

代 号 10701

学 号 1022121396

分 类 号 TP183

密 级 公开

题 (中、英文) 目 基于 EDA 和自适应进化策略的多目标优化方法研究

Multi-objective Optimization Algorithms Based on EDA

and Adaptive Evolution Strategy

作 者 姓 名 任元 指导教师姓名、职务 刘芳 教授

学 科 门 类 工学 学科、专业 计算机应用技术

提 交 论 文 日 期 二〇一三年三月

西安电子科技大学

学位论文独创性声明

秉承学校严谨的学风和优良的科学道德，本人声明所呈交的论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢中所罗列的内容以外，论文中不包含其他人已经发表或撰写过的研究成果；也不包含为获得西安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做了明确的说明并表示了谢意。

申请学位论文与资料若有不实之处，本人承担一切法律责任。

本人签名：_____ 日 期：_____

西安电子科技大学

关于论文使用授权的说明

本人完全了解西安电子科技大学有关保留和使用学位论文的规定，即：研究生在校攻读学位期间论文工作的知识产权单位属西安电子科技大学。学校有权保留送交论文的复印件，允许查阅和借阅论文；学校可以公布论文的全部或部分内容，可以允许采用影印、缩印或其它复制手段保存论文。同时本人保证，毕业后结合学位论文研究课题再撰写的文章一律署各单位为西安电子科技大学。

（保密的论文在解密后遵守此规定）

本学位论文属于保密，在____年解密后适用本授权书。

本人签名：_____ 导师签名：_____

日 期：_____ 日 期：_____

摘要

近年来,人工免疫系统、分布估计算法、密母算法等一些新的方法陆续被用于求解多目标优化问题,本文着重对以上几种经典算法进行了研究分析,提出了两种基于分布式估计和自适应进化策略的多目标优化算法。

基于规则模型的分布估计多目标优化算法(RM-MEDA)是一种分布式估计算法,它充分利用了多目标优化问题的最优解集合在决策空间上的分布呈现出一定的规则性这一特点,通过建模采样产生子代个体。该算法在求解变量之间存在线性和非线性多目标优化问题时表现出较明显的优势。但 RM-MEDA 算法中存在一些缺点和不足。其一,分布式估计算法仅通过采样产生新一代的个体,没有利用已找到最优解的位置信息进行局部搜索,因此在种群分布还未呈现出一定规律性时,用概率模型采样策略产生子代的效果较差。其二,经过研究, RM-MEDA 算法中合理的种群聚类数目和具体的测试问题的 Pareto 最优解集合相关,而该算法中需要人为设定种群聚类数目 K 。针对以上问题,本文的主要工作有:

(1)本文在 EDA(Estimation of Distribution Algorithm)和免疫克隆选择算法思想的基础上,结合基于交叉变异的克隆选择算子以及基于 EDA 的模型采样算子,提出了一种进化算子自适应选择的多目标优化算法(MAOS)。本文算法中的自适应密母算子中利用进化过程中的精英个体信息定义能力和效率两个指标来自适应的调整两种算子在进化的不同阶段采用的比例,从而较好地平衡算法的全局搜索和局部搜索的能力。实验结果表明,本章中的自适应密母算子是有效的,同时对三种不同类型的测试函数, MAOS 与 RM-MEDA 和 NSGAII 相比在收敛性和多样性方面均表现出了一定的优势。

(2)在 RM-MEDA 算法框架中引入合并分裂算子从而提出了一种聚类数目自适应调整的 RM-MEDA 算法 (ACRM-MEDA)来求解多目标优化问题。算法中设计的合并分裂算子通过种群流型信息来自适应地合并冗余的聚类,拆分不合理的聚类,从而通过调整后的聚类建立更为精确的分段线性模型来指导种群的进化方向。实验结果表明对于本章中设置的 6 个不同测试问题, ACRM-MEDA 无论在收敛性和多样性方面均优于 RM-MEDA。同时对于 Pareto set(PS)流型为正弦曲线的复杂的测试问题上,相对于仅引入了合并算子的改进 RM-MEDA 算法(RM-MEDA+C), ACRM-MEDA 能够建立更为合理精确的分段线性模型来提高算法的性能。

关键词: 免疫克隆 分布估计算法 自适应算子选择 密母算法 合并分裂算子

Abstract

Recently, some new evolutionary paradigms have been introduced into multi-objective optimization problem (EMO) community, such as artificial immune system, estimation distribution algorithms, and Memetic algorithms. This article focused on above several classical algorithm analysis and research, we proposed two multi-objective optimization algorithms based on Estimation of Distribution Algorithm (EDA) and adaptive evolution strategy.

A Regularity Model-Based Multi-objective Estimation of Distribution Algorithm (RM-MEDA) is one of the distribution algorithms. It can be induced that the Pareto set, in the decision space, of a continuous multi-objective optimization problem show some certain regularity. Based on this regularity property, RM-MEDA build a piecewise liner model and then sample new trial solutions from the model thus built. RM-MEDA has showed the obvious advantage in solving on a set of test instances with variable linkages. However, RM-MEDA also has some drawbacks. Firstly, distributed estimation algorithm only generates the new solutions through sampling, but fail to use the location information of existing optimal solutions for local search. Thus, the method that adopts the probability model sampling strategy to produce offspring is less effective when the population didn't show the certain regularity in the distribution of population. Secondly, the reasonable number of clusters of the population is related to the Pareto optimal solution set of the specific test questions, but in RM-MEDA, the number of cluster need to artificially set the number of clusters K . To solve the above problems, the main research work of this paper includes the following:

(1)Based on Estimation of Distribution Algorithm (EDA) and Immune Clonal Selection Algorithm (ICSA), this paper proposed a Memetic Algorithm with Adaptive Operator Selection for multi-objective optimization(MAOS). In MAOS, two types of reproducing strategies are combined: one is the recombination and mutation based immune clonal selection operator, the other is EDA based modeling and sampling operator. In MAOS, the adaptive Memetic operator definite two indicators: capacity and efficiency by using the historical information of elite individuals to adaptively adjust both operators used in different stages of evolution, thus MAOS can balance the exploitation and exploration of the algorithm. Experimental studies have shown that the adaptive Memetic operator was efficient. And MAOS performs better than RM-MEDA, NSGAII on a set of test instances with linear and nonlinear variable linkages and ZDT

problems.

(2) We proposed an improved RM-MEDA Algorithm based on adaptive clustering operators (ACRM-MEDA) by introducing the merging and dividing operator into the framework of RM-MEDA. The merging and dividing operator in ACRM-MEDA use the manifold information to merge redundant clustering, divide unreasonable clustering. Thus, through the adjusted clusters we can obtain more accurate piecewise linear model to guide the direction of evolution of the population. Experimental studies have shown that ACRM-MEDA performs better than RM-MEDA in both convergence and diversity on the 6 test problem designed in the paper. Besides, for the complex test problem, the Pareto set(PS) manifold of which is the Sine curve, ACRM-MEDA can obtain the more reasonable and accurate piecewise linear model to improve the performance of the algorithm than RM-MEDA+C which only adopts the merging operator.

**Keyword: Immune Cloning EDA Adaptive Operator Selection
Memetic Algorithm Merging and Dividing Operator**

目录

第一章 绪论	1
1.1 多目标优化问题的产生背景	1
1.2 进化多目标优化方法及其研究现状	1
1.3 分布式估计算法和密母算法和研究的发展	4
1.4 本文的创新动机和主要工作	6
1.5 本文的组织结构	7
第二章 进化多目标优化算法	9
2.1 多目标优化问题的数学描述	9
2.2 免疫多目标优化算法	10
2.2.1 免疫克隆选择算法	10
2.2.2 非支配邻域免疫算法 NNIA	13
2.3 基于 EDA 的进化多目标优化算法	14
2.3.1 EDA 用于求解多目标优化问题	14
2.3.2 基于 EDA 的进化多目标优化算法 RM-MEDA	14
2.4 本章小结	18
第三章 自适应算子选择密母多目标优化算法	19
3.1 引言	19
3.2 自适应选择密母算子	20
3.3 算法 MAOS 的框架设计	22
3.4 仿真实验	24
3.4.1 标准测试函数	25
3.4.2 性能评价指标	28
3.4.3 实验参数设置	28
3.4.4 实验结果及分析	29
3.4.4.1 MAOS 算法有效性分析	29
3.4.4.2 MAOS 算法自适应策略的有效性分析	36
3.4.4.3 MAOS 算法中自适应密母算子的权值参数分析	39
3.5 本章小结	40
第四章 聚类数目自适应调整的 RM-MEDA	43
4.1 引言	43
4.2 算法的基本思想	44
4.3 算法的描述	47
4.4 算法分析	47
4.5 仿真实验	48
4.5.1 标准测试函数	48
4.5.2 实验参数设置	49

4.5.3 实验结果及分析.....	50
4.6 本章小结.....	53
第五章 总结与展望	55
5.1 论文工作总结.....	55
5.2 进一步工作展望.....	56
致谢.....	57
参考文献.....	59
研究成果.....	65

第一章 绪论

1.1 多目标优化问题的产生背景

在实际应用中，普遍存在着需要同时优化两个或两个以上目标的多目标优化问题（Multiobjective Optimization Problem, MOP）。多目标优化问题的研究在其近些年来发展十分迅速，已经逐步形成了一门新兴学科，其在现实生活的各个领域均发挥着广泛和重要的作用，同时很多学者对其进行了不断深入的研究，使多目标优化问题的应用范围已经扩展到了涉及到计算科学，航空航天、生物科学、过程控制、人工智能等许多实际复杂系统的建模、规划以及设计等问题。多目标优化问题往往会涉及到优化若干个相互冲突的目标，因此在多目标优化问题不存在一个能使所有目标同时达到最优的解，多目标优化算法的目的是获得一组具有代表性的 Pareto 最优解的集合（Pareto set, PS），使得这些 Pareto 最优解在目标空间 Pareto 前端(Pareto front, PF)上的分布具有尽可能好的逼近性，宽广性和均匀性^[1]。

多目标优化问题对最优解的定义与单目标优化问题是有很区别的，对于一个特定的多目标优化问题，对于某个目标来说其最优解可能是较好的，但是对于其它优化目标来说最优解的优化效果则可能是较差的。同时与单目标优化问题的最大区别是，多目标优化问题的最优解而是一组最优解的集合，称为 Pareto 最优解集。这个区别也为求解多目标问题带来了许多困难，同时由于基于数学规划原理的多目标优化方法往往不能够很好求解实际优化问题，所以，有必要研究求解多目标优化问题的高效算法及理论。

1.2 进化多目标优化方法及其研究现状

在多目标优化算法研究的初期阶段，一般通过求和加权等方式将多目标优化问题转化为单目标优化问题，然后通过数学规划的方法的最优解，而这种方法的缺陷就是每次只能得到在一种固定权值下的最优解，因此求解效率对权值等参数较为敏感。同时，多目标优化问题的约束函数和目标函数一般可能是不可微或是不连续、非线性的，传统的数学规划方法往往效率较低。

进化算法是模拟生物在自然界中进化过程而形成的一类自适应全局优化概率搜索算法,进化算法同时也是一种启发式搜索算法。近些年许多学者倾向于进化算法来求解多目标优化问题，从而发展成为了一个相对较热的研究方向即进化多目标优化。进化算法通过代与代之间维持的潜在解种群来实现全局搜索，这种从种群到种群的方法对于搜索多目标优化问题的 Pareto 最优解集是很有用的。进化多

目标优化算法体现了现代科学发展中, 生命科学与工程科学相互交叉、相互渗透和相互促进的特征和趋势。Schaffer 在 1985 提出了求解多目标优化问题的开创性算法矢量评价遗传算法 Vector Evaluated Genetic Algorithm(VEGA)^[2]。其第一代进化多目标优化算法的思想始于 Goldberg 的创新性想法, 小生境技术和非支配排序被提出用来求解多目标优化问题。其中, 非支配排序策略的主要步骤为: 首先, 选出当前种群中的非支配个体并设置其等级为 1, 然后将这些非支配个体从竞争中移除; 其次, 继续从当前进化种群中选出新的非支配个体, 并把其等级设置为 2, 该过程一直持续到种群中所有个体都被设置了等级后结束。其中 Goldberg 还使用小生境技术用来保持种群多样性, 从而防止早熟。Goldberg 虽然没有在进化多目标优化的研究中应用到他的思想, 但其对以后从事多目标优化问题研究的学者来说, 具有很大的启发和指导意义。20 世纪 90 年代以后, 很多学者相继提出了多种不同的进化多目标优化算法。1993 年, Fleming 和 Fonseca 提出了 Multi-objective Genetic Algorithm(MOGA)^[3], Deb 和 Srinivas 提出了 Non-Dominated Sorting Genetic Algorithm(NSGA)^[4], Horn 等提出了 Niched Pareto Genetic Algorithm(NPGA)^[5], 这些算法一般被称为第一代进化多目标优化算法, 第一代进化多目标优化算法的特点是采用基于 Pareto 等级的个体选择方法和基于适应度共享机制的种群多样性保持策略。同时随着第一代进化多目标优化算法的研究的不断发展和深入, 一些亟需解决的相关问题也逐步凸显了出来。例如, 是否能找到替代小生境技术来保持种群的多样性等。

从 20 世纪末期开始, 基于精英保留机制的第二代进化多目标优化算法相继被提出来。其中, 在 1999 年, Zitzler 和 Thiele 提出 Strength Pareto Evolutionary Algorithm(SPEA)^[6], 在该算法中, 个体的适应度被称为 Pareto 强度, 该算法中对非支配集中个体的适应度做了如下定义, 即个体的适应度为支配他的个体总数加 1。同时该算法中规定适应度低的个体对应着较高的选择概率。最重要的是, 该方法使精英保留机制在进化多目标优化问题研究中被越来越多的学者们所应用。之后, 他们提出了 SPEA 的改进版本 SPEA2^[7], SPEA2 主要在以下 3 个方面进行了改进, 其分别为及非支配解集的更新、适应度分配策略以及个体分布性的评估方法。2000 年, Knowles 等提出了 Pareto Archived Evolution Strategy(PAES)^[8], 在算法 PAES 中主要采用了(1+1)进化策略对当前一个解进行变异操作, 其后对执行了变异操作后的个体进行函数评价, 同时将其与变异前个体的支配关系作以比较, 采用精英保留策略保留其中较好的, 该算法的特点在于引进了空间超格的机制来保持种群的多样性, 其中每一个个体分配进一个格子。随后, 他们也提出了改进的版本 Pareto Envelope-Based Selection Algorithm(PESA)^[9]和 PESA-II^[10]。PESA 设置了一个内部种群和一个外部种群, 算法迭代时将内部种群的非支配个体会被添加到外部种群中, 当外部种群中添加一个新个体时, 相应的要在外部种群中删除一个

个体，具体的操作方法是在外部种群中删除拥挤系数最大的个体，如果出现多个个体的拥挤系数相同时，则从中选取一个随机地删除。这里把一个个体所对应的超格中所聚集个体的数目定义为该个体的拥挤系数。PESA 是基于个体选择的算法，而在算法 PESAI 中提出了基于区域选择的概念，PESAI 中采用网格选择代替 PESA 中个体选择的方法，从而较好的改善了算法的效率。其后，2002 年，Deb 等学者通过对 NSGA 进行改进，NSGAII^[13] 作为一个非常经典的多目标优化算法被提了出来。这一时期，一些更好的策略被提出来，比如基于聚类的方法，基于拥挤距离的方法，基于空间超格的方法等。

进化多目标优化前沿领域的研究在 2003 年以后呈现出许多新的特点，更多的学者们开始关注如何更有效地求解高维多目标优化问题，一些有别于第二代多目标优化算的 Pareto 占优的新型占优机制陆续被提了出来。Deb 和 Laumanns 和等学者提出了 ε 占优的概念^[14]，同时部分占优^[15]的概念也被 Brockhoff 和 Zitzler 等学者提了出来，此后 Pareto 自适应 ε 占优^[16]的概念也被 Alfredo 和 Coello Coello 等学者提了出来。同时对多目标优化问题本身的研究也在逐步深入，不同性质的多目标优化测试问题相继被提出来。同时，一些新的进化机制也被引入多目标优化领域，如 Coello Coello 等人基于粒子群优化提出的 Multi-objective Particle Swarm Optimization(MOPSO)^[17]。人工免疫系统 Artificial Immune Systems AIS(AIS)是受免疫学启发的自适应系统，它通过模拟免疫学的基本原理、功能和模型来解决复杂问题。Gong 和 Jiao 利用免疫克隆选择进化机制在保持种群多样性的优势提出了非支配近邻免疫多目标优化算法 NNIA^[18](Multi-objective Immune Algorithm with Nondominated Neighbor-based Selection)，它是效率和效果都较好的一个多目标优化算法。NNIA 模拟了免疫响应中具有多样性的抗体共同存活，而仅少数抗体被激活这种现象，采用了基于非支配邻域的个体选择方法这种方法，仅选择出少数相对孤立的非支配个体作为活性抗体，同时根据活性抗体的拥挤程度对其进行按比例克隆复制，对克隆后的抗体群采用了有别于 GA 的重组操作和变异操作，从而能够进一步加强对 Pareto 前沿面中较为稀疏的区域进行搜索。分布估计算法是进化计算领域新兴的分支，它是进化算法和统计学习的有机结合。Zhang 和 Zhou 等人基于分布估计算法提出 Regularity Model Based Multi-Objective Estimation of Distribution Algorithm(RM-MEDA)^{[19][20]}。近两年，Zhang 和 Li 将这种传统的多目标求解策略与进化算法相结合构造了一种新颖的基于分解的多目标进化算法 MOEA/D^[21](A Multi-objective Evolutionary Algorithm Based on Decomposition)，它是一种新的优化模式。MOEA/D 通过同时优化一系列不同的单目标子问题保持了很好的多样性，在解决有复杂 PS 形状的多目标优化问题上，优于基于 Pareto 选择的多目标优化算法。此外，粒子群优化、蚁群算法、人工免疫系统、协同进化算法、密母算法、文化进化算法等一些新的进化范例陆续被用于求解多目标优化问

题。

1.3 分布式估计算法和密母算法和研究的发展

分布估计算法(Estimation of Distribution Algorithms,EDAs)是进化计算领域兴起的一类新型优化算法,分布估计算法的概念最初于 1996 年提出,在 2000 年前后迅速发展,成为当前进化计算领域前沿的研究内容,在 EDAs 中,最重要的是概率模型的学习和采样,而根据不同的概率模型的特点,研究者提出了许多不同算法。EDAs 最初针对变量无关的问题提出,美国卡耐基大学的 Baluja 在 1994 年提出的用于解决二进制编码优化问题的 PBIL(Population-Based Incremental Learning)算法是公认最早的分布估计算法模型^[22]。2004 年,Thierens 等把分布估计算法(EDA)引入了多目标优化领域^[23],其后诸多分布式估计算法^[24,25,26]在求解多目标优化问题中取得了很大的成功。然而此类分布式估计多目标优化算法都忽略了 Pareto 解集在决策空间中分布呈现一定的规律性:一个连续的 m 个目标的多目标优化问题的 Pareto 解集是一个分段连续的 $m-1$ 维流形。而近几年,比较有代表性的算法是 Q.Zhang 等人在 2007 年提出 RM-MEDA^[21],该算法充分利用了连续多目标优化问题的特点,即对于目标个数为 m 的连续多目标问题,由 Karush-Kuhn-Tucker 条件可知,Pareto 最优解集合(Pareto Set,PS)在决策空间上的分布呈现分段连续的 $(m-1)$ -维流形,总之,EDA 算法取得了很大进展,成为目前多目标优化算法研究的热点之一。

同时,与传统的启发式优化算法相比,进化算法的优势在于对解空间的迭代搜索过程中引入了随机性,从而不容易陷入局部最优解。然而,也正是由于随机性的引入,使得进化算法也存在求解效率不高的问题。因此,一些学者通过将具有全局搜索特性的进化算法和具有高效局部搜索能力的启发式算法相结合,提出了很多混合算法,这些混合算法被称为Memetic 算法(MAs)^[27]。MAs是受到 Darwinian提出的自然进化理论和Dawkins提出的Meme概念而提出的Memetic算法。MAs是一种基于种群的全局搜索和基于个体的局部启发式搜索的结合体,它以互补的方式将不同方法的性能相结合,使得这种混合算法具有更好的鲁棒性和有效性^[27]。在MAs中,Meme通常被认为个体对于局部搜索能力的学习过程。Memetic 算法不仅可以应用在单目标优化问题中,而且也越来越多的应用于多目标优化问题^[28]。Memetic算法通过在进化算法中引入启发式局部搜索策略从而提高了进化算法的求解能力,进一步推动了进化算法在实际工程中的有效应用。

目前,多目标 Memetic 大致包括两种类型,其中第一类多目标 Memetic 算法^[29,30,31,32],在局部搜索时使用一个加权求和的适应度函数来决定解的优劣程度,这类多目标 Memetic 算法的典型代表包括 MOGLS 和 PMA 等。在 1998 年,Ishibuchi 等

人首次提出了一种结合局部搜索的多目标混合算法, 即 A Multiobjective Genetic Local Search Algorithm (MOGLS) [29][30]。MOGLS 在选择父代时, 使用了结合随机加权的标量化适应度函数, 并对其子代使用局部搜索的方法。Jaszkiewicz 在 MOGLS 的基础上提出了一种针对多目标组合优化问题的遗传搜索算法[31]。该算法在选择父代时没有对整个种群使用“轮盘赌”选择, 而是在局部搜索中引入了一种概率选择机制。在该文章中认为使用高选择压力的父代选择机制可以提高原有 MOGLS 算法的性能。随后, Jaszkiewicz 又提出了一种改进版本, 被称为 Pareto Memetic Algorithm(PMA)[32]。Jaszkiewicz 同时指出, 在增加种群多样性方面, 标量化函数方法要比支配排序方法好的多。第二类多目标 Memetic 算法的典型代表有 M-PAES, Generalization of Dominance Relation-Based Replacement Rules for Memetic EMO Algorithms 等。Knowles 等在 PAES[8]算法的基础上引入局部搜索策略, 提出了 memetic-PAES algorithm (M-PAES) [33]。M-PAES 算法中沿用了 PAES 中精英保留策略, 但没有使用标量化函数, 新个体由遗传搜索和局部搜索产生, 然后使用 the Pareto-dominance relation and the grid-type partition of the objective space 来决定接受或者拒绝。Murata 等人提出了 Generalization of Dominance Relation-Based Replacement Rules for Memetic EMO Algorithms [34]。该算法在局部搜索算子中混入两种替换规则, 第一种是用支配当前解的另一个解来替换, 另一种是用当前解的一个非支配解来替换, 从而成功地改善了已有算法 SPEA 和 NSGAII 的性能。在以上的研究都表明, 通过引入一种局部搜索方法, 进化算法的收敛速度和收敛精度等性能能够被加强。多目标 Memetic 算法兼顾了全局搜索和局部搜索, 在求解质量和求解效率之间获得了良好的折中, 引起了研究者的广泛关注。因此, 随着 EDA 的发展以及它在求解多目标优化问题表现独有的优势, 许多传统进化算法和 EDA 相结合的 Memetic 算法被提出来, J.M.Pena 等学者提出了 GA-EDA[35], 该算法将构造概率图模型采样产生子代和遗传算法产生子代相结合, 利用了两者的优点, GA 使用重组和变异算子作为从上一代种群的个体中获取新个体的手段, 而 EDA 对新个体构造概率模型采样产生新个体。在 GA-EDA 中, 第 $k+1$ 代种群中的个体来自三个方面: 第 k 代种群, GA 演化产生的后代和 EDA 演化产生的后代, 但没有利用连续多目标问题 Pareto 解集分布的规则性 Pareto 解集分布的规则性。Zhang 和 Zhou 等学者提出 M-MOEA[36]算法用于求解二目标问题, 该算法在进化的奇偶代分别用 EDA 和 NSGAII 的方法产生子代, 在部分测试函数上取得了比 NSGAII 更好的效果。在此基础之上, Zhang 和 Zhou[37]等学者通过采用一个收敛准则来判断是采用遗传操作还是通过建立的概率模型产生新一代种群, 更好的利用了连续多目标优化问题 Pareto 最优解集分布的规则特性。

1.4 本文的创新动机和主要工作

RM-MEDA 作为一种分布式估计算法,它通过建模采样产生个体,对求解决策变量之间存在线性和非线性关联的复杂多目标优化问题非常有效。但 RM-MEDAz 中也存在以下两个缺点:其一, RM-MEDA 中没有利用已找到最优解的位置信息进行局部搜索,而免疫克隆选择算法通过对较优的个体执行克隆、重组、变异操作进化,但忽略了种群内部固有的特征信息。其二, RM-MEDA 算法中种群的聚类数目是固定设置的,而合理的聚类数目往往是和具体测试问题相关的。因此本文中对这分布式算法以及免疫克隆选择算法的思想、流程、优缺点进行了详细的分析,本文主要做了两方面的工作:

(1) 在刘静乐师姐等提出的算法 HIAEDA^[38]中,两种子代繁殖策略的中,在产生新个体时两种算子采用比例是固定的,并未使两种算子在进化的不同阶段发挥各自的作用,因此在基于免疫多目标优化算法以及分布估计算法(EDA)基础上,提出了一种求解复杂多目标优化问题的自适应算子选择密母算法 MAOS(Memetic Algorithm with Adaptive Operator Selection for Muti-objective optimization)。MAOS 的进化过程混合了两种后代产生策略:一种是基于交叉变异的克隆选择算子,用于在父代种群周围进行局部搜索的同时开辟新的搜索区域;另一种是基于 EDA 的模型采样算子,用于学习多目标优化问题自变量之间的相关性,提高算法求解复杂多目标优化问题的能力。论文在分析两种算子的搜索行为的基础上,分别定义了能力和效率两种指标来自适应的调整两种算子在算法不同阶段的使用比例,并利用有限马尔科夫链的性质证明了 MAOS 算法的收敛性。仿真实验结果表明, MAOS 与 NSGAI 算法和基于 EDA 的进化多目标优化算法 RMMEDA 相比,在收敛性和多样性方面均都表现出明显优势,尤其对于函数形式较为复杂多目标优化问题优势更为突出。同时,仿真实验结果也进一步验证了 MAOS 算法的鲁棒性以及其算子自适应选择策略的有效性。

(2) RM-MEDA 算法中指出了 RM-MEDA 中聚类的个数与具体问题相关,而在原算法中对于不同问题的聚类数目 K 设置成了常量(聚类数目 K 为 5),然而对于不同的测试问题,其 PS 形状不同,相应需要合理的聚类数目来更准确的分段逼近真个 PS 流形。因此本章提出了基于自适应聚类算子的 RM-MEDA 算法(ACRM-MEDA)。在该算法中,合并分裂算子来根据种群流形信息自适应调整聚类数目实现子种群的自适应合并和分裂,从而建立更精确的线性模型来产生后代种群。

1.5 本文的组织结构

在本文第一章绪论中,首先概述了多目标优化问题的产生背景,其后对进化多目标优化方法及其研究现状进行了描述,详细阐述了分布式估计算法和 **Memetic** 多目标优化算法的研究现状以及它们在多目标优化问题中的应用,接下来重点分析了本文的创新动机和主要工作,最后给出了本文的组织结构。

在本文的第第二章,首先详细介绍了多目标优化问题的具体描述,其次重点阐述了免疫克隆选择多目标优化算法和基于 **EDA** 的进化多目标优化算法这两类算法,其中详细介绍了基于非支配近邻的 **NNIA** 和 **RM-MEDA** 算法的基本思想、主要流程以及局限性分析,最后给出了本章小结。

在本文的第三章,首先基于免疫算法和 **EDA** 的混合多目标优化算法做了相应的介绍,同时对算法中的混合子代繁殖策略中涉及到的两种算子的搜索行为进行了进一步的分析,讨论了两者在功能上的互补性,在此基础上提出了进化算子自适应选择的多目标优化算法(**MAOS**),并对实验结果进行了对比分析,最后对本章进行了小结。

在本文的第四章,提出了一种聚类数目自适应调整的 **RM-MEDA** 算法(**ACRM-MEDA**)。本章中给出了该算法的基本思想和流程,同时测试了一系列测试函数,并对结果进行了详细分析,最后对本章进行了小结。

在本文的第五章,对本文进行了总结并且给出了文章的不足之处以及进一步需要研究的工作。

第二章 进化多目标优化算法

进化算法作为一种启发式的搜索算法，已经被成功地用于多目标优化问题。进化算法通过父代和子代之间保持的候选解种群来实现全局搜索，对于搜索多目标优化问题的 Pareto 最优解来说，这种从种群到种群的方法是很有用的。针对多目标优化问题的多目标优化算法称为进化多目标优化算法，至今为止已经有三代多目标优化算法被相继提出来，而且近几年进化多目标优化前沿领域的研究呈现出新的特点，其中分布估计算法、人工免疫系统等一些新的范例陆续被用于求解多目标优化问题，从而能够更有效的求解维数更高函数形式更为复杂的多目标优化问题。

本章首先介绍了多目标优化问题的数学模型和相关概念，其次阐述了免疫克隆选择算法的原理和研究现状以及分布式估计算法的思想，并对经典算法 NNIA 和 RM-MEDA 算法的思想、流程以及局限性做了详细的描述。

2.1 多目标优化问题的数学描述

由于最小化问题和最大化问题可以相互转换，一般来说，一个具有 n 个决策变量， m 个最小化目标的多目标优化问题可描述为

$$\begin{aligned} & \text{Minimize} \quad \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x})) \\ & \text{Subject to} \quad \mathbf{x} \in \Omega \end{aligned} \quad (2-1)$$

其中， $\Omega \in \mathbf{R}^n$ 是 n 维决策空间的可行区域， $\mathbf{x} = \{x_1, x_2, \dots, x_n\} \in \Omega$ 为可行区域内的决策向量，目标函数 $F(x)$ 定义了一个由 n 维决策空间向 m 维目标空间的映射函数。在此基础上，给出以下几个重要的定义：

定义 1 可行解 对于某个 $x \in X$ ，如果 x 满足式(2-1)中的约束条件， $g_i(x) \leq 0 (i=1, 2, \dots, q)$ 和 $h_j(x) = 0 (j=1, 2, \dots, q)$ ，则称 x 为可行解。

定义 2 可行解集合 可行解集合由 X 中的所有可行解组成的集合，记为 X_f ，且 $X_f \subseteq X$ 。 X_f 的映射，即目标空间可行区域，定义为 $Y_f = f(X_f)$ 。

定义 3 Pareto 占优 假设 $x_A, x_B \in X_f$ 是式(2-1)所定义的多目标优化问题的两个可行解，称与 x_B 相比， x_A 是 Pareto 占优的，当且仅当

$$\forall i=1, 2, \dots, m \quad f_i(x_A) \leq f_i(x_B) \quad \wedge \quad \exists j=1, 2, \dots, m \quad f_j(x_A) < f_j(x_B) \quad (2-2)$$

记作 $x_A \succ x_B$ ，也叫做 x_A 支配 x_B 。

定义 4 Pareto 最优解 一个解 $x^* \in X_f$ 被称为 Pareto 最优解（或称为非支配解），当且仅当满足式(2-3)的条件：

$$\neg \exists x \in X_f : x \succ x^* \quad (2-3)$$

定义 5 Pareto 最优解集 所有 *Pareto* 最优解的集合称为 *Pareto* 最优解集 PS ，定义如下：

$$P^* \triangleq \{x^* \mid \neg \exists x \in X_f : x \succ x^*\} \quad (2-4)$$

定义 6 Pareto 前沿面 *Pareto* 最优解集 PS 中的所有 *Pareto* 最优解对应的目标矢量组成的曲面成为 *Pareto* 前端 PF ：

$$PF^* \triangleq \{F(x^*) = (f_1(x^*), f_2(x^*), \dots, f_m(x^*))^T \mid x^* \in P^*\} \quad (2-5)$$

2.2 免疫多目标优化算法

人工免疫算法结合了问题的先验知识和生物免疫系统的自适应能力，因而在信息处理方面有较强的鲁棒性，在搜索过程中能更好地收敛到全局最优解。使用免疫算法解决多目标优化问题也逐渐为研究者所重视，它保留了生物免疫系统所具有的全局搜索能力强、并行分布式搜索机制、鲁棒性强等特点，并且在实践中已经取得了较好的效果。同时克隆选择是生物免疫系统理论的重要学说，L.N.De Castro 等从不同的角度模拟生物学抗体克隆选择原理，相继提出了不同的克隆选择算法，其中，非支配邻域算法 NNIA 是比较有代表性的免疫多目标优化算法。

2.2.1 免疫克隆选择算法

克隆选择原理的基本思想是只有那些识别抗原的细胞才能进行扩增，只有这些细胞才能被免疫系统选择并保留下来，而那些不能识别抗原的细胞则不能被选择，更不会进行扩增。由 Burnet 在 1958 年提出的免疫克隆选择原理^[46]是一个受到普遍认可的模型，它描述了在免疫系统中如何寻找与抗原能最佳匹配的抗体，从而响应刺激。基于这个原理，Castro 开创性的提出了克隆选择优化算法 CSA^[47]。这个工作建立了基于免疫克隆选择和超突变的搜索机制的框架。图 2.1 是基于免疫优化算法的免疫克隆选择的搜索原理。它展示了抗体种群中的某个抗体在某一代如何从一个个体转变为一个更好的个体的过程。通过将一个个体的编码，我们可以通过模拟免疫学的认知过程即在抗体空间中找到最匹配的那个抗体去响应相应的抗原来解决优化问题。作为一种全新的人工免疫算法，免疫克隆选择算法和进化算法有许不同点，下面我主要通过四个方面给予解释说明：其一，进化计算的基本思想基于生物进化系统，而克隆算法则是受到免疫系统的启发而设计的。但这两种算法中均包含了变异操作，是细胞基因的变化是免疫和进化的生物基础，而细胞基因水平上的主要操作正是变异操作。其二，免疫克隆选择算法和进化算法的算法实现是又区别的。在进化算法中主要强调的是全局搜索，局部搜索只是作为辅助操作存在的，但克隆算子主要实现局部的并行搜索，克隆算法因而可以

保持较好的种群多样性。第三，在两种算法所使用的操作算子是不同的。交叉算子在标准的遗传算法中是主要算子，而变异算子主要作为辅助算子，而在克隆选择算法中，为了增强算法的局部搜索能力主要仅采用变异操，并没有采用交叉操作，这样不仅可以降低算法寻优的复杂度，同时改善了算法的局部收敛性。第四，免疫克隆选择算法中的克隆算子这种选择机制本身就具有记忆功能，因此本便能够保证算法以概率 1 收敛到最优解。

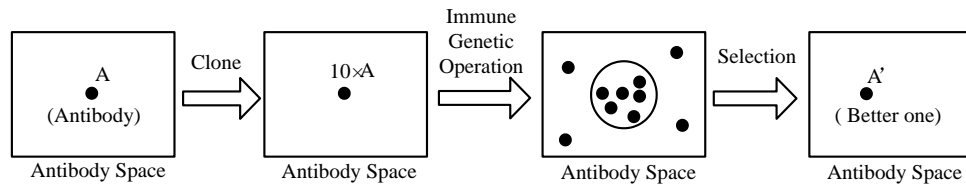


图 2.1 免疫克隆选择优化的搜索机制示意图

2.2.1.1 免疫克隆选择算法的机理及算法描述

在免疫克隆选择算法中，免疫细胞在增殖中的会发生基因突变，这样便保持了免疫细胞的多样性，无性繁殖系通过这些细胞的不断增殖形成。所谓克隆便是细胞的无性繁殖。而当有机体内免疫细胞的多样性达到一度程度时，在机体内每一种侵入机体的抗原都能够被识别出来，此外机体能将免疫细胞克隆，将其激活、分化和增殖，同时通过进行免疫应答的方式清除抗原，这整个过程称为克隆选择。

在 AIS 中，我们把由抗体亲合度定义的抗体随机映射称为克隆操作，其过程主要根据抗原和抗体的亲合度函数 $f(*)$ ，将解空间中的由抗体表示的一个点 $a_i(k) \in A(k)$ 分裂成为 q_i 个相同抗体所代表的点 $a'_i(k) \in A'(k)$ ，之后新的抗体种群可以通过执行了免疫基因操作以及克隆选择操作后获得。其中抗体、抗原、抗体和抗原之间的亲和度分别对应于优化问题的各种约束条件、优化解，目标函数以及目标函数和解的匹配程度。克隆的实质是在抗体进化过程中，在每一代的候选解的周围，根据抗体亲合度的大小进行克隆，产生一个由变异解组成的新群体，从而扩大了算法的搜索范围，有助于增加抗体的多样性从而防止进化早熟以及搜索陷于局部最优的情况出现，此外克隆选择操作还可以加快收敛速度。

2.2.1.2 算法描述

一般来说，免疫克隆选择算法主要包括三个步骤，即克隆操作、免疫基因操作和克隆选择操作。其中定义克隆 Θ 为：

$$\Theta(A) = [\Theta(a_1) \quad \Theta(a_2) \quad \cdots \quad \Theta(a_n)]^T \quad (2-6)$$

其中： $\Theta(a_i) = I_i \times a_i, i=1,2,\dots,n$ ， I_i 为 q_i 维行向量。

$$q_i = g(N, f(a_i)) \quad (2-7)$$

$$q_i = N * \frac{f(a_i)}{\sum_{j=1}^n f(a_j)}, \quad i=1, 2, \dots, n \quad (2-8)$$

$N > n$ 是与克隆规模有关的设定值。克隆过后，种群 A 变为：

$$A' = \{A, A_1', A_2', \dots, A_n'\} \quad (2-9)$$

其中：

$$A_i' = \{a_{i1}, a_{i2}, \dots, a_{iq_i-1}\}, \quad a_{ij} = a_i, \quad j=1, 2, \dots, q_i-1 \quad (2-10)$$

抗体的交叉、变异操作是免疫基因操作(T_g^C)主要组成部分。在生物学中由于单、多克隆抗体所对应的信息交换方式特点是有差异的，单克隆选择算法是指仅采用变异的克隆选择算法；多克隆选择算法是指交叉和变异都采用的克隆选择算法。这里所介绍的单克隆选择和多克隆选择并未完全遵循免疫学中的定义。这里所定义的单克隆算子与多克隆算子相比，仅是能够保留更多的父代特征。在传统的免疫学中，主要是依靠抗体的高频变异来保证抗体多样性的保持和亲和度的成熟而并不是克隆选择算法中的交叉变异算子；因此，克隆选择算法中会更加强调变异算子所产生的作用。和免疫学中所指变异不同的是，在免疫基因操作中，免疫基因变异(T_m^C)主要作用是为了更好的保留抗体原始种群的信息，当 $A \in A'$ ，此操作并不产生作用即：

$$p(T_m^C(a_i) = a_i') = \begin{cases} p_{ij} > 0 & a_i \in A_j' \\ 0 & a_i \in A \end{cases} \quad (2-11)$$

免疫基因操作中的交叉操作(T_c^C)和变异操作一样，为了保留抗体原始种群的信息，当 $A \in A'$ ，交叉操作并不产生作用即：

$$T_c^C(a_i, a_j) = a_i', \quad a_i \in A_j', \quad j=1, 2, \dots, n, \quad a_i \in A \quad (2-12)$$

而克隆选择操作(T_s^C)的定义如下：

$\forall i=1, 2, \dots, n$ ，若存在变异后的抗体 $b = \max\{f(a_{ij}) | j=2, 3, \dots, q_i-1\}$ ，使得：

$$f(a_i) < f(b), a_i \in A \quad (2-13)$$

则用 b 取代原抗体 a_i ，从而更新抗体群，实现信息交换。

因此在免疫克隆选择算法实施过程中，抗体群的状态转移情况可以表示为如下所示的随机过程：

$$A(k) \xrightarrow{\text{克隆操作}} A'(k) \xrightarrow{\text{免疫基因操作}} A''(k) \xrightarrow{\text{克隆选择操作}} A(k+1)$$

2.2.2 非支配邻域免疫算法 NNIA

NNIA(Multi-objective immune algorithm with non-dominated neighbor-based selection)是一种非常典型的免疫多目标选择算法, NNIA 中主要通过模拟免疫响应中具有多样性的抗体共同存活, 而仅有少数抗体被激活这种现象, 通过一种新的个体选择方法即基于非支配邻域的选择方法, 仅有少数的孤立非支配个体被选作为活性抗体, 其后通过计算活性抗体的拥挤距离, 按照其拥挤距离来进行比例克隆复制操作, 对进了克隆操作后的抗体种群采用了和传统于遗传算法不同的重组、变异操作, 从而使 pareto 前沿面中比较稀疏区域的搜索得到了进一步加强。

NNIA 算法主要流程如下:

假设在 t 时刻, 我们采用分别用时变矩阵 C_t 、 A_t 、 D_t 分别表示的优势抗体群、活性抗体群、克隆抗体群, 则非支配邻域免疫算法 NNIA 流程如下:

Step1 初始化: 以随机产生的方式初始抗体种群 B_0 , 设定非支配抗体种群

$D_0 = \Phi$, 活性抗体种群 $A_0 = \Phi$, 克隆抗体种群 $C_0 = \Phi$, 计数器 $t = 0$ 。

Step2 更新非支配抗体种群: 在 B_t 中选出非支配抗体, 组成非支配抗体种群 D_{t+1} ;

Step3 终止判断: 如果满足终止条件, 输出 D_{t+1} 作为算法求解结果, 算法结束; 否则, $t = t + 1$ 。

Step4 对非支配抗体种群进行进化操作产生子代抗体种群:

Step 4.1 选择部分拥挤距离大的非支配抗体组成活性抗体种群 A_t ;

Step4.2 按比例克隆。 对 A_t 执行自适应比例克隆, 得到克隆规模为 CS 的克隆抗体种群 C_t ;

Step 4.3 产生新抗体。 对 C_t 执行免疫交叉算子和超变异算子最终得到 S_t 。

Step5 更新抗体种群: 合并得到抗体种群即 $B_t = D_t \cup S_t$, 转 Step2。

NNIA 中优势抗体的拥挤距离是由当前 Pareto 前沿面上与其邻近的非支配个体决定的分布密度的估计值, 因此算法中的选择操作被叫做非支配邻域选择, 仅有部分具有较大拥挤距离值的非支配个体 (远远少于当前找到的非支配个体) 被选中作为种群中的活性抗体。克隆, 重组, 变异仅适用于被选择的活性抗体。通过算法流程可以看出 NNIA 的多样性保持较好, 但是初期由于对优势个体选择压力过大, 若只采用免疫克隆的进化机制容易陷入局部最优, 从而影响求解效率, 决策性能随之也会下降。

2.3 基于 EDA 的进化多目标优化算法

2.3.1 EDA 用于求解多目标优化问题

分布估计算法^[39]是一种群体进化随机搜索优化算法,与遗传算法不同,在分布估计算法中每一代种群使用概率的方法来表示。在一个优化问题中用一个随机变量来表示每个自变量 x_i , 因此可以得到随机向量 $x = (x_1, x_2, \dots, x_n)$, 这个随机向量和遗传算法中的基因串相对应。这里一个种群就和该随机向量的一个分布相对应。群体性能的一个重要指标就是随机向量的分布, 该群体可以利用随机向量的分来紧凑和整体地表示出来。随机变量之间的概率依赖关系蕴含在概率分布中, 这种概率依赖关系也同时也是一种基因之间的关系, 我们可以通过学习随机变量的分布来学习基因之间的关系。更有价值的群体和个体可以通过在一个概率分布上的采样过程中获得。所以, 分布估计算法的本质就是通过学习每一代的个体相对应的随机向量的分布, 然后通过学习到的分布再生成下一代新的个体。EDA 算法是建立在对分布进行估计之上, 它用建立概率模型以及从概率模型中采样这两步替代遗传算法中的重组和变异算子。概率模型可以捕获问题中的结构关系, 而且, 对个体之前关系的描述与个体位置无关, 不需要设计特殊的编码或操作算子。此外, EDA 还具有很高的可扩展性, 可利用不同的概率分布模型求解不同的问题。它是进化计算领域新兴的分支, 许多基于分布估计思想的多目标优化算法相继被提出来, 其中 RM-MEDA 利用了连续多目标问题的 PS 呈特殊分布的特点, 在求解某一类问题上取得了较好的性能。

2.3.2 基于 EDA 的进化多目标优化算法 RM-MEDA

Zhang 等提出的 RM-MEDA 算法充分利用了连续多目标优化问题的特点, 即对于目标个数为 m 的连续多目标问题, 由 Karush-Kuhn-Tucker 条件可知, Pareto 最优解集合 PS(Pareto Set)在决策空间上的分布呈现分段连续的 $(m-1)$ 维流形^[19]。RM-MEDA 算法的基本思想是建立多个线性 $(m-1)$ -维流形分布的概率模型来分段逼近整个非线性的 PS 流形, 再对各个线性模型进行随机采样产生后代种群。

RM-MEDA 算法迭代过程中种群的个体解将逐渐地向 Pareto 最优解集逼近, 并均匀地分布在主曲线(两个目标)或主曲面(3 个目标)周围。如图 2.2 所示, 种群的个体解均匀地分布在由决策空间潜在维数形成的主曲线两侧。下面对 RM-MEDA 算法的主要流程做以介绍。

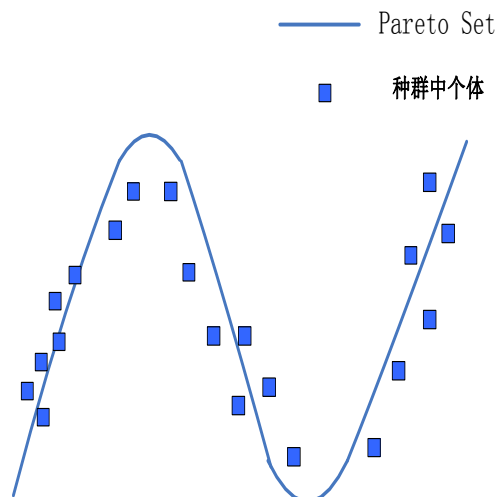


图 2.2 连续两目标问题的 Pareto 解集

算法的主要流程如下：

算法名称：RM-MEDA

Step1 初始化： t 为算法运行的代数， $pop(t) = \{x_1, x_2, \dots, x_n\}$ 为 t 对应的种群中 N 个个体解，所对应的目标函数值为 $F_{x_1}, F_{x_2}, \dots, F_{x_n}$ ，令 $t=0$ ，通过随机的方式来初始化种群 $pop(0)$ ，同时计算种群 $pop(0)$ 中每个个体所对应的目标函数值；

Step2 判断停止条件： 如果满足停止条件，停止并返回 $pop(t)$ 中非劣解集和对应的目标函数向量；

Step3 建立模型：

为 $pop(t)$ 中种群解的分布建立相应的概率模型；

Step4 繁殖子代

通过建立好的线性分段概率模型中产生新一代种群从而形成一个新的解集 Q ，计算解集 Q 中所有解的目标函数值；

Step5 选择

从 $Q \cup pop(t)$ 中选择 N 个个体作为新的 $pop(t+1)$ 。

在步骤 3 中该算法建立了一个分段线性模型,建立好的概率模型是否能够准确地描述种群中个体的分布，很大程度上取决于主曲线和主曲面分析的准确性，主曲线和主曲面分析旨在从 N 维向量空间中的数据点集合中挖掘出其中的中心曲线或者中心曲面。对于 m 个优化目标的 n 维多目标优化问题，当前种群的 PS 可以看作是如下概率模型的独立观测：

$$\xi = \zeta + \varepsilon \quad (2-14)$$

其中, 概率模型 ξ 的中心 ζ 是 n 维决策空间上的 $(m-1)$ -维分段连续流形, ε 是 n 维零均值随机噪声向量。RM-MEDA 用 K 个分段线性的概率模型来近似逼近 ξ 的中心流形 ζ , 即:

$$\zeta = \Psi^1 + \Psi^2 + \dots + \Psi^K \quad (2-15)$$

每个分段线性的概率模型 Ψ^j ($j=1,2,\dots,K$) 是 n 维空间上的一个 $(m-1)$ -维分段连续流形。

如图 2.3 所示, 对于两目标优化问题, Ψ^j 是 n 维决策空间上的线段。在建模过程中, 首先采用 Local PCA 聚类算法^[43]将当前种群划分为 K 个聚类 S^1, S^2, \dots, S^K , 然后对每个聚类 S^j 建立线性模型 Ψ^j 来刻画 S^j 的中心。

记 \bar{x}^j 为 S^j 的均值向量, U_i^j 为 S^j 的第 i 个主分量 (即 S^j 中所有个体协方差矩阵的第 i 个最大特征值对应的特征向量), 将 S^j 中的所有个体向 S^j 的 $(m-1)$ -维主子空间投影, 涵盖了所有投影点的最小线段或超平面记为 Φ^j , 那么:

其中:

$$\Phi^j = \left\{ x \in R^n \mid x = \bar{x}^j + \sum_{i=1}^{m-1} \alpha_i U_i^j, \alpha_i^j \leq \alpha_i \leq b_i^j, i=1, \dots, m-1 \right\} \quad (2-16)$$

$$a_i^j = \min_{x \in S^j} (x - \bar{x}^j)^T U_i^j \quad (2-17)$$

$$b_i^j = \max_{x \in S^j} (x - \bar{x}^j)^T U_i^j \quad (2-18)$$

进一步地, 将模型 Φ^j 沿着 $m-1$ 个主分量的 $U_1^j, U_2^j, \dots, U_{m-1}^j$ 方向两端各扩展 25%, 得到 Ψ^j :

$$\Psi^j = \left\{ x \in R^n \mid x = \bar{x}^j + \sum_{i=1}^{m-1} \alpha_i U_i^j, \right. \\ \left. a_i^j - 0.25(b_i^j - a_i^j) \leq \alpha_i \leq b_i^j + 0.25(b_i^j - a_i^j), i=1, \dots, m-1 \right\} \quad (2-19)$$

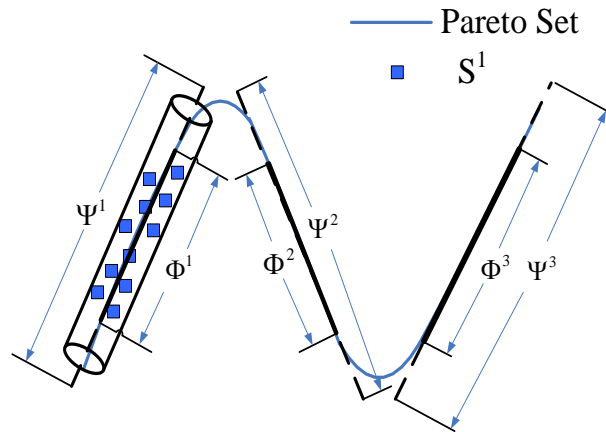


图 2.3 分段线性概率模型

图 2.3 以两目标优化问题为例描述了根据当前种群建立分段线性概率模型的

过程。在图 2.3 的例子中，当前种群被聚为 3 类， Φ^1, Φ^2 和 Φ^3 分别是根据 3 类子群体建立的分段逼近 1 维 PS 流形的最短线段。 Ψ^1, Ψ^2 和 Ψ^3 分别是 Φ^1, Φ^2 和 Φ^3 沿着主分量方向两端各扩展 25% 后的模型，可以看出，与 Φ^1, Φ^2, Φ^3 相比， Ψ^1, Ψ^2 和 Ψ^3 可以更好的逼近图中的 1 维 PS 流形。

在 RM-MEDA 算法 Step4 中所描述算法新的子代生成的过程中，需要产生 N 个新解，在算法 RM-MEDA 中新的子代由概率抽样的方式来产生，由于新繁殖的新解必须能够均匀的分布在 Pareto 解集周围，模型 Ψ^i 中到底需要生成多少个解可以用如下概率公式来计算：

$$\text{Prob}(A^i) = \frac{\text{vol}(\Psi^i)}{\sum_{j=1}^K \text{vol}(\Psi^j)} \quad (2-20)$$

其中当前通过模型 Ψ^i 产生新的个体的事件用 A^i 表示，其中 $\text{vol}(\Psi^i)$ 表示种群 S_i 中所有个体在前 $m-1$ 个主分量方向上投影的线段长度（两目标问题）、面积（三目标问题）、体积（四目标问题）或超体积（众目标问题），因此新解产生于 Ψ^i 的概率 $\text{Prob}(A^i)$ 等于 $\text{vol}(\Psi^i)$ 在整个投影中所占的比例。在 RM-MEDA 中，新的子代的繁殖过程如下：

Step1 通过随机的方式来产生一个整数 $q \in \{1, 2, \dots, K\}$ ，按照如下概率

$$\text{Prob}(q=i) = \frac{\text{vol}(\Psi^i)}{\sum_{j=1}^K \text{vol}(\Psi^j)} \quad (2-21)$$

Step2 产生一个新解，按照公式 $x = x' + \varepsilon'$ ，其中 x' 是一个数据点，其产生于 Ψ^q 中的， ε' 为一个噪音向量，其服从 $N(0, \sigma_q I)$ 的。

Step3 重复步骤一二直到产生 N 个新解。

(3) 选择

RM-MEDA 采用了 NSGAII 算法中的非劣排序选择机制^[12]，称作 NDS 选择。

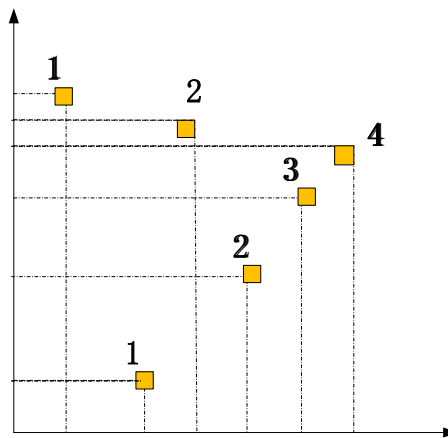


图 2.4 两目标 Pareto 排序

NDS 把 $Q \cup \text{Pop}(t)$ 划分成 s 不同的前沿 F_1, F_2, \dots, F_s ，如上图 2.4 所示，第 i 个

前沿 F_i 包含集合 $\{Q \cup Pop(t)\} \setminus \bigcup_{j=1}^{i-1} F_j$ 所有的非劣解, 因此集合 $\{Q \cup Pop(t)\} \setminus \bigcup_{j=1}^{i-1} F_j$ 中不存在解支配集合 F_i 中的解, F_1 是集合 $Q \cup Pop(t)$ 中最好的非劣前沿, F_2 第二非劣前沿, 依次类推, NDS 选择算法的过程如下:

Step1 利用非劣快速选择排序方法把 $Q \cup Pop(t)$ 划分成不同的前沿 F_1, F_2, \dots, F_s

令 $Pop(t+1) = \emptyset$ 和 $k = 0$

Do

$k = k + 1$

$Pop(t+1) = Pop(t+1) \cup F_k$

Until $|Pop(t+1)| > N$

Step2 *While* $|Pop(t+1)| > N$

对于集合 $F_k \cup Pop(t+1)$ 中的所有成员, 计算它们的拥挤距离, 把拥挤距离最小的元素从集合 $F_k \cup Pop(t+1)$ 中删除。

在算法 RM-MEDA 和 GDE3^[40]、PCX-NSGA-II^[41]、MIDEA^[42]相比, 该方法在求解变量之前有线性相关性的问题上取得了显著的效果, 然而, 在进化过程中, 基于 EDA 的建模方法只是宏观把握种群信息进行建模, 没有利用已找到的最优解的位置信息, 在进化后期会出现所求解集的精度不高的问题。此外, 为了建立概率模型, 算法 RM-MEDA 中且聚类数目 k 较难确定, 因此建立的分段线性模型并不能很好的去逼近测试问题的潜在维数的流行形状,

2.4 本章小结

本章综述了多目标优化算法, 重点描述了基于克隆选择模型和基于分布式估计的两类多目标优化算法, 给出了 NNIA 算法求解多目标问题的流程和它的优缺点, 以及 EDA 方法在求解多目标问题时的优势, 详细阐述了 RM-MEDA 算法, 该方法在求解变量之间有线性和非线性关联的一类问题上效果较好。

第三章 进化算子自适应选择的多目标优化算法

本章在基于 EDA 和人工免疫系统的混合多目标优化算法(HIAEDA)基础上,提出了一种进化算子自适应选择的多目标优化算法(MAOS)。本章中首先根据 HIAEDA 的不足介绍了本算法的改进动机和创新点,接着描述了算法的框架和主要流程,最后根据仿真程序给出了实验结果。

3.1 引言

算法 HIAEDA 中,在迭代进化过程中采用两种后代产生策略:一种是基于交叉变异的克隆选择算子,另外一种是基于 EDA 的模型采样算子。交叉变异算子能够很好的保持种群的多样性,且收敛精度较高,但是由于没有考虑种群个体间的相关性,对于决策变量之间存在非线性相关的复杂多目标优化问题求解能力欠佳。基于 EDA 的模型采样算子,对于求解决策变量之间存在非线性相关的复杂多目标优化问题十分有效。然而其也存在着不足之处:在进化的初期由于种群没有充分收敛,算法建立的概率模型准确性不够,在进化的后期,通过对概率模型随机采样产生的后代逼近精度不高。

HIAEDA 结合两类算子的优势,利用统计学习方法分析种群中抗体的宏观分布特性,利用交叉变异策略提高算法收敛速度和精度,防止算法陷入局部最优而早熟收敛。仿真实验结果表明, HIAEDA 与免疫多目标优化算法和 EDA 算法相比均表现出明显优势,尤其对决策变量之间存在非线性相关的复杂多目标优化问题优势尤为突出。但在 HIEDA 中,算法迭代过程中种群中个体采用两种算子的比例是固定的,并未根据两种算子在种群进化的不同阶段自适应的调整两种算子的比例,来根据两种搜索算子的不同搜索特性来在进化的不同阶段发挥各自的作用。针对 HIEDA 的不足,本文提出了一种求解复杂多目标优化问题的自适应算子选择密母算法 MAOS,其迭代进化过程中采用两种后代产生策略:一种是基于交叉变异的克隆选择算子(Clone Selection Operator, CSO),另外一种是基于 EDA 的模型采样算子(EDA Operator, EO)。图 3.1 采用数值实验的方法分析了两种进化算子的搜索特性。考虑到直观性,图 3.1 以二维决策空间为例,从一组固定的父代抗体(9个)出发,分别采用两种算子产生了相同数量(900个)子代抗体,通过观察子代抗体在决策空间中的分布情况考察两种进化算子的搜索特性。

从图 3.1 中两种进化算子产生后代抗体的分布可以看出,模型采样算子会沿着父代种群构成的线性 PS 流形方向进行搜索,搜索范围不仅仅局限于当前种群个体围成区域的内部,而且沿着当前线性 PS 流形方向朝着两端进行扩展。基于 EDA

的模型采样算子对于决策变量之间线性相关的多目标优化问题十分有效,对于决策变量之间非线性相关的情况,该算子通过建立分段线性模型逼近非线性的 PS 流形。基于 SBX 交叉和 PM 变异的算子在父代种群的周围进行随机搜索,交叉变异算子产生的后代具有良好的多样性,能够在父代种群所在的区域进行局部搜索的同时开辟更为广阔的搜索空间。在实验中我们还发现,两种搜索算子由于搜索特性各有不同,能够在进化的不同阶段发挥各自的作用,且作用的大小与求解的多目标优化问题有关。此外本文算法的自适应 Memetic 算子中利用进化过程中的精英个体信息定义能力(Capacity Rate,CR)和效率(Efficiency Rate,ER)两个指标来自适应的调整两种算子在进化的不同阶段采用的比例,从而平衡算法的开发和探索的能力。

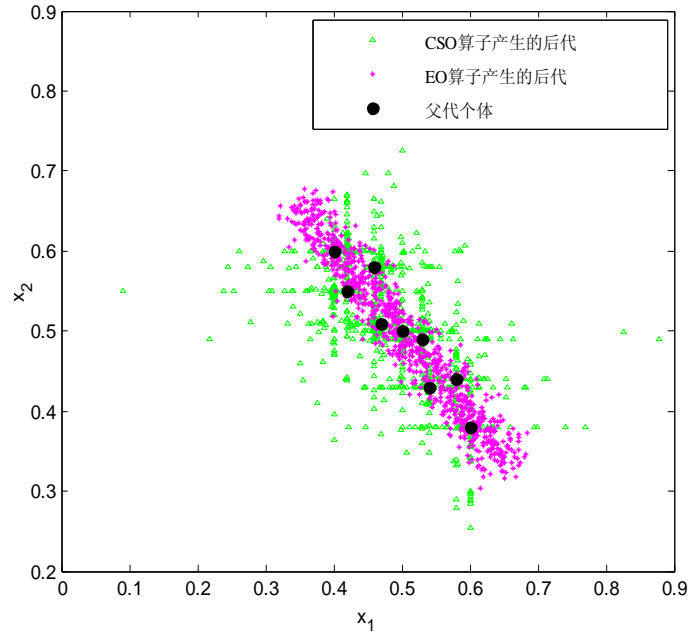


图 3.1 两种进化算子的搜索特性分析

3.2 自适应选择密母算子

本文中设计的自适应选择密母算子主要通过种群迭代进化过程采用自适应选择(Adaptive Operator Selection, ASO)策略^[49],自适应地调整 CSO 和 EO 两种算子在当前种群中所应采用的比例,从而生成新一代的子种群。而某种算子所应采用的比例主要取决于其在进化过程的综合性能。传统的 ASO 策略一般涉及两个方面的内容:信用赋值(Credit Assignment, CA)^[50,51,52,53]和算子选择(Operator Selection, OS)^[54,55]。信用赋值即根据某个算子其在以前的搜索过程制中的性能按照特定的规则给予其相应的权值或激励。而算子选择则

是根据每个算子所获得的权值或激励去确定采用哪一种算子。在现有的文章中^[50,51,52,53]，信用赋值的一般做法都是基于采用该算子后产生新个体的适应度相对于设定好的基准值的改善程度，基准值的一般选取为当代种群中的最优个体或者当前个体的父代。对于多目标优化问题，并不能像单目标优化问题那样直接计算个体之间适应度的差值。本文中则定义了能力指标(Capacity Rate,CR)和效率指标(Efficiency Rate,ER)两种指标来从种群的角度来定义算子的信用赋值。当前种群 $P(t)$ 中，通过两种算子 CSO 和 EO 分别产生子代，算子 CSO 产生的子代数目为 N_{CSO} ，算子 EO 产生的子代数目为 N_{EO} ，通过两种算子共同生成的子代种群为 S' ，其种群规模为 N ，对抗体种群 S' 进行 Pareto 选择，Pareto 最优解构成的集合为 PN ，集合规模为 M 。在集合 PN 中，由算子 CSO 产生的子代个体的数目为 PN_{CSO} ，由算子 EO 产生的子代个体的数目为 PN_{EO} ，则两种算子的能力指标可以由下公式定义：

$$CR_{CSO} = PN_{CSO} / (PN_{CSO} + PN_{EO}) \quad (3-1)$$

$$CR_{EO} = PN_{EO} / (PN_{CSO} + PN_{EO}) \quad (3-2)$$

能力指标定义了在新产生的子代种群所有个体中， CSO 和 EO 这两种算子分别产生非支配解所占整个种群的比例，能力指标主要衡量了算法的探索能力。当前种群 $P(t)$ 和生成子代种群 S' 合并，即 $Q = P(t) \cup S'$ ，采用基于 m 最近邻列表 (m -Nearest Neighbor List) 的选择方法^[11]从 Q 中选出 N 个抗体构成新的种群 $P(t+1)$ 。对抗体种群 $P(t+1)$ 进行 Pareto 选择，Pareto 最优解构成的集合为 PN' ，集合规模为 M' 。在集合 PN' 中，由算子 CSO 产生的子代个体的数目为 PN_{CSO}' ，由算子 EO 产生的子代个体的数目为 PN_{EO}' ，则两种算子的效率指标可以由下公式定义：

$$ER_{CSO} = PN_{CSO}' / PN_{CSO} \quad (3-3)$$

$$ER_{EO} = PN_{EO}' / PN_{EO} \quad (3-4)$$

效率指标定义了 CSO 和 EO 两种算子产生的子代种群和父代种群混合后根据评价指标选出新的进化种群后，两种算子分别所产生的非支配个体所占进化种群中个体的比例，效率指标主要衡量了算法的开发能力。本文中的信用赋值则是通过对能力指标和效率指标设定权重对能力和效率指标加权求和来定义的。

$$CA_{CSO} = wCR_{CSO} + (1-w)ER_{CSO}, 0 \leq w \leq 1 \quad (3-5)$$

$$CA_{EO} = wCR_{EO} + (1-w)ER_{EO}, 0 \leq w \leq 1 \quad (3-6)$$

本文中的算子选择部分则是通过计算出的两种算子信用赋值，在进化种群中按照两种算子信用赋值的比例来分别采用两种算子产生子代个体。

3.3 算法 MAOS 的框架设计

MAOS 算法的抗体编码采用了 0 到 1 之间的实数编码方式, 抗体 $A=(a_1, a_2, \dots, a_n)$ 的每一个基因位 $a_i (i=1, 2, \dots, n)$ 表示决策向量的一维。在已知每一维决策变量的取值范围 $[min_i, max_i]$ 的前提下, 决策变量的取值可以解码为 $v_i = min_i + a_i (max_i - min_i)$ 。

在迭代过程中采用了两种不同类型的子代繁殖策略产生后代抗体, 一种是免疫多目标优化算法中基于交叉变异的子代产生策略, 另外一种 EDA 算法中基于模型采样的子代产生策略。算法的主要流程描述如下:

算法 3.1: MAOS 算法流程

Step1 初始化: 设置种群规模 N , 子种群个数 K , 变异概率 p_m , 设置算法停止条件。

令迭代次数 $t=1$, 随机初始化种群 $P(t)$ 并计算初始种群中抗体的目标函数值;

Step2 判断停止条件: 若算法停止条件满足, 则输出种群 $P(t)$, 否则转 Step3;

Step3 繁殖:

Step 3.1 分组建模: 采用 Local PCA 聚类算法^[21]将当前种群 $P(t)$ 中的抗体聚为 K 类, 分别构成 K 个子种群 S_1, S_2, \dots, S_K , 采用文章第二节中所述的建模方法对各个子种群建立概率模型 $\Psi^1, \Psi^2, \dots, \Psi^K$;

Step 3.2 分配繁殖数量: 假设第 i 个子种群产生后的个数为 N_i , 其中 $i=1, 2, \dots, K$ 并且 $N = \sum_{i=1}^K N_i$, 则 N_i 可以从以下公式获得:

$$N_i = \left\lfloor N * vol(\Psi^i) / \sum_{j=1}^K vol(\Psi^j) \right\rfloor \quad (3-7)$$

其中 $vol(\Psi^i)$ 表示种群 S_i 中所有抗体在前 $m-1$ 个主分量方向上投影的线段长度 (两目标问题)、面积 (三目标问题)、体积 (四目标问题) 或超体积 (众目标问题), $\lfloor x \rfloor$ 表示对实数 x 取下整。

Step 3.3 采用自适应密母算子产生后代: 采用密母算子中的克隆选择算子产生后代种群 S'_i , 同时采用模型采样算子产生后代种群 S''_i

Step4 变异: 记 $S_1 = \bigcup_{i=1}^K S'_i$, $S_2 = \bigcup_{i=1}^K S''_i$, 令 $S = S_1 \cup S_2$ 。对 S 中的每一个抗体采用基于 ε 区间分割的变异算子进行变异操作, 变异概率为 p_m , 得到后代种群 S' ;

Step 4.1: 将多目标优化问题决策空间的每一维子空间 $[min_r, max_r]$ ($r=1, 2, \dots, n$) 等分为 n_r-1 个子区间 $[\omega'_1, \omega'_2], [\omega'_2, \omega'_3], \dots, [\omega'_{n_r-1}, \omega'_{n_r}]$, 其中 min_r 和 max_r 分别为第 r 维决策变量的取值下界和取值上界, $\omega'_1 = min_r$, $\omega'_{n_r} = max_r$, 且 $|\omega'_j - \omega'_{j-1}| \leq \varepsilon$, $j=2, \dots, n_r$;

Step 4.2: 对于被选中变异的抗体 $A=(a_1, a_2, \dots, a_n)$, 对其每一个基因位以概率 p_m 进行变异。若抗体 A 的第 r ($r=1, 2, \dots, n$) 个基因位被选中变异, 则在第 r 维

决策子空间 $[min_r, max_r]$ 的 $n_r - 1$ 个子区间中随机选中一个, 记为 $[\omega_s^r, \omega_{s+1}^r]$, $s = 1, 2, \dots, n_r - 1$, 然后令 $\omega = \omega_s^r + rand \times (\omega_{s+1}^r - \omega_s^r)$, 其中 $rand$ 为 0 到 1 之间的实数, 再将抗体 A 的第 r 个基因位 a_r 变异为: $a_r = \omega / (max_r - min_r)$;

Step 4.3: 将所有变异后得到的新抗体构成后代种群 S' ;

Step5 选择: 令 $Q = P(t) \cup S'$, 采用基于 m 最近邻列表 (m-Nearest Neighbor List) 的选择方法^[46]从 Q 中选出 N 个抗体构成新的种群 $P(t+1)$, 令 $t = t+1$, 转 Step2.

算法 3.2: 自适应密母算子 (输入 K 个抗体子种群 S_i , $i = 1, 2, \dots, K$, 输出种群中 CSO 和 EO 两种算子产生的子种群 S_i' , $i = 1, 2, \dots, K$)

Step1 计算算子信用赋值:

Step1.1 计算算子能力指标 CR 和效率指标 ER:

Step1.2 根据式 4-5, 式 4-6 计算应用赋值 CA_{CSO} , CA_{EO} .

Step2 算子选择: 根据两种算子的信用赋值 CA_{CSO} , CA_{EO} . 各子种群中采用克隆选择算子以及模型采样算子产生分别产生后代种群 S_i' 和 S_i'' , 则 S_i' 中抗体个数为 $N_i^1 = \lfloor N_i * CA_{CSO} / (CA_{CSO} + CA_{EO}) \rfloor$, S_i'' 中抗体个数为 $N_i^2 = N_i - N_i^1$.

在 MAOS 算法的步骤 3.3 中, 采样了基于免疫克隆选择的子代产生策略。克隆选择算子中的免疫基因操作包括重组和变异两个部分, 具体流程如下:

算法 3.3: 克隆选择算子 (输入 K 个抗体子种群 S_i , $i = 1, 2, \dots, K$, 输出 K 个后代种群 S_i')

对于每个抗体子种群 S_i ($i = 1, 2, \dots, K$), 分别执行如下克隆选择算子:

Step1 Pareto 选择: 对抗体子种群 S_i 进行 Pareto 选择, 记 S_i 中 Pareto 最优解构成的集合为 P_i , 集合规模为 M^i 。若 $M^i > N_i^1$ 转 Step2, 否则转 Step3;

Step2 子种群规模控制: 计算子种群 S_i 中每个抗体的拥挤距离^[13], 从中删除拥挤距离最小的抗体, 令 $M^i \leftarrow M^i - 1$, 若 $M^i > N_i^1$ 转 Step2; 否则转 Step3;

Step3 SBX 重组及 PM 变异: 依次对子种群 S_i 中的每个抗体 A_p ($p = 1, 2, \dots, M^i$) 执行模拟二进制交叉 (Simulated Binary Crossover, SBX)^[44] 以及多项式变异操作 (Polynomial Mutation, PM)^[44], 产生后代 A_p'' ;

Step4 返回结果: $S_i' \leftarrow \{A_1'', A_2'', \dots, A_{M^i}''\}$, 返回所有的 S_i' 。

在算法 2 中, 模拟二进制交叉操作和多项式变异操作是求解连续优化问题的进化算法中有效且常用的交叉和变异算子。这两个算子在经典的进化多目标优化算法 NSGAII 中被使用, 并取得了良好的效果。

MAOS 的算子自适应策略具体操作流程如算法 3 所示:

在 MAOS 算法的步骤 5 中, 采用了一种基于邻近距离(Vicinity distance)的 m 最近邻列表的选择方法保持非支配种群的均匀性, 其中 m 为目标函数个数。抗体的邻近距离定义为 $V_{mm} = \prod_{i=1}^m L_2^{NN_i}$, 其中 $L_2^{NN_i}$ 是抗体距离第 i 最近抗体之间的欧氏距离。MAOS 算法中步骤 5 的选择操作具体流程如下:

算法 3.4: 基于 m 最近邻列表的选择方法 (输入种群 Q , 输出规模为 N 的新种群 $P(t+1)$)

Step1 建立 m 最近邻列表: 对于种群 Q 中的每一个抗体, 计算其在种群 Q 中的邻近距离, 并对每一个抗体建立基于邻近距离的 m 最近邻列表;

Step2 判断停止条件: 记当前种群 Q 中抗体数量为 S_Q , 若 $S_Q \leq N$, 则令 $P(t+1) = Q$ 并返回 $P(t+1)$, 否则转 Step3;

Step3 删除最拥挤抗体: 删除前种群 Q 中邻近距离最小的抗体 A_w , 重新计算 m 最近邻列表中包含了被删除抗体 A_w 的抗体的邻近距离, 令 $S_Q = S_Q - 1$;

Step4 更新 m 最近邻列表: 对 m 最近邻列表中包含了被删除抗体 A_w 的抗体, 更新其 m 最近邻列表, 转 Step2。

文献^[46]从理论上证明了基于 m 最近邻列表的选择方法能够获得均匀分布的 Pareto 最优解集, 并通过大量实验结果验证了基于邻近距离的非支配排序方法与基于拥挤距离(Crowding distance)的非支配排序方法^[13]相比具有明显的优势。

这里我们讨论一下 MAOS 算法的计算复杂度, 我们仅考虑种群规模的大小, 假设种群大小为 N 。在最坏情况下, 每一次迭代中 MAOS 的计算复杂度描述如下:

在繁殖策略中, 主要复杂度由算法 Local PCA 决定, 其中假设种群大小为 N , 聚类个数为 K , 因为在聚类过程中, 需要多次更新聚类中心, 设最大更新次数为 $Iter$, 在最坏情况下, 复杂度为: $O(N * Iter * K)$, 其中 N 和 $Iter * K$ 的量级相当因此复杂度可以记为 $O(N^2)$ 。同时在密母算子中: 克隆选择算子中的二进制交叉算子 SBX、超变异算子 PM 的计算复杂度都为 $O(N)$, 其中基于拥挤距离的选择操作的计算复杂度为 $O(N \log(N))$, 我们讨论一下算法 Step5 中基于 m 最近邻列表的选择方法的复杂度为 $O(N^2)$ 。所以算法一次迭代的计算复杂度为:

$$O(N^2) + O(N) + O(N \log(N)) + O(N^2) \quad (3-8)$$

因此在最坏情况下, 算法的计算复杂度可以简单记作 $O(N^2)$ 。

3.4 仿真实验

实验设计中, 我们测试了已有的 3 类标准测试函数, 其中包括变量之间没有关联的 ZDT 系列函数, 变量之间存在线性关联的 ZDT 变形函数以及变量之间存在非线性关联的复杂函数, 并将实验结果和经典算法 RM-MEDA、NSGAII 进行了对

比。

3.4.1 标准测试函数

本文对当前已有的 18 个标准测试函数进行了实验，如下表所示：

表 3.1 本文的测试问题

函数	变量	维数	目标函数值（最小）
ZDT1	[0,1]	30	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - \sqrt{x_1/g(x)}]$ $g(x) = 1 + 9(\sum_{i=2}^n x_i)/(n-1)$
ZDT1L	[0,1]	30	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - (x_1/g(x))^2]$ $g(x) = 1 + 9(\sum_{i=2}^n (x_i - x_1)^2)/(n-1)$
ZDT1N	[0,1]	30	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - (x_1/g(x))^2]$ $g(x) = 1 + 9(\sum_{i=2}^n (x_i^2 - x_1^2))/(n-1)$
ZDT3	$x_1 \in [0,1]$ $x_i \in [-5,5]$ $i = 2, \dots, n$	10	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - \sqrt{x_1/g(x)} - \frac{x_1}{g(x)} \sin(10\pi x_1)]$ $g(x) = 1 + 9(\sum_{i=2}^n x_i)/(n-1)$
ZDT3L	[0,1]	10	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - \sqrt{x_1/g(x)} - \frac{x_1}{g(x)} \sin(10\pi x_1)]$ $g(x) = 1 + 9(\sum_{i=2}^n (x_i - x_1)^2)/(n-1)$
ZDT3N	[0,1]	10	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - \sqrt{x_1/g(x)} - \frac{x_1}{g(x)} \sin(4\pi x_1)]$ $g(x) = 1 + 9(\sum_{i=2}^n (x_i^2 - x_1^2))/(n-1)$

ZDT4	[0,1]	10	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - \sqrt{x_1/g(x)}]$ $g(x) = 1 + 10(n-1) + \sum_{i=2}^n [x_i^2 - 10\cos(4\pi x_i)]$
ZDT4L	[0,1]	10	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - \sqrt{x_1/g(x)}]$ $g(x) = 1 + 10(n-1) + \sum_{i=2}^n [(x_i - x_1)^2 - 10\cos(4\pi x_i)]$
ZDT4N	$[0,1] \times [0,10]^n$	10	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - \sqrt{x_1/g(x)}]$ $g(x) = 1 + 10(n-1) + \sum_{i=2}^n [(x_i^2 - x_1)^2 - 10\cos(2\pi(x_i^2 - x_1))]$
ZDT6	[0,1]	30	$f_1(x) = 1 - e^{(-4x_1) \times [\sin(6\pi x_1)]^6}$ $f_2(x) = g(x)[1 - (f_1(x)/g(x))^2]$ $g(x) = 1 + 9[(\sum_{i=2}^n x_i)/(n-1)]^{0.25}$
ZDT6L	[0,1]	30	$f_1(x) = 1 - e^{(-4x_1) \times [\sin(6\pi x_1)]^6}$ $f_2(x) = g(x)[1 - (f_1(x)/g(x))^2]$ $g(x) = 1 + 9[(\sum_{i=2}^n (x_i - x_1)^2)/(n-1)]^{0.25}$
ZDT6N	[0,1]	30	$f_1(x) = 1 - e^{(-4x_1) \times [\sin(6\pi x_1)]^6}$ $f_2(x) = g(x)[1 - (f_1(x)/g(x))^2]$ $g(x) = 1 + 9[\sum_{i=2}^n (x_i^2 - x_1)^2/(n-1)]^{0.25}$
DTLZ2	[0,1]	$k + x_k - 1$	$f_1(x) = (1 + g(x_k))\cos(x_1\pi/2)\cos(x_2\pi/2)\dots\cos(x_{k-2}\pi/2)\cos(x_{k-1}\pi/2)$ $f_2(x) = (1 + g(x_k))\cos(x_1\pi/2)\cos(x_2\pi/2)\dots\cos(x_{k-2}\pi/2)\sin(x_{k-1}\pi/2)$ \vdots $f_{k-1}(x) = (1 + g(x_k))\cos(x_1\pi/2)\sin(x_2\pi/2)$ $f_k(x) = (1 + g(x_k))\sin(x_1\pi/2)$ <p>where $g(x_k) = 100[x_k + \sum_{x_i \in X_k} ((x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5)))]$</p>

DTLZ2L	[0,1]	30	$f_1(x) = (1 + g(x_k)) \cos(x_1 \pi / 2) \cos(x_2 \pi / 2)$ $f_2(x) = (1 + g(x_k)) \cos(x_1 \pi / 2) \sin(x_{k-1} \pi / 2)$ $f_3(x) = (1 + g(x_k)) \sin(x_1 \pi / 2)$ $\text{where } g(x_k) = \sum_{i=3}^n (x_i - x_1)^2$
DTLZ2N	[0,1]	30	$f_1(x) = (1 + g(x_k)) \cos(x_1 \pi / 2) \cos(x_2 \pi / 2)$ $f_2(x) = (1 + g(x_k)) \cos(x_1 \pi / 2) \sin(x_{k-1} \pi / 2)$ $f_3(x) = (1 + g(x_k)) \sin(x_1 \pi / 2)$ $\text{where } g(x_k) = \sum_{i=3}^n (x_i^2 - x_1)^2$
DTLZ6	[0,1]	$k + x_k - 1$	$f_1(x) = x_1$ $f_2(x) = x_2$ \vdots $f_{k-1}(x) = x_{k-1}$ $f_k(x) = (1 + g(x_k)) h(f_1, f_2, \dots, f_{k-1}, g)$ $\text{where } g(x_k) = 1 + \frac{9}{ x_k } \sum_{x_i \in \mathbf{x}_k} x_i,$ $h(f_1, f_2, \dots, f_{k-1}, g) = k - \sum_{i=1}^k \left[\frac{f_i}{1+g} (1 + \sin(3\pi f_i)) \right]$
DTLZ6L	[0,1]	$k + x_k - 1$	$f_1(x) = x_1$ $f_2(x) = x_2$ \vdots $f_{k-1}(x) = x_{k-1}$ $f_k(x) = (1 + g(x_k)) h(f_1, f_2, \dots, f_{k-1}, g)$ $\text{where } g(x_k) = 1 + \frac{9}{ x_k } \sum_{x_i \in \mathbf{x}_k} (x_i - x_1)^2,$ $h(f_1, f_2, \dots, f_{k-1}, g) = k - \sum_{i=1}^k \left[\frac{f_i}{1+g} (1 + \sin(3\pi f_i)) \right]$

DTLZ6N	[0,1]	$k + x_k - 1$	$f_1(x) = x_1$ $f_2(x) = x_2$ \vdots $f_{k-1}(x) = x_{k-1}$ $f_k(x) = (1 + g(x_k))h(f_1, f_2, \dots, f_{k-1}, g)$ $\text{where } g(x_k) = 1 + \frac{9}{ x_k } \sum_{x_i \in x_k} (x_i^2 - x_1)^2,$ $h(f_1, f_2, \dots, f_{k-1}, g) = k - \sum_{i=1}^k \left[\frac{f_i}{1+g} (1 + \sin(3\pi f_i)) \right]$
--------	-------	-----------------	--

3.4.2 性能评价指标

在实验中，我们使用 inverted generational distance (IGD)^[45]来评估算法的性能。假定 P^* 为沿着 PF 上均匀分布的一组解， P 为 PF 的逼近解，则从 P^* 到 P 的 IGD 定义如下：

$$IGD(P^*, P) = \frac{\sum_{v \in P^*} d(v, P)}{|P^*|} \quad (3-9)$$

其中 $d(v, P)$ 为点 v 到 P 中所有点的最小欧式距离，IGD 指标是衡量所求解的收敛性和多样性的综合指标，IGD 值越小，表明解越接近理想的 Pareto 解。在实验中，对于两目标问题，我们在 PF 上均匀选择 500 个点作为 P^* 。我们利用盒图^[45]来表示每个算法对于 18 个测试问题的 IGD 统计结果，在经济学领域中盒图一直被作为统计分析的重要工具，据的统计分布情况可以通过盒图很好的表示出来，试验中，各个算法独立运行 20 次得到的解的统计特性用该工具来表现。其中，盒子中上下两条线段分表别是样本的上下四分位数，盒子中间的水平线为样本的中位数，盒子上下虚线表示样本的其余本分（野值除外），数据样本的最大值和最小值分别为虚线两端，“+”表示野值，盒子中的切口为数据样本的置信区间。

3.4.3 实验参数设置

在三种算法中，ZDT 及其线性和非线性变形函数群规模都设为 100，DTLZ 及其线性和非线性变形函数群规模都设为 200；所有测试函数均运行 1000 代，RM-MEDA 和 MAOS 中聚类个数都为 5；NSGAII、MAOS 算法中，变异概率 $pm=0.1$ ，SBX 交叉概率 $pc=1$ ；子种群个数为 5，每个函数独立运行次数为 20。

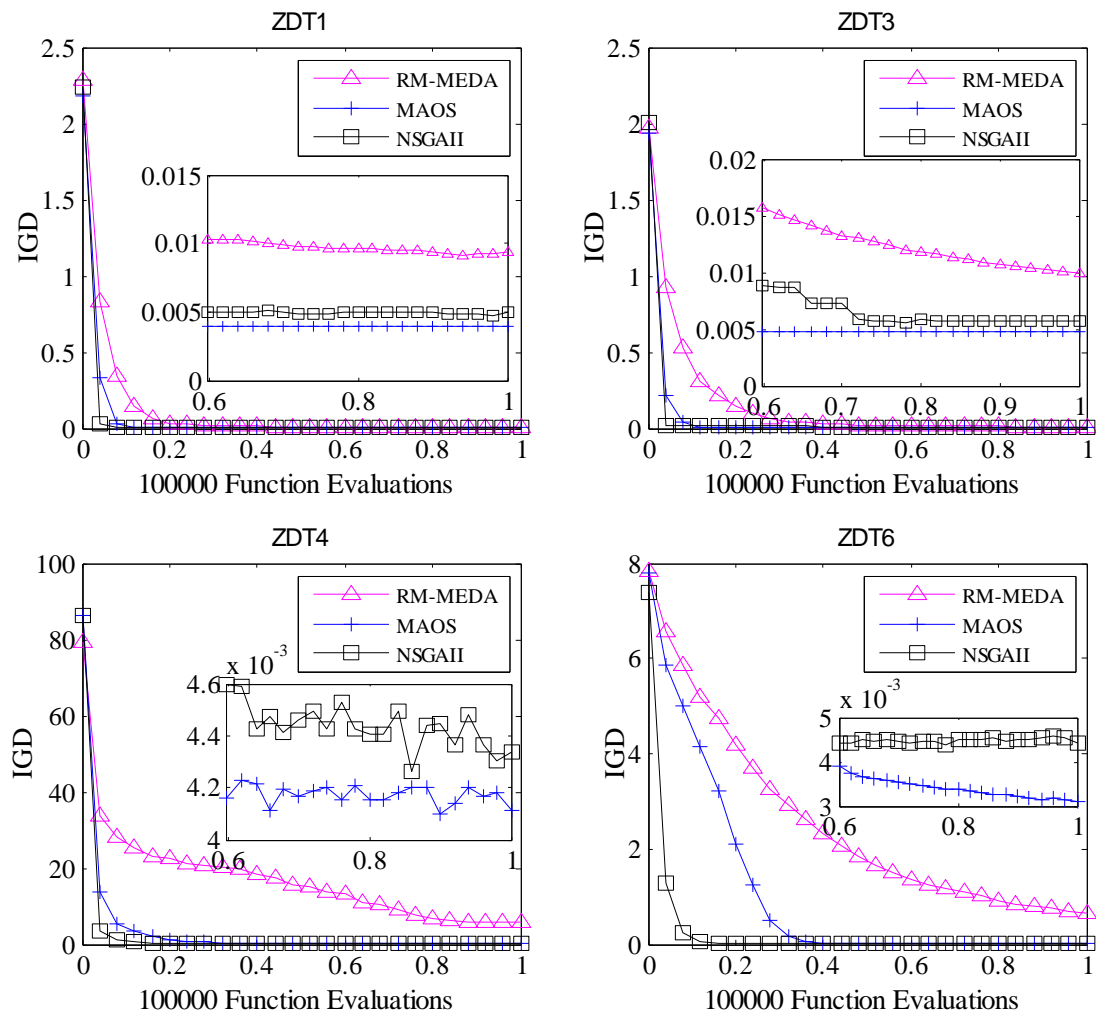
3.4.4 实验结果及分析

3.4.4.1 MAOS 算法有效性分析

在实验中，分别求得 RM-MEDA、NSGAII、MAOS 算法对于 18 个测试问题在进化过程中 IGD 指标的变化趋势，从而看出不同算法在不同进化阶段的收敛速度以及逼近真实 Pareto 前端的精度，此外，每个函数独立运行 20 次，利用盒图来表示每个算法对于 18 个测试问题的 IGD 统计结果，从而评价每个算法在每个测试问题上的收敛性和多样性。

(1) 变量之间没有关联的 ZDT 及 DTLZ 系列函数的实验结果

下图 3.2 是三种算法求解部分 ZDT 系列以及 DTLZ 系列测试问题时，IGD 指标随函数评价次数增加的变化趋势。图 3.3 是三种算法求解 ZDT 及 DTLZ 测试问题时，IGD 指标统计盒图。



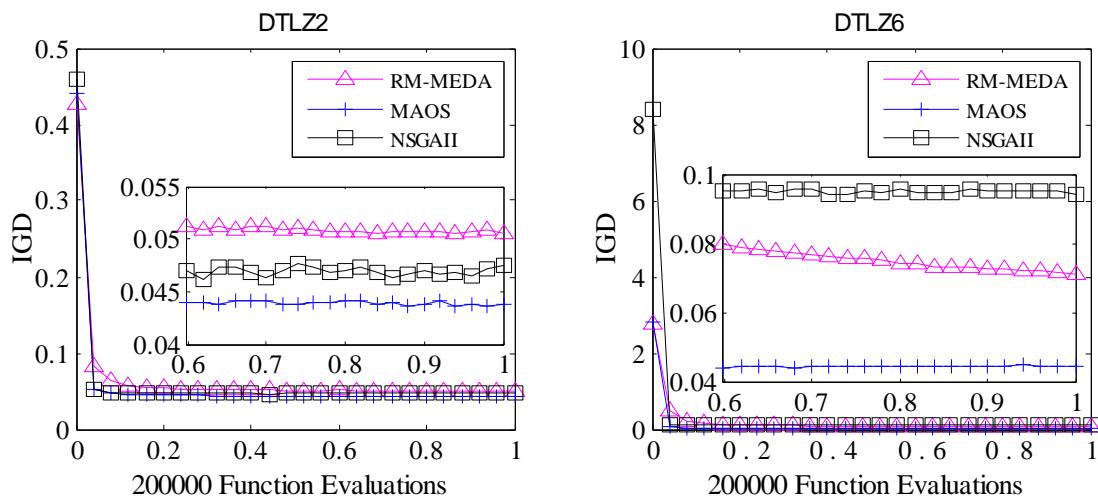
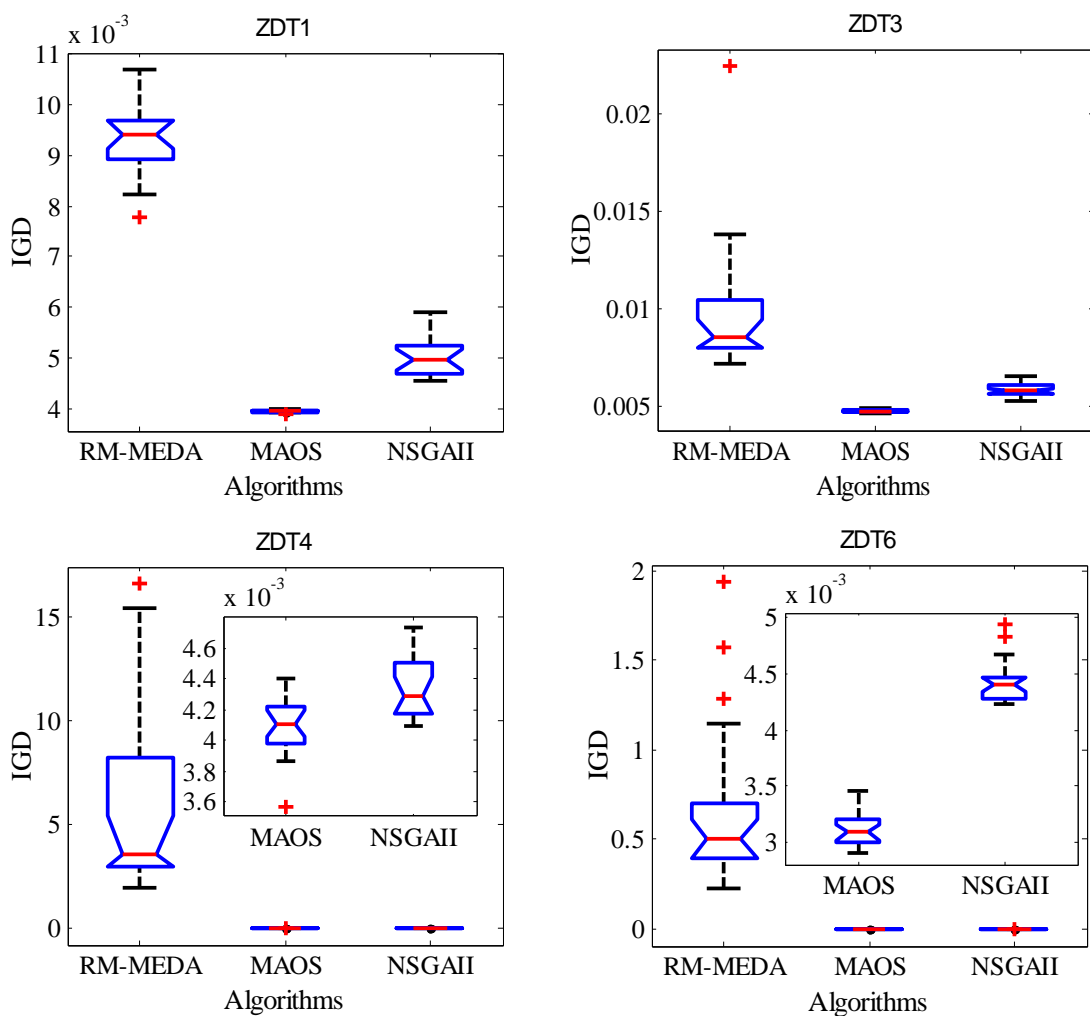


图 3.2 三种算法求解变量之间没有关联的部分 ZDT 和 DTLZ 系列测试问题时, IGD 指标随函数评价次数增加的变化趋势



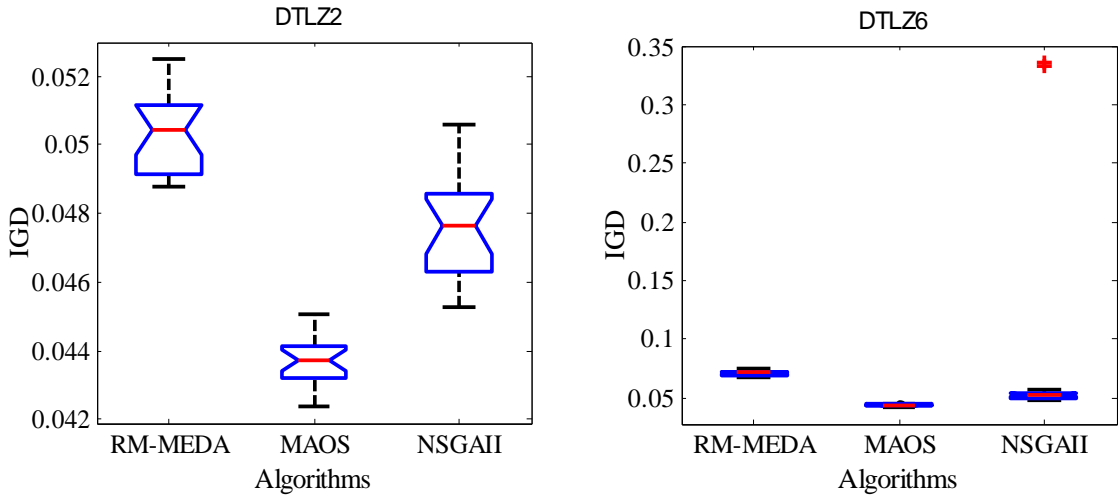


图 3.3 三种算法求解变量之间没有关联的部分 ZDT 和 DTLZ 系列测试问题时, IGD 指标统计盒图

从图 3.2 可以看出, 对于 ZDT 系列函数, 在进化前期, NSGAII 收敛最快, 后期 MAOS 所求解的精度最高。MAOS 算法对于有大量局部 Pareto 最优解的 ZDT4 函数和 Pareto 最优解在最优前沿上分布不均匀的 ZDT6 函数也可以搜索到满意的解集。对于三目标测试问题 DTLZ2 和 DTLZ6, 在进化后期 MAOS 的求解精度均为最好。因为 MAOS 中自适应密母算子中的基于交叉变异的克隆选择算子和模型采样算子在进化的不同阶段发挥各自算子的特点, 很好的保持了种群的多样性, 同时能跳出局部最优, 避免早熟收敛。从图 3.3 可以看出, 对于变量之间没有关联的部分 ZDT 和 DTLZ 系列测试问题, 不论是收敛性还是多样性 MAOS 性能最好, NSGAII 次之, RM-MEDA 最差, RM-MEDA 只是宏观把握种群信息, 它建模时倾向于发现变量之间的固有关联, 而对于函数的变量之间就没有关联, 因此 RM-MEDA 求解效果最差。

(2) 变量之间存在线性关联的部分 ZDT 和 DTLZ 系列变形函数的实验结果

ZDT1L、ZDT2L、ZDT3L、ZDT6L, DTLZ2L、DTLZ6L 分别是 ZDT1、ZDT2、ZDT6, DTLZ2、DTLZ6 的变形, 它们的 PS 相同, 都是一个线段

$$x_1 = x_2 = \dots = x_n, 0 \leq x_i \leq 1, 1 \leq i \leq n. \quad (3-10)$$

它们变量之间的关联通过对 ZDT, 及 DTLZ 系列函数的变量进行如下线性映射得到:

$$x_1 \rightarrow x_1, x_i \rightarrow x_i - x_1, i = 2, \dots, n \quad (3-11)$$

图 3.4 是三种算法求解 ZDT 及 DTLZ 变形测试问题时, IGD 指标随函数评价次数增加的变化趋势。图 3.5 是三种算法求解 ZDT 及 DTLZ 变形测试问题时, IGD 指标统计盒图。

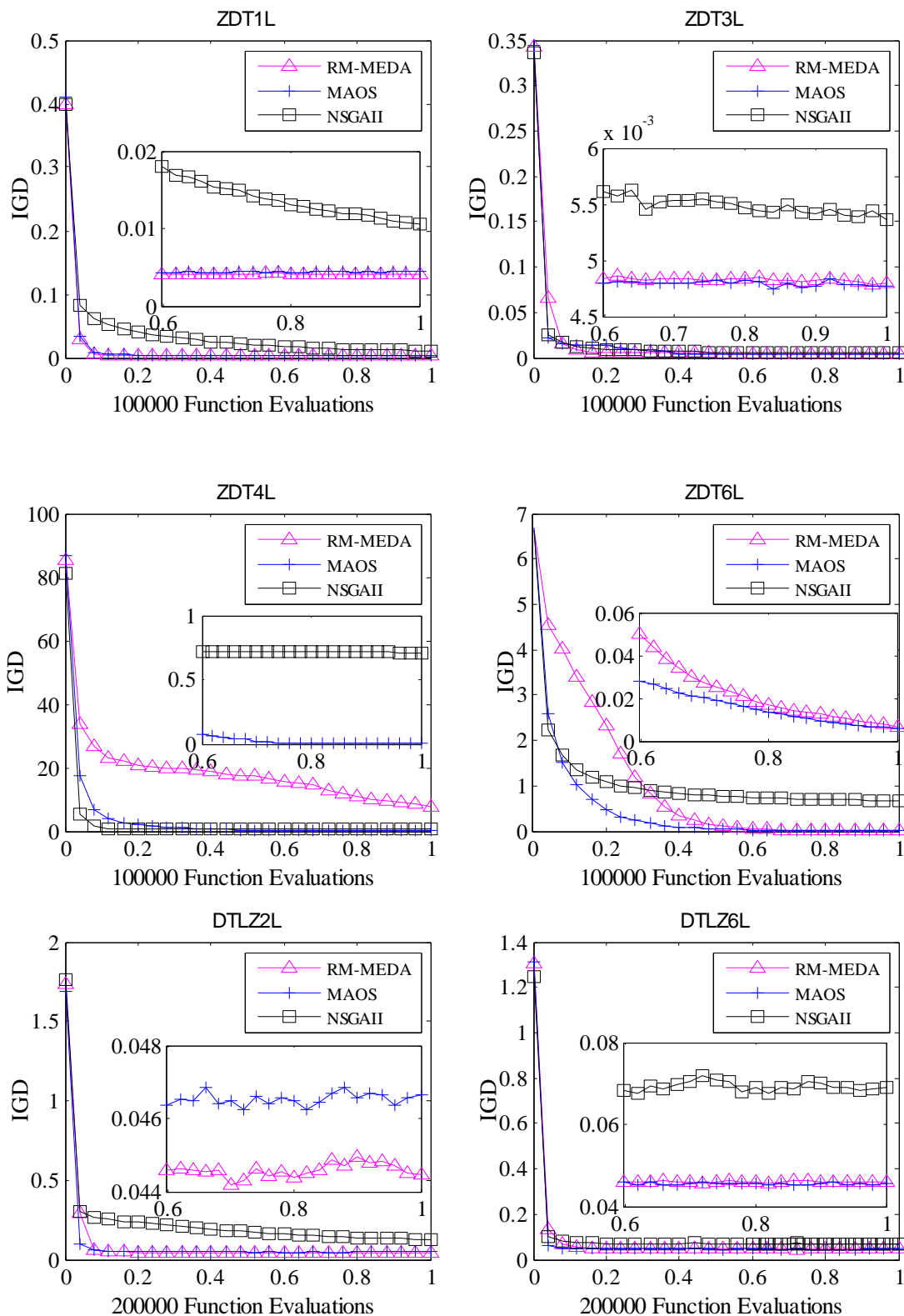


图 3.4 三种算法求解变量之间为线性

关联的部分 ZDT 和 DTLZ 系列测试问题时, IGD 指标随函数评价次数增加的变化趋势

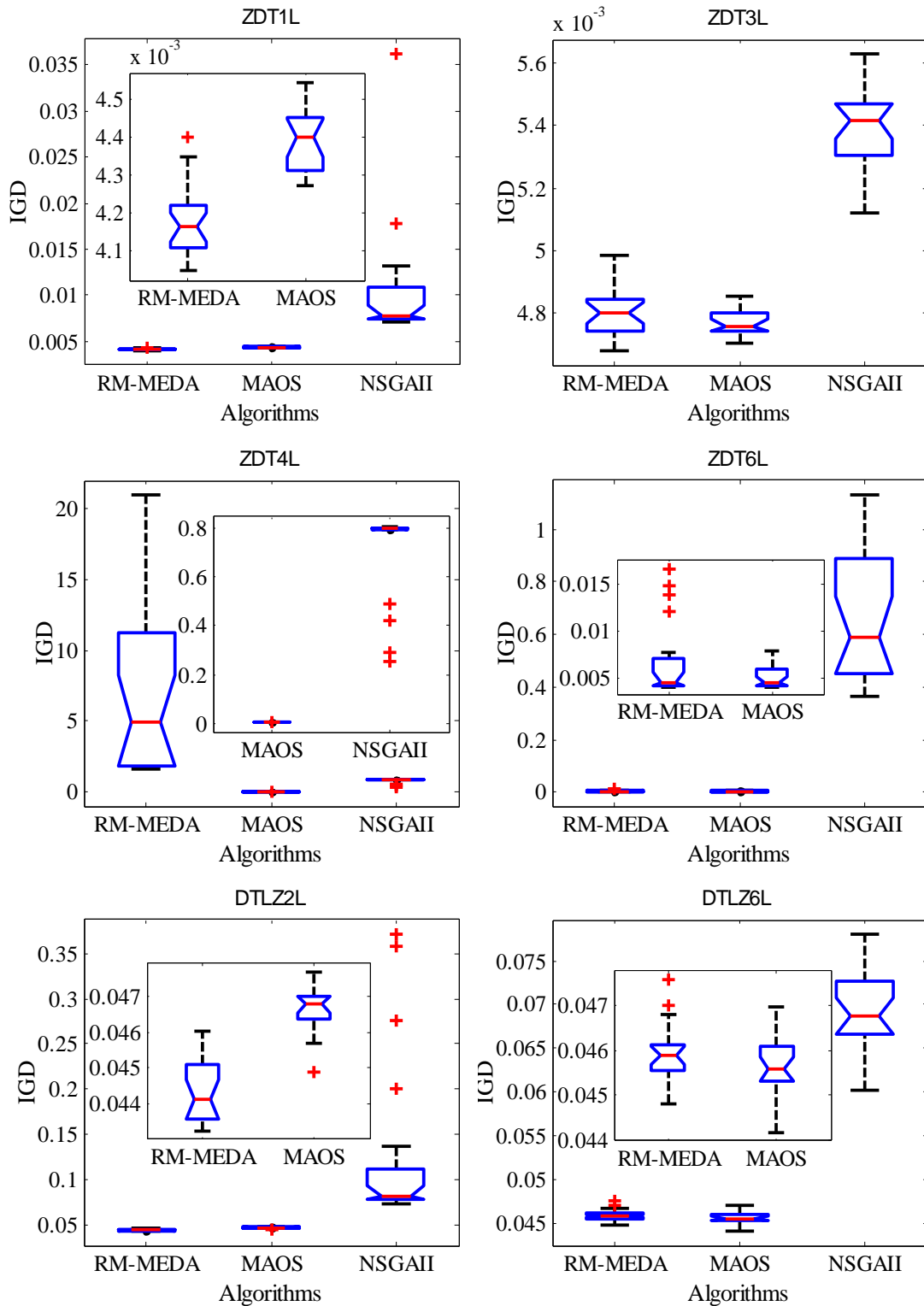


图 3.5 三种算法求解变量之间具有线性关联的部分 ZDT 和 DTLZ 系列测试问题时, IGD 指标统计盒图

从图 3.4 以及图 3.5 可以看出,对于变量之间存在线性关联的 ZDT1L, ZDT3L, ZDT6L 和 DTLZ2L 以及 DTLZ6L 系列变形函数,在收敛性还是多样性, MAOS 和 RM-MEDA 性能相当, NSGAII 性能最差。因为 RM-MEDA 以及 MAOS 算法中

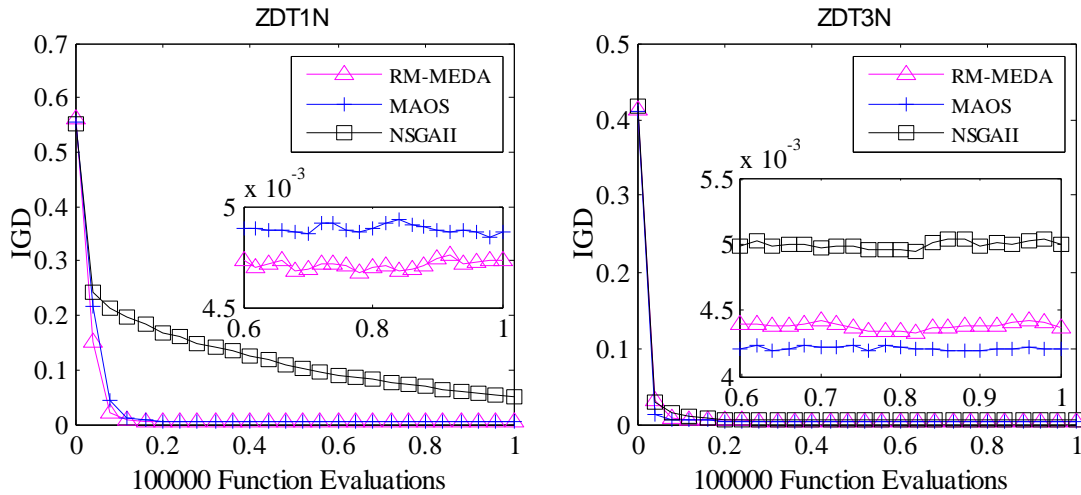
密母算子中采用的模型采样算子很好地利用连续多目标问题 PS 分布呈现一定规则的特点建立多个线性模型,分段逼近真实的 PS,采用模型采点一种方式产生子代,所以效果很好。而 NSGAII 全局搜索能力较弱,对于这几个测试函数,几乎不收敛。同时对于测试函数 ZDT4L 中 RM-MEDA 则没有收敛,这是因为 ZDT4L 测试函数多达 21^9 个 Pareto 最优解,越靠近真正的 Pareto 前沿,局部最优解就越密集,而 RM-MEDA 只是宏观把握种群信息,它建模时倾向于发现变量之间的固有关联,但欠缺局部搜索的能力,容易陷入局部最优。则对于函数形式复杂的 ZDT4L,MAOS 中的密母算子,在进化的不同阶段自适应调整模型采用算子和克隆选择算子的比例,通过建立分段线性模型逼近线性的 PS 流形,同时克隆选择算子在代种群的周围进行了有效的随机搜索,很好地避免了陷入局部最优,很好地收敛到了 Pareto 前端,同时相对于 RM-MEDA 和 NSGAII 得到的解的收敛性、多样性、鲁棒性相对较好。

(3)部分变量之间存在非线性关联的复杂函数的实验结果

ZDT1N、ZDT2N、ZDT3N、ZDT6N, DTLZ2N、DTLZ6N 分别是 ZDT1、ZDT2、ZDT6, DTLZ2、DTLZ6 的变形,这 6 个函数变量之间存在非线性关联,F1 的 PS 是一个有边界的连续曲线(a bounded continuous curve),定义如下

$$x_1 = x_i^2, i = 2, \dots, n, 0 \leq x_1 \leq 1 \quad (3-12)$$

图 3.6 是三种算法求解 ZDT 及 DTLZ 非线性变形测试问题时,IGD 指标随函数评价次数增加的变化趋势。图 3.7 是 3 种算法求解 ZDT 及 DTLZ 非线性变形测试问题时,IGD 指标统计盒图。



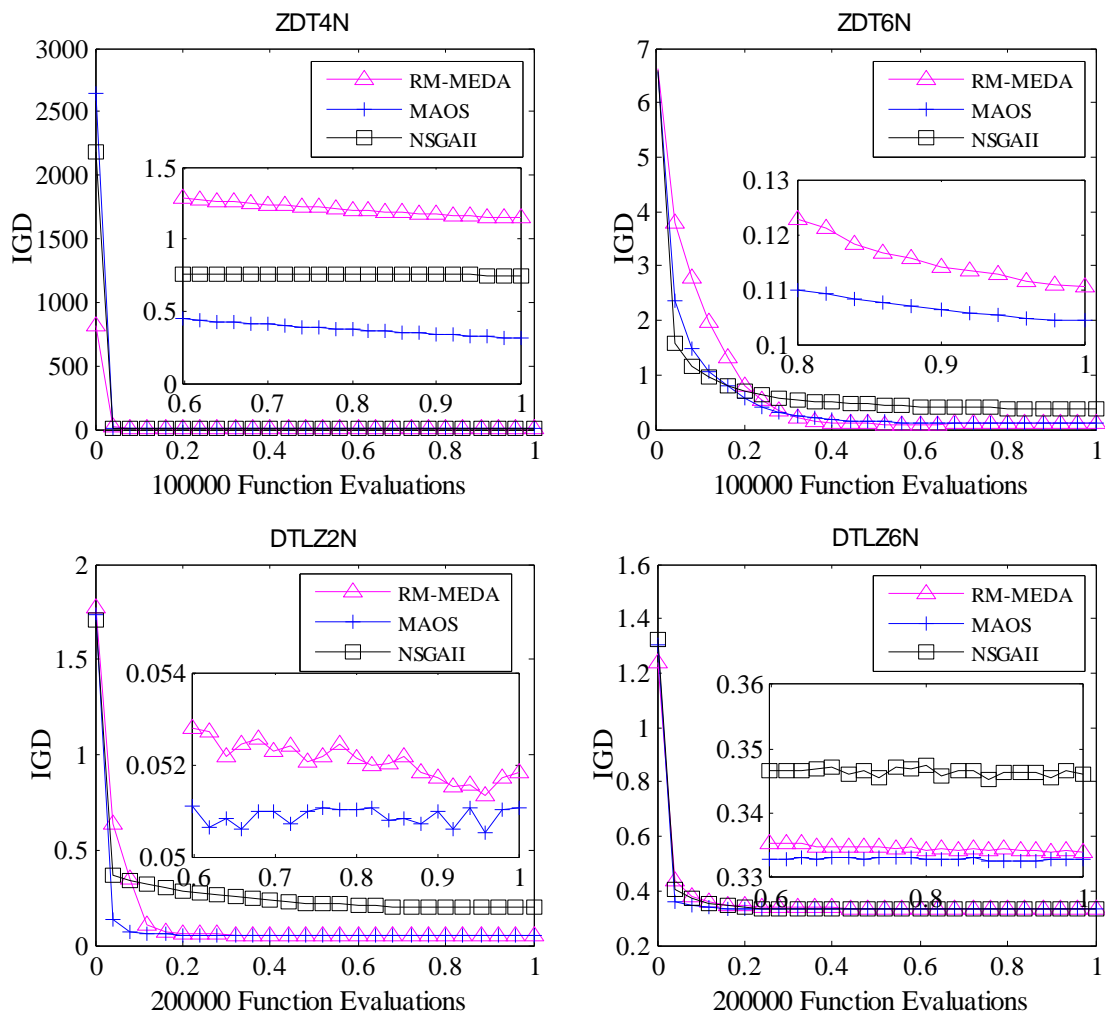
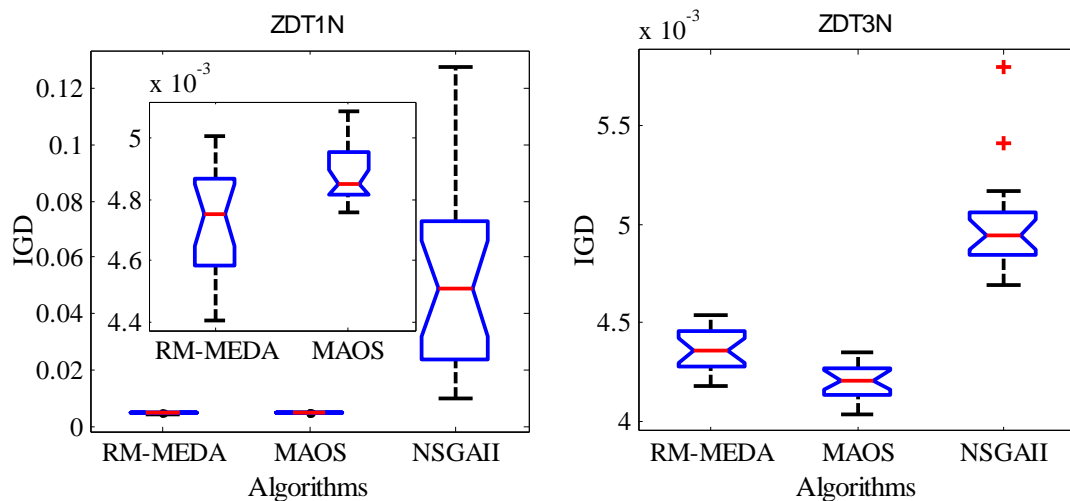


图 3.6 三种算法求解变量之间为非线性

关联的部分 ZDT 和 DTLZ 系列测试问题时, IGD 指标随函数评价次数增加的变化趋势

图 3.7 是三种算法求解变量之间存在非线性关联的测试问题时, 20 次独立运行得到的 IGD 指标的统计盒图。



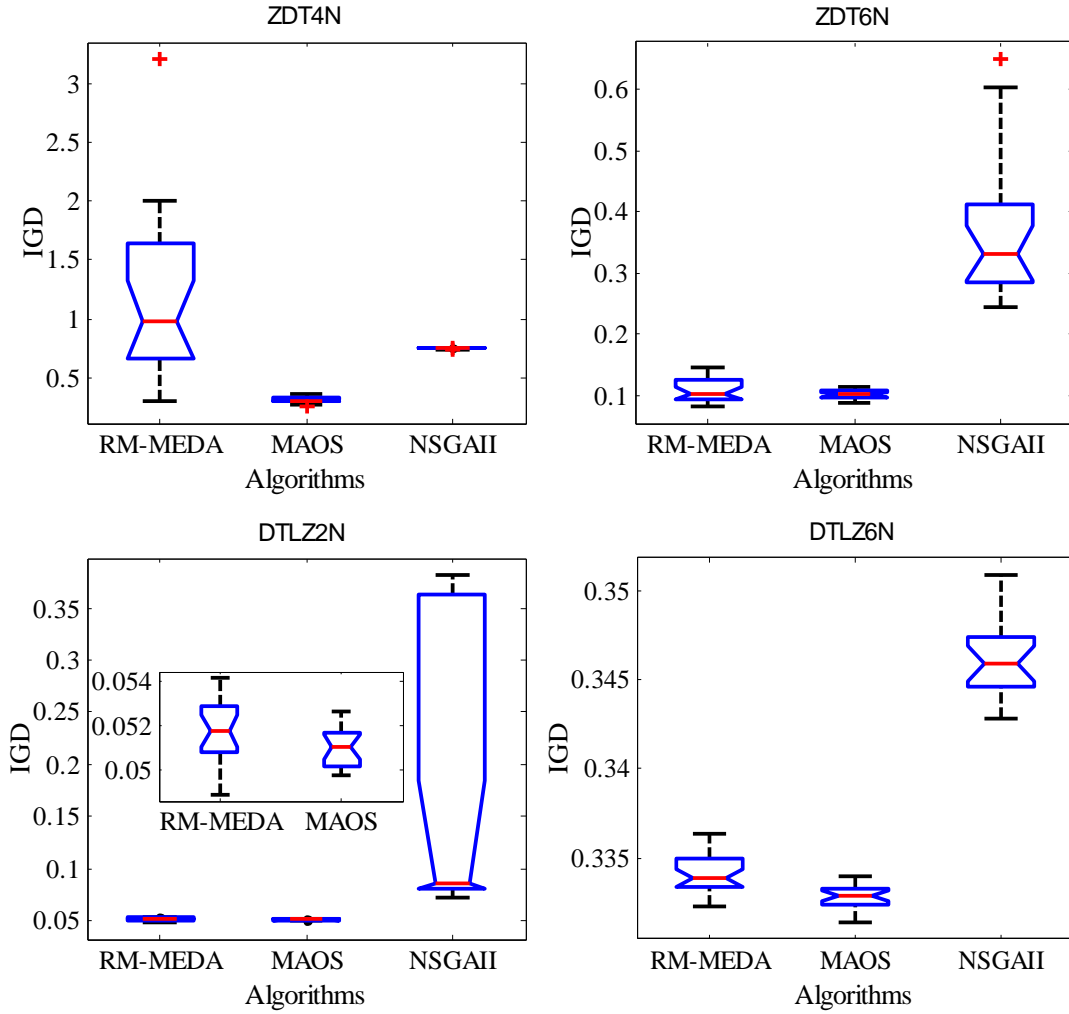


图 3.7 三种算法求解变量之间具有非线性关联的部分 ZDT 和 DTLZ 系列测试问题时, IGD 指标统计盒图

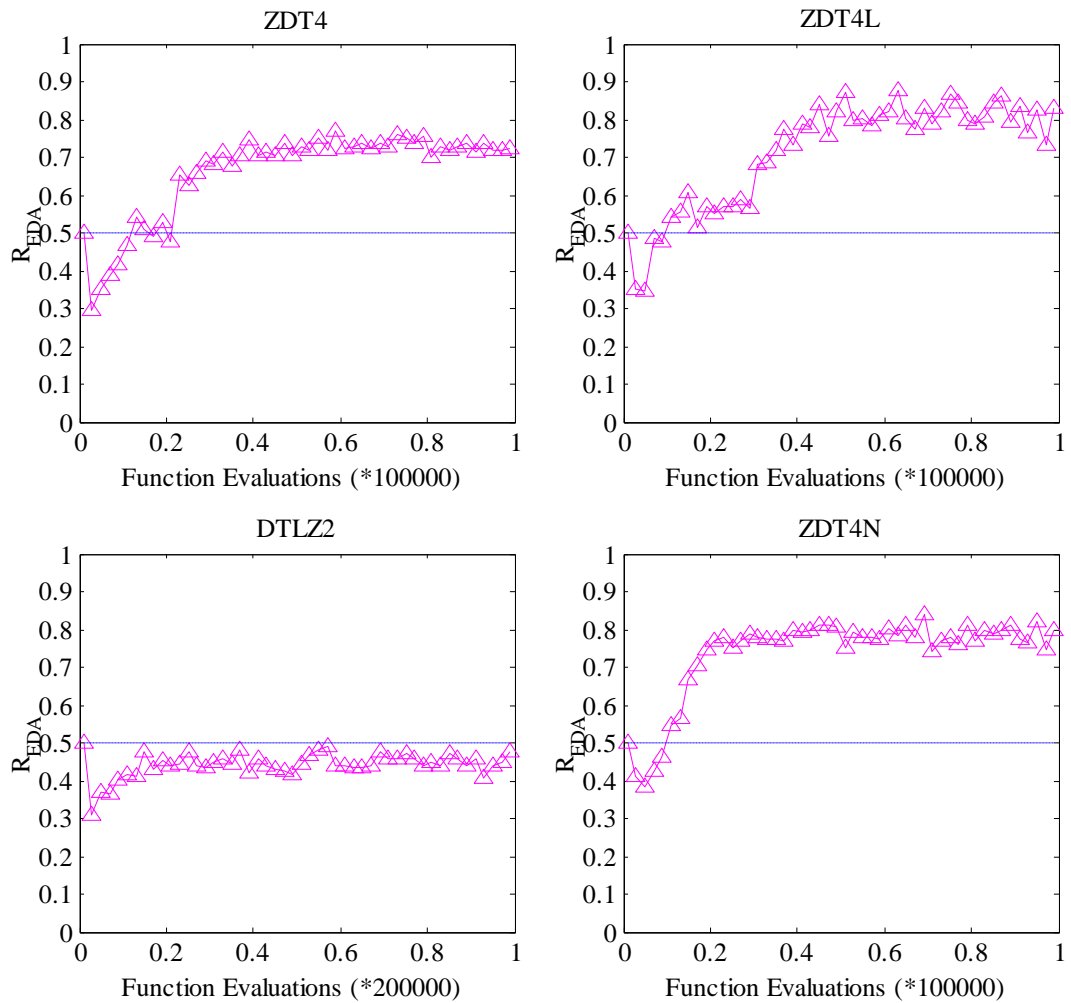
从图 3.6 以及图 3.7 可以看出, 对于变量之间存在非线性关联的复杂函数, MAOS 的性能较稳定, 对于函数形式较为简单的 ZDT1N, MAOS 的性能稍劣于 RM-MEDA。对于其他 5 个测试函数 MAOS 的收敛性还是稳定性均为最优。NSGAII 在以上测试函数上均不收敛, 因为其欠缺种群全局信息的能力, 不适合求解变量之间存在非线性关联的测试函数。RM-MEDA 在 ZDT4N 上不收敛, 陷入局部最优。而 MAOS 中的自适应密母算子中采用了混合子代繁殖策略, 在进化的不同阶段, 自适应的根据能力和效率指标按比例采用不同的子代繁殖策略, 不仅利用了已找到的最优解进行局部搜索, 还宏观把握种群的全局信息进行建模采样。

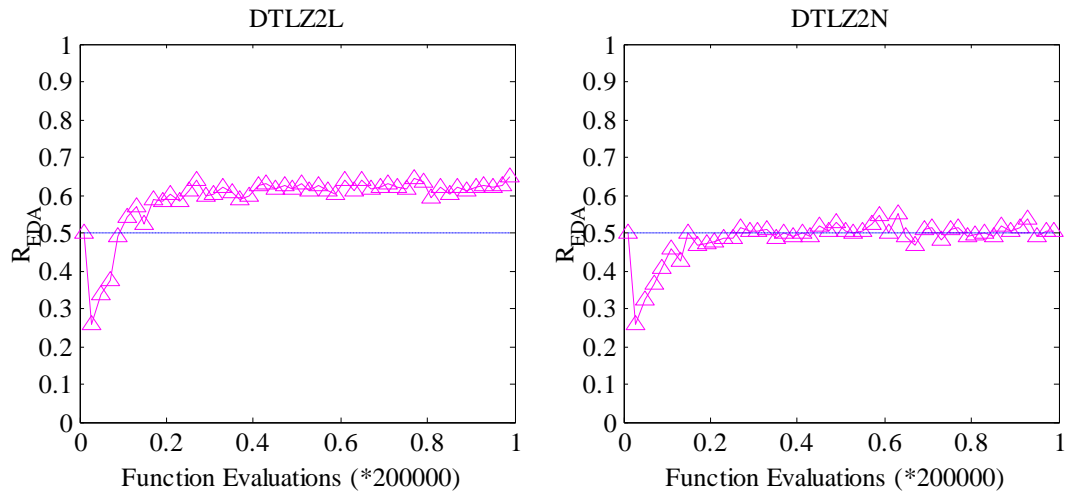
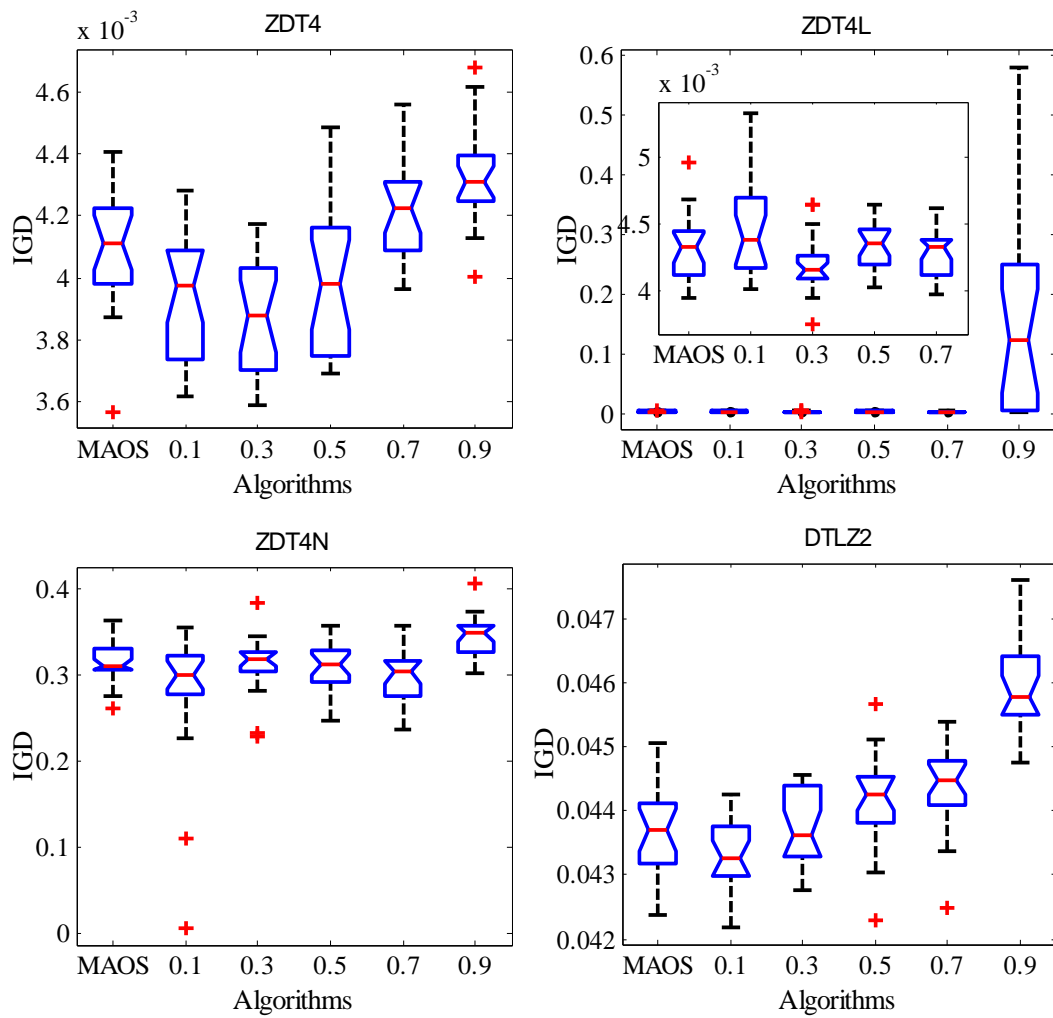
3.4.4.2 MAOS 算法自适应策略的有效性分析

本节将就算法自适应的有效性做以分析, 这部分实验通过对每次迭代中两种算子产生后代的质量进行考察, 分析分析两种算子在进化过程中发挥作用的时机。在实验中, 将每次迭代过程中产生的新抗体放入一个临时种群, 并删除其中被支

配的抗体，构成一个新的 Pareto 最优抗体种群。在这个新种群中，由基于 EDA 的模型采样算子产生的子代抗体的比例记为 R_{EDA} ，那么由基于交叉变异的克隆选择算子产生的子代抗体的比例为 $1-R_{EDA}$ 。图 3.9 给出了对于函数形式最为复杂的测试函数 ZDT4, ZDT4L, ZDT4N, 以及 DTLZ2, DTLZ2L, DTLZ2N 中, R_{EDA} 随着迭代的进行而变化的曲线，图中数据是 20 次独立运行的统计结果。

从图 3.8 可以看出, R_{EDA} 曲线没有收敛到边界值 0 或 1, 这说明在整个进化过程中, 两种算子均发挥了一定的作用, 只是对不同的问题, 在进化的不同阶段发挥作用的大小不同。对于多峰函数的 ZDT4, ZDT4L, ZDT4N 系列问题, 以及三目标的 DTLZ2, DTLZ2L, DTLZ2N 测试问题, R_{EDA} 曲线在进化前期有下降的局势, 在进化前期种群没有充分收敛, 因此模型采样算子能够产生最优个体的比例较少, 此时基于交叉变异的克隆选择算子主要发挥作用吗, 随着进化种群的收敛, 模型采样算子产生的最优解数目增加, 最终和基于交叉变异的克隆选择算子相互合作, 两个算子在进化的不同阶段发挥作用, 从而使得算法具有更好的稳定性。



图 3.8 R_{EDA} 随着函数评价次数的变化曲线

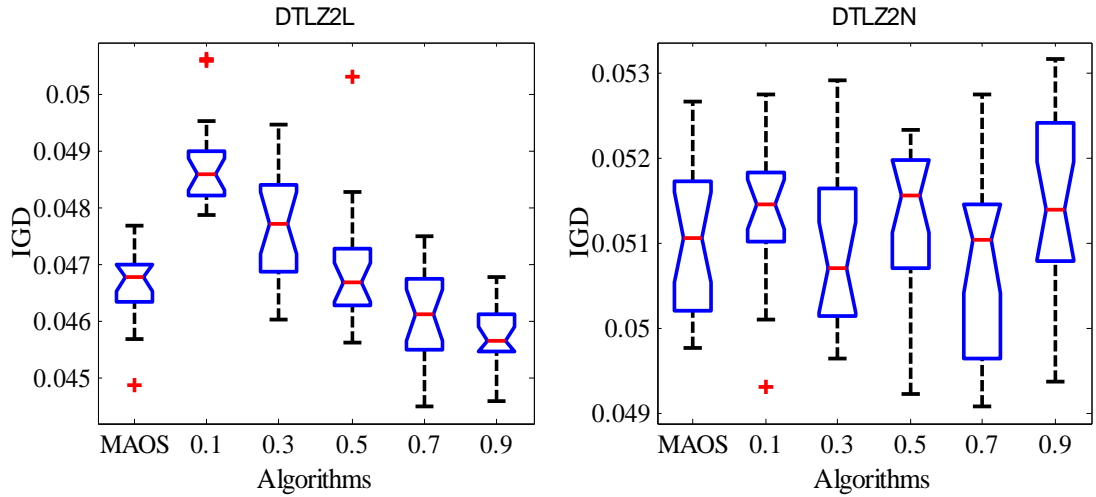


图 3.9 密母算子中两种算子采用不同比例时 IGD 指标统计盒图

本节中以函数形式最为复杂的 ZDT4, ZDT4L, ZDT4N 系列问题, 以及三目标的 DTLZ2, DTLZ2L, DTLZ2N 测试问题为例, 来说明算法中自适应密母算子的有效性, 图 3.9 中给出了当模型采样算子在进化种群中使用比例分别为 0.1, 0.3, 0.5, 0.7, 0.9 时, IGD 指标统计盒图。从统计盒图可以看出 MAOS 的算法性能在两种算子不同比例设定中, 无论收敛性和稳定性均处于较优的水平。

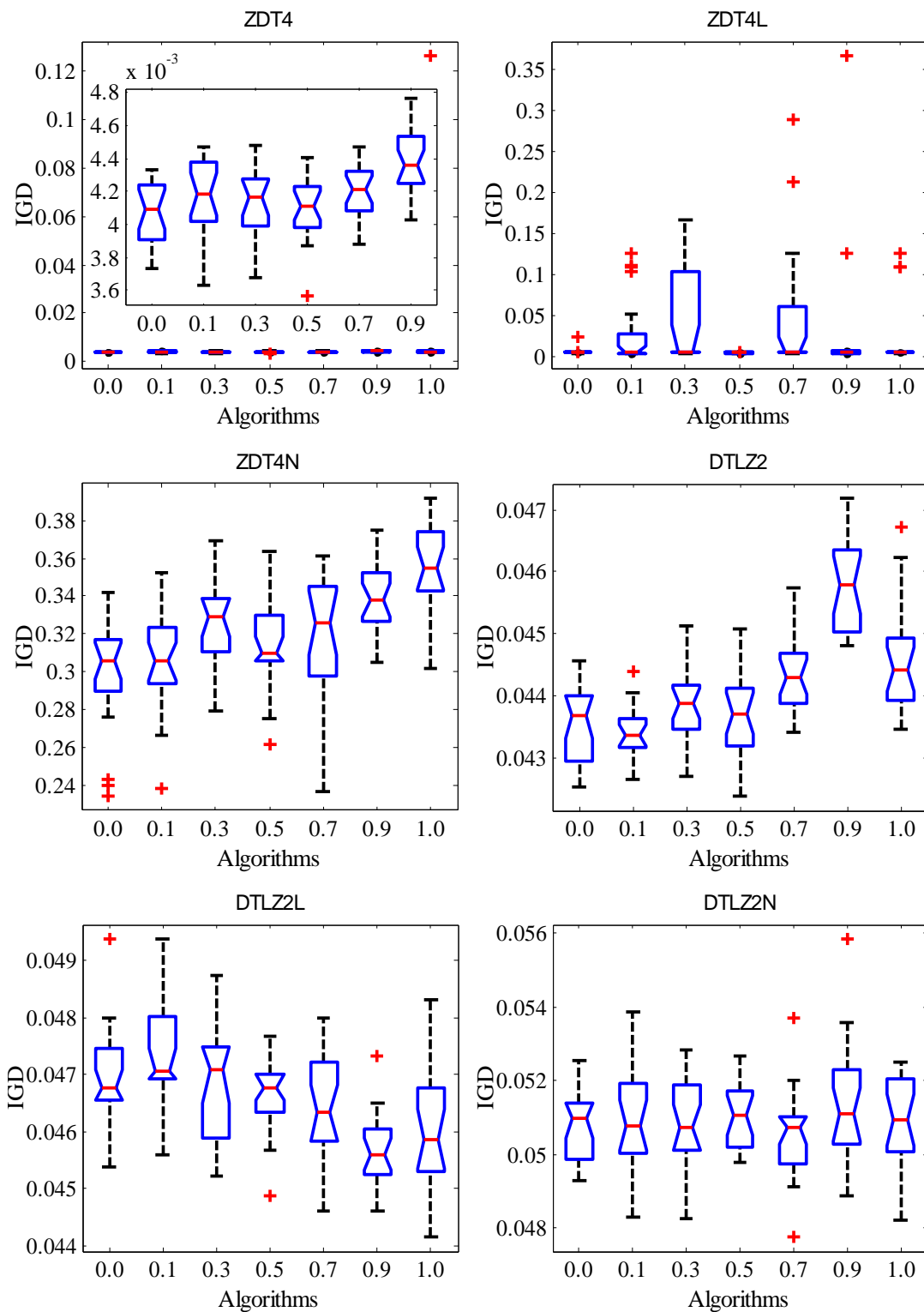
3.4.4.3 MAOS 算法中自适应密母算子的权值参数分析

本文算法中自适应密母算子中涉及到信用赋值中, CA_{CSO} 和 CA_{EO} 权值 w 值的设定, 其中有:

$$CA_{CSO} = wCR_{CSO} + (1-w)ER_{CSO}, 0 \leq w \leq 1 \quad (3-13)$$

$$CA_{EO} = wCR_{EO} + (1-w)ER_{EO}, 0 \leq w \leq 1 \quad (3-14)$$

下图 3.10 中依然以函数形式最为复杂的测试函数 ZDT4, ZDT4L, ZDT4N, 以及 DTLZ2, DTLZ2L, DTLZ2N 为例, w 取值分别为 0.0, 0.1, 0.3, 0.5, 0.7, 0.9, 1.0。从下图的 IGD 指标盒图可以看出当 w 取值为 0.5 时, 算法的收敛性和稳定性均处于较优的水平。

图 3.10 密母算子信用赋值中权值 w 取不同值时 IGD 指标统计盒图

3.5 本章小结

本章是在基于免疫算法和 EDA 的混合多目标优化算法 HIAEDA 基础上, 提出

了一种求解复杂多目标优化问题的进化算子自适应选择的多目标优化算法 **MAOS**, 其通过设定效率和能力两种指标, 在算法进化过程中动态调整密母算子中, 模型采样算子和基于交叉变异的克隆选择算子这两种算子采用的比例, 从而使两种算子在算法进化不同阶段来充分发挥各自的优势, 在挖掘多目标优化问题变量之间相关性的同时, 有效利用进化过程中已找到最优解的位置信息, 加大在稀疏区域的搜索, 仿真实验表明 **MAOS** 与著名的 **NSGAII** 算法和基于 **EDA** 的进化多目标优化算法 **RM-MEDA** 相比, 对于变量之间没有关联的函数以及变量之间有非线性关联的部分复杂多目标优化问题, 在收敛性和多样性方面均都表现出明显优势。同时分针实验同时证明了 **MAOS** 算法中自适应密母算子的有效性, 同时对算法中效率和能力指标的权值参数做了相应的参数分析。

第四章 聚类数目自适应调整的 RM-MEDA

本章通过把基于种群中流形(Manifold)信息而定义的合并分裂算子(Merging and Dividing Operator)加入到 RM-MEDA 算法框架中,从而提出了一种聚类数目自适应调整的 RM-MEDA 算法,该算法中利用合并分裂算子自适应调整 RM-MEDA 算法中的聚类的数目,本章中称该算法为 ACRM-MEDA。

本章首先给出了新算法的创新动机以及创新点,接下来对算法的思想和具体流程进行了详细阐述,测试了大量经典多目标优化问题,并将实验结果和 RM-MEDA、NSGAII 进行了对比。

4.1 引言

一些研究者尝试采用 EDA 算法求解连续多目标优化问题^[23-26],其中 Q. Zhang 等人提出的 RM-MEDA^[20]及其推广算法 MMEA^[25]是最具有代表性的工作。RM-MEDA 充分利用了连续多目标优化问题的特点,即对于目标个数为 m 的连续多目标问题,由 Karush-Kuhn-Tucker 条件可知,Pareto 最优解集合(Pareto Set, PS)在决策空间上的分布呈现分段连续的 $(m-1)$ 维流形(Manifold)^[19]。RM-MEDA 算法的基本思想是对进化种群进行聚类获得多个进化子种群,从而建立多个线性 $(m-1)$ 维流形分布的概率模型来分段逼近整个非线性的 PS 流形,再对各个线性模型进行随机采样产生后代种群。由于 RM-MEDA 充分考虑了 Pareto 最优解之间的相关性,该算法对于求解决策变量之间具体相关的复杂多目标优化问题十分有效。但 RM-MEDA 中同时也指出了 RM-MEDA 中聚类的个数与具体问题相关,而在原算法中对于不同问题的聚类数目 K 设置成了常量(K 为 5),然而对于不同的测试问题,其 PS 形状不同,相应需要合理的聚类数目来更准确的分段逼近整个 PS 流形。Wang^[56]等人针对此问题设计了一种减少冗余聚类的算子(Reducing redundant cluster operator, RRCO)来减少初始聚类中冗余聚类来建立更加精确的线性模型,但该文中设计的 RRCO 算子仅适应的初始聚类大于实际合理聚类的情况,并不能处理当聚类数目小于实际聚类数目的情况。

在本章提出了一种合并分裂算子来根据种群流形信息自适应调整聚类数目实现子种群的自适应合并和分裂,从而建立更精确的线性模型来产生后代种群。通过把合并分裂算子加入到 RM-MEDA 算法框架中,在本章中我们提出了聚类数目自适应调整的 RM-MEDA 算法。

4.2 算法的基本思想

本章介绍的所提算法 ACRM-MEDA 中, 主要通过合并分裂算子来自适应调整初始聚类来建立更为精确的多个线性 $(m-1)$ 维流形分布的概率模型来分段逼近整个非线性的 PS 流形, 再对各个线性模型进行随机采样产生部分后代种群。其中合并算子用来合并冗余的聚类, 分裂算子用来拆分不合理的聚类。通过合并分裂算子对种群聚类的自适应调整, 从而让每个线性模型更加准确的逼近子种群中的 PS 流形。下图 4.1 中为一个变量之间具体非线性关联多目标测试问题(TestA)的 PS 流形, 其中其 PS 流形为一维的分段线性流形: 其变量之间的关系为:

$$x_i = \sin 2\pi x_1, i = 2, 3, 4, \dots, n, 0 \leq x_1 \leq 1 \quad (4-1)$$

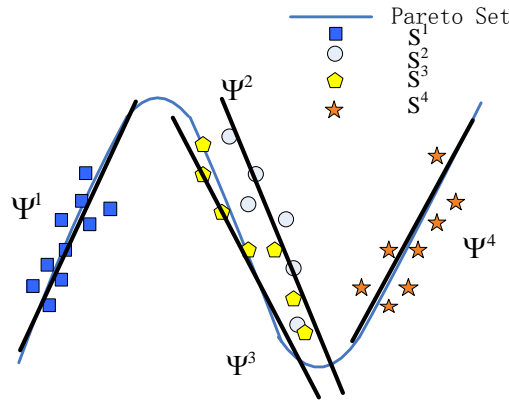


图 4.1 聚类数 K 为 4 时种群 PS 流形的建模情况

如图 4.1 所示, 其中种群的聚类数目为 K 为 4, 因此 RM-MEDA 算法中会生成 4 子种群, 图中分别用 S_1, S_2, S_3, S_4 代表聚类后的子种群, Ψ^1, Ψ^2, Ψ^3 和 Ψ^4 分别是根据 4 类子群体建立的分段逼近 1 维 PS 流形的线性模型。可以从图中看出子种群 S_2 和 S_3 处于 PS 流形的同一段上, 因此建立的分段线性模型 Ψ^3, Ψ^4 是冗余的, 子种群 S_2 和 S_3 应该合并为一类, 从而建立一个线性分段模型来逼近这段 PS 流形。对于此种类型的问题, 本章中定义借鉴 Wang 等人 IRMMEDA^[56] 算法中的 RRCO 算子设计了一种合并算子, 来把符合合并准则的聚类进行合并。

本章中定义判断连个子种群是否能够合并的规则如下:

对于聚类生成的两个子种群 S_i, S_j , Ψ^i, Ψ^j 分别为用来逼近这两个子种群对应流形的线性模型。点 a, 点 b 分别为这两个子种群的中心点。线段 \overline{ab} 为这两个中心点的连线。当满足以下两个准则时, 可以判定两个线性模型 Ψ^i, Ψ^j 是冗余的, 可以合并两个子种群 S_i, S_j :

准则 1: $\langle \Psi^i, \Psi^j \rangle < \theta$, 使线性模型 Ψ^i, Ψ^j 之间夹角小于设定阈值 θ 。

准则 2: $\min(\langle \overline{ab}, \psi_i \rangle, \langle \overline{ab}, \psi_j \rangle) < \langle \psi_i, \psi_j \rangle$ 。

对于以上 2 条准则，准则 1 可以判断出图 4.1 中模型 ψ_2, ψ_3 的这类冗余状况，但对于模型 ψ_1, ψ_4 这样的近似平行状况，虽然两个模型夹角很小，但是仍不符合合并条件，则可以通过准则 2 来避免此类错误合并状况，因为两个模型中心点连线与模型的夹角 $\langle \overline{ab}, \psi_i \rangle, \langle \overline{ab}, \psi_j \rangle$ 会远大于两个线性模型的夹角 $\langle \psi_i, \psi_j \rangle$ ，以上所涉及夹角均根据高维欧式几何理论^[47,48]来计算。

基于以上分析合并算子的具体流程如下：

算法 4.1 合并算子

初始化. 生成现存 K 个子种群对应的集合 $S=\{S_1, S_2, \dots, S_K\}$ ，设置子种群数目临时标记变量 $C=1$ 。

Step 1. 计算每个子种群中抗体的数目，把种群大小为 1 的子种群和其相邻的子种群合并。得到调整后的 K' 个聚类，从而得到新的子种群集合为 $S'=\{S_1, S_2, \dots, S_{K'}\}$ ，计算每个子种群相应的均值点为 $m_i (i=1, \dots, K')$

Step2. While $C \neq K'$ Do

 计算得到满足 $\min(m_c - m_i) (i=1, \dots, K', i \neq c)$ 的聚类 C_{\min} 。判断子种群 C 与子种群 C_{\min} 是否满足合并准则。

If 两个子种群 C 和 C_{\min} 满足合并准则

 子种群 C 和子种群 C_{\min} 进行合并，并为得到的新的子种群用 LCPA 方法重新建立线性模型；

$K' = K' - 1;$

Else

$C = C + 1;$

End If

End While

$K = K'$

下面以图 4.2 为示例进一步解释分裂算子的原理及流程，同样对于多目标测试问题 TestA，当选取聚类种群为 2 时，RM-MEDA 算法中会生成 2 子种群，图中分别用 S_1, S_2 代表聚类后的子种群， Ψ^1, Ψ^2 分别是根据 2 类子群体建立的分段逼近 1 维 PS 流形的线性模型。从图 4.2 可以看出，设定固定聚类数目时，后出现错误的聚类情况，线性模型 Ψ^1 并未正确的逼近 PS 流形，从而导致建立线性模型失败，进化过程中并不能得到较优的解。针对这种情况，本章中定义一种分裂算子来把分裂准则的聚类进行分割，从而建立和 PS 流形方向更为接近的线性模型，来提高算法的性能。

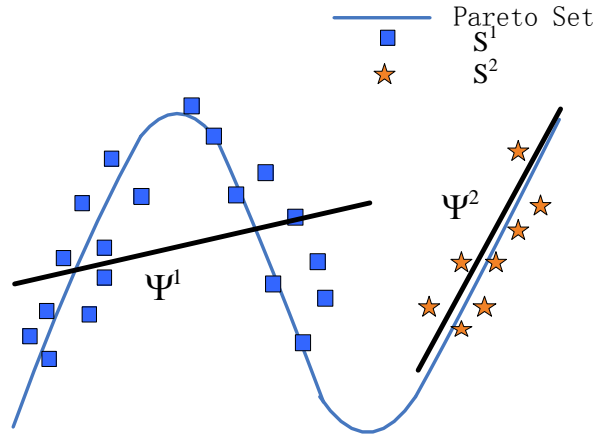


图 4.2 聚类数 K 为 2 时种群 PS 流形的建模情况

对于子种群 S_k 计算其种群中的个体到中心线 L_k 的距离的平方和 D_k 。其中

$$D_k = \sum_{i=1}^{N_k} [d(x^{k,i}, L_k)]^2 \quad (4-2)$$

计算所有子种群 D_k 的均值 D_{mean} ，对于各个子种群从 $D_k (k=1, \dots, K)$ 中选取最大误差 D_k 的聚类，假设第 I 个子种群为有最大平方和的聚类，则如果满足 $\frac{D_I}{D_{mean}} > \alpha$ ，

为算法中设置的阈值参数则把第 I 个子种群从新用 LPCA 聚成 2 类，并建立相应线性模型。假设将该子种群划分两个新的子种群 S_i, S_j ， S_i 的种群大小为 N_i^k ， S_j

的种群大小为 N_j^k ，其中 $N_i^k + N_j^k = N^k$ 。对于 S_i, S_j 分别对应的线性模型为 Ψ^i, Ψ^j 。

分裂算子的具体流程如下：

算法 4.2: 分裂算子

初始化. 生成现存 K 个子种群对应的集合 $S=\{S_1, S_2, \dots, S_K\}$ ，设置子种群数目临时标记变量 $Q=K$ 。

Step 1. 对于每个子种群形成的集合 $S=\{S_1, S_2, \dots, S_K\}$;

Step 2. For $i=1:K$

If 子种群 S_i 满足分裂准则

 生成两个新的子种群，并对这两个子种群重新建立线性模型。

$i=i+1$

$Q=Q+1$

End If

End For

$K=Q$

4.3 算法的描述

ACRM-MEDA 算法的抗体编码采用了 0 到 1 之间的实数编码方式, 抗体 $A=(a_1, a_2, \dots, a_n)$ 的每一个基因位 $a_i (i=1, 2, \dots, n)$ 表示决策向量的一维。在已知每一维决策变量的取值范围 $[\min_i, \max_i]$ 的前提下, 决策变量的取值可以解码为 $v_i = \min_i + a_i(\max_i - \min_i)$ 。

算法的主要流程描述如下:

算法 4.3: ACRM-MEDA 算法流程

Step1 初始化: 设置种群规模 N , 子种群个数 K , 设置算法停止条件。令迭代次数 $t=1$, 随机初始化种群并计算初始种群中个体的目标函数值;

Step2 判断停止条件: 若算法停止条件满足, 则输出种群 $P(t)$, 否则转 Step3;

Step3 繁殖:

Step 3.1 分组建模: 采用 Local PCA(LPCA)聚类算法^[43]将当前种群 $P(t)$ 中的个体聚为 K 类, 分别构成 K 个子种群 S_1, S_2, \dots, S_K , 采用 RM-MEDA 中所述的建模方法对各个子种群建立概率模型 $\Psi^1, \Psi^2, \dots, \Psi^K$;

Step 3.2 调整聚类数目: 采用 4.1 节介绍的合并分裂算子更新聚类子种群数目 K ;

Step3.3 分配繁殖数量: 假设第 i 个子种群产生后的个数为 N_i , 其中 $i=1, 2, \dots, K$, $N = \sum_{i=1}^K N_i$, 则可以从公式(3-1)获得:

$$N_i = \left\lfloor N * \frac{\text{vol}(\Psi^i)}{\sum_{j=1}^K \text{vol}(\Psi^j)} \right\rfloor \quad (4-3)$$

其中 $\text{vol}(\Psi^i)$ 表示种群 S_i 中所有个体在前 $m-1$ 个主分量方向上投影的线段长度 (两目标问题)、面积 (三目标问题)、体积 (四目标问题) 或超体积 (众目标问题), $\lfloor x \rfloor$ 表示对实数 x 取下整。

Step 3.4 采用模型采样算子产生后代: 对于每个子种群 $S_i (i=1, 2, \dots, K)$ 建立的概率模型 $\Psi^1, \Psi^2, \dots, \Psi^K$ 进行随机采样产生后代种群 S_i' , 采样抗体个数为 N_i ;

Step4 选择: 令 $Q = P(t) \cup S_1' \cup S_2' \dots$, 采用非支配排序选择方法^[7]从中选出 N 个个体构成新的种群, 令 $t=t+1$, 转 Step2。

4.4 算法分析

本章把一种合并分裂算子引入到 RM-MEDA 算法框架中, 很好的克服了 RM-MEDA 中设置固定聚类数目固定而不能较为合理准确地建立线性模型的不足。同时本算法中算法复杂度主要由算法 Local PCA 聚类的计算复杂度所决定, 在第三章中已经分析过 Local PCA 的算法复杂度为 $O(N^2)$ 。因此本算法的算法复

杂度为 $O(N^2)$ 。

4.5 仿真实验

实验设计中, 我们测试了已有的 3 类标准测试函数, 其中包括变量之间没有关联的 ZDT 系列函数, 变量之间存在线性关联的 ZDT 变形函数以及变量之间存在非线性关联的复杂函数, 并将实验结果和经典算法 RM-MEDA、NSGAII 进行了对比。

4.5.1 标准测试函数

本文基于 ZDT 测试函数构造了 6 个具有不同流形结构的多目标测试函数, 其中 F1,F2 是变量之间具有线性关联的测试函数, 其 PS 在决策空间是一维线段; F3,F4, F5,F6 为变量之间具体非线性关联的测试函数, 其中 F3,F4 的 PS 流形结构在决策空间为一维 2 次曲线, F5,F6 的 PS 流形结构在决策空间为一维正弦曲线。函数具体形式如下表所示。其中图 4.3 分别为三种不同类型函数的 PS 流形的形状。

表 4.1 本文的测试问题

函数	变量	维数	目标函数值 (最小)
F1	[0,1]	30	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - \sqrt{x_1/g(x)}]$ $g(x) = 1 + 9(\sum_{i=2}^n (x_i - x_1)^2)/(n-1)$
F2	[0,1]	30	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - (x_1/g(x))^2]$ $g(x) = 1 + 9(\sum_{i=2}^n (x_i - x_1)^2)/(n-1)$
F3	[0,1]	30	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - \sqrt{x_1/g(x)}]$ $g(x) = 1 + 9(\sum_{i=2}^n (x_i^2 - x_1)^2)/(n-1)$
F4	[0,1]	30	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - (x_1/g(x))^2]$ $g(x) = 1 + 9(\sum_{i=2}^n (x_i^2 - x_1)^2)/(n-1)$

F5	$[0,1] \times [0,10]^{n-1}$	30	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - \sqrt{x_1/g(x)}]$ $g(x) = 1 + 9(\sum_{i=2}^n (\sin(2\pi x_i) - x_i)^2)/(n-1)$
F6	$[0,1] \times [0,10]^{n-1}$	30	$f_1(x) = x_1$ $f_2(x) = g(x)[1 - (x_1/g(x))^2]$ $g(x) = 1 + 9(\sum_{i=2}^n (\sin(2\pi x_i) - x_i)^2)/(n-1)$

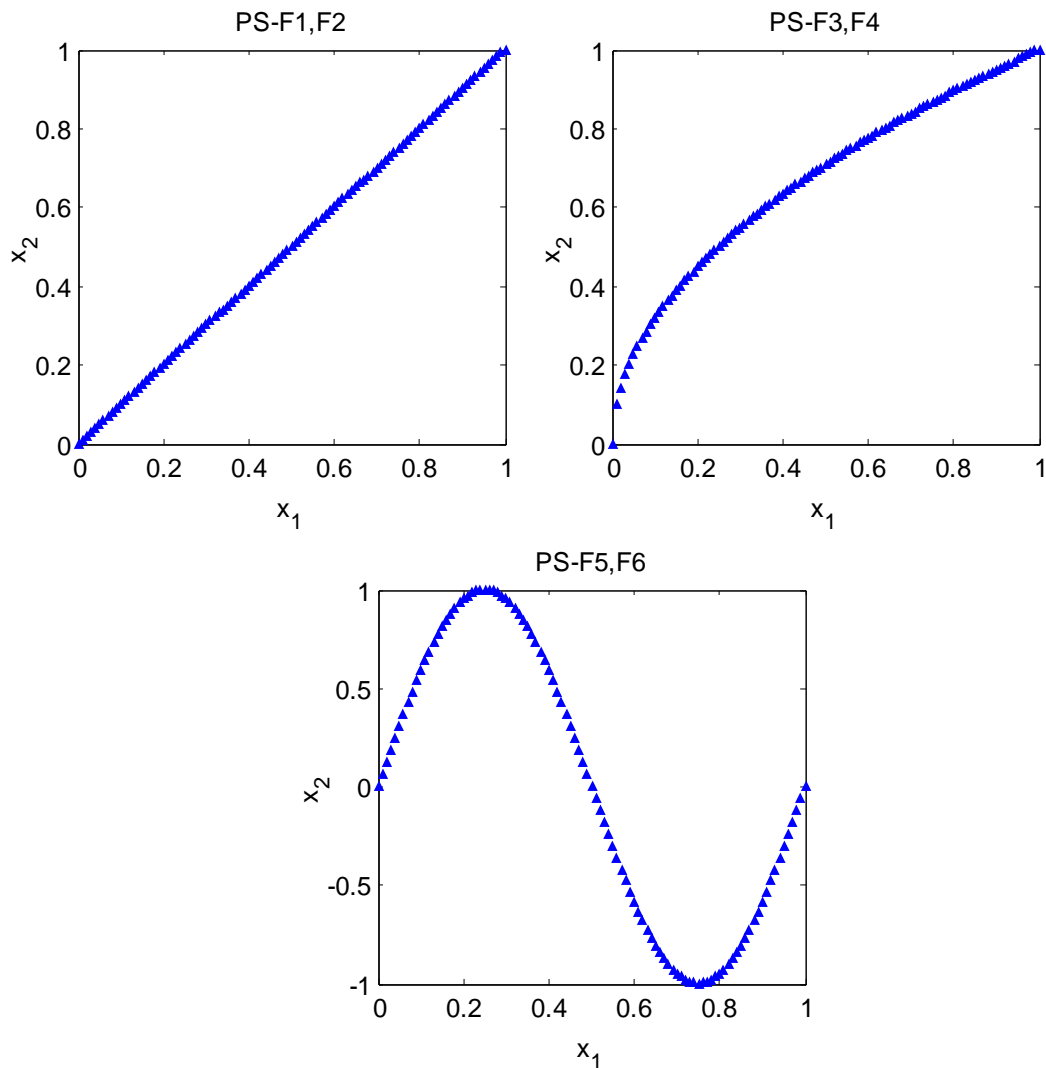


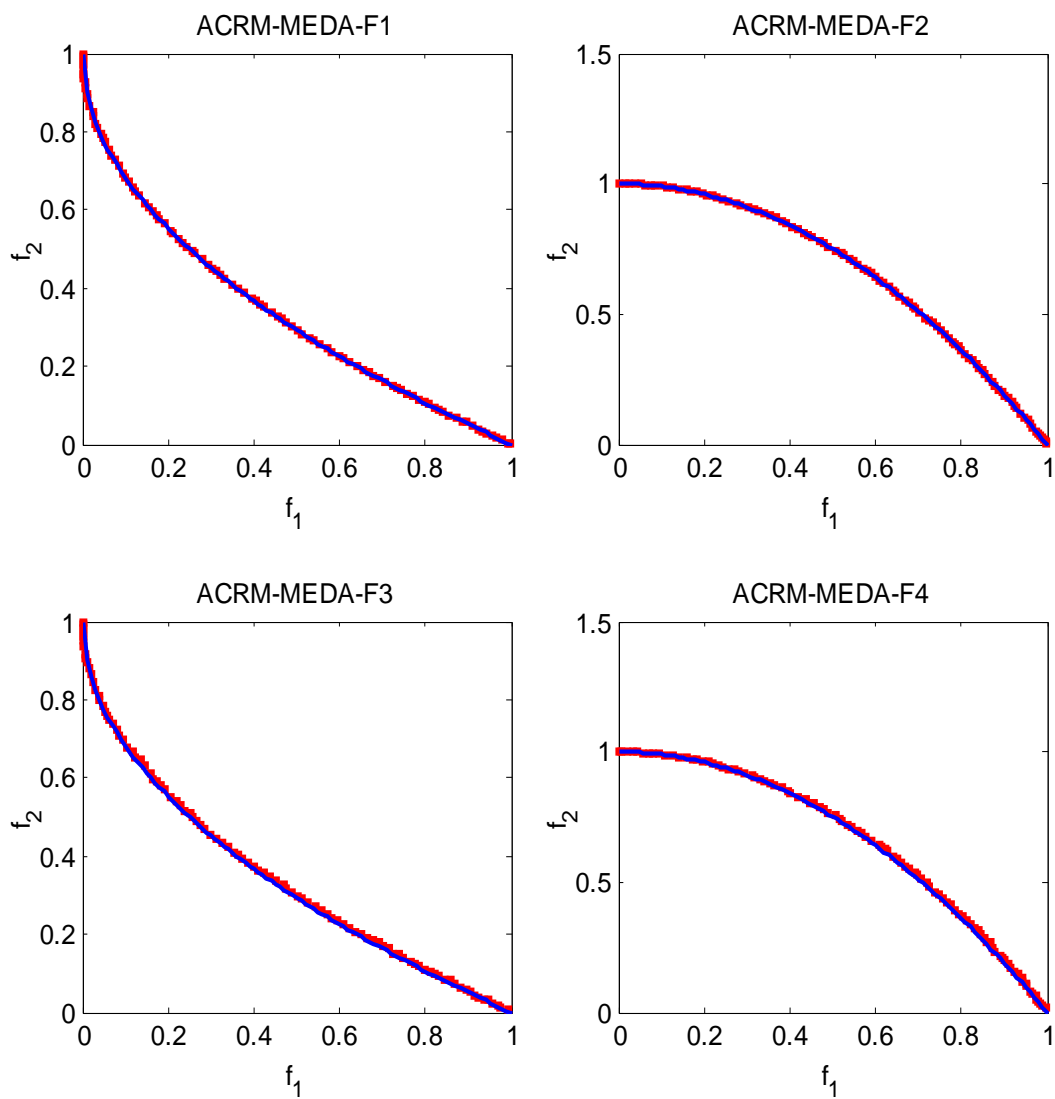
图 4.3 三种不同类型测试函数的 PS 流形

4.5.2 实验参数设置

在三种算法中，所有测试函数的种群规模都设为 100；决策变量的维数为 30；F1,F2,F3,F4 的函数评价次数为 10000，F5，F6 的函数评价次数为 50000；

RM-MEDA, RM-MEDA+C,ACRM-MEDA 的初始聚类数目为 5。每个函数独立运行次数为 20。在算法 RM-MEDA+C 和 ACRM-MEDA 中 θ 设为 5° ，算法 ACRM-MEDA 中分裂算子中的阈值参数 α 设为 3。

4.5.3 实验结果及分析



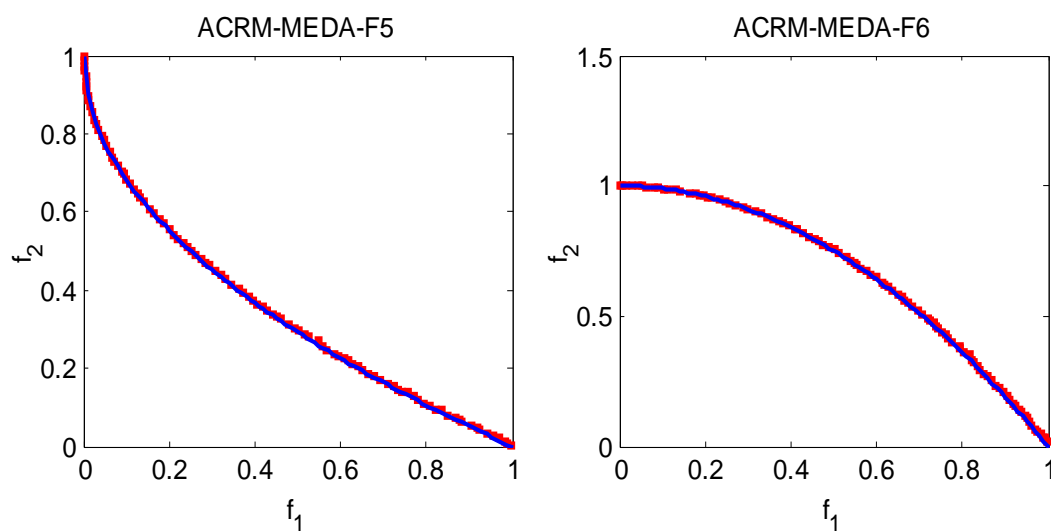
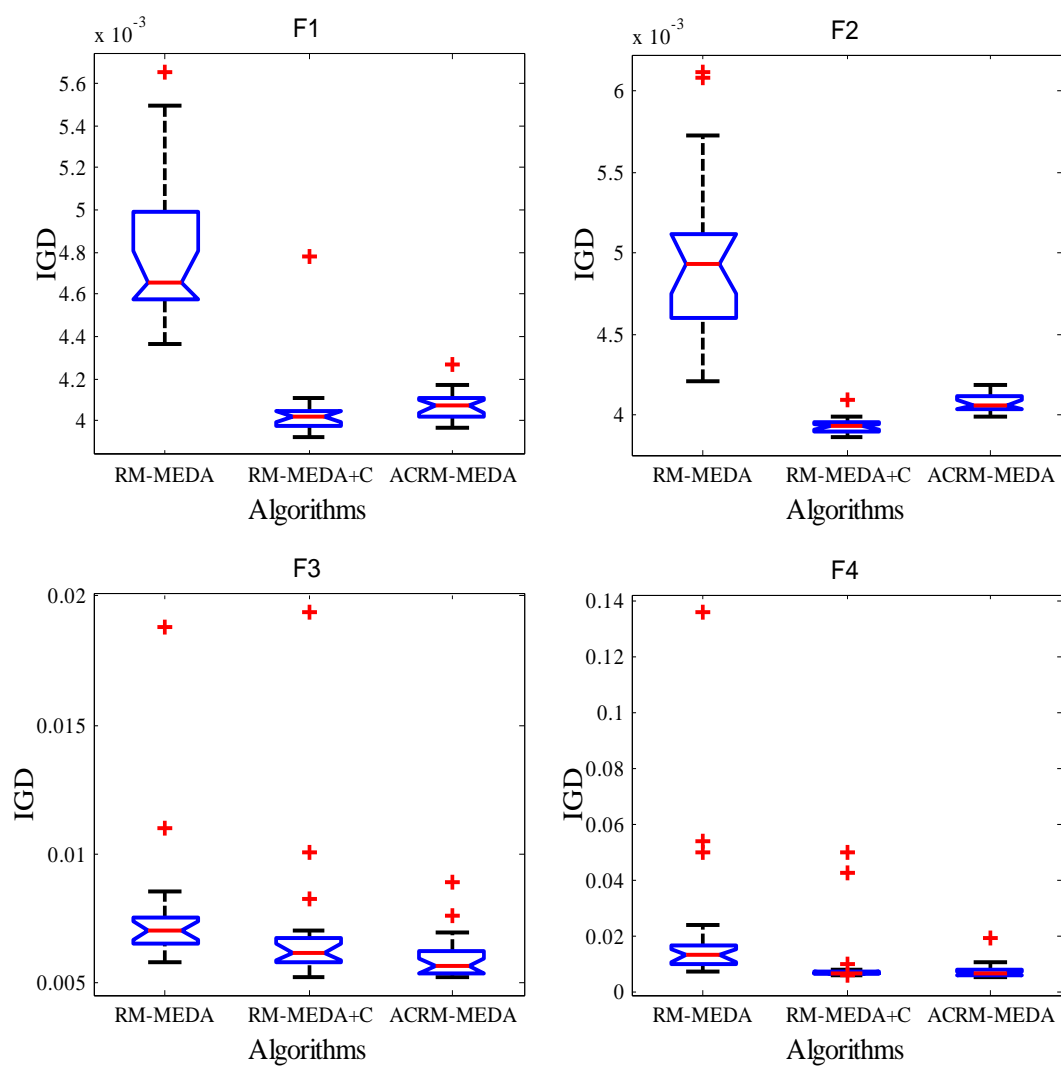


图 4.4 算法 ACRM-MEDA 求解六个函数的 Pareto 前端



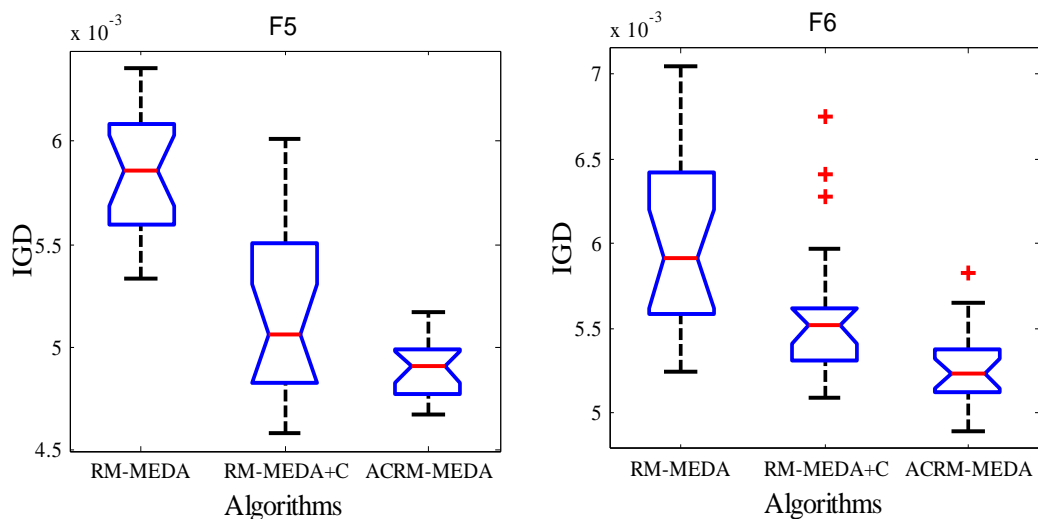
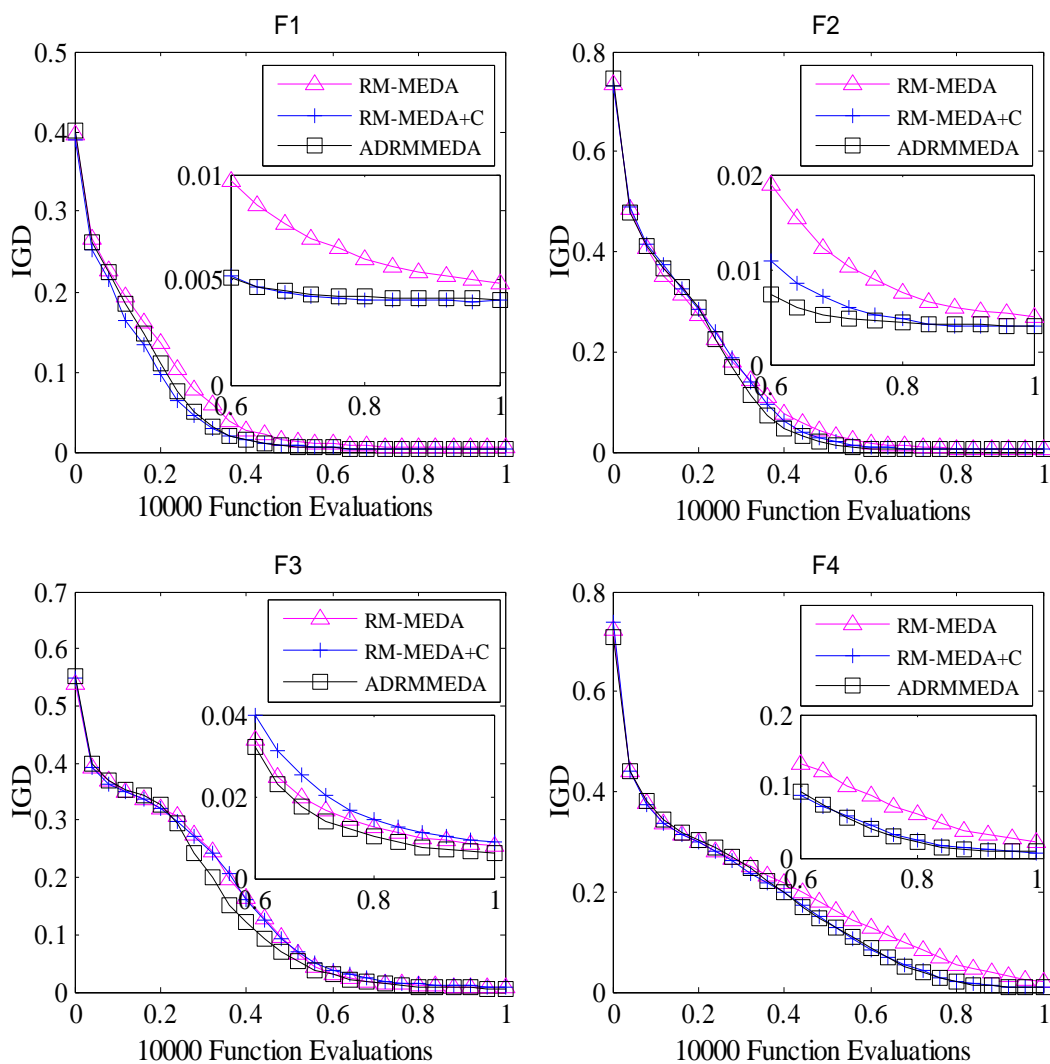


图 4.5 算法 RM-MEDA、RM-MEDA+C、ACRM-MEDA 求解 6 个测试函数的 IGD 指标的统计盒图



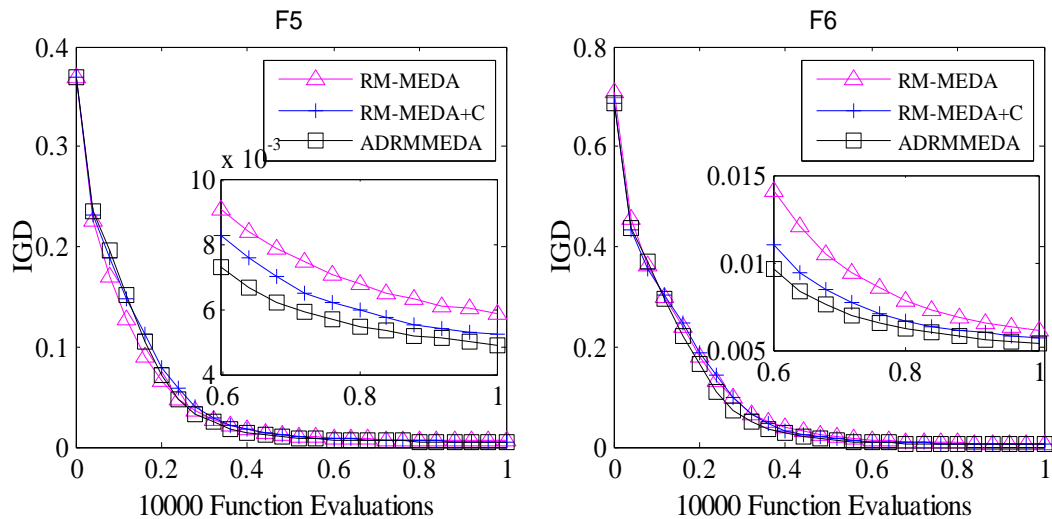


图 4.6 三种算法求解 6 个测试函数时，IGD 指标随函数评价次数增加的变化趋势

图 4.4 中可以看出，对于不同类型的 6 个测试函数，算法 ACRM-MEDA 均能收敛到 Pareto 前端，同时通过图 4.5 和图 4.6 可以看出，对于测试函数 PS 流形为一维线段的测试函数 F1，F2，算法 ACRM-MEDA 的收敛速度最快，RM-MEDA+C 次之，RM-MEDA 的收敛速最慢，从图 4.5 的盒图也可以看出 RM-MEDA+C 的稳定性和多样性最优。这是由于测试函数 F1，F2 的 PS 流形如图 4.3 中所示为一维线段。因此对于测试函数 F1，F2 理论上聚类数目的不同并不会导致错误的线性建模，从实验结果可以看出相对于较少的聚类数目更适合此类测试问题，因此本文中的合并分裂算子优势未能体现出来。对于测试函数流形较为复杂的测试函数 F3 和 F4，其 PS 流形为图 4.3 所示为一维 2 次曲线，理论上至少需要聚成两类才能保证模型的正确性，从实验结果来看 ACRM-MEDA 的性能与 RM-MEDA+C 的性能相当，说明分裂合并算子的中发挥了一定的作用，改善了分段线性模型的精度。对于测试函数 F5,F6 如图 4.3 所示其 PS 流形为一维正弦函数，如果采用合并策略可能出现聚类数目过少的情况,从而出现如图 4.2 所示聚类数目过少的情况，从图 4.5，4.6 的实验结果也可以看出 ACRM-MEDA 的性能优于 RM-MEDA+C，说明本文中设计的分裂算子在一定程度上处理了如图 4.2 所示的错误建模情况，从而建立了更为精确的分段线性模型来产生更为指导种群的进化。

4.6 本章小结

本章针对一个具有代表性的 EDA 算法 RM-MEDA 中存在的缺点，提出了一个改进的 RM-MEDA 算法 ACRM-MEDA，在 ACRM-MEDA 中引入了合并和分裂算子来根据种群的 PS 流形信息自适应的调整子种群的数量，从而可以通过调整后的

聚类子种群建立更为精确的分段线性模型来逼近 PS 流形,来进一步提高算法性能。仿真实验表明,对于本章中设置的 6 个不同测试问题,ACRM-MEDA 无论在收敛性和多样性方面均优于 RM-MEDA,同时对于 PS 流形为正弦曲线的复杂的测试问题上,相对于仅加入了合并算子的 RM-MEDA+C 算法,ACRM-MEDA 能够建立更为合理精确的分段线性模型来提高算法的性能。

第五章 总结与展望

5.1 论文工作总结

多目标优化算法现已成为进化计算领域是一个热点和难点之一，本文中详细介绍了进化多目标优化算法，分布式估计算法以及密母算法的研究现状，分析了这些算法的优势和有待改进之处。在本文中主要针对分现有的具有代表性的分布式估计多目标优化算法进行了改进。第三章算法 MAOS 中通过设定效率和能力两种指标动态调整密母算子中模型采样算子和基于交叉变异的克隆选择算子这两种算子采用的比例，从而使两者算子在进化的不同阶段充分发挥各自的优点，从而达到提高算法性能的目的。在第四章算法 ACRM-MEDA 中，我们设计了一种基于进化种群流形信息的合并分裂算子来自适应的调整算法中子种群的数目，从而使建立的线性模型更加精确的逼近进化种群的 PS 流形。

本文在第一章中首先介绍了多目标优化问题的发展现状、研究意义，随后介绍了分布式估计算法以及多目标优化算法，随后介绍了本文所涉及算法的研究动机，最后给出了本文的研究内容以及组织架构。

在本文第二章中，详细介绍了多目标问题的具体描述，然后重点阐述了基于免疫克隆选择原理的非支配邻域免疫算法 NNIA 以及基于规则模型的多目标优化算法 RM-MEDA，并给出了这两种算法的基本思想，算法流程以及优点和不足之处。

在本文的第三章中，首先介绍了基于 EDA 和人工免疫系统的混合多目标优化算法(HIAEDA)的基本思想，然后说明了算法的改进动机，接着提出了进化算子自适应选择的多目标优化算法(MAOS)。在 MAOS 中，利用进化过程中的精英个体信息定义能力和效率两个指标来自适应的调整密母算子中，模型采样算子和基于交叉变异的克隆选择算子这两种算子采用的比例，使两种算子在算法进化不同阶段来充分发挥各自的优点，从而平衡算法的开发和探索的能力。在仿真实验中，本文将算法 MAOS 和 RM-MEDA 以及 NSGAII 的性能做以比较，对于在文中所涉及三类测试问题，MAOS 在收敛性和多样性方面均表现出明显的优势。在本章最后中还对算法中效率和能力指标的权值参数做了相应的参数分析。

在本文第四章中，我们提出了一种聚类数目自适应调整的 RM-MEDA 算法(ACRM-MEDA)。在算法 ACRM-MEDA 中，我们设计了一种合并分裂算子来通过种群潜在 PS 流形信息来自适应的调整种群的聚类数目，通过建立更加精确的分段线性模型来产生新的子代。实验结果表明，由于在算法框架采用了合并分裂算子，对 RM-MEDA 算法在性能方面有了一定的改进，而且通过与 NSGAII 和 RM-MEDA

的对比,也证明了 ACRM-MEDA 的优良性能。

在本文第五章中,总结了本文的研究内容,并对给出了文章进一步需要研究的工作。

5.2 进一步工作展望

由于时间和本人工作能力等方面的限制,该论文中所涉及的研究还不够深入,本文所提出的相关算法尚处于初级阶段,仍需继续改善和提升。下面对本文中提出的两个算法有待进一步研究的内同作以说明。

对于第三章的算法 MAOS,本文中设计了能力和效率指标来自适应的调整两种算子在算法不同阶段的使用比例,如何设计更合理的自适应算子选择策略,也具有进一步研究的价值。

对第四章的 ACRM-MEDA 算法,算法中设计的合并和分裂算子,同时算法中采用局部 PCA 的方法进行聚类建立线性模型,但其算法时间复杂度较高,如何设计一种更为简单通用的建模方法描述当前种群分布也有待下一步研究。同时合并分裂策略中涉及到相关参数设置,由于时间关系未对本章中参数的敏感性做以分析,如何选取最优的参数也将是下一步工作研究的重点。

致谢

白驹过隙，日月如梭，转眼之间近三年的硕士学习生涯就要结束了。在这两年多期间，我获益良多，不管是在学术研究还是思想生活方面，都让我获得了重大的成长。近三年的认真严谨的学术生涯会对我以后的人生道路产生无尽的影响。在这里，导师们对待科研及生活的态度，以及同窗们的友善热情，都值得我好好学习和珍藏。

首先要衷心感谢我的导师刘芳教授，她在学术上精益求精，孜孜不倦的求真务实的态度让我深切的体会到作为导师这个角色的光荣和艰辛；她在思想上对学生的关心和启迪让我在以后的人生道路上少走弯路，不怕困难，勇往直前；她热情直率的生活态度也深深的感染着我，此时，我要向她表示深深的敬意和感谢，同时还要感谢焦李成教授，他每每在学术上高屋建瓴，深入浅出的话语，总是让我醍醐灌顶，焦老师鼓励我们要向前看，不要拘泥于眼前的困难和挫折，每当科研中遇到困难的时候，他的话语总是能让我重拾信心。

我还要特别感谢戚玉涛老师，戚老师专业知识渊博，学术功底深厚，在科研方面有着敏锐的眼光和独到的见解，本文在选题、研究、以及撰写过程中自始至终都得到了戚老师的悉心指导，在过去一年的周报告中，戚老师和我们一起共同探讨难题，总结分析，并提出新的见解和研究方向，他开阔的视野和敏锐的洞察力给了我深深的启迪，增强了我的科研攻关和解决问题的能力，再次表示衷心的感谢。

感谢智能所的公茂果老师，李阳阳老师，慕彩虹老师，刘若辰老师等，每一次的月报，您们总是给予了宝贵的建议和指导，您们辛苦了！每年智能所的学术之春，学术之秋，您们的报告开阔了我的思维，丰富了我的视野，让我倍受启发。感谢您们耐心指导我如何解决科研中遇到的问题，与您们的交流开拓了我的视野，拓宽了我的思路。再次深表感谢。同时在这里也要感谢师兄马晓亮博士。马师兄踏实认真，热心助人，每每遇到科研问题时，他总是能及时地给予帮助。

感谢与我同窗三年的实验室各位同门，他们是：董航、周确、王杰、安洁、崔白洋、袁嘉琳、赵静、瞿秋蕙、郜国栋，蒋文梅，刘美云，常伟远，马红梅，石俊飞，单雁冰，杨丽，王伟伟，朱亚萍、杨鸽等等；在生活和学习上，你们给予了我很多帮助，怀念和你们在一起的日子，怀念遇到困难时候，我们一起仔细研究解决问题的日子，怀念和你们在一起举杯畅饮的日子，这些都将会是我人生中最美好的回忆。衷心祝愿你们前程似锦。

感谢父母的养育之恩，感谢他们对我学业道路的支持和无私的付出，感谢我的弟弟，他总会在我灰心失意的时候鼓励我，开导我，让我重拾信心和勇气。感

谢你们多年来的支持和鼓励，正是因为你们坚定的支持和无私的奉献才使我顺利地完成了学业，你们对我的恩情永远使我上进的动力，我将铭记于心。

最后感谢参加论文评审和答辩的各位老师对本文的批评指正。

参考文献

- [1] K Deb. Multi-objective optimization using evolutionary algorithms [M]. New York: Wiley, 2001.51-71.
- [2] Schaffer J D. Multiple objective optimization with vector evaluated genetic algorithms. Proceedings of the 1st International Conference on Genetic Algorithms. Lawrence Erlbaum, 1985, 93-100.
- [3] Fonseca CM, Fleming PJ. Genetic algorithm for multi-objective optimization: Formulation, discussion and generation. Forrest S, ed. Proc. of the 5th Int'l Conf. on Genetic Algorithms. San Mateo: Morgan Kauffman Publishers, 1993.416-423.
- [4] Srinivas N, Deb K. Multi-objective optimization using non-dominated sorting in genetic algorithms. Evolutionary Computation. 1994, 2(3).221-248.
- [5] J. Horn, N. Nafpliotis, and D. E. Goldberg. A niched pareto genetic algorithm for multi-objective optimization. International Conference on Evolutionary Computation. 1994, 82-87.
- [6] Zitzler E, Thiele L. Multi-Objective evolutionary algorithms: A comparative case study and the strength Pareto approach. IEEE Trans on Evolutionary Computation. 1999, 3(4).257-271.
- [7] E. Zitzler, M. Laumanns, and L. Thiele, SPEA2: improving the strength pareto evolutionary algorithm for multi-objective optimization, Evolutionary Methods for Design, Optimization and Control. Barcelona, Spain: CIMNE, 2002.95-100.
- [8] Knowles JD, Corne DW. Approximating the non-dominated front using the Pareto archived evolution strategy. Evolutionary Computation, 2000, 8(2).149-172.
- [9] Corne DW, Knowles JD, Oates MJ. The Pareto-envelope based selection algorithm for multi-objective optimization. Parallel Problem Solving from Nature, PPSN VI. LNCS, Berlin: Springer-Verlag, 2000.839-848.
- [10] Corne DW, Jerram NR, Knowles JD, Oates MJ. PESA-II: Region-Based selection in evolutionary multi-objective optimization. Proceedings of the Genetic and Evolutionary Computation Conference .San Francisco: Morgan Kaufmann Publishers, 2001. 283-290.
- [11] Erickson M, Mayer A, Horn J. The niched Pareto genetic algorithm 2 applied to the design of groundwater remediation system. Proc. of the 1st Int'l Conf. on Evolutionary Multi-Criterion Optimization, EMO 2001. Berlin: Springer-Verlag, 2001. 681-695.

- [12] Coello Coello CA, Pulido GT. A micro-genetic algorithm for multi-objective optimization. Proc. of the Genetic and Evolutionary Computation Conf, GECCO 2001. San Francisco: Morgan Kaufmann Publishers, 2001.274–282.
- [13] K. Deb, A fast and elitist multi-objective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation. 2002,6(2).182–197.
- [14] Laumanns M, Thiele L, Deb K, Zitzler E. Combining convergence and diversity in evolutionary multi-objective optimization. Evolutionary Computation. 2001 ,10(3). 263-282.
- [15] Brockoff D, Zitzler E. Are all objective necessary on dimensionality reduction in evolutionary multi-objective optimization.Parallel Problem Solving from Nature, PPSN IX. LNCS.Berlin:Springer-Verlag,2006.533–542.
- [16] Hernández-D áz AG, Santana-Quintero LV, Coello Coello CA, Molina J. Pareto-Adaptive ϵ -dominance. Evolutionary Computation,2007,15(4).493–517.
- [17] Coello Coello CA, Pulido GT, Lechuga MS. Handing multiple objectives with particle swarm optimization. IEEE Trans. on Evolutionary Compuation.2004,8(3). 256-279.
- [18] Gong MG, Jiao LC, Du HF, Bo LF. Multi-objective immune algorithm with non-dominated neighbor-based selection. Evolutionary Computation.2008,16(2). 225–255.
- [19] Zhou AM, Zhang QF, Jin Y, Sendhoff B, Tsang E. Global multi-objective optimization via estimation of distribution algorithm with biased initialization and crossover. Proc. of the Genetic and Evolutionary Computation Conf. GECCO 2007. New York: ACM Press,2007.617–622.
- [20] Zhang QF, Zhou AM, Jin Y. RM-MEDA: A regularity model based multiobjective estimation of distribution algorithm. IEEE Trans. on Evolutionary Computation. 2007,12(1).41–63.
- [21] Zhang Q F, Li F H. A multi-objective evolutionary algorithm based on decomposition: MOEA/D. IEEE Transactions on Evolutionary Computation 11 (2007) 712-731.
- [22] Baluja S. Population-Based Incremental Learning: A method for Integrating Genetic Search Based Function Optimization and Competitive Learning. CMU-CS-94-163,Avariable via.Anonymous ftp atreports.adm.cs.cmu.edu,1994 Technical Report,Carnegie Mellon University(1994).
- [23] D.Thierens and P.A.N.Bosman,Muti-objective Mixture-based Iterated Density Estimation Evolutionary Algorithms , in Proeedings of the Genetic and

- Evolutionary Computation Conference. San Francisco, California: Morgan Kaufmann, 2001, pp. 663~670
- [24] Costa, M. and Minisei, E., MOPED: A Multi-objective Parzen-based Estimation Distribution Algorithm for Continuous Problems, In Fonseca C.M. et al., editors Proceedings of the Second International Conference on Evolutionary Multi-Criterion Optimization, Pages 282~294.
- [25] A Zhou, Q Zhang, Y Jin. Approximating the set of pareto-optimal solutions in both the decision and objective spaces by an estimation of distribution algorithm [J]. IEEE Transactions on Evolutionary Computation, 2009, 13(5): 1167-1189.
- [26] Nazan Khan, David E Goldberg, Martin Pelikan. Multi-objective Bayesian Optimization Algorithm [C]. In Genetic and Evolutionary Computation Conference. Pages: 684-689.
- [27] Hart WE, Krasnogor N, Smith JE Recent advances in memetic algorithms. Springer, Berlin. 2005.
- [28] J. Knowles and D. Corne. Memetic Algorithms for Multi-objective Optimization: Issues, Methods and Prospects. In William E. Hart, N. Krasnogor, and J.E. Smith, editors, Recent Advances in Memetic Algorithms, Springer. Studies in Fuzziness and Soft Computing, 2005.
- [29] H. Ishibuchi and T. Murata. Multi-Objective Genetic Local Search Algorithm. In Toshio Fukuda and Takeshi Furuhashi, Proceedings of the International Conference on Evolutionary Computation, 1996.
- [30] H. Ishibuchi and T. Murata. A multi-objective genetic local search algorithm and its application to flowshop scheduling. IEEE Transactions on Systems, Man, and Cybernetics - Part C: Applications and Reviews, 1998, 28(3): 392~403.
- [31] Jaszkiewicz A. Genetic local search for multiple objective combinatorial optimization, European Journal of Operational Research, 2002.
- [32] A. Jaszkiewicz. Do Multiple-Objective Metaheuristics Deliver on Their Promises Computational Experiment on the Set-Covering Problem. IEEE Transactions on Evolutionary Computation, 2003, 7(2): 133~143.
- [33] J. Knowles and D. Corne. M-PAES: A Memetic Algorithm for Multiobjective Optimization. In 2000 Congress on Evolutionary Computation, 2000, volume 1, 325~332
- [34] T. Murata, S. Kaige, and H. Ishibuchi. Generalization of Dominance Relation-Based Replacement Rules for Memetic EMO Algorithms. Springer. Lecture Notes in Computer Science, 2003: 1234~1245.

- [35] J.M. Pe ña, V. Robles, P. Larra ñaga, V. Herves, F. Rosales, and M.S. Pérez
GA-EDA: Hybrid Evolutionary Algorithm Using Genetic and Estimation of
Distribution Algorithms. *Innovations In Applied Artificial Intelligence Lecture
Notes In Computer Science*.2004,3029(1).361-371.
- [36] Aimin Zhou, Qingfu Zhang,Yaochu Jin, Edward Tsang, and Tatsuya Okabe. A
model-based evolutionary algorithm for bi-objective optimization. *The 2005 IEEE
Congress on Evolutionary Computation*.2005,2568-2575.
- [37] Aimin Zhou, Yaochu Jin, Qingfu Zhang, Bernhard Sendhoff,Edward Tsang.
Combining Model-based and Genetics-based Offspring Generation for
Multi-objective Optimization Using a Convergence Criterion.*In Congress on
Evolutionary Computation*.2006,7, 892-899.
- [38]刘静乐. 基于 EDA 和人工免疫系统的混合多目标优化算法.西安电子科技大学
硕士论文, 2012.2.
- [39] 樊建聪,梁永全,分布估计学习算法研究.山东科技大学博士论文.2010,6,3-4.
- [40] S. Kukkonen and J. Lampinen,.GDE3: The third evolution step of gen-eralized
differential evolution. *Proc. Congr. Evol. Comput. (CEC2005)*, Edinburgh, U.K.,
2005, pp. 443–450.
- [41]K. Deb, A. Sinha, and S. Kukkonen.Multi-objective test problems, linkages, and
evolutionary methodologies.*In Proc. Genetic Evol. Comput. Conf. (GECCO 2006)*,
Seattle, Washington, 2006, vol. 2. 1141–1148.
- [42] P. A. N. Bosman and D. Thierens.The naive MIDEA: A baseline multi- objective
EA, *Proc. 3rd Int. Conf. Evol. Multi-Criterion Op-timization (EMO 2005)*,
Guanajuato, Mexico.2005, vol. 3410, LNCS.428–442.
- [43] N. Kambhatla and T. K. Leen.Dimension reduction by local principal component
Analysis. *Neural Comput*.1997,9(7).1493–1516.
- [44]K Deb and R B Agrawal. Simulated binary crossover for continuous search space
[J]. *Complex Systems*, 1995, 9(2).115–148.
- [45] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. da Fonseca.
Performance assessment of multi-objective optimizers: An analysis and review.
IEEE Trans. Evol.Comput.2003,4,7(2).117–132.
- [46] JD D Yang,L C Jiao,M G Gong,J Feng. Adaptive ranks and K-Nearest neighbor list
based multiobjective immune algorithm [J]. *Computational Intelligence*.2010,
26(4).359–385.
- [47] Q. Li, On the angle between two dimensional planes in the n-dimensional
Euclidean space, *Journal of Mathematics* 6(4)(1986) ,407–410 (in Chinese

with English abstract).

- [48] H. Lu, High Dimension Euclidean Geometry, <http://www.gwjhx.com/>.
- [49] K. Li, A. Fialho, S. Kwong, and Q. Zhang, Adaptive Operator Selection with Bandits for Decomposition based Multi-Objective Optimization, IEEE Trans on Evolutionary Computation, 2012, accepted.
- [50] W. Gong, A. Fialho, and Z. Cai. Adaptive strategy selection in differential evolution. GECCO'10: Proc. 2010 Annual Conference on Genetic and Evolutionary Computation, J. Branke et al., Ed. ACM Press, Jul. 2010, pp. 409–416.
- [51] J. Whitacre, T. Pham, and R. Sarker, Use of statistical outlier detection method in adaptive evolutionary algorithms. GECCO'06: Proc. 2006 Annual Conference on Genetic and Evolutionary Computation, ACMpress. 2006, pp. 1345–1352.
- [52] A. Fialho, R. Ros, M. Schoenauer, and M. Sebag, Comparison-based adaptive strategy selection in differential evolution. PPSN XI: Proc. 11th International Conference on Parallel Problem Solving from Nature, R. Schaefer et al., Ed. Springer. 2010, pp. 194–203.
- [53] J. Maturana, F. Lardeux, and F. Saubion. Autonomous operator management for evolutionary algorithms. Journal of Heuristics. 2010, vol. 16, pp. 881–909.
- [54] D. Thierens. An adaptive pursuit strategy for allocating operator probabilities. GECCO'05: Proc. 2005 Annual Conference on Genetic and Evolutionary Computation, H. G. Beyer, Ed. ACM press. 2005, pp. 1539–1546.
- [55] A. Fialho. Adaptive operator selection for optimization,. Ph.D. dissertation, Universit ´e Paris-Sud XI, Orsay, France, December 2010.
- [56] Yong Wang, Jian Xiang, Zixing Cai. A regularity model-based multiobjective estimation of distribution algorithm with reducing redundant cluster operator. Appl. Soft Comput. 2012, 12(11): 3526–3538

研究成果

在硕士期间所取得研究成果如下：

1. 戚玉涛,刘芳,刘静乐, **任元**等.基于免疫算法和 EDA 的混合多目标优化算法.软件学报（已录用）.
2. 戚玉涛，刘芳， **任元**，刘静乐等. 基于合作模型的协同免疫多目标优化算法.电子学报.（已投稿）
- 3.戚玉涛，刘芳， **任元**等 . 自适应算子选择密母多目标优化算法.(已完成)

