

# 基于不同支付条件的现金流均衡项目调度优化<sup>①</sup>

何正文, 刘人境, 徐 渝

(西安交通大学管理学院, 西安 710049)

**摘要:** 以现金流均衡为目标, 研究不同支付条件下的项目调度问题. 在对所研究问题进行界定的基础上, 构建了基于里程碑事件支付条件的优化模型, 并将其扩展为基于累计时间、累计净值和累计费用等支付条件的优化模型; 针对问题的强 NP-hard 属性, 设计禁忌搜索启发式算法; 在随机生成的标准算例集合上对算法进行测试, 并与其他两种启发式算法进行比较; 最后用一个算例对研究进行说明, 比较分析支付条件对承包商现金流均衡目标的影响. 研究结论如下: 不同支付条件下的现金流均衡结果差异显著, 随着支付次数、支付比例和项目截止日期的增大, 承包商更易实现现金流出与流入的均衡.

**关键词:** 项目调度; 现金流均衡; 支付条件; 优化模型; 禁忌搜索

**中图分类号:** C935; F224.33 **文献标识码:** A **文章编号:** 1007-9807(2011)08-0075-11

## 0 引言

在实际项目管理中, 现金流出与流入的均衡不仅是管理者必须考虑的重要问题之一, 而且在某些特定的情形下, 例如, 项目的实施需要投入大量资金而承包商自身的融资能力又比较有限, 会成为制约项目顺利实施的最关键因素<sup>[1]</sup>. 因此, 在项目调度中, 如何实现现金流的均衡是具有较强现实意义的研究问题. 然而, 对现有项目调度问题的研究现状进行综述<sup>[2-4]</sup>, 可以发现绝大多数研究均以工期最小化 (min-makespan)<sup>[5]</sup>、净现值最大化 (max-npv)<sup>[6]</sup>、费用最小化 (min-cost)<sup>[7]</sup>、资源平衡 (resource leveling)<sup>[8]</sup> 或时间/费用权衡 (time-cost tradeoff)<sup>[9-10]</sup> 等为目标, 很少考虑现金流能否实现均衡. 尽管以资源平衡为目标的项目调度问题<sup>[8]</sup> 研究了如何在项目进行过程中保持资源的平衡, 但其关注的只是资源投入在时间上的平衡, 并未将承包商从业主那里获得的支付纳入统筹范畴内, 所以, 该分支的研究成果并不适用

于上述现金流均衡问题的解决.

鉴于上述实际需求和理论现状, 本文研究现金流均衡项目调度问题 (cash flow balance project scheduling problem, CFBPSP). 在该问题中, 承包商可用资金量随着业主的支付而变化, 同时业主对支付的安排又取决于承包商完成项目的进度, 承包商的项目调度目标是力图实现现金流出与流入的均衡, 以避免在项目实施过程中因可用资金不足而导致项目停滞. 由于现金流入与合同的支付条件密切相关, 因此, 本文将在不同的支付条件下展开对上述问题的研究, 以期为不同情形下承包商的现金流均衡和项目调度提供决策支持.

## 1 问题界定

本文采用基于事件 (event-based) 的研究方法<sup>[11]</sup>, 即项目网络采用 AoA (Activity-on-Arce) 方式表述, 现金流出和流入均与项目的事件相联系. 假定某项目具有  $N$  个活动、 $M$  个事件, 活动  $n$  ( $n =$

① 收稿日期: 2009-06-15; 修订日期: 2010-06-11.

基金项目: 国家自然科学基金资助项目 (70971105); 国家社会科学基金资助项目 (07XJL002); 陕西省自然科学基金资助项目 (2009JM9001); 陕西省科技厅科学计划资助项目 (2008RK11).

作者简介: 何正文 (1967—), 男, 山西运城人, 博士, 副教授. E-mail: zhengwenhe@mail.xjtu.edu.cn. <http://www.cnki.net>

1, 2, …, N) 具有  $Q_n$  种执行模式, 以模式  $q$  ( $q = 1, 2, \dots, Q_n$ ) 执行时的工期和费用分别为  $d_{nq}$  和  $c_{nq}$ , 实现事件  $m$  ( $m = 1, 2, \dots, M$ ) 所需费用为  $e_m$ , 其表达式为

$$e_m = \zeta \sum_{n \in S_{m1}} c_{nq} + (1 - \zeta) \sum_{n \in S_{m2}} c_{nq}$$

其中  $S_{m1}$  为从事件  $m$  开始活动的集合,  $S_{m2}$  为到事件  $m$  结束活动的集合,  $\zeta$  ( $0 \leq \zeta \leq 1$ ) 为活动费用在其开始和结束事件之间的分配比例. 事件  $m$  的挣值为

$$v_m = \sum_{n \in S_{m2}} w_n$$

其中  $w_n$  为活动  $n$  的挣值. 项目截止日期为  $D$ , 合同总价格为  $U$ ,  $U = \sum_{n=1}^N w_n$ .

在项目开始时, 业主对承包商支  $\gamma U$  ( $0 \leq \gamma \leq 1$ ) 的预付款用于项目的开工准备, 这笔预付款在后续支付中随着项目的进展按比例扣回; 每次支付的支付量  $p_k$  ( $k = 1, 2, \dots, K$ ) 基于承包商已完成的累计挣值(即合同价值量)按支付比例  $\theta$  ( $0 \leq \theta \leq 1$ ) 计算, 但需按比例扣留预付款和质量保证金, 质量保证金的比例为  $\eta$  ( $0 \leq \eta \leq 1$ ); 在项目完成时, 业主须向承包商付清除质量保证金  $\eta U$  外的全部剩余合同价款, 质量保证金在质量保证期满后支付. 在项目进行过程中, 各次支付的安排根据如下 4 种支付条件确定:

- 1) 基于里程碑事件支付条件. 业主在双方商定的  $K$  个里程碑事件  $m_k$  ( $k = 1, 2, \dots, K$ ) 上对承包商进行支付, 项目终事件  $M$  必须是最后一次支付的支付事件.
- 2) 基于累计时间支付条件. 从项目开始时刻算起, 累计时间每达到 1 个支付周期  $[D/K]$  业主即对承包商支付 1 次, 最后一次支付安排在项目的完成时刻.
- 3) 基于累计挣值支付条件. 支付安排依赖于承包商已完成的累计挣值, 每达到 1 个规定的额度  $[U/K]$  即对承包商支付 1 次, 最后一次支付安排在项目的完成时刻.
- 4) 基于累计费用支付条件. 支付安排依赖于承包商所发生的累计费用, 每达到 1 个双方商定的额度  $[C/K]$  ( $C$  为项目基准总费用) 即对承包商支付 1 次, 最后一次支付安排在项目的完成时刻.

定义  $G_T$  为在项目进行过程中  $T$  ( $T = 0, 1, \dots, D$ ) 时刻承包商累计现金流出与流入的差额, 为

$$G_T = \sum_{m=1}^M (e_m \sum_{t=0}^T y_{mt}) - \sum_{k=1}^K (p_k \sum_{t=0}^T y_{mkt}) - \gamma U, \quad T = 0, 1, \dots, D$$

其中第 1 项  $\sum_{m=1}^M (e_m \sum_{t=0}^T y_{mt})$  为至  $T$  时刻承包商累计发生的费用总和, 第 2 项  $\sum_{k=1}^K (p_k \sum_{t=0}^T y_{mkt})$  为至  $T$  时刻承包商累计获得的支付总和, 第 3 项  $\gamma U$  为业主对承包商的预付款. 令  $G_{\max}$  为所有  $G_T$  ( $T = 0, 1, \dots, D$ ) 中的最大者, 则现金流均衡项目调度的目标可定义为最小化  $G_{\max}$ . 由于现金流出(即事件费用  $e_m$ ) 取决于活动模式的选择, 现金流入(即支付量  $p_k$ ) 依赖于支付时承包商已完成累计挣值的多少, 而它们的发生时间均与事件实现时间相联系, 因此定义如下两组决策变量

$$x_{nq} = \begin{cases} 1 & \text{活动 } n \text{ 采用第 } q \text{ 种执行模式} \\ 0 & \text{其他} \end{cases}$$

$$n = 1, 2, \dots, N; q = 1, 2, \dots, Q_n$$

$$y_{mt} = \begin{cases} 1 & \text{事件 } m \text{ 在时刻 } t \text{ 实现} \\ 0 & \text{其他} \end{cases}$$

$$m = 1, 2, \dots, M; t = 1, 2, \dots, D$$

至此, 可将本文所研究的 CFBPSP 界定为: 在项目截止日期  $D$  的约束下, 如何基于不同支付条件选择活动的执行模式(即  $x_{nq}$ ) 并安排事件实现时间(即  $y_{mt}$ ) 以使得  $G_{\max}$  最小化.

## 2 CFBPSP 优化模型构建

### 2.1 基于里程碑事件支付条件的 CFBPSP 优化模型

$$\min G_{\max} = \max_{T=0, 1, \dots, D} \{ G_T = \sum_{m=1}^M (e_m \sum_{t=0}^T y_{mt}) - \sum_{k=1}^K (p_k \sum_{t=0}^T y_{mkt}) - \gamma U \} \quad (1)$$

$$\text{s. t. } \sum_{q=1}^{Q_n} x_{nq} = 1 \quad n = 1, 2, \dots, N \quad (2)$$

$$\sum_{t \in E_m}^{L_m} y_{mt} = 1 \quad m = 1, 2, \dots, M \quad (3)$$

$$\sum_{t \in E_{b_n}}^{L_{b_n}} (y_{b_n t}) + \sum_{q=1}^{Q_n} (d_{nq} x_{nq}) \leq \sum_{t \in E_{o_n}}^{L_{o_n}} (y_{o_n t}), \quad n = 1, 2, \dots, N \quad (4)$$

$$\sum_{t \in E_M}^{L_M} (y_{Mt}) \leq D \quad (5)$$

$$e_m = \zeta \sum_{n \in S_{m1}} \sum_{q=1}^{Q_n} (c_{nq} x_{nq}) + (1 - \zeta) \times \sum_{n \in S_{m2}} \sum_{q=1}^{Q_n} (c_{nq} x_{nq}) \quad m = 1, 2, \dots, M \quad (6)$$

$$p_k = (\theta - \gamma - \eta) \left[ \sum_{m=1}^M \left( v_m \sum_{t=0}^{T_k} y_{mt} \right) - \sum_{m=1}^M \left( v_m \sum_{t=0}^{T_{k-1}} y_{mt} \right) \right] \quad k = 1, 2, \dots, K - 1 \quad (7)$$

$$p_K = (1 - \eta) U - \sum_{k=1}^{K-1} p_k \quad (8)$$

$$\begin{aligned} x_{nq} y_{mt} &\in \{0, 1\} \quad n = 1, 2, \dots, N; \\ q &= 1, 2, \dots, Q_n; \quad m = 1, 2, \dots, M; \\ t &= 0, 1, \dots, D \end{aligned} \quad (9)$$

其中  $b_n$  和  $o_n$  分别为活动  $n$  的开始和结束事件;  $[E_m, L_m]$ 、 $[E_{b_n}, L_{b_n}]$ 、 $[E_{o_n}, L_{o_n}]$  和  $[E_M, L_M]$  分别为事件  $m$ 、 $b_n$ 、 $o_n$  和  $M$  的由网络优先关系和项目截止日期决定的时间窗;  $T_k = \sum_{t=0}^D (t y_{mkt})$  和  $T_{k-1} = \sum_{t=0}^D (t y_{m_{k-1}t})$  分别是第  $k$  次和第  $k - 1$  次支付的时间。

在上述 0 - 1 规划优化模型中, 目标要求式 (1) 最小化  $G_{\max}$ ; 约束条件式 (2) 为每个活动选取一种执行模式; 式 (3) 限定各事件的实现时间必须位于其时间窗内; 式 (4) 为优先关系约束, 确保活动  $n$  的结束事件  $o_n$  的实现时间不早于其开始事件  $b_n$  的实现时间与活动  $n$  的工期之和; 式 (5) 为项目完成时间约束, 保证项目终事件  $M$  的实现时间不晚于截止日期  $D$ ; 式 (6) 基于所选取的活动执行模式确定每个事件的费用; 式 (7) 使得各次中间支付的支付量等于承包商在对应时段完成的累计净值, 与支付比例扣除预付款和质量保证金

后的乘积; 式 (8) 确保最后一次支付能够付清除质量保证金外的全部剩余合同价款; 式 (9) 为决策变量的定义域约束。

## 2.2 其他支付条件下的 CFBPSP 优化模型

### 1) 基于累计时间支付条件的 CFBPSP 优化模型

在基于累计时间支付条件下, 业主对承包商的支付时间根据支付周期而定, 由于本文假定支付必须与项目事件相联系, 所以, 可以按照如下方式安排各次支付: 将项目的开始时间计为 0 时刻, 第  $k$  次支付安排在不早于  $k[D/K]$  时刻的最早实现的事件上。上述安排方式实质上决定了如下  $K$  个支付事件

$$m_k = m \mid y_{m\tau} = 1$$

且

$$\tau = \min\{T : T \geq k[D/K]\}, \quad k = 1, 2, \dots, K \quad (10)$$

式 (1) ~ 式 (10) 构成了基于累计时间支付条件的 CFBPSP 优化模型。

### 2) 基于累计净值支付条件的 CFBPSP 优化模型

在基于累计净值支付条件下, 支付时间根据承包商已完成的累计净值而定, 与基于累计时间支付条件中的处理方式类似, 可以按照如下方式确定支付事件: 从项目开始起计算承包商已完成的累计净值, 第  $k$  次支付安排在使该值达到  $k[U/K]$  的最早实现事件上, 由此确定出如下  $K$  个支付事件

$$m_k = m \mid y_{m\tau} = 1$$

且

$$\begin{aligned} \tau &= \min\left\{T : \sum_{m=1}^M \left( v_m \sum_{t=0}^T y_{mt} \right) \right. \\ &\quad \left. \geq k[U/K] \right\}, \quad k = 1, 2, \dots, K \end{aligned} \quad (11)$$

式 (1) ~ 式 (9) 和式 (11) 一起构成基于累计净值支付条件的 CFBPSP 优化模型。

### 3) 基于累计费用支付条件的 CFBPSP 优化模型

在基于累计费用支付条件下, 按照如下方式确定支付事件: 从项目开始起计算承包商累计发生的费用, 第  $k$  次支付安排在使该费用达到  $k[C/K]$  的最早实现事件上, 由此得到  $K$  个支付事件

$$m_k = m \mid y_{m\tau} = 1$$

$$\tau = \min\{T : \sum_{m=1}^M (e_m \sum_{t=0}^T y_{mt}) \geq k [C/K]\}, k = 1, 2, \dots, K \quad (12)$$

式(1) ~ 式(9) 和式(12) 共同构成基于累计费用支付条件的CFBPSP优化模型.

### 2.3 问题的求解难度分析

不失一般性, 现假定质量保证金的比例  $\eta = 1$ . 这意味着, 在项目实施过程中业主不对承包商进行任何支付 ( $\gamma = 0, p_k = 0$ ), 所有合同价款都要等到质量保证期满之后才支付给承包商. 由此可将上述优化模型的目标要求简化为

$$\min G_{\max} = \sum_{m=1}^M (e_m \sum_{t=0}^D y_{mt})$$

此时,CFBPSP 即转化为在项目截止日期约束下, 确定活动执行模式及事件实现时间以最小化项目总费用, 这实质上等同于离散时间 / 费用权衡问题的工期子问题<sup>[9]</sup>. 由于后者已被证明为一个强 NP-hard 问题<sup>[12]</sup>, 因此,CFBPSP 也必然是强 NP-hard 问题.

## 3 禁忌搜索启发式算法

利用禁忌搜索启发式算法求解本文所研究的问题. 该算法已被众多学者应用于项目调度问题的研究中, 并被证明是一种求解此类问题的高效算法<sup>[13-15]</sup>.

### 3.1 解的表示

定义如下两个决策向量:

$$1) \text{ 活动执行模式向量 } \Psi = (q \mid x_{nq} = 1, n = 1, 2, \dots, N)$$

$$2) \text{ 事件实现时间向量 } \Omega = (t \mid y_{mt} = 1, m = 1, 2, \dots, M)$$

并用  $\Pi = \{\Psi, \Omega\}$  表示问题的解, 给定问题的一个可行解  $\Pi = \{\Psi, \Omega\}$ , 目标函数值可按下述步骤计算:

① 依据由向量  $\Psi$  所确定的活动执行模式, 在各活动费用的基础上利用约束条件式(6) 计算事件费用  $e_m$ .

② 根据支付条件分别由式(10) - 式(12) 确定支付事件, 基于由向量  $\Omega$  所给定的事件实现时间及各事件净值, 通过约束条件式(6) 和式(7)

计算各次支付的支付量  $p_k$ .

③ 根据  $G_T$  表达式计算项目进行过程中各时刻的  $G_T$  值, 从中选出最大者即为目标函数值  $G_{\max}$ .

### 3.2 初始解构造

在算法的搜索过程中, 初始可行解  $\Pi^0 = \{\Psi^0, \Omega^0\}$  按如下步骤构造:

① 为每个活动随机地选择 1 种执行模式, 得到 1 个活动执行模式向量  $\Psi$ , 在  $\Psi$  的基础上计算各事件的时间窗, 如果结束事件  $M$  的最早实现时间不晚于项目截止日期  $D$ , 则接受  $\Psi$  为可行的初始活动执行模式向量  $\Psi^0$ ; 否则, 重复该步操作, 直至得到可行的  $\Psi^0$  为止.

② 在不违反优先关系约束的前提下, 为每个事件在其时间窗内随机地安排一个实现时间, 得到一个事件实现时间向量  $\Omega$ , 如果  $\Omega$  中结束事件  $M$  的实现时间不晚于项目截止日期  $D$ , 则接受  $\Omega$  为可行的初始事件实现时间向量  $\Omega^0$ ; 否则, 重复该步操作, 直至得到可行的  $\Omega^0$  为止.

### 3.3 邻点生成

当前解  $\Pi = \{\Psi, \Omega\}$  的邻点  $\Pi^1 = \{\Psi^1, \Omega^1\}$  用如下两个算子随机生成.

1) 模式变换算子 (MC 算子) 随机地选择 1 个活动, 将其由当前  $\Psi$  所决定的执行模式随机地变换为另外一种执行模式, 在新条件下计算各事件的时间窗, 如果结束事件的最早发生时间不晚于项目截止日期, 则该活动的新执行模式与其余仍然保持不变的活动执行模式一起, 构成当前  $\Psi$  的 1 个可行邻点  $\Psi^1$ ; 反之, 重复该算子操作直至获得 1 个可行邻点  $\Psi^1$  为止. 计算在  $\Psi^1$  下的事件时间窗, 检查各事件的实现时间是否位于其新时间窗内, 如果不是, 则在不违反优先关系约束的前提下将它们调整到新时间窗内, 从而在生成可行  $\Psi^1$  的同时将当前  $\Omega$  也变为一个新的  $\Omega^1$ , 得到当前解的邻点  $\Pi^1 = \{\Psi^1, \Omega^1\}$ .

2) 时间改变算子 (TV 算子) 从除开始事件之外的其余事件中随机地选择 1 个事件, 在其时间窗内将该事件的实现时间随机地变动 1 个单位, 调整其他事件的实现时间以确保网络优先关系约束能够得到满足. 检查结束事件的实现时间是否超过项目截止日期, 如果没有超过则得到当前  $\Omega$  的一个可行邻点  $\Omega^1$ ; 反之, 重复该算子的操作直至获得 1 个可行的  $\Omega^1$  为止. 由于在此过程中

$\Psi$  保持不变, 所以由此得到的当前解的邻点为  $\Pi^1 = \{\Psi, \Omega^1\}$ .

在禁忌搜索算法的每一次生成邻点操作时, 以上两个算子均以 0.5 的概率被随机选取, 独立地应用于邻点生成操作中.

### 3.4 移动定义

对应于生成邻点操作所使用的不同算子, 相应的移动定义如下.

1) MC 移动 用一个 3 元向量 (在  $\Psi$  中所选元素的位置, 该元素的初始值, 该元素的新值) 表示. 举例说明, 如果  $\Psi$  中位置 3 上的元素取值由 1 变成了 2, 则 MC 移动表示为 (3, 1, 2), 其含义是活动 3 的执行模式由 1 转变为 2. 该移动的逆向移动表示为 (3, 1) 并被同时加入到禁忌列表中, 以避免活动 3 的执行模式再次变回到 1.

2) TV 移动 用一个 3 元向量 (在  $\Omega$  中所选元素的位置, 该元素的初始值, 该元素的新值) 表示. 举例说明, 如果  $\Omega$  中位置 6 上的元素取值由 13 变成了 12, 那么 TV 移动表示为 (6, 13, 12), 其含义是事件 6 的实现时间由 13 变成了 12. 该移动的逆向移动表示为 (6, 13) 并被同时加入到禁忌列表中, 以避免事件 6 的实现时间重新变回 13.

### 3.5 禁忌列表

在给定禁忌列表长度 (在本文中设置为 15) 下, 采用“先进先出 (first-in-first-out, FIFO)”的原则进行更新. 每当邻点生成算子形成一个移动时, 该移动的逆向移动从底部加入到禁忌列表中, 与此同时, 最早进入列表的逆向移动从顶部移出列表, 列表中其余逆向移动向上递进 1 位. 所有位于禁忌列表中的逆向移动都是被禁止的, 但当 1 个被禁止的逆向移动能够生成比当前最好解还要好的邻点时, 那么它的禁忌状态可以被激活, 即将其从禁忌列表中移出, 其下所有逆向移动向上递进 1 位, 同时将该逆向移动的逆向移动加入到禁忌列表的底部.

### 3.6 搜索步骤

在上述对禁忌搜索关键环节进行界定的基础上, 给出该算法的具体搜索步骤.

步骤 1 输入初始解  $\Pi^0 = \{\Psi^0, \Omega^0\}$  及其对应的目标函数值  $G_{\max}^0$ ; 输入算法终止准则, 即探测可行解总数  $NUM_{\text{stop}}$ ; 初始化禁忌列表; 令计数器  $Num = 0$ ; 令当前解及当前最好解等于初始解;

$$\Pi^{\text{curr}} = \Pi^{\text{opti}} = \Pi^0, G_{\max}^{\text{curr}} = G_{\max}^{\text{opti}} = G_{\max}^0.$$

步骤 2 随机生成当前解的 1 个可行邻点并计算其目标函数值, 记为  $\Pi^1 = \{\Psi^1, \Omega^1\}$ ,  $G_{\max}^1$ , 令  $Num = Num + 1$ . 若  $Num \geq NUM_{\text{stop}}$  转步骤 5; 否则 转步骤 3.

步骤 3 判断生成邻点  $\Pi^1 = \{\Psi^1, \Omega^1\}$  的移动是否位于禁忌列表中, 若不在禁忌列表中, 将当前解更新为该邻点:  $\Pi^{\text{curr}} = \Pi^1, G_{\max}^{\text{curr}} = G_{\max}^1$ . 如果  $G_{\max}^1 < G_{\max}^{\text{opti}}$ , 将当前最好解也更新为该邻点:  $\Pi^{\text{opti}} = \Pi^1, G_{\max}^{\text{opti}} = G_{\max}^1$ . 更新禁忌列表, 转步骤 2; 若在禁忌列表中, 转步骤 4.

步骤 4 判断  $G_{\max}^1 < G_{\max}^{\text{opti}}$  是否成立, 若成立, 激活对应逆向移动的禁忌状态, 将当前解及当前最好解更新为该邻点:  $\Pi^{\text{curr}} = \Pi^{\text{opti}} = \Pi^1, G_{\max}^{\text{curr}} = G_{\max}^{\text{opti}} = G_{\max}^1$ , 更新禁忌列表, 转步骤 2; 否则, 直接转步骤 2.

步骤 5 判断  $G_{\max}^1 < G_{\max}^{\text{opti}}$  是否成立, 若成立, 则将当前最好解更新为该邻点:  $\Pi^{\text{opti}} = \Pi^1, G_{\max}^{\text{opti}} = G_{\max}^1$ . 输出当前最好解  $\Pi^{\text{opti}}, G_{\max}^{\text{opti}}$ , 它们为得到的满意解.

## 4 算法测试

为测试上述禁忌搜索启发式算法的绩效, 本节给出如下两个启发式算法作为对比:

1) 多重迭代<sup>[14]</sup> 该算法从 1 个初始可行解启动, 使用 3.3 中定义的算子生成邻点并选择使目标函数改进最为显著的邻点进行迭代. 在搜索过程中, 如果不存在可使当前解改进的邻点, 则算法从另一个随机生成的初始可行解重新开始迭代. 当搜索的可行解达到某一规定的数值时, 算法终止并输出得到的最好解为问题的满意解.

2) 随机生成 算法使用 3.2 所定义的方法随机生成某一规定数目的可行解, 从这些解中选择最好的解作为问题的满意解.

算法测试在由项目调度问题算例生成器 ProGen<sup>[16]</sup> 随机生成的标准算例进行, ProGen 生成算例时的参数设置见表 1. 算例的其他参数取值如下:  $\gamma = 0.03, \eta = 0.05, \zeta = 0.5, C = \frac{1}{2} \sum_{i=1}^N (c_{n1} + c_{n2})$ . 此外, 在基于里程碑事件支付条

件下,除项目终事件  $M$  外,其他里程碑事件均随机选定.在算例的所有参数中,按全因素试验设计的有4个:项目非虚活动数  $N$ 、支付次数  $K$ 、支付比例  $\theta$  和项目截止日期  $D$ .参数  $N$  的取值为4种, $K$ 、 $\theta$  和  $D$  的取值均为3种.在每种参数组合下生成的算例数为10个,考虑到每个算例均具有4种支付条件,由此总共形成  $10 \times 4 \times 3^3 \times 4 = 4\,320$  个标准测试算例.

表1 ProGen 的参数设置

Table 1 Parameter setting of ProGen

ProGen 参数	取 值
算例的非虚活动数 $N$	10、20、30 或 40
在某一非虚活动数下生成的算例数	10
算例的起始和终止活动数	随机地从 2、3 和 4 中选取
最大紧前和紧后活动数	4
活动执行模式数	2
模式 1 下的活动工期 $d_{n1}$	从 [1, 10] 中均匀地随机选取
模式 1 下的活动费用 $c_{n1}$	从 [10, 20] 中均匀地随机选取
模式 2 下的活动工期 $d_{n2}$	$\rho_1 \cdot d_{n1}$ , 其中 $\rho_1$ 从 [0.8, 1] 中均匀地随机选取
模式 2 下的活动费用 $c_{n2}$	$\rho_2 \cdot c_{n1}$ , 其中 $\rho_2$ 从 [1, 1.2] 中均匀地随机选取
活动的挣值 $w_n$	$\rho_3 \cdot cn_2$ , 其中 $\rho_3$ 从 [1.1, 1.3] 中均匀地随机选取
支付次数 $K$	3, 4, 5
支付比例 $\theta$	0.7, 0.8, 0.9
项目截止日期 $D$	$\rho_4 (C_{\max} - C_{\min}) + C_{\min}$ , 其中, $C_{\max}$ 和 $C_{\min}$ 分别是所有活动均采用模式 1 和 2 时项目网络关键路线的长度, $\rho_4$ 设置为 0.4、0.6 和 0.8

定义如下 4 个指标来反映启发式算法的绩效.

①  $ARD(\%)$ : 满意解距离已知最好解的平均相对偏差;

②  $MRD(\%)$ : 满意解距离已知最好解的最大相对偏差;

③  $ACT(s)$ : 启发式算法的平均计算时间;

④  $MCT(s)$ : 启发式算法的最大计算时间.其中

已知最好解是指在 3 个启发式算法所获得的满意解中,目标函数值最小的一个.算法采用 Visual Basic 6.0 语言编程,在 CPU 主频为 1.6 GHz、内存为 256 MB 的个人计算机上运行.3 个算法均使用相同的停机准则,即在探测了相同总数的可行解(设定为  $N \times 10\,000$ ) 后停止搜索,输出得到的最好解为满意解.

算法的测试结果见表 2.首先从解的质量看,禁忌搜索的  $ARD$  和  $MRD$  要远小于多重迭代和随机生成,而且这种差距随着问题规模(即项目非虚活动数  $N$ ) 的增大而增大,表明禁忌搜索获得的满意解的质量要远好于其他两种算法,当问题变得复杂时这种优势更为显著.其次从计算时间看,禁忌搜索的  $ACT$  和  $MCT$  要大于其他两种启发式算法,说明在探测可行解数目相同的条件下,所需的计算时间较长.上述结果的原因可以解释如下,与禁忌搜索相比,多重迭代不需要管理禁忌列表,而随机生成在此基础上甚至连邻点的生成、判断及迭代也不需要,所以它们的计算速度较快.但是,二者得到的满意解质量较差,表明它们虽然在较短的时间内探测了较多的可行解,但这些可行解的重复率较高,因而算法的搜索效率并不高.与此相反,禁忌搜索尽管在禁忌列表管理上花费了一定的时间,却有效地提高了算法的搜索效率,在可以接受的时间里(所有算例的求解时间不超过 51.07 秒)获得了较好的满意解.因此,对于本文所研究的问题来说,禁忌搜索优于其他两种启发式算法.

表 3 反映了项目关键参数对目标函数值  $G_{\max}$  的影响.由表 3 可见,随着支付次数  $K$ 、支付比例  $\theta$  和项目截止日期  $D$  (用  $\rho_4$  表示) 的增大,  $G_{\max}$  均呈下降的趋势.这一结果解释如下:当支付次数  $K$  增加时,承包商可以更快地从业主那里获得资金补偿,这使得其现金流入可以更为均衡地分布于项目实施过程中,因而现金流出与流入的最大累计差额  $G_{\max}$  相应下降;支付比例  $\theta$  决定了各次支付的支付量,随着  $\theta$  的增加,承包商在各次支付中可以从业主那里得到更多的资金补偿,在现金流出不变的前提下,现金流入的这种增加必然引起  $G_{\max}$  的下降;当项目截止日期延长时,承包商可以更多地选择费用较低而工期较长的模式完成活动,这会减少承包商的现金流出从而导致  $G_{\max}$  下降.

表 2 算法测试结果

Table 2 Test results of algorithms

支付条件	N	禁忌搜索				多重迭代				随机生成			
		ARD (%)	MRD (%)	ACT /s	MCT /s	ARD (%)	MRD (%)	ACT /s	MCT /s	ARD (%)	MRD (%)	ACT /s	MCT /s
基于里程碑事件	10	0.07	0.11	0.83	1.36	0.06	0.24	0.25	0.43	0.09	0.24	0.14	0.24
	20	0.05	0.09	1.91	3.25	1.32	1.98	1.04	1.77	7.76	11.64	0.61	1.04
	30	0.00	0.01	11.99	20.38	6.13	9.19	7.14	12.14	17.28	27.92	3.63	6.17
	40	0.00	0.00	27.87	47.26	10.92	17.88	15.99	27.18	29.16	53.74	7.32	12.44
基于累计时间	10	0.06	0.07	0.94	1.36	0.07	0.14	0.33	0.51	0.14	0.31	0.16	0.25
	20	0.02	0.09	2.67	5.12	2.30	3.68	1.50	3.01	8.35	15.21	0.75	2.00
	30	0.00	0.00	14.71	22.43	8.95	11.19	9.09	15.33	18.63	34.79	4.21	7.32
	40	0.00	0.00	29.57	50.77	12.04	19.88	18.96	28.69	30.59	63.18	8.46	13.99
基于累计挣值	10	0.05	0.10	0.93	1.38	0.04	0.16	0.27	0.49	0.12	0.25	0.16	0.27
	20	0.03	0.07	2.15	4.65	0.95	2.47	1.67	2.89	8.88	13.32	0.69	1.17
	30	0.00	0.00	15.76	27.79	6.03	9.04	8.67	16.73	19.02	30.53	3.59	7.10
	40	0.00	0.00	30.60	48.02	10.02	17.03	16.02	29.23	35.01	59.51	8.26	14.04
基于累计费用	10	0.04	0.06	1.11	1.67	0.08	0.22	0.28	0.56	0.13	0.36	0.18	0.26
	20	0.02	0.03	2.64	3.91	2.01	3.42	1.25	2.08	10.32	15.48	0.77	1.09
	30	0.00	0.00	13.29	23.71	7.11	11.65	9.40	16.70	19.71	29.95	3.72	6.88
	40	0.00	0.00	31.77	51.07	13.79	20.08	17.77	30.65	32.22	60.33	9.39	12.67

表 3 相关参数对目标函数值的影响

Table 3 Influences of the relative parameters on the objective function value

参数	取值	禁忌搜索				多重迭代				随机生成			
		基于里程碑事件支付条件	基于累计时间支付条件	基于累计挣值支付条件	基于累计费用支付条件	基于里程碑事件支付条件	基于累计时间支付条件	基于累计挣值支付条件	基于累计费用支付条件	基于里程碑事件支付条件	基于累计时间支付条件	基于累计挣值支付条件	基于累计费用支付条件
K	3	77.73	70.11	78.68	75.16	97.22	88.89	92.63	95.18	105.55	97.13	109.83	105.66
	4	72.85	64.80	66.15	68.26	77.41	75.61	76.51	77.38	88.57	84.92	81.47	90.56
	5	55.71	61.86	59.43	60.47	73.79	69.61	71.31	69.13	84.38	77.86	70.90	73.86
θ	0.7	79.01	75.88	93.57	87.38	96.32	94.36	97.88	101.51	112.82	104.42	111.95	110.07
	0.8	74.73	68.84	66.15	67.26	87.17	78.61	76.77	81.38	92.57	86.09	82.47	95.56
	0.9	52.38	51.42	44.68	48.61	64.65	60.17	65.62	58.82	72.462	68.91	68.01	65.25
ρ <sub>4</sub>	0.4	79.02	70.01	77.31	74.70	90.20	83.63	87.75	90.65	100.66	95.38	91.82	101.33
	0.6	66.73	64.77	65.15	67.26	81.17	78.77	80.77	81.38	96.57	87.92	88.47	94.16
	0.8	60.87	61.93	61.68	61.95	77.76	71.92	72.02	70.32	81.55	75.05	82.23	76.05

### 5 算例

其中每个活动均有常规和加急两种执行模式,活动的挣值及在不同模式下的工期和费用列于表 4。项目截止日期  $D$  为 35, 合同总价款  $U$  为 22 000,

用图 1 所示的算例对本文的研究进行说明, 项目基准总费用  $C$  为 15 000, 活动费用在其开始

和结束事件之间的分配比例  $\zeta$  为 0.5, 预付款比例  $\gamma$  和质量保证金比例  $\eta$  均为 5%, 支付比例  $\theta$  为 80%, 支付次数  $K$  为 4. 4 种支付条件定义如下:

- ① 基于里程碑事件支付. 当里程碑事件 4、6、8 和 10 实现时进行支付.
- ② 基于累计时间支付. 每过 1 个支付周期  $[D/K]$  进行 1 次支付.
- ③ 基于累计挣值支付. 每当承包商完成的累计挣值达到 1 个  $[U/K]$  进行 1 次支付.

④ 基于累计费用支付. 每当承包商发生的累计费用达到 1 个  $[C/K]$  进行 1 次支付.

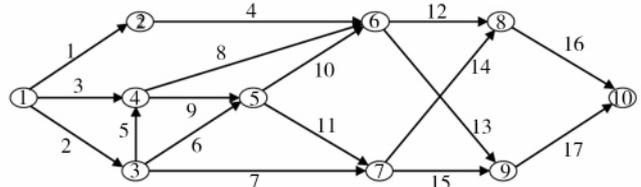


图 1 项目网络图  
Fig. 1 Project network

表 4 活动的挣值及其在不同执行模式下的工期和费用

Table 4 Activities' earned values and their durations and costs under different payment mode

活动代号	挣值	加急模式		常规模式		活动代号	挣值	加急模式		常规模式	
		工期	费用	工期	费用			工期	费用		
1	1 000	4	800	5	600	10	1 500	6	1 200	8	800
2	760	2	600	3	400	11	1 500	6	1 200	7	1 000
3	1 800	1	1 400	2	1 000	12	760	4	600	5	400
4	1 800	5	1 400	6	1 000	13	760	3	600	4	400
5	1 300	1	1 000	2	600	14	1 500	2	1 200	3	800
6	1 000	8	800	10	500	15	1 300	10	1 000	12	700
7	1 000	6	800	8	500	16	2 700	5	2 100	6	1 800
8	1 000	3	800	4	600	17	1 900	4	1 500	5	1 200
9	420	6	300	8	200						

利用禁忌搜索启发式算法求解基于不同支付条件的 CFBPSP 优化模型, 得到的结果见表 5. 由计算结果可见, 随着支付条件的不同, 承包商的满意进度安排不同, 进而导致  $G_{max}$  不同. 对于该项目来说, 从现金流均衡的角度出发, 基于累计费用支付条件对承包商最为有利, 累计现金流出去减去累计现金流入的最大差额 (亦即最大累计资金缺口)  $G_{max}$  为 1 148; 其次为基于里程碑

事件支付条件和基于累计时间支付条件, 它们所对应的  $G_{max}$  分别为 1 248 和 1 700; 而基于累计挣值支付条件对承包商最为不利, 其在项目执行过程中会形成  $G_{max} = 2 100$  的最大累计资金缺口, 比基于累计费用、里程碑事件和累计时间支付条件下的  $G_{max}$  分别高 82.9%、68.3% 和 23.5%, 由此可见支付条件对承包商现金流均衡的重要性.

表 5 不同支付条件下的项目满意进度安排及对应的目标函数值

Table 5 Desirable project schedules and the corresponding objective function values under different payment conditions

		基于里程碑事件支付	基于累计时间支付	基于累计挣值支付	基于累计费用支付
$\Pi^{opti}$	$\Psi^{opti}$	(2 2 2 2 2 1 2 2 2 1 2 1 1 2 1 2 1)	(2 2 2 2 2 2 2 2 1 1 1 2 2 1 1 2 1)	(2 2 2 2 2 2 2 2 2 1 2 2 2 1 1 2 2)	(2 2 2 2 2 1 2 1 2 2 2 1 1 1 1 1 1)
	$\Omega^{opti}$	(0 6 3 6 6 14 21 21 24 31 35)	(0 5 5 9 15 21 21 26 31 35)	(0 13 3 5 13 19 20 25 30 35)	(0 5 3 5 13 21 21 27 31 35)
$G_{max}^{opti}$		1 248	1 700	2 100	1 148

注:  $\Psi^{opti}$  中的“1”和“2”分别代表活动的加急模式和常规模式.

4 种支付条件下承包商累计现金流出、累计现金流入和累计资金缺口随时间的变化曲线见图 2. 以基于累计费用支付条件为例对承包商现金流均衡结果进行说明. 在基于累计费用支付条件下, 依赖于表 5 所给的活动执行模式向量  $\Psi^{opti}$ , 事件 1 ~ 10 的费用分别为 1 000、800、1 150、1 300、

1 400、1 900、1 850、1 950、1 550 和 800, 它们分别发生于由事件实现时间向量  $\Omega^{opti}$  所决定的第 0、5、3、5、13、21、21、27、31 和 35 时刻; 而承包商在项目开始时可以获得 1 100 的预付款, 随后按照累计费用支付条件的规定, 当事件 4、6、8 和 10 实现时, 承包商可进一步获得 3 402、5 754、1 582 和 9 062

的支付; 将现金流出与现金流入综合到一起, 便可发现最大累计资金缺口  $G_{max}$  为发生在第 13 时刻的 1 148。也就是说, 对于该算例项目, 承包商只要具备 1 148 的融资能力就能确保它的顺利实施。

进一步分析支付条件对承包商现金流均衡目标的影响。令  $C_p$  为发生在业主支付时刻的事件总费用, 通过简单计算可以得到在上述满意进度安排下, 基于累计费用、基于里程碑事件、基于累计时间和基于累计挣值支付条件下的  $C_p$  值分别为

9 600、8 900、8 100 和 6 900。将  $C_p$  和  $G_{max}$  同时绘制于图 3 中, 便会发现  $C_p$  随支付方式的变化趋势与  $G_{max}$  的正好相反。由此可以推断, 使得承包商能够将事件费用集中于支付时刻的支付条件有利于现金流均衡目标的实现。这一结果的背后原因如下: 当较多事件费用集中地发生在支付时刻时, 这些费用所引起的现金流出现会因业主支付而形成的现金流入及时得到补偿, 从而避免较大资金缺口的出现; 反之, 现金流出与流入在时间分布上的差异越大, 则承包商的资金缺口越大。

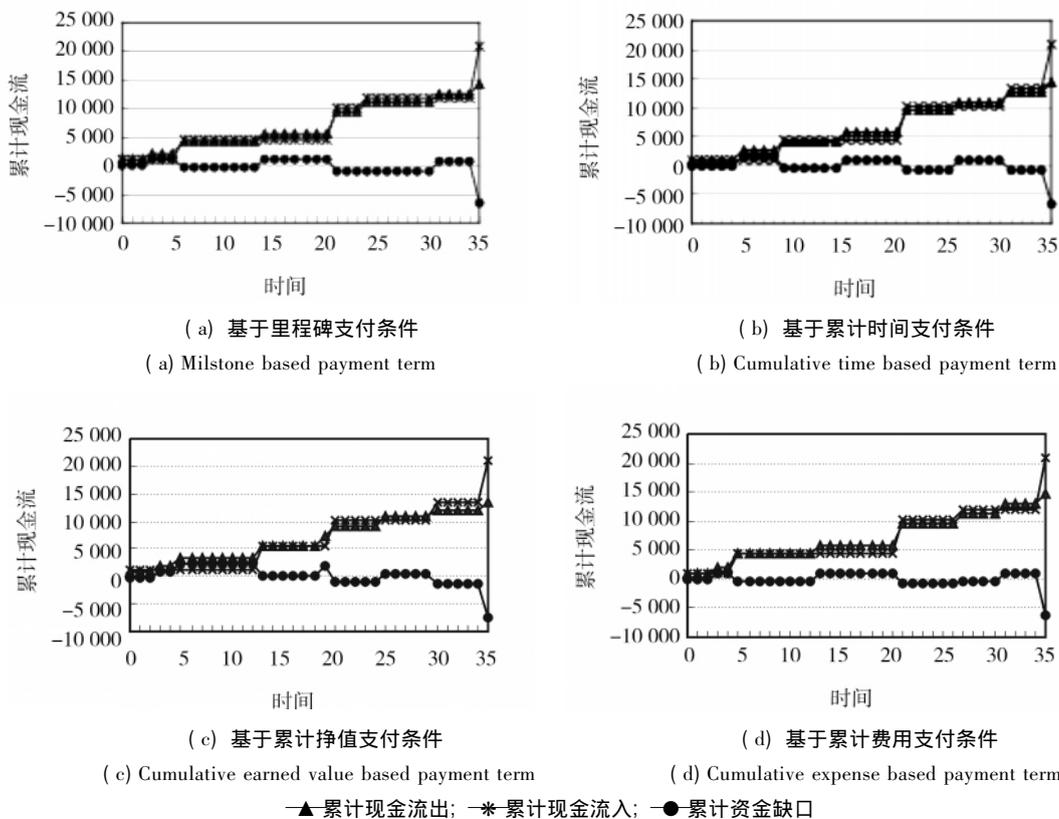


图 2 四种支付条件下承包商累计现金流出、累计现金流入和累计资金缺口随时间变化曲线

Fig. 2 Time-variation curves of the contractor's cumulative cash outflow, cash inflow, and capital gap under four payment conditions

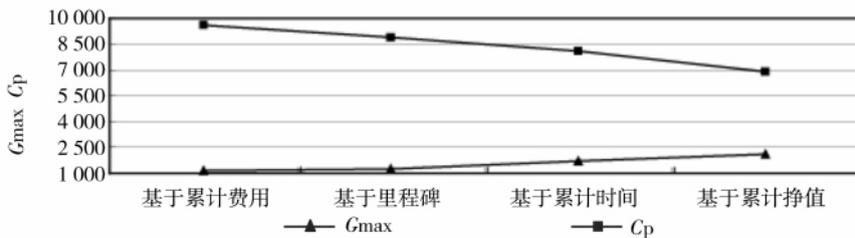


图 3  $G_{max}$  和  $C_p$  随支付条件的变化曲线

Fig. 3 Variation curves of  $G_{max}$  and  $C_p$  with payment conditions

## 6 结束语

本文研究了基于里程碑、累计时间、累计挣值和累计费用等4种支付条件下的现金流均衡项目调度问题,首先对所研究问题进行界定,目标是在项目截止日期的约束下,合理选择活动执行模式并安排事件实现时间,以最小化累计现金流出减去累计现金流入的最大差额;随后,构建了基于里程碑事件支付条件的优化模型,在此基础上加入确定支付事件的附加约束条件,进一步得到基于其他支付条件的优化模型,通过将该问题转化为离散时间/费用权衡问题的工期子问题,分析了问题的求解难度;针对问题的强 NP-hard 属性,设计了禁忌搜索启发式算法,定义了解的表述、初始

解构造、邻点生成及移动、禁忌列表,并给出算法的具体搜索步骤;在随机生成的标准算例集合上对算法进行了测试,并与其他两种启发式算法进行了比较,验证了算法的有效性,通过测试试验还进一步分析了项目关键参数对目标函数值的影响;最后,通过一个算例对本文研究进行说明,分别求解了该算例在4种支付条件下的满意进度安排,分析了累计现金流出与累计现金流入随时间的变化情况,总结出不同支付条件下目标函数值产生差异的原因。

可以得到如下结论:支付条件对于承包商现金流均衡会产生重要影响,当支付次数和支付比例增大以及项目截止日期延长时,承包商更容易实现现金流出与流入的均衡。

### 参考文献:

- [1]陈祥锋,朱道立,应雯珺. 资金约束与供应链中的融资和运营综合决策研究[J]. 管理科学学报,2008,11(3): 70-77.  
Chen Xiang-feng, Zhu Dao-li, Ying Wen-jun. Financial and operation decisions in budget-constrained supply chain [J]. Journal of Management Sciences in China, 2008, 11(3): 70-77. (in Chinese)
- [2]Kolisch R, Padman R. An integrated survey of deterministic project scheduling [J]. Omega, 2001, 29(3): 249-272.
- [3]Herroelen W, Leus R. Project scheduling under uncertainty: Survey and research potential [J]. European Journal of Operational Research, 2005, 165(2): 289-306.
- [4]姚建明,周国华. 大规模定制模式下供应链计划调度优化分析[J]. 管理科学学报,2003,6(5): 58-64.  
Yao Jian-ming, Zhou Guo-hua. Analysis of supply chain optimization planning and scheduling in mass customization [J]. Journal of Management Sciences in China, 2003, 6(5): 58-64. (in Chinese)
- [5]Demeulemeester E, Herroelen W. A branch-and-bound procedure for the multiple resource-constrained project scheduling problem [J]. Management Science, 1992, 38(12): 1803-1818.
- [6]何正文,徐渝,朱少英. Max-npv 项目进度问题研究评述[J]. 管理工程学报,2005,19(4): 60-63.  
He Zheng-wen, Xu Yu, Zhu Shao-ying. A survey of max-npv project scheduling problems [J]. Journal of Industrial Engineering and Engineering Management, 2005, 19(4): 60-63. (in Chinese)
- [7]Yamashita D S, Armentano V A, Laguna M. Scatter search for project scheduling with resource availability cost [J]. European Journal of Operational Research, 2006, 169: 623-637.
- [8]刘士新,王梦光. 一种求解工程调度中资源水平问题的遗传算法[J]. 系统工程理论与实践,2001,21(4): 24-26.  
Liu Shi-xin, Wang Meng-guang. A genetic algorithm for resource leveling problem in project scheduling [J]. Systems Engineering-theory & Practice, 2001, 21(4): 24-26. (in Chinese)
- [9]De P, Dunne E J, Ghosh J B, et al. The discrete time-cost tradeoff problem revisited [J]. European Journal of Operational Research, 1995, 81(2): 225-238.
- [10]张静文,徐渝,何正文,等. 项目调度中的时间-费用权衡问题研究综述[J]. 管理工程学报,2007,21(1): 92-97.  
Zhang Jing-wen, Xu Yu, He Zheng-wen, et al. A review on the time/cost trade-offs problem in project scheduling [J]. Journal of Industrial Engineering and Engineering Management, 2007, 21(1): 92-97. (in Chinese)
- [11]Elmaghraby S. Activity nets: A guided tour through some recent developments [J]. European Journal of Operational Research, 1995, 82(3): 383-408.

- [12] De P, Dunne E J, Ghosh J B, et al. Complexity of the discrete time-cost tradeoff problem for project networks [J]. *Operations Research*, 1997, 45(2): 302–306.
- [13] Mika M, Waligóra G, Weglarz J. Simulated annealing and tabu search for multi-mode resource-constrained project scheduling with positive discounted cash flows and different payment models [J]. *European Journal of Operational Research*, 2005, 164(3): 639–668.
- [14] Mika M, Waligóra G, Weglarz J. Tabu search for multi-mode resource-constrained project scheduling with schedule-dependent setup times [J]. *European Journal of Operational Research*, 2008, 187(3): 1238–1250.
- [15] Lambrechts O, Demeulemeester E, Herroelen W. A tabu search procedure for developing robust predictive project schedules [J]. *International Journal of Production Economics*, 2008, 111(2): 493–508.
- [16] Kolisch R, Sprecher A. PSPLIB—a project scheduling problem library [J]. *European Journal of Operational Research*, 1996, 96(1): 205–216.

## Cash flow balanced project scheduling optimization based on different payment conditions

HE Zheng-wen, LIU Ren-jing, XU Yu

School of Management, Xi'an Jiaotong University, Xi'an 710049, China

**Abstract:** Taking cash flow balance as the objective, this paper studies the project scheduling problem under different payment conditions. Based on the identification of the studied problem, the authors construct the optimization model under the milestone event payment condition and then extend it to the cumulative time, cumulative earned value, and cumulative expense payment conditions. Because of the strong NP-hardness of the problem, a tabu search heuristic algorithm is developed. The algorithm is tested and compared with other two heuristic algorithms on a data set generated randomly. Ultimately, an example is used to illustrate the method and the influences of the payment conditions on the objective of cash flow balance are analyzed. The conclusions are as follows: Payment conditions may exert an important influence on cash flow balance, and with the increase of payment number, payment proportion, and project deadline, it becomes easier for contractor to balance cash outflows and inflows.

**Key words:** project scheduling; cash flow balance; payment condition; optimization model; tabu search