

信息系统学报

CHINA JOURNAL OF INFORMATION SYSTEMS

第3卷第1辑 (总第4辑)

Vol.3 No.1, April 2009

- ◆ 一种求解多模式资源受限项目调度问题的蚁群算法 / 彭武良
- ◆ 业务流程中概念模型到逻辑模型的转换 / 刘曼路, 范绍坤, 赵建良, 窦万春
- ◆ 企业知识的载体、交汇与转载 / 陈智高, 郭文婷
- ◆ 现代服务企业e就绪对电子商务能力的影响: 基于企业资源观的实证研究 / 朱镇, 赵晶
- ◆ 基于ELM的网上用户初始信任影响因素研究 / 周涛, 鲁耀斌
- ◆ 市场、层级与联合: 信息技术对经济组织影响的重新评估 / REIMERS Kai
- ◆ 中国信息系统的国际研究分析及对比 / 卢向华, 冯骏, 黄丽华
- ◆ 国内外信息系统外包理论研究综述 / 李东, 李猛
- ◆ 从NSFC资助项目看我国电子商务的基础研究 / 冯芷艳
- ◆ 电子商务——现状和展望 / 陈禹
- ◆ 新兴电子商务中的安全问题 / 冯登国
- ◆ 中国电子商务的发展及重要研究课题 / 于刚
- ◆ 电子商务大规模发展与运营的特征和趋势 / 梁春晓
- ◆ “移动营销”促进传统服务业实现跨越式发展 / 汤海京

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

信息系统学报.第3卷.第1辑:总第4辑/清华大学经济管理学院编.——北京:清华大学出版社,2009.6
ISBN 978-7-302-20219-6

I. 信… II. 清… III. 信息系统—丛刊 IV. G202-55

中国版本图书馆 CIP 数据核字(2009)第 079849 号

责任编辑:贺 岩

责任校对:王凤芝

责任印制:杨 艳

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:北京市昌平环球印刷厂

装 订 者:北京国马印刷厂

经 销:全国新华书店

开 本:205×282 印 张:7.5 字 数:204 千字

版 次:2009 年 6 月第 1 版 印 次:2009 年 6 月第 1 次印刷

印 数:1~2000

定 价:30.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:032566-01

《信息系统学报》编委会

主 编

陈国青(清华大学)

副主编

黄丽华(复旦大学)

李 东(北京大学)

李一军(哈尔滨工业大学)

毛基业(中国人民大学)

王刊良(西安交通大学)

编辑部主任

黄京华(清华大学)

主编助理

郭迅华(清华大学)

卫 强(清华大学)

编 委

陈华平(中国科技大学)

陈 剑(清华大学)

陈晓红(中南大学)

陈 禹(中国人民大学)

党延忠(大连理工大学)

甘仞初(北京理工大学)

李敏强(天津大学)

刘 鲁(北京航空航天大学)

刘仲英(同济大学)

马费成(武汉大学)

邵培基(电子科技大学)

谢 康(中山大学)

严建援(南开大学)

杨善林(合肥工业大学)

张金隆(华中科技大学)

张朋柱(上海交通大学)

仲伟俊(东南大学)

CHAU Patrick Y. K. (University of Hong Kong)

CHEN Yeshe (Louisiana State University)

HUANG Wayne (Ohio University)

LIANG Ting-Peng (Sun Yat-Sen University, Taiwan)

LU Jie (University of Technology, Sydney)

SHENG Olivia (Utah University)

TAN Bernard (National University of Singapore)

TAN Felix B. (AUT University)

THONG James Y. L. (Hong Kong University of Science & Technology)

WEI Kowk Kee (City University of Hong Kong)

ZHAO Leon (University of Arizona)

ZHU Kevin (University of California, San Diego)

Editorial Board, China Journal of Information Systems

Editor	CHEN Guoqing (Tsinghua University)
Associate Editors	HUANG Lihua (Fudan University) LI Dong (Peking University) LI Yijun (Harbin Institute of Technology) MAO Jiye (Renmin University of China) WANG Kanliang (Xi'an Jiaotong University)
Managing Editor	HUANG Jinghua (Tsinghua University)
Assistants to the Editor	GUO Xunhua (Tsinghua University) WEI Qiang (Tsinghua University)
Members of Editorial Board	CHAU Patrick Y. K. (University of Hong Kong) CHEN Huaping (University of Science & Technology of China) CHEN Jian (Tsinghua University) CHEN Xiaohong (Central South University) CHEN Yeshe (Louisiana State University) CHEN Yu (Renmin University of China) DANG Yanzhong (Dalian University of Technology) GAN Renchu (Beijing Institute of Technology) HUANG Wayne (Ohio University) LI Minqiang (Tianjin University) LIANG Ting-Peng (Sun Yat-Sen University, Taiwan) LIU Lu (Beihang University) LIU Zhongying (Tongji University) LU Jie (University of Technology, Sydney) MA Feicheng (Wuhan University) SHAO Peiji (University of Electronic Science & Technology of China) SHENG Olivia (Utah University) TAN Bernard (National University of Singapore) TAN Felix B. (AUT University) THONG James Y. L. (Hong Kong University of Science & Technology) WEI Kowk Kee (City University of Hong Kong) XIE Kang (Sun Yat-Sen University, Guangzhou) YAN Jianyuan (Nankai University) YANG Shanlin (Hefei University of Technology) ZHANG Jinlong (Huazhong University of Science & Technology) ZHANG Pengzhu (Shanghai Jiao Tong University) ZHAO Leon (University of Arizona) ZHONG Weijun (Southeast University) ZHU Kevin (University of California, San Diego)

主 编 的 话

时值繁花似锦的 2009 年春季,《信息系统学报》总第 4 辑在全国各地信息系统领域研究者的积极支持以及海内外学者的广泛关注下与大家见面了。自创办以来,《信息系统学报》影响日渐扩大,已被信息系统协会(Association for Information Systems, AIS)中国分会(CNAIS)指定为会刊,并全文收入《中国期刊全文数据库》及《中国学术期刊综合评价数据库》,同时成为中国人民大学《复印报刊资料》来源刊。这些成绩既来源于主办及各协办单位的努力投入,也与广大同仁的关心和支持密不可分。

本辑学报秉承前 3 辑所形成的强调学术前沿与严谨研究方法的风格,致力于体现管理与技术并重的特点,收录了多篇国内外学者的最新研究成果,其内容涵盖多个不同的研究主题和多样化的方法论。其中,Kai Reimers 从组织理论的视角探讨了信息技术对经济组织的多方面影响;刘曼路、范绍坤、赵建良等的研究工作则从设计科学的角度对业务流程中概念模型到逻辑模型的转换进行了讨论;陈智高、郭文婷的论文在理论层面上讨论了企业知识的载体、交汇与转载;朱镇、赵晶的文章以及周涛、鲁耀斌的研究工作分别用实证手段研究了现代服务企业 e 就绪对电子商务能力的影响以及网上用户初始信任的影响因素;彭武良的文章在技术方向上提出了一种求解多模式资源受限项目调度问题的蚁群算法。同时,本辑学报还发表了两篇领域研究的综述,即北京大学李东、李猛的对国内外信息系统外包理论研究的综述,以及复旦大学卢向华、冯骏、黄丽华的对中国信息系统研究所作的分析及国际对比。

此外,本辑学报特别开辟了“专家见解”专栏,以“新兴电子商务”为主题,汇集了一批业界和学界的前瞻性观点。近年来,随着我国互联网和移动通信的发展进入全新的阶段,新兴电子商务呈现出移动性、虚拟性、极端数据、个性化以及社会性等鲜明的新特征,带来了一系列重大的、崭新的基础性科学问题以及研究挑战,使得在新形势下重新审视电子商务活动原理与运行规律成为一项紧迫而意义深远的任务。我们希望借助“专家见解”栏目,为推动相关研究的顺利开展添砖加瓦。国家自然科学基金委管理科学部冯芷艳研究员,中国人民大学信息学院陈禹教授,中国科学院信息安全国家重点实验室主任冯登国研究员,上海益实多电子商务有限公司董事长于刚博士,阿里巴巴集团高级研究员、副总裁梁春晓先生,北京闻言科技有限公司董事长汤海京先生分别发表了其真知灼见。今后,《信息系统学报》也将在此栏目中不定期地发表各种相关主题的讨论。我们相信这样的交流,必将有助于学界和业界的互动,共同促进我国信息系统研究与实践的蓬勃发展。

谨向关心和支持《信息系统学报》的国内外学者同仁及各界人士致以深深的谢意。同时感谢参与稿件评审的各位专家的细致工作,并对清华大学出版社在编辑和出版过程中的辛勤工作深表谢意!

主 编 陈国青

副主编 黄丽华 李 东 李一军 毛基业 王刊良

2009 年 4 月于北京

信息系统学报

第 3 卷第 1 辑(总第 4 辑)

目 录

主编的话	V
一种求解多模式资源受限项目调度问题的蚁群算法/彭武良	1
业务流程中概念模型到逻辑模型的转换/刘曼路,范绍坤,赵建良,窦万春	11
企业知识的载体、交汇与转载/陈智高,郭文婷	25
现代服务企业 e 就绪对电子商务能力的影响:基于企业资源观的实证研究/朱镇,赵晶	34
基于 ELM 的网上用户初始信任影响因素研究/周涛,鲁耀斌	48
市场、层级与联合:信息技术对经济组织影响的重新评估/REIMERS Kai	57
 领域综述	
中国信息系统的国际研究分析及对比/卢向华,冯骏,黄丽华	75
国内外信息系统外包理论研究综述/李东,李猛	85
 专家通讯:新兴电子商务专题	
从 NSFC 资助项目看我国电子商务的基础研究/冯芷艳	98
电子商务——现状和展望/陈禹	101
新兴电子商务中的安全问题/冯登国	103
中国电子商务的发展及重要研究课题/于刚	104
电子商务大规模发展与运营的特征和趋势/梁春晓	107
“移动营销”促进传统服务业实现跨越式发展/汤海京	109
 学术动态	 111
院校介绍	113

China Journal of Information Systems

Vol. 3 No. 1, April 2009

CONTENTS

EDITORS' NOTES	V
-----------------------------	---

ARTICLES

ACO for Solving MMRCPSPP/ <i>PENG Wuliang</i>	1
Transformation of Conceptual to Logical Business Process Models / <i>LIU Manlu, FAN Shaokun, ZHAO Jianliang & DOU Wanchun</i>	11
Research on the Repositories, Convergence and Trans-repository of Enterprise Knowledge / <i>CHEN Zhigao & GUO Wenting</i>	25
The Impact of E-readiness on E-business Capability in Modern Service Industry: An Empirical Study from Resource-based View/ <i>ZHU Zhen & ZHAO Jing</i>	34
Factors Impacting Online User Initial Trust based on ELM/ <i>ZHOU Tao & LU Yaobin</i>	48
Markets, Hierarchies and Associations: Reassessing the Impact of Information Technology on the Organization of Economic Activity/ <i>REIMERS Kai</i>	57

REVIEWS

International Information Systems Research On China / <i>LU Xianghua, FENG Jun & HUANG Lihua</i>	75
A Literature Review on Information Systems Outsourcing Research/ <i>LI Dong & LI Meng</i>	85

COMMUNICATIONS: EMERGING E-COMMERCE

NSFC Grants and E-commerce Research in China/ <i>FENG Zhiyan</i>	98
E-commerce: Current Status and Future Trends/ <i>CHEN Yu</i>	101
Security Issues in Emerging E-commerce/ <i>FENG Dengguo</i>	103
Development of E-commerce in China and Key Research Issues/ <i>YU Gang</i>	104
Operational Characteristics and Development Trends in Large Scale E-commerce / <i>LIANG Chunxiao</i>	107
Traditional Service Sectors Forging ahead with Mobile Marketing/ <i>TANG Haijing</i>	109

EVENTS	111
---------------------	-----

SCHOOL INTRODUCTION	113
----------------------------------	-----

一种求解多模式资源受限项目调度问题的蚁群算法*

彭武良

(沈阳理工大学经济管理学院, 沈阳 110168)

摘要 多模式资源受限项目调度问题是一种著名的 NP 完全问题, 本文提出了一种求解该问题的蚁群算法。相对于现有算法, 该算法采用规则池管理和使用大量优先级规则, 提升了算法的通用性和性能。另外, 每只蚂蚁被赋予自治能力、学习能力和预判能力。自治能力体现在每只蚂蚁具有单独的线程, 学习能力体现在蚂蚁可以动态选择更好的优先级规则, 预判能力体现在蚂蚁能够通过分支定界的方法排除不可行路径。最后利用 PSPLIB 标准问题对算法进行了大量的仿真测试, 取得了令人满意的结果。

关键词 多模式资源受限项目调度问题, 计划与调度, 蚁群算法

中图分类号 TP91

1 引言

多模式资源受限项目调度问题是一种著名的 NP 完全问题^[1], 在强资源约束下, 当问题规模超过 20 时, 现有的精确算法已经无能为力^[2-3]。对于该类问题, 比较可行的办法是通过智能优化算法以较小的计算代价求解问题的次优解。蚁群算法是一种群智能优化算法, 近年来在求解项目调度等组合优化问题上的应用引起了研究人员的关注^[4-7]。文献[4]提出了一种基于多代理的蚁群算法求解单模式资源受限项目调度问题 (Single-mode Resource-constrained Project Scheduling Problem, SMRCPSp), 使用了 13 种优先级规则。文献[5]在采用蚁群算法求解 SMRCPSp 时, 采用活动——活动模式定义解构造模型, 结合使用直接评估 (direct evaluation) 和累加评估 (summation evaluation) 的方式确定蚂蚁个体的转移方向。文献[6]使用了 LST 和 WRUP 两个启发式规则, 解构造图采用活动——位置模式, 并在蚂蚁位置转移时平衡了先验知识和新路径探索之间的关系。文献[7]设计了一种基于遗传算法和蚁群算法的元启发式算法, 该算法首先采用遗传算法搜索较好解初始化信息素矩阵, 然后用蚁群算法再次寻优。

目前求解项目调度问题的蚁群算法都是针对 SMRCPSp 进行的。从算法的结构来看, 既有算法都是选择一种或特定几种优先级规则与整个蚁群绑定。我们知道, 优先级规则不具有通用性, 从而也限制了蚁群算法的通用性。另外, 目前既有蚁群算法受其他智能优化算法的影响, 都使用“代”(即算法的一次迭代, 在每次迭代中, 若干蚂蚁同时出发, 同时到达)的概念, 这样处理既与蚂蚁觅食的现实行为不符, 也不利于算法的并行实现。相对于 SMRCPSp, MMRCPSp 更具有普遍意义, 其求解也更困难^[1]。本文根据 MMRCPSp 问题的特点, 提出一种求解该问题的蚁群算法。相对既有的研究, 该算法具有如下几个特点:

* 基金项目: 国家自然科学基金资助项目(60604025)。

通信作者: 彭武良, 沈阳理工大学经济管理学院副教授, e-mail: Peng-wuliang@163.com

- (1) 基于活动列表的编码方式,定义了一种适合 MMRCPSP 问题求解的解构造模型;
- (2) 采用规则池管理大量的优先级规则提升算法的通用性和性能;
- (3) 赋予蚂蚁学习能力:蚂蚁在每次旅行之前,可以从规则池中动态选择新的优先级规则,更适用于问题求解的优先级规则会被越来越多的蚂蚁选择;
- (4) 赋予蚂蚁个体自治能力:为每只蚂蚁设计独立的线程;
- (5) 赋予蚂蚁预判能力,使蚂蚁能够及时排除不可行路径,避免无谓的旅行造成算法时间上的浪费。

2 问题描述

MMRCPSP 的项目计划可采用有向无环图 $G=(V,E)$ 描述,活动用从 1 到 n 的整数标识,即 $j \in V, 1 \leq j \leq n$,其中唯一开始活动 1 和唯一结束活动 n 为虚活动,工期为 0。 $(i,j) \in E$,表示活动 i 为活动 j 的前置活动。对于每个活动 $j \in V$,有一个执行模式集 $M_j = \{1, \dots, |M_j|\}$ 。活动 j 按照模式 m 执行时的活动工期为 d_{jm} 。活动 1 和活动 n 的处理时间为 0。 s_j 和 f_j 分别代表活动 j 的开始和结束时间,活动时间均为整数。项目的开始时间可用活动 1 的开始时间 s_1 表示,项目的结束时间可用活动 n 的结束时间 f_n 表示。项目有可更新资源类型集合 K' ,每种可更新资源 $k \in K'$,有在整个项目周期均可用的资源数量 R^k 。项目有不可更新资源类型集合 K'' ,每种不可更新资源 $k \in K''$,有资源数量上限 N^k 。当活动 j 按照模式 m 执行时,其对可更新资源 k 的需求量用 r_{jmk}^r 表示,其对不可更新资源 k 的需求量用 r_{jmk}^n 表示。同时令在 t 时刻正在执行的活动集合为 $A(t)$ 。则 MMRCPSP 问题模型为:

$$\min f_n \quad (1)$$

$$\text{s. t. } s_j - s_i \geq \sum_{m \in M_i} (x_{im} d_{im}) \quad (i,j) \in E \quad (2)$$

$$\sum_{j \in A(t)} \sum_{m \in M_j} (x_{jm} r_{jmk}^r) \leq R^k \quad 1 \leq t \leq f_n, k = 1, 2, \dots, K' \quad (3)$$

$$\sum_{j \in V} \sum_{m \in M_j} x_{jm} r_{jmk}^n \leq N^k \quad k = 1, 2, \dots, K'' \quad (4)$$

$$\sum_{m \in M_i} x_{im} = 1, i \in V \quad (5)$$

目标函数(1)最小化项目工期。约束条件(2)要求活动的执行时间必须满足紧前关系约束。约束集(3)要求时刻 t 正在执行的活动集合 $A(t)$ 所占用的可更新资源总量不能超过每种资源的总供应量。约束集(4)要求整个项目所占用的每种不可更新资源总量不能超过其总供应量。约束条件(5)要求所有活动必须选择唯一模式 $m \in M_j$ 执行,且执行过程不能中断,其中 x_{im} 为 0-1 决策变量, $x_{im} = 1$ 意味着活动 i 按照模式 m 执行,否则 $x_{im} = 0$ 。显然,当各活动的执行模式选定以后,MMRCPSP 就退化为 SMRCPSP。因此在满足约束条件(5)的情况下,MMRCPSP 和 SMRCPSP 可以采用相同的优先级规则和调度方案。

3 求解 MMRCPSP 问题的蚁群算法

3.1 编码方式和解构造模型

如前所述,MMRCPSP 问题是一个难以求解的 NP 问题,求解算法的执行速度至关重要,本文采

用在效率方面更好的基于活动列表(Activity List Based Encoding)的编码方法。对应基于活动列表的编码方法,定义适合于蚁群算法特点的解构造模型。

解构造模型采用图 $G=(A,P,\Gamma)$ 表示。 A 为活动集合, $|A|=n$, n 为活动总个数。 $j \in A$ 为一个项目活动 ID, 它是蚂蚁在移动过程中需要选择的值。 P 为位置集合, 在活动数为 n 的项目中, $|P|=n$, 位置 $i \in R$ 的意义为蚂蚁所处的位置或蚂蚁移动的步骤。信息素分布 Γ 与解构造图上的节点相关联, 作为解空间参数化概率分布模型的参数, 对于信息素分布矩阵中的一个元素可以通过活动 j 和位置 i 确定, 即 $\tau_{ij} \in \Gamma, i \in P, j \in A, \tau_{ij}$ 的意义为蚁群在第 i 个步骤选择活动 j 时留下的信息素总量。解构造模型可以采用信息素分布矩阵形象地描述, 矩阵的横坐标为 P , 矩阵的纵坐标为 A, Γ 为矩阵中的信息素分布。在 MMRCPSP 中, 因为每个活动可能有多种执行模式, 所以 τ_{ij} 值必须分布在多个执行模式中, 即有:

$$\tau_{ij} = \sum_{m=1}^{M_j} \tau_{ij}(m) \quad (6)$$

其中, $\tau_{ij}(m)$ 为蚁群在第 i 个步骤选择活动 j 的执行模式 m 留下的信息素总量。

一个项目计划实例如图 1 所示, 项目计划包含 6 个活动, 每个活动有 2 个执行模式。信息素分布矩阵的横坐标代表活动列表中的位置, 纵坐标代表活动 ID。蚂蚁共行走 6 步, 依次选择活动 1、3、2、4、5、6, 并选择相应的执行模式, 形成一个活动列表 1:1, 3:1, 2:1, 4:2, 5:2, 6:1。依据该活动列表, 可以按照串行调度方案生成一个完整的计划。

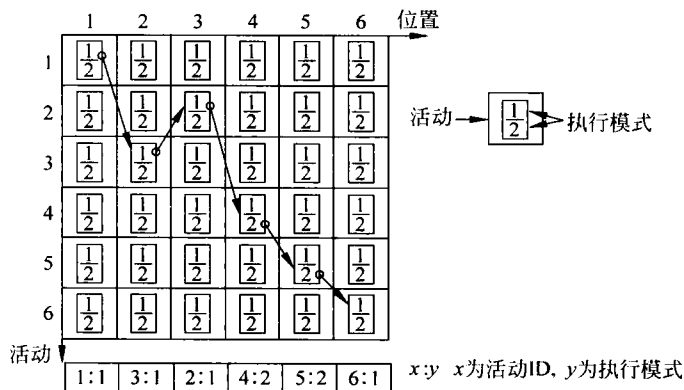


图 1 蚂蚁在信息素分布矩阵中的旅行示例

3.2 蚂蚁个体行为与算法执行步骤

本文摒弃了传统蚁群算法“代”的概念, 将单个蚂蚁视为具有独立行为能力的自治体, 为其设计单独的线程。算法的执行过程如图 2 所示。算法首先初始化若干蚂蚁, 为每个蚂蚁启动单独的线程。在蚂蚁旅行之前, 为蚂蚁个体绑定一个优先级规则, 作为该蚂蚁的行为习惯。蚂蚁在每次旅行完成之后可以改变行为习惯, 通过改变其绑定的优先级规则来实现。蚂蚁在旅行过程中, 通过预判机制选择新的路径。蚂蚁每次完整的旅行过程对应一个活动列表, 将该活动列表输入到串行调度计划生成方案生成项目计划^[8], 并采用对齐技术^[9]进行局部搜索, 以该项目计划的项目工期作为目标函数。最后根据目标函数值在蚂蚁行走的路径上增加信息素。另外, 算法在执行过程中定期对信息素分布矩阵中的信息素进行更新。

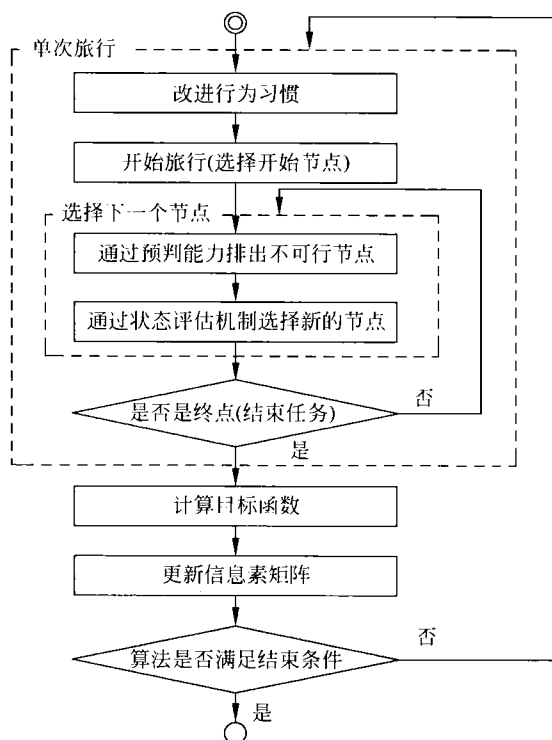


图2 蚂蚁个体的旅行过程

3.3 规则池与蚂蚁改变行为习惯的机制

规则池用于管理大量的优先级规则,并根据优先级规则在算法执行中的应用效果动态更新其被选择概率,使表现较好的优先级规则能够被更多的蚂蚁选择。令 R 为规则池中所维护的规则集合, $g \in R$ 为一个优先级规则,其中 $g=1,2,\dots,|R|$ 为优先级规则的标识。存储在规则池中的优先级规则 g 可用一个三元组表示: $g=(f,\varphi,u)$ 。其中 f 为优先系数的计算方法,优先权系数的计算需要基于项目网络 G ,不同的优先级规则有不同的优先权系数计算方法。 φ 为一个优先级规则的选择概率,蚂蚁在一次巡游开始时,需要依据选择概率 φ 从规则池中随机选取一个优先级规则。显然若规则池中有 $|R|$ 个规则,则有 $\sum_{g=1}^{|R|} \varphi_g = 1$, φ_g 为规则 g 的被选择概率。 u 为效用记录,其值为使用该优先级规则所产生的项目工期的平均值,在未被使用时,其值为 0。蚂蚁在完成一次巡游之后,需要对效用记录进行反馈,反馈的方法如式(7):

$$u_g \leftarrow \begin{cases} \frac{u_g + L_g}{2}, & u_g \neq 0 \\ L_g, & u_g = 0 \end{cases} \quad (7)$$

式中, L_g 为使用优先级规则 g 的蚂蚁单次巡游完毕后,基于活动列表生成的项目计划工期。若 g 第一次被选择,即原效用记录 $u_g=0$,那么新的效用记录则直接为 L_g 。

同时规则池需要定期根据优先级规则的效用记录更新优先级规则的被选择概率,更新方法按照式(8)进行:

$$\varphi_k \leftarrow \begin{cases} \frac{u_{\max} - u_k + \frac{1}{|R|Q}}{\sum_{j=1}^{|R|} (u_{\max} - u_j + \frac{1}{|R|Q})} \sum_{j=1}^h \varphi_j, & u_k \neq 0, u_j \neq 0 \\ \varphi_k, u_k = 0 \end{cases} \quad (8)$$

式中, $u_{\max}$ 为在规则池里所有规则中最高的效用记录值, $u_{\min}$ 为所有规则中最低的效用记录值, 因为目标函数是求极小值, 因此式中通过 $u_{\max} - u_k$ 做了一个转换, 使效用记录值较低的优先级规则获得更高的选择概率。同时, Q 是一个常数, 且 $Q > 1$ 。引入 $\frac{1}{|R|Q}$ 的作用是使较差的规则仍然能够以较小的概率被选择。若优先级规则从来未被选择过, 则其被选择概率保持不变。规则池若不进行定期更新, 我们称之为静态规则池。若规则池按照式(8)进行定期更新, 我们称之为动态规则池。动态规则池对规则被选择概率进行定期更新的时间间隔 Δt , 可通过算法所产生的计划实例个数来确定。

3.4 蚂蚁的预判能力

在 MMRCPSP 问题中, 由于存在紧前关系约束和资源约束, 所以, 蚂蚁在旅行过程中会遇到很多不可行路径。蚂蚁在旅行过程若能通过预判能力排除那些不可行路径, 将会改善算法的执行速度。本文参考分支界定算法的思想, 通过设定资源上界, 排除那些不可行的路径, 提升算法的求解效率。

蚂蚁在位置 m 选择新的路径时, 只能从可行活动集合 E_m 中选择, 否则便违反了紧前关系约束。活动 $j \in E_m$ 的充分必要条件是 $j \notin G_m$, 且 $P_j \subset G_m$, P_j 为 j 的紧前活动。另外, 令所有项目活动构成集合 G , 对于一个不完全的优先级列表 G_m , 令所有属于 G 但不属于 G_m 的活动集合为 $\overline{G_m}$ 。蚂蚁经过的所有节点构成一个不完全活动列表 G_m 。调度 G_m 中的活动可以生成一个不完全项目计划。

定义 1 最小可能资源消耗: 从 G_m 的可行活动集合 E_m 中选择一个活动 j 并加入 G_m , 生成新的不完全活动列表 G_{m+1} 。令所有 $\overline{G_{m+1}}$ 中的活动均选择不可更新资源 k 的资源消耗最小的活动执行模式, 根据 G_{m+1} 产生不完全计划。在该不完全计划的基础上, 放松可更新资源约束, 按照关键路径方法 (CPM) 调度 $\overline{G_{m+1}}$, 可以产生一个临时完全计划, 则该临时完全计划对不可更新资源 k 的消耗定义为资源 k 对于不完全活动列表 G_{m+1} 的最小可能资源消耗, 或 k 对 G_m 和 j 的最小可能资源消耗。

蚂蚁可以根据可行活动集合、最小可能资源消耗和最小可能项目工期进行两个层面的预判: (1) 基本预判: 将不属于可行活动集合的活动排除; (2) 高级预判: 蚂蚁在采用高级预判时, 采用了分支界定算法的定界和剪枝方法, 以每种不可更新资源的总供应量为资源上界, 依据定义 1 进行剪枝, 即在基本预判的基础上, 对于 $k \in K$, 若 k 对 G_m 和 j 的最小可能资源消耗超过了该资源的总供应量 N^k , 则将该活动 j 排除。另外, 有时蚂蚁行走到一个位置时, 通过高级预判机制可能会发现所有的路径均被排除, 此时算法要回溯, 即将当前的节点排除, 回到上一个节点, 重新进行路径选择。

3.5 状态转移评估机制

蚂蚁试图移动到新的位置之前, 首先通过上节所述的预判机制排除不可行路径, 然后再通过状态转移评估机制对余下的可行路径进行评估。主要需要基于两种信息素进行评估: 一是蚂蚁个体的行为习惯, 通过蚂蚁所采用的优先级规则确定; 二是对遗留在信息素分布矩阵上的信息素浓度进行评估, 即信息素评估。

蚂蚁状态转移评估机制有两种: 传统方法是直接评估方法, 直接对新的位置进行评估; 另外一种方法是累加评估方法^[10], 考虑蚂蚁所经过的路线和新的位置, 综合考虑这些位置的信息素与优先级规则,

权衡判断每个新位置的被选择概率。在本文所使用的解构造模型中,其目的是搜索一个能够产生最优调度计划的活动列表。在某个单个位置放置一个活动并无实际意义,有意义的是活动之间排序。因此本文采用累加评估方法,如式(9):

$$p_{ij} = \frac{\left(\sum_{k=1}^i \tau_{kj}\right)^\alpha \eta_j^\beta}{\sum_{h \in \epsilon} \left[\left(\sum_{k=1}^i \tau_{kh}\right)^\alpha \eta_h^\beta\right]} \quad (9)$$

式中, $\sum_{k=1}^i \tau_{kj}$ 为蚂蚁在行走至第 i 个步骤时,其所经过路径上的信息素总量。 ϵ 为可行路径(即可行活动集合)。 η_j 为采用蚂蚁所绑定的优先级规则计算的活动 j 的优先权系数。 α 和 β 均是算法的系统参数, α 为信息素影响因子, β 为启发式信息影响因子,分别表示信息素和优先级规则对蚂蚁路径选择的影响程度。

3.6 信息素更新机制

本文设计了一种总量确定、定期挥发的信息素更新机制。在该机制中,信息素的总量是确定的,在每次挥发完成后,维持留在信息素分布矩阵中的信息素量和已挥发信息素量的比例关系。信息素定期挥发,即每隔一段时间,信息素便挥发一次。信息素挥发的间隔时间通过产生的计划实例个数确定(概念如同前述的规则被选择概率更新的时间间隔)。

设信息素总量为 U ,留在信息素分布矩阵中的信息素总量为 U_r ,挥发掉的信息素量为 U_p ,则有 $U = U_r + U_p$ 。在每次挥发刚刚完成之后,算法保证 $U_r = (1 - \rho)U$ 和 $U_p = \rho U$ 。其中 ρ 为总体挥发因子,为算法的系统参数,它确定了 U_r 和 U_p 比例的动态平衡。信息素挥发时,信息素分布矩阵上每个位置的信息素按照相同的比例 ρ' 挥发,那么在挥发时,需满足式(10):

$$\sum_{i=1}^n \sum_{j=1}^n \sum_{m=1}^{M_j} (\tau_{ij}(m) \rho') = (1 - \rho)U \quad (10)$$

可见, ρ' 值在算法执行过程中并不是常量,它需要通过留在信息素分布矩阵上的信息素量和 ρ 来确定,根据式(11)计算:

$$\rho' = \frac{(1 - \rho)U}{\sum_{i=1}^n \sum_{j=1}^n \sum_{m=1}^{M_j} \tau_{ij}(m)} \quad (11)$$

故挥发时信息素值可按照式(12)计算:

$$\tau_{ij}(m) \leftarrow \rho' \tau_{ij}(m) = \frac{(1 - \rho)U}{\sum_{i=1}^n \sum_{j=1}^n \sum_{m=1}^{M_j} \tau_{ij}(m)} \tau_{ij}(m), \quad 1 \leq m \leq M_j \quad (12)$$

蚂蚁个体在完成单次巡游之后,立即更新信息素分布矩阵,不必等待其他蚂蚁是否巡游完毕,然后进行下一次巡游。信息素增加机制如式(13):

$$\tau_{ij}(m) \leftarrow \tau_{ij}(m) + \Delta \tau_{ij}, \quad 1 \leq m \leq M_j \quad (13)$$

式中, $\Delta \tau_{ij}$ 为信息素增量。 $\tau_{ij}(k)$ 说明在信息素增加时,只对活动 j 选中的模式 k 进行信息素增加。式(14)中的信息素增量 $\Delta \tau_{ij}$ 按照式(14)计算:

$$\Delta \tau_{ij} = \begin{cases} \frac{f_w - f^k}{f_w - f_e} \frac{\rho U}{\Delta t_p}, & f_w \neq f_e \\ \frac{(1 - \rho)U}{\Delta t_p}, & f_w = f_e \end{cases} \quad (14)$$

f_w 为目前最优解, f_r 为目前最差解, Δt_p 为信息素挥发的时间间隔, 例如 $\Delta t = 10$, 意味着每当 10 个蚂蚁巡游完毕, 就挥发一次。显然这种方法适应了本文并行计算的特点, 信息素增加和挥发不再通过算法迭代次数来确定, 只与蚂蚁的巡游次数有关。而在 MMRCPSP 问题中, 蚁群巡游次数总量对应目前生成计划实例的个数。

4 实验测试

4.1 实验环境和数据

作者应用 java 语言实现了本文所提出的蚁群算法和扩展串行调度方案, 测试机内存大小为 1G, CPU 速度为 3G, 操作系统为 Windows XP。实验数据从著名的项目调度问题库 (Project Scheduling Problem Library, PSPLIB) 中选取, 这些问题均是由 Kolish 等^[11]开发的项目实例生成器 ProGen 生成, 在项目优化调度研究中广泛地被用于评估 RCPSP 问题的求解算法。关于 PSPLIB 更多的细节可以参考文献[12]。在 PSPLIB 中, MMRCPSP 测试问题包括 J10、J12、J14、J16、J18、J20 和 J30 共 7 组, 每组包括 640 个问题实例, 每个问题实例分别包括 10、12、14、16、18、20 和 30 个非虚活动。目前 MMRCPSP 问题已知最优解的问题实例最多也只有 20 个非虚活动^[3]。因此这里只能采用 20 个非虚活动以下的问题实例来测试算法的性能。

在规则池中实现了 GRPW、MTS、LST、MINSLK、MINLFT、GRD、GRU 和 SRD^[13] 8 种优先级规则。本文在使用优先级规则进行计算时, 将每种 MMRCPSP 中的每个执行模式按照一个活动进行对待, 同时采用文献[5]的处理方法, 对不同的优先级规则的计算量纲进行了统一。

4.2 系统参数的设定

在本文的蚁群算法中, 主要的系统参数有信息素影响因子 α , 启发式信息影响因子 β , 蚂蚁数量 POP, 挥发率 ρ , 信息素挥发时间间隔 Δt_p , 规则库更新的时间间隔 Δt_r 。参数 α 、 β 的绝对数值并没有意义, 有意义的是二者的比值。令 γ 代表 β 和 α 的比值, 即 $\gamma = \beta / \alpha$ 。本文采用实验的方法测试算法中各参数对算法性能的影响并选取较好的参数值。通过测试发现 Δt_p 和 Δt_r 取值超过 20, 且 $\gamma = 2$, $\rho = 0.4$ 时, 算法有较好的性能。选取不同的蚂蚁数量, 对查找到最优解的所需的计划实例个数影响不大。但蚂蚁数量对算法的 CPU 时间影响非常大。如图 3 所示, 随着蚂蚁的增加, 每个问题所需的平均 CPU 时间显著降低。当蚂蚁数量超过 40 时, 每个问题的求解速度稳定在 450 毫秒左右, 速度不再增加。

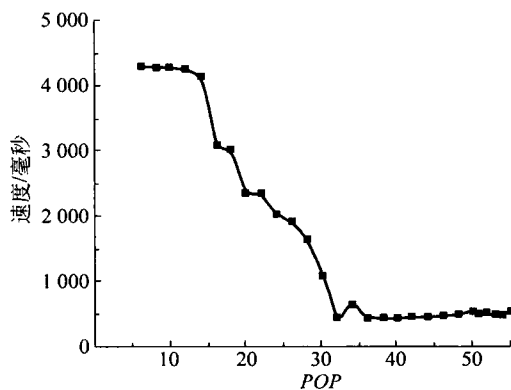


图 3 蚂蚁数量对算法 CPU 时间的影响

其原因是本文所设计的算法中,每个蚂蚁对应一个单独的线程,线程数量越多,越能够充分利用 CPU 的计算能力。但在算法的运行过程中,所有线程共享信息素分布矩阵和规则池,需要通过同步机制(synchronized)保证线程安全。这样,蚂蚁数量太多会造成在信息素增强、信息素挥发、更新规则池时的线程等待,在给定的硬件条件下,不会继续提升算法的速度。蚂蚁数量与算法运行的硬件环境相关,因此在不同的硬件环境下,需要通过实验的方法确定最佳的蚂蚁数量。

4.3 算法不同配置比较

为了测试扩展串行调度方案、规则池和蚂蚁预判能力对算法的影响,本节根据对算法采用不同的配置,分析其计算结果:算法 1 采用基本预判机制和静态规则池;算法 2 采用高级预判机制和静态规则池;算法 3 采用高级预判机制和动态规则池。对算法的三种配置方案进行测试,测试指标为平均计划个数、平均偏差、最大偏差、最优解比率 and 不可行实例比率。各性能指标的定义如下:

- 平均计划个数:算法在发现最优解时生成的平均计划实例数,用于衡量算法的效率;
- 平均偏差:算法在终止后产生的次优解与最优解的平均偏差,用于衡量算法的计算质量;
- 最大偏差:算法最差的计算结果与最优解的偏差,用于衡量算法的稳定性;
- 最优解比率:算法发现的最优解的个数的百分比,用于衡量算法的稳定性和计算质量;
- 不可行实例比率:算法在计算过程中产生的不可行计划实例占总实例的百分比,用于衡量算法的效率。

以 J10、J14 和 J20 三组问题作为测试实例,统计算法上述 5 个性能指标,终止条件为 5 000 个计划实例。在不可更新资源约束下,有些问题实例没有可行解,这里只对可行问题实例进行测试,测试结果如表 1 所示。

表 1 算法不同配置方案的计算性能比较(5 000 个计划实例)

问题规模	可行实例个数	性能指标	算法 1	算法 2	算法 3
J12	547	平均计划个数	658	389	366
		平均偏差	0.79%	0.25%	0.14%
		最大偏差	—	18.4%	11.6%
		最优解比率	57.2%	87.4%	98.4%
		不可行实例比率	48.2%	0%	0%
J16	550	平均计划个数	1 006	724	638
		平均偏差	0.98%	0.91%	0.83%
		最大偏差	—	27.2%	21.6%
		最优解比率	55.0%	76.4%	89.4%
		不可行实例比率	47.1%	0%	0%
J20	540	平均计划个数	1 698	988	770
		平均偏差	1.18%	1.09%	0.93%
		最大偏差	—	37.2%	29.6%
		最优解比率	46.0%	68.4%	80.4%
		不可行实例比率	49.5%	0%	0%

算法 1 使用的是基本预判,蚂蚁在行走完全程之后,采用扩展串行调度方案生成的计划可能无法满足不可更新资源约束,因此会产生大量的不可行项目计划。算法 2 由于采用了高级预判机制,全面消除了不可行解,蚂蚁不会行走不可行的路径,有效地避免了不可行计划的生成,同时也大幅避免了无谓的计算,提升了算法的计算质量。而算法 3 在算法 2 的基础上引入了规则池的动态更新,算法能

够自行发现适应于该问题实例的优先级规则,因此算法的速度和计算质量有了明显的提高。

4.4 与其他算法计算结果比较

目前的 MMRCPSP 问题研究在测试算法性能时,多以算法的 CPU 时间来衡量性能。不同算法的测试软硬件条件难以保持一致,因为算法之间的性能也难以比较。而以计划实例个数为终止条件则有效地避免了这个问题,但从作者能够查阅到的文献来看,这样处理的还很少,故作者仅以能够查到的少数几个文献[1,14,15]来比较算法的计算性能。这些文献均是以 6 000 个计划实例个数为算法终止条件,计算 J10 组问题。本文算法与上述算法的比较结果如表 2 所示。从计算结果可以看出,本文与成熟的智能优化算法如遗传算法相比,在计算性能上有明显的提高。

表 2 与既有的算法比较

作 者	算 法	平均偏差	最优解比率	可行解
Kolisch and Drexl ^[1]	局部搜索	0.50%	91.8%	100%
Ozdamar ^[14]	遗传算法	0.86%	88.1%	100%
Hartmann ^[15]	遗传算法	0.10%	98.01%	100%
本文算法	蚁群算法	0.08%	99.3%	100%

5 结语

本文提出了一种求解 MMRCPSP 问题的蚁群算法。该算法定义了一种适用于 MMRCPSP 的解构造模型,可以通过信息素分布矩阵形象地描述。信息素分布矩阵中的每个节点中的信息素可以与项目活动及活动的每个模式相映射。通过为每个蚂蚁设计单独的线程,突出了蚁群算法在并行计算上的优势。算法采用规则池管理大量优先级规则,提升了算法的适应能力。同时为蚂蚁赋予一定的预判能力,避免产生无谓的不可行项目计划,提升了算法的搜索速度。通过采用标准测试数据进行测试仿真发现,为蚂蚁设计单独的线程,能够大幅提高算法的执行速度。而通过采用预判机制、规则池在提升算法速度的同时,增强了算法的计算质量和通用性。

参 考 文 献

[1] Kolish R,Drexl A. Local search for nonpreemptive multi-mode resource-constrained project scheduling[J]. IIE Transactions,1997,29(11): 987-999.

[2] 毛宁,陈庆新,陈新. MRCPSP 的一种精确算法[J]. 控制理论与应用,2001,18(1): 119-126.

[3] Sprecher A, Drexl A. Multi-mode resource-constrained project scheduling by a simple, general and powerful sequencing algorithm[J]. European Journal of Operational Research,1998,107(2): 431-450.

[4] Joaquín B, Jordi P. Ant colonies for the RCPS problem[C]. Topics in Artificial Intelligence: 5th Catalanian Conference on AI,CCIA 2002,Castellon,Spain,2002: 257-268.

[5] Daniel M,Martin M, Hartmut S. Ant colony optimization for resource-constrained project scheduling[J]. IEEE Transaction on Evolutionary Computation,2002,6(4): 333-346.

[6] 杨波,万仲平,尹德玉. 资源约束项目排序问题的一种修正蚁群算法[J]. 工程数学学报,2007,24(3): 437-445.

[7] Lin-Yu Tseng,Shih-Chieh Chen. A hybrid metaheuristic for the resource-constrained project scheduling problem [J]. European Journal of Operational Research,2006,175(2): 707-721.

[8] Kolisch R. Serial and parallel resource-constrained project scheduling methods revisited: Theory and computation

- [J]. European Journal of Operational Research, 1996, 90(2): 320-333.
- [9] Vicente V, Francisco B, Sacramento Q. Justification and RCPSP: A technique that pays[J]. European Journal of Operational Research, 2005, 165(2): 375-386.
- [10] Merkle D, Middendorf M. An ant algorithm with a new pheromone evaluation rule for total tardiness problems [C]. Proceedings of the Evo Workshops 2000, Berlin, Germany: Springer-Verlag, 2000: 287-296.
- [11] Kolisch R, Sprecher A, Drexel A. Characterization and generation of a general class of resource-constrained project scheduling problems[J]. Management Science, 1995, 41(10): 1693-1703.
- [12] Kolisch R, Sprecher A. PSPLIB—A project scheduling problem library[J]. European Journal of the operational Research, 1997, 96(1): 205-216.
- [13] Hartmann S, Kolisch R. Experimental evaluation of state-of-the-art heuristics for the resource-constrained project scheduling problem[J]. European Journal of Operational Research, 2000, 127(2): 394-407.
- [14] Ozdamar L. A genetic algorithm approach to a general category project scheduling problem [J]. IEEE Transactions on Systems, Man, and Cybernetics. 1999, 29(1): 44-59.
- [15] Hartmann S. Project scheduling with multiple modes: A genetic algorithm[J]. Annals of Operations Research, 2001, 102(2): 111-135.

ACO for Solving MMRCPS

PENG Wuliang

(School of Economics and Management, Shenyang Ligong University, Shenyang 110168)

Abstract An ant colony optimization (ACO) algorithm for solving multi-mode resource-constrained project scheduling problem (MMRCPS), which is a well-known NP hard problem was presented. Compared with the existing studies, the algorithm uses a priority rule pool to manage a large number of priority rules. In addition, each ant is endowed with autonomous ability, learning ability and pre-judgment ability. The autonomous ability is that each ant is provided with a single thread, the learning ability is that each ant can dynamically select the excellent priority rules, and the pre-judgment ability is that each ant can avoid the non-feasible routes based on the branch and bound method. Finally, a full factorial computational experiment was set up using the well-known standard instances in PSPLIB, and computational results reveal that the algorithm is effective for MMRCPS.

Key words Multi-mode resource-constrained project scheduling problem, Planning and scheduling, Ant colony optimization