

资源空窗期及任务可拆分的资源投入问题研究

陆志强[†], 周皓雪

(同济大学 机械与能源工程学院, 上海 201804)

摘要:考虑飞机装配过程中任务可拆分及资源存在空窗期的两大特性,对飞机移动生产线资源投入问题进行模型与算法研究.针对部分任务存在已知拆分模式及拆分惩罚的情形,设计了求解该问题的改进遗传算法,对传统实数交叉操作进行优化,提出了基于染色体适应值的交叉方法,并在数值实验中对相关参数的取值范围进行了敏感性分析;同时,提出了基于任务开始时间选择概率的变异机制.对满足优化条件的任务调度方案,结合空窗期的位置,评判各可拆分任务可否通过选取新的拆分模式重新调度执行,对不同情形进行总结归纳,通过局部操作进一步降低目标资源量.数值实验表明:通过本文算法对求解带资源空窗期的任务不可拆分问题与基本问题的结果对比,得到任务数分别为 10、16、30、60、90 算例的目标值平均增量达到 4.3%;对求解本文问题与任务不可拆分问题的结果对比,平均优化率达 3.5%,证明了本文算法的有效性,同时证明将任务拆分纳入考虑资源空窗期的资源投入问题中,可提高问题求解的灵活性,从而获得较好的调度结果.

关键词:资源投入问题;资源空窗期;任务拆分;遗传算法

中图分类号:F273

文献标志码:A

Resource Investment Problem with Activity Splitting and Resource Window

LU Zhiqiang[†], ZHOU Haoxue

(School of Mechanical Engineering, Tongji University, Shanghai 201804, China)

Abstract: Considering the two characteristics of activity splitting and resource window in the process of aircraft assembly, the model and algorithm of Resource Investment Problem on aircraft mobile production line were studied. Aiming at the situation that some activities have known splitting mode and splitting punishment, an improved genetic algorithm for solving this problem was designed. The traditional real value crossover operation was optimized, and a crossover method based on chromosome fitness value was proposed. Sensitivity analysis was carried out on the range of values of the relevant parameters. A mutation mechanism based on the probability of selection of activity start time was also proposed. For a scheduling scheme that satisfies the optimization conditions, combined with the position of the resource window, after judging whether the splitting activities can be re-scheduled and executed by selecting a new splitting mode and summarizing the different situations, the target resources were further reduced by local operations. The numerical experiments show that, compared with the results of solving the problem of non-split activities with resource window and the basic problem, the average value of the target for the 10, 16, 30, 60, 90 activities is

* 收稿日期:2019-03-07

基金项目:国家自然科学基金资助项目(61473211, 71171130), National Natural Science Foundation of China(61473211, 71171130)

作者简介:陆志强(1968—),男,江苏太仓人,同济大学教授,博士生导师

[†] 通讯联系人, E-mail: zhiqianglu@tongji.edu.cn

4.3%. For the comparison between the results of solving this problem and the non-split problem, the average optimization rate is 3.5%, which proves the effectiveness of the algorithm. At the same time, it is proved that the activity splitting is included in the Resource Investment Problem considering the resource window, which can improve the flexibility of problem solving and obtain better scheduling results.

Key words: resource investment problem; resource window; activity splitting; genetic algorithm

目前,我国的飞机总装大部分仍沿用传统的固定站式装配模式,批量生产能力匮乏.随着社会经济的发展,民用机的需求激增,在航空制造中深入推行基于精益思想的飞机移动式装配线模式,可提高装配效率和质量,按需批量生产是中国飞机制造向世界水平迈进的必经之路.装配线上单架飞机的总装调度决策问题依据不同的生产目的,可将其抽象地分为资源受限项目调度问题^[1](Resource-Constrained Project Scheduling Problem, RCPSP)和资源投入型问题^[2](Resource Investment Problem, RIP).后者即在给定的制造期限约束下,优化配置完成项目所需的线边装配人员、工装、工具及仪器设备等各类资源,达到最小化资源投入总成本的目的.

资源投入问题最早由 Morhing^[3]在一个桥梁建设项目中提出,其证明了此问题为 NP-hard,提出了求解该问题的图解精确算法;Drexel 等^[4]基于拉格朗日松弛和列生成方法针对 RIP 提出了两个下界算法,并且通过算例实验与 Morhing^[3]提出的算法进行了比较;Demeulemeester^[5]参照针对 RCPSP 问题提出的基于深度优先的分支定界方法,构建了求解 RIP 的优化算法.上述精确算法虽然求解精度较好,但是只能应用于小规模问题,对于诸如飞机生产制造的大规模问题,大量国内外学者提出启发式算法与元启发式算法进行求解. Zhu 等^[6]将 RIP 问题的求解划分为排序问题与资源决策问题两个部分,提出了一种多起点迭代搜索的启发式方法; Song 等^[7]利用带有局部搜索的进化算法求解,在优化过程中不允许项目延迟; Qi 等^[8]提出了改进的粒子群优化算法求解多模式下资源可用性成本问题的调度生成方案; Javanmard 等^[9]设计了一种基于遗传和粒子群的算法,通过校准参数和染色体结构,保证解的可行性;任逸飞等^[10]通过基于全局资源水平影响的任务调度评估策略,提出改进型遗传算法优化非关键任务的调度位置,然而该评估策略在求解大规模算例时效率较低.

上述文献均为对基本 RIP 的研究,存在大量的

假设,模型偏理想化,对工况条件复杂的飞机总装过程缺乏指导意义;从算法设计上来看,多数算法局限于将 RIP 转化成 RCPSP 的求解思路,导致某些资源组合可能超出工期约束,针对资源的搜索方向性较差,求解效率低.现代化的飞机生产制造工序繁琐,技术复杂性高,总装过程需要大量不同的专业性人才.因此,在实际生产过程中往往会存在由于关键技术人员紧缺而必须延迟调度的情形,将资源存在的不可用时间段称为资源空窗期.将资源空窗期约束引入到 RIP 模型构建中,更加贴合飞机移动式装配线应用背景.现有文献中仅考虑资源空窗期的 RCPSP,考虑资源特点的 RIP 研究很少.廖广瑞等^[11]针对资源的多技能和时间窗属性,设计了一种基于优先规则的 Rollout 求解算法,并嵌入了一种启发式资源分配方法对资源进行快速分配; Jirachai 等^[12]研究了考虑资源空窗期的多模式 RCPSP 问题,假定任务可拆分执行,设计了求解小规模该类问题的分支定界算法; 綦方中等^[13]研究了基于时间窗的多项目资源分配问题,提出了增加时间窗参数以及惩罚因子的多项目管理资源分配模型.总结上述研究可以发现,空窗期的引入会导致任务调度的灵活性下降,从而加剧空窗期前后位置的资源投入量,增加项目的成本支出.

另一方面,现有 RIP 问题的研究通常假设任务不可拆分,但从飞机装配工艺过程中可以发现,某些任务过程并不是连续的,是可以拆分执行的.从 Javanmard 等^[9]和陆志强等^[14]对于任务可拆分条件下 RIP 的研究中可以看出,考虑任务可拆分特性增加任务调度的柔性,均衡资源使用,降低成本投入.而在以上文献中,虽然考虑了任务可拆分执行的情形,却并未设置相应的拆分惩罚机制.在飞机实际装配生产中,某项任务在执行过程中中断,待到后续某时刻继续执行时,需要重新进行装配准备工作,势必造成时间成本或资源成本的增加.

综上所述,本文将任务拆分及伴有拆分惩罚的

情形纳入考虑资源空窗期的资源投入问题中,构建符合实际应用条件的模型,重点分析同时考虑任务可拆分及资源存在空窗期这两大特性的交互影响,设计相应的改进型遗传算法及优化操作进行求解和验证.最后通过实例分析,以某型号支线客机驾驶舱装配工位的调度为例,验证了本文算法的有效性.

1 问题描述及数学模型

1.1 问题描述

假设项目由若干项任务构成,记给定项目工期为 T ,项目中包含 J 项任务,任务执行时间为 d_j , $\text{pre}(j)$ 为第 j 项任务的紧前任务构成的集合, j^* 为任务 j 的紧前任务, $\text{suc}(j)$ 为其紧后任务构成的集合. J 项任务各有 $M_j(j \in J)$ 种执行模式,其中存在 K 项任务,构成集合 U ,满足 $M_k > 1(k \in K)$,存在拆分执行模式,其余任务满足 $M_j = 1(j \in J, j \notin K)$,不能拆分执行. r_{jmp}^1, r_{jmp}^2 分别表示任务 j 以模式 m 执行下第一段和第二段子任务对资源 p 的需求量, d_{jm}^1, d_{jm}^2 分别表示任务 j 以模式 m 执行下第一段和第二段子任务的执行时间,满足 $d_{jm}^1 + d_{jm}^2 = d_j$. V_{jm}^1, V_{jm}^2 分别表示任务 $j(j \in J)$ 以模式 m 下第一段和第二段子任务所需资源种类,对应的集合大小分别为 S_{jm}^1 和 S_{jm}^2 . θ 为任务选择拆分模式执行下的工期惩罚量.

全部任务共享 P 种可更新资源,每种资源的单位使用成本为 $C_p(p \in P)$.其中资源分关键资源和普通资源,关键资源存在空窗期,部分时刻不可用,普通资源在任意时刻皆可使用; W_p 为资源在项目周期 T 内的可用时间窗集,普通资源满足 $W_p = [1, T]$,引入参量 Z_{pt} ,表示资源 p 在 t 时刻是否可用,若 $t \in W_p$,则 $Z_{pt} = 1$,否则 $Z_{pt} = 0$.任务间存在着一定的时序关系,当满足紧前任务皆完成调度且所需资源充足的情况下,任务即可开始执行.将时间进行离散处理,各任务在时刻 t 对资源 p 的总需求量用 R_{tp} 表示.最小化各时刻 $t(t = 1, 2, \dots, T)$ 每种资源 p 的最大需求量 R_{tp} 的总成本之和是模型的优化目标.

模型中决策变量包括:

$$x_{jmt} = \begin{cases} 1, & \text{表示任务 } j(j = 1, 2, \dots, J) \text{ 以模式 } m \text{ 在时刻 } t \text{ 执行} \\ 0, & \text{否则} \end{cases}$$

$$y_{jm} = \begin{cases} 1, & \text{表示任务 } j(j = 1, 2, \dots, J) \text{ 以模式 } m \text{ 执行} \\ 0, & \text{否则} \end{cases}$$

1.2 数学模型

假设同一时刻只能执行一项任务,且仅当该任

务执行完毕时,占用的资源方可释放,即资源非抢占式.

问题的数学模型为:

$$\min \sum_{p=1}^P (C_p \max R_{tp}) \quad (1)$$

s.t

$$\sum_{m=1}^{M_j} y_{jm} = 1, \forall j \in J \quad (2)$$

$$\sum_{t=1}^T x_{jmt} = (d_{jm} + (m-1)\theta) \times y_{jm}, \forall j \in J, \forall m \in M_j \quad (3)$$

$$R_{tp} = \sum_{j=1}^n r_{jmp} x_{jmt}, \forall t \in T, \forall p \in P, \forall m \in M_j \quad (4)$$

$$\sum_{p \in V_m} Z_{pt} x_{jmt} = S_{jm} x_{jmt}, \forall t \in T, \forall p \in P, \forall m \in M_j \quad (5)$$

$$(d_{j^*m} + (m-1)\theta) x_{j^*mt} \leq \sum_{i=1}^{t-1} x_{j^*mi}, \forall j^* \in \text{pre}(j), m \in M_j \quad (6)$$

$$tx_{jmt} \leq T, \forall j \in J, \forall t \in T, \forall m \in M_j \quad (7)$$

$$x_{jmt}, y_{jm} \in \{0, 1\}, \forall j \in J, \forall t \in T, \forall m \in M_j \quad (8)$$

式(1)为目标函数,表示该问题是优化整个周期内各资源使用的最大总成本;式(2)为约束条件,表示任务只能选择一种执行模式;式(3)表示任务在整个项目工期内的执行时间之和必须满足该任务工期约束;式(4)计算了各时间段 t 内每种资源 p 的使用量;式(5)表示任务只有在所需资源皆可用时才能执行;任务之间需满足时序约束(式(6)),即满足一定的优先关系;式(7)表示所有任务的最晚结束时间不应大于项目给定工期;式(8)定义了决策变量 x_{jmt} 与 y_{jm} 的可行域.

2 算法设计

本文采用直接确定任务位置来获得任务调度计划及资源使用情况的算法设计思路,通过确定任务开始时间的方式来间接获得项目中的资源投入量,针对可拆分任务,将其结束时间也在编码中体现.以遗传算法为搜索框架,设计基于染色体适应值的实数交叉方式及基于任务开始时间选择概率的变异方式进行遗传操作,在解码过程中针对各染色体调度计划,结合资源空窗期与可拆分任务的拆分模式,进行拆分任务的模型更换,进一步降低项目资源投入量,优化目标资源成本.

2.1 基于任务开始时间的编码

本文在可行范围内,采用对任务开始时间进行编码的方式,直观展现各任务调度位置;为体现拆分任务的拆分与否,同时对可拆分任务的结束时间进行编码.当项目中存在 J 项任务和 M 项可拆分任务时,编码的长度为 $J+M$,其中编码第 1 到 J 位表示任务 j 的开始时间;编码第 $J+1$ 到 $J+M$ 位表示第 m 个可拆分任务的结束时间.以图 1 所示算例为例,编码第 8 位表示可拆分任务 3 的结束时间为时刻 3,故可知该任务选择的是连续执行的模式.

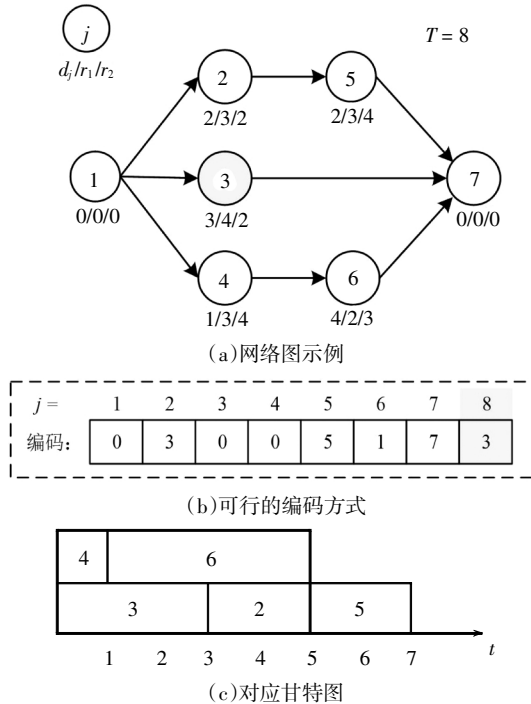


图1 编码

Fig.1 Coding

2.2 实数交叉

经过轮盘赌选择机制获得的种群,根据交叉概率 P_c 选择需要进行交叉操作的染色体组成集合 C , P_c 由式(9)求得,其中, P_{c1} 、 P_{c2} 为预先定义的参数, f_{avg} 、 f_{min} 分别为所有染色体的平均适应度值和最小适应度值.

$$P_c = \begin{cases} P_{c1} - \frac{(P_{c1} - P_{c2})(f - f_{avg})}{f_{avg} - f_{min}}, & f \geq f_{avg} \\ P_{c1}, & f < f_{avg} \end{cases} \quad (9)$$

为提高染色体交叉后的多样性,本文结合文献[15]设计了一种新的基于染色体适应值的交叉方式.不同父体之间适应值的差异不仅对繁殖过程有所裨益,而且可能会为生成更优异的后代提供潜在的搜索方向.定义搜索方向 v 为:

$$v = (\text{fitness}_h - \text{fitness}_l) / \text{fitness}_h \quad (10)$$

式中: fitness_h 和 fitness_l 分别表示父代中适应值较大和较小的染色体.染色体 i 的适应值 fitness_i 通过式(11)求得, O_i 为该染色体编码对应的资源使用量, LB 为项目最低资源用量,通过式(12)求取,其中 r_{jp} 表示任务需要的单位资源的数量.

$$\text{fitness}_i = \frac{LB}{O_i} \quad (11)$$

$$LB = \sum_{p \in P} C_p \left[\frac{\sum_{j \in J} r_{jp} d_j}{T} \right] \quad (12)$$

对应父代染色体 F_h 与 F_l 的编码位 x , 定义子代染色体 CH_1 、 CH_2 通过式(13)所示交叉操作求得:

$$\begin{cases} CH_1^x = F_l^x + \alpha_1(F_h^x - F_l^x)v \\ CH_2^x = F_l^x + \alpha_2(F_h^x - F_l^x)v \end{cases} \quad (13)$$

式中: α_1 为[0,1]上均匀分布的随机数; α_2 为[1,2]上均匀分布的随机数.这样, CH_1^x 位于区间 $[F_l^x, F_h^x]$ 内, CH_2^x 可能处于 $[F_l^x, F_h^x]$ 界外,保证了群体中基因的多样性,避免了收敛于局部最优的可能.

2.3 基于任务开始时间选择概率的变异

经过交叉之后得到的染色体,根据变异概率 P_m 选择需要进行变异操作的染色体组成集合 M , P_m 同样通过式(9)求得.

由于项目存在工期上限,因此各任务 j 的开始时间 S_j 存在决策区间 $[ES_j, LS_j]$.各任务不同开始时间的组合对应不同项目目标资源量,随着遗传算法不断迭代搜索,较低的目标值被发现的同时,所得调度计划中的任务开始时间也趋向于更小的决策区间 $[ES_j', LS_j']$,甚至趋向于某一精确值.定义任务开始时间选择概率 P_s' ($ES \leq S \leq LS$) 和该开始时间下需要多投入的资源量 R_s' .在初始种群中,令 $R_s' = 1$.之后的每条染色体都根据当代的资源用量 R 更新 R_s' 为:

$$R_s' = R_s' + \max(0, R - R_0) \quad (14)$$

R_0 为当前得到的最低资源用量.染色体中需要变异的任务的开始时间 S 被选中的概率 P_s' 为:

$$P_s' = \frac{1}{R_s'} / \sum_{s=ES}^{LS} \frac{1}{R_s'} \quad (15)$$

在迭代过程中,资源用量越大的任务开始时间 S 对应的 R_s' 会不断增大,使得其被选中的概率 P_s' 不断减小;反之,越接近 R_0 的任务开始时间对应的 R_s' 较小,使得相应 P_s' 被选中的概率变大,由此可以不断缩小任务开始时间的决策区间,使得任务调度向着产生更好的解的方向收敛.

2.4 局部优化操作

当部分任务存在可拆分执行模式,且该模式下

某一拆分子任务的执行过程中无需使用存在空窗期的资源时,可考虑通过调整可拆分任务的执行模式及其余部分任务的执行位置,降低相应资源的占用量.

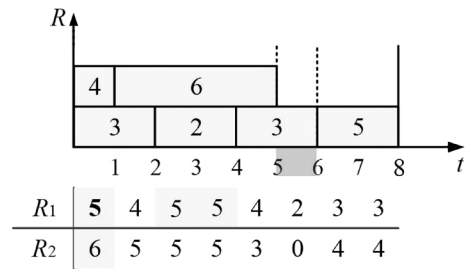
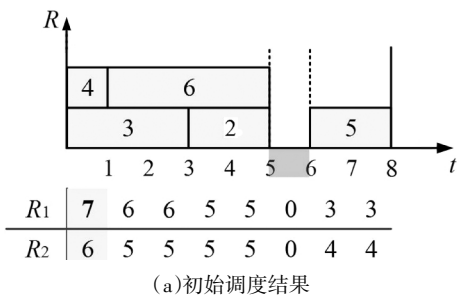
2.4.1 可拆分任务优化调整

针对空窗期位置及任务拆分模式等信息,当资源 g 满足 $g \neq k$, 最大使用量出现的时刻 $H_g = 1, t_{sj} \leq M_{g1} \leq t_{fj}$ 时, 以下两种情形通过更换可拆分任务的拆分模式来降低资源 g 的投入量. 记对应染色体下各资源 $p(p \in P)$ 的最大使用量和最大使用量出现的时刻分别为 $V_p, M_{ph}(h = 1, \dots, H_p)$ (可能存在多个位置资源用量相同), 资源 $k(k \in P)$ 存在空窗期 $[m_k, n_k]$, 可拆分任务 $j(j \in J)$ 在该调度计划下的执行时刻为 $[t_{sj}, f_{fj}]$, 可选拆分模式下前后两段所用资源种类分别为集合 B_j^a 和 B_j^b .

情形 1: $m_k > M_{g1}, k \notin B_j^b$, 区间 $[M_{g1}, n_k]$ 属于任务 j 的浮动区间. 资源 k 的空窗期位于最大资源时刻之后, 任务 j 可以在区间 $[M_{g1}, n_k]$ 内任意时刻开始执行, 且其拆分模式后半段无需使用资源 k , 该情形下可通过更换任务 j 为拆分模式执行, 重调度区间 $[M_{g1}, n_k]$ 内的任务, 降低资源 k 的 V_k 值.

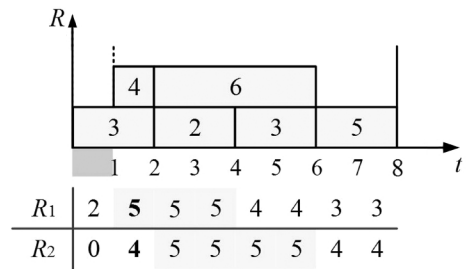
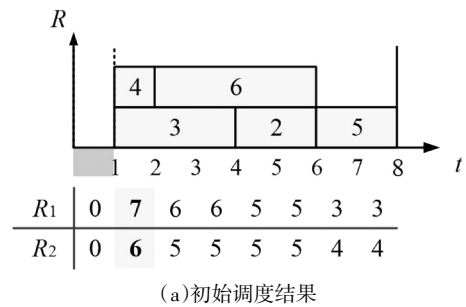
仍以图 1(a)所示项目网络图为例, 假设资源 2 存在空窗期 $[5, 6]$, 任务 3 为可拆分任务, 拆分模式为 $2/2/2$ 和 $1/2/0$, 拆分任务的惩罚时间成本为 $\theta = 1$. 对应染色体编码下初始调度结果如图 2(a)所示, 其中纵坐标 R 代表资源总用量, R_1 和 R_2 为每个任务执行时所消耗的资源量, 横坐标表示时间. 时刻 6 内无可执行任务, 资源 1 和资源 2 的最大资源用量时刻皆为时刻 1, 资源用量分别为 7 和 6. 鉴于可拆分任务 3 满足情形 1, 选取拆分模式执行后, 将任务 2 的开始时间提前一个单位, 任务 3 分别在时间段 $[0, 2], [4, 6]$ 内执行, 资源 1 的用量 V_1 降低了 2 个单位, 调度结果如图 2(b)所示.

情形 2: $n_k > M_{g1}, k \notin B_j^a$, 区间 $[m_k, M_{g1}]$ 属于任务 j 的浮动区间. 资源 k 的空窗期位于最大资源时刻之前, 任务 j 可以在区间 $[m_k, M_{g1}]$ 内任意时刻开始执行, 且其拆分模式前半段无需使用资源 k , 该情形下可通过更换任务 j 为拆分模式执行, 重调度区间 $[m_k, M_{g1}]$ 内的任务, 降低资源 k 的 V_k 值.



(b) 调度结果
图 2 优化情形 1
Fig.2 Optimization situation 1

以图 1(a)所示项目网络图为例, 假设资源 2 存在空窗期 $[0, 1]$, 任务 3 为可拆分任务, 拆分模式分别为 $2/2/0$ 和 $1/2/2$, 拆分任务的惩罚时间成本为 $\theta = 1$. 对应染色体编码下初始调度结果如图 3(a)所示, 时刻 1 内无可执行任务, 资源 1 和资源 2 的最大资源用量时刻皆为时刻 1, 资源用量分别为 7 和 6. 鉴于可拆分任务 3 满足情形 2, 选取拆分模式执行后, 将任务 2 的开始时间提前两个单位, 任务 3 分别在时间段 $[0, 2], [4, 6]$ 内执行, 资源 1 的用量 V_1 降低了 2 个单位, 同时资源 2 的用量 V_2 降低了 1 个单位, 调度结果如图 3(b)所示.



(b) 调度结果
图 3 优化情形 2
Fig.3 Optimization Situation 2

2.4.2 算法步骤

步骤 1 对染色体 F 进行解码操作, 得到各资源 $p(p \in P)$ 的最大使用量和最大使用量出现的时刻 $V_p, M_{ph}(h=1, \dots, H_p)$ 及项目资源水平 X , 记资源 $k(k \in P)$ 的空窗期为 $[m_k, n_k]$, 可拆分任务 $j(j \in J)$ 的执行时刻为 $[S_j, F_j]$, 可选拆分模式下前后两段工期分别为 d_j^a 和

d_j^b . 令 $p = 1$, 转步骤 2.

步骤 2 若 $p > P$, 转步骤 12; 否则, 若资源 p 满足 $p \neq k, H_p = 1, S_j \leq M_{\alpha} \leq F_j$ 转步骤 3; 否则 $p = p + 1$, 转步骤 2 继续判断.

步骤 3 记集合 Z , 令 $i = 2, y = j$, 满足情形 1 转步骤 4; 否则转步骤 8.

步骤 4 $i > J$, 转步骤 5; 否则, 若满足 $S_i = F_y$ 且 $F_i \leq m_k$, 令 $Z = Z \cup i; i = i + 1$, 转步骤 4.

步骤 5 $F_i \leq m_k$, 令 $y = i$, 转步骤 4; 否则, 转步骤 6.

步骤 6 若 $n_k - F_y \geq \theta$, 转步骤 7; 否则转步骤 12.

步骤 7 更新集合 Z 中各任务 z 的开始和结束时间分别为 $S_z = S_z - (d_j - d_j^a), F_z = S_z - d_z$, 可拆分任务 j 的执行时刻为 $[S_j, S_j + d_j^a]$ 和 $[F_z, F_z + d_j^a]$, 重新计算项目资源水平 X , 令 $p = 1$, 转步骤 2.

步骤 8 $i > J$, 转步骤 9; 否则, 若满足 $F_i = S_y$ 且 $S_i \geq n_k$, 令 $Z = Z \cup i; i = i + 1$, 转步骤 8.

步骤 9 $S_i \geq n_k$, 令 $y = i$, 转步骤 8; 否则, 转步骤 10.

步骤 10 若 $S_y - m_k \geq \theta$, 转步骤 7; 否则转步骤 12.

步骤 11 更新集合 Z 中各任务 z 的开始和结束时间分别为 $F_z = F_z + (d_j - d_j^b), S_z = F_z - d_z$, 可拆分任务 j 的执行时刻为 $[S_z - d_j^b, S_z]$ 和 $[F_j - d_j^b, F_j]$, 重新计算项目资源水平 X , 令 $p = 1$, 转步骤 2.

步骤 12 输出项目资源水平 X .

算法框架如图 4 所示.

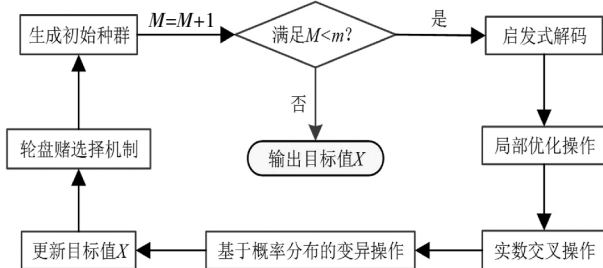


图 4 算法框架

Fig.4 Algorithm framework

3 数值实验

为验证上述改进遗传算法对于求解考虑资源空窗期及任务可拆分特性下的资源投入问题的有效性, 本文运用 C#(Visual Studio 2013)编程实现该算法. 测试平台为 Intel Core i5 处理器, 2.40 GHz 主频, 8 G 内存. 通过对 PSPLIB 中的算例进行改造, 获得了

本文所用的测试算例, 对于实验所需而基本算例中未提供的参数, 通过随机数的方式来确定, 其中工期上限 T 取关键路径长的 1.2 倍, 资源空窗期通过在项目工期内随机产生不同长度的区间获得. 基于不同规模的算例, 对小规模算例选择 2 项任务作为可拆分任务, 生成拆分模式下的执行时间及资源需求; 对大规模算例选择 5 项任务作为可拆分任务. 设定任务拆分的惩罚时间为 $\theta = 1$, 即可拆分任务的后半段执行时间延长一个单位.

3.1 算法参数分析

本文遗传算法设计中交叉操作部分基于传统实数交叉方法进行了改进, 设计了新的基于染色体适应值的交叉方式, 采用参数 α_1, α_2 控制所生成染色体的方向. 选取标准算例库 PSPLIB 中包含 J30、J60 和 J90 共 3 种规模算例的任务对 α_1, α_2 进行敏感性分析, 平均 gap 表示所有规模下不同算例误差百分比的均值. 图 5 为参数 α_1, α_2 的敏感性分析结果, 分别比较取值范围为 $[0, 1]$ 和 $[1, 2]$ 下各算例的计算结果. 遗传算法设定的种群规模 $N=100$.

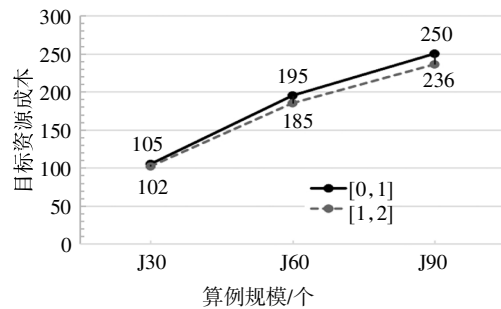


图 5 交叉参数分析

Fig.5 Cross parameter analysis

由图 5 可知, 参数 α_1, α_2 设定在不同区间, 对算例求解的目标值是存在影响的, 随着算例规模的增大, 平均 gap 呈上升趋势. 表明通过 α_2 的取值, 克服了后代个体位置的限制, 使得较弱的个体可能得到更大的改善, 保证了基因多样性, 避免收敛于局部最优. 因此, 当参数 $\alpha_2 \in [1, 2]$ 时可以获得较优的解.

3.2 数值实验分析

3.2.1 算法对比

为了验证本文所提出的改进型遗传算法的有效性, 选取任务数分别为 10、30、60 的各 10 个算例为样本, 与粒子群算法求解的结果进行对比, 求解时间统一设定为 60 s, 结果如图 6 所示, GA 为遗传算法求解结果, PSO 为粒子群算法求解结果. 从图 6 可以看出, 虽然两种算法结果相近, 但是遗传算法结果更优, 说明本文所提出的改进型遗传算法具有优越性.

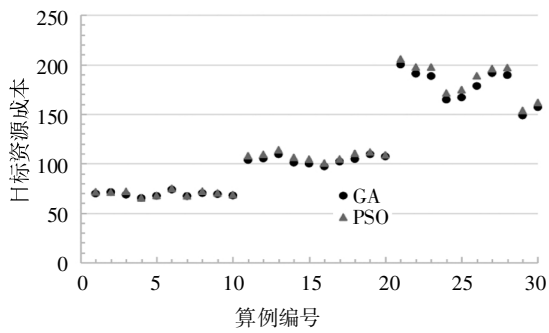


图6 算法对比结果

Fig.6 Algorithm comparison results

3.2.2 数值结果

选取任务数分别为10、16、30、60、90的各50个算例为样本进行数值实验分析,结果如表1~表5所示. GA及GA·NP分别代表每组为5个算例下本文算法和本文算法在考虑任务不可拆分情形下求得的目标值的均值;gap1为这两列值间的差异,用以体现任务拆分的意义. 由于当不考虑空窗期时本文中对于任务拆分的设定条件无任何意义,故GA·N表示本文算法考虑任务不可拆分及资源无空窗期情形下求得的目标值,gap2为GA·NP与GA·N值间的差异,用以体现空窗期对任务调度的影响.v表示每组为5个算例下每种算法求得的目标值的均值,t表示求解时间的均值,单位为s.

$$gap1 = \frac{GA \cdot NP - GA}{GA} \times 100\% \quad (16)$$

$$gap2 = \frac{GA \cdot NP - GA \cdot N}{GA \cdot N} \times 100\% \quad (17)$$

表1 10 jobs 实验结果

Tab.1 Results of 10 jobs

| J10 | GA | | gap1 /% | GA·NP | | gap2 /% | GA·N | |
|------|------|-----|------------|-------|-----|------------|------|-----|
| | v | t/s | | v | t/s | | v | t/s |
| 1 | 65.8 | 5.7 | 2.0 | 67.2 | 3.7 | 3.1 | 65.2 | 4.0 |
| 2 | 67.0 | 4.7 | 1.5 | 68.0 | 3.3 | 5.4 | 64.5 | 3.6 |
| 3 | 64.8 | 6.6 | 4.1 | 67.5 | 4.5 | 5.2 | 64.2 | 3.5 |
| 4 | 62.0 | 6.1 | 0.3 | 62.2 | 4.4 | 3.9 | 59.8 | 4.0 |
| 5 | 63.8 | 6.3 | 0.5 | 64.2 | 3.4 | 2.7 | 62.5 | 3.6 |
| 6 | 69.2 | 7.1 | 1.2 | 70.0 | 3.4 | 2.7 | 68.2 | 3.8 |
| 7 | 63.7 | 6.6 | 0.3 | 63.8 | 5.5 | 6.1 | 60.2 | 5.0 |
| 8 | 66.2 | 6.0 | 2.3 | 67.7 | 5.6 | 4.1 | 65.0 | 5.2 |
| 9 | 65.3 | 6.2 | 1.3 | 66.2 | 6.3 | 5.0 | 63.0 | 5.9 |
| 10 | 64.2 | 5.8 | 1.0 | 64.8 | 5.9 | 5.7 | 61.3 | 4.7 |
| Avg. | 65.2 | 6.1 | 1.5 | 66.2 | 4.6 | 4.4 | 63.4 | 4.3 |

表2 16 jobs 实验结果

Tab.2 Results of 16 jobs

| J16 | GA | | gap1 /% | GA·NP | | gap2 /% | GA·N | |
|------|------|------|------------|-------|------|------------|------|------|
| | v | t/s | | v | t/s | | v | t/s |
| 1 | 85.7 | 15.6 | 4.3 | 89.3 | 12.1 | 4.1 | 85.8 | 11.9 |
| 2 | 77.2 | 17.2 | 4.1 | 80.3 | 13.6 | 3.9 | 77.3 | 10.6 |
| 3 | 84.8 | 16.0 | 4.3 | 88.5 | 10.5 | 4.5 | 84.7 | 10.5 |
| 4 | 83.2 | 16.3 | 3.4 | 86.0 | 12.1 | 3.2 | 83.3 | 11.5 |
| 5 | 81.0 | 14.7 | 2.9 | 83.3 | 11.8 | 2.7 | 81.2 | 11.7 |
| 6 | 83.0 | 15.3 | 2.0 | 84.7 | 11.6 | 2.2 | 82.8 | 11.0 |
| 7 | 81.0 | 15.3 | 3.3 | 83.7 | 11.6 | 3.9 | 80.5 | 11.5 |
| 8 | 83.5 | 19.0 | 2.2 | 85.3 | 12.0 | 2.8 | 83.0 | 11.9 |
| 9 | 80.5 | 15.8 | 2.7 | 82.7 | 11.1 | 3.1 | 80.2 | 12.4 |
| 10 | 76.3 | 16.5 | 2.2 | 78.0 | 11.6 | 4.0 | 75.0 | 11.2 |
| Avg. | 81.6 | 16.2 | 3.1 | 84.2 | 11.8 | 3.4 | 81.4 | 11.4 |

表3 30 jobs 实验结果

Tab.3 Results of 30 jobs

| J30 | GA | | gap1 /% | GA·NP | | gap2 /% | GA·N | |
|------|-------|------|------------|-------|------|------------|-------|------|
| | v | t/s | | v | t/s | | v | t/s |
| 1 | 102.5 | 24.5 | 5.0 | 107.7 | 19.7 | 4.0 | 103.5 | 20.6 |
| 2 | 104.8 | 22.9 | 3.8 | 108.8 | 18.6 | 3.8 | 104.8 | 19.3 |
| 3 | 108.5 | 24.2 | 3.8 | 112.7 | 21.1 | 4.0 | 108.3 | 21.4 |
| 4 | 103.5 | 23.9 | 2.7 | 106.3 | 19.3 | 4.9 | 101.3 | 19.5 |
| 5 | 99.7 | 22.0 | 5.2 | 104.8 | 19.2 | 4.3 | 100.5 | 19.6 |
| 6 | 97.3 | 22.9 | 4.3 | 101.5 | 19.7 | 3.4 | 98.2 | 20.5 |
| 7 | 103.0 | 24.8 | 1.9 | 105.0 | 19.5 | 2.8 | 102.2 | 20.2 |
| 8 | 105.8 | 21.9 | 3.6 | 109.7 | 19.1 | 5.1 | 104.3 | 19.8 |
| 9 | 106.7 | 21.5 | 3.9 | 110.8 | 19.3 | 2.3 | 108.3 | 20.8 |
| 10 | 104.3 | 23.2 | 3.8 | 108.3 | 19.0 | 1.7 | 106.5 | 20.8 |
| Avg. | 103.6 | 23.2 | 3.8 | 107.6 | 19.4 | 3.6 | 103.8 | 20.3 |

表4 60 jobs 实验结果

Tab.4 Results of 60 jobs

| J60 | GA | | gap1 /% | GA·NP | | gap2 /% | GA·N | |
|------|-------|------|------------|-------|------|------------|-------|------|
| | v | t/s | | v | t/s | | v | t/s |
| 1 | 191.8 | 41.3 | 5.3 | 202.0 | 34.4 | 7.6 | 187.7 | 33.8 |
| 2 | 186.5 | 35.4 | 4.6 | 195.2 | 32.5 | 5.0 | 185.8 | 31.8 |
| 3 | 184.5 | 36.0 | 5.8 | 195.2 | 34.5 | 6.0 | 184.2 | 33.3 |
| 4 | 164.7 | 35.8 | 5.3 | 173.3 | 33.4 | 5.9 | 163.7 | 33.9 |
| 5 | 166.5 | 39.7 | 5.8 | 176.2 | 35.0 | 5.6 | 166.8 | 35.7 |
| 6 | 176.2 | 36.2 | 6.6 | 187.8 | 34.4 | 4.7 | 179.3 | 34.6 |
| 7 | 187.0 | 35.4 | 3.7 | 193.8 | 32.6 | 5.8 | 183.2 | 32.7 |
| 8 | 185.3 | 38.5 | 5.0 | 194.7 | 35.5 | 5.4 | 184.7 | 33.7 |
| 9 | 151.3 | 37.3 | 4.8 | 158.7 | 33.7 | 4.0 | 152.5 | 34.4 |
| 10 | 158.3 | 38.7 | 4.4 | 165.3 | 34.0 | 5.4 | 156.8 | 32.6 |
| Avg. | 175.2 | 37.4 | 5.1 | 184.2 | 34.0 | 5.6 | 174.5 | 33.6 |

表 5 90 jobs 实验结果
Tab.5 Results of 90 jobs

| J90 | GA | | gap1 /% | GA·NP | | gap2 /% | GA·N | |
|------|-------|------|------------|-------|------|------------|-------|------|
| | v | t/s | | v | t/s | | v | t/s |
| 1 | 222.2 | 49.1 | 4.7 | 232.7 | 48.4 | 3.6 | 224.5 | 46.4 |
| 2 | 251.5 | 48.3 | 4.1 | 261.8 | 46.1 | 4.0 | 251.8 | 45.1 |
| 3 | 220.0 | 51.1 | 4.9 | 230.8 | 48.2 | 6.4 | 217.0 | 46.1 |
| 4 | 230.2 | 49.2 | 4.5 | 240.5 | 46.9 | 4.0 | 231.3 | 45.4 |
| 5 | 220.7 | 47.8 | 4.9 | 231.5 | 45.9 | 5.1 | 220.2 | 45.6 |
| 6 | 246.5 | 50.3 | 2.7 | 253.2 | 46.1 | 4.8 | 241.5 | 45.8 |
| 7 | 248.5 | 47.6 | 3.4 | 256.8 | 45.7 | 4.3 | 246.3 | 44.9 |
| 8 | 250.3 | 47.7 | 3.9 | 260.0 | 45.9 | 3.5 | 251.2 | 47.0 |
| 9 | 212.8 | 49.3 | 4.0 | 221.3 | 49.2 | 4.2 | 212.3 | 45.1 |
| 10 | 208.0 | 48.6 | 2.9 | 214.0 | 45.6 | 3.8 | 206.2 | 44.7 |
| Avg. | 231.1 | 48.9 | 4.0 | 240.3 | 46.8 | 4.4 | 230.2 | 45.6 |

3.2.3 讨论与分析

从表 6 可以看出, 本文针对求解考虑资源空窗期及任务可拆分条件下的资源投入问题所设计的遗传算法, 能在较短时间内求得较优解. 对比本文算法改进下的求解任务不可拆分问题的结果与基本资源投入问题的结果, 平均增量达 4.3%; 对比本文算法与改进下的求解任务不可拆分问题的结果, 平均优化比率达 3.5%. 结果表明本文提出的改进型遗传算法在求解中具有优越性.

表 6 实验结果汇总

Tab.6 Summary of experimental results

| J | GA | | gap1 /% | GA·NP | | gap2 /% | GA·N | |
|------|-------|------|------------|-------|------|------------|-------|------|
| | v | t/s | | v | t/s | | v | t/s |
| J10 | 65.2 | 6.1 | 1.5 | 66.2 | 4.6 | 4.4 | 63.4 | 4.3 |
| J16 | 81.6 | 16.2 | 3.1 | 84.2 | 11.8 | 3.4 | 81.4 | 11.4 |
| J30 | 103.6 | 23.2 | 3.8 | 107.6 | 19.4 | 3.6 | 103.8 | 20.3 |
| J60 | 175.2 | 37.4 | 5.1 | 184.2 | 34.0 | 5.6 | 174.5 | 33.6 |
| J90 | 231.1 | 48.9 | 4.0 | 240.3 | 46.8 | 4.4 | 230.2 | 45.6 |
| Avg. | 131.3 | 26.4 | 3.5 | 136.5 | 23.3 | 4.3 | 130.7 | 23.0 |

从模型角度来看, 在飞机实际装配过程中, 空窗期约束的存在是客观事实, 同时现实中部分任务可以被拆分执行; 由于文中对任务拆分的模式假定是针对空窗期所设计, 可选模式的执行方案也是根据空窗期进行设置, 故能在一定程度上提高问题调度

的柔性, 从而获得更加贴合实际的调度方案.

3.3 实例应用分析

以某型号支线客机驾驶舱装配工位的调度为例对算法进行验证. 该工位共包含 14 项装配任务, 任务间的时序约束关系、工期及所需资源量如表 7 所示, 任务 1 与任务 14 表示虚任务, 任务 7 及任务 13 为可拆分任务, 存在拆分执行模式. 主要考虑关键人力资源 R_1 、关键设备资源 R_2 、物料配送能力 R_3 和线边空间存储能力 R_4 共 4 类对飞机生产过程影响较大的资源类型, 已知关键人力资源 R_1 存在空窗期 [4, 6].

表 7 驾驶舱装配工位装配计划

Tab.7 Assembly order of cockpit assembly workstation

| 任务 | 执行模式 | 装配工时 | 紧后任务 | 资源用量 |
|----|------|------|----------|----------|
| 2 | 1 | 2 | 6,7 | 1,3,4,8 |
| 3 | 1 | 7 | 8,11,13 | 0,4,3,10 |
| 4 | 1 | 5 | 6,8,9 | 0,5,9,9 |
| 5 | 1 | 3 | 10,11 | 0,3,5,8 |
| 6 | 1 | 4 | 11,13 | 1,4,8,10 |
| 7 | 1 | 3 | 6,9,10 | 0,5,7,7 |
| | 2 | 2 | 6,9,10 | 0,4,6,5 |
| 8 | 1 | 1 | 12 | 1,3,4,10 |
| | 2 | 1 | 6,9,10 | 0,2,6,2 |
| 9 | 1 | 2 | 10,11,12 | 1,5,2,8 |
| 10 | 1 | 2 | 13 | 1,3,10,7 |
| 11 | 1 | 1 | 14 | 0,2,6,7 |
| 12 | 1 | 1 | 14 | 0,4,7,3 |
| 13 | 1 | 2 | 14 | 1,2,5,3 |
| | 2 | 1 | 14 | 1,4,4,2 |
| 14 | 1 | 1 | 14 | 1,3,4,2 |
| | 2 | 1 | 14 | 1,3,4,2 |

分别利用 CPLEX、本文提出的改进遗传算法来求解该实例, 调度方案分别如图 7 和图 8 所示.

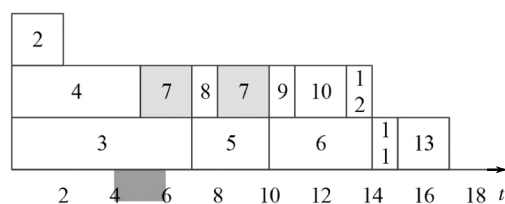


图 7 CPLEX 调度结果

Fig.7 CPLEX scheduling result

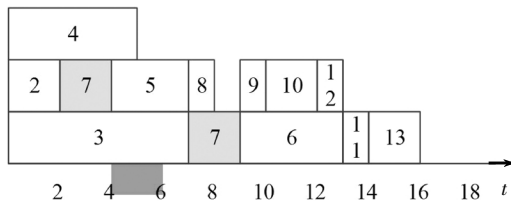


图8 本文算法调度结果

Fig.8 Algorithm scheduling result

图7和图8中,横轴表示时间,由于项目的执行需要4种资源,无法用一张图表明各资源使用量,所以纵坐标不代表任何参量.图示两种方法求得实例的资源成本均为59,说明本文算法在此实例下能够求得最优解.

4 结论

本文以飞机移动式装配线为背景,在研究基本资源投入问题的基础上,同时考虑任务可拆分及资源存在空窗期两大特性,设计了改进型遗传算法进行求解,结果表明:

1)考虑空窗期下的问题调度缺乏灵活性,调度结果所用资源成本相对较高;将任务拆分纳入考虑资源空窗期的资源投入问题中,能通过任务拆分调度的灵活性降低空窗期对问题调度带来的不良影响,利用对任务不同调度位置的决策,合理配置现有资源的使用,从而获得相对较好的调度结果.

2)本文算法在求解效率及求解精度上表现良好,且能用于求解大规模算例.

参考文献

- [1] 王琰, 陆志强. 基于多重约束的飞机移动装配线作业任务调度优化[J]. 工业工程与管理, 2011, 16(6): 115—120.
WANG Y, LU Z Q. Job scheduling optimization of aircraft moving assembly line under multiple constraints[J]. Industrial Engineering and Management, 2011, 16(6): 115—120. (In Chinese)
- [2] 宗保氏, 陆志强. 项目拆分与资源投入调度问题的集成优化[J]. 上海交通大学学报, 2018, 52(7): 793—800.
ZONG B S, LU Z Q. Integrated optimization of project splitting and resource investment project scheduling [J]. Journal of Shanghai Jiaotong University, 2018, 52(7): 793—800. (In Chinese)
- [3] MORHING R H. Minimizing costs of resource requirements in project networks subject to a fixed completion time[J]. Operations Research, 1984, 32(1): 89—120.

- [4] DREXL A, KIMMS A. Optimization guided lower and upper bounds for the resource investment problem[J]. Operational Research Society, 2001, 52(3): 340—351.
- [5] DEMEULEMEESTER E L. Minimizing resource availability costs in time-limited project networks [J]. Operations Research and the Management Science, 1995, 41(10): 1590—1598.
- [6] ZHU X, RUIZ R, LI S Y, et al. An effective heuristic for project scheduling with resource availability cost [J]. European Journal of Operational Research, 2016, 257(3): 746—762.
- [7] SONG Y, LIU J, WIMMERS M O, et al. A differential evolution algorithm with local search for resource investment project scheduling problems [C]//Evolutionary Computation. Sendai: IEEE, 2015: 1725—1731.
- [8] QI J J, LIU Y J, JIANG P, et al. Schedule generation scheme for solving multi-mode resource availability cost problem by modified particle swarm optimization [J]. Journal of Scheduling, 2015, 18(3): 285.
- [9] JAVANMARD S, AFSHAR-NADJAFI B, NIAKI S T A. Preemptive multi-skilled resource investment project scheduling problem: Mathematical modelling and solution approaches [J]. Computers & Chemical Engineering, 2017, 96(4): 55—68.
- [10] 任逸飞, 陆志强. 多技能资源投入项目调度问题的建模与优化[J]. 同济大学学报(自然科学版), 2017, 45(11): 1713—1721.
REN Y F, LU Z Q. Modeling and optimization of resource investment project scheduling problem with multi-skill [J]. Journal of Tongji University (Natural Science), 2017, 45(11): 1713—1721. (In Chinese)
- [11] 廖广瑞, 刘振元, 毕阳. 多技能资源时间窗约束下项目调度研究 [C]//第26届中国控制与决策会议论文集. 长沙: 控制与决策, 2014: 4885—4891.
LIAO G R, LIU Z Y, BI Y. Project Scheduling with Time Window Constraints on Multi-Skill Resources [C]// 26th Chinese Control and Decision Conference (CCDC). Changsha: Control and Decision, 2014: 4885—4891. (In Chinese)
- [12] JIRACHAI B, DAVID S K. Properties of multi-mode resource-constrained project scheduling problems with resource vacations and activity splitting [J]. European Journal of Operational Research, 2006, 175(1): 279—295.
- [13] 蔡方中, 胡丹, 叶雷宏. 基于时间窗和关键链的多项目资源分配问题的研究[J]. 科技管理研究, 2013, 33(13): 229—232.
QI F Z, HU D, YE L H. Study on resource allocation of multi-project based on time window and critical chain[J]. Science and Technology Management Research, 2013, 33(13): 229—232. (In Chinese)
- [14] 陆志强, 石婷. 作业任务可拆分的资源投入问题的建模与优化[J]. 计算机集成制造系统, 2018, 24(3): 602—611.
LU Z Q, SHI T. Modeling and optimization of resource investment problem with activity splitting[J]. Computer Integrated Manufacturing System, 2018, 24(3): 602—611. (In Chinese)
- [15] 陈小平, 于盛林. 实数遗传算法交叉策略的改进[J]. 电子学报, 2003, 31(1): 71—74.
CHEN X P, YU S L. Improvement on crossover strategy of real-valued genetic algorithm [J]. Acta Electronica Sinica, 2003, 31(1): 71—74. (In Chinese)