

考虑认知动态变化的多技能项目调度模型

朱 翀, 张艳梅, 唐小毅

中央财经大学信息学院, 北京
Email: jcuchong@foxmail.com

收稿日期: 2021年8月29日; 录用日期: 2021年9月14日; 发布日期: 2021年9月29日

摘 要

许多互联网企业并未对人力资源实施精密细致的规划管理,在开发复杂软件项目时面临着人才调度困难,开发效率低下的困境,选择高效的多技能员工调度策略成为现代软件项目管理的新趋势。现有研究较少对员工的认知能力建模且未考虑企业人才培养战略目标的实现。因此,本文针对员工认知能力动态变化的多技能项目调度问题,以项目工期、项目成本以及技能增长效益为目标建立了考虑变动工作效率的优化模型,基于学习-遗忘曲线理论(LFCM)设计了工作效率随时间动态变化的计算方法,并提出一种考虑工作效率的新型工资分配模式。设计了改进的NSGA-III求解模型,算法引入了包含优先级和员工配置属性的多维染色体编码,使用一种新型进度生成策略(SSGS-SMTS)获得相应编码下调度方案的目标函数值,对不同维度的染色体分别采用特殊的交叉和变异操作以保证解的合理性和多样性。最后,结合工程实例进行验证,表明了所提模型及算法的有效性。

关键词

多技能, 人力资源, 项目调度, 变动工作效率, NSGA-III

Multi-Skill Project Scheduling Model Considering Cognitive Dynamic Change

Chong Zhu, Yanmei Zhang, Xiaoyi Tang

School of Information, Central University of Finance and Economics, Beijing
Email: jcuchong@foxmail.com

Received: Aug. 29th, 2021; accepted: Sep. 14th, 2021; published: Sep. 29th, 2021

Abstract

Most internet companies have not implemented precise and meticulous planning and scheduling

文章引用: 朱翀, 张艳梅, 唐小毅. 考虑认知动态变化的多技能项目调度模型[J]. 管理科学与工程, 2021, 10(3): 321-336. DOI: 10.12677/mse.2021.103040

of human resources, and are facing the dilemma of difficult talent scheduling and low development efficiency when implementing complex software projects. Choosing an efficient multi-skilled staff scheduling strategy has become a new trend in modern software project management. Existing research seldom models the cognitive ability of staff scientifically and often ignores the realization of corporate strategic objectives in talent cultivation. Hence, aiming at the multi-skill project scheduling problems in which the cognitive ability of staff changes dynamically, an optimization model considering variable perform efficiency with the objectives of project duration, project cost and skill growth benefit is established. Based on the LFCM (Learning and Forgetting Curve Model) theory, the calculation method of perform efficiency which changes with time dynamically is designed. The improved NSGA-III is adopted to solve the model. The algorithm introduces a multi-dimensional chromosome coding which includes the attributes of priority and staff allocation with the SSGS-SMTS (Serial Schedule Generation Scheme Combining with Semaphore Mechanism and Topological Sorting) to obtain the values of objective function under corresponding scheduling scheme. Special crossover and mutation are adopted for chromosomes of different dimensions in order to ensure the rationality and diversity of solutions. Finally, the excellence of the proposed model and algorithm is verified through a project example.

Keywords

Multi-Skill, Human Resources, Project Scheduling, Dynamic Perform Efficiency, NSGA-III

Copyright © 2021 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

随着人才的价值在现代社会发展和高新科技企业发展过程中日益凸显, 员工的高效调度也被赋予强烈的时代意义。Burleson [1]最先提出利用替代的劳动力资源可提升项目性能, 减少工艺短缺的发生, 提高现有劳动力的利用率。随之多技能项目调度问题(Multi-Skill Project Scheduling Problem, MSPSP)作为与人力资源调度问题内容深度交叉的问题之一, 近年来获得了国内外学者的跟踪与关注, 它是经典资源受限项目调度问题(RCPSP)的扩展, 概念由 Néron [2]首次提出, 这类问题中受限资源具有多技能属性, 其中人力资源是最常见的多技能资源, 除了需要确定项目中的资源分配, 还需要考虑员工的技能水平, 相较于 RCPSP, 该问题具有更强的针对性和现实意义。

随着用户需求的细化, 软件结构和功能已经变得越来越复杂, 软件开发项目的实现通常涉及具有多重优先级约束的多个任务, 即每个项目都包含多个任务, 并且每个任务必须由具有所需技能的员工合作执行。而技术人才往往是稀缺资源, 如何高效调度这些员工完成项目是互联网企业遇到的一大难题。此外, 技术人员通常具有多种技能, 例如, 一名员工掌握了几种不同的计算机语言, 例如 C++, Java 和 Python, 意味着他能够完成软件系统开发流程中的多项工作。同时, 一些研究强调[3], 认知因素会影响项目执行过程中员工的工作效率, 因此有必要考虑员工的学习效应和遗忘效应, 对其认知能力进行科学建模。软件项目开发计划问题属于多技能项目调度问题, 考虑项目开发期间员工技能水平的动态变化时, 该问题的求解将更加困难。因此, 本文建立了考虑认知动态变化的多技能项目调度模型, 该模型同时考虑了软件项目工期, 项目成本和技能增长效益的三个目标。设计项目工期目标是要考虑到软件系统上线的时间需求; 项目成本目标是为了制订开发预算; 技能增长效益目标是为了企业人才的培养, 需要关注员工技能水平的提升。该模型适用于流程复杂的软件开发项目的实施, 能够充分调用员工的多项技能, 并考虑

员工的认知能力的影响, 获得较为科学合理的员工调度结果, 为企业提供有关人才配置、工期估算以及成本预算的决策支持。

本文的主要贡献包括 3 个方面:

1) 基于学习-遗忘曲线理论(LFCM)对员工的认知能力建模, 提出一种结合信号量机制和拓扑排序算法的进度生成策略(SSGS-SMTS)获得多技能员工的调度方案, 员工能够在具有复杂流程的项目中充分运用所掌握的多项技能参与工作。

2) 提出一种新型工资分配模式, 员工的绩效工资由完成工作的执行效率决定, 工作效率受到员工认知能力的影响, 更加现实合理。

3) 运用改进的 NSGA-III 算法求解模型, 能够同时优化多重目标, 融合了企业的人才培养需求, 可为多技能员工调度方案的制订提供有效参考。

2. 相关工作

随着知识获取途径的多元化, 员工学习技能的方法也更加丰富, 在互联网企业中员工掌握多种研发技能十分常见, 其主要特征是, 一个员工掌握多种技能且技能水平高低不一, 有关多技能员工的调度问题被归纳为 MSPSP, 是目前国内外研究的热点问题。

Heimerl 等[4]研究 IT 领域的多技能人员调度问题, 建立了以成本最小化为目标的非线性混合整数规划模型, 通过 CPLEX 求解器求解模型。Correia 等[5]提出了一种以最小化项目完工时间为目标的混合整数线性规划模型, 通过设置一些二元变量的最优值以限制搜索空间, 并利用多组额外不等式如时间窗约束等改进该模型。Liu 等[6]提出一种时间持续优化模型, 将多技能的人员进行组合优化项目工期以提高工程建设工作的连续性和效率, 采用约束规划方法求解。这些研究仅考虑单个目标的优化, 使用精确解法求解, 但对于复杂度较大的调度问题容易陷入组合爆炸而无法在有限时间内解决, 智能优化算法属于启发式算法, 更适合求解这类问题, 本文选择的 NSGA-III 作为最新出现能够有效求解高维多目标模型的智能优化算法之一, 在 MSPSP 以往研究中尚未应用。

Nie 等[7]设计了一种结构化方法为具有多种不同技能熟练度的员工分配工作, 运用一种结合遗传算法的模因算法进行求解得到最短工期。Kazemipoor 等[8]考虑多项目中任务的多技能员工安排调度, 以与预期完工时间的最小偏差为优化目标, 通过差分进化算法求解确定资源配置情况。Yannibelli 等[9]以员工的总技能熟练度最大和项目工期最短为目标同时优化, 设计了一种新型进化算法生成 Pareto 近似解集作为问题的解。这些研究选择忽略项目执行过程中员工技能水平的变化, 假设员工的技能熟练度在项目期间恒定不变, 而实际上人的认知能力决定了其固有的学习能力和遗忘能力, 应将该现象科学建模从而帮助企业决策者制订更符合人认知变化过程的项目调度与计划方案。

Gutjahr 等[10]提出了一种竞争力驱动的项目组合选择模型, 以项目中获得的经济收益和从期望能力的增长中获得的战略收益的加权平均值最大化为目标, 确定人员的调度计划。Attia 等[11]考虑了员工在项目中的学习和遗忘效应, 员工的技能水平随着项目的进行不断变化, 目标函数由六个子目标复合而成, 包括三种类型的成本、两种类型的惩罚数和一种虚拟利润。陈蓉等[12]注意到产品研发项目执行过程中人员随机离职的现象, 采用离散马尔可夫链描述人员离职过程, 考虑人才培养战略收益、研发周期以及研发成本三个目标建立了新产品研发项目组合调度随机多目标约束优化模型, 采用自适应 Pareto 抽样算法求解模型。这些研究考虑了人的认知能力, 但对员工学习效应和遗忘效应的建模尚不完善, 且工资依据工作时长计算, 强调员工的时间价值却忽视了更加重要的效率价值, 这不仅对技能水平较高的员工短时间完成指派工作不公平, 也不利于企业长远目标的实现。本文认为应对高效率完成工作的员工给予更多奖励, 对工资分配模式进行改进。

本文提出的考虑认知动态变化的多技能项目调度模型能够有效填补现有工作的不足, 该模型考虑了员工在软件开发期间的学习效应和遗忘效应, 并提出一种考虑工作效率的新型工资分配模式, 采用改进的 NSGA-III 求解, 以项目工期、项目成本以及技能增长效益为优化目标, 融合了企业的人才培养需求。最后结合工程实例进行验证, 表明了本文所提方法的有效性以及应用价值。

3. 问题建模

3.1. 问题描述

在一个已确定开发流程和依赖关系的软件项目中, 存在 n 个任务需要完成, 规定任务有偏序关系, 任务之间存在串行或并行结构。每个任务包含一个或多个对应某种技能的工作, 每个工作需要一定数量的员工合作完成, 项目所需技能总数为 r 种, 现共有 m 个员工可供调用, 员工可以一专多能, 即精通本职专业技术, 同时掌握与本职工作密切相关的多种技术, 各技能的应用水平不同, 定义技能熟练度 $\beta(0 \leq \beta \leq 1)$ 描述技能的应用水平, 技能熟练度越高, 工作效率越高。项目期间同时考虑员工学习效应和遗忘效应对工作效率的影响, 技能熟练度随着工作时间增加而增加并随着闲置时间增加而降低。

在满足逻辑约束、员工人数、技能需求等约束条件下, 以项目成本最低、项目工期最短以及技能增长效益最大为优化目标, 需要确定员工合理配置的调度计划方案。问题假设如下。

- 1) 项目期间员工数恒定, 不考虑员工的离职和增派情况, 员工一旦被分配到某工作, 则相应任务开始执行, 不可中断;
- 2) 考虑员工认知能力对工作效率的影响, 员工的学习效应和遗忘效应遵循 3.2 节提出的 LFCM 模型;
- 3) 员工的技能熟练度的计算以工作为界限, 不考虑员工工作时间以外各个时点的技能水平变化;
- 4) 项目开发以人为关键资源, 项目成本仅考虑员工工资, 不考虑物料成本或设备成本等;
- 5) 每个任务中可能包含多个工作, 每个工作对应一种技能的工作, 每个工作需要不同人数的员工合作完成, 被指派到同一工作的多个员工必须同时开工并同时完工;
- 6) 每个员工不能同时执行两个及以上技能的工作, 即每个单位时间内只能执行一个任务中的一种工作。

3.2. 员工认知能力建模

LFCM (Learning and Forgetting Curve Model) 模型[13]是基于经典的 WLC 学习曲线模型提出的较为完善的学习 - 遗忘曲线模型, 该模型将遗忘效应与学习效应进行类比, 揭示了学习和遗忘程度与中断时间的关系。本文基于 LFCM 理论建立考虑学习 - 遗忘效应的技能熟练度的计算方法:

$$\beta_{pki}^f = \beta_{pki}^s + L \cdot T^{a_k} - F \cdot T^{b_k}$$

其中 β_{pki}^s 表示员工 p 使用技能 k 开始执行任务 i 时刻的熟练度; T 表示员工 p 使用技能 k 执行任务 i 的持续时间, T' 表示员工 p 使用技能 k 执行任务 i 的初始时刻与之前最后一次使用技能 k 执行任务的结束时刻之差, 即员工闲置时间; L 表示学习曲线系数, F 表示遗忘曲线系数, 皆与初始技能熟练度有关, 通常情况下, 当技能水平较低时, 由于基础知识的学习难度较低, 水平提升速度较快, 并且知识不容易遗忘; 而技能水平较高时, 深层次知识的学习难度相对较高, 水平提升速度会放缓, 并且知识容易遗忘。通常情况下人的遗忘速度小于学习速度, 本文规定 $L = -\ln(\beta_{pki}^s/2)/10$, $F = \ln(3\beta_{pki}^s)/15$, 初始技能熟练度越大, 学习曲线系数越小且遗忘曲线系数越大; a_{pk} 表示员工 p 对于技能 k 的学习因子, $a_{pk} = -\ln(\lambda_{pk})/\ln(2)$, λ 为学习率, λ 一般在 75%~95%, 学习因子越大则学习效应越强; b_{pk} 表示员工 p 对于技能 k 的遗忘因子, $b_{pk} = -\ln(1-\mu_{pk})/\ln(2)$, μ 为遗忘率, μ 一般在 3%~15%, 遗忘因子越

大则遗忘效应越弱，学习率和遗忘率均与员工个人因素密切相关。员工的学习 - 遗忘特征如图 1 所示，考虑到人的生理条件限制，规定技能熟练度有学习上限 $\hat{\beta}$ 和遗忘下限 $\underline{\beta}$ 。 $\Delta\beta(T) > \Delta\beta(T')$ ， $T = T'$ 表示相同时间内员工的学习效应强于遗忘效应。此外，每次中断后员工的学习效应和遗忘效应均发生改变，随着技能熟练度的提升，学习速度放缓，遗忘速度加快。

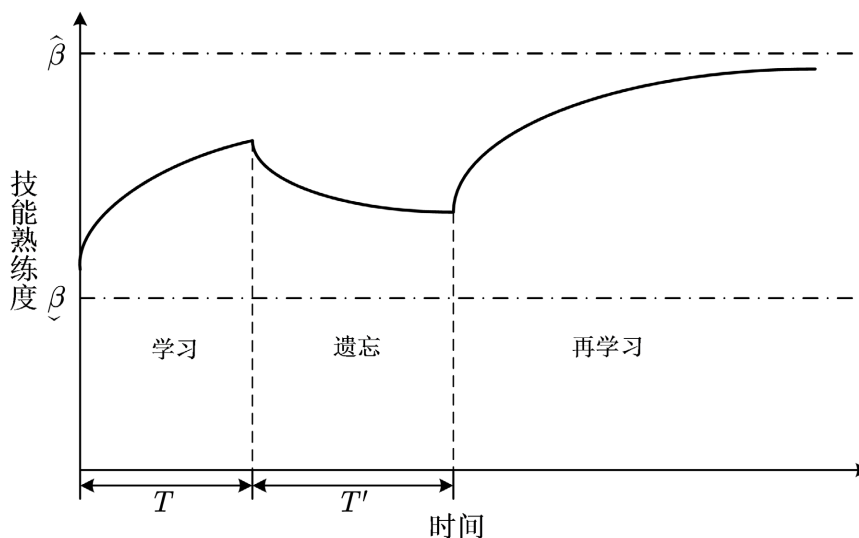


Figure 1. Staff learning-forgetting characteristics

图 1. 员工学习 - 遗忘特征

3.3. 符号定义

1) 参数

符号	描述
I	任务 i 集合, $I = \{1, 2, \dots, n\}$
K	技能 k 集合, $K = \{1, 2, \dots, r\}$
P	员工 p 集合, $P = \{1, 2, \dots, m\}$
α_{ki}	若任务 i 包含需要技能 k 执行的工作, 则值为 1 否则值为 0
J_{ki}	任务 i 中使用技能 k 执行的工作, 工作集合 $P = \{1, 2, \dots, m\}$, $j = \sum_{i=1}^n \sum_{k=1}^r \alpha_{ki}$
A	关键路径上的任务集合, $A = \{a_1, a_2, \dots, a_x\}$, $a_x \in [1, n]$
d_{ki}	工作 J_{ki} 所需要的员员工数
T_{ki}^{\min}	工作 J_{ki} 的最短持续时间, 指技能熟练度为 1 的员工完成该工作所需时间
β_{pk}^s	项目开始时员工 p 使用技能 k 的熟练度, 代表员工的工作效率, $\beta_{pk} \in [0, 1]$, 若员工 p 不具备技能 k 则值为 0
$\bar{\beta}_{ki}$	工作 J_{ki} 的平均熟练度
B_i	任务 i 的开工时间
PF_i	任务 i 的紧前任务集合
T_i	任务 i 的持续时间
B_{ki}	工作 J_{ki} 的开工时间

Continued

φ_{ki}	工作 J_{ki} 的优先级, $\varphi_{ki} \in [1, j]$
T_{ki}	工作 J_{ki} 的持续时间
S_{ki}	分配给工作 J_{ki} 的员工集合
c_{ki}^{\max}	工作 J_{ki} 的最高提成, 指技能熟练度为 1 的员工完成该工作所获报酬
w	员工的基本工资
ε_k	技能 k 的发展权重, 指技能对企业的重要程度
a_{pk}	员工 p 对技能 k 的学习因子
L	学习曲线系数
b_{pk}	员工 p 对技能 k 的学习因子
F	遗忘曲线系数

2) 变量

符号	描述
x_{pki}	若员工 p 参与工作 J_{ki} , 则值为 1, 若员工 p 具备技能 k 但不参与工作 J_{ki} 则值为 0, 若员工 p 不具备技能 k 则值为 -1
y_{pkit}	若员工 p 在 t 时刻参与工作 J_{ki} , 则值为 1, 否则值为 0
X_{pki}	若员工 p 参与工作 J_{ki} 则值为 1, 否则值为 0
β_{pki}^s	员工 p 开始工作 J_{ki} 时刻的技能熟练度
β_{pki}^f	员工 p 结束工作 J_{ki} 时刻的技能熟练度

3.4. 模型建立

$$\min T = \sum_{i \in A} T_i \tag{1}$$

$$\min C = m \cdot w + \sum_{i=1}^n \sum_{k=1}^r (\bar{\beta}_{ki} \cdot c_{ki}^{\max}) \tag{2}$$

$$\max R = \sum_{i=1}^n \sum_{k=1}^r \sum_{p=1}^m \varepsilon_k \cdot (\beta_{pki}^f - \beta_{pki}^s) \tag{3}$$

$$\bar{\beta}_{ki} = \frac{\sum_{p=1}^m \beta_{pki}^s \cdot X_{pki}}{d_{ki}} \tag{4}$$

$$T_i = \max_{k \in K} \{T_{ki}\}, T_{ki} = \frac{T_{ki}^{\min}}{\bar{\beta}_{ki}} \tag{5}$$

$$B_i - B_j \geq T_j, \forall i \in I, \forall j \in PF_i \tag{6}$$

$$\beta_{pki}^s = \begin{cases} \beta_{pk}^s & \forall x_{pki'} \neq 1, i' \in PF_{i'} \\ \beta_{pki'}^f, \arg \min_{i' \in PF_i} \{B_{ki'} - (B_{ki'} + T_{ki'})\} & \exists x_{pki'} = 1, i' \in PF_{i'} \end{cases} \tag{7}$$

$$\beta_{pki}^f = \begin{cases} \beta_{pki}^s + L \cdot T_{ki}^{a_{pk}} & \forall x_{pki'} \neq 1, i' \in PF_i \\ \beta_{pki}^s + L \cdot T_{ki}^{a_{pk}} - F \cdot \left(\min_{i' \in PF_i} \{B_{ki} - (B_{ki} + T_{ki'})\} \right)^{b_{pk}} & \exists x_{pki'} \neq 1, i' \in PF_i \end{cases} \quad (8)$$

$$L = -\ln(\beta_{pki}^s/2)/10, F = \ln(3\beta_{pki}^s)/15$$

$$B_{ai} - B_{bj} \geq T_{bj}, \varphi_{aj} \geq \varphi_{bi}, PF_i = PF_j, S_{ai} \cap S_{bj} \neq \emptyset \quad (9)$$

$$B_{ai} - B_{bi} \geq T_{bi}, \varphi_{bi} \geq \varphi_{ai}, S_{ai} \cap S_{bi} \neq \emptyset \quad (10)$$

$$\sum_{p=1}^m X_{pki} = d_{ki} \quad (11)$$

$$\sum_{i=1}^n \sum_{k=1}^r y_{pkit} \leq 1 \quad (12)$$

$$x_{pki} \in \{0, 1, -1\}, y_{pkit} \in \{0, 1\}, X_{pki} \in \{0, 1\}; \forall p, k, i, t \quad (13)$$

$$x_{pki} = \begin{cases} 1 & \forall t, y_{pkit}=1 \\ 0 & \forall t, y_{pkit}=0; \forall p, k, i \\ -1 & \alpha_{ki} = 0 \end{cases} \quad (14)$$

$$X_{pki} = \begin{cases} 1 & x_{pki} = 1 \\ 0 & x_{pki} \neq 1 \end{cases}; \forall p, k, i \quad (15)$$

其中式(1)表示使项目工期最小化的目标函数，总工期为关键路径上任务的持续时间的加和；式(2)表示使项目成本最小化的目标函数，项目成本为发放给所有员工工资的总和，包含基本工资和提成两部分，基本工资所有员工固定一致，提成与完成工作的平均效率正相关；式(3)表示使技能增长效益最大化的目标函数，总效益等于所有员工参与各个工作后技能熟练度增值之和；式(4)表示工作平均熟练度的计算方法；式(5)表示任务持续时间的计算方法；式(6)表示任务的逻辑约束条件，即任务必须在其紧前任务全部结束后方可开工；式(7)表示员工开始工作时刻技能熟练度的计算方法；式(8)表示员工结束工作时刻技能熟练度的计算方法，规定技能熟练度的学习上限为1，遗忘下限为0.3，若计算得到的熟练度值大于1或小于0.3，则将该值重置为上限或下限值；式(9)表示当并行任务中存在分配了同一员工完成多个工作的情况，工作指派发生人力资源冲突时，员工先被分配给优先级较高的工作使用技能，即优先级较高的工作先开工；式(10)表示当同一任务中存在一名员工被分配到多个工作的情况时，优先级较高的工作先开工；式(11)表示安排在任务*i*使用技能*k*的员工数应等于完成任务*i*所需掌握技能*k*的员员工数；式(12)表示每名员工在任意时刻只能参与一种工作；式(13)、式(14)和式(15)表示三个决策变量的取值范围和相互关系。

4. 算法设计

4.1. 编码

针对多技能调度问题的特点，本文设计了一种新型多维染色体编码方案，相较于现有文献[14][15]中的实数单维编码模式更加简洁直观：染色体的每列基因值代表项目中任务分解后对应某种技能的工作的相关属性：1) 优先级染色体。表示工作的优先级：编码为随机不重复整数 φ_{ki} ，当并行工作发生人力资源冲突时，具有较高优先级编码的工作先开工；2) 员工配置属性染色体。表示工作所分配员工的状态：每一员工对应染色体的某一横行，编码为决策变量 x_{pki} ，若员工*p*使用技能*k*参与任务*i*，则值为1，否则值为0，若员工无技能*k*则值为-1。

		任务 I		I_1		I_2		I_3		I_i		I_n		
多维染色体编码方案	优先级 φ	3	5	4	1	9	7	...		φ_{ki}	...	12	20	
	员工 P	P_1	1	0	-1	1	0	1	...		x_{1ki}	...	1	0
		P_2	1	-1	1	0	-1	1	...		x_{2ki}	...	0	-1
		\vdots	\vdots				\vdots	\vdots			\vdots	\vdots		\vdots
		P_m	-1	1	0	-1	-1	1	...		P_{mki}	...	0	-1

Figure 2. Multi-dimensional chromosome coding scheme
图 2. 多维染色体编码方案

本文设计的多维染色体编码方案如图 2 所示：任务 I_1 需要掌握两种技能的员工合作完成，两种工作的优先级分别为 3 和 5，在任务 I_1 的工作 1 中，员工 P_1 和员工 P_2 的染色体编码均为 1，表示员工 P_1 和员工 P_2 都被安排完成工作 1，在任务 I_1 的工作 2 中员工 P_1 的染色体编码为 0，表示不安排员工 P_1 完成工作 2，在任务 I_1 的工作 2 中员工 P_2 的染色体编码为 -1，表示员工 P_2 不具备完成工作 2 的技能，在初始化时置为 -1。如果任务 I_1 中的两个工作发生人力资源冲突，由于工作 2 的优先级较高，首先为工作 2 分配员工，工作 2 完成后工作 1 方可开工，以此类推其他并行工作发生冲突的解决方法。

4.2. SSGS-SMTS 进度生成策略

本文提出一种结合信号量机制和拓扑排序算法的进度生成策略(Serial Schedule Generation Scheme combining with Semaphore Mechanism and Topological Sorting, SSGS-SMTS)获得调度方案并获取项目工期目标函数值。SSGS 由 Kelley 等[16]首次提出，强调以任务为阶段变量，每个阶段依据优先规则在可行任务集中选择合适的任务完成调度，本文以 SSGS 为基础加以改进，将每个工作抽象成项目网络拓扑图中的一个结点，将员工的分配抽象为操作系统进程的一次调度，将员工抽象为进程申请的资源，算法具体步骤如下。

Step1: 根据多维染色体编码方案，可获得为每个工作 J^* 分配的员工作集 S^* 以及相应的优先级 φ^* ，首先根据项目的网络拓扑图计算出所有工作的入度 $indegree(J^*)$ ，将所有员工置为空闲状态 $condition(P^*)=0$ ，令当前时刻 $T=0$ ，所有工作的访问标志初始化为 $visit(J^*)=0$ ；

Step2: 选取 $indegree(J^*)=0$ 且 $visit(J^*)=0$ 的工作加入起始队列 R ，将 R 中工作按照优先级大小降序排列；

Step3: 取出 R 的队首工作 J^r ，判断为其分配的员工作集 S^r 中是否存在处于忙碌状态 $condition(P^*)=1$ 的员工，如果不存在则将 J^r 加入执行队列 E ，并将 S^r 中所有员工的状态置为忙碌，否则将 J^r 加入阻塞队列 W ；

Step4: 如果 $R \neq \emptyset$ ，继续执行 Step3，否则从执行队列 E 中取出具有最短持续时间 t^{min} 的工作 J^e ，将当前时刻改为 $T = T + t^{min}$ ，将 E 中的所有工作的持续时间改为 $duration(J^*) = duration(J^*) - t^{min}$ ，将 S^e 中所有员工的状态置为空闲，并把从工作 J^e 出发到达的所有工作 J^* 的入度改为 $indegree(J^*) = indegree(J^*) - 1$ ，令 $visit(J^e) = 1$ ；

Step5: 如果 $W = \emptyset$ ，转 Step6，否则对 W 中所有工作 J^w 进行判断，若为其分配的员工作集 S^w 中

不存在处于忙碌状态 $condition(P^*)=1$ 的员工，则将 J^w 加入起始队列 R ，并将其从 W 中释放；

Step 6: 如果项目中仍然存在访问标志 $visit(J^*)=0$ 的工作，转步骤 2 继续循环，否则记录下当前时刻 T ，即为项目工期，所有工作被执行完毕，算法终止。

4.3. 交叉操作

本文分别对优先级染色体和员工配置属性染色体执行以下交叉操作。

1) 优先级染色体：优先级染色体编码为随机整数，本文采用 SEC 交叉策略[17]：在一条父代染色体上选择一组基因，在另一条父代染色体上找到这些基因的位置，保持未选中基因不变，按选中基因的出现顺序，交换两父代染色体中基因的位置。

2) 员工配置属性染色体：员工配置属性染色体的编码由 m 行 j 列构成，每一行代表一个员工的工作分配属性，每一列代表一个工作的员工配置属性。每列基因值为 1 的个数应为该工作所需人数，若在单行进行行交叉，极大概率会引起染色体失效，因此设计了多列合并交叉的交叉策略，即在两条父代染色体的相同位置随机选取两列基因分别作为交叉始位和交叉终位，将交叉位之间的多列基因合并交换即生成新的可行子代染色体，如图 3 所示。

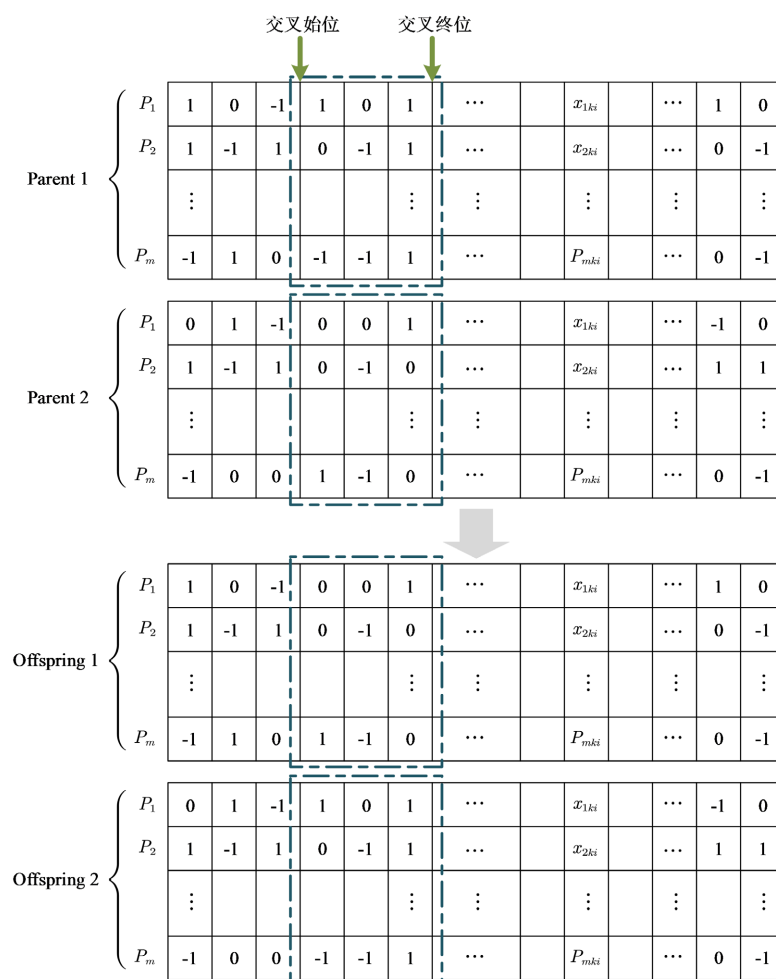


Figure 3. Staff allocation chromosome crossover operation

图 3. 员工配置属性染色体交叉操作

4.4. 变异操作

本文分别对优先级染色体和员工配置属性染色体执行以下变异操作。

1) 优先级染色体：为避免染色体变异后工作优先级重复从而导致资源冲突时难以判断工作执行顺序，因此采取一种随机变异策略：在父代染色体上随机选择两个基因作为变异位，交换两变异位上对应基因。

2) 员工配置属性染色体：为确保变异后的员工配置属性染色体仍为可行染色体，即当增派员工做某项工作的同时在该项工作中调离相应人数的员工，本文对员工配置属性染色体采取一种反转变异策略：随机选中染色体某列上基因值分别为 1 和 0 的两个基因作为变异位，将变异位上对应基因交换，如图 4 所示。

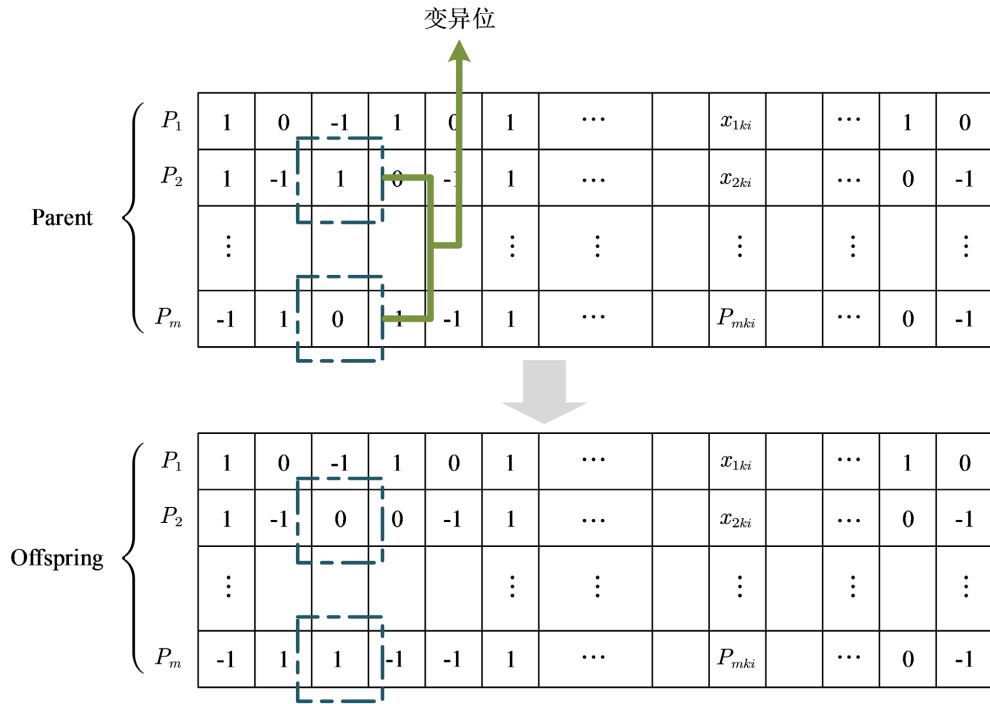


Figure 4. Staff allocation chromosome mutation operation
图 4. 员工配置属性染色体变异操作

4.5. 算法流程

综上所述，本文设计了改进的 NSGA-III [18]算法求解模型，算法流程如图 5 所示。

5. 工程实例验证

5.1. 实例描述

现有位于北京市的一家主营业务为软件研发的科技有限公司 Y，以该公司承包位于北京市朝阳区的某商业银行系统的开发项目 Px 为例，对本文的模型及算法进行验证：项目 Px 可被分解成 20 个具有串行或并行关系的子任务，每个任务包含至少需要一种技能的工作，将任务分解后共形成 40 个对应单种技能的不同工作，每个工作需要若干员工配合完成。公司 Y 共有可调度员工 15 人，每个员工掌握 1~3 种技能不等，每种技能的掌握水平不一，员工的技能熟练度由公司主管、项目经理以及技术负责人等成立专

家小组评价确定，规定所有员工的基本工资为 2000 元。项目的逻辑关系约束、各工作所需人数、各工作的最短持续时间和最高提成以及员工的技能分布描述和对应技能水平数据可从 <https://github.com/JcuChong/MSPSP-LFCM> 获取。

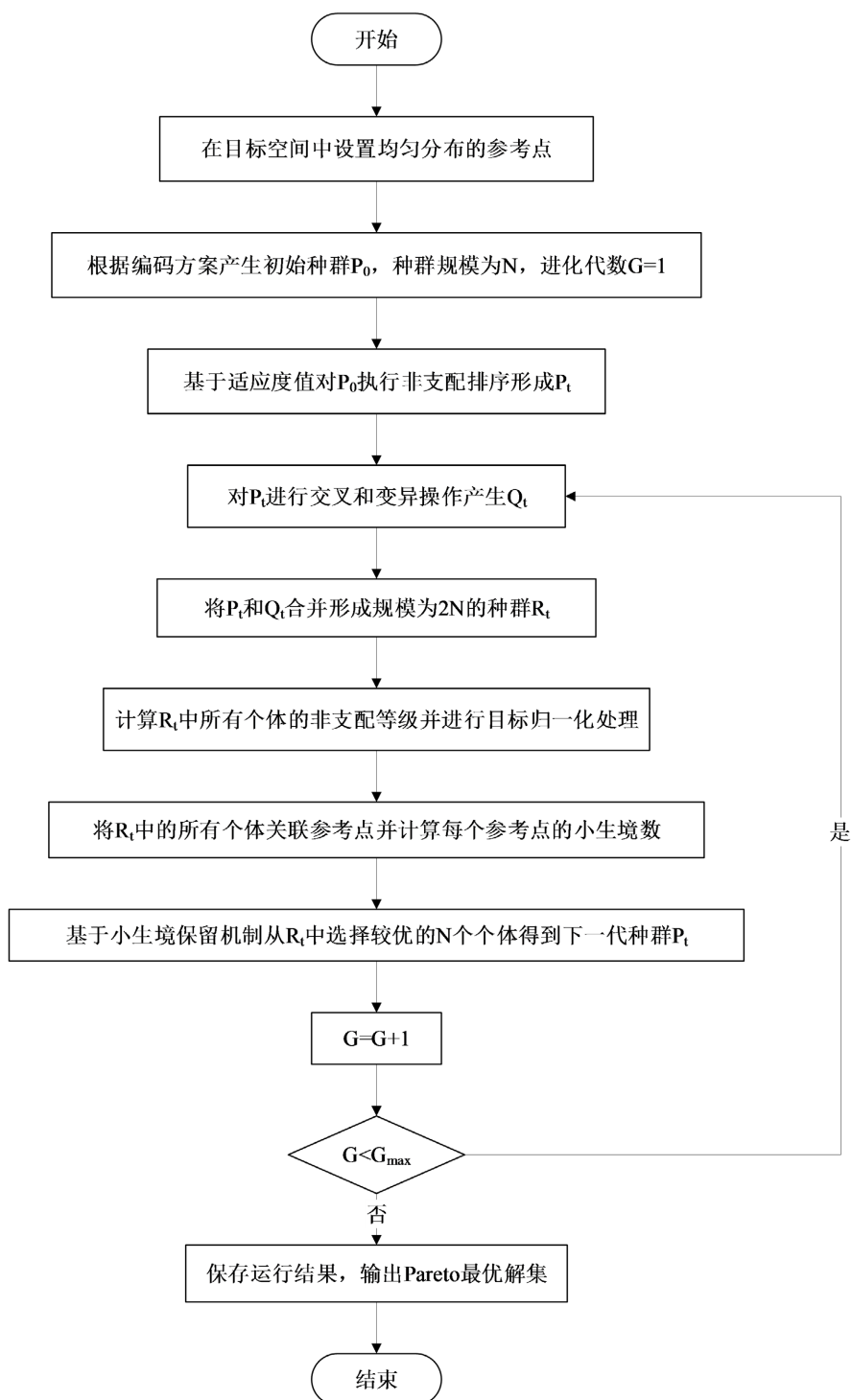


Figure 5. Algorithm flow chart

图 5. 算法流程图

5.2. 结果分析

根据上文提出的考虑认知动态变化的多技能项目调度模型，采用 NSGA-III 算法求解，以 MATLAB R2019a 实现，程序运行环境为 i5-10210U CPU，主频 1.60 GHz。算法参数设置如下：种群规模 $N_p = 500$ ，交叉概率 $P_c = 0.5$ ，变异概率 $P_m = 0.5$ ，最大进化代数 $G_{\max} = 3000$ ，员工学习率 λ 均取 0.92，员工遗忘率 μ 均取 0.04。

本文选取 30 次实验中获得的最优解，抽取了部分具有代表性的可行员工调度方案进行分析。表 1 计算了四种方案下的工资分配以及项目成本和项目工期，方案一下项目工期最短，方案二下项目成本最低，方案三下技能增长效益最大，方案四下三个目标值处于适中水平，受到学习 - 遗忘效应影响，员工工资由参与工作时的实际执行效率决定，计算得到最低项目成本为 137,009.29 元，最短项目工期为 86.98 天，最大技能增长效益为 7.5835，项目经理可结合实际需求确定其中一种调度方案执行。图 6 为项目工期最短(方案一)下工作调度与计划甘特图，每个工作内标识了其归属的任务，不同颜色的矩形对应需要不同技能的工作，清晰展示了所有任务的开工时间与完工时间。图 7 为方案一下的员工调度图，矩形内的数字表示工作编号，该图动态描述了可调度员工的工作分配情况，员工利用率达到 78.63%，易知调度安排紧凑，该方案可推荐给有赶工需求的项目经理。

图 8 绘制了方案一下员工 P3 和 P14 的技能熟练度变化趋势，这两位员工同时掌握三种技能，且技能水平不一，具有一定代表性，其初始熟练度分别为 1、0.8、0.6 和 0.6、0.8、1，图中的标记点表示项目工作的开始和结束，从图中可以直观发现当技能熟练度在工作期间上升且在闲置期间下降，员工的学习能力和遗忘能力在多次中断后都会发生改变，工作效率随时间动态变化，符合 LFCM 模型理论，验证了本文提出的考虑学习 - 遗忘效应的技能熟练度计算方法的可行性。图 9 为利用本章提出的 NSGA-III 求解模型获得的 Pareto 最优解以及拟合得到的 Pareto 最优前沿曲面，解集在曲面周围分布较为均匀，说明解的多样性较好。

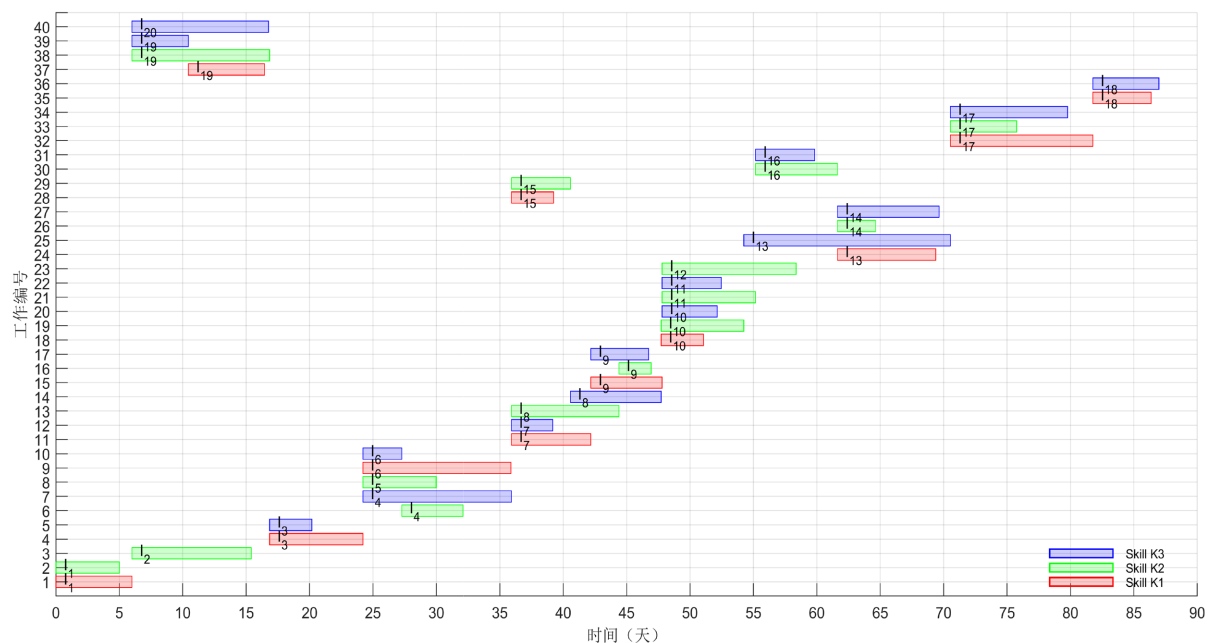


Figure 6. Work scheduling Gantt chart under the Scheme No.1

图 6. 方案一下工作调度甘特图

Table 1. Wage distribution and the optimal values of each objective under typical staff scheduling plans
表 1. 典型工人调度计划方案下的工资分配及最优项目成本和项目工期

员工 P 编号	工资总额(元)				员工 P 编号	工资总额(元)			
	方案一	方案二	方案三	方案四		方案一	方案二	方案三	方案四
1	8653.75	6629.17	8257.95	6837.96	9	9382.25	7211.06	8242.90	8227.73
2	8102.29	7144.56	7246.31	7699.41	10	11365.37	12936.2	11137.08	11358.74
3	10901.65	14932.32	11974.67	14111.38	11	10936.31	12924.32	11738.45	13665.61
4	11379.85	9231.01	12862.33	11984.47	12	8587.04	4831.56	5583.07	4626.99
5	8001.83	4352.16	5185.33	5155.44	13	8633.08	10126.75	8973.34	8531.48
6	7984.31	6832.38	9741.69	9262.79	14	11309.48	11407.92	11291.56	11386.45
7	9434.55	11046.63	11986.92	11288.09	15	8227.01	7623.77	8313.39	7766.79
8	9888.54	9779.49	8183.05	7325.07	项目 工期(天)	86.98	135.15	104.67	132.83
项目 成本(元)	142787.31	137009.29	140718.05	139228.38	技能增长 效益	6.8886	6.4871	7.5835	7.3810

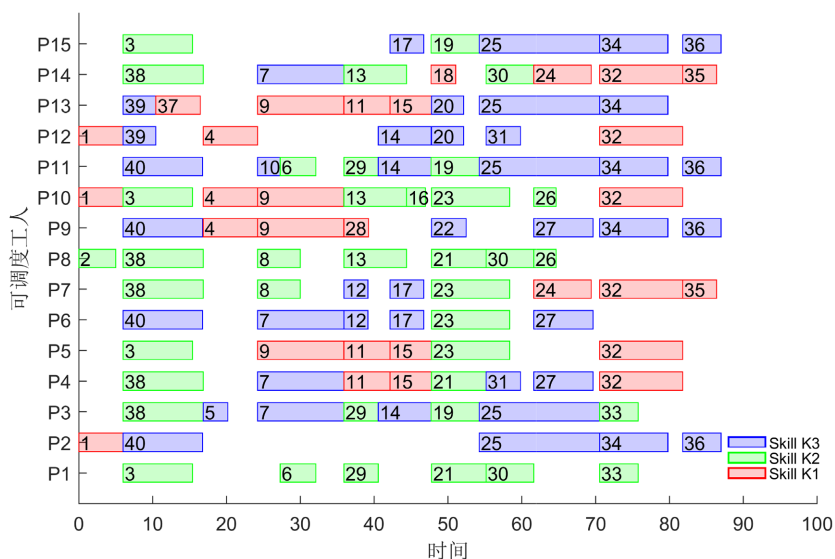


Figure 7. The staff scheduling plan under Scheme No.1 which holds the lowest project duration

图 7. 项目工期最短方案一下工人调度图

Table 2. NSGA-III vs. NSGA-II the results of 30 experiments under the optimal values of each objective
表 2. NSGA-III vs. NSGA-II 30 次实验下各目标最优时实验结果对比

描述指标	项目工期		项目成本		技能增长效益	
	NSGA-III	NSGA-II	NSGA-III	NSGA-II	NSGA-III	NSGA-II
最优解	86.98	91.64	137009.29	137103.85	7.5835	7.3113
均值	87.82	94.72	137356.10	137595.80	7.5762	7.2883
均值的标准误差	0.50	0.89	656.75	832.57	0.05	0.08
标准差	0.86	1.81	1857.56	2202.78	0.13	0.19
变异系数	0.009793	0.019109	0.013524	0.016009	0.017159	0.026069

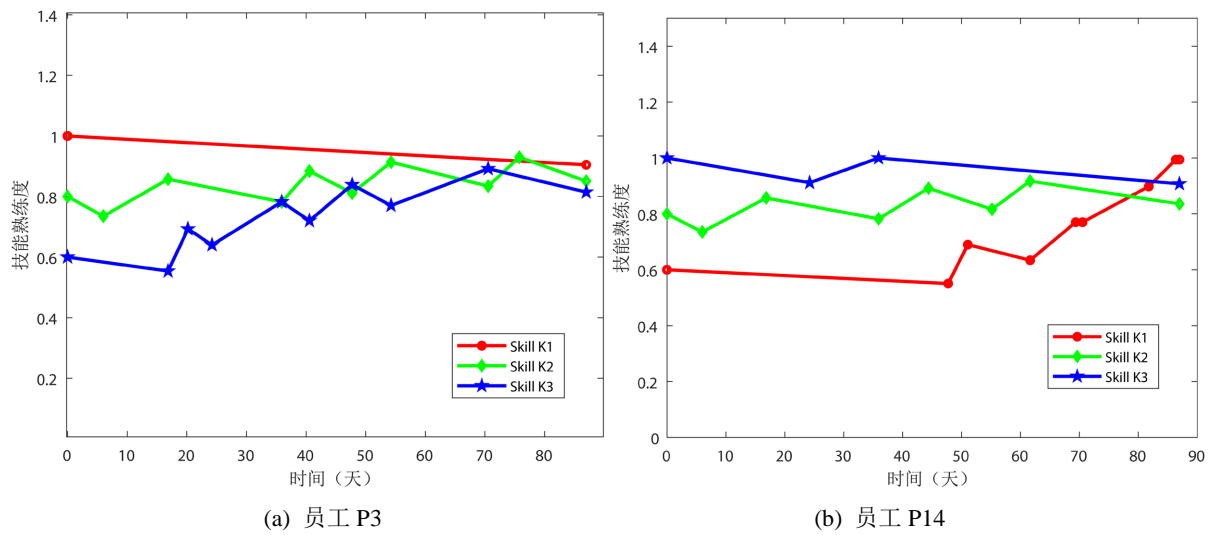


Figure 8. Changes in the proficiency of workers under Scheme No.1
图 8. 方案一下工人熟练度变化趋势

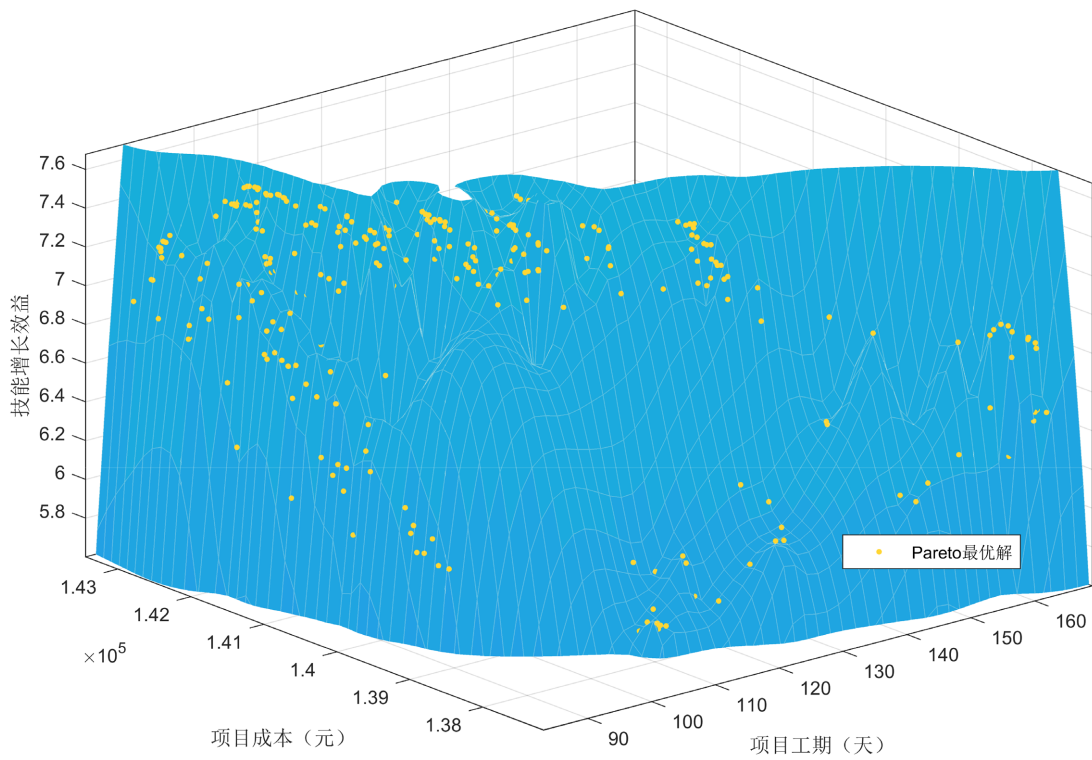


Figure 9. The Pareto optimal solution and the Pareto fitting front surface
图 9. Pareto 最优解以及拟合得到的 Pareto 最优前沿曲面

5.3. 算法对比

为了测试算法效率，在相同参数下分别使用本文提出的 NSGA-III 和 Chen [15] 采用的 NSGA-II 求解模型，生成的三个目标值的迭代过程如图 10 所示，绿色代表 NSGA-III，红色代表 NSGA-II，通过对比可知，NSGA-II 在进化后期容易陷入局部最优，在高维解空间中搜索效率较低，而 NSGA-III 的种群多样

性较好, 算法收敛速度较快, 进化到 1500 代左右已经接近完全收敛。表 2 为算法 30 次实验下各目标最优时不同指标对比, 结果显示所用算法的标准差以及变异系数更小, 进一步反映了算法收敛性和稳定性较好, 结果也显示模型目标的优化效果更佳, 表明本文所提出算法的有效性。

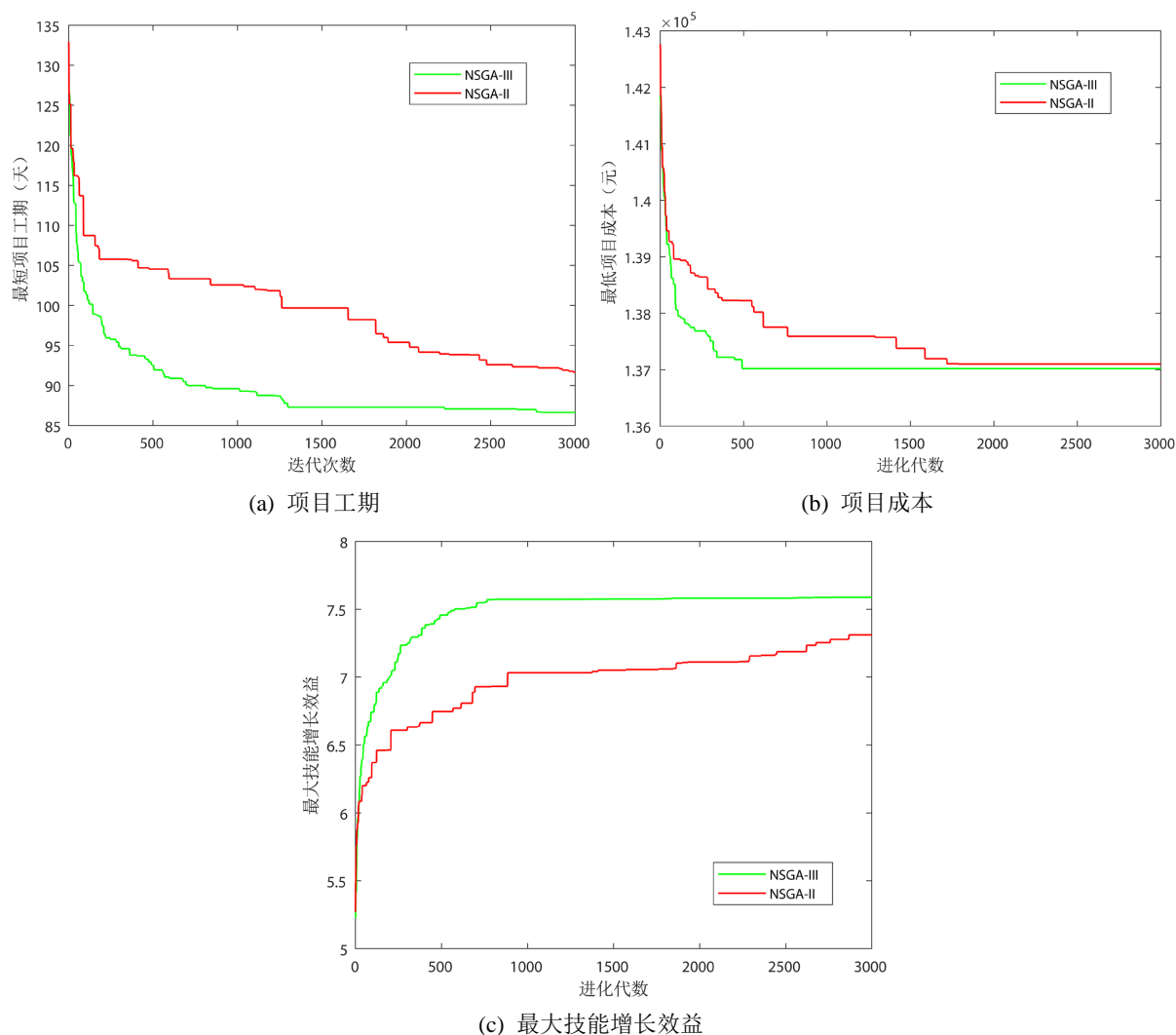


Figure 10. The iteration process of each objective

图 10. 各目标值迭代过程

6. 总结与展望

本文研究了软件开发项目中员工认知动态变化的多技能调度问题, 建立了多目标优化模型, 基于 LFCM 理论设计了工作效率随时间动态变化的计算方法, 并提出一种考虑变动工作效率的新型工资分配模式, 设计了改进的 NSGA-III 求解模型, 可获得贴近人的认知能力的多技能员工调度方案。结合项目实例验证了本文研究方法的实用性和有效性, 能够为企业承包项目时有关人才配置、工期估算以及成本预算的决策提供合理依据, 丰富了项目计划与调度领域的理论, 具有较强的现实意义和应用价值。

在现实中, 项目实施还受到加班赶工、劳动法保护等其他因素的影响, 下一步将考虑项目的时间窗约束, 任务分解等更多的情景。此外, 员工认知能力受许多因素的影响, 未来将考虑员工心情的起伏对

工作效率的影响, 同事之间的竞争效应对技能水平的影响, 以及多种技能学习和遗忘过程中的复合作用等。

基金项目

国家自然科学基金青年项目(No.61602536); 北京市社会科学基金重点项目(No.16YJA001)。

参考文献

- [1] Burleson, R.C., Haas, C.T., Tucker, R.L., *et al.* (1998) Multiskilled Labor Utilization Strategies in Construction. *Journal of Construction Engineering and Management*, **124**, 480-489. [https://doi.org/10.1061/\(ASCE\)0733-9364\(1998\)124:6\(480\)](https://doi.org/10.1061/(ASCE)0733-9364(1998)124:6(480))
- [2] Néron, E. (2002) Lower Bounds for the Multi-Skill Project Scheduling Problem. *Proceeding of the Eighth International Workshop on Project Management and Scheduling*, 2002, 274-277.
- [3] Baloff, N. (1966) Startups in Machine-Intensive Production Systems. *Journal of Industrial Engineering*, **17**, 25-32.
- [4] Heimerl, C. and Kolisch, R. (2010) Scheduling and Staffing Multiple Projects with a Multi-Skilled Workforce. *OR Spectrum*, **32**, 343-368. <https://doi.org/10.1007/s00291-009-0169-4>
- [5] Correia, I., Lourenço, L.L. and Saldanha-da-Gama, F. (2012) Project Scheduling with Flexible Resources: Formulation and Inequalities. *OR Spectrum*, **34**, 635-663. <https://doi.org/10.1007/s00291-010-0233-0>
- [6] Liu, S.S. and Wang, C.J. (2012) Optimizing Linear Project Scheduling with Multi-Skilled Crews. *Automation in Construction*, **24**, 16-23. <https://doi.org/10.1016/j.autcon.2011.12.009>
- [7] Nie, H. and Liu, B. (2013) Combining MILP with Memetic Algorithm for Scheduling and Staffing Construction Project with a Multi-skilled Workforce. 2013 *International Conference on Computational and Information Sciences*, Shiyang, 21-23 June 2013, 1150-1153. <https://doi.org/10.1109/ICCIS.2013.305>
- [8] Kazemipoor, H., Tavakkoli-Moghaddam, R., Shahnazari-Shahrezaei, P., *et al.* (2013) A Differential Evolution Algorithm to Solve Multi-Skilled Project Portfolio Scheduling Problems. *The International Journal of Advanced Manufacturing Technology*, **64**, 1099-1111. <https://doi.org/10.1007/s00170-012-4045-z>
- [9] Yannibelli, V. and Amandi, A. (2013) Project Scheduling: A Multi-Objective Evolutionary Algorithm That Optimizes the Effectiveness of Human Resources and the Project Makespan. *Engineering Optimization*, **45**, 45-65. <https://doi.org/10.1080/0305215X.2012.658782>
- [10] Gutjahr, W.J., Katzensteiner, S., Reiter, P., *et al.* (2008) Competence-Driven Project Portfolio Selection, Scheduling and Staff Assignment. *Central European Journal of Operations Research*, **16**, 281-306. <https://doi.org/10.1007/s10100-008-0057-z>
- [11] Attia, E.A., Duquenne, P. and Le-Lann, J.M. (2014) Considering Skills Evolutions in Multi-Skilled Workforce Allocation with Flexible Working Hours. *International Journal of Production Research*, **52**, 4548-4573. <https://doi.org/10.1080/00207543.2013.877613>
- [12] 陈蓉, 梁昌勇, 叶春森, 等. 考虑随机离职的新产品研发项目组组合多技能员工调度模型[J]. 系统工程理论与实践, 2018, 38(1): 164-176.
- [13] Jaber, M.Y. and Bonney, M. (1996) Production Breaks and the Learning Curve: The Forgetting Phenomenon. *Applied Mathematical Modelling*, **20**, 162-169. [https://doi.org/10.1016/0307-904X\(95\)00157-F](https://doi.org/10.1016/0307-904X(95)00157-F)
- [14] 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+1730.
- [15] Chen, R., Liang, C., Gu, D., *et al.* (2017) A Multi-Objective Model for Multi-Project Scheduling and Multi-Skilled Staff Assignment for IT Product Development Considering Competency Evolution. *International Journal of Production Research*, **55**, 6207-6234. <https://doi.org/10.1080/00207543.2017.1326641>
- [16] Kelley, J.E. (2003) The Critical-Path Method: Resource Planning and Scheduling. *Industrial Scheduling*, **1963**, 399-421.
- [17] 陈小平, 于盛林. 实数遗传算法交叉策略的改进[J]. 电子学报, 31(1): 71-74.
- [18] Deb, K. and Jain, H. (2014) An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints. *IEEE Transactions on Evolutionary Computation*, **18**, 577-601. <https://doi.org/10.1109/TEVC.2013.2281535>