

求解广义优先关系下多技能人员项目 调度问题的改进布谷鸟搜索算法^{*}

段鹏飞^{1a†}, 余杰², 聂慧^{1b}, 杨辉华^{1a,2}

(1. 桂林电子科技大学 a. 电子工程与自动化学院; b. 信息与通信工程学院, 广西 桂林 541004; 2. 北京邮电大学
自动化学院, 北京 100876)

摘要: 为解决传统的完成—开始时序不能满足描述真实项目调度顺序要求的问题, 引入广义优先关系 (GPRs) 及改进的 AON 描述任务的时序约束。提出将布谷鸟搜索算法应用于求解广义优先关系下的多技能人力资源项目调度问题 (MS-RCPSP/GPRs) 中的构想, 建立了基于改进布谷鸟搜索算法 (ICS) 的求解方法, 采用 Powell 局部改进技术和精英保留策略, 并给出了算法流程。基于相关案例生成器生成该问题的数据集, 实验结果表明 ICS 是一种求解 MS-RCPSP/GPRs 的有效方法, 对解决实际问题具有重要意义。

关键词: 广义优先关系; 多技能人力资源调度问题; 布谷鸟搜索算法; Powell 局部搜索; 回溯操作

中图分类号: TP301.6

文献标志码: A

文章编号: 1001-3695(2018)05-1315-05

doi:10.3969/j.issn.1001-3695.2018.05.008

Improved cuckoo search algorithm for multi-skilled resource constrained project scheduling problems with generalized precedence relations

Duan Pengfei^{1a†}, Yu Jie², Nie Hui^{1b}, Yang Huihua^{1a,2}

(1. a. College of Electronic Engineering & Automation, b. School of Information & Communication Engineering, Guilin University of Electronic Technology, Guilin Guangxi 541004, China; 2. School of Automation, Beijing University of Posts & Telecommunications, Beijing 100876, China)

Abstract: Aiming at the problem that the traditional finish-start precedence relation couldn't satisfy the requirement of describing the real project scheduling precedence relations, this paper introduced the generalized precedence relations (GPRs) and the improved AON to describe task timing constraints. It applied cuckoo search algorithm to solve the multi-skilled resource constrained project scheduling problem with generalized precedence relations (MS-RCPSP/GPRs), designed and introduced an methodology based on the improved cuckoo search (ICS) algorithm to optimize those problems, used Powell technology and improved local elite retention strategy, and gave the algorithm process. The relevant case generator generated the data set of the problem. The experimental results show that ICS is an effective method to solve MS-RCPSP/GPRs, which is of great significance to solve practical problems.

Key words: generalized precedence relations; multi-skilled RCPSP; cuckoo search algorithm; Powell local search; backtracking

0 引言

随着经济的不断发展, 人力资源管理工作在企业、社团、组织中的地位日益上升。多技能人力资源管理也是项目管理领域中的热点, 因此引出了资源受限项目调度问题中 (RCPSP) 的多技能人力资源调度问题 (MS-RCPSP), 而广为研究的多技能人力资源受限项目调度问题, 任务间只存在最小时间间隔 (time-lag) 为零的 FS (finish-start) 关系。这种传统的 FS 紧前约束, 比较简单实用, 但不能满足描述真实项目调度顺序的要求, 从而引入带最小最大时间间隔 (minimum and maximum time lags) 的 FS (完成—开始)、SS (开始—开始)、SF (开始—完成)、FF (完成—完成) 时序约束来描述更加复杂的项目调度问题, 它们被称为广义优先关系 (generalized precedence relations,

GPRs)^[1]。

国外的相关研究有, Bartusch 等人^[1]通过对任一对任务的开始时间和结束时间之间引入任意的最小和最大时间间隔, 形成带时间窗的项目网络, 它是广为人知的 CPM (critical path methods) 网络的更为一般的形式。Dhib 等人^[2]研究了资源具有一种或多种技能、但任务对技能水平有下限要求的多技能人力资源调度问题, 并给出了两个下界。Montoya 等人^[3]提出了一种包含列生成方法并且考虑了特定任务和时间分解法的分支定界法用来求解 MS-RCPSP 问题。实验表明该方法能够在中小型问题实例中在可接受的时间内求得问题的最优解。Myszkowski 等人^[4]提出了一种求解多技能的资源受限项目调度问题的混合蚁群算法。结果表明, 融合定向搜索过程的蚁群算法是一种有效的方法, 相比经典算法能够获得更好的结果。

收稿日期: 2016-12-28; **修回日期:** 2017-03-01 **基金项目:** 国家自然科学基金资助项目 (21365008, 61562013); 广西重点研发计划资助项目 (桂科 AB16380293)

作者简介: 段鹏飞 (1991-), 男 (通信作者), 硕士研究生, 主要研究方向为最优化方法 (2460927670@qq.com); 余杰 (1993-), 男, 硕士研究生, 主要研究方向为最优化方法; 聂慧 (1972-), 女, 副研究员, 主要研究方向为人力资源管理; 杨辉华 (1972-), 男, 湖南常德人, 教授, 博导, 博士, 主要研究方向为机器学习。

国内也有相关研究,如郭研等人^[5]以项目群的总工期及总费用最小为目标,建立了多技能员工受限的多项目调度模型,进而提出一种新的云多目标微粒群算法求解方法。Liao等人^[6]针对MS-RCPSP问题,建立了一种数学模型,设计了一种基于优先规则的Rollout求解算法,算法中嵌入了一种启发式资源分配方法,实验结果表明该算法针对大规模问题能够求得满意解。王一帆等人^[7]提出了一种求解多技能人力资源约束的项目调度问题的两阶段算法,效果比一般启发式算法更好,但是大部分研究都只是针对传统FS紧前约束关系下的多技能人力资源项目调度,对于(MS-RCPSP/GPRs)问题还鲜有研究。

本文针对广义优先关系下对于多技能人力资源和项目任务分配调度问题,设计了一种基于改进布谷鸟搜索算法的求解模型。采用Wang等人^[8]提出的案例生成器得到问题的基本案例,对不同规模的案例进行求解,并与遗传算法、粒子群算法和CPLEX软件的求解结果进行对比。

1 问题描述及模型建立

1.1 广义优先关系转换

对于广义优先关系,通过Bartusch等人^[1]于1988年提出的一种转换方法,可把带时间间隔的所有GPRs转换成“开始—开始”的最小时间间隔关系。Bartusch证明可以实现上述四种关系的相互转换,但为方便起见,一般是全部转换为SS型优先关系。以FS型转换SS型为例:

$$f_i + l_{ij}^{\min} \leq s_j \rightarrow s_i + l_{ij} \leq s_j \quad (1)$$

式(1)的转换过程可以由 $l_{ij} = d_i + l_{ij}^{\min}$, $f_i = s_i + d_i$ 得出。 $f_i = s_i + d_i$ 表示任务 i 的开始时间加上工期等于其完成时间; d_i 表示任务 i 的工期; $f_i + l_{ij}^{\min} \leq s_j$ 表示任务 j 在任务 i 完成的最小时间间隔 l_{ij}^{\min} 后才能开始。

$$f_i + l_{ij}^{\max} \leq s_j \rightarrow s_j + l_{ij} \leq s_i \quad (2)$$

式(2)的转换过程同上。其中: $l_{ij} = -d_i - l_{ij}^{\max}$, $f_i + l_{ij}^{\max} \leq s_j$ 表示任务 j 在任务 i 完成的最大时间间隔 l_{ij}^{\max} 内必须开始。

为简洁和统一地描述GPRs, Bartusch等人进一步将上述GPRs统一表示为

$$s_j \geq s_i + l_{ij} \quad (3)$$

其中:对任意任务 i ,开始时间 $s_i \geq 0$ 。为了描述的简洁性,本文采用式(3)统一描述GPRs的时序约束。

为统一、直观地描述任务间的各种广义优先关系,对AON网络改进如下:对于具有GPRs的任务 i 和任务 j ,若有最小时间间隔 l_{ij}^{\min} ,则用AON网络上一条正向的带权 l_{ij} 的弧表示;若有最大时间间隔 l_{ij}^{\max} ,则用AON网络上一条反向的带权 l_{ij} 的弧将其转换成最小时间间隔(图1中虚线)。广义优先关系下的AON网络实例如图1所示。

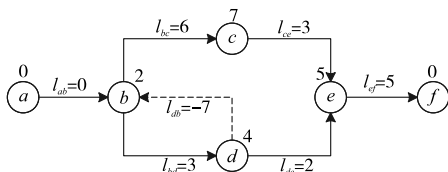


图1 广义优先关系下的AON网络实例

1.2 问题描述及模型

本文研究的是非抢占式的广义优先关系下的多技能资源受限的项目调度问题,其优化目标是求得使项目工期最短的任务调度序列和人力资源分配策略。

问题的数学模型描述如下:一个项目由 n 个任务的集合 $P = \{p_1, p_2, \dots, p_n\}$ 组成, g 种可更新资源组成集合 $R = \{r_1, r_2, \dots, r_g\}$;每种资源可利用资源数量为常量 R_s ;任务 p 的执行时间为 d_p 。任务在执行过程中不能中断,即某个任务开始则必须完成该任务。在任务执行的每个阶段 t ,任务 p 对资源 s 的需求 $r_{ps} \leq R_s$ 。由于部分任务间存在一定的优先关系,所以有 $S_j \geq S_i + l_{ij}$ 。

目标函数是在满足广义优先关系和资源受限双重条件下,求解工程的最短工期以及每个任务的开始时间。一个项目调度方案可以表示为 $B = \{B_1, \dots, B_n\}$, B_p 表示任务 p 的开始时间。

本文提出一种混合整数线性规划(MILP)模型来描述该问题。具体数学模型如下:

$$\min \max \{B_1 + d_1, B_2 + d_2, \dots, B_n + d_n\} \quad (4)$$

$$\text{s. t. } \sum_{t=1}^T Z_{pt} = 1, \forall p \in P \quad (5)$$

$$S_j \geq S_i + l_{ij} (\langle i, j \rangle \in E) \quad (6)$$

$$\sum_{p=1}^n \sum_{s=1}^g X_{ptsk} \leq 1, \forall t = 1, 2, \dots, T, k \in R \quad (7)$$

$$B_p = \sum_{t=1}^T t \cdot Z_{pt}, \forall p \in P \quad (8)$$

$$r_{psq} \leq \sum_{k=1}^m \text{RCM}_{ks} - U_s \quad (9)$$

$$p \in P, s \in S, k \in R, q \in \{B_p, \dots, B_p + d_p\} \quad (10)$$

$$X_{ptsk} \in \{0, 1\}, \forall p \in P, \forall t = 1, 2, \dots, T, \forall s \in S, \forall k \in R \quad (11)$$

$$Z_{pt} \in \{0, 1\}, \forall p \in P, \forall t = 1, 2, \dots, T \quad (12)$$

各个符号的说明如下:

P 表示项目中任务 p 的集合, $p = \{1, 2, \dots, n\}$;

S 表示项目所需技能 s 的集合, $s = \{1, 2, \dots, g\}$;

R 表示项目所有工人 k 的集合, $k = \{1, 2, \dots, m\}$;

R_s 表示具有技能 s 的人力资源集合, $s \in S$;

r_{pts} 表示任务 p 在 t 时间段对技能 s 的资源需求量, $t = 1, 2, \dots, T, s \in S, k \in R$;

d_p 表示执行任务 p 的持续时间;

B_p 表示任务 p 的开始时间;

F_p 表示任务 p 的紧前任务集合;

T 表示完成项目的最长可能工期, $T = \sum_{n=1}^n d_p$;

RCM_{ks} 表示第 k 个人是否具有技能 s ,如果具有技能 s , $\text{RCM}_{ks} = 1$,否则 $\text{RCM}_{ks} = 0$;

若 $0 < \text{RCM}_{ks} < 1$,则表示人力资源 k 具备多种技能;

U_s 表示已经占用技能 s 的资源集合, $s \in S$;

X_{ptsk} 表示若第 k 个人使用技能 s 在 t 时间段内执行任务 p ,则 $X_{ptsk} = 1$,否则 $X_{ptsk} = 0$;

Z_{pt} 表示若任务 p 在 t 时间段开始执行,则 $Z_{pt} = 1$,否则 $Z_{pt} = 0$ 。

式(4)表示求项目最短工期;式(5)表示在整个工期内每个任务只能开始一次;式(6)表示两个任务之间的广义优先关系;式(7)表示每个工人在每个时间段只能使用一种技能执行一个任务;式(8)表示任务 p 开始时间的公式;式(9)表示任务 p 对技能为 s 的人力资源的请求不大于可利用技能为 s 的人力资源数;式(10)~(12)为决策约束变量。

2 模型求解算法设计

2.1 布谷鸟搜索算法

布谷鸟搜索算法(cuckoo search, CS)^[9]也叫做杜鹃搜索,是由剑桥大学Yang教授和Deb于2009年提出的一种新兴启发算法。CS算法通过模拟某些种属布谷鸟的寄生育雏来有效地求解最优化问题的算法。同时,CS算法也采用了相关的

Levy 飞行搜索机制。

在自然界中,布谷鸟寻找适合自己产蛋的鸟巢位置是随机的或是类似随机的方式。为模拟布谷鸟寻巢,假设了以下三个理想规则^[10]:

a) 布谷鸟随机选择一个鸟巢宿主来孵化蛋,且每次只产一个鸟蛋;

b) 最好的鸟巢将会被保留到下一代;

c) 可用鸟巢数量 n 是固定的,一个鸟巢的宿主能发现一个外来鸟蛋的概率为 $P_a \in [0, 1]$ 。

在这三个理想规则的基础上,布谷鸟寻找鸟巢的路径和位置更新公式如下:

$$x_i^{(t+1)} = x_i^{(t)} + \alpha \oplus \text{Levy}(\lambda) \quad i = 1, 2, \dots, n \quad (13)$$

其中: $x_i^{(t)}$ 表示第 i 个鸟巢在第 t 代的鸟巢位置; \oplus 为点对点乘法; α ($\alpha > 0$) 为步长控制量,一般取 $\alpha = 1$; $\text{Levy}(\lambda)$ 表示莱维搜索路径,随机步长满足 Levy 分布:

$$\text{Levy}(s, \lambda) \sim s^{-\lambda}, 1 < \lambda \leq 3 \quad (14)$$

此处 s 代表由莱维飞行得到的随机步长。通过位置更新后,用随机数 $r \in [0, 1]$ 与 P_a 比较,若 $r > P_a$ 则对鸟巢的位置 $x_i^{(t+1)}$ 进行随机改变,反之不变。最后保留适应度值较好的一组鸟巢位置。CS 算法伪代码如下:

算法 1 布谷鸟搜索算法^[9]

```

objective function  $f(X)$ ,  $X = (x_1, x_2, \dots, x_d)^T$ 
initial population of  $n$  host nests  $X_i (i = 1, 2, \dots, n)$ 
while  $t < \text{max generation or stop criterion}$ 
    get a cuckoo randomly by Levy flights
    evaluate fitness  $F_i$ 
    choose a nest  $j$  among  $n$  randomly
    if  $F_i > F_j$ 
        replace  $j$  by the new solution
    end if
    a fraction  $P_a \in [0, 1]$  of worse nests are abandoned and new ones are built
    keep the best solutions
    rank the solutions and find the current best
end while
post process results and visualization
 $t$  represents the number of iterations

```

2.2 Powell 局部搜索

Powell 搜索算法也称做方向加速法,是利用共轭方向可以加快收敛速度的性质形成的一种搜索方法。该方法不需要对目标函数进行求导,当目标函数的导数不连续的时候也能应用。Powell 算法有一套完整的理论体系,其计算效率高与其他直接法,且该方法使用一维搜索,而不是跳跃的探测步,因此, Powell 算法是一种十分有效的直接搜索法。

算法计算步骤如下:

a) 选取初始数据。

选取初始点 x_0 , n 个线性无关的搜索方向 d_0, d_1, \dots, d_{n-1} , 给定允许误差 Err , 令 $k > 0$ 。

b) 进行基本搜索。

令 $y_0 = x_1$, 依次沿 d_0, d_1, \dots, d_{n-1} 进行一维搜索。对一切 $j = 1, 2, \dots, n$, 记

$$f(y_{j-1} + \lambda_{j-1} d_{j-1}) = \min f(y_{j-1} + \lambda d_{j-1}) \quad (15)$$

其中: $y_j = y_{j-1} + \lambda_{j-1} d_{j-1}$ (16)

c) 进行加速搜索。

取加速方向 $d_n = y_n - y_0$; 若 $\|d_n\| < \text{Err}$, 迭代终止, 得到 y_n 为问题的近似最优解; 否则, 从点 y_n 出发沿 d_n 进行一维搜索, 求出 λ_n , 使得

$$f(y_n + \lambda_n d_n) = \min f(y_n + \lambda d_n) \quad (17)$$

记 $x_{k+1} = y_n + \lambda_n d_n$, 转 d)。

d) 调整搜索方向。

在原来 n 个方向 d_0, d_1, \dots, d_{n-1} 中, 除去 d_0 增添 d_n , 构成新的搜索方向, 返回 b)。

2.3 编码方案

针对广义优先关系下多技能人力资源受限项目调度问题的特性, 提出基于任务优先级的编码方案。采用实数对任务执行顺序优先级进行编码, 项目由 n 个任务的集合 $P = [p_1, p_2, \dots, p_n]$ 组成, 其中 p_i 表示任务 i 的优先级值, 设定 p_i 的上下界为 $[-5, 5]$ 。

调度任务的选择取决于任务间的时序优先关系约束和资源约束, 在就绪任务集中根据任务的优先权系数选择将要处理的任務。本文引入的优先权系数说明如下:

overlap 表示任务之间在时间上的重叠系数;

jobHardness 表示任务获得所需资源的难易程度系数;

skillHardness 表示任务所用技能获得所需人员的难易程度系数;

conflict 表示任务所需资源间的冲突系数;

jobstime 表示任务调度的开始时间。

对于上述任务优先权系数的计算, 主要采用基于任务序列图的优先规则(最短工期优先规则、最多紧后任务优先规则等)、基于关键路径的优先规则(最早开始时间优先规则、最小总时差优先规则等)、基于资源的优先规则(最大资源利用优先规则、最大资源需求优先规则等)。

2.4 回溯操作

在广义优先关系网络中会存在环形结构, 因此在任务执行过程中会出现两种特殊情况, 分别是 $ES_k > LS_k$ 和满足资源约束的任务开始时间 $t^* > LS_k$ 。遇到这两种情况需要进行如下处理。

1) $ES_k > LS_k$ 假设任务 k 存在多个最大时距约束, 可能存在 $ES_k > LS_k$, 如图 2 所示。其中, $S_i = S_j = 1$, 因此, 由任务 i 确定 k 的开始时间限制为 $S_k^i \in [4, 6]$, 由任务 j 确定 k 的开始时间限制为 $S_k^j \in [2, 3]$, 则 $S_k^i \cap S_k^j = \emptyset$, 即 $ES_k > LS_k$ 。

出现此种情况, 需要找出限制其最晚开始时间变小的任务, 然后在满足其时间窗约束和资源约束情况下, 增加其开始时间, 如上例, 增加 S_j 的开始时间, 即 $S_j + 1$, 则 $S_k^i \cap S_k^j = 4$ 。

2) $t^* > LS_k$ 假设任务 k 存在多个最大时距约束, 且 $ES_k \leq LS_k$, 在求任务满足资源约束条件的最早开始时间时, 可能存在 $t^* > LS_k$, 如图 3 所示。其中, $S_i = S_j = 1$, 因此, 由任务 i 确定 k 的开始时间限制为 $S_k^i \in [4, 6]$, 由任务 j 确定 k 的开始时间限制为 $S_k^j \in [3, 5]$, 则 $S_k^i \cap S_k^j = [4, 5]$, 然而满足资源情况下的最早开始时间为 7。为解决此种情况, 需要找出限制其最晚开始时间变小的任务, 然后在满足其时间窗约束和资源约束的情况下, 增加其开始时间, 大小为 $t^* - LS_k$ 。如果超出其时间窗约束, 需要递归(嵌入)调用该方法, 直至初始任务节点, 则判断任务不可行。

2.5 模型求解

针对非抢占式的广义优先关系下的多技能资源受限的调度问题, 将问题的求解过程分为两个步骤^[11]。首先求解任务的调度序列, 再求解该调度序列如何进行技能工人的分配。其主要算法流程如算法 2 所示。

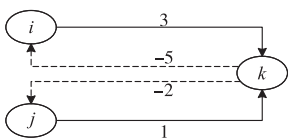


图2 $ES_k > LS_k$ 示例图

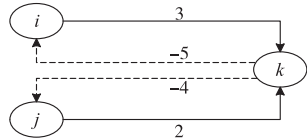


图3 $t^* > LS_k$ 示例图

算法 2 模型求解算法

- a) 对于给定的案例,根据 benders 分解法^[11]将问题求解过程分为两步,首先进行项目任务的调度,再进行任务中所需技能人员的分配;
- b) 采用布谷鸟搜索算法产生初始种群及初始鸟巢,一个布谷鸟找到的一组鸟巢代表一个调度序列;
- c) 评估每组鸟巢的适应度值,若该广义优先关系网络中存在环形结构,则进行回溯操作;
- d) 对技能人员的分配引入 hardness、overlap 等系数作为优先级进行调度分配。
- e) 判断该任务所需技能人员是否分配成功,并保留当前较优鸟巢;
- f) 在当前较优鸟巢附近采用 Powell 局部搜索算法获得最新鸟巢,判断任务所需技能人员是否成功分配,并更新当前较优鸟巢;
- g) 根据 Levy 飞行公式更新鸟巢,并保留当前最好的鸟巢;
- h) 当未达到最大迭代次数或不满足停止条件则算法继续;
- i) 满足最大迭代次数或停止条件,保存当前最好的鸟巢;
- j) 输出最优调度分配结果。

3 实验设计与结果

本文的数据集基于 Wang 等人^[8]设计的 MS-RCPSP/GPRs 案例生成器生成,为了测试算法的求解效果,对数据集中的案例采用不同算法进行对比实验,并分析求解效果。本文所提算法的仿真实验在 MATLAB R2013a 上进行,部分对比实验采用 IBM 大型约束规划软件 CPLEX 进行求解。

3.1 实验案例及算法参数

MS-RCPSP/GPRs 实验案例按照工作规模分为 Job10、Job20、Job30 共三个案例集合,各案例集合中均包含有 270 个案例。

种群规模 N_p 、发现概率 P_a 是布谷鸟搜索算法的两个重要参数,针对不同规模的问题,算法参数设置不尽相同。为了对比不同算法的求解性能,本文考虑固定启发式算法的初始参数设置如表 1 所示。

表 1 算法参数设置

种群规模 N_p	迭代次数 iter	发现概率 P_a	步长 α
200	50	0.25	0.1

3.2 结果与分析

对于本文所用数据集中的案例,首先采用 CPLEX 软件直接进行求解,并根据 CPLEX 的求解结果把案例分为 OPT、FEA、UNK 三种不同子集,分别表示在 CPU 时间限定下,CPLEX 可以求得最优解、可行解和结果未知的案例。采用 CPLEX 求解 Job10、Job20 和 Job30 数据集的时间上限分别为 120 s、240 s、480 s。

对以上数据集分别采用 CS、GA、PSO 以及 ICS 四种启发式算法进行求解并与 CPLEX 的求解效果进行对比。

评价指标说明如下:

- a) CPU(s) 表示算法对单个案例的平均求解时间。
- b) SolveRate 表示算法对案例的求解率。
- c) Dev 表示算法结果与案例中项目网络图的关键路径长度(CPL)之间相对差值的百分比。

结果如表 2~7 所示。

表 2 Job10 案例求解情况

案例子集	算法	CPU/s	SolveRate	dev
OPT	CS	3.11	1	0.264 7
	GA	3.83	1	0.273 9
	PSO	3.52	1	0.264 7
	CPLEX	2.47	1	0.264 7
FEA	CS	101.7	1	0.271 4
	GA	170.6	1	0.285 6
	PSO	162.8	0.994 6	0.283 9
	CPLEX	120.2	1	0.272 8
UNK	CS	0	0	0
	GA	0	0	0
	PSO	0	0	0
	CPLEX	0	0	0

表 3 Job20 案例求解情况

案例子集	算法	CPU/s	SolveRate	dev
OPT	CS	4.65	1	0.026 1
	GA	5.17	1	0.032 2
	PSO	5.87	0.996	0.03
	CPLEX	5.03	1	0.026 7
FEA	CS	212.3	1	0.161 2
	GA	260.8	1	0.199 8
	PSO	249.1	0.997 2	0.201 2
	CPLEX	240.9	1	0.188 4
UNK	CS	193.3	0.95	0.442 6
	GA	254.1	0.933 3	0.493
	PSO	278.9	0.942 5	0.467 3
	CPLEX	240	0	-1

表 4 Job30 案例求解情况

案例子集	算法	CPU/s	SolveRate	dev
OPT	CS	5.88	1	0.003 1
	GA	6.26	1	0.005 7
	PSO	7.2	0.997 2	0.008 4
	CPLEX	6.33	1	0.004 2
FEA	CS	220.9	1	0.017 7
	GA	228.5	0.944 4	0.020 2
	PSO	245.4	0.964 5	0.021 3
	CPLEX	480	1	0.159 8
UNK	CS	274.8	0.965 5	0.216 8
	GA	289.1	0.937 9	0.219 4
	PSO	279.6	0.936 5	0.220 4
	CPLEX	480	0	-1

表 2 给出了 Job10 案例的求解情况,因为 Job10 任务规模较小,没有不可知案例。对于其中的 OPT 案例,所有算法均能在有效时间求得最优解,在时间上 CPLEX 比 CS 算法略胜一筹;而对于 FEA 案例,CS 算法与其他算法在 dev 上差别并不大。

从表 3、4 可以看出,当工作规模逐步增大后,CPLEX 求解效果在不断下降。由表 3 可以得出,对于求解 OPT 案例集,在时间上 CS 相比 GA 算法平均每个案例缩短 0.52 s,dev 值也降低了 0.006 1;而对于 FEA 案例集,CS 算法相对于其他启发式算法的优势则更加明显。由表 4 可以看出,对于 UNK 案例集

CPLEX 难以在一定时间内求得较优解,而 CS 算法能够快速求得较优解且求解效果优于其他启发式搜索算法。

从表 5~7 可以看出,ICS 在不同规模的案例集中,对于 OPT 部分案例的求解在 CPU 和 dev 上都比原 CS 算法更优。在 Job30 的 OPT 数据集中,ICS 算法在运行时间上平均每案例减少 0.32 s,dev 值降低 0.000 5。对于 FEA、UNK 部分案例,虽然在求解时间上略有牺牲,但是 ICS 算法在 dev 值方面分别降低了 0.005 2 和 0.031 4,且在 SolveRate 求解率方面提高了 0.007 8,因此可以得出 ICS 算法在总体的求解性能和结果上优势明显。

表5 Job10 案例 CS 与 ICS 求解结果

案例子集	算法	CPU/s	SolveRate	dev
OPT	CS	3.11	1	0.264 7
	ICS	2.93	1	0.264 7
FEA	CS	101.7	1	0.271 4
	ICS	110.9	1	0.219 8
UNK	CS	0	0	0
	ICS	0	0	0

表6 Job20 案例 CS 与 ICS 求解结果

案例子集	算法	CPU/s	SolveRate	dev
OPT	CS	4.65	1	0.026 1
	ICS	4.32	1	0.021 5
FEA	CS	212.3	1	0.161 2
	ICS	218.9	1	0.142 7
UNK	CS	193.3	0.95	0.442 6
	ICS	196.5	0.966 7	0.398 1

表7 Job30 案例 CS 与 ICS 求解结果

案例子集	算法	CPU/s	SolveRate	dev
OPT	CS	5.88	1	0.003 1
	ICS	5.56	1	0.002 6
FEA	CS	220.9	1	0.017 7
	ICS	230.4	1	0.012 5
UNK	CS	274.8	0.965 5	0.216 8
	ICS	281.2	0.973 3	0.185 4

4 结束语

本文针对 MS-RCPSP/GPRs 问题的特点,采用文献[1]中所提方法对问题进行了转换,并提出了采用布谷鸟搜索算法求

解该类问题;同时为克服 CS 算法后期收敛慢的缺点引入 Powell 局部搜索策略。实验结果证明本文提出的 ICS 算法是一种求解 MS-RCPSP/GPRs 问题的有效方法,对于解决实际问题具有重要意义。

下阶段研究工作考虑从以下两方面入手:a)引入技能员工的薪资、技能熟练程度等因素,进一步对模型进行完善;b)实际项目中,单目标优化并不能完全满足企业需求,因此研究工期和成本等的多目标优化问题具有一定理论和实际价值。

参考文献:

- [1] Bartusch M, Mhring R H, Rademacher F J. Scheduling project networks with resource constraints and time windows[J]. *Annals of Operational Research*, 1988, 16(1): 201-240.
- [2] Dhib C, Kooli A, Soukha L A, et al. Lower bounds for the multi-skill project scheduling problem[C]//Proc of the 8th International Workshop on Project Management and Scheduling. Berlin: Springer, 2012: 471-476.
- [3] Montoya C, Bellenguez-Morineau O, Pinson E, et al. Branch-and-price approach for the multi-skill project scheduling problem[J]. *Optimization Letters*, 2014, 8(5): 1721-1734.
- [4] Myszkowski P B, Skowronski M E, et al. Hybrid ant colony optimization in solving multi-skill resource-constrained project scheduling problem[J]. *Soft Computing*, 2015, 19(12): 3599-3619.
- [5] 郭研, 李南, 李兴森. 基于云多目标微粒群算法的多项目调度方法研究[J]. *计算机工程与应用*, 2012, 48(21): 15-20.
- [6] Liao Guangrui, Liu Zhenyuan, Bi Yang. Project scheduling with time window constraints on multi-skill resources[C]//Proc of the 26th Chinese Control and Decision Conference. [S. l.]: IEEE Press, 2014: 4885-4891.
- [7] 王一帆, 刘士新, 陈迪. 求解多技能人力资源约束的项目调度问题的两阶段算法[J]. *东北大学学报: 自然科学版*, 2014, 35(2): 184-189.
- [8] Wang Yifan, Chen Di, Liu Shixin, et al. An instance generator for project scheduling problems with multi-skilled personnel constraints[C]//Proc of the 24th Chinese Control and Decision Conference. [S. l.]: IEEE Press, 2012: 3430-3435.
- [9] Yang Xinshe, Deb S. Cuckoo search via Lévy flights[C]//Proc of World Congress on Nature & Biologically Inspired Computing. Piscataway, NJ: IEEE Press, 2009: 210-214.
- [10] Ballestín F, Barrios A, Valls V. An evolutionary algorithm for the resource-constrained project scheduling problem with minimum and maximum time lags[J]. *Journal of Scheduling*, 2011, 14(4): 391-406.
- [11] Li Haitao, Womer K. Scheduling projects with multi-skilled personnel by a hybrid MILP/CP benders decomposition algorithm[J]. *Journal of Scheduling*, 2009, 12(3): 281-298.
- [12] Suh B, Hong L, Piroli P, et al. Want to be retweeted? Large scale analytics on factors impacting retweet in Twitter network[C]//Proc of IEEE International Conference on Social Computing. Piscataway, NJ: IEEE Press, 2010: 177-184.
- [13] Huang Dongxu, Zhou Jing, Yang Feisheng, et al. Understanding retweeting behaviors in Twitter[J]. *Journal of Computational Information Systems*, 2015, 11(13): 4625-4634.
- [14] Zhao Hongyong, Zhu Linhe. Dynamic analysis of a reaction-diffusion rumor propagation model[J]. *International Journal of Bifurcation & Chaos*, 2016, 26(6): 1650101.
- [15] 顾亦然, 夏玲玲. 在线社交网络中谣言的传播与抑制[J]. *物理学报*, 2012, 61(23): 238701.
- [16] Liu Xiongding, Li Tao, Wang Yuanmei, et al. Spreading dynamics of a social information model with overlapping community structures on complex networks[J]. *Open Access Library Journal*, 2016, 3(5): 1-11.
- [17] Liu Hongli, Xie Yun, Hu Haibo, et al. Affinity based information diffusion model in social networks[J]. *International Journal of Modern Physics C*, 2014, 25(5): 1440004.
- [18] 王金龙, 刘方爱, 朱振方. 一种基于用户相对权重的在线社交网络信息传播模型[J]. *物理学报*, 2015, 64(5): 050501.
- [19] Sasaki W, Furukawa Y, Nishiyama Y, et al. SmileWave: sensing and analysis of smile-based emotional contagion over social network: poster abstract[C]//Proc of International Conference on Information Processing in Sensor Networks. Piscataway, NJ: IEEE Press, 2016: 64-65.
- [20] Ren Donghao, Zhang Xin, Wang Zhenhuang, et al. Weibo events: a crowd sourcing weibo visual analytic system[C]//Proc of Pacific Visualization Symposium. Piscataway, NJ: IEEE Press, 2014: 330-334.
- [21] 费斯汀格. 认知失调理论[M]. 郑全全, 译. 浙江: 浙江教育出版社, 1999.
- [22] CML. UCI microblogPCU 数据库[EB/OL]. [2016-08-21]. <https://archive.ics.uci.edu/ml/datasets/microblogPCU>.

(上接第 1303 页)