足优先关系与资源约束的基线进度计划^[3,4]. 经典的 RCPSP 已经被证明是强 NP-hard 问题^[5], 带有活动重叠的资源受限项目调度问题可以看作经典 RCPSP 扩展, 当任意活动之间不存在重叠时(即重叠时间为零), 带有活动重叠的 RCPSP 便成为经典的 RCPSP. 可见, 经典的 RCPSP 是带有活动重叠 RCPSP 的特例. 因此, 带有活动重叠的 RCPSP 也

① 收稿日期: 2014-10-10; 修订日期: 2016-02-14.

基金项目: 国家自然科学基金资助项目(71571005; 71271019); 教育部人文社会科学基金资助项目(16YJC630159).

通讯作者: 徐哲(1964—), 女, 江苏无锡人, 教授, 博士生导师. Email: zhexubuaa@163.com

是强 NP-hard 问题。

采用设计结构矩阵 (design structure matrix, DSM) 建立重叠模型是常用的方法, Cho 和 Eppinger^[6] 基于 DSM 仿真模型估计复杂产品开发项目的工期与成本, 考虑了资源受限并且对返工、迭代做出了更符合项目实际的假设; 李洪波和徐哲^[7] 建立了复杂产品开发流程仿真模型, 该模型同时考虑了活动随机重叠和资源冲突; 进一步, 李洪波等^[8] 将 DSM 调度策略嵌入仿真模型, 提出了基于 DSM 仿真与非支配排序遗传算法的混合优化算法, 求解复杂产品研发项目多目标流程优化的 Pareto 最优解; Maheswari 和 Varghese^[9] 利用 DSM 时间因子矩阵建立活动重叠量化模型, 进而估算项目工期; 白思俊和万小兵^[10] 考虑了活动重叠对项目工期的定量影响, 基于时间因子矩阵来计算项目工期; 杨青和黄建美^[11] 采用信息输出时间因子和信息输入时间因子描述顺序重叠和反馈重叠, 利用遗传算法求解项目最小工期; 张春生和严广乐^[12] 利用 DSM 描述活动重叠, 提出混合粒子群算法对项目进度进行优化。除了以 DSM 为工具, 利用优化模型求解重叠问题也是研究的热点。Gerk 和 Qassim^[13] 综合考虑活动赶工、重叠、替代 3 种项目加速工具, 提出了混合整数非线性模型确定 3 种加速工具的最优分配; Lin 等^[14-15] 指出, 重叠可以缩短项目工期但会带来返工与沟通成本, 由此提出分析模型, 对活动重叠导致的正面与负面影响进行权衡, 并给出最优的沟通策略和重叠水平; Wang 和 Lin^[16] 建立了重叠流程模型, 分析流程结构 (活动执行顺序、活动之间的重叠程度) 对项目工期的影响。

最近几年有学者开始研究基于活动重叠的项目调度问题。Berthaut 等^[17] 将重叠模式引入资源受限项目调度问题, 对活动重叠模式与活动执行模式进行分析, 建立了带有重叠模式的非线性 0-1 整数规划模型; 随后, Berthaut 等^[18] 又将返工成本和沟通成本引入重叠模式, 建立了带有重叠模式的时间/成本权衡问题, 并将该问题的非线性 0-1 整数规划模型转换为线性 0-1 整数规划模型; Greze 等^[19] 建立了带有活动重叠的 RCPSp 的模型, 提出了求解带有活动重叠的 RCPSp 的启发式算法, 分析了多种目标函数, 以项目进度加速与成本增加权衡为目标和以最大化收益为目标两者

相比, 后者可以更好地避免损失。Bozejko 等^[20] 在道路施工调度中, 利用重叠模式进行加速以实现成本最小的目标。Koyuncu 和 Erol^[21] 采用粒子群算法对带有重叠的研发项目调度问题进行求解。于静等^[22] 将活动重叠引入到 RCPSp 中, 将活动重叠分为自然重叠和强制重叠, 采用设计结构矩阵重叠对时间和资源的影响, 以最小化研发项目工期为目标, 求解带有活动重叠的资源受限项目调度问题优化模型。

通过对以上文献的梳理分析可以看出, 带有活动重叠的项目调度研究逐渐开始引起学者的关注, 但目前研究尚存在较大的局限性。首先, 对于活动重叠的描述和建模缺乏深入研究, 对活动重叠对于项目调度的影响分析研究不够。例如, 在重叠描述模型上, 大多数研究假设重叠时间与返工时间是简单的线性关系^[22], 但是, 在项目实际执行过程中, 活动重叠导致下游活动的返工时间可能是不确定的, 建立返工时间的概率分布模型十分必要; 其次, 引入活动重叠, 优化调度模型的约束条件不仅包含活动优先关系约束和资源约束, 还包含活动重叠关系约束, 以及重叠产生下游活动返工时间和返工资源的约束, 使得调度问题的复杂度进一步提高。近年来, 已有少数学者开始关注带有活动重叠的项目调度问题, 但仅研究了 0-1 整数规划模型和时间/成本权衡模型, 提出了一些效率较低的启发式算法^[17-18]。另外, 在实验设计上, 现有研究缺乏基于大量算例的结果分析和问题参数对算法求解效果的影响分析。

本文将主要研究带有活动重叠和资源限制的项目调度问题, 体现在以下 3 个方面: ①深入分析了活动重叠对于项目调度的影响, 对活动重叠及其不确定性进行了详细描述与建模, 提出了活动重叠导致下游活动返工时间的概率分布模型; ②建立带有活动重叠和资源限制的优化调度模型, 并设计了求解大规模问题的遗传算法; ③基于一定量的算例, 验证了模型和算法的有效性, 并分析了带有活动重叠的项目调度问题中, 网络参数、资源参数以及重叠参数对项目工期的影响。

1 问题描述

在复杂产品系统研发项目中, 活动的执行模

式通常可以分为串行、并行、重叠3种^[23]. 在串行执行模式中, 上游活动完全结束后, 才将信息传递给下游活动; 在并行执行模式中, 活动之间没有信息传递; 重叠模式是指上游活动在完成之前将信息传递给下游活动, 下游活动在接收信息之后立刻开始执行.

本文基于以下4个假设开展分析: ① 活动之间的信息传递具有单向性, 即上游活动可以将信息传递给下游活动, 但是下游活动不能传递信息到上游活动, 因此, 活动重叠只会导致下游活动返工, 对上游活动没有影响; ② 活动之间的信息传递具有瞬时性, 即信息传递过程中不占用时间和资源, 上游活动释放信息时, 下游活动立即开始; ③ 活动重叠时, 下游活动结束时间大于上游活动结束时间, 由此保证下游活动可以接收到上游活动释放的最终信息; ④ 假设两个活动重叠时, 下游活动的返工时间不会超过其与上游活动的重叠时间, 否则活动重叠执行不能起到缩短项目工期的效果.

1.1 两个活动重叠的描述

两个活动重叠如图1所示, 活动*i*和活动*j*重叠执行, d_i 和 d_j 分别表示活动*i*和活动*j*的工期; 两个活动的重叠时间用 o_{ij} 表示; 重叠导致的返工时间用 rw_{ij} 表示. 若上游活动传递的信息为最终信息, 则下游活动无需进行返工, 这种重叠称为自然重叠, 图1中返工时间 $rw_{ij} = 0$; 若上游活动传递的信息为初始信息, 则下游活动执行后期需要返工以修正初始信息带来的误差, 这种重叠称为强制重叠, 图1中返工时间 $rw_{ij} > 0$.

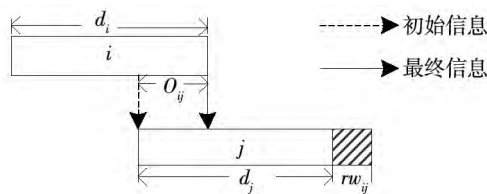


图1 两个活动重叠示意图

Fig. 1 Overlapping process of two activities

在项目调度中, 一般是将资源与时间做离散化处理的, 资源的需求与供给是针对某一时间段(小时、天、周等)而言的. 假设各类活动(正常、重叠、返工)在单位时间上消耗的资源量相同, 那么下游活动返工时间确定了也就可以确定返工增加的资源需求量. 在活动重叠描述模型中, 大多假定

重叠时间和返工时间是简单的线性关系. 但是, 在项目实际执行过程中, 由于活动重叠导致下游活动是否返工以及返工量的大小是依据上游活动在信息释放点释放给下游活动的信息量而变化的, 而上游活动在信息释放点能够释放多少信息是受到很多不确定因素影响的. 例如, 项目总体进展情况和上游活动的进展情况等. 因此, 活动重叠导致下游活动的返工时间可能是不确定的. 基于以上分析, 建立下游活动返工时间的概率分布模型十分必要. 调度问题中时间是离散化处理的, 假设活动的重叠时间为 Δ , 下游返工时间为 σ , 均取整数. 这样, 可将重叠时间划分为 Δ 个时间段, 考察每个重叠时间段, 相当于进行了 Δ 次独立试验, 在每次试验中, 两个活动重叠导致发生下游活动返工或不返工. 那么 Δ 次试验中发生返工的次数服从二项分布. 每次试验导致下游活动返工时间均为单位时间, 假设发生返工的概率为 β , 则返工时间 σ 服从二项分布 $C_{\Delta}^{\sigma} \beta^{\sigma} (1 - \beta)^{\Delta - \sigma}$, $\sigma = 0, 1, 2, \dots, \Delta$, 记为 $\sigma \sim B(\Delta, \beta)$. 例如, 假设重叠时间为5个时间单位, 下游活动发生返工的概率为0.4, 则返工时间的取值有以下6种可能: 0, 1, 2, 3, 4, 5. 其概率分布如图2所示.

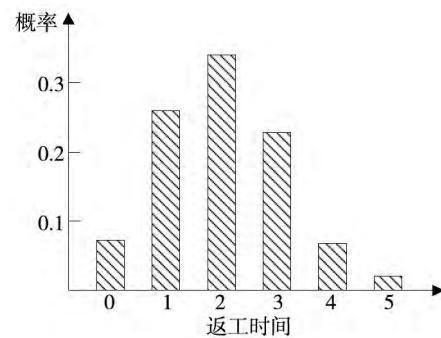


图2 $\Delta = 5, \beta = 0.4$ 时返工时间的概率分布

Fig. 2 Probability distribution of rework time with $\Delta = 5, \beta = 0.4$

在研发项目执行过程中, 活动的进展是通过里程碑进行控制的, 因此, 本文定义上游活动的初始信息是在里程碑节点释放给下游活动, 在不同的里程碑节点释放信息对应不同的重叠模式, 一个活动在执行过程中可能在多个里程碑节点与下游活动重叠执行. 因此, 本文假设重叠活动对之间存在多种可能的重叠模式, 该假设使得可重叠的活动对在调度过程中发生重叠的机率大大增加, 也就可以增加缩短项目工期的可能性. 活动重叠模式的个数取决于上游活动中里程碑的个数, 如

图 3 所示. 假设活动 i 和活动 j 重叠执行, 重叠模式集合可表示为 $M_{ij} = \{m_{ij1}, m_{ij2}, \dots, m_{ijM}\}$.

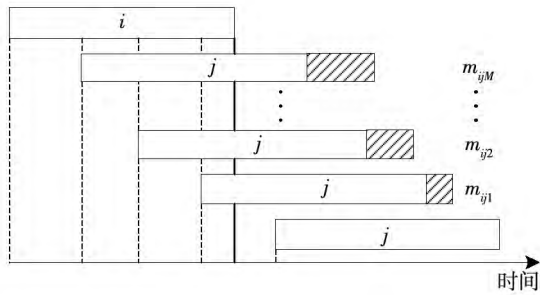


图 3 两个活动多种重叠模式示意图

Fig. 3 Multiple overlapping modes of two activities

1.2 多个活动重叠的描述

当项目执行过程中出现多个活动重叠时,即多个上游活动与一个下游活动重叠,多个活动重叠如图 4 所示. 由于活动重叠时,上游活动释放信息后,下游活动立即开始执行,为保证多个活动重叠的顺利进行,本文假设所有的上游活动在同一时刻释放初始信息(信息释放点一般是项目的里程碑时间节点). 不失一般性,假设每对活动重叠导致的下游活动的返工是相互独立的. 因此,下游活动总的返工时间是该活动所有重叠造成的返工时间之和. 图 4 中,假设活动 a_1, \dots, a_n 等多个活动都与活动 j 重叠,这些活动在同一时刻释放初始信息给活动 j ,活动 j 接收到信息后立刻开始执行. 活动 a_1 与 j 重叠产生的返工时间为 rw_{a_1j} ,活动 a_n 与活动 j 重叠产生的返工时间为 rw_{a_nj} ,那么活动 j 的总返工时间为 $(rw_{a_1j} + \dots + rw_{a_nj})$. 多个活动的重叠模式的描述与两个活动的重叠模式描述基本一致,不再赘述.

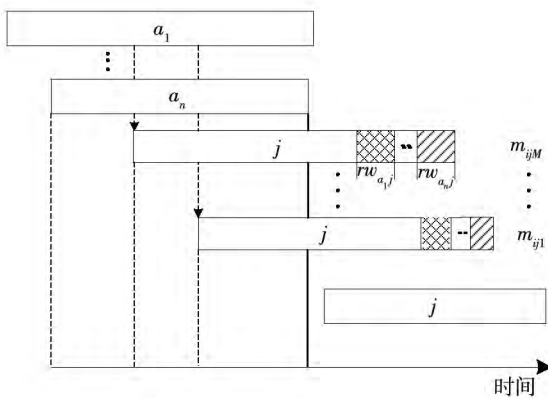


图 4 多个活动重叠模式示意图

Fig. 4 Overlapping process of multiple activities

2 带有活动重叠的优化调度模型

本文采用基于活动的研究方法,即项目表示为一个 AoN(activity-on-node) 网络图表示,1 个项目包含 $n+1$ 个活动,用集合 $V = \{0, 1, \dots, n\}$ 表示,其中,活动 0 和活动 n 分别是虚的开始和结束活动,不占用时间和资源;活动的工期表示为 $d = \{d_0, d_1, \dots, d_n\}$;项目实施需要 K 种可更新资源,第 k 种可更新资源在单位时间内的可用量为 R_k ;活动 i 在执行时单位时间内对资源 k 的需求量为 r_{ik} ;项目中满足优先关系的活动对记作 A ,重叠活动对记作 A_o ,非重叠活动对记作 A_n ,因此, $A = A_o \cup A_n$;假定时间和资源都是整数.

考虑活动之间存在的优先关系、重叠关系和资源约束,优化目标为最小化项目期望工期,建立的带有活动重叠的项目优化调度模型为

$$\min E(s_n) \tag{1}$$

$$\text{s. t. } s_j \geq s_i + d_i, \forall (i, j) \in A_n \tag{2}$$

$$s_j \geq s_i + d_i - o_{ijm} \\ \forall (i, j) \in A_o, m \in \{1, \dots, M\} \tag{3}$$

$$s_j + d_j > s_i + d_i \\ \forall (i, j) \in A_o, m \in \{1, \dots, M\} \tag{4}$$

$$\sum_{i \in S_t} r_{ik} \leq R_k, \forall k, t \tag{5}$$

式(1)中, s_n 为虚结束活动 n 的开始时间,即整个项目的工期,由于活动重叠导致的下游活动返工时间为随机变量,目标函数为最小化项目工期的期望值;式(2)表示非重叠活动对之间的时间约束关系;式(3)表示重叠活动对之间的时间约束关系,如果调度过程中两个活动重叠执行,那么下游活动的开始时间等于上游活动释放初始信息的时间,如果由于资源限制不能重叠执行,那么下游活动的开始时间不早于上游活动的完成时间;式(4)表示重叠活动对中上、下游活动结束时间关系约束,保证下游活动可以接收上游活动释放的最终信息;式(5)为资源约束条件,对于非重叠活动 $S_t = \{i \mid s_i \leq t < s_i + d_i\}$,对于重叠活动 $S_t = \{i \mid s_i \leq t < s_i + d_i + rw_i\}$.

3 遗传算法设计

遗传算法作为全局优化启发式算法是求解调度问题的常用方法^[24]。本文采用遗传算法对上述优化模型进行求解。对于求解项目调度的遗传算法而言,关键是对染色体的解码过程。本文根据活动重叠的特征,专门设计了针对带有活动重叠模式的资源受限项目调度问题的串行进度生成机制,是相对于 RCPSP 的串行进度生成机制的改进。

针对本文建立的优化调度模型,算法求解步骤如下:首先,对问题进行编码形成初始种群,然后,对初始种群中的每个个体进行解码,由于返工时间为随机变量,通过仿真得到工期的期望值,此值将作为适应值对种群进行评价,若满足停止规则(最大迭代次数),则算法停止,否则,执行遗传操作(选择、交叉、变异),形成新的种群后再重复执行上述步骤。

3.1 染色体编码

本文设计的遗传算法采用活动列表对染色体进行编码。染色体中的每个基因代表活动编号,每个活动在染色体中出现且仅出现 1 次。此外,生成的活动列表是优先可行的,即每个活动在列表中出现的位置必须在其所有紧前活动之后。例如,对于网络图 $G = (N, A)$, $N = \{1, 2, 3, 4, 5, 6\}$, $A = \{(1, 3), (3, 5), (2, 4), (4, 6)\}$, 则可定义染色体 $I_1 = (1, 3, 2, 5, 4, 6)$, $I_2 = (2, 4, 6, 1, 3, 5)$ 。

3.2 染色体解码

本文提出了带有活动重叠的串行进度生成机制对染色体进行解码,即以活动为阶段变量。解码步骤如下:按活动列表顺序选取活动 i ,计算活动 i 的紧前活动中最晚结束的活动,判断该活动是否可以与活动 i 重叠执行,若可以重叠执行,且满足资源约束,选择一个重叠模式使该活动尽早开始执行,同时,判断其紧前活动中其它活动是否可以与活动 i 重叠,以及是否满足资源约束;否则,活动 i 与其紧前活动不重叠执行,在满足资源约束的条件下安排其尽早开始。重复以上过程,直到所有的活动都参加排序。算法伪代码如表 1:

表 1 串行进度生成机制算法伪代码

Table 1 Algorithm pseudo code of serial schedule generation mechanism

```

初始化:
生成活动列表 AL;
 $S = \{s_1, s_2, \dots, s_n\} \leftarrow \vec{0}$ ;
对  $l = 1$  to  $N$ 
 $i \leftarrow AL(l)$ ;
计算  $Preceding\{i\}$ ; //活动  $i$  的紧前活动集合
计算活动  $i$  的紧前活动中最晚结束的活动记为  $h$ ,其完成时间为  $FT_h$ ;
若  $(h, i) \in A_0$ 
    活动  $h$  重叠信息释放的时刻记为  $\{RT_h(1), RT_h(2), \dots, RT_h(m)\}$ ;
    对  $j = 1$  to  $m$ 
         $RT_h \leftarrow RT_h(j)$ ;
         $u_k \leftarrow \sum_{i \in S_i} r_{ik}$ ; //  $k = 1$  to  $K$ 
         $S_i = \{i' \mid ST_{i'} \leq ST_i \leq ST_i + d_i + rw_{ii'}\}$ 
    若 所有  $u_k \leq R_k$ 
        对  $l = 1$  到  $length(Preceding\{i\} \setminus \{h\})$ 
             $j \leftarrow Preceding\{i\}(l)$ ;
            若  $(j, i) \in A_0$  和  $\exists ?RT_j = RT_h$ 
                活动  $i$  的开始时间  $ST_i \leftarrow RT_h$ ;
                活动  $i$  完成时间  $FT_i \leftarrow ST_i + d_i + \sum rw_{ii'}$ ;
            End If
        End For
    End If
    Break;
End If
End For
活动  $i$  的开始时间  $ST_i \leftarrow FT_h$ ;
活动  $i$  的完成时间  $FT_i \leftarrow FT_h + d_i$ ;
Flag = 1;
While Flag = = 1
    Flag = 0;
     $u_k \leftarrow \sum_{i \in S_i} r_{ik}$ ; //  $k = 1$  to  $K$ ,  $S_i = \{i' \mid ST_{i'} \leq ST_i \leq FT_i\}$ 
    If  $u_k > R_k$ 
        Flag = 1;
    End If
     $ST_i = ST_i + 1$ ;
     $FT_i = FT_i + 1$ ;
End While
End For

```

3.3 初始种群的产生

设初始种群包含 N 个个体,采用基于后悔值的偏倚随机抽样方法产生 N 个活动列表。样本算法中的优先规则采用最大紧后活动数。假设活动 j

的优先系数为 $v(j)$,活动 j 不被调度的后悔值为 $\rho_j = \max_{i \in D} v(i) - v(j)$,活动 j 被调度的概率计算式

$$\text{为 } \psi(j) = \frac{(\rho_j + 1)^\alpha}{\sum_{i \in D} (\rho_i + 1)^\alpha} \text{ ,通过调整系数 } \alpha (\alpha > 0)$$

的大小来改变偏好的大小^[25] .

3.4 遗传算子设计

1) 选择算子. 采用排序选择的方式选择个体,对每个活动列表对应的重叠后的工期进行计算,经过多次仿真后,得到的期望工期作为适应值,按照从小到大的顺序对适应值进行排序,保留适应值小的活动列表作为新的种群.

2) 交叉算子. 两点交叉,在种群中选取两个个体,并随机选取两个整数 k_1, k_2 ,且 $1 \leq k_1 < k_2 \leq n$,以交叉概率 p_c 选取两个染色体作为父代,记作 F 和 M ,生成的两个子代记作 S 和 D ,交换 F 和 M 的角色,即可生成子代 S ^[25] . 例如: 染色体 $I_F = (1 \ 3 \ 2 \ 5 \ 4 \ 6)$, $I_M = (2 \ 4 \ 6 \ 1 \ 3 \ 5)$,选择 $k_1 = 1, k_2 = 3$,以概率交叉后,得到 $I_S = (1 \ 2 \ 4 \ 3 \ 5 \ 6)$, $I_D = (2 \ 1 \ 3 \ 4 \ 6 \ 5)$.

3) 变异算子. 以变异概率 p_m 选定种群中的某个个体,交换个体中基因 J_q 和 J_{q+1} ,其中, q 代表基因位置 ($q = 1, \dots, n$) ,交换的同时保证活动列表是优先可行的^[25] . 例如, $I_F = (1 \ 3 \ 2 \ 5 \ 4 \ 6)$,

6) 变异后得到的染色体为 $I' = (1 \ 2 \ 3 \ 4 \ 6 \ 5)$.

4 实验设计与结果分析

4.1 实验设计

本节中的算例来自 PSPLIB^[26] 中 J60 问题库 (包含 480 个算例),该库采用全因子实验设计方法进行构造,控制变量采用以下 3 个特征参数^[27]: ① 网络复杂度 (network complexity, NC): 用 AON 网络 $G = (V, E)$ 中每个节点的平均箭线数量 $NC = |E|/|V|$ 表示, NC 值越大表示网络复杂度越高; ② 资源系数 (resource factor, RF): 反映活动对可更新资源的总需求强度. 定义 $RF = \frac{1}{N} \times$

$$\frac{1}{K} \sum_{j=1}^N \sum_{k=1}^K \begin{cases} 1, & \text{若 } r_{jk} > 0 \\ 0, & \text{否则} \end{cases} \text{ , } RF \text{ 值越大表示活动}$$

对资源的总需求程度越高; ③ 资源强度 (resource strength, RS): 反映资源需求量与资源供给量之间的

$$\text{关系. 定义 } RS = \frac{R_k - R_k^{\min}}{R_k^{\max} - R_k^{\min}} \text{ 其中, } R_k^{\min} \text{ 和 } R_k^{\max}$$

分别表示对资源 k 的最小需求量和最大需求量, RS 值越低表示资源越稀缺. 在本文实验设计中,这 3 个参数的取值水平分别为 $NC(1.5, 1.8, 2.1)$, $RF(0.25, 0.5, 0.75)$, $RS(0.2, 0.5, 0.7, 1)$.

另外,根据本文研究的问题,本文提出 2 个重叠与返工参数: ① 可重叠活动对占总的活动对百分比 $Percentage$,取值为 0.3、0.5 和 0.7,该参数可以控制项目中重叠活动对的数量; ② 每个时间段重叠导致下游活动返工的概率 β ,取值为 0.2、0.4 和 0.6.

本文基于 480 个算例定义不同的参数取值,共生成了 $480 \times 3 \times 3 = 4320$ 个算例,算法通过 Matlab 编程实现,并在 2×8 核 Xeon E5-2690 和 128G 内存的服务器上进行测试. 实验参数设置如下: 重叠活动对中上游活动的信息释放点,在活动工期范围内依次选取 3 个时刻点; 种群个数为 20; 交叉概率为 0.8,变异概率为 0.1; 最大进化代数设为 50 代; 仿真次数为 20.

4.2 结果分析

本文利用 PSPLIB 网站上更新的结果进行分析,该网站列出了 PSPLIB J60 问题库中所有算例 RCPSP 的最新结果. 将本文的研究结果与该库的研究结果进行比较,说明活动重叠对项目工期缩短的影响效果是合理的. 将没有重叠的项目工期,记为 $Solution$; 采用本文提出的基于串行进度生成机制的遗传算法得到的带有活动重叠的项目期望工期,记为 $Solution_P$; 最后,计算两者的相对偏差 $RD = (Solution - Solution_P) / Solution$ 和平均相对偏差 Dev .

为了考察活动重叠导致项目工期缩短的效果,采用平均相对偏差 Dev 评价网络参数、资源参数、重叠与返工参数对项目工期缩短效果的影响. 考虑 $Percentage$ 和 β 各取值 3 种情况的 9 种组合情况. 图 5 列出了在参数 $Percentage$ 和 β 不同取值情况下, Dev 随 RS, RF 变化的折线图. 从图 5 可以看出,9 种情况时 Dev 折线的变化趋势一致. 因

此,考虑到论文篇幅的限制,仅在表2中列出了 $\beta = 0.4$, $Percentage = 0.5$ 情况下 Dev 随参数 NC , RS RF 变化的计算结果开展讨论。

考察图5可以得出以下结论: ① Dev 随着 $Percentage$ 的增大或 β 的减小而增大, 这表明重叠活动对增加或返工概率减小时, 采用活动重叠模式均可以使项目工期明显缩短; ② 随着 RS 的增大, RF 对 Dev 的影响减弱, 这说明资源可用量充足的情况下, 活动对资源的需求程度的大小不会影响项目按计划执行, 这时, 项目工期缩短的效果

差异不大; ③ RS 增大, Dev 随之增大, 这说明随着资源强度的增大, 资源的可用量相对较大, 当执行过程中重叠活动对增加, 采用重叠模式可使工期明显缩短。

考察表2列出的计算结果, 可以得出以下结论: ① 当 NC 变化时, Dev 的变化并不明显, 说明网络结构的复杂程度对活动重叠的影响不大; ② RF 减小, Dev 随之增大, 这说明随着资源系数减小, 活动对资源的需求量减少, 使得能够重叠的活动对增加, 进而使得项目工期明显缩短。

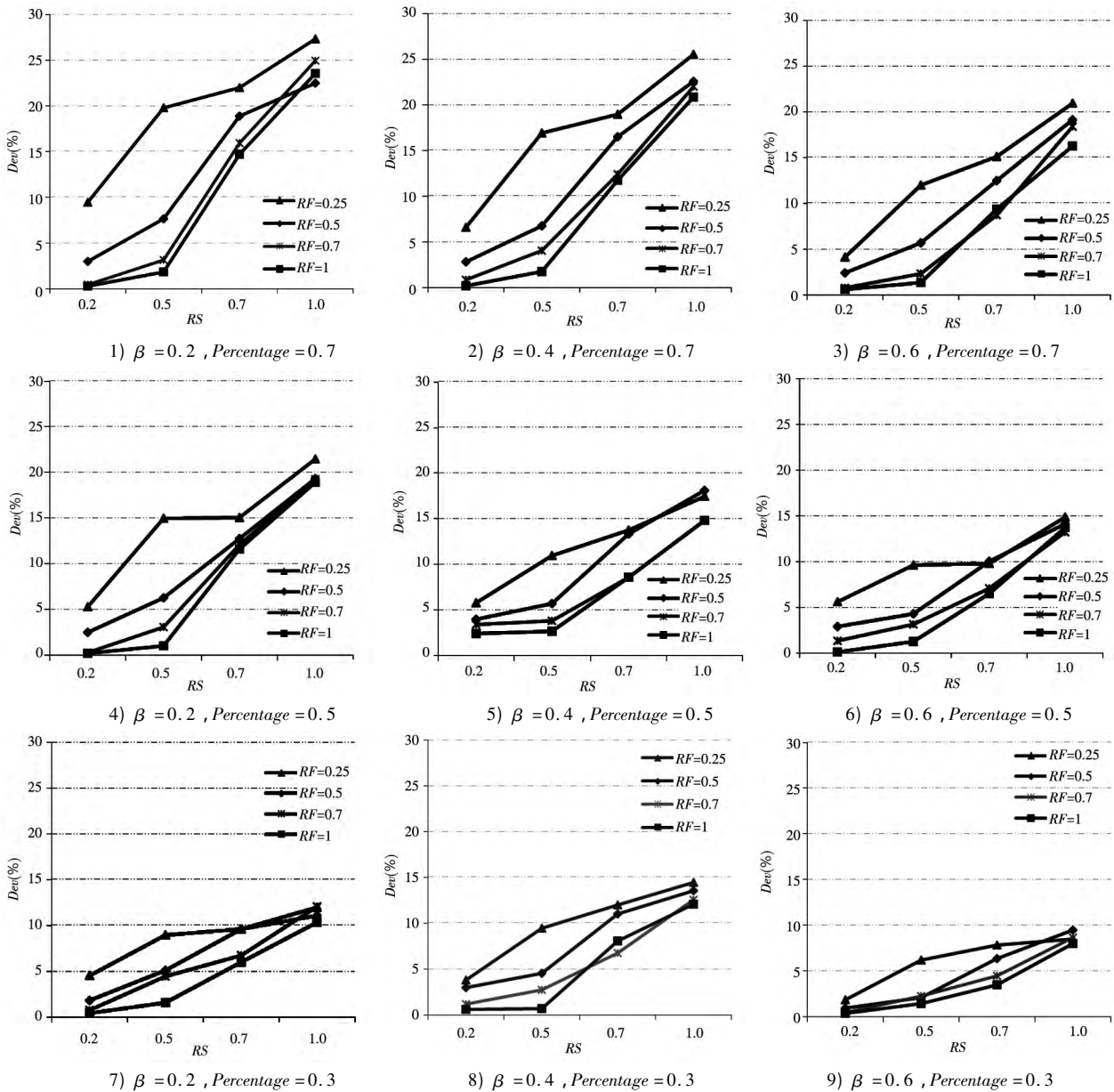


图5 Dev 随 $Percentage$, β 以及 RS RF 的变化折线图

Fig. 5 Line chart of Dev with $Percentage$, β , RS , RF

表 2 $\beta = 0.4$, $Percentage = 0.5$ 下 Dev (%) 的计算结果

Table 2 Results of Dev with $\beta = 0.4$, $Percentage = 0.5$

活动个数	NC	RF	RS			
			0.2	0.5	0.7	1.0
60	1.5	0.25	5.13	11.88	15.01	16.91
		0.5	2.29	7.20	15.06	16.86
		0.75	4.05	5.60	8.21	12.62
		1.0	3.54	4.92	12.47	13.74
	1.8	0.25	6.46	11.38	13.10	18.66
		0.5	3.33	3.90	15.51	18.55
		0.75	3.82	3.48	9.19	15.61
		1.0	2.30	1.51	9.90	14.92
	2.1	0.25	5.89	9.63	13.12	16.71
		0.5	6.20	5.91	9.50	19.02
		0.75	2.26	2.41	8.44	16.32
		1.0	1.46	1.68	3.26	15.73

5 结束语

本文基于活动重叠导致下游活动返工不确定的假设,研究了带有活动重叠的资源受限项目调度问题.对活动重叠和重叠模式进行了详细的描述,提出了活动重叠导致下游活动返工时间的二项分布概率模型,建立了复杂产品研发系统项目

调度优化模型,并利用本文提出的遗传算法对 PSPLIB 中 J60 问题库算例进行了测试研究.研究结果表明,在以下 4 种情形下,采用活动重叠模式可使项目工期明显缩短:①资源可用量充足;②活动对资源的需求量较小;③可重叠的活动对较多;④下游活动返工概率较小.项目管理者可以应用本文提出的模型与算法,依据实际项目中的资源、重叠与返工参数制定合理的项目进度计划.

参考文献:

- [1]李洪波,徐哲.鲁棒项目调度研究综述[J].系统工程,2014,32(2):123-131.
Li Hongbo, Xu Zhe. Robust project scheduling: A review[J]. Systems Engineering, 2014, 32(2): 123-131. (in Chinese)
- [2]汪峥,严洪森,刘霞岭,等.并行工程产品开发过程量化建模与计划定制[J].管理科学学报,2000,3(4):46-59.
Wang Zheng, Yan Hongsen, Liu Xialing, et al. Quantitative modeling and planning of the product development process in concurrent engineering[J]. Journal of Management Sciences in China, 2000, 3(4): 46-59. (in Chinese)
- [3]苏志雄,乞建勋,张立辉.广义优先关系下隐性时间、隐性时差和伪时差[J].管理科学学报,2015,18(12):68-78.
Su Zhixiong, Qi Jianxun, Zhang Lihui. Hidden times, hidden floats and false floats under generalized precedence relations[J]. Journal of Management Sciences in China, 2015, 18(12): 68-78. (in Chinese)
- [4]何正文,刘入境,胡信布.基于合同双方交互作用的项目调度优化[J].管理科学学报,2014,17(8):48-59.
He Zhengwen, Liu Renjing, Hu Xinbu. Project scheduling optimization based on interaction between two parties of contracts[J]. Journal of Management Sciences in China, 2014, 17(8): 48-59. (in Chinese)
- [5]Blazewicz J, Lenstra J K, Rinnoy K. Scheduling projects subject to resource constraints: Classification and complexity[J]. Discrete Applied Mathematics, 1983, 5(1): 11-24.
- [6]Cho S H, Eppinger S D. A simulation-based process model for managing complex design projects[J]. IEEE Transactions on Engineering Management, 2005, 52(3): 316-328.

- [7]李洪波,徐哲. 考虑活动随机重叠和资源冲突的复杂产品开发流程仿真建模[J]. 系统工程与电子技术,2012,34(7): 1412-1418.
Li Hongbo, Xu Zhe. Simulation modeling of complex product development process considering stochastic activities overlapping and resource conflict[J]. Systems Engineering and Electronic, 2012, 34(7): 1412-1418. (in Chinese)
- [8]李洪波,徐哲,于静. 基于DSM的研发项目流程多目标仿真优化[J]. 系统工程理论与实践,2015,35(1): 142-149.
Li Hongbo, Xu Zhe, Yu Jing. Multi-objective simulation optimization for the process of R&D projects based on DSM[J]. Systems Engineering-Theory & Practice, 2015, 35(1): 142-149. (in Chinese)
- [9]Maheswari J U, Varghese K. Project scheduling using dependency structure matrix[J]. International Journal of Project Management, 2005, 23(3): 223-230.
- [10]白思俊,万小兵. 基于设计结构矩阵的项目进度周期[J]. 系统工程理论与实践,2008,28(11): 51-54.
Bai Sijun, Wan Xiaobing. Project schedule cycle based on design structure matrix[J]. System Engineering-Theory & Practice, 2008, 28(11): 51-54. (in Chinese)
- [11]杨青,黄建美. 基于活动重叠的DSM项目时间计算及排序优化[J]. 系统工程理论与实践,2011,31(3): 495-504.
Yang Qing, Huang Jianmei. Project time calculation and optimization based on DSM activities overlapping[J]. System Engineering-Theory & Practice, 2011, 31(3): 495-504. (in Chinese)
- [12]张春生,严广乐. 基于活动重叠的DSM项目进度优化与仿真[J]. 运筹与管理,2013,22(3): 36-44.
Zhang Chunsheng, Yan Guangle. Study on optimization and simulation of DSM project schedule based on activities overlapping[J]. Operations Research and Management Science, 2013, 22(3): 36-44. (in Chinese)
- [13]Gerk J E Y, Qassim R Y. Project acceleration via activity crashing, overlapping and substitution[J]. IEEE Transactions on Engineering Management, 2008, 55(4): 590-601.
- [14]Lin J, Chai K H, Brombacher A C, et al. Optimal overlapping and functional interaction in product development[J]. European Journal of Operational Research, 2009, 196(3): 1158-1169.
- [15]Lin J, Qian Y, Cui W, et al. Overlapping and communication policies in product development[J]. European Journal of Operational Research, 2010, 201(3): 737-750.
- [16]Wang J T, Lin Y I. An overlapping process model to assess schedule risk for new product development[J]. Computer and Industrial Engineering, 2009, 57(2): 460-474.
- [17]Berthaut F, Greze L, Pellerin R, et al. Optimal resource-constrained project scheduling with overlapping modes[EB/OL]. <https://www.cirrelt.ca/DocumentsTravail/CIRRELT-2011-09.pdf>, 2011a.
- [18]Berthaut F, Pellerin R, Perrier N, et al. Time cost tradeoffs in resource-constrained project scheduling with overlapping modes[EB/OL]. <https://www.cirrelt.ca/DocumentsTravail/CIRRELT-2011-10.pdf>, 2011b.
- [19]Greze L, Pellerin R, Leclaire P, et al. A heuristic method for resource-constrained project scheduling with activity overlapping[J]. Journal of Intelligent Manufacturing, 2014, 25(4): 797-811.
- [20]Bozejko W, Hejducki Z, Uchronski M, et al. Solving resource-constrained construction scheduling problems with overlaps by metaheuristic[J]. Journal of Civil Engineering and Management, 2014, 20(5): 649-659.
- [21]Koyuncu E, Erol R. PSO based approach for scheduling NPD projects including overlapping process[J]. Computers & Industrial Engineering, 2015, 85(7): 316-327.
- [22]于静,徐哲,李洪波. 带有活动重叠的资源受限项目调度问题建模与求解[J]. 系统工程理论与实践,2015,35(5): 1236-1245.
Yu Jing, Xu Zhe, Li Hongbo. Modeling and solving the resource-constrained project scheduling problem with activities overlapping[J]. Systems Engineering-Theory & Practice, 2015, 35(5): 1236-1245. (in Chinese)
- [23]Krishnan V, Eppinger S, Whitney D. Accelerating product development by the exchange of preliminary product design information[J]. ASME Journal of Mechanical Design, 1995, 117(4): 491-498.
- [24]宋莉波,徐学军,孙延明,等. 一种求解柔性工作车间调度问题的混合遗传算法[J]. 管理科学学报,2010,13(11): 49-54.

- Song Libo , Xu Xuejun , Sun Yanming , et al. A hybrid genetic algorithm for flexible job shop scheduling problem [J]. Journal of Management Sciences in China , 2010 , 13(11) : 49 – 54. (in Chinese)
- [25] Kolish R. Serial and parallel resource-constrained project scheduling methods revisited: Theory and computation [J]. European Journal of Operational Research , 1996 , 90(2) : 320 – 333.
- [26] Kolish R , Sprecher A. PSPLIB-A project scheduling problem library [J]. European Journal of Operational Research , 1996 , 96(1) : 205 – 216.
- [27] Kolish R , Sprecher A , Drexl A. Characterization and generation of a general class of resource-constrained project scheduling problems [J]. Management Science , 1995 , 41(10) : 1693 – 1703.

Project scheduling optimization with overlapping modes and resource constraints

*YU Jing*¹ , *XU Zhe*^{2*} , *XIE Fang*²

1. School of Management , Tianjin University of Technology , Tianjin 300383 , China;

2. School of Economics and Management , Beihang University , Beijing 100191 , China

Abstract: Overlapping activities are an effective means to accelerate the duration of R&D projects. Project scheduling with overlapping modes and resource constraints is an extension of the classical problem. Firstly , based on an in-depth analysis of the influence of overlapping activities on project scheduling , a detailed description and modeling of overlapping and its uncertainty is proposed. Then , a binomial probability model for reworking on downstream activities is built. Thirdly , an optimization model is built whose objective is to minimize the expectation of duration , and a serial schedule generation mechanism is designed as decoding operator for genetic algorithm. Finally , a heuristic method is used to test on the PSPLIB of J60 problems (480 cases) . The duration of the project is analyzed as the network and resource parameters and overlapping parameters change. The results show that: the smaller the intensity of activities on resource requirements or the lower the scarcity of resources , the larger the number of overlapped activities , and the more the shortening in the project duration; Network complexity has little effect on shortening the duration; The more overlapping activities or the smaller reworking probability on downstream activities , the more obvious the impact of shortening the duration.

Key words: project scheduling; genetic algorithm; overlapping; resource; rework