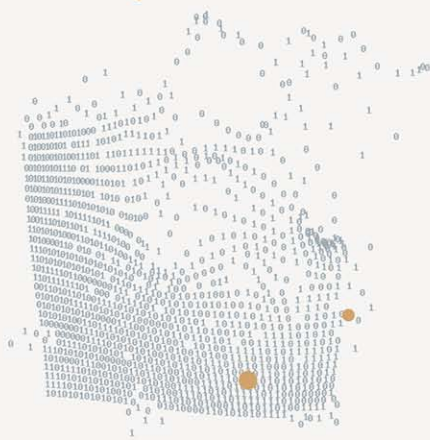


# 项目调度的数学模型 与启发式算法

寿涌毅 著



MATHEMATICAL MODELS  
AND HEURISTICS OF  
PROJECT SCHEDULING

ZHEJIANG UNIVERSITY PRESS  
浙江大学出版社

## 图书在版编目(CIP)数据

项目调度的数学模型与启发式算法 / 寿涌毅著. --  
杭州 : 浙江大学出版社, 2019.12  
ISBN 978-7-308-19924-7

I. ①项… II. ①寿… III. ①项目管理 - 数学模型 -  
启发式算法 - 研究 IV. ①F224.5

中国版本图书馆CIP数据核字(2020)第006899号

## 项目调度的数学模型与启发式算法

寿涌毅 著

---

责任编辑 陈佩钰  
文字编辑 陈逸行  
责任校对 高士吟  
封面设计 棱智广告  
出版发行 浙江大学出版社  
(杭州市天目山路148号 邮政编码310007)  
(网址: <http://www.zjupress.com>)  
排 版 杭州朝曦图文设计有限公司  
印 刷 杭州高腾印务有限公司  
开 本 710mm×1000mm 1/16  
印 张 15.25  
字 数 225千  
版 印 次 2019年12月第1版 2019年12月第1次印刷  
书 号 ISBN 978-7-308-19924-7  
定 价 68.00元

---

版权所有 翻印必究 印装差错 负责调换

浙江大学出版社市场运营中心联系方式: 0571-88925591; <http://zjdxcbbs.tmall.com>

# 前 言

在现代商业社会与人们的日常生活中,项目无所不在。事实上,项目管理实践至少可以追溯到4500年前,古埃及人留下的令人称奇的金字塔就是醒目的证据。即便是多项目管理,也至少可以追溯到2000年前,雄伟的长城就是多项目管理的杰作。现代项目管理是在第二次世界大战之后发展起来的,阿波罗登月计划就是非常典型的现代项目,这时人们已经开始运用运筹学来指导项目管理实践,尤其是在项目调度领域发展出了各种技术。

笔者从博士阶段开始进行项目管理领域的学术研究,任教至今一直从事项目管理领域的教学与科研。本书是笔者在项目调度领域的第二本著作,已出版的第一本著作《资源受限多项目调度的模型与方法》专注于项目调度问题的数学模型和求解算法,本书对第一本著作做了修订和更新,修正了前书的错漏之处,对项目调度问题做了拓展并相应补充了粒子群优化算法,在粒子群优化算法与蚁群优化算法基础上进一步构建了更一般化的多智能体优化算法架构。此外,本书还将项目调度问题拓展到资源组合选择与调度问题,定义了相关数学模型并给出了相应的求解算法。本书是笔者在项目调度研究领域学术成果的汇总,也希望能对该领域有所贡献。

本书共三大部分。第一部分包括第1章到第3章,介绍项目调度研究所涉及的问题。第1章从项目生命周期角度,解释了项目调度的前因后果。第2章对项目调度问题涉及的各种要素逐一进行回顾,包括任务工期、约束条件,以及常见的各类项目调度目标函数,从而建立起单项目调度问题数学模型,并在此基础上拓展到多项目调度问题数学模型。第3章综述了现有文献中的各类项目调度问题特征参数,介绍了常用的单项目调度算例库,并采用全因子设计方法,构造了多项目调度算例库。

第二部分包括第4章到第6章,覆盖了项目调度的各类传统网络技术和简单启发式算法。第4章为项目网络时间分析。第5章阐述基于优先规则的启发式算法,详细分析常见的进度生成机制和各类优先规则,以及基于进度生成机制的多项目调度抽样算法。第6章则在逆向进度生成机制基础上给出正向逆向迭代算法,该算法可用于改进已有的项目进度计划。

第三部分包括第7章到第12章,介绍了项目调度领域的元启发式算法以及对一些扩展问题的求解。第7章介绍项目调度领域最常见的元启发式算法,即遗传算法,详细分析各种可行的染色体编码方案与各类遗传算子。第8章介绍群体智能算法中较常见的蚁群优化算法,并构造了用于多目标项目调度的多种群蚁群算法。第9章介绍粒子群优化算法,并构造了用于抢占式项目调度的粒子群优化算法。第10章介绍多智能体优化算法,给出了适用于抢占式项目调度的多智能体优化系统架构设计。第11章针对多项目调度问题设计了多单位组合拍卖机制,用于分解多项目调度问题,从而更有效地进行资源分配和进度安排。第12章拓展介绍了项目组合选择与调度问题,并构造了用于求解问题的双层决策算法。

在此要特别感谢笔者历年来的研究生,包括傅奥、李敏、王伟、姚伟建、宋淳江、赖昌涛、李菲、王承超、彭晓峰、李盈、相雯雯、胡文锦等。他们参与了相关的研究课题,本书的算法和文字中也包含了他们的不懈努力和智力贡献。

还要感谢国家自然科学基金面上项目、国家自然科学基金青年科学基金项目、浙江省自然科学基金杰出青年项目及教育部留学回国人员科研启动基金提供科研资助。感谢浙江大学管理学院为本书的出版所提供的支持和帮助。最后,本书的顺利出版离不开浙江大学出版社各位编辑的辛勤工作,在此一并致谢。

# 目 录

## 第 1 章 项目计划与项目调度 / 1

- 1.1 项目计划 / 1
- 1.2 项目调度 / 4
- 1.3 多项目管理 / 4

## 第 2 章 项目调度问题数学模型 / 7

- 2.1 任务工期 / 7
  - 2.1.1 确定型时间估计 / 8
  - 2.1.2 概率型时间估计 / 9
- 2.2 约束条件 / 10
  - 2.2.1 优先关系 / 10
  - 2.2.2 资源约束 / 14
  - 2.2.3 时间约束 / 16
- 2.3 目标函数 / 17
  - 2.3.1 时间类目标函数 / 18
  - 2.3.2 资源类目标函数 / 20
  - 2.3.3 财务类目标函数 / 21
  - 2.3.4 质量类目标函数 / 21
  - 2.3.5 常规目标函数与非常规目标函数 / 21

2.4 单项目调度问题数学模型 / 23

2.4.1 基本模型 / 23

2.4.2 拓展模型 / 25

2.4.3 分类系统 / 27

2.5 多项目调度问题数学模型 / 31

**第3章 项目调度问题特征参数与算例库 / 36**

3.1 单项目调度问题特征参数 / 36

3.2 单项目调度算例库 / 40

3.2.1 Patterson算例库 / 40

3.2.2 PSPLIB算例库 / 41

3.3 多项目调度问题特征参数 / 43

3.4 多项目调度算例库 / 46

3.4.1 构造方式 / 47

3.4.2 全因子多项目算例库 / 48

**第4章 项目网络时间分析 / 50**

4.1 任务开始时间 / 50

4.1.1 正向计算 / 51

4.1.2 逆向计算 / 52

4.2 任务时差 / 54

4.2.1 总时差 / 55

4.2.2 自由时差 / 55

4.2.3 安全时差 / 56

4.3 扩展分析 / 57

**第5章 基于优先规则的启发式算法 / 58**

5.1 进度生成机制 / 59

- 5.1.1 串行进度生成机制 / 59
- 5.1.2 并行进度生成机制 / 63
- 5.1.3 多项目进度生成机制 / 66
- 5.2 任务优先规则 / 69
- 5.3 简单启发式算法 / 76
- 5.4 多项目调度抽样算法 / 78
  - 5.4.1 算法设计 / 78
  - 5.4.2 算法测试与分析 / 80

## 第6章 正向逆向迭代算法 / 83

- 6.1 正向逆向调度 / 83
  - 6.1.1 逆向调度算法 / 83
  - 6.1.2 迭代算法 / 87
- 6.2 改进型迭代算法 / 91
  - 6.2.1 基于局部搜索的迭代算法 / 91
  - 6.2.2 逆向正向改进算法 / 92
  - 6.2.3 改进算法的比较与应用 / 94
- 6.3 多项目调度迭代算法 / 95
  - 6.3.1 算法流程 / 95
  - 6.3.2 算法测试与分析 / 96

## 第7章 项目调度遗传算法 / 98

- 7.1 遗传算法 / 98
  - 7.1.1 简单遗传算法 / 99
  - 7.1.2 选择算子 / 101
  - 7.1.3 交叉算子 / 103
  - 7.1.4 变异算子 / 105
- 7.2 单项目调度遗传算法 / 107

7.2.1 编码方案 / 107

7.2.2 其他部分 / 110

7.3 多项目调度遗传算法 / 112

7.3.1 设计方案 / 112

7.3.2 算法测试与分析 / 114

## 第8章 项目调度蚁群算法 / 117

8.1 蚁群算法 / 117

8.1.1 蚂蚁系统 / 118

8.1.2 蚁群系统 / 121

8.1.3 蚁群优化算法 / 123

8.1.4 蚁群算法的改进 / 124

8.2 单项目调度蚁群优化算法 / 126

8.3 多目标项目调度蚁群算法 / 128

8.3.1 多目标项目调度问题 / 128

8.3.2 多种群蚁群算法 / 129

8.3.3 算法测试与分析 / 132

## 第9章 项目调度粒子群优化算法 / 135

9.1 粒子群优化算法 / 135

9.1.1 粒子群优化 / 135

9.1.2 算法流程 / 137

9.2 抢占式项目调度 / 139

9.3 抢占式项目调度粒子群优化算法 / 142

9.3.1 编码方案 / 142

9.3.2 进度生成机制 / 144

9.3.3 适应值函数 / 145

9.3.4 粒子更新机制 / 145

## 9.3.5 算法测试与分析 / 147

**第 10 章 项目调度多智能体优化算法 / 154**

## 10.1 多智能体优化 / 155

## 10.2 抢占式项目调度多智能体优化算法 / 157

## 10.2.1 基础服务平台 / 158

## 10.2.2 智能体交互系统 / 158

## 10.2.3 协商过程 / 159

## 10.2.4 调度智能体 / 160

## 10.2.5 任务智能体 / 163

## 10.3 计算示例 / 164

## 10.3.1 初始化阶段 / 164

## 10.3.2 再调度阶段 / 166

## 10.3.3 迭代改进阶段 / 175

**第 11 章 多项目调度拍卖算法 / 176**

## 11.1 拍卖理论与拍卖算法 / 176

## 11.1.1 拍卖理论 / 176

## 11.1.2 组合拍卖与优化 / 178

## 11.2 多单位组合拍卖算法 / 182

## 11.2.1 拍卖品组合 / 182

## 11.2.2 竞买人效用 / 184

## 11.2.3 拍卖人效用 / 185

## 11.2.4 组合拍卖流程 / 186

## 11.3 计算示例 / 187

**第 12 章 项目组合选择调度问题及算法 / 191**

## 12.1 数学模型 / 192

- 12.2 双层决策算法 / 194
  - 12.2.1 上层决策 / 194
  - 12.2.2 下层决策 / 197
- 12.3 计算测试与分析 / 198
  - 12.3.1 算例设计 / 198
  - 12.3.2 算法参数设计 / 199
  - 12.3.3 计算结果分析 / 200
- 参考文献 / 202

# 第1章 项目计划与项目调度

项目管理的重要性已经毋庸置疑。根据统计,全球经济活动有30%是采用项目形式执行的(Turner, 2008)。按项目方式管理(management-by-projects)在全世界范围内日益流行,也使得项目管理成为商学研究的重要领域。Kwak和Anbari(2009)对英国《金融时报》认定的商学研究顶级期刊进行了统计,发现其中有16种都涉及项目管理。

Kolltveit等(2007)则对项目管理研究领域的文献做了内容分析,发现多数学者都非常重视任务视角的研究,尤其强调项目的计划与控制。事实上,项目管理最重要的工作,就是项目的计划、调度、控制,这三部分也恰恰就是Kerzner(2017)著名的项目管理手册副标题的三个关键词。

## 1.1 项目计划

任何一个项目都会经过不同的发展阶段,因此可以区分其生命周期。但是,鉴于项目的复杂性和多样性,目前并没有统一的项目生命周期划分标准(Kerzner, 2017)。Demeulemeester和Herroelen(2002)认为可以将项目生命周期划分为五个阶段,分别是概念阶段、定义阶段、计划阶段、调度阶段、控制阶段。

在概念阶段,需要明确项目需求。在明确需求后,就转入定义阶段,首先就是要确定项目目标(objectives)。项目目标可能是开发新技术,可能是占领新市场,也可能是降低生产成本,不一而足。很多情况下,项目不仅仅只有一个目标,而是有多个互相关联的目标。比如,加速新产品研发和占领新兴市场就是两个互相关联的目标。有的时候,目标非常明确,比如在年末降低

10%的生产成本;但有的时候目标又比较模糊,比如在市场上领先于竞争对手。通常,项目经理无法同时满足所有目标,因此需要给这些目标赋予不同的优先级。以比较数学的术语来说,项目经常是具有不同权重的模糊的多目标问题。

在确定项目目标后,需要考虑如下四个问题(Kerzner,2017):(1)要实现项目目标,需要完成哪些主要工作,这些工作要素是如何互相关联的?(2)哪些职能部门应当为达成项目目标负责,为落实主要工作负责?(3)公司或组织是否能够提供项目所需的资源?(4)项目的信息流是什么样的?

这四个问题非常基础和重要。第一个问题涉及如何分解项目目标,并确定为实现项目目标所必须完成的各项工作。通常,可以用工作说明(statement of work,SOW)对这些工作和工作范围加以描述。Kerzner(2017)阐述了SOW对于项目管理的重要性和编制SOW的基本流程。在有必要的时候,还可以为SOW编制单独的项目明细表(project specifications),以确认关于设备、材料、工时的各项估计。毫无疑问,项目明细表的细小变动都有可能显著影响项目进度和项目成本。为了进一步分析项目的主要工作,尤其是工作之间的相互关系,一般需要开发项目的工作分解结构(work breakdown structure,WBS)。编制WBS是项目计划过程中非常重要的一步。WBS是以项目交付物(deliverables)为导向的一个树状结构,用于分解整个项目的各项工作,并为后续项目调度与控制提供基础。关于编制WBS的详细说明,可以参考美国项目管理协会(Project Management Institute,PMI)的WBS实践标准(Project Management Institute,2001)。

在完成SOW和WBS的编制后,需要落实执行WBS各项工作的职能部门。经常可以在WBS基础上,结合组织结构确定职责矩阵(Kerzner,2017)。职责矩阵(responsibility matrix)也经常被称为直线责任图(linear responsibility chart,LRC)或职责分配矩阵(responsibility allocation matrix,RAM)。职责矩阵的目的,在于将WBS中的各项工作分配给具体的部门、岗位或者个人,并明确各参与者的不同职责。

在明确了各职能部门的职责后,需要确定是否有项目所需的资源来支持各责任部门即将开展的各项工作。项目通常会涉及的资源主要包括资金、人

力、设备、设施、材料、信息及技术。项目经理首先要在WBS的基础上明确项目需要哪些资源,也就是编制项目资源需求(resource requirement)计划,用于说明WBS各组成部分所需资源的种类和数量。因为项目经理并不直接控制上述所需资源,而且这些资源往往为不同的部门所调用,所以如何确保资源在所需时间准时到位就成为一个关键问题。因此,就需要知道各类资源的所需时间,这就要求项目经理需要编制项目进度计划,以回答不同任务分别在什么时候开始。这一过程,就是项目调度(project scheduling)过程,通常是在WBS和资源需求计划基础上进行的。但是,由于项目进度计划所需的资源未必都能非常理想地得到满足,所以通常还需要参照资源供给情况进行调整,也就是再计划。其中,人力资源是非常特殊的一类资源,因而人力资源配备(staffing)也成为特别棘手的一个问题(Tukel and Rom,1998)。但是,由于传统的网络计划技术只能进行时间分析,资源调度(resource scheduling)常常为项目经理所忽视(Herroelen,2005)。

在完成工作分解、职责分配及资源需求计划后,还有一项重要的工作,就是信息管理。这也容易被许多项目组织所忽视。由于项目的各组成部分和资源往往分散在不同职能部门,甚至不同企业,而这些任务之间又需要相互沟通和反馈,因此对项目的信息流加以规划和管理,就成为影响项目执行效率的重要因素。在项目进度计划中,通常只有具体的工作才分配工时。传统的项目管理软件并不提供信息流管理方面的支持。但是,Brooks(1995)指出,信息沟通所消耗的时间,恰恰是造成“人月神话”的重要原因。目前,项目信息流管理工具还不是很多,设计结构矩阵(design structure matrix,DSM)是这方面的重要发展(Browning,2001,2016;Yassine,2004)。

项目计划非常重要。《中庸》说,“凡事豫则立,不豫则废”。西方也有类似的谚语,“If you fail to plan,you plan to fail”。但是,世上并不存在完美的计划。虽然对于项目成功和失败并没有一致的理解(Baccarini,1999;Kerzner,2017),但是实证研究表明项目失败率非常显著(El Emam and Koru,2008),因此人们难免有“为什么项目总是失败”的感叹(Matta and Ashkenas,2003)。Pinto和Mantel(1990)发现计划失效是项目失败的重要原因之一,对于研发项目而言,不合理的调度尤其容易导致项目失败。

## 1.2 项目调度

项目计划阶段的主要工作在于明确需要执行哪些任务以达成项目目标。而项目调度则在于如何有效地完成上述任务。项目调度是项目计划的下一个阶段,所提供的项目基准计划(baseline plan)则成为后续项目控制的基准。简言之,项目调度本质上就是安排WBS中所列举的各项任务的开始时间,并为任务分配所需的资源。

从早期的项目管理技术开始,项目调度就成为许多项目管理软件的基本功能。传统的项目计划方法,如关键路线法(critical path method,CPM)或计划评审技术(project evaluation and review technique,PERT),在编制初始计划时并不考虑有限资源对项目进度的负面影响,而是在形成初始计划后再考虑资源约束,进而对时间计划进行后续调整。这样的计划方式所得的进度计划往往不是最优的方案。因此,在CPM与PERT之后,学术界与产业界已开发出各类精确算法或启发式算法,对实际项目调度问题进行更合理的求解。

如何编制项目基准计划,使其满足任务之间的优先关系,能够有效利用项目组织的有限资源,并且实现项目的既定目标,这就是项目调度阶段的主要工作和最大挑战,也是本书后续各章的主题。

## 1.3 多项目管理

当我们谈论项目管理时,容易陷入一个误区,就是以单个的项目为分析对象,假设它们可以成为孤立的封闭系统,从而使我们的各种分析和决策都能够尽量简化。但现实并非如此。根据Lova和Tormos(2001)在西班牙瓦伦西亚地区的调查,84%的企业在多项目环境中开展项目作业,项目之间共享各类资源。Turner(2008)则估计至少有90%的项目是在多项目环境中执行的。

多项目(multiple projects/multi-project)是指同时进行的多个项目,这些项目的规模、重要性、进展状况可能各不相同,但是共享一组有限资源

(Fricke and Shenhar, 2000)。多项目也经常被称为项目组合(portfolio)。与多项目相类似的一个概念是项目群(program),也译为项目集,是指若干有内在联系的项目,为了组织利益而进行集成与管理(Turner, 2008)。而当这些项目分别独立管理时,由于项目之间的相互联系,组织利益无法得到有效保障(Lycett et al., 2004)。比较项目组合与项目群两个概念的差异,可以发现,项目群有着共同的交付物(产出),而项目组合则有着共同的资源投入(Turner, 2008)。一个项目群也可以是一个项目组合的构成部分(Project Management Institute, 2006)。

程铁信等(2004)回顾了国际项目管理协会(International Project Management Association, IPMA)自20世纪60年代以来历次国际会议的主题,发现项目管理研究热点从网络计划技术、项目管理实施、项目管理组织逐步向集成化项目管理发展。而对于集成化项目管理来说,多项目管理就成为非常核心的问题。

与单项目管理相比,多项目管理更加困难重重,面临更高的失败风险(Gutjahr, 2015)。Van Der Merwe(1997)指出,多项目管理最主要的风险即来自缺乏对项目的协调与控制。而项目协调与控制的困难,很大程度上来自多项目管理中管理界面增加所导致的巨大复杂性(Payne, 1995)。Engwall and Jerbrant(2003)通过案例研究指出,资源分配是多项目管理的最大挑战,因为存在多个项目之间的资源竞争和冲突。Fricke和Shenhar(2000)的案例研究也证实,资源管理是多项目管理与单项目管理的重要区别。这就涉及多项目调度中非常核心的资源配置问题。如果进一步将环境动态性纳入考虑,那么由于项目之间的竞争造成不确定性进一步上升,多项目管理将更加困难(Eskerod, 1996)。因此,面对这样的动态复杂性,需要对项目进行系统思考(Yeo, 1993)。

由于多项目管理的复杂性,就比较容易理解为什么存在大量有关单项目调度的文献,而多项目调度方面的文献则相对较少(Herroelen, 2005),尽管项目实践中大量存在着多项目环境。事实上,多项目调度可以在许多场合得到应用,例如涉及多个项目之间的人员调配(Dean et al., 1992),新产品开发项目之间的资源分配(方炜、欧立雄, 2005),甚至于互联网服务器的计算资源配

置都可以看作一个多项目调度问题(Lee and Lei,2001)。多项目调度也在企业实践中取得了一定成效,例如Bowers等(1996)尝试在服装制造企业中推广多项目调度方法,研究发现多项目调度方法能够使案例企业劳动生产率提高5.1%。

基于以上考虑,本书尝试对项目调度进行研究,不仅考虑单项目调度,同时也针对多项目调度的特点设计相应的多项目调度数学模型与求解算法,以期对提升项目管理绩效有所贡献。

## 第2章 项目调度问题数学模型

任何一个项目都需要完成一系列具体的任务,这些任务之间一般存在逻辑上的优先关系,同时每项任务的执行又需要特定的资源。项目调度就是要合理分配这些资源,恰当地安排这些任务的开始时间,从而优化既定的一个或若干个目标函数。

资源受限项目调度问题(resource-constrained project scheduling problem, RCPSP),或称资源约束下项目调度问题,即要求在满足项目内部优先关系和资源约束的前提下,合理安排项目的进度计划,从而优化项目预期目标。

本章旨在建立资源受限项目调度问题的数学模型,包括多项目调度问题的模型。从项目调度问题涉及的任务工期、约束条件及目标函数出发,首先介绍常见的任务工期估计方法,然后对各类约束条件进行详细综述,总结各类项目调度目标函数,并在此基础上给出资源受限项目调度问题的数学模型。最后,进一步讨论多项目调度问题的特点,并给出资源受限多项目调度问题的数学模型。

### 2.1 任务工期

项目一般由一系列的任务(task)构成。在不同情景中,任务也可能被称为活动(activity)或工作(job)。当一个项目通过工作分解结构(WBS)明确了所需要执行的任务之后,需要分析各个任务的属性以及任务之间的相互关系,然后才可以对任务进行调度,从而编制合理的项目进度计划。在编制WBS时即需要明确的任务属性,主要包括:任务工期、任务所需资源、任务所