电子科技大学 UNIVERSITY OF ELECTRONIC SCIENCE AND TECHNOLOGY OF CHINA

博士学位论文

DOCTORAL DISSERTATION

(电子科技大学图标)

论文题目	多目标进化算法及其在矿用挖掘机
	提升机构减速器中的应用

学科专业	机械工程
学 号	201311080101
作者姓名	付国忠
指导教师	黄洪钟 教 授

分类号	密级	
UDC ^{注1}		
ODC		

学 位 论 文

多目标进化算法及其在矿用挖掘机

	提升机	几构减速器中	的应用	
		(题名和副题名)		
		付国忠		
		(作者姓名)		
指导教师		黄洪钟	教 授	
	<u> </u>	已子科技大学	成都	
•		(姓名、职称、单位名	3称)	
申请学位级是	剧博士	学科专业	机械工程	
提交论文日期	期	论文答辩日期		
学位授予单位	立和日期	电子科技大学	年 月	
答辩委员会	主席			
评阅人				

注 1: 注明《国际十进分类法 UDC》的类号。

Multi Objective Evolutionary Algorithm and Its Application in Gear Reducer of Mining Excavator Hoisting Mechanism

A Doctor Dissertation Submitted to University of Electronic Science and Technology of China

Discipline:	Mechanical Engineering	
Author:	Guo-Zhong Fu	
Supervisor:	Prof. Hong-Zhong Huang	
	School of Mechanical and Electrical	
School:	Engineering	

独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作 及取得的研究成果。据我所知,除了文中特别加以标注和致谢的地方外, 论文中不包含其他人已经发表或撰写过的研究成果,也不包含为获得 电子科技大学或其它教育机构的学位或证书而使用过的材料。与我一 同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明 并表示谢意。

作者签名:	日期:	年	月	日
-------	-----	---	---	---

论文使用授权

本学位论文作者完全了解电子科技大学有关保留、使用学位论文的规定,有权保留并向国家有关部门或机构送交论文的复印件和磁盘,允许论文被查阅和借阅。本人授权电子科技大学可以将学位论文的全部或部分内容编入有关数据库进行检索,可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。

(保密的学位论文在解密后应遵守此规定)

作者签名:	 导师签名:			
	日期:	年	月	日

摘要

现实世界中,许多实际问题如工程设计、投资优化、资源分配和路径规划等多为优化问题。起源于达尔文进化论,进化算法(Evolutionary Algorithm,EAs)遵循"适者生存,不适者淘汰"的基本准则,依靠群体寻优的方式,不需要借助优化问题的梯度信息,可以适用于求解复杂优化问题。同时,EAs 每次迭代生成一组优化解而非单一优化解,极大地提升了计算效率。多目标进化算法(Multi-Objective Evolutionary Algorithms,MOEAs)作为 EAs 的一个重要分支,受到广泛关注,并服务于科学研究与工程实际应用。近年来,分解多目标进化算法(Multi-Objective Evolutionary Algorithm Based on Decomposition,MOEA/D)作为应用最为广泛的MOEAs 基本框架,在求解某些问题时会出现性能退化,算法本身存在一定不足。鉴于此,本文以提升 MOEA/D 性能为目的,从决策空间和目标空间两个维度展开研究。本文的主要研究成果和创新点包括:

- (1) 针对不同 MOEAs 中不同变化算子在收敛性和多样性上的表现不同,提出了一种基于混合性能指标综合评价的变化算子自适应选择机制。根据混合性能指标评估当前变化算子生成子代在收敛性和多样性上的表现,自适应判断后续进化过程采用的变化算子,以期降低变化算子在进化过程中生成无效子代。同时,针对乌鸦算法在求解单目标优化问题时展现出良好的探索性能,将其与差分进化算子相结合,组成变化算子选择池。实验结果表明该方案能有效平衡 MOEA/D 在决策空间的探索与开发,进而平衡优化解在收敛性和多样性上的表现。
- (2)针对 MOEA/D 中固定均匀分布权值向量求解非凸、离散优化问题时获得的优化解分布不均匀,提出了一种自适应权值向量生成机制。进化过程中当前种群个体与当前权值向量协同进化,去除不可行权值向量,结合最优权值向量生成器,自动更新权值向量,进而使 MOEA/D 优化解均匀分布于真实帕累托最优前沿面。实验结果表明该方案能有效改善 MOEA/D 的多样性。
- (3)针对多数情况下决策者更倾向于获得帕累托最优前沿面上自己偏好的一组解而非整个真实帕累托最优前沿,本文提出一种基于满意度的交互式多目标优化算法。通过引入满意度函数表征决策者偏好信息,同时根据交互式偏好信息的方式,提升决策者对优化问题的认识,更好地指导进化过程向决策者偏好方向驶进。实验结果表明基于满意度的交互式优化方法能够有效探索真实帕累托前沿面的边界,同时与其他优秀算法对比也能保证一定的收敛性。
 - (4) 针对大型矿用挖掘机提升机构减速器特殊工况与设计需求,提出了一种

基于安全系数设计的多目标优化模型。结合前面提出的优化算法对该优化问题进行求解,计算结果表明所提出的算法能够有效解决这一实际优化问题。

关键词: 多目标进化算法,变化算子自适应选择,权值向量自适应调整,交互式满意度优化,减速器

ABSTRACT

Many real world problems such as engineering design, portfolio optimization, resource allocation, route planning, etc. are indeed of optimization criteria. Originated from the Darwinian Evolution Theory, evolutionary algorithms (EAs) comply with the principle of Survival of the Fittest. Derived from the population-based search mechanism, EAs are able to solve complicated optimization problem in absence of the gradient information. Meanwhile, a set of solutions rather than one single solution are obtained after each iteration, the efficiency of EAs outperform the convential algorithms. Multi Objective Evolutionary Algorithms (MOEAs), a key branch of EAs, has prospered over these years, and been successively applied to solve many scientific research and practical applications. Multiobjective Evolutionary Algorithm Based on Decomposition (MOEA/D), the seminal framework of MOEAs, degenerates in some occasions. In light of this, this dissertation is devoted to intensively extend and modify MOEA/D in two scenarios (decision spapce and objective space) to improve its performance. The major work and core contribution of this dissertation include:

- (1) To alleviate the difference performance in convergence and diversity derived from the selection of different operators, a hybrid indicator based adaptive operator selection mechanism is proposed. By collaboratively evaluating hybrid indicators of the solutions generated by temporary operator, it can decide which operator to select in the next generations so as to reduce unuseful iterations. Meanwhile, as CSA shows significant performance in addressing the exploration of search space, we integrated it with DE operator to formulate a new selection pool. Experiment results indicate that the selection mechanism is efficient in most cases to address the balance between exploration and exploiation in search space, so as to balance convergence and diversity.
- (2) To alleviate the nonuniformly distributed solutions generated by a fixed set of evenly distributed weight vectors in the presence of nonconvex and disconnected problems, an adaptive vector generation mechanism is proposed. A coevolution stragtrgy and a vector generator are synergistically cooperated to remedy the weight vectors. Optimal weight vectors are generated to replace the useless weight vectors to assure the optimal solutions distribute evenly. Experiment results indicate that this mechanism is efficient to improve the diversity of MOEA/D.

- (3) In light of the fact that the Decision Makers' are more likely to generate the preferred solutions rather than a whole approximation of the true Pareto Front. An interactive satisfactory theory is introduced to articulate the preference on different objectives. The preference of the decision maker is articulated by satisfactory function. By interactively adjusting the preference, a more in-depth understanding of the MOPs is obtained to make sure that the solutons are evolved towards the preferred region of the true Pareto front. Experiment results indicate that the satisfactory optimization based interactive articulation of preference is able to explore the bounded parts of the true Pareto front to some extent and maintain a good convergence compared to efficient algorithms.
- (4) A safety factor based multi-objective optimization model with respect to the twostage gear reducer of a heavy mining excavator is formulated in accordance with its working condition and design's demand. The algorithms proposed in the early sections are utilized to optimize this particular multi objective problem. The results indicate that the proposed algorithms are able to solve this particular optimization problem.

Keywords: multi-objective evolutionary algorithm, adaptive operator selection, adaptive weight adjustment, interactive satisfactory optimization, gear reducer

目 录

第一	草	绪 论		1
:	1.1	研究二	工作的背景与意义	1
	1.2	多目标	示优化问题概述	4
		1.2.1	多目标优化问题数学基础	4
		1.2.2	多目标优化解集性能评价指标	6
		1.2.3	多目标优化算法测试函数	10
	1.3	多目标	示进化算法	11
		1.3.1	多目标进化算法发展历程	11
		1.3.2	多目标进化算法研究现状	12
		1.3.3	减速器多目标优化研究现状	14
		_	综述总结与问题提出	_
	1.4	本文的	り结构安排	16
第二	章	基于混	合性能指标综合评价的变化算子自适应选择多目标进化算法	17
4	2.1	引言.		17
4	2.2	改进型	일乌鸦搜索变化算子	18
		2.2.1	差分进化变化算子	18
		2.2.2	乌鸦搜索变化算子	18
		2.2.3	乌鸦搜索变异因子	20
4	2.3	基于沿	昆合性能指标综合评价的变化算子自适应选择策略	21
		2.3.1	算法思路	21
		2.3.2	算法详解	21
4	2.4	实验证	设计	24
		2.4.1	测试函数集	24
		2.4.2	性能评价指标	25
		2.4.3	对比算法	25
		2.4.4	实验设置	26
2	2.5	实验约	吉果与分析	27
		2.5.1	Median Attainment Surface	27
		2.5.2	实验数值统计结果	29
		2.5.3	盒图	33

	2.5.4 结果分析	46
2.6	5 本章小结	46
第三章	基于自适应权值向量的分解多目标进化算法	47
3.1	1 引言	47
3.2	2 PICEA-w 权值向量更新机制	48
	3.2.1 PICEA-w 算法基本流程	48
	3.2.2 PICEA-w 算法存在的一些问题	50
	3.2.3 最优权值向量生成方案	51
3.3	3 基于自适应权值向量的分解多目标进化算法	51
	3.3.1 算法思路	51
	3.3.2 算法详解	51
3.4	4 实验设计	53
	3.4.1 测试函数集	53
	3.4.2 性能评价指标	53
	3.4.3 对比算法	53
	3.4.4 实验设置	55
3.5	5 实验结果与分析	55
	3.5.1 Median Attainment Surface	55
	3.5.2 实验数值统计结果	57
	3.5.3 盒图	61
	3.5.4 结果分析	
3.6	6 本章小结	75
第四章	基于满意度的交互式多目标优化算法	76
4.]	1 引言	76
4.2	2 满意度理论	
	4.2.1 满意度定义	
	4.2.2 多目标满意度优化	
4.3	3 基于满意度理论的交互式进化算法	
	4.3.1 二元关系	78
	4.3.2 权值优化	79
	4.3.3 搜索机制	80
4.4	4 实验设计	
	4.4.1 测试函数集	81

		4.4.2	性能评价指标	82
		4.4.3	对比算法	82
		4.4.4	实验设置	82
4	4.5	实验结	吉果与分析	82
		4.5.1	Median Attainment Surface	82
		4.5.2	实验数值统计结果	85
		4.5.3	盒图	88
		4.5.4	结果分析	93
4	4.6	本章小	卜结	93
第五	章	大型矿	用挖掘机提升机构减速器多目标优化设计	95
	5.1	引言		95
	5.2	大型矿	广用挖掘机提升机构减速器多目标优化模型	96
		5.2.1	提升机构减速器基本构成	97
		5.2.2	目标函数	97
		5.2.3	约束条件	102
	5.3	优化组	吉果	103
		5.3.1	MOEA/D-AW 计算结果	105
		5.3.2	与先进算法对比	108
;	5.4	本章小	卜结	.111
第六	章	全文总	结与展望	.112
(6.1	全文总	总结	.112
(6.2	后续コ	工作展望	.113
致	谢			.114
参考:	文南	状		.115
攻读 [·]	博士	上学位其	朝间参与的科研项目及取得的成果	130

主要符号及缩略语

SOPs 单目标优化问题(Single-Objective Optimization Problems)

MOPs 多目标优化问题(Multi-Objective Optimization Problems)

MaOPs 高维多目标优化问题(Many-Objective Optimization Problems)

EAs 进化算法(Evolutioanry Algorithms)

EMO 进化多目标优化(Evolutionary Multi-Objective Optimization)

MOEAs 多目标进化优化算法(Multi-Objective Evolutionary Algorithms)

MOEA/D 分解多目标优化算法(Multiobjective Evolutionary Algorithm Based

on Decomposition)

CSA 乌鸦算法(Crow Search Algorithm)

DE 差分进化(Differential Evolution)

MOEA/D-IHO 基于混合性能指标综合评价的自适应变化算子选择 MOEA/D

(Hybrid Indicator Based Adaptive Operator Selection MOEA/D)

MOEA/D-AW 基于自适应权值向量的 MOEA/D(Adaptive Weights Vector Based

MOEA/D)

第一章 绪 论

1.1 研究工作的背景与意义

起源于法国数学家拉格朗日求解约束条件下函数极值,最优化问题一直以来都是工程实践和科学研究中的一个重要问题^[1,2]。按照目标函数数量,最优化问题又可以分为单目标优化问题(Single-Objective Optimization Problems,SOPs)和多目标优化问题(Multi-Objective Optimization Problems,MOPs)。MOPs 与 SOPs 的区别主要体现在: 1)SOPs 的目标函数为 1,寻优过程中迭代解目标函数值之间的优劣关系为一个全序(Total Order)关系,可以通过比较大小的方式选择优势解,除去多峰问题,一般而言 SOPs 只有单一最优解; 2)MOPs 的目标函数大于 1,且目标函数之间相互冲突,寻优过程中迭代解目标函数值之间的优劣关系为一个偏序(Partial Order)关系,无法通过比较大小的方式选择优化解,因此 MOPs 的"优化解"为一组相互妥协的解的集合(即帕累托最优解集)^[3-16]。

国际上普遍将 1896 年意大利裔政治经济学家 Pareto 针对社会资源分配提出的 最优概念作为 MOP 的理论基础[11,17-24]。Pareto 结合 Edgeworth 提出的无差异曲线 (Indifference Curve)理论,指出在社会资源分配问题上,如果在不影响某一个体 收入的前提下无法提升其他个体的收入则称之为最优分配。而帕累托最优解的正 式概念于 1951 年由诺贝尔经济学奖获得者 Koopmans^[25]首次提出,并用来指导社 会生产与社会分配问题的分析。同年,Kuhn 和 Tucker^[26]结合 Koopmans^[25]提出的 有效向量概念,从数学角度制定了非线性规划问题的最优条件。1968年,Johnsen[27] 在其研究报告中系统地归纳了多目标决策模型,奠定了多目标优化学科蓬勃发展 的基础。多目标优化问题逐渐成为一门独立的学科并受到经济学者的广泛关注始 于 1970 年^[18,28,29], 其中 Zeleny^[28]于 1975 年出版了在京都召开的 International Meeting of The Institute of Management Sciences 大会中多准则决策(Multi Criteria Decision Making,MCDM)分会的 15 篇与多目标优化相关的文章,标志着多目标 优化问题在管理科学领域逐渐成为一个研究热点。从 Pareto 针对 MOP 首次提出帕 累托最优的概念到 MOP 在管理科学中的广泛应用,求解 MOP 也正式归纳为求解 帕累托最优解集的数学规划问题。扩展到自然科学领域,无论是科学探索研究还是 工程实际应用,大多数待解决的问题都包含多个相互冲突的优化目标。正因为 MOPs 广泛存在于工程设计、投资优化、资源分配和路径规划等各个领域[16,17,30-44], 因此开展多目标优化算法研究,具有十分重要的理论和工程实用价值。

一般而言, MOPs 的分析过程主要分为两个阶段: 首先, 求得一组帕累托优化

解集;其次,决策者根据"更高阶"的决策方法(或者偏好信息)从帕累托优化解 集中选择单一或一组最终优化解。一般而言,传统方法主要通过某种聚合机制,将 相互冲突的多个目标函数转化为单一"复合"目标函数,再通过解析算法求得相对 应的全局最优解,即一个帕累托优化解。应用最为广泛的算法包括线性加权求和 (Weighted Sum Method) [45]、 ε 约束(ε -Constraint Method)[46]、目标规划(Goal Programming) [47]、字典序(Lexicographic Ordering)和加权测度(Weighted Metric Method)[17]等算法。然而,这些经典算法在求解过程中每次只能获得一个优化解, 只有通过多次运行并且设置不同单一"复合"目标函数,才能获得足够多的帕累托 最优解。同时,由于每次迭代优化过程相互独立,迭代过程中的信息无法共享,导 致了算力的浪费。此外,每个算法都有其局限性,例如:线性加权求和法在求解非 凸(Non-Convex)问题时,往往无法收敛到帕累托优化解,并且多数情况下优化解 体现出的函数关系并不能与预设的权值向量相匹配,还需要重新设定新的权值向 量,加剧了算力的浪费。 ε 约束法虽然能够求解加权求和法无法解决的非凸问题, 然而不同的 ε 取值,会直接影响帕累托最优解的求解过程。目标规划法虽然通过设 定期望目标将 MOP 转化为易于理解的物理问题,但是同加权求和法与 ε 约束法一 样,求解过程同样需要借助决策者信息设定相应的期望目标顺序,而这些信息又无 法从物理层面上直观地进行解释,因此也限制了其在复杂优化问题上的应用场景。 加权测度法虽然通过设置不同的权值向量可以获得不同帕累托最优解,但是该方 法会获得一定数量的弱帕累托最优解,并且该方法也无法保证可以获得所有的帕 累托最优解。同时, 在处理非凸问题时, 理想点的选取也直接决定能否获得帕累托 最优解。

"进化"作为大自然馈赠给人类的礼物,我们接受并学习进化理论,将起源于达尔文进化论的"适者生存,不适者淘汰"基本机制与计算机技术相结合,进而衍生出了进化算法(Evolutionary Algorithm, EAs)。EAs 使计算机具备"进化"能力,不依靠梯度信息便能够解决复杂的优化问题。同时,由于 EAs 是一种基于群体的优化算法,在每次迭代过程中,可以并行获得一组优化解。鉴于此,EAs 自上世纪50 年代末由 Fraser 和 Bremermann 提出^[48]不久,Rosenberg^[49]便于 1967 年在其就读密歇根大学期间撰写的博士论文中采用一种初级的进化算法来解决 MOPs,尽管在他的论文中 MOP 被当成 SOP 进行分析,但是也可以作为进化算法用来解决MOPs 的开篇之作。然而 EAs 的并行计算机制受限于当时计算机运算能力,一直未能受到人们的重视。直到上世纪 80 年代,才由 Schaffer^[50]采用进化算法来解决MOPs,并归纳总结为矢量评估遗传算法(Vector Evaluated Genetic Algorithm,VEGA),同时 VEGA 也是 Holland^[51]提出的遗传算法(Genetic Algorithm,GA)首

次用来解决 MOPs。随着计算机计算能力的爆发式增长,多目标进化算法(Multi-Objective Evolutionary Algorithms,MOEAs)成为 EAs 领域的重要分支:进化多目 标优化(Evolutionary Multi-Objective Optimization,EMO)^[52]。针对 MOEAs 的研 究在过去近30年期间取得了长足的发展,《进化计算》(Evolutionary Computation, MIT Press)、《IEEE 进化计算汇刊》(IEEE Transactions on Evolutionary Computation)、 《IEEE 控制论汇刊》(IEEE Transactions on Cybernetics)等国际进化计算领域顶级 期刊,每年都会大篇幅出版与 MOEAs 相关的研究论文。其中,根据 Web of Science 统计,影响因子最高的《IEEE 进化计算汇刊》截至 2018 年 3 月他引次数排名前 10 的论文有 6 篇与多目标进化算法直接相关。其中,总排名第1 的是 Deb 等[53]提 出的基于精英保留策略的非支配排序遗传算法(Non-dominated Sorting Genetic Algorithm, NSGA-II),总排名第 3 的是 Zitzler 等[54]提出的基于聚类技术的增强帕 累托进化算法(Strength Pareto Evolutionary Algorithm, SPEA),总排名第 5 的是 Zhang 等[55]提出的分解多目标进化算法(Multiobjective Evolutionary Algorithm Based on Decomposition,MOEA/D),总排名第 7 的是 Liang 等^[56]提出的基于综合 学习的粒子群优化算法(Comprehensive Learning Particle Swarm Optimization, CLPSO),总排名第 8 的是 Zitzler 等[57]关于 MOP 性能评价指标的分析与综述文 章, 总排名第 9 的是 Coello 等[58]提出的基于搜索空间分区的多目标粒子群算法 (Multi-Objective Particle Swarm Optimization, MOPSO)。同时他引排名第 10 的是 Das 和 Suganthan^[59]关于差分进化(Defferential Evolutionary, DE)的综述性文章, 这篇综述着重讨论 DE 在多目标优化领域的应用。《进化计算》的他引次数排名前 10 的文章中也有 4 篇直接与 MOEAs 相关[60-63], 这里不再一一赘述。除此之外, EAs 领域顶级会议 IEEE Congress on Evolutionary Computation (CEC) 和 Genetic and Evolutionary Computation Conference (GECCO) 每年都设专场集中讨论 EMO 领域的最新研究成果,EMO领域知名学者也相继出版了关于MOEAs的专著[64-67]。

随着研究问题的深入,人们发现随着目标数量的增加(目标函数大于 3),常用的基于帕累托支配基本框架的 MOEAs,如 NSGA-II,Non-dominated Sorting Genetic Algorithm^[53]和 SPEA2,Improved Strength Pareto Evolutionary Algorithm^[68]的性能会急剧下降^[69-72]。鉴于此,基于松弛支配关系的 MOEAs^[63,73-75],基于性能指标的 MOEAs^[76-78]、基于分解的 MOEAs^[55,79,80]和基于偏好信息的 MOEAs^[81-84]相继涌现,并在处理 MaOPs 时体现出优于基于帕累托支配基本框架 MOEAs 的性能。然而,这几类方法目前还存在着一些不足,依然处在探索改进阶段。如何更好地平衡各类 MOEAs 优化解在目标空间中多样性与收敛性之间的关系是目前许多学者着重探讨的问题。此外,目前多数学者着重探讨优化解在目标空间中的表现(多样

性与收敛性的平衡),很少就搜索空间中的探索与开发展开深入的讨论。因此,本文着重探讨 MOEAs 在提升多样性与收敛性之间平衡的同时,进一步探讨 MOEAs 在搜索空间中探索与开发的平衡对 MOEAs 性能的影响。同时,本文在国家 863 项目 "75 立方米大型露天矿用挖掘机研制"的子课题"大型矿用挖掘机提升机构动态可靠性分析与优化设计关键技术研究"的资助下开展大型矿用挖掘机提升机构减速器多目标优化设计关键技术研究,探讨如何更为有效地将实际应用问题与MOEAs 相结合,以更好地求解工程实际应用中涌现的多目标优化问题。

1.2 多目标优化问题概述

1.2.1 多目标优化问题数学基础

多目标优化问题作为典型的偏序问题,其优化目标由多个相互冲突的目标组成,不失一般性,本文以最小化作为优化目标,其函数表达如下:

$$\min_{\mathbf{x} \in D} \mathbf{f}(\mathbf{x}) = \min \left\{ f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_M(\mathbf{x}) \right\}$$
s.t.
$$\mathbf{x} = \left\{ x_1, x_1, \dots, x_n \right\} \in D \in \mathbb{R}^n$$
(1-1)

式中,D为n维决策空间中的可行域, \mathbf{x} 为D中一个决策向量,M为目标函数数量, $f(\Theta)$ 为决策向量到M维目标空间 Ω 的映射关系表达式, $f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_M(\mathbf{x})$ 为相互冲突的目标函数。

MOPs 中不存在一个决策向量 \mathbf{x}_i 能够使 $f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_M(\mathbf{x})$ 同时在可行域内取最小值,"最优"概念在 MOPs 中发生了转变,研究学者普遍采用 Pareto 提出的帕累托最优(Pareto Optimality)作为求解 MOP 中的基本准则^[4,17,85,86],具体如下。

定义 1: 帕累托支配 (Pareto Dominate)

假设在可行域D中,任意两个决策向量 $\mathbf{u} = \{u_1, u_2, ..., u_n\}^T$ 和 $\mathbf{v} = \{v_1, v_2, ..., v_n\}^T$ 同时满足如下条件:

- 1) $\forall i = \{1, 2, ..., M\}, f_i(\mathbf{u}) \leq f_i(\mathbf{v}),$
- 2) $\exists j = \{1, 2, \dots, M\}$, $f_j(\mathbf{u}) < f_j(\mathbf{v})$,

则u帕累托支配v,简称u支配v,记为u~v。

定义 2: 帕累托最优解(Pareto Optimal Solution)

在可行域D中,当且仅当没有其他决策向量 \mathbf{u} 支配 \mathbf{u}^* 时,则称 \mathbf{u}^* 为帕累托最优解,也可称之为非劣解(Non-Inferior Solution)或非支配解(Non-Dominated Solution)。

定义 3: 弱帕累托最优解(Weakly Pareto Optimal Solution)

n维决策向量 $\mathbf{u} = \{u_1, u_2, \dots, u_n\}^T$ 被称为弱帕累托最优解,当且仅当不存在一个

n 维决策向量 $\mathbf{v} = \{v_1, v_2, ..., v_n\}^T$ 满足对于所有 $j = \{1, 2, ..., M\}$, $f_j(\mathbf{v}) < f_j(\mathbf{u})$ 。所有帕累托最优解都是弱帕累托最优解,而弱帕累托最优解不一定是帕累托最优解。

定义 4: 帕累托最优解集(Pareto Optimal Set)

所有帕累托最优解组成的集合称为帕累托最优解集,记为PS。

定义 5: 帕累托最优前沿 (Pareto Optimal Frontier)

帕累托最优解集映射到目标空间组成的集合称为帕累托最优前沿,记为*PF*。结合上述定义,MOP 归纳总结为在决策空间可行域内通过特定迭代机制,择优选取一组非支配决策向量,即帕累托最优解集,作为该 MOP 的优化解。

如图 1-1 所示,决策向量 \mathbf{x}_a 、 \mathbf{x}_1 、 \mathbf{x}_2 和 \mathbf{x}_b 按照定义 1 和定义 2 称之为可行域 D 中的帕累托最优解,其组成的集合按照定义 3 称之为帕累托最优解集(PS)。 \mathbf{z}_a 、 \mathbf{z}_1 、 \mathbf{z}_2 和 \mathbf{z}_b 分别为决策向量 \mathbf{x}_a 、 \mathbf{x}_1 、 \mathbf{x}_2 和 \mathbf{x}_b 通过目标函数 $f(\Theta)$ 映射在目标空间 Ω 中的目标向量,其组成的集合按照定义 5 称之为帕累托最优前沿 (PF)。决策向量 \mathbf{x}_3 之所以不是帕累托最优解,因为其在目标空间 Ω 中对应的目标向量 \mathbf{z}_3 在 f_1 和 f_2 两个目标维度都要大于目标向量 \mathbf{z}_a 、 \mathbf{z}_1 、 \mathbf{z}_2 和 \mathbf{z}_b 相对应的值,按照定义 1 决策向量 \mathbf{x}_3 分别被决策向量 \mathbf{x}_a 、 \mathbf{x}_1 、 \mathbf{x}_2 和 \mathbf{x}_b 支配,记为 \mathbf{x}_a \prec \mathbf{x}_3 , \mathbf{x}_1 \prec \mathbf{x}_3 , \mathbf{x}_2 \prec \mathbf{x}_3 , \mathbf{x}_b \prec \mathbf{x}_a , \mathbf{x}_a \prec \mathbf{x}_a \prec

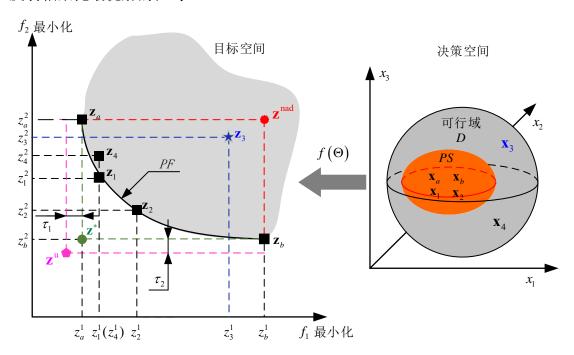


图 1-1 帕累托最优概念示意图

除去上述定义外,还有几个特殊解(理想点、乌托邦点和最差点)在多目标优化算法中经常被用来归一化不同量纲的目标函数^[17,87],现对其定义如下。

定义 6: 理想点 (Ideal Objective Vector)

对每个 $i = \{1, 2, ..., M\}$,由可行域D中所有决策向量 \mathbf{x} 映射到目标空间 Ω 对应的目标函数 $f_i(\mathbf{x})$ 的下确界 z_i^* 组成的向量称为理想点 \mathbf{z}^* ,记为 $\mathbf{z}^* = \{z_1^*, z_2^*, ..., z_m^*\}$ 。

定义 7: 乌托邦点 (Utopian Objective Vector)

某些算法需要借助一个在每个目标维度 $i = \{1, 2, ..., m\}$ 都优于理想点的特殊点来指导其优化过程,因此引入了"乌托邦点"。其定义如下: 对每个 $i = \{1, 2, ..., M\}$,由 \mathbf{z}^* 每个元素 \mathbf{z}^*_i 减去一个足够小的值 $\boldsymbol{\tau}_i$ 称为乌托邦点,记为 $\mathbf{z}^*_i = \mathbf{z}^*_i - \boldsymbol{\tau}_i$ 。

定义 8: 最差点 (Nadir Objective Vector)

对每个 $i = \{1, 2, ..., M\}$,由可行域D中所有决策向量 \mathbf{x} 映射到目标空间 Ω 对应的目标函数 $f_i(\mathbf{x})$ 的上确界组 z_i^{nad} 成的向量称为最差点 $\mathbf{z}^{\mathrm{nad}}$,记为 $\mathbf{z}^{\mathrm{nad}} = \{z_1^{\mathrm{nad}}, z_2^{\mathrm{nad}}, ..., z_m^{\mathrm{nad}}\}$ 。

如图 1-1 中所示 \mathbf{z}^* 、 \mathbf{z}^* 和 \mathbf{z}^{nad} 分别对应为理想点、乌托邦点和最差点。一般通过 $f_i^{\text{norm}}(\mathbf{x}) = \frac{f_i(\mathbf{x}) - z_i^*}{z_i^{\text{nad}} - z_i^*}$, i = 1, 2, ..., M 对目标函数进行归一化处理,以达到统一量纲的目的。

1.2.2 多目标优化解集性能评价指标

MOPs 的求解本质上是获得一组能够代表真实帕累托最优解集的近似优化解集。因此,评价近似解集的优劣要从两个维度展开^[88]: 1)收敛性,评价近似解集与真实帕累托最优前沿之间的逼近程度; 2)多样性,评价近似解集在真实帕累托最优前沿面上的分布程度,如图 1-2 所示。"收敛性"可以评定所得解集是否为真实帕累托最优解,而"多样性"则可以评定所得解集能否从广度和均匀度上近似代表真实帕累托最优前沿。

根据评价方式不同,多目标优化解集性能评价指标又可以分为定性指标与定量指标。

(1) 定性指标

定性指标通过直观地观察综合判定优化解集在收敛性与多样性上的优劣。其中最为成熟的指标为 Attainment Surface^[89]。该方法主要通过将离散优化解集连接成一个线或面的形式,表征该组优化解在性能上的优劣。同时,由于 MOEAs 从本质上讲为基于随机搜索的算法,每次获得的优化解集不尽相同,因此在 EMO 领域普遍采用 Median Attainment Surface^[89]对优化结果进行定性表达,如图 1-3 所示。然而,这类指标受限于图像显示技术,在目标函数大于 3 的情况下,很难直观地进

行表征。同时,许多算法在性能上差异不是很大,很难通过观察评价其优劣程度。

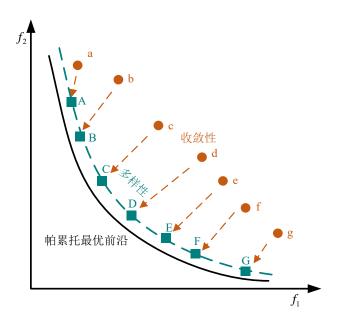


图 1-2 优化解集收敛性与多样性示意图

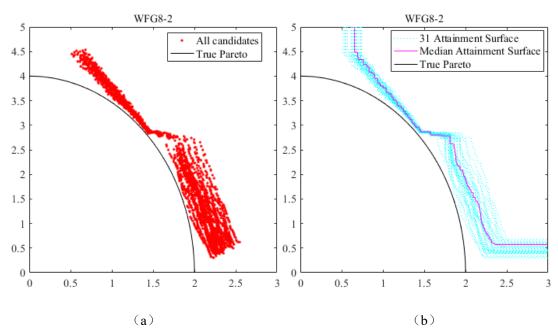


图 1-3 帕累托最优解集与 Attainment Surface。(a) 31 个帕累托最优解;(b) 31 个 Attainment Surface 以及 Median Attainment Surface 示意图

(2) 定量指标

优化解集在收敛性与多样性上的表现可以通过定量计算来进行表征^[57,90],其中可以同时评价收敛性与多样性的指标包括:超体积^[91](Hypervolume,HV)和 逆世代距离^[92](Inverted Generational Distance,IGD)。单独评价收敛性的指标包

括: 世代距离^[93](Generational Distance,GD)和集合覆盖率^[57](Set Coverage,C)。单独评价多样性的指标包括: 纯多样性^[94](Pure Diversity,PD)和最大广度 ^[54](Maximum Spread,MS)。

1)超体积(HV)

HV 指标用来计算在 M 维目标空间中近似解集 A 中每一点 z 与参考点围成的超体积的合集,如图 1-4(a)所示。图 1-4(a)为 HV 在 2 维空间中的表征,图中阴影部分即为超体积。通常而言,参考点可以选取为最差点 z^{nad} 或者宽松设置的 z^{nad} ,然而有关学者指出参考点较 z^{nad} 大些更为合适 z^{nad} 的函数表达如下:

$$HV(A) = \bigcup_{\mathbf{x} \in A} vol(\mathbf{z}) \tag{1-2}$$

式中, $vol(\cdot)$ 为**z**与参考点围成的超体积。HV从一定程度上可以同时评价解集的收敛性与多样性,并且单调递增。HV 值越大,表征相对应解集的性能更佳,如图 1-4(b)所示。图 1-4(b)中解集 $\{A,B,C,D,E\}$ 的HV 值大于解集 $\{a,b,c,d,e\}$,而直观上也可以判定 $\{A,B,C,D,E\}$ 整体性能优于 $\{a,b,c,d,e\}$ 。

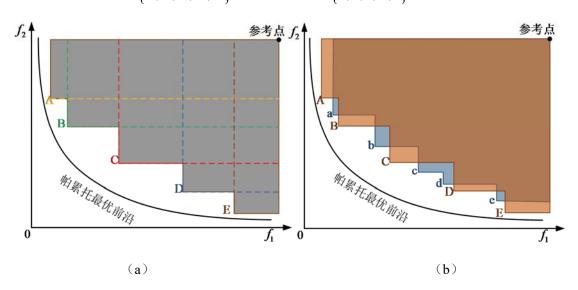


图 1-4 超体积性能评价指标计算示意和对比示意。(a) 计算示意图:(b) 对比示意图

2) 逆世代距离 (IGD)

设 P^* 为真实帕累托最优前沿上一组均匀分布的点。IGD 指标主要计算 P^* 中每一点 p^* 与近似解集A中最近点的距离的均值。IGD 的函数表达如下:

$$IGD(A) = \frac{1}{\left|P^*\right|} \sqrt{\sum_{\mathbf{p}^* \in P^*} \min_{\mathbf{z} \in A} d\left(\mathbf{p}^*, \mathbf{z}\right)^2}$$
 (1-3)

式中, \mathbf{z} 为近似解集 A 中任意一点, $d(\bullet)$ 为 \mathbf{p}^* 与 \mathbf{z} 之间的欧氏距离, $\left|P^*\right|$ 为 P^* 中点的个数。当 $\left|P^*\right|$ 值足够大,以至于 P^* 能够表征整个真实帕累托最优前沿,IGD 从某

种程度上也能够同时表征解集的收敛性与多样性。IGD 值越小,表示解集 A 与真实帕累托最优前沿足够接近,并且 P^* 中每个点都能够在 A 中找到对应的点加以表征。

3) 世代距离(Generational Distance, GD)指标

GD 指标计算近似解集 A 中每一点到真实帕累托最优前沿最近点距离的均值。 GD 的函数表达如下:

$$GD(A) = \frac{1}{|A|} \sqrt{\sum_{\mathbf{z} \in A} \min_{\mathbf{p} \in PF} d(\mathbf{z}, \mathbf{p})^{2}}$$
 (1-4)

式中, \mathbf{z} 为近似解集 \mathbf{A} 中任意一点, $\mathbf{d}(\bullet)$ 为 \mathbf{p} 与 \mathbf{z} 之间的欧氏距离, $|\mathbf{A}|$ 为 \mathbf{A} 中点的个数。 \mathbf{GD} 只能用来评价解集的收敛性, \mathbf{GD} 值越小,表示 \mathbf{A} 的收敛性越好。

4) 集合覆盖率 (C)

设A和B为两个近似解集。C指标不同于其他指标,为二元评价指标,C(A,B)主要用来计算解集B中被解集A支配的比例。C(A,B)的函数表达如下:

$$C(A,B) = \frac{\left| \left\{ \mathbf{y} \in B \middle| \exists \mathbf{x} \in A : \mathbf{x} \prec \mathbf{y} \right\} \right|}{|B|}$$
 (1-5)

如果C(A,B)大于C(B,A),那么在收敛性上解集A优于解集B。

5) 纯多样性 (PD)

PD指标计算近似解集A中点之间累积差异性,其中点之间的差异性通过计算两点之间的p<1的闵可夫斯基距离进行度量。PD的函数表达如下:

$$PD = \max_{\mathbf{z}_{i} \in A} \left\{ PD(\mathbf{z}_{i}) + d(\mathbf{z}_{i}, A \setminus \{\mathbf{z}_{i}\}) \right\}$$
 (1-6)

式中, $d(\mathbf{z}, A \setminus \{\mathbf{z}\})$ 为近似解集A中任意一点 \mathbf{z}_i 与A中距离其最近的点 $\mathbf{z}_j \in A \setminus \{\mathbf{z}_i\}$ 之间的p < 1的闵可夫斯基距离,其函数表达如下:

$$d(\mathbf{z}_{i},\mathbf{z}_{j}) = \left(\sum_{k}^{m} (\mathbf{z}_{i}^{k} - \mathbf{z}_{j}^{k})^{p}\right)^{1/p}$$
(1-7)

式中,根据PD的设计者建议,p值取为 $0.1^{[94]}$ 。

6) 最大广度 (MS)

MS 指标计算解集A在每个目标维度上极值点组成的超长方体对角线长度,以表征解集A在目标空间中能代表的最大广度。MS 的函数表达如下:

$$MS = \sqrt{\frac{1}{m} \sum_{i=1}^{m} \left(\frac{\max_{k=1}^{|A|} f_{i}^{k} - \min_{k=1}^{|A|} f_{i}^{k}}{f_{i}^{\max} - f_{i}^{\min}} \right)^{2}}$$
 (1-8)

式中, f_i^{max} 和 f_i^{min} 分别为解集A中所有解在对应目标函数上的最大值和最小值。MS 值越大,表示解集A 的广度越好。

上述 6 个性能指标中,只有 GD 和 IGD 的计算需要借助真实帕累托最优前沿,而 HV、 C 和 PD 的计算不需要借助真实帕累托最优前沿。此外 HV 和 MS 的计算还需要借助参考点信息,而 C 和 PD 的计算不需要借助任何其他信息。

1.2.3 多目标优化算法测试函数

测试函数通过人为设置障碍使优化解难以收敛到真实帕累托最优前沿或使优化解无法满足分布性上的要求,因此经常作为比较不同算法优劣的测试问题。同时,由于现实问题复杂多变,单靠几个单独的测试函数很难满足问题通用性上的要求,因此设计一个合理的测试函数集便显得尤为重要。

2000年,Zitler^[60]等提出的 ZDT 测试函数集是 EMO 领域首个综合性的测试函数集,涵盖了 6 种不同特性的 2 目标优化问题,并且决策变量的个数可以向任意规模进行扩展。其中,ZDT5 的决策变量为二进制,而其他 5 个函数的决策变量为连续型变量。此外,ZDT3 为离散凸函数,ZDT4 的帕累托最优前沿面具有 21°个局部最优前沿,因此可以用来测试算法处理多模态的能力,ZDT2 与 ZDT6 为凹函数,其中 ZDT6 的帕累托最优前沿面不均匀,集中分布在目标空间中右下角,可以很好地测试算法在多样性上的表现。

鉴于 ZDT 测试函数集只能测试算法在 2 目标优化问题上的性能,Deb 等[88]于 2002 年提出的 DTLZ 测试函数集很好地解决了这一问题,并很大程度上推动了 MaOPs 的发展。DTLZ 测试函数集包括 9 个测试函数,其中 7 个无约束优化问题 (DTLZ1 至 DTLZ7) 和 2 个约束优化问题 (DTLZ8 至 DTLZ9)。本文主要探讨无约束优化问题,因此只对 DTLZ1 至 DTLZ7 展开描述。DTLZ 测试函数集的决策变量由位置决策向量和距离决策向量两部分组成,通过 3 种距离函数、2 种形状函数和 2 种转换函数将决策变量映射到目标空间,以分别控制帕累托最优前沿面的形状和收敛情况。DTLZ1 采用的形状函数为简单的决策变量相加,因此其形状是一个平面; DTLZ2-DTLZ4 的形状为球面; DTLZ5 和 DTLZ6 由于在 DTLZ2 至 DTLZ4 的基础上加入了转换函数,因此其形状最终退化成曲线; DTLZ7 的形状区别于其他 DTLZ 测试问题,其形状为不连续和多峰的。

然而 ZDT 测试函数集和 DTLZ 测试函数集中大部分测试函数都缺少连续空间上欺骗性 (Deceptive)的要求,并且这两种测试函数集的决策变量可分离,鉴于此 Huband 等^[97,98]提出带有欺骗函数的 WFG 测试函数集,同时保留了不可分和退化的特性。而本文也主要以 WFG 测试函数集为对比函数的基准测试函数,详细描述

在后续对比实验中展开讨论。

1.3 多目标进化算法

1.3.1 多目标进化算法发展历程

多目标进化算法(Multi-Objective Evolutionary Algorithms,MOEAs)正式进入 我们的视线要追溯到 1985 年 Goldberg^[99]开创性的工作,Goldberg 开启了进化算法 在多目标优化领域的广泛研究与探索。

在此之后, EMO 的发展从时间上进行划分可以分为 3 个阶段[86,100]。

(1) 起源阶段

主要以 1.1 节中提及的 Rosenberg 于 1967 年在密西根大学期间的博士论文和 Schaffer 于 1984 年提出的 VEGA 为主,此外 Goldberg^[99]于 1985 年首次将帕累托最优概念与进化算法相结合,引入非支配分层准则,对种群中所有个体进行分层排序。首先,选取种群中不被其他个体支配的非支配个体并赋以最高排序 R_1 ,在剩下的个体中重新选取新的非支配个体赋以次高排序 R_2 ,依此类推,直至种群中所有个体都被赋以相对应的排序 R_i ,i=1,3,...,r,并与相同排序的个体组成相对应的非支配层。同时,引入小生境技术(Niching Technique)以保证种群在进化过程中得以避免收敛到局部最优。Goldberg 开创性的工作,也为后续 MOEAs 的研究提供了基本思路。

(2) 初级阶段

主要以 1993 年 Fonseca 和 Fleming^[101]提出的多目标遗传算法(Multiple Objective Genetic Algorithm,MOGA),1994 年 Srinivas 和 Deb^[102]提出的非支配排序遗传算法(Nondominated Sorting Genetic Algorithm,NSGA),和同年 Horn 等^[103]于第一届 CEC 大会上(IEEE Conference on Evolutionary Computation)提出小生境遗传算法(Niched-Pareto Genetic Algorithm,NPGA)为主。

MOGA 作为较为著名的 MOEA,提出了一种个体排序的方法,首先将群体中所有非支配个体的适应度设为 1,其他个体的适应度相继设为支配他的个体数组总和,最后通过 Goldberg^[99]提出的适应度共享机制从适应度相同的个体中选择相应个体执行后续操作。NSGA 借鉴了 Goldberg 的分层排序的思想,但是其在每次迭代过程中需要对群体中所有个体进行帕累托排序,大大降低了算法的计算效率。NPGA 通过随机选取两个个体 \mathbf{x}_i , \mathbf{x}_j 并与整个群体中大约 10%的个体 \mathbf{s}_i 战力比,如果 \mathbf{x}_i 被 \mathbf{s}_i 也的个体支配,而 \mathbf{x}_j 不是,那么在本次竞争中 \mathbf{x}_j 获胜,而如果出现平局(\mathbf{x}_i 和 \mathbf{x}_j 全被 \mathbf{s}_i 也的个体所支配,或者不被支配),胜负则通过适应度共享以保证多样性。

虽然这一阶段的算法采用的思路都较为简单,但是从理论层面上为 MOEA 设定了一个基本框架,即:首先,通过在决策空间随机生成一定数量 N 的样本(群体);然后,按照一定的"配对选择"机制从群体中选取一定数量的样本(父代)执行变化算子以生成新的样本点(子代);最后,通过"环境选择",从父代与子代中择取适应度高的样本(群体)组成新的群体,进行下一步迭代优化。值得提出的是,作为最早应用在 MOEA 中并且目前广泛应用的变化算子主要为 GA 的"交叉"算子和"变异"算子。随着研究的不断深入,相继出现了其他执行变化操作的算法,具有代表性的包括诸如模拟退火^[104](Simulated Annealing,SA)、蚁群算法^[105](Ant Colony Optimization,ACO)、粒子群算法^[106](Particle Swarm Optimization,PSO)、差分进化^[107](Differential Evolutionary,DE)、人工免疫系统^[108](Immune System)和分布式估计^[109](Estimation of Distribution Algorithm,EDA)。其中,DE 由于其优异的性能正被广泛应用于 MOEAs,并有逐渐替代 GA 的趋势^[110-118]。其中 MOEA/D 的变体 MOEA/D-DRA 采用 DE 作为变化算子取得了 2009 年 CEC 会议上求解无约束多目标优化问题的冠军^[111]。

(3) 成熟发展阶段

这一阶段的开始以 1998 年 Zitzler 等^[91]提出的增强帕累托进化算法(Strength Pareto Evolutionary Algorithm,SPEA)为代表。具有代表性的成果包括 Zitzler 等^[68] 提出的基于聚类技术的增强帕累托进化算法 2 代(SPEA2),Deb 等^[53]提出的基于精英策略的非支配排序遗传算法(Non-dominated Sorting in Genetic Algorithm,NSGA-II)和 Knowles 和 Corne^[61]提出的帕累托档案进化策略(Pareto Archive Evolution Strategy,PAES)。这些标志性的算法都是基于帕累托支配的基本框架展开讨论的。

SPEA 通过引入外部档案,记录当前种群搜索到的最优解集,利用提出的聚类技术控制外部档案中优化解的综述不超过档案最大数。SPEA2 在 SPEA 的基础上,提出了一种新型适应度分配方案,同时引入邻域密度概念指导整个搜索过程,最后提出一种能够保留边界信息的档案删减方法。NSGA-II 在 NSGA 的基础上,通过快速非劣排序法提升了算法效率,同时引入拥挤度对比,提升了算法在多样性上的表现,同时通过父代与子代共同对比的方式实现了精英保留策略。PAES 采用(1+1)的进化策略生成子代,并与历史档案进行对比,保留较为优秀的个体。其中 PAES 通过引入自适应目标空间分格的方式保证种群的多样性。

1.3.2 多目标进化算法研究现状

近些年,随着 EMO 领域许多学者研究发现基于帕累托支配的基本框架在解决

高维多目标优化问题^[21](Many Objective Optimization Problems,MaOPs)时,整个种群中非支配解的比例急剧增加,以至于几乎所有解之间都是非支配关系。因此,基于帕累托支配关系的环境选择机制将无法从群体中择取有效的父代以生成子代,整个群体的进化过程很大程度上依赖于初始种群建立与变化算子的效果,整个收敛速度会变慢,在面对一些复杂问题时甚至会出现收敛停滞的现象^[119]。因此,采用不同框架的 MOEAs 不断被提出^[11,30,100]。其中最具代表性的主要包括:

(1) 基于松弛支配关系的 MOEA 框架(Relaxed Dominance Based MOEA)

基于松弛支配关系的 MOEA 框架主要通过修改解集的目标函数值,增加群体中支配解的比例,以降低传统支配处理高维多目标优化问题的难度。常见的松弛支配关系包括,Ikeda 等 $^{[120]}$ 提出的 $_{\alpha}$ 支配和 Laumanns 等 $^{[63]}$ 提出的 $_{\varepsilon}$ 支配。

 α 支配关系的定义为: 如果解x在某一目标上略差于解y,然而在其他目标上明显优于解y,那么解x可称为 α 支配解y,目前尚未有 α 基于支配关系的算法用来解决高维多目标优化问题。

而 ε 支配关系由于其定义简单,直接将传统的支配关系扩到($1+\varepsilon$)倍,因此被广泛应用于 MOEA 中,并衍生出了基于相似支配关系思想的算法。代表性的有Deb 等 $^{[121]}$ 提出的 ε 多目标进化算法(ε -Multi Objective Evolutionary Algorithm, ε -MOEA),Sato 等提出的解支配区域控制方法(Control the Dominance Area of Solutions,CDAS) $^{[122]}$ 以及近期 Yang 等 $^{[116]}$ 提出的基于动态网格的进化算法(Grid Based Evolutionary Algorithm,GrEA)。大量实验结果显示,采用动态网格的 GrEA 在处理大部分测试问题时都要明显优于 ε -MOEA。因此 GrEA 也成为目前 EMO 领域较为知名的算法 $^{[21]}$,并用于新设计算法的比较算法 $^{[123-125]}$ 。

(2) 基于偏好的 MOEA 框架(Preference Based MOEA)

基于偏好的 MOEA 框架通过引入决策者的偏好信息以更好地指导 MOEA 向偏好方向进化。这一框架引入主要源于基于群体搜索概念的 MOEA 往往会获得一组庞大的非支配解集,而决策者最终仅仅需要从获得自己偏好的部分解,因此需要借助决策者的偏好信息以指导搜索向帕累托最优前沿上自己偏好的一部分行进。而根据偏好信息在优化过程中参与形式,Miettinen^[17]将多目标优化问题归纳总结为 3 类,先验偏好信息法(Priori-articulation of Preference Method),后验偏好信息法(Proteriori-articulation of Preference Method)和交互式偏好信息法(Interactive Preference Method)。

先验偏好信息法主要通过提前设定决策者针对不同优化目标的偏好信息,将 优化目标相互冲突的 MOP 转化为 1 个 SOP 进行求解。后验偏好信息法中决策者 信息的介入主要发生在完成一次完整搜索后,决策者从获得的一组新的解集中根 据相应的偏好信息选取最为偏好的单个或一组解。而交互式偏好信息法中偏好信息的介入发生在寻优搜索过程中,决策者通过对中间解集的不断学习提升其对于求解问题的理解,同时不断更新偏好信息,以更好地指导整个搜索过程向更新的偏好方向行进。自 Fonseca 和 Flemming 提出将帕累托支配与决策者偏好信息相结合对个体进行排序以来,基于偏好信息的 MOEAs 也受到了许多学者的关注^[69,70,126-129]。多数取得的成果以交互式偏好信息法为主,代表性成果包括,Li 和 Silva^[69]提出的 EMOSA(Evolutionary Multi-objective Simulated Annealing),Deb 和 Chaudhuri^[70]提出的 I-EMO(Interactive Evolutionary Multi-Objective Optimziation Tool),Thiele等^[82]提出的 PBEA(Preference Based Evolutionary Algorithm)。EMOSA 通过解集当前位置,借助决策者信息自动调整权值向量以指引搜索过程向帕累托最优前沿的不同区域驶进。I-EMO 通过适当选择现有的不同多目标优化方法,结合决策者信息,已达到获得决策者偏好解的目的。PBEA 将决策者的偏好信息以参考点的形式引入,结合该参考点计算标量化到达函数(Achievement Scalarizing Function)以实现适应度分配,进而指导搜索过程向参考点附近区域驶进。

(3) 基于性能指标的 MOEA 框架(Indicator Based MOEA)

基于性能指标的 MOEA 框架主要通过计算优化过程中生成的近似解集的性能指标,进而指导优化解集的选择过程。受限于计算复杂,这一框架近些年的发展较为缓慢[77,78,130]。

(4) 基于分解的 MOEA 框架(Decomposition Based MOEA)

基于分解的 MOEA 框架可以追溯到 1998 年。Ishibuchi 和 Muratatat^[50]提出的多目标遗传局部搜索(Multi-Objective Genetic Local Search,MOGLS)通过一组随机生成的权值向量根据加权求和的方式确定每个个体的适应度,然后根据比例选择方法确定每个个体被选择执行交配的概率。同时,从外部档案中随机选取一定数量的解构成新的种群。最后,通过局部搜索对生成的新种群进行改进。该方法采用较为简单的加权和方式,一定程度上限制其使用范围。而现在基于分解的 MOEA 框架中大部分工作的开展都是基于 Zhang 等^[55]提出的 MOEA/D 算法。目前针对MOEA/D 的研究主要可以分为基于基于权值向量生成^[46,80,131],不同分解方法^[38,72,123,132,133],建立计算资源分配^[111,134,135],变化算子选择^[110,136]这几大框架。

1.3.3 减速器多目标优化研究现状

工程机械中,减速器作为直接传递转矩的重要子系统,直接影响整个系统的性能表现。在过去几十年里,减速器单目标优化问题已经得到了广泛的研究^[137-144],这些问题大都以最小化减速器齿轮体积为优化目标,即最小化减速器重量。随着竞

争的加剧,单纯以最小化体积作为优化目标已无法满足市场对产品的更高要求。将疲劳寿命、传递扭矩、传动效率、齿轮磨损等相互冲突的优化目标加入到减速器优化设计中,从多个角度对减速器进行多目标优化设计,更加符合实际需求。在过去二十年间,减速器多目标优化设计引起了越来越多的学者的关注。初期,多数学者为了简化计算过程,通过线性加权^[145-150]、目标规划^[151]等方法将多个目标转化为单一目标函数,并借助单目标优化方法进行求解。然而,将多目标优化问题通过加权求和或目标规划的形式转换为单目标优化问题,属于先验偏好信息法,其转换过程完全依赖决策者的先验信息,缺乏科学依据。

Huang 等 $^{[152]}$ 借助交互式物理规划方法一定程度上消除决策者偏好信息对优化 求解的影响。Deb 等 $^{[31]}$ 将 NSGA-II 成功应用到多级齿轮减速器优化设计中,并取 得满意的结果。后续相关学者针对不同的目标函数,采用 NSGA-II $^{[153-156]}$,同样对减速器多目标优化问题进行求解,取得满意的结果。车林仙等 $^{[157]}$ 从提升多目标进化算法性能为出发点,提出一种网格占优约束多目标差分计划算法(ε -CMODE),有效求解齿轮机构多目标优化问题的同时,得到均匀分布的帕累托最优前沿。

1.3.4 综述总结与问题提出

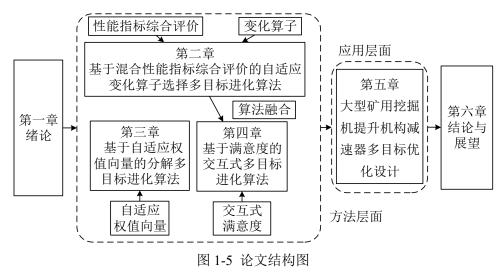
综上,基于分解的 MOEA 以其优异性能成为目前多目标进化算法的研究热点,并广泛应用于路径规划^[158-160]、天线设计^[161,162]、网络优化^[163-167]等实际工程优化中。鉴于目前减速器多目标优化问题中减速器的结构较为简单,目标函数的数量多为 2目标或 3 目标,采用基于帕累托支配和松弛帕累托支配基本框架的 NSGA-II 和 ε-MOEA 等算法,在一定程度上可以获得较为满意的优化解。然而,本文的工程应用对象为大型矿用挖掘机提升机构减速器,其结构较传统标准化减速器复杂,同时需要考虑的优化目标大于 3,为此传统基于帕累托支配和松弛帕累托支配基本框架的多目标进化算法会出现性能退化。MOEA/D 通过调整其自带生成机制和环境选择策略能够提升 MOEA/D 在处理 MaOPs 问题时的性能。目前尚未找到 MOEA/D 在减速器优化中的应用的相关文献,因此,提升 MOEA/D 在多样性和收敛性上综合表现,进而应用到大型矿用挖掘机提升机构减速器优化设计中为本文主主要研究思路。

同时,由于权值向量的取值与解集多样性表现直接相关,因此从权值向量自适应生成角度出发,探讨能够有效提升解集多样性的算法显得尤为重要。同时,近期许多学者不断指出^[168,169]变化算子作为决策空间执行探索与开发的控制单元,也会影响解集的收敛性和多样性。然而,目前针对此类问题的研究较少,因此从平衡决策空间探索与开发角度出发,探讨能够有效平衡搜索过程探索与开发的变化算子

选择策略也显得尤为重要。此外,基于交互式偏好信息的 MOEAs 往往能够帮助偏好者求得偏好解,而实际工程设计优化问题中,决策者最终往往只需要单独最优解或一小组优化解集指导实际设计,因此,探讨能够有效地将决策者偏好信息交互地与 MOEAs 相结合在工程实际应用中显得尤为重要。

1.4 本文的结构安排

本文总共分为六章,章节的安排以提升多目标优化问题综合性能为主线,以矿用挖掘机提升机构减速器多目标优化问题为工程背景,本文的总体框架如图 1-1 所示。



各章节具体内容概述如下:

第一章为绪论,介绍本文的研究背景与意义、多目标优化问题数学基础、性能评价指标、测试函数、多目标进化算法的发展历程和研究现状,并简要概述本文主要研究内容和行文结构。

第二章将介绍基于混合性能指标综合评价的自适应变化算子多目标进化算法,研究自适应变化算子选择策略解决 MOEAs 在搜索空间平衡探索和开发的问题。

第三章将介绍基于自适应权值向量的分解多目标进化算法,研究通过自适应 更新权值向量提升分解多目标进化算法在目标空间多样性的问题。

第四章将介绍基于满意度的交互式多目标进化算法,研究通过交互式满意度 策略获得决策者偏好解的问题。

第五章综合考虑大型矿用挖掘机提升机构减速器的设计要求和实际工况,完善矿用挖掘机提升机构减速器多目标优化问题,验证之前提出 MOEAs 的有效性。第六章将总结全文,并概要展望以后的研究工作。

第二章 基于混合性能指标综合评价的变化算子自适应选择 多目标进化算法

2.1 引言

当前,针对 MOEAs 的研究以平衡优化解集在收敛性和多样性上的综合表现为主。相关研究指出,对于目标空间中解集在多样性和收敛性上的要求加剧了在决策空间中平衡探索(Exploration)与开发(Exploitation)的难度^[170]。变化算子作为决策空间执行搜索的关键因素,其表现受参数调整以及求解具体问题的影响较大。不同变化算子处理不同问题性能不尽相同,设计者需要有足够的先验知识作为保障,以选取合适的变化算子执行搜索空间的探索与开发。

Zhang 等[111]在 2009 年为了提升 MOEA/D 在多样性与收敛性上的表现,提出了使用差分进化代替原始版本中的模拟二进制交叉算子(Simulated Binary Crossover,SBX),使算法在求解问题时性能大幅提升。这一发现促使学者们就变化算子影响 MOEAs 性能展开进一步研究。目前针对平衡变化算子在搜索空间中探索与开发能力的相关研究还处于初级阶段,代表性研究主要以变化算子自适应选择策略(Adaptive Operator Selection,AOS)[114]为主。

AOS 策略主要由两部分组成: 1) 信任度分配(Credit Assignment)和 2) 变化算子选择(Operator Selection)。这两部分是一个递进关系,通过对不同算子当前状态下的性能表现进行评定,进而依据某种评定的性能表现选择下一次迭代中的变化算子。采用基于该策略的 FRRMAB[114]和 MOEA/D-UCB-tuned^[136]在性能上有较为突出的表现。其中 MOEA/D-UCB-tuned 在 FRRMAB 的基础上,将多臂赌博机算法 UCB1 替换为 UCB-tuned。UCB1 和 UCB-tuned 本质上都是一种概率选择机制,而变化算子的搜索过程本身也是基于概率。此外,上述两种算法在迭代过程中变化算子池中的所有变化算子在每次搜索过程中全部都要参与,加剧了算法的复杂度。鉴于此,为了消除概率的影响,同时避免变化算子池中每个变化算子每次都参与的情况。本章引入一种基于混合性能指标综合评价的自适应算子选择策略(IHO),以 MOEA/D 为主体框架,借助迭代过程中前后解集在性能指标上的综合变化,择优选择算子以平衡在搜索空间中的探索与开发。同时,提出在探索能力上表现优异的基于乌鸦算法变形的新型变化算子,与传统在大多数问题上开发能力较强的 DE/rand/1 算子相结合,组成新的变化算子选择池。

本章首先简单介绍 CSA 的基本原理,结合 DE 等交叉算子的特性,提出一种新的变异算子(Crow Search Operator, CSO)。其次,根据综合考虑迭代过程中所

得解在多样性和收敛性性能上的不断变化,选取合适的变化算子,生成新的子代。 最后,将所提 IHO 变化算子选择策略与 MOEA/D 相结合形成 MOEA/D-IHO,并 详细介绍 MOEA/D-IHO 算法流程和与先进算法的实验对比。

2.2 改进型乌鸦搜索变化算子

2.2.1 差分进化变化算子

目前为止 DE 因其简单的设置与良好的性能,成为 EAs 领域最受欢迎的执行变化操作的变化算子。DE 有许多种不同的变异策略,其中应用最广的为"DE/rand/1",其函数表达如下:

$$\mathbf{v}_{i}^{iter} = \mathbf{x}_{r_{i}^{i}}^{iter} + F(\mathbf{x}_{r_{i}^{i}}^{iter} - \mathbf{x}_{r_{i}^{i}}^{iter})$$
 (2-1)

式中, \mathbf{v}_{i}^{iter} 为第iter 次迭代过程中与决策向量 \mathbf{x}_{i}^{iter} 相对应的变异向量, \mathbf{r}_{i}^{i} , \mathbf{r}_{2}^{i} , \mathbf{r}_{3}^{i} 为 1 到 n 的随机生成整数,n 为决策向量的维度。F 为区间[0,1]内可调常数:调整 $\mathbf{x}_{\mathbf{r}_{i}^{i}}^{iter}$ 和 $\mathbf{x}_{\mathbf{r}_{i}^{i}}^{iter}$ 在变异操作中的参与程度。

本文 DE 的交叉策略全部采用二项式交叉的方式,其函数表达如下:

$$u_{i,j}^{iter} = \begin{cases} v_{i,j}^{iter} & \text{if } \text{rand} \le CR \text{ or } j = j_{rand} \\ \mathbf{x}_{i,j}^{iter} & \text{otherwise} \end{cases}, j = \{1, 2, ..., n\}$$
 (2-2)

式中,CR 为区间[0,1]内可调常数:交叉率,控制着继承自变异向量 $v_{i,j}^{iter}$ 的比例。

2.2.2 乌鸦搜索变化算子

类似于受启发于鸟群飞翔与鱼群觅食这一自然群体现象的群体智能算法,乌鸦算法(Crow Search Algorithm,CSA)源于乌鸦(自然界中最聪明的鸟类)的一系列智能行为,于近年由 Askarzadeh 提出,并作为一种新的群体智能优化算法以解决高度非线性多峰特性目标函数的问题^[171]。区别于传统的粒子群算法(Particle Swarm Optimization,PSO),CSA 算法借鉴乌鸦在自然界中智能行为,引入防追踪机制,极大地降低在处理多峰问题时陷入局部最优的概率^[172]。

乌鸦算法的基本概念如下:

- (1) 与 PSO 算法一样, CSA 算法中的乌鸦也以群体的方式生活在一起;
- (2) 乌鸦能够记住他们藏匿食物的位置;
- (3) 乌鸦会跟踪其它乌鸦以试图窃取其它乌鸦的食物:
- (4) 每一个乌鸦又会防止它们自己藏匿的食物被其它乌鸦偷走。

在每一次迭代中,每一个乌鸦 \mathbf{x}_i ,i=1,2,...,N在D维空间中第iter次迭代时的

位置表示为 $\mathbf{x}_{i}^{iter} = [x_{i,1}^{iter}, x_{i,2}^{iter}, \dots, x_{i,d}^{iter}]$,每一个乌鸦 i 记录其藏匿食物位置 \mathbf{m}_{i}^{iter} , $i = 1, 2, \dots, N$ 也会随着迭代的进行而更新。CSA 的基本流程如图 2-1 所示,具体求解某一优化问题时,目标函数、约束条件与设计变量均应提前假定。

首先,确立固定参数,包括群体尺寸N、飞行距离 fl、最大迭代次数 $iter_{max}$ 和知晓概率 AP。在D维可行域中随机生成N组D维决策变量 \mathbf{x}_i , i=1,2,...,N 作为乌鸦群体的初始位置,并将其设为每个乌鸦所对应的食物最佳藏匿位置。

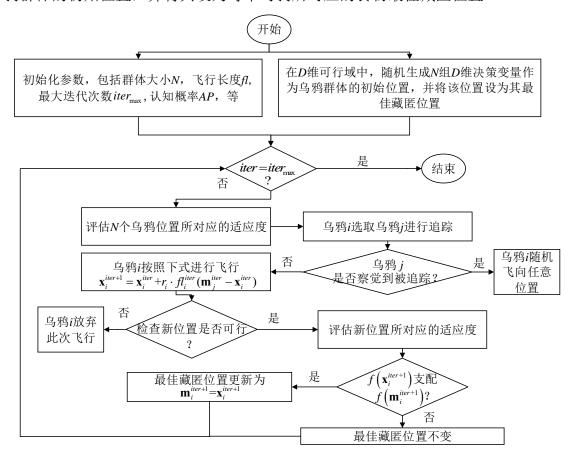


图 2-1 乌鸦算法基本流程图

其次,假设在第iter次迭代过程中,乌鸦i随机选取乌鸦j进行跟踪,并且乌鸦j会飞向自己当前的食物藏匿位置。由于乌鸦本身的智力行为,跟踪时会出现两种情形:

情形 1: 乌鸦 j 并未察觉到乌鸦 i 在跟踪它,乌鸦 i 能够飞向乌鸦 j 的最佳藏匿位置附近,于是乌鸦 i 的新位置如下:

$$\mathbf{x}_{i}^{iter+1} = \mathbf{x}_{i}^{iter} + r_{i} \cdot f l_{i}^{iter} \left(\mathbf{m}_{i}^{iter} - \mathbf{x}_{i}^{iter} \right)$$
 (2-3)

式中, r_i 为 0 到 1 的一个随机数, fl_i^{iter} 为乌鸦 i 在第 iter 次迭代时的飞行距离。乌鸦 i 的飞行轨迹如图 2-2 所示。为了提升搜索多样性,在 $\left(\mathbf{m}_{j}^{iter}-\mathbf{x}_{i}^{iter}\right)^{T}$ 的基础

上乘以飞行距离参数 fl_i^{irer} 和 0-1 随机数 r_i ,这样乌鸦 i 在下一次(iter+1)的位置便会出现在乌鸦 j 在第 iter+1 次的食物最佳藏匿位置附近。同时,如图 2-4 所示,如若 $r_i \cdot fl_i^{iter} > 1$,该跟踪过程倾向于全局搜索,当 $r_i \cdot fl_i^{iter} < 1$ 时,该跟踪过程倾向于集约化,收敛于局部最优。

情形 2: 乌鸦 j 察觉到乌鸦 i 正在跟踪,乌鸦 j 改变其飞行轨迹飞到错误的地方以免暴露其当前食物藏匿位置 \mathbf{m}_{i}^{ier} 。

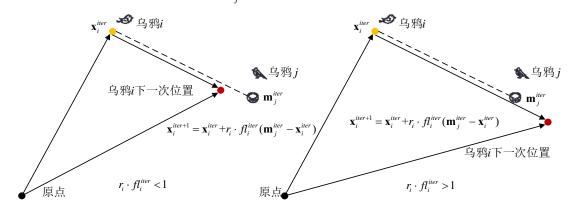


图 2-2 乌鸦算法乌鸦"飞行"示意图

综上, 乌鸦i的第iter次迭代后的新位置如下:

$$\mathbf{x}_{i}^{iter+1} = \begin{cases} \mathbf{x}_{i}^{iter} + r_{i} \cdot fl_{i}^{iter} \left(\mathbf{m}_{j}^{iter} - \mathbf{x}_{i}^{iter}\right) & r_{i} \geq AP_{j}^{iter} \\ \text{random location} & r_{i} < AP_{j}^{iter} \end{cases}$$
(2-4)

式中, AP_j^{iter} 为乌鸦 j 在第 iter 次迭代时的晓知概率 $^{[171]}$ 。如若随机生成数 r_i 大于 AP_j^{iter} ,那么它代表乌鸦 j 并未察觉到此次追踪,反之乌鸦 j 放弃回到自己食物藏匿点。

每一次迭代过后,还需要检查新位置的可行性,即是否满足约束条件。

综上, CSA 算法的优点主要有如下几点:

- (1) 群体中每个乌鸦都有类人的智力特征,例如自我意识(认知被追踪的能力和寻求更佳食物藏匿点的能力)和记忆能力(藏匿和获取食物)。
- (2) 算法中需调整参数只有飞行距离 fl 和晓知概率 AP ,这样降低了算法受参数调整的影响。
- (3)算法多样性通过乌鸦i随机选取乌鸦j进行跟踪,并将第iter+1次最新位置 \mathbf{x}_{i}^{iter+1} 与自己当前食物藏匿地点 \mathbf{m}_{i}^{iter} 进行比较进而更新其食物藏匿地点得以保证。

2.2.3 乌鸦搜索变异因子

对于单目标优化问题, CSA 算法中的每个乌鸦的最佳藏匿位置很容易通过简

单的比较大小来获取。然而,对于多目标优化问题,由于目标之间相互冲突,乌鸦当前位置与其最佳藏匿位置之间的比较就变得复杂。受启发于 MOEA/D-DE 的基本思路^[58],同样地将能够有效处理单目标优化问题的 CSA 优化算法扩展到处理多目标优化问题层面。

目前,CSA 向 EMO 领域进行扩展的研究很少[172-174]。在已有案例中,鉴于 CSA 中控制整个搜索过程的关键参数为:飞行距离 fl 和晓知概率 AP,学者们也主要着重于 fl 与 AP 的设定,同时对式(2-4)中的随机搜索进行修改。

本节基于式(2-3)的搜索机制,通过额外增加一个乌鸦k的形式将 CSA 扩展到 EMO 领域。由于原始 CSA 中执行每一次飞行的时候,乌鸦j都被设定为搜寻自己最佳位置,而这一设置很有可能会导致搜索过程陷入局部最优。因此,本文提出将乌鸦j的飞行轨迹替换为 $(\mathbf{m}_k^{iter} - \mathbf{x}_j^{iter})$,因此乌鸦i的跟踪飞行轨迹也变为 $\mathbf{x}_i^{iter+1} = \mathbf{x}_i^{iter} + r_i \cdot fl_i^{iter} (\mathbf{m}_k^{iter} - \mathbf{x}_i^{iter})$ 。为了提高本章所提变化算子的探索能力,式(2-2)中的随机位置为 $\mathbf{x}_i^{iter} + r_i \cdot ||U - L||$ 。至此,本文提出的乌鸦搜索变化算子(Crow Search Operator,CSO)中变异操作的函数表达如下:

$$\mathbf{x}_{i}^{iter+1} = \begin{cases} \mathbf{x}_{i}^{iter} + r_{i} \cdot f l_{i}^{iter} \left(\mathbf{m}_{k}^{iter} - \mathbf{x}_{i}^{iter} - \mathbf{x}_{i}^{iter} \right) & r_{i} \geq A W_{j}^{iter} \\ \mathbf{x}_{i}^{iter} + r_{i} \cdot \left\| U - L \right\| & r_{i} < A W_{j}^{iter} \end{cases}$$
(2-5)

式中, U和L分别代表决策向量的上界和下界。

CSO 同样采用二项式交叉操作,具体数学表达式如式 (2-2)。

2.3 基于混合性能指标综合评价的变化算子自适应选择策略

2.3.1 算法思路

基于性能综合指标评价的变化算子自适应选择策略主要采用三种性能指标对历史解集性能进行综合表征,包括世代距离性能指标 $GD^{[93]}$,最大广度性能指标 $MS^{[54]}$,和集合覆盖率性能指标 $C^{[57]}$ 。其中,GD的计算需要借助真实帕累托最优前沿面的信息,而这会降低算法的应用范围。本文采取前后两组优化解集之间的GD值进行计算,以其中一组解集替代传统GD计算中的帕累托最优前沿面。

算法根据三个性能指标的综合评价,判断当前搜索过程的的进化倾向性,依据 倾向性选择相对应的变化算子继续执行后续进化。

2.3.2 算法详解

MOEA/D-IHO 算法框架如算法 2.1 所示。

算法 2.1: MOEA/D-IHO

```
种群数量N, 邻域尺寸T, 最大迭代次数iter_{max}
初始化
           生成一组均匀分布权值向量W \leftarrow \{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_N\}
  1.
           根据权值向量W 初始化种群 P^0 \leftarrow \{p_1^0, p_2^0, ..., p_N^0\}
  2.
           根据初始种群P^0初始化理想点z_0^*
  3.
           计 算 距 离 权 值 向 量 \mathbf{w}, 最 近 的 T 组 向 量 , 并 组 成 邻 域
  4.
            \lambda_i \leftarrow \{\mathbf{w}_i, \mathbf{w}_i, \dots, \mathbf{w}_T\}
  5.
            iter \leftarrow 1
           while iter < iter_{max} do
  6.
              分别设定C与GD的判断因子为:cj = 0.1(1-0.9iter/iter_{max}),
  7.
              gdj = 0.01(1-0.9iter/iter_{max}), msj = 1e-4
  8.
              if iter < 10
                 P^{iter} \leftarrow GenerateOffspring(P^{iter}, DE)
  9.
              elseif mod(iter, 10) == 0
  10.
                 \{C\_diff, GD, MS\} \leftarrow CalIndicator\{(P^{iter-1}, ..., P^{iter-8})\}
  11.
                 if mean\{GD(i)\} < gdj or mean\{C\_diff\} > cj
  12.
                    P^{iter} \leftarrow GenerateOffspring(P^{iter}, DE), \quad flag \leftarrow c
  13.
                 elseif std\{MS(i)\} > msj
  14.
                    P^{iter} \leftarrow GenerateOffspring(P^{iter}, CSO), \quad flag \leftarrow d
  15.
  16.
                 else
                    P^{iter} \leftarrow GenerateOffspring(P^{iter}, (DE, CSO)), \quad flag \leftarrow m
  17.
  18.
                 end if
  19.
              else
                 switch flag
  20.
  21.
                    P^{iter} \leftarrow GenerateOffspring(P^{iter}, DE)
  22.
  23.
                    break
  24.
                 case d
                    P^{iter} \leftarrow GenerateOffspring(P^{iter}, CSO)
  25.
  26.
                    break
  27.
                 case m
  28.
                    P^{iter} \leftarrow GenerateOffspring(P^{iter}, (DE, CSO))
  29.
                    break
  30.
                 end switch
  31.
              end if
  32.
           end while
  33.
           return P 中所有非支配解
```

算法 2.1 首先生成一组均匀分布的权值向量 $W \leftarrow \{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_N\}$,根据 $\mathbf{w}_i, i = 1, 2, ..., N$ 确定 N 个子问题 p_i ,并组成初始化种群 $P^0 \leftarrow \{p_1^0, p_2^0, ..., p_N^0\}$,计算 距离权值向量 \mathbf{w}_i 最近的 T 组向量,组成子问题 p_i 的邻域 $\lambda_i \leftarrow \{\mathbf{w}_i, \mathbf{w}_j, ..., \mathbf{w}_T\}$ 。其中,权值向量根据 Das 和 Dennis 的生成方式产生^[175],步骤 3 中初始理想点 z_0^* 的生成主要依靠当前种群中所有个体在目标函数值的最小值进行估计。

IHO 策略主要通过算法 2.1 中步骤 7 至步骤 32 完成,其中主体内容主要为步骤 11 至步骤 19。在算法初期,步骤 7 首先设定用于判断 C、GD和MS 性能指标值的判断因子,以综合控制步骤 11 所求性能指标值对历史解集收敛性和多样性的判断。这里将两个判断因子同时进行考虑,主要出于完整性的考量,首先GD的均值小于设定的值,只能说明之前 9 次迭代过程中GD变化不是特别明显,在算法后期可能GD的值会非常小,而此时则需要C值在加以辅助。当迭代次数小于 10,步骤 9 被激活,以 DE 为变化算子生成子代,函数 2.1 和函数 2.2 详细阐述了该过程。

函数 2.1: Generate Offspring

输入: 变化算子 $Op = \{DE, CSO, (DE, CSO)\}$, 第 iter 次迭代时刻种群 p^{iter}

输出: p^{iter}

- 1. **for** $k \leftarrow 1: N$ **do**
 - 从 p_i^{iter} 的邻域内随机选择三个个体,通过op变化算子生成子代
- 2. $u_i^{iter} \circ$
- 3. $P^{iter} \leftarrow Update(u_i^{iter}, p_i^{iter})$
- 4. end for

函数 2.2: Update

输入: 子代, u_i^{iter} , 子问题, p_i^{iter}

输出: 子问题, p_i^{iter}

根据 p_i^{iter} 的权值向量 $\mathbf{w}_i = \{w_1, w_2, ..., w_M\}$, 计算 u_i^{iter} 以及 p_i^{iter} 中其他

- 1. 解的切比雪夫函数值, $f_{Tch}\left(u_i^{iter}\right)$, $f_{Tch}\left(nox_{i,k}^{iter}\right)$, $nox_{i,k}^{iter} \in p_i^{iter}$, $k = \{1, 2, ..., T\}$
- 2. **for** $k \leftarrow 1:T$ **do**
- 3. **if** $f_{Tch}(u_i^{iter}) < f_{Tch}(nox_{i,k}^{iter})$ **then**
- 4. $nox_{i,h}^{iter} \leftarrow u_{i}^{iter}$
- 5. end if
- 6. end for

当迭代次数大于 10,步骤 11 至步骤 19 被激活,首先通过步骤 11 计算之前 9 组集生成的性能指标值,函数 2.3 详细阐述了该计算过程。当满足条件 12 时,说明之前 9 次搜索在收敛性上表现良好。因此,步骤 13 被激活,继续以 DE 为变化算子生成子代,以强化收敛性表现。当满足条件 14 时,说明之前 9 次搜索在收敛性上变化甚微,在多样性上表现良好。因此,步骤 15 被激活,以 CSO 为变化算子生成子代,强化多样性表现。当满足条件 12 时,表明当前算法陷入困境,因此采用双变化算子先 DE/rand/1 后 CSO 的策略,以试图跳出此困境。

函数 2.3: CalIndicator

输入: 之前9组解{P^{iter-1},...,P^{iter-9}}

输出: $\{C_diff(i), GD(i), MS(i)\}, i = 1,..., 8$

- 1. $i \leftarrow 1$
- 2. **for** $k \leftarrow 1:8$ **do**

计算 P^{iter-i} 和 $P^{iter-i-1}$ 的二元收敛性指标C, $C(P^{iter-i},P^{iter-i-1})$ 和

- 3. $C(P^{iter-i-1}, P^{iter-i})$,计算 $C(P^{iter-i}, P^{iter-i-1})$ 和 $C(P^{iter-i-1}, P^{iter-i})$ 之间差值的绝对值 $C_{-}diff(i) = |C(P^{iter-i-1}, P^{iter-i}) C(P^{iter-i}, P^{iter-i-1})|$
- 4. 计算 P^{iter-i} 和 $P^{iter-i-1}$ 的收敛性指标GD, $GD(i) = GD(P^{iter-i}, P^{iter-i-1})$
- 5. 计算 P^{iter-i} 的最大广度指标MS, $MS(i) = MS(P^{iter-i})$
- 6. end if

2.4 实验设计

2.4.1 测试函数集

本章主要采用 Huband 等[98]提出的 2 目标、4 目标和 7 目标 WFG2-WFG9 测试函数集。其各个函数的特性如表 2-1 所示。

测试函数	目标函数	可分性	多峰性	偏差	几何形状
NAECO.	$f_{_{1:M-1}}$	不可分	单峰	多项式, 平滑	ローエケル
WFG2	$f_{\scriptscriptstyle M}$	不可分	多峰	多项式, 平滑	凸,不连续
WFG3	$f_{_{1:M}}$	不可分	单峰	多项式, 平滑	线性,退化
WFG4	$f_{_{1:M}}$	可分	多峰	多项式,平滑	凹
WFG5	$f_{_{1:M}}$	可分	欺骗性	多项式, 平滑	凹

表 2-1 WFG 测试函数集函数特性

测试函数	目标函数	可分性	多峰性	偏差	几何形状
WFG6	$f_{_{1:M}}$	不可分	单峰	多项式, 平滑	Щ
WFG7	$f_{_{1:M}}$	可分	单峰	参数依靠	Ш
WFG8	$f_{_{1:M}}$	不可分	单峰	参数依靠	Ш
WFG9	$f_{_{1:M}}$	不可分	多峰, 欺骗性	参数依靠	Щ

表 2-1 WFG 测试函数集函数特性(续)

其中,WFG 的决策变量区间为 $z_i = [0,2i], i = 1,2,...,n$ 。

2.4.2 性能评价指标

目前 EMO 领域应用最为广泛的评价指标 HV 和 IGD 被用作本章的主要评价指标 [90]。其中 HV 的计算需要借助参考点信息,一般采取最差点的 1.2 倍作为参考点。同时辅以 GD 和 PD 两个性能指标,指标 GD 被用以评价 MOEAs 求解标准测试集的收敛性。指标 PD 不仅可以在广度上对 MOEAs 进行评价,同时能够保证对分布均匀性进行度量,被认为是目前评价多样性最为有效的指标。然而,PD 的计算复杂度随着解集数量递增呈指数性增长,因此需要合理设置对比算法中群体的规模。除此之外,还借助 Median Attainment Surface [89]对 2 目标优化解集在目标空间中的分布进行综合对比。

2.4.3 对比算法

为了评估本章所提出的 MOEA/D-IHO 算法的性能,鉴于目前尚无采用同等策略的 MOEAs,本章将其与以下四种采取不同框架的优秀算法进行比较,四种算法分别为基于帕累托支配基本框架的 NSGA-II^[53]、基于松弛帕累托支配基本框架的 GrEA^[176]、基于性能指标基本框架的 HypE^[78]和基于分解基本框架的 MOEA/D-DE^[111]。

Deb 等^[53]提出的基于精英策略的非支配排序遗传算法(Non-dominated Sorting in Genetic Algorithm-II,NSGA-II)作为基于帕累托支配基本框架下应用最为广泛的多目标进化算法,NSGA-II 在 NSGA 的基础上,通过快速非劣排序法提升了算法效率,同时引入拥挤度对比,提升了算法在多样性上的表现,同时通过父代与子代一同对比的方式实现了精英保留策略。

Yang 等^[75]提出的基于动态网格的进化算法 (Grid Based Evolutionary Algorithm, GrEA) 作为基于松弛支配基本框架下的行之有效的多目标进化方法,通过网格支配(Grid Domination)关系提升算法在收敛性上的表现。在适应度评价机制中,引

入网格排序(Grid Ranking,GR)和网格拥挤距离(Grid Crowding Distance,GRD)用以对个体分别从收敛性和多样性两个角度进行区分。除此之外,GrEA 为了避免两个个体可能会出现相同的 GR 和 GRD 值,引入网格坐标点间距(grid coordinate point distance,GCPD)用以计算个体与其所在网格的最优点之间的距离,并取GCPD 值最小的解作为适应度较高的个体用以区分 GR 和 GRD 相同的个体。大量实验结果显示,采用动态网格的 GrEA 在处理大部分测试问题时都要明显优于 ε -MOEA。

Bader 和 Zitzler^[78]提出的基于超体积性能指标多目标优化算法(Hypervolume Estimation Algorithm for Multiobjective Optimization,HypE)作为基于性能指标基本框架近期提出的代表性算法,区别于传统精确求解超体积数值的方法,通过蒙特卡洛采样近似求解超体积,在保证算法精度的同时降低算法运行时间,提升了算法的可用度。对每一个解的适应度计算,主要通过从当前种群中去除该解与另外k-1个解后,与另外k-1个解相关联的超体积损失的期望,其中k取值大于等于当前种群个数的一半。由于 HypE 的适应度计算基于超体积值,而超体积能够综合表征算法在多样性和分布性上的表现,因此 HypE 能够平衡算法在收敛性和分布性之间的综合表现。

Li 和 Zhang^[110]在 MOEA/D 的基础上,将 SBX 变化算子替换为 DE 变化算子以生成子代,提出了 MOEA/D-DE。MOEA/D 作为基于分解基本框架的代表性算法,主要通过一组均匀分布的权值向量,将原 MOP 分解为 N 个与权值向量相关的子问题。同时最小化每一个子问题的聚合函数,经过数次迭代,MOEA/D 便能够获得 N 个最优解。其中,通过计算与子问题 j 相对应权值向量 w_j 距离最近的 T 个权值向量,确定子问题 j 的邻域 B_j 。将子问题 j 生成的子代 x_j 代入其邻域 B_j 的聚合函数中,比较该子代 x_j 与相邻子问题当前个体在聚合函数上的表现,如果优于相邻子问题对应的个体,则将该相邻子问题对应的个体更新为 x_i 。

2.4.4 实验设置

本章数值实验中涉及到的具体参数设置如下:

- (1) 群体规模:对于 2、4、7目标 WFG 测试集优化问题,NSGA-II、GrEA和 HypE 的 N 全部设置为 100,MOEA/D-DE 和 MOEA/D-IHO 的 N 分别设置为 100、126 和 210。
- (2)MOEA/D-DE 和 MOEA/D-IHO 邻域T 为常数 10,GrEA 中 div 设置遵从原作设为 $12^{[116]}$,HypE 中用于计算 HV 的样本量设为 10000。
 - (3) WFG 测试集中k = 18, l = 14。

- (4)所有算法均采用 DE/rand/1 和二项式变异变化算子的组合,其中 DE/rand/1 中 CR = 1.0, F = 0.5,多项式变异中 $\eta_m = 20$, $p_m = 1/32$, CSO 中 AW = 0.8。
 - (5) 所有算法独立运行 31 次,每次运行迭代 250 次。

此外,在性能指标的计算时,所有非支配解的个数都设置为100。

2.5 实验结果与分析

为了检测 MOEA/D-IHO 的有效性,所有对比算法独立运行 31 次。计算结果分别从 Median Attainment Surface、数值统计和盒图三个方面展开。绘制所有对比算法运行 31 次后获得的近似帕累托最优前沿面的 Median Attainment Surface,如图 2-3 所示。每个算法求解得到 $HV \setminus IGD \setminus GD$ 和 PD 四个性能评价指标计算值的统计结果,以各指标值平均值作为对比基准,分别列于表 2-2 至 2-5 中。并绘制了每个算法在 $HV \setminus IGD \setminus GD$ 和 PD 四个性能评价指标计算值的盒图。其中,所有图例中,IHO 代表算法 MOEA/D-IHO,MOEADDE 代表算法 MOEA/D-DE。每个示例中 WFGX-Y 代表 Y 个目标的测试函数 WFGX, $X \in \{2,3,...,9\}$ 。

2.5.1 Median Attainment Surface

Median Attainment Surface 直观地将各个算法优化解在目标空间中进行对比,同时辅以每个测试函数的真实帕累托最优前沿,便于查看各个算法优化解在目标空间的多样性和收敛性情况。从图 2-3 我们可以看出:

- (1) 在对离散型凸函数 WFG2 问题的求解上,所有算法都不能收敛到真实帕累托最优前沿面。其中 HypE 和 GrEA 在收敛性上的表现优于其它算法,而 MOEA/D-IHO 在多样性上表现最优。
- (2) 在线性凸函数 WFG3 问题的求解上,所有算法都有较好的收敛性,其中 MOEA/D-IHO 在多样性上的表现优于其它算法。
- (3) 在多峰可分凹函数 WFG4 问题的求解上 MOEA/D-IHO 的综合表现都优于其它算法。
- (4) 在带有欺骗性可分凹函数 WFG5 问题的求解上 MOEA/D-IHO 的综合表现略优于其它算法,这与 MOEA/D-DE 的性能比较接近。
- (5) 在不可分单峰凹函数 WFG6 问题的求解上 MOEA/D-IHO 的综合表现远优于其它算法。同样地,在求解 WFG7 至 WFG9 问题时,MOEA/D-IHO 的综合性能表现也要明显优于其它算法。

根据直观观察, MOEA/D-IHO 几乎在所有测试函数上的表现都优于其它算法。由于每个算法优化解集之间距离较近,还需要借助数值统计的均值和盒图综合表

征各算法在收敛性上的表现。

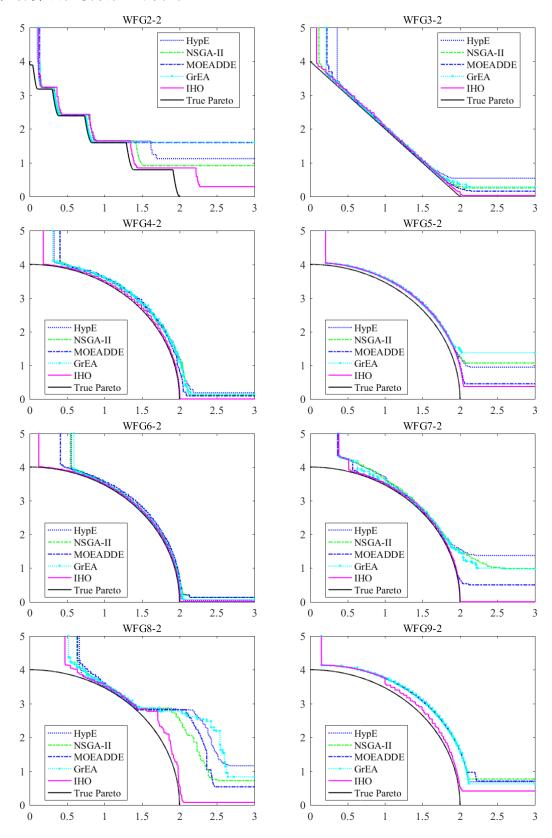


图 2-3 2 目标 WFG 测试集 Median Attainment Surface

2.5.2 实验数值统计结果

本节对 31 次独立运算结果进行平均值统计,每种问题中表现最好的算法都进行加粗处理。如表 2-2 所示,MOEA/D-IHO 几乎在所有测试函数问题上所得平均 *HV* 性能指标都优于其他算法,HypE 在 WFG3 问题上的表现较为突出,GrEA 在 WFG5-7、WFG6-4、WFG6-7、WFG9-4 和 WFG9-7 上的表现优于其它算法。

表 2-2 5 种算法在 WFG2 至 WFG9 测试问题上所得平均 HV 值的比较

	目			算法		
测试函数	标	NSGA-II	GrEA	MOEA/D-DE	НурЕ	MOEA/D-IHO
-	2	0.621258	0.62229	0.5734194	0.623452	0.638516
WFG2	4	0.769097	0.847839	0.8341935	0.875516	0.925194
W1 G2	7	0.755129	0.873871	0.8556129	0.959645	0.979452
	2	0.607677	0.606613	0.6129677	0.601194	0.634677
WEC2	4					
WFG3		0.188032	0.161581	0.1430323	0.367355	0.282613
	7	0	0	0.1464194	0.237548	0.153871
	2	0.387677	0.387484	0.391871	0.379484	0.42371
WFG4	4	0.408387	0.57129	0.4593548	0.452677	0.670194
	7	0.460258	0.600065	0.5486129	0.493806	0.731774
	2	0.373323	0.367419	0.3925806	0.380774	0.402516
WFG5	4	0.456871	0.564742	0.4731613	0.542387	0.600935
	7	0.483	0.594903	0.4849032	0.560129	0.575161
	2	0.395258	0.392161	0.3955806	0.381	0.432839
WFG6	4	0.582419	0.665516	0.5543548	0.606645	0.577935
	7	0.643548	0.760903	0.6446774	0.674161	0.688097
	2	0.349645	0.353129	0.4003548	0.352548	0.420129
WFG7	4	0.419871	0.554742	0.477	0.57029	0.679839
	7	0.470613	0.567355	0.4543871	0.573161	0.718871
	2	0.264129	0.266677	0.2785484	0.266742	0.384129
WFG8	4	0.344516	0.433516	0.3734839	0.519742	0.605903
	7	0.420097	0.480935	0.4217419	0.603	0.555323
	2	0.351419	0.354774	0.3704839	0.353194	0.376129
WFG9	4	0.470258	0.558581	0.4602903	0.552032	0.529
	7	0.450258	0.599323	0.4469355	0.586323	0.509323

如表 2-3 所示,MOEA/D-IHO 在所有测试函数问题上所得平均 IGD 性能指标与在所有测试函数问题上所得平均 HV 性能指标不同,表现较差。MOEA/D-IHO 在WFG2-4、WFG-2-7、WFG4-4、WFG4-7、WFG7-4、WFG7-7 和 WFG8-4 问题上所得平均 IGD 性能指标差距较大,在其他测试问题上差距较小或表现居中。其中NAGA-II 的表现最好,其次是 MOEA/D-DE。

表 2-3 5 种算法在 WFG2 至 WFG9 测试问题上所得平均 IGD 值的比较

加小五二十五	目			算法		
测试函数	标	NSGA-II	GrEA	MOEA/D-DE	НурЕ	MOEA/D-IHO
	2	0.621258	0.62229	0.573419	0.623452	0.638516
WFG2	4	0.769097	0.847839	0.834194	0.875516	0.925194
	7	0.755129	0.873871	0.855613	0.959645	0.979452
	2	0.607677	0.606613	0.612968	0.601194	0.634677
WFG3	4	0.188032	0.161581	0.143032	0.367355	0.282613
	7	0	0	0.146419	0.237548	0.153871
	2	0.387677	0.387484	0.391871	0.379484	0.42371
WFG4	4	0.408387	0.57129	0.459355	0.452677	0.670194
	7	0.460258	0.600065	0.548613	0.493806	0.731774
	2	0.373323	0.367419	0.392581	0.380774	0.402516
WFG5	4	0.456871	0.564742	0.473161	0.542387	0.600935
	7	0.483	0.594903	0.484903	0.560129	0.575161
	2	0.395258	0.392161	0.395581	0.381	0.432839
WFG6	4	0.582419	0.665516	0.554355	0.606645	0.577935
	7	0.643548	0.760903	0.644677	0.674161	0.688097
	2	0.349645	0.353129	0.400355	0.352548	0.420129
WFG7	4	0.419871	0.554742	0.477	0.57029	0.679839
	7	0.470613	0.567355	0.454387	0.573161	0.718871
	2	0.264129	0.266677	0.278548	0.266742	0.384129
WFG8	4	0.344516	0.433516	0.373484	0.519742	0.605903
	7	0.420097	0.480935	0.421742	0.603	0.555323
	2	0.351419	0.354774	0.370484	0.353194	0.376129
WFG9	4	0.470258	0.558581	0.46029	0.552032	0.529
	7	0.450258	0.599323	0.446935	0.586323	0.509323

如表 2-4 所示,MOEA/D-IHO 除去 WFG8-2 和 WFG9-2,在 2 目标 WFG 测试 函数问题上所得平均 PD 性能指标为零,与 Median Attainment Surface 和表 2-2 中 平均 HV 性能指标统计结果相悖,因此验证 PD 在评价 2 目标多样性时表现不稳定。 MOEA/D-IHO4 目标和 7 目标 WFG 测试函数问题上所得平均 PD 性能指标表现优异。

表 2-4 5 种算法在 WFG2 至 WFG9 测试问题上所得平均 PD 值的比较

	目			算法		
测试函数	标	NSGA-II	GrEA	MOEA/D-DE	НурЕ	MOEA/D-IHO
	2	3.004E-01	2.141E+01	0	1.550E+00	0
WFG2	4	5.527E+03	9.003E-01	6.978E+01	1.446E+03	6.758E+04
	7	4.036E+05	2.281E+06	0	2.123E+05	1.549E+05
	2	2.009E+00	1.011E-01	2.913E+00	1.162E+01	0
WFG3	4	6.668E+03	5.471E+04	0	3.934E+02	2.660E+04
	7	3.047E+06	2.426E+05	0	1.323E+05	3.085E+06
	2	9.443E-01	1.667E+02	0	6.086E+00	0
WFG4	4	3.398E+04	3.427E+04	6.198E+04	6.279E+03	0
	7	5.447E+06	1.811E+07	6.993E+04	1.381E+06	3.521E+06
	2	0	4.071E+01	7.148E-01	1.260E+01	0
WFG5	4	9.612E+01	2.572E+04	0	3.082E+02	0
	7	1.507E-01	2.118E+07	1.240E+07	8.971E+04	3.698E+06
	2	1.190E+00	1.227E+02	1.811E-01	9.436E+00	0
WFG6	4	3.209E-01	3.601E+03	3.233E+03	3.538E+03	5.027E+03
	7	0	1.320E+06	5.512E+04	2.082E+06	1.467E+04
	2	1.709E+00	2.870E+01	2.884E-01	8.070E+00	0
WFG7	4	1.685E+04	3.337E+04	1.323E+01	4.213E+03	9.553E+04
	7	6.930E+06	2.647E+07	0	3.676E+05	2.519E+06
	2	6.814E-01	1.445E+02	0	4.752E+00	1.996E-03
WFG8	4	2.549E+04	1.046E+04	0	4.958E+03	0
	7	9.055E+06	2.106E+07	1.117E+03	7.736E+05	6.485E+06
	2	4.035E-01	6.680E+01	1.288E+00	1.466E+01	1.291E-01
WFG9	4	1.204E+03	2.202E+04	9.323E+04	3.224E+02	0
	7	3.546E+03	2.409E+07	1.313E+06	1.154E+05	4.565E+06

如表 2-5 所示,MOEA/D-IHO 除去 WFG2-2、WFG3-4、WFG3-7 和 WFG9-2,在其他 WFG 测试函数上所得平均 GD 性能指标上表现最好。在 WFG2-2 和 WFG3-4 测试函数上所得平均 GD 性能指标表现较差,劣于其他函数。在 WFG3-7 测试函数上所得平均 GD 性能指标表现居中。在 WFG9-2 测试函数上所得平均 GD 性能指标表现居中。有 WFG9-2 测试函数上所得平均 GD 性能指标表现略差于最优算法 MOEA/D-DE,可以忽略不计。

表 2-5 5 种算法在 WFG2 至 WFG9 测试问题上所得平均 GD 值的比较

加小子云。坐	目			算法		
测试函数	标	NSGA-II	GrEA	MOEA/D-DE	НурЕ	MOEA/D-IHO
	2	0.004955	0.004965	0.00475	0.003701	0.010991
WFG2	4	0.080263	0.040691	0.036647	0.029933	0.016372
	7	0.289916	0.204183	0.052116	0.300758	0.044903
	2	0.007295	0.004114	0.003611	0.002645	0.002572
WFG3	4	0.170954	0.141411	0.142976	0.00503	0.162426
	7	0.428855	0.36449	0.203646	0.005081	0.208219
	2	0.0089	0.008682	0.007367	0.00853	0.001659
WFG4	4	0.086373	0.039016	0.034563	0.025631	0.006866
	7	0.198685	0.140827	0.037047	0.084969	0.020622
	2	0.008113	0.0088	0.006737	0.006807	0.006203
WFG5	4	0.074931	0.027381	0.033069	0.023092	0.012737
	7	0.182132	0.145522	0.048732	0.091241	0.034786
	2	0.005084	0.005755	0.005474	0.005779	0.002698
WFG6	4	0.059011	0.024351	0.019087	0.024593	0.017639
	7	0.191847	0.115112	0.05098	0.084835	0.035532
	2	0.021043	0.019116	0.00678	0.016016	0.006597
WFG7	4	0.092356	0.04708	0.036851	0.024484	0.007797
	7	0.203223	0.154687	0.034559	0.072818	0.030754
	2	0.039735	0.031432	0.037811	0.03607	0.007996
WFG8	4	0.125122	0.083256	0.046558	0.026668	0.015227
	7	0.225277	0.196023	0.037758	0.079089	0.03162
	2	0.016948	0.016553	0.013283	0.017389	0.013854
WFG9	4	0.0668	0.028889	0.024739	0.028743	0.018829
	7	0.213437	0.14315	0.065477	0.106394	0.057415

2.5.3 盒图

盒图用以展示所得数据的离散情况。

如图 2-4 所示,MOEA/D-IHO 在 2 目标 WFG 测试集 HV 盒图上表现最优。

如图 2-5 所示,MOEA/D-IHO 除去 WFG2-2,在 2 目标 WFG 测试集 IGD 盒图上表现最优。

如图 2-6 所示,MOEA/D-IHO 除去 WFG3-2 和 WFG4-2,在 2 目标 WFG 测试 集 PD 盒图上表现与其他算法相似。

如图 2-7 所示,MOEA/D-IHO 除去 WFG2-2 和 WFG3-2 测试问题在 2 目标 WFG 测试集 GD 盒图上表现最优。

如图 2-8 所示,MOEA/D-IHO 在 4 目标 WFG2、WFG4、WFG5、WFG7 和 WFG8 测试集 HV 盒图上表现最优。在其他测试函数 HV 盒图上表现与最优算法之间的差距也并不大。

如图 2-9 所示,MOEA/D-IHO 除去 WFG3-4 测试问题在 4 目标 WFG 测试集 IGD 盒图上表现最优。

如图 2-10 所示,MOEA/D-IHO 在 4 目标 WFG2 和 WFG6 测试集 PD 盒图上表现最优。

如图 2-11 所示,MOEA/D-IHO 除去 WFG3-4 测试问题在 4 目标 WFG 测试集 GD 盒图上表现最优。

如图 2-12 所示,MOEA/D-IHO 在 7 目标 WFG2、WFG4、WFG7 和 WFG8 测试问题上 HV 盒图上表现最优。在其它测试问题盒图上表现也具有竞争力。

如图 2-13 所示,MOEA/D-IHO 除去 WFG3 测试问题在 7 目标 WFG 测试集 IGD 盒图上表现最优。

如图 2-14 所示,GrEA 在 7 目标 WFG 测试集 PD 盒图上综合表现最优。 MOEA/D-IHO 在大部分问题上都处于中间水平。

如图 2-15 所示,MOEA/D-IHO 除去 WFG3-7 测试问题在 7 目标 WFG 测试集 GD 盒图上表现最优。

结合 2.5.2 节中的 5 个算法在 4 个性能评价指标平均值的统计情况,对 MOEAs 在性能函数上的表现往往需要结合多重信息综合进行评定,仅依靠一种评价方式 很难体现优化算法的真实性能。2.5.2 节侧重 31 次运算结果的平均值,而本节侧重 31 次运算结果的离散分布情况,鉴于 MOEAs 寻优过程本质上是一种概率搜索,因此需要结合不同统计结果对不同 MOEAs 进行评价。

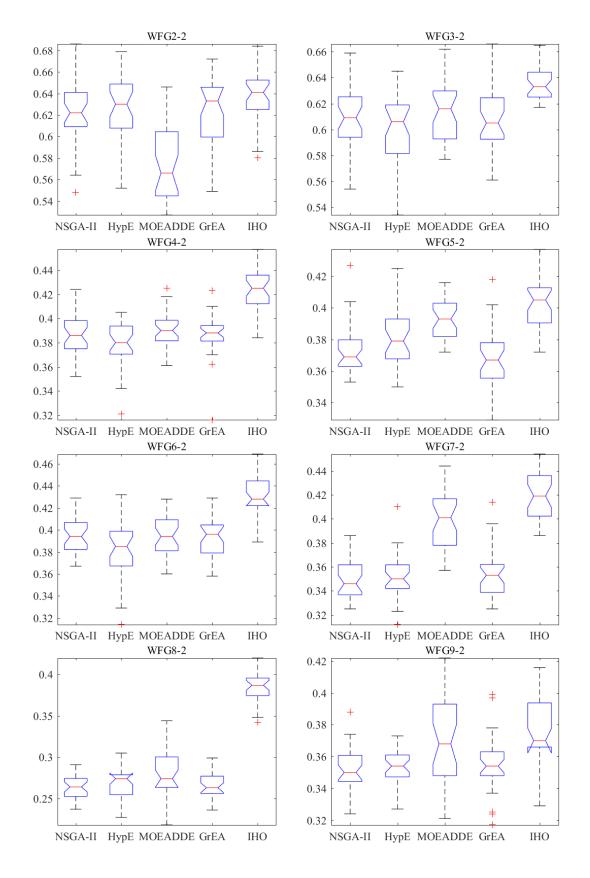


图 2-4 2 目标 WFG 测试集 HV 盒图

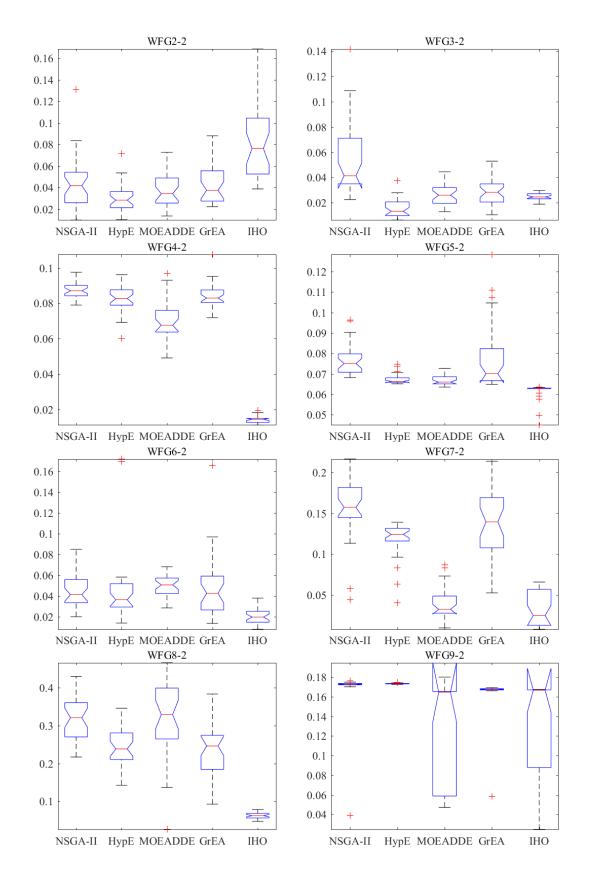


图 2-5 2 目标 WFG 测试集 IGD 盒图

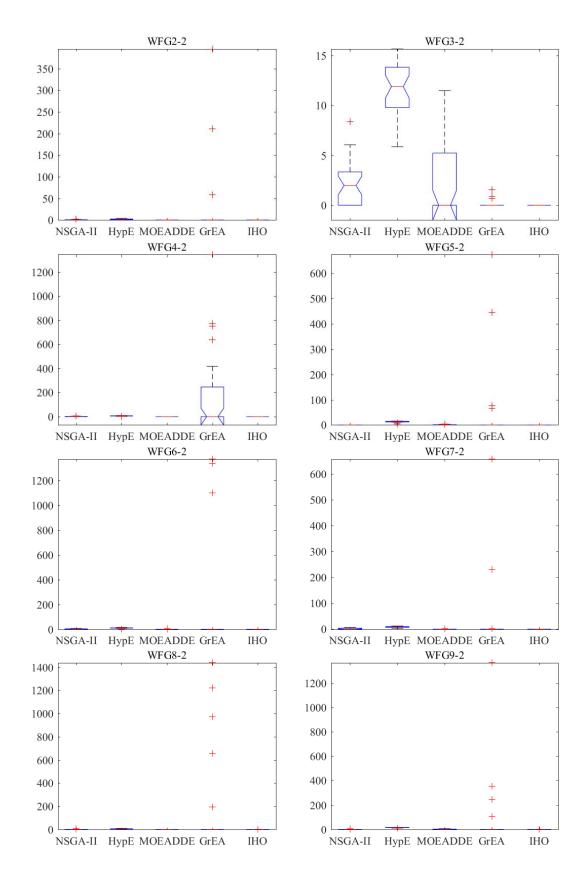


图 2-62目标 WFG 测试集 PD 盒图

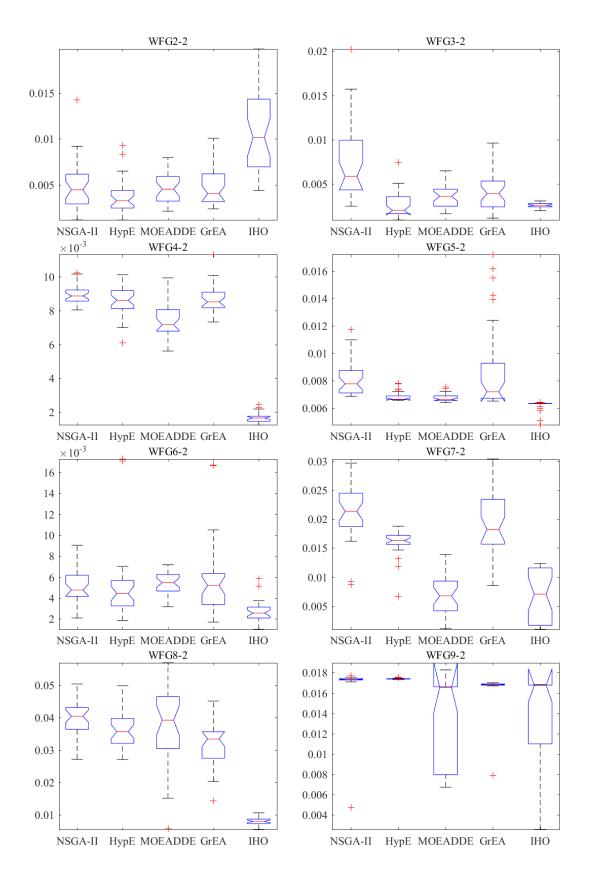


图 2-72目标 WFG 测试集 GD 盒图

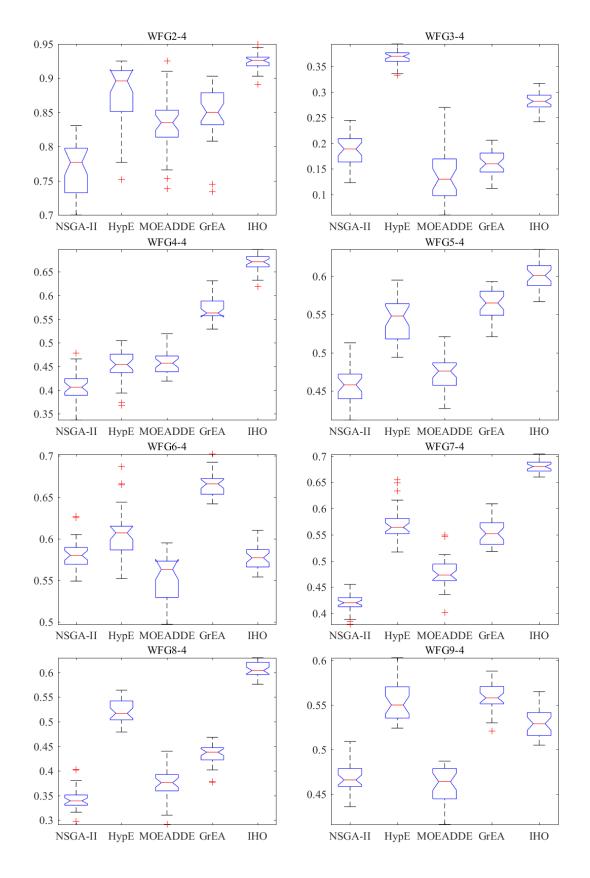


图 2-8 4 目标 WFG 测试集 HV 盒图

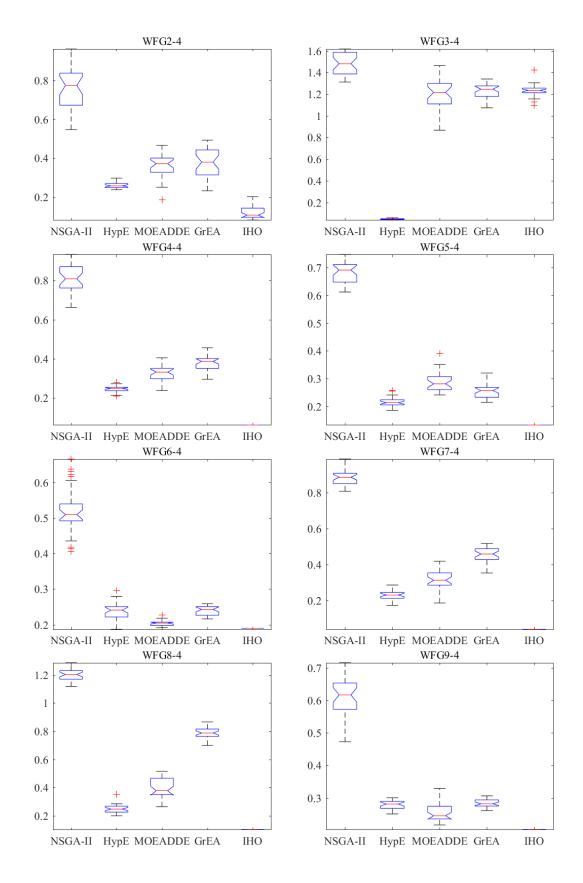


图 2-9 4 目标 WFG 测试集 IGD 盒图

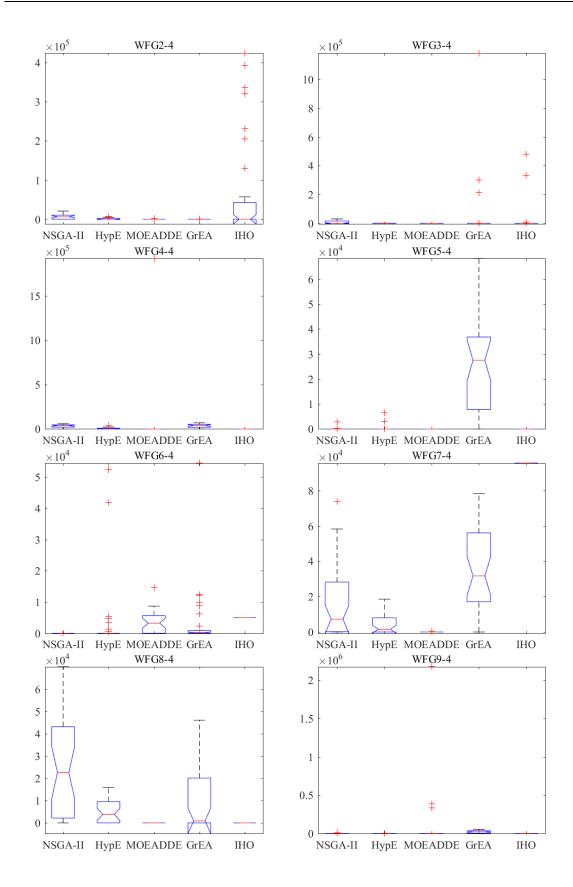


图 2-10 4 目标 WFG 测试集 PD 盒图

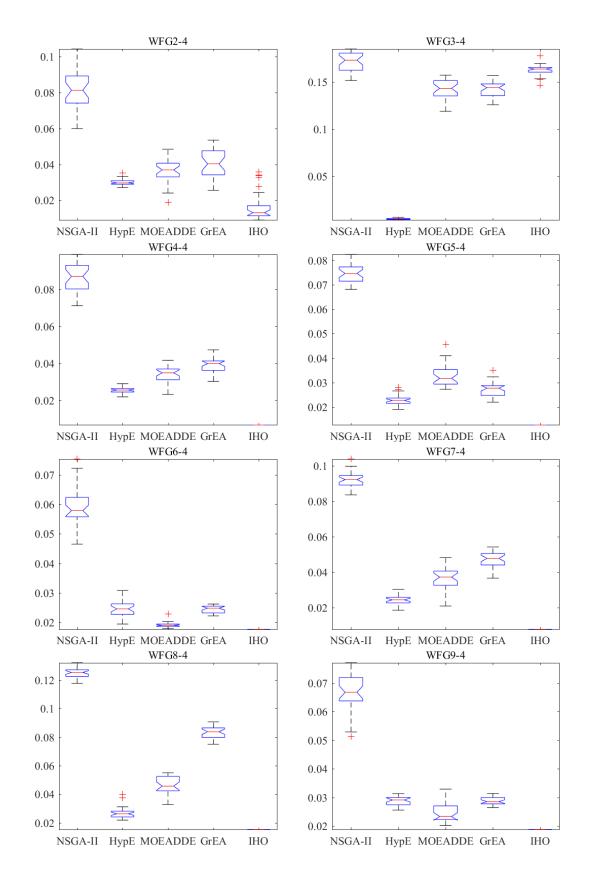


图 2-11 4 目标 WFG 测试集 GD 盒图

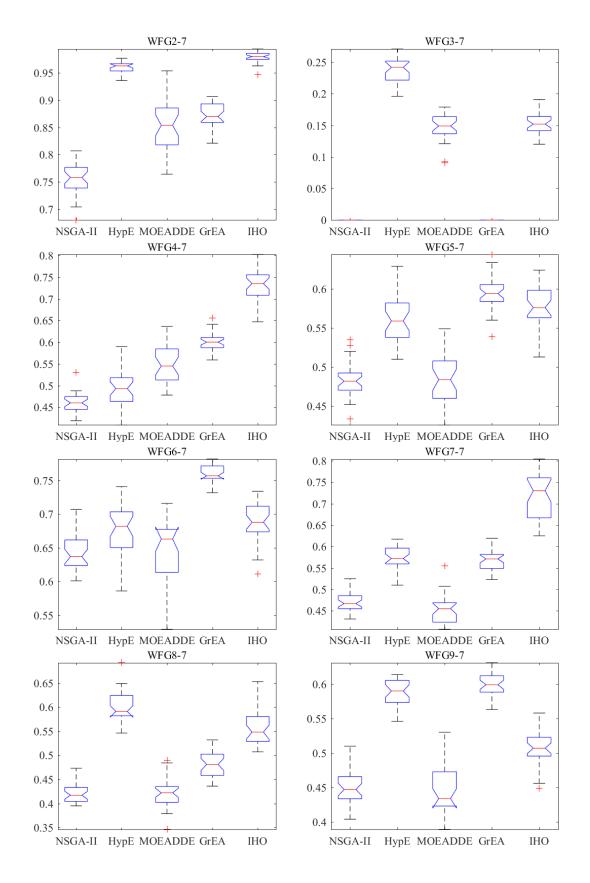


图 2-12 7 目标 WFG 测试集 HV 盒图

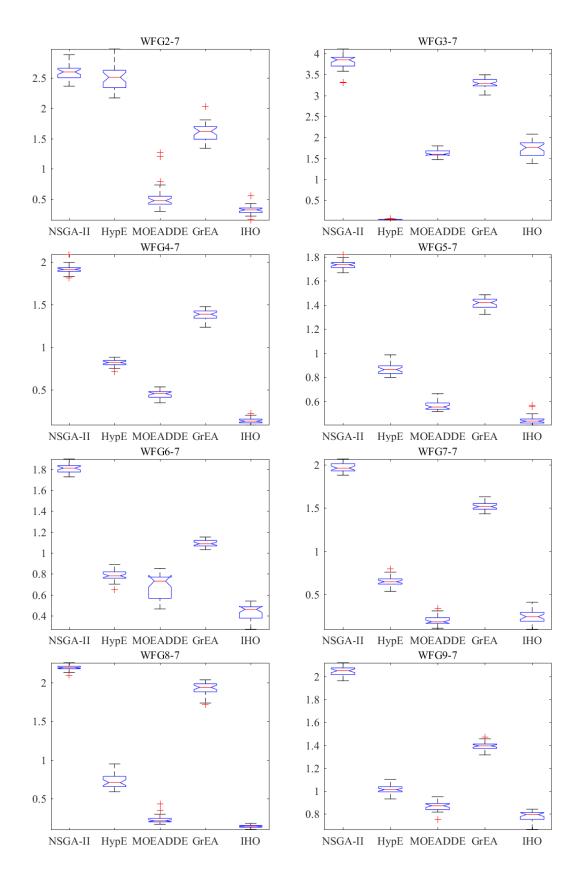


图 2-13 7 目标 WFG 测试集 IGD 盒图

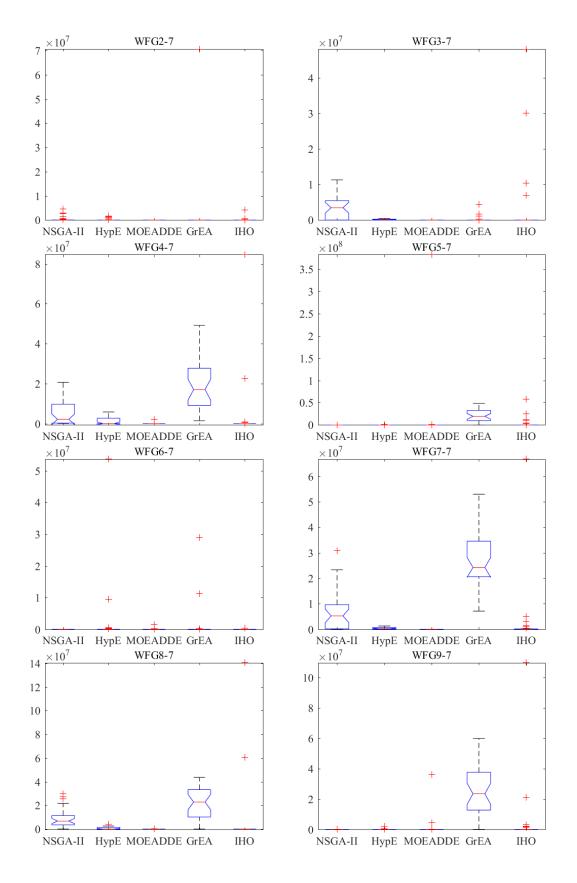


图 2-14 7 目标 WFG 测试集 PD 盒图

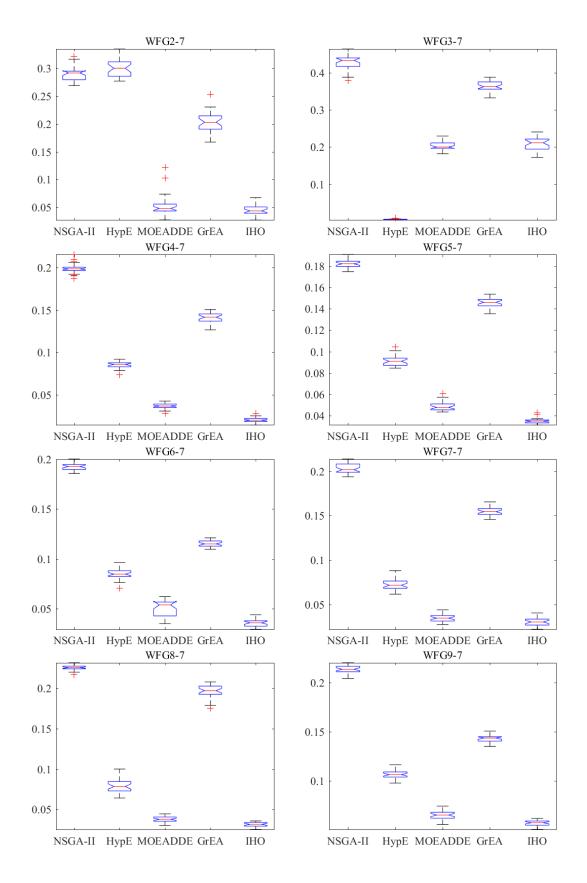


图 2-15 7 目标 WFG 测试集 GD 盒图

2.5.4 结果分析

结合 2.5.1 至 2.5.3 节性能指标对比情况,可以得出 MOEA/D-IHO 除去 WFG3-2 测试问题在 2 目标 WFG 测试函数集所有性能指标盒图对比中基本表现最优。在 4 目标、7 目标 WFG 测试函数集中性能指标 *IGD* 和 *GD* 盒图对比中也一直表现优异,然而在 4 目标、7 目标 WFG 测试函数集中性能指标 *HV* 盒图对比中表现时好时坏,而在 4 目标、7 目标 WFG 测试函数集中性能指标 *PD* 表现一般。这也一定程度上验证了基于 MOEA/D 的算法框架,在求解高维多目标优化问题时急需在多样性表现上深入研究。

2.6 本章小结

本章根据 EAs 中变化算子对制约其在搜索空间中平衡探索与开发的问题,提出了一种基于混合性能指标综合评价机制的变化算子自适应选择策略 IHO,根据历史优化解在 3 种不同性能指标下进行综合评定,择优选取相对应的变化算子在搜索空间中强化探索或开发。同时,根据 DE 算子基本构建框架,将一种新型探索型单目标变化算子乌鸦搜索算法扩展到多目标维度。以目前综合性能表现优异的MOEA/D 为基本框架,对 MOEA/D-IHO 算法在 2 目标、4 目标和 7 目标 WFG 测试集性能指标 HV、IGD、PD,GD与其他四种基于帕累托支配基本框架、基于松弛支配基本框架、基于性能指标基本框架和基于分解基本框架的优秀 MOEAs 进行对比,验证了本章提出的 IHO 策略在平衡搜索空间探索与开发问题上性能优异。

第三章 基于自适应权值向量的分解多目标进化算法

许多学者指出 MOEA/D 在求解 MaOPs 时经常会出现多样性保持不足的情况,第二章提出的 MOEA/D-IHO 为解决该问题提供了一种有效的思路。进一步,本章以 MOEA/D 权值向量为切入点,探究通过修改权值向量提升 MOEA/D 多样性表现的可能。

3.1 引言

MOEA/D 在其原始版本中,认为借助一组在目标空间均匀分布的权值向量最终同样会得到一组在多样性上表现良好的优化解,然而 MOEA/D 设计者 Zhang 等 [111]应用 MOEA/D-DE 求解其提出的 F5 优化问题时发现,不同权值向量的取值会得到不同的结果。如图 3-1 所示, $\mathbf{w} = \{w_1, w_2, w_3, w_4, w_5, w_6\}$ 为一组在目标空间均匀分布的权值向量,然而其所获得的最优解集 $\mathbf{s} = \{s_1, s_2, s_3, s_4, s_5, s_6\}$ 却在多样性上表现较差。

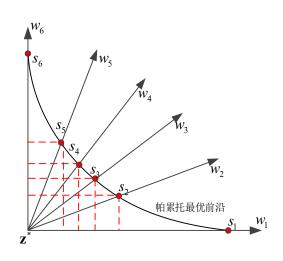


图 3-1 均匀分布权值向量对应不均匀分布优化解

许多学者试图通过重新设计权值向量的生成方式,使优化解能够均匀分布于目标空间,进而提升其多样性。目前为止,比较有代表性的权值向量生成方式包括:单纯形网格法^[175]、双层单纯形网格法^[80]和均匀设计法^[112]。这些方法要想获得分布均匀的优化解,对求解问题本身的帕累托最优前沿有一定的要求,并不适用于求解具有离散、退化或奇异形状的优化问题。鉴于此,现如今在权值向量方面的研究主要以自适应模式为主,根据优化解的稀疏程度,自动增添或删减权值向量以达到提

升多样性的需求。在基于分解的 MOEA 框架研究领域,已有相关学者针对这一问题设计出行之有效的算法,Deb 等[177]在 NSGA-III[80]的基础上提出一种自适应参考点删减的 NSGA-III(A-NSGA-III),通过构建参考点的(m-1)单纯形格,删除拥挤参考点周围与优化解无任何关联的参考点。Qi 等[46]提出一种自适应权值调整策略(MOEAD-AWA),在优化过程中首先假设算法收敛到真实帕累托最优前沿,然后周期性地删除掉拥挤的权值向量,并在稀疏区域添加新的权值向量。Cheng 等[123]通过引入两组参考向量用以分别应对不同的问题。然而这些方式有一定的局限性,有些算的设计专门是为了解决某些问题。同时,Giagkiozis 等[131,178]指出,采用自适应策略生成新的权值向量可能会对整个进化过程产生收敛性不足的问题。鉴于此,Wang 等[179]提出一种基于权值向量和优化解协同进化的方法 PICEA-w,通过计算权值向量领域内优化解的切比雪夫函数值,择优选取优势解和优势向量,共同进入下一次优化过程。然而,其在添加权值向量的过程采用随机生成的方式,并未充分利用当前解信息。因此本文在借鉴 PICEA-w 优势权值向量生成机制的基础上,引入权值向量生成机制,以充分利用生成解在目标空间中的分布情况,反向指导权值向量的增删。

本章首先简单介绍 PICEA-w 的基本原理。其次,根据自适应权值向量生成机制,结合 PICEA-w 协同进化方案提出一种自适应权值向量生成机制(Adaptive Weight Generation,AW)。最后,将 AW 机制与 MOEA/D 相结合形成 MOEA/D-AW,详细介绍 MOEA/D-AW 算法流程并与先进算法的实验对比。

3.2 PICEA-w 权值向量更新机制

3.2.1 PICEA-w 算法基本流程

PICEA-w 基本流程如算法 3.1 所示。算法 3.1 中主要涉及五个函数,如表 3-1 所示。

函数名称	描述
weightGenerator	生成随机权值向量
updateA	根据新获取的非支配解更新外部档案 A
thetaConfiguration	生成函数 coEvolve 所需角度值
coEvolve	生成新的解集与新的权值向量
trimA	限制外部档案 A 总数

表 3-1 PICEA-w 中涉及的函数

算法 3.1: PICEA-w

种群数量N,初始种群S, N_w 个权值向量 $W = \{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_{N_w}\}$,最大迭代次数 $iter_{max}$,目标函数数量M,外部档案A尺寸Asize

输出 S , W , A , BestF

- 1. $BestF \leftarrow \phi$
- 2. $S \leftarrow initializeS(N)$
- 3. $F S \leftarrow obj(S)$
- 4. $BestF \leftarrow updateA(BestF, F S)$
- 5. $W \leftarrow weightGenerator(N_{yy})$
- 6. while $iter < iter_{max}$ do
- 7. $Sc \leftarrow geneticOperator(S)$
- 8. $F _Sc \leftarrow obj(Sc)$
- 9. $(Joint S, Joint F) \leftarrow Union(S, Sc, F S, F Sc)$
- 10. $Wc \leftarrow weightGenerator(N_w)$
- 11. $\theta \leftarrow thetaConfiguration(iter, \pi/2)$
- 12. $(S, F \mid S, W) \leftarrow coEvolve(JointF, JointS, JointW, \theta)$
- 13. $BestF \leftarrow updateA(BestF, F _ S, Asize)$
- 14. end while

表 3-1 中各个函数功能简要描述如下(详尽表述可参见 PICEA-w 原文[179]):

(1) weightGenerator

函数 weightGenerator 主要用以生成 N_w 个权值向量 $\mathbf{w}_j = \{w_1, ..., w_i, w_M\}$, $j = 1, 2, ..., N_w$, \mathbf{w}_i 的函数表达如下:

$$w_{1} = 1 - {}^{M} \sqrt{rand()}$$

$$w_{i} = \left(1 - \sum_{k=1}^{i-1} w_{k}\right) \left(1 - {}^{M} \sqrt{rand()}\right)$$

$$\vdots$$

$$w_{M} = 1 - \sum_{j=1}^{i-1} w_{k}$$

$$(3-1)$$

式中, N_w 为权值向量个数,rand()为 0-1 间的随机数,M 为目标个数。

(2) updateA

函数 updateA 主要用以更新外部档案 A 中的非支配解集,其中当 A 尺寸超过 Asize 时,执行 trimA 以保证 A 尺寸维持在 Asize 。

(3) *trimA*

函数 trimA 根据 Zitzler 等 $^{[68]}$ 提出的 SPEA2 中聚类技术,将 AF 尺寸固定在初始值。

(4) thetaConfiguration

函数 thetaConfiguration 生成一个角度 θ 作为函数 coEvolve 的输入,其函数表达如下:

$$\theta = \frac{\pi}{2} \frac{iter}{iter_{\text{max}}}$$
 (3-2)

式中, iter 为当前迭代次数, itermax 为最大迭代次数。

(5) *coEvolve*

函数 coEvolve 的输入参数为 θ 、JointS、JointF 和 JointW,通过协同进化算法,输出优化解集S、优化目标集 F_S 和优化权重W。JointS 为第 iter 次进化过程中父代解集和子代解集的合集,JointF 为父代解集和子代解集在目标空间映射的合集。JointW 父代优化权值向量和随机权值向量的合集。

函数 coEvolve 中协同进化的基本步骤如下:

- 1) 对于 JointW 中的每个向量 \mathbf{w}_j , $j=1,2,...,2N_w$, 计算得出 \mathbf{w}_j 和 JointF 中 \mathbf{F}_i , i=1,2,...,2N 之间的角度,并得到一个 $2N\times 2N_w$ 的角度矩阵 $\mathbf{\Phi}$ 。当 \mathbf{F}_i 和 \mathbf{w}_j 之间角度小于 $\mathbf{\theta}$, 计 \mathbf{F}_i , i=1,2,...,2N 为解 \mathbf{w}_j 的相邻向量。 $\mathbf{\Phi}$ 中大于 $\mathbf{\theta}$ 值的元素设为 \inf 。计算 \mathbf{F}_i 和 \mathbf{w}_j 的切比雪夫标量化方程,用以评定 \mathbf{F}_i 性能。将 $\mathbf{\Phi}$ 中表征相邻关系的元素替换为切比雪夫标量化值,并形成一个 $2N\times 2N_w$ 的性能矩阵 \mathbf{P} 。 依据 \mathbf{w}_j 对其邻阶 \mathbf{F}_i 按照降序进行排序,生成一个 $2N\times 2N_w$ 顺序矩阵 \mathbf{R} ,其中 1 代表最佳性能。
- 2) 基于顺序矩阵 \mathbf{R} ,选取 $N \wedge \mathbf{F}_i$ 作为新的 F_S 。对 \mathbf{R} 进行升序排序,其中第一列标记为为最优性能,第二列其次,以此类推。选取前 N 个最优性能 \mathbf{F}_i 。
- 3) 基于 \mathbf{R} 选取 N_{w} 个 \mathbf{w}_{j} 作为新的 W。其依据 \mathbf{F}_{i} 挑选使其成为新 $F_{-}S$ 的 \mathbf{w}_{j} ,如果 \mathbf{F}_{i} 同时在多个权值向量下表现最优,选取距离 \mathbf{F}_{i} 最远的 \mathbf{w}_{j} 作为当前群体的最优权值向量。

3.2.2 PICEA-w 算法存在的一些问题

PICEA-w 在每次迭代过程中权值向量通过式 3-1 随机生成,在进化过程中,算法 3.1 步骤 10 产生的随机变量很有可能与上一代进化后的权值向量相重合,并没有充分利用到已知解在目标空间中的分布情况,因此有很大几率会出现多样性不足的情况。为此,本章提出一种自适应向量生成方案,借助算法 3.1 步骤 10 的协同进化过程与权值向量生成机制相结合,以获取算法在目标空间多样性的良好表现。

3.2.3 最优权值向量生成方案

给定一个参考点 $\mathbf{z}_i^* = (\mathbf{z}_1^*, \mathbf{z}_2^*, \cdots, \mathbf{z}_M^*)$,通过求解该参考点的切比雪夫标量化方程,即可获得该参考点下的最优权值向量 $\mathbf{w} = \{\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_M\}$ 。最优权值向量具体计算过程如下:

$$\frac{f_1(s) - \mathbf{z}_1^*}{\mathbf{w}_1} = \frac{f_2(s) - \mathbf{z}_2^*}{\mathbf{w}_2} = \dots = \frac{f_M(s) - \mathbf{z}_M^*}{\mathbf{w}_M}$$
(3-3)

同时,由于 $\mathbf{w}_1 + \mathbf{w}_2 + \cdots + \mathbf{w}_M = 1$,因此最优权值向量计算过程又可以表示为:

$$\mathbf{w} = (\mathbf{w}_{1}, \dots, \mathbf{w}_{M}) = \left(\frac{f_{1}(s) - \mathbf{z}_{1}^{*}}{\sum_{i=1}^{M} f_{i}(s) - \mathbf{z}_{i}^{*}}, \dots, \frac{f_{M}(s) - \mathbf{z}_{M}^{*}}{\sum_{i=1}^{M} f_{i}(s) - \mathbf{z}_{i}^{*}}\right)$$
(3-4)

3.3 基于自适应权值向量的分解多目标进化算法

3.3.1 算法思路

本章提出的基于自适应权值向量的分解多目标进化算法主要借助 PICEA-w 的协同进化算法与最优权值向量生成方案,结合 MOEA/D 基本框架,对 PICEA-w 协同进化算法进行修正,除替换 PICEA-w 中随机权值向量生成机制外, MOEA/D-AW的协同进化策略中依靠当前父代信息对权值向量进行更新,以指导下一次进化过程。

3.3.2 算法详解

算法 3.2 首先生成一组均匀分布的权值向量 $W \leftarrow \{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_N\}$,根据 $\mathbf{w}_i, i = 1, 2, ..., N$ 确定 N 个子问题 p_i ,并组成初始化种群 $P^0 = \{p_1^0, p_2^0, ..., p_N^0\}$,计算 距离权值向量 \mathbf{w}_i 最近的 T 组向量,组成子问题 p_i 的邻域 $\lambda_i \leftarrow \{\mathbf{w}_i, \mathbf{w}_j, ..., \mathbf{w}_T\}$ 。其中,权值向量根据 Das 和 Dennis 的生成方式产生,步骤 3 中初始理想点 z_0^* 的生成主要 依靠当前种群中所有个体目标函数值的最小值进行估计。

AW 策略主要通过算法 3.2 中步骤 7 完成,其中主体内容主要为算法 3.1 步骤 10 到 12 的组合,其中权值向量生成发生在 coEvolve 中,一旦判定解 P_i^{ier} 对应的权值向量无法满足邻域的需求,随机根据式 (3.4) 生成最优权值向量替换原有向量。以更新后的子问题生成子代,执行步骤 8,其中具体生成方法如函数 3.1 和 3.2 所示。其中,解集的多样性通过更新权值向量的方式得以保证,因此本文采取收敛性较好的 DE 算子生成子代。

算法 3.2: MOEA/D-AW

初始化 种群数量N,邻域尺寸T,最大迭代次数 $iter_{max}$

- 1. 生成一组均匀分布权值向量 $W \leftarrow \{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_N\}$
- 2. 根据权值向量W 初始化种群 $P^0 = \{p_1^0, p_2^0, ..., p_N^0\}$, 并求得目标函数 $F P^0 \leftarrow obj(P^0)$
- 3. 根据初始种群 P^0 初始化理想点 z_0^*
- 4. 计算距离权值向量 \mathbf{w}_i 最近的 T 组向量,并组成邻域 $\lambda_i \leftarrow \{\mathbf{w}_i, \mathbf{w}_i, ..., \mathbf{w}_T\}$
- 5. $iter \leftarrow 1$;
- 6. **while** $iter < iter_{max}$ **do**
- 7. $(F \ P^{iter}, P^{iter}, W) \leftarrow coEvolve(F \ P^{iter}, P^{iter}, W)$
- 8. $P^{iter} \leftarrow GenerateOffspring(P^{iter}, DE)$
- 9. $F_P^{iter} \leftarrow obj(P^{iter})$
- 10. end while
- 11. **return** P中所有非支配解

函数 3.1: Generate Offspring

输入: 变化算子DE, 第iter 次迭代时刻种群 P^{iter}

输出: Piter

- 1. **for** $i \leftarrow 1:N$ **do**
- 2. 从 p_i^{iter} 的邻域内随机选择三个个体,通过 DE 算子生成子代 u_i^{iter} 。
- 3. $P^{iter} \leftarrow Update(u_i^{iter}, p_i^{iter})$
- 4. end for

函数 3.2: Update

输入: 子代, u_i^{iter} , 子问题, p_i^{iter}

输出: 子问题, p_{\cdot}^{iter}

- 1. 根据 p_i^{iter} 的权值向量 $\mathbf{w}_i = \{w_1, w_2, ..., w_M\}$, 计算 u_i^{iter} 以及 p_i^{iter} 中其他解的切比雪夫函数值, $f_{Tch}\left(u_i^{iter}\right)$, $f_{Tch}\left(nox_{i,k}^{iter}\right)$, $nox_{i,k}^{iter} \in p_i^{iter}$, $k = \{1, 2, ..., T\}$
- 2. **for** $k \leftarrow 1:T$ **do**
- 3. **if** $f_{Tch}(u_i^{iter}) < f_{Tch}(nox_{i,k}^{iter})$ **then**
- 4. $nox_{i,k}^{iter} \leftarrow u_i^{iter}$
- 5. end if
- 6. end for

3.4 实验设计

3.4.1 测试函数集

本章主要采用 Huband 等提出的 2 目标、4 目标和 7 目标 WFG2-WFG9 测试函数集,其各个函数的特性如表 3-2 所示。

测试函数	目标函数	可分性	多峰性	偏差	几何形状
WE 62	$f_{_{1:M-1}}$	不可分	单峰	多项式,平滑	
WFG2	$f_{\scriptscriptstyle M}$	不可分	多峰	多项式,平滑	凸,不连续
WFG3	$f_{_{1:M}}$	不可分	单峰	多项式, 平滑	线型,退化
WFG4	$f_{_{1:M}}$	可分	多峰	多项式, 平滑	Щ
WFG5	$f_{_{1:M}}$	可分	欺骗性	多项式, 平滑	Щ
WFG6	$f_{_{1:M}}$	不可分	单峰	多项式, 平滑	Щ
WFG7	$f_{_{1:M}}$	可分	单峰	参数依靠	Ш
WFG8	$f_{_{1:M}}$	不可分	单峰	参数依靠	Ш
WFG9	$f_{_{1:M}}$	不可分	多峰,欺骗性	参数依靠	凹

表 3-2 WFG 测试函数集函数特性

其中,WFG 的决策变量区间为 $z_i = [0,2i], i = 1,2,...,n$ 。

3.4.2 性能评价指标

目前 EMO 领域应用最为广泛的评价指标 HV 和 IGD 被用作本章的主要评价指标 PV 和 PV 的计算需要借助参考点信息,一般采取最差点的 1.2 倍作为参考点。同时辅以 PV 可以一种的性能指标,指标 PV 用以评价 MOEAs 求解标准测试集的收敛性。指标 PV 不仅可以在广度上对 MOEAs 进行评价,同时能够保证对分布均匀性进行度量,被认为是目前评价多样性最为有效的指标。

3.4.3 对比算法

为了评估本章所提出的 MOEA/D-AW 算法的性能,将其与以下四种用于解决 MaOPs 的先进算法进行比较,分别为: RVEAa^[123]、A-NSGA-III^[177]、PICEA-g^[84]和 EFR-RR^[124]。

Cheng 等^[123]在参考向量指引进化算法(Reference Vector Guided Evolutionary Algorithm, RVEA)的基础上,针对 RVEA 求解不规则帕累托最优前沿面问题时性能出现退化的情况,提出了一种参考向量自适应更新(Regenaration)策略, RVEAa。

RVEA 通过预设一组分布均匀的参考向量将目标空间划分为与参考向量等数量的子空间。同 NSGA-II 中的精英策略相同,在每一次进化过程中,RVEA 采取将父代与子代结合的方式,择优生成下一代。RVEA 通过引入角度惩罚距离(Angle-Penalized Distance,APD)以平衡算法在收敛性与多样性上的表现,并且 APD 的计算与进化次数相关,以保证 RVEA 在进化初期倾向于收敛性,在进化末期倾向于多样性。RVEA 在 RVEA 的基本框架下,在根据参考向量划分群体生成 N 组子群后,将空子群相对应的参考向量替换为介于当前种群目标函数最大值与最小值的随机单位向量,非空子群相对应的参考向量得以保留。

Jain 和 Deb^[177]在 NSGA-III^[80]的基础上通过自适应增添和删减参考点的方式,提出了 NSGA-III 自适应版本 A-NSGA-III。同 NSGA-III 相似,NSGA-III 首先将关键非支配层前面所有非支配层的个体归入下一代种群,主要对关键非支配层个体进行选择。在选择之前,根据下一代种群中解到参考线的距离将下一代种群中个体与参考点进行关联。根据参考点 j 已关联解个数 ρ_j ,优先选择关联解个数最小的参考点 j,并从关键层中选取距离 j 最近的解加入到下一代种群。其中,当 ρ_j = 0,如果在关键非支配层中无法找到距离参考点 j 距离最小的解,则将参考点 j 删除。而 A-NSGA-III 在 NSGA-III 的基础上,在 ρ_j > 1 的参考点附近根据目标函数数量增加相应数量的参考点,并删除此时所有参考点中 ρ_j = 0 的参考点,以期达到 ρ_j = 1 的理想状态,进而使优化解能够均匀分布于帕累托最优前沿面。

Wang 等^[84]在基于偏好的协同进化算法^[180](Preference Inspired Coevolutionary Algorithm,PICEA)的基础上,通过引入目标向量的形式,提出了基于目标向量偏好的协同进化算法(Preference Inspired Coevolutionary Algorithm Using Goals,PICEA-g)。PICEA-g 借助当前种群中目标函数取值范围,设置一组目标向量将目标空间进行划分,将传统帕累托支配关系和基于目标向量的适应度计算公式相结合,同 NSGA-II 类似,同样采取将父代与子代相结合的方式,对该种群进行适应度分配,并择优选取子代。PICEA-g 目前在各个 WFG 测试函数的对比试验中表现优异,相较其他算法性能表现总是处于前两位^[181]。

Yuan 等^[124]在集成适应度排序法^[182](Ensemble Fitness Ranking,EFR)的基础上引入排序限制策略,归纳为 EFR-RR 法(Ensemble Fitness Ranking with Ranking Restriction Scheme)。在 EFR 中,解集的适应度仅仅依赖于聚合函数值,然而这种选择机制会选取与权值向量相距甚远的解,使算法在多样性保持上表现较差。鉴于此,EFR-RR 提出一种限制排序的策略,通过限制解只在与其相近的权值向量构成的适应度函数上进行排序,进而借助均匀分布的权值向量生成一组分布性表现良好的优化解。

RVEAa、A-NSGA-III 和 PICEA-g 的参考向量(其中 PICEA-g 为目标向量)根据当前种群自适应进行调整,以获取多样性表现较好的优化解。EFR-RR 通过限制适应度排序只发生在解与其邻近的适应度函数上以获得均匀分布的优化解。

3.4.4 实验设置

本章数值实验中涉及到的具体参数设置如下:

- (1) 群体规模: *N*: 对于 2、4、7目标 WFG 测试集优化问题, A-NSGA-III、RVEAa、PICEA-g 和 EFR-RR 的 *N* 全部设置为 100, MOEA/D-AW 的 *N* 分别设置为 100、126 和 210。
 - (2) RVEAa 中惩罚变化率 $\alpha = 2$,参考向量调整频率 fr = 0.1。
 - (3) PICEA-g 中目标向量个数 $N_{goal} = N*m/2$, m 为目标数量。
 - (4) EFR-RR 中邻近权值向量个数 K=2。
 - (5) WFG 测试集中k = 18, l = 14。
- (6) 为公平起见,所有算法均采用模拟二项式交叉和多项式变异变化算子,模拟二项式交叉中 pc=1.0 , $\eta_c=15$,二项式变异中 $\eta_m=20$, $p_m=1/32$ 。
 - (7) 所有算法独立运行 31 次,每次运行迭代 250 次。 此外,在性能指标的计算时,所有非支配解的个数都设置为 100。

3.5 实验结果与分析

为了检测 MOEA/D-AW 的有效性,所有对比算法独立运行 31 次。计算结果分别从 Median Attainment Surface、数值统计和盒图三个方面展开。绘制所有对比算法运行 31 次后获得的近似帕累托最优前沿面的 Median Attainment Surface,如图 3-2 和 3-3 所示。每个算法求解得到 HV、IGD、GD 和 PD 四个性能评价指标计算值的统计结果,以各指标平均值作为对比基准,分别列于表 3-3 至 3-6 中。绘制了每个算法在 HV、IGD、GD 和 PD 四个性能评价指标计算值的盒图。

其中,在所有图例中,ANSGAIII 代表 A-NSGA-III,PICEAg 代表 PICEA-g,EFRRR 代表 EFR-RR,AW 代表算法 MOEA/D-AW。每个示例中 WFGX-Y 代表 Y 个目标的测试函数 WFGX, $X \in \{2,3,...,9\}$ 。

3.5.1 Median Attainment Surface

Median Attainment Surface 直观地将各个算法优化解在目标空间中进行对比,同时辅以每个测试函数的真实帕累托最优前沿,便于查看各个算法优化解在目标空间的多样性和收敛性情况。从图 3-2 可以看出:

- (1) 在离散型凸函数 WFG2 问题的求解上,所有算法都不能收敛到真实帕累 托最优前沿面。RVEAa 的收敛性表现最差,所有算法的多样性表现接近。
- (2) 在线性凸函数 WFG3 问题的求解上,所有算法都有较好的收敛性,其中 PICEA-g 有一定的波动性,收敛性略差。在多样性表现上,PICEA-g 和 MOEA/D-AW 要优于其它算法。
- (3) 在多峰可分凹函数 WFG4 问题的求解上, MOEA/D-AW 在收敛性和多样性上都优于其它算法。
- (4) 在带有欺骗性可分凹函数 WFG5 问题的求解上,所有算法都有较好的收敛性,PICEA-g 有一定波动型,收敛性略差。在多样性表现上,PICEA-g 和 MOEA/D-AW 要优于其它算法。

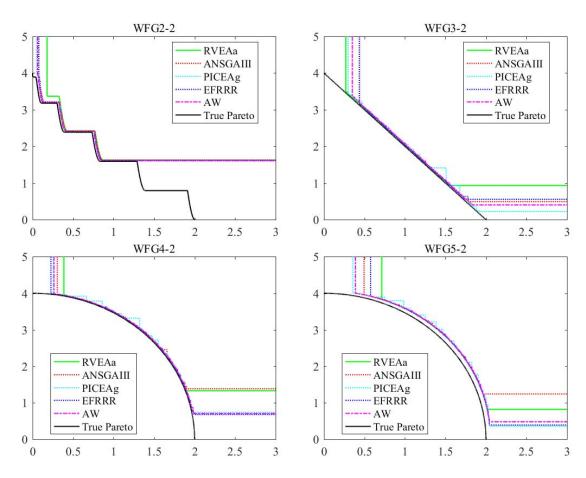


图 3-2 2 目标 WFG2 至 WFG5 测试函数 Median Attainment Surface

从图 3-3 可以看出:

(1)在不可分单峰凹函数 WFG6 问题的求解上,所有算法都有较好的收敛性,其中 PICEA-g 有一定的波动性,收敛性略差。在多样性表现上,PICEA-g 和 MOEA/D-AW 要优于其它算法。

- (2) 在可分单峰凹函数 WFG7 问题的求解上,所有算法都不能收敛到真实帕累托最优前沿面。EFR-RR 在多样性表现上最优。
- (3) 在不可分单峰凹函数 WFG8 问题的求解上,所有算法都不能收敛到真实帕累托最优前沿面,EFR-RR 和 RVEAa 的收敛性较好。
- (4) 在带有欺骗性多峰不可分凹函数 WFG9 问题的求解上,所有算法都有较好的收敛性, PICEA-g 有一定波动型,收敛性略差。在多样性表现上, PICEA-g 和 MOEA/D-AW 要优于其它算法。

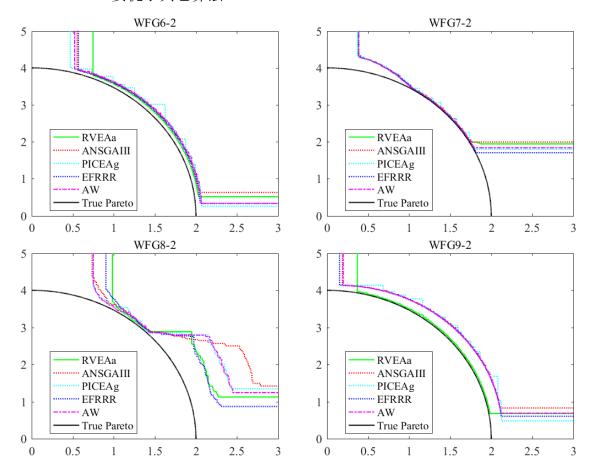


图 3-3 2 目标 WFG6 至 WFG9 测试函数 Median Attainment Surface

3.5.2 实验数值统计结果

由于所有算法从本质上讲都是一种随机搜索方法,单靠一次的运行结果很难判定一个算法好坏,因此本节对 31 次独立运算结果进行平均值统计,对比每个算法在 4 个不同性能评价指标上的表现。其中每种问题中表现最好的算法都进行加粗处理。

如表 3-3 所示, PICEA-g 在大部分测试函数问题上所得平均 HV 性能指标优于

其它算法,但是优势并不明显。MOEA/D-AW 在大部分 4 目标和 7 目标 WFG 测试函数上优于其他算法或与最优算法接近。

表 3-3 5 种算法在 WFG2 至 WFG9 测试问题上所得平均 HV 值的比较

测试	目			算法		
函数	标	ANSGAIII	RVEAa	PICEAg	EFRRR	MOEA/D-AW
	2	0.5683871	0.5500968	0.5739032	0.5643871	0.5702581
WFG2	4	0.7934194	0.764	0.8054839	0.8194839	0.8082581
	7	0.8746774	0.8352903	0.8840323	0.8402581	0.8999032
	2	0.5710968	0.5456774	0.5999355	0.5912903	0.5864839
WFG3	4	0.3281935	0.2037097	0.3644194	0.3294839	0.1830645
	7	0.0893548	0	0.2494194	0.0548387	0
	2	0.3740645	0.384	0.4030968	0.4102581	0.4017097
WFG4	4	0.5639032	0.6194194	0.7011935	0.7180323	0.7030323
	7	0.8199355	0.7131935	0.6676774	0.8089355	0.8238387
	2	0.3583226	0.3581613	0.38	0.3760323	0.3689032
WFG5	4	0.5999677	0.5777419	0.6954194	0.6979355	0.6706774
	7	0.7905484	0.652871	0.7366452	0.7626774	0.7820968
	2	0.3527742	0.3514194	0.3772258	0.3721613	0.3708387
WFG6	4	0.6134839	0.6013548	0.6923226	0.6930645	0.664129
	7	0.7943226	0.6944194	0.8665161	0.7997419	0.7867097
	2	0.3279677	0.3345806	0.3455484	0.3497419	0.3434194
WFG7	4	0.6065806	0.615129	0.7440968	0.7061935	0.7161935
	7	0.845871	0.7185806	0.8206129	0.766	0.8727097
	2	0.2635806	0.245129	0.2549032	0.2653871	0.2746452
WFG8	4	0.5103226	0.4592903	0.5910323	0.5504516	0.5702581
	7	0.7334194	0.6075161	0.7737742	0.6537097	0.7068387
	2	0.356129	0.3681613	0.3618387	0.3717097	0.3656774
WFG9	4	0.5714194	0.5823226	0.5999355	0.5946452	0.6084839
	7	0.6807742	0.6577419	0.6996774	0.6535806	0.7032581

如表 3-4 所示,PICEA-g 在绝大多数 4 目标、7 目标测试函数问题上所得平均 *IGD* 性能指标优于其他算法。MOEA/D-AW 除在 WFG2-7 测试函数问题上所得平均 *IGD* 性能指标优于其他算法,在其他测试问题上表现较差。综合表 3-3 对比结

果,MOEA/D-AW 在大多数测试函数问题上所得平均 *IGD* 性能指标主要受限于其收敛性。

表 3-4 5 种算法在 WFG2 至 WFG9 测试问题上所得平均 IGD 值的比较

测试	目			算法		
函数	标	ANSGAIII	RVEAa	PICEAg	EFRRR	MOEA/D-AW
	2	0.0277422	0.0563067	0.0249234	0.0753122	0.0271739
WFG2	4	0.3189261	0.3191342	0.2444377	0.3978739	0.2655842
	7	2.4785011	2.573738	3.0557009	2.9327871	2.2366478
	2	0.0228255	0.0291779	0.0203973	0.0169032	0.0280808
WFG3	4	0.7068468	0.6187601	0.0185076	0.2263272	1.4919033
	7	0.7939793	1.8655692	0.0085836	0.482924	3.8750395
	2	0.0112434	0.0067703	0.0108588	0.0076598	0.0114086
WFG4	4	0.1441249	0.2072207	0.1192475	0.1264188	0.1587242
	7	0.9685581	1.1217202	0.5526627	0.8014194	1.0555186
	2	0.0680199	0.0646356	0.0649533	0.0648996	0.0683153
WFG5	4	0.2463052	0.211099	0.1530422	0.1562725	0.1783047
	7	0.9916907	1.1705984	0.7270069	0.8763536	1.117266
	2	0.0789296	0.0717543	0.0757306	0.0769455	0.0772058
WFG6	4	0.2947874	0.2691742	0.1706372	0.1645013	0.21404
	7	0.9112308	1.2399054	0.5960213	0.7120275	1.0834489
	2	0.0967214	0.0666666	0.0968906	0.0866432	0.0885118
WFG7	4	0.1834088	0.1851346	0.1139877	0.145567	0.1421494
	7	0.9009015	1.2479099	0.5760978	1.0492375	0.9631205
	2	0.228603	0.267558	0.2398115	0.2928287	0.2446978
WFG8	4	0.4418091	0.6349837	0.2242573	0.5935174	0.3420176
	7	1.1035881	1.7921342	0.5728614	1.5081538	1.4359456
	2	0.1559923	0.1164602	0.124575	0.1231714	0.1269312
WFG9	4	0.3008735	0.2706696	0.2342858	0.2465482	0.2482205
	7	1.3706167	1.5075472	1.0934077	1.3964426	1.3268733

如表 3-5 所示,MOEA/D-AW 在 WFG3-4、WFG3-7、WFG4-4、WFG4-7、WFG5-7和 WFG6-4 测试问题上所得平均 PD 性能指标表现最好。MOEA/D-AW 在 RVEAa 表现最好的测试问题上所得平均 PD 性能指标与 RVEAa 相差不大,处于同一量级。

表 3-5 5 种算法在 WFG2 至 WFG9 测试问题上所得平均 PD 值的比较

测试	目			算法		
函数	标	ANSGAIII	RVEAa	PICEAg	EFRRR	MOEA/D-AW
	2	2.5856E-01	9.9694E-01	9.9272E-03	1.1440E+00	2.4108E-01
WFG2	4	1.5262E+04	1.3787E+04	5.3622E+02	2.8578E+03	5.6991E+03
	7	1.0326E+06	2.8411E+06	2.8449E+04	7.3095E+05	1.5429E+06
	2	3.1062E+00	5.9609E+00	9.3079E-02	8.1628E+00	3.7246E+00
WFG3	4	1.2490E+03	1.1823E+04	2.1369E+01	2.5384E+01	2.1157E+04
W1 03	7	2.3024E+05	6.2118E+06	2.1784E+02	6.2226E+04	9.8409E+06
	2	3.0417E+00	6.1710E+00	2.9812E-02	3.2246E+00	3.0944E+00
WFG4	4	1.1445E+04	4.3697E+04	1.2715E+03	6.2506E+03	4.5500E+04
	7	4.7345E+06	1.3482E+07	1.5931E+03	4.4490E+06	1.4077E+07
	2	3.6109E+00	5.5950E+00	3.2521E-02	3.7871E+00	3.3018E+00
WFG5	4	3.4718E+04	4.4549E+04	1.2287E+03	1.1603E+04	3.9428E+04
	7	9.2095E+06	1.5182E+07	1.9216E+04	6.2197E+06	1.5289E+07
	2	3.2065E+00	4.9179E+00	4.6837E-02	3.7488E+00	2.4938E+00
WFG6	4	3.2313E+04	4.3719E+04	7.1540E+02	2.2820E+03	4.4265E+04
	7	8.0712E+06	1.3226E+07	2.8716E+04	1.7753E+06	1.2286E+07
	2	2.7451E+00	8.0203E+00	2.1564E-02	4.1566E+00	2.1970E+00
WFG7	4	6.3097E+03	5.3304E+04	7.3130E+02	3.3870E+03	4.4569E+04
	7	1.1350E+06	2.0854E+07	3.3026E+04	5.7026E+06	9.4309E+06
	2	3.8185E-01	2.5569E+00	3.0534E-02	1.7046E+00	7.3051E-01
WFG8	4	4.9118E+03	5.5100E+04	4.6357E+02	5.4929E+03	3.9892E+04
	7	1.2470E+06	2.0567E+07	3.4924E+04	4.4069E+06	1.7187E+07
	2	2.5016E+00	5.7800E+00	2.1657E-02	3.4753E+00	3.0163E+00
WFG9	4	4.9685E+04	5.9847E+04	1.0100E+03	5.6476E+03	4.4488E+04
	7	7.8656E+06	2.2798E+07	9.1184E+04	5.5552E+06	1.3535E+07

如表 3-6 所示,PICEA-g 与 RVEAa 有较好的表现。MOEA/D-AW 在 WFG2-4、WFG2-7 和 WFG5-4 测试问题上所得平均 GD 性能指标表现最好,在其他测试问题上 MOEA/D-AW 所得平均 GD 性能指标与最优算法之间的差距不大。

表 3-6 5 种算法在 WFG2 至 WFG9 测试问题上所得平均 GD 值的比较

测试	目			算法		
函数	标	ANSGAIII	RVEAa	PICEAg	EFRRR	MOEA/D-AW
	2	0.0028059	0.0063916	0.0025988	0.010613	0.0027418
WFG2	4	0.0336244	0.0329557	0.025543	0.0566835	0.0219159
	7	0.1830739	0.1899149	0.2304059	0.3316881	0.1615903
	2	0.0023069	0.0029552	0.0020615	0.0017083	0.0028253
WFG3	4	0.0891084	0.0627838	0.0020188	0.033096	0.1260125
	7	0.0665427	0.1382586	0.0007797	0.0564796	0.2610865
	2	0.0011696	0.0007209	0.0011423	0.0008048	0.0011955
WFG4	4	0.0136766	0.0192918	0.0114128	0.0144728	0.0123177
	7	0.0704044	0.0825899	0.0433635	0.0892277	0.0700605
	2	0.0068192	0.0064841	0.0065255	0.0065168	0.0068536
WFG5	4	0.022822	0.0196447	0.0142876	0.0174276	0.0137266
	7	0.0704448	0.0845458	0.053274	0.0945654	0.0731169
	2	0.0079202	0.0072055	0.0076123	0.0077377	0.0077505
WFG6	4	0.0274299	0.0249955	0.0158777	0.0183691	0.0163788
WFG6	7	0.0654238	0.0893581	0.044954	0.0784874	0.0711229
	2	0.014487	0.0124651	0.0156522	0.0145996	0.0143956
WFG7	4	0.0171622	0.0175017	0.0109997	0.0179542	0.0111352
	7	0.0658009	0.0903263	0.0451136	0.1167012	0.0646982
	2	0.0314387	0.0376814	0.0343243	0.0352841	0.0315172
WFG8	4	0.054451	0.0644364	0.02797	0.0772175	0.0314716
	7	0.0804631	0.1283871	0.0444722	0.1703424	0.0956273
	2	0.0160276	0.0124825	0.0132742	0.0135198	0.0133882
WFG9	4	0.0278622	0.0250021	0.021662	0.0272734	0.0189518
ī	7	0.0967642	0.1060724	0.0787347	0.1505702	0.0859494

3.5.3 盒图

盒图用以展示所得数据的离散情况。如图 3-4 所示,MOEA/D-AW 在 2 目标 WFG 测试集 HV 盒图上表现与其他优秀算法相似,表现较优。

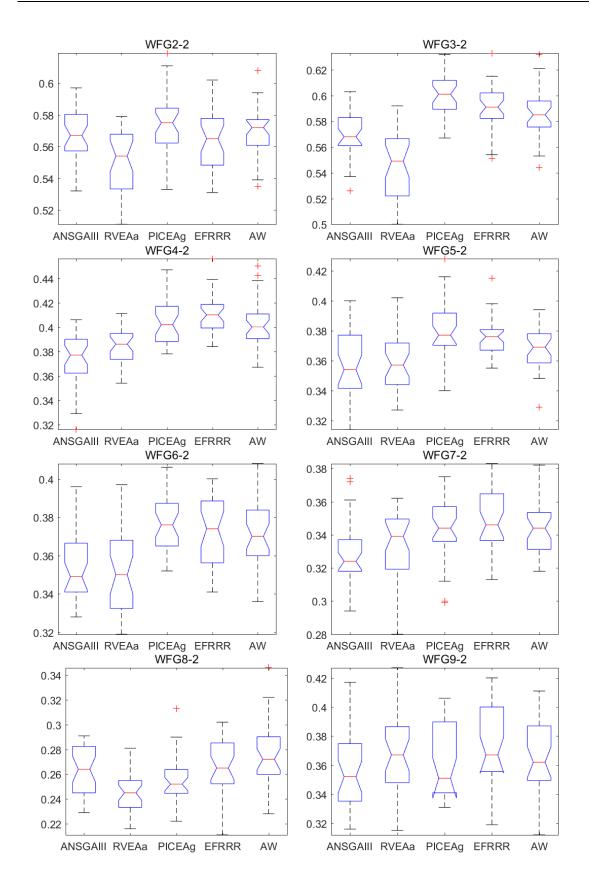


图 3-4 2 目标 WFG 测试集 HV 盒图

如图 3-5 所示,MOEA/D-AW 除去 WFG2-2、WFG4-2、WFG5-2 和 WFG7-2 测试问题,在 2 目标 WFG 测试集 IGD 盒图上表现与其他优秀算法相似。

如图 3-6 所示,MOEA/D-AW 在 2 目标 WFG 测试集 PD 盒图上表现劣于最优算法,但是差距不大。

如图 3-7 所示,MOEA/D-AW 除去 WFG2-2 测试问题在 2 目标 WFG 测试集 GD 盒图上表现与其他优秀算法相似。

如图 3-8 所示,MOEA/D-AW 在 4 目标 WFG9 测试问题 HV 盒图上表现最优,MOEA/D 在 4 目标 WFG3 测试问题 HV 盒图上表现最差,MOEA/D-AW 在其他 4 目标测试问题 HV 盒图上表现与优秀算法 PICEA-g 和 EFR-RR 相近。

如图 3-9 所示,MOEA/D-AW 除去 WFG3-4 测试问题,在 4 目标 WFG 测试集 IGD 盒图上表现与优秀算法 PICEA-g 相似。

如图 3-10 所示,MOEA/D-AW 在 4 目标 WFG3、WFG4 和 WFG6 测试问题 *PD* 盒图上表现最优,MOEA/D-AW 在其他 4 目标 WFG2 测试问题 *PD* 盒图上表现与 PICEA-g 和 EFR-RR 相似,劣于 A-NSGA-III 和 RVEAa。MOEA/D-AW 在其他 WFG 测试集 *PD* 盒图上表现与最优算法相近。

如图 3-11 所示,MOEA/D-AW 在 4 目标 WFG2、WFG4、WFG5、WFG6、WFG7 和 WFG9 测试问题 GD 盒图上表现最优,在 4 目标 WFG3 测试问题 GD 盒图上表现最差,在 4 目标 WFG8 测试问题 GD 盒图上与最优算法表现相近。

如图 3-12 所示,MOEA/D-AW 在 7 目标 WFG2、WFG4、WFG5、WFG7 和 WFG9 测试问题 HV 盒图上表现最优,在 7 目标 WFG3 测试问题盒图上表现最差 直接为零。

如图 3-13 所示, MOEA/D-AW 除去 7 目标 WFG2 测试问题, 在 7 目标 WFG 测试集 *IGD* 盒图上表现劣于最优算法,与其他算法表现相近。

如图 3-14 所示,MOEA/D-AW 在 7 目标 WFG3、WFG4、WFG5 和 WFG6 测试问题 PD 盒图上表现最优,在其他 WFG 测试问题 PD 盒图上表现处于中间水平,略差于最优算法。

如图 3-15 所示,MOEA/D-AW 除去 7 目标 WFG2 和 WFG9 测试问题,在 7 目标 WFG 测试集 GD 盒图上表现都劣于最优算法,与其他算法表现相近。

结合 3.6.2 节中的 5 个算法在 4 个性能评价指标平均值的表现情况,得出对 MOEAs 在性能函数上的表现往往需要借助多重信息进行综合评定,仅依靠一种评价方式很难体现优化算法的真实性能。

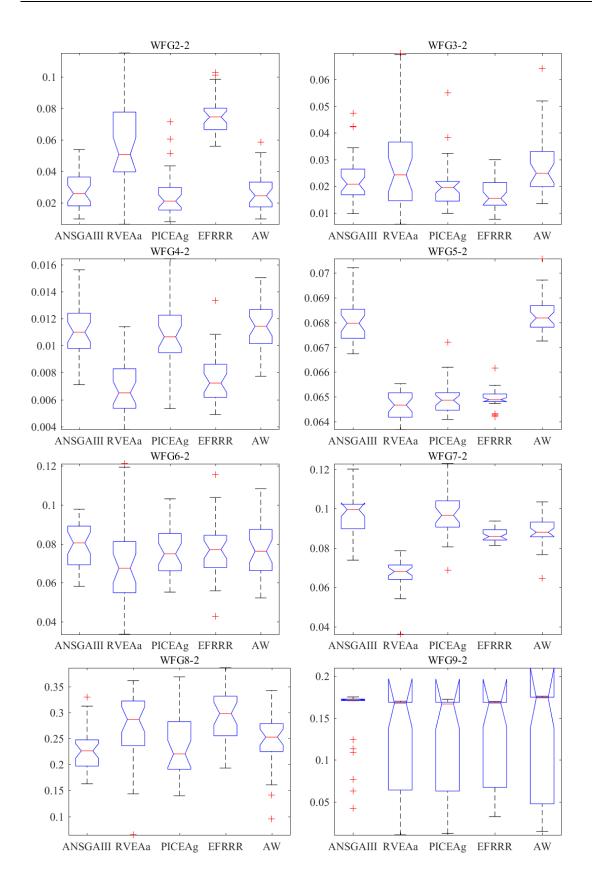


图 3-5 2 目标 WFG 测试集 IGD 盒图

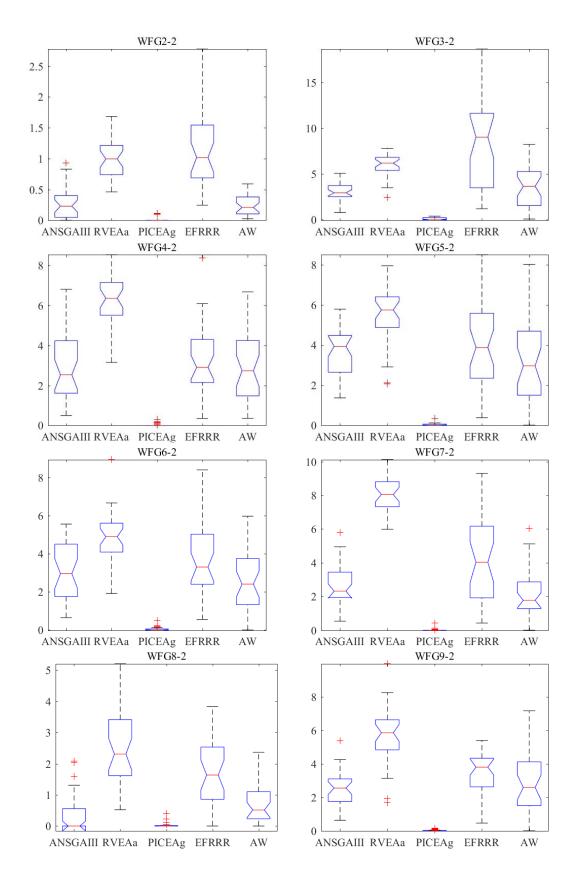


图 3-62目标 WFG 测试集 PD 盒图

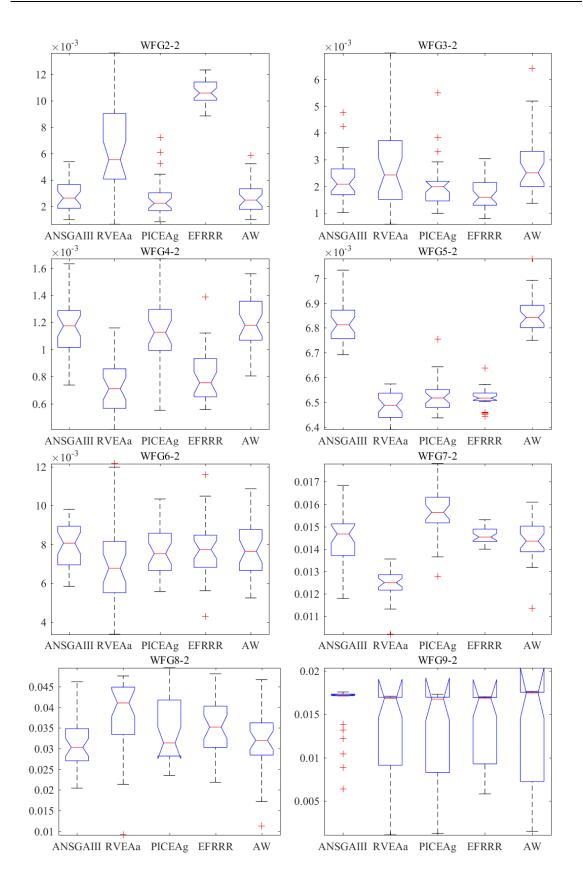


图 3-7 2 目标 WFG 测试集 GD 盒图

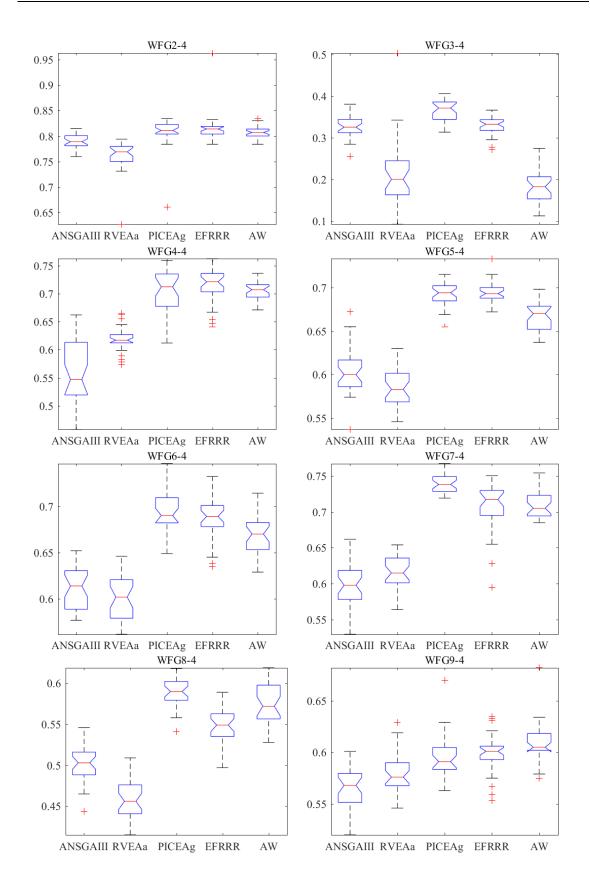


图 3-8 4 目标 WFG 测试集 HV 盒图

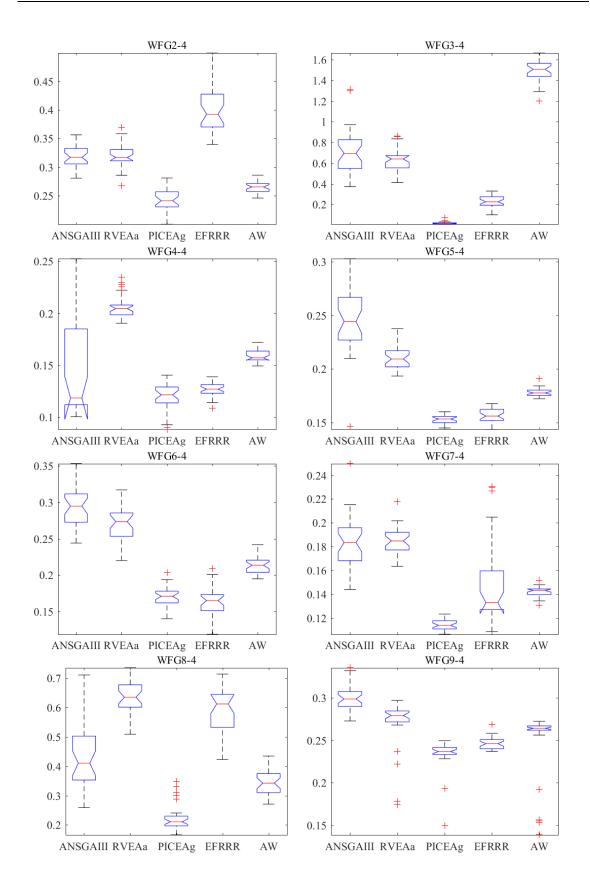


图 3-9 4 目标 WFG 测试集 IGD 盒图

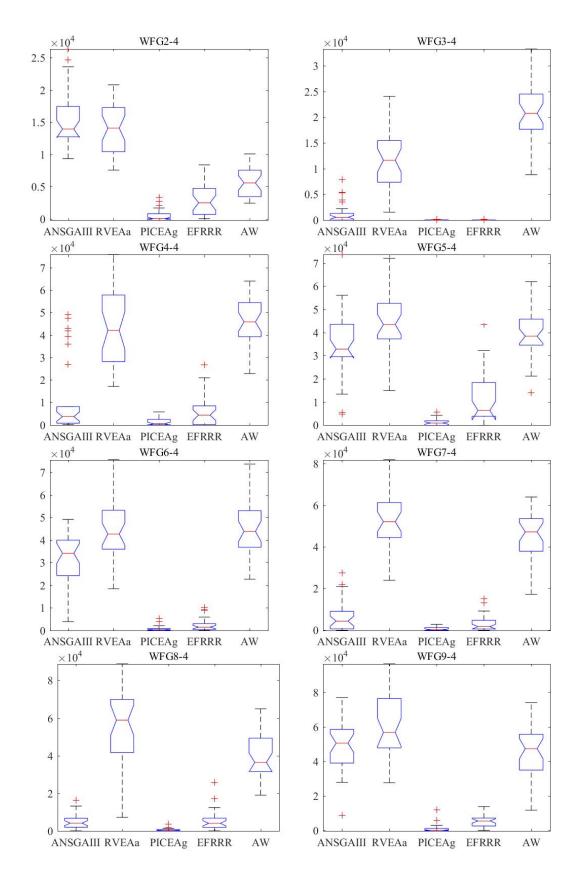


图 3-10 4 目标 WFG 测试集 PD 盒图

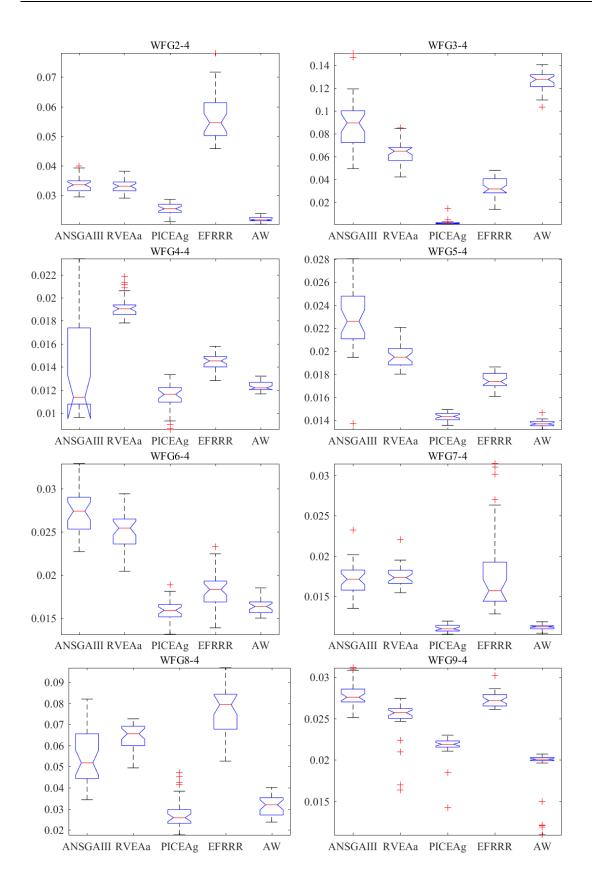


图 3-11 4 目标 WFG 测试集 GD 盒图

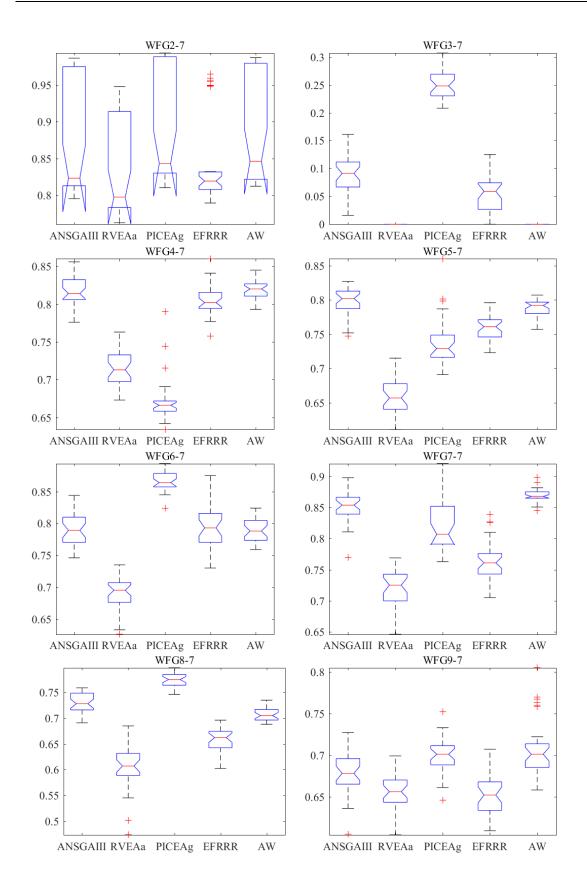


图 3-12 7 目标 WFG 测试集 HV 盒图

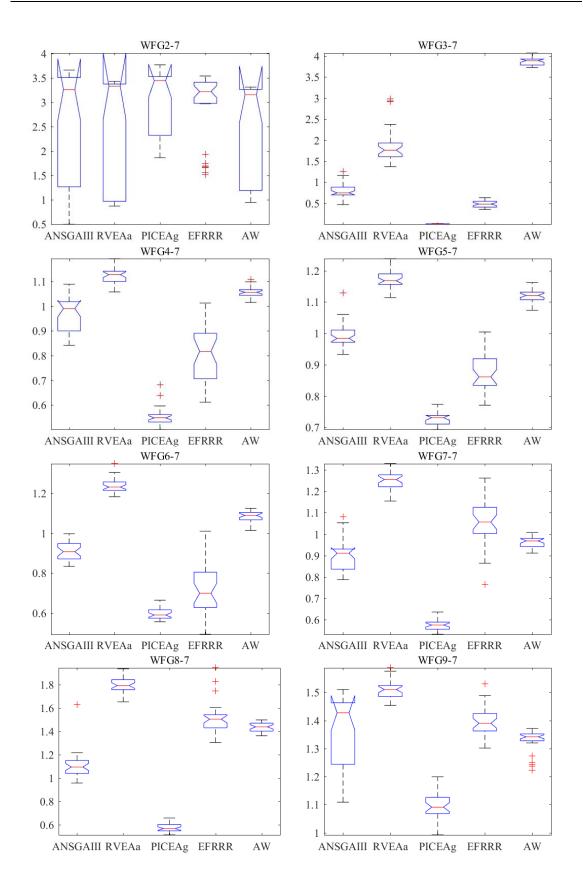


图 3-13 7 目标 WFG 测试集 IGD 盒图

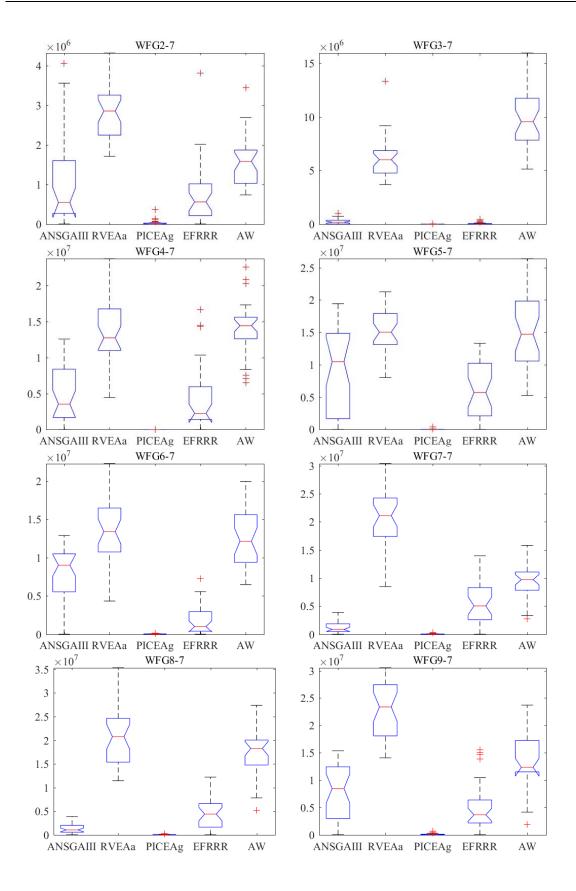


图 3-14 7 目标 WFG 测试集 PD 盒图

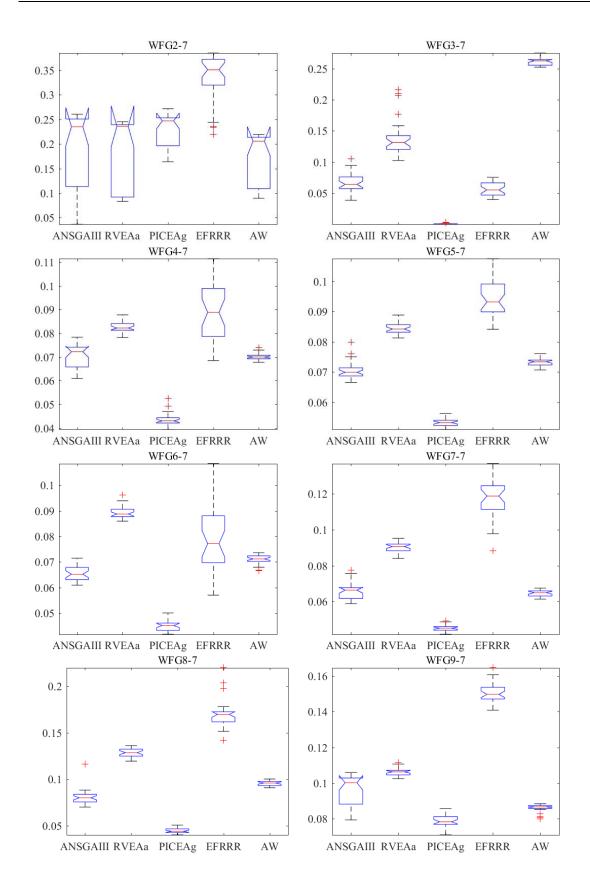


图 3-15 7 目标 WFG 测试集 GD 盒图

3.5.4 结果分析

结合 3.5.1 至 3.5.3 节性能指标对比情况,可以得出 MOEA/D-AW 在 WFG 测试集性能指标盒图对比中,收敛性指标 IGD 和 GD 表现一般,劣于目前优秀算法。在多样性指标 HV 和 PD 上表现较为优异。实验结果验证了 MOEA/D-AW 算法在多样性上的综合表现。

3.6 本章小结

本章从提升 MOEA/D 优化解多样性,尤其从分布均匀性的角度出发,提出了一种基于协同进化的权值向量自适应生成机制。该机制结合当前进化过程中子代和权值向量的综合信息,一方面利用当前子代多样性信息(尤其是目标空间解集分布稀疏程度),共同进化以剔除不可行权值向量,另一方面该机制利用了当前子代收敛性信息(最优点),生成最优权值向量进入到下一次进化过程中。MOEA/D-AW算法与当前优秀 MOEAs 在 2 目标、4 目标和 7 目标 WFG 测试集在性能指标 HV、IGD、PD和GD上的表现进行对比,验证了本章提出的 AW 策略在多样性上表现优异。

第四章 基于满意度的交互式多目标优化算法

工程实际应用中,由于决策者的倾向,最终决策者会从一组多样性保持好的解集中选出最需要的一组解。此时,决策者不再需要尽可能遍布帕累托最优前沿的优化解集,而更倾向于获得目标空间中符合其偏好的一组解。因此便需要决策者的偏好信息来指导整个多目标优化过程。然而,由于 MOP 中的目标之间冲突关系很难直观判定,决策者在搜索之处对 MOP 的认知并不清晰,这就给决策者执行决策带来了一定困难。鉴于此,本章提出一种基于满意度的交互式多目标优化算法,决策者根据不断更新的优化解,结合满意度理论及时调整相应的权值向量以指导整个搜索过程向自己最为满意的方向行进。

4.1 引言

1960 年 Simon^[183]首次提出满意度优化的概念并得出满意解。Takatsu^[184]于1981 年建立了满意度理论的基本数学模型。Goodrich 等^[185]在控制论的深入研究中根据满意度目标函数来进行控制,开创了满意度优化在工程领域的应用。之后满意度优化理论被广泛应用到多个领域并取得一定的成果^[129,186,187]。

满意度优化从某种意义上与基于偏好信息的优化方法类似,都是挑选决策者 更倾向的一组解,但是两者之间存在一定的差异。满意度优化直接对优化解进行区 分,而基于偏好信息的优化方法则对目标函数进行区分,进而指导优化解的生成。 目前,基于偏好信息的优化方法,根据偏好信息在优化过程中的参与方式,主要分 为先验偏好信息、后验偏好信息和交互偏好信息[24,188,189]。先验偏好信息方式通过 不同形式,例如,权重、边界、参考点和词典顺序等,对各个目标的偏好程度进行 表征[17,190]。然而,决策者往往并不能对各个目标直接的偏好信息进行全面的区分, 进而导致所获得的最优解无法令人满意,将会诱发替代性偏好解的产生。后验偏好 信息方式的基本原理是从所获得的帕累托非支配解中根据偏好信息选择偏好的一 组优化解[18,84,189]。然而,从大量的帕累托优化解集中挑选决策者在各个目标之间 不同偏好的解集仍是一个巨大挑战。而交互式偏好信息方式,通过不断学习获得的 优化解,更新对于优化目标之间关系的认知,同时,由于每次只挑选满意解进行下 一次进化,一定程度上节省了计算成本。鉴于交互式偏好信息在集成决策者偏好信 息上的优越性,多位学者针对其展开了深入的研究[126,191-194]。目前,根据偏好信息 构建的不同方式,交互式偏好多目标优化主要分为基于偏好点和基于目标函数分 类两种形式[195]。Thiele 等[82]将决策者偏好信息以效用函数参考点的方式引入,更 新参考点信息后的效用函数被认为可以引导搜索向偏好的方向行进。Miettien等[196] 提出的同步 NIMBUS 算法,通过计算基于相同参考点的不同标量函数,以缓解由不同标量化函数带来的优化解偏差。Deb等[128]根据树方法将决策者信息和通过 PI-EMO-CF 算法获得的部分或完整偏好排序进行合并,进而引导整个寻优过程向帕累托最优前沿面偏好部分行进。上述所有方法的根本落脚点都是在于偏好信息的表征上。本文提出了一种基于满意度理论的交互式多目标进化算法,直接对优化解进行满意度区分,挑选更为满意的优化解进入下一代进化过程。结合第二章提出的MOEA/D-IHO 算法在搜索空间平衡探索与开发的优异表现,将两者相结合,力求能够获得较为满意的结果。

本章首先简单介绍满意度理论的基础知识。其次,提出一种基于满意度的交互式偏好信息表征策略(Interactive Satisfaction,IS)。最后,将 IS 机制与 MOEA/D-IHO 相结合形成 MOEA/D-IS,并详细介绍 MOEA/D-IS 算法流程和与先进算法的实验对比。

4.2 满意度理论

4.2.1 满意度定义

给定可行解集 $X, X \subseteq R^n$,其质量函数如下[187]:

$$f: \mathbf{X} \to Q$$

$$q = f(\mathbf{x}) \in Q, \mathbf{x} \in X$$
(4-1)

式中,f 为质量函数,即目标函数,Q, $Q \in \mathbf{R}^m$ 为质量集,q 为解 \mathbf{x} , $\mathbf{x} \in X$ 的质量。 给定可行解集 \mathbf{X} , $\mathbf{X} \subseteq \mathbf{R}^n$ 与质量集 Q,与之相对应的满意度函数 $h(\cdot)$ 如下:

$$h: Q \to [0, 1]^{m}$$

$$s_{\tilde{F}}(\mathbf{x}) = h(q) = h(f(\mathbf{x}))$$

$$\forall \mathbf{x} \in \mathbf{X}, q \in Q, s_{\tilde{F}}(\mathbf{x}) \in [0, 1]^{m}$$

$$(4-2)$$

式中, \tilde{F} 为与质量函数 f 相对应的解集 $X, X \subseteq \mathbb{R}^n$ 中的质量函数集, $s_{\tilde{F}}(\cdot)$ 为 \tilde{F} 的满意度函数,代表任意解 $x, x \in X$ 的满意度。为了行文方便, \tilde{F} 可写成 $\{X, f, h\}$,其中 f 和 h 分别为质量映射与满意映射。实际应用场景中, $X \times f$ 和 h 为满意度表征中的三个主要元素。

4.2.2 多目标满意度优化

假设决策变量为 $X = \{(x_1, x_2, ..., x_n) | , i = 1, 2, ..., n \}, X \subseteq \mathbf{R}^n$ 。 质量集的函数表达如下:

$$Q = \{(q_1, q_2, ..., q_m) | , i = 1, 2, ..., m \}, Q \subseteq \mathbf{R}^m$$
 (4-3)

式中, m 为质量函数数量, $q_k = f_k(x), k = 1, 2, ..., m$ 。

给定一个满意度函数 $h_k: \mathbf{R} \to [0,1]$ 和 $s_k = h_k(q_k), k = 1,2,...,m$,集合的满意度函数如下:

$$S = (s_1, s_2, ..., s_m) = [h_1(q_1), h_2(q_2), ..., h_m(q_m)], S \subseteq [0, 1]^m, i = 1, 2, ..., m \quad (4-4)$$

总体满意度方程定义为 f_{op} :[0,1]" \rightarrow [0,1]。通常,根据问题的不同特性 f_{op} 的建立有几种不同的形式^[197,198]。本章采用应用最为广泛的线性权值求和法来对满意度进行标定,其函数表达如下:

$$sw = f_{op}(s) = \sum_{i=1}^{m} \omega_i s_i$$
 (4-5)

式中, $\omega_i \in [0,1], \sum_{i=1}^m \omega_i = 1$ 。

因此,多目标满意度优化(Multi-Objective Satisfactory Optimization,MOSO)问题可以表示为满足不等式约束 $g_i(x) \le 0$ 的前提下,最大化总体满意度 sw 的优化问题,其函数表达如下:

Max
$$sw = f_{op}(s), \{s = h(q), q = f(x)\}$$

 $x \in X \subseteq \mathbf{R}^n, q \in Q \subseteq \mathbf{R}^m$ (4-6)
s.t. $g_i(x) \le 0$ $i = 1, 2, ..., k$

本章中,定义式(4-6)为总体满意度方程并着重对权值的标定展开研究。

4.3 基于满意度理论的交互式进化算法

4.3.1 二元关系

为了区分群体中任意两个个体,首先定义一个二元关系,如表4-1所示。

符号	描述	阈值状态
≈	等效	$ d < \varepsilon$
>	优于	$\varepsilon \le d < \alpha$
>>	远优于	$\alpha \le d < \beta$
\prec	劣于	$-\alpha < d \le -\varepsilon$
	远劣于	$-\beta < d \le -\alpha$

表 4-1 个体二元关系

表 4-1 中参数 d 为任意两个个体之间的满意区分度,其函数表达如下:

$$d = sw_i - sw_j = \sum_{k=1}^{m} \omega_k s_{ik} - \sum_{k=1}^{m} \omega_k s_{jk}$$
 (4-7)

式中,当d为正数时,表示个体 x_i 优于个体 x_j ,反之亦然。当 $|d| < \varepsilon$ 时,表示两个个体等价。为了下文行文方便,本章只取"等价"、"优于"和"远优于"3种关系来表征任意两个个体之间的二元关系。

表 4-1 中参数 ε 为一个正数,如 10^{-5} , α 和 β 为可调参数并分别对应"优于" \succ 和"远优于" \succ 上的上限,并且满足 ($0 < \varepsilon < \alpha < \beta < 1$)。二元关系" \succ "和" \succ "实际上是用来度量不同个体的偏好程度。为了更直观地表征其物理意义,6 个个体之间的二元关系 \succ 和 \succ 示意如图 4-1 所示。

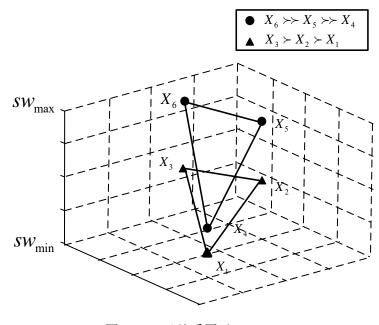


图 4-1 二元关系图示

从图 4-1 可以看出,"远优于" \rightarrow 意味着更大的偏好倾向。实际应用中,参数 α 和 β 用来界定决策者的偏好度,而它们的初始化可以借助于决策者的先验信息 或者经验。

4.3.2 权值优化

从式(4-5)与式(4-7)可知, MOSO 算法主要通过在每次迭代过程中设定相对应的权值来定义并区分不同解的满意度。因此,建立如下模型,将特定解得偏好程度转化为权值偏好度以更好地代表总体满意度。

假设L为决策者选择的一组候选者,l为L的 ℓ_0 范数(非零元素总数),L的函数表达如下:

$$L = \{x_{i}, x_{i-1}, ..., x_{i-l-1} \mid x_{i} \in \mathbf{R}^{n}, i = k, ..., k-l-1\}$$
(4-8)

L中的个体之间的偏好关系如下:

$$x_k \approx x_{k-1} \succ x_{k-2} \succ \dots \succ x_{k-l-1}$$
 (4-9)

权值优化的目标为最小化 x_k , x_{k-l-1} 相对应的满意度之间的权值 ℓ_p 范数,而优化过程的可行域由式(4-10)确定。具体优化问题函数表达如下:

Find
$$(\omega_{1}, \omega_{2}, ..., \omega_{m})$$

Min $r(s_{k}; p, \omega) = ||s_{k} - s_{k-l-1}||_{p,\omega}$

$$= \left(\sum_{i=1}^{m} \omega_{i}^{p} |s_{ki} - s_{(k-l-1),i}|^{p}\right)^{1/p}$$
s.t. $g_{1}: |cd_{k}| < \varepsilon$
 $g_{2}: \alpha \le cd_{k-l} < \beta$
 $g_{3}: \varepsilon \le cd_{k-2} < \alpha$
 \vdots
 $g_{l}: \varepsilon \le cd_{k-l} < \alpha$

约束条件 g_i , j=1,2,...,l中的系数c函数表达如下:

$$c = \frac{1}{sw_{\text{max}} - sw_{\text{min}}} \tag{4-11}$$

系数c的引入避免在进化过程中动态调整参数 α 和 β 。 sw_{max} 和 sw_{min} 的取值随着迭代的进行而不断变化。

式(4-10)求得的最优权值向量可以分配给相对应的满意度目标函数,作为选择进入下一代的分类依据。

4.3.3 搜索机制

交互式满意度优化的关键在于在搜索过程中能够探索整个目标空间,因此对于 MOEAs 本身的要求较高,本文结合第二章提出的 MOEAE/D-IHO 作为基础 MOEAs,结合之前提出的 IS 策略,组成 MOEA/D-IS 算法。

基于满意度的交互式偏好信息搜索机制如下,见图 4-2。

步骤 1: 初始化种群。

步骤 2: 决策者根据式(4-7)中定义的二元关系选取满意的部分个体进行交配生成子代。借助 MOEA/D-IHO 算法执行进化操作。

步骤 3: 运行 iter 次以后,借助表 4-1 中定义的二元关系区分具有不同满意度的个体。同时,通过求解式(4-10)中的权值向量优化问题,对满意度计算不同解

的总体满意度sw。

步骤 4: 选取满意度大的个体进入到下一代中,如果决策者无法基于偏好信息区分每个个体,算法停止,否则回到步骤 2 中。

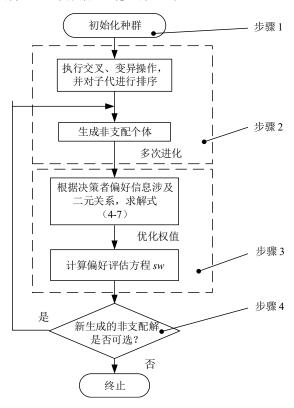


图 4-2 交互式满意度算法流程图

4.4 实验设计

4.4.1 测试函数集

第二章中多数算法在求解 2 目标 WFG2、WFG7、WFG8 和 WFG9 问题时无法 完全收敛到帕累托最优前沿的特定区域。因此本章特定选取 WFG2、WFG7、WFG8 和 WFG9 为测试函数,验证所提 MOEA/D-IS 的有效性。

测试函数	目标函数	可分性	多峰性	偏差	 几何形状
侧风团数	日你凶奴	可力压	夕 峰比	/	几円形机
WEC2	$f_{_{ m l:}M-1}$	不可分	单峰	多项式,平滑	凸,不连续
WFG2	$f_{\scriptscriptstyle M}$	不可分	多峰	多项式,平滑	口,小廷练
WFG7	$f_{_{1:M}}$	可分	单峰	参数依靠	凹
WFG8	$f_{\scriptscriptstyle 1:M}$	不可分	单峰	参数依靠	凹

表 4-2 WFG 测试函数集函数特性

表 4-2 WFG 测试函数集函数特性(续)

测试函数	目标函数	可分性	多峰性	偏差	几何形状
WFG9	$f_{_{1:M}}$	不可分	多峰,欺骗性	参数依靠	Щ

其中, WFG 的决策变量区间为 $z_i = [0,2i], i = 1,2,...,n$ 。

4.4.2 性能评价指标

本章以性能指标 GD 为主,用指标 GD 评价 MOEAs 的收敛性。

4.4.3 对比算法

为了评估本章所提出的 MOEA/D-IS 算法的性能,鉴于 MOEA/D-IS 的基础框架为第二章所提的 MOEA/D-IHO,本章选取与 2.4.3 节相同的先进算法进行对比,同时对比 MOEA/D-IHO,以验证所提出方案的有效性。

4.4.4 实验设置

本章中实验设置直接参考 2.4.4 节中对于实验的综合设置。除此之外, 3 组决策者分别介入整个优化过程,以求获取不同满意度的优化解。在后续实验结果对比中, IS-1、IS-2 和 IS-3 分别代表三种不同偏好者信息。

4.5 实验结果与分析

为了检测 MOEA/D-IS 的有效性,所有对比算法独立运行 31 次。计算结果分别从 Median Attainment Surface、数值统计和盒图三个方面展开。绘制 3 种不同决策者偏好信息的 MOEA/D-IS 与所有对比算法运行 31 次后获得的近似帕累托最优前沿面的 Median Attainment Surface,如图 4-3 至 4-5 所示。对比 3 种不同决策者偏好信息的 MOEA/D-IS 与每个对比算法在 GD 性能评价指标计算值的统计结果,以各指标平均值作为对比基准,分别列于表 4-3 至 4-6 中。并绘制了 3 种不同决策者偏好信息的 MOEA/D-IS 与每个对比算法在 GD 性能评价指标计算值的盒图。

其中,在所有图例中,IS-1、IS-2、IS-3 代表本章提出的 MOEA/D-IS 算法对应 3 中不同的偏好信息,N 代表 NSGA-II 算法、H 代表 HyPE 算法、M 代表 MOEAD/DE 算法、G 代表 GREA 算法、I 代表 MOEA/D-IHO 算法。每个示例中 WFGX-Y 代表 Y个目标的测试函数 WFGX, $X \in \{2,3,...,9\}$ 。

4.5.1 Median Attainment Surface

Median Attainment Surface 直观地将各个算法优化解在目标空间中进行对比,

同时辅以每个测试函数的真实帕累托最优前沿,便于查看各个算法优化解在目标空间的多样性和收敛性情况。

如图 4-3 所示,在 WFG2-2、WFG7-2、WFG8-2 和 WFG9-2 测试函数对比中,本章所提出的 MOEA/D-IS 在偏好信息 IS-1 的交互作用下,最终都能收敛到帕累托最优前沿的右侧边界部分驶进,同时在 IS-1 的交互作用下,MOEA/D-IS 在所有测试函数收敛性上都优于其他对比算法。

如图 4-4 所示,在 WFG2-2、WFG7-2、WFG8-2 和 WFG9-2 测试函数对比中,本章所提出的 MOEA/D-IS 在偏好信息 IS-2 的交互作用下,最终都能向帕累托最优前沿的左侧边界部分区域驶进,同时在 IS-2 的交互作用下,MOEA/D-IS 在WFG2-2 和 WFG8-2 测试函数上的收敛效果优于其他对比函数,验证了算法的有效性。

如图 4-5 所示,在 WFG2-2、WFG7-2、WFG8-2 和 WFG9-2 测试函数对比中,本章所提出的 MOEA/D-IS 在偏好信息 IS-3 的交互作用下,最终都能向帕累托最优前沿的中间部分区域驶进,同时在 IS-3 的交互作用下,MOEA/D-IS 在所有测试函数上的收敛效果优于其他对比函数,验证了算法的有效性。

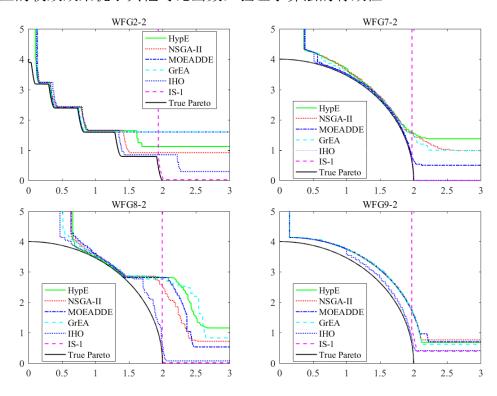


图 4-3 偏好 1WFG2-2、WFG7-2、WFG8-2 和 WFG9-2 测试函数 Median Attainment Surface 对比

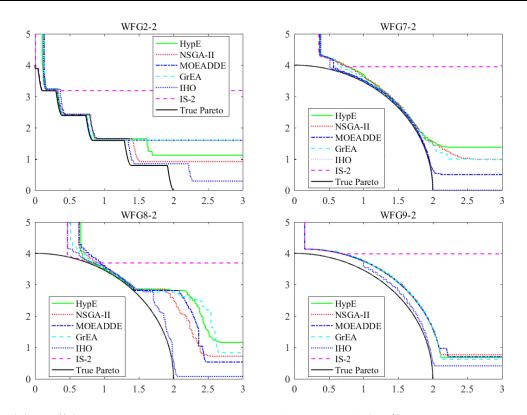


图 4-4 偏好 2WFG2-2、WFG7-2、WFG8-2 和 WFG9-2 测试函数 Median Attainment Surface 对比

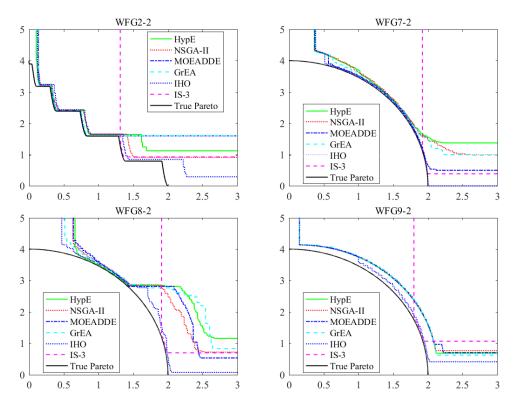


图 4-5 偏好 3WFG2-2、WFG7-2、WFG8-2 和 WFG9-2 测试函数 Median Attainment Surface 对比

4.5.2 实验数值统计结果

由于所有算法从本质上讲都是一种随机搜索方法,单靠一次的运行结果很难 判定一个算法好坏,因此本节对 31 次独立运算结果进行平均值统计,对比每个算 法在 GD 性能评价指标上的表现。其中每种问题中表现最好的算法都进行加粗处理。

如表 4-3 所示,MOEA/D-IS-1 在 WFG7-2、WFG8-2、WFG8-7 和 WFG9-2 测试函数问题上所得平均 GD 性能指标优于其他算法。MOEA/D-IHO 在 WFG2-4、WFG2-7、WFG8-4、WFG9-4 和 WFG9-7 测试函数问题上所得平均 GD 性能指标优于其他算法,在这些测试函数问题上,MOEA/D-IS-1 与 MOEA/D-IHO 所得平均 GD 性能指标相差不大。HypE、NSGA-II、GREA 和 MOEA/D-DE 在 WFG2-2 测试函数问题上所得平均 GD 性能指标表现接近,其中 HypE 表现最优,MOEA/D-IS-1表现次之,优于 MOEA/D-IHO。MOEA/D-DE、HypE、MOEA/D-IHO 和 GrEA 在 WFG7-4 测试函数问题上所得平均 GD 性能指标表现接近,其中 MOEA/D-DE 表现最优,MOEA/D-IS-1 和 NSGA-II 表现相似,表现略差。MOEA/D-DE 在 WFG7-7测试函数问题上所得平均 GD 性能指标表现最优,MOEA/D-IHO 和 MOEA/D-IS-1表现次之,MOEA/D-DE、MOEA/D-IHO 和 MOEA/D-III 和 NSGA-II、GrEA和 HypE 的表现,其中 MOEA/D-IS-1 的表现略优于 MOEA/D-IHO。

表 4-3 6 种算法在 WFG2、WFG7、WFG8 和 WFG9 测试问题上所得平均 GD 值的比较

测试	目			算法			
函数	标	NSGA-II	GrEA	MOEA/D-DE	НурЕ	MOEA/D-IHO	MOEA/D-IS-1
	2	0.004955	0.004965	0.004750	0.003701	0.010991	0.006142
WFG2	4	0.080263	0.040691	0.036647	0.029933	0.016372	0.024013
	7	0.289916	0.204183	0.052116	0.300758	0.044903	0.228467
	2	0.072188	0.069488	0.056350	0.069376	0.053948	0.014891
WFG7	4	0.309508	0.272124	0.247832	0.250388	0.255460	0.306482
	7	0.654615	0.620609	0.326770	0.571524	0.395798	0.383915
	2	0.039735	0.031432	0.037811	0.036070	0.007996	0.000892
WFG8	4	0.125122	0.083256	0.046558	0.026668	0.015227	0.019010
	7	0.225277	0.196023	0.037758	0.079089	0.031620	0.012039
	2	0.016948	0.016553	0.013283	0.017389	0.013854	0.008669
WFG9	4	0.066800	0.028889	0.024739	0.028743	0.018829	0.023461
	7	0.213437	0.143150	0.065477	0.106394	0.057415	0.073164

如表 4-4 所示, MOEA/D-IS-2 在 WFG2-2、WFG8-2 和 WFG8-7 测试函数问题 上所得平均GD性能指标优于其他算法。MOEA/D-IHO 在 WFG2-4、WFG2-7、 WFG7-2、WFG8-4、WFG9-4 和 WFG9-7 测试函数问题上所得平均 GD 性能指标优 于其他算法,其中 MOEA/D-IS-2 在 WFG2-4 和 WFG9-7 测试问题上所得平均 GD 性能指标要远优于 NSGA-II、GrEA 和 HypE,在 WFG2-7、WFG7-2、WFG8-4 和 WFG9-4 测试问题上所得平均 GD 性能指标与其他算法表现接近,略差于 MOEA/D-IHO。MOEA/D-DE 在 WFG7-4 测试问题上所得平均GD性能指标优于其他算法, 其中 MOEA/D-IS-2 和 NSGA-II 表现相似,表现略差。MOEA/D-DE 在 WFG7-7 测 试问题上所得平均GD性能指标优于其他算法,MOEA/D-IHO 和 MOEA/D-IS-2 表 现次之,都远优于其他算法,其中 MOEA/D-IS-2 略优于 MOEA/D-IHO。HypE、 NSGA-II、GREA 和 MOEA/D-DE 在 WFG2-2 测试函数问题上所得平均 GD 性能指 标表现接近, 其中 HypE 表现最优, MOEA/D-IS-2 表现次之, 优于 MOEA/D-IHO。 MOEA/D-DE、HypE、MOEA/D-IHO 和 GrEA 在 WFG7-4 测试函数问题上所得平 均GD性能指标表现接近,其中MOEA/D-DE表现最优,MOEA/D-IS-2与NSGA-II 类似,表现略差于其他算法。MOEA/D-DE 在 WFG7-7 测试函数问题上所得平均 GD 性能指标表现最优, MOEA/D-IHO 和 MOEA/D-IS-2 表现次之, MOEA/D-DE、 MOEA/D-IHO 和 MOEA/D-SI-I 都远优于 NSGA-II、GrEA 和 HypE 的表现,其中 MOEA/D-IS-1 的表现略优于 MOEA/D-IHO。MOEA/D-DE 在 WFG9-2 测试函数问 题上所得平均GD性能指标表现最优,MOEA/D-IHO和 MOEA/D-IS-2表现与之相 近, MOEA/D-DE、MOEA/D-IHO 和 MOEA/D-IS-2 都略优于 NSGA-II、GrEA 和 HypE 的表现,其中 MOEA/D-IHO 的表现略优于 MOEA/D-IS-2。

如表 4-5 所示,MOEA/D-IS-3 在 WFG7-2、WFG8-2、WFG8-7 和 WFG9-2 测试函数问题上所得平均GD 性能指标优于其他算法。MOEA/D-IHO 在 WFG2-4 和 WFG8-4 测试函数问题上所得平均GD 性能指标优于其他算法,MOEA/D-IS-3 表现次之,远优于其他算法,MOEA/D-IHO 在 WFG2-7 测试函数问题上所得平均GD 性能指标优于其他算法,MOEA/D-DE 表现次之,MOEA/D-IS-3、NSGA-II 和 GrEA 表现相似,优于 HypE。MOEA/D-IHO 在 WFG9-4 测试函数问题上所得平均GD 性能指标优于其他算法,MOEA/D-IS-3 和 MOEA/D-DE 表现次之,优于其他算法。MOEA/D-IHO 在 WFG9-4 测试函数问题上所得平均GD 性能指标优于其他算法,MOEA/D-IS-3 和 MOEA/D-DE 表现次之,远优于其他算法,MOEA/D-IS-3 和 MOEA/D-DE 表现次之,远优于其他算法,MOEA/D-IS-3 和 MOEA/D-DE 表现次之,远优于其他算法。MOEA/D-IS-3 在 WFG2-2 和 WFG7-7 测试问题上所得平均GD 性能指标表现与 NSGA-II 类似,略差于其他算法。

表 4-4 6 种算法在 WFG2、WFG7、WFG8 和 WFG9 测试问题上所得平均 GD 值的比较

测试	目			算法			
函数	标	NSGA-II	GrEA	MOEA/D-DE	НурЕ	MOEA/D-IHO	MOEA/D-IS-2
	2	0.004955	0.004965	0.004750	0.003701	0.010991	0.000614
WFG2	4	0.080263	0.040691	0.036647	0.029933	0.016372	0.024013
	7	0.289916	0.204183	0.052116	0.300758	0.044903	0.228467
	2	0.072188	0.069488	0.056350	0.069376	0.053948	0.055639
WFG7	4	0.309508	0.272124	0.247832	0.250388	0.255460	0.306482
	7	0.654615	0.620609	0.326770	0.571524	0.395798	0.383915
	2	0.039735	0.031432	0.037811	0.036070	0.007996	0.003868
WFG8	4	0.125122	0.083256	0.046558	0.026668	0.015227	0.019010
	7	0.225277	0.196023	0.037758	0.079089	0.031620	0.012039
	2	0.016948	0.016553	0.013283	0.017389	0.013854	0.014332
WFG9	4	0.066800	0.028889	0.024739	0.028743	0.018829	0.023461
	7	0.213437	0.143150	0.065477	0.106394	0.057415	0.073164

表 4-5 6 种算法在 WFG2、WFG7、WFG8 和 WFG9 测试问题上所得平均 GD 值的比较

测试	目			算法			
函数	标	NSGA-II	GrEA	MOEA/D-DE	НурЕ	MOEA/D-IHO	MOEA/D-IS-3
	2	0.004955	0.004965	0.004750	0.003701	0.010991	0.005326
WFG2	4	0.080263	0.040691	0.036647	0.029933	0.016372	0.024013
	7	0.289916	0.204183	0.052116	0.300758	0.044903	0.228467
	2	0.072188	0.069488	0.056350	0.069376	0.053948	0.031619
WFG7	4	0.309508	0.272124	0.247832	0.250388	0.255460	0.306482
	7	0.654615	0.620609	0.326770	0.571524	0.395798	0.383915
	2	0.039735	0.031432	0.037811	0.036070	0.007996	0.002295
WFG8	4	0.125122	0.083256	0.046558	0.026668	0.015227	0.019010
	7	0.225277	0.196023	0.037758	0.079089	0.031620	0.012039
	2	0.016948	0.016553	0.013283	0.017389	0.013854	0.007652
WFG9	4	0.066800	0.028889	0.024739	0.028743	0.018829	0.023461
	7	0.213437	0.143150	0.065477	0.106394	0.057415	0.073164

4.5.3 盒图

盒图用以展示所得数据的离散情况。

如图 4-6 所示,MOEA/D-IS-1 在 2 目标 WFG2、WFG7、WFG8 和 WFG9 测试函数 GD 盒图上表现最优。

如图 4-7 所示,MOEA/D-IS-1 在 4 目标 WFG2、WFG8 和 WFG9 测试函数 GD 盒图上表现略差于 MOEA/D-IHO,但优于其他算法,MOEA/D-IS-1 在 4 目标 WFG7 测试函数 GD 盒图上表现与 NSGA-II 相似,差于其他算法。

如图 4-8 所示,MOEA/D-IS-1 在 7 目标 WFG2 和 WFG7 测试函数 GD 盒图上表现处于中等水平。MOEA/D-IS-1 在 4 目标 WFG8 测试函数 GD 盒图上表现最优,在 4 目标 WFG9 测试函数 GD 盒图上表现与 MOEA/D-IHO 和 MOEA/D-DE 表现接近,优于其他算法。

如图 4-9 所示,MOEA/D-IS-2 在 2 目标 WFG2、WFG7、WFG8 和 WFG9 测试函数 GD 盒图上表现最优。

如图 4-10 所示,MOEA/D-IS-2 在 4 目标 WFG2、WFG8 和 WFG9 测试函数 GD 盒图上表现略差于 MOEA/D-IHO,但优于其他算法,MOEA/D-IS-2 在 4 目标 WFG7 测试函数 GD 盒图上表现与 NSGA-II 相似,差于其他算法。

如图 4-11 所示,MOEA/D-IS-2 在 7 目标 WFG2 和 WFG7 测试函数 GD 盒图上表现处于中等水平。MOEA/D-IS-2 在 4 目标 WFG8 测试函数 GD 盒图上表现最优,在 4 目标 WFG9 测试函数 GD 盒图上表现与 MOEA/D-IHO 和 MOEA/D-DE 表现接近,优于其他算法。

如图 4-12 所示,MOEA/D-IS-3 在 2 目标 WFG2、WFG7、WFG8 和 WFG9 测试函数 GD 盒图上表现最优。

如图 4-13 所示,MOEA/D-IS-3 在 4 目标 WFG2、WFG8 和 WFG9 测试函数 GD 盒图上表现略差于 MOEA/D-IHO,但优于其他算法,MOEA/D-IS-3 在 4 目标 WFG7 测试函数 GD 盒图上表现与 NSGA-II 相似,差于其他算法。

如图 4-14 所示,MOEA/D-IS-3 在 7 目标 WFG2 和 WFG7 测试函数 GD 盒图上表现处于中等水平。MOEA/D-IS-3 在 4 目标 WFG8 测试函数 GD 盒图上表现最优,在 4 目标 WFG9 测试函数 GD 盒图上表现与 MOEA/D-IHO 和 MOEA/D-DE 表现接近,优于其他算法。

综上,MOEA/D-IS 算法,在 2 目标 WFG2、WFG7、WFG8 和 WFG9 测试函数 GD 盒图上表现最优,在 4 目标 WFG2、WFG7、WFG8 和 WFG9 测试函数 GD 盒图上表现次之,在 7 目标 WFG2、WFG7、WFG8 和 WFG9 测试函数 GD 盒图上表现比在 2 目标和 4 目标测试函数上表现略差。

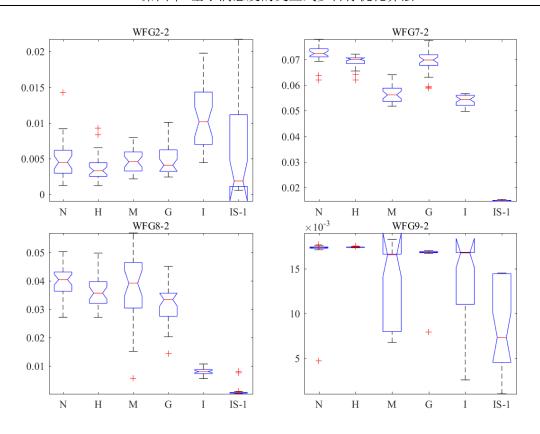


图 4-6 偏好 1WFG2-2、WFG7-2、WFG8 和 WFG9-4 测试函数 GD 盒图

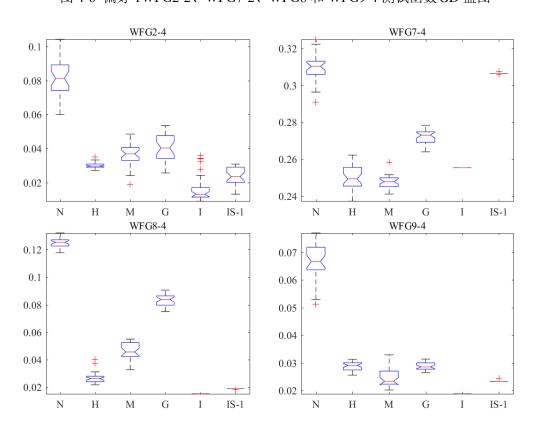


图 4-7 偏好 1WFG2-4、WFG7-4、WFG8-4 和 WFG9-4 测试函数 GD 盒图

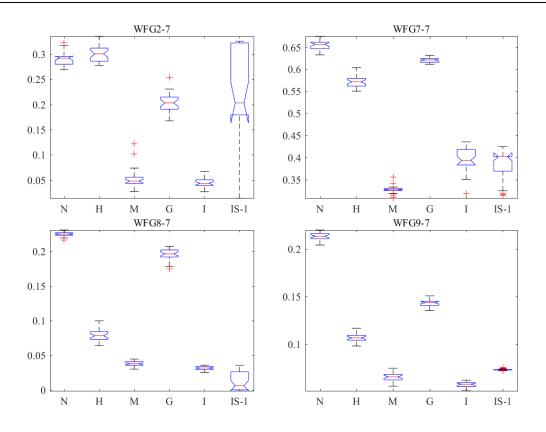


图 4-8 偏好 1WFG2-7、WFG7-7、WFG8-7 和 WFG9-7 测试函数 GD 盒图

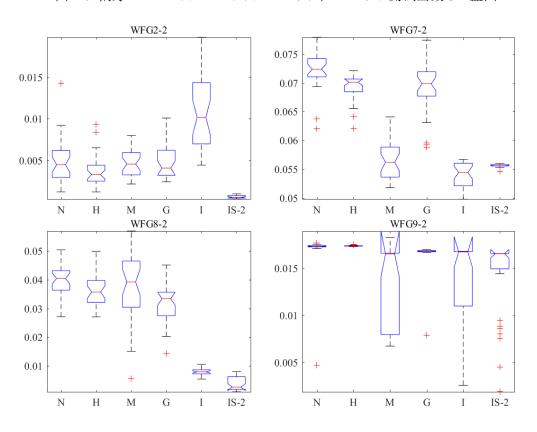


图 4-9 偏好 2WFG2-2、WFG7-2、WFG8 和 WFG9-4 测试函数 GD 盒图

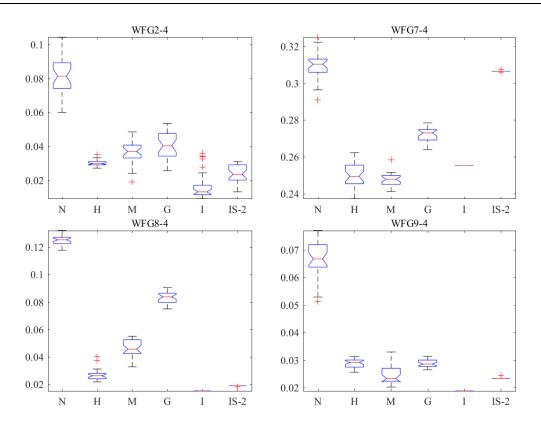


图 4-10 偏好 2WFG2-4、WFG7-4、WFG8-4 和 WFG9-4 测试函数 GD 盒图

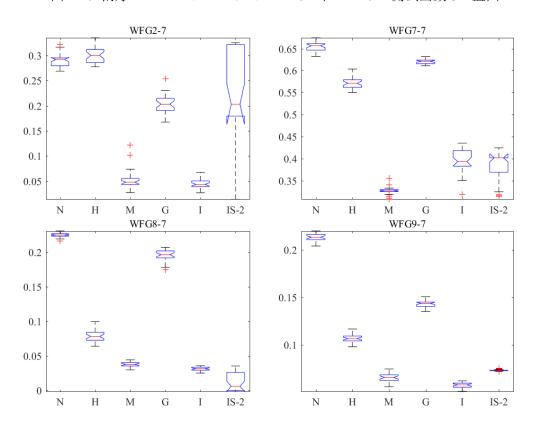


图 4-11 偏好 2WFG2-7、WFG7-7、WFG8-7 和 WFG9-7 测试函数 GD 盒图

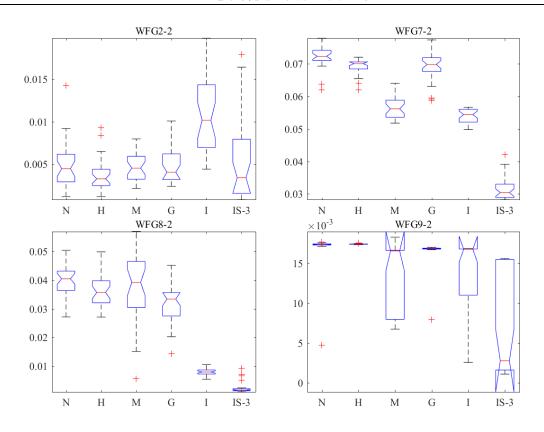


图 4-12 偏好 3WFG2-2、WFG7-2、WFG8 和 WFG9-4 测试函数 GD 盒图

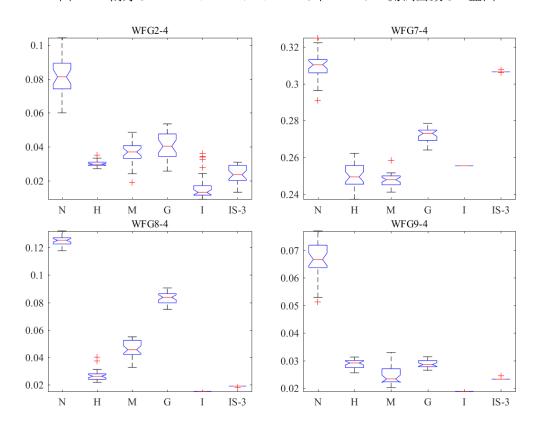


图 4-13 偏好 3WFG2-4、WFG7-4、WFG8-4 和 WFG9-4 测试函数 GD 盒图

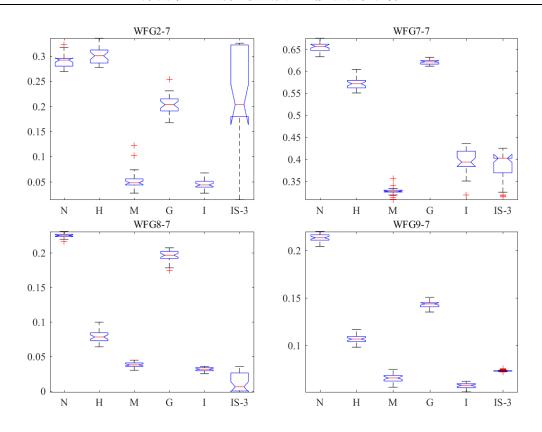


图 4-14 偏好 3WFG2-7、WFG7-7、WFG8-7 和 WFG9-7 测试函数 GD 盒图

4.5.4 结果分析

结合 4.5.1 节 Median Attainment Surface 对比情况,可以得出 MOEA/D-IS 借助 3 个不同决策者偏好信息都能向帕累托最优前沿面相对应部分驶进,结合 4.5.2 和 4.5.3 节中平均 GD 性能指标和 GD 盒图,验证了所获得偏好解的收敛性。然而,在 4 目标和 7 目标 WFG 测试函数性能指标对比中,MOEA/D-IS 的表现不如在 2 目标 WFG 测试函数中表现优异。这主要因为,GD 的计算结果为近似解集到真实帕累托前沿面最近距离值的平均值。MOEA/D-IS 所获取的近似解集为 MOEA/D-IHO 近似解集的一部分,MOEA/D-IHO 近似解集的其他部分 GD 性能指标有可能会优于 MOEA/D-IS 获取的部分解集。然而,在这种情况下 MOEA/D-IS 依然在某些 4 目标和 7 目标测试函数问题上所得平均 GD 性能指标最优,在某些 4 目标和 7 目标测试函数问题上所得平均 GD 性能指标略差于其他算法,一定程度上验证了 MOEA/D-IS 在收敛性上的优异表现。同样的结论也适用于 4.5.3 节 4 目标和 7 目标 GD 盒图对比结果。

4.6 本章小结

本章根据偏好信息在优化过程中不同的参与形式,提出了一种基于满意度的

交互式多目标进化算法。借助满意度理论,对决策者的偏好信息进行度量,以第二章提出的 MOEA/D-IHO 为基本框架,对 MOEA/D-IS 算法在 2 目标、4 目标和 7 目标 WFG2、WFG7、WFG8 和 WFG9 测试函数性能指标 GD 与基准算法和其他优秀 MOEAs 进行对比,验证了本章提出的 IS 策略在获取偏好解上的有效性。

第五章 大型矿用挖掘机提升机构减速器多目标优化设计

实际工程中,传统机械产品的设计优化问题大都是多目标优化问题。一般设计者主要以功能与成本为出发点,力求在保证性能达标的基础上将成本降到最低以获得最大效益。随着社会的不断发展,机械产品设计过程在协同满足功能与成本的前提下,还需要考虑产品在运行过程中涌现的诸多可靠性和安全性问题,即性能问题。但是,每增加一个目标函数都会给优化过程增添一定程度的阻碍,因此采取行之有效的多目标优化方法,在设计阶段综合考虑机械产品在成本、功能和性能三个方面,以期获得较为满意的优化结果,指导实际工程应用。前三章内容分别从3个不同角度给出求解 MOPs 时行之有效的算法,但是这些算法有效性的验证都还停留在测试函数集的层面上,未曾通过实际工程案例加以验证。因此,本章将以工程实际需求为出发点,附以验证本文所提出不同算法的性能。

5.1 引言

当前,大型矿用挖掘机作为开采露天矿山的主要采装设备,正向高速、重载、精密和复杂化的方向发展,人们对大型挖掘机工作装置的可靠性和安全性要求越来越高^[122]。即不仅要求它在保证较低的成本下能够传递较大的载荷,而且要求工作装置本身必须具备良好的可靠性,从而降低大型挖掘机的运营成本并提高设备运营过程中的安全性^[199],这便对其在初始设计阶段提出了更高的优化设计要求。

提升机构作为大型矿用挖掘机重要工作装置,主要由提升减速器、提升卷筒、提升电机、提升钢丝绳、联轴器和润滑系统组成,其主要功能为通过提升减速器传动提升电机扭矩带动提升卷筒转动进而提起和放下铲斗,并与推进机构协同工作共同完成挖掘任务。提升减速器作为直接传递转矩的工作装置,实际工作过程中,由于复杂的受力状态、恶劣的工作条件和使用环境,提升机构减速器齿轮(尤其是连接提升卷筒的大齿轮和传动轴齿轮)经常会发生磨损甚至断齿的情况。在工作过程中,减速器发生可观测形变,噪声巨大,严重影响挖掘机正常开采任务的完成,有时减速器部分螺栓振松甚至振断,直接导致整个挖掘机停工整修。此外,由于减速器的尺寸巨大,其生产制造过程为单件生产,给更换和维修带来了困难的同时,也增加了维护成本。鉴于上述情况,生产厂家急需对已投产的产品进行重新设计,在优化设计阶段将与齿轮寿命相关因素与性能函数同时考虑进去,在保证一定成本的前提下,以期提升现有产品的综合竞争力。

针对减速器进行多目标优化的工作相对较少, Thompson 等[148]提出以最小接

触疲劳系数和最小体积为优化目标,采用较为简单的加权和方法进行求解,并且将 减速器简化为齿轮接触对,仅以齿轮的体积作为整体体积优化目标,同时齿轮体积 的计算也采用圆柱体的近似形式。Deb 等[31]以 NSGA-II 为基础算法,以最小化齿 轮体积和最大化能量传输为目标,针对齿数为定数和齿数变化的两种情形,展开细 致研究与讨论。Huang 等[152]在 Thompson 提出的优化模型基础上增加了减速器承 载能力这一目标函数,同时提出一种交互式物理规划的多目标优化方法,通过将决 策者的偏好信息引入到优化过程中,有效解决了该三级齿轮减速器多目标优化问 题。Li 等[44]构建了一个包括最大化承载能力,最小化体积和最小化输出轴转角误 差的多目标优化模型,同时基于模糊控制理论提出了一种自适应遗传算法用以求 解该减速器优化模型。此外,孙志礼等[200]提出了在减小体积的同时,以提升传动 效率为另一个优化目标,在其构建的模型下,一定程度减小体积的同时,传动效率 有显著的提升。同样,周鹏等[201]提出一种同时保证行星齿轮减速器体积最小,传 功效率最高和承载能力最大的减速器多目标优化模型,通过构建比较矩阵、判断矩 阵和进行一致性检验,确定分配到各个目标函数上的权值,进而采用单目标优化方 法进行求解。这些方法一定程度上都很好地解决了其所构建的多目标优化问题。然 而,在这些多目标优化问题的构建中,都忽略了齿轮的具体构形和减速器整体体积 对于工程设计的具体指导意义。在齿轮构形的问题上, Wang 等[202]指出,由于顶隙 和齿轮变位的存在,将齿轮视为一圆柱体(以分度圆直径为圆柱体直径)将造成3.5% 的计算误差,同时不同齿轮构形对于齿轮体积的计算也有相当的影响。在减速器整 体体积计算问题上, Buiga[140,141]等为计算齿轮、轴和箱体的计算给出一个指导性建 议。同时,上述研究对象多为抽象齿轮对和简单的工业减速器,而本章涉及的研究 对象本身尺寸巨大,同时工作环境与上述研究对象大相径庭,在满足功能最大化, 体积最小化的基础上,还需要考虑运行的稳定性。

本章在分析大型矿用挖掘机提升机构减速器具体构形的基础上,结合实际工况提出了能够指导实际生产设计的多目标优化模型,同时借助前文提出的多目标进化算法,对提升机构减速器进行多目标优化设计。

5.2 大型矿用挖掘机提升机构减速器多目标优化模型

大型矿用挖掘机的作业环境为露天野外,设备长期承受风吹、日晒、雨淋、雪侵、雷击、粉尘的考验。复杂多样的气候环境和地质条件增加了大型矿用挖掘机正常运行的不稳定性,而提升机构减速器作为大型矿用挖掘机提升机构传递扭矩的关键装置,也直接经受该不稳定性,加剧其在运行过程中失效的可能性。因此,在提升机构减速器优化设计过程中,在降低成本的同时,还应当充分考虑提升减速器

整体性能。减速器将主从电机输出扭矩传递到卷筒,带动钢丝绳提升铲斗完成挖掘工况,由于工作载荷巨大,减速器需要传递的扭矩巨大。同时由于铲斗需要往复运动,减速器承受重载循环载荷,对其齿轮的接触疲劳寿命也提出了更高的要求。

5.2.1 提升机构减速器基本构成

提升减速器的主要构成如图 5-1 所示,提升机构减速器采用对称式两级齿轮的布局。其中主从提升电机在工作过程中提供转矩,经测试发现,两台电机转速的时域曲线几乎完全重合,同时两台电机转矩偏差也在 1%以下,因此后续计算中所涉及到的转矩直接将单个电机的转矩乘以 2。图 5-1 中序列号相对应构件的描述如表5-1 所示。

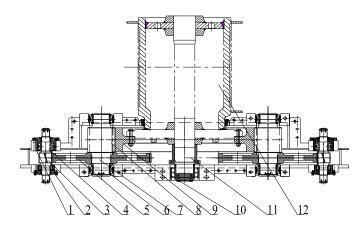


图 5-1 提升机构减速器主要构成

编号	名称	编号	名称
1	减速器箱体	7	中间轴
2	输入轴双列圆锥滚子轴承	8	二级直齿小齿轮
3	一级斜齿小齿轮	9	二级直齿大齿轮
4	输入轴	10	输出轴双列圆锥滚子轴承
5	一级斜齿大齿轮	11	输出轴
6	中间轴双列圆锥滚子轴承	12	提升卷筒

表 5-1 提升机构减速器序列号与相关描述

如表 5-1 所示,减速器一级为斜齿轮传动,二级为直齿轮传动。电机与一级小齿轮相连,以实现主从电机与减速器的刚性连接。

5.2.2 目标函数

由于本章研究对象的齿轮为英制齿轮,主要符合美国齿轮标准(AGMA)。本

文目标函数的选取参考 Huang 等[152]与 Thompson 等[148]的文献,分别选取最小化减速箱体积,最大化接触疲劳寿命和最大化承载能力为相对应的目标函数。其中,减速箱体积主要考虑箱体、齿轮和轴颈三者体积构成;接触疲劳寿命与接触疲劳系数成反比,因此取接触疲劳系数最小化为优化目标;由于承载能力取最大化,因此对其目标函数乘以-1,以统一所有目标函数均为最小化。

$$\min V$$

$$\min SF = \min \left\{ SF_1, SF_2 \right\}$$

$$\max T_{capacity} = \max \left\{ T_1, T_2 \right\}$$
(5-1)

(1) 最小体积

由于第一级大齿轮直径将近两米,同时该减速器为双电机对称设计,因此该减速器的整体尺寸相较于普通机械减速箱要大很多。鉴于此,需要将箱体体积和轴径体积同时纳入减速器总体体积计算当中。假设齿轮、箱体与轴颈材料相同,因此将三者体积通过简单的线性相加便能够代表减速器总体体积,其函数表达如下:

$$V = V_{gears} + V_{shaft} + V_{house}$$
 (5-2)

式中, V_{sears} 为所有齿轮体积总和, V_{shaft} 为所有齿轮轴体积, V_{house} 为箱体体积。

1) 齿轮体积

如图 5-1 所示,此减速器中有 7 个齿轮,齿轮体积 V_{eqgs} 函数表达如下:

$$V_{\textit{gears}} = 2 \cdot V_{p1} + 2 \cdot V_{w1} + 2 \cdot V_{p2} + V_{w2} \tag{5-3}$$

式中, V_{p1} 为第一级斜齿小齿轮体积, V_{w1} 为第一级斜齿大齿轮体积, V_{p2} 为第二级直齿小齿轮体积, V_{w2} 为第二级直齿大齿轮体积。

该减速器所有小齿轮为实心式齿轮,所有大齿轮为腹板式齿轮,如图 5-3 所示。同时由于齿轮本身直径过大,通常在计算齿轮体积时,将齿项圆到分度圆之间的齿轮倒放在齿根圆到分度圆之间,体积的计算可以简化为计算分度圆和齿厚构成的圆柱体。然而由于顶隙和齿轮变位的存在,倒放部分与齿根圆之间会形成间隙,直接以圆柱体体积作为齿轮体积将造成近 3.5%的误差,因此本文对齿轮体积进行建模时,采用了 Wang 等[202,203]提出的考虑该间隙的齿轮体积构建方法,具体每个齿轮体积计算函数表达如下:

$$V_{p1} = \frac{\pi \left(\left(d_{p1} \right)^2 - \left(d_{s1} \right)^2 \right)}{4} b_{p1} - V_{p1}^c$$
 (5-4)

$$V_{w1} = \frac{\pi \left(\left(d_{w1} \right)^2 - \left(d_{w1}^v \right)^2 + \left(d_{w1}^n \right)^2 - \left(d_{s2} \right)^2 \right) b_{w1}}{4} + \frac{\pi \left(\left(d_{w1}^v \right)^2 - \left(d_{w1}^n \right)^2 - 6\pi \left(d_{w1}^p \right)^2 \right) c_1}{4} - V_{w1}^c$$
(5-5)

$$V_{p2} = \frac{\pi \left(\left(d_{p2} \right)^2 - \left(d_{s2} \right)^2 \right)}{4} b_{p2} - V_{p2}^c$$
 (5-6)

$$V_{w2} = \frac{\pi \left(\left(d_{w2} \right)^2 - \left(d_{w2}^v \right)^2 + \left(d_{w2}^n \right)^2 - \left(d_{s3} \right)^2 \right) b_{w2}}{4} + \frac{\pi \left(\left(d_{w2}^v \right)^2 - \left(d_{w2}^n \right)^2 - 6\pi \left(d_{w2}^p \right)^2 \right) c_2}{4} - V_{w2}^c$$
(5-7)

式(5-4)-(5-7)中,下标 p_1 、 w_1 、 p_2 和 w_2 分别代表一级斜齿小齿轮、一级斜齿大齿轮、二级直齿小齿轮和二级直齿大齿轮。下标 s_1 、 s_2 和 s_3 分别代表输入轴、中间轴和输出轴。 d_i , $i = \{s_1, s_2\}$ 为轴径, d_i , $i = \{p_1, w_1, p_2, w_2\}$ 为齿轮分度圆直径,具体函数表达如下:

$$d_{i} = \begin{cases} m_{1}n_{i}/\cos\psi, & i = \{p_{1}, w_{1}\}\\ m_{2}n_{i}, & i = \{p_{2}, w_{2}\} \end{cases}$$
 (5-8)

 V_{i}^{c} , $i = \{p_{1}, w_{1}, p_{2}, w_{2}\}$ 为间隙体积,具体函数表达如下:

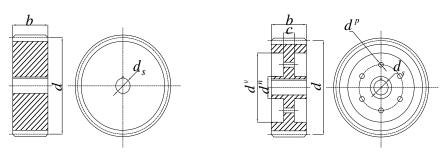
$$V_i^c = s_i^a \cdot x_i^* \cdot b_i \cdot n_i, i = \{p_1, w_1, p_2, w_2\}$$
 (5-9)

式 (5-9) 中, s_i^a 为直齿、斜齿横切面齿顶厚,具体函数表达如下:

$$s_i^a = d_i^a \left(\frac{\pi}{2n_0} + \frac{2\chi_i \tan \alpha}{n_0} + (\text{inv } \alpha - \text{inv } \alpha_i^a) \right), i = \{p_1, w_1, p_2, w_2\}$$
 (5-10)

式 (5-10) 中, d_i^a 为齿顶圆直径,具体函数表达如下:

$$d_i^a = d_i + 2m_1(1 + \chi_i), i = \{p_1, w_1, p_2, w_2\}$$
 (5-11)



(a) 实心式齿轮

(b) 腹板式齿轮

图 5-3 齿轮结构。(a) 实心式齿轮;(b) 腹板式齿轮

式 (5-10) 中,n 为当量齿数,具体函数表达如下:

$$n_{v} = \begin{cases} \frac{n_{i}}{\cos^{3} \psi}, i = p_{1}, w_{1} \\ n_{i}, i = p_{2}, w_{2} \end{cases}$$
 (5-12)

式(5-10)中, α 为分度圆压力角, α_i^a 为齿顶圆压力角,具体函数表达如下:

$$\alpha_i^a = \arccos\left(d_i^b/d_i^a\right) \tag{5-13}$$

式(5-13)中, d_i^b 为齿轮基圆直径,表示为 $d_i^b = d_i \cos \alpha, i = \{p_1, w_1, p_2, w_2\}$ 。 式(5-9)中, x_i^* 为齿根圆与齿顶圆通过分度圆镜像后形成的圆弧之间的距离,具体函数表达如下:

$$x_{i}^{*} = (x^{*} - 2\chi_{i})m_{j}, i = \{p_{1}, w_{1}, p_{2}, w_{2}\}, j = \{1, 2\}$$
 (5-14)

式(5-14)中, x^* 为变位系数。式(5-9)中, b_i , $i = \{p_1, w_1, p_2, w_2\}$ 为齿宽,并且满足如下条件:

$$b_{w1} = b_{p1} - 10$$
, $b_{w2} = b_{p2} - 10$ (5-15)

式(5-9)中, n_i , $i = \{p_1, w_1, p_2, w_2\}$ 为齿数。式(5-5)和式(5-7)中, d_i^v , $i = \{w_1, w_2\}$ 为轮辋内径, d_i^n , $i = \{w_1, w_2\}$ 为轮毂外径, d_i^p , $i = \{w_1, w_2\}$ 为螺栓孔直径, c_i , $i = \{1, 2\}$ 为腹板厚度。

 d_i^v , $i = \{w_1, w_2\}$, d_i^n , $i = \{w_1, w_2\}$, d_i^p , $i = \{w_1, w_2\}$ 和 c_i , $i = \{1, 2\}$ 的函数表达分别如下:

$$d_i^v = d_i^a - 12m_i, i = \{w_1, w_2\}, j = \{1, 2\}$$
 (5-16)

$$d_i^n = 1.7d_i, i = \{w_1, w_2\}, j = \{s_2, s_3\}$$
 (5-17)

$$d_i^p = 0.3(d_i^v - d_i^n)d_i, i = \{w_1, w_2\}$$
 (5-18)

$$c_i = 0.2(b_i - 10), i = \{1, 2\}, j = \{p_1, p_2\}$$
 (5-19)

2) 箱体体积

对于箱体体积,其数学表达式分为两部分,其中一部分主要用来安装一级齿轮,表示为 V_1^h ,另外一部分用来安装二级齿轮,表示为 V_2^h 。

具体列出表达式之前,首先提出如下三个假设条件:

假设 1: 箱体尺寸主要与齿宽、中心距和齿顶圆相关联,箱体的厚度主要由齿 宽进行确定,长度主要由齿轮中心距进行确定,高度主要由齿顶圆直径进行确定。

假设 2: 箱体为密封件,轴承在箱体体积计算中忽略不计。

假设 3: 箱体壁厚一致,箱体体积直接由箱体内部空间体积进行代替。

鉴于上述三个假设,箱体体积函数表达如下:

$$V_{bouse} = V_1^h + V_2^h (5-20)$$

式 (5-20) 中, V_1^h 和 V_2^h 的函数表达分别如下:

$$V_1^h = \left(2a_1\cos\varphi + 2a_2 + d_{p1}^a + 40\right) \cdot \left(\frac{d_{w2}^a + d_{w1}^a}{2} + 40\right) \cdot (b_{p1} + 40)$$
 (5-21)

$$V_{2}^{h} = \left(2a_{2} + d_{p2}^{a} + 40\right) \cdot \left(\frac{d_{w2}^{a}}{2} + 40\right) \cdot b_{w1} + \left(\frac{d_{w2}^{a}}{2} + 20\right) \cdot b_{w1} + \left(\frac{d_{w2}^{a}}{2} + 20\right) \tan W \cdot \left(\frac{d_{w2}^{a}}{2} + 20\right)\right) \cdot b_{w1},$$

$$\tan W = \tan \left(\arcsin \left(\frac{d_{w2}^{a}/2 + 20}{a_{2} + 20}\right) / 2\right)$$

$$(5-22)$$

式(5-21)中, a_1 和 a_2 分别为一级、二级齿轮中心距, φ 为一级齿轮中心距与水平线夹角,如图 5-4 所示, d_i^a , $i = \{p_1, w_1, p_2, w_2\}$ 为齿顶圆直径, b_i , $i = \{p_1, w_1\}$ 为齿宽。

3)轴体积

假设减速器中所有齿轮轴材料一致,齿轮轴体积函数表达如下:

$$V_{shaft} = 2V_{s1} + 2V_{s2} + V_{s3} (5-23)$$

式(5-23)中,下标 V_{s1} 、 V_{s2} 和 V_{s3} 分别代表输入轴、中间轴和输出轴体积,其函数表达分别如下:

$$V_{s1} = (b_{n1} + 40)\pi d_{s1}^2 / 4 \tag{5-24}$$

$$V_{s2} = (b_{p1} - 10 + b_{p2} + 40)\pi d_{s2}^{2} / 4$$
 (5-25)

$$V_{s3} = (b_{p1} - 10 + b_{p2} + 40)\pi d_{s3}^{2}/4$$
 (5-26)

式中, d_i , $i = \{s_1, s_2, s_3\}$ 为齿轮轴直径。

(2) 最大接触疲劳寿命

鉴于矿用挖掘机工况多为重载、冲击等,因此为了保证其运行的稳定性,需要对其接触疲劳寿命加以优化。与接触疲劳寿命相关的系数为接触疲劳寿命系数 C_l ,接触疲劳寿命系数值越小,齿轮接触疲劳寿命越长 $[^{148}]$ 。而本章中接触疲劳寿命目标函数 SF_1 和 SF_2 分别为一级、二级接触疲劳寿命系数的平方,其函数表达分别如式(5-27)和式(5-28)所示,因此取其极小化便可以满足最大化接触疲劳寿命的目标。

$$SF_{1} = \frac{4C_{p}^{2}K_{V}K_{o}0.93K_{m}T_{input}}{\cos\alpha\sin\alpha S_{fe}^{2}C_{R}^{2}} \cdot \frac{d_{p1} + d_{w1}}{b_{p1}d_{p1}^{2}d_{w1}}$$
(5-27)

$$SF_{2} = \frac{4C_{p}^{2}K_{V}K_{O}K_{m}T_{input}}{\cos\alpha\sin\alpha S_{fe}^{2}C_{R}^{2}} \cdot \frac{d_{p2} + d_{w2}}{b_{p2}d_{p2}^{2}d_{w2}} \cdot \frac{d_{w1}}{d_{p1}}$$
(5-28)

式(5-27)和式(5-28)中的参数选取如下 $^{[204]}$: C_p 为弹性系数,鉴于提升机构减速器中所有齿轮均为钢材,弹性系数 C_p 取 191,单位为 $\sqrt{\mathrm{MPa}}$ 。 K_v 为动载系数,由于齿轮为高精度啮合齿轮,同时加工方式为插齿成形的方式, K_v 的函数表达式为 $\left(78+\sqrt{V}\right)/78$,其中V为节线速度,单位为 $\mathrm{ft/min}$ 。 K_o 为过载系数,由于驱动电机均匀地输出扭矩并且减速器经受工作载荷为中度冲击,因此 K_o 取值为 1.25。 K_m 未安装校正因子,根据齿宽不同,一级齿轮设为 1.5,二级齿轮设为 1.8。 α 为压力角。 S_{fe} 为接触疲劳强度,设为 632 MPa , C_R 为接触可靠性因子,由于规定可靠度要求为 0.99,此处设为 1。

(3) 最大承载能力

$$T_{1} = \frac{S'_{n}(H_{1})C_{s}(H_{1})k_{r}k_{t}k_{ms}m_{1}b_{p1}J(N_{p1})d_{p1}}{2K_{V}K_{O}0.93K_{m}}$$
(5-29)

$$T_{2} = \frac{S'_{n}(H_{2})C_{s}(H_{2})k_{r}k_{t}k_{ms}m_{2}b_{p2}J(N_{p2})d_{p1}d_{p2}}{2K_{V}K_{O}K_{m}d_{w1}}$$
(5-30)

式(5-29)和式(5-30)中参数选取如下^[204]: k_r 为弯曲可靠度校正因子,鉴于规定的可靠度要求为 0.99, k_r 设为 0.814。 k_t 为温度因子,由于齿轮均为钢材,同时工作环境温度小于 71°C,因此 k_t 设为 1。 k_m 为平均应力修正系数,由于输入和输出齿轮都为单向受弯,因此取值为 1.4。J 为 Lewis 弯曲强度几何系数,其取值由《机械零件设计原理》^[204]中关于直齿与斜齿的不同构型进行确定。 S_n 为屈服极限,取值 324MPa。 C_s 为接触系数,结合实际加工情况进行选取。

5.2.3 约束条件

约束条件的选取主要参考 Huang 等提出的模型^[152],并且假设所有齿轮轴无键槽。

(1) 弯曲应力约束

$$g_{1} = \frac{2T}{m_{1}b_{p1}J(N_{p1})d_{p1}}K_{V}K_{O}0.93K_{m} - S_{n}'(H_{1})C_{s}(H_{1})k_{r}k_{t}k_{ms} \leq 0 \quad (5-31)$$

$$g_{2} = \frac{2Td_{w1}}{m_{2}b_{p2}J(N_{p2})d_{p1}d_{p2}}K_{V}K_{O}K_{m} - S_{n}'(H_{2})C_{s}(H_{2})k_{r}k_{t}k_{ms} \leq 0 \quad (5-32)$$

(2) 轴径约束

$$g_3 = \sqrt[3]{\frac{16T}{\pi \cdot \tau_{\text{max}}}} - d_{s1} \le 0 \tag{5-33}$$

$$g_4 = \sqrt[3]{\frac{16T}{\pi \cdot \tau_{\text{max}}} \frac{d_{w1}}{d_{p1}}} - d_{s2} \le 0$$
 (5-34)

$$g_5 = \sqrt[3]{\frac{16T}{\pi \cdot \tau_{\text{max}}} \frac{d_{w1}d_{w2}}{d_{p1}d_{p2}}} - d_{s3} \le 0$$
 (5-35)

(3) 干涉约束

$$g_6 = d_{w1}^a - 2\sqrt{\left(d_{w1}^b/2\right)^2 + a_1^2 \sin^2 \alpha_n} \le 0$$
 (5-36)

$$g_7 = d_{w2}^a - 2\sqrt{\left(d_{w2}^b/2\right)^2 + a_2^2 \sin^2 \alpha} \le 0$$
 (5-37)

(4) 齿数约束

$$g_8 = 17 - n_{p1} / \cos^3 \psi \le 0 \tag{5-38}$$

$$g_0 = 17 - n_{\text{out}} / \cos^3 \psi \le 0 \tag{5-39}$$

$$g_{10} = 17 - n_{p2} \le 0 \tag{5-40}$$

$$g_{11} = 17 - n_{w2} \le 0 \tag{5-41}$$

(5) 齿宽约束

$$0.9 - b_{n1} / m_1 n_{n1} \le 0 ag{5-42}$$

$$b_{p1}/m_1n_{p1} - 1.4 \le 0 (5-43)$$

$$0.9 - b_{p2} / m_2 n_{p2} \le 0 (5-44)$$

$$b_{n2}/m_2n_{n2} - 1.4 \le 0 (5-45)$$

(6) 齿数互质

$$\left(n_{p1}, n_{w1}\right) = 1\tag{5-46}$$

$$(n_{p2}, n_{w2}) = 1 (5-47)$$

5.3 优化结果

本章以 MOEA/D-AW 为求解算法,其具体参数设置如表 5-2 所示。

表 5-2 参数描述与取值

样本量	变化算子	变化算子取值
N = 126	DE/rand/1	CR = 0.5, F = 1

出于对知识产权的保护,初始设计中涉及变量取值及其约束范围修改后取值 如表 5-3 所示。同样地,减速器中其他参数修改后取值如表 5-4 所示。

表 5-3 设计变量描述及其取值

Category	符号	Description	值	范围
	m_1	法向模数	12.7mm	[12.7, 25.4]
	n_{p1}/n_{w1}	齿数	17/140	[15, 25]/[120, 150]
, <i>L</i> TZ	b_{p1}	齿宽	215mm	[195, 300]
一级	χ_{p1}/χ_{w1}	变位系数	0.39/-0.39	[-0.39, 0.39]
	d_{s1}	输入轴轴径	175mm	[150, 200]
	d_{s2}	中间轴轴径	365mm	[320, 380]
	m_2	模数	25.4mm	[12.7, 25.4]
	n_{p2}/n_{w2}	齿数	17/105	[17, 28]/[100, 130]
	b_{p2}	齿宽	540/530mm	[450, 600]
二级	χ_{p2}/χ_{w2}	变位系数	0.4/-0.3	[0.2, 0.5]/[-0.5, -0.2]
	d_{s3}	输出轴轴径	360	[350, 400]
	φ	一级齿轮中心线与水平线夹角	39°	[30, 45]

本章中对约束条件的处理较为简单,采取直接将整个搜索过程严格限制在可行域内的方式。由于在减速器参数设计中,并非所有设计变量都为连续型变量,如齿轮模数为离散型变量,齿数为整数型变量并且相互啮合齿轮齿数互质。为了解决此类问题,本章在优化过程对模数及齿数做连续化对待,在得到连续型帕累托优化解后进行简单处理。对连续型变量值进行修正,取连续型模数优化结果相近的离散值作为最终模数值,对连续型齿数进行圆整,以取得整数型齿数值。同时,在取相近离散值与圆整的过程中做二次验算,以验证所取结果满足约束条件限制。此外,在对齿数进行圆整的过程中,为了满足齿数互质的约束条件,本章首先建立一组向量包含圆整后的齿数 n,n+1,n-1,并结合与之相啮合齿的另外一组向量进行比对,从这两组向量中选取出满足互质条件的两个齿数。

表 5-4 参数描述与取值

符号	描述	值	单位
α	压力角	20°	度
Ψ	螺旋角	15°	度
$h_{\!\scriptscriptstyle n}^*,h^*$	法向齿顶高系数	1	无
χ_n^*, χ^*	法向顶隙系数	0.25	无
C_p	弹性系数	191	\sqrt{MPa}
K_{V}	动态系数	1.51/1.25	无
K_{O}	过载系数	1.25	无
K_{m}	Mounting correction factor	1.5/1.8	无
$S_{\it fe}$	接触疲劳极限	632	MPa
C_R	Surface reliability factor	1	无
k_r	Bending reliability correction factor	0.814	无
k_{t}	温度系数	1	无
k_{ms}	平均应力因子	1.4	无
$S_{n}^{'}$	Standard R.R. Moore endurance limit	324	MPa
C_s	Surface factor	0.4325	无
$ au_{ m max}$	Torsional stress limit	630	MPa
T	输入扭矩	3.3×10^4	$N\cdot m$
ω	输入速度	700	rpm
H_1, H_2	硬度	60	HRC

5.3.1 MOEA/D-AW 计算结果

运行 MOEA/D-AW 算法 31 次,取其中间第 16 次运行结果进行展示。对连续型非支配解进行圆整得到 21 个圆整非支配解,进而计算圆整非支配解集在目标空间的映射作为最终优化结果,分别如表 5-5 和表 5-6 所示。其中,表 5-5 和表 5-6 中加粗行分别代表完全支配初始设计的优化解和其在目标空间的映射,表 5-6 中第一行加粗斜体为初始设计所对应的目标函数值。为了直观地对比非支配解与初始设计,绘制非支配解集在目标空间中的映射与初始设计分别在五个优化目标上的柱状图,如图 5-5 所示。图 5-5 中,最左侧对应初始设计分别对应减速箱总体积V、一级齿轮接触疲劳寿命系数 SF_1 、二级齿轮接触疲劳寿命系数 SF_2 ,一级齿轮承载能力 T_1 和二级齿轮承载能力 T_2 的取值。此外,图 5-6 为所得优化解集分别在五个优化目标取值的均值与初始设计之间的对比图,相对应的变化率如表 5-7 所示。

表 5-5 圆整非支配解集

n_{p1}	n_{w1}	n_{p2}	n_{w2}	m_1	m_2	b_{p1}	b_{p2}	χ_{p1}	χ_{w1}	χ_{p2}	χ_{w2}	d_{s1}	d_{s2}	d_{s3}	θ
17	120	28	100	12.7	16.9	200	600	0.39	-0.39	0.2	-0.5	200	320	450	45
25	120	28	100	12.7	14.5143	295	600	0.39	-0.39	0.2	-0.5	200	320	450	45
25	120	28	100	12.7	14.5143	300	600	0.39	-0.39	0.2	-0.5	200	320	450	45
17	120	28	100	12.7	20.32	200	595	0.39	-0.39	0.2	-0.5	200	320	450	45
23	120	28	100	12.7	16.9	265	590	0.39	-0.39	0.2	-0.5	200	325	450	45
25	120	28	100	12.7	16.9	295	600	0.39	-0.39	0.2	-0.5	200	320	450	45
17	120	28	100	12.7	20.32	200	600	0.39	-0.39	0.2	-0.5	200	320	450	45
25	120	28	100	12.7	16.9	295	590	0.38	-0.38	0.2	-0.5	200	320	450	45
25	120	28	100	12.7	16.9	300	600	0.39	-0.39	0.2	-0.5	200	320	450	45
25	120	28	100	12.7	16.9	290	600	0.39	-0.39	0.2	-0.5	200	320	450	45
25	120	28	100	12.7	20.32	300	600	0.38	-0.38	0.44	-0.5	200	320	450	45
18	120	28	100	16.9	16.9	300	600	0.38	-0.38	0.2	-0.49	200	320	450	45
22	120	28	100	12.7	16.9	265	590	0.39	-0.39	0.2	-0.5	200	320	450	45
23	120	28	100	12.7	16.9	265	600	0.39	-0.39	0.2	-0.5	200	320	450	45
19	120	28	100	16.9	16.9	300	600	0.39	-0.39	0.2	-0.5	200	320	450	45
17	120	26	129	20.32	25.4	300	600	0.37	-0.37	0.2	-0.49	195	320	600	44.3
21	120	28	100	12.7	20.32	245	600	0.36	-0.36	0.2	-0.5	200	320	450	45
25	120	28	100	12.7	20.32	290	600	0.39	-0.39	0.2	-0.5	200	320	455	45
25	124	26	127	12.7	25.4	300	600	0.39	-0.39	0.5	-0.5	200	320	450	45
25	120	28	100	12.7	20.32	300	595	0.39	-0.39	0.2	-0.5	200	320	450	45
25	122	26	129	12.7	25.4	300	600	0.27	-0.27	0.5	-0.47	200	320	450	45

表 5-6 初始设计和圆整非支配解集在每个优化目标上的取值

V	SF_1	SF_2	$T_1(-)$	<i>T</i> ₂ (-)
6.5975	0.0147	0.0143	26.0482	18.9440
3.3749	0.0119	0.0077	24.7910	18.3010
3.9665	0.0042	0.0076	50.1180	18.6040
4.0232	0.0042	0.0076	50.9670	18.6040
4.3503	0.0119	0.0055	24.7910	25.7770
4.3285	0.0054	0.0062	42.0860	22.8570
4.7842	0.0042	0.0057	50.1180	24.760

表 5-6	初始设计和圆整非支配解集在每个优化目标上的取值((金)
10 5-0		、ケスノ

V	SF_1	SF_2	$T_1(-)$	$T_{2}(-)$
6.5975	0.0147	0.0143	26.0482	18.9440
4.3568	0.0119	0.0054	24.7910	25.9940
4.7764	0.0042	0.0058	50.1180	24.3480
4.8523	0.0042	0.0057	50.9670	24.7600
4.7160	0.0043	0.0057	49.2680	24.7600
6.2425	0.0042	0.0040	50.9670	34.9520
5.7533	0.0043	0.0074	65.5190	19.1690
4.3095	0.0058	0.0064	40.5930	22.0890
4.3366	0.0054	0.0061	42.0860	23.2440
5.7830	0.0039	0.0071	68.4420	20.0190
14.4727	0.0034	0.0039	87.2670	36.8490
5.1865	0.0068	0.0046	36.1330	30.6740
6.0434	0.0043	0.0040	49.2680	34.9520
11.9271	0.0042	0.0030	50.9670	47.7990
6.2083	0.0042	0.0041	50.9670	34.6610
12.1484	0.0042	0.0029	50.9670	48.4690

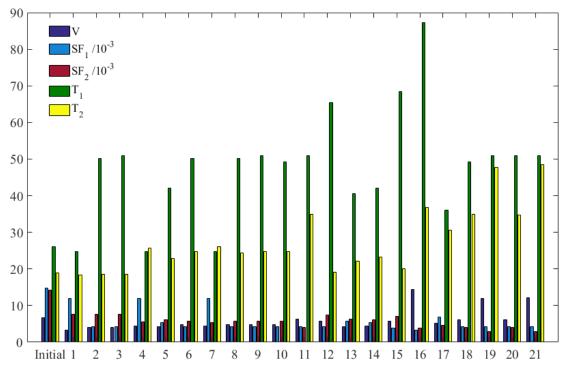


图 5-4 初始设计与圆整非支配解集对比

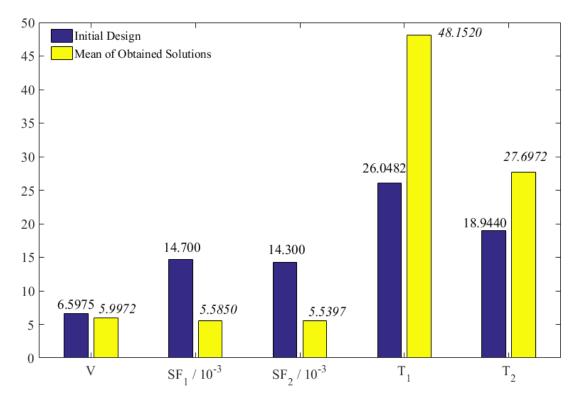


图 5-5 初始设计与圆整非支配解集均值对比

从表 5-7 可以看出,非支配解在优化目标 SF_1 、 SF_2 、 T_1 和 T_2 面都有显著提升。根据计算所得优化解,决策者在实际设计中根据其对体积、接触疲劳寿命与承载能力三方面的特定需求,选取相对应非支配解。

	V	SF_1	SF_2	$T_1(-)$	$T_{2}(-)$
初始设计	6.5975	0.0147	0.0143	26.0482	18.9440
优化解平均值	5.9972	0.0056	0.0055	48.1520	27.6972
变化率	9.10%	62.01%	61.26%	84.86%	46.21%

表 5-7 初始设计与优化解平均值对比

5.3.2 与先进算法对比

如 1.3 节所示,评估某一进化算法的性能时,收敛性与多样性为两个重要的评价指标 $^{[30,132,205]}$ 。选择之前第三章中发现综合性能较高的算法 $PICEA-g^{[84]}$ 与第二章提出的 MOEA/D-IHO 作为对比,验证算法在求解本章优化问题的性能。之所以没有验证第四章提出的 MOEA-IS,因为其所获得优化解为真实帕累托最优前沿某一部分的近似表达,而我们又对于工程实际问题的真实帕累托最优前沿一无所知,无法通过 GD 指标验证其收敛性的好坏,因此本章将不与之进行对比。MOEA/D-IHO

和 PICEA-g 的设置分别如 2.4.4 节和 3.4.4 节所示。

由于本章算例为一具体工程算例,并不清楚其真实帕累托前沿,因此GD与IGD这两个需要真实帕累托前沿作为辅助的性能指标便失去了效用。鉴于此,本节采用HV和PD来综合表征解集在多样性和收敛性上的表现。其中HV的计算采取蒙特卡洛(Monte Carlo)抽样,同时计算过程中所需要的参考点以三种算法在 31次运算过程中在每个目标函数上所对应的最大值乘以 1.2。3 种算法在 HV 和PD上的表现如盒图 5-6 所示。

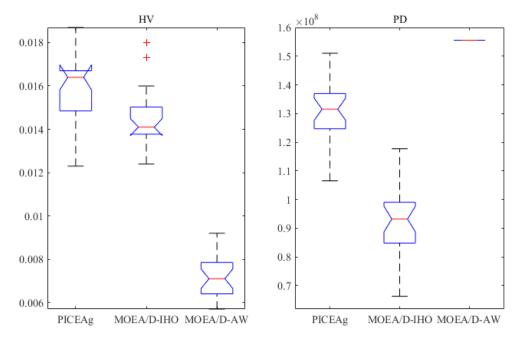


图 5-6 PECEAg、MOEA/D-AW 和 MOEA/D-IHOHV 和 PD 盒图

如图 5-6 所示,本文第三章提出的 MOEA/D-AW 算法与目前处理高维多目标 优化问题表现优异的 PICEA-g 和 MOEA/D-IHO 在 HV 盒图上差距明显,但是在 PD 盒图上表现要优于 PICEA-g 和 MOEA/D-IHO。这一结果在一定程度上佐证了第三章关于 MOEA/D-AW 收敛性不足的结论。MOEA/D-IHO 在 HV 盒图上表现与 PICEA-g 表现接近,PD 盒图上表现差于其他两个算法,这一结果可以归咎于本章 处理约束条件采取的方法,优于变化算子 CSO 的引入,搜索过程在整个决策空间的探索能力较强,很大程度上会出现在可行域外,因此限制了其在多样性上的表现。

此外,根据 5.3.1 节中计算结果,本节假设二级齿轮接触疲劳寿命系数 SF_2 和二级齿轮承载能力 T_2 在优化过程中分别与一级齿轮接触疲劳寿命系数 SF_1 和一级齿轮承载能力 T_1 的进化过程相似,同时 SF_1 和 T_1 在优化过程中的的变化幅度也分别大于 SF_2 和 T_2 。鉴于此,为了更直观地对比三种算法,直接在 3 维空间中展示减速器总体积 V 、一级齿轮接触疲劳寿命系数 SF_1 和一级齿轮承载能力 T_1 的方式,同时

借鉴 Median Attainment Surface^[89]的基本思想,分别列出三种算法获得的非支配优化解集在目标空间的映射,如图 5-7 所示。

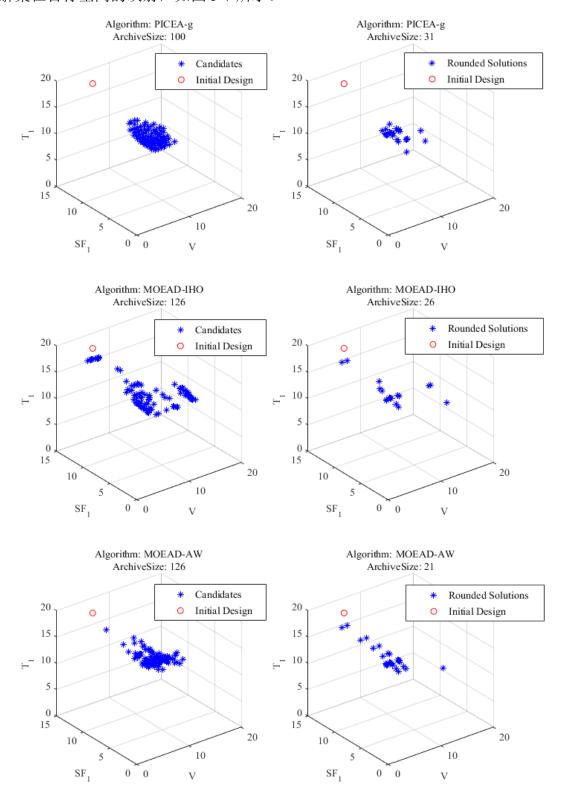


图 5-7 PECEAg、MOEA/D-AW 和 MOEA/D-IHO 算法第 16 次优化解集在目标空间中的映射

值得注意的是,为了使三个不同量纲的优化目标能够在目标空间中看起更为直接,以箱体总体积为基础量纲,在不影响优化过程的基础上分别通过式(5-48)和式(5-49)对 SF_1 、 SF_2 和 T_1 、 T_2 做出相应的调整。

$$sSF_i = sSF_i \cdot 1000, i = 1,2$$
 (5-48)

$$sT_i = 20 - T_i/10, i = 1,2$$
 (5-49)

结合图 5-6 和图 5-7 图示信息,MOEA/D-AW 和 MOEA/D-IHO 在目标空间中广度上的表现要优于 PICEA-g,因此通过第四章提出的 MOEA/D-IS 算法,决策者交互地制定偏好信息,获得的解集能更好地指导实际生产。

5.4 本章小结

本章提出了考虑整体体积、疲劳寿命和承载能力为优化目标的大型矿用挖掘机提升机构减速器优化模型,该模型具有很强的工业背景。为了使优化结果真实可信,首先对该减速器构形进行了细致深入的研究,充分考虑真实齿轮构形对齿轮体积的影响,并应用第三章提出的算法进行求解,优化结果与初始设计对比,验证了算法的可行性。同时,与本文第二章提出的 MOEA-IHO 算法和目前求解 MaOPs 综合性能极佳的 PICEA-g 在多样性上进行比对,也验证了 MOEA/D-AW 算法在多样性保持上的优越性。

第六章 全文总结与展望

6.1 全文总结

本文主要以提升 MOEAs 收敛性和多样性为主线,分别从 1) 平衡决策空间探索与开发,2) 提升 MOEA/D 在目标空间多样性(尤其是均匀性),和 3) 借助决策者信息指导多目标进化方向 3 个角度对 MOEAs 展开了深入的研究。最后,结合国家 863 项目 "75 立方米大型露天矿用挖掘机研制"子课题"大型矿用挖掘机提升机构动态可靠性分析与优化设计关键技术研究",以国产某型号大型矿用挖掘机提升机构减速器为研究对象,以提出的 MOEAs 为工具,求解该减速器多目标优化问题。

本文的主要研究内容及创新成果如下:

- (1)从平衡进化算法在搜索空间探索与开发的角度出发,提出了一种基于混合性能指标综合评价的变化算子自适应选择机制。该机制利用性能评价指标对过往解集的综合评价,选择后续变化操作中适用的变化算子,一方面借助了过往解集在目标空间收敛性和多样性方面综合表现的指导信息,另一方面利用不同变化算子在搜索空间探索与开发的倾向性不同,在不同进化阶段从变化算子选择池中选择不同变化算子,从而提升个体在目标空间中在收敛性和多样性方面的综合表现。其中,针对乌鸦算法在求解单目标优化问题时展现的良好的探索性能,将其与差分进化算子相结合,组成变化算子选择池。以 MOEA/D 为算法基本框架,提出MOEA/D-IHO 算法,与其他 4 个算法在 WFG 测试函数集上进行测试分析,验证了算法的有效性。
- (2) 第三章从提升 MOEA/D 优化解多样性,尤其从分布均匀性的角度出发,提出了一种基于协同进化的权值向量自适应生成机制。该机制结合当前进化过程中子代和权值向量的综合信息,一方面利用当前子代多样性信息(尤其是目标空间解集分布稀疏程度),共同进化以剔除不可行权值向量,另一方面该机制利用了当前子代收敛性信息(最优点),生成最优权值向量进入到下一次进化过程。与其他4 个求解 MaOPs 的优秀算法在 WFG 测试集上进行测试分析,验证了算法的有效性。
- (3)第四章从获取决策者偏好优化解的角度出发,提出了一种基于交互式满意度的偏好信息表征方法。该方法通过不断学习进化过程中生成的新的子代信息,更新决策者对于问题的认识,结合满意度理论不断调整偏好信息,从进化过程中生成的新的子代中,挑选决策者最为满意的个体进行下一次进化。以第二章提出的

MOEA/D-IHO 为基本框架,提出 MOEA/D-IS 算法,与 MOEA/D-IHO 和第二章中的其他 4 个对比算法在 WFG2、WFG7、WFG8 和 WFG9 测试函数上进行测试分析,验证了算法在获取决策者偏好优化解,尤其在探索真实帕累托前沿面边界优化解方面优势明显,同时与其它优秀算法相比还保持可观的收敛性。

(4)以大型矿用挖掘机提升机构减速器多目标优化设计为背景,结合减速器实际应用工况,提出了更为切合实际需求的多目标优化模型,分别以减速器总体积最小,减速器一级、二级齿轮接触疲劳寿命最大,和减速器一级、二级齿轮承载能力最大为优化目标。在减速器总体积优化目标建模中,以齿轮总体积、轴总体积和箱体体积的总和作为体积优化目标。齿轮体积建模中,一方面,由于顶隙和变位系数的存在,对传统齿轮体积计算方法进行修正,另一方面,针对不同齿轮构形搭建不同齿轮体积计算公式。结合之前第二、三章提出的优化算法对该减速器开展多目标优化。分析结果验证了所提算法解决这一实际优化问题的可行性。

6.2 后续工作展望

本文虽然在上述方面取得了阶段性的成果,但是仍然有一些问题值得进一步研究,具体概括如下:

- (1)在自适应变化算子研究中,本文根据性能指标做综合评价时,需要针对不同问题调整 $C \setminus GD$ 和MS的判断因子 $cj \setminus gdj$ 和msj。因此,针对不同的求解问题,如何自适应调整该关键参数仍是一个十分困难的问题。
- (2) 在权值向量自适应生成机制研究中,本文基于 MOEA/D 基本框架的 MOEA/D-AW 算法,在收敛性表现上出现了一定退化。因此,针对这一问题,在 AW 机制中加入自适应分解方法策略将是解决 AW 策略收敛性不足的研究方向,值得进一步深入探讨。
- (3) 在求解 MOPs 过程中,本文用于对比的测试函数集局限于边界约束的优化问题。第五章通过将搜索过程完全限制在可行域内的方式满足所有约束条件,降低了算法的运算效率。因此,开展先进约束条件处理技术研究,与本文提出的 3 种进化算法相结合,是具有有挑战性和有意义的工作。
- (4) 在减速器优化设计建模过程中,本文以安全系数为指导,一定程度上会造成计算结果偏保守。因此,结合可靠性理论,开展基于可靠性的多目标优化设计研究,也是一件十分有意义的工作。

致 谢

至此,漫长而又短暂的博士求学生涯即将完结,内心激动而又平静。衷心的感谢所有帮助过我的老师、朋友和亲人们。

首先,由衷地感谢我的导师黄洪钟教授,感谢他在五年前选择我作为他的博士研究生,使我能够加入电子科技大学系统可靠性与安全性研究中心。在日常的科研工作与学习生活当中,黄洪钟教授以身作则,悉心指导我正确的工作、学习方法,其谆谆教诲是我受益终身的宝贵财富。同时,在日常生活当中,黄老师对我悉心照顾,尽其所能在方方面面帮助我以实现自我提升。"亦师亦父,若宽若言",短短八个字包含了我对黄老师深深的敬佩与感激之情。

感谢李海庆副教授,作为我的硕士研究生导师,李老师对我科研学习的启蒙性指导,为我在博士研究生期间开展科研与学习奠定了良好的基础。

感谢 Tongdan Jin 教授、汪忠来教授、许焕卫副教授在我学术论文的撰写过程中给予的指导、帮助与鼓励。

感谢电子科技大学系统可靠性和安全性研究中心的刘宇教授、朱顺鹏副教授、 张小玲副教授、李彦锋副教授、孟德彪副教授和彭卫文老师,感谢他们在我博士研究生期间给予的关怀与帮助。感谢系统可靠性与安全性研究中心的各位师兄、师姐及其他同学的关心与帮助,感谢庞煜师姐、杨圆鉴师兄、吕志强师兄、王海琨师兄、 左芳君师姐、刘征师姐、高会英师姐、殷毅超同学、米金华同学、张小强同学、黄 承赓同学、周杰同学、郭骏宇同学、李贺同学、黄鹏同学、李享同学,感谢他们在 科研上给予我的帮助与支持,同时感谢他们在日常生活当中对我的照顾,使我的异 乡求学过程充满了爱与关怀。

最后,我要感谢我的父母和我的爱人!感谢父母抚养我成人,并无私地支持我安心完成我的学业,父母养育之恩无以为报。感谢我的爱人,感谢她能够在我博士研究生求学期间将自己托付于我,给予我温馨的支持与关怀,使我能够坚韧的走下去,并激励我奋勇直前。

参考文献

- [1] 袁亚湘. 非线性优化计算方法[M]. 北京: 科学出版社, 2008
- [2] 袁亚湘, 孙文瑜. 最优化理论与方法[M]. 北京: 科学出版社, 1997
- [3] 黄洪钟,赵正佳,关立文,等.基于遗传算法的方案智能优化设计[J]. 计算机辅助设计与图形学学报, 2002, (05): 437-441
- [4] 徐玖平, 李军. 多目标决策的理论与方法[M]. 北京: 清华大学出版社, 2005
- [5] Y. Tian, R. Cheng, X. Zhang, et al. PlatEMO: A MATLAB platform for evolutionary multiobjective optimization[J]. IEEE Computational Intelligence Magazine, 2017, 12(4): 73-87
- [6] S. Jiang, S. Yang. An improved multiobjective optimization evolutionary algorithm based on decomposition for complex pareto fronts[J]. IEEE Transactions on Cybernetics, 2016, 46(2): 421-437
- [7] W. Tong, S. Chowdhury, A. Messac. A multi-objective mixed-discrete particle swarm optimization with multi-domain diversity preservation[J]. Structural and Multidisciplinary Optimization, 2016, 53(3): 471-488
- [8] G. Zavala, A. J. Nebro, F. Luna, et al. Structural design using multi-objective metaheuristics. Comparative study and application to a real-world problem[J]. Structural and Multidisciplinary Optimization, 2016, 53(3): 545-566
- [9] H. Wang, L. Jiao, X. Yao. Two_Arch2: An improved two-archive algorithm for many-objective optimization[J]. IEEE Transactions on Evolutionary Computation, 2015, 19(4): 524-541
- [10] X. Cai, Y. Li, Z. Fan, et al. An external archive guided multiobjective evolutionary algorithm based on decomposition for combinatorial optimization[J]. IEEE Transactions on Evolutionary Computation, 2015, 19(4): 508-523
- [11] C. von Lücken, B. Barán, C. Brizuela. A survey on multi-objective evolutionary algorithms for many-objective problems[J]. Computational Optimization and Applications, 2014,
- [12] N. Al Moubayed, A. Petrovski, J. McCall. D2MOPSO: MOPSO based on decomposition and dominance with archiving using crowding distance in objective and solution spaces[J]. Evolutionary Computation, 2014, 22(1): 47-77
- [13] R. C. Purshouse, K. Deb, M. M. Mansor, et al. A review of hybrid evolutionary multiple criteria decision making methods[C]. IEEE Congress on Evolutionary Computation, Beijing, 2014, 1147-1154

- [14] P. Norio Takahashi, Z. Ren, D. Zhang, et al. Multi-objective optimization approach to reliability-based robust global optimization of electromagnetic device[J]. COMPEL: The International Journal for Computation and Mathematics in Electrical and Electronic Engineering, 2013, 33(1/2): 191-200
- [15] H. Garg, S. P. Sharma. Multi-objective reliability-redundancy allocation problem using particle swarm optimization[J]. Computers & Industrial Engineering, 2013, 64(1): 247-255
- [16] L. H. Lee, E. P. Chew, S. Teng, et al. Finding the non-dominated Pareto set for multi-objective simulation models[J]. IIE Transactions, 2010, 42(9): 656-674
- [17] K. Miettinen. Nonlinear multiobjective optimization[M]. Boston: Kluwer Academic Publishers, 1999
- [18] S. M. Elsayed, R. A. Sarker, E. Mezura-Montes. Self-adaptive mix of particle swarm methodologies for constrained optimization[J]. Information Sciences, 2014, 277: 216-233
- [19] M. Ehrgott. Multicriteria optimization[M]. Heidelberg: Springer, 2005
- [20] A. Trivedi, D. Srinivasan, K. Sanyal, et al. A survey of multiobjective evolutionary algorithms based on decomposition [J]. IEEE Transactions on Evolutionary Computation, 2016: 1-1
- [21] B. Li, J. Li, K. Tang, et al. Many-objective evolutionary algorithms: A survey[J]. ACM Computing Surveys, 2015, 48(1): 1-35
- [22] G. R. Zavala, A. J. Nebro, F. Luna, et al. A survey of multi-objective metaheuristics applied to structural optimization[J]. Structural and Multidisciplinary Optimization, 2014, 49(4): 537-558
- [23] L. Bianchi, M. Dorigo, L. M. Gambardella, et al. A survey on metaheuristics for stochastic combinatorial optimization[J]. Natural Computing, 2009, 8(2): 239-287
- [24] R. T. Marler, J. S. Arora. Survey of multi-objective optimization methods for engineering[J]. Structural and Multidisciplinary Optimization, 2004, 26(6): 369-395
- [25] T. C. Koopmans. Analysis of production as an efficient combination of activities[C]. Activity Analysis of Production and Allocation, New York, 1951, 33-97
- [26] H. Kuhn, A. Tucker. Nonlinear programming[C]. Berkeley Symposium and Mathematical Statitics and Probability, Berkeley, 1951, 481-492
- [27] 焦李成,尚荣华,马文萍,等. 多目标优化免疫算法、理论和应用[M]. 北京: 科学出版社, 2012
- [28] M. Zeleny. Multiple criteria decision making Kyoto 1975[M]. New York: Springer, 1975
- [29] C.-L. Hwang, A. S. M. Masud. Multiple objective decision making-methods and applications: A State-of-the-art survey[M]. New York: Springer, 1979

- [30] A. Zhou, B.-Y. Qu, H. Li, et al. Multiobjective evolutionary algorithms: A survey of the state of the art[J]. Swarm and Evolutionary Computation, 2011, 1(1): 32-49
- [31] K. Deb, S. Jain. Multi-speed gearbox design using multi-objective evolutionary algorithms[J]. Journal of Mechanical Design, 2003, 125(3): 609
- [32] K. Deb. Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques[M]. Boston: Springer, 2014, 403-449
- [33] O. L. De Weck. Multiobjective optimization: History and promise[C]. China-Japan-Korea Joint Symposium on Optimization of Structural and Mechanical Systems, Kanazawa, 2004, Invited Keynote Paper GL2-2
- [34] N. M. Okasha, D. M. Frangopol. Lifetime-oriented multi-objective optimization of structural maintenance considering system reliability, redundancy and life-cycle cost using GA[J]. Structural Safety, 2009, 31(6): 460-474
- [35] H. A. Taboada, J. F. Espiritu, D. W. Coit. MOMS-GA: A multi-objective multi-state genetic algorithm for system reliability optimization design problems[J]. IEEE Transactions on Reliability, 2008, 57(1): 182-191
- [36] K. Sinha. Reliability-based multiobjective optimization for automotive crashworthiness and occupant safety[J]. Structural and Multidisciplinary Optimization, 2007, 33(3): 255-268
- [37] K. Deb, D. Padmanabhan, S. Gupta, et al. Reliability-based multi-objective optimization using evolutionary algorithms[C]. International Conference on Evolutionary Multi-Criterion Optimization, Matsushima, 2007, 66-80
- [38] A. Chinchuluun, P. M. Pardalos. A survey of recent developments in multiobjective optimization[J]. Annals of Operations Research, 2007, 154(1): 29-50
- [39] G. S. Mahapatra, T. K. Roy. Fuzzy multi-objective mathematical programming on reliability optimization model[J]. Applied Mathematics and Computation, 2006, 174(1): 643-659
- [40] S. Kuriakose, M. S. Shunmugam. Multi-objective optimization of wire-electro discharge machining process by Non-Dominated Sorting Genetic Algorithm[J]. Journal of Materials Processing Technology, 2005, 170(1-2): 133-141
- [41] E. Zio, L. R. Golea, C. M. Rocco S. Identifying groups of critical edges in a realistic electrical network by multi-objective genetic algorithms[J]. Reliability Engineering & System Safety, 2012, 99: 172-177
- [42] C. B. Geoffrey, M. K. Ilan. Multi-objective aircraft optimization for minimum cost and emissions over specific route networks[C]. International Congress of the Aeronautical Sciences, Anchorage, 2008

- [43] G. Yuanping, C. Xianbin, Z. Jun. Multiobjective evolutionary algorithm with constraint handling for aircraft landing scheduling[C]. IEEE Congress on Evolutionary Computation, Hong Kong, 2008, 3657-3662
- [44] L. Rui, C. Tian, W. Jianwei, et al. Multi-objective optimization design of gear reducer based on adaptive genetic algorithm[C]. International Conference on Computer Supported Cooperative Work in Design, Xi'an, 2008, 229-233
- [45] L. Zadeh. Optimality and non-scalar-valued performance criteria[J]. IEEE transactions on Automatic Control, 1963, 8(1): 59-60
- [46] Y. Y. Haimes. On a bicriterion formulation of the problems of integrated system identification and system optimization[J]. IEEE Transactions on Systems, Man, and Cybernetics, 1971, 1(3): 296-297
- [47] A. Charnes, W. W. Cooper. Goal programming and multiple objective optimizations: Part 1[J]. European Journal of Operational Research, 1977, 1(1): 39-54
- [48] K. Sastry, D. E. Goldberg, G. Kendall. Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques[M]. Boston: Springer, 2014, 93-117
- [49] R. S. Rossenberg. Simulation of genetic populations with biochemical properties[D]. Ann Harbor: University of Michigan, 1967
- [50] J. D. Schaffer. Multiple objective optimization with vector evaluated genetic algorithms[C]. International Conference on Genetic Algorithms and Their Applications, New Jersey, 1985, 93-100
- [51] J. H. Holland. Adaptation in natural and artificial systems[M]. Ann Arbor: The University of Michigan Press, 1975
- [52] 公茂果, 焦李成, 杨咚咚, 等. 进化多目标优化算法研究[J]. 软件学报, 2009, 20(02): 271-289
- [53] K. Deb, A. Pratap, S. Agarwal, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II[J]. IEEE Transactions on Evolutionary Computation, 2002, 6(2): 182-197
- [54] E. Zitzler, L. Thiele. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach[J]. IEEE Transactions on Evolutionary Computation, 1999, 3(4): 257-271
- [55] Q. Zhang, H. Li. MOEA/D: A multiobjective evolutionary algorithm based on decomposition[J]. IEEE Transactions on Evolutionary Computation, 2007, 11(6): 712-731

- [56] J. J. Liang, A. K. Qin, P. N. Suganthan, et al. Comprehensive learning particle swarm optimizer for global optimization of multimodal functions[J]. IEEE Transactions on Evolutionary Computation, 2006, 10(3): 281-295
- [57] E. Zitzler, L. Thiele, M. Laumanns, et al. Performance assessment of multiobjective optimizers: An analysis and review[J]. IEEE Transactions on Evolutionary Computation, 2003, 7(2): 117-132
- [58] C. A. C. Coello, G. T. Pulido, M. S. Lechuga. Handling multiple objectives with particle swarm optimization[J]. IEEE Transactions on Evolutionary Computation, 2004, 8(3): 256-279
- [59] S. Das, P. N. Suganthan. Differential evolution: A survey of the state-of-the-art[J]. IEEE Transactions on Evolutionary Computation, 2011, 15(1): 4-31
- [60] E. Zitzler, K. Deb, L. Thiele. Comparison of multiobjective evolutionary algorithms: Empirical results[J]. Evolutionary Computation, 2000, 8(2): 173-195
- [61] J. D. Knowles, D. W. Corne. Approximating the nondominated front using the Pareto archived evolution strategy[J]. Evolutionary computation, 2000, 8(2): 149-172
- [62] C. M. Fonseca, P. J. Fleming. An overview of evolutionary algorithms in multiobjective optimization[J]. Evolutionary Computation, 1995, 3(1): 1-16
- [63] M. Laumanns, L. Thiele, K. Deb, et al. Combining convergence and diversity in evolutionary multiobjective optimization[J]. Evolutionary Computation, 2002, 10(3): 263-282
- [64] S. M. Elsayed, R. A. Sarker, D. L. Essam. Adaptive configuration of evolutionary algorithms for constrained optimization[J]. Applied Mathematics and Computation, 2013, 222: 680-711
- [65] K. C. Tan, E. F. Khor, T. H. Lee. Multiobjective evolutionary algorithms and applications[M]. London: Springer, 2005
- [66] J. Knowles, D. Corne, K. Deb. Multiobjective problem solving from nature[M]. New York: Springer, 2008
- [67] K. Deb. Multi-objective optimization using evolutionary algorithms[M]. New York: John Wiley & Sons, 2001
- [68] E. Zitzler, M. Laumanns, L. Thiele. SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization[C]. Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems, Athens, 2001, 95-100
- [69] R. C. Purshouse, P. J. Fleming. Evolutionary many-objective optimisation: an exploratory analysis[C]. IEEE Congress on Evolutionary Computation, Canberra, 2003, 2066-2073
- [70] E. J. Hughes. Evolutionary many-objective optimisation: many once or one many?[C]. IEEE Congress on Evolutionary Computation, 2005, 222-227 Vol.1

- [71] T. Wagner, N. Beume, B. Naujoks. Pareto-, aggregation-, and indicator-based methods in manyobjective optimization[C]. International Conference on Evolutionary Multi-Criterion Optimization, Matsushima, 2007, 742-756
- [72] I. Giagkiozis, P. J. Fleming. Methods for multi-objective optimization: An analysis[J]. Information Sciences, 2015, 293: 338-350
- [73] K. Deb, M. Mohan, S. Mishra. Evaluating the ε-domination based multi-objective evolutionary algorithm for a quick computation of Pareto-optimal solutions[J]. Evolutionary Computation, 2005, 13(4): 501-525
- [74] K. Ikeda, H. Kita, S. Kobayashi. Failure of Pareto-based MOEAs: does non-dominated really mean near to optimal?[C]. IEEE Congress on Evolutionary Computation, Seoul, 2001, 957-962 vol. 2
- [75] S. Yang, M. Li, X. Liu, et al. A grid-based evolutionary algorithm for many-objective optimization[J]. IEEE Transactions on Evolutionary Computation, 2013, 17(5): 721-736
- [76] E. Zitzler, S. Künzli. Parallel Problem Solving from Nature[M]. Heidelberg: Springer, 2004, 832-842
- [77] N. Beume, B. Naujoks, M. Emmerich. SMS-EMOA: Multiobjective selection based on dominated hypervolume[J]. European Journal of Operational Research, 2007, 181(3): 1653-1669
- [78] J. Bader, E. Zitzler. HypE: An algorithm for fast hypervolume-based many-objective optimization[J]. Evolutionary Computation, 2011, 19(1): 45-76
- [79] H. Ishibuchi, T. Murata. A multi-objective genetic local search algorithm and its application to flowshop scheduling[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, 1998, 28(3): 392-403
- [80] K. Deb, H. Jain. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints[J]. IEEE Transactions on Evolutionary Computation, 2014, 18(4): 577-601
- [81] K. Deb, S. Chaudhuri. I-EMO: An interactive evolutionary multi-objective optimization tool[C]. International Conference on Pattern Recognition and Machine Intelligence, Kolkata, 2005, 690-695
- [82] L. Thiele, K. Miettinen, P. J. Korhonen, et al. A preference-based evolutionary algorithm for multi-objective optimization[J]. Evolutionary Computation, 2009, 17(3): 411-436
- [83] R. Wang, R. C. Purshouse, P. J. Fleming. Preference-inspired co-evolutionary algorithms using weight vectors[J]. European Journal of Operational Research, 2015, 243(2): 423-441

- [84] R. Wang, R. C. Purshouse, P. J. Fleming. Preference-inspired coevolutionary algorithms for many-objective optimization[J]. IEEE Transactions on Evolutionary Computation, 2013, 17(4): 474-494
- [85] D. A. V. Veldhuizen. Multiobjective evolutionary algorithms: classifications, analyses, and new innovations[D]. Dayton: Air Force Institute of Technology, 1999
- [86] M. Thakur, S. S. Meghwani, H. Jalota. A modified real coded genetic algorithm for constrained optimization[J]. Applied Mathematics and Computation, 2014, 235: 292-317
- [87] K. Deb, K. Sindhya, J. Hakanen. Decision Sciences: Theory and Practice[M]. Boca Raton: CRC Press, 2016, 145-184
- [88] K. Deb, L. Thiele, M. Laumanns, et al. Scalable multi-objective optimization test problems[C]. IEEE Congress on Evolutionary Computation, Honolulu, 2002, 825-830
- [89] C. M. Fonseca, P. J. Fleming. On the performance assessment and comparison of stochastic multiobjective optimizers[C]. International Conference on Parallel Problem Solving from Nature, Berlin, 1996, 584-593
- [90] G. G. Yen, Z. He. Performance metric ensemble for multiobjective evolutionary algorithms[J]. IEEE Transactions on Evolutionary Computation, 2014, 18(1): 131-144
- [91] E. Zitzler, L. Thiele. Multiobjective optimization using evolutionary algorithms A comparative case study[C]. International Conference on Parallel Problem Solving from Nature, Amsterdam, 1998, 292-301
- [92] P. Czyzżak, A. Jaszkiewicz. Pareto simulated annealing—A metaheuristic technique for multiple-objective combinatorial optimization[J]. Journal of Multi-Criteria Decision Analysis, 1998, 7(1): 34-47
- [93] D. A. Van Veldhuizen, G. B. Lamont. Evolutionary computation and convergence to a pareto front[C]. Late Breaking Papers at The Genetic Programming 1998 Conference, 1998, 221-228
- [94] H. Wang, Y. Jin, X. Yao. Diversity assessment in many-objective optimization[J]. IEEE Transactions on Cybernetics, 2017, 47(6): 1510-1522
- [95] K. Deb, K. Miettinen, S. Chaudhuri. Toward an estimation of nadir objective vector using a hybrid of evolutionary and local search approaches[J]. IEEE Transactions on Evolutionary Computation, 2010, 14(6): 821-841
- [96] H. Ishibuchi, Y. Hitotsuyanagi, N. Tsukamoto, et al. Many-objective test problems to visually examine the behavior of multiobjective evolution in a decision space[C]. International Conference on Parallel Problem Solving from Nature, Krakow, 2010, 91-100

- [97] S. Huband, L. Barone, L. While, et al. A scalable multi-objective test problem toolkit[C]. International Conference on Evolutionary Multi-Criterion Optimzation, Guanajuato, 2005, 280-295
- [98] S. Huband, P. Hingston, L. Barone, et al. A review of multiobjective test problems and a scalable test problem toolkit[J]. IEEE Transactions on Evolutionary Computation, 2006, 10(5): 477-506
- [99] D. E. Goldberg. Genetic algorithms in search, optimization, and machine learning[M]. Boston: Addison Wesley Longman Publishing, 1989
- [100] I. Giagkiozis, R. C. Purshouse, P. J. Fleming. An overview of population-based algorithms for multi-objective optimisation[J]. International Journal of Systems Science, 2015, 46(9): 1572-1599
- [101] C. M. Fonseca, P. J. Fleming. Genetic algorithms for multiobjective optimization: Formulation discussion and generalization[C]. International Conference in Genetic Algorithm, San Francisco, 1993, 416-423
- [102] N. Srinivas, K. Deb. Muiltiobjective optimization using nondominated sorting in genetic algorithms[J]. Evolutionary Computation, 1994, 2(3): 221-248
- [103] J. Horn, N. Nafpliotis, D. E. Goldberg. A niched Pareto genetic algorithm for multiobjective optimization[C]. IEEE Conference on Evolutionary Computation, 1994, 82-87
- [104] S. Kirkpatrick, C. D. Gelatt, M. P. Vecchi. Optimization by simulated annealing[J]. Science, 1983, 220(4598): 671-680
- [105] M. Dorigo, G. D. Caro. Ant colony optimization: a new meta-heuristic[C]. IEEE Congress on Evolutionary Computation, Washington, 1999, 1470-1477
- [106] J. Kennedy, R. Eberhart. Particle swarm optimization[C]. IEEE International Conference on Neural Networks, Perth, 1995, 1942-1948
- [107] R. Storn, K. Price. Differential evolution A simple and efficient heuristic for global optimization over continuous spaces[J]. Journal of Global Optimization, 1997, 11(4): 341-359
- [108] A. Tarakanov, D. Dasgupta. A formal model of an artificial immune system[J]. Biosystems, 2000, 55(1): 151-158
- [109] P. larranaga, J. A. lozano. Estimation of distribution algorithms A new tool for evolutionary computation[M]. Norwell: Kluwer Academic Publishers, 2002
- [110] H. Li, Q. Zhang. Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II[J]. IEEE Transactions on Evolutionary Computation, 2009, 13(2): 284-302

- [111] Q. Zhang, W. Liu, H. Li. The performance of a new version of MOEA/D on CEC09 unconstrained MOP test instances[C]. IEEE Congress on Evolutionary Computation, Trondheim, 2009, 203-208
- [112] Y.-N. Wang, L.-H. Wu, X.-F. Yuan. Multi-objective self-adaptive differential evolution with elitist archive and crowding entropy-based diversity measure[J]. Soft Computing, 2009, 14(3): 193
- [113] K. Li, Á. Fialho, S. Kwong. Multi-objective differential evolution with adaptive control of parameters and operators[C]. International Conference on Learning and Intelligent Optimization, Lion, 2011, 473-487
- [114] K. Li, Á. Fialho, S. Kwong, et al. Adaptive operator selection with bandits for a multiobjective evolutionary algorithm based on decomposition[J]. IEEE Transactions on Evolutionary Computation, 2014, 18(1): 114-130
- [115] K. Li, Q. Zhang, S. Kwong, et al. Stable matching-based selection in evolutionary multiobjective optimization[J]. IEEE Transactions on Evolutionary Computation, 2014, 18(6): 909-923
- [116] S. M. Venske, R. A. Gonçalves, M. R. Delgado. ADEMO/D: Multiobjective optimization by an adaptive differential evolution algorithm[J]. Neurocomputing, 2014, 127: 65-77
- [117] Q. Lin, Z. Liu, Q. Yan, et al. Adaptive composite operator selection and parameter control for multiobjective evolutionary algorithm[J]. Information Sciences, 2016, 339: 332-352
- [118] W. Ning, B. Guo, Y. Yan, et al. Distance-dependent parameter adaption for multi-objective evolutionary algorithm based on decomposition[J]. Soft Computing, 2017,
- [119] R. C. Purshouse, P. J. Fleming. On the evolutionary optimization of many conflicting objectives[J]. IEEE Transactions on Evolutionary Computation, 2007, 11(6): 770-784
- [120] D. Thierens. An adaptive pursuit strategy for allocating operator probabilities[C]. Genetic and Evolutionary Computation Conference, Washington, 2005, 1539-1546
- [121] D. E. Goldberg. Probability matching, the magnitude of reinforcement, and classifier system bidding[J]. Machine Learning, 1990, 5(4): 407-425
- [122] 王晓明. 矿用机械式挖掘机技术概论[J]. 科技创新与生产力, 2010, (12): 70-72+77
- [123] R. Cheng, Y. Jin, M. Olhofer, et al. A reference vector guided evolutionary algorithm for manyobjective optimization[J]. IEEE Transactions on Evolutionary Computation, 2016, 20(5): 773-791

- [124] Y. Yuan, H. Xu, B. Wang, et al. Balancing convergence and diversity in decomposition-based many-objective optimizers[J]. IEEE Transactions on Evolutionary Computation, 2016, 20(2): 180-198
- [125] Z. He, G. G. Yen. Visualization and performance metric in many-objective optimization[J]. IEEE Transactions on Evolutionary Computation, 2016, 20(3): 386-402
- [126] K. Klamroth, K. Miettinen. Integrating approximation and interactive decision making in multicriteria optimization[J]. Operations Research, 2008, 56(1): 222-234
- [127] H.-Z. Huang, Z. Tian, M. J. Zuo. Intelligent interactive multiobjective optimization method and its application to reliability optimization[J]. IIE Transactions, 2005, 37(11): 983-993
- [128] K. Deb, A. Sinha, P. J. Korhonen, et al. An interactive evolutionary multiobjective optimization method based on progressively approximated value functions[J]. IEEE Transactions on Evolutionary Computation, 2010, 14(5): 723-739
- [129] K. Deb, A. Kumar. Interactive evolutionary multi-objective optimization and decision-making using reference direction method[C]. Genetic and Evolutionary Computation Conference, London, 2007, 781-788
- [130] L. While, L. Bradstreet, L. Barone. A fast way of calculating exact hypervolumes[J]. IEEE Transactions on Evolutionary Computation, 2012, 16(1): 86-95
- [131] I. Giagkiozis, R. C. Purshouse, P. J. Fleming. Generalized decomposition[C]. International Conference on Evolutionary Multi-Criterion Optimization, Sheffield, 2013, 428-442
- [132] R. Wang, Q. Zhang, T. Zhang. Decomposition-based algorithms using Pareto adaptive scalarizing methods[J]. IEEE Transactions on Evolutionary Computation, 2016, 20(6): 821-837
- [133] H. Ishibuchi, Y. Sakane, N. Tsukamoto, et al. Adaptation of scalarizing functions in MOEA/D: An adaptive scalarizing function-based multiobjective evolutionary algorithm[C]. International Conference on Evolutionary Multi-Criterion Optimization, Nantes, 2009, 438-452
- [134] 袁源. 基于分解的多目标进化算法及其应用[D]. 北京: 清华大学, 2015
- [135] C. A. Coello Coello. Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art[J]. Computer Methods in Applied Mechanics and Engineering, 2002, 191(11): 1245-1287
- [136] R. A. Gonçalves, C. P. Almeida, A. Pozo. Upper confidence bound (UCB) algorithms for adaptive operator selection in MOEA/D[C]. International Conference on Evolutionary Multi-Criterion Optimization, Cham, 2015, 411-425
- [137] N. Marjanovic, B. Isailovic, V. Marjanovic, et al. A practical approach to the optimization of gear trains with spur gears[J]. Mechanism and Machine Theory, 2012, 53: 1-16

- [138] C. Gologlu, M. Zeyveli. A genetic approach to automate preliminary design of gear drives[J]. Computers & Industrial Engineering, 2009, 57(3): 1043-1051
- [139] S. Golabi, J. J. Fesharaki, M. Yazdipoor. Gear train optimization based on minimum volume/weight design[J]. Mechanism and Machine Theory, 2014, 73: 197-217
- [140] O. Buiga, L. Tudose. Optimal mass minimization design of a two-stage coaxial helical speed reducer with Genetic Algorithms[J]. Advances in Engineering Software, 2014, 68: 25-32
- [141] L. Tudose, O. Buiga, C. Ştefanache, et al. Automated optimal design of a two stage helical gear reducer[J]. Structural and Multidisciplinary Optimization, 2010, 42(3): 429-435
- [142] K. Tamboli, S. Patel, P. M. George, et al. Optimal design of a heavy duty helical gear pair using particle swarm optimization technique[J]. Procedia Technology, 2014, 14: 513-519
- [143] L. P. Pomrehn, P. Y. Papalambros. Discrete optimal design formulations with application to gear train design[J]. Journal of Mechanical Design, 1995, 117(3): 419-424
- [144] T. Yokota, T. Taguchi, M. Gen. A solution method for optimal weight design problem of the gear using genetic algorithms[J]. Computers & Industrial Engineering, 1998, 35(3): 523-526
- [145] 王志刚, 张均富, 王进戈. 齿轮减速器的多目标可靠性优化设计[J]. 机械设计与研究, 2011, 27(6): 44-47
- [146] 蒋春明, 阮米庆. 汽车机械式变速器多目标可靠性优化设计[J]. 汽车工程, 2007, 29(12)
- [147] 李瑰, 张保成, 马学军. 基于层次分析法的齿轮减速器多目标优化设计[J]. 煤矿机械, 2015, 36(5): 32-35
- [148] D. F. Thompson, S. Gupta, A. Shukla. Tradeoff analysis in minimum volume design of multistage spur gear reduction units[J]. Mechanism and Machine Theory, 2000, 35(5): 609-627
- [149] 王安麟, 蒋涛, 昝鹏宇, 等. 大型行星齿轮系统可靠性设计的多目标优化[J]. 中国工程机械学报, 2009, 7(3): 253-257
- [150] 王颖, 王三民, 郭家舜. 高速重载齿轮传动多目标优化设计研究[J]. 机械设计与制造, 2012, (9): 7-9
- [151] 叶秉良, 郭绍义, 戚金明. 考虑磨损问题的圆柱齿轮减速器多目标可靠优化[J]. 煤矿机械, 2008, 29(6): 18-21
- [152] H.-Z. Huang, Z.-G. Tian, M. J. Zuo. Multiobjective optimization of three-stage spur gear reduction units using interactive physical programming[J]. Journal of Mechanical Science and Technology, 2005, 19(5): 1080-1086
- [153] 颜伏伍, 王洪建, 田韶鹏, 等. 基于第二代非支配排序遗传算法的变速器齿轮系多目标可靠性优化[J]. 汽车工程, 2010, 32(3): 234-237
- [154] 赵宁, 杨杰. 高重合度圆柱齿轮传动多目标优化设计[J]. 机械传动, 2012, 36(7): 43-46

- [155] D. Miler, D. Žeželj, A. Lončar, et al. Multi-objective spur gear pair optimization focused on volume and efficiency[J]. Mechanism and Machine Theory, 2018, 125: 185-195
- [156] 成晓升, 余军合, 战洪飞. 基于 NSGAII 的齿轮减速器多目标优化研究[J]. 机电工程, 2014, 31(5): 568-572
- [157] 车林仙, 何兵. 圆柱齿轮传动多目标优化的差分进化算法[J]. 机械传动, 2013, 37(11): 61-66
- [158] Y. Mei, K. Tang, X. Yao. Decomposition-based memetic algorithm for multiobjective capacitated arc routing problem[J]. IEEE Transactions on Evolutionary Computation, 2011, 15(2): 151-165
- [159] R. Shang, J. Wang, L. Jiao, et al. An improved decomposition-based memetic algorithm for multi-objective capacitated arc routing problem[J]. Applied Soft Computing, 2014, 19: 343-361
- [160] S. B. Gee, W. A. Arokiasami, J. Jiang, et al. Decomposition-based multi-objective evolutionary algorithm for vehicle routing problem with stochastic demands[J]. Soft Computing, 2016, 20(9): 3443-3453
- [161] R. Carvalho, R. R. Saldanha, B. N. Gomes, et al. A Multi-Objective Evolutionary Algorithm Based on Decomposition for Optimal Design of Yagi-Uda Antennas[J]. IEEE Transactions on Magnetics, 2012, 48(2): 803-806
- [162] D. Ding, G. Wang. Modified multiobjective evolutionary algorithm based on decomposition for antenna design[J]. IEEE Transactions on Antennas and Propagation, 2013, 61(10): 5301-5307
- [163] M. Gong, L. Ma, Q. Zhang, et al. Community detection in networks by using multiobjective evolutionary algorithm with decomposition[J]. Physica A: Statistical Mechanics and its Applications, 2012, 391(15): 4050-4060
- [164] M. Gong, Q. Cai, X. Chen, et al. Complex network clustering by multiobjective discrete particle swarm optimization based on decomposition[J]. IEEE Transactions on Evolutionary Computation, 2014, 18(1): 82-97
- [165] Q. Cai, M. Gong, S. Ruan, et al. Network structural balance based on evolutionary multiobjective optimization: A two-step approach[J]. IEEE Transactions on Evolutionary Computation, 2015, 19(6): 903-916
- [166] A. Konstantinidis, C. Charalambous, A. Zhou, et al. Multi-objective mobile agent-based sensor network routing using MOEA/D[C]. IEEE Congress on Evolutionary Computation, Barcelona, 2010, 1-8

- [167] A. Konstantinidis, K. Yang. Multi-objective energy-efficient dense deployment in Wireless Sensor Networks using a hybrid problem-specific MOEA/D[J]. Applied Soft Computing, 2011, 11(6): 4117-4134
- [168] C. Qian, Y. Yu, Z.-H. Zhou. An analysis on recombination in multi-objective evolutionary optimization[J]. Artificial Intelligence, 2013, 204: 99-119
- [169] W. Khan, Q. Zhang. MOEA/D-DRA with two crossover operators[C]. UK Workshop on Computational Intelligence, Essex, 2010, 1-6
- [170] M. Črepinšek, S.-H. Liu, M. Mernik. Exploration and exploitation in evolutionary algorithms: A survey[J]. ACM Computing Surveys, 2013, 45(3): 1-33
- [171] A. Askarzadeh. A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm[J]. Computers & Structures, 2016, 169: 1-12
- [172] G.-Z. Fu, H.-Z. Huang, Y.-F. Li, et al. Multi-objective design optimization for a two-stage transmission system under heavy load condition[J]. Mechanism and Machine Theory, 2018, 122: 308-325
- [173] H. Nobahari, A. Bighashdel. MOCSA: A multi-objective crow search algorithm for multi-objective optimization[C]. Conference on Swarm Intelligence and Evolutionary Computation, Kerman, 2017, 60-65
- [174] S. Hinojosa, D. Oliva, E. Cuevas, et al. Improving multi-criterion optimization with chaos: a novel multi-objective chaotic crow search algorithm[J]. Neural Computing and Applications, 2018, 29(8): 319-335
- [175] I. Das, J. E. Dennis. Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems[J]. SIAM Journal on Optimization, 1998, 8(3): 631-657
- [176] M. Li, S. Yang, X. Liu. Diversity comparison of Pareto front approximations in many-objective optimization[J]. IEEE Transactions on Cybernetics, 2014, 44(12): 2568-2584
- [177] 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[J]. IEEE Transactions on Evolutionary Computation, 2014, 18(4): 602-622
- [178] 马晓亮. 基于局部学习和均匀分解的多目标进化算法研究[D]. 西安: 西安电子科技大学, 2014
- [179] S. M. J. Spence, M. Gioffrè, A. Kareem. An efficient framework for the reliability-based design optimization of large-scale uncertain and stochastic linear systems[J]. Probabilistic Engineering Mechanics, 2015,

- [180] J. D. Lohn, W. F. Kraus, G. L. Haith. Comparing a coevolutionary genetic algorithm for multiobjective optimization[C]. IEEE Congress on Evolutionary Computation, Honolulu, 2002, 1157-1162
- [181] 李丙栋. 超多目标演化算法及其应用研究[D]. 合肥: 中国科学技术大学, 2017
- [182] Y. Yuan, H. Xu, B. Wang. Evolutionary many-objective optimization using ensemble fitness ranking[C]. Genetic and Evolutionary Computation Conference, Vancouver, 2014, 669-676
- [183] H. A. Simon. The new science of management decision[J]. The Ford distinguished lectures, 1960, 3: 1-8
- [184] S. Takatsu. Latent satisficing decision criterion[J]. Information Sciences, 1981, 25(2): 145-152
- [185] M. A. Goodrich, W. C. Stirling, R. L. Frost. A theory of satisficing decisions and control[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans, 1998, 28(6): 763-779
- [186] T. Kiyota, Y. Tsuji, E. Kondo. An interactive fuzzy satisficing approach using genetic algorithm for multi-objective problems[C]. IFSA World Congress and NAFIPS International Conference, 2001, 757-762 vol.2
- [187] Y. Tao, H.-Z. Huang, B. Yang. An interactive preference-weight genetic algorithm for multicriterion satisficing optimization[C]. International Conference on Advances in Natural Computation, Xi'an, 2006, 643-652
- [188] M. Rosenblatt. Remarks on a multivariate transformation[J]. The Annals of Mathematical Statistics, 1952, 23(3): 470-472
- [189] J. Branke, K. Deb, K. Miettinen, et al. Multiobjective optimization: Interactive and evolutionary approaches[M]. Berlin: Springer, 2008
- [190] K. Miettinen, M. M. Mäkelä. On scalarizing functions in multiobjective optimization[J]. OR spectrum, 2002, 24(2): 193-213
- [191] K. Miettinen, P. Eskelinen, F. Ruiz, et al. NAUTILUS method: An interactive technique in multiobjective optimization based on the nadir point[J]. European Journal of Operational Research, 2010, 206(2): 426-434
- [192] P. Eskelinen, K. Miettinen. Trade-off analysis approach for interactive nonlinear multiobjective optimization[J]. OR Spectrum, 2012, 34(4): 803-816
- [193] A. B. Ruiz, M. Luque, K. Miettinen, et al. An interactive evolutionary multiobjective optimization method: Interactive WASF-GA[C]. International Conference on Evolutionary Multi-Criterion Optimization, Guimarães, 2015, 249-263

- [194] G. Bonori, M. Barbieri, F. Pellicano. Optimum profile modifications of spur gears by means of genetic algorithms[J]. Journal of Sound and Vibration, 2008, 313(3): 603-616
- [195] K. Miettinen, F. Ruiz, A. P. Wierzbicki. Multiobjective Optimization: Interactive and Evolutionary Approaches[M]. Berlin: Springer, 2008, 27-57
- [196] K. Miettinen, M. M. Mäkelä. Synchronous approach in interactive multiobjective optimization[J]. European Journal of Operational Research, 2006, 170(3): 909-922
- [197] H. Garg, M. Rani, S. P. Sharma. An approach for analyzing the reliability of industrial systems using soft-computing based technique[J]. Expert Systems with Applications, 2014, 41(2): 489-501
- [198] H. Garg, M. Rani, S. P. Sharma, et al. Bi-objective optimization of the reliability-redundancy allocation problem for series-parallel system[J]. Journal of Manufacturing Systems, 2014, 33(3): 335-347
- [199] 王新中. 国内外矿用挖掘机发展状况[J]. 矿山机械, 2004, (09): 52-53+5
- [200] 孙志礼, 李昌, 韩兴. 基于行星减速器的多目标可靠性优化设计方法研究[J]. 机械与电子, 2007, (10): 15-17
- [201] 周鹏, 郭迎福, 刘厚才. 行星齿轮减速器多目标优化设计[J]. 机械工程师, 2014, (05): 1-3
- [202] C. Wang, Z. Y. Shi. Accurate model for the minimum volume design of single stage spur gear train[C]. ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, 2015, V010T11A001-V010T11A001
- [203] I. Doltsinis, Z. Kang. Robust design of structures using optimization methods[J]. Computer Methods in Applied Mechanics and Engineering, 2004, 193(23-26): 2221-2237
- [204] R. C. Juvinall, K. M. Marshek. Fundamentals of machine component design[M]. Danvers: John Wiley & Sons, 2012
- [205] O. P. Yadav, S. S. Bhamare, A. Rathore. Reliability-based robust design optimization: A multiobjective framework using hybrid quality loss function[J]. Quality and Reliability Engineering International, 2010, 26(1): 27-41

攻读博士学位期间参与的科研项目及取得的成果

一、攻读博士学位期间参与的研究项目

- [1] 国家 863 项目: "75 立方米大型露天矿用挖掘机研制",课题编号: 2012AA062001,子课题: "大型矿用挖掘机提升机构动态可靠性分析与优化设计关键技术研究"
- [2] 国家自然科学基金项目:基于时变不确定性分析的复杂机械系统多学科设计优化,项目编号: 51475082
- [3] 总装技术基础科研项目:"机械(结构)可靠性设计与分析软件平台研究",项目编号: YXCDZ20121ZL01
- [4] 中国人民解放军原总装备部十二五预研项目: 典型装备使用寿命设计分析与验证技术-XX 发射装置使用寿命验证技术研究,项目编号: 51319020202
- [5] 国防科工局国防科技工业技术基础科研项目:"复杂装备系统的剩余寿命预测及维护决策理论研究",项目编号: 1420130001

二、攻读博士学位期间取得的成果

- [1] <u>Guo Zhong Fu</u>, Hong-Zhong Huang, Yan-Feng Li, Bo Zheng, Tongdan Jin. Multi-objective design optimization for a two-stage transmission system under heavy load condition[J]. Mechanism and Machine Theory, 2018, 122: 308-325 (SCI, IF: 2.577, JCR 3 🗵)
- [2] <u>Guo-Zhong Fu</u>, Yan-Feng Li, Ye Tao, Hong-Zhong Huang. An interactive preference-based evolutionary algorithm for multi-criterion satisficing optimization[J]. Journal of Intelligent & Fuzzy System, 2018, 34(4): 2503-2511 (SCI, IF: 1.261, JCR 4 区)
- [3] <u>Guo Zhong Fu</u>, Hong-Zhong Huang, Jie Zhou. An adaptive hybrid evolutionary algorithm and its application in redundancy optimization for aeroengine[J]. International Journal of Turbo & Jet-Engines, (己录用, SCI 源刊)(SCI, IF: 0.470, JCR 4 区)
- [4] Xiaoqiang Zhang, Huiying Gao, <u>Guo-Zhong Fu</u>, Hong-Zhong Huang. Reliability analysis of filtering reducer considering the temperature correction and shock load of space[J]. Journal of Shanghai Jiao Tong University (Science), 2018, 23(3): 456-474 (EI 检索)
- [5] Zi-Chen Li, Yi-Ping Liu, <u>Guo-Zhong Fu</u>. Machine tool fault information management system software development[J]. Journal of Donghua University, 2014, 31(6): 760-762. (EI 检索)

- [6] Peng-Hui Liu, <u>Guo-Zhong Fu</u>, Cheng-Geng Huang, Hai-Qing Li, Hong-Zhong Huang. Application of hybrid programming with Matlab and C# in reliability analysis software[J], Journal of Donghua University, 2015, 32(6): 957-960
- [7] <u>Guo-Zhong Fu</u>, Zhuo-Hong Huang, Hai-Qing Li, Yan-Feng Li, Yu Liu, Hong-Zhong Huang. Fault tree analysis on kinematic accuracy of wafer stage using BDD and DFTA technique[C]. International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering, Sichuan, 2013, 260-262 (EI 检索)
- [8] <u>Guo-Zhong Fu</u>, Hong-Zhong Huang, Yan-Feng Li, Yuan-Jian Yang, Yi-Chao Yin. A service life estimation method for offshore electronic control module based on salt spray test[C]. International Conference on Mathematical Methods in Reliability, Tokyo, 2015
- [9] <u>Guo-Zhong Fu</u>, Yuan-Jian Yang, Hua-Ming Qian, Hong-Zhong Huang. Global optimization of a heavy-load gear reducer[C]. International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering, Chengdu, 2017, 876-879
- [10] Zhuohong Huang, Haiqing Li, <u>Guo-Zhong Fu</u>, Rong Yuan, Zhonglai Wang. Software development for mechanical reliability analysis[C]. International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering, Sichuan, 2013, 276-279(EI 检索)
- [11] 米金华,李彦锋,黄洪钟,朱顺鹏,刘宇,**付国忠**,杨圆鉴,殷毅超,张小玲.一种复杂机电系统使用寿命评估方法.中国,发明专利,专利号: ZL2015 1 0088525.X
- [12] <u>付国忠</u>,朱顺鹏,杨圆鉴,殷毅超,米金华,刘宇,黄洪钟,汪忠来 一种基于盐雾环境试验的印制电路板使用寿命评估方法.中国,发明专利,专利申请号: 201510173745.2
- [13] 黄洪钟, <u>付国忠</u>, 刘宇, 李彦锋, 米金华, 罗大双, 张龙龙, 杨圆鉴, 彭卫文. 基于齐次坐标方法的微动台动态可靠性分析方法. 中国, 发明专利, 专利申请号: 201410820168.7
- [14] 黄洪钟, 钱正坤, 黄土地, 李彦锋, <u>付国忠</u>, 米金华, 郭骏宇. 一种基于 Kriging 模型齿轮 传动可靠性评估方法. 中国, 发明专利,专利申请号: 201710446255.4