

申 报	系列：教师系列
	教学科研并重型
	专业：计算机软件与理论
	职称：副教授

业绩成果材料

（申报人的业绩成果材料包括论文、科研项目、获奖以及其他成果等）

单 位（二级单位） 数学与信息学院 软件学院

姓 名 彭超达

材料核对人：

单位盖章：

核对时间：

华南农业大学制

目 录

一、教学研究业绩

1. 教学研究项目：关于“课程思政”视角下的《数字图像处理》课程建设研究项目的立项通知（合同）及有关佐证材料4
2. 教材参编：十四五规划教材《C 语言程序设计教程》9

二、科研项目

1. 主持：关于国家自然科学基金-青年科学基金项目的批准通知书18
2. 主持：关于广州市科技计划项目的合同22
3. 主持：关于广州市科技计划项目支持方向为优秀博士“续航”项目的任务书 33
4. 主参：关于国家自然科学基金项目有关佐证材料45
5. 主参：关于“十四五”国家重点研发计划子课题有关佐证材料66
6. 主参：关于广东省基础与应用基础研究基金项目有关佐证材料70
7. 主参：关于广东省基础与应用基础研究基金自然科学基金项目有关佐证材料 81

三、论文、著作等

1. 检索证明（五篇） 90
2. 以第一作者发表本专业论文情况
 - 2.1. A two-phase framework of locating the reference point for decomposition-based constrained multi-objective evolutionary algorithms (Knowledge-Base

Systems,2022,中科院 TOP 一区)	95
2.2. Constrained Multi-Objective Optimization for UAV-Enabled Mobile Edge Computing Offloading Optimization and Path Planning (IEEE Wireless Communications Letters,2022,中科院二区)	114
2.3. A decomposition-based constrained multi-objective evolutionary algorithm with a local infeasibility utilization mechanism for UAV path planning (Applied Soft Computing,2022,中科院 TOP 二区)	123
2.4. Joint Energy and Completion Time Difference Minimization for UAV-Enabled Intelligent Transportation Systems: A Constrained Multi-Objective Optimization Approach (IEEE Transactions on Intelligent Transportation Systems,2024,中科院 TOP 一区)	139
3. 以通讯作者发表本专业论文情况	
3.1. Code Multi-View Hypergraph Representation Learning for Software Defect Prediction (IEEE Transactions on Reliability,2024,中科院二区)	161

四、科研成果

1. 知识产权

1.1. 专利授权证书：一种基于改进多目标进化算法的无人机群路径规划方法	178
1.2 专利实审通知：基于改进约束多目标优化进化算法的 AGA-UAV 路径规划方法	180

五、其他业绩

1. 指导学生学科竞赛

1.1. 蓝桥杯全国总决赛 C/C++程序设计大学 B 组二等奖....	181
--------------------------------------	-----

1.2. 蓝桥杯全国总决赛 C/C++程序设计大学 A 组三等奖....	182
1.3. 蓝桥杯全国总决赛 C/C++程序设计大学 B 组优秀奖....	183
1.4. 蓝桥杯广东省赛区 C/C++程序设计大学 A 组一等奖....	184
1.5. 蓝桥杯广东省赛区 C/C++程序设计大学 B 组一等奖....	185
1.6. 蓝桥杯广东省赛区 C/C++程序设计大学 B 组一等奖....	186
1.7. 蓝桥杯广东省赛区 C/C++程序设计大学 B 组三等奖....	187
1.8. 2022 年第二届中国高校大数据挑战赛三等奖.....	188
2. 指导学生创新创业训练项目	
2.1. 2022 年全国青年科普创新实验暨作品大赛华南农业大学 学校赛优秀奖.....	189
2.2. 2023 年大学生创新训练计划项目立项证明.....	190
3. 个人荣誉	
3.1. 2022 年度“十佳工作者”荣誉获得证明.....	191

华南农业大学文件

华南农教〔2021〕69 号

关于公布华南农业大学 2021 年度校级质量工程 和教改项目立项的通知

各学院、部处、各单位：

根据《广东省教育厅关于开展 2021 年度广东省本科高校教学质量与教学改革工程项目申报推荐工作的通知》（粤教高函〔2021〕18 号）和《广东省本科高校“十四五”教学质量与教学改革工程建设实施方案》（粤教高〔2021〕3 号）精神，以及学校《关于 2021 年度校级教育教学改革与研究项目申报的通知》《关于开展 2021 年度质量工程项目申报推荐工作的通知》等要求，为进一步推进学校教学改革，建设一流本科教育，提升人才培养质量，学校组织开展了 2021 年度校级质量工程和教改项目遴选工作。

- 1 -

经项目负责人申请、所在单位推荐、学校组织专家评审、校内公示无异议等程序，确定“食品化学（全英）在线开放课程”等 53 个项目为 2021 年度校级质量工程项目；确定“农业院校生物学拔尖人才培养模式改革与实践”等 147 个项目为 2021 年度校级教改项目，具体立项情况详见附件 1 和附件 2。

请各项目负责人按照项目建设任务及要求，及时开展各项工作，加快推进学校人才培养改革，并力争取得高水平的教学成果。各单位要切实加强对项目建设的督促、指导，以确保项目能如期高质量完成。

特此通知。

- 附件：1. 华南农业大学 2021 年校级质量工程立项名单
2. 华南农业大学 2021 年校级教改项目立项名单

华南农业大学
2021 年 11 月 30 日

（联系人：王 欢，电 话：85280052）

公开方式：主动公开

华南农业大学党政办公室

2021 年 11 月 30 日印发

- 2 -

附件 2

华南农业大学 2021 年校级教改项目立项名单

项目编号	项目名称	项目负责人	项目类别
JG21001	农业院校生物学拔尖人才培养模式改革与实践	邓诣群	重点项目
JG21002	基于思政的“互联网+赛教融合”《养牛学》课程改革与实践	李耀坤	重点项目
JG21003	基于 ESG 整合思政与专业教育的《财务报告分析》混合式教学改革与实践	陈艳艳	重点项目

JG21133	“课程思政”视角下的《数字图像处理》课程建设研究	崔金荣	自筹项目
JG21134	概率论与数理统计混合式一流课程建设与实践	朱艳科	自筹项目
JG21135	产教深度融合以项目驱动的实践教学研究与建设	蔡贤资	自筹项目
JG21136	《FPGA 技术及应用》课程教学方法创新与实践	陈 楚	自筹项目
JG21137	疫情下光电信息科学与工程专业课程教学实践与探索	刘建斌	自筹项目
JG21138	“金课”导向下农林院校《现代汉语》课程特色思政教学体系的探索与实践	李 颖	自筹项目

项目编号: JG21 _ _ _

华南农业大学教育教学研究和改革项目

申 报 书

项目类别 自 筹 项 目

项目名称 “课程思政”视角下的《数字图像处理》课程
建设研究

项目负责人 崔 金 荣

职 称 讲 师

手机号码 18588528183

所在单位 (公章)

申报日期 2021 年 8 月 15 日

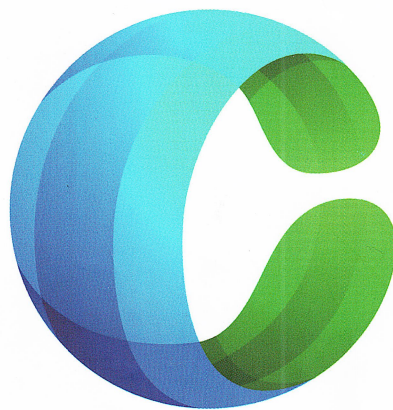
华南农业大学 本科生院 制

2021 年 6 月

	主要成员 (不含申请者)	姓名	性别	出生 年月	职称	工作单位	分工	签名 ²
		黄栋	男	1987 年 11 月	副教授	数与信学院	课题研究，并筹划 题目的协调实施	黄栋
		刘心	女	1977 年 6 月	副教授	数与信学院	指导形成课题方 案，对思政进 行挖掘	刘心
		彭超达	男	1990 年 9 月	讲师	数与信学院	对相关文献进 行研究，搜集 整理相关资料	彭超达



普通高等教育农业农村部“十四五”规划教材



C语言程序设计 教程

第二版

肖磊 陈湘骥 主编

首批全国优秀出版社



中国农业出版社

第二版编写人员

主 编 肖 磊 陈湘骥

参 编 梁 云 彭超达

宋鸿陟 王海燕

张义青

目 录

第二版前言

第一版前言

第 1 章 C 语言与程序设计概述	1
1.1 计算机与程序设计	1
1.1.1 指令与程序	1
1.1.2 程序设计语言	1
1.2 程序设计的基本过程	3
1.2.1 程序设计的步骤	3
1.2.2 算法的常用描述方法	5
1.3 C 语言的发展与特点	6
1.3.1 C 语言的发展历程	6
1.3.2 C 语言的特点	7
1.4 C 语言程序的组成和结构	8
1.4.1 简单的 C 程序例子	8
1.4.2 C 语言程序的结构	9
1.5 C 语言程序的开发	10
1.5.1 C 语言程序的开发步骤	10
1.5.2 使用 IDE 开发 C 语言程序	11
习题	15
第 2 章 基本数据类型和表达式	17
2.1 C 语言的基本语法单位	17
2.1.1 基本符号	17
2.1.2 关键字	17
2.1.3 标识符	18
2.2 数据与数据类型	18
2.3 基本数据类型	19
2.3.1 常量与变量	19
2.3.2 整型数据	21
2.3.3 实型数据	23
2.3.4 字符型数据	25

2.4 运算符和表达式	27
2.4.1 算术运算符和算术表达式	28
2.4.2 赋值运算符和赋值表达式	29
2.4.3 逗号运算符与逗号表达式	30
2.4.4 sizeof 运算符	30
2.4.5 不同类型数据混合运算与数据类型转换	31
2.4.6 位运算符	33
习题	35
第3章 C语言程序设计初步	37
3.1 C语言语句的作用与分类	37
3.2 数据的输入和输出	38
3.2.1 格式输出函数 printf	38
3.2.2 格式输入函数 scanf	41
3.2.3 字符输入/输出函数	43
3.3 常用的数学函数	45
3.4 顺序结构程序示例	46
3.5 程序设计中注意的问题	48
3.5.1 源程序代码风格和注释	48
3.5.2 程序设计错误	49
3.5.3 调试程序	50
习题	50
第4章 选择结构程序设计	52
4.1 关系运算符与关系表达式	52
4.2 逻辑运算符与逻辑表达式	53
4.3 选择结构控制语句	55
4.3.1 单分支 if 语句	55
4.3.2 双分支 if 语句	57
4.3.3 使用嵌套 if 语句实现多分支结构	60
4.3.4 switch 语句	63
4.4 条件运算符和条件表达式	66
4.5 选择结构程序综合举例	68
习题	73
第5章 循环结构程序设计	74
5.1 循环结构解决的问题	74
5.2 while 语句	74
5.3 do-while 语句	77

5.4	for 语句	78
5.5	循环语句的比较	80
5.6	循环嵌套	81
5.7	改变循环的执行流程	83
5.7.1	break 语句	83
5.7.2	continue 语句	84
5.8	goto 语句	85
5.9	循环结构程序综合举例	86
	习题	90
第 6 章	数组	91
6.1	一维数组	91
6.1.1	一维数组的定义	91
6.1.2	一维数组的引用	92
6.1.3	一维数组的初始化	93
6.1.4	一维数组程序举例	94
6.2	二维数组	97
6.2.1	二维数组的定义	97
6.2.2	二维数组的引用	98
6.2.3	二维数组的初始化	99
6.2.4	二维数组程序举例	100
6.3	字符数组与字符串	104
6.3.1	字符数组的定义	104
6.3.2	字符数组的初始化	105
6.3.3	字符数组的引用	105
6.3.4	使用字符数组存放字符串	106
6.3.5	字符串的输入和输出	106
6.3.6	字符串处理函数	107
6.3.7	字符数组程序举例	110
	习题	115
第 7 章	函数	116
7.1	函数的基本概念	116
7.2	函数的基本应用	117
7.2.1	函数的定义	117
7.2.2	函数的调用	118
7.2.3	函数参数的值传递	120
7.2.4	函数的原型声明	121
7.3	数组作为函数参数	123

7.3.1 一维数组作为函数参数	123
7.3.2 多维数组作为函数参数	126
7.4 函数的嵌套调用	128
7.5 函数的递归调用	129
7.6 局部变量与全局变量	131
7.7 变量的存储方式	135
7.7.1 局部变量的存储类别	135
7.7.2 全局变量的存储类别	137
7.8 内部函数与外部函数	138
7.9 编译预处理	139
7.9.1 宏定义	139
7.9.2 文件包含	141
7.9.3 条件编译	141
7.10 函数程序举例	142
习题	146
第8章 结构体与共用体	148
8.1 结构体	148
8.1.1 结构体类型的定义	148
8.1.2 结构体变量的定义	149
8.1.3 结构体变量的使用	150
8.1.4 结构体类型程序举例	152
8.2 结构体数组	153
8.2.1 结构体数组的定义	153
8.2.2 结构体数组的使用	153
8.2.3 结构体数组的初始化	154
8.2.4 结构体数组程序举例	155
8.3 结构体类型数据在函数之间的传递	156
8.3.1 结构体变量在函数之间传递数据	156
8.3.2 结构体数组在函数之间传递数据	157
8.4 共用体	158
8.4.1 共用体类型的定义	159
8.4.2 共用体变量的定义	159
8.4.3 共用体变量的使用	160
8.5 枚举类型	162
8.5.1 枚举类型的定义	162
8.5.2 枚举类型变量的定义与使用	162
8.6 用 typedef 定义类型名	164
习题	165
· 4 ·	

第 9 章 指针	166
9.1 地址和指针的概念	166
9.2 指针变量和指针运算	167
9.2.1 指针变量	167
9.2.2 指针运算	167
9.3 指针与数组	171
9.3.1 数组元素的指针	171
9.3.2 通过指针引用数组元素	171
9.3.3 字符指针与字符串	173
9.3.4 指针与多维数组	175
9.3.5 指针数组	178
9.4 指针与函数	179
9.4.1 指针作为函数的参数	179
9.4.2 指针作为函数的返回值	182
9.4.3 指向函数的指针	183
9.5 多级指针	185
9.6 动态内存空间分配	186
9.7 指针与链表	188
9.7.1 结构体指针	188
9.7.2 链表的概念	189
9.7.3 链表的基本操作	190
9.8 命令行参数	196
习题	197
第 10 章 文件	199
10.1 文件的基本知识	199
10.1.1 文件的概念	199
10.1.2 文件的数据存储	200
10.1.3 文件缓冲区	200
10.1.4 文件类型指针	201
10.2 文件的打开与关闭	201
10.2.1 打开文件	201
10.2.2 关闭文件	203
10.3 文件的顺序读写	203
10.3.1 字符读写函数	203
10.3.2 字符串读写函数	205
10.3.3 二进制读写函数	206
10.3.4 格式化读写函数	208

10.4 文件的随机读写	209
10.4.1 文件位置标记	209
10.4.2 文件位置标记的定位	209
习题	211
附录 A 常用字符与 ASCII 码对照表	212
附录 B 运算符及其特征	213
附录 C 常用 C 语言库函数	214
参考文献	218

图书在版编目 (CIP) 数据

C 语言程序设计教程 / 肖磊, 陈湘骥主编. -- 2 版.
北京: 中国农业出版社, 2024. 6. -- (普通高等教育
农业农村部“十四五”规划教材). -- ISBN 978-7-109-
-32098-7

I. TP312.8

中国国家版本馆 CIP 数据核字第 2024GD4624 号

C 语言程序设计教程

C YUYAN CHENGXU SHEJI JIAOCHENG

中国农业出版社出版

地址: 北京市朝阳区麦子店街 18 号楼

邮编: 100125

责任编辑: 李 晓

版式设计: 王 晨 责任校对: 周丽芳

印刷: 中农印务有限公司

版次: 2015 年 8 月第 1 版 2024 年 6 月第 2 版

印次: 2024 年 6 月第 2 版北京第 1 次印刷

发行: 新华书店北京发行所

开本: 787mm×1092mm 1/16

印张: 14.75

字数: 368 千字

定价: 36.50 元

版权所有·侵权必究

凡购买本社图书, 如有印装质量问题, 我社负责调换。

服务电话: 010-59195115 010-59194918

国家自然科学基金资助项目批准通知

(包干制项目)

彭超达 先生/女士:

根据《国家自然科学基金条例》、相关项目管理办法规定和专家评审意见,国家自然科学基金委员会(以下简称自然科学基金委)决定资助您申请的项目。项目批准号: 62202177, 项目名称: 面向农业无人机自适应路径规划的约束多目标进化算法研究, 资助经费: 30.00万元, 项目起止年月: 2023年01月至 2025年12月, 有关项目的评审意见及修改意见附后。

请您尽快登录科学基金网络信息系统(<https://isisn.nsf.gov.cn>), **认真阅读《国家自然科学基金资助项目计划书填报说明》并按要求填写《国家自然科学基金资助项目计划书》(以下简称计划书)**。对于有修改意见的项目,请您按修改意见及时调整计划书相关内容;如您对修改意见有异议,须在电子版计划书报送截止日期前向相关科学处提出。

请您将电子版计划书通过科学基金网络信息系统(<https://isisn.nsf.gov.cn>)提交,由依托单位审核后提交至自然科学基金委。自然科学基金委审核未通过者,将退回的电子版计划书修改后再行提交;审核通过者,打印纸质版计划书(一式两份,双面打印)并在项目负责人承诺栏签字,由依托单位在承诺栏加盖依托单位公章,且将申请书纸质签字盖章页订在其中一份计划书之后,一并报送至自然科学基金委项目材料接收工作组。纸质版计划书应当保证与审核通过的电子版计划书内容一致。**自然科学基金委将对申请书纸质签字盖章页进行审核,对存在问题的,允许依托单位进行一次修改或补齐。**

向自然科学基金委提交电子版计划书、报送纸质版计划书并补交申请书纸质签字盖章页截止时间节点如下:

1. **2022年10月8日16点:** 提交电子版计划书的截止时间;
2. **2022年10月14日16点:** 提交修改后电子版计划书的截止时间;
3. **2022年10月19日:** 报送纸质版计划书(一式两份,其中一份包含申请书纸质签字盖章页)的截止时间。
4. **2022年10月28日:** 报送修改后的申请书纸质签字盖章页的截止时间。

请按照以上规定及时提交电子版计划书，并报送纸质版计划书和申请书纸质签字盖章页，逾期不报计划书或申请书纸质签字盖章页且未说明理由的，视为自动放弃接受资助；未按要求修改或逾期提交申请书纸质签字盖章页者，将视情况给予暂缓拨付经费等处理。

附件：项目评审意见及修改意见表

国家自然科学基金委员会

2022年9月7日

附件：项目评审意见及修改意见表

项目批准号	62202177	项目负责人	彭超达	申请代码1	F0214
项目名称	面向农业无人机自适应路径规划的约束多目标进化算法研究				
资助类别	青年科学基金项目	亚类说明			
附注说明					
依托单位	华南农业大学				
直接费用	30.00 万元	起止年月	2023年01月 至 2025年12月		
<p>通讯评审意见：</p> <p><1>具体评价意见：</p> <p>一、该申请项目是否面向国家需求并试图解决技术瓶颈背后的基础问题？请结合应用需求详细阐述判断理由。</p> <p>农业无人机在数字农业发展中有着广泛的应用场景，而无人机路径规划是无人机应用的关键问题之一。本课题针对现有无人机规划在模型和算法上的若干瓶颈问题，研究维度自适应的无人机路径规划约束多目标优化方法，能够为农业无人机运用提供有效的技术支撑，面向国家数字农业需求开展基础技术研究，有很好的资助价值。</p> <p>二、请评述申请项目所提出的科学问题与预期成果的科学价值。</p> <p>针对无人机路径规划问题的瓶颈和挑战，申请书提出可变维度进化算子设计、基于历史信息的约束处理方法设计、约束多目标进化算法下代理模型训练三个关键问题；通过解决这些关键问题，构建高效自适应的约束多目标优化方法，有望在无人机路径规划能力上取得较大提升，具有较高的科学意义和实用价值。</p> <p>三、请评述申请人的创新潜力及研究方案的创新性和可行性。</p> <p>申请人长期优化算法和无人机路径相关研究，在IEEE Transactions on Cybernetics、Applied Soft Computing等领域知名期刊发表了多篇相关研究成果，具有很好的研究基础和创新潜力。申请书针对自适应维度的决策变量策略、约束多目标进化算法框架、数据驱动的约束多目标进化算法下代理模型三个主要方面，制定了较为细致的研究方案，技术路线合理可行。</p> <p>四、其他建议</p> <p>作为青年基金，预期成果中发表论文数量偏多；建议突出高水平学术成果及其转化价值。</p> <p><2>具体评价意见：</p> <p>一、该申请项目是否面向国家需求并试图解决技术瓶颈背后的基础问题？请结合应用需求详细阐述判断理由。</p> <p>该项目面向农业无人机的路径规划，研究约束多目标进化算法。该项目既面向国家数字农业发展的需求，有助于推广农业无人机的应用，同时具有复杂约束的多目标进化计算也是具有挑战性的研究课题。因此，本项目符合“需求牵引、突破瓶颈”特征。</p> <p>二、请评述申请项目所提出的科学问题与预期成果的科学价值。</p> <p>项目拟将进化计算路径规划和B-Spline曲线路径表达相结合，设计符合无人机路径特征的编码方式，并研究维度可变的约束多目标进化算法和数据驱动的效率提升。科学问题凝练合理，预期成果既有学术价值也有实际应用价值。</p> <p>三、请评述申请人的创新潜力及研究方案的创新性和可行性。</p> <p>申请人在相关方向有良好的前期成果积累。项目的研究方案合理，技术路线清晰，有一定的创新性，具体方案可行，有望取得有价值的研究成果。</p> <p>四、其他建议</p>					

<p><3>具体评价意见：</p> <p>一、该申请项目是否面向国家需求并试图解决技术瓶颈背后的基础问题？请结合应用需求详细阐述判断理由。</p> <p>本项目拟建立一个面向农业无人机自适应路径规划的高效、快速约束多目标进化算法通用框架。首先，建立的约束多目标进化算法使用了基于决策变量维度可变的策略，能够自动适应不同农业作业的路径规划场景，同时，目标函数、约束函数、环境条件或者场景的改变不需要重新进行模型调参。其次，新算法框架设计的优秀进化算子和约束处理方法能高效地求解农业无人机路径规划问题，提出的基于数据驱动算法求解效率提升策略可以有效地降低农业无人机路径规划问题昂贵的计算代价，最终使得整个算法框架对不同的农业作业场景有优秀的求解高效性与鲁棒性。最后，建立的约束多目标进化算法通用框架可直接服务于农业无人机路径规划，给出相应的优化方案，提高农业管理效率和生产效益。研究内容与研究方法有新意。</p> <p>二、请评述申请项目所提出的科学问题与预期成果的科学价值。</p> <p>该项目使用基于维度可变的决策变量策略，相对于普通进化算子，其设计更为复杂，需要考虑进行杂交变异操作的父代个体避免因维度不同而无法进行操作的问题。同时，由于不同维度的个体所包含的进化有效信息不同，极大地增加了进化算子搜索最优解区域的难度。因此，设计一种具有针对性的高效进化算子是一个复杂且具有挑战性的问题，等等。该项目所关注问题的科学价值较高，对相关前沿领域研究有较高参考价值。</p> <p>三、请评述申请人的创新潜力及研究方案的创新性和可行性。</p> <p>研究内容具体，研究目标明确，方案设计合理可行，申请人具有较好的前期研究工作的经验和基础。</p> <p>四、其他建议</p> <p>无</p>	<p>修改意见：</p> <p style="text-align: right;">信息科学部</p> <p style="text-align: right;">2022年9月7日</p>
--	--

项目编号：202201010576

基础与应用基础研究项目 合同书

项目名称：面向复杂环境无人机路径规划的约束多目标进化算法研究

承担单位：华南农业大学

项目负责人：彭超达

计划类别：基础研究计划

专题名称：基础与应用基础研究项目

支持方向：一般项目（博士青年科技人员类）

组织单位：华南农业大学

起止时间：2022年04月01日 至 2024年03月31日

主管处室：基础研究处

广州市科学技术局
(二〇二二年制)

填写说明

一、本合同书的项目编号由市科学技术局（以下简称市科技局）统一确定。

二、本合同书由申报书在后台自动转换生成，如有错漏之处需修正，请联系市科技局项目责任处室退回承担单位修正。

三、项目经费分为直接费用和间接费用。

基础与应用基础研究专题项目试点实施“包干制”，经费支出不设科目比例限制，由项目研究团队自主调剂使用，按照市科研项目经费“包干制”管理有关规定执行，同时应符合以下要求：

（1）经费支出应实际用于项目研究支出，使用范围限于设备费、材料费、测试化验加工费、燃料动力费、差旅/会议/国际合作与交流费、出版/文献/信息传播/知识产权事务费、劳务费、专家咨询费、依托单位管理费用、绩效支出以及其他合理支出。

（2）经费支出应按照市级财政科研项目资金开支范围和标准使用；

（3）间接费用是指项目承担单位在组织实施项目过程中发生的无法直接列支的相关费用，主要用于补偿项目承担单位为了项目研究提供的现有仪器设备及房屋，水、电、气、暖消耗，有关提高科研管理、服务能力等费用，以及绩效支出等。间接费用按照不超过项目直接费用扣除设备购置费后的一定比例核定，具体比例按《广州市财政局 广州市科学技术局 广州市审计局关于市级财政科研项目资金绩效提升和管理监督办法》规定确定。

（4）不得列支基建费；

（5）项目验收时应提交经费决算表

四、本合同书仅适用于广州市基础与应用基础研究项目

一、基本信息

项目负责人	姓名	彭超达	证件类型	身份证	证件号码	445221199009201918	性别	男
	出生年月	1990年09月20日	民族	汉族	国籍	中国	学历	博士研究生
	学位	博士	学位授予国家(或地区)	中国	职务	无	职称	讲师(高校)
	所学专业	控制科学与工程	手机号码	18819457743	办公电话	18819457743	电子邮箱	chaodapeng@scau.edu.cn

项目承担单位	单位名称	华南农业大学	统一社会信用代码或组织机构代码	124400004554165634
	注册时间	1952-01-01	单位类型	高等院校
	注册地址	广东省广州市天河区五山路483号		
	办公地址	广东省广州市天河区五山路483号		
	联系人	姓名	倪慧群	
		手机号码	15920301530	
		电子邮箱	kjcgxk@scau.edu.cn	
	开户银行	广东广州工行五山支行		
	开户户名	华南农业大学		
	银行帐号	3602002609000310520		
研究平台				

项目 基本 信息	项目名称	面向复杂环境无人机路径规划的约束多目标进化算法研究		
	所属学科	信息-自动化-人工智能与知识工程-进化算法及应用		
	申请金额	5.00万元	研究期限	2022年04月01日-2024年03月31日
项目 摘要	无人机路径规划作为无人机的一个研究重点，对我国经济发展具有重要的战略意义。建立无人机路径规划数学模型和设计求解算法是无人机路径规划的核心研究问题。现有无人机路径规划数学模型存在模型通用性低，随着环境复杂化程度增加，模型建立难度增加。同时求解算法存在着约束处理能力效果差、寻优能力不足等问题。针对上述情况，本项目拟进一步开展：1、建立基于模块化无人机路径规划约束多目标优化数学模型，降低模型建立的资源投入。2、提出一种融合邻域惩罚和短暂忽略约束策略的约束多目标进化算法，并设计一种高效的进化算子，减轻求解算法寻优难度大的负担。最后，将设计的算法用于求解面向复杂环境无人机实际路径规划问题。本项目的研究为无人机路径规划的实际应用提供强有力的理论支撑，具有较强的创新性和明确的社会需求。			

二、项目预期成果

论文及专著情况	国家统计局刊物以上刊物发表论文（篇）		3		科技报告（篇）		0	
	其中，被SCI/EI/ISTP收录论文数（篇）		3		培养人才（人）		1	
	专著（册）		0		引进人才（人）		0	
专利情况(项)	发明专利		实用新型专利		外观设计专利		国外专利	
	申请	授权	申请	授权	申请	授权	申请	授权
	0	0	0	0	0	0	0	0
其他	无							

三、项目经费预算

本项目总投入： ¥（ 5.00 ）万元，其中，市财政补助经费：¥（5.00）万元，自筹经费：¥（ 0 ）万元。

(单位：万元)

1. 经费下达计划			
资金来源	小计	市科技局经费	自筹资金
2023年	5.00	5.00	/
合计	5.00	5.00	/

注：本项目市科技局经费试点实施“包干制”，经费支出不设科目比例限制，由项目研究团队自主调剂使用，按照市科研项目经费“包干制”管理有关规定执行。

202201010576

四、合同条款

第一条 甲、乙、丙方根据《中华人民共和国科学技术进步法》《广东省自主创新促进条例》《广州市科技创新条例》及《中华人民共和国民法典》等国家有关法规和规定，经协商一致，特订立本合同，作为甲、乙、丙方在合同执行中共同遵守的依据。

第二条 甲方、乙、丙三方应当严格履行《广州市科技计划项目管理办法》《广州市级财政科研项目资金绩效提升和管理监督办法》《广州市科技创新发展专项资金管理办法》《广州市科技计划项目经费“包干制”改革试点工作方案》《广州市科技计划项目全过程管理简政放权改革工作方案》的规定要求。

第三条 甲方应：

1. 根据财政经费预算安排，及时进行拨付项目经费。
2. 赋予乙方和丙方广州市科技业务管理阳光政务平台（以下简称阳光政务平台）的使用权限，保障丙方进行项目全过程管理的使用需求。
3. 根据甲方需要，在不影响乙方工作的前提下，定期或不定期对乙方项目的实施情况和经费使用情况进行检查或抽查。
4. 审核丙方提交的年度工作报告，制定下一年度的资金切块方案。
5. 对丙方进行周期绩效考核和检查评估，重新评估丙方资格。

第四条 乙方应：

1. 作为项目具体组织实施的责任主体，为本单位提供的与本项目有关的全部材料真实、合法、有效性负责，同意甲方向行业协会等第三方机构直接调取乙方与本项目相关的数据、信息、材料，包括但不限于工商登记信息、审计报告等；
2. 按照《合同书》规定的内容组织实施项目，接受并配合甲方、丙方以及各级财政、审计部门，或上述部门委托的机构进行评估、稽查、审计、检查和绩效评价，并按要求提供项目任务与预算执行情况和有关财务资料；
3. 按照市财政科技经费管理“包干制”相关要求对项目经费单独设账，专款专用；
4. 保证自筹资金按时到位和其它配套条件的落实；
5. 在项目研究开发过程中优先考虑使用“广东省科技资源共享服务平台”的仪器设备，项目购置的设备仪器若符合入网条件应及时办理入网手续对社会共用共享，提高设备仪器的使用率。按照《中华人民共和国采购法》要求，对符合政府采购范围的设备仪器，执行政府采购；
6. 项目合同执行期内需进行变更的，按照《广州市科技计划项目管理办法》《广州市级财政科研项目资金绩效提升和管理监督办法》《广州市科技创新发展专项资金管理办法》《广州市科技计划项目全过程管理简政放权改革工作方案》相关程序办理；
7. 项目合同执行期满后3个月内向丙方提出验收申请，并出具在广州注册会计市协会备案的验收专项审计报告，提前完成合同规定任务的可提前申请验收；
8. 按照相关规定，在项目验收时提交科技报告，办理《验收证书》和科技成果登记手续；

9. 在项目实施期间和项目结题验收后3年内，配合甲方开展对财政资金年度绩效跟踪，按照甲方要求提供相关信息和数据，完成年度报告填报任务；

第五条 丙方应：

1. 明确项目管理依据的管理办法或管理规程，承担项目全过程管理职责；
2. 自主安排立项评审和结题验收工作，充分利用阳光政务平台，推进项目全过程管理的网络化电子化，主动配合推行合同电子签章；
3. 严格落实信息公开制度，公示遴选和结题验收结果，并及时处理异议；
4. 及时报送相关材料，按广州市科学技术局要求，每年按时提交拟立项项目清单，报送年度工作总结；
5. 按广州市科学技术局要求配合开展绩效评价和监督检查工作；
6. 主动追回终止项目未使用和不合规支出的市财政科技经费；
7. 按照本单位相关项目管理办法组织项目验收工作，并按相关规定做好存档工作；
8. 协助甲方对项目的实施过程进行跟踪、检查和提供相关信息，并对所提供信息的客观真实性负责；
9. 负责监管乙方严格遵守本合同规定的任务；

第六条 甲方同意给予乙方人民币（5.00万）的资助，立项后一次性拨付。

第七条 合同终止：

1. 项目因故无法继续进行的，按照相关规定实施合同终止。
2. 发现存在以下情况之一的，立即启动终止程序：
 - ①因不可抗拒因素导致项目无法继续进行、没有必要继续进行或无法完成合同预期目标任务的；
 - ②不接受项目监督检查、检查不合格限期整改后仍未通过的或拒不配合项目验收工作的；
 - ③无正当理由项目合同执行期满后3个月以后仍未提交验收申请的；
 - ④项目承担单位已迁出本市，或已停止经营活动，或已注销的；
 - ⑤发现在项目申报、实施过程中有违法、欺骗等事实的；
 - ⑥存在其他导致项目不能正常实施的原因。
3. 合同终止由乙方提出申请，丙方审定，也可由丙方强制实施。具体由丙方按照《广州市科技创新发展专项项目全过程管理简政放权改革试点工作方案》的规定要求进行办理。
4. 合同终止后，乙方或承接乙方法定义务的责任人应停止使用该项目财政经费；上缴尚未使用和使用不符合规定的财政经费。

第八条 对合同正常执行期及项目整改期之外的经费开支，不属于财政项目经费列支范围。

第九条 在履行本合同的过程中，乙方发现可能导致项目失败或部分失败的情形时，应及时通知甲方，并采取适当措施减少损失，没有及时通知并采取适当措施，致使损失扩大的，应当就扩大的损失承担责任。

第十条 在履行本合同的过程中，如遇到市财政计划改变等不可抗力情况，甲方对所核拨经费的数量和时间可进行相应变更。

第十一条 成果转化：

1. 本项目技术成果及知识产权的归属、转让和实施技术成果所产生的经济利益的分享，除另有约定外，按国家和省、市有关规定执行；正式发表的论文、论著应标注“广州市科技计划项目资助”字样及项目编号；项目所取得的技术成果和知识产权应优先广州产业化或推广转让。

第十二条 属技术保密的项目，经协商订立如下技术保密条款：

1. 本合同书保密内容范围为：本合同及其补充协议和附件、乙方因履行本合同所接触或知晓的甲方工作秘密（包括但不限于甲方的任何技术性资料、以及甲方为完成本合同提供的任何其他信息资料并且在提供时未说明是公开信息的）、在合同履行过程中，乙方接触到的，或履行合同产生的任何国家、商业、工作信息（包括但不限于计算机系统数据信息、审计工作资料、技术文档及相关敏感资料等）。

2. 本合同书保密期限为：\

3. 乙方及乙方人员（包括但不限于项目组人员、乙方雇员、代理人、顾问等工作人员，下同）采取有效的保密措施以避免泄露给任何第三方；乙方增强对项目组人员的保密教育，每年至少开展一次保密自查，并与可能知悉保密内容的人员签订技术保密保护协议，确保项目组人员遵守保密协议，乙方应保密义务不得低于本合同书的约定；甲乙双方应建立技术保密制度。

4. 乙方在合同履行的过程中，对接触到的相关信息，乙方及项目组人员承担保密责任；乙方应将项目组人员的身份证复印件、劳动合同、学历职称证明、项目经验等资料提供给甲方，更换项目负责人时需事先征得甲方书面同意并提交上述资料。

5. 在本合同有效存续期间及合同终止后，未经甲方事先的书面同意，不得以任何方式记录、复制、拍摄、摘抄、收藏、公布、发表、公开、披露、散播本合同项下保密信息的任何部分，或对其加以任何形式的利用或使用；乙方及乙方人员未经甲方书面同意不得私自下载、拷贝计算机内项目相关数据信息，不得擅自携带记载项目内容的载体（例如移动硬盘、U盘等）和打印资料外出，严禁将工作系统的程序、口令等泄露给他人。

6. 属技术保密的项目必须经相关负责技术保密部门审查、批准后，方可发表或用于境外合作与交流。

7. 乙方应当制定泄密应急预案，一旦发现本单位持有的国家科学技术秘密可能泄露或者已经泄露，应当在24小时内向甲方报告，同时启动应急预案，并协助有关部门查处泄密事件。

8. 乙方应严格遵守国家、省市规定的其他技术保密相关法律、法规和政策。在项目实施过程中，乙方或项目合作单位及其相关人员违反科学技术保密管理相关规定，给国家安全和利益造成损害的，应当依法追究单位和相关人员的法律责任。

第十三条 廉洁责任

甲方、丙方、评审机构及其工作人员不得索取、收受利益相关方财物或其他不正当利益，严格遵守中央八项规定精神及其实施细则。

乙方应严格遵守国家、省、市关于科技专项经费使用的有关法律、法规，相关政策以及廉洁建设的各项规定，积极开展人员廉洁从业教育，防范科技项目组成员在科研活动中出现“法律、行政法规、部门规章或规范性文件规定的其他相关违规行为”。

第十四条 科研诚信和科技伦理要求

乙方应建立健全促进科研诚信和科技伦理的规章制度，落实以下职责：

1. 建立健全本单位学术论文发表诚信承诺制度、科研过程可追溯制度、科研成果检查和报告制度等成果管理制度。对本项目形成的科研成果的署名、研究数据真实性、实验可重复性等进行诚信审核和学术把关。防范科技项目组成员在项目申报、研发过程中出现提供虚假信息或材料，抄袭、剽窃他人科研成果，捏造、变造或篡改科研数据等违反科研诚信和科技伦理要求的情形。

2. 加强对科技项目参加人员的科研诚信和科技伦理教育，督促科技项目组成员恪守科学道德准则，遵守科研活动规范。对在科研诚信和科技伦理方面存在问题情节较严重的，应及时调整出项目团队并及时以书面形式报告甲方；

3. 加强对项目合作单位的科研诚信管理，正确履行管理、指导、监督职责，全面落实科研诚信和科技伦理要求；

4. 乙方或项目合作单位及其相关人员被记入科研严重失信行为数据库或相关社会领域信用“黑名单”，乙方应及时以书面形式报告甲方；

5. 乙方应严格遵守国家、省市规定的其他科研诚信管理和科技伦理相关法律、法规和政策。

6. 其他：在项目实施过程中，对乙方或项目合作单位及其相关人员有严重违背科研诚信和科技伦理要求的行为，甲方和相关部门可依照相关法律、法规规定对乙方采取责令改正、终止或撤销项目并追回财政性资金、记入科研诚信严重失信行为数据库等处理处罚措施。

第十五条 争议解决

因本合同书所产生的争议，各方应友好协商解决；协商不成的，各方同意由本合同签订地人民法院管辖。

第十六条 书面通知与送达

甲方在本合同履行过程中向乙方或丙方发出或者提供的所有书面通知、文件、文书、资料等，均以本合同所列明的乙方或丙方地址送达。乙方或丙方如果迁址，应当书面通知甲方；未履行书面通知义务的，甲方按原地址邮寄相关材料即视为已履行送达义务。

第十七条 鼓励开展科普工作

鼓励项目承担单位和人员结合科研任务对适合进行科学普及的项目内容加强科普工作。

本合同一式四份，各份具有同等效力。甲方和丙方各存一份，乙方存二份。本合同签订各方均负有相应的法律责任，不受机构、人事变动而影响。

说明：本《合同书》中，凡是三方约定无需填写的条款，在该条款的空白处划（\）。

附件：项目承担单位（乙方）及项目负责人承诺书

承诺书

本单位/本人作为广州市科技计划项目承担单位/项目负责人，将严格遵守广州市科技计划管理相关规定，严格履行自身责任，加强对项目组人员及合作单位的管理，在此郑重承诺：

（一）确保与本项目有关的全部材料真实、合法、有效，未侵犯其他方知识产权等权利；

（二）严格遵守《广州市科技计划项目管理办法》《广州市级财政科研项目资金绩效提升和管理监督办法》《广州市科技创新发展专项资金管理办法》《广州市科技计划项目全过程管理简政放权改革工作方案》等相关规定，实施项目和经费管理。

（三）严格遵守国家、省、市关于科研诚信和科技伦理的有关法律、法规，相关政策以及各项规定，加强项目实施过程中的科研诚信及科技伦理管理，恪守科研道德准则。

（四）_____

如有违反，本单位/本人愿意接受相关部门做出的各项处理决定，包括但不限于终止项目，停拨或核减经费，追回项目经费，取消一定期限广州市科技计划项目申报资格，记入科研诚信严重失信行为数据库，将不良行为向社会公开以及主要责任人接受相应党纪政纪处理等。

项目承担单位签章：

日期：2022.6.7



项目负责人签章：

日期：2022.6.7

刘绍达

合同书各方签章

广州市科学技术局（甲方）：广州市科学技术局

项目经办人：李磊

联系电话：020-83124052

责任处室负责人：莫雪华



项目承担单位（乙方）：华南农业大学

二级部门：华南农业大学数学与信息学院

项目负责人：彭超达

项目经费汇入账号

帐户名：华南农业大学

帐号：3602002609000310520

开户银行：广东广州工行五山支行

财务负责人：肖斐

财务负责人联系电话：020-5288032



组织单位（丙方）：华南农业大学

项目经办人：倪慧群



任务书编号：2025A04J5508

广州市科技计划项目 任务书

项目名称：	面向时间敏感无人机路径规划的动态约束多目标进化算法研究
承担单位：	华南农业大学
项目负责人：	彭超达
计划类别：	基础研究计划
专题名称：	2025年度基础与应用基础研究专题
支持方向：	优秀博士“续航”项目
组织单位：	华南农业大学
起止时间：	2025-01-01 至 2026-12-31
主管处室：	引进智力管理处（科技人才处）

广州市科学技术局制

二〇二五年

填写说明

1. 任务书甲方为广州市科学技术局；乙方为项目承担单位；丙方为项目组织单位。

2. 任务书基于项目申报书转换而成，请按照“广州科技GI（广州科技大脑）”提示在线填写核实，若存在不填写内容的栏目，请用“无”表示；任务书中的单位名称应为规范全称，并与单位公章一致。

3. 乙方与合作单位的合作协议自动从项目申报书中读取，如需变化调整，须待任务书签订后，按要求及时办理重大变更。

4. 乙方完成项目任务书在线填写，依次提交丙方和甲方审核确认后，按要求完成签章。具备电子签章条件的单位，在“广州科技GI（广州科技大脑）”完成任务书签署；不具备电子签章条件的单位，经与业务主管处室沟通对接后，可下载电子版项目任务书用A4纸双面打印装订签章，一式六份报甲方和丙方签章，其中甲方两份丙方两份，项目承担单位和项目负责人各一份。

5. 涉密项目请在“广州科技GI（广州科技大脑）”下载项目任务书模板，按保密要求离线填写报送。

6. 项目申报书是项目任务书填报的重要依据，未经甲方许可，乙方不得修改考核指标，调整主要研究内容。项目任务书将作为项目实施管理、验收结题和监督评估的重要依据。

7. 项目任务书中的“备注”，包括重要的必须补充的内容。

8. “广州科技GI（广州科技大脑）”是项目管理过程中重要通知和文书的电子送达平台，电子送达与书面送达具有同等法律效力。为确保电子送达渠道畅通，乙方和项目负责人应及时更新维护“广州科技GI（广州科技大脑）”的单位和个人信息。

9. 项目涉及科技伦理、科技安全（如临床研究、生物安全、信息安全等）和科技保密相关问题的，申报单位须严格执行国家有关法律法规和伦理原则，完成相关审查工作；项目负责人在项目任务书签订环节，须提供符合国家有关法律法规和伦理准则要求的审查

批准文件，项目承担单位负责审核批准文件的真实性和有效性。

广州市科技项目任务书2024-12-31

一、项目基本信息

项目 基本 信息	项目名称	面向时间敏感无人机路径规划的动态约束多目标进化算法研究
	申请市财政科技经费	10(万元)
	研究期限	2(年)
项目 摘要	实时路径规划是无人机技术的关键研究领域，提高路径规划的最优性和效率是重点研究问题。与现有深度、强化学习等主要方法不同，本项目创新地将该问题建模为动态约束多目标优化问题,并设计一种多分布评价策略约束多目标进化算法,以提高实时路径规划的最优性。为提高求解效率，设计一种轻量化代理模型降低算法计算复杂度。本研究为动态约束多目标优化和实时无人机路径规划提供新的理论和解决策略,具有较高的科学研究和实际价值。	

二、项目单位情况

项目承担单位	单位名称	华南农业大学	统一社会信用代码	124400004554165634
	注册时间	1952-01-01	单位类型	高等院校
	注册地址	广东省广州市天河区五山路483号		
	办公地址 (联系地址)	广东省广州市天河区五山路483号		
	联系人	姓名	夏杰	
		手机号码	13711345768	
		电子邮箱	kjcgxk@scau.edu.cn	
	开户银行	广东广州工行五山支行		
	开户户名	华南农业大学		
银行账号	3602002609000310520			

注：如果办公地址（联系地址）等相关信息有变更，项目单位应当在变更之日起三个工作日内告知我单位变更后的地址。

三、项目负责人信息

姓名	彭超达	证件类型	身份证
证件号码	445221199009201918	性别	男
出生日期	1990-09-20	民族	汉族
国籍	中国	学历	博士研究生
学位	博士	学位授予国家 (或地区)	中国
职务	无	职称	中级
所学专业	控制科学与工程	手机号码	18819457743
办公电话	020-85280001	电子邮箱	chaodapeng@scau. edu. cn
获2021或2022年度 广州市基础研究计 划基础与应用基础 研究(博士青年科 技人员)项目编号	202201010576		

四、项目经费信息

本项目总投入：¥（10）万元，其中，市财政科技经费：¥（10）万元，自筹经费：¥（0）万元。

经费下达计划			
资金来源	小计	市财政科技经费	自筹经费
2025	10	10	0
总计	10	10	0

（单位：万元）

注：本专题纳入“包干制”，市财政科技经费按市科技计划项目经费“包干制”相关规定执行。

五、预期代表性成果

项目负责人在项目实施期内，以该项目作为资助项目获得以下5种情形之一且经费使用符合规定的，由组织单位审核后通过验收。

（一）项目实施期内，以第一作者/通讯作者发表论文1篇或以上（须标注资助项目编号）；

（二）项目实施期内，以第一完成人申请或授权专利、软件著作权1项或以上；

（三）项目实施期内，获省级以上科技计划项目或人才项目支持1项或以上；

（四）项目实施期内，获省级以上科技奖励（含列入获奖团队成员名单）1项或以上；

（五）项目实施期内，获得职称晋升。

六、备注

专题补充约定条款：

甲方对未履行勤勉尽责义务的相关责任主体，自作出处理结论之日起，依照法律法规规定或任务书约定实施惩戒5年，取消相关责任主体申报市科技计划项目、申领市科技计划项目经费的资格。

预期代表性成果需在实施期内获得。

项目承担单位（乙方）及项目负责人承诺书

承诺书

本单位/本人作为广州市科技计划项目承担单位/项目负责人，将严格遵守广州市科技计划管理相关规定，严格履行自身责任，加强对项目组人员及合作单位的管理，在此郑重承诺：

（一）确保与本项目有关的全部材料真实、合法、有效，未侵犯其他方知识产权等权利，不存在多头申报、重复申报行为；

（二）严格遵守《广州市科技创新条例》《广州市科技计划项目管理办法》《广州市科技计划项目经费管理办法》《广州市科技计划科技报告管理办法》等相关规定，实施项目和经费管理；

（三）严格遵守国家、省、市关于科研诚信、科技伦理、科技安全（如临床研究、生物安全、信息安全等）和科技保密的有关法律、法规，相关政策以及各项规定，加强项目实施过程中的科研诚信、科技伦理、科技安全（如临床研究、生物安全、信息安全等）和科技保密管理，恪守科研道德准则。

如有违反，本单位/本人愿意接受相关部门做出的各项处理决定，包括但不限于终止项目、停拨经费、核减经费、追回经费，取消一定期限广州市科技计划项目申报资格，记入科研失信行为数据库，将不良行为向社会公开等。

项目承担单位：华南农业大学

日期：2024年12月26日

项目负责人：彭超达

日期：2024年12月24日

电子送达确认书

告知事项	<p>1. 为便于本项目承担单位（受送达人）及时收到相关通知和文书，请受送达人知悉项目管理过程中重要通知和文书（如：项目验收结果通知书、配合结余资金审计通知书、项目终止通知书等）的送达方式是通过“广州科技GI（广州科技大脑）”平台（以下简称平台）电子送达。</p> <p>2. 确认的送达方式适用于行政执法全过程程序。</p> <p>3. 请受送达人在项目管理过程中及时、主动查看平台相应模块，以免错过相关通知和文书。</p> <p>4. 此电子送达方式，以发送方设备显示发送成功视为送达。但接收方证明其到达平台的日期与发送方对应系统显示发送日期不一致的，以受送达人证明到达平台的日期为准。</p>
电子送达	<p>受送达人同意：“广州科技GI（广州科技大脑）”是项目管理过程中重要通知和文书的电子送达平台，电子送达与书面送达具有同等法律效力。为确保电子送达渠道畅通，受送达人同意“广州科技GI（广州科技大脑）”作为电子送达平台。</p> <p>“广州科技GI（广州科技大脑）”网站地址：https://gzsti.gzsi.gov.cn。</p>
受送达人确认	<p>我单位已阅读本确认书的告知事项，接受并确认了上栏送达方式。</p> <p style="text-align: right;">受送达人：华南农业大学</p> <p style="text-align: right;"><u>2024年12月26日</u></p>
备注	

任务书签署

甲乙丙三方根据《广州市科技计划项目管理办法》《广州市科技计划项目经费管理办法》《广州市科技计划科技报告管理办法》等有关文件规定，以及有关法律、政策和管理要求，签署本任务书。

签订地点：广州市越秀区

广州市科学技术局（甲方）：广州市科学技术局
局项目经办人：陈良 联系电话：83124036
责任处室负责人：洪雪妍

2024年12月31日

项目承担单位（乙方）：华南农业大学
二级部门：华南农业大学数学与信息学院
项目负责人：彭超达
项目经费汇入账号
账户名：华南农业大学 账号：3602002609000310520
开户银行：广东广州工行五山支行

2024年12月26日

组织单位（丙方）：华南农业大学
项目经办人：夏杰

2024年12月27日



项目批准号	U23A20174
申请代码	C1302
归口管理部门	
依托单位代码	51064208A0499-0932



U23A201741006896

国家自然科学基金 资助项目计划书 (预算制项目)

资助类别：联合基金项目

亚类说明：重点支持项目

附注说明：区域创新发展联合基金

项目名称：稻田杂草信息多尺度感知及机-药协同精准除草机理研究

直接费用：262万元 执行年限：2024.01-2027.12

负责人：齐龙

通讯地址：广州市天河区五山路483号华南农业大学工程学院

邮政编码：510642 电 话：020-85283707

电子邮件：qilong@scau.edu.cn

依托单位：华南农业大学

联系人：唐家林 电 话：020-85280070

填表日期：2023年11月27日

国家自然科学基金委员会制

Version: 1.006.896



国家自然科学基金资助项目计划书填报说明 （预算制项目）

- 一、项目负责人收到《国家自然科学基金资助项目批准通知》（以下简称《批准通知》）后，请认真阅读本填报说明，参照国家自然科学基金相关项目管理办​​法和新修订的《国家自然科学基金资助项目资金管理办法》（以下简称《资金管理办法》，请查阅国家自然科学基金委员会官方网站首页“政策法规”栏目），按《批准通知》的要求认真填写和提交《国家自然科学基金资助项目计划书》（以下简称《计划书》）。
- 二、填写《计划书》时要科学严谨、实事求是、表述清晰、准确。《计划书》经国家自然科学基金委员会相关项目管理部门审核批准后，将作为项目研究计划执行、检查和验收的依据。
- 三、《计划书》各部分填写要求如下：
 - （一）简表：由系统自动生成。
 - （二）摘要及关键词：各类获资助项目都应当填写中、英文摘要及关键词。
 - （三）项目组主要成员：计划书中列出姓名的项目组主要成员由系统自动生成，与申请书原成员保持一致，不可随意调整。如果《批准通知》所附“项目评审意见及修改意见表”中“修改意见”栏目有调整项目组成员相关要求的，待项目开始执行后，按照项目成员变更程序另行办理。
 - （四）资金预算表：根据批准的项目资助额度，按规定调整项目预算，并按照《国家自然科学基金项目计划书预算表编制说明》填报资金预算表和预算说明书。
 - （五）正文：
 1. 面上项目、地区科学基金项目：如果《批准通知》所附“项目评审意见及修改意见表”中“修改意见”栏目没有修改要求的，只需选择“研究内容和研究目标按照申请书执行”即可；如果《批准通知》中上述栏目明确要求调整研究期限或研究内容等的，须选择“根据研究方案修改意见更改”并填报相关修改内容。
 2. 重点项目、重点国际（地区）合作研究项目、重大项目、国家重大科研仪器研制项目、原创探索计划项目：须选择“根据研究方案修改意见更改”，根据《批准通知》的要求填写研究（研制）内容，不得自行降低、更改研究目标（或仪器研制的技术性能与主要技术指标、验收技术指标等）或缩减研究（研制）内容。此外，还要突出以下几点：
 - （1）研究的难点和在实施过程中可能遇到的问题（或仪器研制风险），拟采用的研究（研制）方案和技术路线；
 - （2）项目主要参与者分工，合作研究单位（如有）之间的关系与分工，重大项目还需说明课题之间的关联；
 - （3）详细的年度研究（研制）计划。
 3. 创新研究群体项目：须选择“根据研究方案修改意见更改”，按下列提纲撰写：
 - （1）研究方向；



- (2) 结合国内外研究现状，说明研究工作的学术思想和科学意义（限两个页面）；
 - (3) 研究内容、研究方案及预期目标（限两个页面）；
 - (4) 年度研究计划；
 - (5) 研究队伍的组成情况。
4. 基础科学中心项目：须选择“根据研究方案修改意见更改”，根据《批准通知》的要求和现场考察专家组的意见和建议，进一步完善并细化研究计划，按下列提纲撰写：
 - (1) 五年拟开展的研究工作（包括主要研究方向、关键科学问题与研究内容）；
 - (2) 研究方案（包括骨干成员之间的分工及合作方式、学科交叉融合研究计划等）；
 - (3) 年度研究计划；
 - (4) 五年预期目标和可能取得的重大突破等；
 - (5) 研究队伍的组成情况。
5. 对于其他类型项目，参照面上项目的方式进行选择和填写。



简表

项目负责人信息	姓 名	齐 龙	性 别	男	出生年月	1979年11月	民 族	汉族
	学 位	博士			职称	研究员		
	是否在站博士后	否			电子邮件	qilong@scau.edu.cn		
	电 话	020-85283707			个人网页	http://yjsglxt.scau.edu.cn/open/wxtkss/teacherinfo.aspx?jsbh=30003448		
	工 作 单 位	华南农业大学						
	所 在 院 系 所	工程学院						
依托单位信息	名 称	华南农业大学					代码	51064208A0499
	联 系 人	唐家林			电子邮件	kyc.jhk@scau.edu.cn		
	电 话	020-85280070			网站地址	http://kjc.scau.edu.cn/		
合作单位信息	单 位 名 称							
	东北农业大学 湖南农业大学							
项目基本信息	项 目 名 称	稻田杂草信息多尺度感知及机-药协同精准除草机理研究						
	资 助 类 别	联合基金项目				亚 类 说 明	重点支持项目	
	附 注 说 明	区域创新发展联合基金						
	申 请 代 码	C1302:农艺农机学				C1301:农业信息学		
	基 地 类 别	农业装备技术全国重点实验室						
	执 行 年 限	2024.01-2027.12						
	直 接 费 用	262万元						



项目摘要

中文摘要:

针对稻田杂草绿色防控的技术瓶颈，本项目创新提出“空地多尺度”感知、“机-药协同”控草的高效智能精准除草理念，具体研究内容和方法如下：1) 引入伪装目标检测思想，解决无人机视角下稻田杂草弱目标识别的难题，准确获取杂草密度和发生区域信息；2) 构建基于注意力机制的杂草视觉表征提取网络，提出同类杂草表征相互补充、不同种类杂草表征相互区分的深度聚类辨识算法，满足地面智能除草机具对杂草种类精准辨识的要求；3) 通过机械方式改变杂草叶片的表面结构和疏水特性，阐明除草剂在受损杂草叶面的润湿性能、传导效率及增效机制；4) 形成适配于不同繁殖特性杂草的控草处方策略，创新机-药协同作业的执行机构和视觉伺服控制系统，实现分类精准对靶作业。本项目实施所构建的模型和方法不仅可以为稻田杂草绿色防控提供技术支撑，减少除草剂的施用量，提高杂草防控效果，也可以为其他农田杂草精准防控技术的研究提供参考和借鉴。

Abstract:

To break the technical bottleneck of green weed control in rice fields, this project innovatively proposes a novel solution for efficient, intelligent, and precise weed control, which includes "ground-air multiscale" weeds perception and "mechanical-chemical integration" weed control strategy. The methods are described as follows. 1) By introducing the idea of camouflage object detection, the problem of identifying weeds in paddy field under the perspective of unmanned aerial vehicles can be effectively resolved, yielding accurate results on the perception of weed density and location. 2) In order for the intelligent ground weeding equipment to identify the species of weeds precisely, a deep neural framework, which complements the same representations and distinguishes different representations, is established based on the attention mechanism for feature extraction. 3) By mechanically modifying the surface structure and hydrophobic properties of weed leaves, the wetting performance, conduction efficiency, and performance of herbicides on the damaged leaf surface are elucidated; 4) The weed control strategies adapted to different reproductive characteristics of weeds are proposed accordingly, and then the execution mechanism and visual servo control system for precise weeding operations are developed based on the "mechanical-chemical integration" strategy. The models and methods proposed by this project can not only provide technical support for green weeding in paddy field with improved weeding effectiveness and reduced herbicide use, but also provide guidance and reference for the research on the precise weed control of other crop fields.

关键词(用分号分开): 农艺农机融合; 田间信息自动获取; 稻田杂草防控; 智能精准除草; 机械-药剂协同控草

Keywords(用分号分开): Agricultural machinery and agronomy integration; Automatic field- information acquisition; Weed control in paddy fields; Intelligent and precise weeding; Mechanical-chemical integration on weed control



项目组主要成员

编号	姓名	出生年月	性别	职称	学位	单位名称	电话	证件号码	项目分工	每年工作时间 (月)
1	齐龙	1979. 11	男	研究员	博士	华南农业大学	020-85283707	230106197911092814	项目负责人	6
2	马锐军	1991. 03	男	讲师	博士	华南农业大学	15989140276	445202199103058530	杂草视觉表征、种类精准辨识及定位方法	6
3	冯骁	1988. 11	男	讲师	博士	华南农业大学	15017513002	230107198811252316	机械作用下杂草叶片创面形成机理	6
4	彭超达	1990. 09	男	讲师	博士	华南农业大学	18819457743	445221199009201918	低空高质量稻田杂草图像快速采集策略	6
5	安静	1989. 11	女	讲师	博士	华南农业大学	15801026018	340621198911290328	机械与药剂协同作用下杂草凋亡机制	6
6	陈瑛	1957. 09	女	教授	博士	University of Manitoba	012043964861	GB331222	稻田杂草叶片本构模型	4
7	王奇	1990. 08	男	副教授	博士	东北农业大学	0451-55190950	231222199008094034	执行机构创新设计与精准对靶作业方法	6
8	刘敏	1986. 03	男	讲师	博士	湖南农业大学	073184618163	430581198603072014	基于稻田杂草繁殖特性的控草作业处方策略	6
9	冯健昭	1981. 10	男	讲师	博士	华南农业大学	020-85280320-604	440602198110310637	低空稻田图像的杂草密度与发生区域信息获取模型	6



总人数	高级	中级	初级	博士后	博士生	硕士生
21	3	6	0	0	2	10



国家自然科学基金预算制项目预算表

项目批准号：U23A20174

项目负责人：齐龙

金额单位：万元

序号	科目名称	金额
1	一、基金资助项目直接费用合计	262.0000
2	1、设备费	15.0000
3	其中：设备购置费	15.0000
4	2、业务费	171.0000
5	3、劳务费	76.0000
6	二、其他来源资金	0.0000
7	三、合计	262.0000

注：请按照项目研究实际需要合理填写各科目预算金额。



预算说明书

（请按照《国家自然科学基金项目申请书预算表编制说明》等的有关要求，按照政策相符性、目标相关性和经济合理性原则，实事求是编制项目预算。填报时，直接费用应按设备费、业务费、劳务费三个类别填报，每个类别结合科研任务按支出用途进行说明。对单价 ≥ 50 万元的设备详细说明，对单价 < 50 万元的设备费用分类说明，**对合作研究单位资质及资金外拨情况、自筹资金进行必要说明。**）

本项目直接经费预算 262 万元，其中外拨资金 20 万元到合作单位湖南农业大学、45 万元到合作单位东北农业大学。预算说明如下：

1、设备费：15 万元。

购买高计算性能计算机工作站 1 台，共 5 万元，用于图像处理、训练深度学习网络、离散元模型仿真模拟。购置多通道数据采集仪和力传感器，共 6 万元，用于采集杂草叶片创面形成受力数据。购置无人机 1 台，共 4 万元，用于拍摄稻田杂草图像。

2、业务费：171 万元（其中湖南农业大学 14 万元，东北农业大学 35 万元）。

（1）材料费：61 万元（其中湖南农业大学 5.2 万元，东北农业大学 14.3 万元）。

所需电控及视觉类材料（40.8 万元）：除草机械臂用无刷电机组件（位置编码器、电机驱动器等）0.5 万元*12（套），共 6 万元；机器人底盘用大力矩减速电机组件（位置编码器、电机驱动器、谐波减速器等）2 万元*4（套），共 8 万元；DSP 及 ARM 控制器开发模块 0.5 万元*16（套），共 8 万元；嵌入式深度学习开发实验板 0.6 万元*2（套），共 1.2 万元；电子元器件 0.3 万元*6（套），共 1.8 万元；磷酸铁锂电池材料 0.7 万元*4（套），共 2.8 万元；视觉元器件（镜头、滤光片、光源等）0.8 万元*5（套），共 4 万元；相机定影组件、减震模块等材料 1.8 万元*5（套），共 9 万元。

所需液力及机械材料（9.2 万元）：高速电磁阀、过滤器，蓄能器，溢流阀，接头等材料，共 3.6 万元；液压泵 0.3 万元*2（套），共 0.6 万元；型材、标准件、管路等机械材料，4 年共计 5 万元。

盆栽试验所需耗材（5 万元）：开展室内盆栽试验所需的营养基质和塑料方盒，共 1 万元；开展室内试验所需的载玻片、枪头、离心管、称量纸、手套、口罩、玻璃器皿等实验室耗材一批，共 3 万元。购买乙二醇、有机溶剂、表面活性剂等常规化学试剂，共 1 万元。

田间试验所需耗材（6 万元）：开展田间试验所需的种子、育苗基质、地膜、化肥、除草剂、插地牌等生产资料，共 6 万元。

（2）测试化验加工费：56 万元（其中湖南农业大学 0 万元，东北农业大学 10 万元）。

除草剂在杂草受损表面的润湿性能和传导效率测试（8 万元）：测定除草剂在杂草叶片上的润湿性能（界面张力仪）4 万元；除草剂在杂草植株中的传导效率（气相色谱仪）4 万元。

杂草叶片本构模型参数测试（10 万元）：工装夹具制作费用 5 万元；细胞拉力仪测试费用 5 万元。

六自由度并联机械臂等加工费用（18 万元）：3D 打印零部件加工费用 6 万元；金属零部件加工费用 12 万元。

末端执行器及药液喷头等加工费用（12 万元）：3D 打印零部件费用 6 万元；金属零部件加工费用 6 万元。

加工移动式地面载具费用（8 万元）：3D 打印零部件费用 4 万元；金属零部件加工费用 4 万元。

（3）燃料动力费：10 万元（其中湖南农业大学 2.4 万元，东北农业大学 0.5 万元）。

用于培养杂草需要用到的人工气候箱、温室等实验室仪器、设备的用电。共需用电 20000 千瓦时/年 $\times 0.6$ 元/千瓦时 $\times 4$ 年=4.8 万元。用于田间试验的除草装备的能源消耗费用 5.2 万元。



预算说明书

（4）差旅/会议/国际合作与交流费：27 万元（其中湖南农业大学 4.4 万元，东北农业大学 7.2 万元）。

装置试制发生的差旅费（1.6 万元）：试制装置期间需前往加工企业，按照每年往返 20 次计算，每次 200 元，4 年共计 1.6 万元。

田间试验发生的差旅费（8 万元）：参加田间试验人员每次 10 人，2 年，早晚 2 季，每人交通费 500 元，计 2 万元；2 年工作共 40 天，每天每人住宿和伙食补助 150 元，计 6 万元。

国内学术会议差旅费（8.4 万元）：每年平均 6 人次参加国内学术会议，交通费每人 2000 元，每天每人住宿 500 元（一般会期 3 天），4 年共计 8.4 万元。

国际交流差旅费（9 万元）：3 人次参加国际会议，差旅费、交通及公杂费共计 9 万元。

（5）出版物/文献/信息传播费：17 万元（其中湖南农业大学 2 万元，东北农业大学 3 万元）。

论文出版费：预计发表论文 12 篇，平均每篇 5000 元，共计 6 万元；申请专利 10 件，平均每件 5000 元，共计 5 万元；EDEM 软件使用授权费每年 1.5 万元，4 年共计 6 万元。

3、劳务费：76 万元（其中湖南农业大学 6 万元，东北农业大学 10 万元）。

共有 2 名博士研究生和 10 名硕士研究生参与本项目的研究工作，其中博士研究生每月补助 3000 元，硕士研究生每月补助 1000 元，每年工作 10 个月。4 年总计劳务费为 64 万元；开展田间试验聘请临时工的劳务费，4 年共计 8 万元。支付聘用专家进行技术咨询和指导费用，预计邀请专家 20 人次，2000 元/人次。共计 4 万元。

本项目两个合作研究单位分别为湖南农业大学和东北农业大学，为高等学校，具有基础科研资质。本项目没有自筹资金。



报告正文

一、 研究内容、研究目标以及拟解决的关键科学问题

1.1 研究内容

1.1.1 基于无人机遥感的稻田杂草密度与发生区域信息快速获取方法研究

（1）基于无人机遥感的高质量稻田杂草图像快速采集策略研究

为提高稻田杂草图像质量和采集效率，研究无人机平台下稻田杂草图像的解译性、采集效率与无人机工作参数的关系，构建低空高质量稻田杂草图像快速采集的多目标优化模型，设计基于深度神经网络的约束多目标群体智能优化算法，探索杂草生长期与最优解集的映射关系，形成复杂环境下稻田杂草图像低空采集策略。

（2）基于低空稻田图像的杂草密度与发生区域信息获取模型研究

研究低空稻田图像中弱目标杂草的快速识别算法，构建杂草密度与图像发生区域信息检测模型；探索无人机平台的“北斗”信息与杂草图像位置的空间匹配关系，实现稻田杂草发生区域的快速定位。基于上述模型和定位方法，建立稻田除草作业“底方图”，指导地面作业机具前往草害发生区域进行除草作业。

1.1.2 基于地面作业机具的杂草视觉表征、种类精准辨识及定位算法研究

（1）近地稻田杂草图像的视觉表征快速提取算法研究

利用地面作业机具搭载图像采集系统，构建基于注意力机制的深度网络框架，研究面向实际稻田环境的杂草图像视觉表征提取算法；利用深度模型中的表征依存关系，探索杂草表征的关联学习机制，研究稻田杂草关键表征的有效强化方法；优化所构网络的参数和计算复杂度，形成轻量级近地稻田杂草图像表征提取网络。

（2）联合表征融合和深度聚类的杂草种类精准辨识模型研究

针对稻田杂草种类多样、禾本科杂草与水稻形似难辨、杂草簇生遮叠等问题，构建表征多尺度融合框架，在此基础上结合深度聚类策略，研究深度表征融合与聚类学习的协同机制，在深度表征空间中提升不同杂草类型区分度和同种类杂草汇聚度，建立高精度的杂草种类辨识模型。

（3）基于深度语义图像建模的稻田杂草定位方法研究

构建基于深度语义的分割网络，研究面向稻田杂草的语义信息定位方法；



探索基于语义表征的深度图像学习机制，研究表型信息驱动的杂草立体深度图像建模方法，实现不同种类杂草形态结构的精准解析。根据上述构建的模型，测定稻田复杂环境下杂草位置信息及作业定位点，为后续的除草作业提供所需的执行参数。

1.1.3 机械-药剂协同作用下不同类型杂草凋亡机制研究

（1）杂草叶片创面形成机理研究

研究杂草叶片在拉伸、剪切等作用下的变形和破坏规律，构建稻田杂草叶片的本构模型；建立杂草叶片的离散元模型，分析不同作业参数条件下机械部件对杂草叶片的破坏规律，探索精准形成理想叶片创面的机械作用方法。

（2）机械-药剂协同作用下杂草凋亡机制研究

采用机械-药剂协同的方式，实现杂草“破而不断”，药剂可在植株体内有效渗透和传导。研究杂草叶片表面不同组织受损后，除草剂在杂草植株上的润湿性能和传导效率；分析不同伤口类型对除草剂剂量效应的影响，探索杂草叶片破损后除草剂的加成及增效规律。

1.1.4 控草作业处方策略及除草执行机构创新研究

（1）基于稻田杂草繁殖特性的智能精准控草作业处方研究

通过小区试验验证单一机械除草及机械-药剂协同除草的杂草防控效果，结合稻田杂草的繁殖特性，形成适配于不同种类杂草的控草作业策略。在田间作业时，根据稻田杂草位置信息，形成稻田除草作业的精细“处方图”。

（2）执行机构创新设计与精准对靶作业方法研究

创新设计兼具切割、刮擦和喷药功能的除草执行机构，构建机构的运动学模型，分析其主要结构与工作参数对工作范围、运动速度、负载能力和稳定性等性能的影响规律，确定最佳参数组合；基于杂草的表征信息及生物力学特性，构建执行机构与稻田杂草的多相耦合动力学模型，分析运动轨迹、角度和速度等作业参数对杂草损伤程度的影响规律，探索在稻田复杂环境中，执行机构的视觉伺服控制与位姿误差补偿方法。

（3）系统集成及田间除草试验

开展智能系统匹配与协同作业试验，研制集成无人机遥感和地面作业机具信息的控制系统及装备原型样机，根据智能辨识的杂草密度、种类、发生区域信息和控草作业策略，验证田间除草效果。



1.2 研究目标

本项目的申请，旨在构建“空地多尺度”获取稻田杂草密度、发生区域、种类和位置等信息的感知模型，提出杂草图像信息快速精准解析算法；探明机械-药剂协同作用下不同繁殖种类杂草的凋亡机制；形成适配于不同杂草类型的控草作业策略，创新机械-药剂协同作业的执行机构和视觉伺服的控制系统，实现精准对靶作业；最终形成高效智能精准除草作业体系，在保证除草效果的同时减少除草剂的使用，为杂草的绿色防控提供理论基础和关键技术支撑。

1.3 拟解决的关键科学问题

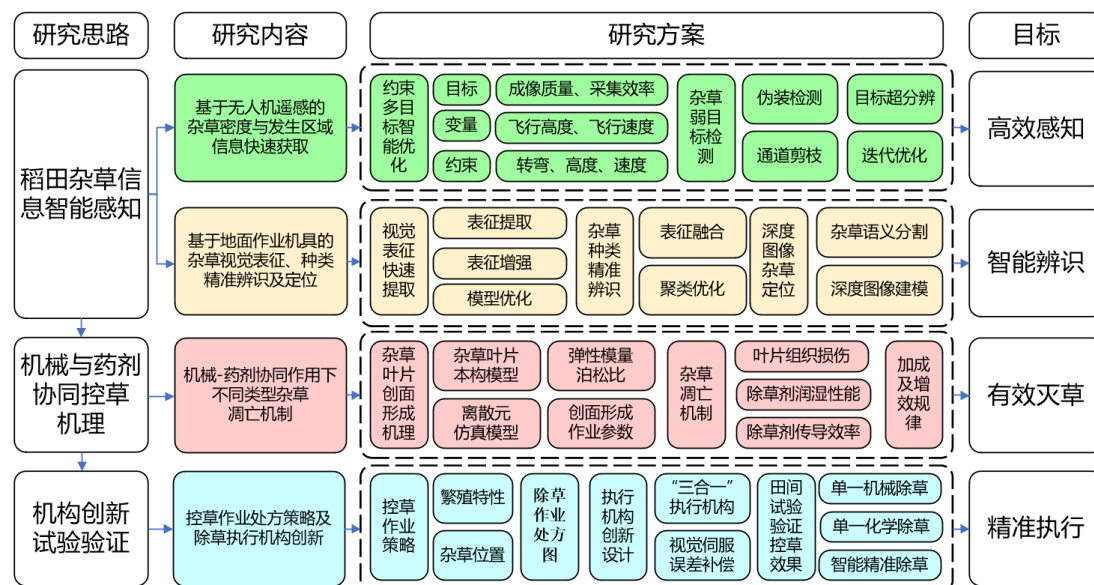
1.3.1 面向无人机低空遥感的弱目标杂草信息超分辨问题。

1.3.2 近地稻田杂草图像特征信息的准确表征及有效增强问题。

1.3.3 除草剂在受损杂草表面的润湿性能、传导效率及增效机制。

二、本项目拟采取的技术路线和研究方案

2.1 技术路线



2.2 研究方案

本项目通过“空地多尺度”感知稻田杂草信息，采用机械除草和化学除草相结合方式对靶作业，实现高效智能精准除草。具体研究方案如下：在稻田杂草信息感知方面，利用无人机巡田，获取杂草密度及发生区域信息，引导地面作业机具到达杂草发生区域进行除草作业，提高机具作业效率；通过作业机具



上搭载的视觉系统，精准获取杂草的种类和位置信息，指导执行机构按杂草种类和位置信息分类对靶作业。在除草方式的决策和执行机构的设计方面，开展机械除草及机械-药剂协同作用下杂草调亡机制研究，针对稻田不同繁殖特性（营养繁殖和种子繁殖）的杂草特点，形成适配于不同类型杂草的控草作业策略；创新设计可实现单一机械除草或机械-药剂协同作业的执行机构及控制系统，实现精准除草作业。本项目具体研究方案如下。

2.2.1 基于无人机遥感的稻田杂草密度与发生区域信息快速获取方法

（1）高质量稻田杂草图像快速采集策略：拟采用分辨率、清晰度、信噪比等作为指标评价低空稻田杂草图像被识别算法所识别到的图像质量。以无人机的飞行高度和飞行速度作为决策变量，飞行力学（即转弯半径、飞行高度和飞行速度）为约束条件，构建图像质量和采集效率的多目标约束函数。设计基于深度神经网络的多目标函数高效约束模型，根据目标与决策搜索空间设计区域自适应优化策略，以获取高质量稻田杂草图像快速采集策略。

（2）杂草密度与发生区域信息获取模型构建：拟将低空稻田杂草检测的问题转换为伪装目标检测的问题，即将杂草视为伪装在稻田中的弱目标，以优化杂草伪装目标边界细节模糊问题为出发点，构建基于迭代超分辨的伪装目标检测网络。拟引入超分辨重建的迭代优化与反馈机制，使模型更关注不同尺度、更细颗粒度的杂草表征，从而实现杂草弱目标的识别。拟通过带有“北斗”定位和惯性传感器等设备的无人机，测得无人机相机拍照瞬间相机的高度、经纬度坐标以及相机三轴方向的角度信息，形成作业底方图，以指导地面作业机具开展除草作业。

2.2.2 基于地面作业机具的杂草视觉表征、种类精准辨识及定位方法

（1）近地稻田杂草图像的视觉表征快速提取算法：拟将输入杂草图像切分为固定大小的分块，结合位置嵌入，将图像进行序列化输入分层级联 Transformer 网络，以提取更具语义信息的杂草视觉表征；拟从表征的感知融合和表征的局部-全局耦合两个层面设计增强算法，促使杂草图像视觉信息得到更好的表达。设计基于知识蒸馏的网络框架，将上述构建的杂草表征的提取和强化网络迁移到具有较少权值参数的网络中，以实现近地稻田杂草图像表征提取网络的轻量级优化。

（2）联合表征融合和深度聚类的杂草种类精准辨识模型：拟利用不同尺度的卷积核对所提取的杂草表征进行多尺度挖掘。随后，构建深度聚类网络。拟



采用最小二乘回归理论分别学习同类杂草的汇聚度转换矩阵和不同种类杂草的区分度转换矩阵。在同类杂草聚类子空间，以区分度转换矩阵为约束进行类内聚类关系的优化，从而迫使类内聚类产生一致且有判别度的回归；采用相同的方式，在不同种类杂草聚类子空间，以汇聚度转换矩阵为约束增大类间区分度距离。最后，将学习到的汇聚度和区分度转换矩阵进行全局表征的聚类优化，确定最佳的杂草类别归属。

（3）基于深度语义图像建模的稻田杂草定位方法：拟采用基于改进 U-Net 的深度网络进行杂草语义分割，以获得稻田和杂草像素的分类得到其对应的位置信息。随后，将提取的杂草表征输入 PatchMatch 模块中，结合语义分割网络获取的杂草像素轮廓初始化杂草深度图像。在网络迭代训练过程中，拟利用杂草表型先验信息，结合注意力机制最大化深度层特征之间的相关性。拟将语义分割网络获取的语义信息与杂草的深度图像表征进行融合，采用最近邻算法快速查找图像块之间的近似最近邻匹配，从而实现杂草深度图像信息的细节优化，为对靶除草作业提供精准的深度图像信息。

2.2.3 机械-药剂协同作用下不同类型杂草凋亡机制

（1）杂草叶片创面形成机理：拟根据不同稻田杂草特性，构建杂草叶片的本构模型，使用定制刀具去除杂草叶片上表皮区域形成叶片创面，分析在拉伸应力和剪切应力作用下的对杂草的叶片变形和破坏规律；测定叶片的弹性模量和泊松比，使用 EDEM 构建叶片离散元仿真模型，通过试验数据对离散元模型中键与粒子的参数进行辨识；并分析球形砂轮在不同速度、角度条件下对叶面造成创口的规律，确定精准形成特定深度和范围的叶片创面所需的作业参数。

（2）机械-药剂协同作用下杂草凋亡机制：拟通过测定除草剂液滴在杂草叶片不同伤口上的表面自由能、亲水性、疏水性、接触角和扩散半径，分析除草剂在杂草叶片不同伤口上的润湿性能。以五氟磺草胺和敌草快作为药剂试样，在杂草倒 3 叶上进行伤口处理，并涂抹田间推荐剂量的除草剂，分别于处理后不同时间收取杂草新叶、处理叶片、功能叶、成熟叶及茎部组织，使用气相色谱仪测定各组织中五氟磺草胺或敌草快含量的变化，解析除草剂在杂草叶片不同伤口上的传导效率。最后，对供试杂草叶片进行损伤处理后，与不同浓度除草剂复配组合处理，以用单独机械伤草、除草剂处理及清水为对照处理后测定其株防效和鲜重防效，评定机械伤草后除草剂的加成及增效规律。

2.2.4 控草作业处方策略及除草执行机构创新



（1）基于稻田杂草繁殖特性的控草作业处方：于水稻播种或插秧后 7d 进行单一机械除草和机械-药剂协同除草两张不同方式的杂草处理，并在处理后 15d、30d、45d 调查防治效果。对于不同繁殖特性的杂草，采用独立样本 T-test（双尾）检验的方法分析单一机械除草和机械-药剂协同除草的除草效果差异，以除草效果显著高者为其除草策略，若无显著差异则以单一机械除草为其除草策略，从而形成适配于不同繁殖特性杂草的智能精准除草方案。

（2）执行机构创新设计与精准对靶作业方法：以搭载 9 台独立驱动器的并联机械臂为基础，创新设计基于多自由度并联机械臂的除草执行机构。设计具有切割、刮擦和喷药功能的“三合一”末端执行器。研究机械-药剂协同作业集成控制方法，依据除草策略调整砂轮转速、末端执行器位姿和除草剂喷射压力等作业参数，确保机械-药剂协同作用达到最佳效果。采用视觉伺服控制的方法实现执行机构的精确定位和误差补偿。集成运动学、动力学模型以及视觉伺服控制模型，计算随机动载荷条件下执行机构的位姿误差和补偿量，实现执行机构在稻田环境下的高精度控制。

（3）系统集成及田间试验：研制集成无人机遥感和地面作业机具信息的除草装备原型样机，在华南双季稻区开展田间除草试验，分别采用单一机械除草、单一化学除草和智能精准除草（本项目提出）三种不同除草方式进行处理，并设置不除草的对照组。为测定不同除草处理对水稻生长和产量的影响，在水稻生长的分蘖中期、幼穗分化期、抽穗期和成熟期开展田间试验，测试杂草防控效果。

三、研究的难点和在实施过程中可能遇到的问题及拟采取的研究方案与技术路线

难点 1：本项目研究内容一和研究内容二的关键问题是面向无人机低空遥感的弱目标杂草信息超分辨，以及近地稻田杂草图像特征信息的准确表征及有效增强。现有的深度网络模型（如表征提取、目标检测、深度聚类 and 语义分割等）可类比应用并解决上述关键问题。在实施过程中开源的网络模型或软件/框架（如 MATLAB, Pytorch 等）可能被禁用。针对这一问题，我们拟采用的研究方案和技术路线如下：1）项目参与人在表征提取模型构建、注意力机制和语义分割方面，具有扎实的研究基础，可基于论文实现代码的复现和算法的创新；2）采用国产自主研发的深度学习架构/平台（如百度的 Paddle, 阿里的 X-Deep Learning, Truffer 等）进行神经网络模型搭建，确保软件系统的平稳运行。



难点 2: 本项目内容三在实施过程中可能存在的问题是营养繁殖的杂草难以用种子繁殖，而不同生长周期的营养枝生长、繁殖能力不同，长势难以控制。针对这一问题，我们拟采用的研究方案和技术路线如下：1）从田间采集长势良好的营养繁殖杂草的枝条、地下块茎等营养繁殖体，选择其适宜的土壤介质在温室中进行种植，对其进行扩繁；2）从扩繁材料中选取长势一致的营养体进行再次种植，待其生长至可供处理时，挑选生长一致的植株用于试验处理，确保试验处理时杂草处于同一生长状态，以减少试验误差。

难点 3: 本项目内容四在实施过程中可能存在除草执行机构加工精度不足导致除草过程中产生较大运动误差的问题，进而影响作业精度。针对这一问题，我们拟采用的研究方案和技术路线如下：1）在零部件加工过程中，选用高强度材料（如碳纤维），采用金属增材制造技术，以提高齿轮、铰链等关键部件的稳定性和加工精度；2）在零部件组装过程中，采用精密的组装工艺减小装配误差。同时，保持良好润滑降低机械传动部件损耗，以提高除草机构整机运行的可靠性。

四、项目主要参与者分工，合作研究单位之间的关系与分工

本项目牵头单位为华南农业大学，合作研究单位为东北农业大学和湖南农业大学。

华南农业大学的齐龙研究员为项目负责人，负责制定项目实施工作方案，全面掌握项目进展情况。主要参与者来自水稻生产机械化团队和作物生理生态科研团队，负责稻田杂草信息智能感知、机械与药剂协同作用下杂草凋亡机制、对靶精准作业控制等研究内容。其中，马锐军博士负责杂草视觉表征、种类精准辨识及定位方法的研究；冯骁博士负责机械作用下杂草叶片创面形成机理的研究；彭超达博士负责低空高质量稻田杂草图像快速采集策略的研究；安静博士负责机械与药剂协同作用下杂草凋亡机制的研究；冯健昭博士负责低空稻田图像的杂草密度与发生区域信息获取模型的研究。

项目参与单位东北农业大学的主要参与者王奇博士，长期致力于水田杂草物理特性测定、机械除草关键技术与除草部件的研究，负责除草执行机构的创新设计。湖南农业大学主要参与者刘敏博士，是柏连阳院士领衔的杂草生物学及安全防控湖南省重点实验室的核心成员之一，负责水田杂草精准防控作业处方策略的研究。此外，项目参与者加拿大曼尼托巴大学陈瑛教授负责稻田杂草叶片本构模型的构建。



综上所述，本项目牵头单位华南农业大学与合作单位研究领域特色明显，申请团队前期基础扎实，团队人员合理，能共同保证项目按计划高质量完成。

五、详细的年度研究计划及预期研究结果

5.1 年度研究计划

第一年度：2024.1~2024.12

开展稻田优势杂草种群调查，明确稻田主要防治靶标；研究无人机视角下高质量稻田杂草图像快速采集策略，并采集低空稻田杂草图像；构建基于深度学习的稻田杂草表征提取网络框架，实现杂草表征的增强和模型优化算法；分析杂草叶片力学性能，构建叶片的本构模型；研究除草剂在破损叶片上润湿性能和传导效率。发表高水平论文 2-3 篇；申请发明专利 2-3 件。

第二年度：2025.1~2025.12

构建基于无人机遥感图像的稻田杂草密度和发生区域感知算法；研究面向稻田杂草辨识任务的聚类优化模型；针对不同繁殖特性的杂草开展除草效果验证试验，分析杂草受损后除草剂的加成及增效规律；设计机械-药剂协同除草执行机构及末端执行器，构建运动学、动力学模型，优化设计参数。邀请国内外专家进行学术交流。发表高水平论文 2-3 篇；申请发明专利 2-3 件。

第三年度：2026.1~2026.12

构建面向稻田杂草的深度语义分割模型，实现杂草的位置检测和结构实例分割；开展小区试验，测试单一机械除草和机械-药剂协同除草对稻田杂草的防除效果，形成适配于不同繁殖特性杂草的精准智能除草作业策略；开展除草执行机构与视觉伺服控制系统集成验证试验。参加国际会议与国外本领域专家交流，或参加国内学术会议。发表高水平论文 3-4 篇；申请发明专利 2-3 件。

第四年度：2027.1~2027.12

研制集成无人机遥感和地面作业机具信息的除草装备原型样机；在华南双季水稻产区开展稻田田间除草试验，验证智能精准除草效果，并根据试验结果优化执行机构设计；总结研究过程中形成的基础理论、基本方法、关键技术等，对相关的学术成果、影像、图片资料等进行整理，撰写研究报告，准备验收汇报。发表高水平论文 3-4 篇；申请发明专利 2-3 件。

5.2 预期研究结果

本项目完成后，将形成比较完整的稻田高效智能精准杂草防控理论和方法。



提出空地多尺度的稻田杂草发生信息精准感知模型和算法，探明机械-药剂协同作用下杂草凋亡机制，创制出新型的智能除草装置和控制系统。项目研究成果拟以论文、专利、人才培养等形式体现，具体如下：

- （1）发表高水平学术论文 10~12 篇；
- （2）申请发明专利 8~10 件；
- （3）培养青年骨干教师 5~7 名，研究生 10~12 名。



国家自然科学基金项目负责人、依托单位承诺书

国家自然科学基金项目负责人承诺书

本人郑重承诺：我接受国家自然科学基金的资助，严格遵守中共中央办公厅、国务院办公厅《关于进一步加强科研诚信建设的若干意见》《关于进一步弘扬科学家精神加强作风和学风建设的意见》《关于加强科技伦理治理的意见》等规定，及国家自然科学基金委员会关于资助项目管理、项目资金管理等各项规章，在《计划书》填写及项目执行过程中：

（一）按照《批准通知》《国家自然科学基金资助项目计划书填报说明》的要求填写《计划书》，未自行降低、更改目标任务或约定要求，或缩减研究（研制）内容；

（二）树立“红线”意识，严格履行科研合同义务，按照《计划书》负责实施本项目（批准号：U23A20174），切实保证研究工作时间，按时报送有关材料，及时报告重大情况变动，不违规将科研任务转包、分包他人，不以项目实施周期外或不相关成果充抵交差；

（三）遵守科研诚信、科技伦理规范和学术道德，认真开展研究工作，对资助项目发表的论著和取得的科研成果按规定进行标注，不在非本项目资助的成果或其他无关成果上标注本项目批准号，反对无实质学术贡献者“挂名”，不在成果署名、知识产权归属等方面侵占他人合法权益，并如实报告本人及项目组成员发生的违背科研诚信要求的任何行为；

（四）尊重科研规律，弘扬科学家精神，严谨求实，追求卓越，反对浮夸浮躁、投机取巧，不人为夸大学术或技术价值，不传播未经科学验证的现象和观点；

（五）将项目资金全部用于与本项目研究工作相关的支出，并结合科研活动需要，科学合理安排项目资金支出进度；

（六）做好项目组成员的教育和管理，确保遵守以上相关要求。

如违背上述承诺，本人愿接受国家自然科学基金委员会和相关部门做出的各项处理决定。

项目负责人（签字）：

年 月 日

依托单位科研管理部门：

负责人（签章）：

年 月 日

依托单位财务管理部门：

负责人（签章）：

年 月 日

国家自然科学基金项目依托单位承诺书

我单位同意承担上述国家自然科学基金项目，将保证项目负责人及其研究队伍的稳定和研究项目实施所需的条件，严格遵守国家自然科学基金委员会有关资助项目管理、项目资金管理、科研诚信管理和科技伦理管理等各项规定，并督促实施。

依托单位（公章）

年 月 日



本栏目由自然科学基金委填写

负责人（签章）：
年 月 日

负责人（签章）：
年 月 日

子课题编号：2021YFD2000201-01-02

密 级：公开

“十四五”国家重点研发计划 子课题任务书

子课题名称(编号)：	基于显微图像的土壤气相特性信息田间原位快速感知方法（2021YFD2000201-01-02）
子课题承担单位：	华南农业大学
子课题负责人：	杨秀丽
课题名称（编号）：	土壤有机质和结构复合特性快速现场检测（2021YFD2000201-01）
课题承担单位：	吉林大学
执行期限：	2021 年 12 月 至 2026 年 11 月



中华人民共和国科学技术部制

六、子课题参加人员基本情况表

填表说明: 1. 职称分类: A、正高级 B、副高级 C、中级 D、初级 E、其他;
 2. 投入本课题的全时工作时间(人月)是指在课题实施期间该人总共为课题工作的满月度工作量; 累计是指课题组所有人员投入人月之和;
 3. 课题固定研究人员需填写人员明细;
 4. 是否有工资性收入: Y、是 N、否;
 5. 人员分类代码: A、子课题负责人 B、子课题骨干 C、其他研究人员;
 6. 工作单位: 填写单位全称, 其中高校要具体填写到所在院系。

序号	姓名	性别	出生日期	身份证号码 (军官证、护照)	技术 职称	职务	学位	专业	投入本课题的 全时工作时间 (人月)	人员 分类	是否有 工资性 收入	工作单位
1	杨秀丽	女	1971.05.17	142331197105171048	C	无	博士	农业电气化 与自动化	30	A	Y	华南农业大学工程学院
2	彭超达	男	1990.09.20	445221199009201918	C	无	博士	控制科学与 工程	30	B	Y	华南农业大学数学与信息学院 软件学院
3	唐震宇	男	1995.10.30	340823199510301936	E	无	学士	农业工程	30	B	N	华南农业大学工程学院
4	傅灯斌	男	1997.03.05	500234199703058094	E	无	学士	农业机械化 工程	30	B	N	华南农业大学工程学院
5	王志琪	男	1997.12.27	150404199712272612	E	无	学士	机械	30	B	N	华南农业大学工程学院
6	安尔泰	男	1999.07.10	342626199907100374	E	无	学士	机械	30	B	N	华南农业大学工程学院
7	冯健	男	1998.06.13	411328199806137833	E	无	学士	机械	30	B	N	华南农业大学工程学院
8	林敬川	男	2000.02.07	442000200002076616	E	无	学士	农业机械化 及其自动化	30	C	N	华南农业大学工程学院

9	雷俊求	男	2000.09.11	440781200009111511	E	无	学士	农业机械化 及其自动化	30	C	N	华南农业大学工程学院
								固定研究人员合计	270	/	/	/
								流动人员或临时聘用人员合计	0	/	/	/
								累计	270	/	/	/

课题承担单位（甲方）：（公章）

课题负责人（签字）：

年 月 日

子课题承担单位（乙方）：（公章）

子课题负责人（签字）：

年 月 日

受理编号: c23140500000850

项目编号: 2023A1515012885

文件编号: 粤基金字(2023) 2号

广东省基础与应用基础研究基金项目

任务书

项目名称: 面向多组学数据的高阶融合网络建模

项目类别: 广东省自然科学基金-面上项目

项目起止时间: 2023-01-01 至 2025-12-31

管理单位(甲方): 广东省基础与应用基础研究基金委员会

依托单位(乙方): 华南农业大学

通讯地址: 广东省广州市天河区五山路483号

邮政编码: 510642

单位电话: 020-85283435

项目负责人: 王海燕

联系电话: 15013255687



(广东科技微信公众号)



(查看任务书信息)



(受理纸质材料二维码)

广东省基础与应用基础研究
基金委员会
二〇二〇年制

填写说明

一、项目任务书内容原则上要求与申报书相关内容保持一致，不得无故修改。

二、项目承担单位通过广东省科技业务管理阳光政务平台下载项目任务书，按要求完成签名盖章后扫描上传到广东省科技业务管理阳光政务平台。

三、签名盖章说明。请分别在单位工作分工及经费分配情况页、人员信息页、签约各方页等地方按要求签字或盖章，签章不合规或错漏将不予受理。其中，人员信息页要求所有参与人员本人亲笔签名，代签或印章无效，漏签将不予受理。

四、本任务书自签字并加盖公章之日起生效，各方均应负本任务书的法律责任，不应受机构、人事变动影响。

五、根据《广东省科学技术厅广东省财政厅关于深入推进省基础与应用基础研究基金项目经费使用“负面清单+包干制”改革试点工作的通知》（粤科规范字[2022]2号），2022年度及以后立项资助的全部省基金项目（包括省自然科学基金、省市联合基金、省企联合基金项目等）均适用“负面清单+包干制”，项目提交申请书和任务书时无需编制费用明细科目预算。

一、主要研究内容和要达到的目标

主要研究内容如下：

（一）基于张量二部图的高阶融合低维表示学习模型构建

多组学数据具有多源差异性、信息互补和语义一致的优势，提取组学间的高阶相关统计，加深对数据整体的理解是重要的科学挑战。本项目拟在前期的张量学习方法上（工作基础2），利用张量理论的优势设计有效的高阶融合学习算法。在这一框架下，设计多组学张量二部图学习，深入研究多组学数据间的互补信息和高阶相关性。同时迭代利用所学二部图张量进行Tucker分解，量化各组学的重要性并从各组学中抽取适量信息识别数据的统一低维度表示，提升多组学数据表示学习的性能。

（二）基于流形度量的高阶超图网络融合学习模型构建

本项目拟在前期Grassmann流形度量的学习模型框架下（工作基础1），开展后续的研究工作。基于充分调研流形学习的数学理论和超图网络建模学习的基础上，在对组学数据进行高阶超图网络构建的同时进行共享子空间识别。模型使得超图刻画组学内部具有高阶局部亲和力的样本关系网络；更使得多组学子空间在Grassmann流形上充分对齐，有效避开欧式度量复杂多组学数据产生的不准确性，实现新度量空间下的多组学整合分析方法研究。

（三）基于高阶相似度的组学网络融合学习模型构建

多组学数据在有噪声的情况下，难以刻画准确的样本关系。本项目拟设计基于高阶相似度的组学高阶融合网络模型，用于挖掘多组学数据中可靠的样本关联关系，减小高噪声的生物数据分析挑战，以期实现关键生物标志物挖掘及精准疾病诊断。通过建立组学高阶张量相似度及对应高阶相似度表示，寻找隐藏在组学样本中的高阶关联性，有效缓解由于数据噪声带来的影响。

具体研究目标如下：

- 1) 建立基于张量二部图的高阶融合低维表示学习模型，利用嵌入在单组学二部图中的空间结构和互补信息，通过张量理论充分挖掘组学间的高阶相关关系，实现疾病关联基因模块的准确挖掘。
- 2) 建立基于流形度量的高阶超图网络融合学习模型，利用高阶超图网络学习具有高阶局部亲和力的样本关系网络，同时使得多组学子空间在流形度量空间充分对齐，学习有效的共享子空间学习，实现流形度量空间下的多组学整合分析研究，提高共表达模块识别的鲁棒性，减少疾病诊断的假阳率。
- 3) 建立基于高阶相似度的组学网络融合学习模型，利用四阶张量建模组学样本点对点-点对关系，为点-点关系提供补充，充分挖掘组学样本相似度信息，运用高阶相似度理论挖掘建模学习多组学共享基因模块，实现去噪声和整合等任务，提升临床精准医疗的准确率。

二、项目预期获得的研究成果及形式

论文及专著情况	国家统计源刊物以上刊物 发表论文（篇）		3		科技报告（篇）			
	其中被SCI/EI/ISTP收录 论文数（篇）		3		培养人才（人）			
	专著（册）				引进人才（人）			
专利情况(项)	发明专利		实用新型专利		外观设计专利		国外专利	
	申请	授权	申请	授权	申请	授权	申请	授权
	2							

三、项目进度和阶段目标

(一) 项目起止时间: 2023-01-01 至 2025-12-31		
(二) 项目实施进度及阶段主要目标:		
开始日期	结束日期	主要工作内容
2023-01-01	2023-12-31	<p>(1) 研究计划: 依托于张量计算理论, 通过张量二部图学习、张量Tucker分解的有机结合, 建立基于张量二部图的高阶融合低维表示学习模型。</p> <p>(2) 阶段目标:</p> <p>建立基于张量二部图的高阶融合低维表示学习模型;</p> <p>深入了解国内外学术界最新发展, 参加重要国内外学术会议1-2次;</p> <p>发表高水平论文1篇。</p>
2024-01-01	2024-12-31	<p>(1) 研究计划: 在充分文献方法调研的基础上, 结合超图及流形学习度量理论, 完成基于流形度量的高阶超图网络融合学习模型构建, 给出高效的数值求解方案。</p> <p>(2) 阶段目标:</p> <p>建立基于流形度量的高阶超图网络融合学习模型;</p> <p>深入了解国内外学术界最新发展, 参加重要国内外学术会议1-2次;</p> <p>发表高水平论文1篇, 申请国家发明专利1项。</p>
2025-01-01	2025-12-31	<p>(1) 研究计划: 结合张量谱分析理论, 建立基于高阶相似度的组学网络融合学习模型, 给出高效的数值求解方案。</p> <p>(2) 阶段目标:</p> <p>建立基于高阶相似度的组学网络融合学习模型、参加重要国内外学术会议1-2次;</p> <p>发表高水平论文1篇, 申请国家发明专利1项。</p>

四、项目总经费及省基金委经费预算

1. 省基金委经费下达总额：（大写）壹拾万圆整；（小写）10万元；

2. 省基金委经费年度下达计划：

年度	2023 年	年	年	年	年
经费(万元)	10.00				

2023A1515012885

五、人员信息


项目负责人

姓名	证件号码	年龄	性别	职称	学历	在项目中承担的任务	所在单位	签名
王海燕	340621198704047560	36	女	讲师	博士研究生	项目负责人	华南农业大学	王海燕

项目组主要成员

姓名	证件号码	年龄	性别	职称	学历	在项目中承担的任务	所在单位	签名
黄栋	441622198711022098	36	男	副教授	博士研究生	算法检验	华南农业大学	黄栋
彭超达	445221199009201918	33	男	副教授	博士研究生	算法检验	华南农业大学	彭超达
王金凤	130105197810080929	45	女	副教授	博士研究生	算法检验	华南农业大学	王金凤
张广煜	440106199105151814	32	男	讲师	博士研究生	算法检验	华南农业大学	张广煜
崔金荣	410927198412094042	39	女	讲师	博士研究生	算法检验	华南农业大学	崔金荣

六、工作分工及财政经费分配

承担/参与单位名称 (盖章)	工作分工	省级财政科技资金分配 (万元)
 华南农业大学	项目负责人负责跟进年度执行目标的达成，并完成主要的科研工作；为保障项目顺利进行，合作老师负责辅助检验算法及实验数据；同时计划引进一名硕士生，负责数据清洗及辅助算法调试。	10.00
	合计	10.00

七、任务书条款

第一条 甲方与乙方根据《中华人民共和国民法典》及国家有关法规和规定，按照《广东省科学技术厅关于广东省基础与应用基础研究基金（省自然科学基金、联合基金等）项目管理的实施细则（试行）》《广东省省级科技计划项目验收结题工作规程（试行）》等规定，为顺利完成（2023）年面向多组学数据的高阶融合网络建模专项项目（文件编号：粤基金字（2023）2号）经协商一致，特订立本任务书，作为甲乙双方在项目实施管理过程中共同遵守的依据。

第二条 甲方的权利义务：

1. 按任务书规定进行经费核拨的有关工作协调。
2. 根据甲方需要，在不影响乙方工作的前提下，定期或不定期对乙方项目的实施情况和经费使用情况进行检查或抽查。
3. 根据《广东省科研诚信管理办法(试行)》等规定对乙方进行科技计划信用管理。

第三条 乙方的权利义务：

1. 确保落实自筹经费及有关保障条件。
2. 按任务书规定，对甲方核拨的经费实行专款专用，单独列账，并随时配合甲方进行监督检查。
3. 经费使用按照广东省级财政科研项目经费使用等有关规定进行管理。
4. 项目依托单位应制定经费使用“负面清单+包干制”内部管理制度并报甲方备案。
5. 使用财政资金采购设备、原材料等，按照《广东省实施〈中华人民共和国招标投标法〉办法》有关规定，符合招标条件的须进行招标。
6. 项目任务书任务完成后，或任务书规定的任务、指标及经费投入等提前完成的，乙方可提出验收结题申请，并按甲方要求做好项目验收结题工作。
7. 若项目发生需要终止结题的情况，乙方须提出终止结题申请，并按甲方要求做好项目终止结题工作。
8. 在每年规定时间内向甲方如实提交上年度工作情况报告，报告内容包含上年度项目进展情况、经费决算和取得的成果等。
9. 按照国家和省有关规定，提交科技报告及其他材料。
10. 利用甲方的经费获得的研究成果，项目负责人和参与者应当注明获得“广东省基础与应用基础研究基金（英文：Guangdong Basic and Applied Basic Research Foundation）（项目编号）”资助或作有关说明。
11. 乙方要恪守科学道德准则，遵守科研活动规范，践行科研诚信要求，不得抄袭、剽窃他人科研成果或者伪造、篡改研究数据、研究结论；不得购买、代写、代投论文，虚构同行评议专家及评议意见；不得违反论文署名规范，擅自标注或虚假标注获得科技计划（专项、基金等）等资助；不得弄虚作假，骗取科技计划（专项、基金等）项目、科研经费以及奖励、荣誉等；不得有其他违背科研诚信要求的行为。
12. 确保本项目开展的研究工作符合我国科技伦理管理相关规定。

第四条 在履行本任务书的过程中，如出现广东省相关政策法规重大改变等不可抗力情况，甲方有权对所核拨经费的数量和时间进行相应调整。

第五条 在履行本任务书的过程中，当事人一方发现可能导致项目整体或部分失败的情形时，应及时通知另一方，并采取适当措施减少损失，没有及时通知并采取适当措施，致使损失扩大的，应当就扩大的损失承担责任。

第六条 本项目技术成果的归属、转让和实施技术成果所产生的经济利益的分享，除双方另有约定外，按国家和广东省有关法规执行。

第七条 根据项目具体情况，经双方另行协商订立的附加条款，作为本任务书正式内容的一部分，与本任务书具有同等效力。

第八条 本任务书一式三份，各份具有同等效力。甲、乙方及项目负责人各执一份，三方签字、盖章后即生效，有效期至项目结题后一年内。各方均应负任务书的法律责任，不应受机构、人事变动的影响。

第九条 乙方必须接受甲方聘请的本项目任务书监理单位的监督和管理。监理单位按照甲方赋予的权利对本项目任务书的履行进行审核、进度调查，对项目任务书变更、经费使用情况进行监督管理及组织项目验收。

说明：1. 本任务书中，凡是当事人约定无需填写的内容，应在空白处划（/）。

2. 委托代理人签订本任务书的，应出具合法、有效的委托书。

八、本任务书签约各方

管理单位（甲方）：

广东省基础与应用基础研究基金委员会（盖章）

法定代表人（或法人代理）：

曾路

（签章）

2023 年 02 月 14 日

依托单位（乙方）：

华南农业大学

（盖章）

法定代表人（或法人代理）：

刘雅红

（签章）

联系人（项目主管）姓名：

倪慧群

（签章）

Email: kjcgxk@scau.edu.cn

电话: 020-85283435 / 15920301530

开户单位名称: 华南农业大学

开户银行名称: 广东广州工行五山支行

开户银行帐号: 3602002609000310520

2023 年 2 月 24 日

联系人（项目负责人）姓名：

王海燕

（签名）王海燕

Email: cshywang@scau.edu.cn

电话: 15013255687

2023 年 02 月 26 日

2024年度广东省基础与应用基础研究基金自然科学基金面上和青年提升
拟立项项目及资金安排表

单位：万元

序号	主管部门	项目名称	申报单位	负责人	拟立项 金额	2024年 拟拨付 金额	项目 类型
总计（3600项）					58500	58500	
一	省直部门						
(一)	省教育厅						
1	南方医科大学				3705	3705	
		电磁扰动技术在早期发现颅脑损伤患者病情变化的基础与应用方法研究	南方医科大学	包贇	30	30	青年提升
		MTDH-PGAM1轴调控糖酵解介导FOXO3a乳酸化修饰促进糖尿病肾病肾小管上皮细胞损伤的机制研究	南方医科大学	陈晓雯	30	30	青年提升
		FAM134B-PTEN-Ca2+诱导mtDNA氧化释放在GO致神经细胞焦亡的机制研究	南方医科大学	冯晓黎	30	30	青年提升
		肺IM通过外泌体诱发癌症相关血栓栓塞的机制研究	南方医科大学	李建龙	30	30	青年提升
		IL-17A+MRC1+腹腔巨噬细胞通过JAK2/STAT3信号驱动重症中暑肠道损伤的作用和机制研究	南方医科大学	李莉	30	30	青年提升
		ECHS1调控磷脂代谢重编程诱导肿瘤相关巨噬细胞极化促进结直肠癌耐药的分子机制	南方医科大学	李睿	30	30	青年提升
		星形胶质细胞通过与中间神经元的相互作用调控LTD及恐惧记忆	南方医科大学	刘吉红	30	30	青年提升
		颞下颌关节骨关节炎中Wnt16调控软骨下骨异常骨改建的分子机制研究	南方医科大学	刘显文	30	30	青年提升
		DHCR24的介导星形胶质细胞脂质代谢障碍在ACP损伤下丘脑中的机制探究	南方医科大学	刘忆	30	30	青年提升
		MDM2介导小胶质细胞糖代谢重编程调控抗肿瘤免疫促进胶质瘤免疫抑制微环境形成的机制研究	南方医科大学	刘卓浩	30	30	青年提升
		OGDH通过YTHDF3的琥珀酰化促进炎症牙髓干细胞成牙本质向分化的机制研究	南方医科大学	罗海芸	30	30	青年提升
		5-氨基戊酸诱导NOL8通过液-液相分离促进β-Catenin核蓄积介导结直肠癌肝转移的机制研究	南方医科大学	史蛟龙	30	30	青年提升
		HIF-1α调节弥漫大B细胞淋巴瘤ROS介导耐药的机制研究	南方医科大学	魏小磊	30	30	青年提升
		RAGE/β-catenin轴通过Neuropilin-1调控气道上皮细胞线粒体融合介导TDI激素抵抗型哮喘气道炎症的机制研究	南方医科大学	赵文驱	30	30	青年提升

序号	主管部门	项目名称	申报单位	负责人	拟立项 金额	2024年 拟拨付 金额	项目 类型
		AMP通过AMPK/FoxO1调控AS肉鸡肺血管重构发生的作用机制	华南农业大学	李英	15	15	面上
		稻瘟菌非经典分泌型效应子的高通量筛选与功能分析	华南农业大学	李云锋	15	15	面上
		新型高光学纯吡啶骨架杀虫剂的设计、合成和作用机制研究	华南农业大学	李兆栋	15	15	面上
		辣椒素生物合成的可变剪切调控机制研究	华南农业大学	廖毅	15	15	面上
		OsATL15-MADS55互作介导水稻定向积累噻虫嗪防控稻飞虱的分子机制	华南农业大学	林菲	15	15	面上
		呕吐毒素诱发内质网应激影响胆汁酸稳态的机制研究	华南农业大学	林如琴	15	15	面上
		不对称式毛竹衍生碳复合材料的构筑及其电磁屏蔽性能研究	华南农业大学	林秀仪	15	15	面上
		基于最大和次大邻接特征值的一些图谱问题的研究	华南农业大学	刘木伙	15	15	面上
		杨树阿拉伯糖基木聚糖的合成及其对材性的影响	华南农业大学	刘思雯	15	15	面上
		铁还原菌的质子传递机制解析与合成生物学调控	华南农业大学	刘维	15	15	面上
		甲萘醌抑制耐药质粒接合转移的机制研究	华南农业大学	刘艺云	15	15	面上
		蓝莓花色苷调节肠道菌群抑制黄曲霉毒素B1诱发肝细胞焦亡的机制研究	华南农业大学	刘韵乐	15	15	面上
		基于多组学探讨烟酰胺单核苷酸对间歇性禁食所致能量代谢重编程的调节效应	华南农业大学	柳春红	15	15	面上
		物联网场景中面向无服务器边缘计算架构的工作流调度与资源优化	华南农业大学	罗浩宇	15	15	面上
		狂犬病病毒不同毒株感染激活RIG-I信号通路的差异性研究	华南农业大学	罗永文	15	15	面上
		云计算环境下的多用户密文检索技术	华南农业大学	马莎	15	15	面上
		AuroraB偶联DNA损伤修复和细胞周期G1/S阻滞响应T-2毒素压力的分子机制	华南农业大学	母培强	15	15	面上

顺序号	2414050002700
项目类别	面上项目

广东省基础与应用基础研究基金
自然科学基金项目（面上项目）

申 请 书

业务类别：	广东省自然科学基金-面上项目
项目名称：	物联网场景中面向无服务器边缘计算架构的工作流调度与资源优化
申请人姓名：	罗浩宇
依托单位：	华南农业大学
邮政编码：	510642
通讯地址：	广东省广州市天河区五山路483号
申请人电话：	13022092227
申请人电子邮箱：	haoyuluo@scau.edu.cn
单位联系人：	倪慧群
单位电话：	020-85283435
申请日期：	2023年06月02日

广东省基础与应用基础研究基金委员会
二 0 二 三 年 制



（广东科技微信公众号）



（受理纸质材料二维码）

一、基本信息

研究项目情况	名称	物联网场景中面向无服务器边缘计算架构的工作流调度与资源优化						
	类别	广东省自然科学基金-面上项目			研究类型	应用基础研究		
	申报学科	名称1	并行与分布式软件 (F020206)		代码1	F020206		
		名称2	软件工程 (F020202)		代码2	F020202		
	申请金额 (万元)	20.00		研究期限	2024年01月01日 -- 2026年12月31日			
申请人情况	姓名	罗浩宇	性别	男	身份号码	360302198907132536	民族	汉族
	职称	副教授	学位	博士	最终学位授予国	中国大陆		
	依托单位	华南农业大学						
	二级部门	华南农业大学数学与信息（软件）学院						
项目联系人	姓名	罗浩宇		电子邮箱	haoyuluo@scau.edu.cn			
	固定电话	020-85285393		移动电话	13022092227			
	证件类型	身份证		证件号码	360302198907132536			

项目 组 情 况	主要成员（ 不包括申请者）	姓名	证件号码	职 称	所在单位全称	项目中的分工
		彭超达	445221199009201918	副教授	华南农业大学	面向分布式排队网络的并发工作流调度算法
		黄立峰	431202199002010815	讲师	华南农业大学	面向突发负载的区域化资源自动伸缩和放置
		严思源	445302199412010333	未取得	华南农业大学	算法验证、原型系统开发
		张金鹏	130427199809041934	未取得	华南农业大学	算法验证、原型系统开发
	总人数	按职称统计				
	5	高级	中级	初级	其他	
		2	1	0	2	
		按学位统计				
		博士	硕士	学士	其他	
		3	0	2	0	
在读研究生（含博士、硕士）人数		研究单位数统计（含牵头单位）				
2		1				

研 究 内 容 和 意 义	摘 要	边缘计算作为新型计算技术架构对优化物联网性能、提升终端设备的智能化实时感知能力产生重大影响。然而在智慧农业等典型物联网场景中，由于边缘节点的计算资源和能量供有限，当前主流边缘计算架构的资源管理方式面临资源利用率低、峰值可扩展性不足等问题，物联网应用的服务质量难以保障。为此，本项目将无服务器边缘计算模式引入物联网场景，研究 workflow 调度与资源优化方法，利用无服务器技术基于事件驱动、资源按需加载和自动扩容等特点实现边缘节点资源的高效利用和应用请求的快速响应。项目研究以无服务器边缘计算架构下 workflow 执行优化方法为主线，突破 FaaS 函数链的容器冷启动优化、面向突发负载的区域化资源管理和分布式排队网络中的并发 workflow 调度等关键技术，在此基础上设计无服务器 workflow 执行优化框架并开发系统原型。本项目能有效缓解部分物联网场景中有限计算资源和能量供应与物联网应用的响应能力高要求之间的矛盾，为服务质量保障提供了新思路。
	主要研究内容和要达到的目标：	<p>本项目将无服务器技术与边缘计算框架结合，利用无服务器技术的事件驱动、资源按需加载和自动扩容等特征，为物联网应用的服务质量保障提供新的解决思路。考虑到无服务器计算架构中容器实例冷启动带来的影响以及物联网场景中业务请求调用频率差异大且具有短时突发性等问题，本项目研究边缘计算框架下的无服务器 workflow 执行优化方法，通过对容器化服务全生命周期的管理和 workflow 任务的优化调度，在物联网场景中有限计算资源的前提下提高对 workflow 请求的响应能力。研究内容由以下四个部分组成：</p> <p>（1）冷启动优化的 FaaS 函数放置方法，包括无服务器 workflow 调用模式感知与预测、生命周期感知的自适应 FaaS 容器预热和重用混合策略。</p> <p>（2）面向突发负载的区域化资源自动伸缩和放置策略，包括基于统计学习的区域突发负载预测方法和基于排队理论的资源自动伸缩和放置。</p> <p>（3）面向分布式排队网络的 workflow 调度算法。</p> <p>（4）面向无服务器边缘计算环境的工作流执行优化框架设计及验证。</p> <p>本项目拟达到的研究目标包括方法创新、关键技术突破和方法有效性评估三个方面，具体如下：</p> <p>（1）边缘计算架构下无服务器 workflow 执行优化方法创新。针对传统边缘计算架构在处理物联网场景中的 workflow 应用请求时资源利用率较低和服务质量难以保障的困境，以无服务器计算模式在基础设施管理方面独特的优势为契机，以 FaaS 容器冷启动优化和并发 workflow 任务调度及资源管理为基本手段，形成无服务器边缘计算架构下 workflow 执行优化新方法，有效缓解当前物联网场景中有限的计算资源及能量供应与物联网应用的服务质量高要求之间的矛盾。</p> <p>（2）关键技术突破。突破面向 FaaS 函数链的容器冷启动优化技术，在减少冷启动所带来的响应时延的同时，提高系统资源利用率；突破面向突发负载的区域化资源管理技术，实现局部计算资源的弹性伸缩以应对物联网场景中局部短时突发的处理需求；突破分布式排队网络中的并发 workflow 调度方法，在尽可能少的资源消耗的前提下提高物联网平台的响应能力，保障服务质量。</p> <p>（3）原型系统开发与方法有效性评估验证。以上述关键技术为核心，设计面向无服务器边缘计算环境的工作流执行优化框架并开发系统原型。确立多维综合的无服务器 workflow 执行优化方法和关键技术的评估验证体系，能够客观说明本项研究的方法和关键技术对“物联网场景中 workflow 应用的服务质量保障”任务的促进作用。</p>
	主 题 词	边缘计算，无服务器计算，workflow

二、项目计划进度

序号	开始日期	结束日期	主要工作内容
1	2024-01-01	2024-12-31	研究冷启动优化的FaaS函数放置方法，具体包括： ①优化研究方案并对其进行细化； ②研究无服务器工作流调用模式感知与预测； ③研究生命周期感知的自适应FaaS容器预热与重用混合策略。
2	2025-01-01	2025-12-31	研究面向突发负载的区域化资源自动伸缩和放置策略，具体包括： ①研究基于统计学习的区域突发负载预测方法； ②研究基于排队理论的资源自动伸缩和放置； ③并发工作流调度算法预研。
3	2026-01-01	2026-12-31	研究面向分布式排队网络的并发工作流调度算法，以及面向无服务器边缘计算环境的工作流执行优化框架设计及验证，具体包括： ①研究面向分布式排队网络的并发工作流调度算法； ②工作流执行优化框架设计。 ③原型系统开发

三、经费预算

1. 省基金委经费下达总额（单位：万元）：	20.00（不超过）
2. 省基金委经费年度下达计划—本年度下达计划经费	
年度：	2024年
经费（万元）：	20.00（不超过）

四、项目预期获得的研究成果及形式

论文及专著情况	国家统计源期刊以上刊物发表论文（篇）		5		科技报告（篇）		0	
	其中被SCI/EI/ISTP收录论文数（篇）		4		培养人才（人）		3	
	专著（册）		0		引进人才（人）		0	
专利情况(项)	发明专利		实用新型专利		外观设计专利		国外专利	
	申请	授权	申请	授权	申请	授权	申请	授权
	2	0	0	0	0	0	0	0
其他								

五、前期研究基础

1. 论文收录与被引用情况统计表

论文收录情况（单位：篇）						全部论文在近5年内在SCI被引用情况	
	CSCD	CSTPCD	《SCI光盘版》	《SCI网络版》	《EI》	他人引用次数(次)	单篇被引用最高次数(次)
第一作者论文				5	5		
通讯作者论文				1	2		
其他论文				6	11		
合计	0	0	0	12	18	0	0

六、工作分工及财政经费分配

序号	单位名称	工作分工	省级财政科技资金 分配 (万元)
1	华南农业大学	无	20.00
合计			20.00

检索证明

根据委托人提供的论文材料，委托人华南农业大学 数学与信息学院、软件学院彭超达1篇论文收录情况如下表。

序号	论文名称	发表刊物及发表的年月卷期/页码等	作者排名	论文等级	作者文中单位	收录情况	影响因子	中科院大类分区
1	A two-phase framework of locating the reference point for decomposition-based constrained multi-objective evolutionary algorithms	KNOWLEDGE-BASED SYSTEMS 出版年: 2022 MAR 5 卷期: 239 文献号: 107933 文献类型: Article	第一作者	T2类	华南农业大学	SCI	IF2-year=8.139 IF5-year=8.153 (2021)	计算机科学 1区 Top期刊: 是 (2022)

说明: 论文等级和中科院大类分区按《华南农业大学学术论文评价方案(试行)》划分。

报告免责声明: 如未盖章, 报告无效



检索证明

根据委托人提供的论文材料，委托人华南农业大学 数学与信息学院、软件学院彭超达1篇论文收录情况如下表。

序号	论文名称	发表刊物及发表的年月卷期/页码等	作者排名	论文等级	作者文中单位	收录情况	影响因子	中科院大类分区
1	Constrained Multi-Objective Optimization for UAV-Enabled Mobile Edge Computing: Offloading Optimization and Path Planning	IEEE WIRELESS COMMUNICATIONS LETTERS 出版年: 2022 APR 卷期: 11 4 页码: 861-865 文献类型: Article	第一作者	A类	华南农业大学	SCI	IF2-year=5.281 IF5-year=4.874 (2021)	计算机科学 2区 Top期刊: 否 (2022)

说明: 论文等级和中科院大类分区按《华南农业大学学术论文评价方案(试行)》划分。

报告免责声明: 如未盖章, 报告无效



检索证明

根据委托人提供的论文材料，委托人华南农业大学 数学与信息学院、软件学院彭超达1篇论文收录情况如下表。

序号	论文名称	发表刊物及发表的年月卷期/页码等	作者排名	论文等级	作者文中单位	收录情况	影响因子	中科院大类分区
1	A decomposition-based constrained multi-objective evolutionary algorithm with a local infeasibility utilization mechanism for UAV path planning	APPLIED SOFT COMPUTING 出版年: 2022 MAR 卷期: 118 文献号: 108495 文献类型: Article	第一作者	A类	华南农业大学	SCI	IF2-year=8.263 IF5-year=7.595 (2021)	计算机科学 2区 Top期刊: 是 (2022)

说明: 论文等级和中科院大类分区按《华南农业大学学术论文评价方案(试行)》划分。

报告免责声明: 如未盖章, 报告无效



检索证明

根据委托人提供的论文材料，委托人华南农业大学 彭超达(学科类型:自然科学) 1 篇论文收录情况如下表。

序号	论文名称	发表刊物及发表的年月卷期/页码等	作者排名	论文等级	作者文中单位	收录情况	影响因子	中科院大类分区
1	Joint Energy and Completion Time Difference Minimization for UAV-Enabled Intelligent Transportation Systems: A Constrained Multi-Objective Optimization Approach	IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS 出版年: 2024 出版日期: OCT 卷期: 25 10 页码: 14040-14053 文献类型: Article	第一作者	A 类	华南农业大学	SCI	IF2-year=8.4 IF5-year=9.5 (2024)	计算机科学 2 区 Top 期刊: 是 OA 期刊: 否 (2025)

说明: 论文等级和中科院大类分区按《华南农业大学学术论文评价方案(试行)》划分。

报告免责声明: 如未盖章, 报告无效

检索员: 卢炳卫
华南农业大学图书馆
2025-09-16

检索证明

根据委托人提供的论文材料，委托人华南农业大学数学与信息学院 软件学院 彭超达 1 篇论文收录情况如下表。

序号	论文名称	发表刊物及发表的年月卷期/页码等	作者排名	论文等级	作者文中单位	收录情况	影响因子	中科院大类分区
1	Code Multiview Hypergraph Representation Learning for Software Defect Prediction	IEEE TRANSACTIONS ON RELIABILITY 出版年：2024 出版日期：DEC 卷期：73 4 页码：1863-1876 文献号： 文献类型：Article	通讯作者	A 类	华南农业大学 数学与信息学院	SCI	IF2-year=5.0 IF5-year=5.2 (2023)	计算机科学 2 区 Top 期刊：否 (2023)

说明：论文等级和中科院大类分区按《华南农业大学学术论文评价方案（试行）》划分。

报告免责声明:如未盖章,报告无效





Volume 239, 5 March 2022 ISSN 0950-7051

Knowledge-Based SYSTEMS





Knowledge-Based Systems

Table of contents

Volume 239,5 March 2022

Editorial Board

Article 108300

Research Articles

Subspace clustering via adaptive least square regression with smooth affinities

Lai Wei, Fanfan Zhang, Zhengwei Chen, Rigui Zhou, Changming Zhu
Article 107950

SPR: Similarity pairwise ranking for personalized recommendation

Junrui Liu, Zhen Yang, Tong Li, Di Wu, Ruiyi Wang
Article 107828

MOLS-Net: Multi-organ and lesion segmentation network based on sequence feature pyramid and attention mechanism for aortic dissection diagnosis

Qingyang Zhou, Jiaohua Qin, Xuyu Xiang, Yun Tan, Yu Ren
Article 107853

A multi-module generative adversarial network augmented with adaptive decoupling strategy for intelligent fault diagnosis of machines with small sample

Kaiyu Zhang, Qiang Chen, Jinglong Chen, Shuilong He, ... Zitong Zhou
Article 107980

Self-attention neural architecture search for semantic image segmentation

Zhenkun Fan, Guosheng Hu, Xin Sun, Gaige Wang, ... Chi Su
Article 107968

A multi-method forecasting algorithm: Linear unbiased estimation of combine forecast

Korkut Bekiroglu, Emrah Gulay, Okan Duru
Article 107990

An explainable recommendation framework based on an improved knowledge graph attention network with massive volumes of side information

Ryotaro Shimizu, Megumi Matsutani, Masayuki Goto
Article 107970

Optimal synthesis of mechanisms using repellency evolutionary algorithm

Qiujun Huang, Yicheng Yu, Kai Zhang, Shengquan Li, ... Tao Mei
Article 107928

Brightness-gradient difference feature guided shadow removal method

Yu Sang, Shihui Zhang, Huan He, Qunpeng Li, Xiaowei Zhang
Article 107986

An ensemble learning method based on deep neural network and group decision making

Xiaojun Zhou, Jingyi He, Chunhua Yang
Article 107801

Unsupervised deep hashing through learning soft pseudo label for remote sensing image retrieval

Yuxi Sun, Yunming Ye, Xutao Li, Shanshan Feng, ... Kuai Dai
Article 107807

On streaming disaster damage assessment in social sensing: A crowd-driven dynamic neural architecture searching approach

Yang Zhang, Ruohan Zong, Ziyi Kou, Lanyu Shang, Dong Wang
Article 107984

A two-phase framework of locating the reference point for decomposition-based constrained multi-objective evolutionary algorithms

Chaoda Peng, Hai-Lin Liu, Erik D. Goodman, Kay Chen Tan
Article 107933

Ontology-aware dynamically adaptable free-form natural language agent interface for querying databases

Muhammed Jassem Al-Muhammed, Deryle W. Lonsdale
Article 108012

Easy Domain Adaptation for cross-subject multi-view emotion recognition

Chuangquan Chen, Chi-Man Vong, Shitong Wang, Hongtao Wang, Miaoqi Pang
Article 107982

An elitism-based multi-objective evolutionary algorithm for min-cost network disintegration

Qian Li, Sanyang Liu, Yiguang Bai, Xingshi He, Xin-She Yang
Article 107944

Combining max–min ant system with effective local search for solving the maximum set k-covering problem

Yupeng Zhou, Xiaofan Liu, Shuli Hu, Yiyuan Wang, Minghao Yin
Article 108000

DNN compression by ADMM-based joint pruning

Geonseok Lee, Kichun Lee
Article 107988

Self-attention-based adaptive remaining useful life prediction for IGBT with Monte Carlo dropout

Dengyu Xiao, Chengjin Qin, Jianwen Ge, Pengcheng Xia, ... Chengliang Liu
Article 107902

An adaptive and general model for label noise detection using relative probabilistic density

Shuyin Xia, Longhai Huang, Guoyin Wang, Xinbo Gao, ... Zizhong Chen
Article 107907

Continuous temporal network embedding by modeling neighborhood propagation process

Yanru Zhou, Senlin Luo, Limin Pan, Lu Liu, Dandan Song
Article 107998

Self-supervised signal representation learning for machinery fault diagnosis under limited annotation data

Huan Wang, Zhiliang Liu, Yipei Ge, Dandan Peng
Article 107978

CFs-focused intelligent diagnosis scheme via alternative kernels networks with soft squeeze-and-excitation attention for fast-precise fault detection under slow & sharp speed variations

Yuanhong Chang, Jinglong Chen, Qiang Chen, Shen Liu, Zitong Zhou
Article 108026

An efficient approach for multiple probabilistic inferences with Deepwalk based Bayesian network embedding

Jiahui Wang, Kun Yue, Liang Duan, Zhiwei Qi, Shaojie Qiao
Article 107996

A referenced iterated greedy algorithm for the distributed assembly mixed no-idle permutation flowshop scheduling problem with the total tardiness criterion

Yuan-Zhen Li, Quan-Ke Pan, Rubén Ruiz, Hong-Yan Sang
Article 108036

An online portfolio strategy based on trend promote price tracing ensemble learning algorithm

Hong-Liang Dai, Chu-Xin Liang, Hong-Ming Dai, Cui-Yin Huang, Rana Muhammad Adnan
Article 107957

CJE-TIG: Zero-shot cross-lingual text-to-image generation by Corpora-based Joint Encoding

Han Zhang, Suyi Yang, Hongqing Zhu
Article 108006

Robust gravitation based adaptive k -NN graph under class-imbalanced scenarios

Yuanting Yan, Tianxiao Zhou, Zhong Zheng, Hao Ge, ... Yanping Zhang
Article 108002

Hybridised differential evolution and equilibrium optimiser with learning parameters for mechanical and aircraft wing design

Kittinan Wansasueb, Sorasak Panmanee, Natee Panagant, Nantiwat Pholdee, ... Ali Riza Yildiz
Article 107955

QLogicE: Quantum Logic Empowered Embedding for Knowledge Graph Completion

Panfeng Chen, Yisong Wang, Xiaomin Yu, Renyan Feng
Article 107963

MVE-FLK: A multi-task legal judgment prediction via multi-view encoder fusing legal keywords

Shuxin Yang, Suxin Tong, Guixiang Zhu, Jie Cao, ... Yu Wen
Article 107960

Fuzzy clustering of Acute Lymphoblastic Leukemia images assisted by Eagle strategy and morphological reconstruction

Arunita Das, Amrita Namtirtha, Animesh Dutta
Article 108008

ARFDNet: An efficient activity recognition & fall detection system using latent feature pooling

Santosh Kumar Yadav, Achleshwar Luthra, Kamlesh Tiwari, Hari Mohan Pandey, Shaik Ali Akbar
Article 107948

FKPIndexNet: An efficient learning framework for finger-knuckle-print database indexing to boost identification

Geetika Arora, Avantika Singh, Aditya Nigam, Hari Mohan Pandey, Kamlesh Tiwari
Article 108028

Privacy-preserving indoor localization based on inner product encryption in a cloud environment

Zhiheng Wang, Yanyan Xu, Yuejing Yan, Yiran Zhang, ... Xue Ouyang
Article 108005

An efficient method for mining sequential patterns with indices

Huy Minh Huynh, Loan T.T. Nguyen, Nam Ngoc Pham, Zuzana Komínková Oplatková, ... Bay Vo
Article 107946

Hybrid SOM based cross-modal retrieval exploiting Hebbian learning

Parminder Kaur, Avleen Kaur Malhi, Husanbir Singh Pannu
Article 108014

Synchronization of multiple neural networks with reaction–diffusion terms under cyber–physical attacks

Yanyi Cao, Yuting Cao
Article 107939

LSTM based decision support system for swing trading in stock market

Shouvik Banik, Nonita Sharma, Monika Mangla, Sachi Nandan Mohanty, Shitharth S.
Article 107994

Multi-local Collaborative AutoEncoder

Jielei Chu, Hongjun Wang, Jing Liu, Zhiguo Gong, Tianrui Li
Article 107844

Biased unconstrained non-negative matrix factorization for clustering

Ping Deng, Fan Zhang, Tianrui Li, Hongjun Wang, Shi-Jinn Horng
Article 108040

Hybrid recommendations and dynamic authoring for AR knowledge capture and re-use in diagnosis applications

Iñigo Fernández del Amo, John Ahmet Erkoyuncu, Maryam Farsi, Dedy Ariansyah
Article 107954

PulseNet: A multitask learning network for remote heart rate estimation

Ruo-Nan Yin, Rui-Sheng Jia, Zhe Cui, Hong-Mei Sun
Article 108048

PredRANN: The spatiotemporal attention Convolution Recurrent Neural Network for precipitation nowcasting

Chuyao Luo, Xinyue Zhao, Yuxi Sun, Xutao Li, Yunming Ye
Article 107900

Intra-domain and cross-domain transfer learning for time series data—How transferable are the features?

Erik Otović, Marko Njirjak, Dario Jozinović, Goran Mauša, ... Ivan Štajduhar
Article 107976

Personalized Knowledge Distillation for Recommender System

SeongKu Kang, Dongha Lee, Wonbin Kweon, Hwanjo Yu
Article 107958

Semi-supervised NPC segmentation with uncertainty and attention guided consistency

Lin Hu, Jiaxin Li, Xingchen Peng, Jianghong Xiao, ... Yan Wang
Article 108021

Benchmark construction and experimental evaluations for incoherent ontologies

Qiu Ji, Weizhuo Li, Shiqi Zhou, Guilin Qi, Yuanfang Li
Article 108090

Publication classification prediction via citation attention fusion based on dynamic relations

Caixia Jing, Liqing Qiu, Xiangbo Tian, Tingyu Hao
Article 108056

Association Rules Enhanced Knowledge Graph Attention Network

Zhenghao Zhang, Jianbin Huang, Qinglin Tan
Article 108038

Deep learning for missing value imputation of continuous data and the effect of data discretization

Wei-Chao Lin, Chih-Fong Tsai, Jia Rong Zhong
Article 108079

A new weakly supervised strategy for surgical tool detection

Yao Xue, Siming Liu, Yonghui Li, Ping Wang, Xueming Qian
Article 107860

Feature separation and double causal comparison loss for visible and infrared person re-identification

Qiang Liu, Xiaohai He, Mozhi Zhang, Qizhi Teng, ... Linbo Qing
Article 108042

Artificial intelligence-enabled non-intrusive vigilance assessment approach to reducing traffic controller’s human errors

Fan Li, Chun-Hsien Chen, Ching-Hung Lee, Shanshan Feng
Article 108047

Smart-VPoseNet: 3D human pose estimation models and methods based on multi-view discriminant network

Hao Wang, Minghui Sun
Article 107992

Disagreement-based class incremental random forest for sensor-based activity recognition

Chunyu Hu, Yiqiang Chen, Lisha Hu, Han Yu, Dianjie Lu
Article 108044

The automatic design of parameter adaptation techniques for differential evolution with genetic programming

Vladimir Stanovov, Shakhnaz Akhmedova, Eugene Semenko
Article 108070

Dual constraints and adversarial learning for fair recommenders

Haifeng Liu, Nan Zhao, Xiaokun Zhang, Hongfei Lin, ... Wenqi Fan
Article 108058

Special issue on Deep Learning

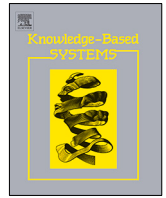
Multi-label modality enhanced attention based self-supervised deep cross-modal hashing

Xitao Zou, Song Wu, Nian Zhang, Erwin M. Bakker
Article 107927

Special issue on Explainability of Machine Learning in Methodologies and Applications

A two-stage deep transfer learning model and its application for medical image processing in Traditional Chinese Medicine

Xu Zhang, Zhikui Chen, Jing Gao, Wei Huang, ... Jianing Zhang
Article 108060



A two-phase framework of locating the reference point for decomposition-based constrained multi-objective evolutionary algorithms

Chaoda Peng^a, Hai-Lin Liu^{b,*}, Erik D. Goodman^c, Kay Chen Tan^d

^a College of Mathematics and Informatics, South China Agricultural University, Guangzhou, Guangdong Province, China

^b School of Mathematics and Statistics, Guangdong University of Technology, Guangzhou city, Guangdong Province, China

^c BEACON Center for the Study of Evolution in Action, Michigan State University, East Lansing, MI, USA

^d Department of Computing, The Hong Kong Polytechnic University, Hong Kong SAR

ARTICLE INFO

Article history:

Received 8 September 2021

Received in revised form 12 November 2021

Accepted 8 December 2021

Available online 20 December 2021

Keywords:

Multi-objective evolutionary algorithm

Referent point

Decomposition

Constraint-handling technique

ABSTRACT

Reference point is a key component in decomposition-based constrained multi-objective evolutionary algorithms (CMOEAs). A proper way of updating it requires considering constraint-handling techniques due to the existing constraints. However, it remains unexplored in this field. To remedy this issue, this paper firstly designs a set of benchmark problems with difficulties that a CMOEA must update the reference point effectively. Then a two-phase framework of locating the reference point is proposed to enhance performance of the current decomposition-based CMOEAs by evolving two populations—the main and external population. At the first phase, the external population evolves along with the main population to identify the approximate locations of the constrained and unconstrained Pareto front (PF). At the second phase, a location estimation mechanism is designed to estimate the best fit reference point between the two PFs for the main population by evolving the external population. Besides, a replacement strategy is used to drive the main population to the promising regions. Experimental studies are conducted on 26 benchmark problems, and the results highlight the effectiveness of the proposed framework.

© 2021 Elsevier B.V. All rights reserved.

1. Introduction

Constrained multi-objective optimization problems (CMOPs) have been frequently encountered in a wide range of domains, including science, economics and engineering [1–7]. Usually a CMOP has two or three objectives and a set of constraints. Problems with no less than four objectives and a set of constraints are called constrained many-objective optimization problems [8–10]. Without loss of generality, CMOPs can be formulated as:

$$\begin{aligned} \min F(\mathbf{x}) &= (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x}))^T \\ \text{s.t.} \quad & c_i(\mathbf{x}) \leq 0 \quad i = 1, 2, \dots, q \\ & h_i(\mathbf{x}) = 0 \quad i = q + 1, \dots, l \\ & \mathbf{x} = (x_1, x_2, \dots, x_n)^T \in \mathbf{D} \end{aligned} \quad (1)$$

where $\mathbf{x} = (x_1, x_2, \dots, x_n)$ is an n -dimensional decision vector, $F(\mathbf{x})$ is an m -dimensional objective function. $c_i(\mathbf{x})$ is an inequality

constraint, and $h_i(\mathbf{x})$ is an equality constraint, l is the number of the constraints, including equality and inequality constraints. In this paper, the overall constraint violation is taken as the constraint violation of a solution \mathbf{x} as in [1].

The recent decades have witnessed the fast development of decomposition-based constrained multi-objective evolutionary algorithms (CMOEAs) [1–5,7,9,11,12]. These CMOEAs decompose a CMOP into a set of subproblems with a set of weight vectors, and collaboratively solve these subproblems. The weight vectors are taken as reference lines, and a Pareto optimal solution associated with a reference line is expected to be obtained along this reference line. Reference point is one of the key components in decomposition-based CMOEAs as the weight vectors originate from it. In general, the reference point is unknown beforehand. Thus it has to be estimated. By taking a minimization problem as an example, the reference point can be specified by the following equation.

$$z_i^{\min} = \min \{f_i(\mathbf{x}) \mid \mathbf{x} \in \mathbf{D}\}, \quad i = 1, 2, \dots, m \quad (2)$$

Where m is the number of objectives, z_i^{\min} is the i th component of $\mathbf{z}^{\min} = (z_1^{\min}, z_2^{\min}, \dots, z_m^{\min})$, which is the minimum objective value of $f_i(\mathbf{x})$.

* Corresponding author.

E-mail addresses: ChaodaPeng@scau.edu.cn (C. Peng), hliu@gdut.edu.cn (H.-L. Liu), goodman@egr.msu.edu (E.D. Goodman), kctan@polyu.edu.hk (K.C. Tan).

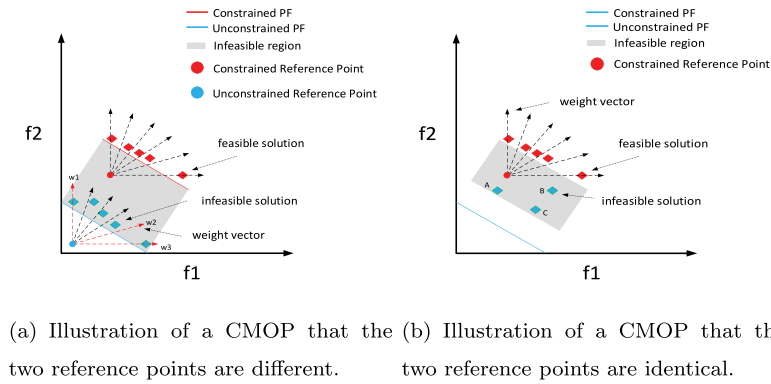


Fig. 1. Illustration of the constrained and unconstrained reference points.

The most commonly used way of updating the reference point is using the current population, including feasible and infeasible solutions. However, due to the existence of constraints, it may not work well on some CMOPs, especially for a problem with a characteristic that the extreme points of the constrained and unconstrained Pareto front (PF) are different. In this paper, the reference point updated only by feasible solutions is called constrained reference point, while if it is updated by feasible and infeasible solutions, it is called unconstrained reference point as shown in Fig. 1. Usually, the unconstrained reference point is no worse than the constrained reference point in terms of objective values. For a problem with the characteristic as shown in Fig. 1(a), some weight vectors, such as w_1 , w_2 and w_3 , do not have intersection with the constrained PF, resulting in a waste of computational resources.

This leads us to think: can a decomposition-based CMOEA only use feasible solutions to update its reference point? As presented in Fig. 1(b), the constrained reference point is taken as the reference point for a CMOEA, and all the solutions are moved to the first quadrant except three infeasible solutions A, B and C. That is, the three solutions are ignored by any of the weight vectors. Nevertheless, the utilization of them can help the algorithm go through the infeasible region. Apparently, updating the reference point only with feasible solutions may stop a CMOEA from traversing through infeasible regions.

Few efforts have been given to address this issue [13,14]. In [13], the authors proposed an estimation method for the reference point and conducted numerical experiments to verify the effect of the reference point specified in three different manners, i.e., pessimistic, optimistic and dynamic manner. The experimental results indicate that the dynamic manner outperforms the others. However, it is designed only for unconstrained multi-objective evolutionary algorithms based on decomposition. In [14], an estimation method of the ideal and nadir point was proposed to handle differently scaled objectives by normalization and to handle real-world CMOPs with unknown PFs. In this method, it firstly updates the ideal and nadir point by the feasible solutions with better non-dominated ranking, following by the infeasible solutions with smaller constraint violations till at least 95% of the population is inside the area formed by the ideal and nadir point. However, for the constraint-handling techniques with an ability of maintaining infeasibility, this method may not work well since it will waste some of the weight vectors selecting the same solutions. Actually, a proper way of updating the reference point relies on what kind of constraint handling technique that a CMOEA uses, since they have different abilities to traverse through infeasible regions. To our best knowledge, this has not been explored in the community of decomposition-based CMOEAs. Beside, several constrained multi-objective benchmark problems are proposed recently [1,5,7,8,15,16]. However, none

of them has focused on providing decomposition-based CMOEAs with difficulties of updating the reference point. Since they may occur in real-world problems, it is necessary to design such a set of artificial problems to further help researchers study the behaviors of decomposition-based CMOEAs and design appropriate ones to that domain. Table 1 lists some of the popular and recent decomposition-based CMOEAs and CMOPs.

Based on the above considerations, we design a set of test problems called RCMOPs with challenges of updating the reference point for decomposition-based CMOEAs. A two-phase framework of locating the reference point is proposed by evolving two populations, i.e., the main and external population. To improve the robustness of the proposed method with no need to consider what kind of constraint-handling technique a CMOEA uses, the external population with two phases serves this purpose. At the first phase, the external population evolves without considering constraint violations along with the main population, aiming at finding the constrained and unconstrained PF. During this phase, two reference points, i.e., the constrained and unconstrained reference point, are updated accordingly. At the second phase, the proposed method will encounter two scenarios that the two reference points are different or identical. For the first case, a location estimation mechanism is proposed to find the best estimated reference point between the constrained and unconstrained PF, since the shape of a PF is unknown a priori. As for the second case, either the constrained reference point or the unconstrained reference point can be taken as the reference point since a use of any of them does not waste any weight vector. The remained computational resources of the external population are reassigned to evolve the main population. The major contributions of this paper are summarized as follows.

1. A set of artificial benchmark problems regarding to the challenges of updating the reference point has been introduced in detail.
2. A local estimation mechanism is proposed to estimate the best fit location of the reference point for a CMOEA.
3. The proposed two-phase framework can be applied to any kind of decomposition-based CMOEA without extra parameters.
4. Systematic experiments have been conducted on 26 CMOPs to study performance of three representative decomposition-based CMOEAs.

The remainder of this paper is organized as follows. Section 2 presents a brief literature review of constrained multi-objective optimization. Section 3 introduces a set of RCMOPs and discusses their properties in detail. In Section 4, we introduce the proposed framework thoroughly. A series of experimental comparisons and further discussion are presented in Section 5. Section 6 draws the conclusion.

Table 1
Summary of the popular and recent CMOEAs and CMOPs.

Algorithm or Proposed CMOPs	Main Contributions	To consider the reference point
PPS [2]	Propose an algorithm with push and pull search for CMOPs by temporarily disregarding constraint violations.	No
M2M-DW [3]	Propose an algorithm with directed weight vectors for CMOPs by considering good infeasible individuals.	No
CCMODE [17]	Propose a cooperative differential evolution framework for CMOPs by taking advantages of the existing constraint-handling techniques.	No
C-TAEA [9]	Propose a two-archive evolutionary algorithm for CMOPs by balancing convergence, diversity and feasibility simultaneously.	No
CHT-DAE [11]	Propose a detect-and-escape strategy to guide the search to the promising regions.	No
ToP and DOC [1]	Propose a set of CMOPs, i.e., DOC, which considers decision and objective constraints, and design a framework named ToP for solving DOC.	No
M2M-IDW and DCMOPs [5]	Propose a set of CMOPs with deceptive constraints, i.e., DCMOPs, and design a cooperative evolution framework for solving DCMOPs.	No
UCMOPs [7]	Propose a set of CMOPs with unbalanced constraints.	No
MW [15]	Propose a set of CMOPs which reflect the characteristics of the real-world applications.	No
DAS-CMOPs [16]	Propose a general toolkit to construct difficulty adjustable and scalable CMOPs.	No

2. Related work

Generally speaking, the existing constraint-handling techniques in CMOEAs can be grouped into the following five categories.

The first category is based on penalty functions [18–22]. They are one of the most commonly used constraint-handling techniques. But there is a drawback that the penalty factor is a problem-dependent value. A large penalty factor may result in heavy selection pressure. The algorithms may easily get stuck in one of the disjoint feasible regions and converge to locally optimal regions. A small factor may cause a broad search region, and the algorithms may converge to an infeasible region at the end. In [21], a constraint-handling technique based on an adaptive penalty function and a distance measure was proposed to solve CMOPs. The number of the feasible solutions in this method is used to guide the search process either to find more feasible solutions or to find more feasible non-dominated solutions. This method is easy to implement and does not require any parameter tuning. However, each individual is punished with the same penalty factor in the current population, which may be too small or too large for some individuals.

In order to avoid tuning penalty factor, another type of constraint-handling techniques preferring feasible solutions over infeasible ones is carried out [1,8,23,24]. This type of methods prefer feasible solutions in the population, and tend to remove infeasible solutions. However, infeasible solutions may carry useful information about the optimal direction of objectives, which can help a CMOEA find potential optimal solutions. In [24], the authors extended or modified two constraint-handling techniques, i.e., constraint domination principle (CDP) [23] and stochastic ranking (SR) [25], yielding two CMOEAs, to solve CMOPs. The experimental results suggest that CDP outperforms SR on the test problems used in the paper.

The third category is using repair operator to drive infeasible solutions to feasible regions [6,26,27]. In [27], a simulated annealing algorithm was proposed to solve CMOPs. It can accelerate progress of the movement from an infeasible individual to a feasible one. However, this type of methods may not work well on a problem with small ratio of feasible regions, since finding a feasible solution itself is an issue.

The fourth category is based on preservation of infeasible solutions [3,7,28]. In general, infeasible individuals containing important information cannot be frequently replaced by feasible individuals, since they may benefit the evolution process. A constraint-handling technique based on directed weight vectors was proposed in [3]. In this method, a set of infeasible weight

vectors are used to maintain a number of well-distributed infeasible solutions. Furthermore, they are dynamically changed along with the evolution process, to select the infeasible solutions with better objective values and smaller constraint violations. The experimental results show its effectiveness of solving CMOPs considered in the paper.

The last category of algorithms temporarily disregards constraint violations [2,5,9,29], which has drawn great interests of researchers recently. In [2], a push and pull search framework consisting of two stages, i.e., the push and pull stage, was proposed to solve CMOPs. In the push stage, the population evolves without considering any constraint, which can help an algorithm traverse through infeasible regions quickly. Then in the pull stage, an improved epsilon constraint-handling technique is applied to pull the infeasible individuals found in the push stage to feasible regions. In [5], the authors proposed a cooperative evolutionary framework based on an improved version of directed weight vectors for handling CMOPs with deceptive constraints (DCMOPs). Due to the existence of the deceptive constraints, most of the current constraint-handling techniques do not work well on DCMOPs, since they may be misled to the less promising regions. To address this issue, an evolutionary framework with two switchable phases is proposed. The first phase uses two sub-populations—one to explore feasible regions and the other to explore the entire space, respectively. And they provide useful information about the optimal direction of objective improvement for each other. The second phase mainly aims at finding Pareto optimal solutions. The experimental results indicate the superiority of the proposed method over the compared algorithms.

3. Proposed benchmark problems

As pointed out in [30] that real-world problems cannot be taken as benchmark problems, since they may need special software or hardware. Though several new constrained multi-objective test suites are designed with different characteristics recently [1,5,7,8,15,16], none of them has provided decomposition-based CMOEAs with difficulties of updating the reference point. To further enrich this domain, seven RPCMOPs, including five 2-D problems and two 3-D problems, are proposed in this paper. They are introduced one by one in this section. Due to the limitation of the space, the formulas of the proposed test problems are put in the supplementary document.

RPCMOP1: In RPCMOP1, there is an infeasible region represented by the gray area before approaching the PF. Part of the PF

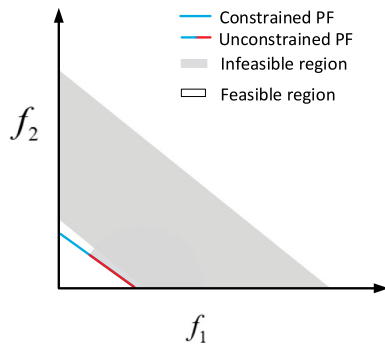


Fig. 2. Illustration of RPCMOP1.

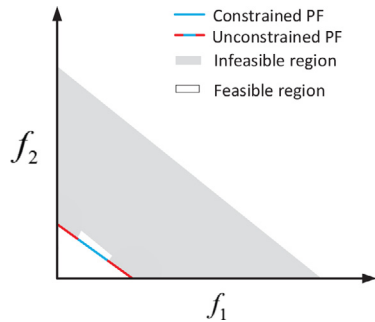


Fig. 3. Illustration of RPCMOP2.

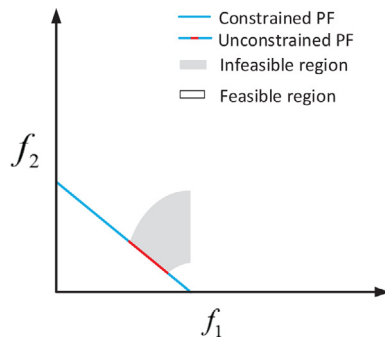


Fig. 4. Illustration of RPCMOP3.

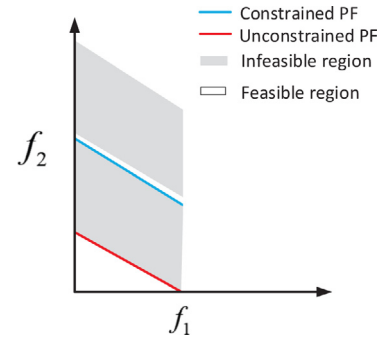


Fig. 5. Illustration of RPCMOP4.

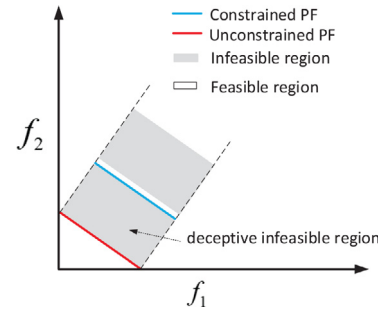


Fig. 6. Illustration of RPCMOP5.

(the red line segment) is covered by the infeasible region, therefore it becomes infeasible, as shown in Fig. 2. A decomposition-based CMOEA must first have an ability to traverse through the infeasible region. However, this may lead to an excessive exploration around the unconstrained PF, resulting in using the unconstrained reference point as the reference point for the algorithm. A special mechanism of finding the best fit reference point between the constrained and unconstrained reference point must be designed to handle the problems with such the characteristics.

RPCMOP2: As presented in Fig. 3, there is an infeasible region represented by the gray area before approaching the PF in RPCMOP2. Two parts of the PF (the red line segments) are covered by the infeasible region. A decomposition-based CMOEA may suffer from a waste of a number of weight vectors, especially for an algorithm with a strong ability to explore infeasible regions, since some of the weight vectors out of the constrained reference point select the same individuals.

RPCMOP3: In RPCMOP3, an inner part of the original PF is covered by an infeasible region. The constrained and unconstrained reference point are identical, as presented in Fig. 4. It provides

a decomposition-based CMOEA with a difficulty that only using feasible solutions in current population to update the reference point may not help the algorithm get through the infeasible barrier, especially for a constraint-handling technique based on the preference on feasible solutions over infeasible ones.

RPCMOP4: In RPCMOP4, the original PF (the red line segment) totally becomes infeasible, and the constrained PF (the blue line segment) lies in between the feasible region and infeasible region as shown in Fig. 5. Two reference points, i.e., the constrained and unconstrained reference point, are different on axis f_2 . A decomposition-based CMOEA may perform poorly on this test problem due to a waste of computational resources with a constraint-handling technique based on preserving infeasible solutions, since usually the solutions found in the population are used to update the reference point, and infeasible solutions may have better objective values than the feasible ones, resulting in taking the unconstrained reference point as the reference point for the algorithm.

RPCMOP5: As shown in Fig. 6, the original PF (the red line segment) is rendered infeasible, and the constrained PF (the blue line segment) lies in between the feasible region and deceptive infeasible region [5]. It provides a decomposition-based CMOEA with a difficulty that the algorithm must handle constraints properly to locate the reference point, otherwise it may be driven away from the feasible region by the deceptive constraints, resulting in using the unconstrained reference point as the reference point.

RPCMOP6: RPCMOP6 is a 3-D test problem, in which part of the original PF becomes infeasible (the red area) as shown in Fig. 7. RPCMOP6 is an extension of RPCMOP1, providing decomposition-based CMOEAs with the same difficulties that RPCMOP1 has.

RPCMOP7: As shown in Fig. 8, RPCMOP7 is a 3-D test problem, and the original PF fully becomes infeasible, which is represented by the red area. The constrained PF lies in between the feasible and deceptive infeasible area. RPCMOP7 is an extension of RPCMOP5, providing decomposition-based CMOEAs with the same difficulties that RPCMOP5 has.

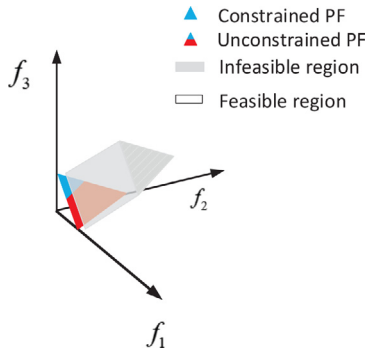


Fig. 7. Illustration of RPCMOP6.

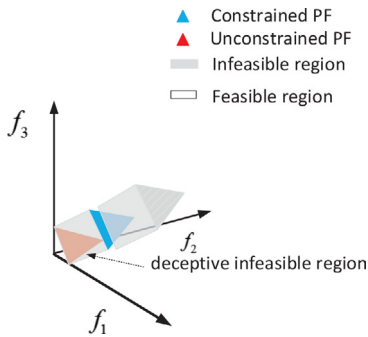


Fig. 8. Illustration of RPCMOP7.

4. Proposed framework

In this section, the details of the proposed framework are presented in the following subsections.

4.1. Pseudo-code of the proposed framework

This paper proposes a two-phase framework of locating the reference point for decomposition-based CMOEAs. Its pseudo-code is given in Algorithm 1. To better understand the working procedure of the proposed framework, we also present the flowchart as shown in Fig. 9.

The proposed framework includes two phases. The first phase aims to find the appropriate location of the unconstrained and constrained reference point. At a generation t during this phase, we have the main and external population P_t and E_t both with a size N , and a CMOEA denoted as **MOEA/D-CHT**. Noted that **MOEA/D** denotes a decomposition-based multi-objective evolutionary algorithm, and **CHT** denotes a constraint-handling technique in the pseudo-code (see line 6, 11 and 29 in Algorithm 1). Two populations are evolved by using the genetic operations in **MOEA/D-CHT** when a decomposition-based CMOEA is specified. Two reference points—the unconstrained and constrained reference point are updated according to Section 4.2. Besides, the feasible individuals found in the external population are stored into the archive A . If the size of A exceeds N , the best N feasible individuals regarding to the achievement scalarizing function (ASF) [31] are chosen into A .

$$ASF(\mathbf{x}|\mathbf{w}) = \max_{1 \leq i \leq m} \left(\frac{f_i(\mathbf{x}) - z_i^{\min}}{w_i} \right) \quad (3)$$

where \mathbf{w} is a weight vector, $\mathbf{z}^{\min} = (z_1^{\min}, z_2^{\min}, \dots, z_m^{\min})$ is the reference point given by Eq. (4). The external population E_t evolves without taking constraints into account (see line 11–15)

with a purpose to maximumly explore the whole region. When the whole population changes slightly at a generation, which can be regarded that it does not evolve any more. That is, the exploration of the two reference points is done at this moment. The proposed framework will switch to the second phase (see line 35–37).

At the second phase, the proposed framework firstly checks whether the unconstrained and constrained reference point are different or not (see line 18). It will undergo two scenarios as follows:

1. If they are different, then a location estimation mechanism is designed to find the best estimated reference point between the unconstrained and constrained reference point according to Algorithm 2 (see line 21 in Algorithm 1). Due to the fact that the external population evolves without considering feasibility at the first phase, it may end up with getting stuck in the infeasible regions. To draw the external population back to feasible regions, a replacement mechanism is designed to use the feasible solutions in A to replace the solutions in E_{t+1} (see line 16). Each of the N weight vectors is associated with the closest individual in A , and this individual is used to compare with the one associated with the same weight vector in E_{t+1} based on CDP, better one survives.
2. If they are identical, then there is no need to estimate the reference point between the two reference points. In this case, the rest of computational resources of the external population is reassigned to the main population (see line 23).

Remark. The idea of the two-phase strategy has also been exploited in [1,2,5,9]. However, these papers used the two-phase strategy to cope with constraints, by which the algorithms can traverse through infeasible regions quickly. In contrast, this paper uses this idea to estimate the locations of the constrained and unconstrained referent point, and find the best location of the reference point for a decomposition-based multi-objective evolutionary algorithm. Therefore, the idea of this paper is quite different from these papers.

4.2. The local estimation mechanism

The proposed framework updates the reference point as follows:

$$z_i^{\min} = CRP_i - \lambda (CRP_i - URP_i), i = 1, 2, \dots, m \quad (4)$$

Where λ is a value in $[0, 1]$. When $\lambda = 0$, the constrained reference point is taken as the reference point, while $\lambda = 1$, the unconstrained reference point is taken as the reference point. The pseudo-code of updating the reference point is given in Algorithm 2.

Each newly generated individual is used to update the constrained and unconstrained reference point. The constrained reference point is only updated by feasible individuals. While the unconstrained reference point is updated both by feasible and infeasible individuals. Finally the reference point for a given decomposition-based CMOEA is determined by Eq. (4).

The proposed framework consists of two phases. At the first phase, the unconstrained reference point is used as the reference point (i.e., λ is set to 1). As discussed in Section 1, the constrained reference point cannot be used as the reference point at the early stage of the evolution process since it may stop a decomposition-based CMOEA traversing through infeasible regions.

At the second phase, when the unconstrained and constrained reference point are identical, either one of them can be used

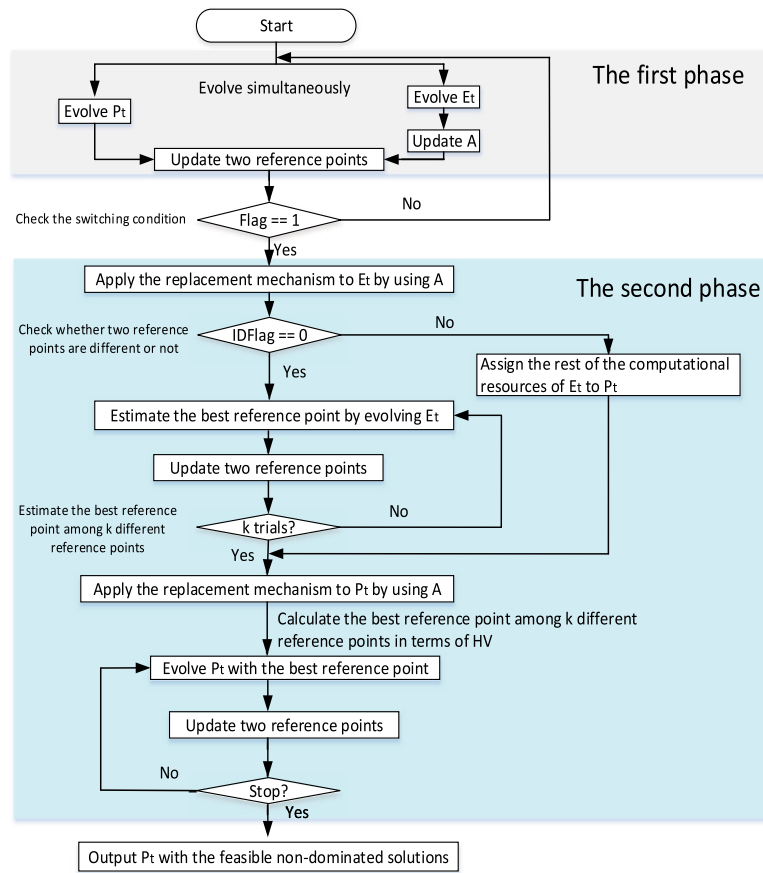


Fig. 9. The flowchart of the proposed framework.

as the reference point since they are the same according to Eq. (4). However, when they are different, a location mechanism is proposed to find the most suitable location at the line segment formed by these two reference points (see line 4–18 in Algorithm 2). k different locations between the two reference points are investigated. Each of the k locations is corresponding to one λ in λ_s , where λ_s is given in Eq. (5). The rest of the computational resources of the external population is divided into k equal parts, in which each part is assigned to evolve E_t under the same experimental conditions except the values of λ . Then the hyper-volume [32] (HV) values of these k final populations are calculated. The reference point for calculating HV value is set to 1.1 times of the average maximum values of each objective in k obtained populations. The value of λ in λ_s regarding to the best HV value is used in Eq. (4) at the second phase.

$$\lambda_s = \left\{ \frac{k-i-1}{k-1} \mid i = 0, 1, 2, \dots, k-1, k \geq 2 \right\} \quad (5)$$

4.3. The switching condition

Algorithm 3 presents the idea of the switching condition, in which the switching condition is triggered when the population changes slightly at a small range. To serve this purpose, a set of well-distributed trial weight vectors with a size of $\lfloor \alpha N \rfloor$ are used, which are selected from N well-distributed weight vectors by using the Max-Min method [3].

The switching condition is checked in every 20 generations according to Algorithm 1. The set *old_TA* stores the historical individuals selected by the trial weight vectors, and another set *new_TA* stores the individuals selected from E_{t+1} with the same trial weight vectors in every 20 generations. Then the inverted

generation distance metric (IGD) [33] is applied to compare the closeness between these two sets in terms of the convergence and diversity (see line 3–12). It is noted that a normalization scheme is applied to these two sets by shrinking the objective values into $[0, 1]$. If the IGD value is smaller than a given parameter r , then the population is considered that it does not evolve any more, therefore the switching condition is triggered.

4.4. Computational complexity of the proposed framework

The computational complexity of the proposed framework is analyzed as below. The proposed framework mainly includes evolving P_t and E_t , updating A and two reference points, the replacement mechanism, the switch condition and the location estimation mechanism. Therefore, the computational complexity of the proposed algorithm can be determined by the seven components. Suppose the computational complexity of an embedded algorithm is $O(\Delta)$. At the first phase, the proposed framework evolves two populations P_t and E_t that consumes $O(\Delta)$, updates the archive A that consumes $O(mN^2)$ and two reference points that consumes $O(mN)$. Thus the computational complexity of the proposed algorithm at the first phase is $\max(O(\Delta), O(mN^2))$. The second phase includes the local estimation mechanism, the switching condition, updating two reference points, the replacement mechanism and evolving P_t . For the local estimation mechanism, it repeats the same procedure to find the best fit reference point by evolving E_t with K times. It is noted that the proposed framework preserves the fair comparison with other algorithms, the generations for finding the best fit reference point are subtracted from the total generations. Hence, it consumes $O(\Delta)$ for a generation. The switching condition consumes $O(m\beta^2)$ mainly

Algorithm 1: The Proposed Framework

```

1: Initialize two populations  $P_0$  and  $E_0$ , an archive  $A$ , a set of trial weight vectors  $TW$ , a temporary archive  $TA$ , the maximum generation number  $max\_gen$ , and the
   constrained and unconstrained reference point- $CRP$  and  $URP$ .
2: Initialize a decomposition-based CMOEA as MOEA/D-CHT.
3: Set the switching condition  $Flag = 0$ , the complementary condition  $CFlag = 0$ , the stopping condition  $stop\_gen = \frac{max\_gen}{2}$ ,  $\lambda = 1$  and  $t = 1$ .
4: while  $t < stop\_gen$  do
5:   if  $Flag == 0$  then
6:     function MOEA/D-CHT ( $P_t$ )
7:       Generate a new solution  $y$  by genetic operators.
8:       updateRefPoint ( $y$ ,  $CRP$ ,  $URP$ ,  $\lambda$ ).
9:       Select the best  $N$  individuals into  $P_{t+1}$ .
10:    end function
11:    function MOEA/D ( $E_t$ ) %Evolving  $E_t$  without considering constraints..
12:      Generate a new solution  $y$  by genetic operators.
13:      updateRefPoint ( $y$ ,  $CRP$ ,  $URP$ ,  $\lambda$ ).
14:      Select the best  $N$  individuals into  $E_{t+1}$ .
15:    end function
16:    Update  $A$  with the feasible solutions found in  $E_{t+1}$ .
17:  else
18:    Set  $IDFlag = 0$  if  $CRP$  and  $URP$  are different.
19:    if  $IDFlag == 0$  then
20:      Applied the replacement mechanism to update  $E_t$  by using  $A$ .
21:       $\lambda^{best}$ ,  $A = \text{findTheBestRefPoint}(E_t, CRP, URP, A)$ .
22:    else
23:      Assign the rest of the computational resources of  $E_t$  to  $P_t$ :  $stop\_gen = max\_gen - t$ .
24:    end if
25:     $CFlag = 1$ .
26:    Applied the replacement mechanism to update  $P_t$  by using  $A$ .
27:  end if
28:  if  $CFlag == 1$  then
29:    function MOEA/D-CHT ( $P_t$ ) %Evolving  $P_t$  with the best reference point..
30:      Generate a new solution  $y$  by genetic operators.
31:      updateRefPoint ( $y$ ,  $CRP$ ,  $URP$ ,  $\lambda^{best}$ ).
32:      Select the best  $N$  individuals into  $P_{t+1}$ .
33:    end function
34:  end if
35:  if  $mod(t, 20) == 0$  &  $Flag == 0$  then
36:     $Flag, TA = \text{checkSwitchingCondition}(E_{t+1}, TA, TW)$ .
37:  end if
38:   $t = t + 1$ .
39: end while
40: Output the feasible non-dominated solutions in  $P_t$ .

```

Algorithm 2: findTheBestRefPoint(CDP, E_t, CRP, URP, A)

```

1:  $RP = \emptyset$ .
2:  $r\_gen = \frac{stop\_gen-t}{k} + t$ .
3: Initialize  $\lambda_s$  by using Eq. (5).
4: for  $i = 1 : k$  do
5:    $TP = E_t$ ;
6:   Set  $c = t$ .
7:   while  $c < r\_gen$  do
8:     function MOEA/D-CDP ( $TP_c$ )
9:       Generate a new solution  $y$  by genetic operators.
10:      updateRefPoint ( $y$ ,  $CRP$ ,  $URP$ ,  $\lambda_{s_i}$ ).
11:      Select the best  $N$  individuals into  $TP_{c+1}$ .
12:    end function
13:     $c = c + 1$ .
14:  end while
15:   $RP_i = TP_c$ .
16:  Update  $A$  with the feasible solutions found in  $TP_c$ .
17: end for
18: Calculate the HV values of  $k$  populations in  $RP$ , and return the  $\lambda$  regarding to
   the best HV value.
19: Output the value  $\lambda$  related to the best estimated reference point and  $A$ .

```

Algorithm 3: checkSwitchingCondition(E_{t+1}, TA, TW, r)

```

1:  $old\_TA = TA$ .
2: Set  $new\_TA = \emptyset$ .
3: for  $i = 1 : \lfloor aN \rfloor$  do
4:   Obtain the individual  $y$  in terms of the best ASF value  $ASF1$  in  $E_{t+1}$  with the
      $i$ -th trial weight vector according to Eq. (3).
5:    $new\_TA = new\_TA \cup \{y\}$ .
6:   Calculate the ASF value  $ASF2$  of the  $i$ -th individual in  $TA$  with the  $i$ -th trial
     weight vector according to Eq. (3).
7:   if  $ASF1 < ASF2$  then
8:     Update the  $i$ -th individual in  $TA$  with  $y$ .
9:   end if
10: end for
11: Applied normalization scheme to  $old\_TA$  and  $new\_TA$ .
12: Calculate the IGD value:  $IGD\_V = IGD(Old\_TA, new\_TA)$ .
13: if  $IGD\_V < r$  then
14:    $Flag = 1$ .
15: else
16:    $Flag = 0$ .
17: end if
18: Output the switching condition  $Flag$  and the temporary archive  $TA$ .

```

for calculating IGD value. Having $\beta = \lfloor aN \rfloor$, which is at the scale of N , the computational complexity of the switching condition is $O(mN^2)$. And the replacement mechanism consumes $O(mN^2)$. While the rest of the components has been analyzed above. The computational complexity at the second phase is $\max(O(\Delta), O(mN^2))$. As a result, the total computational complexity of the propose algorithm is $\max(O(\Delta), O(mN^2))$.

5. Experimental studies

In this section, a series of experiments on three sets of constrained multi-objective benchmark problems, i.e., the proposed

test problems, CF test problems [17] and CTP test problems [34], are conducted to study performance of the proposed framework. Three CMOEAs, i.e., CMOEA/D-CDP [24], MOEA/D-PPS [2] and M2M-DW [3], with three ways of updating the reference point—only using feasible solutions, both using feasible and infeasible solutions and the proposed method (denoted as OUFs, BUFIS and TPM), are used in our experiments. Besides, the IGD and HV metric are used to evaluate performance of the three CMOEAs. 100 and 150 feasible solutions are chosen from the final obtained population by using the Max-Min method to calculate the IGD and HV values of 2D and 3D problems respectively. The reference point is set to 1.1 times of the extremely largest values of the PF.

Generally speaking, a smaller IGD value implies the population found by an algorithm is closer to the PF, while a larger HV value implies that an algorithm achieves better performance in terms of convergence and diversity.

Due to the limitation of the space, some of the experiments regarding to the computational time of the proposed framework, strength and weakness of the proposed framework, performance of the proposed framework on a real-world problem named OSY [35] are presented in the supplementary document.

5.1. Parameter settings

The parameters related to the proposed framework and the other two CMOEAs are set in this subsection. The proposed framework includes three parameters, i.e., a , r and k . Therein, a indicates the number of the representative points that are used to measure the similarity between two populations. A smaller value of a indicates that two populations are easier to get similar. r controls the similarity of two populations. A smaller value of r indicates that two populations are more similar. That is, the proposed framework spends more computational resources evolving the external population. While k defines the number of λ (or the size of λs). When the parameter k is specified, λs can be given by Eq. (5). A larger value of k indicates that more locations between the unconstrained and constrained reference point are investigated to find the most suitable reference point for a decomposition-based CMOEA. Their details are discussed in Sections 5.4–5.6, respectively.

1. Each of the CMOEAs in this paper runs 30 independent times, and stops after 3×10^5 and 5×10^5 function evaluations (FEs) for 2D and 3D problems, respectively.
2. The population size N is set to 100 and 300 for problems with two and three objectives, respectively.
3. The parameter a is set to $\lfloor 15\% \rfloor$.
4. The parameter r is set to 0.01.
5. The parameter k is set to 5, and λs is set to $\{0, 0.25, 0.5, 0.75, 1\}$ by using Eq. (5).
6. The number of the feasible and infeasible weight vectors in M2M-DW is set to (70, 30) and (180, 120) for 2D and 3D problems, respectively.
7. The rest of parameter settings of the three CMOEAs remain the same as in the original papers.

5.2. Comparisons among three variants of updating the reference point

Tables 2–4 display experimental results of three CMOEAs with the three variants of updating the reference point over 30 independent runs regarding to the mean value and standard deviation of IGD and HV on RPCMOPs respectively. It is worth noting that if the number of the runs that the final obtained population of a CMOP contains no feasible solutions is more than half of the 30 runs, the IGD and HV values of the CMOP will be assigned NAN. While if it is less than half of the 30 runs, the runs without finding any feasible solution are disregarded when calculating IGD and HV values. Besides, the overall ranking (OR) is used to verify the overall performance of CMOEAs on the test instances. A CMOEA is ranked 1 on a test instance when it achieves the best result, and is ranked 2 when it achieves the second best result, and so on. Afterward, the OR of a CMOEA is calculated by summing up the rankings of all the test instances. The smaller value indicates an algorithm has better overall performance among the compared algorithms. Furthermore, Wilcoxon's rank sum test [36] at 0.05 significance level is performed between the proposed method and the other two methods of updating the reference point under the

three CMOEAs. The plus sign implies that the distribution of the performance metric values achieved by the proposed method is significantly better than the compared method of updating the reference point. The asymp sign implies that the distribution of the performance metric values obtained by the proposed method and the compared method has no difference. The minus sign implies that the distribution of the performance metric values achieved by the proposed method is worse than the compared method. Better results are highlighted in light gray shade.

The advantages of the proposed framework can be intuitively observed from Tables 2–4. The proposed framework consistently achieves better results than the other two methods, i.e., OUFs and BUFIS, under the three different CMOEAs on each RPCOMP in terms of the mean values of IGD and HV. The main reason is given as follows. When only using feasible solutions to update the reference point, the algorithms may be stopped from traversing through infeasible regions. As shown in Section 3, each RPCOMP except RPCOMP3 has an infeasible barrier before approaching the PF, therefore, the three CMOEAs with the way of updating reference point only by feasible solutions, i.e., MOEA/D-PPS-OUFs, CMOEA/D-CDP-OUFs and M2M-DW-OUFs, fail to find the PFs on these RPCOMP. From these three tables, we can also see that the method based on updating reference point both with feasible and infeasible solutions seems to achieve more comparable results than OUFs. However, they waste some of the computational resources since parts of the PF on most of the RPCOMP become infeasible, and the weight vectors on these areas are wasted. The proposed method consisting of two phases can relatively estimate the right location of the reference point for a decomposition-based CMOEA. At the first phase, the proposed method aims at finding the locations of the unconstrained and constrained reference point. During this phase, the unconstrained reference point is used as the reference point. Then the most suitable location of the reference point for a decomposition-based CMOEA is specified between the unconstrained and constrained reference point at the second phase.

Fig. 10 presents non-dominated fronts obtained at the median run based on IGD values of the three different methods of updating the reference point by M2M-DW. Due to the limitation of the space, figures obtained by MOEA/D-PPS and CMOEA/D-CDP can be found in the supplementary document.

The advantages of the proposed method are obvious as shown in Fig. 10. M2M-DW with the proposed method of updating the reference point, i.e., M2M-DW-TPM (the first column), obtains a set of better-distributed feasible non-dominated solutions compared with the other two variants. A closer look at this figure is that M2M-DW-OUFs updates the reference point only with the feasible solutions, and prevents itself traversing through infeasible regions, resulting in a failure on six RCMOPs. While M2M-DW-BUFIS updates the reference point with infeasible solutions, some weight vectors which have no intersections with the constrained PFs select the feasible solutions around the boundary between the feasible and infeasible regions with a waste of computational resources.

5.3. Experimental results on CF and CTP test problems

Table 5 displays the experimental results of M2M-DW with the three variants of updating the reference point over 30 independent runs regarding to the average value and standard deviation of IGD and HV on CFs and CTPs. The overall ranking (OR) is used to verify the overall performance of a CMOEA on the test instances. A CMOEA achieves the best result on a test instance is ranked 1, the second best result is ranked 2, and so on. Afterward, the OR of a CMOEA is calculated by summing up the rankings of all the test instances. The smaller value indicates a better result

Table 2

Experimental results of MOEA/D-PPS over 30 independent runs in terms of the mean value and standard deviation of IGD and HV on RPCMOPs.

Problem	IGD			HV		
	TPM	OUFs	BUFIS	TPM	OUFs	BUFIS
RPCMOP1	2.6144e-03(2.3194e-04)	2.8289e+00(1.5868e-04)+	4.4490e-03(6.4465e-05)+	2.0242e-01(2.0216e-04)	0.0000e+0(0.0000e+0)+	2.0119e-01(1.0546e-04)+
RPCMOP2	2.1320e-03(1.9304e-04)	2.8288e+00(1.1765e-04)+	3.1812e-03(6.3402e-05)+	1.2032e-01(8.8139e-05)	0.0000e+0(0.0000e+0)+	1.1980e-01(7.1221e-05)+
RPCMOP3	4.5756e-03(3.7371e-05)	4.8496e-02(1.3371e-01)≈	4.5768e-03(5.1467e-05)≈	6.5622e-01(1.4835e-04)	6.2365e-01(9.8245e-02)≈	6.5619e-01(1.8655e-04)≈
RPCMOP4	8.7284e-03(3.9967e-03)	5.1537e+00(1.3259e+00)+	1.5411e-02(8.2974e-04)+	9.1534e-01(6.0354e-03)	4.0435e-02(1.7371e-01)+	9.0058e-01(3.8932e-03)+
RPCMOP5	4.9200e-03(6.3801e-04)	3.7479e+00(3.7947e-05)+	NAN	8.9607e-01(6.6335e-04)	0.0000e+0(0.0000e+0)+	NAN
RPCMOP6	1.5480e-02(1.4388e-03)	5.1979e+00(4.3462e-04)+	2.5691e-02(5.0323e-04)+	1.0636e-01(3.2725e-04)	0.0000e+0(0.0000e+0)+	1.0441e-01(3.7962e-04)+
RPCMOP7	2.9290e-02(1.5718e-03)	3.0618e+00(1.1274e-04)+	NAN	1.3430e+00(1.7055e-03)	0.0000e+0(0.0000e+0)+	NAN
OR	7	20	15	7	20	15

Table 3

Experimental results of CMOEA/D-CDP over 30 independent runs in terms of the mean value and standard deviation of IGD and HV on RPCMOPs.

Problem	IGD			HV		
	TPM	OUFs	BUFIS	TPM	OUFs	BUFIS
RPCMOP1	2.4333e-03(1.7970e-04)	2.8288e+00(1.3319e-04)+	2.8300e+00(7.6979e-04)+	2.0265e-01(1.5169e-04)	0.0000e+0(0.0000e+0)+	0.0000e+0(0.0000e+0)+
RPCMOP2	2.1592e-03(2.3694e-04)	2.8287e+00(7.9596e-05)+	2.8302e+00(1.0832e-03)+	1.2035e-01(1.3958e-04)	0.0000e+0(0.0000e+0)+	0.0000e+0(0.0000e+0)+
RPCMOP3	4.6083e-03(8.4216e-05)	2.4015e-01(2.2053e-01)+	4.6498e-03(1.4188e-04)≈	6.5603e-01(3.2676e-04)	4.8030e-01(1.5980e-01)+	6.5593e-01(5.4614e-04)≈
RPCMOP4	7.2286e-03(2.7467e-03)	5.1345e+00(1.3939e+00)+	4.9607e+00(1.6228e+00)+	9.1757e-01(4.2266e-03)	6.1076e-02(2.3243e-01)+	7.1626e-02(2.3459e-01)+
RPCMOP5	5.0259e-03(6.9180e-04)	3.7479e+00(3.8860e-05)+	3.7488e+00(3.7914e-04)+	8.9616e-01(8.1887e-04)	0.0000e+0(0.0000e+0)+	0.0000e+0(0.0000e+0)+
RPCMOP6	1.4598e-02(8.3405e-04)	5.1977e+00(3.9380e-04)+	5.1983e+00(1.2166e-03)+	1.0711e-01(2.2347e-04)	0.0000e+0(0.0000e+0)+	0.0000e+0(0.0000e+0)+
RPCMOP7	2.8058e-02(1.6255e-03)	3.0617e+00(1.2518e-04)+	3.0622e+00(5.1018e-04)+	1.3453e+00(1.5251e-03)	0.0000e+0(0.0000e+0)+	0.0000e+0(0.0000e+0)+
OR	7	16	19	7	16	14

Table 4

Experimental results of M2M-DW over 30 independent runs in terms of the mean value and standard deviation of IGD and HV on RPCMOPs.

Problem	IGD			HV		
	TPM	OUFs	BUFIS	TPM	OUFs	BUFIS
RPCMOP1	2.4950e-03(1.7856e-04)	2.8286e+00(4.6879e-05)+	4.7273e-03(2.7477e-05)+	2.0285e-01(1.3799e-04)	0.0000e+0(0.0000e+0)+	2.0121e-01(5.2600e-05)+
RPCMOP2	1.8730e-03(2.5975e-04)	2.8285e+00(4.5751e-05)+	3.8331e-03(2.3356e-05)+	1.2062e-01(1.3838e-04)	0.0000e+0(0.0000e+0)+	1.1964e-01(2.9521e-05)+
RPCMOP3	5.0901e-03(1.7336e-05)	6.3450e-02(1.5127e-01)≈	5.0916e-03(3.6103e-05)≈	6.5607e-01(1.2221e-04)	6.1308e-01(1.1120e-01)≈	6.5597e-01(6.7522e-04)≈
RPCMOP4	6.7581e-03(2.4662e-03)	4.9436e+00(1.6741e+00)+	1.9820e-02(3.3415e-03)+	9.1960e-01(3.5734e-03)	9.2015e-02(2.8077e-01)+	8.9892e-01(5.0124e-03)+
RPCMOP5	4.4537e-03(3.4115e-05)	3.4982e+00(9.4973e-01)+	9.9541e-03(1.3040e-04)+	8.9697e-01(1.0263e-04)	5.9809e-02(2.2761e-01)+	8.8931e-01(1.7583e-04)+
RPCMOP6	1.3652e-02(8.7474e-04)	5.1970e+00(1.5982e-04)+	2.4637e-02(3.1534e-04)+	1.0792e-01(2.0664e-04)	0.0000e+0(0.0000e+0)+	1.0506e-01(1.8200e-04)+
RPCMOP7	2.9285e-02(8.1244e-04)	3.0614e+00(7.3466e-05)+	4.4504e-02(9.9695e-04)+	1.3461e+00(7.7556e-04)	0.0000e+0(0.0000e+0)+	1.3294e+00(1.3455e-03)+
OR	7	21	14	7	21	14

among the compared algorithms. Besides, Wilcoxon's rank sum test at 0.05 significance level is performed between the proposed method and the other two methods of updating the reference point under the three different CMOEAs. The plus sign implies that the distribution of the performance metric values achieved by the proposed method is significantly better than the compared method of updating the reference point. The asymp sign implies that the distribution of the performance metric values obtained by the proposed method and the compared method has no difference. The minus sign implies that the distribution of the performance metric values achieved by the proposed method is worse than the compared method. Better results are marked in light gray shade. Due to the limitation of the space, experimental results obtained by MOEA/D-PPS and CMOEA/D-CDP are put in the supplementary document.

The superiority of the proposed method can be further observed from Table 5. As for M2M-DW, M2M-DW-TPM works better than M2M-DW-OUFs and M2M-DW-BUFIS on CF1, CF3, CF5-CF8, CF10, CTP1, CTP2, CTP6-CTP8 and CF2, CF3, CF5, CF7, CF8, CF10, CTP1, CTP2, CTP5-CTP8 in terms of mean value of IGD and HV values respectively. Besides, we can observe that M2M-DW-TPM achieves the best overall performance on these two test problems in terms of OR.

5.4. Investigation of the parameter a

We investigate sensitivity of the parameter a in this subsection. RPCMOP2 and RPCMOP5 are used since they provide a CMOEA with different difficulties that are usually used to test performance of a CMOEA. The proposed framework with M2M-DW and five values of parameter a (5%, 15%, 25%, 35% and 45%) are adopted in this numerical experiment, while the rest of the parameters remain the same as in Section 5.1. The result is presented in Table 7.

From Table 6, we can see that M2M-DW-TPM is not sensitive to the parameter a based on the obtained mean values and deviations of IGD and HV on RPCMOP2 and RPCMOP5.

5.5. Investigation of the parameter r

We investigate sensitivity of the parameter r on RPCMOP2 and RPCMOP5 in this subsection. The proposed framework with M2M-DW and five values of parameter r (0.001, 0.01, 0.1, 0.5 and 1) are adopted in this numerical experiment, while the rest of the parameters remain the same as in Section 5.1. The result is presented in Table 7.

From Table 7, we can see that M2M-DW-TPM is not sensitive to the parameter r based on the obtained mean values and deviations of IGD and HV on RPCMOP2 and RPCMOP5.

5.6. Investigation of the parameter k

We investigate sensitivity of the parameter k on two test problems, i.e., RPCMOP2 and RPCMOP5, in this subsection. The proposed framework with M2M-DW and five values of parameter k (2, 4, 5, 6 and 8) are adopted in this numerical experiment, while the rest of the parameters remain the same as in Section 5.1. The result is presented in Table 7.

From Table 8, we can see that M2M-DW-TPM is not sensitive to the parameter k based on the obtained mean values and deviations of IGD and HV on RPCMOP2 and RPCMOP5.

6. Conclusion

We proposed a two-phase framework of locating the reference point for decomposition-based CMOEAs in this paper. A set of RPCMOPs which provides decomposition-based CMOEAs with difficulties of updating the reference point were introduced in Section 3.

In Section 4, we introduced the proposed framework in detail. To eliminate the difference brought by different constraint-handling techniques, an external population was used to help find the most suitable reference point for a decomposition-based CMOEA. The proposed framework consisted of two phases. At

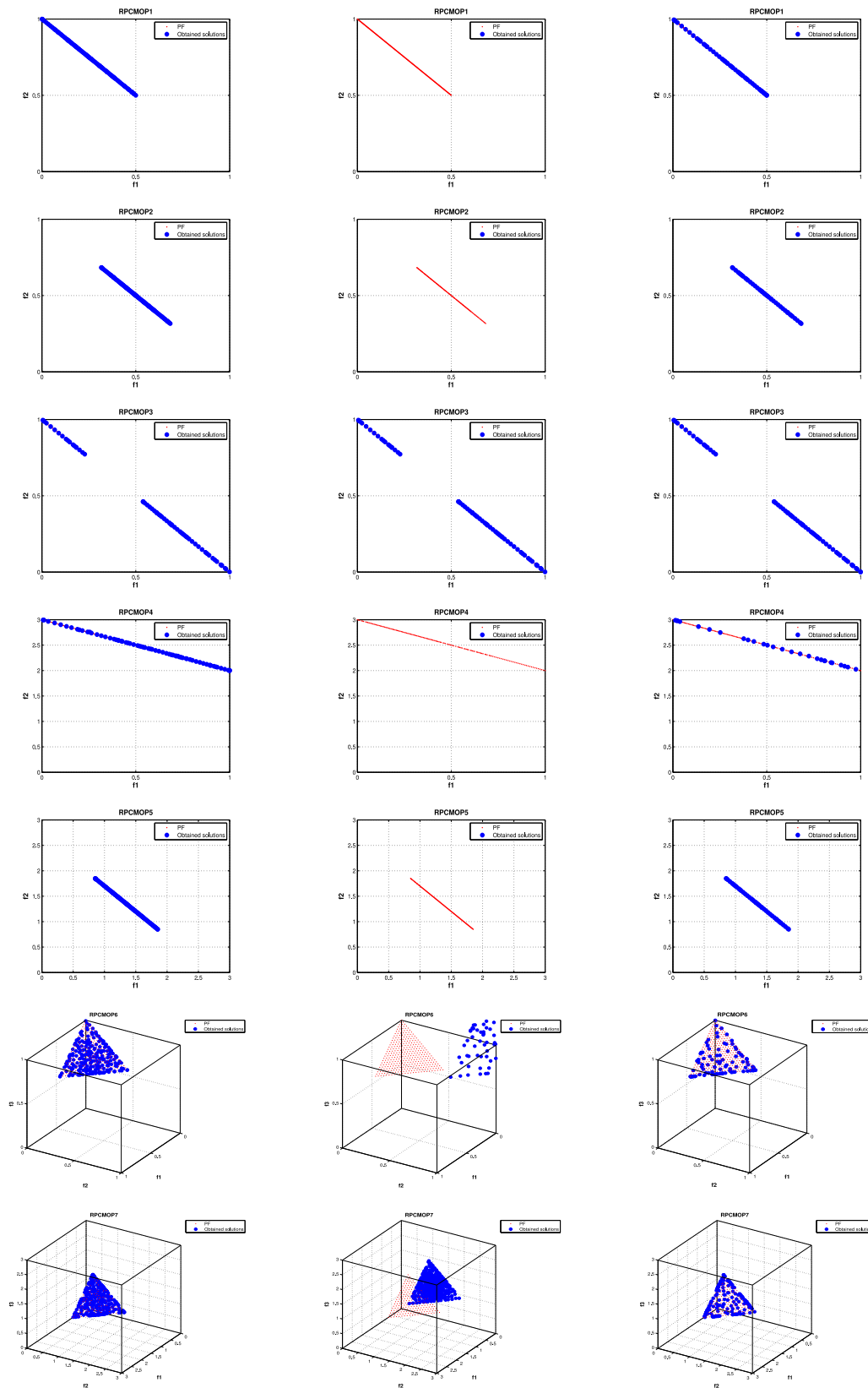


Fig. 10. Non-dominated fronts obtained at the median run based on IGD values of M2M-DW-TPM (the first column), M2M-DW-OUFS (the second column) and M2M-DW-BUFIS (the last column).

the first phase, the external population evolved without considering constraint violations along with the main population, aiming at finding the constrained and unconstrained PF. During

this phase, two reference points, i.e., the constrained and unconstrained reference point, were updated according to Section 4.2. At the second phase, the proposed framework firstly checked

Table 5

Experimental results of M2M-DW over 30 independent runs in terms of the mean value and standard deviation of IGD and HV on CF and CTP test problems.

Problem	IGD			HV		
	TPM	OUF5	BUFIS	TPM	OUF5	BUFIS
CF1	2.0108e-03(3.5171e-04)	2.4403e-03(1.2775e-03)≈	2.0969e-03(9.1973e-04)≈	6.8187e-01(5.6626e-04)	6.8166e-01(1.1274e-03)≈	6.8191e-01(9.3299e-04)≈
CF2	9.3485e-03(4.5410e-03)	6.4137e-03(8.9726e-04)	6.2910e-03(9.2013e-04)	8.1470e-01(2.9583e-03)	8.1432e-01(3.5518e-03)≈	8.1438e-01(2.8475e-03)≈
CF3	6.2946e-02(5.2139e-02)	7.3720e-02(7.6974e-02)≈	6.4775e-02(8.6692e-02)≈	4.0740e-01(5.3851e-02)	3.9867e-01(6.9015e-02)≈	4.0632e-01(8.1452e-02)≈
CF4	1.8614e-02(4.8328e-03)	2.2395e-02(2.3778e-02)≈	1.7898e-02(3.4187e-03)≈	6.1829e-01(7.5651e-03)	6.1874e-01(1.9983e-02)≈	6.2170e-01(5.5241e-03)≈
CF5	1.6147e-01(8.9643e-02)	2.3305e-01(8.9195e-02)+	2.1768e-01(9.5343e-02)+	4.4438e-01(6.7240e-02)	4.0314e-01(6.7446e-02)+	4.0820e-01(7.2936e-02)≈
CF6	2.0535e-02(3.5100e-03)	4.0260e-02(2.5079e-02)+	2.0563e-02(5.7258e-03)≈	8.2012e-01(3.9607e-03)	8.1152e-01(1.6791e-02)≈	8.2238e-01(4.4747e-03)≈
CF7	1.5917e-01(6.7675e-02)	2.4166e-01(1.0372e-01)+	2.3321e-01(1.3595e-01)+	6.1416e-01(5.9373e-02)	5.5367e-01(1.0380e-01)+	5.5840e-01(1.1894e-01)≈
CF8	7.5588e-02(5.9556e-03)	9.0815e-02(6.9417e-03)+	8.8932e-02(5.9953e-03)+	6.5586e-01(9.9842e-03)	6.2707e-01(1.0737e-02)+	6.3030e-01(1.0491e-02)+
CF9	3.5872e-02(2.1164e-03)	3.3121e-02(1.6866e-03)-	3.3327e-02(1.8201e-03)-	7.3134e-01(1.9597e-03)	7.3210e-01(1.8532e-03)≈	7.3200e-01(2.2714e-03)≈
CF10	2.8013e-01(1.5041e-01)	3.4202e-01(8.4486e-02)+	3.9575e-01(9.3362e-02)+	3.4646e-01(1.3620e-01)	3.2476e-01(6.6792e-02)≈	2.6046e-01(7.2181e-02)+
CTP1	9.7039e-03(9.3018e-04)	1.0438e-02(6.5085e-04)+	1.0430e-02(1.5706e-03)≈	4.5149e-01(4.0557e-04)	4.5129e-01(3.7095e-04)≈	4.5141e-01(5.8826e-04)≈
CTP2	5.6658e-03(1.3146e-03)	6.4802e-03(1.8815e-03)+	7.2938e-03(2.4913e-03)+	5.0488e-01(1.1352e-03)	5.0373e-01(1.4393e-03)+	5.0286e-01(2.3995e-03)+
CTP3	4.2315e-03(5.6957e-04)	3.8948e-03(6.2380e-04)-	3.8531e-03(5.6875e-04)-	4.8187e-01(8.8281e-04)	4.8236e-01(1.1634e-03)≈	4.8257e-01(1.0606e-03)-
CTP4	4.2904e-02(4.0914e-03)	4.4694e-02(3.7294e-03)+	3.9724e-02(4.7483e-03)-	4.3084e-01(4.9005e-03)	4.2907e-01(4.2733e-03)≈	4.3456e-01(6.3116e-03)-
CTP5	6.4862e-03(1.2049e-03)	5.7133e-03(1.1373e-03)-	9.6602e-03(3.2038e-03)+	4.9828e-01(1.0204e-03)	4.9730e-01(1.5553e-03)+	4.9757e-01(9.3319e-04)+
CTP6	1.7242e-02(2.2452e-03)	4.8113e-01(1.4185e+00)+	2.5007e-02(1.0594e-03)+	2.0670e+00(4.3141e-03)	1.8623e+00(6.3140e-01)+	2.0528e+00(2.1495e-03)+
CTP7	1.2172e-02(7.8112e-05)	1.2181e-02(7.8439e-05)≈	1.2189e-02(1.0398e-04)≈	6.6140e-01(2.9600e-04)	6.6131e-01(4.2599e-04)≈	6.6139e-01(3.4662e-04)≈
CTP8	1.4652e-02(2.7164e-03)	1.9253e+00(2.2244e+00)≈	2.1092e-02(1.1025e-03)+	1.3487e+00(3.4816e-03)	7.6449e-01(6.7997e-01)+	1.3403e+00(1.2539e-03)+
OR	27	44	37	27	48	33

Table 6Experimental results on investigation of the parameter a over 30 independent runs in terms of the mean value and standard deviation of IGD and HV on RPCMOP2 and RPCMOP5.

a	RPCMOP2		RPCMOP5	
	IGD	HV	IGD	HV
$a = 0.05$	1.9300e-03(2.1944e-04)	1.2059e-01(1.3493e-04)	4.4334e-03(2.3776e-05)	8.9702e-01(6.6818e-05)
$a = 0.15$	1.8730e-03(2.5975e-04)	1.2062e-01(1.3838e-04)	4.4537e-03(3.4115e-05)	8.9697e-01(1.0263e-04)
$a = 0.25$	1.8500e-03(1.7987e-04)	1.2064e-01(1.0201e-04)	4.4408e-03(2.7860e-05)	8.9711e-01(6.2051e-05)
$a = 0.35$	1.8611e-03(1.7892e-04)	1.2071e-01(1.0225e-04)	4.4501e-03(2.7921e-05)	8.9678e-01(6.1121e-05)
$a = 0.45$	1.8446e-03(1.8251e-04)	1.2104e-01(1.0437e-04)	4.4346e-03(2.7916e-05)	8.9698e-01(3.6828e-05)

Table 7Experimental results on investigation of the parameter r over 30 independent runs in terms of the mean value and standard deviation of IGD and HV on RPCMOP2 and RPCMOP5.

r	RPCMOP2		RPCMOP5	
	IGD	HV	IGD	HV
$r = 0.001$	1.8548e-03(1.8638e-04)	1.2062e-01(1.1216e-04)	4.4804e-03(4.2653e-05)	8.9684e-01(1.1439e-04)
$r = 0.01$	1.8730e-03(2.5975e-04)	1.2058e-01(1.3838e-04)	4.4537e-03(3.4115e-05)	8.9697e-01(1.0263e-04)
$r = 0.1$	1.7698e-03(1.4973e-05)	1.2066e-01(2.7115e-04)	4.4456e-03(2.0744e-05)	8.9710e-01(8.7405e-05)
$r = 0.5$	1.7664e-03(1.0738e-05)	1.2067e-01(2.1396e-05)	4.4462e-03(2.0773e-05)	8.9720e-01(8.7417e-05)
$r = 1$	1.7687e-03(1.0747e-05)	1.2074e-01(2.1446e-05)	4.4826e-03(3.8669e-05)	8.9695e-01(1.7522e-04)

Table 8

Experimental results over 30 independent runs in terms of the mean value and standard deviation of IGD and HV on RPCMOP2 and RCMOP5.

k	RPCMOP2		RPCMOP5	
	IGD	HV	IGD	HV
$k = 2$	1.7492e-03(5.0695e-06)	1.2071e-01(1.4381e-05)	4.4490e-03(3.2588e-05)	8.9708e-01(9.2871e-05)
$k = 4$	1.7558e-03(1.3424e-05)	1.2067e-01(3.4456e-05)	4.4754e-03(3.5232e-05)	8.9701e-01(9.1612e-05)
$k = 5$	1.8730e-03(2.5975e-04)	1.2062e-01(1.3838e-04)	4.4537e-03(3.4115e-05)	8.9697e-01(1.0263e-04)
$k = 6$	2.0594e-03(4.0259e-04)	1.2050e-01(2.0108e-04)	4.6828e-03(5.1765e-04)	8.9664e-01(6.4047e-04)
$k = 8$	1.9722e-03(2.5965e-04)	1.2052e-01(1.5176e-04)	4.7106e-03(5.5355e-04)	8.9564e-01(2.8808e-03)

whether the unconstrained and constrained reference point were different or not. And then the location estimation mechanism was designed to help find the most suitable reference point between the unconstrained and constrained reference point as shown in Algorithm 2.

Subsequently, we conducted a series of experiments to study performance of the three CMOEAs on PRCMOPs, CFs, CTPs and a real-world problem OSY. All the experimental results highlighted the advantages of the proposed framework on solving the CMOPs used in this paper. Furthermore, the parameters of the proposed framework were studied and the results showed that the proposed framework was insensitive to these parameters on the test instances studied.

CRediT authorship contribution statement

Chaoda Peng: Conceptualization, Methodology, Software, Data Curation, Writing – original draft. **Hai-Lin Liu:** Conceptualization,

Investigation, Supervision, Writing – review & editing. **Erik D. Goodman:** Investigation, Supervision, Validation, Writing – review & editing. **Kay Chen Tan:** Validation, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 62172110; in part by the Natural Science Foundation of Guangdong Province, China 2020A1515011500; and in part by the Programme of Science and

Technology of Guangdong Province, China 2021A0505110004, 2020A0505100056.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at <https://doi.org/10.1016/j.knosys.2021.107933>.

References

- [1] Z.-Z. Liu, Y. Wang, Handling constrained multiobjective optimization problems with constraints in both the decision and objective spaces, *IEEE Trans. Evol. Comput.* (2019).
- [2] Z. Fan, W. Li, X. Cai, H. Li, C. Wei, Q. Zhang, K. Deb, E. Goodman, Push and pull search for solving constrained multi-objective optimization problems, *Swarm Evol. Comput.* 44 (2019) 665–679.
- [3] C. Peng, H.-L. Liu, F. Gu, An evolutionary algorithm with directed weights for constrained multi-objective optimization, *Appl. Soft Comput.* 60 (2017) 613–622.
- [4] J. Wang, G. Liang, J. Zhang, Cooperative differential evolution framework for constrained multiobjective optimization, *IEEE Trans. Cybern.* (99) (2018) 1–13.
- [5] C. Peng, H.-L. Liu, E.D. Goodman, A cooperative evolutionary framework based on an improved version of directed weight vectors for constrained multiobjective optimization with deceptive constraints, *IEEE Trans. Cybern.* (2020) 1–13.
- [6] L. Jiao, J. Luo, R. Shang, F. Liu, A modified objective function method with feasible-guiding strategy to solve constrained multi-objective optimization problems, *Appl. Soft Comput.* 14 (2014) 363–380.
- [7] C. Peng, H.-L. Liu, E.D. Goodman, Handling multi-objective optimization problems with unbalanced constraints and their effects on evolutionary algorithm performance, *Swarm Evol. Comput.* (2020) 100676.
- [8] H. Jain, K. Deb, An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part II: handling constraints and extending to an adaptive approach, *IEEE Trans. Evol. Comput.* 18 (4) (2013) 602–622.
- [9] K. Li, R. Chen, G. Fu, X. Yao, Two-archive evolutionary algorithm for constrained multi-objective optimization, *IEEE Trans. Evol. Comput.* (2018).
- [10] Y. Zhou, M. Zhu, J. Wang, Z. Zhang, Y. Xiang, J. Zhang, Tri-goal evolution framework for constrained many-objective optimization, *IEEE Trans. Syst. Man Cybern.: Syst.* (99) (2018) 1–14.
- [11] Q. Zhu, Q. Zhang, Q. Lin, A constrained multiobjective evolutionary algorithm with detect-and-escape strategy, *IEEE Trans. Evol. Comput.* (2020).
- [12] H. Wang, T. Cai, K. Li, W. Pedrycz, Constraint handling technique based on lebesgue measure for constrained multiobjective particle swarm optimization algorithm, *Knowl.-Based Syst.* 227 (2021) 107131.
- [13] R. Wang, J. Xiong, H. Ishibuchi, G. Wu, T. Zhang, On the effect of reference point in MOEA/D for multi-objective optimization, *Appl. Soft Comput.* 58 (2017) 25–34.
- [14] H. Fukumoto, A. Oyama, Impact of estimation method of ideal/nadir points on practically-constrained multi-objective optimization problems for decomposition-based multi-objective evolutionary algorithm, in: 2019 IEEE Symposium Series on Computational Intelligence, SSCI, IEEE, 2019, pp. 2138–2145.
- [15] Z. Ma, Y. Wang, Evolutionary constrained multiobjective optimization: Test suite construction and performance comparisons, *IEEE Trans. Evol. Comput.* 23 (6) (2019) 972–986.
- [16] Z. Fan, W. Li, X. Cai, H. Li, C. Wei, Q. Zhang, K. Deb, E. Goodman, Difficulty adjustable and scalable constrained multiobjective test problem toolkit, *Evol. Comput.* 28 (3) (2020) 339–378.
- [17] Q. Zhang, A. Zhou, S. Zhao, P.N. Suganthan, W. Liu, S. Tiwari, Multiobjective Optimization Test Instances for the CEC 2009 Special Session and Competition, Vol. 264, Technical Report, University of Essex, Colchester, UK and Nanyang Technological University, Singapore, 2008, Special session on performance assessment of multi-objective optimization algorithms.
- [18] A. Panda, S. Pani, A symbiotic organisms search algorithm with adaptive penalty function to solve multi-objective constrained optimization problems, *Appl. Soft Comput.* 46 (2016) 344–360.
- [19] G.G. Yen, An adaptive penalty function for handling constraint in multi-objective evolutionary optimization, in: *Constraint-Handling in Evolutionary Optimization*, Springer, 2009, pp. 121–143.
- [20] P.D. Surry, N.J. Radcliffe, The COMOGA method: constrained optimisation by multi-objective genetic algorithms, *Control Cybernet.* 26 (1997) 391–412.
- [21] Y.G. Woldesenbet, G.G. Yen, B.G. Tessema, Constraint handling in multiobjective evolutionary optimization, *IEEE Trans. Evol. Comput.* 13 (3) (2009) 514–525.
- [22] A. Ben Hadj-Alouane, J.C. Bean, A genetic algorithm for the multiple-choice integer program, *Oper. Res.* 45 (1) (1997) 92–101.
- [23] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE Trans. Evol. Comput.* 6 (2) (2002) 182–197.
- [24] M.A. Jan, R.A. Khanum, A study of two penalty-parameterless constraint handling techniques in the framework of MOEA/d, *Appl. Soft Comput.* 13 (1) (2013) 128–148.
- [25] T.P. Runarsson, X. Yao, Stochastic ranking for constrained evolutionary optimization, *IEEE Trans. Evol. Comput.* 4 (3) (2000) 284–294.
- [26] K. Harada, J. Sakuma, I. Ono, S. Kobayashi, Constraint-handling method for multi-objective function optimization: Pareto descent repair operator, in: *International Conference on Evolutionary Multi-Criterion Optimization*, Springer, 2007, pp. 156–170.
- [27] H.K. Singh, T. Ray, W. Smith, C-PSA: Constrained Pareto simulated annealing for constrained multi-objective optimization, *Inform. Sci.* 180 (13) (2010) 2499–2513.
- [28] M. Miyakawa, K. Takadama, H. Sato, Two-stage non-dominated sorting and directed mating for solving problems with multi-objectives and constraints, in: *Proceedings of the 15th Annual Conference on Genetic and Evolutionary Computation*, 2013, pp. 647–654.
- [29] F. Ming, W. Gong, H. Zhen, S. Li, L. Wang, Z. Liao, A simple two-stage evolutionary algorithm for constrained multi-objective optimization, *Knowl.-Based Syst.* 228 (2021) 107263.
- [30] R. Tanabe, A. Oyama, A note on constrained multi-objective optimization benchmark problems, in: *2017 IEEE Congress on Evolutionary Computation, CEC, IEEE*, 2017, pp. 1127–1134.
- [31] A.P. Wierzbicki, The use of reference objectives in multiobjective optimization, in: *Multiple Criteria Decision Making Theory and Application*, Springer, 1980, pp. 468–486.
- [32] E. Zitzler, L. Thiele, Multiobjective optimization using evolutionary algorithms—a comparative case study, in: *International Conference on Parallel Problem Solving from Nature*, Springer, 1998, pp. 292–301.
- [33] C.A.C. Coello, G.B. Lamont, D.A. Van Veldhuizen, et al., *Evolutionary Algorithms for Solving Multi-Objective Problems*, Vol. 5, Springer, 2007.
- [34] K. Deb, A. Pratap, T. Meyarivan, Constrained test problems for multi-objective evolutionary optimization, in: *International Conference on Evolutionary Multi-Criterion Optimization*, Springer, 2001, pp. 284–298.
- [35] A. Osyczka, S. Kundu, A new method to solve generalized multicriteria optimization problems using the simple genetic algorithm, *Struct. Optim.* 10 (2) (1995) 94–99.
- [36] C.M. Fonseca, J.D. Knowles, L. Thiele, E. Zitzler, A tutorial on the performance assessment of stochastic multiobjective optimizers, in: *Third International Conference on Evolutionary Multi-Criterion Optimization*, Vol. 216, 2005.

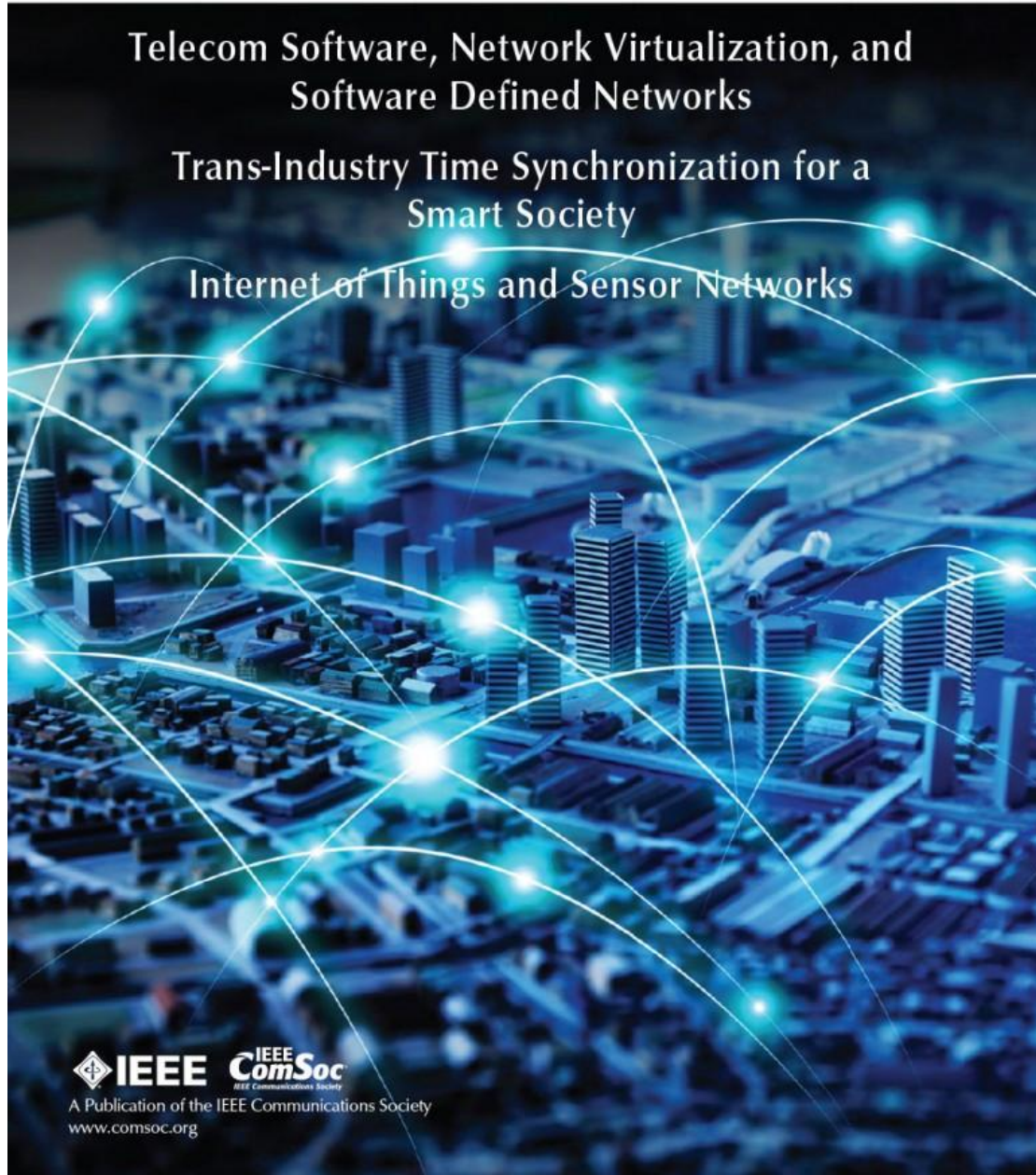
IEEE COMMUNICATIONS MAGAZINE

April 2022, Vol. 11, No. 4

Telecom Software, Network Virtualization, and
Software Defined Networks

Trans-Industry Time Synchronization for a
Smart Society

Internet of Things and Sensor Networks



A Publication of the IEEE Communications Society
www.comsoc.org

IEEE WIRELESS COMMUNICATIONS LETTERS

A PUBLICATION OF THE IEEE COMMUNICATIONS SOCIETY, THE IEEE SIGNAL PROCESSING SOCIETY, AND THE IEEE VEHICULAR TECHNOLOGY SOCIETY



APRIL 2022

VOLUME 11

NUMBER 4

IWCLAF

ISSN (2162-2345)

Exemplary Reviewers 2021	<i>Kai-Kit Wong</i>	667
Electromagnetic Interference in RIS-Aided Communications	<i>Andrea de Jesus Torres, Luca Sanguinetti, and Emil Björnson</i>	668
Exploiting Distributed IRSs for Enabling SWIPT	<i>Diluka Loku Galappaththige and Gayan Aruma Baduge</i>	673
RF Beamforming and Subcarrier Allocation Using Beam Squint in mmWave Systems	<i>Nancy Varshney and Swades De</i>	678
Age of Information Analysis for Finite Blocklength Regime in Downlink Cellular Networks	<i>Hyeonjun Sung, Minsu Kim, Sungjin Lee, and Jemin Lee</i>	683
A Novel Range-Free Node Localization Method for Wireless Sensor Networks	<i>Yong Jin, Lin Zhou, Lu Zhang, Zhentao Hu, and Jing Han</i>	688
Proceed From Known to Unknown: Jamming Pattern Recognition Under Open-Set Setting	<i>Hao Han, Wen Li, Zhibin Feng, Gui Fang, Yifan Xu, and Yuhua Xu</i>	693
Low-Density Spreading Design Based on an Algebraic Scheme for NOMA Systems	<i>Goldwyn Millar, Michel Kulhandjian, Ayse Alaca, Saban Alaca, Claude D'Amours, and Halim Yanikomeroglu</i>	698
Flexible Index Mapping Scheme for Packet-Level Index Modulation	<i>Kosuke Suzuki, Koichi Adachi, Mai Ohta, Osamu Takyu, and Takeo Fujii</i>	703
Learning of Time-Frequency Attention Mechanism for Automatic Modulation Recognition	<i>Shangao Lin, Yuan Zeng, and Yi Gong</i>	707

(Contents Continued on Page 665)



Beam Tracking for Distributed Millimeter-Wave Massive MIMO Systems Based on the Unscented Kalman Filter	Pengcheng Zhu, Huixin Lin, Jialong Bao, Jiamin Li, and Dongming Wang	712
Ground-Assisted Federated Learning in LEO Satellite Constellations	Nasrin Razmi, Bho Matthiesen, Armin Dekorsy, and Petar Popovski	717
Selection Combining Over the Extended η - μ Fading Channels	Osamah S. Badarneh, Mustafa K. Alshawaqfeh, Fares S. Almeahmadi, and Hugerles S. Silva	722
Secrecy Outage Probability of MIMO Diversity Schemes Over Integer and Real-Valued Nakagami Fading Channels	Donghun Lee	727
Client Scheduling in Wireless Federated Learning Based on Channel and Learning Qualities	Jichao Leng, Zihuai Lin, Ming Ding, Peng Wang, David Smith, and Branka Vucetic	732
Improving IoT-Over-Satellite Connectivity Using Frame Repetition Technique	Bisma Manzoor, Akram Al-Hourani, and Bassel Al Homssi	736
Antenna Selection for Energy-Efficient Dual-Functional Radar-Communication Systems	Iman Valiulahi, Christos Masouros, Abdelhamid Salem, and Fan Liu	741
Robust Beamforming Design for IRS-Aided Secure SWIPT Terahertz Systems With Non-Linear EH Model	Zhengyu Zhu, Jinlei Xu, Gangcan Sun, Wanming Hao, Zheng Chu, Cunhua Pan, and Inkyu Lee	746
Enhanced Index Modulation-Based Frequency Hopping: Resist Power-Correlated Reactive Jammer	Yuxin Shi, Kang An, Xinjin Lu, and Yusheng Li	751
Cooperative Activation and Caching Strategy for Low-Latency and Energy-Efficient Small-Cell Networks	Jingjing Luo, Qinsong Wang, Fu-Chun Zheng, Lin Gao, and Shushi Gu	756
AoA-Based Positioning for Aerial Intelligent Reflecting Surface-Aided Wireless Communications: An Angle-Domain Approach	Tao Zhou, Kui Xu, Zhexian Shen, Wei Xie, Dongmei Zhang, and Jianhui Xu	761
Asynchronous Transmission for Cooperative Free-Space Optical Communication System	Yejun Liu, Yanqin He, Kun Chen, and Lei Guo	766
Optimal Beamwidth and Altitude for Maximal Uplink Coverage in Satellite Networks	Bassel Al Homssi and Akram Al-Hourani	771
Single-Channel Anti-Jamming Receiver With Harmonic-Based Space-Time Adaptive Processing	Gang Ni, Chong He, and Ronghong Jin	776
Multiple Antennas-Based Secure Communications With Channel Inversion Power Control	Jinsong Hu, Xiaoqiang Shi, Youjia Chen, Haifeng Zheng, and Feng Shu	781
Generalized Multistream Spatial Modulation Based on Frequency-Phase Modulated Signals	Nodar Ugrelidze, Sergo Shavgulidze, and Juergen Freudenberger	786
Achievable Rate Upper-Bounds of Uplink Multiuser OTFS Transmissions	Ruoxi Chong, Shuangyang Li, Jinhong Yuan, and Derrick Wing Kwan Ng	791
Scalable Channel Estimation and Reflection Optimization for Reconfigurable Intelligent Surface-Enhanced OFDM Systems	Jiancheng An, Qingqing Wu, and Chau Yuen	796
Capacity Improvement for Intelligent Reflecting Surface-Assisted Wireless Systems With a Limited Number of Passive Elements	Jung-Chieh Chen	801
Digital Twin-Aided Intelligent Offloading With Edge Selection in Mobile Edge Computing	Tan Do-Duy, Dang Van Huynh, Octavia A. Dobre, Berk Canberk, and Trung Q. Duong	806
ML-Based Massive MIMO Channel Prediction: Does It Work on Real-World Data?	Muhammad K. Shehzad, Luca Rose, Stefan Wesemann, and Mohamad Assaad	811
Performance Analysis for Rate Splitting Uplink NOMA Transmission in High Throughput Satellite Systems	Huaicong Kong, Min Lin, Zining Wang, Jin-Yuan Wang, Wei-Ping Zhu, and Jiangzhou Wang	816
Age of Information and Throughput in Random Access-Based IoT Systems With Periodic Updating	Yun Han Bae and Jung Woo Baek	821
Extreme Age of Information for Wireless-Powered Communication Systems	Nikolaos I. Miridakis, Zheng Shi, Theodoros A. Tsiftsis, and Guanghua Yang	826
On the Statistics of the Maximum of α - \mathcal{F} Variates and Their Applications	Hugerles S. Silva, Guilherme B. L. C. Lourenço, Wamberto J. L. Queiroz, Arnaldo S. R. Oliveira, Francisco Madeiro, and Osamah S. Badarneh	831
High-Performance Passive Eigen-Model-Based Detectors of Single Emitter Using Massive MIMO Receivers	Qijuan Jie, Xichao Zhan, Feng Shu, Yaohui Ding, Baihua Shi, Yifan Li, and Jiangzhou Wang	836

Learning to Estimate RIS-Aided mmWave Channels	841
..... <i>Jiguang He, Henk Wymeersch, Marco Di Renzo, and Markku Juntti</i>	
Deep Compressed Sensing-Based Cascaded Channel Estimation for RIS-Aided Communication Systems	846
..... <i>Wenwu Xie, Jian Xiao, Peng Zhu, Chao Yu, and Liang Yang</i>	
Opportunistic Fountain Coding With Coordinative Routing	851
..... <i>Tong Peng, Sangarapillai Lambotharan, Gan Zheng, and Mohammad Shikh-Bahaei</i>	
Hardware-Impaired PHY Secret Key Generation With Man-in-the-Middle Adversaries	856
..... <i>Mehdi Letafati, Hamid Behroozi, Babak Hossein Khalaj, and Eduard A. Jorswieck</i>	
Constrained Multi-Objective Optimization for UAV-Enabled Mobile Edge Computing: Offloading Optimization and Path Planning	861
..... <i>Chaoda Peng, Xumin Huang, Yuan Wu, and Jiawen Kang</i>	
Hybrid Uplink and Downlink Transmissions for Full-Duplex UAV Communication With RIS	866
..... <i>Kaiyuan Tian, Bin Duo, Sixian Li, Yong Zuo, and Xiaojun Yuan</i>	
Channel Inversion Power Control Aided Covert Communications in Uplink NOMA Systems	871
..... <i>Maochun Wang, Weiwei Yang, Xingbo Lu, Caibo Hu, Bing Liu, and Xin Lv</i>	

Constrained Multi-Objective Optimization for UAV-Enabled Mobile Edge Computing: Offloading Optimization and Path Planning

Chaoda Peng, Xumin Huang^{ID}, Yuan Wu^{ID}, *Senior Member, IEEE*, and Jiawen Kang

Abstract—An unmanned aerial vehicle (UAV) is employed to sequentially visit the specific waypoints and provide offloading services for nearby devices. Most of the current works optimized the UAV-enabled offloading according to a single criterion while neglecting necessary optimizations and constraints for flight safety of the UAV. This motivates us to study the optimization problem of the UAV from a multi-objective viewpoint by considering the UAV's flight safety. A constrained multi-objective optimization problem (CMOP) involving two objective functions about the energy-efficient offloading and safe path planning is formulated for the UAV. To solve the formulated CMOP, we present a constrained decomposition-based multi-objective evolution algorithm. To further improve the algorithm, we particularly utilize the infeasible individuals with great objective values, which provide useful information for improving the optimized objective values during the evolution process. Finally, experimental results demonstrate that compared with the existing works, our scheme is beneficial to simultaneously reduce energy consumption and ensure safe flight for the UAV.

Index Terms—Computation offloading, 3D path planning, constrained multi-objective optimization, and evolutionary algorithm.

I. INTRODUCTION

DUE TO the flexible deployment and mobility, unmanned aerial vehicle (UAV) has been widely exploited to provide various services, e.g., event and data detection [1], [2], reliable connectivity and proximal computing for users, particularly in the scenarios where communication infrastructures are damaged and network congestion is continuously aggravated. This results in a new computing paradigm called by UAV-enabled mobile edge computing (MEC).

Manuscript received January 1, 2022; accepted January 31, 2022. Date of publication February 4, 2022; date of current version April 11, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 62001125 and Grant 62102099; in part by the Science and Technology Development Fund of Macau SAR under Grant 0060/2019/A1 and Grant 0162/2019/A3; in part by the FDCT-MOST Joint Project under Grant 0066/2019/AMJ; in part by the Key Project in Higher Education of Guangdong Province under Grant 2020ZDZX3030; and in part by the Special Fund for Talents of South China Agricultural University under Grant 221114. The associate editor coordinating the review of this article and approving it for publication was K. Ota. (*Corresponding author: Xumin Huang.*)

Chaoda Peng is with the College of Mathematics and Informatics, South China Agricultural University, Guangzhou 510642, China (e-mail: chaodapeng@scau.edu.cn).

Xumin Huang is with the School of Automation, Guangdong University of Technology, Guangzhou 510006, China, and also with the State Key Laboratory of Internet of Things for Smart City, University of Macau, Macau, China (e-mail: huangxu_min@163.com).

Yuan Wu is with the State Key Laboratory of Internet of Things for Smart City and the Department of Computer and Information Science, University of Macau, Macau, China (e-mail: yuanwu@um.edu.mo).

Jiawen Kang is with the School of Automation, Guangdong University of Technology, Guangzhou 510006, China (e-mail: kavinkang@gdut.edu.cn).

Digital Object Identifier 10.1109/LWC.2022.3149007

Many research efforts have been devoted to jointly optimizing the task offloading and trajectory design for the UAV scheduling. The authors in [3] considered that the UAV was approximately stationary and used binary offloading decisions for the devices with offloading requests. The similar problem was extended to jointly optimize the user association and horizontal location of the UAV to maximize the overall data rate for the users [4]. The joint optimization of resource allocation and 3D trajectory of the UAV from the viewpoint of energy efficiency was studied in [5]. The use of the UAV as an aerial communication platform was proposed to tackle the traffic offloading problems in a variety of application scenarios such as cellular hotspot areas [6] and community communications [7]. A dual-role UAV playing both as an edge-server and a traffic relay was proposed in [8]. Furthermore, wireless power transfer was integrated into the UAV-enabled MEC to establish the on-demand power links and communication channels for wireless devices [9].

However, most of the current works formulated the network-wide optimization problem as a single-objective optimization problem, while the UAV scheduling could consider both the efficiency of task processing and the safe path planning of the UAV as a joint criterion. Toward feasible deployment of the UAV, there may exist several important yet conflicting objectives which need to be jointly optimized. Technically, it is not suitable to simply sum the different objectives with fixed weights. Moreover, the current UAV's trajectory was determined by assigning processing order to the devices and the trajectory design was based on the straight flight among the specific locations. The practical UAV path planning that accounts for the obstacle avoidance and safe flight requirements has not been widely studied yet.

Motivated by the above considerations, we investigate a constrained multi-objective optimization problem (CMOP) for UAV-enabled MEC, which aims at simultaneously achieving the energy-efficient offloading and safe path planning for the UAV. Given the locations of the devices, the UAV flies from one place to another to provide offloading services for the devices. In this multi-objective optimization, we simultaneously optimize the transmission power of the devices, computing resource, flying velocity and 3D path of the UAV. The main contributions of this letter are summarized as follows.

- A CMOP for UAV-enabled MEC is investigated to simultaneously study the energy-efficient offloading and safe path planning for the UAV.
- A constrained multi-objective evolutionary algorithm with a mechanism of utilizing the useful infeasible individuals is developed to tackle the proposed problem.

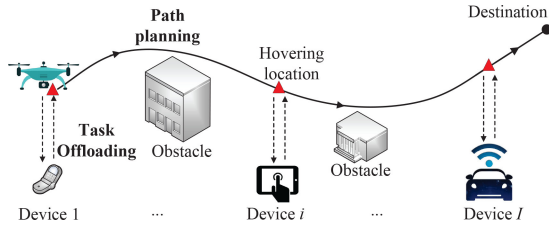


Fig. 1. System model.

- Numerical results are provided to demonstrate the effectiveness of our proposed algorithm, especially its robustness in obtaining the feasible optimal solutions.

II. PROBLEM FORMULATION

We consider a system model in Fig. 1 where a single UAV with computing resources is scheduled to sequentially visit I hovering locations to serve I devices, and finally flies toward the destination. For simplicity, the UAV flies above the devices at a constant height H . We denote the location of device i as $(x'_i, y'_i, 0)$, and the i -th hovering location of the UAV refers to (x_i, y_i, H) . We describe the task of device i by input data size S_i and computation workloads W_i . Similar to [10], we do not consider the small output data size compared with the input data size and the following delay of receiving the output data. Let B_i and p_i^{tx} represent the channel bandwidth and transmitter power of device i , respectively. Thus, the uplink data rate of device i is expressed by $r_i^{\text{UL}} = B_i \log_2(1 + p_i^{\text{tx}} g_i / \sigma^2)$, where the free-space path loss model $g_i = g_0 / H^2$ in [11], [12] is used, g_0 is the received power at the reference distance $d_0 = 1$ m and σ^2 is the noise power. For the UAV, let $f_{\text{UAV},i}$ indicate the computing resources allocated to device i . The hovering duration of the i -th hovering location is calculated by $\tau_i^{\text{H}} = t_i^{\text{d}} + t_i^{\text{w}}$, where the data transmission and workload processing time are $t_i^{\text{d}} = S_i / r_i^{\text{UL}}$ and $t_i^{\text{w}} = W_i / f_{\text{UAV},i}$, respectively. Given a real path between the i -th and the $(i+1)$ -th hovering locations, the length of the i -th path segment is measured by L_i , and the flying time of the UAV is $\tau_i^{\text{F}} = L_i / v_i$ under the assumption that the UAV is flying at a constant velocity denoted by v_i .

At the i -th hovering location, energy consumption for receiving the offloading data, handling the offloading data and staying hovering is equal to $p_{\text{UAV}}^{\text{rx}} t_i^{\text{d}}$, $\varepsilon_{\text{UAV}} f_{\text{UAV},i}^2 W_i$, and $p_{\text{UAV}}^{\text{H}} \tau_i^{\text{H}}$, respectively, where $p_{\text{UAV}}^{\text{rx}}$, ε_{UAV} and $p_{\text{UAV}}^{\text{H}}$ are the receiver power, effective switched capacitance of the processor, and hovering power, respectively. To maintain the stable flying motion with the constant velocity, the flying power of the UAV is estimated by $p_{\text{UAV}}^{\text{F}}$ according to the method in [13]. From the i -th hovering location to the $(i+1)$ -th hovering location, the total energy consumption of the UAV, which is denoted by E_i , is given as follows:

$$E_i = p_{\text{UAV}}^{\text{rx}} t_i^{\text{d}} + \varepsilon_{\text{UAV}} f_{\text{UAV},i}^2 W_i + p_{\text{UAV}}^{\text{H}} \tau_i^{\text{H}} + p_{\text{UAV}}^{\text{F}} \tau_i^{\text{F}} \quad (1)$$

The delay constraint is given by:

$$C_1: h_1 = T_i - \tau_i^{\text{H}} - \tau_i^{\text{F}}, \text{ and } h_1 \geq 0 \quad (2)$$

where T_i is the maximum tolerable time duration required by the UAV. We try to reduce E_i to achieve the energy-efficient task offloading in UAV-enabled MEC.

At the same time, we enable the safe flight for the UAV. Before defining the whole UAV's path, we simulate the flying environment of the UAV by considering the existence of obstacles. Referring to [14], we use the following 3D model:

$$z(x, y) = \sin(y + \kappa_1) + \kappa_2 \sin(x) + \kappa_3 \cos(y) + \kappa_4 \cos(y) + \kappa_5 \cos(\kappa_6 \sqrt{x^2 + y^2}) + \kappa_7 \sin(\kappa_7 \sqrt{x^2 + y^2}) \quad (3)$$

where $\kappa_1, \kappa_2, \kappa_3, \kappa_4, \kappa_5, \kappa_6$ and κ_7 are experimentally studied constants, and they can be configured to produce the consistent surface of the obstacles such as a building, valley and mountain. B-spline curve is used in this letter to model the UAV's path since it is defined only by a set of control points that can represent a complicated path. It has been widely utilized in industrial applications such as computer graphic representations and computer aided manufacturing [14]. Suppose that we have λ control points $CP_1 = (x_1, y_1, z_1)$, $CP_2 = (x_2, y_2, z_2), \dots, CP_\lambda = (x_\lambda, y_\lambda, z_\lambda)$, and then the corresponding B-spline curve, i.e., the UAV's path, consists of J path points $B_1 = (x'_1, y'_1, z'_1), B_2 = (x'_2, y'_2, z'_2), \dots, B_J = (x'_J, y'_J, z'_J)$.

To derive a collision-free path, an objective function with respect to the safe flight is considered. We project the path points and mesh points of the obstacles into horizontal plane coordinate, and obtain the mesh points of the obstacles within the safe distance denoted by d_s , which guarantees that the UAV flies away from the known obstacles. By referring to [15], we set the objective function which is related to the safe path as:

$$D_s = \sum_{j=1}^J \sum_{k=1}^K \left(\frac{d_s}{d_{j,k}} \right)^2 \quad (4)$$

where K is the number of mesh points of all obstacles within the safe distance, and $d_{j,k}$ indicates the Euclidean distance between the j -th path point and the k -th mesh point. We aim to reduce D_s since a smaller value of D_s means that the UAV can reduce the risk of colliding with the obstacles.

In addition, the UAV's path should satisfy the following three constraints, i.e., C_2 , C_3 , and C_4 . Constraint C_2 ensures that the UAV flies above the minimum flight altitude, namely,

$$C_2: h_2 = \sum_{j=1}^J \left[d_j^{\min} \right]^-, \text{ and } h_2 = 0 \quad (5)$$

where $[\bullet]^- = \min(\bullet, 0)$, $d_j^{\min} = z'_j - z(x'_j, y'_j) - h^{\min}$, and h^{\min} is the minimum flight altitude, and $(x'_j, y'_j, z(x'_j, y'_j))$ is the j -th mesh point regarding to Eq. (3). $h_2 = 0$ means that each path point is above the minimum flight height. However, $h_2 < 0$ means that there are some path points below the minimum flight height, i.e., violating the feasible conditions of the safety path. As we will illustrate in Section III, we will leverage infeasible individuals for improving the performance of our evolutionary algorithm. Thus, the value of h_2 (when it is negative) will be used in Eq. (10) at the beginning of Section III for evaluating how much an individual violates the feasibility conditions.

The following constraint C_3 restricts the upper flight altitude of the UAV:

$$C_3: h_3 = \sum_{j=1}^J \left[d_j^{\max} \right]^-, \text{ and } h_3 = 0 \quad (6)$$

where $d_j^{\max} = h^{\max} - z'_j$, and h^{\max} is the maximum flight altitude. $h_3 = 0$ means that each path point is under the maximum flight height.

The constraint C_4 ensures that the turning angle of the UAV along the path cannot surpass the maximum value θ_{\max} ,

$$C_4: h_4 = \sum_{j=2}^{J-1} [\Delta\theta_j]^- , \text{ and } h_4 = 0 \quad (7)$$

where $\Delta\theta_j = \theta^{\max} - \theta(\mathbf{B}_{j,j-1}, \mathbf{B}_{j+1,j})$, $\mathbf{B}_{m,n}$ means the vector from point B_m to point B_n , and $\theta(\mathbf{B}_{j,j-1}, \mathbf{B}_{j+1,j})$ is the angle of two vectors $\mathbf{B}_{j,j-1}$ and $\mathbf{B}_{j+1,j}$,

$$\theta(\mathbf{B}_{j,j-1}, \mathbf{B}_{j+1,j}) = \cos^{-1} \left(\frac{\mathbf{B}_{j,j-1} \cdot \mathbf{B}_{j+1,j}}{\|\mathbf{B}_{j,j-1}\| \|\mathbf{B}_{j+1,j}\|} \right) \quad (8)$$

Finally, the proposed CMOP is given as follows.

$$\begin{aligned} \min \quad & \begin{cases} G_1(\mathbf{x}) = D_s \\ G_2(\mathbf{x}) = \sum_{i=1}^I E_i \end{cases} \\ \text{s.t.} \quad & C_1 \sim C_4, \mathbf{x} \in \mathcal{D} \end{aligned} \quad (9)$$

where $G_1(\mathbf{x})$ and $G_2(\mathbf{x})$ are the two objective functions related to the safe path planning and energy consumption, respectively. \mathbf{x} is a $(3\lambda + 3I)$ dimensional decision variable in the given decision space \mathcal{D} , which includes two parts: the first part is a sequence of λ control points in sequence and the second part refers to $\{p_i^{\text{tx}}, f_{\text{UAV},i}, v_i, \forall i\}$. The λ control points are represented by an one-dimensional vector with the dimension of 3λ , and the decision variables of offloading optimization are also represented by an one-dimensional vector with the dimension of $3I$. As a summary, we will use $\mathbf{x} = (x_1, y_1, z_1, \dots, x_\lambda, y_\lambda, z_\lambda, p_1^{\text{tx}}, f_{\text{UAV},1}, v_1, \dots, p_I^{\text{tx}}, f_{\text{UAV},I}, v_I)$ in our proposed system.

III. PROPOSED ALGORITHM

The above problem (9) is a complicated CMOP. Referring to [16], we calculate the constraint violation of an individual \mathbf{x} according to the constraints from C_1 to C_4 :

$$CV(\mathbf{x}) = \sum_{i=1}^4 |cv_i(\mathbf{x})| \quad (10)$$

where $cv_i(\mathbf{x}) = \min(0, h_i(\mathbf{x}))$. $cv_i(\mathbf{x}) = 0, \forall i$ means that \mathbf{x} is a feasible individual, while $cv_i(\mathbf{x}) \neq 0, \exists i$ means that \mathbf{x} is an infeasible individual. To solve the problem (9), a multi-objective evolutionary algorithm in [17] with a dynamic infeasibility allocation mechanism is proposed. It has three main components, i.e., the initialization, the reproduction, and the constraint-handling technique with a dynamic infeasibility allocation mechanism. The details are shown as follows.

Step 1 (Initialization): In the first phase, we initialize a population P_0 with N individuals, and calculate the values of the two objective functions $G_1(\mathbf{x})$, $G_2(\mathbf{x})$ and four constraints

Algorithm 1: The Proposed Constraint-Handling Technique

Input:

- The combined population M_t .
- The M unit center vectors.
- The N weight vectors.

Output:

- K sub-populations $\Omega_1, \dots, \Omega_M$.

```

1: Update  $\alpha$  by using Eq. (13).
2: A dominated infeasible individuals trim scheme is applied to  $M_t$  by eliminating
   the infeasible individuals which do not dominate any feasible individual.
3: The individuals in  $M_t$  are decomposed into  $K$  sub-populations  $\Omega_1, \dots, \Omega_M$  by
   using Eq. (11).
4: for each sub-population  $\Omega_i$  do
5:   if  $\|\Omega_i\| < s_i$  then
6:     Select all the individuals in  $\Omega_i$  and randomly select  $s_i - \|\Omega_i\|$  individuals
       from  $M_t$  as the next sub-population for  $\Omega_i$ .
7:   else
8:      $\delta = \alpha s_i$ . % The number of feasible solutions should be saved in advance.
9:     Obtain the number of the feasible individuals in  $\Omega_i$ :  $\delta'$ .
10:    if  $\delta' < \delta$  then
11:      Sort the individuals in  $\Omega_i$  in ascending order of  $CV(\mathbf{x})$  regarding to
        Eq. (10), and then the best  $\delta$  individuals are stored into  $\Omega_i$ .
12:    else
13:      Select the best  $\delta$  feasible individuals in  $\Omega_i$  by using the weight vectors
         $\mathbf{V}^1, \mathbf{V}^2, \dots, \mathbf{V}^{s_i}$  according to Eq. (12).
14:    end if
15:    if  $\delta < s_i$  then
16:      Select the best  $s_i - \delta$  individuals from the rest of the population in  $\Omega_i$ 
        in terms of ASF.
17:    end if
18:  end if
19: end for

```

$C_1 \sim C_4$. The current generation t is set to 1. A set of N weight vectors $\mathbf{V}^1, \mathbf{V}^2, \dots, \mathbf{V}^N$ are evenly chosen from the hyperplane to select a set of individuals, since a weight vector is corresponding to a Pareto optimal solution in the context of multi-objective optimization. The N weight vectors are decomposed into M sub-populations $\Omega_1, \Omega_2, \dots, \Omega_M$ by using a set of M unit center vectors, and the size of a sub-population s_i is determined by the number of the weight vectors assigned into the Ω_i . Each weight vector is assigned to its closest unit center vector according to Eq. (11).

$$\Omega_i = \left\{ \mathbf{u} \mid \langle \mathbf{u}, \mathbf{w}^i \rangle \leq \langle \mathbf{u}, \mathbf{w}^j \rangle, 1 \leq j \leq M \right\} \quad (11)$$

where \mathbf{u} is a vector and \mathbf{w}^i is a unit center vector.

Step 2 (Reproduction): At the generation t , each individual \mathbf{x} is used to produce an offspring by using genetic operators [17]. Afterwards, an offspring population O_t is generated.

Step 3 (Selection with a dynamic infeasibility allocation mechanism): Combining the parent population P_t with its offspring population O_t as M_t , the next step is to select the best N members from the combined population M_t . To handle the constraints of problem (9) effectively, how to utilize infeasible individuals is a significant issue. Hereby, a constraint-handling technique with a dynamic infeasibility allocation mechanism is proposed as shown Algorithm 1.

To maintain the population with the same size in each generation, each sub-population must select the best s_i individuals for itself (see lines 5 - 18). This will encounter two scenarios:

- 1) When $\|\Omega_i\|$ is smaller than s_i , all the individuals in Ω_i with $s_i - \|\Omega_i\|$ randomly selected individuals are stored into Ω_i (see lines 5 - 7).
- 2) Otherwise, the δ best individuals regarding to the constraint violations are selected into Ω_i (see lines 10 - 14).

Note that when δ is still smaller than s_i , $s_i - \delta$ individuals from the rest of the population are selected into Ω_i in terms of an achievement scalarizing function (ASF) [18] (see lines 15 - 17).

$$ASF(\mathbf{x}|\mathbf{V}) = \max_{i=1,2} \left(\frac{G_i(\mathbf{x}) - Z_i}{V_i} \right) \quad (12)$$

where $\mathbf{Z} = (Z_1, Z_2)$ with each element $Z_i = \min(G_i(\mathbf{x}))$, $G_i(\mathbf{x})$ is the i -th objective function of problem (9).

Parameter α is used to control the algorithm either towards exploring more regions or finding feasible optimal solutions by deciding how many infeasible individuals can be saved into the next generation, which is given in Eq. (13). When the current population does not have any feasible individual, α is set to 1 (see lines 10 - 12). The proposed algorithm will select the infeasible individuals with smaller constraint violations, guiding the search towards the feasible regions.

$$\alpha = \begin{cases} 1 & \xi = 0 \\ \frac{t}{\beta t^{\max}} & \text{otherwise} \end{cases} \quad (13)$$

where ξ is the ratio of the feasible individuals in the combined population M_t , t^{\max} is the maximum generation number, and β is to control the number of the generations to explore the infeasible regions. When the current population has at least a feasible individual, the proposed constraint-handling technique starts to guide the search towards the feasible regions. In other words, with the increase of α , the algorithm tends to save more individuals with smaller constraint violations in each sub-population. To accelerate the convergence of the algorithm towards the promising feasible regions, we only explore the infeasible regions in the first βt^{\max} generations.

Step 4 (Output the results): When $t < t^{\max}$, go to *Step 2*. Otherwise, output all the feasible optimal individuals in P_t .

IV. EXPERIMENTAL STUDIES

We perform the experiments to verify performance of our algorithm. Two recent constrained multi-objective evolutionary algorithms, i.e., ToP [16] and PPS [19], are used as the baseline algorithms for the purpose of performance comparisons.

- 1) Each algorithm runs 30 independent times, and stops after 3×10^4 function evaluations.
- 2) The parameters related to the terrain with an area of $200 \times 200 \times 20 \text{ m}^3$ are set as follows: $\kappa_1 = 5$, $\kappa_2 = 5$, $\kappa_3 = 1.2$, $\kappa_4 = 1$, $\kappa_5 = 3$, $\kappa_6 = 1.8$ and $\kappa_7 = 1$.
- 3) $\lambda = 6$, $d_s = 10 \text{ m}$, $h^{\min} = 2 \text{ m}$, $h^{\max} = 20 \text{ m}$, $\theta_{\max} = 2\pi/3$.
- 4) For simplicity, we consider $I = 1$ device. $p_{\text{UAV}}^H = 59.2 \text{ W}$, $\varepsilon_{\text{UAV}} = 10^{-27}$, $B_1 = 10 \text{ MHz}$, $H = 5 \text{ m}$, $g_0 = -30 \text{ dB}$, $\sigma^2 = 10^{-10} \text{ W}$, $p_1^{\text{tx}} \in [0.01, 0.2] \text{ W}$, $S_1 = 80 \text{ MB}$, $W_1 = 10 \text{ giga CPU cycles}$, $v_1 \in [1, 20] \text{ m/s}$, $f_{\text{UAV},1} \in [0.1, 1.5] \text{ GHz}$, and T_1 is set to 50 seconds.
- 5) The location of device 1 and the destination is set to (50, 30, 5) and (165, 165, 5), respectively.
- 6) Two parameters of the genetic operator \mathcal{F} and CR are set to 0.5 and 0.1 respectively, and let $\eta = 21$, $N = 100$, $M = 10$, $\beta = 0.4$.

TABLE I
THE MEAN AND STD VALUE OF IGD AND HV METRIC.
BETTER RESULTS ARE MARKED BOLD

Algorithm	IGD	HV
ToP	1.38E+03(8.91E+01)	4.93E+06(1.27E+06)
PPS	1.00E+03(3.83E+02)	4.09E+06(7.25E+05)
Our Algorithm	6.66E+02(1.81E+02)	5.81E+06(3.30E+05)

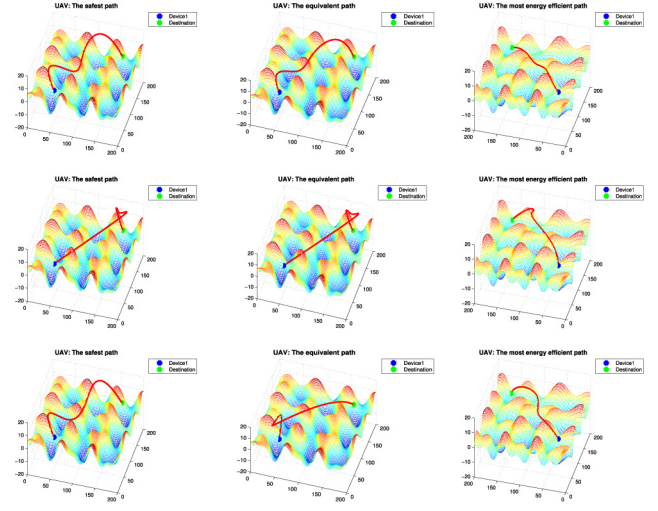


Fig. 2. Paths under three different preferences obtained by the proposed algorithm (the first row), ToP (the second row), and PPS (the third row).

IGD and HV metrics are two commonly used performance indexes to evaluate the overall performance of multi-objective evolutionary algorithms in terms of convergence and diversity of the obtained solution set [16]. A smaller IGD value indicates that an algorithm achieves better performance regarding to convergence, while a larger HV value implies that an algorithm achieves better performance regarding to both the convergence and diversity. All feasible solutions are chosen from the final obtained population to calculate the IGD and HV values. The reference point for HV metric is (5000, 5000).

Table I presents the mean and standard deviation (STD) value of IGD and HV metric among the three algorithms. Compared with PPS and ToP, our method has achieved better results of IGD and HV values. Specifically, our algorithm obtains a set of better feasible non-dominated solutions in terms of convergence and diversity, which enables the algorithm to provide more choices with a wider range of preferences.

Fig. 2 illustrates the paths derived by three preferences in terms of the median run of IGD values among the three algorithms. For the UAV, the first column is obtained by the weight vector [1, 0], which pays all the attention to the safe flight. The second column is obtained by the weight vector [0.5, 0.5], which fairly considers both for the safe flight and the energy consumption. The last column is obtained by the weight vector [0, 1], which pays all the attention to the energy consumption. We can observe that our algorithm is able to find much smoother paths on three obtained paths with different preferences compared with the other two algorithms.

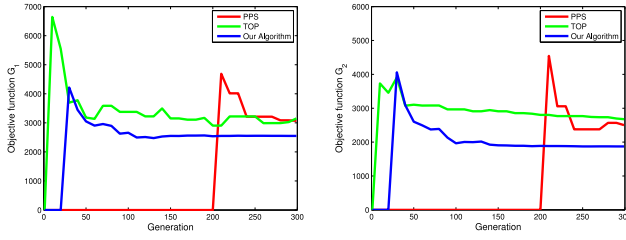


Fig. 3. Convergence graphs of the two objective functions.

 TABLE II
 INVESTIGATION OF THE SENSITIVITY OF PARAMETER β

β	IGD	HV	$G_1(\mathbf{x})$	$G_2(\mathbf{x})(J)$
0.2	6.19E+02(2.36E+02)	5.89E+06(3.26E+05)	2.98E+03	2.29E+03
0.3	6.68E+02(2.24E+02)	5.77E+06(3.77E+05)	2.97E+03	2.34E+03
0.4	6.71E+02(1.94E+02)	5.94E+06(2.82E+05)	2.90E+03	2.34E+03
0.5	6.66E+02(1.81E+02)	5.81E+06(3.30E+05)	2.97E+03	2.33E+03
0.6	7.09E+02(1.89E+02)	5.80E+06(3.45E+05)	2.98E+03	2.34E+03

 TABLE III
 INVESTIGATION OF THE DOMINATED INFEASIBLE INDIVIDUALS TRIM SCHEME

Scheme	IGD	HV	$G_1(\mathbf{x})$	$G_2(\mathbf{x})(J)$
Yes	6.66E+02(1.81E+02)	5.81E+06(3.30E+05)	2.97E+03	2.33E+03
No	7.37E+02(2.22E+02)	5.74E+06(3.67E+05)	3.02E+03	2.33E+03

Fig. 3 shows the convergence of the three algorithms in terms of the two objective functions G_1 and G_2 under the condition of the weight vector $[0.5, 0.5]$ at the median run based on IGD values respectively. Our algorithm can consistently find more feasible optimal solutions when the generation reaches βt^{\max} . Besides, our algorithm achieves lower energy consumption for the UAV compared with PPS and ToP.

Finally, we investigate sensitivity of parameter β related to the constraint-handling ability and effectiveness of the dominated infeasible individuals trim scheme. The experimental results with the mean value and STD of IGD and HV metric, and the average values of G_1 and G_2 among 30 runs are summarized in Table II and Table III respectively. Table II shows that our algorithm is not sensitive to β according to the mean values and STD of IGD and HV. The experimental results in Table III also show that the dominated infeasible individuals trim scheme (see line 2 in Algorithm 1) is helpful for the performance improvement of our algorithm.

V. CONCLUSION

We studied a CMOP for UAV-enabled MEC to simultaneously reduce the energy consumption and ensure the safe flight for the UAV. A constrained decomposition-based multi-objective evolutionary algorithm with the dynamic infeasibility allocation mechanism was designed as the solution. Both the feasible and infeasible individuals were utilized for improving the algorithm performance. Finally, extensive experimental

results were provided to demonstrate the effectiveness and efficiency of our algorithm. In our future work, we will investigate a joint computation offloading and deployment optimization scheme for the multi-UAV scenario.

REFERENCES

- [1] J. Dong, K. Ota, and M. Dong, "UAV-based real-time survivor detection system in post-disaster search and rescue operations," *IEEE J. Miniaturization Air Space Syst.*, vol. 2, no. 4, pp. 209–219, Dec. 2021.
- [2] X. Diao, J. Zheng, Y. Cai, Y. Wu, and A. Anpalagan, "Fair data allocation and trajectory optimization for UAV-assisted mobile edge computing," *IEEE Commun. Lett.*, vol. 23, no. 12, pp. 2357–2361, Dec. 2019.
- [3] H. Guo and J. Liu, "UAV-enhanced intelligent offloading for Internet of Things at the edge," *IEEE Trans. Ind. Informat.*, vol. 16, no. 4, pp. 2737–2746, Apr. 2020.
- [4] X. Xi, X. Cao, P. Yang, J. Chen, T. Quek, and D. Wu, "Joint user association and UAV location optimization for UAV-aided communications," *IEEE Wireless Commun. Lett.*, vol. 8, no. 6, pp. 1688–1691, Dec. 2019.
- [5] H. Mei, K. Yang, Q. Liu, and K. Wang, "Joint trajectory-resource optimization in UAV-enabled edge-cloud system with virtualized mobile clone," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 5906–5921, Jul. 2020.
- [6] J. Lyu, Y. Zeng, and R. Zhang, "UAV-aided offloading for cellular hotspot," *IEEE Trans. Wireless Commun.*, vol. 17, no. 6, pp. 3988–4001, Jun. 2018.
- [7] Z. Ning *et al.*, "5G-enabled UAV-to-community offloading: Joint trajectory design and task scheduling," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 11, pp. 3306–3320, Nov. 2021.
- [8] T. Wang, Y. Li, and Y. Wu, "Energy-efficient UAV assisted secure relay transmission via cooperative computation offloading," *IEEE Trans. Green Commun. Netw.*, vol. 5, no. 4, pp. 1669–1683, Dec. 2021.
- [9] H.-T. Ye, X. Kang, J. Joung, and Y.-C. Liang, "Optimization for wireless-powered IoT networks enabled by an energy-limited UAV under practical energy consumption model," *IEEE Wireless Commun. Lett.*, vol. 10, no. 3, pp. 567–571, Mar. 2021.
- [10] S. Bi, L. Huang, H. Wang, and Y.-J. A. Zhang, "Lyapunov-guided deep reinforcement learning for stable online computation offloading in mobile-edge computing networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 11, pp. 7519–7537, Nov. 2021.
- [11] Y. Wu, X. Guan, W. Yang, and Q. Wu, "UAV swarm communication under malicious jamming: Joint trajectory and clustering design," *IEEE Wireless Commun. Lett.*, vol. 10, no. 10, pp. 2264–2268, Oct. 2021.
- [12] T. Ma, H. Zhou, B. Qian, and A. Fu, "A large-scale clustering and 3D trajectory optimization approach for UAV swarms," *Sci. China Inf. Sci.*, vol. 64, pp. 1–16, Apr. 2021.
- [13] Y. Zeng and R. Zhang, "Energy-efficient UAV communication with trajectory optimization," *IEEE Trans. Wireless Commun.*, vol. 16, no. 6, pp. 3747–3760, Jun. 2017.
- [14] X. Yu, C. Li, and J. Zhou, "A constrained differential evolution algorithm to solve UAV path planning in disaster scenarios," *Knowl. Based Syst.*, vol. 204, Sep. 2020, Art. no. 106209.
- [15] X. Yu, C. Li, and G. G. Yen, "A knee-guided differential evolution algorithm for unmanned aerial vehicle path planning in disaster management," *Appl. Soft Comput.*, vol. 98, Jan. 2020, Art. no. 106857.
- [16] Z.-Z. Liu and Y. Wang, "Handling constrained multiobjective optimization problems with constraints in both the decision and objective spaces," *IEEE Trans. Evol. Comput.*, vol. 23, no. 5, pp. 870–884, Oct. 2019.
- [17] C. Peng, H.-L. Liu, and E. D. Goodman, "Handling multi-objective optimization problems with unbalanced constraints and their effects on evolutionary algorithm performance," *Swarm Evol. Comput.*, vol. 55, Jun. 2020, Art. no. 100676.
- [18] A. P. Wierzbicki, "The use of reference objectives in multiobjective optimization," in *Multiple Criteria Decision Making Theory and Application*. Berlin, Germany: Springer, 1980 pp. 468–486.
- [19] Z. Fan *et al.*, "Push and pull search for solving constrained multi-objective optimization problems," *Swarm Evol. Comput.*, vol. 44, pp. 665–679, Feb. 2019.

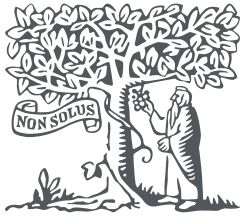


Volume 118, March 2022 ISSN 1568-4946

Applied Soft Computing

Available online at www.sciencedirect.com

ScienceDirect



Applied Soft Computing

Table of contents

Volume 118, March 2022

Editorial Board

Article 108664

Customer-oriented product design using an integrated neutrosophic AHP & DEMATEL & QFD methodology

Ali Karasan, Esra Ilbahar, Selcuk Cebi, Cengiz Kahraman
Article 108445

Optimization of energy-efficient open shop scheduling with an adaptive multi-objective differential evolution algorithm

Lijun He, Yulian Cao, Wenfeng Li, Jingjing Cao, Lingchong Zhong
Article 108459

Entropy measure for a fuzzy relation and its application in attribute reduction for heterogeneous data

Liangdong Qu, Jiali He, Gangqiang Zhang, Ningxin Xie
Article 108455

Discrete sparrow search algorithm for symmetric traveling salesman problem

Zhen Zhang, Yang Han
Article 108469

CMML: Combined metaheuristic-machine learning for adaptable routing in clustered wireless sensor networks

Hojjatollah Esmaili, Behrouz Minaei Bidgoli, Vesal Hakami
Article 108477

An interval type-2 fuzzy sets based Delphi approach to evaluate site selection indicators of sustainable vehicle shredding facilities

Muhammet Deveci, Vladimir Simic, Selman Karagoz, Jurgita Antucheviciene
Article 108465

SCSTCF: Spatial-Channel Selection and Temporal Regularized Correlation Filters for visual tracking

Jianming Zhang, Wenjun Feng, Tingyu Yuan, Jin Wang, Arun Kumar Sangaiah
Article 108485

An automata algorithm for generating trusted graphs in online social networks

Nina Fatehi, Hadi Shahriar Shahhoseini, Jesse Wei, Ching-Ter Chang
Article 108475

Multi-objective optimization of hexahedral pyramid crash box using MOEA/D-DAE algorithm

Weiwei Wang, Shijuan Dai, Wanzhong Zhao, Chunyan Wang
Article 108481

A triangular hashing learning approach for olfactory EEG signal recognition

Hui-Rang Hou, Qing-Hao Meng, Biao Sun
Article 108471

Impersonation fraud detection on building access control systems: An approach based on anomalous social and spatio-temporal behaviors

Gabriel Mariano de Castro Silva, Jaime Simão Sichman
Article 108310

A biased genetic algorithm hybridized with VNS for the two-dimensional knapsack packing problem with defects

Qiang Luo, Yunqing Rao, Xiaoqiang Guo, Bing Du
Article 108479

A simulated annealing based heuristic for a location-routing problem with two-dimensional loading constraints

Kamyla Maria Ferreira, Thiago Alves de Queiroz
Article 108443

Multi-dimensional recurrent neural network for remaining useful life prediction under variable operating conditions and multiple fault modes

Yiwei Cheng, Chao Wang, Jun Wu, Haiping Zhu, C.K.M. Lee
Article 108507

Genetic programming for feature extraction and construction in image classification

Qinglan Fan, Ying Bi, Bing Xue, Mengjie Zhang
Article 108509

Cross-network representation learning for anchor users on multiplex heterogeneous social network

Amina Amara, Mohamed Ali Hadj Taieb, Mohamed Ben Aouicha
Article 108461

Extended rule-based opinion target extraction with a novel text pre-processing method and ensemble learning

Kürşat Mustafa Karaoğlu, Oğuz Fındık
Article 108524

A multimodal approach to chaotic renewable energy prediction using meteorological and historical information

Hui Hwang Goh, Ronghui He, Dongdong Zhang, Hui Liu, ... Kai Chen Goh
Article 108487

A fuzzy mathematical model for tumor growth pattern using generalized Hukuhara derivative and its numerical analysis

Rubeena Khaliq, Pervaiz Iqbal, Shahid Ahmad Bhat, Aadil Rashid Sheergojri
Article 108467

Solving time varying many-objective TSP with dynamic θ -NSGA-III algorithm


Rashi Gupta, Satyasai Jagannath Nanda
Article 108493

A study of multi-objective restricted multi-item fixed charge transportation problem considering different types of demands

Amiya Biswas, Leopoldo Eduardo Cárdenas-Barrón, Ali Akbar Shaikh, Avijit Duary, Armando Céspedes-Mota
Article 108501

Community-based anomaly detection using spectral graph filtering

Rodrigo Francisquini, Ana Carolina Lorena, Mariá C.V. Nascimento
Article 108489

A fast parameter optimization approach based on the inter-cluster induced distance in the feature space for support vector machines 

Jiapeng Wang, Jiaxiang Luo
Article 108519

Cybersecurity of multi-cloud healthcare systems: A hierarchical deep learning approach

Lav Gupta, Tara Salman, Ali Ghubaish, Devrim Unal, ... Raj Jain
Article 108439

Randomization-based machine learning in renewable energy prediction problems: Critical literature review, new results and perspectives

J. Del Ser, D. Casillas-Perez, L. Cornejo-Bueno, L. Prieto-Godino, ... S. Salcedo-Sanz
Article 108526

A decomposition-based constrained multi-objective evolutionary algorithm with a local infeasibility utilization mechanism for UAV path planning

Chaoda Peng, Shaojian Qiu
Article 108495

A novel improved SMA with quasi reflection operator: Performance analysis, application to the image segmentation problem of Covid-19 chest X-ray images

Sukanta Nama
Article 108483

Session-based social and dependency-aware software recommendation

Dengcheng Yan, Tianyi Tang, Wenxin Xie, Yiwen Zhang, Qiang He
Article 108463

A stacking neuro-fuzzy framework to forecast runoff from distributed meteorological stations

Marvin Querales, Rodrigo Salas, Yerel Morales, Héctor Allende-Cid, Harvey Rosas
Article 108535

Churn prediction in digital game-based learning using data mining techniques: Logistic regression, decision tree, and random forest

Mai Kiguchi, Waddah Saeed, Imran Medi
Article 108491

Human-like motion planning of autonomous vehicle based on probabilistic trajectory prediction

Peng Li, Xiaofei Pei, Zhenfu Chen, Xingzhen Zhou, Jie Xu
Article 108499

Scene recognition using multiple representation network

Chaowei Lin, Feifei Lee, Lin Xie, Jiawei Cai, ... Qiu Chen
Article 108530

Advanced traffic congestion early warning system based on traffic flow forecasting and extenics evaluation

Ping Jiang, Zhenkun Liu, Lifang Zhang, Jianzhou Wang
Article 108544

Medical image fusion via discrete stationary wavelet transform and an enhanced radial basis function neural network

Zhen Chao, Xingguang Duan, Shuangfu Jia, Xuejun Guo, ... Fucang Jia
Article 108542

Tree trunk texture classification using multi-scale statistical macro binary patterns and CNN

Safia Boudra, Itheri Yahiaoui, Ali Behloul
Article 108473

Applying the quantum approximate optimization algorithm to the minimum vertex cover problem

Y.J. Zhang, X.D. Mu, X.W. Liu, X.Y. Wang, ... C. Dong
Article 108554

A statistical feature data mining framework for constructing scholars’ career trajectories in academic data

Zhou Shao, Sha Yuan, Jing Xu, Yongli Wang
Article 108550

Carbon price forecasting system based on error correction and divide-conquer strategies

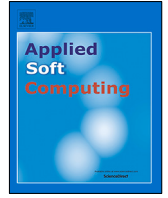
Xinsong Niu, Jianzhou Wang, Lifang Zhang
Article 107935

Soft computing for recommender systems and sentiment analysis

Lorenzo Malandri, Carlos Porcel, Frank Xing, Jesus Serrano-Guerrero, Erik Cambria
Article 108246

Corrigendum to “An advanced YOLOv3 method for small-scale road object detection” [Appl. Soft Comput. 112 (2021) 107846]

Kun Wang, Maozhen Liu, Zhaojun Ye
Article 108289



A decomposition-based constrained multi-objective evolutionary algorithm with a local infeasibility utilization mechanism for UAV path planning

Chaoda Peng, Shaojian Qiu *

College of Mathematics and Informatics, South China Agricultural University, Guangzhou, China

ARTICLE INFO

Article history:

Received 19 July 2021

Received in revised form 25 October 2021

Accepted 17 January 2022

Available online 29 January 2022

Keywords:

Multi-objective optimization

UAV path planning

Constraint-handling technique

Evolutionary algorithm

ABSTRACT

Unmanned Aerial Vehicle (UAV) path planning problems can be treated as constrained multi-objective optimization problems, which often have complicated constraints in real-world scenarios. Algorithms for solving them require a powerful constraint-handling technique to utilize infeasible information. However, this has seldom been explored in this field. To remedy this issue, this paper proposes a decomposition-based constrained multi-objective evolutionary algorithm (M2M-DW) with a local infeasibility utilization mechanism for UAV path planning. Therein, M2M-DW is adopted as a solution optimizer since it can utilize infeasible individuals. However, this may result in poor performance due to the arbitrary use of infeasible individuals. To solve this issue, a local infeasibility utilization mechanism is proposed to effectively utilize the infeasible information. Besides, an improved mutation scheme is designed to further explore the promising regions. Experimental studies are conducted on three sets of UAV path planning problems with different difficulties, and the results highlight the effectiveness of the proposed algorithm in terms of reliability and stability in finding a set of feasible optimal solutions.

© 2022 Elsevier B.V. All rights reserved.

1. Introduction

Unmanned aerial vehicles (UAVs) have been applied to a wide range of real-world applications, such as search and rescue work, crop dusting and agricultural surveillance, traffic monitoring, network provisioning, military use and so on [1–3]. For example, in rush-hour time, the number of mobile devices may rapidly increase and cause network congestion and high delay, and UAVs carrying base station with computing resources can be appointed to the place with an ability to help alleviate network overload and execution latency. For such an application, how to fly a UAV to the destination without any collision is one of the critical issues in drone technologies, i.e., UAV path planning.

In general, UAV path planning problem can be modeled as a constrained multi-objective optimization problem (CMOP) as follows:

$$\begin{aligned} \min F(\mathbf{x}) &= (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x})) \\ \text{s.t.} \quad & c_i(\mathbf{x}) \geq 0 \quad i = 1, 2, \dots, q \\ & h_i(\mathbf{x}) = 0 \quad i = q + 1, \dots, l \\ & \mathbf{x} = (x_1, x_2, \dots, x_n) \in D \end{aligned} \quad (1)$$

where $\mathbf{x} = (x_1, x_2, \dots, x_n)$ is an n -dimensional decision vector, $F(\mathbf{x})$ is an m -dimensional objective function. $c_i(\mathbf{x})$ and $h_i(\mathbf{x})$ are the inequality and equality constraint respectively, l is the number of the constraints.

In this paper, the constraint violation of a solution \mathbf{x} on the i th constraint is calculated by the following equation [4]:

$$G_i(\mathbf{x}) = \begin{cases} \min\{0, c_i(\mathbf{x})\} & 1 \leq i \leq q \\ \min\{0, |h_i(\mathbf{x})|\} & q + 1 \leq i \leq l \end{cases} \quad (2)$$

The constraint violation of a solution \mathbf{x} is calculated as:

$$G(\mathbf{x}) = \sum_{i=1}^l |G_i(\mathbf{x})| \quad (3)$$

UAV path planning problems can be treated as constrained single-objective optimization problems (CSOPs) [5–7] or CMOPs [8–12]. When taking UAV path planning problem as a CSOP, most algorithms construct an objective function with a weighted-sum method, which incorporates requirements and constraints related to the UAVs and their flight path. In [6], a safety-enhanced UAV path planning with spherical vector-based particle swarm optimization algorithm was proposed. It firstly converts the path planning problem into an objective function by weightedly summing the path optimality, safety and feasibility constraints with a weight vector. Then a particle swarm algorithm based on a

* Corresponding author.

E-mail address: qiushaojian@scau.edu.cn (S. Qiu).

spherical vector is designed to solve the problem. In [7], the UAV path planning problem is also designed as a weighted-sum objective function by considering traveling distance, safety, flight height, flight angle and limited UAV slope. However, determining a weight vector is a time-consuming task, especially if there are many requirements and constraints. It is hard to find an appropriate weight vector based on the preference of the decision-makers. While taking it as a CMOP, the advantage is that it can obtain a set of optimal solutions with various preferences in a single run.

Constrained multi-objective evolutionary algorithms (CMOEAs) have stimulated the interest of many researchers during the past few decades [4,13–31]. They can be divided into five groups based on the constraint-handling techniques they use—i.e., penalty functions [13,14], methods based on the preference of feasible solutions over infeasible solutions [4,15–17], methods based on a repair operator [18–20], methods based on preserving informative infeasible solutions [22–25] and methods based on temporarily disregarding constraint violations [26–28]. They are introduced as follows:

- (1) Penalty function is one of the most popular constraint-handling techniques. Nevertheless, the main drawback is that the penalty factor is problem-dependent. In [32], a constraint-handling technique based on an adaptive function and a distance measure was proposed to handle with CMOPs. Therein, the number of the feasible solutions is used to decide the search either towards finding more feasible solutions or finding optimal solutions. The paper is easy to implement and does not need any parameter tuning. It however uses the same penalty factor to punish the individuals in the whole population, which may be too large or small for some search regions.
- (2) To avoid setting penalty factor, methods based on the preference of feasible solutions over infeasible solutions are used to deal with CMOPs. They prefer feasible solutions, which may easily converge to locally optimal regions, especially there are more than one feasible regions. In [4], a constraint-domination principle (CDP) was used to select solutions with feasibility preference. If an individual \mathbf{x} is said to constraint-dominate another individual \mathbf{y} if \mathbf{x} and \mathbf{y} are feasible individuals and \mathbf{x} dominates \mathbf{y} , or \mathbf{x} has a smaller constraint violation than \mathbf{y} .
- (3) Another methods based on a repairing operator are designed to avoid setting penalty factor. They try to fix infeasible solutions to feasible solutions. However, for some problems with small feasibility ratios, finding a feasible solution itself is a problem. In [19], a Pareto descent repairing operator was designed to drive the infeasible towards feasible regions. It studies the guidelines for a method to handle constraints and then explains the concepts and calculations to meet the guidelines.
- (4) Methods based on preserving infeasible solutions have caught the attention of many researchers since the utilization of the infeasible solutions with better objective values is beneficial to the evolution process. Peng et al. proposed a CMOEA based on the directed weight vectors (M2M-DW) [23]. It designs two types of weight vectors, i.e., feasible and infeasible weight vectors. The infeasible weight vectors are used to select a set of infeasible individuals which participate in the genetic operations. Besides, they are changed dynamically to select the infeasible individuals with better objective values and smaller constraint violations along with the evolution process. The experimental results highlight its effectiveness.
- (5) The last type of constraint-handling techniques is based on temporarily disregarding constraint violations during the

evolution process. Fan et al. proposed a CMOEA based on push and pull search mechanism (PPS) [26]. It divides the evolution process into two phases. At the first phase, it does not consider constraint violations while takes it into account to direct the search to the feasible regions at the second phase. But it may not guide search back to the feasible regions, resulting in poor performance on some CMOPs with complicated constraints.

UAV path planning problems are highly constrained in complicated scenarios, and algorithms for solving must have an ability to handle constraints with an appropriate manner. However, to the best of our knowledge, this has seldom been explored in the community of constrained multi-objective UAV path planning problems [8,10,12,33,34]. In [8], a 3-D offline path planner for UAVs by using NSGA-II [15] with CDP was proposed. The proposed algorithm can generate a number of feasible paths, which allows decision-makers to select a path based on their interests. Nevertheless, the constraint-handling technique prefers feasible solutions over the infeasible solutions, which may produce local optimal solutions. In [34], the UAV path planning problem was modeled as a constraint satisfaction problem (CSP). Then a multi-objective evolutionary algorithm based on the branch and bound method is used to find feasible optimal paths. These algorithms do not take advantages of the useful infeasible individuals to improve their performance when solving UAV path planning problems.

Motivated by the above considerations, M2M-DW with a local infeasibility utilization method is proposed for solving UAV path planning problems in this paper. At first, three sets of UAV path planning problems with different difficulties are presented by minimizing the traveling distance and the risk of a UAV subjected to four constraints, including the minimum flight altitude, maximum flight altitude, minimum flight angle and minimum/maximum flight scope. To effectively handle constraints, M2M-DW is adopted since it has an ability to utilize the useful infeasible individuals. However, its manner of utilizing infeasible solutions is arbitrary, which may result in poor performance. To solve this issue, a local infeasible utilization method is proposed to make good use of infeasible solutions, aiming at guiding the search to the promising regions. Besides, an improved mutation scheme is designed to enhance the search ability of the proposed algorithm. The main contributions of this paper are summarized as follows:

- Three sets of UAV path planning problems with different difficulties are conducted to verify the performance of the algorithms related to their constraint-handling ability.
- The proposed constraint-handling technique based on a local infeasibility utilization not only maintains a set of well-distributed infeasible non-domination solutions, but also makes better use of the informative infeasible solutions, improving its performance in terms of finding a set of well-distributed Pareto optimal solutions.
- An improved mutation scheme effectively enhances the search ability of the proposed algorithm.
- The effectiveness of the proposed algorithm and the sensitivity of the parameters have been experimentally investigated.

The remainder of this paper is organized as follows. Section 2 reviews the literature in the field of UAV path planning. Section 3 presents the details of the UAV path planning problem. Section 4 introduces the proposed algorithm thoroughly. Section 5 shows a series of experiments. Finally, the conclusion is drawn in Section 6.

2. Related work

Algorithms for solving UAV path planning problems can be roughly grouped into five categories, including methods based on geometric search [35–37], methods based on traditional mathematics [38,39], methods based on heuristic algorithms [40,41], methods based on the artificial potential field [42,43], and methods based on artificial intelligence algorithms [5–12,33].

The first category is the methods based on geometric search. This type of methods usually divide the whole region into a number of connected regions, and each of them forms a vertex of the graph which the UAV path passes through. In [35], the paper presented an end-to-end solution in the battlefield scenario for multiple UAVs to strike a number of known targets. The Voronoi diagram is used to generate a graph as the input for the algorithms to find the best path. A sampling-based path planning solver for UAV collision avoidance was proposed in [36]. It uses a method based on the closed-loop rapidly-exploring random tree to produce a search graph. The main drawback of these methods is that they are unsuitable for UAV path planning problems with constraints.

The second category is the methods based on traditional mathematics (such as dynamic programming algorithm, linear programming algorithm). A fast-dynamic mixed integer linear programming method was proposed for efficient UAV path planning in various flight formations [38]. An elaborate construction of constraint equations is designed to enforce the formation to visit pre-defined waypoints and avoid collisions with obstacles. In [39], the UAV path planning problem for search and rescue scenarios was modeled by using the Markov decision process and solved by dynamic programming algorithms. These methods need gradient information for the problems, which limits their scalable capacity.

The third category is the methods based on heuristic algorithms (such as A^* [40], D^* [41]). In [40], a minimum-time trajectory planning algorithm under intermittent measurements was proposed. It presents a robust perception-aware bi-directional A^* algorithm for differentially flat systems and a derivative-free Kalman filter is used to approximate the belief dynamics in the flat space. The main shortcoming of these methods is that they easily obtain the local optimal path when there are some obstacles in the way of from the start point to the end point. Besides, they may suffer from slow speed and execution of running the algorithms if the search space increases.

The fourth category is the methods based on the artificial potential field that takes a UAV as a particle moving under the influence of a potential field related to the objective function and obstacles. In [42], a UAV path planner with the artificial potential field updated by the optimal control theory was proposed. It firstly remodels the UAV path planning problem based on the artificial potential field into a constrained optimization problem, then the problem is translated into an unconstrained optimization problem, which is later reformed into an optimal control problem. The UAV path planning problem is solved with the help of the optimal control method. The experimental results show that the proposed method finds a shorter and smoother path than the other compared algorithms based on the artificial potential field. In [43], a collision-free trajectory planning algorithm for multi-rotor UAVs in a wind condition based on the modified potential field was proposed. The modified potential field is used to avoid collision for multi-rotor UAVs under uncertain environments. Usually, this type of methods have difficulty to cope with local minima since they do not take the optimality of the solution into account.

The last category is the methods based on artificial intelligence algorithms (such as genetic algorithm, particle swarm optimization algorithm, pigeon-inspired algorithm, gray wolf optimizer

and so on). They have attracted the attention of many researchers in the field of UAV path planning due to their natural advantages that they are cable of handling constraints and can search the global optimal solutions in complicated scenarios. In [12], the authors considered the path planning problem as a CMOP in a three-dimensional terrain disaster scenario. To solve this problem, an adaptive selection mutation constrained differential evolution algorithm is proposed to guide the search direction to the promising regions. However, it uses CDP as the constraint-handling technique, which may produce local optimal solutions. In [10], a multi-objective pigeon-inspired algorithm to UAV distributed flocking among obstacles was proposed. The constraints related to flight safety are divided into the hard constraints which must be satisfied and soft constraints which will be optimized. When an individual does not meet the hard constraints, it will be reproduced within the search space. UAV path planning under the complex environments usually contains many constraints, and the algorithm may not work on this type of problems since generating an individual that satisfies the hard constraints is a problem. Although these algorithms achieve competitive results in UAV path planning problems, their performance however remains an issue for complicated scenarios with more constraints.

3. UAV path planning problems

3.1. Representation of the terrain and UAV path

Prior to building the UAV path planning model, the flying environment must be given, which is often represented by a 3-D model by using Eq. (4) [7].

$$z(x, y) = \sin(y + \gamma_1) + \gamma_2 \sin(x) + \gamma_3 \cos(y) + \gamma_4 \cos(y) + \gamma_5 \cos(\gamma_6 \sqrt{x^2 + y^2}) + \gamma_7 \sin(\gamma_7 \sqrt{x^2 + y^2}) \quad (4)$$

Where $\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5, \gamma_6$ and γ_7 are experimentally studied constants that a combination of them produces a smooth surface simulating a terrain with valleys and mountains.

In general, B-Spline curve is used to model the UAV path since it is defined only by a few number of control points that can represent a complicated path [8]. Moreover, it is smooth at least up to the first order and easy to implement. In this paper, the control points are taken as variables for the proposed algorithm.

Suppose $C_0(x_0, y_0, z_0), C_1(x_1, y_1, z_1), \dots, C_{r-1}(x_{r-1}, y_{r-1}, z_{r-1})$ are r control points, and then the corresponding B-Spline curve, i.e., the UAV path, is generated with the r control points. The UAV path composes of $(s + 1)$ path points $B_0 = (x'_0, y'_0, z'_0), B_1 = (x'_1, y'_1, z'_1), \dots, B_s = (x'_s, y'_s, z'_s)$, and it is generated as follows:

$$\begin{cases} x'_i = \sum_{j=0}^{r-1} B_{j,d}(t_i) \cdot x_j \\ y'_i = \sum_{j=0}^{r-1} B_{j,d}(t_i) \cdot y_j \\ z'_i = \sum_{j=0}^{r-1} B_{j,d}(t_i) \cdot z_j \end{cases} \quad , \quad i = 0, 1, 2, \dots, s \quad (5)$$

Where d is its degree, which reflects the smoothness of the curve. Larger value of d implies a smoother curve. $B_{j,d}(t_i)$ is the blending function of the curve and defined recursively in terms of a non-decreasing sequence of real numbers, i.e., a knot vector

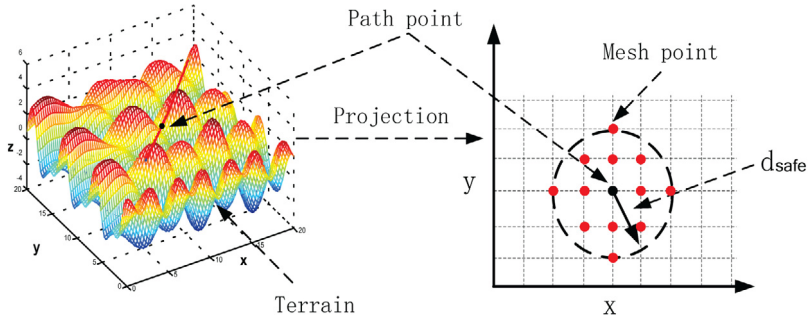


Fig. 1. Projection of a path point and mesh points to the horizontal plane.

$U = \{u_0, u_1, \dots, u_m\}$. The uniform non-periodic knot vector is one of the most commonly used ones as follows:

$$\begin{cases} u_j = 0 & j \leq d \\ u_j = j - d & d < j \leq s - 1 \\ u_j = s - d & s \leq j \end{cases} \quad (6)$$

Then the blending function $B_{j,d}(t_i)$ is computed as follows:

$$B_{j,1}(t_i) = \begin{cases} 1 & u_j \leq t_i < u_{j+1} \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

$$B_{j,d}(t_i) = \frac{t_i - u_j}{u_{j+d-1} - u_j} \cdot B_{j,d-1}(t_i) + \frac{u_{j+d} - t_i}{u_{j+d} - u_{j+1}} \cdot B_{j+1,d-1}(t_i) \quad (8)$$

Parameter t_i is a value in the range between 0 and $r + d$ with a constant step, therefore providing a set of discrete points of the B-Spline curve.

3.2. Model of UAV path planning

This paper considers the UAV path planning problem as a two-objective problem with four constraints, which is presented as follows.

The first objective function is the overall length of the path:

$$f_1 = \varepsilon \sum_{i=1}^{s-1} \|B_{i+1} - B_i\| \quad (9)$$

Where ε is a factor which makes two objective functions have a similar scale.

The second objective is to ensure a collision-free path. The objective function is the same as in [12]. It firstly finds the nearest mesh points with a given path point. To achieve this purpose, it projects the path point and all the mesh points of a terrain into the horizontal plane. Then the nearest mesh points that are within the range of the safe distance (denoted as d_{safe}) are obtained as shown in Fig. 1. The second objective is calculated as follows:

$$f_2 = \sum_{j=0}^s \sum_{i=1}^{np} \frac{1}{\left(\frac{d_{i,j}}{d_{safe}}\right)^2} \quad (10)$$

Where np is the number of the nearest mesh points. $d_{i,j}$ is the distance between the j th path point and i th mesh point. s is related to the total number of the path points in Eq. (5).

The first constraint is that a UAV must fly above the minimum flight altitude, it is given as follows:

$$h_1 = \sum_{i=1}^s h_{1i} = 0 \quad (11)$$

$$h_{1i} = \begin{cases} d_{min}^i & d_{min}^i < 0 \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

Where $d_{min}^i = z'_i - z(x'_i, y'_i) - h_{min}$, h_{min} is the minimum flight altitude, and $(x'_i, y'_i, z(x'_i, y'_i))$ is the i th mesh point regarding to Eq. (4).

The second constraint is the turning angle of a UAV along the path that cannot surpass its maximum turning angle θ_{max} :

$$h_2 = \sum_{i=2}^{s-1} h_{2i} = 0 \quad (13)$$

$$h_{2i} = \begin{cases} \Delta\theta_i & \Delta\theta_i < 0 \\ 0 & \text{otherwise} \end{cases} \quad (14)$$

Where $\Delta\theta_i = \theta_{max} - \theta(\mathbf{B}_{i,i-1}, \mathbf{B}_{i+1,i})$, $\mathbf{B}_{i+1,i}$ is the vector from point B_{i+1} to point B_i , $\theta(\mathbf{B}_{i,i-1}, \mathbf{B}_{i+1,i})$ is the angle of two vectors $\mathbf{B}_{i,i-1}$ and $\mathbf{B}_{i+1,i}$ as follows:

$$\theta(\mathbf{B}_{i,i-1}, \mathbf{B}_{i+1,i}) = \cos^{-1} \left(\frac{\mathbf{B}_{i,i-1} \cdot \mathbf{B}_{i+1,i}}{\|\mathbf{B}_{i,i-1}\| \|\mathbf{B}_{i+1,i}\|} \right) \quad (15)$$

The third constraint is the limited upper flight altitude:

$$h_3 = \sum_{i=1}^s h_{3i} = 0 \quad (16)$$

$$h_{3i} = \begin{cases} d_{max}^i & d_{max}^i < 0 \\ 0 & \text{otherwise} \end{cases} \quad (17)$$

Where $d_{max}^i = h_{max} - z'_i$, h_{max} is the maximum flight altitude.

The fourth constraint is related to UAV manoeuvrability, i.e., the minimum climbing slope S_{min}^i and maximum climbing slope S_{max}^i [44], it is defined as follows:

$$h_4 = \sum_{i=1}^{s-1} h_{4i} = 0 \quad (18)$$

$$h_{4i} = \begin{cases} 0 & S_i \in (S_{min}^i, S_{max}^i) \\ S_i - S_{min}^i & S_i \leq S_{min}^i \\ S_{max}^i - S_i & S_i \geq S_{max}^i \end{cases} \quad (19)$$

Where,

$$S_i = \frac{z'_{i+1} - z'_i}{\sqrt{(x'_{i+1} - x'_i)^2 + (y'_{i+1} - y'_i)^2}},$$

$$S_{min}^i = 2.5063 \times 10^{-9} z_i'^2 - 6.3014 \times 10^{-6} z_i' - 0.3257, \\ S_{max}^i = -1.5377 \times 10^{-10} z_i'^2 - 2.6997 \times 10^{-5} z_i' + 0.4211.$$

3.3. Three sets of UAV path planning problems

In this paper, three sets of UAV path planning problems with different difficulties are considered by varying the parameters of the terrain and UAV path planning model. The parameters of the three sets of UAV path planning problems are listed in Table 1.

For the first set of the problems, i.e., UAV1-1 and UAV1-2, it considers a simple path planning which the UAV should fly above the terrain. As for the second set of the problems, i.e., UAV2-1 and

Table 1

The parameters for three sets of UAV path planning.

Problem	$(\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5, \gamma_6, \gamma_7)$	ε	r	d_{safe}	θ_{max}	h_{min}	h_{max}	start point	end point	h_1	h_2	h_3	h_4
UAV1-1	(1,2,0,3,0,0,2,2,0,1)	5	4	1.2	$\frac{\pi}{4}$	2	16	(1,4,10)	(17,18,12)	✓	✓	✗	✗
UAV1-2	(1,2,0,3,0,0,2,2,0,1)	5	4	1.2	$\frac{\pi}{4}$	2	16	(1,4,10)	(17,18,12)	✓	✓	✓	✓
UAV2-1	(1,0,5,1,0,1,2,0,1)	20	5	1.2	$\frac{\pi}{4}$	2	5.5	(4,3,4)	(17,16,4)	✓	✓	✗	✗
UAV2-2	(1,0,5,1,0,1,2,0,1)	20	5	1.2	$\frac{\pi}{4}$	2	5.5	(4,3,4)	(17,16,4)	✓	✓	✓	✓
UAV3-1	(5,5,1,2,1,3,1,8,1)	15	6	1.2	$\frac{\pi}{4}$	2	10	(4,3,5)	(17,19,5)	✓	✓	✗	✗
UAV3-2	(5,5,1,2,1,3,1,8,1)	15	6	1.2	$\frac{\pi}{4}$	2	10	(4,3,5)	(17,19,5)	✓	✓	✓	✓

UAV2-2, it considers a scenario that a UAV should avoid collisions near the start and end point. While for UAV3-1 and UAV3-2, both of them need a UAV to avoid collisions along their flight path, which makes them more difficult for the path planners to find a feasible path. Besides, from the table, it can be observed that the second problem of each set has two more constraints than the first problem, which makes it more dependent on the ability of handling constraints.

4. The proposed algorithm

4.1. The proposed framework

To effectively handle UAV path planning problems, an algorithm must have an ability to make good use of the infeasible individuals with better objective values. M2M-DW [23] is adopted to serve this purpose. However, the manner of utilizing infeasible solutions is arbitrary, a local infeasibility utilization mechanism is designed in this paper to further make better use of infeasible solutions. The detail of the proposed local infeasibility utilization mechanism can be found in Algorithm 2. Besides, this paper designs an improved mutation scheme to enhance the search ability of the proposed algorithm. The pseudo-code¹ is given in Algorithm 1.

Algorithm 1: The framework of the proposed algorithm

```

1: Generate  $N_1$  infeasible weight vectors,  $N_2$  feasible weight vectors and  $K$  unit center vectors.
2: Initialize population  $Q_0$  with a size of  $N$  ( $N = N_1 + N_2$ ) and the ideal point  $\mathbf{z} = (z_1, z_2, \dots, z_m)$  by  $z_i = \min(f_i(\mathbf{x}_1), f_i(\mathbf{x}_2), \dots, f_i(\mathbf{x}_N))$ . Set the stopping criterion and current generation  $t = 1$ .
3: Decompose  $(N_1 + N_2)$  weight vectors into  $K$  sub-regions  $\Omega_i$  ( $i = 1, 2, \dots, K$ ) by using Eq. (21). Set  $s_i = \|\Omega_i\|$  as the size of the  $i$ th sub-region.
4: while the stopping criterion is not satisfied do
5:   for  $i = 1$  to  $K$  do
6:     for  $j = 1 : \|\Omega_i\|$  do
7:       if  $\text{rand} < 0.9$  then
8:         Select three individuals  $\mathbf{x}'$ ,  $\mathbf{x}''$  and  $\mathbf{x}'''$  according to Algorithm 2.
9:       else
10:        Randomly select three individuals  $\mathbf{x}'$ ,  $\mathbf{x}''$  and  $\mathbf{x}'''$  from the current population.
11:      end if
12:      Generate an offspring  $\mathbf{y}$  with  $\mathbf{x}_j$ ,  $\mathbf{x}'$ ,  $\mathbf{x}''$  and  $\mathbf{x}'''$  according to Eqs. (22) and (24).
13:       $M_t = M_t \cup \{\mathbf{y}\}$ .
14:    end for
15:  end for
16:  Update  $\mathbf{z}$  with each  $\mathbf{x}_j$  in  $M_t$ :  $z_i = \min(z_i, f_i(\mathbf{x}_j))$ .
17:  Set the combined population  $C_t = Q_t \cup M_t$ , and reset  $\Omega_1, \dots, \Omega_K$  by using  $C_t$  according to Eq. (21).
18:   $Q_{t+1} = \bigcup_{i=1}^K P_i$ .
19:  Update the infeasible weight vectors according to Eq. (28).
20:  Update the current generation  $t = t + 1$ .
21: end while
22: Output all the feasible non-domination individuals in  $Q_t$ .

```

At first, M2M-DW transforms an m -dimensional CMOP into an $(m+1)$ -dimensional unconstrained multi-objective optimization problem by using Eq. (20). As shown in Fig. 2, the proposed algorithm generates N_1 infeasible weight vectors on infeasible region,

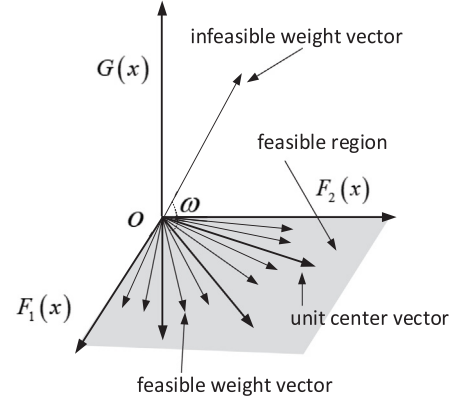


Fig. 2. Illustration of the feasible, infeasible and unit center weight vector.

i.e., the space above the horizontal plane F_1OF_2 , N_2 feasible weight vectors and K unit center vectors on feasible region, i.e., the horizontal plane F_1OF_2 (see line 1 in Algorithm 1). They are evenly generated from a hyperplane, and the details can be found in [23]. The N_1 infeasible weight vectors aim to preserve a set of infeasible solutions during the evolution process, providing useful genetic information about the optimal direction of improvement for the objective functions.

$$\min F(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x}), G(\mathbf{x})) \quad (20)$$

s.t. $\mathbf{x} \in D$

The $(N_1 + N_2)$ weight vectors used in the proposed algorithm are decomposed into K sub-populations $\Omega_1, \Omega_2, \dots, \Omega_K$, and the size of each sub-population is set to s_i . Each weight vector is assigned to its closest unit center vector according to Eq. (21).

$$\Omega_i = \{\mathbf{u} | \langle \mathbf{u}, \mathbf{w}^i \rangle \leq \langle \mathbf{u}, \mathbf{w}^j \rangle, j = 1, 2, \dots, K\} \quad (21)$$

where $\langle \mathbf{u}, \mathbf{w}^i \rangle$ is the angle between vector \mathbf{u} and the i th unit center vector $\mathbf{w}^1, \mathbf{w}^2, \dots, \mathbf{w}^K$ are the K predefined unit center vectors (see line 3 in Algorithm 1).

At a generation t , the proposed algorithm performs genetic operations on the individuals in each sub-population. To utilize the informative infeasible solutions, a local infeasibility utilization mechanism is designed (see line 8 in Algorithm 1), which the pseudo-code can be found in Algorithm 2.

In this paper, a DE operator named DE/current-to-rand/1 is performed on individuals \mathbf{x}_j , \mathbf{x}' , \mathbf{x}'' and \mathbf{x}''' to generate a new individual $\mathbf{y} = (y_1, y_2, \dots, y_n)$ [4]:

$$\mathbf{y} = \mathbf{x}_j + F \times (\mathbf{x}' - \mathbf{x}'') + F \times (\mathbf{x}'' - \mathbf{x}''') \quad (22)$$

Where F is the scaling factor, which is randomly obtained from the F pool in every generation before performing the genetic operations. When \mathbf{y} violates the boundary constraint, it is fixed back to its domain:

$$y_i = \begin{cases} 2L_i - y_i & y_i < L_i \\ 2U_i - y_i & \text{otherwise} \end{cases} \quad (23)$$

¹ The Matlab code of this paper can be freely obtained from: <https://github.com/ChaodaPeng/M2M-DW-for-UAV-Path-planning.git>.

Where U_i and L_i are the upper and lower boundary of the i th decision variable y_i .

Then an improved version of mutation operator is proposed to enhance the search ability of the proposed algorithm. It is given as follows:

$$y_i = \begin{cases} y_i + \sigma_i (U_i - L_i) & \text{rand} \leq CR \\ y_i & \text{otherwise} \end{cases} \quad (24)$$

$$\sigma_i = \begin{cases} 2\text{rand}^{\frac{1}{\eta+1}} - 1 & \text{rand} \leq 0.5 \\ 1 - (2 - 2\text{rand})^{\frac{1}{\eta+1}} & \text{otherwise} \end{cases} \quad (25)$$

Where CR is the control parameter, which is updated by Eq. (26). rand is a uniformly distributed random number in $(0, 1)$. Each component of the individual \mathbf{y} is fixed within its boundary constraint.

$$CR = \max \left(\left(1 - \frac{t}{\max_t + 1} \right)^\mu, 0.1 \right) \quad (26)$$

Where t is the current generation number, and \max_t is the maximum generation number. The parameter CR decreases from 1 to 0.1 along with the generation t . That is, a larger value of CR enlarges the exploration ability of the proposed algorithm at the beginning stage of the evolution process. As the generation number t increases, the proposed algorithm starts to focus on the exploitation with the decreasing CR . The UAV path problems in complex scenarios have many constraints, which complicates the feasible search space, therefore the algorithms for solving them should have a strong ability of exploring more regions at the early stage in case of getting stuck in the local optimal regions.

The combined population C_t is divided into K sub-populations by using Eq. (21) (see line 17 in Algorithm 1). For the i th sub-population with $\|\Omega_i\|$ individuals, if $\|\Omega_i\| < s_i$, these individuals are assigned to this sub-population, and then $\|\Omega_i\| - s_i$ individuals are randomly selected into the i th sub-population from the combined population. Otherwise, we use the achievement scalarizing function (ASF) [45] to select the best $\|\Omega_i\|$ individuals.

$$\text{ASF}(\mathbf{x}|\mathbf{v}) = \max_{1 \leq i \leq m+1} \left(\frac{F_i(\mathbf{x}) - z_i}{w_i} \right) \quad (27)$$

Where \mathbf{v} is a weight vector, $\mathbf{z} = (z_1, z_2, \dots, z_{m+1})$ is the ideal point, $z_i = \min_{\mathbf{x} \in D} F_i(\mathbf{x})$. It is noted that when selecting the individuals \mathbf{x}' , \mathbf{x}'' and \mathbf{x}''' in Algorithm 2, the proposed algorithm calculates ASF value of an individual only with its objective functions.

Finally, each infeasible weight is updated by decreasing the angle ω :

$$\omega = \frac{\pi k}{2} \quad (28)$$

Where $k = 1 - \text{rand}^{(1 - \frac{t}{\max_t})^\lambda}$. λ is usually set to a value from 2 to 5. In this paper, it is set to 3 as suggested in [23].

4.2. The proposed local infeasibility utilization mechanism

The proposed local infeasibility utilization mechanism is introduced in this subsection. The pseudo-code is presented in Algorithm 2.

In the original M2M-DW, it randomly selected individuals for the genetic operations. Although there is a possibility that an individual can participate in the genetic operations with other individuals which have better fitness, the manner is arbitrary. This may result in poor performance when handling CMOPs with complex constraints. To make better use of the infeasible individuals with better objective values, the local infeasibility utilization mechanism is designed in this paper. The pairing individuals

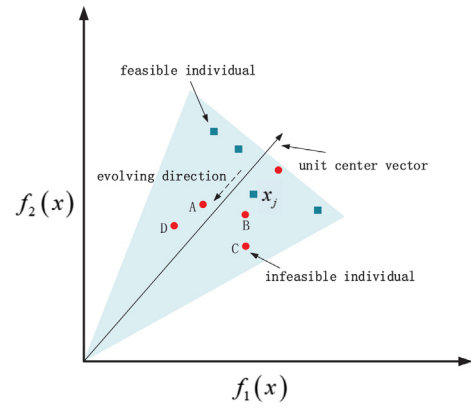


Fig. 3. Illustration of the local infeasibility utilization mechanism.

Algorithm 2: The local infeasibility utilization mechanism

Input:

- The population Q_t .
- The i th sub-population Ω_i .
- The ideal point \mathbf{z} .
- The i th unit center vector \mathbf{w}^i .
- The j th individual \mathbf{x}_j .

Output:

- Three selected individuals \mathbf{x}' , \mathbf{x}'' and \mathbf{x}''' .

- 1: Calculate the ASF values of the individuals in Q_t associated with the i th unit center vector \mathbf{w}^i according to Eq. (27).
- 2: Calculate the ASF value of the individual \mathbf{x}_j associated with the i th unit center vector \mathbf{w}^i according to Eq. (27).
- 3: Randomly select the first individual \mathbf{x}' that has better ASF values than \mathbf{x}_j among the individuals in Q_t if exists. Otherwise, randomly select \mathbf{x}' from the i th sub-population Ω_i . The rest of two individuals \mathbf{x}'' and \mathbf{x}''' are randomly selected from Ω_i .

with better fitness in the same sub-region are likely selected for the genetic operations. As shown in Fig. 3, for an individual \mathbf{x}_j , one of the individuals A, B, C and D will be selected as the individual \mathbf{x}' in Eq. (22). If there is no any individual that has better objective values than the individual \mathbf{x}_j , then \mathbf{x}' is randomly selected from the current sub-population. The individuals \mathbf{x}'' and \mathbf{x}''' are selected randomly from the current sub-population to preserve the diversity of the genetic information.

5. Experimental studies

In this section, a series of experiments on the three sets of UAV path planning problems are conducted to verify the performance of the proposed algorithm. Three decomposition-based CMOEAs, i.e., M2M-DW [23], PPS [26] and MOEA/D-CDP [16], are used as the compared algorithms. Due to the unknown Pareto front of these problems, performance metrics prior to knowing the Pareto front, such as the inverted generation distance metric (IGD) [46], usually are not suitable for the used problems. Therefore, only hypervolume metric (HV) [47] is used to evaluate the performance of the four algorithms in this paper. All feasible solutions are chosen from the final obtained population to calculate the HV values. The reference points are set to (200, 200) for UAV1-1 and UAV1-2, (800, 1000) for UAV2-1 and UAV2-2, and (800, 800) for UAV3-1 and UAV3-2, respectively. A larger HV value implies that an algorithm achieves better performance related to convergence and diversity.

Table 2
Comparison of the mean and Std values of HV on six UAV path planning problems.

Problem		M2M-DW	MOEA/D-CDP	PPS	Our algorithm
UAV1-1	Mean	8.9053e+03 (3)	1.0537e+04 (2)	8.8971e+03 (4)	1.0593e+04 (1)
	Std	2.8138E+02	7.2170E+01	3.1040E+02	6.8103E+01
UAV1-2	Mean	5.5440e+03 (3)	9.2104e+03 (2)	5.5120e+03 (4)	9.4487e+03 (1)
	Std	8.5604E+02	3.6969E+02	9.5296E+02	2.3218E+01
UAV2-1	Mean	3.3337e+05 (3)	3.3168e+05 (4)	3.3401e+05 (1)	3.3363e+05 (2)
	Std	4.0265E+02	8.7583E+02	3.2944E+02	3.4188E+02
UAV2-2	Mean	3.3538e+05 (2)	3.3434e+05 (3)	1.4791e+05 (4)	3.3571e+05 (1)
	Std	1.4982E+02	4.2016E+02	1.1434E+04	2.6744E+02
UAV3-1	Mean	2.9366e+05 (4)	3.0560e+05 (3)	3.0598e+05 (2)	3.0681e+05 (1)
	Std	1.6524E+03	9.8659E+02	1.7046E+03	9.9166E+02
UAV3-2	Mean	2.9613e+04 (4)	1.6310e+05 (2)	1.2046e+05 (3)	1.7540e+05 (1)
	Std	3.8354E+04	1.7890E+04	5.7912E+04	1.8297E+04
Overall Ranking		19	16	18	7

5.1. Parameter settings

The parameters of this paper are set as follows:

- Each of the algorithms in this paper runs 21 independent times, and stops after 3×10^4 function evaluations (FEs).
- The population size N is set to 100, N_1 and N_2 are set to 30 and 70 respectively.
- The F pool is set to {0.5, 0.7, 0.9}, η is set to 20.
- The number of sub-populations K is set to 10.
- The parameter η related to Eq. (25) is set to 25.
- The parameter μ related to Eq. (26) is set to 0.2.
- The rest of the parameters in M2M-DW, PPS, and MOEA/D-CDP remain the same as in the original papers.

5.2. Experimental results

5.2.1. Comparison of the proposed algorithm with the other algorithms

Table 2 displays the experimental results of the four CMOEAs over 21 independent runs regarding to the mean value and standard deviation (Std) of HV on the six UAV path planning problems. For each UAV path planning problem, a CMOEA that achieves the best mean value of HV is ranked 1 within a bracket as shown in the table, the one achieving the second best mean value of HV is ranked 2, and so for the rest of the CMOEAs. Afterward, the overall ranking of a CMOEA is obtained by summing up all the rankings of each UAV path planning problem. The smaller overall ranking indicates a better result among the compared algorithms. Better results are marked bold.

The advantages of the proposed algorithm can be intuitively observed from Table 2. The proposed algorithm consistently achieves better results than the other three decomposition-based methods, i.e., M2M-DW, PPS and MOEA/D-CDP, on each UAV path planning problem in terms of mean values of HV and the overall ranking except UAV2-1. The main reason is given as follows. For M2M-DW and the proposed algorithm, it can be observed that the proposed algorithm is superior over M2M-DW, even these two algorithms have an ability to utilize infeasible solutions, the manner of utilizing infeasible solutions of M2M-DW is arbitrary, which implies the effectiveness of the proposed local infeasibility utilization and the improved mutation scheme. Although M2M-DW and PPS can make use of infeasible solutions, they achieve worse results than MOEA/D-CDP. It is because the arbitrary manner of infeasibility utilization results in the degeneration of finding optimal feasible solutions. A closer look at this table that MOEA/D-CDP seems to achieve the second best result according to the overall ranking, however, it obtains the highest Std values than the other three compared algorithms, especially

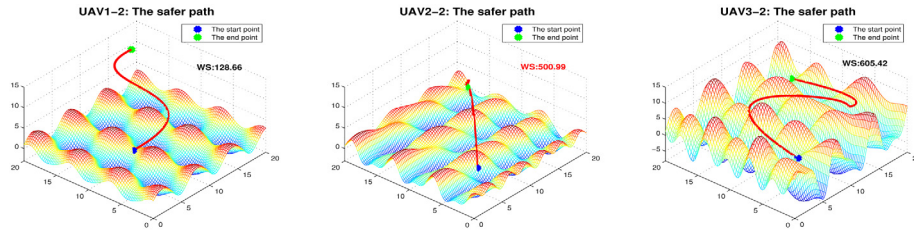
on UAV1-2, UAV2-2, and UAV3-2, which are highly constrained problems. Therefore, MOEA/D-CDP is not stable for solving these UAV path planning problems.

Figs. 4 and 5 present the paths obtained with two preferences at the median run of HV values of the four compared algorithms. Fig. 4 shows the path obtained by the weight vector (0.8, 0.2) on test instances UAV1-2, UAV2-2 and UAV3-2, which considers the path with the shorter traveling distance. Fig. 5 shows the path obtained by the weight vector (0.2, 0.8), which considers the path with more safety but the longer flight path among the final obtained Pareto optimal solutions. Besides, a weighted sum (denoted as WS in the figures) of the two objective functions $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$ associated with the given weight vector is presented in each figure. A smaller value indicates the presented figure is a better choice with the corresponding weight vector. The best weighted sum value is marked red.

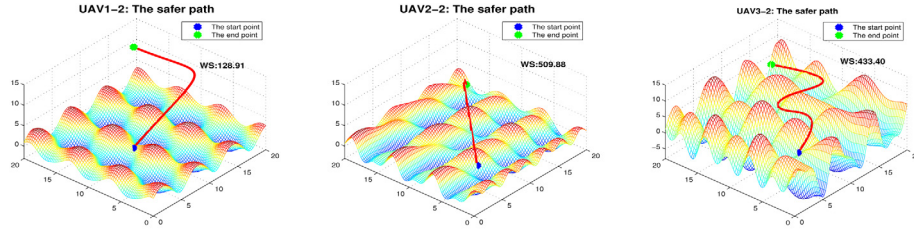
From Figs. 4 and 5, it can be observed that our algorithm obtained shorter and safer paths under the considerable range of preferences on most of the UAV path planning problems. The reason is clear that the proposed algorithm has a strong ability to utilize the useful infeasible individuals with the local infeasibility utilization mechanism, which is able to guide the search to the more promising regions than the other compared algorithms. Focusing on the last columns of the two figures, as UAV3-2 is the highly constrained problem, the paths obtained by M2M-DW, MOEA/D-CDP and PPS are either longer or less safe than the ones obtained by the proposed algorithm. That is, the three compared algorithms fail to effectively handle the constraints and result in poor performance, which implies the effectiveness of the proposed algorithm again.

Fig. 6 presents the convergence graphs of HV values at the median run for the four compared algorithms on problems UAV1-1, UAV1-2, UAV2-1, UAV2-2, UAV3-1 and UAV3-2. The X axis indicates the generation number t , and the Y axis indicates the HV values of the obtained population consisting of feasible non-domination solutions at generation t . It is noted that the reference point for each problem is set the same as in this paper. When there are no feasible solutions inside the area formed by the reference point and two objectives, then HV value will be assigned to zero, and when there is no any feasible solution in the current population, the HV value will be assigned to -1.

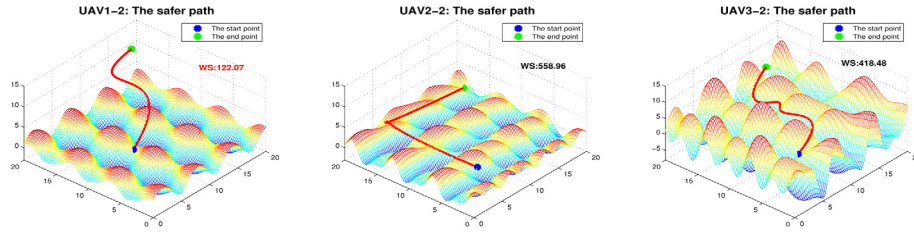
The advantages of the proposed algorithm can be observed from these convergence graphs once again. The proposed algorithm stably finds the feasible optimal promising regions for each UAV path planning problem, and obtains better results on five out of six problems compared with M2M-DW, MOEA/D-CDP and PPS. It can be observed that PPS is unstable to find the optimal solutions since it disregards the feasible solutions during the push stage. That is, it finds some feasible solutions and suddenly



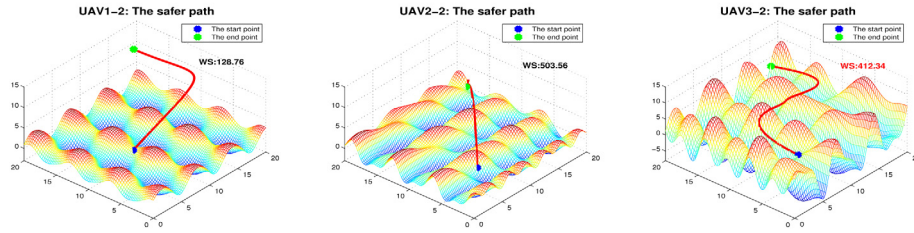
(a) The paths obtained by M2M-DW.



(b) The paths obtained by MOEA/D-CDP.



(c) The paths obtained by PPS.



(d) The paths obtained by our algorithm.

Fig. 4. The paths obtained with the safer flight based on the median run of HV value of M2M-DW, MOEA/D-CDP, PPS and our algorithm on UAV1-2, UAV2-2 and UAV3-2.

Table 3

The sensitivity of the parameter μ .

UAV2-2	$\mu = 0.05$	$\mu = 0.1$	$\mu = 0.15$	$\mu = 0.2$	$\mu = 0.25$
Mean	3.3573E+05	3.3533E+05	3.3564E+05	3.3571E+05	3.3561E+05
Std	1.7260E+02	1.9299E+02	2.1919E+02	2.6744E+02	1.6681E+02

converges back to the infeasible regions since the solutions in the infeasible regions have better objective values than the feasible ones.

5.2.2. Investigation of the parameter μ

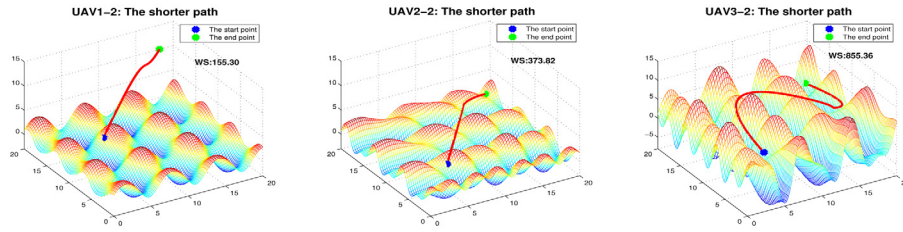
We investigate the sensitivity of the parameter μ related to the improved mutation scheme on UAV2-2 in this subsection. The proposed algorithm with five values of parameter μ (0.05,

0.1, 0.15, 0.2 and 0.25) are adopted in this numerical experiment, while the rest of the parameters remain the same as in Section 5.1. The result is presented in Table 3.

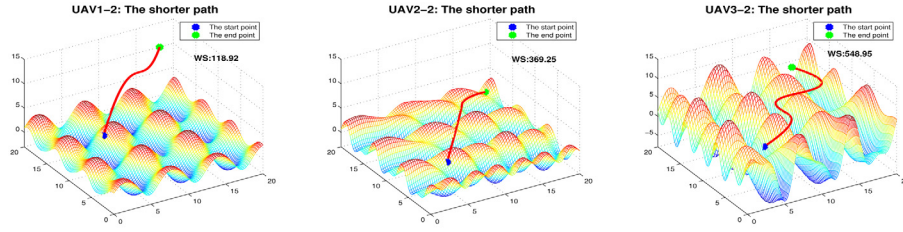
The parameter μ controls the possibility of the improved mutation scheme. A smaller value of μ indicates that an individual will have less mutation operation. That is, the ability of exploring the search space of the proposed algorithm is decreasing. From Table 3, we can see that the proposed algorithm is not sensitive to the parameter μ based on the obtained mean and Std value of HV on UAV2-2.

5.2.3. Investigation of the F pool

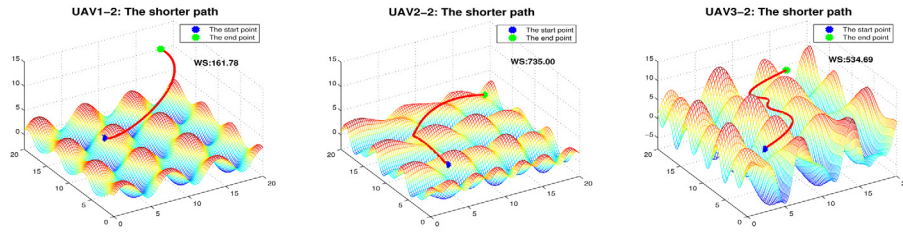
We investigate the sensitivity of the parameter F pool on UAV2-2 in this subsection. The proposed algorithm with five values of F pool, i.e., {0.1, 0.3, 0.5}, {0.1, 0.5, 0.9}, {0.5, 0.8, 1}, {0.5, 0.7, 0.9} and {0.2, 0.5, 0.8}, are adopted in this numerical



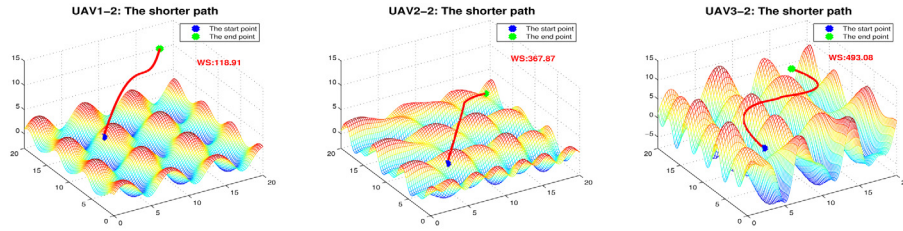
(a) The paths obtained by M2M-DW.



(b) The paths obtained by MOEA/D-CDP.



(c) The paths obtained by PPS.



(d) The paths obtained by our algorithm.

Fig. 5. The paths obtained with the shorter traveling distance based on the median run of HV value of M2M-DW, MOEA/D-CDP, PPS and our algorithm on UAV1-2, UAV2-2 and UAV3-2.

experiment, while the rest of the parameters remain the same as in Section 5.1. The result is presented in Fig. 7.

From Fig. 7, we can see that the proposed algorithm is not sensitive to the parameter F pool on UAV2-2 since it obtains similar mean values of HV.

6. Conclusion

M2M-DW with the local infeasibility utilization method and the improved mutation scheme was proposed for solving UAV path planning problems in this paper. UAV path planning represented by B-Spline curve was first formulated as a multi-objective optimization problem, i.e., minimizing the traveling distance and the risk of a UAV, with four constraints including the minimum flight altitude, maximum flight altitude, minimum flight angle

and minimum/maximum flight scope, and then three sets of UAV path planning problems with different difficulties were presented in Section 3.

Subsequently, we conducted a series of experiments to study the performance of the four CMOEAs on the three sets of UAV path planning problems in Section 4. The experimental results highlighted the superiority of the proposed algorithm over the compared algorithms in terms of finding a set of well-distributed and well-converged feasible non-dominated solutions. Furthermore, the parameters of the proposed algorithm were investigated and the results showed that the proposed algorithm was insensitive to the parameter μ and F pool on the test instance studied. In conclusion, the proposed algorithm can provide smoother paths, which makes it more appropriate for UAV path planning.

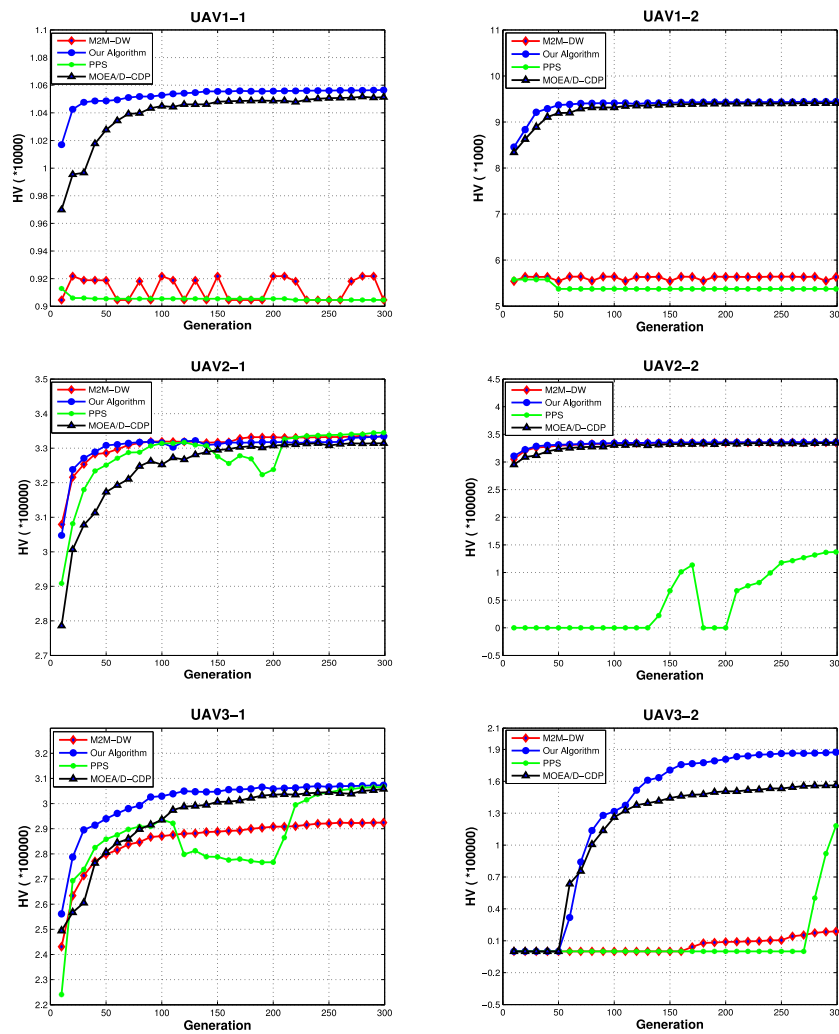


Fig. 6. Convergence graphs for the four compared algorithms based on the median run of HV value on six UAV path planning problems.

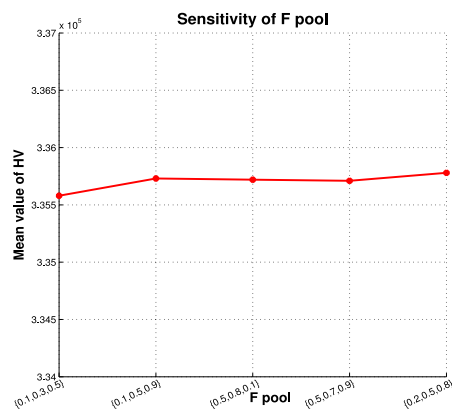


Fig. 7. Investigation of the parameter F pool on UAV2-2.

CRediT authorship contribution statement

Chaoda Peng: Conceptualization, Methodology, Software, Writing – original draft. **Shaojian Qiu:** Visualization, Investigation, Validation, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work is supported by the Special Fund for Talents of South China Agricultural University under Grant 221114.



References

- [1] Q. Hu, Y. Cai, G. Yu, Z. Qin, M. Zhao, G.Y. Li, Joint offloading and trajectory design for UAV-enabled mobile edge computing systems, *IEEE Internet Things J.* 6 (2) (2018) 1879–1892.
- [2] H. Guo, J. Liu, UAV-enhanced intelligent offloading for internet of things at the edge, *IEEE Trans. Ind. Inform.* 16 (4) (2019) 2737–2746.
- [3] Y. Liu, K. Xiong, Q. Ni, P. Fan, K.B. Letaief, UAV-assisted wireless powered cooperative mobile edge computing: Joint offloading, CPU control, and trajectory optimization, *IEEE Internet Things J.* 7 (4) (2019) 2777–2790.
- [4] Z.-Z. Liu, Y. Wang, Handling constrained multiobjective optimization problems with constraints in both the decision and objective spaces, *IEEE Trans. Evol. Comput.* 23 (5) (2019) 870–884.
- [5] I.K. Nikolos, E.S. Zografos, A.N. Brintaki, UAV path planning using evolutionary algorithms, in: *Innovations In Intelligent Machines-1*, Springer, 2007, pp. 77–111.

- [6] M.D. Phung, Q.P. Ha, Safety-enhanced UAV path planning with spherical vector-based particle swarm optimization, *Appl. Soft Comput.* 107 (2021) 107376.
- [7] X. Yu, C. Li, J. Zhou, A constrained differential evolution algorithm to solve UAV path planning in disaster scenarios, *Knowl.-Based Syst.* 204 (2020) 106209.
- [8] S. Mittal, K. Deb, Three-dimensional offline path planning for UAVs using multiobjective evolutionary algorithms, in: 2007 IEEE Congress On Evolutionary Computation, IEEE, 2007, pp. 3195–3202.
- [9] B.N. Coelho, V.N. Coelho, I.M. Coelho, L.S. Ochi, R. Haghazadeh, D. Zuidema, M.S. Lima, A.R. da Costa, A multi-objective green UAV routing problem, *Comput. Oper. Res.* 88 (2017) 306–315.
- [10] H. Qiu, H. Duan, A multi-objective pigeon-inspired optimization approach to UAV distributed flocking among obstacles, *Inform. Sci.* 509 (2020) 515–529.
- [11] Z. Sun, J. Wu, J. Yang, Y. Huang, C. Li, D. Li, Path planning for GEO-uav bistatic SAR using constrained adaptive multiobjective differential evolution, *IEEE Trans. Geosci. Remote Sens.* 54 (11) (2016) 6444–6457.
- [12] X. Yu, C. Li, G.G. Yen, A knee-guided differential evolution algorithm for unmanned aerial vehicle path planning in disaster management, *Appl. Soft Comput.* 98 (2021) 106857.
- [13] A. Homayfar, C.X. Qi, S.H. Lai, Constrained optimization via genetic algorithms, *Simulation* 62 (4) (1994) 242–253.
- [14] J.A. Joines, C.R. Houck, On the use of non-stationary penalty functions to solve nonlinear constrained optimization problems with GA's, in: Proceedings Of The First IEEE Conference On Evolutionary Computation. IEEE World Congress On Computational Intelligence, IEEE, 1994, pp. 579–584.
- [15] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE Trans. Evol. Comput.* 6 (2) (2002) 182–197.
- [16] M.A. Jan, R.A. Khanum, A study of two penalty-parameterless constraint handling techniques in the framework of MOEA/D, *Appl. Soft Comput.* 13 (1) (2013) 128–148.
- [17] H. Jain, K. Deb, An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part II: Handling constraints and extending to an adaptive approach., *IEEE Trans. Evol. Comput.* 18 (4) (2014) 602–622.
- [18] H.K. Singh, T. Ray, W. Smith, C-PSA: Constrained Pareto simulated annealing for constrained multi-objective optimization, *Inform. Sci.* 180 (13) (2010) 2499–2513.
- [19] K. Harada, J. Sakuma, I. Ono, S. Kobayashi, Constraint-handling method for multi-objective function optimization: Pareto descent repair operator, in: International Conference On Evolutionary Multi-Criterion Optimization, Springer, 2007, pp. 156–170.
- [20] L. Jiao, J. Luo, R. Shang, F. Liu, A modified objective function method with feasible-guiding strategy to solve constrained multi-objective optimization problems, *Appl. Soft Comput.* 14 (2014) 363–380.
- [21] J. Wang, G. Liang, J. Zhang, Cooperative differential evolution framework for constrained multiobjective optimization, *IEEE Trans. Cybern.* 49 (6) (2018) 2060–2072.
- [22] M. Miyakawa, K. Takadama, H. Sato, Directed mating using inverted PBI function for constrained multi-objective optimization, in: IEEE Congress On Evolutionary Computation, IEEE, 2015, pp. 2929–2936.
- [23] C. Peng, H.-L. Liu, F. Gu, An evolutionary algorithm with directed weights for constrained multi-objective optimization, *Appl. Soft Comput.* 60 (2017) 613–622.
- [24] S.Z. Martínez, C.A.C. Coello, A multi-objective evolutionary algorithm based on decomposition for constrained multi-objective optimization, *IEEE Congr. Evol. Comput.* (2014) 429–436.
- [25] C. Peng, H.-L. Liu, E.D. Goodman, Handling multi-objective optimization problems with unbalanced constraints and their effects on evolutionary algorithm performance, *Swarm Evol. Comput.* (2020) 100676.
- [26] Z. Fan, W. Li, X. Cai, H. Li, C. Wei, Q. Zhang, K. Deb, E. Goodman, Push and pull search for solving constrained multi-objective optimization problems, *Swarm Evol. Comput.* 44 (2019) 665–679.
- [27] K. Li, R. Chen, G. Fu, X. Yao, Two-archive evolutionary algorithm for constrained multiobjective optimization, *IEEE Trans. Evol. Comput.* 23 (2) (2018) 303–315.
- [28] C. Peng, H.-L. Liu, E.D. Goodman, A cooperative evolutionary framework based on an improved version of directed weight vectors for constrained multiobjective optimization with deceptive constraints, *IEEE Trans. Cybern.* 51 (11) (2020) 5546–5558.
- [29] A. Sadollah, H. Eskandar, J.H. Kim, Water cycle algorithm for solving constrained multi-objective optimization problems, *Appl. Soft Comput.* 27 (2015) 279–298.
- [30] R. Datta, R.G. Regis, A surrogate-assisted evolution strategy for constrained multi-objective optimization, *Expert Syst. Appl.* 57 (2016) 270–284.
- [31] Z. Ma, Y. Wang, Evolutionary constrained multiobjective optimization: Test suite construction and performance comparisons, *IEEE Trans. Evol. Comput.* 23 (6) (2019) 972–986.
- [32] H. Li, Q. Zhang, Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II, *IEEE Trans. Evol. Comput.* 13 (2) (2009) 284–302.
- [33] C.R. Atencia, J. Del Ser, D. Camacho, Weighted strategies to guide a multi-objective evolutionary algorithm for multi-UAV mission planning, *Swarm Evol. Comput.* 44 (2019) 480–495.
- [34] C. Ramirez-Atencia, D. Camacho, Constrained multi-objective optimization for multi-UAV planning, *J. Ambient Intell. Humaniz. Comput.* 10 (6) (2019) 2467–2484.
- [35] R.W. Beard, T.W. McLain, M.A. Goodrich, E.P. Anderson, Coordinated target assignment and intercept for unmanned air vehicles, *IEEE Trans. Robot. Autom.* 18 (6) (2002) 911–922.
- [36] Y. Lin, S. Saripalli, Sampling-based path planning for UAV collision avoidance, *IEEE Trans. Intell. Trans. Syst.* 18 (11) (2017) 3179–3192.
- [37] T.W. McLain, R.W. Beard, Coordination variables, coordination functions, and cooperative timing missions, *J. Guid. Control Dyn.* 28 (1) (2005) 150–161.
- [38] M. Radmanesh, M. Kumar, Flight formation of UAVs in presence of moving obstacles using fast-dynamic mixed integer linear programming, *Aerosp. Sci. Technol.* 50 (2016) 149–160.
- [39] P. Fabiani, V. Fuertes, A. Piquereau, R. Mampey, F. Teichteil-Königsbuch, Autonomous flight and navigation of VTOL UAVs: from autonomy demonstrations to out-of-sight flights, *Aerosp. Sci. Technol.* 11 (2–3) (2007) 183–193.
- [40] B. Penin, P.R. Giordano, F. Chaumette, Minimum-time trajectory planning under intermittent measurements, *IEEE Robotics Autom. Lett.* 4 (1) (2018) 153–160.
- [41] A. Stentz, Optimal and efficient path planning for unknown and dynamic environments, Tech. rep., Carnegie-mellon Univ Pittsburgh PA Robotics Inst, 1993.
- [42] Y. Chen, G. Luo, Y. Mei, J. Yu, X. Su, UAV path planning using artificial potential field method updated by optimal control theory, *Int. J. Syst. Sci.* 47 (6) (2016) 1407–1420.
- [43] H. Heidari, M. Saska, Collision-free trajectory planning of multi-rotor UAVs in a wind condition based on modified potential field, *Mech. Mach. Theory* 156 (2021) 104140.
- [44] E. Besada-Portas, L. de la Torre, M. Jesus, B. de Andrés-Toro, Evolutionary trajectory planner for multiple UAVs in realistic scenarios, *IEEE Trans. Robotics* 26 (4) (2010) 619–634.
- [45] A.P. Wierzbicki, The use of reference objectives in multiobjective optimization, in: Multiple Criteria Decision Making Theory And Application, Springer, 1980, pp. 468–486.
- [46] C.A.C. Coello, G.B. Lamont, D.A. Van Veldhuizen, et al., Evolutionary algorithms for solving multi-objective problems, vol. 5, Springer, 2007.
- [47] E. Zitzler, L. Thiele, Multiobjective optimization using evolutionary algorithms—a comparative case study, in: International Conference On Parallel Problem Solving From Nature, Springer, 1998, pp. 292–301.

[IEEE.org](#)
[IEEE Xplore Digital Library](#)
[IEEE Standards](#)
[IEEE Spectrum](#)
[More Sites](#)

[f](#)
[X](#)
[in](#)
[v](#)
[+](#)
[G](#)
[@](#)

[About](#)
[Membership](#)
[Conferences](#)
[Publications](#)
[Awards](#)
[Educational Activities](#)
[Technical Activities](#)
[Standards Activities](#)
[News](#)

IEEE Transactions on Intelligent Transportation Systems (T-ITS)

Home » Publications » IEEE Transactions on Intelligent Transportation Systems (T-ITS)

On This Page

- [Scope](#)
- [Submission Information: Information for Authors](#)
- [Open Access Publishing option of the IEEE Transactions on ITS](#)
- [Alternative Option for Open Access Publishing in ITS](#)
- [Submitting Final Files: Information for authors of accepted papers](#)

Publications Menu

- [IEEE Author Tools](#)
- [Open Journal of Intelligent Transportation Systems \(OJ-ITS\)](#) >
- [Transactions on Intelligent Transportation Systems \(T-ITS\)](#) >
 - [Overview](#)
 - [T-ITS Editorial Board](#)
 - [T-ITS Special Issues](#)
 - [T-ITS Previous Issues](#)

There is no open call for new Associate Editors for the IEEE Transactions on Intelligent Transportation Systems now. The next call will be opened in August 2025.

The IEEE Transactions on Intelligent Transportation Systems (T-ITS) is published monthly. All issues of T-ITS are digitally archived in IEEE Xplore. (See also a [list of previous issues](#).) Members of the IEEE ITS Society have access to the electronic version of the Transactions free of charge. IEEE T-ITS is a top-ranked publication in the ITS field.

T-ITS Impact Factor: 7.9 (2023)

Editor-in-Chief

Prof. Simona Sacone, EIC, Transactions on ITS
 Dept. of Computer Science, Bioengineering, Robotics and Systems Engineering,
 University of Genova, Italy
simona.sacone@unige.it

Scope

The IEEE Transactions on Intelligent Transportation Systems covers fundamental and applied research on all scientific and technical aspects of modern transportation systems, including but not limited to sensing, communications, controls, planning, design, and implementation of intelligent transportation systems. It covers theory, methodologies, modeling and simulation, experimentation, and evaluations of transportation systems that include multi-modal transportation, surface transportation traffic, coordinated multiple vehicles, infrastructure, and other road users (pedestrians, bicyclists, etc.) and their interactions. The Transactions also serve as a forum for the design, analysis, and control of information technology as it is applied to transportation systems. Read more on [IEEE Xplore](#).

IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS

A PUBLICATION OF THE IEEE INTELLIGENT TRANSPORTATION SYSTEMS SOCIETY

OCTOBER 2024

VOLUME 25

NUMBER 10

ITISFG

(ISSN 1558-0016)

EDITORIAL

Scanning the Issue *S. Sacone* 12846

SURVEY PAPERS

Pedestrian Simulation Challenges: Modeling Techniques and Emerging Positioning Technologies for ITS Applications *V. Papathanasopoulou, H. Perakis, I. Spyropoulou, and V. Gikas* 12876

Multi Criteria Methodology for Aircraft Trajectory Planning Algorithm Selection: A Survey *A. Guitart, C. Demouge, D. Delahaye, and E. Feron* 12893

Passenger Emergency Evacuation in Subway Station Systems: A Bibliometric Analysis and Systematic Review *X. Yang, M. Zhou, and H. Dong* 12912

A Critical Review of Subway Train Timetabling and Rescheduling Problems *L. Kang, N. Buhigiro, H. Sun, and J. Wu* 12930

Toward Smart Skies: Reviewing the State of the Art and Challenges for Intelligent Air Transportation Systems (IATS) *S. Wandelt and C. Zheng* 12943

REGULAR PAPERS

Reducing Power Consumption and Latency of Autonomous Vehicles With Efficient Task and Path Assignment in the V2X-MEC Based on Nash Equilibrium *R. Xiong, J. Cheng, Q. Yuan, K. Ma, L. Li, C. Zhang, and H. Zeng* 12954

Hash-Based Gaussian Mixture Model (HGMM) for Roadside LiDAR Smart Infrastructure Applications *T. T. Zhang, Y. Ge, A. Chen, M. Sartipi, and P. J. Jin* 12968

Modeling and Robust H_∞ Control Synthesis of the CAV-HDV Heterogeneous Traffic System With Different Car-Following Modes *Y. Gong and W.-X. Zhu* 12980

Privacy-Preserving Travel Recommendation Based on Stay Points Over Outsourced Spatio-Temporal Data *L. Han, W. Luo, R. Lu, Y. Zheng, A. Yang, J. Lai, Y. Cheng, and Y. Zhang* 12999

Low-Rank Tensor and Hybrid Smoothness Regularization-Based Approach for Traffic Data Imputation With Multimodal Missing *Z. Zeng, B. Liu, J. Feng, and X. Yang* 13014

Online Spatio-Temporal Correlation-Based Federated Learning for Traffic Flow Forecasting *Q. Liu, S. Sun, M. Liu, Y. Wang, and B. Gao* 13027

Linkage-Based Object Re-Identification via Graph Learning *L. Wei, Z. Wang, C. Lang, L. Liang, T. Wang, S. Feng, and Y. Li* 13040

STORM: A Spatio-Temporal Context-Aware Model for Predicting Event-Triggered Abnormal Crowd Traffic *Y. Hong, H. Zhu, T. Shou, Z. Wang, L. Chen, L. Wang, C. Wang, and L. Chen* 13051

Enhance Sample Efficiency and Robustness of End-to-End Urban Autonomous Driving via Semantic Masked World Model *Z. Gao, Y. Mu, C. Chen, J. Duan, P. Luo, Y. Lu, and S. E. Li* 13067

Probabilistic Game-Theoretic Traffic Routing *E. Benenati and S. Grammatico* 13080

(Contents Continued on Page 12841)



A Novel Factor Graph Framework for Tightly Coupled GNSS/INS Integration With Carrier-Phase Ambiguity Resolution	Z. Shen, X. Li, X. Wang, Z. Wu, X. Li, Y. Zhou, and S. Li	13091
An Asynchronous Shuffled Frog-Leaping With Feasible Jaya Algorithm for Uncertain Task Rescheduling Problem in UAV Emergency Networks	Q. Luan, H. Cui, X. Yu, L. Zhang, and B. Sikdar	13106
Adaptive Speed Optimization Strategy at Signalized Intersections Based on the Penetration Rate of Connected Automated Vehicles	R. Fan, J. Lu, D. Z. W. Wang, and F. Zhang	13122
A POV-Based Highway Vehicle Trajectory Dataset and Prediction Architecture	V. Katariya, G. A. Noghre, A. D. Pazho, and H. Tabkhi	13136
Order Dispatching via GNN-Based Optimization Algorithm for On-Demand Food Delivery	J.-F. Chen, L. Wang, Y. Liang, Y. Yu, J. Feng, J. Zhao, and X. Ding	13147
Spatio-Temporal Corridor-Based Motion Planning of Lane Change Maneuver for Autonomous Driving in Multi-Vehicle Traffic	Y. Yoon, C. Kim, H. Lee, D. Seo, and K. Yi	13163
Comparative Analysis of Driver Overtaking Behavior Near Low-Speed Automated Vehicles and Human-Driven Vehicles	C. M. Craig, B. A. Drahos, D. Tian, and N. L. Morris	13184
Impact of Smartphone Activity on Pedestrian Safety: A Case Study in Seoul	Y. Han and H. Lee	13194
Dynamic Passenger Route Guidance in the Multimodal Transit System With Graph Representation and Attention Based Deep Reinforcement Learning	H. Hao, E. Yao, R. Chen, L. Pan, and Y. Wang	13204
Determination of Optimum Coordinate Transformation Parameters for GNSS and LiDAR-Based Localization in Automated Vehicles	V. İlçi, M. Yavuz, Y. Şişman, K. Par, and A. U. Peker	13217
Compare and Focus: Multi-Scale View Aggregation for Crowd Counting	S. Jiang, J. Cai, H. Zhang, Y. Liu, and Q. Liu	13231
Hyper-Anchor Based Lane Detection	B. Liu and Q. Ling	13240
Multi-Scale Learnable Gabor Transform for Pedestrian Trajectory Prediction From Different Perspectives	A. Feng, C. Han, J. Gong, Y. Yi, R. Qiu, and Y. Cheng	13253
A Model-Based Research on Performance Evaluation and Topology Optimization of Series-Parallel Lithium-Ion Battery Packs	X. Liu, F. Qu, Q. Luo, H. Liang, P. Sun, and X. Du	13264
PedAST-GCN: Fast Pedestrian Crossing Intention Prediction Using Spatial-Temporal Attention Graph Convolution Networks	Y. Ling, Z. Ma, Q. Zhang, B. Xie, and X. Weng	13277
Functional Safety and Performance Analysis of Autonomous Route Management for Autonomous Train Control System	H. Song, L. Li, Y. Li, L. Tan, and H. Dong	13291
Robust Transit Frequency Setting Problem With Demand Uncertainty	X. Guo, B. Mo, H. N. Koutsopoulos, S. Wang, and J. Zhao	13305
A Deep-Learning Approach to Detect and Classify Heavy-Duty Trucks in Satellite Images	X. Liu, Y. Li, L. Sizemore, X. Xie, and J. Wu	13323
Deep Reinforcement Learning-Based Adaptive Computation Offloading and Power Allocation in Vehicular Edge Computing Networks	B. Qiu, Y. Wang, H. Xiao, and Z. Zhang	13339
A Pavement Crack Translator for Data Augmentation and Pixel-Level Detection Based on Weakly Supervised Learning	J. Zhong, Y. Ma, M. Zhang, R. Xiao, G. Cheng, and B. Huang	13350
Consideration of Human Vision in Crowd Simulations	B. Yu, J. Ye, and D. Ou	13364
Arterial Signal Timing Based on Probe Vehicle Trajectories Under Cyclic Stochastic Demand	W. Ma, X. Li, C. Yu, Z. Su, and S. Liu	13375
Monitoring Outdoor Parking in Urban Areas With Unmanned Aerial Vehicles	S. Kim, Y. Tak, E. Barmounakis, and N. Geroliminis	13393
Dual-Signature Blockchain-Based Key Sharing Protocol for Secure V2V Communications in Multi-Domain IoV Environments	D. Abbasinezhad-Mood and H. Ghaemi	13407
Toward C-V2X Enabled Connected Transportation System: RSU-Based Cooperative Localization Framework for Autonomous Vehicles	Z. Huang, S. Chen, Y. Pian, Z. Sheng, S. Ahn, and D. A. Noyce	13417
A Recurrent CNN for Online Object Detection on Raw Radar Frames	C. Decourt, R. VanRullen, D. Salle, and T. Oberlin	13432
Stability and Safety Analysis of Connected and Automated Vehicle Platoon Considering Dynamic Communication Topology	X. Wang, C. Xu, X. Zhao, H. Li, and X. Jiang	13442
Reactive Composition of UAV Delivery Services in Urban Environments	W. Lee, B. Shahzaad, B. Alkouz, and A. Bouguettaya	13453
RdmkNet & Toronto-Rdmk: Large-Scale Datasets for Road Marking Classification and Segmentation	J. Du, L. Ma, J. Li, N. Qin, J. Zelek, H. Guan, and J. Li	13467

Real-Time Collision Mitigation Strategies for Autonomous Vehicles	13483
..... <i>L. Bosia, L. Manganotto, M. Anghileri, S. Dolci, P. Astori, and C.-D. Kan</i>	
Fixed Frequency Highly Efficient Resonant Converter With Low Component Count for CC and CV Charges of Electric Vehicles Batteries	13494
..... <i>S. Askari and H. Farzanehfard</i>	
STARRIS-Assisted IoV NOMA Networks With Hardware Impairments and Imperfect CSI	13501
..... <i>Y. Zhang, L. Yang, X. Li, K. Guo, H. Liu, and Y. Bian</i>	
Intelligent Assessment of Pavement Structural Conditions: A Novel FeMViT Classification Network for GPR Images	13511
..... <i>Z. Liu, S. Wang, X. Gu, D. Wang, Q. Dong, and B. Cui</i>	
PBAG: A Privacy-Preserving Blockchain-Based Authentication Protocol With Global-Updated Commitment in IoVs	13524
..... <i>X. Feng, K. Cui, L. Wang, Z. Liu, and J. Ma</i>	
Public Bike Scheduling Strategy Based on Demand Prediction for Unbalanced Life-Value Distribution	13546
..... <i>H. Sun, Z. Tang, M. Cao, Y. Wang, Z. Yang, H. Xue, R. Xue, L. He, and H. Xiong</i>	
Evaluating Autonomous Vehicle Safety Performance Through Analysis of Pre-Crash Trajectories of Powered Two-Wheelers	13560
..... <i>R. Zhou, Z. Lin, G. Zhang, H. Huang, H. Zhou, and J. Chen</i>	
Potential and Electro-Mechanical Coupling Analysis of a Novel HTS Maglev System Employing Double-Sided Homopolar Linear Synchronous Motor	13573
..... <i>J. Zheng, S. Nie, H. Jing, Y. He, M. Li, Y. Ma, and Z. Ding</i>	
AodeMar: Attention-Aware Occlusion Detection of Vessels for Maritime Autonomous Surface Ships	13584
..... <i>N. Wang, Y. Wang, Y. Feng, and Y. Wei</i>	
Koopman-Based Hybrid Modeling and Zonotopic Tube Robust MPC for Motion Control of Automated Vehicles ...	13598
..... <i>H. Zheng, Y. Li, L. Zheng, and E. Hashemi</i>	
Detecting Traffic Anomalies During Extreme Events via a Temporal Self-Expressive Model	13613
..... <i>M. Nouri, E. Konyar, M. R. Gahrooeri, and M. Ilbeigi</i>	
Domain Adaptive LiDAR Point Cloud Segmentation via Density-Aware Self-Training	13627
..... <i>A. Xiao, J. Huang, K. Liu, D. Guan, X. Zhang, and S. Lu</i>	
Estimation of Vehicular Journey Time Variability by Bayesian Data Fusion With General Mixture Model	13640
..... <i>X. Wu, A. H. F. Chow, L. Zhuang, W. Ma, W. H. K. Lam, and S. C. Wong</i>	
Safe Reinforcement Learning in Autonomous Driving With Epistemic Uncertainty Estimation	13653
..... <i>Z. Zhang, Q. Liu, Y. Li, K. Lin, and L. Li</i>	
An Ensemble Learning Approach With Attention Mechanism for Detecting Pavement Distress and Disaster-Induced Road Damage	13667
..... <i>S. Wang, H. Jiao, X. Su, and Q. Yuan</i>	
A Freeway Traffic Flow Prediction Model Based on a Generalized Dynamic Spatio-Temporal Graph Convolutional Network	13682
..... <i>R. Gan, B. An, L. Li, X. Qu, and B. Ran</i>	
Egocentric Vulnerable Road Users Trajectory Prediction With Incomplete Observation	13694
..... <i>H. Liu, C. Liu, F. Chang, Y. Lu, and M. Liu</i>	
LAM-Depth: Laplace-Attention Module-Based Self-Supervised Monocular Depth Estimation	13706
..... <i>J. Wei, S. Pan, W. Gao, and P. Guo</i>	
Long-Term Energy Consumption Minimization in NOMA-Enabled Vehicular Edge Computing Networks	13717
..... <i>L. P. Qian, X. Dong, M. Wu, Y. Wu, and L. Zhao</i>	
Enhancing Timeliness in Asynchronous Vehicle Localization: A Signal-Multiplexing Network Measuring Approach	13729
..... <i>H. Zhao, Z. Zhang, L. Xu, Y. Wang, and Y. Shen</i>	
Optimal Deployment of Connected and Autonomous Vehicle Dedicated Lanes: A Trade-Off Between Safety and Efficiency	13744
..... <i>C. Hu, C. Han, A. Pervez, J. Hao, G. Xu, J. Tang, and H. Huang</i>	
Two-Layer MPC Architecture for Efficient Mixed-Integer-Informed Obstacle Avoidance in Real-Time	13767
..... <i>A. L. Gratzner, M. M. Broger, A. Schirrer, and S. Jakubek</i>	
A Simulation Platform for Truck Platooning Evaluation in an Interactive Traffic Environment	13785
..... <i>J. Hu, X. Yan, G. Wang, M. Tu, X. Zhang, H. Wang, D. Gruyer, and J. Lai</i>	
Efficient Uncertainty-Aware Collision Avoidance for Autonomous Driving Using Convolutions	13805
..... <i>C. Zhang, X. Wu, J. Wang, and M. Song</i>	
A Vision-Transformer-Based Convex Variational Network for Bridge Pavement Defect Segmentation	13820
..... <i>H. Qi, X. Kong, Z. Jin, J. Zhang, and Z. Wang</i>	
OUR-Net: A Multi-Frequency Network With Octave Max Unpooling and Octave Convolution Residual Block for Pavement Crack Segmentation	13833
..... <i>P. Li, M. Wang, Z. Fan, H. Huang, G. Zhu, and J. Zhuang</i>	
DSANet-KD: Dual Semantic Approximation Network via Knowledge Distillation for Rail Surface Defect Detection	13849
..... <i>W. Zhou, J. Hong, X. Ran, W. Yan, and Q. Jiang</i>	

YOLC: You Only Look Clusters for Tiny Object Detection in Aerial Images	13863
..... C. Liu, G. Gao, Z. Huang, Z. Hu, Q. Liu, and Y. Wang	
Hierarchical Competition Learning for Pairwise Wheel Grounding Points Estimation	13876
..... Y. Zhang and Z. Yuan	
A Coarse-to-Fine Deep Learning Based Framework for Traffic Light Recognition	13887
..... Z. Yao, Q. Liu, J. Fu, Q. Xie, B. Li, Q. Ye, and Q. Li	
A Stochastic Predictive Adaptive Cruise Control System With Uncertainty-Aware Velocity Prediction and Parameter Self-Learning	13900
..... J. Wang, X. Gong, P. Wang, Y. Wang, R. Wang, L. Guo, Y. Hu, and H. Chen	
Integrating the Safety Control Against Cyber-Attacks on the Global Information in Coupled Map Car-Following Model Under Connected Vehicles Platoon Environment	13914
..... G. Peng, X. Li, and H. Tan	
UP-CrackNet: Unsupervised Pixel-Wise Road Crack Detection via Adversarial Image Restoration	13926
..... N. Ma, R. Fan, and L. Xie	
Evaluating Pedestrian Trajectory Prediction Methods With Respect to Autonomous Driving	13937
..... N. Uhlemann, F. Fent, and M. Lienkamp	
Fine-MVO: Toward Fine-Grained Feature Enhancement for Self-Supervised Monocular Visual Odometry in Dynamic Environments	13947
..... W. Wei, Y. Ping, J. Li, X. Liu, and Y. Zhou	
Driver-Centric Data-Driven Model Predictive Vehicular Platoon With Longitudinal-Lateral Dynamics	13961
..... Y. Wu, Z. Zuo, Y. Wang, Q. Han, J. Li, and H. Xu	
A Study of TMA Aircraft Conflict-Free Routing and Operation: With Mixed Integer Linear Programming, Multi-Agent Path Finding, and Metaheuristic-Based Neighborhood Search	13976
..... Y. Zhang, S. Zhang, Y. Zhang, and Y. Yin	
Small Fixed-Wing Unmanned Aerial Vehicle Path Following Under Low Altitude Wind Shear Disturbance	13991
..... Z. Zhang, C. He, H. Chen, Y. Zhang, H. Wang, Y. Cai, L. Chen, H. Li, and T. Lu	
Supapixel-Guided Multi-Type Rail Segmentation via Contextual Information Aggregation	14004
..... X. Ni, P. W. Fieguth, Z. Ma, B. Shi, Y. Qiu, Y. Chen, and H. Liu	
SR-Adv: Salient Region Adversarial Attacks on 3D Point Clouds for Autonomous Driving	14019
..... S. Zheng, W. Liu, Y. Guo, Y. Zang, S. Shen, C. Wen, M. Cheng, P. Zhong, and C. Wang	
MG-GCT: A Motion-Guided Graph Convolutional Transformer for Traffic Gesture Recognition	14031
..... X. Guo, Q. Zhu, Y. Wang, and Y. Mo	
Joint Energy and Completion Time Difference Minimization for UAV-Enabled Intelligent Transportation Systems: A Constrained Multi-Objective Optimization Approach	14040
..... C. Peng, Z. Wu, X. Huang, Y. Wu, J. Kang, Q. Huang, and S. Xie	
Real-Time Vehicle Tracking-Based Data Forwarding Using RLS in Vehicular Named Data Networking	14054
..... S. A. Khan and H. Lim	
Adaptive and Interactive Multi-Level Spatio-Temporal Network for Traffic Forecasting	14070
..... Y. Zhang, P. Wang, B. Wang, X. Wang, Z. Zhao, Z. Zhou, L. Bai, and Y. Wang	
High Precision Time Synchronization Strategy for Low-Cost Embedded GNSS/MEMS-IMU Integrated Navigation Module	14087
..... P. Yan, J. Jiang, H. Tan, Q. Zheng, and J. Liu	
Optimizing Train-to-Train Rescue and Rescheduling in Metro Systems	14100
..... Z. Wang, T. Tang, S. Su, A. D'Ariano, T. Bosi, and B. Su	
Learning Driver-Irrelevant Features for Generalizable Driver Behavior Recognition	14115
..... H. Gao, M. Hu, and Y. Liu	
Electric Vehicle Next Charge Location Prediction	14128
..... R. Marlin, R. Jurdak, A. Abuadbba, S. Ruj, and D. Miller	
RIS-Assisted Multi-Aperture FSO Communication Network for High-Speed Train: Second-Order Statistical Analysis	14140
..... A. Girdher and A. Bansal	
Improving Urban Travel Time Estimation Using Gaussian Mixture Models	14154
..... A. Gemma, L. Mannini, U. Crisalli, and E. Cipriani	
Aircraft Navigation in GNSS-Denied Environments via Radio SLAM With Terrestrial Signals of Opportunity	14164
..... Z. M. Kassas, N. Khairallah, J. J. Khalife, C. Lee, J. Jurado, S. Wachtel, J. Duede, Z. Hoeffner, T. Hulse, R. Quirarte, and R. Tay	
How Far Ahead Should Autonomous Vehicles Start Resolving Predicted Conflicts? Exploring Uncertainty-Based Safety-Efficiency Trade-Off	14183
..... G. Li, Z. Li, V. L. Knoop, and J.W.C. van Lint	
Enhancing Robustness of Deep Reinforcement Learning Based Adaptive Traffic Signal Controllers in Mixed Traffic Environments Through Data Fusion and Multi-Discrete Actions	14196
..... T. Yang and W. Fan	
Unsupervised Reinforcement Learning for Multi-Task Autonomous Driving: Expanding Skills and Cultivating Curiosity	14209
..... Z. Ma, X. Liu, and Y. Huang	
Efficient Alternative Route Planning in Road Networks	14220
..... A. Fahmin, B. Shen, M. A. Cheema, A. N. Toosi, and M. E. Ali	

Robust Trajectory and Resource Allocation for UAV Communications in Uncertain Environments With No-Fly Zone: A Deep Learning Approach	W. Lee and K. Lee	14233
AnonMAKE: Toward Secure and Anonymous Mutually Authenticated Key Exchange Protocol for Vehicular Communications	A. Karati and L.-C. Chang	14245
Joint Optimization of UAV Trajectory and Communication Resources With Complete Avoidance of No-Fly-Zones	K. Heo, G. Park, and K. Lee	14259
Landmark-Based Vehicle Self-Localization Using Automotive Polarimetric Radars	F. Weishaupt, J. F. Tilly, N. Appenrodt, P. Fischer, J. Dickmann, and D. Heberling	14266
A Physics-Informed and Attention-Based Graph Learning Approach for Regional Electric Vehicle Charging Demand Prediction	H. Qu, H. Kuang, Q. Wang, J. Li, and L. You	14284
A Coordination Graph Based Framework for Network Traffic Signal Control	H. Zhu, F. Sun, K. Tang, T. Han, and J. Xiang	14298
Scalable Algorithms for Bicriterion Trip-Based Transit Routing	P. Agarwal and T. Rambha	14313
Ground-VIO: Monocular Visual-Inertial Odometry With Online Calibration of Camera-Ground Geometric Parameters	Y. Zhou, X. Li, S. Li, X. Wang, and Z. Shen	14328
A Spatial-State-Based Omni-Directional Collision Warning System for Intelligent Vehicles	W. Zhao, S. Gong, D. Zhao, F. Liu, N. Sze, M. Quddus, and H. Huang	14344
A Parallel Genetic Algorithm With Variable Neighborhood Search for the Vehicle Routing Problem in Forest Fire-Fighting	X. Li, N. Chen, H. Ma, F. Nie, and X. Wang	14359
Transportation Mode Recognition Based on Low-Rate Acceleration and Location Signals With an Attention-Based Multiple-Instance Learning Network	C. Siargkas, V. Papapanagiotou, and A. Delopoulos	14376
Optimal Drive-By Sensing in Urban Road Networks With Large-Scale Ridesourcing Vehicles	S. Guo and X. Qian	14389
Robust Longitudinal Control for Vehicular Platoons Using Deep Reinforcement Learning	A. Alves Neto and L. A. Mozelli	14401
State-Space Modeling and Feedback Control for Real-Time Automatic Train Timetable Rescheduling of Intercity HSRs	J. Kang, D. Cui, X. Dai, H. Zhao, Y. Hu, and T. Chai	14411
Large-Scale High-Altitude UAV-Based Vehicle Detection via Pyramid Dual Pooling Attention Path Aggregation Network	Z. Ying, J. Zhou, Y. Zhai, H. Quan, W. Li, A. Genovese, V. Piuri, and F. Scotti	14426
Transductive Transfer Learning-Assisted Hybrid Deep Learning Model for Accurate State of Charge Estimation of Li-Ion Batteries in Electric Vehicles	S. K. R. Moosavi, M. H. Zafar, A. Saadat, Z. Abaid, W. Ni, A. Jamalipour, and F. Sanfilippo	14445
Energy-Efficient Cooperative Secure Communications in mmWave Vehicular Networks Using Deep Recurrent Reinforcement Learning	Y. Ju, Z. Gao, H. Wang, L. Liu, Q. Pei, M. Dong, S. Mumtaz, and V. C. M. Leung	14460
Braking Torque Distribution Reconfiguration Strategy of Vehicle With Faults of In-Wheel Motor Drive System	C. Xing, Y. Zhu, J. Wang, and Y. Lin	14476
Fast and Robust LiDAR-Inertial Odometry by Tightly-Coupled Iterated Kalman Smoother and Robocentric Voxels	J. Liu, Y. Zhang, X. Zhao, Z. He, W. Liu, and X. Lv	14486
Converting Urban Trips to Multi-Dimensional Signals to Improve Trip Purpose Inference	M. Shojaaee Zade, M. Mesbah, M. Habibian, and H. Farooqi	14497
Toward Efficient and Secure Object Detection With Sparse Federated Training Over Internet of Vehicles	Y. Qian, L. Rao, C. Ma, K. Wei, M. Ding, and L. Shi	14507
Toward Explainable End-to-End Driving Models via Simplified Objectification Constraints	C. Zhang, D. Deguchi, J. Chen, and H. Murase	14521
Smart Contract-Based Decentralized Data Sharing and Content Delivery for Intelligent Connected Vehicles in Edge Computing	C. Li, Y. Zhang, J. Wu, Y. Luo, and S. Yu	14535
Deep Learning Approach for Driver Speed Intention Recognition Based on Naturalistic Driving Data	K. Cheng, D. Sun, J. Jian, D. Qin, C. Chen, G. Liao, Y. Kan, and C. Lv	14546
CD-BASA: An Efficient Cross-Domain Batch Authentication Scheme Based on Blockchain With Accumulator for VANETs	Q. Zhong, X. Zhao, Y. Xia, and X. Liu	14560
Privacy-Preserving Distributed Optimal Control for Vehicular Platoon With Quantization	Y. He, Y. Chen, C. Pan, and I. Ali	14572
An Improved NSGAI for Integrated Container Scheduling Problems With Two Transshipment Routes	L. Zhong, W. Li, K. Gao, L. He, and Y. Zhou	14586
PU-Ray: Domain-Independent Point Cloud Upsampling via Ray Marching on Neural Implicit Surface	S. Lim, K. El-Basyouny, and Y. H. Yang	14600

Travel Demand Forecasting: A Fair AI Approach	<i>X. Zhang, Q. Ke, and X. Zhao</i>	14611
Night-Time Vehicle Detection Based on Hierarchical Contextual Information	<i>H. Zhang, K.-F. Yang, Y.-J. Li, and L. L.-H. Chan</i>	14628
Dynamic Rescheduling for Optimal Transportation Incident Management in Inland Waterways	<i>J. M. Nadales, D. Muñoz de la Peña, D. Limon, and T. Alamo</i>	14642
Tracking Control for High-Speed Train With Coupler Constraints	<i>J.-P. Zhang, H. Yang, K. Zhang, and C.-H. Xie</i>	14654
RSO-SLAM: A Robust Semantic Visual SLAM With Optical Flow in Complex Dynamic Environments	<i>L. Qin, C. Wu, Z. Chen, X. Kong, Z. Lv, and Z. Zhao</i>	14669
OTCLDA: Optimal Transport and Contrastive Learning for Domain Adaptive Semantic Segmentation	<i>Q. Fan, X. Shen, S. Ying, and S. Du</i>	14685
A Security Protocol for Vehicle Platoon Verification Using Optical Camera Communications	<i>M. Plattner, E. Sonnleitner, and G. Ostermayer</i>	14698
EPQ-GAN: Evolutionary Perceptual Quality Assessment Generative Adversarial Network for Image Dehazing	<i>K. Ashwini, H. Nenavath, and R. K. Jatoth</i>	14710
YOLOX-RDD: A Method of Anchor-Free Road Damage Detection for Front-View Images	<i>J. Li, Z. Qu, S.-Y. Wang, and S.-F. Xia</i>	14725
A Hybrid Reinforcement Learning-Based Method for Generating Privacy-Preserving Trajectories in Low-Density Traffic Environments	<i>Z. Zhang, W.-C. Wong, and B. Sikdar</i>	14740
Virtual Inertia-Based Control and Stability Analysis for Proton Exchange Membrane Fuel Cell Hybrid Electric Vehicles	<i>X. Li, Q. Li, T. Wang, W. Chen, and Z. You</i>	14758
Data-Driven Cooperative Differential Game Based Energy Management Strategy for Hybrid Electric Propulsion System of a Flying Car	<i>S. Ruan, Y. Ma, Z. Wei, C. Zhang, and C. Xiang</i>	14770
Social-Aware Assisted Edge Collaborative Caching Based on Deep Reinforcement Learning Joint With Digital Twin Network in Internet of Vehicles	<i>G. Chen, W. Duan, J. Sun, Q. Zeng, and Y.-D. Zhang</i>	14785
Characteristics of Air Traffic Flow in Terminal Airspace: A Multiplex Recurrence Network Analysis	<i>F. Jiang and Z. Zhang</i>	14803
Adaptive Dual-Channel Event-Triggered Fuzzy Control for Autonomous Underwater Vehicles With Multiple Obstacles Environment	<i>S. Liu, R. Zhang, D. Zhao, and H. Song</i>	14816
Decomposition-Based Multiobjective Evolutionary Optimization With Tabu Search for Dynamic Pickup and Delivery Problems	<i>J. Cai, Q. Zhu, Q. Lin, Z. Ming, and K. C. Tan</i>	14830
Aperiodic Coordination Scheduling of Multiple PPLs in Shipboard Integrated Power Systems	<i>B. Qin, H. Wang, W. Li, F. Li, W. Wang, and T. Ding</i>	14844
Traffic Parameters Estimation With Partial Vehicle Trajectories by the Iterative Partial Backpropagation Maximum Likelihood Estimation (IPB-MLE) Framework	<i>K. Du, S. Wang, and H. K. Lo</i>	14855
Indian Traffic Sign Detection and Classification Through a Unified Framework	<i>R. Uikey, H. R. Lone, and A. Agarwal</i>	14866
Optimizing Bus Operations at Autonomous Intersection With Trajectory Planning and Priority Control	<i>W. Wu, M. Xiong, T. Liu, J. Sun, and Y. Li</i>	14876
MGINS: A Lane-Level Localization System for Challenging Urban Environments Using Magnetic Field Matching/GNSS/INS Fusion	<i>X. Niu, L. Ding, Y. Wang, and J. Kuang</i>	14890
SVCE: Shapley Value Guided Counterfactual Explanation for Machine Learning-Based Autonomous Driving	<i>M. Li, H. Sun, Y. Huang, and H. Chen</i>	14905
Preserving Label-Related Domain-Specific Information for Cross-Domain Semantic Segmentation	<i>M. Liao, S. Tian, Y. Zhang, G. Hua, W. Zou, and X. Li</i>	14917
Heading Measurement Frame Based on Atmospheric Scattering Beams for Intelligent Vehicle	<i>X. Wu, H. Cao, C. Wang, C. Shen, J. Tang, and J. Liu</i>	14932
Designing High Speed Weigh-in-Motion System With Principal Component Regression in Wallonia (Belgium) Toward Direct Weight Enforcement	<i>L. Warscotte and J. Boreux</i>	14948
An Efficient Image Privacy Preservation Scheme for Smart City Applications Using Compressive Sensing and Multi-Level Encryption	<i>X. He, L. Li, H. Peng, and F. Tong</i>	14958
Dynamic Voxels Based on Ego-Conditioned Prediction: An Integrated Spatio-Temporal Framework for Motion Planning	<i>T. Zhang, M. Fu, W. Song, Y. Yang, and A. Alahi</i>	14973
Metaverse Meets Intelligent Transportation System: An Efficient and Instructional Visual Perception Framework	<i>J. Wang, Y. Chen, X. Ji, Z. Dong, M. Gao, and C. S. Lai</i>	14986

Blockchain Assisted Intra-Twin and Inter-Twin Authentication Scheme for Vehicular Digital Twin System	<i>D. Gautam, G. Thakur, P. Kumar, A. K. Das, and Y. Park</i>	15002
MECFNet: Reconstruct Sharp Image for UAV-Based Crack Detection	<i>C. Liu, J. Zhao, C. Zhu, X. Xia, and H. Long</i>	15016
Privacy-Preserving Cruise Control for Heterogeneous Platoon Vehicle System Under Actuator Faults and Uncertainties	<i>J. Liu and J. Dong</i>	15029
MDD-ShipNet: Math-Data Integrated Defogging for Fog-Occlusion Ship Detection	<i>N. Wang, Y. Wang, Y. Feng, and Y. Wei</i>	15040
Problem-Specific Knowledge Based Multi-Objective Meta-Heuristics Combined Q-Learning for Scheduling Urban Traffic Lights With Carbon Emissions	<i>Z. Lin, K. Gao, N. Wu, and P. N. Suganthan</i>	15053
Overtaking Feasibility Prediction for Mixed Connected and Connectionless Vehicles	<i>L. Zhao, H. Qian, A. Hawbani, A. Y. Al-Dubai, Z. Tan, K. Yu, and A. Y. Zomaya</i>	15065
Multi-Scale Semantic Map Distillation for Lightweight Pavement Crack Detection	<i>X. Wang, Z. Mao, Z. Liang, and J. Shen</i>	15081
BEVRefiner: Improving 3D Object Detection in Bird's-Eye-View via Dual Refinement	<i>B. Wang, H. Zheng, L. Zhang, N. Liu, R. M. Anwer, H. Cholakkal, Y. Zhao, and Z. Li</i>	15094
Detection and Statistics System of Pavement Distresses Based on Street View Videos	<i>Z. Zhang, F. Liu, Y. Huang, and Y. Hou</i>	15106
<hr/> SHORT PAPERS		
Design of Intelligent Control Under Machine Learning Supervision and Signal Compression Mechanism Design for NCSs Under DoS Attacks	<i>X. Cai, K. Shi, Y. Sun, J. Cao, O.-M. Kwon, C. Qiao, and Z. Tian</i>	15116
Intelligent Reflective Surface Assisted Integrated Sensing and Wireless Power Transfer	<i>Z. Li, Z. Zhu, Z. Chu, Y. Guan, D. Mi, F. Liu, and L.-L. Yang</i>	15122

Joint Energy and Completion Time Difference Minimization for UAV-Enabled Intelligent Transportation Systems: A Constrained Multi-Objective Optimization Approach

Chaoda Peng^{ID}, *Member, IEEE*, Zexiong Wu, Xumin Huang^{ID}, Yuan Wu^{ID}, *Senior Member, IEEE*,
Jiawen Kang, *Senior Member, IEEE*, Qiong Huang^{ID}, and Shengli Xie^{ID}, *Fellow, IEEE*

Abstract—An unmanned aerial vehicle (UAV)-enabled intelligent transportation system utilizes a set of UAVs to collect and process surveillance data for transportation management. Subsequently, the processing results of the UAVs are transmitted to a control center that makes a centralized transportation management decision based on the fusion of all processing results. When performing the monitoring tasks, the UAVs can access to an edge server for offloading. To reduce the energy consumption and improve the fusion performance, the control center schedules the UAVs to perform the tasks in an energy-efficient manner while synchronizing the completion time of the UAVs. As a result, the control center studies a constrained multi-objective optimization problem (CMOP), in which two objectives, i.e., the total energy consumption of the UAVs and total completion time difference among the UAVs, are simultaneously considered. To tackle the CMOP, we develop an improved constrained multi-objective evolutionary algorithm. Particularly, we design an improved genetic operator and repairing constraint-handling technique to improve the overall performance of the proposed algorithm in seeking Pareto optimal solutions for the CMOP. Numerical results demonstrate that compared with the baseline algorithms, the proposed algorithm has great advantages in finding better solutions with the enhanced diversity and convergence for the CMOP.

Index Terms—UAV-enabled intelligent transportation system, energy optimization, time difference minimization, constrained multi-objective optimization, evolutionary algorithm.

I. INTRODUCTION

INTELLIGENT Transportation System (ITS) relies on recent advances in the area of Internet of Things (IoT) to facilitate data collection, data analysis, and information fusion for improving the transportation management. Due to great advantages of easy deployment, flexible mobility, and lower cost, unmanned aerial vehicles (UAVs) have been widely applied in diverse IoT services and applications, e.g., performing data collection on land [1] or over the sea [2], providing aerial edge computing [3], [4], or acting as aerial base stations [5], [6] for ground IoT devices. UAVs can also be scheduled to support diverse traffic surveillance applications in ITS [7]. In UAV-enabled ITS, UAVs serve as aerial agents for accident reporting, flying police eyes to track target vehicles, and airborne cameras to monitor traffic flow and road conditions [8]. In a mission planning period, a control center employs a set of UAVs to reach specific monitoring locations and stay hovering to collect and process the surveillance data. Besides, mobile edge computing brings cloud computing capability closer to IoT devices and deploys an edge server at the network edge to support computation offloading with lower service latency, less bandwidth consumption, and improved data security [9]. When processing a monitoring task, a UAV can locally perform or access a nearby edge server to offload the task. Subsequently, data processing results are transmitted to the control center for the fusion. To ensure the overall performance of the mission planning, the control center should consider a joint optimization problem of UAV association, task offloading, and resource allocation for UAV-enabled ITS.

However, a number of challenges still need to be addressed. First, different UAVs have obvious differences in flight distance and parameters, hardware configuration, and energy consumption profiles while different task executions necessitate different monitoring locations, service programs, computing workloads, and time. The association between the UAVs and tasks is optimized to reduce the total energy consumption of the UAVs (i.e., the first objective denoted as G_1). Second,

Manuscript received 17 July 2023; revised 17 February 2024 and 20 April 2024; accepted 22 April 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62202177, Grant 62001125, and Grant 62102099; in part by the National Key Research and Development Program of China under Grant 2020YFB1807802; in part by Guangzhou Basic Research Program under Grant 2023A04J0340 and Grant 202201010576; in part by the Science and Technology Development Fund of Macau, SAR, under Grant 0158/2022/A; and in part by MYRG-GRG2023-00083-IOTSC-UMDF. The Associate Editor for this article was L. Wang. (*Corresponding authors: Xumin Huang; Yuan Wu.*)

Chaoda Peng, Zexiong Wu, and Qiong Huang are with the College of Mathematics and Informatics, South China Agricultural University, Guangzhou 510642, China (e-mail: ChaodaPeng@scau.edu.cn; zexiongwu@stu.scau.edu.cn; qhuang@scau.edu.cn).

Xumin Huang is with the School of Automation, Guangdong University of Technology, Guangzhou 510006, China, and also with the State Key Laboratory of Internet of Things for Smart City, University of Macau, Taipa, Macau, China (e-mail: huangxu_min@163.com).

Yuan Wu is with the State Key Laboratory of Internet of Things for Smart City and the Department of Computer and Information Science, University of Macau, Taipa, Macau, China (e-mail: yuanwu@um.edu.mo).

Jiawen Kang and Shengli Xie are with the School of Automation, Guangdong University of Technology, Guangzhou 510006, China (e-mail: kavinkang@gdut.edu.cn; shlxi@gdut.edu.cn).

Digital Object Identifier 10.1109/TITS.2024.3395993

1558-0016 © 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See <https://www.ieee.org/publications/rights/index.html> for more information.

the control center aims at facilitating the unified scheduling of UAVs and particularly considers the total completion time difference among all employed UAVs (i.e., the second objective denoted as G_2). In the conventional multi-UAV-enabled mobile edge computing, the joint optimization problem of task offloading and resource allocation between multiple UAVs and the edge server has been investigated for the mission planning period, according to different optimization objectives. In the previous works, different UAVs are selected to perform the tasks within different deadlines, and data processing results of different tasks are independently utilized. But the control center in UAV-enabled ITS necessitates to fuse the data processing results of all UAVs to make a centralized transportation management decision. To alleviate the negative effect of temporal asynchronization on the fusion performance, the control center would like to simultaneously receive the data processing results from the UAVs. To this end, an effective method is required to reduce the total completion time difference among all employed UAVs. This harmonization also ensures that the employed UAVs concurrently enter the standby state for quickly joining the next mission together such that the control center can always handle sufficient UAVs on demand in each mission planning period. Thus, the control center has two objectives G_1 and G_2 that are jointly optimized. Last but not least, the control center necessitates a set of solutions rather than a single one to handle different tradeoffs between the two objectives. We are motivated to study a multi-objective optimization problem involving with the two objectives such that a variety of solutions can be provided to satisfy diverse preferences of the control center. To the best of our knowledge, such a multi-objective optimization problem has not been investigated yet in the community of UAV-enabled ITS.

To address the above challenges, we study the joint optimization of energy consumption and completion time difference for UAV-enabled ITS from a multi-objective optimization perspective. In a mission planning period, the control center has a set of standby UAVs for the executions of several monitoring tasks. UAVs are dispatched to depart from their start locations and reach to specific monitoring locations to perform the associated tasks. To facilitate the task processing, a part of the tasks are offloaded to the nearest edge server while the residual ones are locally performed. For the above system model, we investigate a constrained multi-objective optimization problem (CMOP) that involves the simultaneous optimization of the two objectives G_1 and G_2 , which are presented to achieve the energy-efficient data collection and processing, and balance time consumption for the employed UAVs, respectively. In the CMOP, we also consider feasible constraints regarding UAV association, task offloading, and resource allocation.

Solving the CMOP requires an efficient algorithm to seek a set of Pareto optimal solutions by satisfying the constraints, where a Pareto optimal solution refers to a solution which no improvement can be made in one objective without worsening the other objectives [10]. In practice, we obtain the non-dominated solutions to approximate the Pareto optimal solutions. More details of the above two kinds of solutions are shown in Section IV. Evolutionary algorithms have been

widely developed for solving different CMOPs over the past decades, owing to the inherent characteristics such as easy implementation without the need for gradient information and the capacity of finding global optima [10], [11]. In an evolutionary algorithm, genetic operators and constraint-handling techniques are two fundamental components that significantly affect the performance of the algorithm in terms of obtaining a set of Pareto optimal solutions [12], [13]. To address the studied CMOP, we develop a constrained multi-objective evolutionary algorithm based on an improved genetic operator and repairing constraint-handling technique under the framework of CMOEA/D-CDP [14]. The improved genetic operator based on the data types of the optimized decision variables is designed to enhance the search ability of the proposed algorithm. The repairing constraint-handling technique is designed to convert infeasible solutions into feasible ones, accelerating the convergence of the proposed algorithm towards feasibility. The main contributions of this paper are summarized as follows:

- We present a multi-source information fusion system model for UAV-enabled ITS where the control center dispatches a set of UAVs to perform the monitoring tasks. In a mission planning period, the UAVs collect the surveillance data and collaborate with the edge server to generate the data processing results. Then the results are gathered to the control center for the fusion.
- We study a CMOP to achieve the simultaneous optimization on the task and UAV sides. To facilitate the unified scheduling of the UAVs, UAV association, task offloading, and resource allocation are jointly optimized to complete the monitoring tasks in an energy-efficient manner while achieving the time balancing among all employed UAVs.
- We propose a constrained multi-objective evolutionary algorithm with an improved genetic operator and repairing constraint-handling technique to address the CMOP. The numerical results demonstrate that compared with the baseline algorithms, the proposed algorithm has great advantages in seeking a set of better non-dominated solutions with the enhanced diversity and convergence for the CMOP.

The remainder of this paper is organized as follows. Section II provides a comprehensive overview of recent works. Section III presents the system model and formulates a CMOP for UAV-enabled ITS. Section IV introduces the proposed algorithm based on CMOEA/D-CDP to solve the CMOP. The simulations and performance evaluations are shown in Section V. Section VI concludes this paper.

II. RELATED WORK

A. UAV-Enabled ITS

Research efforts have been devoted to a variety of optimization schemes for UAV-enabled ITS. For example, a UAV can be applied as an aerial base station to assist the terrestrial communication and perform computational tasks for vehicles. UAVs were scheduled to serve mobile vehicles along an optimal path and enhance the downlink throughput

of the Internet of Vehicles (IoV). A UAV was assumed to serve a single vehicle in a time slot, where the association between the vehicles and UAVs with the power allocations was optimized to improve the UAV-to-vehicle communication performance [15]. A stable relay selection problem was generated when a UAV becomes a relay for IoV routing protocols. Considering dynamic mobility and reputation values of all UAVs, the problem was tackled by using a matching game theoretic approach [16]. UAV-assisted communication was exploited to provide continuous line-of-sight (LoS) links to vehicles when they prepared to offload the tasks to other vehicles or nearby edge servers [17]. UAVs could be employed as flying base stations with caching capability to improve the efficiency of data dissemination and facilitate the file sharing process among the vehicles [18].

Besides, the UAV deployment enables ground vehicles to gain aerial computing services. Computation-intensive tasks in ITS environment were first collected by task gathering nodes, and a UAV flew above the nodes to process the tasks together [19]. Furthermore, a UAV was assigned with a dual role, i.e., task performer and mobile relay, where the UAV was scheduled to process offloading tasks of some vehicles, in the meantime, played as a mobile relay that helps forward offloading tasks of some vehicles to the nearest edge server [20]. When providing aerial computing for platooning vehicles, the UAV can receive energy replenishment from the vehicles when necessary [21]. In UAV-assisted aerial computing networks, UAVs may harvest available computing resources from the surrounding entities to enhance the computing capabilities. In this regard, UAVs served as mobile data collectors in smart cities can offload the data to proper vehicles for remote processing. The optimal matching between UAVs and vehicles was investigated by modeling the transaction process of offloading data as a bargaining game [22]. The similar problem was studied in a post-disaster rescue scenario [23]. In UAV-enabled traffic monitoring, network nodes with idle resources were employed to complete a part of computational tasks from the UAVs [24].

B. Multi-Objective Optimization for UAV Scheduling

Nowadays, UAV scheduling has been investigated from a multi-objective optimization perspective in diverse applications. Multiple UAVs were scheduled for data collection, and a multi-objective optimization problem of UAV deployment was studied to optimize the network-wide uplink throughput while reducing the total energy consumption of all UAVs [25]. In [26], the UAVs were scheduled for collaborative beamforming, and the tradeoff between the data transmission performance and energy consumption was also tackled by using a multi-objective optimization approach. In [27], a single UAV was dispatched to charge a set of sensor nodes that further utilized the harvested energy to submit sensory data to the UAV. The achievable sum rate of all nodes in the uplink and the total transmit power of the UAV in the downlink were simultaneously optimized. Compared with the above works, the following works consider the mobility of a UAV. In [28], the UAV was scheduled to successively visit target devices with the fly-hover-fly trajectory. When staying hovering, the

UAV performed the data collection and wireless charging tasks. The control policies of the UAV over multiple objectives was addressed by using a deep reinforcement learning algorithm. The authors considered three objectives for the UAV but transformed the original multi-objective optimization problem into a single-objective optimization problem by a weighting-sum method. With the increasing problem size, it is difficult to determine the weighting parameters among the objectives. In [29], the UAV was deployed to sequentially visit the specific waypoints, and perform offloading tasks for local users. Subsequently, the joint optimization of energy-efficient offloading and safe path planning for the UAV was studied by a multi-objective evolutionary algorithm.

C. Performance Comparison

Compared with the previous works, we aim at scheduling the UAVs to collaboratively perform the tasks in an energy-efficient manner while minimizing the total completion time difference among the UAVs using a multi-objective optimization approach. Most of the previous works study a UAV-enabled task offloading system where the UAVs perform the tasks that are completely unassociated with each other, and output results of the tasks are independently utilized. Different from the works, our work considers a UAV-enabled multi-source information fusion system where all data processing results of the UAVs need to be integrated together and be temporally aligned before the fusion. The control center expects the UAVs to complete the task processing in a synchronized manner such that the temporal aligning operation on the data processing results is facilitated to ensure the fusion performance. As a result, our work formulates a CMOP to simultaneously minimize the total energy consumption of the UAVs and total completion time difference among the employed UAVs. After that, we develop a constrained multi-objective evolutionary algorithm under the framework of CMOEA/D-CDP to tackle the CMOP. An improved genetic operator and repairing constraint-handling technique are designed to improve the performance of the proposed algorithm in terms of the convergence and diversity of the non-dominated solutions.

III. SYSTEM MODEL AND PROBLEM FORMULATION

A. System Model

As shown in Fig. 1, a control center refers to different information at the different traffic junctions to make a centralized transportation management decision. According to the detection demand, the control center dispatches sufficient UAVs to hover above a number of interesting ground locations and collect the surveillance data, e.g., image and video. For example, camera-equipped UAVs are scheduled to detect free parking spaces in a parking area, issue an accident report of an intersection, and predict the dynamic traffic flow in a highway. The UAVs further collaborate with an edge server to process the data to acquire available information that is gathered to the control center for the multi-source data fusion. We provide more details of the network entities as follows:

- *Edge server*: The edge server provides service caching for UAV-enabled ITS. Each monitoring task is associated

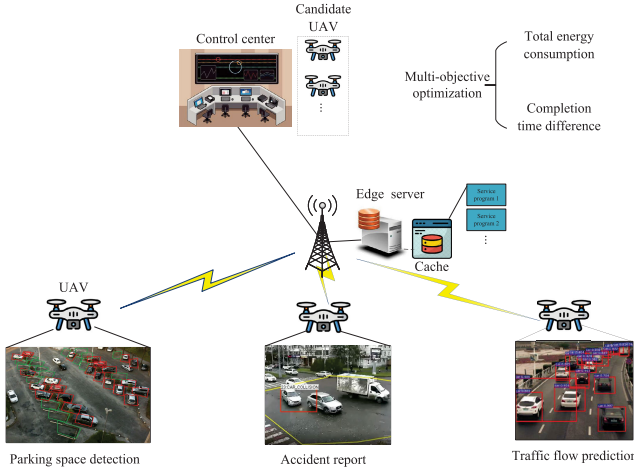


Fig. 1. UAV-enabled ITS.

with a surveillance service, in which the task is run by a service program, including executable.EXE files, library, and database. A UAV can be assigned with an arbitrary task in different mission planning periods. The UAV with limited storage space is difficult to store all service programs in advance. As an option, the control center caches popular service programs on an edge server co-located with a base station and stores the residual ones on a remote cloud server. Subsequently, the UAVs retrieve the service programs from the edge/cloud servers when necessary.

- **UAV:** A UAV is responsible for the data collection and processing. When the UAV is assigned with a monitoring task, the UAV flies to the monitoring location at a constant velocity, and stays hovering to gather sufficient surveillance data. After that, the UAV can locally process the collected data by requesting the service program from the edge server, or directly transmit the data to the edge server for remote processing. If the data processing task necessitates heavy workloads, it is more suitable to let the UAV offload the task to the edge server so as to avoid the large task execution delay.
- **Control center:** The control center is a centralized manager that is responsible for the unified scheduling of the UAVs. The control center holds prior knowledge of any UAV, e.g., energy consumption profile, to achieve the energy-efficient task processing by properly assigning the tasks to the UAVs and allocating the available bandwidth/computing resources among them. In addition, the control center collects all data processing results of the UAVs for the fusion and performs the temporal aligning operation to alleviate the negative effect of temporal asynchronization on the fusion performance. This motivates the control center to take into account the total completion time difference among all employed UAVs instead of the specific task completion time of any UAV. As mentioned above, the control center has two objectives, i.e., G_1 and G_1 . The control center adopts a multi-objective optimization method to realize the two objectives according to different preferences between them.

B. Mathematical Model

We provide basic mathematical formulations on a UAV and the edge sever as follows.

The control center has I standby UAVs, where a UAV is indexed by i , $1 \leq i \leq I$. There are J monitoring tasks corresponding to J monitoring locations. A monitoring task/location is indexed by j , $1 \leq j \leq J$ and $I \geq J$ in general. For the UAV scheduling, we introduce two binary decisions x_i and $y_{i,j}$. x_i refers to offloading decision of UAV i such that $x_i = 1$ means UAV i accesses the edge server for computation offloading, while $x_i = 0$ means UAV i locally processes the given task. $y_{i,j}$ refers to the association between UAV i and task j . $y_{i,j} = 1$ means that UAV i is scheduled to reach monitoring location j , while $y_{i,j} = 0$ means the UAV is not scheduled to perform task j . To reach a desired location, UAV i flies at a constant velocity v_i . According to the evaluation method in [30], flying power of the UAV denoted as P_i^F is expressed by

$$P_i^F = c_1 v_i^3 + \frac{c_2}{v_i}, \quad (1)$$

where c_1 and c_2 are two coefficients. Considering the straight-line flight, flying time of the UAV for reaching monitoring location j is calculated by $t_{i,j}^F = d_{i,j}/v_i$, where $d_{i,j}$ is the distance between start location of UAV i and monitoring location j . We also consider a constant hovering power of UAV i denoted as P_i^H when the UAV stays hovering.

When $y_{i,j} = 1$, we introduce the communication and energy models of UAV i . In the data collection process, sampling frequency is s_i data samples per second, and sensing power is P_i^S . The UAV needs to collect α_j data samples such that the sensing time of the UAV is $t_{i,j}^S = \alpha_j/s_i$. When UAV i keeps the stable hovering state at the monitoring location j , the communication channel between the UAV and the nearest base station is dominated by a line-of-sight link. Similar to the work [31], we neglect the impact of channel impairments such as shadowing or small-scale fading, and let the channel power gain between the UAV and the nearest base station follow a free-space path loss model as $g_0 d_j^{-2}$, where g_0 denotes the channel power at the reference distance of 1 meter and d_j is the distance between monitoring location j and the base station. Furthermore, we adopt frequency division multiple access to avoid the co-channel interference among multiple UAVs when they simultaneously communicate with the base station. Using the Shannon principle, we measure the achievable uplink and downlink rates of UAV i at the monitoring location j by

$$\begin{cases} r_{i,j}^{UL} = b_i \log_2(1 + \frac{P_i^{TX} g_0 d_j^{-2}}{N_0}) = b_i k_{i,j}^{UL}, \\ r_{i,j}^{DL} = b_i \log_2(1 + \frac{P_{BS}^{TX} g_0 d_j^{-2}}{N_0}) = b_i k_{i,j}^{DL}, \end{cases} \quad (2)$$

where b_i is the available bandwidth; P_i^{TX} and P_{BS}^{TX} are the transmit power of the UAV and base station, respectively; N_0 is the noise power spectrum density. After collecting sufficient data as the input data, the UAV generates a data processing task with computational workloads W_j . It is noted that compared with the input data size, the output data size

is much smaller. Both the time and energy consumption of delivering the output data are neglected.

If $x_i = 0$, we denote this case as “L”. Since the UAV chooses to locally process the task, it retrieves the service program from the edge server or the cloud server. When the UAV downloads the service program, the data transmission time and energy consumption are

$$\begin{cases} t_{i,j}^{COM,L} = \frac{\beta_j}{r_{i,j}^{DL}} + (1 - \gamma_j) \frac{\beta_j}{r_{ES}}, \\ e_{i,j}^{COM,L} = p_i^{RX} \frac{D_j}{r_{i,j}^{DL}}, \end{cases} \quad (3)$$

where β_j is the data size of the service program; γ_j is a binary indicator where $\gamma_j = 1$ means the service program is cached on the edge server, and $\gamma_j = 0$ means that the service program is stored on the cloud server; r_{ES} refers to the data rate between the edge server and cloud server; p_i^{RX} is the receive power of the UAV. It is noted that we neglect the wired data transmission time between the edge server and base station. In the case “L”, the workload processing time of UAV i and energy consumption are

$$\begin{cases} t_{i,j}^{CMP,L} = \frac{W_j}{f_i^L}, \\ e_{i,j}^{CMP,L} = \kappa_i (f_i^L)^2 W_j, \end{cases} \quad (4)$$

where f_i^L is computing capability of UAV i and κ_i is a hardware parameter on the effective switched capacitance depending on the chip architecture. We calculate the total time consumption and total energy consumption of UAV i for completing task j , which are expressed by $T_{i,j}^L$ and $E_{i,j}^L$, respectively. They are given as follows:

$$\begin{cases} T_{i,j}^L = t_{i,j}^F + T_{i,j}^{H,L}, \\ T_{i,j}^{H,L} = t_{i,j}^S + t_{i,j}^{COM,L} + t_{i,j}^{CMP,L}, \\ E_{i,j}^L = p_i^F t_{i,j}^F + p_i^S t_{i,j}^S + e_{i,j}^{COM,L} + e_{i,j}^{CMP,L} + p_i^H T_{i,j}^{H,L}, \end{cases} \quad (5)$$

where $T_{i,j}^{H,L}$ refers to the hovering time of the UAV with respect to $x_i = 0$.

If $x_i = 1$, we denote this case as “O” where the UAV chooses to offload the task. When the UAV transmits the input data, the data transmission time of the UAV and workload processing time on the edge server are

$$\begin{cases} t_{i,j}^{COM,O} = \frac{\alpha_j}{r_{i,j}^{UL}}, \\ t_{i,j}^{CMP,O} = (1 - \gamma_j) \frac{\beta_j}{r_{ES}} + \frac{W_j}{f_i^O}, \end{cases} \quad (6)$$

where f_i^O is the computing capability of the edge server allocated to this offloading task. The energy consumption of the UAV and that of the edge server are

$$\begin{cases} e_{i,j}^{COM,O} = p_i^{TX} \frac{\alpha_j}{r_{i,j}^{UL}}, \\ e_{i,j}^{CMP,O} = \kappa_{ES} (f_i^O)^2 W_j. \end{cases} \quad (7)$$

In the case “O”, the total time consumption and energy consumption of UAV i for completing the task j are expressed by $T_{i,j}^O$ and $E_{i,j}^O$, respectively. They are given as follows:

$$\begin{cases} T_{i,j}^O = t_{i,j}^F + T_{i,j}^{H,O}, \\ T_{i,j}^{H,O} = t_{i,j}^S + t_{i,j}^{COM,O} + t_{i,j}^{CMP,O}, \\ E_{i,j}^O = p_i^F t_{i,j}^F + p_i^S t_{i,j}^S + e_{i,j}^{COM,O} + e_{i,j}^{CMP,O} + p_i^H T_{i,j}^{H,O}, \end{cases} \quad (8)$$

where κ_{ES} is the hardware parameter of the edge server and $T_{i,j}^{H,O}$ refers to the hovering time of the UAV when $x_i = 1$. For the edge server, we pay attention to the energy consumption of performing the offloading workloads.

C. Problem Formulation

In this study, we investigate how to schedule the UAVs to reach all monitoring locations, and further preform the offloading optimizations to simultaneously minimize the total energy consumption of the UAVs and the total completion time difference among the employed UAVs while satisfying the feasible constraints. As mentioned above, the two objectives are denoted as G_1 and G_2 , respectively. To achieve the satisfactory tradeoffs between G_1 and G_2 , we study the joint optimization problem of UAV association, task offloading, and resource allocation problem from a multi-objective optimization perspective.

We define a decision variable vector as $\mathbf{x} = \{x_i, y_{i,j}, b_i, f_i^O\}_{\forall i,j}$. Given \mathbf{x} , we calculate G_1 by

$$G_1(\mathbf{x}) = \sum_{1 \leq j \leq J} \sum_{1 \leq i \leq I} y_{i,j} [(1 - x_i) E_{i,j}^L + x_i E_{i,j}^O]. \quad (9)$$

Let a subset $\mathcal{S} \subseteq \{1, 2, \dots, I\}$ represent the set of J selected UAVs, $\mathcal{S} = \{i | \sum_{j=1}^J y_{i,j} = 1, 1 \leq i \leq I\}$. For a UAV $i \in \mathcal{S}$, we calculate the completion time by

$$\tau_i = \sum_{1 \leq j \leq J} y_{i,j} [(1 - x_i) T_{i,j}^L + x_i T_{i,j}^O], i \in \mathcal{S}. \quad (10)$$

During each mission planning period, the control center gathers data processing results from all employed UAVs for the fusion. The fusion performance at the control center is influenced by the differences in completion time of different tasks assigned to different UAVs. Besides, when the total completion time difference among the employed UAVs is reduced, the control center can promptly adopt the unified scheduling of the UAVs for the next mission planning period. Hence, the control center aims to balance the total completion time difference among the employed UAVs, which is expressed by G_2 as follows:

$$G_2(\mathbf{x}) = \sum_{i \in \mathcal{S}} \frac{|\tau_i - \hat{\tau}|}{\vartheta}, \quad (11)$$

where $\hat{\tau}$ is the average value of $\{\tau_i\}_{\forall i \in \mathcal{S}}$ and ϑ is a presetting reference value.

As a result, we formulate a CMOP for the control center as follows:

$$\min \begin{cases} G_1(\mathbf{x}), \\ G_2(\mathbf{x}), \end{cases}$$

$$\begin{aligned}
\text{s.t. } C_1 : & \sum_{1 \leq i \leq I} y_{i,j} \leq 1, \quad \forall j, \\
C_2 : & \sum_{1 \leq j \leq J} y_{i,j} = 1, \quad \forall i, \\
C_3 : & \sum_{1 \leq j \leq J} \sum_{1 \leq i \leq I} y_{i,j} b_i \leq B, \\
C_4 : & \sum_{1 \leq j \leq J} \sum_{1 \leq i \leq I} y_{i,j} x_i f_i^{ES} \leq F, \\
C_5 : & \sum_{1 \leq j \leq J} \sum_{1 \leq i \leq I} [y_{i,j} (\beta_j + x_i \alpha_i)] \leq S, \\
C_6 : & x_i \in [0, 1], y_{i,j} \in [0, 1], b_i \geq 0, f_i^o \geq 0, \quad \forall i.
\end{aligned} \tag{12}$$

Feasible solutions are derived when several constraints are satisfied. The constraint (C_1) ensures that a single UAV is allocated with one task at most. The constraint (C_2) ensures that each task is completed. The constraint (C_3) ensures that the total bandwidth of all selected UAVs is smaller than an upper bound B , where B is the entire available bandwidth of the base station. The constraint (C_4) ensures that the total number of computing resources allocated to those UAVs with offloading requests is smaller than an upper bound F , where F is the maximal computing capability of the edge server. The constraint (C_5) ensures that in the worst case, the total data storage of all the selected UAVs involving with the offloading input data and requested service programs is smaller than an upper bound S , where S is the storage capacity of the edge server. The constraint (C_6) ensures the feasible domain of the decision variables x_i , $y_{i,j}$, b_i , and f_i^o , respectively.

It is noted that the above problem is a typical multi-objective optimization problem. Some previous works about the UAV networks, e.g., [28], also study the multi-objective optimization problems and propose to transform them into single-objective ones by using the weighting method. However, weighting the objectives is not beneficial to achieve the simultaneous optimization of multiple conflicted objectives, since the weighting method is not suitable to seek a set of distinct Pareto optimal solutions for the multi-objective optimization problems with non-convex Pareto fronts. The weighting method is not applied in this paper. Moreover, the control center as the decision maker necessitates a set of solutions rather than only one solution to handle different tradeoffs among the objectives according to different preferences. Thus, we are motivated to apply a multi-objective optimization approach for the control center.

IV. PROPOSED ALGORITHM

A. Preliminary Study

To tackle the proposed CMOP, we propose a multi-objective evolutionary algorithm with an improved genetic operator to seek promising solutions by adaptively disturbing the decision variables according to their data types and a repairing constraint-handling technique to convert infeasible solutions into feasible ones. The proposed algorithm follows the framework of the decomposition-based constrained multi-objective

evolutionary algorithm (CMOEA/D-CDP) [14]. Prior to introducing the proposed algorithm, we present a preliminary study to show some basic concepts of constrained multi-objective optimization and give a brief introduction of CMOEA/D-CDP.

A typical CMOP is expressed as follows:

$$\begin{aligned}
\min \mathcal{G}(\mathbf{x}) &= (G_1(\mathbf{x}), G_2(\mathbf{x}), \dots, G_m(\mathbf{x}))^T, \\
\text{s.t. } g_i(\mathbf{x}) &\leq 0, \quad i = 1, 2, \dots, q, \\
h_i(\mathbf{x}) &= 0, \quad i = q + 1, \dots, p, \\
\mathbf{x} &\in \mathbb{D},
\end{aligned} \tag{13}$$

where \mathbf{x} is a vector in the decision variable space \mathbb{D} , $\mathcal{G}(\mathbf{x})$ is an objective vector that consists of m objective functions, $g_i(\mathbf{x}) \leq 0$ and $h_i(\mathbf{x}) = 0$ are inequality constraints and equality constraints, respectively. p is the total number of the inequality constraints and equality constraints. The degree of constraint violation $CV(\mathbf{x})$ of an individual \mathbf{x} is given by Eq. (14).

$$\begin{aligned}
CV(\mathbf{x}) &= \sum_{i=1}^p cv_i(\mathbf{x}), \\
cv_i(\mathbf{x}) &= \begin{cases} \max\{0, g_i(\mathbf{x})\}, & i = 1, 2, \dots, q, \\ \max\{0, |h_i(\mathbf{x})| - \delta\}, & i = q + 1, \dots, p, \end{cases}
\end{aligned} \tag{14}$$

where δ is a small value which means the tolerance value for equality constraints. It is notable that $CV(\mathbf{x}) \geq 0$. If $CV(\mathbf{x}) = 0$, \mathbf{x} is a feasible solution. Otherwise, \mathbf{x} is an infeasible solution.

The superiority of one solution over another one is defined by the Pareto dominance, as given in Eq. (16). A solution is called a Pareto optimal solution when no other solution can dominate it (see **Definition 2**). It is worth noting that the optimal solutions found by multi-objective optimization algorithms, e.g., evolutionary algorithms, are the approximate Pareto optimal solutions [14]. In the context of multi-objective optimization, the approximate Pareto optimal solutions obtained by an algorithm are also called non-dominated solutions. The set of Pareto optimal solutions in objective space and decision space is called Pareto front (see **Definition 3**) and Pareto optimal solution set (see **Definition 4**), respectively.

Definition 1 (Pareto dominance): For two solutions $\mathbf{x}, \mathbf{y} \in \mathbb{D}$, if \mathbf{x} and \mathbf{y} satisfy the following relation:

$$\begin{cases} G_i(\mathbf{x}) \leq G_i(\mathbf{y}), & \forall i \in \{1, 2, \dots, m\}, \\ G_i(\mathbf{x}) < G_i(\mathbf{y}), & \exists i \in \{1, 2, \dots, m\}, \end{cases} \tag{16}$$

\mathbf{x} is said to dominate \mathbf{y} , denoted as $\mathbf{x} <_d \mathbf{y}$.

Definition 2 (Pareto optimal solution): For a solution \mathbf{x}^* , if there is no any solution $\mathbf{y} \in \mathbb{D}$ that $\mathbf{y} <_d \mathbf{x}^*$, \mathbf{x}^* is called as a Pareto optimal solution.

Definition 3 (Pareto optimal solution set): The set of all \mathbf{x}^* in \mathbb{D} is called Pareto optimal solution set.

Definition 4 (Pareto front): The set of all $\mathcal{G}(\mathbf{x}^*)$ is called Pareto front.

The decomposition-based multi-objective evolutionary algorithm (MOEA/D) is a well-known method for addressing multi-objective optimization problems, due to its capacity

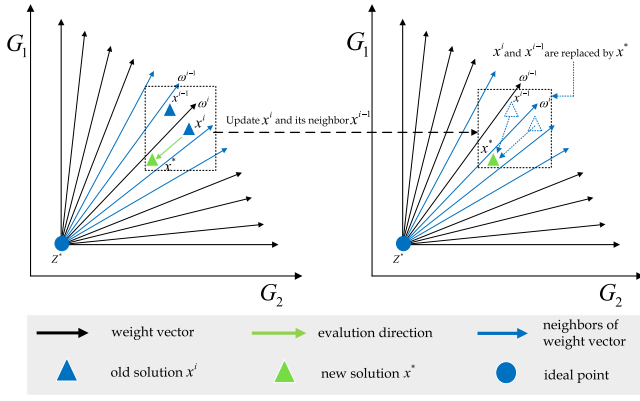


Fig. 2. Illustration of CMOEA/D-CDP.

to preserve population diversity and its lower computational complexity. Furthermore, it has been enhanced with CDP to tackle different CMOPs, yielding a new algorithm called CMOEA/D-CDP. CDP is used for the comparison between two solutions according to the feasibility and fitness. For any two solutions $\mathbf{x}, \mathbf{y} \in \mathbb{D}$, \mathbf{x} is better than \mathbf{y} if one of the conditions holds:

- If \mathbf{x} is feasible, and \mathbf{y} is infeasible.
- If \mathbf{x} and \mathbf{y} are feasible, and \mathbf{x} has smaller fitness.
- If \mathbf{x} and \mathbf{y} are infeasible, and \mathbf{x} has smaller constraint violation.

The primary concept behind CMOEA/D-CDP, as demonstrated in Fig. 2, is to decompose a CMOP into a series of scalar sub-problems, and solve them collaboratively. This is achieved by a set of N well-distributed weight vectors (\mathbf{V}) denoted as $\omega^1, \omega^2, \dots, \omega^N$, where N is the population size. In this context, each weight vector corresponds to a sub-problem. CMOEA/D-CDP utilizes a neighborhood information mechanism to enhance population evolution. It establishes neighborhood relationships among the sub-problems by identifying the T nearest weight vectors to each sub-problem using Euclidean distance. In the updating scheme, each new individual updates its neighbors using a decomposition-based method, e.g., Achievement Scalarizing Function (ASF), as given in Eq. (17). To preserve population diversity during evolution process, CMOEA/D-CDP maintains a manner by selecting the mating parents either from the corresponding neighbors of an individual or the entire population based on the probability δ . Additionally, the algorithm updates the neighbors of each individual at most n_r times.

$$ASF(\mathbf{x}^n | \omega^n) = \max \left(\frac{G_i(\mathbf{x}^n) - z_i^*}{\omega^n}, 1 \leq i \leq m \right), \quad (17)$$

where $\mathbf{z}^* = (z_1^*, \dots, z_m^*)$ is the ideal point used to shift the population to the first quadrant, z_i^* is the minimum value found so far for objective G_i . The main parameter settings of CMOEA/D-CDP, as suggested in [14], are given in Table I.

B. Representation of Encoding Scheme

The solution to Problem (12) is encoded as a mixed integer-float individual, as shown in Fig. 3. An individual is divided into three parts based on their data types,

TABLE I
MAIN PARAMETER SETTINGS IN CMOEA/D-CDP

Parameter	Description	Suggested value
\mathbf{V}	Set of all the weight vectors	Uniformly sampled from a hyperplane
\mathbb{B}_i	Set of the neighbors of ω^i	T closest weight vectors to ω^i
T	Neighborhood size	$0.1N$
δ	Probability of selecting manner for the mating parents	0.9
n_r	Times of updating neighbors	$0.01N$

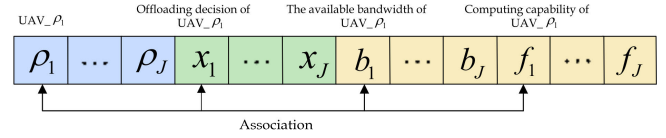


Fig. 3. Illustration of genetic encoding scheme.

i.e., integer or float. The first part (denoted as $unit_1$) is $\{\rho_1, \rho_2, \dots, \rho_J\}$, where ρ_j is an integer variable representing the index of a UAV that handles the j -th task. The second part (denoted as $unit_2$) is $\{x_1, x_2, \dots, x_J\}$, where x_j is a binary variable indicating whether the UAV offloads the task to the edge server or not. The third part (denoted as $unit_3$) is $\{b_1, b_2, \dots, b_J, f_1, f_2, \dots, f_J\}$, where b_j and f_j are float variable representing the assigned bandwidth and computing speed to the ρ_j -th UAV, respectively. To this end, the decision variable to Problem (12) is represented as $\mathbf{x} = (\rho_1, \dots, \rho_J, x_1, \dots, x_J, b_1, \dots, b_J, f_1, \dots, f_J)$. Throughout this study, χ_i implies the i -th element of \mathbf{x} .

Under this genetic encoding scheme, each region is guaranteed to assign at least one UAV. The constraints C_1 and C_2 are equally transformed to the following constraint:

$$C_7 : \sum_{j=1}^J IF_{i,j} = 1, \quad (18)$$

$$IF_{i,j} = \begin{cases} 1, & \text{if } \rho_i = \rho_j, \\ 0, & \text{otherwise,} \end{cases} \quad (19)$$

where $IF_{i,j}$ is equal to 1 when $\rho_i = \rho_j$ holds, and 0 otherwise. It is clear that the aforementioned constraint is met when each UAV is assigned to only one region, provided that it is selected to assist in executing the task associated with that region. In the context of evolutionary algorithm, the constraint C_6 is taken as the decision variable boundary constraint, meaning that each solution is ensured to be within the range of the decision variable boundary.

C. Proposed Algorithm

To improve the solution performance, we revise the conventional CMOEA/D-CDP by exploiting an improved genetic operator and repairing constraint-handling technique. The improved genetic operator is beneficial to seek promising solutions by adaptively disturbing the decision variables $\{\rho_i, x_i, b_i, f_i\}_{\forall i}$ according to their data types. The repairing constraint-handling technique is beneficial to improve the convergence performance of the proposed algorithm towards

feasibility. Our algorithm is executed generation by generation, as presented in **Algorithm 1**.

Algorithm 1 The Proposed Algorithm Framework

Input: The population size: N ,
 the neighborhood size: T ,
 the probability of selecting manner for the
 mating parents: δ ,
 the times of updating neighbors: n_r .

Output: The final population \mathbb{P}_t

- 1 Initialize a set of N evenly distributed weight vectors
 $\mathbb{V} \leftarrow \{\omega^1, \omega^2, \dots, \omega^N\}$;
- 2 Initialize \mathbb{B}_i by finding the T closest vectors to the
 weight vector ω^i , $i = 1, 2, \dots, N$;
- 3 Initialize a population \mathbb{P}_1 with size N ;
- 4 Initialize the ideal point \mathbf{z}^* based on \mathbb{P}_1 ;
- 5 $t \leftarrow 1$;
- 6 **while** $t \leq T_{max}$ **do**
- 7 **for** $i \leftarrow 1$ **to** N **do**
- 8 **if** $rand < \delta$ **then**
- 9 $\omega^{r1}, \omega^{r2}, \omega^{r3} \leftarrow$ three randomly selected
 weight vectors in \mathbb{B}_i ;
- 10 $\mathbf{x}^{r1}, \mathbf{x}^{r2}, \mathbf{x}^{r3} \leftarrow$ three mating individuals
 from \mathbb{P}_t , corresponding to $\omega^{r1}, \omega^{r2}, \omega^{r3}$;
- 11 **else**
- 12 $\mathbf{x}^{r1}, \mathbf{x}^{r2}, \mathbf{x}^{r3} \leftarrow$ three randomly selected
 mating individuals from \mathbb{P}_t ;
- 13 **end**
- 14 $\mathbf{x}^* \leftarrow \text{Algorithm 2}(\mathbf{x}^i, \mathbf{x}^{r1}, \mathbf{x}^{r2}, \mathbf{x}^{r3})$;
- 15 Update \mathbf{z}^* with \mathbf{x}^* ;
- 16 $v \leftarrow 0$;
- 17 **while** $v < n_r$ **do**
- 18 $\omega' \leftarrow$ a randomly selected weight vector in
 \mathbb{B}_i ;
- 19 Update the solution in \mathbb{P}_t corresponding to
 ω' with \mathbf{x}^* in terms of CDP;
- 20 $v \leftarrow v + 1$;
- 21 **end**
- 22 $(\mathbf{x}^*, \mathbb{P}_t) \leftarrow \text{Algorithm 3}(\mathbf{x}^*, \mathbb{B}_i, \mathbb{P}_t)$;
- 23 $\mathbb{P}_{t+1} \leftarrow \mathbb{P}_t$;
- 24 $t \leftarrow t + 1$;
- 25 **end**
- 26 **end**
- 27 **return** \mathbb{P}_t

At the beginning of the proposed algorithm (see Lines 1 - 5), it initializes a set of N weight vectors \mathbb{V} , where weight vectors are evenly distributed on the hyperplane, as in [14]. An initial population $\mathbb{P}_1 = \{\rho_i^j, x_i^j, b_i^j, f_i^j\}_{\forall i,j}$ is also produced, where ρ_i^j is the index of the UAV, x_i^j is the offloading decision, b_i^j is the assigned bandwidth, and f_i^j is the assigned computing speed, all of which pertain to the i -th task of the j -th individual. The parameters related to CMOEA/D-CDP are set at the same time. In this study, the maximum generation number is taken as the stopping criterion.

Subsequently, the proposed algorithm enters the loop. It maintains a population \mathbb{P}_t with size N at each generation. Therein, each individual \mathbf{x}^i participates in genetic operations with its mating parents that are chosen either from the neighbors of \mathbf{x}^i or from the whole population according to CMOEA/D-CDP. As a result, a new individual \mathbf{x}^* is produced (see Lines 8-13). The detail of the genetic operations is presented in **Algorithm 2**. At the same time, \mathbf{x}^* is used to update its neighbors at most n_r in terms of CDP (see Lines 16-21). When comparing two individuals via CDP, as described in Section IV-A, the fitness refers to the ASF value, as defined in Eq. (17). Subsequently, a repairing constraint-handling technique is applied to \mathbf{x}^* when this individual violates the constraints of Problem (12) (see Line 22). It is worth noting that the proposed algorithm aims to obtain Pareto optimal solutions in a greedy manner to fix infeasible solutions to feasible solutions by utilizing the constraint information in Problem (12). The detail of the repairing constraint-handling technique is presented in **Algorithm 3**.

When the generation number t reaches the maximum generation number T_{max} , the proposed algorithm stops and takes all the feasible non-dominated solutions in \mathbb{P}_t as the Pareto optimal solutions. Otherwise, it goes back to Line 6.

D. Proposed Improved Genetic Operator

In this subsection, we introduce the detail of the improved genetic operator. As described in Section IV-B, an individual consists of three parts where the first two parts, i.e., $unit_1$ and $unit_2$, are discrete, while the third part, i.e., $unit_3$, is continuous. An improved genetic operator is designed to deal with different data types of the decision variables in this study. The pseudo-code is presented in **Algorithm 2**.

The differential evolutionary operator (DE) called DE/current-to-rand/1 [29] and polynomial mutation operator [32] are adopted due to their effectiveness on solving CMOPs. When performing DE/current-to-rand/1, as given in Eq. (20), it requires three mating parents that are randomly chosen either from the neighbors or from the whole population (see Lines 8-13 in **Algorithm 1**) for each individual \mathbf{x}^i . As a result, a new individual $\mathbf{y} = \{\rho_i, x_i, b_i, f_i\}_{\forall i}$ is generated, in which \mathbf{y} has J new assigned UAVs with the corresponding offloading decisions, assigned bandwidth, and computing speed.

$$\mathbf{y} = \mathbf{x}^i + \mathcal{F}(\mathbf{x}^{r1} - \mathbf{x}^i) + \mathcal{F}(\mathbf{x}^{r2} - \mathbf{x}^{r3}), \quad (20)$$

where \mathcal{F} is the DE control parameter, its value is within $[0, 1]$. After the DE operation, if an element of \mathbf{y} is out of boundary constraint, it is randomly fixed back to its inside boundary.

Subsequently, a mutation operation is performed on \mathbf{y} . Since the structure of \mathbf{y} is composed of three parts, i.e., $unit_1$, $unit_2$, and $unit_3$, three different ways of mutating \mathbf{y} are executed accordingly (see Lines 5-24). Each element of \mathbf{y} executes the mutation operation with a probability CR . For the k -th element of \mathbf{y} , if it is in $unit_1$, either replacing a used UAV with a unused UAV or swapping two randomly chosen tasks of \mathbf{y} is performed with equal probability (see Lines 5-18). If it is in $unit_2$, the element is a binary variable. The value of χ_k is flipped from 1 to 0 and vice versa (see Lines 19-21).

Algorithm 2 The Improved Genetic Operation

Input: A solution: \mathbf{x}^i ,
the three mating individuals: \mathbf{x}^{r1} , \mathbf{x}^{r2} , \mathbf{x}^{r3}
Output: An offspring: \mathbf{x}^*

- 1 $\mathbf{y} \leftarrow$ a new individual generated by \mathbf{x}^i with \mathbf{x}^{r1} , \mathbf{x}^{r2} ,
and \mathbf{x}^{r3} according to Eq. (20);
- 2 $\mathbf{y} \leftarrow$ a solution obtained by fixing it back to the
boundary constraints;
- 3 **for** $k \leftarrow 1$ to $4J$ **do**
- 4 **if** $rand < CR$ **then**
- 5 **if** $\chi_k \in unit_1$ **then**
- 6 **if** $rand < 0.5$ **then**
- 7 $\rho'_{ik} \leftarrow$ a randomly chosen index of an
unused UAV based on \mathbf{y} ;
- 8 $x'_{ik}, b'_{ik}, f'_{ik} \leftarrow$ the values randomly
generated within their domains;
- 9 /*Replace $\rho_{ik}, x_{ik}, b_{ik}, f_{ik}$ with $\rho'_{ik}, x'_{ik},$
 b'_{ik}, f'_{ik} */
- 10 $\rho_{ik}, x_{ik}, b_{ik}, f_{ik} \leftarrow \rho'_{ik}, x'_{ik}, b'_{ik}, f'_{ik}$;
- 11 **else**
- 12 $p_1, p_2 \leftarrow$ two randomly selected
different indexes in $[1, J]$;
- 13 /*Swap $\rho_{ip_1}, x_{ip_1}, b_{ip_1}, f_{ip_1}$ and $\rho_{ip_2},$
 $x_{ip_2}, b_{ip_2}, f_{ip_2}$ */
- 14 $\rho^*_t, x^*_t, b^*_t, f^*_t \leftarrow \rho_{ip_1}, x_{ip_1}, b_{ip_1}, f_{ip_1}$;
- 15 $\rho_{ip_1}, x_{ip_1}, b_{ip_1}, f_{ip_1} \leftarrow$
 $\rho_{ip_2}, x_{ip_2}, b_{ip_2}, f_{ip_2}$;
- 16 $\rho_{ip_2}, x_{ip_2}, b_{ip_2}, f_{ip_2} \leftarrow \rho^*_t, x^*_t, b^*_t, f^*_t$;
- 17 **end**
- 18 **end**
- 19 **if** $\chi_k \in unit_2$ **then**
- 20 $\chi_k \leftarrow 1 - \chi_k$;
- 21 **end**
- 22 **if** $\chi_k \in unit_3$ **then**
- 23 $\chi_k^{new} \leftarrow$ a decision generated by the
polynomial mutation operator on χ_k ;
- 24 **end**
- 25 **end**
- 26 **end**
- 27 $\mathbf{y} \leftarrow$ a solution obtained by fixing it back to the
boundary constraints;
- 28 $\mathbf{x}^* \leftarrow \mathbf{y}$;
- 29 **return** \mathbf{x}^*

While if it is in $unit_3$, the polynomial mutation is used as it is a continuous variable. The polynomial mutation is given as follows:

$$\chi_k^{new} = \begin{cases} y_k + \sigma_k (U_k - L_k), & \text{if } rand < CR, \\ y_k, & \text{otherwise,} \end{cases} \quad (21)$$

$$\sigma_k = \begin{cases} (2 * rand)^{\frac{1}{\eta+1}} - 1, & \text{if } rand < 0.5, \\ 1 - (2 - 2 * rand)^{\frac{1}{\eta+1}}, & \text{otherwise,} \end{cases} \quad (22)$$

where $rand$ is a uniformly random number from $[0, 1]$, η is the distribution index, and CR is another DE operator parameter

Algorithm 3 Proposed Repairing Constraint-Handling Technique

Input: A solution: \mathbf{x}^* ,
the set of the neighbors of ω^i : \mathbb{B}_i ,
the current population: \mathbb{P}_t
Output: A repaired solution: \mathbf{x}^* ,
an updated population: \mathbb{P}_t

- 1 Calculate the constraint violations C_7, C_3 - C_5 of \mathbf{x}^* on
Problem (12);
- 2 **while** $C_7(\mathbf{x}^*) \neq 1$ **do**
- 3 **for** $i \leftarrow 1$ to J **do**
- 4 $\mathbb{I} \leftarrow$ the UAV indexes regarding the rest of
tasks having the same index with ρ_i ;
- 5 Replace the UAV indexes in \mathbb{I} with randomly
selected indexes associated with the unused
UAVs;
- 6 **end**
- 7 **end**
- 8 **if** $C_3(\mathbf{x}^*) > 0$ **then**
- 9 $B_r \leftarrow$ a randomly generated value in $(0, B]$;
- 10 $b_1^{new}, b_2^{new}, \dots, b_J^{new} \leftarrow$ a random division of B_r
into J parts;
- 11 $b_1, b_2, \dots, b_J \leftarrow b_1^{new}, b_2^{new}, \dots, b_J^{new}$;
- 12 **end**
- 13 **if** $C_4(\mathbf{x}^*) > 0$ **then**
- 14 $F_r \leftarrow$ a random value in $(0, F]$;
- 15 $f_1^{new}, f_2^{new}, \dots, f_J^{new} \leftarrow$ a random division of F_r
into J parts;
- 16 $f_1, f_2, \dots, f_J \leftarrow f_1^{new}, f_2^{new}, \dots, f_J^{new}$;
- 17 **end**
- 18 **while** $C_5(\mathbf{x}^*) > 0$ **do**
- 19 $x_j \leftarrow$ a random decision in \mathbf{x}^* with the value of 1;
- 20 $x_j \leftarrow 0$;
- 21 **end**
- 22 Update \mathbf{z}^* with \mathbf{x}^* ;
- 23 $v \leftarrow 0$;
- 24 **while** $v < n_r$ **do**
- 25 $\omega' \leftarrow$ a randomly selected weight vector in \mathbb{B}_i ;
- 26 Update the solution in \mathbb{P}_t corresponding to ω' with
 \mathbf{x}^* in terms of CDP;
- 27 $v \leftarrow v + 1$;
- 28 **end**
- 29 **return** $(\mathbf{x}^*, \mathbb{P}_t)$

whose value is in $[0, 1]$. U_k and L_k are the upper boundary and lower boundary of the k -th element of \mathbf{y} , respectively. After the mutation operation, if an element of \mathbf{y} is out of boundary constraint, it is randomly replaced with a value from its inside boundary.

E. Proposed Repairing Constraint-Handling Technique

This subsection introduces the proposed repairing constraint-handling technique that can convert infeasible solutions into feasible ones. The pseudo-code is presented in **Algorithm 3**.

TABLE II
COMMONLY UTILIZED SIMULATION PARAMETERS
AMONG THE THREE CMOPs

Parameter	Description	setting
I	Number of UAVs	10
c_1	Coefficient regarding UAV flying power	9.26×10^{-4}
c_2	Coefficient regarding UAV flying power	2250
α_j	Input data size of task j	[5.5, 10] Megabytes
W_j	Computational workload of task j	[1, 2] Giga CPU cycles
β_j	Data size of service program of task j	[5, 20] Megabytes
v_i	Flying speed of UAV i	[10, 50] m/s
$d_{i,j}$	Distance between UAV i and monitoring location j	[50, 100] m
P_i^H	Hovering power of UAV i	[50, 60] W
s_i	Data samples per second of UAV i	[3.75, 6.25] Megabytes/s
P_i^S	Sensing power of UAV i	[1, 125, 6.25] W
$k_{i,j}^{UL}$	Coefficient regarding uplink data rate of UAV i at monitoring location j	[1, 125, 2.5] Megabytes
$k_{i,j}^{DL}$	Coefficient regarding downlink data rate of UAV i at monitoring location j	[1, 125, 2.5] Megabytes
p_i^{RX}	Receive and transmit power of UAV i	[0.1, 0.2] W
p_i^{TX}	Transmit power of UAV i	[0.1, 0.2] W
κ_i	Hardware parameter of UAV i	$[1, 5] \times 10^{-27} \text{ W} \cdot \text{s}^3$
κ_{ES}	Hardware parameter of the edge server	$10^{-26} \text{ W} \cdot \text{s}^3$
r_{ES}	Data rate between the edge server and cloud server	1.125 Megabytes/s
ϑ	Reference value	10^{-3}

An offspring \mathbf{x}^* is produced after the genetic operation. When \mathbf{x}^* has positive degree of constraint violations, it implies that it violates some constraints related to Problem (12). For example, \mathbf{x}^* is infeasible when one UAV is sent to execute multiple tasks. A repairing constraint-handling technique is designed to fix it back to feasibility one by one constraint.

When \mathbf{x}^* violates the constraint C_7 , it is apparent that the constraint can be easily handled by substituting the duplicate UAVs with the unused UAVs (see Lines 2-7). When \mathbf{x}^* violates the constraint C_3 , the bandwidth B_r is firstly generated, and is divided into J parts that are used to replace the bandwidth of \mathbf{x}^* (see Lines 8-12). When it violates the constraint C_4 , the same mechanism is used to restrict the total usage of computing speed without exceeding F for the J tasks (see Lines 13-17). When it violates the constraint C_5 , the tasks for offloading are randomly changed to process the task locally one by one till the constraint is satisfied (see Lines 18-21).

V. EXPERIMENTAL STUDIES

A. Parameter Settings and Test Instances

We perform extensive simulation experiments to evaluate the overall performance of the proposed algorithm for UAV-enabled ITS. We pay attention to an application scenario of UAV-enabled ITS where UAVs run lightweight object detection models such as SSD-MobileNet and Yolo-tiny-v3 for traffic surveillance. The current models are of only several tens of megabytes that are suitable to be deployed on general embedded devices like UAVs. For example, proper models are optimized to have rather small model sizes while not sacrificing the detection accuracy. After that, they are efficiently deployed for vehicle detection [33] and crowd density detection [34] in the ITS environment. We further consider three case studies, leading to three CMOPs. The commonly utilized parameter settings for a UAV i , task j , and the edge server in the three cases are presented in Table II, while the distinctive parameter settings are presented in Table III. In Algorithm 1, three parameters for the genetic operators are set as follows: $F = 0.5$, $CR = 0.1$, and $\eta = 21$. During

TABLE III
SIMULATION PARAMETERS AMONG THE THREE CMOPs

Parameters	Description	Suggested value		
		CMOP1	CMOP2	CMOP3
J	Number of monitoring tasks	4	6	8
B	Maximum bandwidth B	10 MHz	20 MHz	25 MHz
F	Maximum computing speed F	5 GHz	10 GHz	15 GHz
S	Maximum data storage S	70 Megabytes	100 Megabytes	110 Megabytes
γ	Caching decisions of the service programs	[0, 1, 0, 1]	[0, 0, 1, 0, 1, 1]	[0, 0, 1, 0, 1, 0, 0, 1]

TABLE IV
COMPUTATIONAL COMPLEXITY OF THE SIX ALGORITHMS

Algorithm	Computational complexity
CMOPSO	$O(mN^2JT_{max})$
M2M-DW	$O(mNJT_{max})$
PPS	$O(mNn_rJT_{max})$
CMOEAD/CDP	$O(mNn_rJT_{max})$
ShiP	$O(mN^2JT_{max})$
Our Algorithm	$O(mNn_rJT_{max})$

the comparison experiments, we conduct 30 independent runs of six algorithms, each with a population size $N = 100$. The termination condition T_{max} is set to 800.

B. Performance Comparison

Meta-heuristic algorithms, such as particle swarm optimization algorithm [35] and evolutionary algorithm [36], are the promising methodologies for solving the proposed CMOP. In this study, we compare the proposed algorithm with five algorithms particularly designed for CMOPs, i.e., M2M-DW [37], PPS [32], CMOEAD/CDP [14], ShiP [38], and CMOPSO [39]. We compare the performance of all algorithms on the proposed CMOP with the three different parameter settings, as presented in Table III. The computational complexities of the six algorithms are shown in Table IV, where m is the number of objectives, N is the size of the population, J is the number of tasks, n_r is the times of updating neighbors, and T_{max} is the maximum generation number. As illustrated in Table IV, the baseline algorithms such as CMOPSO and ShiP exhibit a computational complexity of $O(mN^2JT_{max})$. This complexity arises from their utilization of non-dominated sorting and crowding distance to select individuals during the evolutionary process. In contrast, the proposed algorithm leverages the framework of CMOEAD/CDP, wherein each offspring is updated a maximum of n_r times, leading to a complexity of $O(mNn_rJT_{max})$. Given that n_r is typically smaller than N , the computational complexities of PPS, CMOEAD/CDP, and the proposed algorithm are comparatively lower.

To evaluate the performance of the algorithms, we adopt two commonly utilized performance metrics, i.e., inverted generational distance (IGD) [29] and hypervolume (HV) [32], and focus on their mean values and standard deviation (STD). To calculate the aforementioned performance metrics, we consider 100 feasible non-dominated solutions from the final population.

IGD can measure the distance between the approximate Pareto front generated by an algorithm and the true Pareto front. It is defined as the average shortest distance from each solution in the true Pareto front to the approximate Pareto front. The smaller IGD value implies that the algorithm obtains a better approximate Pareto front. The calculation of IGD is shown as follows:

$$IGD_t(\mathbb{PF}^*, \mathbb{PF}_t) = \frac{\sum_{\mathbf{x} \in \mathbb{PF}^*} d(\mathbf{x}, \mathbb{PF}_t)}{\|\mathbb{PF}^*\|}, \quad (23)$$

where \mathbb{PF}^* represents the true Pareto front, \mathbb{PF}_t represents the Pareto front obtained by an algorithm at generation t , and $d(\mathbf{x}, \mathbb{PF}_t)$ is the Euclidean distance between individual \mathbf{x} in \mathbb{PF}^* and the closest individual in \mathbb{PF}_t to \mathbf{x} . Given that the true Pareto front of the CMOPs is unknown, we use the Pareto front generated by the proposed algorithm as a proxy. The Pareto front is subsequently adjusted slightly to the bottom-left to serve as the reference Pareto front for all algorithms.

HV can measure the convergence and diversity of the population generated by an algorithm. It is defined as the volume covered by a reference point and the obtained Pareto front. The larger HV value implies that the algorithm obtains a better approximate Pareto front. The calculation of HV is shown as follows:

$$HV_t = \text{volume} \left(\bigcup_{i=1}^{|\mathbb{PF}_t|} v_i \right), \quad (24)$$

where v_i represents a hypercube associated each individual \mathbf{x}_i in \mathbb{PF}_t and a reference point. The reference point for the three cases is set to [15000, 25000], [20000, 30000], and [33000, 65000], respectively.

Table V shows comparison results of the six compared algorithms on the three CMOPs. The performance of the algorithms is measured by the mean values and STD of IGD and HV, respectively. We find that compared to ShiP, the proposed algorithm yields an approximate 69% decrease in the mean value of IGD and approximate 4% increase in the mean value of HV. The proposed algorithm achieves the best results among the six algorithms, as highlighted in Table V. The proposed algorithm obtains the highest mean HV and the lowest mean IGD on each CMOP. Compared with the baseline algorithms, the proposed algorithm can seek a set of better distributed and better converged non-dominated solutions. As a result, the control center can make a wider choice among the obtained solutions regarding the preferences on the two objectives G_1 and G_2 .

Figs. 4-6 illustrate the non-dominated solutions of the six algorithms obtained at the median run in terms of HV. The vertical axis represents the first objective G_1 while the horizontal axis represents the second objective G_2 . From the figures, the proposed algorithm is superior to all baseline algorithms. As for CMOP1, we can observe that the proposed algorithm finds more approximate Pareto front than the compared algorithms with respect to less completion time difference and energy consumption. Although ShiP achieves a part of similar approximate Pareto front with the proposed algorithm, the proposed algorithm finds a wider range of non-dominated

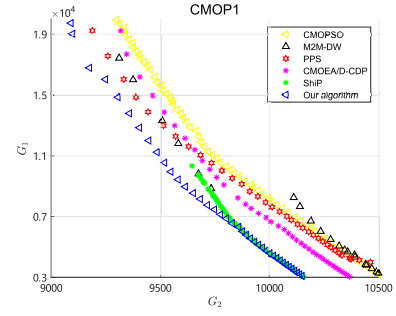


Fig. 4. Obtained non-dominated solutions of CMOP1.

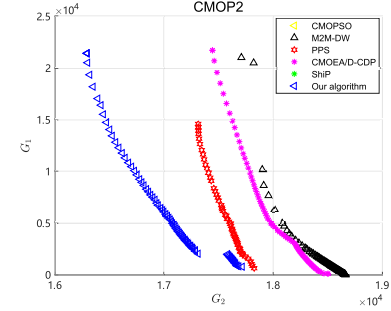


Fig. 5. Obtained non-dominated solutions of CMOP2.

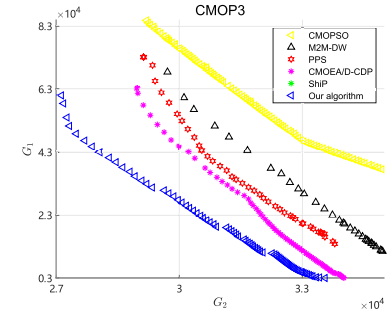


Fig. 6. Obtained non-dominated solutions of CMOP3.

solutions. As for CMOP2 and CMOP3, the proposed algorithm outperforms the baseline algorithms, and consistently finds better non-dominated solutions for Problem (12). The above results demonstrate the effectiveness and superiority of the proposed algorithm on all considered test instances.

Fig. 7 presents the convergence graphs of the six algorithms on the three CMOPs at the median run in terms of HV. The horizontal axis represents the generation number t while the vertical axis represents the HV. Although a small number of monitoring tasks are required to handle in CMOP1, we find that the proposed algorithm works slightly better than CMOPSO, M2M-DW, CMOEAD-CDP, and ShiP. The advantage of the proposed algorithm is further shown by the results on CMOP2 and CMOP3, where more monitoring tasks are required, and the constraints in Problem (12) are more difficult to satisfy compared with CMOP1. To enhance solution performance, the proposed algorithm designs an improved genetic operator for enhancing search capabilities, along with a repairing constraint-handling technique that utilizes constraint information to transform infeasible solutions

TABLE V
COMPARISON RESULTS OF SIX COMPARED ALGORITHMS

Algorithm	CMOP1		CMOP2		CMOP3	
	IGD	HV	IGD	HV	IGD	HV
CMOPSO	5.21e+02(2.08e+02)	1.28e+08(5.43e+06)	6.70e+03(5.21e+03)	4.19e+07(3.28e+07)	1.41e+04(1.02e+04)	8.35e+07(6.27e+07)
DW	6.35e+02(2.06e+02)	1.30e+08(5.20e+06)	2.22e+03(9.60e+02)	5.39e+07(2.08e+07)	6.70e+03(6.82e+03)	7.86e+07(5.56e+07)
PPS	6.43e+02(3.58e+02)	1.28e+08(8.68e+06)	2.69e+03(2.87e+03)	6.73e+07(1.56e+07)	4.59e+03(2.46e+03)	1.08e+08(4.73e+07)
CMOEA/D-CDP	5.55e+02(2.49e+02)	1.30e+08(6.41e+06)	1.41e+03(7.47e+02)	7.35e+07(1.72e+07)	3.10e+03(1.66e+03)	1.11e+08(5.11e+07)
ShiP	1.03e+03(7.75e+02)	1.29e+08(1.01e+07)	5.21e+03(0.00e+00)	5.71e+07(0.00e+00)	0.00e+00(0.00e+00)	0.00e+00(0.00e+00)
Our Algorithm	3.18e+02(6.34e+01)	1.36e+08(1.12e+06)	4.24e+02(9.98e+01)	9.97e+07(2.41e+06)	1.08e+03(2.14e+02)	2.18e+08(8.84e+06)

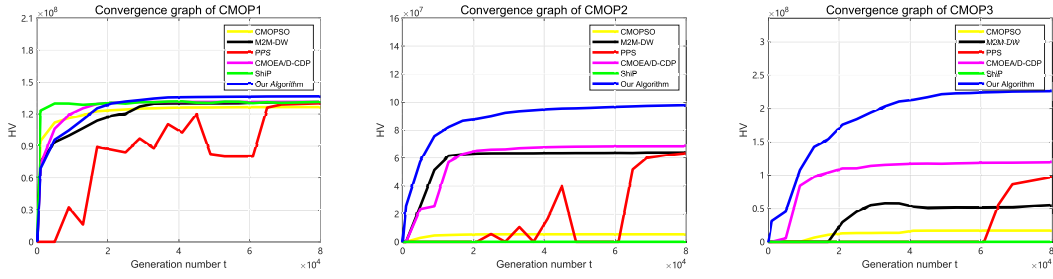


Fig. 7. Convergence graphs of the six algorithms on the three CMOPs at the median run in terms of HV.

TABLE VI

COMPARISON OF OUR ALGORITHM WITH AND WITHOUT THE IMPROVED GENETIC OPERATION

	IGD	HV
Yes	4.24e+02(9.98e+01)	9.97e+07(2.41e+06)
No	9.52e+02(3.97e+02)	8.49e+07(1.12e+07)

TABLE VII

COMPARISON OF OUR ALGORITHM WITH AND WITHOUT PROPOSED REPAIRING CONSTRAINT-HANDLING TECHNIQUE

	IGD	HV
Yes	4.24e+02(9.98e+01)	9.97e+07(2.41e+06)
No	5.34e+02(1.60e+02)	9.69e+07(3.79e+06)

into feasible ones, thereby accelerating the convergence of the proposed algorithm. Unlike the baseline algorithms that underutilize constraint information, the proposed algorithm achieves superior results. For example, in CMOP2, our investigations demonstrate that the proposed algorithm attains a notably enhanced solution performance, as evidenced by a reduced mean value of IGD and an augmented mean value of HV. Specifically, the mean value of HV exhibits a remarkable increase of approximately 26%, attributing this notable improvement directly to the utilization of the improved genetic operator and the repairing constraint-handling technique.

C. Investigation of the Improved Genetic Operator

We study the impact of the improved genetic operator on the performance improvements of the proposed algorithm.

In this experiment, we consider CMOP2 to investigate the effectiveness of the improved genetic operator. The proposed algorithm can adopt the improved genetic operator (denoted as “Yes”) or the original genetic operator (denoted as “No”). It is noted that the original genetic operator uses DE/current-to-rand/1 and the polynomial mutation. The comparative results are presented in Table VI. The improved genetic operator is

based on our encoding scheme in Section IV-B, in which the design enables the proposed algorithm to effectively explore promising regions and seek superior solutions. As a result, the mean value of IGD decreases by approximately 55%, and the mean value of HV increases by approximately 15%. This demonstrates that the improved genetic operator can enhance the ability of the proposed algorithm for seeking Pareto optimal solutions.

D. Investigation of the Repairing Constraint-Handling Technique

In this experiment, we also consider CMOP2 to investigate the effectiveness of the proposed repairing constraint handling technique. This design enables the proposed algorithm to effectively convert infeasible solutions into feasible ones. From Table VII, we observe that the mean value of IGD decreases by approximately 20% while the mean value of HV increases by approximately 2%. This demonstrates that the repairing constraint-handling technique can enhance the ability of the proposed algorithm in handling constraints.

VI. CONCLUSION

We investigated the joint minimization of energy consumption and completion time difference for UAV-enabled ITS by using a constrained multi-objective optimization approach. UAVs were employed to collect and process the surveillance data with the help of the edge server. In the formulated CMOP, we aimed at simultaneously achieving the energy-efficient data collection and processing, and reducing the total completion time difference among the employed UAVs. We jointly optimized the UAV association, task offloading, and resource allocation. To tackle the CMOP, we adopted a constrained decomposition-based multi-objective evolutionary algorithm. To obtain high-quality solutions, we designed the improved genetic operator to seek more promising solutions and repairing constraint-handling technique to accelerate the proposed algorithm convergence towards feasibility. Extensive numerical results demonstrated that compared with the baseline algorithms, the proposed algorithm achieved the better non-dominated solutions on different test instances. In the future, we will consider the real implementation of UAV-enabled ITS applications such as road traffic monitoring, and discuss the joint optimization of trajectory planning and computation offloading.

REFERENCES

- [1] T. Ma et al., "UAV-LEO integrated backbone: A ubiquitous data collection approach for B5G Internet of Remote Things networks," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 11, pp. 3491–3505, Nov. 2021.
- [2] X.-H. Lin et al., "Joint optimization of resource allocation and flight trajectory for UAV-IoT underwater detecting systems," *IEEE Trans. Veh. Technol.*, vol. 72, no. 12, pp. 16482–16498, Dec. 2023.
- [3] J. Xu, K. Ota, and M. Dong, "Aerial edge computing: Flying attitude-aware collaboration for multi-UAV," *IEEE Trans. Mobile Comput.*, vol. 22, no. 10, pp. 5706–5718, Oct. 2023.
- [4] W. Lu et al., "Secure NOMA-based UAV-MEC network towards a flying eavesdropper," *IEEE Trans. Commun.*, vol. 70, no. 5, pp. 3364–3376, May 2022.
- [5] Z. Chang, H. Deng, L. You, G. Min, S. Garg, and G. Kaddoum, "Trajectory design and resource allocation for multi-UAV networks: Deep reinforcement learning approaches," *IEEE Trans. Netw. Sci. Eng.*, vol. 10, no. 5, pp. 2940–2951, Sep./Oct. 2023.
- [6] M. Dai, Y. Wu, L. Qian, Z. Su, B. Lin, and N. Chen, "UAV-assisted multi-access computation offloading via hybrid NOMA and FDMA in marine networks," *IEEE Trans. Netw. Sci. Eng.*, vol. 10, no. 1, pp. 113–127, Jan. 2023.
- [7] F. Outay, H. A. Mengash, and M. Adnan, "Applications of unmanned aerial vehicle (UAV) in road safety, traffic and highway infrastructure management: Recent advances and challenges," *Transp. Res. A, Policy Pract.*, vol. 141, pp. 116–129, Nov. 2020.
- [8] H. Menouar, I. Guvenc, K. Akkaya, A. S. Uluagac, A. Kadri, and A. Tuncer, "UAV-enabled intelligent transportation systems for the smart city: Applications and challenges," *IEEE Commun. Mag.*, vol. 55, no. 3, pp. 22–28, Mar. 2017.
- [9] X. Hu, L. Wang, K.-K. Wong, M. Tao, Y. Zhang, and Z. Zheng, "Edge and central cloud computing: A perfect pairing for high energy efficiency and low-latency," *IEEE Trans. Wireless Commun.*, vol. 19, no. 2, pp. 1070–1083, Feb. 2020.
- [10] Y. Xiang, X. Yang, H. Huang, and J. Wang, "Balancing constraints and objectives by considering problem types in constrained multiobjective optimization," *IEEE Trans. Cybern.*, vol. 53, no. 1, pp. 88–101, Jan. 2023.
- [11] Z. Zhang, Z. Hao, and H. Huang, "Hybrid swarm-based optimization algorithm of GA & VNS for nurse scheduling problem," in *Information Computing and Applications*, B. Liu and C. Chai, Eds. Berlin, Germany: Springer, 2011, pp. 375–382.
- [12] K. Qiao et al., "Dynamic auxiliary task-based evolutionary multitasking for constrained multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 27, no. 3, pp. 642–656, Jun. 2023.
- [13] M. Zuo, D. Gong, Y. Wang, X. Ye, B. Zeng, and F. Meng, "Process knowledge-guided autonomous evolutionary optimization for constrained multiobjective problems," *IEEE Trans. Evol. Comput.*, vol. 28, no. 1, pp. 193–207, Feb. 2024.
- [14] M. A. Jan and R. A. Khanum, "A study of two penalty-parameterless constraint handling techniques in the framework of MOEA/D," *Appl. Soft Comput.*, vol. 13, no. 1, pp. 128–148, Jan. 2013.
- [15] X. Liu, B. Lai, B. Lin, and V. C. M. Leung, "Joint communication and trajectory optimization for multi-UAV enabled mobile Internet of Vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 15354–15366, Sep. 2022.
- [16] H. Abualola, H. Otok, H. Barada, M. Al-Qutayri, and Y. Al-Hammadi, "Matching game theoretical model for stable relay selection in a UAV-assisted Internet of Vehicles," *Veh. Commun.*, vol. 27, Jan. 2021, Art. no. 100290.
- [17] J. Hu, C. Chen, L. Cai, M. R. Khosravi, Q. Pei, and S. Wan, "UAV-assisted vehicular edge computing for the 6G Internet of Vehicles: Architecture, intelligence, and challenges," *IEEE Commun. Stand. Mag.*, vol. 5, no. 2, pp. 12–18, Jun. 2021.
- [18] R. Zhang, R. Lu, X. Cheng, N. Wang, and L. Yang, "A UAV-enabled data dissemination protocol with proactive caching and file sharing in V2X networks," *IEEE Trans. Commun.*, vol. 69, no. 6, pp. 3930–3942, Jun. 2021.
- [19] R. Liu, A. Liu, Z. Qu, and N. N. Xiong, "An UAV-enabled intelligent connected transportation system with 6G communications for Internet of Vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 2, pp. 2045–2059, Feb. 2023.
- [20] L. Zhao, K. Yang, Z. Tan, X. Li, S. Sharma, and Z. Liu, "A novel cost optimization strategy for SDN-enabled UAV-assisted vehicular computation offloading," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 6, pp. 3664–3674, Jun. 2021.
- [21] Y. Liu et al., "Joint communication and computation resource scheduling of a UAV-assisted mobile edge computing system for platooning vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 8435–8450, Jul. 2021.
- [22] M. Dai, Z. Su, Q. Xu, and N. Zhang, "Vehicle assisted computing offloading for unmanned aerial vehicles in smart city," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 3, pp. 1932–1944, Mar. 2021.
- [23] Y. Wang et al., "Task offloading for post-disaster rescue in unmanned aerial vehicles networks," *IEEE/ACM Trans. Netw.*, vol. 30, no. 4, pp. 1525–1539, Aug. 2022.
- [24] A. Alioua, H.-E. Djeghri, M. E. T. Cherif, S.-M. Senouci, and H. Sedjelmaci, "UAVs for traffic monitoring: A sequential game-based computation offloading/sharing approach," *Comput. Netw.*, vol. 177, Aug. 2020, Art. no. 107273.
- [25] L. Liu, A. Wang, G. Sun, and J. Li, "Multiobjective optimization for improving throughput and energy efficiency in UAV-enabled IoT," *IEEE Internet Things J.*, vol. 9, no. 20, pp. 20763–20777, Oct. 2022.
- [26] G. Sun, J. Li, Y. Liu, S. Liang, and H. Kang, "Time and energy minimization communications based on collaborative beamforming for UAV networks: A multi-objective optimization method," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 11, pp. 3555–3572, Nov. 2021.
- [27] S. M. Hashir, A. Mehrabi, M. R. Mili, M. J. Emadi, D. W. K. Ng, and I. Krikidis, "Performance trade-off in UAV-aided wireless-powered communication networks via multi-objective optimization," *IEEE Trans. Veh. Technol.*, vol. 70, no. 12, pp. 13430–13435, Dec. 2021.
- [28] Y. Yu, J. Tang, J. Huang, X. Zhang, D. K. C. So, and K.-K. Wong, "Multi-objective optimization for UAV-assisted wireless powered IoT networks based on extended DDPG algorithm," *IEEE Trans. Commun.*, vol. 69, no. 9, pp. 6361–6374, Sep. 2021.
- [29] C. Peng, X. Huang, Y. Wu, and J. Kang, "Constrained multi-objective optimization for UAV-enabled mobile edge computing: Offloading optimization and path planning," *IEEE Wireless Commun. Lett.*, vol. 11, no. 4, pp. 861–865, Apr. 2022.
- [30] Y. Zeng and R. Zhang, "Energy-efficient UAV communication with trajectory optimization," *IEEE Trans. Wireless Commun.*, vol. 16, no. 6, pp. 3747–3760, Jun. 2017.
- [31] Q. Tang, Z. Yu, C. Jin, J. Wang, Z. Liao, and Y. Luo, "Completed tasks number maximization in UAV-assisted mobile relay communication system," *Comput. Commun.*, vol. 187, pp. 20–34, Apr. 2022.
- [32] Z. Fan et al., "Push and pull search for solving constrained multi-objective optimization problems," *Swarm Evol. Comput.*, vol. 44, pp. 665–679, Feb. 2019.

- [33] W. Y. B. Lim et al., "Towards federated learning in UAV-enabled Internet of Vehicles: A multi-dimensional contract-matching approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 8, pp. 5140–5154, Aug. 2021.
- [34] S. Wang, Z. Pu, Q. Li, and Y. Wang, "Estimating crowd density with edge intelligence based on lightweight convolutional neural networks," *Expert Syst. Appl.*, vol. 206, Nov. 2022, Art. no. 117823.
- [35] W.-N. Chen, J. Zhang, H. Chung, W.-L. Zhong, W.-G. Wu, and Y. H. Shi, "A novel set-based particle swarm optimization method for discrete optimization problems," *IEEE Trans. Evol. Comput.*, vol. 14, no. 2, pp. 278–300, Apr. 2010.
- [36] H. Huang, Y. Xu, Y. Xiang, and Z. Hao, "Correlation-based dynamic allocation scheme of fitness evaluations for constrained evolutionary optimization," *IEEE Trans. Evol. Comput.*, early access, 2024, doi: 10.1109/TEVC.2023.3302897.
- [37] C. Peng, H.-L. Liu, and E. D. Goodman, "A cooperative evolutionary framework based on an improved version of directed weight vectors for constrained multiobjective optimization with deceptive constraints," *IEEE Trans. Cybern.*, vol. 51, no. 11, pp. 5546–5558, Nov. 2021.
- [38] Z. Ma and Y. Wang, "Shift-based penalty for evolutionary constrained multiobjective optimization and its application," *IEEE Trans. Cybern.*, vol. 53, no. 1, pp. 18–30, Jan. 2023.
- [39] X. Zhang, X. Zheng, R. Cheng, J. Qiu, and Y. Jin, "A competitive mechanism based multi-objective particle swarm optimizer with fast convergence," *Inf. Sci.*, vol. 427, pp. 63–76, Feb. 2018.



Chaoda Peng (Member, IEEE) received the Ph.D. degree from the School of Automation, Guangdong University of Technology, Guangzhou, China, in 2019. He was a Visiting Ph.D. Student with the Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI, USA. He is currently with the College of Mathematics and Informatics, South China Agricultural University, Guangzhou. His current research interests include constrained evolutionary multi-objective optimization, UAV path planning, and mobile edge computing.



Zexiong Wu is currently pursuing the B.S. degree in mathematics and applied mathematics with South China Agricultural University. His research interests include mobile edge computing, evolutionary computation, and multi-objective optimization.



Xumin Huang received the Ph.D. degree from Guangdong University of Technology, China, in 2019. He is currently an Associate Professor with the School of Automation, Guangdong University of Technology. He is also a Macau Young Scholar with the State Key Laboratory of Internet of Things for Smart City, University of Macau, Macau, China. His research interests include resource and service optimizations for connected vehicles, the Internet of Things, blockchain, and edge intelligence.



Yuan Wu (Senior Member, IEEE) received the Ph.D. degree in electronic and computer engineering from The Hong Kong University of Science and Technology in 2010. He is currently an Associate Professor with the State Key Laboratory of Internet of Things for Smart City and the Department of Computer and Information Science, University of Macau, Macau, China. His research interests include resource management for wireless networks, edge computing and edge intelligence, and integrated sensing and communication. He received the Best Paper Award from IEEE ICC'2016, IEEE TCGCC'2017, IWCMC'2021, and WCNC'2023. He served as the Track/Symposium Co-Chair for IEEE VTC'2017-Fall, VTC'2021-Spring, VTC'2022-Spring, and ICC'2023. He is on the Editorial Board of IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING, and IEEE INTERNET OF THINGS JOURNAL.



Jiawen Kang (Senior Member, IEEE) received the Ph.D. degree from Guangdong University of Technology, China, in 2018. He was a Post-Doctoral Researcher with Nanyang Technological University, Singapore, from 2018 to 2021. He is currently a Professor with Guangdong University of Technology. His main research interests include blockchain, security, and privacy protection in wireless communications and networking.



Qiong Huang received the Ph.D. degree from the City University of Hong Kong in 2010. He is currently a Professor with the College of Mathematics and Informatics, South China Agricultural University, Guangzhou, China. He has published more than 170 research papers in international conferences and journals. His research interests include information security and intelligent agriculture. He served as a program committee member for many international conferences.



Shengli Xie (Fellow, IEEE) received the B.S. degree in mathematics from Jilin University, Changchun, China, in 1983, the M.S. degree in mathematics from Central China Normal University, Wuhan, China, in 1995, and the Ph.D. degree in control theory and applications from the South China University of Technology, Guangzhou, China, in 1997. He is currently a Full Professor and the Head of the Institute of Intelligent Information Processing, Guangdong University of Technology, Guangzhou. He has coauthored two books and more than 150 research papers in refereed journals and conference proceedings. His research interests include blind signal processing, machine learning, and the Internet of Things. He is a Foreign Full Member (an Academician) of Russian Academy of Engineering. He received the Second Prize of the National Natural Science Award of China in 2009. He was awarded the Highly Cited Researcher in 2020. He is an Associate Editor of IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS.

IEEE TRANSACTIONS ON RELIABILITY

published by the
IEEE RELIABILITY SOCIETY

DECEMBER 2024

VOLUME 73

NUMBER 4

IERQAD

(ISSN 1558-1721)

TReL-CPS: SPECIAL SECTION ON AI AND ML FOR RESILIENCE ASSESSMENT AND ENHANCEMENT OF
CYBER-PHYSICAL SYSTEMS

Editorial: AI and ML for Resilience Assessment and Enhancement of Cyber-Physical Systems	W. Shieh	1709
Decentralized Graphical-Representation-Enabled Multi-Agent Deep Reinforcement Learning for Robust Control of Cyber-Physical Systems	D. Cao, J. Hu, Y. Liu, and W. Hu	1710
A Condition-Based Maintenance Policy Considering Batch Sizes for Warm Standby Systems With Priority to Repair	F. Qi, Y. Wang, and F. Li	1721
L-TLA: A Lightweight Driver Distraction Detection Method Based on Three-Level Attention Mechanisms	Z. Guo, Q. Liu, L. Zhang, Z. Li, and G. Li	1731
Deep Learning-Based Segment Trend Removal Approach for Prognostics and Health Management Signals of Rail Vehicles	S. Zhou, Y. Liu, P. Zhao, and B. Chen	1743
Multiscale BLS-Based Lightweight Prediction Model for Remaining Useful Life of Aero-Engine	T. Xu, G. Han, H. Zhu, C. Lin, and J. Peng	1757
A Lightweight Bearing Compound Fault Diagnosis Method With Gram Angle Field and Ghost-ResNet Model	Y. Gu, R. Chen, P. Huang, J. Chen, and G. Qiu	1768
Concurrent Classifier Error Detection (CCED) in Large Scale Machine Learning Systems	P. Reviriego, Z. Wang, Á. Alonso, Z. Gao, F. Niknia, S. Liu, and F. Lombardi	1782
Active Labeling Aided Semi-Supervised Safety Assessment With Task-Related Unknown Scenarios	C. Liu, X. He, M. Li, Y. Zhang, and Z. Ding	1792
A Novel Machine Learning Based Fault Diagnosis Method for All Gas-Path Components of Heavy Duty Gas Turbines With the Aid of Thermodynamic Model	J. Li and Y. Ying	1805

REGULAR PAPERS

Software Testing, Verification, and Quality Assurance

Software Fault Localization Based on Network Spectrum and Graph Neural Network	X. Gou, A. Zhang, C. Wang, Y. Liu, X. Zhao, and S. Yang	1819
Combining Metamorphic Testing and Machine Learning to Enhance OpenStreetMap	M. Méndez, A. Becerra-Terón, J. M. Almendros-Jiménez, M. G. Merayo, and M. Núñez	1834
Modeling and Verification Methods for Spatio-Temporal Consistency of CPS in Uncertain Environments	S. Pan, C. Wang, W. Xie, J. Lu, Q. Huang, and Z. Zuo	1849
Code Multiview Hypergraph Representation Learning for Software Defect Prediction	S. Qiu, M. Huang, Y. Liang, C. Peng, and Y. Yuan	1863
Adjusted Trust Score: A Novel Approach for Estimating the Trustworthiness of Software Defect Prediction Models	X. Wan, Z. Zheng, F. Qin, X. Lu, and K. Qiu	1877
UniAda: Universal Adaptive Multiobjective Adversarial Attack for End-to-End Autonomous Driving Systems	J. Zhang, J. W. Keung, Y. Xiao, Y. Liao, Y. Li, and X. Ma	1892

Network Reliability and Security

Component Diagnosis Strategy of Star Graphs Interconnection Networks	Z. Wan, L. Lin, Y. Huang, and S.-Y. Hsieh	1907
--	---	------

(Contents Continued on Page 1708)



DPkCR: Distributed Proactive k -Connectivity Recovery Algorithm for UAV-Based MANETs	
..... <i>M. Tosun, U. C. Cabuk, E. Haytaoglu, O. Dagdeviren, and Y. Ozturk</i>	1918
Fault-Tolerant Communication in HSDC: Ensuring Reliable Data Transmission in Smart Cities	
..... <i>H. Dong, M. Lv, and W. Fan</i>	1933
Subsystem Reliability Analysis of Data Center Network BCube	<i>Y. Wang, W. Fan, J. Fan, J. Zhou, and B. Cheng</i> 1946
<i>Degradation and Maintenance</i>	
Adaptive Two-Stage Model for Bearing Remaining Useful Life Prediction Using Gaussian Process Regression With Matched Kernels	<i>X. Zheng, W. Fan, C. Chen, and Z. Peng</i> 1958
Joint Multimission Selective Maintenance and Inventory Optimization for Multicomponent Systems Considering Stochastic Dependency	<i>X. Kong, J. Yang, W. Chen, and J. Pan</i> 1967
Cooperative Interaction Observer-Based Security Control for T-S Fuzzy Cyber-Physical Systems Against Sensor and Actuator Attacks	<i>X. Huang, C. Chang, J. Li, S. Xiao, and Q. Su</i> 1982

CORRECTIONS	
Corrections to “Guest Editorial: Crisis Management—From Nuclear Accidents to Outbreaks of COVID-19 and Infectious Diseases”	<i>W. Kuo and J. He</i> 1993
Erratum to “Effective Prediction of Bug-Fixing Priority via Weighted Graph Convolutional Networks”	<i>S. Fang, Y.-shuai Tan, T. Zhang, Z. Xu, and H. Liu</i> 1994

Code Multiview Hypergraph Representation Learning for Software Defect Prediction

Shaojian Qiu¹, *Member, IEEE*, Mengyang Huang¹, Yun Liang¹, Chaoda Peng¹, *Member, IEEE*,
and Yuan Yuan¹

Abstract—Software defect prediction technology aids the reliability assurance team in identifying defect-prone code and assists the team in reasonably allocating limited testing resources. Recently, researchers assumed that the topological associations among code fragments could be harnessed to construct defect prediction models. Nevertheless, existing graph-based methods only concentrate on features of single-view association, which fail to fully capture the rich information hidden in the code. In addition, software defects may involve multiple code fragments simultaneously, but traditional binary graph structures are insufficient for representing these multivariate associations. To address these two challenges, this article proposes a multiview hypergraph representation learning approach (MVHR-DP) to amplify the potency of code features in defect prediction. MVHR-DP initiates by creating hypergraph structures for each code view, which are then amalgamated into a comprehensive fusion hypergraph. Following this, a hypergraph neural network is established to extract code features from multiple views and intricate associations, thereby enhancing the comprehensiveness of representation in the modeling data. Empirical study shows that the prediction model utilizing features generated by MVHR-DP exhibits superior area under the curve (AUC), F-measure, and matthews correlation coefficient (MCC) results compared to baseline approaches across within-project, cross-version, and cross-project prediction tasks.

Index Terms—Code multiview fusion, code representation learning, hypergraph construction, software defect prediction, software reliability.

I. INTRODUCTION

THE difficulty of reliability assurance is increasing with the continuous growth of software scale and complexity [1]. In response, software defect prediction emerges as a technology to assess the predisposition of new files and modules toward defects. This facilitates judicious resource allocation by the

reliability assurance team and aids in determining if the software aligns with delivery standards [2].

In the defect prediction task, finding out the defect-related features of the code is the key to building a high-quality prediction model [3]. When delving into code representation, diverse features can be derived in accordance with distinct code metrics. For instance, Halstead features are rooted in operands and operators, McCabe features stem from code dependencies, chidamber-kemerer (CK) features are rooted in object-oriented concepts, and structural-semantics features are based on abstract syntax trees (AST) [4]. These code feature representations predominantly encapsulate the intricacies and structural insights in the code and have gained widespread traction in constructing defect prediction models [5], [6].

Recently, researchers have focused on intricate linkages among code fragments. They attempt to extract information from the correlations between codes (e.g., dependency and similarity associations) to enhance the comprehensiveness of modeling data [7], [8], [9]. Qu et al. [10] applied a node2vec embedding technique to encode program dependency structures into a low-dimensional vector space. Then, these vectors enriched with code context information were harnessed for constructing a defect prediction model. The literature [11] has explored the analysis of software defect distributions through class-dependent networks, using k-core decomposition to further augment the performance of predictive models. Moreover, Zeng et al. [12] combined node2vec features with traditional handcrafted features to construct data for a graph convolutional network to obtain a more profound representation of code associations.

In practice, the preceding graph-association-based methodologies encounter two challenges. First, intricate multiview associations exist within the realm of code fragments. However, the abovementioned methods predominantly concentrate on capturing code dependencies, neglecting the valuable contextual insights offered by other code views. Second, software defects may be caused by the collaborative work of multiple (three or more) code fragments. Nevertheless, the traditional simplistic graph is tailored solely to representing binary associations rather than multivariate associations [13], [14], so it is challenging to discover the interrelationships between multiple code fragments.

To elucidate the context of multivariate associations among codes, we present a motivation case (an abridged multithreaded scenario within an actual project) in Fig. 1. On the left side of Fig. 1, three classes denoted as T , M , and P , are defined. The T class inherits from the *Thread* class and overrides the

Manuscript received 13 November 2023; revised 25 January 2024; accepted 22 April 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62202177, in part by the Natural Science Foundation of Guangdong Province under Grant 2022A1515110564, and in part by the Guangzhou Science and Technology Plan Project under Grant 202206010091 and Grant 2023B03J1363. Associate Editor: D. Lo. (*Corresponding author: Chaoda Peng.*)

Shaojian Qiu, Mengyang Huang, Yun Liang, and Chaoda Peng are with the College of Software Engineering, South China Agricultural University, Guangzhou 510642, China (e-mail: qiushaojian@scau.edu.cn; huangmengyang@stu.scau.edu.cn; sdliangyun@163.com; chaodapeng@scau.edu.cn).

Yuan Yuan is with the School of Computer Science and Engineering, Beihang University, Beijing 100191, China, and also with the State Key Laboratory of Software Development Environment, Zhongguancun Laboratory, Beijing 100095, China (e-mail: yuan21@buaa.edu.cn).

Digital Object Identifier 10.1109/TR.2024.3393415

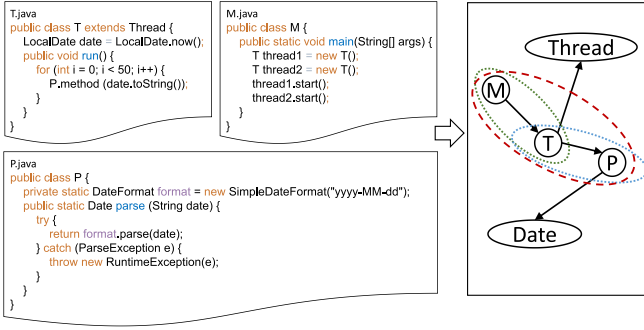


Fig. 1. Motivation case of multivariate association between codes.

run method. Within this *run* method, the *parse* function of the *P* class is executed. Notably, due to the static access modifier of the variable *format*, all instances of the *P* class share this variable. However, the variable's data type, *DateFormat*, is thread-unsafe. In our scenario, the *main* method of class *M* generates two *T* threads, each of which executes the *run* method. Consequently, multiple instances of *P* are invoked by diverse threads within the project concurrently, thereby potentially triggering thread exceptions. On the right side of Fig. 1, a simplified graph is employed to depict the dependencies among classes *M*, *T*, and *P*. Suppose we exclusively consider the dependency between *M* and *T* (i.e., the section encompassed by the green dashed line) or solely focus on the calling relationship between *T* and *P* (i.e., the portion enclosed by the blue dashed line), the aforementioned thread-unsafe exception cannot be identified. In this context, the properties of code multivariate correlations should be exploited in the defect prediction process.

In summary, code representation learning with the characteristics of both multiple view and multivariate associations remains an area that has not yet been fully explored. To address these challenges, we propose an innovative code multiview hypergraph representation learning method for software defect prediction (MVHR-DP). This method leverages the capabilities of hypergraphs to model multivariate associations and extends its functionality to encompass multiple views for defect information mining. First, MVHR-DP constructs the hypergraph structures of each view in accordance with different feature perspectives. Second, MVHR-DP melds this multiview information into an adaptable-dimensional hypergraph representation. Finally, we design a hypergraph neural network with hyperedge convolution operation to mine the multiview and multivar-association information between the codes, thereby improving the features' completeness of modeling data and the effect of the defect prediction model.

The main contributions of this article are as follows.

- 1) We introduce a novel hypergraph structure tailored to capture the intricacies of multivariate associations between codes. Specifically, we put forth a hypergraph-based representation learning approach that enhances the efficacy of the extracted code features.
- 2) We design a code multiview fusion method to strengthen the completeness of code features and construct a defect prediction framework based on multiview and multivar-association code features.

- 3) We conduct comprehensive experiments involving 29 pairs of within-project, 19 pairs of cross-version, and 90 pairs of cross-project defect prediction (CPDP) tasks. The empirical outcomes substantiate our claims, demonstrating that the prediction model constructed using MVHR-DP-generated features yields superior area under the curve (AUC), F-measure, and matthews correlation coefficient (MCC) results compared to other state-of-the-art defect prediction techniques.

The rest of this article is organized as follows. Section II presents related work. Section III presents preliminary knowledge for MVHR-DP. Section IV presents the technical specification of MVHR-DP. Sections V and VI elaborate on the experimental design and results. Section VII discusses some additional experiments and threats to validity. Finally, Section VIII concludes this article.

II. RELATED WORK

The key to constructing an effective defect prediction model is finding the code's defect-related feature representations. Currently, the mainstream code features used in software defect prediction are static and process features. Notably, static features revolve around metrics concerning code size and inherent complexity [6], [15]. For instance, the Halstead features rely on operand and operator counts, McCabe features derive from control flow complexity, while CK features are rooted in object-oriented principles. Guided by the assumption that "greater program complexity correlates with heightened defect likelihood," researchers have employed these static features to construct training data for defect prediction models. However, static features cannot contain information about code changes. Consequently, researchers have delved into the realm of code process features. For instance, Pornprasit et al. [16] quantified code change content using fine-grained alterations and maintainer insights drawn from commit logs. Kondo et al. [17] conducted defect prediction based on features extracted from code change paths and context metrics. Similarly, Zeng et al. [18] extracted attributes from commit messages and code changes, subsequently employing these features to predict defects.

The utilization of static and process features (referred to as handcrafted features) in the aforementioned studies can be influenced by subjective human choices. It might fail to capture the code's full structural and semantic intricacies. Consequently, some researchers have turned to deep learning techniques to extract defect-related attributes from the internal architecture of the code. For instance, Wang et al. [19], Li et al. [20], and Dam et al. [21] have endeavored to derive semantic features from the code's AST using deep belief networks (DBNs), convolutional neural networks (CNNs), and long short-term memory neural networks, respectively. In a different approach, Phan et al. [22] transformed the source code into a program control flow graph (CFG) and employed CNNs to conduct convolution operations on the CFG's topological sequence. This facilitated the extraction of more profound semantic features from the code. In addition, Chen et al. [23] introduced defect prediction using the software visualization and deep transfer learning (DTL-DP) method. DTL-DP leveraged a more visualization program

representation, enhancing the efficiency of defect-related feature extraction.

The code features employed in constructing defect prediction models above mainly capture intrinsic aspects of code instances, such as complexity features or structural-semantic characteristics. However, it should be noted that there is valuable association information embedded within the programs [7], [24]. Extracting code features from associated data can significantly augment the comprehensiveness of code representation. Currently, several researchers [10], [11], [12] have ventured into utilizing network embedding techniques to glean pertinent insights from the code contextual associations. Qu et al. [10] employed the node2vec network embedding approach to translate class-dependent structures into low-dimensional vector representations while preserving the code's topological information. These vector representations were then harnessed to construct a defect prediction model. Building upon this, Zeng et al. [12], [25] fused node2vec features with traditional static attributes and semantic features derived from ASTs. This amalgamation was subjected to graph convolution operations to draw out more intricate association information, thereby enhancing the comprehensiveness of code feature representation. In addition, Gong et al. [9] introduced self-ego metrics and global metrics from social network analysis to further enrich the feature representation obtained from the code dependency network.

Despite the advances achieved by these code association-based feature extraction methods, they often focus solely on dependencies between codes, inadvertently disregarding the contextual information encapsulated within other code views. In addition, defects in the code may involve multiple pieces of code simultaneously, and such defects cannot be effectively expressed with a simple graph structure. To address these challenges, this article proposes the MVHR-DP approach to enhance defect prediction performance.

III. PRELIMINARIES

A. Software Defect Prediction

Software defect prediction assesses whether new code carries defects, thereby judiciously aiding developers and the reliability assurance team in allocating testing resources. A defect prediction task typically unfolds in several stages. First, historical source code files are gathered from software repositories, and these files are labeled as either defective or clean based on released data. Subsequently, corresponding data features are extracted from the code. These instances and their labels are then employed to train classifiers (e.g., logistic regression and random forests). Ultimately, new instances are input into the trained model to predict whether they possess defects [26].

The set of instances used to construct the classifier model is known as the training set, while the set of instances employed to evaluate the trained classifier's performance is termed the testing set. This study concentrates on the following three primary scenarios: within-project defect prediction (WPDP), where both training and testing sets originate from the same version of the same project; cross-version defect prediction (CVDP), where the training and testing sets emerge from consecutive versions of the same project; and CPDP, where one project is employed as

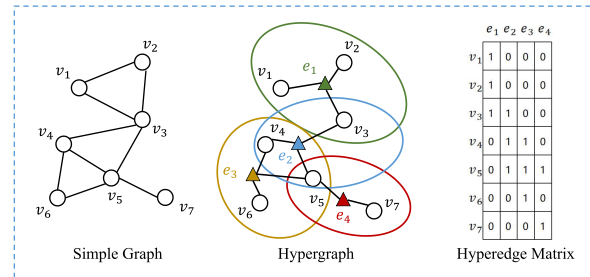


Fig. 2. Process of hypergraph expansion.

the testing data while another is selected as the training set [27]. In CVDP and CPDP tasks, the training and testing sets are also called source and target projects, respectively.

B. Multiview Fusion

In reality, objects often possess multiple facets that can be perceived from diverse angles or approaches. Drawing an analogy to code representation, the features inherent to code can be encapsulated through complexity measurements or the view derived from topological structures. A more comprehensive and robust information representation can be achieved by fusing these various views.

Leveraging the benefits offered by knowledge from multiple perspectives, researchers have introduced multiview fusion methods [28]. This approach entails modeling a specific view while leveraging insights from other views pertaining to the same data. This synergistic optimization of data representation is proven to be beneficial.

In multiview code representation, we denote \mathbf{X}^i to signify data from the i th view. Within this framework, \mathbf{x}_n^i represents the n th code instance from the i th view. The i th view of the code can, thus, be succinctly expressed as $\mathbf{X}^i = (\mathbf{x}_1^i, \mathbf{x}_2^i, \dots, \mathbf{x}_n^i)$. By extending the process of extracting code features to encompass the multiview fusion paradigm, we can bolster the comprehensiveness of features in the modeling data.

C. Hypergraph Construction

In general, simple graphs are commonly employed to model data and their interrelations. Illustrated on the left side of Fig. 2, nodes within a simple graph symbolize data points, while edges connecting nodes denote associations. Represented by adjacency matrices, simple graphs are characterized by edges linking only two vertices—constrained to a degree of 2 per edge. Nonetheless, in practical situations, multivariate associations can often exist between nodes. Take the defect prediction task as an example: a defect might arise from multiple code fragments. Consequently, finding a more fitting graph structure to depict such multivariate associations becomes imperative.

To handle the abovementioned case, researchers have introduced the concept of a hypergraph structure [13]. Depicted on the middle and right side of Fig. 2, hyperedges within a hypergraph link two or more vertices. Hypergraphs excel in portraying and mining nonlinear high-order data dependencies. In the context of this article, a hypergraph is defined as $G = (V, E, \mathbf{W})$,

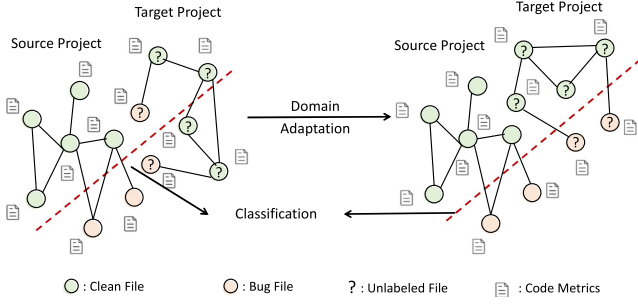


Fig. 3. Domain adaptation for software defect prediction.

encompassing a vertex set V , a hyperedge set E , and a diagonal matrix \mathbf{W} that captures hyperedge weights. A hypergraph can be articulated using a hyperedge matrix \mathbf{H} , where \mathbf{H} is sized $|V| \times |E|$, and the h in \mathbf{H} can be delineated as

$$h(v, e) = \begin{cases} 1, & \text{if } v \in e \\ 0, & \text{if } v \notin e. \end{cases} \quad (1)$$

For a vertex $v \in V$, its degree is defined as

$$d(v) = \sum_{e \in E} w(e) h(v, e). \quad (2)$$

For an edge $e \in E$, its degree is defined as

$$\mathcal{E}(e) = \sum_{v \in V} h(v, e). \quad (3)$$

According to this, the degree matrix \mathbf{D}_v of the hypergraph vertex and the hyperedge degree matrix \mathbf{D}_e can be obtained.

In our approach, hypergraphs are adopted to represent multivariate associations among codes. A comprehensive explanation of this methodology will be presented in the forthcoming method section.

D. Domain Adaptation

Illustrated on the left side of Fig. 3, inherent differences in personnel, functionalities, and coding conventions across distinct projects give rise to variations in data distribution. Consequently, a discrepancy emerges between the source project ($P^s = \{\mathbf{X}^s | \mathbf{Y}^s\}$) and the unlabeled target project ($P^t = \{\mathbf{X}^t\}$). Thus, a software defect prediction model fashioned using the source project often encounters challenges when directly applied to the target project. To handle this dilemma, domain adaptation is frequently employed. The primary aim of domain adaptation is to alleviate the impact of data shift that occurs during the transference of knowledge from a source domain to a target domain [29]. Within the general domain adaptation process, a feature extractor, denoted as a generator f_g , is fine-tuned to acquire data representations from both the source and target domains. Concurrently, a domain classifier, functioning as a discriminator, is trained to differentiate between data representations of the source domain $f_g(\mathbf{X}^s)$ and the target domain $f_g(\mathbf{X}^t)$. This is achieved by minimizing the domain classification loss \mathcal{L}_d . The overarching objective of the domain adaptation process can be encapsulated within the following loss function \mathcal{L} :

$$\mathcal{L} = \mathcal{L}_c(f_g(\mathbf{X}^s), \mathbf{Y}^s) + \mathcal{L}_d(f_g(\mathbf{X}^s), f_g(\mathbf{X}^t)). \quad (4)$$

Domain invariant representations can be obtained when the model converges loss \mathcal{L}_c and domain divergence \mathcal{L}_d is minimized [30].

In recent years, graph representation learning has garnered extensive utilization within the software engineering area. Demonstrated in Fig. 3, the application of domain adaptation to the defect prediction task, grounded in graph representation learning, holds the potential to enhance both the transfer capability of generated association features and the overall generalizability of the prediction model. Within our proposed approach, we apply domain adaptation in code multiview hypergraph representation learning to improve the performance of CVDP and CPDP tasks.

IV. METHOD

This article introduces the MVHR-DP approach, a multiview hypergraph representation learning method, which centers on amalgamating multiple code views and incorporating multivariate associations. Fig. 4 offers a comprehensive view of the framework of MVHR-DP. The representation learning process encompasses four pivotal steps, as follows.

- 1) We extract three distinct categories of metric sets from files as the basis for building code views. These views encompass traditional static code metrics (TSM), complex network metrics (CNM), and low-dimensional network embedding metrics (NEM).
- 2) For each code view, we meticulously construct the corresponding hypergraph employing the nearest K neighbor algorithm. Subsequently, we amalgamate the hypergraph associations gleaned from the three distinct code views into a multiview hypergraph with an adjustable-dimensional structure.
- 3) We introduce a hypergraph neural network that facilitates the hyperedge convolution operation. By constructing a hidden layer, we efficiently mine the multivariate associations amidst code instances.
- 4) To corroborate the effectiveness of our proposed approach, we build defect prediction models catering to WPDP, CVDP, and CPDP scenarios.

Each of these steps is meticulously detailed as follows.

A. Code Multiview Feature Generation

In this step, we introduce three types of code metrics [10], [12], [25] as the basis for code views. The details are as follows.

TSM: TSM provides a quantitative evaluation by analyzing various static structures and characteristics of software code. Compared to alternative metrics, TSM is formulated based on rules derived from human analysis, thereby encapsulating specific intrinsic logical information within the code. Notably, it operates independently of historical software data, enabling the detection of defects in new software versions that may have yet to undergo multiple iterations. This study introduces a comprehensive set of TSM comprising 20 metrics covering diverse aspects, including the measure of function abstraction, average method complexity, number of public methods, etc. We derive these TSM metrics utilizing the widely recognized

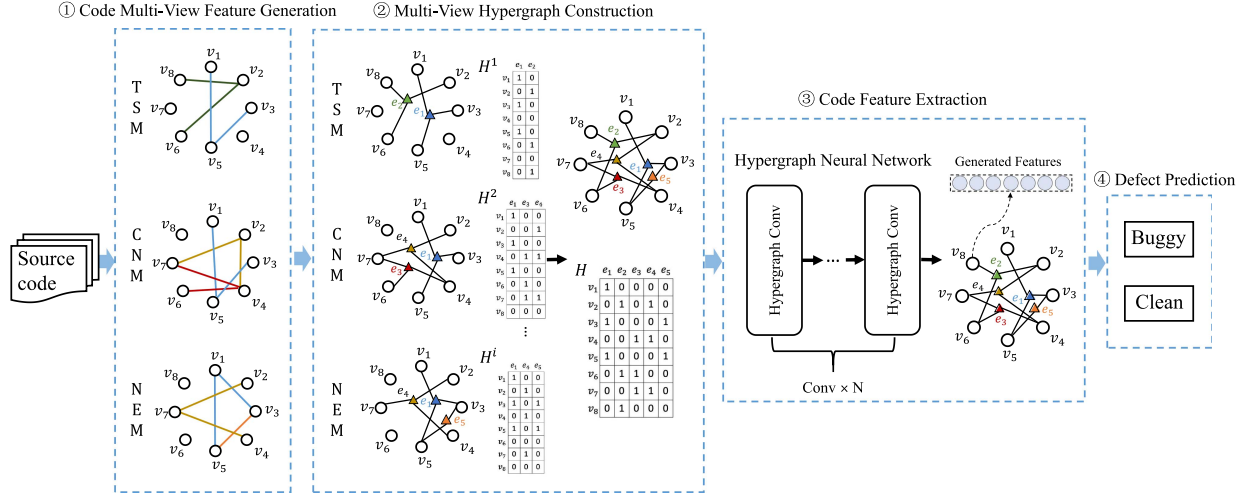


Fig. 4. Framework of MVHR-DP.

tool *Understand*.¹ These metrics are extensively used within software defect prediction studies [25], [31], [32].

CNM: This category is obtained by analyzing the network of code dependencies, which implies the information regarding the three types of connections in code files. The first type pertains to inherit: when class v_1 inherits from class v_2 or implements the interface v_2 , a directed edge $e_{12} = (v_1, v_2)$ is established. The second type involves aggregation: when class v_1 incorporates the attributes of class v_2 , a directed edge $e_{12} = (v_1, v_2)$ is formed. The third type revolves around parameter: if the method of class v_1 invokes the method of class v_2 , a directed edge $e_{12} = (v_1, v_2)$ is created. The category encompasses 17 metrics, including measurements like the number of weak components, neighborhood, and degree within the code class dependency network. These code network metrics have also been integral to software defect detection tasks [12], [33], [34]. The code network and requisite metrics for generating CNM can be acquired via the publicly available API Dependencyfinder.²

NEM: NEM arises from class-dependent networks of codes through network embedding learning. In this context, we leverage the node2vec method to map each class node onto a low-dimensional vector [10], [35]. While there may exist a certain degree of information overlap with the CNM, NEM goes beyond by incorporating additional topological insights extracted from the code network. This incorporation significantly enriches the comprehensiveness of code feature representation, providing a more nuanced understanding of the intricate relationships within the codebase. By leveraging these supplementary insights, NEM aims to enhance the precision and effectiveness of code defect prediction models.”

Distinct code metrics inherently carry complementary information. To extract code features more comprehensively and precisely, thereby enhancing the overall effectiveness of the modeling data, we leverage the three aforementioned feature sets

as the foundation for generating nodes and hyperedges within the multiview hypergraph representation.

B. Multiview Hypergraph Construction of Code

During this phase, we initiate the construction of a hypergraph for each distinct view, culminating in the integration of these individual hypergraphs into a versatile multidimensional hypergraph, facilitating multiview representation learning.

In our approach, the i th view of each code instance is encapsulated by the notation $\mathbf{X}^i = (\mathbf{x}_1^i, \mathbf{x}_2^i, \dots, \mathbf{x}_n^i)$. At this juncture, each instance is treated as a node within the graph structure. We proceed by crafting dedicated hyperedges for each view based on the similarity distances observed between code instances. Constructing a hypergraph structure based on code similarity allows us to establish connections among corresponding feature views, enabling the identification of code fragments with similar characteristics and the discovery of differences between codes. We iteratively designate a node as a centroid and employ the Euclidean distance metric to identify its K nearest neighbors. Consequently, each hyperedge takes shape by connecting a node with its closest K nodes. Following traversal of all nodes within a specific view, we compile the hyperedge correlation matrix \mathbf{H} boasting dimensions $n \times n$, wherein a cumulative total of $n \times (K + 1)$ elements equate to 1, while the remainder adopts a value of 0.

Subsequently, we amalgamate the hypergraph associations derived from the three code views into an encompassing multiview hypergraph and concatenate the hyperedge matrices across all views. Furthermore, since the preceding method of constructing a hypergraph disregards class dependencies, potentially resulting in the omission of pertinent association information, we introduce class dependencies as supplementary data to initialize the hyperedge set E [10], [14]. We then employ TSM+NEM+CNM as the collective node features, effectively constituting the hypergraph vertex set V . Ultimately, we construct a fusion hypergraph $G = (V, E, \mathbf{W})$ that incorporates multiview information.

¹Understand: <https://scitools.com>

²Dependencyfinder: <https://depfind.sourceforge.io/>

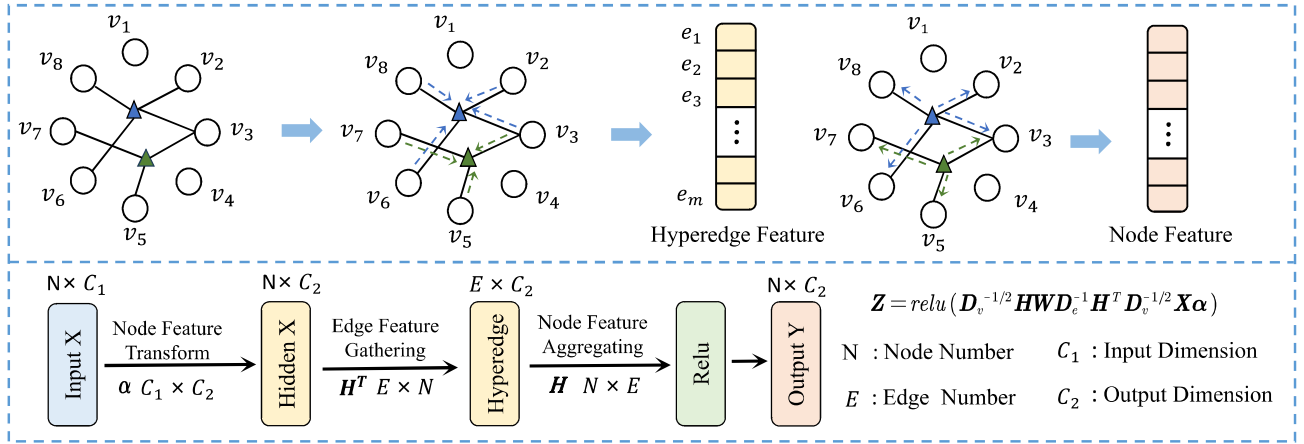


Fig. 5. Illustration of the hyperedge convolution.

C. Code Feature Extraction

We construct a hypergraph neural network to extract high-order code association information. Given the code multiview hypergraph $G = (V, E, W)$ constructed in the previous step, where the matrix W is regarded as the weight of the hyperedge. Let $\theta = D_v^{-1/2} H W D_e^{-1} H^T D_v^{-1/2}$ and $\Delta = I - \theta$, I is the identity matrix. Since this Δ is a semipositive definite matrix, we can use eigen decomposition $\Delta = \Phi \Lambda \Phi^T$ to obtain orthonormal eigenvectors $\Phi = \text{diag}(\phi_1, \dots, \phi_n)$ and nonnegative eigenvalue diagonal matrix $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n)$. Then, the Fourier transform for the instance feature $X = (x_1, \dots, x_n)$ in hypergraph is defined as $\hat{X} = \Phi^T X$, where the eigenvectors are regarded as the Fourier bases, and the eigenvalues are interpreted as frequencies. The spectral convolution of instance feature X and filter f can be denoted as

$$f * X = \Phi ((\Phi^T f) \odot (\Phi^T X)) = \Phi f(\Lambda) \Phi^T X \quad (5)$$

where \odot denotes the element-wise Hadamard product and $f(\Lambda) = \text{diag}(f(\lambda_1), \dots, f(\lambda_n))$ is a function of the Fourier coefficients. However, the time complexity of the inverse forward Fourier transform is $O(n^2)$. To solve the problem of the high time complexity, we follow the K -order polynomial parameterization $f(\Lambda)$ used by [36]. Finally, we get the hyperedge convolution operation as follows:

$$f * X = \alpha_0 X - \alpha_1 D_v^{-1/2} H W D_e^{-1} H^T D_v^{-1/2} X \quad (6)$$

where α_0 and α_1 are filter parameters on all hypergraph nodes. We further use a single parameter α to avoid the overfitting problem, defined as

$$\begin{cases} \alpha_1 = -\frac{1}{2}\alpha \\ \alpha_0 = \frac{1}{2}\alpha D_v^{-1/2} H W D_e^{-1} H^T D_v^{-1/2}. \end{cases} \quad (7)$$

The hyperedge convolution operation can be simplified to the following expression:

$$f * X = \alpha D_v^{-1/2} H W D_e^{-1} H^T D_v^{-1/2} X. \quad (8)$$

When we have n instance features of dimension C_1 , $X \in \mathbb{R}^{n \times C_1}$ in hypergraph, our hyperedge convolution can be

expressed as follows:

$$Z = \text{relu}(D_v^{-1/2} H W D_e^{-1} H^T D_v^{-1/2} X \alpha). \quad (9)$$

As illustrated in Fig. 5, $\alpha \in \mathbb{R}^{C_1 \times C_2}$ constitutes a parameter that is learned during the training process. The filter α is applied across the nodes within the hypergraph to extract features. Following the hypergraph convolution module, we obtain the refined high-order code information $Z \in \mathbb{R}^{n \times C_2}$. Notably, the hypergraph convolution layer facilitates an information transformation between node-edge-node, significantly enhancing code features using the hypergraph structure and effectively extracting high-order association information among code fragments. In our approach, the weight matrix W is set as the unit matrix I , implying that the weights of all hyperedges are uniform, and $C_1 = C_2$. Multiple hyperedge convolution operations are performed, and the rectified linear unit (ReLU) activation function is employed to construct a hypergraph CNN module. This module assists in capturing the multivariate association information between codes, leading to an enhanced representation of code features.

D. Defect Prediction

To bolster the generality and efficacy of our proposed approach, we formulate prediction models for WPDP, CVDP, and CPDP tasks, respectively. Addressing the class imbalance issue within the dataset, we employ the SMOTETomek algorithm, as outlined in the literature [37], to enhance the training effectiveness of the classifier.

1) *Prediction Model for WPDP*: WPDP technique is instrumental in identifying potential defects, enabling development teams to effect timely improvements and elevate software reliability. Employing the developed multiview hypergraph representation, we construct a dedicated model tailored to the WPDP task. The model's schematic is depicted in Fig. 6.

This WPDP process unfolds in two primary stages. First, we employ the methodology elucidated in the preceding section to generate modeling data by the multiview hypergraph construction. Second, employing cross-validation, we partition the

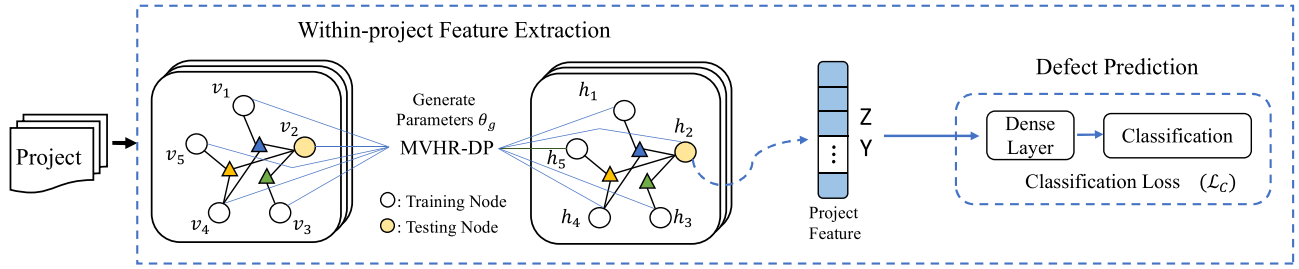


Fig. 6. Prediction model for WPDP task.

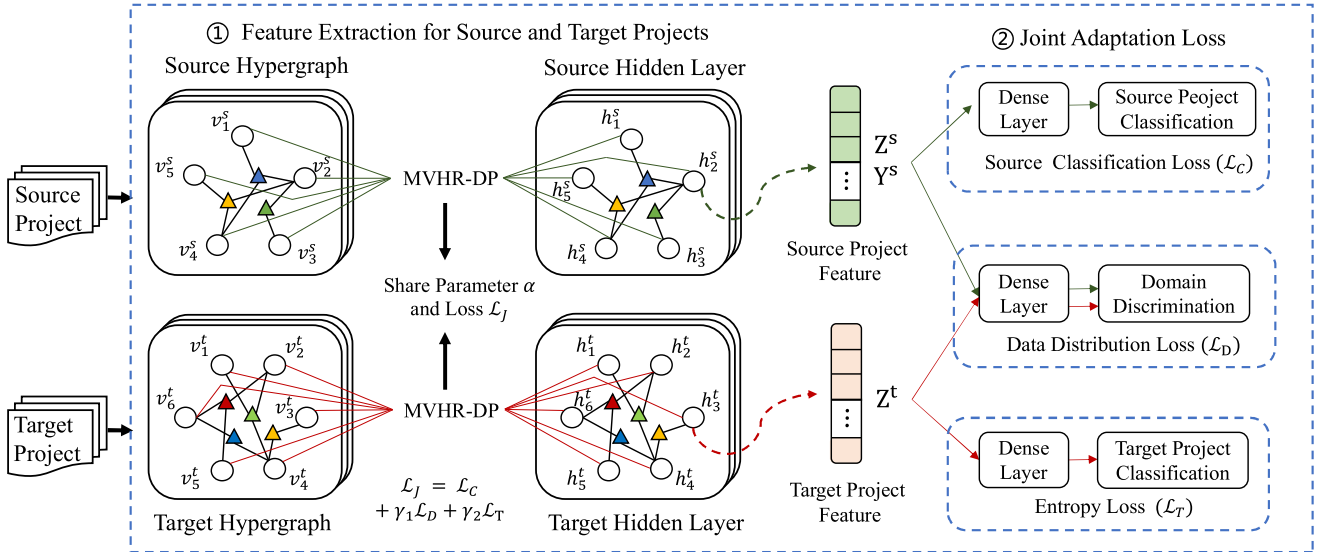


Fig. 7. Prediction model for CVDP and CPDP tasks.

dataset into distinct training and testing subsets, with 80% of code instances allocated to the training set and the remaining 20% to the testing set.

Within the hypergraph CNN module, node h signifies the updated hidden layer representation of node v , while α denotes the parameter steering node updates. Consequently, the high-order hidden layer features of the code are extracted and denoted as Z . The crux of the feature extraction procedure unfolds as shown in (9).

The code instance set and hypergraph adjacency matrix are X , H , and project labels Y . D_e , D_v , respectively, represent the hyperedge degree matrix and node degree matrix of the multiview hypergraph structure of the code.

The extracted features are passed through a classifier to predict whether the file is defective. The classifier and its parameters are f_c and θ_c . We use logistic regression as the classifier and classification cross-entropy loss $\mathcal{L}_C(f_c(Z; \theta_c), Y)$ is as follows:

$$\mathcal{L}_C(f_c(Z; \theta_c), Y) = -\frac{1}{N} \sum_{i=1}^N y_i \log(f_c(Z; \theta_c)) \quad (10)$$

where N represents the number of project files and y_i denotes the label of the i th node in the label set Y .

2) *Prediction Model for CVDP and CPDP*: In practical scenarios, commencing a new project often entails a dearth of labeled data. Consequently, other projects or historical versions are frequently employed as the training set (source project) to train the prediction model. Subsequently, this model is harnessed to forecast software defects in novel projects, which serve as the testing set (target project). Nonetheless, such a methodology encounters challenges rooted in the discordances of structural information and data distribution across different projects.

To address this quandary while concurrently extracting code association features, we introduce an inventive joint adaptation loss during constructing prediction models for CVDP and CPDP tasks. The model architecture is depicted in Fig. 7.

Leveraging the proposed code multiview fusion representation learning approach, we construct hypergraph structures for both the source project ($P^s = \{X^s|Y^s\}$) and the target project ($P^t = \{X^t\}$). Perform feature extraction on them, respectively, and the extraction process is shown in (9). The source and target projects share a parameter α of the update code instances. The high-order hidden layer features of the code extracted by the hypergraph CNN module are denoted as Z^s and Z^t .

To better learn knowledge and improve the transfer ability of code-associated features to assist in the CVDP and CPDP task. We proposed a joint adaptation loss consisting of an adversarial

module, a source classifier, a domain discriminator, and a target classifier working together to learn domain invariant code representations. The overall objective is as follows:

$$\mathcal{L}_J = \mathcal{L}_C(f_c(\mathbf{Z}^s; \theta_c), \mathbf{Y}^s) + \gamma_1 \mathcal{L}_D(\mathbf{Z}^s, \mathbf{Z}^t) + \gamma_2 \mathcal{L}_T(f_c(\mathbf{Z}^t; \theta_c)) \quad (11)$$

where γ_1 and γ_2 are the balance parameters. The \mathcal{L}_C , \mathcal{L}_D , and \mathcal{L}_T represent the source classification loss, the domain adaptation loss, and the target entropy loss, respectively. Also, use regression as classifier f_c with parameter θ_c for source and target projects classification. The details are introduced as follows.

1) *Source classification loss*: The classification loss $\mathcal{L}_C(f_c(\mathbf{Z}^s; \theta_c), \mathbf{Y}^s)$ is to minimize the cross-entropy loss for the labeled data in the source project

$$\mathcal{L}_C(f_c(\mathbf{Z}^s; \theta_c), \mathbf{Y}^s) = -\frac{1}{N^s} \sum_{i=1}^{N^s} y_i \log(f_c(\mathbf{Z}_i; \theta_c)) \quad (12)$$

where N^s represents the number of source project files and y_i denotes the label of the i th node in the source project label set \mathbf{Y}^s .

2) *Domain adaptation loss*: The domain adaptation loss $\mathcal{L}_D(\mathbf{Z}^s, \mathbf{Z}^t)$ ensure the similarity of code representations \mathbf{Z}^s and \mathbf{Z}^t obtained through hypergraph CNN feature extraction, thereby mitigating the impact of data distribution differences. To achieve this, we use gradient reversal layer [38] as a discriminator to differentiate code representations

$$\mathcal{L}_D(\mathbf{Z}^s, \mathbf{Z}^t) = -\frac{1}{N^s + N^t} \sum_{i=1}^{N^s + N^t} m_i \log(\hat{m}_i) + (1 - m_i) \log(1 - \hat{m}_i) \quad (13)$$

where $m_i \in \{0, 1\}$ denotes the ground truth of the file project and \hat{m}_i denotes the domain prediction for the i th file in the source and target projects, respectively.

3) *Target entropy loss*: To utilize valuable information in the target project. We employ an entropy loss for the target classifier

$$\mathcal{L}_T(f_c(\mathbf{Z}^t; \theta_c)) = -\frac{1}{N^t} \sum_{i=1}^{N^t} \hat{y}_i \log(\hat{y}_i). \quad (14)$$

In other words, it serves to enforce that the decision boundary does not cross data-dense regions and will prevent information loss of the target project [39], where \hat{y}_i are the classification prediction for the i th file in the target project.

\mathcal{L}_C , \mathcal{L}_D , \mathcal{L}_T are jointly optimized via our objective function in (11), and all parameters are optimized using the standard backpropagation algorithms.

Algorithm 1 provides a pseudocode representation to enhance the clarity of the MVHR-DP process and details. As depicted in Step 1, we extract three distinct categories of metric sets, serving as the foundation for constructing a multiview hypergraph. Subsequently, MVHR-DP commences by establishing hypergraph structures for each code view, which are then amalgamated into a comprehensive fusion hypergraph. Following this, a hypergraph neural network is utilized in Step 2 to extract code features and intricate associations. Ultimately, as delineated in Step 3, we

Algorithm 1: MVHR-DP.

Input:

The source code of project files P .

The number of code views M .

Output:

Prediction models f_{WPDP} , f_{CPDP} , f_{CVDP} .

Step 1: Multi-View hypergraph construction

1: Initialize the hypergraph G , which to be fused;

2: **for** all iteration i from 1 **to** M **do**

3: Initialize the hypergraph G^i for the i -th view;

4: **for** j -th file **in** P **do**

5: Generate metric x_j^i to collection X^i ;

6: **end for**

7: **for** x_j^i **in** X^i **do**

8: Create hyperedge e_j from top-k neighbors;

9: Add e_j to i -th code hypergraph G^i ;

10: **end for**

11: Fuse multi-view hypergraph by appending G^i to G ;

12: **end for**

Step 2: Code feature extraction

13: Construct hypergraph neural network by G ;

14: **if** perform WPDP task **then**

15: Train network by loss \mathcal{L}_C , as (10);

16: **else if** perform CVDP or CPDP tasks **then**

17: Train network by loss \mathcal{L}_J , as (11);

18: **end if**

19: Generate feature Z through trained network, as (9);

Step 3: Defect Prediction

20: Build f_{WPDP} , f_{CPDP} , f_{CVDP} by extracted feature representation Z .

build a defect prediction model tailored to specific defect tasks to perform defect prediction.

V. EXPERIMENT DESIGN

A. Datasets

We carried out experiments on 10 Java open-source projects (a total of 29 project versions) released on the PROMISE database, which is widely used in software defect prediction research [12], [25], [40]. Based on this data repository, a total of 29 pairs of WPDP tasks, 19 pairs of CVDP tasks, and 90 pairs of CPDP tasks can be formed. We download the corresponding code version from each project's official website for our experiments. Table I shows detailed information about these projects, including project name, version, description, number of files, defect file number, and defect rate. The dataset consists of projects with different sizes (the number of project files ranges from 103 to 919) and defect rates (lowest 6.67% and highest 92.89%).

B. Experimental Setup

To verify the effectiveness of MVHR-DP, we focus on the following three research questions (RQs).

RQ1: How much improvement can MVHR-DP achieve in WPDP tasks?

TABLE I
DATASETS

Projects	Description	Version	Nodes	Bug file	Defective%
ant	A cross-platform construction tool	1.5	291	32	11.00%
		1.6	347	92	26.51%
		1.7	738	164	22.22%
camel	Enterprise integration framework	1.2	577	215	37.26%
		1.4	820	145	17.68%
		1.6	919	188	20.46%
ivy	Java-based dependency manager	1.4	240	16	6.67%
		2.0	351	40	11.40%
jEdit	Text editor developed in Java	3.2.1	269	89	33.09%
		4	301	74	24.58%
		4.1	311	79	25.40%
log4j	A Java-based versatile logging framework	1.0	128	33	25.78%
		1.1	103	37	35.92%
		1.2	197	183	92.89%
lucene	An open-source library	2.0	192	91	47.40%
		2.2	244	144	59.02%
		2.4	337	202	59.94%
poi	Java library to access Microsoft format files	2.0	311	37	11.90%
		2.5.1	381	246	64.57%
		3.0	431	279	64.73%
velocity	Java-based template engine	1.4	193	146	75.65%
		1.5	213	142	66.67%
		1.6	228	78	34.21%
xalan	A library for transforming XML files	2.4	714	110	15.41%
		2.5	794	383	48.24%
		2.6	820	374	45.61%
xerces	An open-source Java library for parsing XML	1.2	431	69	16.01%
		1.3	445	66	14.83%
		1.4.4	332	213	64.16%

RQ2: Is the proposed MVHR-DP method better than the comparative methods in CVDP?

RQ3: Do code features learned from MVHR-DP outperform related methods in CPDP?

In our work, we conduct experiments on three RQs separately. To evaluate the performance of our end-to-end model MVHR-DP, we compared eight related defect prediction methods. The experimental setup and comparison methods for each RQ are as follows.

- 1) *Traditional*: Software defect prediction on traditional handcrafted code features.
- 2) *DBN*: A defect prediction method uses DBN to automatically learn semantic features of code from AST [41].
- 3) *Defect prediction via convolutional neural network (DPCNN)*: A method that applies standard CNN to extract features from source code and combine CNN-learned features with traditional handcrafted features [20].
- 4) *Code bidirectional encoder representations from transformers (CodeBERT)*: Recently, bidirectional encoder representations from transformers (BERT)-based technology has been widely used in code representation learning. We selected the representative method CodeBERT as a comparison method, a bimodal pretrained model for programming language and natural language [42].
- 5) *Node2defect*: A network embedding method that adopts the node2vec technology to extract features from the class-dependent network of the code [10].
- 6) *DTL-DP*: A method that visualizes the program as an image and applies AlexNet and a self-attention mechanism to extract semantic features of code [23].

7) *Unsupervised domain adaptation for defective prediction (UDA-DP)*: A method to adopt the unsupervised domain adaptation using pseudolabels to learn defective features from source programs directly [43].

8) *Graph convolutional network for defect prediction (GCN2defect)*: The graph convolution operation is performed on the class-dependent network to obtain a code association representation [12]. The node features are combined with node2vec and handcrafted software code features.

For RQ1: In the WPDP task, training and testing data are extracted from the same version of each project. Five-fold cross-validation is used in WPDP. Specifically, the dataset is randomly split into five folds. Here, 80% of the instances are used for training a prediction model, and 20% of instances are used to evaluate the model. We repeated the experiment 20 times and finally took the average result.

For RQ2 and RQ3: In the CVDP task, we conduct experiments involving two consecutive versions of the same project. The older version serves as the training set or source project, while the subsequent new versions constitute the testing set or target project (e.g., using ant-1.5 as the training set and ant-1.6 as the testing set). We carried out a total of 19 pairs of CVDP tasks. Shifting to the CPDP task, our experimental dataset comprises the latest versions of the projects. Each project is utilized for training a prediction model, leaving the remaining nine projects for testing (e.g., ant-1.7 serves as the training set, and the latest versions of the remaining nine projects are utilized as the testing set). Notably, all models undergo nine rounds of testing, resulting in 90 pairs of CPDP tasks. To verify the effectiveness of proposed domain adaptation in CVDP and CPDP tasks, we introduced a new comparison, MVHR-CVDP, and MVHR-CPDP represent training with a joint adaptation loss, while MVHR-DP does not include this modification.

Parameter Setting of MVHR-DP: The hyperparameters of the MVHR-DP model proposed in this article include the number of neighbor nodes K for constructing the hyperedge, the feature dimension extracted by the hyperedge convolution, the threshold of SMOTETomek, joint adaptation loss balance parameters, and the network training parameters. Specifically, the K is set to 5, and the output feature dimension is set to 64. The SMOTETomek threshold is set to 0.4. Regarding the learning parameters, the epoch is set to 200, and the learning rate is set to 0.01. Adam is used as the optimizer, the gamma is set to 0.9, and the weight decay is set to 5×10^{-4} . Joint adaptation loss balance parameters γ_1 and γ_2 are set to 0.6 and 0.8, respectively. Except for the traditional method, all other methods involve some randomness. Therefore, we execute these random methods 20 times and record the average results.

C. Evaluation Indicator

To evaluate the prediction performance, we use evaluation indicators AUC, F-measure, and MCC, which are widely used in software defect prediction research [1], [44].

In a defect prediction task, four results can be obtained: predict the truly defective instance as defective (true positive, TP); predict the truly clean instance as defective (false positive,

FP); predict the truly defective instance as clean (false negative, FN); predict the truly clean instance as clean (true negative, TN). Then, the evaluation indicator AUC, F-measure, and MCC are calculated as follows:

$$\text{AUC} = \int_0^1 \text{TP}(\text{FP})d\text{FP} \quad (15)$$

$$\text{F-measure} = \frac{2 \times \text{TP}}{2 \times \text{TP} + \text{FP} + \text{FN}} \quad (16)$$

$$\text{MCC} = \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{\sqrt{(\text{TP} + \text{FP})(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TN} + \text{FN})}}. \quad (17)$$

AUC measures the model's performance across all possible classification thresholds. It represents the model's ability to distinguish between positive and negative classes, with higher values indicating better performance. An AUC of 50 denotes random guessing, while 100 represents perfect performance.

F-measure is the harmonic mean of precision and recall. It is beneficial in dealing with imbalanced datasets as it takes into account both the correct identification of the positive class and the erroneous labeling of negative instances. The best value of F-measure is 100, and the worst is 0.

MCC is a balanced indicator that provides useful information even in situations of class imbalance. It ranges from -100 (entirely incorrect predictions) to 100 (perfect predictions), with 0 denoting random guess performance.

D. Statistical Test

In our comparative experiments, we apply the Scott-Knott ESD test [45] to examine the performance of MVHR-DP. The Scott-Knott ESD test is a means-comparison method that uses hierarchical clustering to divide a set of measurements (e.g., AUC) into statistically different pairs with non-negligible differences. The Scott-Knott ESD test consists of two steps: 1) Finding a partition that maximizes the measurement between pairs; 2) dividing the results into two pairs or combining them into one pair. For a detailed process description of the Scott-Knott ESD test, please refer to [45].

VI. EXPERIMENTAL RESULTS

In this section, we will analyze the experimental results of MVHR-DP for three RQs. The Scott-Knott ESD test results are shown in Figs. 8–10, which comprehensively demonstrates the performance and ranking of each method. The horizontal line in the middle of the box plot represents the median and the blue diamond indicates the average value of our examined method.

A. RQ1: How Much Improvement Can MVHR-DP Achieve in WPDP Tasks?

The results of the experimental analysis are concisely depicted in Fig. 8, which illustrates the performance ranking of nine distinct methods through the Scott-Knott ESD test across 29 pairs of WPDP tasks under the three evaluation indicators. Significantly, MVHR-DP stands out as the most effective approach, demonstrating superior performance with the highest average values for AUC, F-measure, and MCC. Specifically, MVHR-DP achieves noteworthy results across all three evaluation indicators, with respective values of 85.0, 68.0, and 50.0.

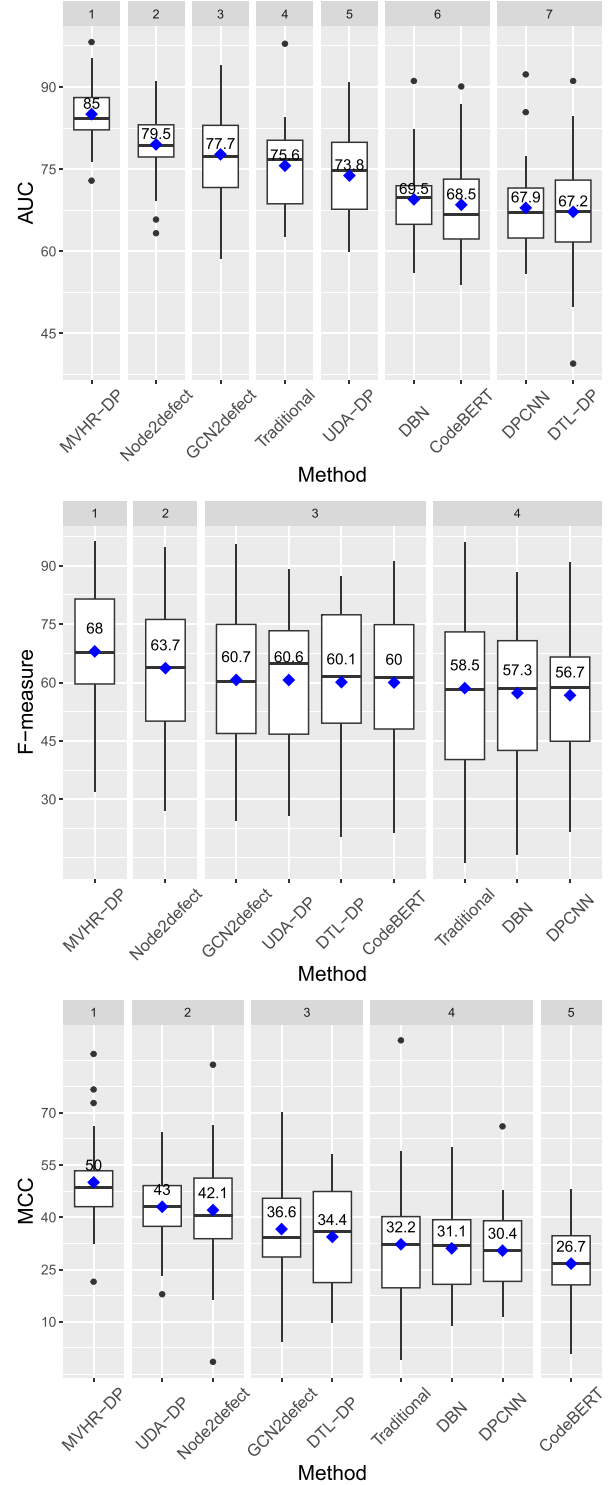


Fig. 8. Scott-Knott ESD test of 29 pairs WPDP tasks on nine methods (for RQ1).

The values are improved by 6.92%–26.49%, 6.75%–19.93%, and 16.28%–87.27% compared to the other eight methods. Furthermore, the median of MVHR-DP was also higher than all other methods. As illustrated in Fig. 8, the gap between the upper and lower edges of our method is smaller than that of the most compared methods, indicating that MVHR-DP has more robust stability.

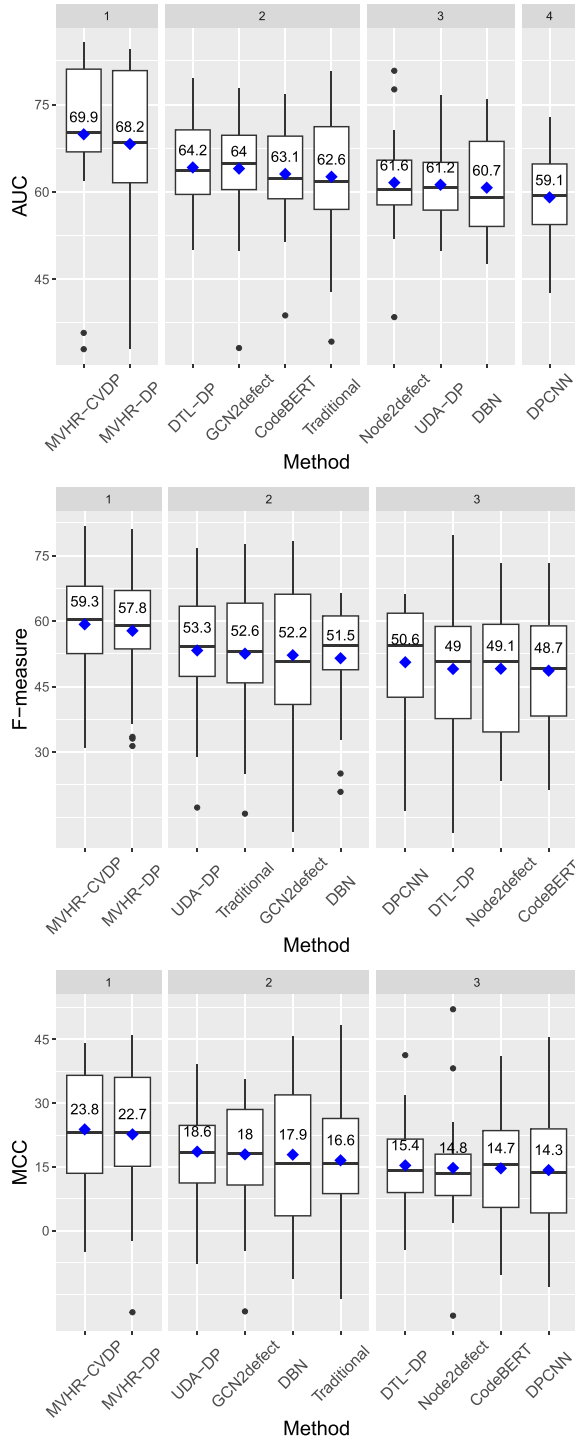


Fig. 9. Scott-Knott ESD test of 19 pairs CVDP tasks on ten methods (for RQ2).

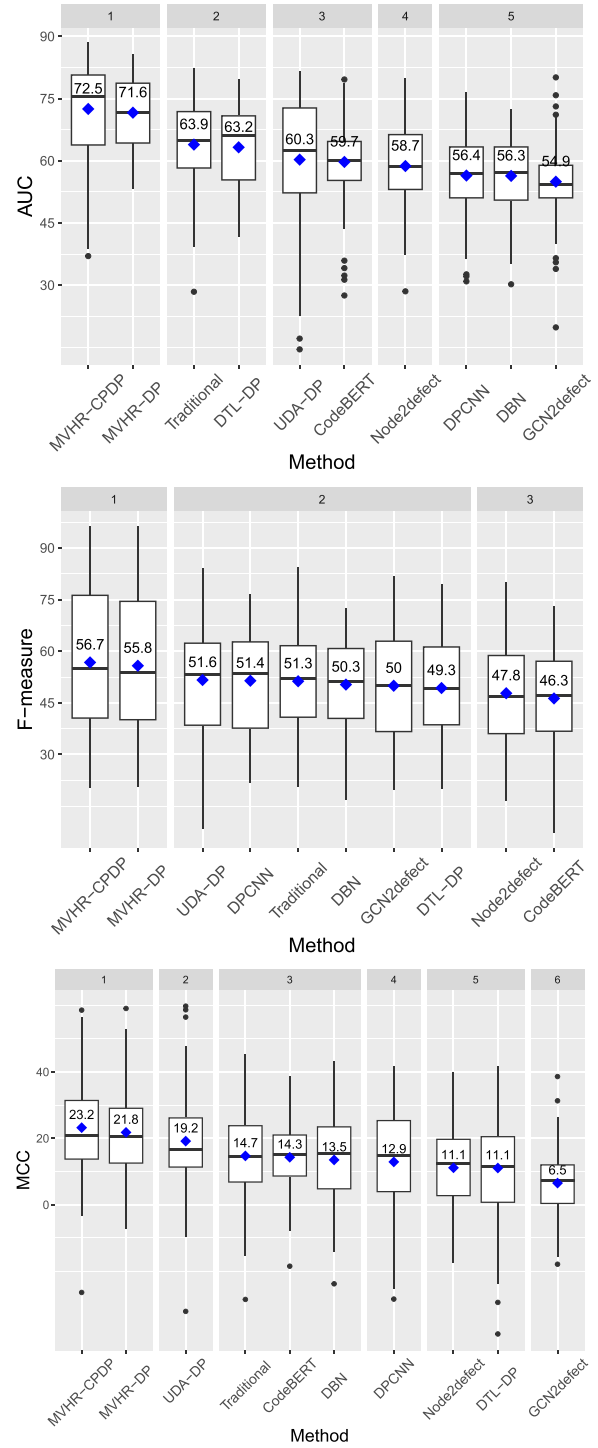


Fig. 10. Scott-Knott ESD test of 90 pairs CPDP tasks on ten methods (for RQ3).

Answer to RQ1: The above statistical test results show that MVHR-DP is better than the baseline methods in WPDP. The average AUC, F-measure, and MCC are all in the first echelon, reflecting the superiority and stability of MVHR-DP.

B. RQ2: Is the Proposed MVHR-DP Method Better Than the Comparative Methods in CVDP?

We form a pair of CVDP tasks using successive versions of the same project. A total of 29 versions of ten projects can form 19 pairs of CVDP tasks. Fig. 9 shows the performance ranking of ten different methods by the Scott-Knott ESD test. Our

MVHR-CVDP method performs stronger than the remaining nine methods under AUC, F-measure, and MCC. MVHR-DP achieves the best average results are 69.9, 59.3, and 23.8, exhibiting improvements of 8.89%–18.27%, 11.26%–21.77%, 27.96%–66.43% compared to the other nine methods, respectively. It is discernible that the MVHR-CVDP is positioned among the best-performance echelons based on the Scott-Knott ESD test ranking. Notably, the experimental results comparing MVHR-DP and MVHR-CVDP confirm that the performance is further improved by adding the joint adaptation loss.

Answer to RQ2: *The experimental results illustrate that the code features learned from MVHR-CVDP outperform compared methods in 19 pairs of CVDP tasks. The code feature representation learned by MVHR-CVDP has advantages and performs excellently in most CVDP tasks.*

C. RQ3: Do Code Features Learned From MVHR-DP Outperform Related Methods in CPDP?

In total, the ten latest projects in our dataset can be composed of 90 pairs of CPDP tasks. As can be seen from Figure, in the CPDP tasks, our MVHR-CPDP outperforms the other nine methods under AUC, F-measure, and MCC. In the 90 pairs of CPDP tasks, MVHR-DP achieves the best average result. The best results under the three evaluation indicators are 72.5, 56.7, and 23.2, reflecting improvements ranging from 13.46% to 32.06%, 9.88% to 22.46%, and 20.83% to 357% compared to the other nine methods, respectively. It can be seen that the MVHR-CPDP method belongs to the first echelon in the Scott-Knott ESD test ranking. Simultaneously, it validates that joint adaptation loss can further enhance model performance, improving its effectiveness in CPDP tasks.

Answer to RQ3: *The experimental results reflect that the defect prediction model built with MVHR-CPDP outperforms compared methods in 90 pairs of CPDP tasks. In summary, MVHR-DP is the preferred method for software reliability assurance teams performing defect prediction tasks in most cases.*

VII. DISCUSSION

A. Can Mining of Multivariate Associations Help Enhance MVHR-DP?

In MVHR-DP, we construct the hypergraph for each distinct view to mine the multivariate association properties of the code. In this section, we discuss the effectiveness of utilizing multivariate association features in defect prediction tasks through an ablation experiment. We initially considered one of the original metrics (Ori). Then, we expanded upon it by integrating binary association data from code dependencies, employing graph convolution for feature extraction (+Dep). Lastly, we heightened the approach by incorporating multivariate associations and utilizing our constructed hypergraph framework for feature extraction (+HG).

TABLE II
ABLATION EXPERIMENT OF MULTIVARIATE ASSOCIATION USING AUC

projects \ metrics	TSM			CNM			NEM		
	Ori	+Dep	+HG	Ori	+Dep	+HG	Ori	+Dep	+HG
ant	80.1	73.6	82.6	74.6	74.6	73.7	72.8	71.8	76.7
camel	67.6	71.1	72.0	73.4	72.9	79.9	76.9	77.9	78.3
ivy	66.2	70.0	70.5	68.7	73.6	75.3	70.5	65.1	74.9
jEdit	81.3	81.9	85.3	79.1	81.7	83.7	79.8	79.7	84.4
log4j	74.9	74.9	75.8	78.5	76.9	78.5	68.7	63.7	79.4
lucene	72.0	72.0	75.2	69.7	71.3	68.4	68.8	72.4	70.9
poi	76.6	79.0	77.9	73.5	79.9	79.4	82.7	78.6	79.3
velocity	76.5	84.7	80.9	74.1	73.3	83.6	82.4	82.3	83.1
xalan	74.4	76.8	78.3	69.6	73.1	77.9	73.3	77.1	77.4
xerces	80.4	87.5	85.3	82.3	86.6	91.8	90.2	88.2	91.9
AVG	75.0	77.1	78.4	74.3	76.4	79.2	76.6	75.7	79.6

The bold values represent the best AUC results in the ablation experiment.

As shown in Table II, we conducted 29 pairs of WPDP experiments by employing the three mentioned feature extraction methods separately on three different metrics (TSM, CNM, and NEM) across ten different projects. Considering space limitations, we only listed the average AUC value of the projects. The results indicate that in most cases, +HG achieved the best results, demonstrating that capturing code properties of multivariate associations can improve the effectiveness of code features learned from MVHR-DP.

B. What is the Improvement Made by the Code Multiview Features Learned From MVHR-DP?

To verify the effectiveness of multiview association for code feature extraction, we perform various combinations of the three code views: TSM (T), CNM (C), and NEM (N). The three views can be combined to form seven different combinations. Besides, the abovementioned views do not consider code class dependencies, so we added code class dependencies based on the combination of T+C+N views as supplements (MVHR-DP). We construct a code multiview hypergraph structure for each combination of code views and perform feature extraction for defect prediction. To ensure fairness, the experimental parameters for all views are maintained consistently.

Fig. 11 presents the heatmap with different view combinations based on AUC, illustrating the results for eight view combinations across 29 WPDP tasks. Due to space limitations, we have calculated the average of the prediction results for all versions of each project. This manifestation of heatmap effectively shows the variations in prediction results that arise from different view combinations [46]. It can be seen that the T+C+N view combination demonstrates superior performance compared to the other seven combinations. On this basis, MVHR-DP, which adds class dependencies on the T+N+C view, has achieved excellent results. MVHR-DP achieves significant results across AUC; the average value is 85.0. In general, the effectiveness of code multiview features learned from MVHR-DP can be improved by combining code multiview features.

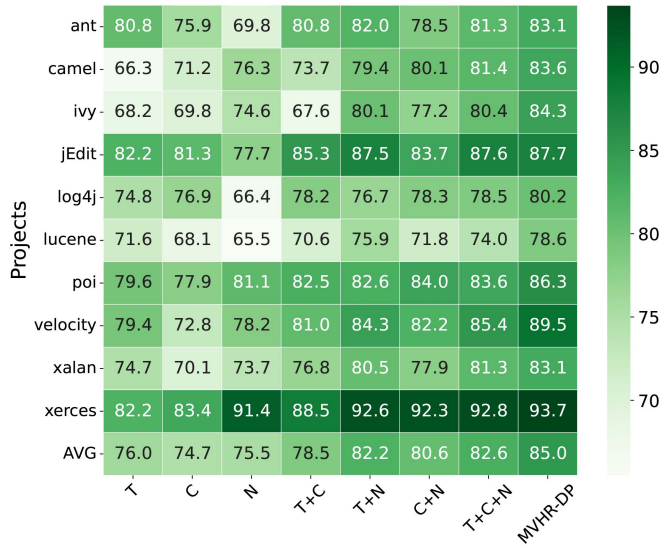
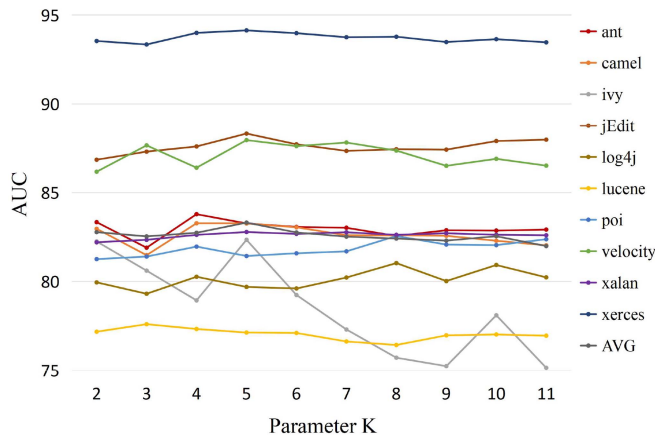


Fig. 11. AUC heatmap with different view combinations.

Fig. 12. AUC of MVHR-DP under different parameter K .

C. How Does Parameter K Used to Construct the Hyperedge Affect the Experiment?

This section discusses the K parameter setting of the number of neighbor nodes used to construct hyperedges in the MVHR-DP method to achieve the best results. Since we focus on the effect of K parameters on model performance, we only use K as a variable for this experiment. Other network parameters remain unchanged. We conduct WPDP experiments on 29 pairs of defect prediction tasks for ten projects, recording the average value of each project with the evaluation metric AUC. Fig. 12 shows the performance of MVHR-DP at different K s. We set K between 2 and 11. As shown in Fig. 12, when $K = 5$, the F-measure values of most WPDP tasks peak. In addition, for the remaining WPDP tasks, when K is set to 5, the F-measure value is not much different from the maximum value. Based on the abovementioned experimental results, the optimal K is about 5 in our experiments.

D. Threats to Validity

1) *Implementation of Compared Methods:* We implement some of the compared methods in the experiments (e.g., DTL-DP and GCN2defect) using the source code available online. For baseline methods that do not provide source code, we try to ensure implementation by strictly following the responding details in the original paper. Our implementation may not reflect all the details in the compared methods.

2) *Experimental Results Might not be Generalizable:* We conduct experiments using open-source software defect datasets, which vary in project scale and the number of defects, to facilitate the generalization of our method. However, we cannot guarantee that MVHR-DP will also achieve the same improvement on other software datasets. To reduce external validity, more software defect datasets need to be investigated in our future work.

3) *AUC, F-Measure, and MCC Might not be the Only Appropriate Indicators:* We selected AUC, F-measure, and MCC, which are commonly used in software defect prediction studies, as evaluation indicators. Different metrics may lead to different results. This article does not compare other performance metrics (e.g., G-mean and balanced accuracy) and more evaluation indicators that need to be used in the future.

4) *Parameter Selection of Network Does not Take all Options Into Account:* In our experiments, we tried to adjust the parameters of the network to get better prediction performance. However, it is impractical to evaluate all possible combinations of parameters. We evaluated several combinations of parameters within a specific range based on previous research experience. There may be a more appropriate combination of parameters for better predictive performance.

VIII. CONCLUSION

To overcome the challenges of underutilized code multiview information and the need to uncover multivariate association code features, this article proposes MVHR-DP to enhance the effectiveness of extracted code features for defect prediction. MVHR-DP fuses code multiview information into an adjustable-dimensional hypergraph, and a hypergraph neural network is constructed to mine the multiple-view and multivariate association information for improving the feature completeness of modeling data. Empirical study shows that the prediction model built with the MVHR-DP-generated features can achieve better AUC, F-measure, and MCC results in both WPDP, CVD, and CPDP tasks. In most cases, we believe that MVHR-DP can be a good option for reliability assurance teams to perform defect prediction tasks. The data, source code, and detailed experimental results supporting this study's findings are openly available on GitHub at <https://github.com/insoft-lab/MVHR-DP>.

Several problems remain to be investigated in future work. First, we will incorporate more software code data, both open-source and business projects, to further validate the performance of our method. Second, we will combine more code views to improve the effectiveness of MVHR-DP, i.e., the structure-based and code-visualization-based views.

REFERENCES

- [1] N. Li, M. Shepperd, and Y. Guo, "A systematic review of unsupervised learning techniques for software defect prediction," *Inf. Softw. Technol.*, vol. 122, 2020, Art. no. 106287.
- [2] S. Qiu, H. Huang, W. Jiang, F. Zhang, and W. Zhou, "Defect prediction via tree-based encoding with hybrid granularity for software sustainability," *IEEE Trans. Sustain. Comput.*, Feb. 2023, doi: 10.1109/TSUSC.2023.3248965. [Online]. Available: <https://ieeexplore.ieee.org/document/10052729>
- [3] X. Chen, D. Zhang, Y. Zhao, Z. Cui, and C. Ni, "Software defect number prediction: Unsupervised vs supervised methods," *Inf. Softw. Technol.*, vol. 106, pp. 161–181, 2019.
- [4] Z. Xu et al., "LDFR: Learning deep feature representation for software defect prediction," *J. Syst. Softw.*, vol. 158, 2019, Art. no. 110402.
- [5] T. Shippey, D. Bowes, and T. Hall, "Automatically identifying code features for software defect prediction: Using AST N-grams," *Inf. Softw. Technol.*, vol. 106, pp. 142–160, 2019.
- [6] H. Wang, W. Zhuang, and X. Zhang, "Software defect prediction based on gated hierarchical LSTMs," *IEEE Trans. Rel.*, vol. 70, no. 2, pp. 711–727, Jun. 2021.
- [7] J. Xu, F. Wang, and J. Ai, "Defect prediction with semantics and context features of codes based on graph representation learning," *IEEE Trans. Rel.*, vol. 70, no. 2, pp. 613–625, Jun. 2021.
- [8] J. Wang and Q. Wang, "Analyzing and predicting software integration bugs using network analysis on requirements dependency network," *Requirements Eng.*, vol. 21, pp. 161–184, 2016.
- [9] L. Gong, G. K. Rajbahadur, A. E. Hassan, and S. Jiang, "Revisiting the impact of dependency network metrics on software defect prediction," *IEEE Trans. Softw. Eng.*, vol. 48, no. 12, pp. 5030–5049, Dec. 2022.
- [10] Y. Qu et al., "node2defect: Using network embedding to improve software defect prediction," in *Proc. IEEE/ACM 33rd Int. Conf. Automated Softw. Eng.*, 2018, pp. 844–849.
- [11] Y. Qu et al., "Using k-core decomposition on class dependency networks to improve bug prediction model's practical performance," *IEEE Trans. Softw. Eng.*, vol. 47, no. 2, pp. 348–366, Feb. 2021.
- [12] C. Zeng, C. Y. Zhou, S. K. Lv, P. He, and J. Huang, "GCN2defect: Graph convolutional networks for SMOTETomek-based software defect prediction," in *Proc. 32nd Int. Symp. Softw. Rel. Eng.*, 2021, pp. 69–79.
- [13] A. Antelmi, G. Cordasco, M. Polato, V. Scarano, C. Spagnuolo, and D. Yang, "A survey on hypergraph representation learning," *ACM Comput. Surv.*, vol. 56, no. 1, pp. 1–38, 2023.
- [14] Y. Gao, Y. Feng, S. Ji, and R. Ji, "HGNN+: General hypergraph neural networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 3, pp. 3181–3199, Mar. 2023.
- [15] Z. Xu et al., "A comprehensive comparative study of clustering-based unsupervised defect prediction models," *J. Syst. Softw.*, vol. 172, 2021, Art. no. 110862.
- [16] C. Pornprasit and C. Tantithamthavorn, "DeepLineDP: Towards a deep learning approach for line-level defect prediction," *IEEE Trans. Softw. Eng.*, vol. 49, no. 1, pp. 84–98, Jan. 2023.
- [17] M. Kondo, D. M. German, O. Mizuno, and E.-H. Choi, "The impact of context metrics on just-in-time defect prediction," *Empirical Softw. Eng.*, vol. 25, no. 1, pp. 890–939, 2020.
- [18] Z. Zeng, Y. Zhang, H. Zhang, and L. Zhang, "Deep just-in-time defect prediction: How far are we?," in *Proc. 30th Int. Symp. Softw. Testing Anal.*, 2021, pp. 427–438.
- [19] S. Wang, T. Liu, and L. Tan, "Automatically learning semantic features for defect prediction," in *Proc. 38th Int. Conf. Softw. Eng.*, 2016, pp. 297–308.
- [20] J. Li, P. He, J. Zhu, and M. R. Lyu, "Software defect prediction via convolutional neural network," in *Proc. 17th Int. Conf. Softw. Qual., Rel. Secur.*, 2017, pp. 318–328.
- [21] H. K. Dam et al., "Lessons learned from using a deep tree-based model for software defect prediction in practice," in *Proc. 16th Int. Conf. Mining Softw. Repositories*, 2019, pp. 46–57.
- [22] A. V. Phan, M. Le Nguyen, Y. L. H. Nguyen, and L. T. Bui, "DGCNN: A convolutional neural network over large-scale labeled graphs," *Neural Netw.*, vol. 108, pp. 533–543, 2018.
- [23] J. Chen et al., "Software visualization and deep transfer learning for effective software defect prediction," in *Proc. 42nd Int. Conf. Softw. Eng.*, 2020, pp. 578–589.
- [24] X. Li, W. Li, Y. Zhang, and L. Zhang, "DeepFL: Integrating multiple fault diagnosis dimensions for deep fault localization," in *Proc. 28th Int. Symp. Softw. Testing Anal.*, 2019, pp. 169–180.
- [25] C. Zhou, P. He, C. Zeng, and J. Ma, "Software defect prediction with semantic and structural information of codes based on graph neural networks," *Inf. Softw. Technol.*, vol. 152, 2022, Art. no. 107057.
- [26] R. Kumar, A. Chaturvedi, and L. Kailasam, "An unsupervised software fault prediction approach using threshold derivation," *IEEE Trans. Rel.*, vol. 71, no. 2, pp. 911–932, Jun. 2022.
- [27] Z. Li, J. Niu, X.-Y. Jing, W. Yu, and C. Qi, "Cross-project defect prediction via landmark selection-based kernelized discriminant subspace alignment," *IEEE Trans. Rel.*, vol. 70, no. 3, pp. 996–1013, Sep. 2021.
- [28] S. Huang, Z. Kang, and Z. Xu, "Auto-weighted multi-view clustering via deep matrix decomposition," *Pattern Recognit.*, vol. 97, 2020, Art. no. 107015.
- [29] L. Duan, I. W. Tsang, and D. Xu, "Domain transfer multiple kernel learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 3, pp. 465–479, Mar. 2012.
- [30] Q. Dai, X.-M. Wu, J. Xiao, X. Shen, and D. Wang, "Graph transfer learning via adversarial domain adaptation with graph convolution," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 5, pp. 4908–4922, May 2023.
- [31] C. Liu, D. Yang, X. Xia, M. Yan, and X. Zhang, "A two-phase transfer learning model for cross-project defect prediction," *Inf. Softw. Technol.*, vol. 107, pp. 125–136, 2019.
- [32] X. Yu, J. Keung, Y. Xiao, S. Feng, F. Li, and H. Dai, "Predicting the precise number of software defects: Are we there yet?," *Inf. Softw. Technol.*, vol. 146, 2022, Art. no. 106847.
- [33] C. Ni, X. Chen, F. Wu, Y. Shen, and Q. Gu, "An empirical study on pareto based multi-objective feature selection for software defect prediction," *J. Syst. Softw.*, vol. 152, pp. 215–238, 2019.
- [34] W. Ma, L. Chen, Y. Yang, Y. Zhou, and B. Xu, "Empirical analysis of network measures for effort-aware fault-proneness prediction," *Inf. Softw. Technol.*, vol. 69, pp. 50–70, 2016.
- [35] J. Zhang, Y. Dong, Y. Wang, J. Tang, and M. Ding, "Prone: Fast and scalable network representation learning," in *Proc. Int. Joint Conf. Artif. Intell.*, 2019, vol. 19, pp. 4278–4284.
- [36] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*.
- [37] S. Feng, J. Keung, X. Yu, Y. Xiao, and M. Zhang, "Investigation on the stability of smote-based oversampling techniques in software defect prediction," *Inf. Softw. Technol.*, vol. 139, 2021, Art. no. 106662.
- [38] Y. Ganin et al., "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 2096–2030, 2016.
- [39] P. Stojanov, Z. Li, M. Gong, R. Cai, J. Carbonell, and K. Zhang, "Domain adaptation with invariant representation learning: What transformations to learn?," in *Proc. Int. Conf. Adv. Neural Inf. Process. Syst.*, 2021, vol. 34, pp. 24791–24803.
- [40] X. Yu et al., "Improving ranking-oriented defect prediction using a cost-sensitive ranking SVM," *IEEE Trans. Rel.*, vol. 69, no. 1, pp. 139–153, Mar. 2020.
- [41] S. Wang, T. Liu, J. Nam, and L. Tan, "Deep semantic feature learning for software defect prediction," *IEEE Trans. Softw. Eng.*, vol. 46, no. 12, pp. 1267–1293, Dec. 2020.
- [42] Z. Feng et al., "CodeBERT: A pre-trained model for programming and natural languages," 2020, *arXiv:2002.08155*.
- [43] X. Huang et al., "UDA-DP: Unsupervised domain adaptation for software defect prediction," in *Proc. IEEE Int. Conf. Softw. Anal., Evol. Reengineering*, 2023, pp. 308–318.
- [44] R. Moussa and F. Sarro, "On the use of evaluation measures for defect prediction studies," in *Proc. 31st Int. Symp. Softw. Testing Anal.*, 2022, pp. 101–113.
- [45] C. Tantithamthavorn, S. McIntosh, A. E. Hassan, and K. Matsumoto, "An empirical comparison of model validation techniques for defect prediction models," *IEEE Trans. Softw. Eng.*, vol. 43, no. 1, pp. 1–18, Jan. 2017.
- [46] X. Cheng, H. Wang, J. Hua, G. Xu, and Y. Sui, "DeepWukong: Statically detecting software vulnerabilities using deep graph neural network," *ACM Trans. Softw. Eng. Methodol.*, vol. 30, no. 3, pp. 1–33, 2021.

证书号第6139394号



发明专利证书

发明名称：一种基于改进多目标进化算法的无人机群路径规划方法

发明人：彭超达;黄海源;何才杰;齐龙;张文晖;张金鹏

专利号：ZL 2023 1 0405415.6

专利申请日：2023年04月17日

专利权人：华南农业大学

地址：510640 广东省广州市天河区五山路483号

授权公告日：2023年07月14日

授权公告号：CN 116126032 B

国家知识产权局依照中华人民共和国专利法进行审查，决定授予专利权，颁发发明专利证书并在专利登记簿上予以登记。专利权自授权公告之日起生效。专利权期限为二十年，自申请日起算。

专利证书记载专利权登记时的法律状况。专利权的转移、质押、无效、终止、恢复和专利权人的姓名或名称、国籍、地址变更等事项记载在专利登记簿上。



局长
申长雨

申长雨



证书号 第6139394号

专利权人应当依照专利法及其实实施细则规定缴纳年费。本专利的年费应当在每年04月17日前缴纳。未按照规定缴纳年费的，专利权自应当缴纳年费期满之日起终止。

申请日时本专利记载的申请人、发明人信息如下：

申请人：

华南农业大学

发明人：

彭超达;黄海源;何才杰;齐龙;张文晖;张金鹏



226300

南通高新技术产业开发区铂金时代 17 幢 317 室 南通启佑专利商标
代理事务所(普通合伙)
李晓星(0513-86125669)

发文日:

2023 年 11 月 11 日



申请号或专利号: 202311092374.6

发文序号: 2023111100040040

申请人或专利权人: 华南农业大学

发明创造名称: 基于改进约束多目标优化进化算法的 AGA-UAV 路径规划方法

发明专利申请进入实质审查阶段通知书

上述专利申请, 根据申请人提出的实质审查请求, 经审查, 符合专利法第 35 条及实施细则第 96 条的规定, 该专利申请进入实质审查阶段。

提示:

1. 根据专利法实施细则第 51 条第 1 款的规定, 发明专利申请人自收到本通知书之日起 3 个月内, 可以对发明专利申请主动提出修改。

2. 申请文件修改格式要求:

对权利要求修改的应当提交相应的权利要求替换项, 涉及权利要求引用关系时, 则需要将相应权项一起替换补正。如果申请人需要删除部分权项, 申请人应该提交整理后连续编号的部分权利要求书。

对说明书修改的应当提交相应的说明书替换段, 不得增加和删除段号, 仅只能对有修改部分段进行整段替换。如果要增加内容, 则只能增加在某一段中; 如果需要删除一个整段内容, 应该保留该段号, 并在此段号后注明: “此段删除” 字样。段号以国家知识产权局回传的或公布/授权公告的说明书段号为准。

对说明书附图修改的应当以图为单位提交相应的替换附图。

对说明书摘要文字部分修改的应当提交相应的替换页。对摘要附图修改的应当重新指定。

同时, 申请人应当在补正书或意见陈述书中标明修改涉及的权项、段号、图、页。

审 查 员: 自动审查

联系电话: 010-62356655

审查部门: 初审及流程管理部



210307
2022.10

纸件申请, 回函请寄: 100088 北京市海淀区蓟门桥西土城路 6 号 国家知识产权局专利局受理处收
电子申请, 应当通过专利业务办理系统以电子文件形式提交相关文件。除另有规定外, 以纸件等其他形式提交的文件视为未提交。



第十四届蓝桥杯大赛软件赛国赛比赛结果

姓名： 莫宇涵

证件号码： 441521200207256512

比赛科目： 第十四届软件赛C/C++程序设计大学B组

指导老师： 彭超达

比赛奖项： 二等奖



获奖查询

第十四届蓝桥杯大赛软件赛国赛比赛结果

姓名：冯雨泽

证件号码：340822200308080710

比赛科目：第十四届软件赛C/C++程序设计大学A组

指导老师：彭超达

比赛奖项：三等奖

蓝桥杯大赛

获奖证书

华南农业大学李洋阳：

荣获第十四届蓝桥杯全国软件和信息技术专业人才大赛全国总决赛C/C++程序设计大学B组优秀奖。

特发此证，以资鼓励。

证书编号：021450076

证件号码：441781200302286637

工业和信息化部
人才交流中心

蓝桥杯大赛组委会
组织委员会

2023年6月16日

获奖查询

第十四届蓝桥杯大赛软件类国赛比赛结果

姓名：李洋阳

证件号码：441781200302286637

比赛科目：第十四届软件类C/C++程序设计大学B组

指导老师：彭超达

比赛奖项：优秀奖



获奖查询

第十四届蓝桥杯大赛软件赛省赛比赛结果

姓名：冯雨泽

证件号码：340822200308080710

比赛科目：第十四届软件赛C/C++程序设计大学A组

指导老师：彭超达

比赛奖项：一等奖

是否进入决赛：是

蓝桥杯大赛

获奖证书

华南农业大学李洋阳：

荣获第十四届蓝桥杯全国软件和信息技术专业人才大赛广东赛区C/C++程序设计大学B组一等奖。

特发此证，以资鼓励。

证书编号：021411962

证件号码：441781200302286637

工业和信息化部
人才交流中心

蓝桥杯大赛组委会
组织委员会

2023年4月23日

获奖查询

第十四届蓝桥杯大赛软件类省赛比赛结果

姓名：李洋阳

证件号码：441781200302286637

比赛科目：第十四届软件类C/C++程序设计大学B组

指导老师：彭超达

比赛奖项：一等奖

是否进入决赛：是

蓝桥杯大赛

获奖证书

华南农业大学莫宇涵：

荣获第十四届蓝桥杯全国软件和信息技术专业人才大赛广东赛区C/C++程序设计大学B组一等奖。

特发此证，以资鼓励。

证书编号：021412030

证件号码：441521200207256512

工业和信息化部
人才交流中心

蓝桥杯大赛组委会
组织委员会

2023年4月23日

第十四届蓝桥杯大赛软件赛省赛比赛结果

姓名： 莫宇涵

证件号码： 441521200207256512

比赛科目： 第十四届软件赛C/C++程序设计大学B组

指导老师： 彭超达

比赛奖项： 一等奖

是否进入决赛： 是



第十四届蓝桥杯大赛软件赛省赛比赛结果

姓名： 萧瀚阳

证件号码： 440307200212162319

比赛科目： 第十四届软件赛C/C++程序设计大学B组

指导老师： 彭超达

比赛奖项： 三等奖

是否进入决赛： 否

2022年第二届中国高校大数据挑战赛

获奖证书

华南农业大学

温树文

指导老师：彭超达

在2022年第二届中国高校大数据挑战赛中荣获：

三等奖

特颁此证，以资鼓励。

2022年11月
证书编号：bdc220153





奖状

梁子佳，温义豪，马誉津，汤崇健：

你们的作品《基于自适应智能识别的校园安全出行系统》在
2022年全国青年科普创新实验暨作品大赛华南农业大学校赛中荣
获

优秀奖

指导老师：彭超达
特发此证，以资鼓励。



共青团华南农业大学委员会
2022年6月15日

华南农业大学大学生创新训练计划项目 立项证明

兹证明 数学与信息学院、软件学院 黄海源
(202125310408)主持，2023 年大学生创新训练计划项目
获得立项，项目名称：基于进化计算和深度学习的末端物流
无人机自主路径规划器研究，项目编号：202310564064，
项目级别：国家级，指导教师：彭超达。

华南农业大学学生工作部（研究生工作部）
创新创业学院
2023 年 6 月 20 日



中共华南农业大学数学与 信息学院 软件学院委员会 文件

数信软件党发〔2022〕5号



关于表彰数学与信息学院 软件学院 2022年度“十佳工作者”的决定

各系（中心、室）、各党支部、各教职工：

2022年，学院全体教职工坚持以习近平新时代中国特色社会主义思想为指导，学习贯彻党的二十大精神，坚持以立德树人为根本，围绕学院实现内涵式高质量发展和助力学校“双一流”建设，在党的建设、人才培养、学科建设、师资队伍、科学研究、社会服务、疫情防控等方面均取得了可喜的成绩。为表彰先进、树立典型，激发广大教职工的积极性、主动性和创造性，营造积极向上的学院文化氛围，决定对张伟峰等“十佳教学工作者”、张娜等“十佳科技工作者”、喻财铃等“十佳服务工作者”予以表彰。

希望受到表彰的教职工珍惜荣誉，坚持不忘初心、牢记使命

- 1 -

命，坚持谦虚谨慎、艰苦奋斗，坚持持之以恒、再接再厉，在落实立德树人根本任务和实现高水平科技自立自强以及优化治理服务体系等工作中取得新业绩。希望全体教职工向“十佳工作者”学习，不忘立德树人初心，牢记为党育人使命，深入学习贯彻党的二十大精神，深入学习贯彻习近平总书记关于教育的重要论述，积极躬行《新时代高校教师职业行为十项准则》，努力推动学院实现内涵式高质量发展，助力学校“双一流”建设。

附件：数学与信息学院 软件学院 2022 年度“十佳工作者”名单

中共数学与信息学院 软件学院委员会 数学与信息学院 软件学院

2022 年 12 月 26 日

附件

数学与信息学院 软件学院 2022 年度“十佳工作者”名单
(排名不分先后)

一、“十佳教学工作者”

张伟峰 朱艳科 林利云 杨志程 黄沛杰 殷建军
林旭东 王国华 黄文玲 严 颖

二、“十佳科技工作者”

张 娜 肖 莉 袁利国 郑佳悦 彭超达 王海燕
张广煜 崔金荣 邓 金 张连宽(科技推广)

三、“十佳服务工作者”

喻财铃 陈 思 徐海婵 张秋燕 邱 华 黄小虎
张 猜 周运华 余平祥 陈炜颖

公开方式：主动公开

数学与信息学院 软件学院党政办公室 2022 年 12 月 26 日印发

荣誉证书

彭超达 同志：

在 2022 年度 科技 工作中成绩突出，授予“十佳工作者”
称号。特发此证，以资鼓励。

华南农业大学数学与信息学院



2022 年 12 月