

广义优先关系下工序机动时间的隐形消耗^①

苏志雄, 魏汉英, 李 岚

(南昌工程学院工商管理学院, 南昌 330099)

摘要: 在带有广义优先关系 (generalized precedence relations, 简称 GPRs) 的工序网络中, 发现了新的奇异现象. 传统观念中, 如果某工序的机动时间被消耗, 则必然发生在以下两种情况中: 1) 该工序主动消耗自身机动时间; 2) 由于该工序的前继工序消耗各自的机动时间, 导致该工序被动地消耗自身机动时间. 然而新发现的奇异现象是, 即使脱离上述两种情况, 某工序的机动时间也会被消耗. 该现象称为工序机动时间的隐性消耗, 出现在带有 GPRs 的工程项目中. 在 GPRs 网络的基础上, 研究了该奇异现象的特性, 分别针对工序的经典时差和隐性时差, 提出了相应的机动时间隐性消耗的量化方法. 对于带有 GPRs 的项目调度问题, 工序机动时间隐性消耗的现象会弱化现有的基于机动时间的模型和优化算法. 因此, 为了进一步提高项目调度的效率和准确性, 对该奇异现象的理论研究是不可或缺的.

关键词: 工序网络; 广义优先关系; 奇异现象; 机动时间

中图分类号: O221 **文献标识码:** A **文章编号:** 1007-9807(2018)03-0069-13

0 引 言

在当前的生产和运作管理中, 日益紧张激烈的时间竞争已成为最显著的特征之一. 时间早在上世纪 80 年代就被认为是竞争优势来源^[1], 使得项目管理者越来越重视时间在项目中的功能. 机动时间是该竞争优势来源的具体化表现, 主要用于避免和缓和潜在的项目延期, 以及保护项目的重要工序. 某工序的机动时间不仅刻画了该工序对于项目的重要性, 而且还反映了项目的结构特性, 为项目计划与进度提供指导依据. 鉴于此, 机动时间长期被视作项目优化的重要参数, 多数项目管理方法, 特别是项目调度, 均以机动时间为基础^[2-5]. 根据项目优化问题的不同, 相应的处理方法也是千差万别、各有精妙, 但是总结其原理能够发现, 一些传统观念在其中起着重要的指导作用, 例如: 1) 延迟关键工序会导致总工期延迟, 反之

则反; 2) 机动时间会越用越少; 3) 如果工序都不使用各自机动时间, 并且总工期也不变, 那么任何工序的机动时间不会改变; 等等.

然而, 自从 Roy^[6] 引入工序之间的广义优先关系 (generalized precedence relations, GPRs) 的概念后, 人们发现在带有 GPRs 的项目中存在很多奇异现象, 足以颠覆上述传统观念^[7-12]. GPRs 表示项目中不同工序之间在开始/结束时间上的最小和最大时距, 涵盖了工序之间的各类优先关系. Elmaghraby 和 Kamburowski^[7]、张立辉等^[9] 发现了缩短(延长)关键工序的工期反而造成总工期延长(缩短)的奇异现象, 与上述传统观念 1) 相悖. Qi 和 Su^[10] 进一步发现了机动时间竟然可以越用越多, 与上述传统观念 2) 相悖; 苏志雄等^[11] 在此基础上提出了隐性时差和伪时差等新机动时间类型, 较好地诠释和应对了该现象. 机动时间是带有 GPRs 的项目调度的重要参数, 特别是在在问

^① 收稿日期: 2016-04-19; 修订日期: 2017-07-07.

基金项目: 国家自然科学基金资助项目(71271080); 江西省自然科学基金资助面上项目(20171BAA208001).

作者简介: 苏志雄(1983), 男, 山西朔州人, 讲师, 博士生. Email: suzhixiongbaner@126.com

题建模上^[13-15]. 因此, 对上述奇异现象的研究可能会导致对项目优化研究的革新. 已有研究者基于这些奇异现象研究了带有 GPRs 的项目调度的新方法. 例如, Elmaghraby 和 Kamburowski^[7] 研究了如何在求解带有 GPRs 的时间-费用权衡问题时规避奇异现象; 受此启发, 苏志雄等^[16] 通过研究成功避免了奇异现象对该问题的影响, 实现了问题的初始最优化和等效化简; 而 Zhang 等^[17] 通过考虑奇异现象, 提出了重复性项目(可视为特殊的带有 GPRs 的项目)中时间-费用权衡的新方法.

目前机动时间理论研究的主要表现之一为, 在项目开始前, 对机动时间进行计算和特性分析, 并制定合理的计划方案等. 根据上述表现的特点, 可视为从静态的角度研究机动时间. 本文从新的角度——动态的角度, 研究项目在运行过程中工序机动时间的特性规律, 发现了带有 GPRs 的工程项目中的新奇异现象——工序机动时间的隐性消耗. 该现象与第 1 段所述的传统观念 3) 相悖. 根据常理, 如果延迟某工序的开始时间, 或者延长其工期, 该工序就会消耗自身的机动时间. 因此, 传统观念认为, 只有以下两种情况会导致某工序机动时间的消耗: 1) 该工序主动消耗自身机动时间; 2) 受其他工序的影响, 该工序被动地消耗自身机动时间, 例如, 该工序的紧前工序延迟, 使得该工序被迫推迟开始时间. 然而本文发现, 即使没有上述两种情况, 甚至所有工序都未使用机动时间, 且总工期也未变, 某些工序的机动时间仍然会被消耗, 这种看不见的消耗可称为隐性消耗. 工序机动时间的隐性消耗揭示了 GPRs 相关理论的新缺口, 可能导致 GPRs 下机动时间的现有计算方法失效, 从而使与机动时间有关的项目优化问题的现有理念发生动摇, 几乎弱化了以机动时间为基础的所有方法.

工序机动时间的隐性消耗主要发生在项目运行过程中, 这意味着即使实际条件与计划时的预期条件一致, 项目的实际运行结果也可能与计划的预期结果有所差异. 例如, 在计划中, 根据机动时间, 将某工序进行延期调整, 进而实现项目优化; 但实际进度中, 如果在该工序即将运行的时候才按计划施行调整, 其机动时间的隐性消耗将可能导致总工期延迟、甚至项目不可行等严重后果

(参见第 2 节). 因此在带有 GPRs 的项目中, 机动时间的隐性消耗对项目计划的实施提出了更高的要求, 如对上例而言, 只有在机动时间出现隐性消耗之前完成计划中的调整, 才能实现预期优化效果, 否则将背离目标. 机动时间的隐性消耗也可能增大不确定性项目调度问题^[18-21]的实际难度, 因为在不确定环境下, 工序的实际状态(如开始时间、工期等)只有在其运行时才可能知晓^[20], 但此时机动时间的隐性消耗可能已发生, 从而导致原计划的调度方案不可行.

另外, 机动时间的隐性消耗还能导致 GPRs 下其他新的项目调度问题, 如反应性项目调度. 经典的项目调度问题常假设项目环境是确定且稳定不变的, 在此情况下建立项目进度计划以达到目标函数的最优化. 但是在执行过程中, 活动的工期、资源的可用量等随时可能发生变化, 导致项目无法按原计划执行, 因此必须根据实际情况对基准进度进行调整, 或者制定新的计划以适应实际环境, 该过程被称为反应性项目调度^[22]. 何正文等^[23] 对该问题的前期研究进行了综述分析, 指出相比其他调度问题, 反应性项目调度的研究成果较为有限, 尚未形成完整的体系. 而最新研究中, 王艳婷等^[24]、Suwa 和 Morita^[25] 以及 Zheng 等^[26] 分别研究了活动工期随机中断下的反应性项目调度、反应性资源受限项目调度、以及强化项目进度平滑性的反应性项目调度等问题, 分别提出了相应的启发式算法. 然而, 现有研究鲜有针对带有 GPRs 的项目, 而该项目的反应性调度具有新的特征和困难.

当带有 GPRs 的项目情况出现变化, 需要进行反应性调度时, 即使之前的进度完全按照计划进行, 但是由于在项目运行中, 工序的机动时间可能出现隐性消耗, 因此很多工序的时间参数与计划相比已经大不相同, 特别是其机动时间已大幅缩水. 显然, 若在此时进行调度优化, 运用已知的时间参数必然导致错误的甚至极差的结果, 必须用新的方法重新计算相应的时间参数, 才能保证预期的调度效果. 上述反应性调度本身属于项目调度问题, 但是对于带有 GPRs 的项目而言, 工序机动时间的隐性消耗可能使得现有项目调度理论和方法难以保证将其有效解决, 因此需要引起重视.

总体来说, GPRs 下机动时间的隐性消耗反映了现有时间参数观念和计算方法依然存在缺陷, 阻碍项目管理理论的发展, 并在实际中产生严重的负面效益. 本文将深入研究“机动时间隐性消耗”这一奇异现象, 分析其产生的机理和条件, 给出量化方法, 揭示机动时间的新特性和规律, 为更有效的带有 GPRs 的项目管理提供理论依据和方法指导.

1 GPRs 及其时间参数

GPRs 包含项目中工序之间结束 - 开始 (FTS)、结束 - 结束 (FTF)、开始 - 开始 (STS) 和开始 - 结束 (STF) 型最小时距和最大时距^[6]. 文献 [7] 提出的 GPRs 网络是当前国际上通用的表示 GPRs 的方法, 该图是 AoA (activity - on - arc, 也称双代号) 网络. 例如, 图 1 所示的就是这样一个 GPRs 网络.

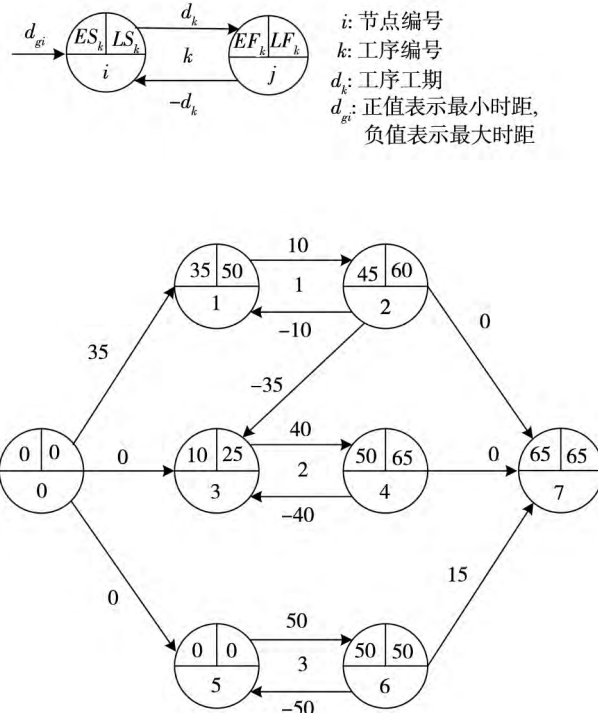


图 1 GPRs 网络

Fig. 1 Example of an activity network under GPRs

GPRs 网络的时间参数主要包括已广泛使用的经典时间参数和新提出的隐性时间参数. 文献 [7] 给出了经典时间参数及其计算公式, 主要包括各工序 k 的最早开始时间 ES_k 、最早结束时间

EF_k 、最迟开始时间 LS_k 、最迟结束时间 LF_k 、总时差 TF_k 、自由时差 FF_k 和安全时差 SF_k . 而文献 [11] 提出了 GPRs 条件下工序的隐性时间参数及其计算方法, 包括工序 k 的隐性最早开始时间 \widetilde{ES}_k 、隐性最迟结束时间 \widetilde{LF}_k 和隐性时差 \widetilde{TF}_k , 分别表示当该工序的工期延长时, 其开始时间可能早于经典最早开始时间的量, 结束时间可能迟于经典最迟结束时间的量, 以及总时差可能大于经典总时差的量.

2 机动时间的隐性消耗现象

传统观念中, 对于任意工序 k , 如果所有其他工序都没有使用各自的机动时间, 并且该工序 k 还没有开始, 那么它的机动时间理应不会受影响, 更不会被消耗. 该结论在 CPM 网络中已得到充分验证, 并且所有基于该工序网络的经典项目调度理论和方法 (如资源限制项目调度、时间 - 费用权衡、资源均衡) 等, 均以该结论为基本前提和依据^[2, 3]. (注: 若项目中工序之间只存在零时滞结束 - 开始型优先关系, 即 GPRs 中的一类特殊优先关系, 可用不同于 GPRs 网络的更为简单的网络表示^[2], 其各工序的时间参数可用关键路线法 (简称 CPM) 计算, 因此有文献也称该网络为 CPM 网络^[27]).

但是本文发现, 在 GPRs 网络中, 即使在上述情况下, 工序 k 的机动时间也可能被消耗. 该现象称之为 GPRs 下机动时间的隐性消耗, 现列举以下两例辅助以阐述.

2.1 现象 1

经典总时差表示, 工序在不影响总工期条件下, 其开始时间的最大延迟量. 以图 1 所示 GPRs 网络为例, 图 2 所示甘特图表示各工序都在其最早时间开始. 根据文献 [7], 工序 1 的经典总时差为

$$TF_1 = LS_1 - ES_1 = 50 - 35 = 15$$

说明该工序的开始时间最多可以延迟 15 天而不影响总工期. 根据图 2, 假设该项目的当前进度为 10 天, 即运行到了第 10 天, 则工序 2 已开始, 而工序 1 还未开始, 并且所有工序都未使用自身机动时间. 按照传统观念, 此时计划延迟工序 1 的开

始时间是可行的;而若将其延迟 15 天,对应的甘特图如图 3 所示.该图显示,为了满足工序 1 和工序 2 之间的优先关系,工序 2 的开始时间也将不得不延迟 15 天,即延迟到第 25 天.但是由于工序 2 此时已经开始,其开始时间不能再延迟,所以,将工序 1 的开始时间延迟 15 天实际上不可行.这意味着该工序的经典总时差此时已少于 15 天,相当于被隐性消耗了.进一步验证了,一旦工序 2 开始,工序 1 的开始时间甚至 1 天都不能延迟,相当于原先 15 天的经典总时差都被隐性消耗了.

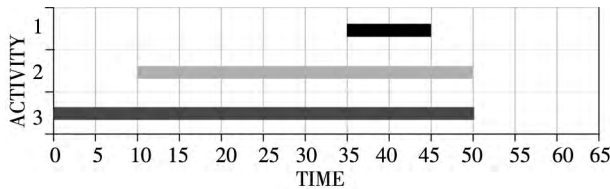


图 2 图 1 例子的甘特图

Fig. 2 Gantt chart of the example in Fig. 1

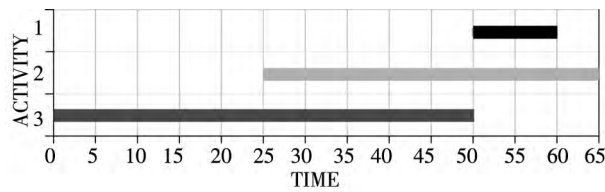


图 3 将工序 1 延迟 15 天的甘特图

Fig. 3 Gantt chart after delaying the activity 1 by 15

2.2 现象 2

根据隐性时差的概念,拥有隐性时差的工序的工期延长量即使超过经典总时差,也不会影响总工期,并且也满足任何工序的时间约束条件.例如如图 4 中的工序 1,根据文献 [7, 11],其经典总

时差和隐性时差为 $TF_1 = 30$ 天, $\widetilde{TF}_1 = 10$ 天.根据定义表明在不影响项目总工期的条件下,该工序的工期最多可延长 $TF_1 + \widetilde{TF}_1 = 30 + 10 = 40$ 天.假设该项目运行到了第 5 天,且工序 2 和工序 3 都在其最早开始时间开始,显然工序 1 此时尚未使用其机动时间.但如果此时将该工序的工期延长 40 天,如图 5 所示,总工期却延迟了 5 天,与上述结论相悖.不仅如此,为了保证工序之间的优先关系,工序 3 的开始时间也不得不延迟 25 天,这也与“工序 3 在其最早开始时间即 0 天开始”相悖.经验证明,如果要想使总工期不推迟,且满足任何工序的时间约束条件,工序 1 的工期最多只能延长 20 天,如图 6 所示.该现象表明,在项目

运行到第 5 天时,工序 1 虽未使用机动时间,但是不仅其经典总时差被隐性消耗掉 10 天,其隐性时差同样消失,被隐性消耗掉了 10 天.“隐性时差被隐性消耗”的现象,意味着 GPRs 网络可能隐藏着更复杂的结构特征.

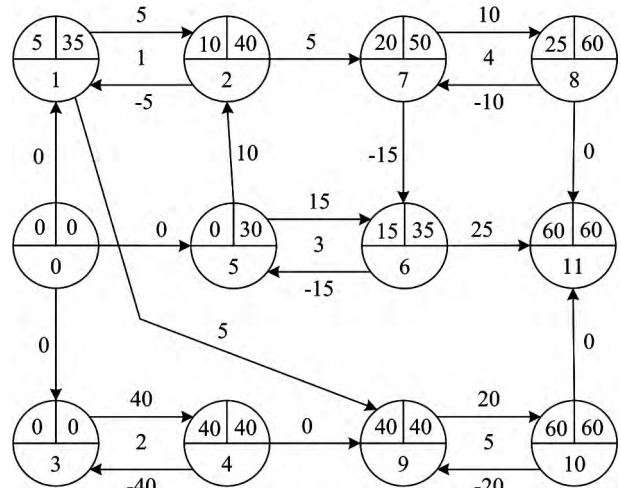


图 4 GPRs 网络

Fig. 4 Example of an activity network under GPRs

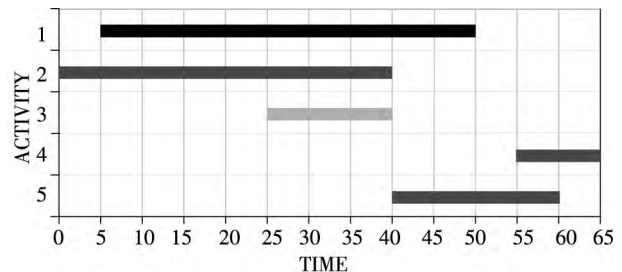


图 5 将工序 1 的工期延长 40 天的甘特图

Fig. 5 Gantt chart after prolonging the duration of the activity 1 by 40

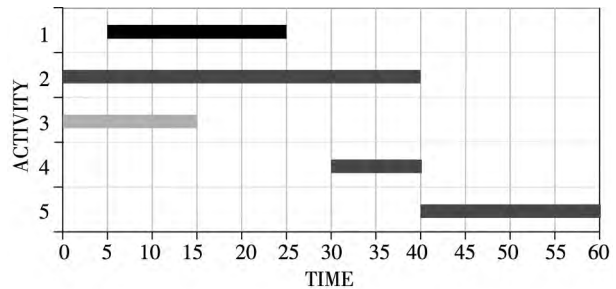


图 6 将工序 6 的工期延长 20 天的甘特图

Fig. 6 Gantt chart after prolonging the duration of the activity 1 by 20

3 机动时间隐性消耗的机理分析

经验证明, GPRs 网络中出现的“机动时间隐性消耗的现象”,在经典 CPM 网络中却不会出现.

因此, 通过对比分析这两类工序网络的差异来进行上述现象的机理分析.

通过对比 GPRs 网络和 CPM 网络的绘制方法和结构可以发现, 两者之间最明显的差异在于, GPRs 网络中可存在负弧(其长度为负数, 用于表示工序之间的最大时距). 这是导致 GPRs 下机动时间隐性消耗的主要原因所在.

GPRs 网络中, 某长度为负的弧(或路线)从工序 i 指向工序 j 的情况如图 7 所示($\delta < 0$, $-\delta$ 表示工序间的时距). 根据图 7(a) ~ 图 7(d) 所示情况, 假设各工序工期不变, GPRs 下工序经典机动时间出现隐性消耗的充分必要条件可归纳为定理 1:

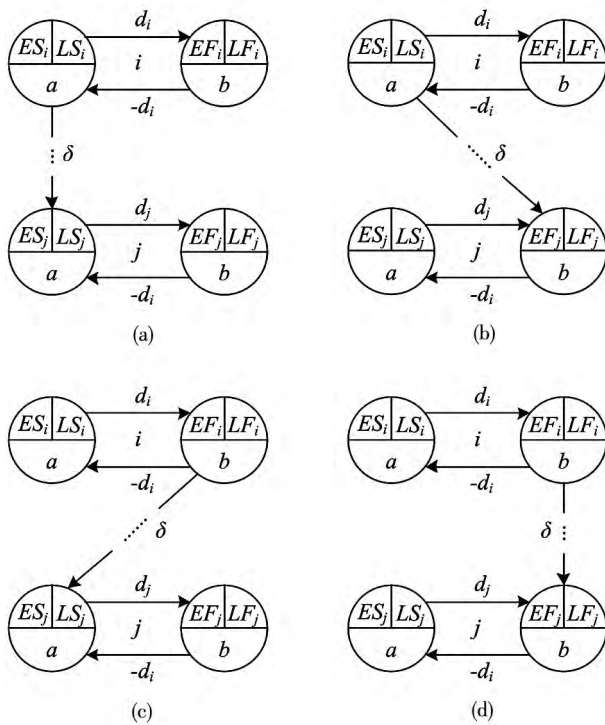


图 7 负弧连接工序 i 与 j 的情况汇总

Fig. 7 Cases of an arc with negative length connecting activities i and j

定理 1 GPRs 网络中, 当项目进行到时间 T , 且已执行的各工序并未使用各自机动时间, 对于尚未开始的非关键工序 i , 即 $T < ES_i$, 其经典机动时间出现隐性消耗的充要条件是: 图 7 中工序 j 是非关键工序, $ES_j \leq T < ES_i$, 且满足以下条件之一:

- (1) 对于图 7(a), $ES_j - \delta < LS_i$;
- (2) 对于图 7(b), $EF_j - \delta < LS_i$;
- (3) 对于图 7(c), $ES_j - \delta < LF_i$;

(4) 对于图 7(d), $EF_j - \delta < LF_i$.

证明 (1) 充分性证明.

根据 AoA 网络表示 GPRs 的方法, 若工序间存在最大时距, 则在 GPRs 网络中用负弧表示. 将视角从弧扩展至路线, 根据题设, 分别考虑图 7(a) ~ 图 7(d) 所示的负弧(路线)从工序 i 指向工序 j 的情况. 为便于表述及遵循惯例, 将工序 j 称为 i 的图示后继工序, 工序 i 称为 j 的图示前继工序.

1) 图 7(a) 所示情况, 负弧(路线)从工序 i 的开始节点指向工序 j 的开始节点, 表示工序 j 开始后最多 $-\delta$ 单位时间(如: 天), 工序 i 必须开始. 可用下式表示如下(s_i 表示工序 i 的开始时间)

$$s_j - \delta > s_i \tag{1}$$

根据 GPRs 网络中时间参数的计算方法^[7]

$$ES_j \geq ES_i - (-\delta)$$

因此, $ES_j \geq ES_i$ 和 $ES_j < ES_i$ 的情况均可能存在, 即具有 $ES_j \leq s_j \leq T < ES_i$ 的可能性.

根据文献[7]

$$TF_i = LS_i - ES_i = LF_i - EF_i \tag{2}$$

若工序 i 是非关键工序, 即经典总时差 $TF_i > 0$, 则 TF_i 取决于其最迟开始时间 LS_i . 亦根据文献[7]

$$LS_i = \min\{S_{i1}, S_{i2}, S_{i3}, S_{i4}\} \tag{3}$$

式中

$$S_{i1} = \min_k\{LS_k - \delta_{ik}\} \tag{4}$$

表示存在长度为 δ_{ik} 的弧(路线)从工序 i 的开始节点指向工序 k 的开始节点

$$S_{i2} = \min_k\{LF_k - \delta_{ik}\} \tag{5}$$

表示存在长度为 δ_{ik} 的弧(路线)从工序 i 的开始节点指向工序 k 的结束节点

$$S_{i3} = \min_k\{LS_k - \delta_{ik}\} - d_i \tag{6}$$

表示存在长度为 δ_{ik} 的弧(路线)从工序 i 的结束节点指向工序 k 的开始节点

$$S_{i4} = \min_k\{LF_k - \delta_{ik}\} - d_i \tag{7}$$

表示存在长度为 δ_{ik} 的弧(路线)从工序 i 的结束节点指向工序 k 的结束节点. 由于图 7(a) 所示的工序 j 与式(4)中的工序 k 具有一致性, 即 $k := j$, $\delta_{ik} := \delta_{ij} = \delta < 0$, 因此工序 j 的最迟开始时间 LS_j 的变化有可能会影响工序 i 的最迟开始时间 LS_i , 进而根据影响工序 i 的总时差 TF_i .

根据式(3)和式(4)若 LS_j 减小为 LS'_j , 使得 $LS'_j - \delta < LS_i$, 则 $S'_i = LS'_j - \delta < LS_i$, 所以

$$LS'_i = \min\{S'_i, S_{i2}, S_{i3}, S_{i4}\} = S'_i < LS_i \quad (8)$$

根据式(2)

$$TF'_i = LS'_i - ES_i < LS_i - ES_i = TF_i \quad (9)$$

即 TF_i 减小.

据上分析, 若 $ES_j \leq s_j < LS_j$ 且 $s_j \leq T < ES_i$, 说明工序 j 是非关键工序, 可先于工序 i 开始, 当项目进行到时间 T 时, 工序 j 已开始, 而其图示前继工序 i 尚未开始. 根据假设, 已执行的工序均未消耗各自机动时间, 因此可知 $s_j = ES_j$. 由于此时工序 j 已开始, 其开始时间必然不能再推迟, 因此也可认为 $s_j := ES_j$ 是该工序的新最迟开始时间 LS'_j , 即 LS_j 减小为 $LS'_j = ES_j$. 如果 $ES_j - \delta < LS_i$, 即 $LS'_j - \delta < LS_i$ 根据式(8), $LS'_i < LS_i$ 即导致了工序 i 的最迟开始时间 LS_i 减小为 LS'_i . 再根据式(9) $TF'_i < TF_i$ 进而使得工序 i 的经典总时差 TF_i 的减小 TF'_i 表现为第 2.1 节所描述的隐性消耗.

综上所述, 对于图 7(a) 所示情况, 当项目进行到时间 T , 且已执行的工序均未消耗各自机动时间, 则非关键工序 i 出现工序机动时间隐性消耗的充分条件包括: $ES_j \leq T < ES_i$, 工序 j 是非关键工序, 且 $ES_j - \delta < LS_i$.

2) 图 7(b) 所示情况, 负弧(路线)从工序 i 的开始节点指向工序 j 的结束节点, 表示工序 j 结束后最多 $-\delta$ 单位时间, 工序 i 必须开始, 可用式(10)表示如下(f_j 表示工序 j 的结束时间)

$$f_j - \delta > s_i \quad (10)$$

对式(10)转化

$$f_j - \delta > s_i \Leftrightarrow (s_j + d_j) - \delta > s_i \\ \Leftrightarrow s_j - (\delta - d_j) > s_i$$

与式(1)形同, 对应地, 图 7(b) 可转化为图 8, 与图 7(a) 形同. 因此根据上述(1)的证明过程可得, 对于图 7(b) 所示情况, 当项目进行到时间 T , 且已执行的工序均未消耗各自机动时间, 则非关键工序 i 出现工序机动时间隐性消耗的充分条件包括 $ES_j \leq T < ES_i$, 工序 j 是非关键工序, 且

$$ES_j + (r + d_j) < LS_i \Leftrightarrow (ES_j + d_j) + r < LS_i \\ \Leftrightarrow EF_j + r < LS_i$$

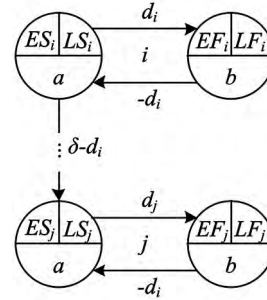


图 8 图 7(b) 的等效转化

Fig. 8 An equivalent transformation of Fig. 7(b)

3) 类似地, 可以证明, 对于图 7(c) 和图 7(d) 所示情况, 定理所示的充分条件正确.

(2) 必要性证明.

假设当项目运行到时间 T , 已执行的工序均未使用各自机动时间. 如果此时某尚未开始的非关键工序 i (即 $T < ES_i$) 的总时差 TF_i 出现隐性消耗, 根据式(2), 必然是其 ES_i 增大或 LS_i 减小. 由于各工序均未使用各自机动时间, 因此排除 ES_i 增大的可能, 只需考虑 LS_i 减小的可能.

根据式(3)~式(7), 由于各工序及项目工期均不变, 因此若 LS_i 减小, 则必然源于工序 i 的图示后继非关键工序的 LS 的减小; 而 LS 最先减小, 并进而造成其他工序(包括工序 i) 的 LS 减小的非关键工序 j , 其 LS_j 减小的原因只有一个: 当项目进行到时间 T 时, 工序 j 已开始, 即 $s_j = ES_j \leq T$, 故其开始时间无法再延迟, 可视为 $LS_j := s_j = ES_j$ 可知, 由于工序 j 是 i 的图示后继工序, 即存在从工序 i 指向 j 的弧(路线), 而 $ES_j \leq T < ES_i$, 根据时间参数计算方法^[7], 该弧(路线)的长度 δ 满足 $ES_j \geq ES_i + \delta$, $ES_j < ES_i$ 所以 $\delta < 0$.

如果从工序 i 指向 j 的弧(路线)形如图 7(a) 所示, 根据式(3)和式(4), 若 LS_i 受 $LS_j \rightarrow LS'_j = s_j = ES_j$ 的影响而减小, 则其必然满足

$$LS_j - \delta < LS_i \Rightarrow ES_j - \delta < LS_i$$

因此, 定理 1 “工序 j 是非关键工序, $ES_j \leq T < ES_i$, 且 $ES_j - \delta < LS_i$ ”是必要条件.

同理可证, 定理 1 “工序 j 是非关键工序, $ES_j \leq T < ES_i$, 且 + 条件(2) 或(3) 或(4)”同样是工序 i 的经典机动时间隐性消耗的必要条件. 证毕.

进一步地, GPRs 下工序隐性机动时间出现隐性消耗的充分条件之一可归纳为定理 2:

定理 2 对于带有 GPRs 的项目中的工序 i ,

当项目进行到时间 T ($T < ES_i$) 时, 如果该工序 i 的隐性最早开始时间 $\widetilde{ES}_i < T$, 则其隐性机动时间会被隐性消耗.

证明 根据文献[11], 工序 i 的隐性机动时间取决于其隐性最早开始时间 \widetilde{ES}_i 与经典最早开始时间 ES_i 相比提前的量. 当项目运行到时刻 T , 若 $\widetilde{ES}_i < T < ES_i$, 说明工序 i 虽尚未开始, 但其开始时间最多只能提前到 T 而非 \widetilde{ES}_i , 意味着其隐性机动时间减少了 $T - \widetilde{ES}_i$, 即出现隐性消耗. 证毕.

根据定理 1 和定理 2, 得出有关工序隐性机动时间的隐性消耗的推论 1:

推论 1 GPRs 网络中, 如果某负弧(路线)连接两个工序 i 和 j , 并且 $ES_j < ES_i$, 则当项目进行到时间 $T \in [ES_j, ES_i)$ 时, 可能出现工序隐性机动时间的隐性消耗.

证明 与定理 1 的情况和证明类似, 根据文献[11], 工序 i 的隐性时差 \widetilde{TF}_i 同样取决于其图示后继工序 j (如图 7 所示) 的最迟时间 LS_j 或 LF_j , 因此, 工序 j 早于其图示前继工序 i 开始, 也可能会导致工序 i 的隐性机动时间的隐性消耗. 证毕.

2.1 节和 2.2 节分别列举了工序的经典机动时间和隐性机动时间的隐性消耗的例子. 根据这两类机动时间的差别, 将分别研究两者的隐性消耗.

4 经典机动时间的隐性消耗的计算方法

4.1 算法描述

根据文献[7, 11], GPRs 网络中工序的经典总时差对确定工序开始时间的可延迟量有效, 而对工序工期的可延长量无效. 若某工序已经开始, 则该工序的开始时间自然无法再延迟, 也不需要考虑其经典机动时间的情况. 因此, 当项目实施到时刻 T 时, 需要考虑的是在该时刻还未开始的工序 k 的机动时间情况(这里主要考虑总时差).

假设 GPRs 网络中, 工序 i 和 j 由负弧连接, 即工序 j 是工序 i 的图示紧后工序, 且 $ES_j < ES_i$. 工序 j 的开始时间为 s_j , $ES_j \leq s_j < ES_i$, 当项目进行到 $T \in [s_j, ES_i)$ 时, 未开始的工序 k 的经典总时差的隐性消耗的计算方法如下:

步骤 1 计算 T 时刻未开始的工序 k 的经典总时差 TF_k .

步骤 2 令 $ES'_j = s_j$, $LS'_j = s_j$, 计算工序 k 的经典时间参数 ES'_k 、 EF'_k 、 LS'_k 、 LF'_k 和 TF'_k .

步骤 3 计算工序 k 的经典总时差的隐性消耗 ΔTF_k

$$\Delta TF_k = TF_k - TF'_k \quad (11)$$

4.2 算法正确性分析

任何工序一旦开始, 其开始时间必然不能再变动, 所以如果工序 j 在某时刻 s_j 已经开始, 且 $ES_j \leq s_j < ES_i$, 则可视为其最早和最迟开始时间均被赋予新值 s_j , 即 $ES'_j = s_j$, $LS'_j = s_j$.

设项目此时运行到时间 T , $s_j < T < ES_i$. 根据假设条件, 工序 i 是工序 j 的图示紧前工序, 则在该时间 T , 工序 j 已开始, 但其图示紧前工序 i 却未开始. 根据文献[7], 工序 i 的最迟开始时间 LS_i 可能会受工序 j 的最迟开始时间 LS'_j 的影响而改变. 若 $LS'_j < LS_j$, 则可能导致工序 i 的最迟开始时间 $LS'_i < LS_i$. 而若 $LS'_i < LS_i$ 为真, 则根据文献[7], 工序 i 的经典总时差 TF_i 变化如下

$$TF'_i = LS'_i - ES_i < LS_i - ES_i = TF_i$$

即该工序的经典总时差减少了 $\Delta TF_i = TF_i - TF'_i$, 式(11)正确. 由于工序 i 实际并未开始, 也未消耗总时差, 因此该部分总时差的消耗 ΔTF_i 就是隐性消耗. 另外, LS_i 的减小(即 LS'_i)也可能导致其他未开始工序 k 的 LS_k 减小, 从而也产生机动时间的隐性消耗 ΔTF_k . 4.1 节算法正确.

5 隐性机动时间的隐性消耗的计算方法

5.1 算法描述

如果工序 k 有隐性时差, 那么该工序同样可能出现隐性时差的隐性消耗. 根据文献[11], 工序 k 的隐性时差主要产生于该工序的工期延长的情况下, 决定其最大延长量. 与第 4 节所述的经典

机动时间的隐性消耗不同,即使工序 k 已开始,但只要还未结束,其工期就可以延长.因此,当项目实施到某时刻 T 时,需要考虑的是还未结束的工序的隐性时差的隐性消耗.

由于工序的隐性时差包括前端隐性时差和后端隐性时差^[11],并且根据第3节,这两部分时差的隐性消耗机理不同,因此分别进行研究.

5.1.1 前端隐性时差的隐性消耗的计算方法

当项目实施到时间 T 且 $T < ES_k$,即从经典时间参数的角度看,工序 k 尚未开始,但该工序的前端隐性时差可能出现隐性消耗,计算方法如下:

步骤1 计算工序 k 隐性最早开始时间 \widetilde{ES}_k .

步骤2 如果 $\widetilde{ES}_k < T$,计算工序 k 的前端隐性总时差的隐性消耗 $\Delta^* \widetilde{TF}_k$

$$\Delta^* \widetilde{TF}_k = T - \widetilde{ES}_k \tag{12}$$

显然,如果 $T \geq ES_k$,则工序 k 的前端隐性时差被全部隐性消耗,即

$$\Delta^* \widetilde{TF}_k = \widetilde{TF}_k \tag{13}$$

5.1.2 后端隐性时差的隐性消耗的计算方法

当项目实施到时间 T ,工序 k 不论开始与否,只要未结束,其后端隐性时差均可发生隐性消耗.

与4.1节的假设相同,设GPRs网络中,工序 i 和 j 之间是由负弧相连,且 $ES_j < ES_i$.工序 j 的开始时间为 s_j , $ES_j \leq s_j < ES_i$,当项目进行到 $T \in [s_j, ES_i)$ 时,未结束的工序 k 的后端隐性时差的隐性消耗的计算方法如下:

步骤1 计算工序 k 的后端隐性时差 \widetilde{TF}_k^* .

步骤2 令 $ES_{j'} = s_j, LS_{j'} = s_j$,计算工序 k

的经典时间参数 LF_k^* 和隐性时间参数 \widetilde{LF}_k^* 和 \widetilde{TF}_k^* .

步骤3 计算工序 k 的后端隐性时差的隐性消耗 $\Delta \widetilde{TF}_k^*$

$$\Delta \widetilde{TF}_k^* = \widetilde{TF}_k^* - TF_k^* \tag{14}$$

结合5.1.1节,工序 k 在时刻 T 的隐性时差的隐性消耗 $\Delta \widetilde{TF}_k$ 为

$$\Delta \widetilde{TF}_k = \Delta^* \widetilde{TF}_k + \Delta \widetilde{TF}_k^* \tag{15}$$

5.2 算法正确性分析

根据文献[11],GPRs下工序 k 的隐性时差、前端隐性时差和后端隐性时差计算如下

$$\begin{aligned} \widetilde{TF}_k &= \widetilde{TF}_k^* + \widetilde{TF}_k^* \\ \widetilde{TF}_k^* &= ES_k - \widetilde{ES}_k \\ \widetilde{TF}_k^* &= \widetilde{LF}_k - LF_k \end{aligned} \tag{16}$$

工序 k 的隐性最早开始时间 \widetilde{ES}_k 决定其前端隐性时差, $\widetilde{ES}_k < ES_k$,而隐性最迟结束时间 \widetilde{LF}_k 决定其后端隐性时差, $\widetilde{LF}_k > LF_k$.

先考虑工序 k 的前端隐性时差.当项目运行到时间 T 且 $\widetilde{ES}_k < T < ES_k$,说明虽然工序 k 尚未开始,但是其隐性开始时间已经被推迟,即此时其隐性开始时间最早只能为 $\widetilde{ES}_k' = T > \widetilde{ES}_k$,根据文献[11]可知

$$\widetilde{TF}_k' = ES_k - \widetilde{ES}_k' < ES_k - \widetilde{ES}_k = \widetilde{TF}_k^*$$

所以工序 k 的前端隐性时差已经被消耗 $\Delta^* \widetilde{TF}_k$

$$\Delta^* \widetilde{TF}_k = \widetilde{ES}_k' - \widetilde{ES}_k = T - \widetilde{ES}_k$$

式(12)正确.而如果 $ES_k < T$,说明工序 k 已经开始,不能再提前,因此不会再有前端隐性时差,可视为 \widetilde{TF}_k^* 已被全部隐性消耗,式(13)正确.所以5.1.1节算法正确.

再考虑工序 k 的后端隐性时差.与4.2节算法的正确性分析类似,当项目实施到时间 T ,某些在 T 时刻已经开始了的工序的“计划最迟开始时间”已被实际开始时间所替代,而在GPRs网络中负弧的影响下,该变化可能会导致其他工序的最迟时间也发生变化.根据文献[11],工序 k 的后端隐性时差是由自身及其图示后继工序的“最迟时间”决定的,因此这些“最迟时间”的变化可能会导致该工序 k 的后端隐性时差 \widetilde{TF}_k^* 的变化,通过对比原先的 \widetilde{TF}_k^* 和当前的 \widetilde{TF}_k^* ,可得出隐性

消耗 ΔTF_k^* , 即 $\Delta TF_k^* = TF_k^* - TF_k^*$ 式 (14) 正确.

工序 k 的前后端隐性时差的隐性消耗的总和 就是该工序隐性时差的隐性消耗, 式 (15) 正确. 所以, 5.1.2 节算法正确.

6 应用举例

6.1 案例描述

根据第 3 节的机理分析, 机动时间隐性消耗的最主要原因之一是 GPRs 网络中存在用于表示最大时距的负弧, 而在带有 GPRs 的现代工程项目中, 工序之间的最大时距广泛存在, 因此在项目调度中, 随时可能受到工序机动时间隐性消耗的影响.

该节主要考虑工序机动时间隐性消耗对带有 GPRs 的反应性资源均衡的影响. 以某公路建设工程项目为例. 该工程的一部分 (子项目) 是修筑一段长 6.3 km 的公路, 从简化的角度考虑, 其结构分为 3 个结构层: 基层 (挖沟、筑堤、砂石铺路等, 记为 A)、底层 (水泥浇筑等, 记为 B)、面层 (路

面修筑等, 记为 C). 3 层都采用线性流水施工, 其中底层 B 的施工速度最快. 鉴于居住、施工等多方面因素考虑, 施工单位决定将公路分 3 段施工, 每段 2.1 km, 即 3 层分别为 $A_k, B_k, C_k, k = 1, 2, 3$, 且根据施工速度 A_k, B_k, C_k 的工期分别为 78 h、70 h、105 h. 由于 B_k 需要在 A_k 上作业, 并且 B_k 的施工速度快于 A_k 的施工速度, 因此为了保障 B_k 所需的施工面, 计划 A_k 的结束时间至少要比 B_k 的结束时间早 7 h; 同理, 由于 C_k 必须在 B_k 上作业, 因此计划在 B_k 开始后至少 12 h, C_k 才能开始. 另外, 由于 B 有水泥浇筑作业, 因此 B_k 和 B_{k+1} 之间的时间间隔不能超过 12 h.

根据上述分析, 以及各作业 (工序) 之间的优先关系, 可得该子项目的 GPRs 网络, 如图 9 所示. 假设该项目在运行初期可用资源充足, 期间各工序均可在其最早时间开始, 但当项目进行到时间 94 h, 出现了意料之外的资源限制情况, 届时需要及时制定资源均衡方案并于此时开始执行, 即进行反应性资源均衡. (注: 各工序的工期不变, 即确保施工速度不变, 只可调整工序的开始时间.)

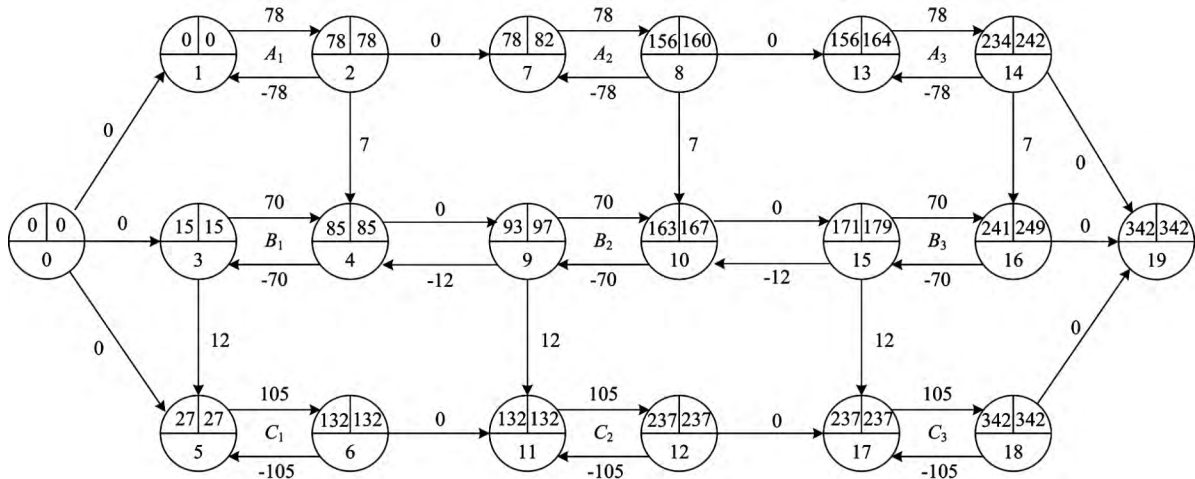


图 9 第 6 节案例的 GPRs 网络

Fig. 9 Activity network under GPRs of the illustration in section 6

6.2 理论模型

反应性资源均衡问题是资源均衡问题的一个类型, 可以借助资源均衡模型来求解. Rieck 等^[4] 针对带有 GPRs 的资源均衡问题, 通过拓展 Easa 模型, 建立了更为高效的混合整数线性规划模型, 如下:

假设各工序的工期不变, 且项目总工期不变,

而通过调整工序的开始时间实现资源均衡效果. 令 M 表示项目中所有工序的集合, M_{STS} 、 M_{STF} 、 M_{FTS} 和 M_{FTF} 分别表示由“开始-开始”型、“开始-结束”型、“结束-开始”型和“结束-结束”型最小和最大时距相关联的工序的集合, \bar{d} 表示项目总工期, \mathfrak{R} 表示总体资源, c_k 和 Y_k 分别表示资源 $k \in \mathfrak{R}$ 的单位费用和阈值, r_{ik} 表示工序 i 对资源

$k \in \mathfrak{R}$ 的需求量.

设置 0-1 变量 x_{iq} : 若工序 i 的开始为 $ES_i + q$, 令 $x_{iq} = 1$; 否则, 令 $x_{iq} = 0$. 设置辅助变量 z_{ikt} 和 z_{kt} , z_{ikt} 表示工序 i 在时间 t 对资源 $k \in \mathfrak{R}$ 的需求量, 而 z_{kt} 表示资源 $k \in \mathfrak{R}$ 在时间 t 的总需求量.

Rieck 资源均衡模型

$$\begin{aligned} & \min \sum_{k \in \mathfrak{R}} c_k \sum_{t=0}^{\bar{d}-1} z_{kt} \\ & \text{s. t.} \\ & \sum_{q=1}^{TF_i} x_{iq} \leq 1, i \in M \\ & ES_j - ES_i + \sum_{q=1}^{TF_j} qx_{jq} - \sum_{q=1}^{TF_i} qx_{iq} \geq \delta_{ij}, i, j \in M_{STS} \\ & EF_j - ES_i + \sum_{q=1}^{TF_j} qx_{jq} - \sum_{q=1}^{TF_i} qx_{iq} \geq \delta_{ij}, i, j \in M_{STF} \\ & ES_j - EF_i + \sum_{q=1}^{TF_j} qx_{jq} - \sum_{q=1}^{TF_i} qx_{iq} \geq \delta_{ij}, i, j \in M_{FTS} \\ & EF_j - EF_i + \sum_{q=1}^{TF_j} qx_{jq} - \sum_{q=1}^{TF_i} qx_{iq} \geq \delta_{ij}, i, j \in M_{FTF} \\ & z_{ikt} \geq \left(1 - \sum_{q=1}^{TF_i} qx_{iq}\right) r_{ik} + \sum_{q=1}^{\min\{t-ES_i, TF_i\}} x_{iq} r_{ik}, i \in M, \\ & k \in \mathfrak{R}, ES_i \leq t < ES_i + d_i \\ & z_{ikt} \geq \sum_{q=1+\min\{t-ES_i-d_i, TF_i\}}^{\min\{t-ES_i, TF_i\}} x_{iq} r_{ik}, i \in M, k \in \mathfrak{R}, \\ & ES_i + d_i \leq t < ES_i + d_i + TF_i \end{aligned}$$

$$\begin{aligned} z_{kt} & \geq \sum_{i \in V} z_{ikt} - Y_k, k \in \mathfrak{R}, t \in T \setminus \{\bar{d}\} \\ x_{iq} & \in \{0, 1\}, i \in M, q = \{1, \dots, TF_i\} \end{aligned}$$

注: 文献 [4] 的模型将所有优先关系均转化为“开始-开始”型时距, 而本文将其均还原为初始优先关系; δ_{ij} 在此只表示 GPRs 网络中从工序 i 指向工序 j 的弧的长度

6.3 结果分析

本节运用 6.2 节 Rieck 模型求解 6.1 节案例. 简便起见, 假设本例中 $c = 1, Y = 1, r_i = 1$. 在该反应性资源均衡中, 需要调整的是开始时间大于 94h 的工序. 根据图 9, 符合该条件的工序为 C_2, A_3, B_3 和 C_3 , 其时间参数如表 1 所示.

表 1 工序时间参数

Table 1 Time parameters of the activities

工序	ES	EF	TF
C_2	132	237	0
A_3	156	234	8
B_3	171	241	8
C_3	237	342	0

1) 运用 Rieck 模型解决该问题, 在表 1 中时间参数的基础上, 可算得该模型的最优解为

$$s_{C_2}^* = 0, s_{A_3}^* = 156, s_{B_3}^* = 179, s_{C_3}^* = 0$$

对应的方案效果示意图如图 10 所示. 图 10 中, 工序 B_2 的开始时间为 97, 但由于 $ES_{B_2} = 93$, 即当项目进行到时间 94, 工序 B_2 已开始, 其开始时间不能再推迟到 97, 因此上述基于表 1 时间参数的结果是错误的. 该例验证了, 根据现有经典时间参数解决带有 GPRs 的反应性项目调度问题, 难以保证结果的准确性. 下面对上述错误进行分析和修正.

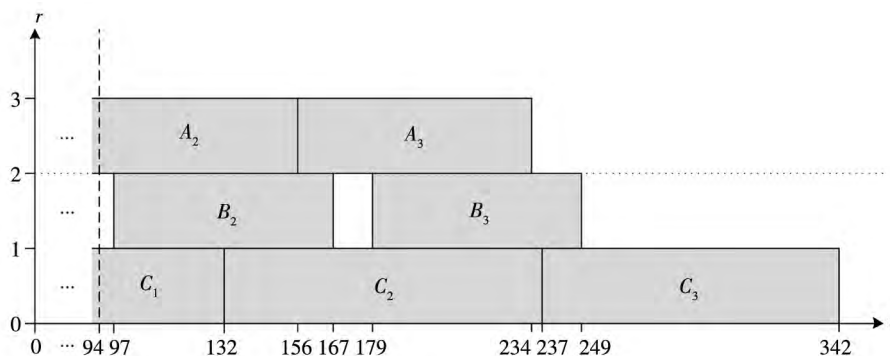


图 10 基于经典机动时间的资源均衡示意图

Fig. 10 Form of resource profile after levelling based on the values of classical time floats

2) 根据图 9, 有负弧 (15, 10) 从工序 B_3 指向 B_2 符合图 7(b) 所示情况, 并且工序 B_2 已经开始 $EF_{B_2} - \delta_{15, 10} = 163 - (-12) = 175 < LS_{B_3} = 179$ 满足定理 1 充分条件 (2), 因此, 会出现工序经典机动时间的隐形消耗.

运用 5.1 节算法, 当项目进行到 $T = 94$ h, 由于工序没有使用机动时间, 所以 B_2 已开始, 即 $s_{B_2} = ES_{B_2} = 93$ h, 而 B_2 的紧前工序 B_3 尚未开始. 令 $LS_{B_2} = s_{B_2} = 93$ h 根据文献 [7], 计算各工序的经典时间参数, 如图 11 所示, 进而计算尚未开始的工序的经典总时差, 得 $TF_{A_3}^* = 4$, $TF_{B_3}^* = 4$ 而 A_3 和 B_3 原先的经典总时差均为 8h, 说明其存在隐形消耗, 且消耗量均为 4h.

根据题设, 各工序的工期不会变化, 因此不需要考虑隐性机动时间及其隐性消耗.

3) 根据 2) 的结论, 当项目进行到 $T = 94$ h, 在充

分考虑了工序机动时间的隐性消耗后, 此时工序 C_2 、 A_3 、 B_3 和 C_3 的实际时间参数如表 2 所示.

表 2 $T = 94$ h 时工序的实际时间参数

Table 2 Time parameters of the activities in $T = 94$ h

工序	ES	EF	TF
C_2	132	237	0
A_3	156	234	4
B_3	171	241	4
C_3	237	342	0

将表 2 时间参数代入 Rieck 模型, 算得最优解如下 $s_{C_2}^* = 0$, $s_{A_3}^* = 156$, $s_{B_3}^* = 175$, $s_{C_3}^* = 0$ 对应的方案效果示意图如图 12 所示. 经验表明, 所有工序均满足既定的优先关系, 并且工序 B_2 的开始时间为 93, 等于其最早开始时间 ES_{B_2} , 符合题设条件, 是真实的最优解. 该案例验证了, 对于带有 GPRs 的反应性项目调度问题, 如果缺少考虑工序机动时间的隐形消耗, 很可能导致错误的结果.

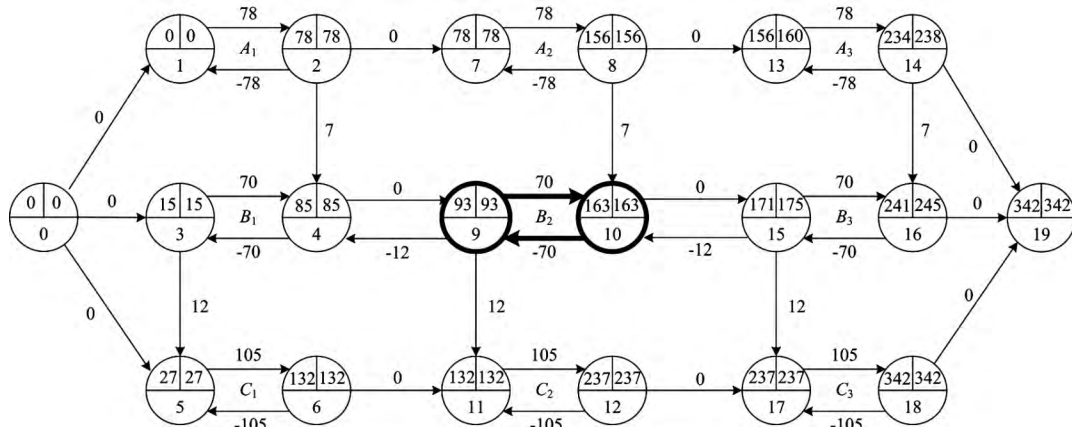


图 11 $T = 94$ h 时未完成的工序的时间参数值

Fig. 11 Time parameters of unfinished activities in $T = 94$ h

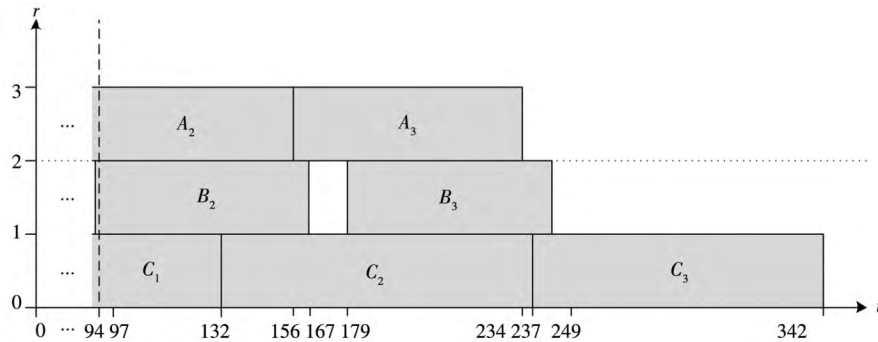


图 12 基于机动时间隐性消耗的资源均衡示意图

Fig. 12 Form of resource profile after levelling based on the hidden consumptions of time floats

7 结束语

本文发现了一些奇异现象: 工序的机动时间

存在隐性消耗, 例如, 在项目的运行过程中, 即使所有工序都没有消耗各自的机动时间, 某工序的机动时间仍会被消耗掉, 哪怕该工序尚未开始. 该

现象动摇了与机动时间相关问题(如项目调度问题)的现有理论基础,可能弱化以机动时间为基础的求解方法. 本文研究表明,机动时间隐性消耗的现象主要存在于带有 GPRs 的项目的运行过程中,在项目前期难以预料,因此,该现象对预定项目计划的实际实施在时间上(或时效性上)提出了更高的要求,并且对不确定性项目调度、反应性

项目调度等方面的影响尤为显著. 若疏忽了机动时间隐性消耗,则必然会导致错误的甚至严重的后果. 本文基于该奇异现象研究了 GPRs 下机动时间的新特性,针对工序的经典机动时间和隐性时差,分别给出了其隐性消耗现象的存在条件和量化方法,为诸如带有 GPRs 的反应性项目调度等提供了指导方向和理论依据.

参考文献:

- [1] George S J. Time—the next source of competitive advantage [J]. *Harvard Business Review*, 1988, 66(4): 41–51.
- [2] Elmaghraby S. Activity nets: A guided tour through some recent developments [J]. *European Journal of Operational Research*, 1995, 82(3): 383–408.
- [3] Mubarak S A. *Construction Project Scheduling and Control* [M]. New York: John Wiley & Sons, Inc, Hoboken, 2010.
- [4] Rieck J, Zimmermann J, Gather T. Mixed-integer linear programming for resource leveling problems [J]. *European Journal of Operational Research*, 2012, 221(1): 27–37.
- [5] Peng W L, Huang M. A critical chain project scheduling method based on a differential evolution algorithm [J]. *International Journal of Production Research*, 2014, 52(13): 3940–3949.
- [6] Roy B. Graphes et ordonnancement [J]. *Revue Francaise de Recherche Operationelle*, 1962, 25: 323–326.
- [7] Elmaghraby S E, Kamubowski J. The analysis of activity networks under generalized precedence relations [J]. *Management Science*, 1992, 38: 1245–1263.
- [8] 杨冰. 网络计划计算模型的统一 [J]. *系统工程理论与实践*, 2002, (3): 51–55.
Yang Bing. Unifying calculating models for the network planning [J]. *Systems Engineering: Theory & Practice*, 2002, (3): 51–55. (in Chinese)
- [9] 张立辉, 潘楚云, 邹鑫. 重复性项目调度模型与网络模型的转化研究 [J]. *管理科学学报*, 2014, 17(6): 49–59.
Zhang Lihui, Pan Chuyun, Zuo Xin. A method for converting repetitive scheduling models to network model [J]. *Journal of Management Sciences in China*, 2014, 17(6): 49–59. (in Chinese)
- [10] Qi J X, Su Z X. Analysis of an anomaly: The increase in time float following consumption [J]. *The Scientific World Journal*, 2014, 2014: 415870–1–12.
- [11] 苏志雄, 乞建勋, 张立辉. 广义优先关系下的隐性时间、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)
- [12] 阚芝南, 孔峰, 乞建勋. 搭接网络中的路长悖论及其特性研究 [J]. *中国管理科学*, 2014, 22(5): 121–130.
Kan Zhinan, Kong Feng, Qi Jianxun. Research on path length paradox and its characteristics under spliced network [J]. *Chinese Journal of Management Science*, 2014, 22(5): 121–130. (in Chinese)
- [13] Bianco L, Caramia M, Giordani S. Resource levelling in project scheduling with generalized precedence relationships and variable execution intensities [J]. *OR Spectrum*, 2016, 38(2): 405–425.
- [14] Pérez Á, Quintanilla S, Lino P, et al. A multi-objective approach for a project scheduling problem with due dates and temporal constraints infeasibilities [J]. *International Journal of Production Research*, 2014, 52(13): 3950–3965.
- [15] Kaveh K D, Madjid T, Abtahi A R, et al. Solving multi-mode time-cost-quality trade-off problems under generalized precedence relations [J]. *Optimization Methods and Software*, 2015, 30(5): 965–1001.
- [16] 苏志雄, 乞建勋, 阚芝南. 等效化简带有广义优先关系的时间—费用权衡问题 [J]. *运筹与管理*, 2015, 24(5): 175–183.
Su Zhixiong, Qi Jianxun, Kan Zhinan. Simplification of time-cost tradeoff problem with generalized precedence relations [J]. *Operations Research and Management Science*, 2015, 24(5): 175–183. (in Chinese)

- [17] Zhang L H , Zou X , Qi J X. A trade-off between time and cost in scheduling repetitive construction projects [J]. *Journal of Industrial and Management Optimization* , 2015 , 11(4) : 1423 – 1434.
- [18] 胡雪君, 崔南方, 赵 雁. 基于活动工期风险和资源约束风险的缓冲大小计算方法 [J]. *控制与决策* , 2016 , 31(8) : 1513 – 1518.
Hu Xuejun , Cui Nanfang , Zhao Yan. Buffer sizing method based on activity duration risk and resource constraint risk [J]. *Control and Decision* , 2016 , 31(8) : 1513 – 1518. (in Chinese)
- [19] 张宏国, 杜 雷, 谢璟琦. 不确定周期约束下项目调度算法 [J]. *计算机工程与设计* , 2011 , 32(12) : 4083 – 4086.
Zhang Hongguo , Du Lei , Xie Jingqi. Project scheduling algorithm with uncertain period-constrained [J]. *Computer Engineering and Design* , 2011 , 32(12) : 4083 – 4086. (in Chinese)
- [20] Xiong J , Chen Y W , Zhou Z B. Resilience analysis for project scheduling with renewable resource constraint and uncertain activity durations [J]. *Journal of Industrial and Management Optimization* , 2016 , 12(2) : 719 – 737.
- [21] 于 静, 徐 哲. 基于活动重叠的研发项目调度模型与算法 [M]. 北京: 科学出版社, 2015.
Yu Jing , Xu Zhe. Research Project Scheduling Models and Algorithms based on Activities Overlapping [M]. Beijing: Science Press , 2015. (in Chinese)
- [22] Herroelen W , Leus R. Project scheduling under uncertainty: Survey and research potential [J]. *European Journal of Operational Research* , 2005 , 165(2) : 289 – 306.
- [23] 何正文, 宁敏静, 徐 渝. 前瞻性及反应性项目调度方法研究综述 [J]. *运筹与管理* , 2016 , 25(5) : 278 – 287.
He Zhengwen , Ning Minjing , Xu Yu. A survey of proactive and reactive project scheduling methods [J]. *Operations Research and Management Science* , 2016 , 25(5) : 278 – 287. (in Chinese)
- [24] 王艳婷, 何正文, 刘人境. 随机工期下反应性多模式项目调度优化 [J]. *系统管理学报* , 2017 , 26(1) : 85 – 93.
Wang Yanting , He Zhengwen , Liu Renjing. Reactive strategies in the multi-mode project scheduling with stochastic activity durations [J]. *Journal of Systems & Management* , 2017 , 26(1) : 85 – 93. (in Chinese)
- [25] Suwa H , Morita D. Reactive project scheduling method to enhance project progress under uncertainty [J]. *Journal of Advanced Mechanical Design , Systems , and Manufacturing* , 2016 , 10(3) : 1 – 8.
- [26] Zheng W B , He Z W , Wang N M , et al. Proactive and reactive resource-constrained max-NPV project scheduling with random activity duration [J]. *Journal of the Operational Research Society* , 2017 , online: 1 – 12.
- [27] Anagnostopoulos K P , Kotsikas L. Time-cost trade off in CPM networks by simulated annealing [J]. *Operational Research* , 2001 , 1(3) : 315 – 329.

Hidden consumptions of activity floats under generalized precedence relations

SU Zhi-xiong , WEI Han-ying , LI Lan

Business Administration College , Nanchang Institute of Technology , Nanchang 330099 , China

Abstract: This discovers new anomalies in activity network under generalized precedence relations (GPRs) . In conventional ideas , only two causations could result in consumptions of time floats of an activity: (1) the activity actively consumes time floats , and (2) it passively consumes time floats because of the impaction of consumptions of time floats of its preceding activities. However , we discover that even without the two causations , the time floats of an activity also can be consumed. These anomalies are hidden consumptions of activity floats , and may appear in projects with GPRs. We study the anomalies based on activity networks under GPRs , and design algorithms to quantize the hidden consumptions of classic total float and hidden float respectively. The hidden consumptions of activity floats may weaken current models and optimization approaches based on time floats so that has indispensable application to project scheduling with GPRs.

Key words: activity networks; generalized precedence relations; anomalies; time floats