

分类号：

单位代码： 10140

密 级：公开

学 号： 4031431968

遼寧大學

硕士学位论文

中文题目： 带精英策略的非支配排序遗传算法优化研究

Research on Optimization of Elitist Non-dominated

英文题目： Sorting Genetic Algorithm

论文作者： 郭 军

指导教师： 冯勇 教授

专 业： 计算机应用技术

完成时间： 二〇一七年四月

申请辽宁大学硕士学位论文

带精英策略的非支配排序遗传算法优化研究

Research on Optimization of Elitist Non-dominated
Sorting Genetic Algorithm

作 者：_____郭 军

指导教师：_____冯勇 教授

专 业：_____计算机应用技术_____

答辩日期：_____2017年 5月 25 日_____

二〇一七年五月 · 中国辽宁

辽宁大学学位论文原创性声明

本人郑重声明：所呈交的学位论文是本人在导师的指导下独立完成的。论文中取得的研究成果除加以标注的内容外，不包含其他个人或集体已经发表或撰写过的研究成果，不包含本人为获得其他学位而使用过的成果。对本文的研究做出重要贡献的个人和集体均已在文中进行了标注，并表示谢意。本人完全意识到本声明的法律结果由本人承担。

学位论文作者签名: 年 月 日

学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交学位论文的原件、复印件和电子版，允许学位论文被查阅和借阅。本人授权辽宁大学可以将本学位论文的全部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和汇编学位论文。同时授权中国学术期刊（光盘版）电子杂志社将本学位论文收录到《中国

博士学位论文全文数据库》和《中国优秀硕士学位论文全文数据库》并通过网络向社会公众提供信息服务。学校须按照授权对学位论文进行管理，不得超越授权对学位论文进行任意处理。

保密（ ），在 年后解密适用本授权书。（保密：请在括号内划“√”）

授权人签名：

指导教师签名：

日期： 年 月 日 日期： 年 月 日

辽宁大学学位论文原创性声明

本人郑重声明：所呈交的学位论文是本人在导师的指导下独立完成的。论文中取得的研究成果除加以标注的内容外，不包含其他个人或集体已经发表或撰写过的研究成果，不包含本人为获得其他学位而使用过的成果。对本文的研究做出重要贡献的个人和集体均已在文中进行了标注，并表示谢意。本人完全意识到本声明的法律结果由本人承担。

学位论文作者签名: 年 月 日

学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交学位论文的原件、复印件和电子版，允许学位论文被查阅和借阅。本人授权辽宁大学可以将本学位论文的全部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和汇编学位论文。同时授权中国学术期刊（光盘版）电子杂志社将本学位论文收录到《中国

博士学位论文全文数据库》和《中国优秀硕士学位论文全文数据库》并通过网络向社会公众提供信息服务。学校须按照授权对学位论文进行管理，不得超越授权对学位论文进行任意处理。

保密（ ），在 年后解密适用本授权书。（保密：请在括号内划“√”）

授权人签名：

指导教师签名：

日期： 年 月 日 日期： 年 月 日

摘要

在实际工程及科研领域中有大量需要同时对不止一个目标进行优化求解的问题，此类问题通常被称为多目标优化问题（**Multi-objective optimization problems**, **Mops**）。多目标优化问题的最优解往往并不唯一，并且多个优化目标之间一般存在相互冲突关系，在对某一目标优化的同时往往会使其他目标劣化，使得各目标难以同时达到各自的最优值，因此该类问题的求解就是要得出一组在多目标间优化性能均衡的解。实现对多目标优化问题的求解具有重要的现实意义和理论意义，该问题的求解已得到工程界和学术界的广泛关注。

带精英策略的非支配排序遗传算法（**Elitist Non-dominated Sorting Genetic Algorithm**, **NSGA-II**）是目前最流行的多目标优化算法之一，它具有较高的求解效率，可在一次运行过程中得出多个高质量的解，成为了其它多目标优化算法进行性能对比的基准算法。随着**NSGA-II**应用的推广深入，一些问题也显现出来，首先**NSGA-II**在整个运行过程中运行参数保持不变，不能根据环境变化对其自适应调整，难以对解空间实现高效搜索；其次**NSGA-II**具有较强的全局搜索能力但是局部搜索能力较弱，从而导致算法收敛性能差，无法找到全局最优解集或需要较长时间才能找到全局最优解集。

针对**NSGA-II**现存问题，本文从以下两个方面进行具体研究：

第一方面，本文针对**NSGA-II**不能根据环境自适应调整运行参数的问题，给出了一种基于非支配排序分层的适应性策略。本文所提策略根据算法的运行阶段、运行代数及当前临时种群非支配个体数动态调整进化个体的运行参数，使算法具备运行自适应性，提高算法全局搜索能力，从而起到抑制早熟、加快收敛速度、增加种群多样性的作用。

第二方面，本文针对**NSGA-II**局部优化能力较弱的问题，给出了一种基于模拟退火算法的**NSGA-II**优化改进。本文所提改进首先对传统模拟退火算法进行调整，即以支配关系作为个体优劣的评价标准，在搜索过程中只接受优于原始解的新解，并根据当前进化代所对应的退火温度对搜索范围进行动态调节；然后将调整后的模拟退火算法引入到**NSGA-II**中，形成综合算法，即加入局部搜索的非支配排序遗传算法，通过对每一代进化生成的父代及子代种群个体进行退火搜索，从而起到提高算法局部优化能力的作用；最后将加入局部搜索的非支配排序遗传算法与第一阶段所给出的适应性策略相结合，给出了一种加入局部搜索的自适应非支配排序遗传算法，通过提高算法进化适应性及局部优化能力以提升原算法求解全局最优解的能力，从而起到抑制早熟、加快收敛速度、增加种群多样性的作用，实现对**NSGA-II**的优化改进。

本文通过模拟仿真实验对所提改进进行验证，实验分为三部分：第一部分

通过实验对比表明本文所提基于非支配排序分层的适应性策略能有效提高原算法的全局搜索能力；第二部分通过实验对比表明本文所提加入局部搜索的非支配排序遗传算法能有效提高原算法的局部搜索能力；第三部分通过实验对融合本文所提基于非支配排序分层的适应性策略和加入局部搜索的非支配排序遗传算法的综合算法，即加入局部搜索的自适应非支配排序遗传算法进行性能验证，表明本文对 **NSGA-II** 进行的两点优化能有效结合，在收敛性及多样性两方面对算法的性能确有提升，从而证明了本文对 **NSGA-II** 优化研究工作的有效性。

关键词： **NSGA-II**，适应性策略，全局搜索，模拟退火算法，局部搜索

ABSTRACT

In the field of practical engineering and research, there are large numbers of problems which need to be optimized for more than one goal simultaneously, such problems are usually called multi-objective optimization problems. The optimal solution of multi-objective optimization problems is usually not unique, and there are usually conflicting relationships between multiple optimization goals, the optimization of a goal usually deteriorates the other goals, it is difficult for each goal to achieve its own optimal value at the same time, therefore, the solution of this kind of problem is to achieve a set of solutions which have balanced optimization performance among multiple objectives. It has important practical and theoretical significance to solve the multi-objective optimization problem. The solution of multi-objective optimization problems has received extensive attention of engineering and academia.

The non-dominated sorting genetic algorithm with elitist strategy is one of the most popular multi-objective optimization algorithms, it has higher solving efficiency, can produce a number of high quality solutions in a single run, so it has become a benchmark algorithm for performance comparison of the other multi-objective optimization algorithms. With the popularization and application of NSGA-II, some problems have emerged. First of all, the running parameters of NSGA-II remain unchanged during the whole operation, it can not be adjusted adaptively according to the change of environment, so it is difficult to search efficiently for the solution space. Secondly, NSGA-II has a strong global search capability, but its local search ability is weak, which leads to the poor convergence performance of the algorithm, and it can not find the global optimal solution set, or it takes a long time to find the global optimal solution set.

In view of the existing problems of NSGA-II, this paper makes a concrete research from the following two aspects:

In the first aspect, an adaptive strategy based on non-dominated sorting layer is proposed in this paper to solve the problem that NSGA-II can not adjust the running parameters adaptively according to the environment. The strategy proposed in this paper dynamically adjusts the running parameters of evolutionary individuals according to the running phase of the algorithm, the evolution time, and the number of non-dominated individuals in the current temporary population. The strategy proposed in this paper can improve the self-adaptive abilities and the global search ability of the

original algorithm, thus suppress premature, accelerate convergence speed and increase population diversity.

In the second aspect, an optimization improvement of NSGA-II which is based on simulated annealing algorithm is proposed to solve the problem that weak local optimization ability of NSGA-II. The improvement proposed in this paper first adjusts the traditional simulated annealing algorithm, that is, the dominance relation is used as the evaluation criterion of individual, only a new solution that is superior to the original solution is accepted in the search process, and the search range is dynamically adjusted according to the annealing temperature corresponding to the current evolution. Then, the simulated annealing algorithm is introduced into NSGA-II to form a synthesis algorithm, that is, an adaptive non-dominated sorting genetic algorithm with local search, each generation of parent and offspring populations are searched by a simulated annealing algorithm in synthesis algorithm. Finally, the non-dominated sorting genetic algorithm with local search is combined with the adaptive strategy given in the first stage, that is, an adaptive non-dominated sorting genetic algorithm with local search is proposed. The proposed algorithm improves the evolutionary adaptability and local optimization ability of the algorithm, so that the ability of the original algorithm to solve the global optimal solution is improved, thus, it can suppress premature, accelerate convergence speed and increase population diversity, realize the optimization and improvement of NSGA-II.

Analog simulation experiment is performed to validate the improvements in this paper. The experiment is consisted of three parts: The first part of the experiment shows that the adaptive strategy based on non-dominated sorting layer which is proposed in this paper can effectively improve the global search ability of the original algorithm. The second part of the experiment shows that the non-dominated sorting genetic algorithms proposed in this paper can effectively improve the local search ability of the original algorithm. The third part of the experiment shows that the adaptive non-dominated sorting genetic algorithm with local search which integrated the adaptive strategy based on non-dominated sorting layer and the non-dominated sorting genetic algorithms proposed can improve the ability of the original algorithm, it shows that the two points optimization of NSGA-II can be effectively combined, in the two aspects of convergence and diversity indicate the performance of the algorithm which proposed in this paper can improve the ability of the original algorithm. The validity of the research work on NSGA-II optimization in this paper is proved.

Key Words: NSGA-II, Adaptive strategy, Global search, Simulated annealing algorithm, Local search

目 录

第 1 章 绪论	1
1.1 研究背景.....	1
1.2 国内外研究现状.....	2
1.2.1 国外研究现状	2
1.2.2 国内研究现状	3
1.3 研究目的和意义.....	4
1.3.1 研究目的	4
1.3.2 研究意义	5
1.4 主要研究内容.....	6
1.5 论文的组织结构.....	7
第 2 章 文献综述	9
2.1 多目标优化问题.....	9
2.1.1 多目标优化问题的数学描述	9
2.1.2 多目标优化中的相关定义	9
2.1.3 多目标问题的传统解法	11
2.2 多目标优化算法.....	13
2.2.1 多目标优化算法的基本框架	13
2.2.2 多目标优化算法的常用策略	14
2.2.3 多目标优化算法的性能指标	17
2.3 带精英策略的非支配排序遗传算法.....	20
2.3.1 NSGA-II 的关键技术	20
2.3.2 NSGA-II 的基本流程	23
2.4 本章小结.....	24
第 3 章 基于非支配排序分层的适应性策略	26

3.1 传统的适应性策略.....	26
3.1.1 基于进化阶段的适应性策略	26
3.1.2 基于个体的适应性策略	27
3.1.3 传统适应性策略的局限性分析	30
3.2 基于非支配排序分层的适应性策略.....	31
3.2.1 交叉率及变异率调整模型	31
3.2.2 交叉分布指数及变异分布指数调整模型	34
3.3 本章小结.....	36
第 4 章 加入局部搜索的自适应非支配排序遗传算法	37
4.1 局部搜索算法与传统遗传算法的组合.....	37
4.1.1 爬山算法	37
4.1.2 模拟退火算法	40
4.1.3 加入局部搜索的遗传算法	43
4.2 加入局部搜索的多目标优化算法.....	45
4.3 加入局部搜索的非支配排序遗传算法.....	47
4.3.1 对模拟退火算法的调整	47
4.3.2 加入局部搜索的非支配排序遗传算法.....	48
4.4 加入局部搜索的自适应非支配排序遗传算法.....	49
4.5 本章小结.....	51
第 5 章 实验与分析	53
5.1 实验准备.....	53
5.2 模拟仿真测试函数.....	53
5.3 性能对比实验.....	56
5.3.1 基于非支配排序分层的适应性策略性能对比实验	56
5.3.2 加入局部搜索的非支配排序遗传算法性能对比实验	61
5.3.3 加入局部搜索的自适应非支配排序遗传算法性能对比实验	65
5.4 本章小结.....	68
第 6 章 总结与展望	69

6.1 总结.....	69
6.2 未来工作展望.....	70
致 谢.....	71
参考文献	72
攻读学位期间发表的学术论文以及参加科研情况	75

图表目录

图目录

图 2-1 两个目标的帕累托支配关系示意图.....	10
图 2-2 两个目标的帕累托前沿面分布示意图.....	11
图 2-3 多目标优化算法基本框架流程图.....	14
图 2-4 两目标最小化问题的NDS方法分层示意图.....	15
图 2-5 精英保留机制的两种实现方法.....	17
图 2-6 收敛性指标 γ 的示意图.....	18
图 2-7 多样性准则 Δ 的示意图.....	20
图 2-8 NSGA-II 精英选择过程示意图.....	22
图 2-9 拥挤距离示意图.....	23
图 2-10 NSGA-II 流程图	24
图 3-1 自适应交叉率与变异率变化趋势图.....	28
图 3-2 改进型自适应交叉率与变异率变化趋势图.....	29
图 3-3 NSGA-II 新种群产生的过程示意图.....	32
图 4-1 爬山算法的示意图.....	38
图 4-2 单一个体爬山搜索的流程图.....	40

图表目录

图 4-3 单一个体模拟退火搜索的流程图.....	43
图 4-4 加入局部搜索的遗传算法流程图.....	45
图 4-5 加入局部搜索的非支配排序遗传算法流程图.....	49
图 4-6 加入局部搜索的自适应非支配排序遗传算法流程图	51
图 5-1 ZDT1-ZDT4、 ZDT6 Pareto 前沿面示意图.....	55
图 5-2 ZDT1-ZDT4临时种群非支配个体数变化趋势图.....	57
图 5-3 ZDT6 临时种群非支配个体数变化趋势图.....	58
图 5-4 ZDT1 、 ZDT2 临时种群非支配个体数变化趋势图.....	61
图 5-5 ZDT3 、 ZDT4 、 ZDT6 临时种群非支配个体数变化趋势图.....	62
图 5-6 ZDT1 、 ZDT2 临时种群非支配个体数变化趋势图.....	65
图 5-7 ZDT3 、 ZDT4 、 ZDT6 临时种群非支配个体数变化趋势图.....	66

表目录

表 5-1 ZDT测试函数集.....	54
表 5-2 收敛性准则 γ	59
表 5-3 多样性准则 Δ	60
表 5-4 收敛性准则 γ	63
表 5-5 多样性准则 Δ	64
表 5-6 收敛性准则 γ	67

图表目录

表 5-7 多样性准则 Δ	68
----------------------------	----

第 1 章 绪论

1.1 研究背景

在日常生产和生活实践中很多事情都是相互矛盾、彼此间存在制约关系的,在对某件事情寻求更优的同时往往会损害其他方面的利益,如何使多方面的事情同时达到令人满意的优化效果是人们所需要解决的问题,这一类问题就是典型的多目标优化问题。多目标优化问题的求解就是要得到在多个优化目标间性能均衡的解,如果只一味追求单个目标的优化往往会造成其它目标的劣化,使整体优化效果下降。在生产实践中对利益的追求和对环境的保护就是这样的例子:如果只追求生产利益的最大化而忽略对环境的保护往往会造成严重的环境问题,对人们的健康生存构成威胁。这样的例子在生产实践中比比皆是,如果忽略了事物间的关联性,大力追求单一事物最优化的同时没有考虑对其他事物的影响,往往会造成难以想象的后果。多目标优化问题在日常场景中也广泛存在,比如:出行旅游时,旅游者希望以较少的开销获得更有价值的旅行经历;在安排公交车发车时刻时,公交公司希望在乘客获得较好乘车体验的同时公司自身可以获得更好的经济收益;在购物时,购物者希望以较少的经济支出而购买到具有较大效用的商品。由此可见,多目标优化问题与人们日常生活息息相关,对其进行求解具有十分重要的现实意义。

传统多目标优化解法往往采用将多目标优化问题转化为单目标优化问题的方式对其进行求解^[1],但该种方式存在较大局限性,为进一步提升多目标优化问题的求解效率,众多学者对其提出了新的解决方案,其中以进化算法为主的解决方案成为当今该领域的主要研究方向。进化算法通过对自然界中生物进化过程的模拟来实现对问题的求解,是生命科学与计算科学相互融合和相互促进的一个具体体现。进化算法采用群体计算和随机搜索的方式,通过维持代与代之间的潜在解,达到逐渐逼近问题最优解的目的,该类算法不需要预先了解问题的背景知识,也不对所求问题的数学特性做任何限制,一次运算能够得出多个最终解,具有较高搜索效率,对多目标优化问题具有较好适应性。

1.2 国内外研究现状

1.2.1 国外研究现状

国外的多目标优化研究出现较早，其研究发展过程以是否引入进化算法及 Pareto 最优理论划分为三部分，第一部分是利用数学方法进行求解的传统方法；第二部分是以进化算法的引入为标志；第三部分是以 Pareto 最优理论的引入为标志。传统方法具有较多不足之处，比如：对所求解问题有过多限制、在运行之前需要人为根据问题的背景知识预设参数、一次运算只能得到一个最优解、难以对所得解进行优劣程度判断等。多目标进化算法不对问题的数学特性做任何要求，也不需要问题本身背景知识有任何了解，理论上可以处理所有类型的问题，每次执行都可以得到多个最终解^[2]，能够有效解决传统方法中的不足^[3-7]。

第一个被学术界公认的多目标进化算法是1985年 Schaffer 所提出的向量评价遗传算法（Vector-evaluated Genetic Algorithms，VEGA）^[8]。VEGA 未直接使用 Pareto 占优的概念进行个体的选择，是一种非 Pareto 优化算法，在 VEGA 中个体选择采用成比例的选择机制，各目标函数均采用独立的子群体对其进行搜索寻优，子群体执行一次搜索后需要合并成一个新种群，并从新种群中为各目标函数选择出新一代子种群，如此循环执行直到得出问题最优解。Fourman 在 VEGA 基础之上进行了一定改进^[9]，改进算法在每次进化操作中均采用锦标赛的方式从多目标函数中随机选择一个子目标作为该次进化的目标函数，并采用单目标优化问题的处理方法对种群执行进化操作。在 Fourman 的研究基础之上Kurasawe提出了改进算法，在改进算法中目标函数的选择可以是随机的，也可以是被指定的，也可以是在运行过程中对两种方式进行动态选择的^[10]。以上算法并不是完全基于 Pareto 理论的，只能获得小部分 Pareto 最优解，而且无法很好地保持群体多样性^[11]。最早提出将 Pareto 最优理论引入到进化算法中的人是 Goldberg，他提出将 Pareto 排序思想引入到遗传算法中，并以此提高算法的求解效率^[12]。虽然 Goldberg 并没有对所提想法进行实现，但是他的这些思路为后来的多目标优化研究指明了方向。Fonseca 和 Fleming 于1993年以 Pareto 排序为基础提出了多目标遗传算法（Multi-objective Genetic Algorithm，MOGA）^[13]，此后基于 Pareto 理论的多目标优化方法逐渐盛行起来，多目标优化问题的研究得到了快速发展。

1.2.2 国内研究现状

国内关于多目标优化问题的研究开始较晚，但也在理论研究和实际应用领

域获得了较大的成绩。比如：尚荣华等人通过实验表明 **NSGA-II** 对于多目标聚类学习和分类学习框架 (**MSCC**) 的优化效果优于多目标粒子群算法^[14]；刘爱军等人提出改进的 **NSGA-II**，即对拥挤密度排序法和精英策略进行改进并提出自适应交叉和变异策略^[15]；刘江首先针对 **NSGA-II** 对种群多样性考虑不足的问题提出了新的选择算子和变异算子，然后对擂台法进行改进并将其引入到 **NSGA-II** 中以提高 **NSGA-II** 非支配排序的速度^[16]；张丽科提出了改进蜜蜂进化型遗传算法引导的 **NSGA-II** 两阶段优化算法，以提高 **NSGA-II** 在求解复杂苛刻优化问题时的性能^[17]；宋晓鹏等人提出根据公交车车容量及城市道路信号控制等因素建立公交发车间隔优化模型，并利用 **NSGA-II** 进行求解^[18]；陶文化等人提出一种基于差分进化和 **NSGA-II** 的混合算法，以提高 **NSGA-II** 的收敛速度及 Pareto 最优解集空间分布的均匀性^[19]；王聪等人提出了一种新的多目标指数罚函数，并将其作为 **NSGA-II** 的适应度函数，以提高 **NSGA-II** 的收敛速度及 Pareto 最优解集空间分布的均匀性^[20]；耿焕同等人针对 **NSGA-II** 多样性保持策略不能客观反映个体间真实拥挤程度的问题提出了一种基于自适应混合非支配个体排序策略的改进型 **NSGA-II**^[21]；边毅等人提出利用 GPU 并行技术实现 **NSGA-II** 中适应度函数计算和交叉操作，通过异构计算提升 **NSGA-II** 的运行速度^[22]；陈志旺等人针对约束多目标区间优化问题提出了一种交互多属性决策 **NSGA-II**，即以多属性决策模型的属性集作为 **NSGA-II** 的目标函数，并通过对多属性决策模型的求解来选取符合决策者偏好的满意解^[23]；谢承旺等人针对 **NSGA-II** 种群多样性保持策略的不足提出了一种带差分局部搜索的改进型 **NSGA-II**，即利用差分进化中变异算子的定向引导作用改进 **NSGA-II** 的解群分布性^[24]；王超学等人针对 **NSGA-II** 早熟收敛及多样性不足的问题提出了一种改进的非支配排序合作型协同进化遗传算法，即利用外部档案存储每一代进化过程中产生的精英个体，并对其不断进行更新，以加快算法的收敛速度，同时给出了一种新的种群之间协同进化的方式，增强候选解的多样性^[25]；熊盛武等人提出采用岛并行模型，使子群体以不同交叉率和变异率独自演化，并以一定周期交换各自外部集合中的个体，提高了原始算法的优化性能^[26]；郑强提出将 **NSGA-II** 应用在用于滤除电网谐波的无源滤波器的设计及工业中以柴油为原料进行化合反应的模型参数辨识，验证了 **NSGA-II** 在工业设计中的有效性^[27]；李莉提出将 **NSGA-II** 分别与前馈式神经网络和支持向量机相结合，减少了原始模型的预测误差率^[28]；王鲁在 **NSGA-II** 基础上提出改进，对非支配排序分层中各层个体的选择比例和精英策略的应用代数进行动态调节，提高了种群多样性^[29]；李中林针对 **NSGA-II** 中个体没有明确适应度的问题提出了一种个体适应度计算方法，并给出了一种改进的局部搜索方法，对算法早熟收敛或陷入局部最优解起到了抑制作用^[30]；刘旭红等人将算术交叉算子引入到 **NSGA-II** 中，并提出了一种累积排序适应度赋值策略，有效促进算法收敛，并提高种群多样性^[31]；程刚提出了一种自适应划分策略，并将其引入到 **NSGA-II** 中，提高了解的多样性^[32]；赵森针对 **NSGA-II** 中交叉操作搜索性能较弱以及精英保留机制影响

种群多样性的问题，提出了一种基于渐变交叉分布指数和分层策略的改进算法，促进算法收敛，对种群多样性起到增加作用^[33]。

1.3 研究目的和意义

1.3.1 研究目的

在实际工程和科学研究中多目标优化问题随处可见，对其进行求解具有十分重要的现实意义及学术价值。NSGA-II 是同类算法中应用最为广泛的一个算法，能够在一次执行中得到一组帕累托最优解，但其求解过程中仍存在较多不足。随着社会发展的进步，对多目标优化问题进行高效求解的需求也越来越迫切，为进一步提升该类问题的求解效率，本文针对 NSGA-II 中存在的一些不足提出相应改进，提高其求解性能，使其能更好满足实际应用中的需求。

本文的研究目的是：首先，针对 NSGA-II 不能根据环境自适应调整运行参数的问题，给出了一种基于非支配排序分层的适应性策略，该策略通过提升 NSGA-II 运行自适应性，提高其全局搜索能力；其次，针对 NSGA-II 局部优化能力较弱的问题，对传统模拟退火算法进行适当调整并将其引入到 NSGA-II 中，给出了一种加入局部搜索的非支配排序遗传算法，提高原算法的局部搜索能力；最后将本文所给出的加入局部搜索的非支配排序遗传算法与第一阶段所给出的适应性策略相结合，给出了一种加入局部搜索的自适应非支配排序遗传算法，通过提高原算法进化适应性及局部优化能力以提升其求解全局最优解的能力，从而起到抑制早熟、加快收敛速度、增加种群多样性的作用。

1.3.2 研究意义

（1）理论意义

首先，本文针对 NSGA-II 在运行过程中运行参数保持不变，不能对运行参数进行自适应调整的问题提出了一种基于非支配排序分层的适应性策略，该策略根据算法的运行阶段、运行代数及当前临时种群非支配个体数对个体赋予相应的交叉率、变异率、交叉分布指数、变异分布指数，从而使算法具备运行的自适应性，提高算法全局搜索能力并抑制早熟；然后，本文针对 NSGA-II 局部搜索能力较弱的问题，对传统模拟退火算法进行适当调整并将其引入到 NSGA-II 中，给出了一种加入局部搜索的非支配排序遗传算法，该算法设置一定的搜索次数，在该搜索次数内对每一代进化生成的父代及子代种群个体进行退火搜索，并在运行过程中根据进化代数对退火温度进行调节，根据当前进化代所对应的退火温度动态调节当前进化代的搜索范围，通过提高算法的局部搜

索能力提升原算法的全局寻优能力；最后，本文将所提基于非支配排序分层的适应性策略和加入局部搜索的非支配排序遗传算法相融合，给出了一种加入局部搜索的自适应非支配排序遗传算法，提高了原算法的局部搜索能力及全局搜索能力，对NSGA-II优化性能起到了提升作用。

（2）现实意义

多目标优化问题在现实生活中十分常见，提高多目标优化问题的解决效率对人们日常生产、生活实践有着积极意义，传统多目标优化解法一般将原问题转化为单目标优化问题，采用如此做法虽然更便于求解但是却使建立的优化模型偏离现实情况，难以实现对现实场景的真实表达，而多目标进化算法往往难以克服其自身缺陷而需与其它策略相结合，因此对多目标优化问题的研究仍有较大的探索空间。随着科学技术的发展、社会的进步，多目标优化算法越来越清晰的显示出其在生产、生活中的重要作用，在现实中，优化问题往往是较为复杂的，具有多个参考指标或评价标准，如何对其得出一个性能均衡的优化解决方案一直是人们重点研究内容。本文对NSGA-II所进行的改进可以有效提升原始算法的优化性能，使其更好地解决工业生产指标及设计参数的选择、工厂生产计划的安排、公交车发车时刻的安排、政府决策计划的制定、电子商务中商品推荐等，提高生产效率及人们生活质量，具有良好应用前景及重要现实意义。

1.4 主要研究内容

本文主要研究内容可以概括为下几部分：

首先，本文通过对相关文献的研究及学习，充分了解了多目标优化问题的发展历史、背景及意义，深入分析多目标优化问题传统解法的不足，在此基础上对多目标优化算法中常用策略及性能评价指标进行总结，给出了基本框架，并介绍了NSGA-II的关键技术及基本流程。

其次，针对NSGA-II在运行过程中运行参数保持不变，不能根据环境的变化对运行参数进行自适应调整，从而导致求解效率较低的问题给出了一种基于非支配排序分层的适应性策略。所提策略根据算法的运行阶段、运行代数及当前种群非支配层个体数对不同个体赋予相应的交叉率、变异率、交叉分布指数、变异分布指数，通过提高算法的自适应性提升了算法的性能。

再次，针对NSGA-II局部搜索能力较弱，从而导致难以搜索到全局最优解集的问题，给出了一种基于模拟退火算法的NSGA-II优化改进。该改进算法首先对传统模拟退火算法进行调整，即以支配关系作为个体优劣的评价标准，在搜索过程中只接受优于原始解的新解，并根据当前进化代所对应的退火温度对搜索范围进行动态调节；然后将调整后的模拟退火算法引入到NSGA-II中，形成综合算法，即加入局部搜索的非支配排序遗传算法，通过对每一代进化生成

的父代及子代种群个体进行退火搜索，从而起到提高算法局部优化能力的作用；最后将加入局部搜索的非支配排序遗传算法与前文所给出的适应性策略相结合，给出了一种加入局部搜索的自适应非支配排序遗传算法，通过提高算法进化适应性及局部优化能力以提升原算法求解全局最优解的能力，从而起到抑制早熟、加快收敛速度、增加种群多样性的作用，实现对 NSGA-II 的优化改进。

最后，通过模拟仿真实验对所提改进进行验证。实验分为三部分：第一部分通过实验对比表明本文所提基于非支配排序分层的适应性策略能有效提高算法的全局搜索能力；第二部分通过实验对比表明本文所提加入局部搜索的非支配排序遗传算法能有效提高原算法的局部搜索能力；第三部分通过实验对融合本文所提基于非支配排序分层的适应性策略和加入局部搜索的非支配排序遗传算法的综合算法，即加入局部搜索的自适应非支配排序遗传算法进行性能验证，表明本文对 NSGA-II 进行的两点优化能有效结合，在收敛性及多样性两方面对算法的性能确有提升，从而证明了本文对 NSGA-II 优化研究工作的有效性。

1.5 论文的组织结构

本文对 NSGA-II 进行优化研究，对其在运行过程中运行参数保持不变，不能根据环境的变化对运行参数进行自适应调整、局部搜索能力较弱、无法找到全局最优解集或需要较长时间才能找到全局最优解集的问题，进行了研究及改进。本文共包括六章，具体结构如下：

第 1 章 绪论。本章节是论文的开篇，介绍了本文的研究背景、国内外研究现状，并根据 NSGA-II 中存在的问题，阐述了本文研究的目的和意义，最后在章末对本文的主要研究内容和具体论文结构进行了介绍。

第 2 章 文献综述。本章为后续具体研究进行相关知识介绍及理论铺垫，首先对多目标优化问题的相关概念及传统解法进行介绍，并对传统解法中存在的问题进行了具体阐述；然后根据多目标优化算法的主要研究方向，对其通用框架及常用策略进行了介绍，并给出了常用的性能评价标准；最后对 NSGA-II 关键技术进行介绍，并详细给出了基本流程，为后续对其进行优化改进奠定理论基础。

第 3 章 基于非支配排序分层的适应性策略。本章针对 NSGA-II 在运行过程中运行参数保持不变，不能根据环境的变化对运行参数进行自适应调整，从而导致搜索效率较低的问题，将单目标优化算法中常用的适应性策略进行适当改进，提出了一种基于非支配排序分层的适应性策略。该策略融合了基于个体的适应性策略和基于进化阶段的适应性策略，根据算法的运行阶段、运行代数及当前临时种群非支配个体数对个体动态赋予相应的交叉率、变异率、交叉分布指数及变异分布指数。

第 4 章 加入局部搜索的自适应非支配排序遗传算法。本章针对 NSGA-II

局部搜索能力较弱,从而导致难以搜索到全局最优解集的问题,给出了一种基于模拟退火算法的NSGA-II优化改进。该改进算法首先对传统模拟退火算法进行调整,即以支配关系作为个体优劣的评价标准,在搜索过程中只接受优于原始解的新解,并根据当前进化代所对应的退火温度对搜索范围进行动态调节;然后将调整后的模拟退火算法引入到NSGA-II中,形成综合算法,即加入局部搜索的非支配排序遗传算法,通过对每一代进化生成的父代及子代种群个体进行退火搜索,从而起到提高算法局部优化能力的作用;最后将加入局部搜索的非支配排序遗传算法与前文所给出的适应性策略相结合,给出了一种加入局部搜索的自适应非支配排序遗传算法,通过提高算法进化适应性及局部优化能力以提升原算法求解全局最优解的能力,从而起到抑制早熟、加快收敛速度、增加种群多样性的作用,实现对NSGA-II的优化改进。

第5章 实验与分析。本章利用模拟仿真实验对所提改进进行验证,实验分为三部分:第一部分通过实验对比表明本文所提基于非支配排序分层的适应性策略能有效提高NSGA-II全局搜索能力,从而起到抑制早熟、加快收敛速度、增加种群多样性的作用;第二部分通过实验对比表明本文所提加入局部搜索的非支配排序遗传算法能有效提高原算法的局部搜索能力;第三部分通过实验对融合本文所提基于非支配排序分层的适应性策略和加入局部搜索的非支配排序遗传算法的综合算法,即加入局部搜索的自适应非支配排序遗传算法进行性能验证,表明本文对NSGA-II进行的两点优化能有效结合,在收敛性及多样性两方面对算法的性能确有提升,从而证明了本文对NSGA-II优化研究工作的有效性。

第6章 总结与展望。本章对本文主要研究工作进行总结,再次阐述了本文对NSGA-II的优化改进,表明了本文的核心观点,并指出了本文的不足之处,为进一步的研究指明方向。

第2章 文献综述

本章针对论文所涉及的理论知识进行详细介绍，为进一步的研究工作做理论支撑。本章内容包含三个部分：第一部分对多目标优化问题的数学表现形式进行了具体描述，并对相关定义进行介绍；第二部分对问题的传统解法给出了具体介绍及详细分析，并给出多目标优化算法的基本框架、常用策略和性能评价标准；第三部分对NSGA-II的关键技术和基本流程进行介绍。

2.1 多目标优化问题

2.1.1 多目标优化问题的数学描述

优化问题包括最大化和最小化问题，根据对偶理论可知两者之间可以相互转化，因此多目标优化问题等价于多目标最小化问题，本文所研究的问题均为最小化问题，以下给出多目标优化的一般数学表达式，如式（2.1）所示。

$$\begin{aligned} \min \quad & F(x) = (f_1(x), f_2(x), \dots, f_m(x))^T \\ \text{s.t.} \quad & g_i(x) \leq 0 \quad (i = 1, 2, \dots, p) \\ & h_j(x) = 0 \quad (j = 1, 2, \dots, q) \end{aligned} \quad (2.1)$$

其中， $x = (x_1, x_2, \dots, x_n)^T \in X \subset \mathbb{R}^n$ 是 n 维决策向量， X 为 n 维决策空间， $y = (f_1(x), f_2(x), \dots, f_m(x))^T \in Y \subset \mathbb{R}^m$ 是 m 维目标向量， Y 为 m 维目标空间， $F(x)$ 为由 m 个从 X 向 Y 映射的目标分量函数构成的目标函数， $f_i(x)$ 为目标函数 $F(x)$ 的第 i 个目标分量函数， $g(x)$ 、 $h(x)$ 分别为对决策空间进行约束的 p 个不等式函数和 q 个等式函数，约束函数共同决定决策向量 x 的可行域。

2.1.2 多目标优化中的相关定义

定义 1. (可行解集) 可行解集 X_f 为在可行域范围内满足 (2.1) 式中约束函数 $g(x)$ 和 $h(x)$ 的解 x 的集合，即 $X_f = \{x \in X \mid g(x) \leq 0 \text{ 且 } h(x) = 0\}$ 。

定义 2. (帕累托支配) 可行解 $u, v \in X_f \subset \mathbb{R}^n$ ，在满足存在 $\forall i = 1, 2, \dots, m$ 使

得 $f_i(u) \leq f_i(v)$ ，并且 $\exists j=1,2,\dots,m$ 使得 $f_j(u) < f_j(v)$ 成立，则称 u 帕累托支配于 v ，记为 $u \prec v$ 。

定义 3. (帕累托最优解) 假设 $x^* \in X_f$ 且不存在 $\bar{x}^* \in X_f$ ，使得 $\bar{x}^* \prec x^*$ 成立，则称 x^* 是 (2.1) 式的帕累托最优解。

定义 4. (帕累托最优解集) 帕累托最优解集 (Pareto Set, PS) 是所有帕累托最优解的集合，即 $PS = \{x^*\} = \{x \in X_f \mid \nexists x' \in X_f : x' \prec x\}$ 。

定义 5. (Pareto 前沿) Pareto 前沿 (Pareto Front, PF) 是 Pareto 最优解在目标空间中的几何图形，即 $PF = \{F(x) \mid x \in PS\}$ 。

定义 6. (当前种群的非支配解集 NDS) 假设 $Pop(t)$ 为算法的第 t 代种群，即当前种群， $x^* \in Pop(t)$ ，当且仅当 $\nexists x' \in Pop(t)$ 使得 $x' \prec x^*$ ，则称 x^* 为 $Pop(t)$ 的非支配解， $Pop(t)$ 中所有非支配解 x^* 构成的集合被称为 $Pop(t)$ 的非支配解集 NDS(Non-dominated Set, NDS)。

定义 7. (极端解和极端解集) 若个体 $x \in NDS$ 且 $\nexists x' \in NDS$ 使得 x' 任意目标函数值上小于 x ，则称 x 为 NDS 的极端解，NDS 中所有极端解组成的集合被称为 NDS 的极端解集 NDES(Non-dominated Extreme Set, NDES)。

为进一步对多目标优化中相关概念进行解释，以下给出两个目标的帕累托支配关系示意图和两个目标的帕累托前沿面分布示意图，分别如图 2-1 和图 2-2 所示。

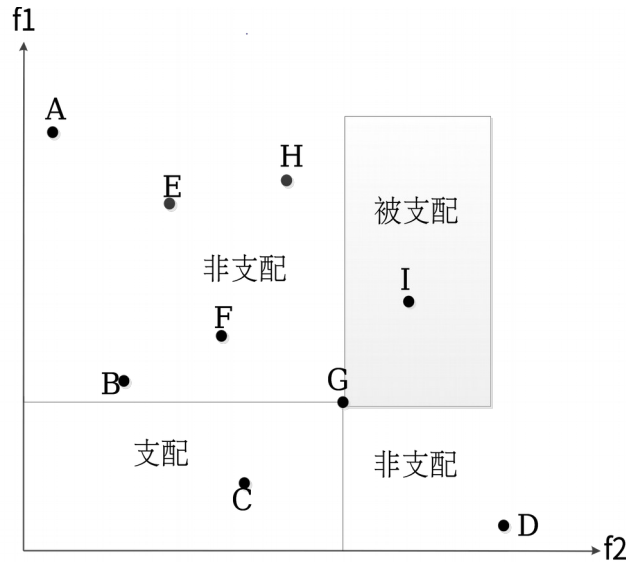


图 2- 两个目标的帕累托支配关系示意图

在图 2-1 中，以 G 点为例，被 G 点支配的区域在其右上部分，即 I 点被 G 点支配；支配于 G 点的区域在其左下部分，即 G 点被 C 点支配； G 点的左上部分及右下部分区域为其非支配区域，即 G 点与 A 、 B 、 E 、 H 、 F 、 D 为非支配关系。

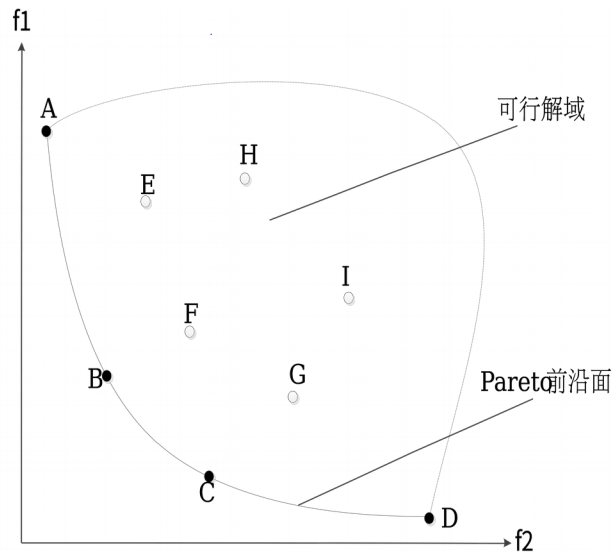


图 2- 两个目标的帕累托前沿面分布示意图

在图 2-2 中，弧线 $ABCD$ 是帕累托前沿面，其上的点均为帕累托最优解，其与虚线所构成区域为目标空间的可行解区域，可行解区域内的所有点均直接或间接受帕累托前沿面上的解支配。从图中可知， B 支配 E 、 F 、 H 、 I ； F 被 B 支配，但是支配 H 、 I ； E 、 F 、 G 无支配关系； H 、 I 无支配关系。

2.1.3 多目标问题的传统解法

多目标优化传统解法将原始问题转化成单目标求优问题，然后再对转化后的问题进行计算，以下对其中几种经典方法进行具体介绍。

(1) 加权法

加权法就是对原始问题中各子目标函数分别赋予一定权值，然后对其累加求和形成一个单目标函数，使原始问题转化为已于解决的形式^[34]。以下给出加权法的计算模型，如式 (2.2) 所示：

$$g(x) = \sum_{i=1}^m w_i f_i(x) \quad 0 \leq w_i \leq 1 \text{ 且 } \sum_{i=1}^m w_i = 1 \quad (2.2)$$

其中， $g(x)$ 为新的目标函数， $f_i(x)$ 为原始多目标函数， w_i 为权值系数。权值 w_i 代表着每个目标函数的相对重要程度，一般需人为根据主观设定，权值的设定对求解效率有着重要影响。加权法一次运算只能获取一个 Pareto 最优解，每次运算都需要重新设定权值，由于权值设定的主观性，优化得到的最优解往往与实际最优解有较大差距，因此该方法难以在实际应用中获得良好表现。

(2) 约束法

约束法就是根据某种前提条件从原始问题的 m 个目标中选择一个作为转化后的目标函数，而使其余 $m-1$ 个目标函数变换为约束条件^[34]。以下给出约束法的计算模型，如式 (2.3) 所示：

$$\begin{aligned} \min y &= f_c(x) \\ g(x) &= (g_1(x), g_2(x), \dots, g_p(x)) \leq 0 \\ \text{s.t. } h(x) &= (h_1(x), h_2(x), \dots, h_q(x)) = 0 \\ f_i(x) &\leq \varepsilon_i, (1 \leq i \leq m, i \neq c) \end{aligned} \quad (2.3)$$

其中， $f_i(x)$ 为所求原始问题的各子目标函数， m 为多目标函数的个数， $f_c(x)$ 为所选取的目标函数， $g(x)$ 、 $h(x)$ 为所求问题原有约束条件。参数 ε_i 为对多目标函数 $f_i(x)$ 进行约束的参数值，需要决策者根据多目标问题的各个目标本身的特点和所求问题的背景人为设定。该方法中所设置的参数值 ε_i 需要决策者对多目标问题的背景知识有足够了解，否则就难以得出较高优化水平的解，甚至无法求得任何解。

(3) 目标规划法

目标规划法对原始问题各子目标函数设置期望值，通过最小化各目标函数与所设定期望值绝对差值之和，实现对原问题的求解^[35]，其计算模型如式 (2.4) 所示：

$$\min y = \sum_{i=1}^m |f_i(x) - G_i| \quad (2.4)$$

其中， $f_i(x)$ 为多目标问题的各子目标函数，每个目标函数的期望值是

G_i , m 为多目标函数的个数。目标规划法不需要指定各个目标的权值, 但需要知道各目标函数的期望值, 该方法中的各目标函数的期望值仍需要决策者人为给出。

由以上分析可知, 传统解法一般存在以下几个缺点:

第一点, 每次运行只可以得出一个最优解, 需要多次运行才能得出问题的最优解集, 无法满足实际应用中对多个决策方案的需求;

第二点, 各目标函数之间往往具备不同的量纲及不同的物理意义, 没有共同的度量标准, 难以对其进行定量比较, 无法得到准确的优劣程度;

第三点, 对原有多目标函数进行单目标函数的转化需要设定相应的参数值, 而参数值的设定需要决策者依据经验及对问题背景的了解, 具有较大的主观性, 导致最终优化结果难以达到令人满意的程度。

2.2 多目标优化算法

多目标优化问题的传统解法存在较多不足, 求解效果往往差强人意, 现实应用中存在着较多局限性, 随着 Pareto 最优理论的引入, 尤其是在以 Pareto 最优理论为基础的多目标进化算法成为了该领域主要研究方向之后, 该问题得到了较好解决^[36]。

2.2.1 多目标优化算法的基本框架

多目标优化算法以 Pareto 支配理论和进化算法为基础, 其具体搜索过程在本质上与进化算法基本一致, 二者之间最大的区别在于对解优劣判断标准不相同, 也就是说在进化过程的个体选择阶段如何选择更优个体使其得到保存并参与到进化操作是两者的最大区别。在某种意义上来说, 多目标优化算法由进化算法与某些特定策略相融合而成, 是进化算法的一种特殊实现, 因此各种多目标优化算法均可以使用一个共同的基本框架来进行描述, 具体步骤如下所示。

Step1. 随机生成规模为 N 的初始种群 P_t ($t=0$) ;

Step2. 选择某个进化算法对种群 P_t 执行进化操作得到新种群 Q_t ;

Step3. 将种群 P_t 、 Q_t 合并从而得到临时种群 $P_t \cup Q_t$, 并从 $P_t \cup Q_t$

中求解出其非支配解集 R_t ;

Step4. 当前迭代次数自加 1 , 判断当前迭代次数是否到达预设上限, 如果

到达则停止运行并输出 R_t ；否则将 R_t 复制给 P_{t+1} ，转到 Step2。

多目标优化算法基本框架流程如图 2-3 所示。

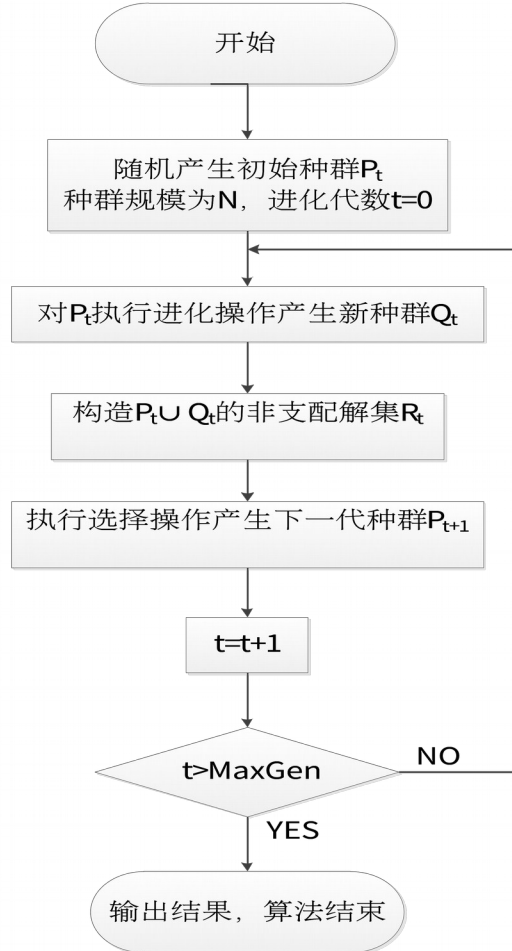


图 2- 多目标优化算法基本框架流程图

2.2.2 多目标优化算法的常用策略

多目标优化问题的两个求解目标为^[37]:

首先，提高算法的收敛性。使算法最终解尽最大可能接近帕累托最优前沿，尽量减少所求解与真实解集之间距离；

其次，维护种群个体多样性。由于真实应用中往往需要多个具有较大差异的选择方案，因此一般要求解出多个具有显著差别的最终解。

在设计算法时，为保证收敛性需要确定个体优劣判断标准与选择策略，从而促使种群个体向 Pareto 最优前沿方向进行进化；为确保算法的进化方向往往需要采用一定的精英保留机制，适当的精英保留机制能够有效防止进化过程中

由于随机因素所导致的最优个体丢失问题，保证算法的全局收敛；为确保算法种群多样性一般需要采取有效的群体多样性维护策略，防止算法早熟收敛，并提高最终解的差异性。由此可知，适应度赋值、多样性保持以及精英个体保留是影响算法性能的关键因素，下面从这三个方面来介绍多目标优化算法中常用策略。

（1）适应度赋值

多目标优化算法根据解之间的帕累托支配关系来判断解的优劣，因此适应度赋值策略需要对种群个体进行支配关系的层次划分，根据具体划分方式的不同一般可以将其分为非支配排序（**Non-dominated sorting**，**NDS**）和多目标排序（**Multi-objective ranking**，**MOR**）两种具体策略。

①NDS方法

NDS方法将进化种群作为候选解集，利用 **Pareto** 支配关系对其进行非支配解集和支配解集划分，对非支配解集个体赋予较高级别，如赋值为 1，并将支配解集作为候选解集；对候选解集进行非支配解集和支配解集划分，对划分出的非支配解集个体赋予较高级别，一般采用递增方式在上次划分赋值级别上加 1，并将支配解集作为候选集；再次对候选解集进行划分，如此循环操作直至种群中所有个体都已赋予级别。以下给出一个两目标最小化问题**NDS**方法分层示意图，如图 2-4 所示。

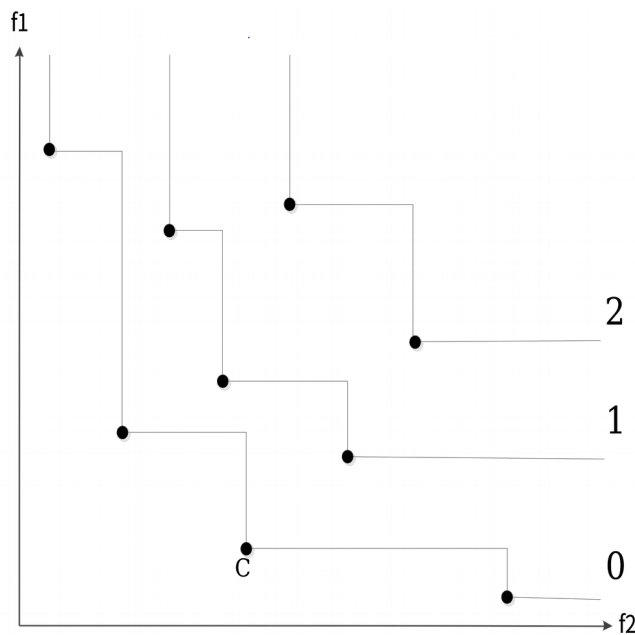


图 2- 两目标最小化问题的 **NDS** 方法分层示意图

② MOR 方法

MOR 方法一般只找出种群中全局 **Pareto** 最优个体，即全局非支配解集，

并对其中所有个体赋予较高的级别，如赋予级别为 1，因此 MOR 方法只对种群进行一次分层处理，得到全局非支配解集和支配解集。与 NDS 方法相同，MOR 方法将支配解集作为候选解集，而后根据非支配解集中个体在支配解集中支配的个体个数对其进行适应度赋值，一般来说所支配个体个数越多其适应度越大。对于支配解集中个体来说，其适应度往往由支配其的非支配解个体适应度所决定，其被支配的次数越多其适应度也就越大。因此，在 MOR 方法中个体适应度越大其级别越低，非支配解集中个体适应度最低而级别也最高，即适应度越小的个体越优。

（2）多样性维护

在多目标优化问题中维护种群多样性能够抑制早熟，提高最优解求解效率，并使最终解具有较好差异性，由此可知，多目标优化算法中多样性维护策略不仅可以使算法最终解得到较好分布性，同时也对全局优化性能起到促进作用，其常用策略有两种，分别为：小生境技术和密度估计法。

①小生境技术^[38]

小生境技术中基于共享机制的最具有代表性，在该技术中一般使用共享函数来计算个体间相似程度，在个体共享度求解过程中，为节省计算开销，往往设定共享半径，并将其与半径之内其它个体间共享函数值累加求和。共享度越大个体，意味着其周围其它个体越多，反之亦然，为维持种群多样性，共享度越小个体具有越大机会参与到下一代进化操作中。

②密度估计法

在种群的进化过程中根据个体的密度信息对其进行选择，一般来说，个体周边其他个体越多，越拥挤，密度也越大，被选择复制的机会越小，反之亦然。该方式中个体密度一般用个体与周围其它个体间距离之和或者用周围其它个体的个数来表示。

（3）精英保留机制

多目标优化算法中采用精英保留机制能够防止随机进化过程中优秀解的丢失，其精英保留机制的主要实现方式有^[39]： $(\mu+\lambda)$ 选择法和外部归档法。

① $(\mu+\lambda)$ 选择法

$(\mu+\lambda)$ 选择法是在 Schwefel^[40] 提出的种群 $(\mu+\lambda)$ 进化策略基础上建立的，采用种群融合的思想将 μ 个父代种群中个体和经过进化操作产生的 λ 个子代种群个体融合成一个种群，即为临时种群；然后通过对临时种群中个体进行优劣判断从中选择出若干个体作为下一代进化种群。该方法并不像传统的进化方法那样直接用子代群体替换父代群体作为新一代种群，而是采用先融合后选择的方式，在实现精英个体保留的同时增加了种群个体的采样规模。

②外部归档法

外部归档法需要构造一个归档集，归档集的存在是为了对种群进化过程中所搜索到的精英个体进行保存，而归档集中个体并不直接参与进化操作。归档集在算法开始时一般为空，在进化过程中由于更优秀个体的产生，同时受计算机内存资源的限制以及对算法时间复杂度及空间复杂度的考虑，归档集在进化过程中需要根据自身所保存个体与进化种群中个体的 **Pareto** 支配关系和额外信息不断进行自我更新。

以下给出两种机制的示意图，如图 2-5 所示。

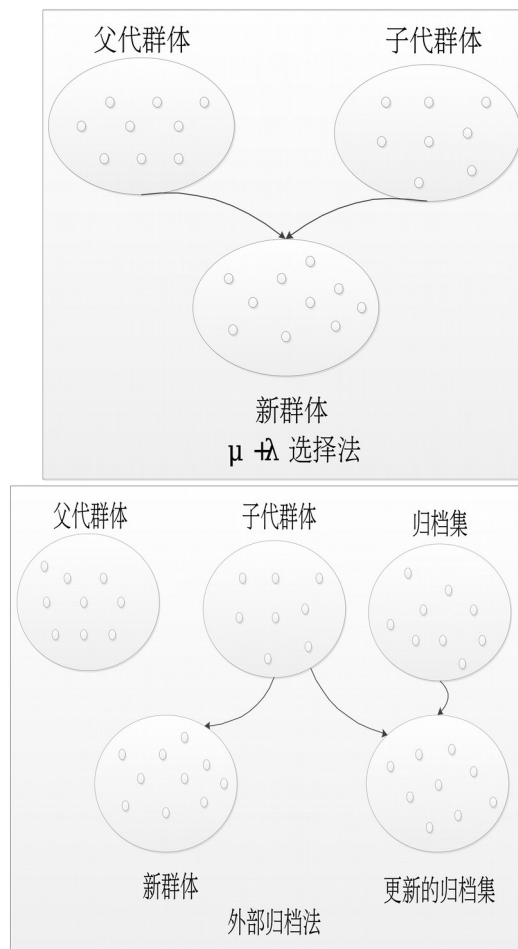


图 2- 精英保留机制的两种实现方法

2.2.3 多目标优化算法的性能指标

多目标优化算法的性能评价标准一般包括两个方面：一是算法从运行开始到产生结果所需要耗费的资源，这包括对计算资源的时间及存储空间的度量；二是对解质量优劣进行评价。对于前者来说，一般可以采用时间复杂度和空间

复杂度来进行估计，而在时间评价标准上，由于多目标优化算法运行的随机性，也可以采用对目标函数的评价次数或算法的运行代数来度量；对于后者，根据求解目标可知需要在收敛性和多样性两方面进行估计，也就是说要使求得的解尽可能的逼近问题的真实解，同时使所求得的解集具备较好的分布性。在实际应用中，对多目标优化算法进行性能比较，可以根据需要有针对性地选择某些指标进行评价。针对多目标优化算法中解质量的评价一般用以下两方面指标来度量^[41]：

（1）收敛性

求得的最优解集应尽可能逼近问题真实帕累托前沿面，即求得的最优解与问题真实最优解之间距离越小越好。以下给出两种典型的方法：

①收敛性准则 γ ^[37]

采用准则 γ 对算法性能进行评价时，首先需要预先知道所求问题的最优解集，对此一般采用某种成熟的多目标优化算法对原问题进行一定次数的寻优并将所求出的最优解集 P^* 近似作为问题的真实最优解集，有文献指出 P^* 的大小一般为 500，并且要求 P^* 尽可能均匀分布。该方法需要计算出待评价算法解与已知近似最优解集 P^* 中解的最小距离，并对其取均值。以下给出具体定义，如式（2.5）所示，及示意图如图 2-6 所示：

$$\gamma(P, P^*) = \frac{1}{|P|} \min_{p \in P} \{ \|p - p^*\|, p^* \in P^* \} \quad (2.5)$$

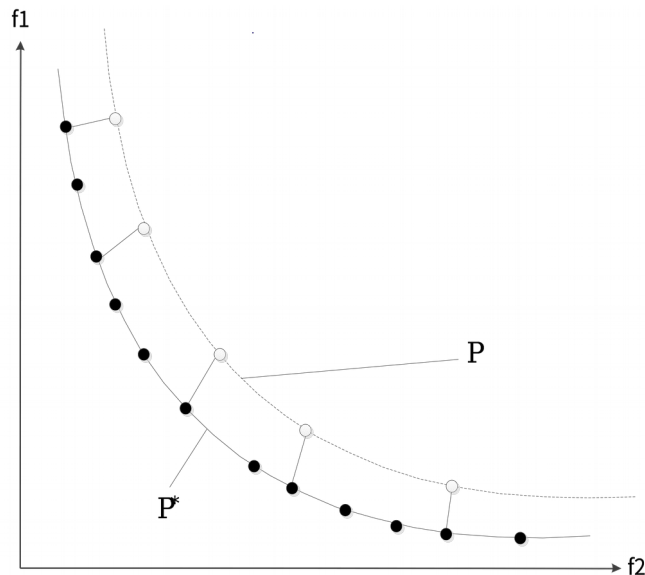


图 2-收敛性指标 γ 的示意图

当收敛指标 γ 越小时代表求解出的最优解集 P 与真实最优解集 P^* 之间距离越小，贴合程度越高。当 $\gamma=0$ 时，则代表所求出的最优解集与问题帕累托最优前沿面完全重合。

②两个解集之间的覆盖率（Coverage of sets, CS）^[41]

对同一个问题的两种不同求解方式所得出的最优解集 A 、 B 进行收敛性优劣判断，该种方式不需要与问题真实最优解集进行对照。 $CS(A, B)$ 表示 B 中被 A 支配的解的个体数目与集合 B 的个体总数的比例，如式（2.6）所示。

$$CS(A, B) = \frac{|\{u \in B \mid \exists v \in A: v \preceq u\}|}{|B|} \quad (2.6)$$

其中， \preceq 表示弱支配关系，两个解集间的覆盖率描述了两个解集之间的相互覆盖水平，是对解集的评价， $CS(A, B) \in [0, 1]$ 。当 $CS(A, B)=1$ 时表示 B 中的解全部被集合 A 中的解弱支配；当 $CS(A, B)=0$ 时表示 B 中没有解被 A 中的解所弱支配。一般来说， $A \nsubseteq B \cap \Phi$ ，在对两个解集进行覆盖率评价的话一般需要同时判断 $CS(A, B)$ 与 $CS(B, A)$ ，而 $CS(A, B)$ 与 $CS(B, A)$ 的和不一定等于 1。

（2）分布性

算法所求解分布性体现在两个方面，第一方面，解集多样性，即要求所得解在解集的前端曲线或前端曲面上尽可能的均匀分布；第二方面，解集分布广度，即要求所得解集的前端曲线或前端曲面对问题真实 Pareto 前沿面有很好的覆盖。以下给出常用的分布性度量方法，多样性准则 Δ ^[37]，定义如式（2.7）所示。

$$\Delta(P, P^*) = \frac{d_f + d_l + \sum_{i=1}^{|P|-1} |d_i - \bar{d}|}{d_f + d_l + (|P|-1)\bar{d}} \quad (2.7)$$

其中， P 为待评价的解集， P^* 为问题真实最优解集， d_i 为解集 P

中两个连续解向量间距离， \bar{d} 为全部 d_i 的平均值， d_f 和 d_l 为解集 P 的边界解与 P^* 中极端解之间的欧几里得距离。多样性准则 Δ 描述了所得解在所得解集的前端曲线或前端曲面上均匀分布程度，并对分布广度进行了考虑。准则 Δ 值越小，代表解集多样性越好。以下给出多样性准则 Δ 的示意图，如图2-

7 所示。

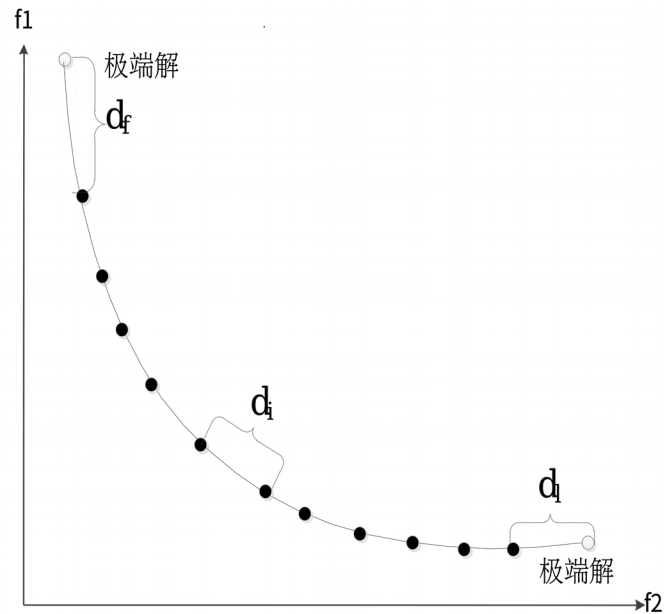


图 2- 多样性准则 Δ 的示意图

2.3 带精英策略的非支配排序遗传算法

带精英策略的非支配排序遗传算法（NSGA-II）同时兼备了良好分布性及较快收敛速度的特点，可在一次运行过程中得出多个高质量的解，与其他多目标优化算法相比在性能上具有较大的提升，已经成为了其它多目标优化算法进行性能对比的基准算法，被国内外学者广泛引用。根据数据统计，进化计算领域的权威期刊 IEEE Transactions on Evolutionary Computation 从1997年创刊至2008年底发表的文章中，Deb等提出的NSGA-II是被SCI引用次数最多的两篇文章之一。

2.3.1 NSGA-II的关键技术

第一代非支配排序遗传算法 (Non-dominated Sorting Genetic Algorithm, NSGA) 是由 Srinivas 和 Deb 于1994年提出的^[42]。而后，Deb等人在NSGA基础上进行改进，并提出了NSGA-II^[37]，其改进之处主要为采用了以下几方面关键技术：

- （1）快速非支配排序方法

快速非支配排序方法通过保存各解之间优劣关系的比较结果，并且利用索引查找机制，将计算复杂度由 $O(mN^3)$ 降到了 $O(mN^2)$ ，其中 m 为目标数量， N 为种群大小。在 NSGA 中根据支配关系对种群中 N 个个体进行层次划分，首先需要对种群中没有确定非支配排序分层序号的个体进行非支配排序，其中每个没有确定分层序号的个体都需要与其它没有确定分层序号的个体进行支配关系判断，因此该步操作的时间复杂度为 $O(mN^2)$ 。在该步操作结束之后便将种群划分为非支配解集和支配解集两部分，如果支配解集不为空，则需要再次对支配解集中个体进行层次划分，如此循环操作直至种群中所有个体都被划分，也就是说当种群中所有个体的非支配排序分层序号都被确定则结束排序操作，因此该种排序方法的时间复杂度为 $O(mN^3)$ 。而 NSGA-II 中所提的改进排序方法由以下两部分组成：

第一部分：为种群所有解设置两个变量 n_p 和 S_p ，并对其进行初始化，其中， $p=1,2,L,N$ ， n_p 用来记录所有解中支配解 p 的解个数， S_p 用来记录所有解中被解 p 支配的解集。该步操作中需要对种群中所有解进行双重遍历，计算每个解的 n_p 和 S_p ，因此时间复杂度为 $O(mN^2)$ 。

第二部分：根据第一部分排序后的结果进行分层，并初始化分层序号为 1。首先，将种群中所有 n_p 为 0 的个体移除种群，将当前非支配排序分层的序号赋予这些个体，这些个体即为该非支配排序分层上的个体；然后将这些个体 S_p 中所对应的个体 n_p 减 1，并将分层序号递增加 1，如此反复循环执行直到种群中所有个体的分层序号被赋予，因此该部分操作的时间复杂度为 $O(N^2)$ ，而此种排序方法的时间复杂度为 $O(mN^2) + O(N^2)$ ，即为 $O(mN^2)$ 。

在种群 P_t 上实施改进后的排序分层方法，其具体过程如下：

Step1. 对种群 P_t 中所有个体的 n_p 和 S_p 进行初始化，令 $n_p = 0$ ， $S_p = \Phi$ ， $p = 1, 2, L, N$ ；

Step2. 对种群 P_t 中的所有个体进行非支配判断，假设 p 、 q 为种群中任意的两个个体，如果 p 支配 q ，则 $S_p = S_p \cup \{q\}$ ， $n_q = n_q + 1$ ，如果 q 支配 p ，则 $S_q = S_q \cup \{p\}$ ， $n_p = n_p + 1$ ；

Step3. 对种群当前分层序号 k 进行初始化，令 $k = 1$ ；

Step4. 找出 P_t 中 $n_p = 0$ 的个体，将其从 P_t 中移出，并将其加入到分层集合 F_k 中，即 $F_k = F_k \cup \{p\}$ ；

Step5. 判断 F_k 是否为空，如果不为空，则将 F_k 中所有个体 S_p 中所对应的个体 n_p 减 1，并且 $k = k + 1$ ，跳转到 Step2；如果为空则结束操作。

（2）精英选择策略

采用 $(\mu + \lambda)$ 精英选择策略，将同一代的父代种群 P_t 与子代种群 Q_t 进行合并，从而形成临时种群 R_t ，并对 R_t 排序分层，利用拥挤选择算子从 R_t 中选取 N 个最为优秀的个体作为下一代父代种群。以下给出 NSGA-II 精英选择过程的示意图，如图 2-8 所示。

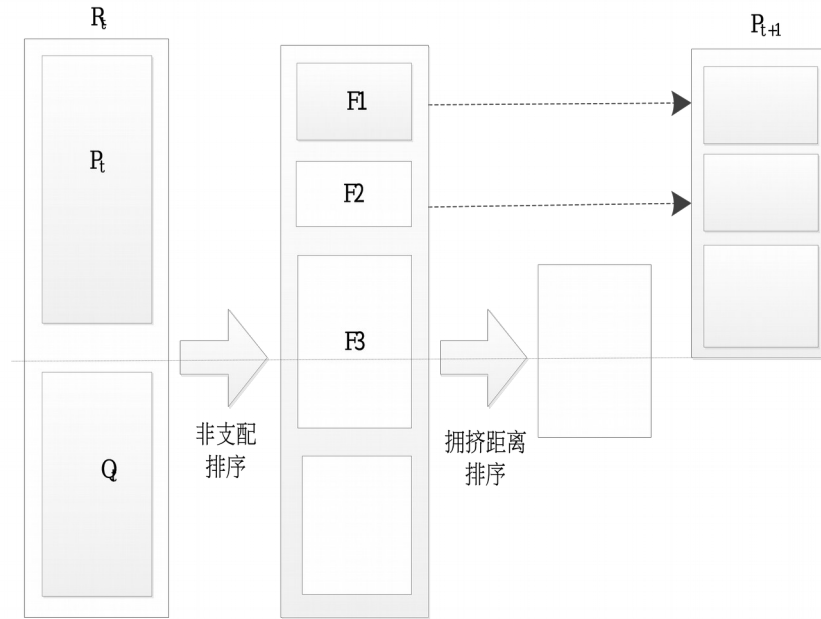


图 2- NSGA-II 精英选择过程示意图

（3）拥挤距离

拥挤距离用来表示同一排序分层内个体的聚集程度，并以此作为同一排序分层内个体的优劣评判标准，从而保持种群多样性。拥挤距离是对同一排序分层内个体在每一个目标上和与它相邻两个个体距离之差进行绝对值求和。个体

i 在第 k 个目标 f_k 上的拥挤距离为 $|f_k^{i+1} - f_k^{i-1}|$, $k=1,2,\dots,m$, m 为目标
的个数, f_k^{i+1} 、 f_k^{i-1} 是个体 i 在第 k 个目标上相邻两个个体的目标值。个体 i
的拥挤距离 d_i 如式 (2.8) 所示, 以下给出其示意图, 如图 2-9 所示。

$$d_i = \sum_{k=1}^m \left(|f_k^{i+1} - f_k^{i-1}| \right) \quad (2.8)$$

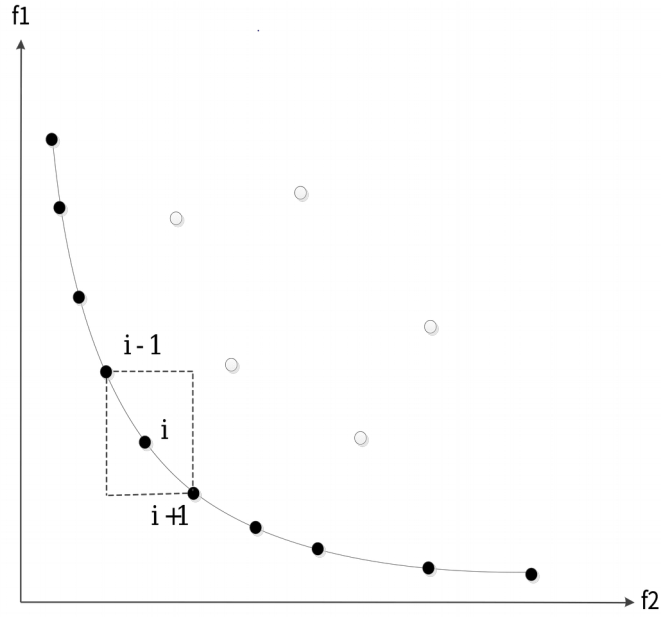


图 2-拥挤距离示意图

(4) 拥挤选择算子

为保持种群个体的分布性及多样性, 在使用 $(\mu+\lambda)$ 精英选择策略时需要根据个体的非支配排序分层序号和拥挤距离对个体进行选择, 从而选择出最优秀的 N 个个体。设个体 i 的非支配排序分层序号为 i_{rank} , 拥挤距离为 d_i 。
对于任意两个个体 i 、 j , 如果 $i_{rank} < j_{rank}$, 则选择个体 i ; 如果 $i_{rank} > j_{rank}$,
则选择个体 j ; 如果 $i_{rank} = j_{rank}$, 则根据两个个体拥挤距离来进行优劣判断, 数值越大代表其周围其他个体数量越少, 密度越稀疏, 此时认为该个体更优, 反之亦然。

2.3.2 NSGA-II的基本流程

NSGA-II的基本流程图如图2-10所示，其具体描述为：

Step1. 对运行参数，交叉率、变异率、交叉分布指数、变异分布指数进行初始化，在解空间内随机生成个体数为 N 的初始种群 P_t ， $t=0$ ，并将其作为父代种群；

Step2. 父代种群 P_t 进行二元竞赛选择、模拟二进制交叉、多项式变异操作，产生种群 Q_t ，并将 Q_t 作为子代种群；

Step3. 将 P_t 和 Q_t 融合在一起，作为规模为 $2N$ 的临时种群 R_t ，对其进行快速非支配排序和拥挤度计算，根据个体的非支配排序分层序号和个体的拥挤距离选择出最优的 N 个个体作为下一代进化操作中的父代种群 P_t ，

$t=t+1$ ；

Step4. 判断迭代次数是否等于预设上限值，如果等于则结束运行；否则，跳转到 Step2。

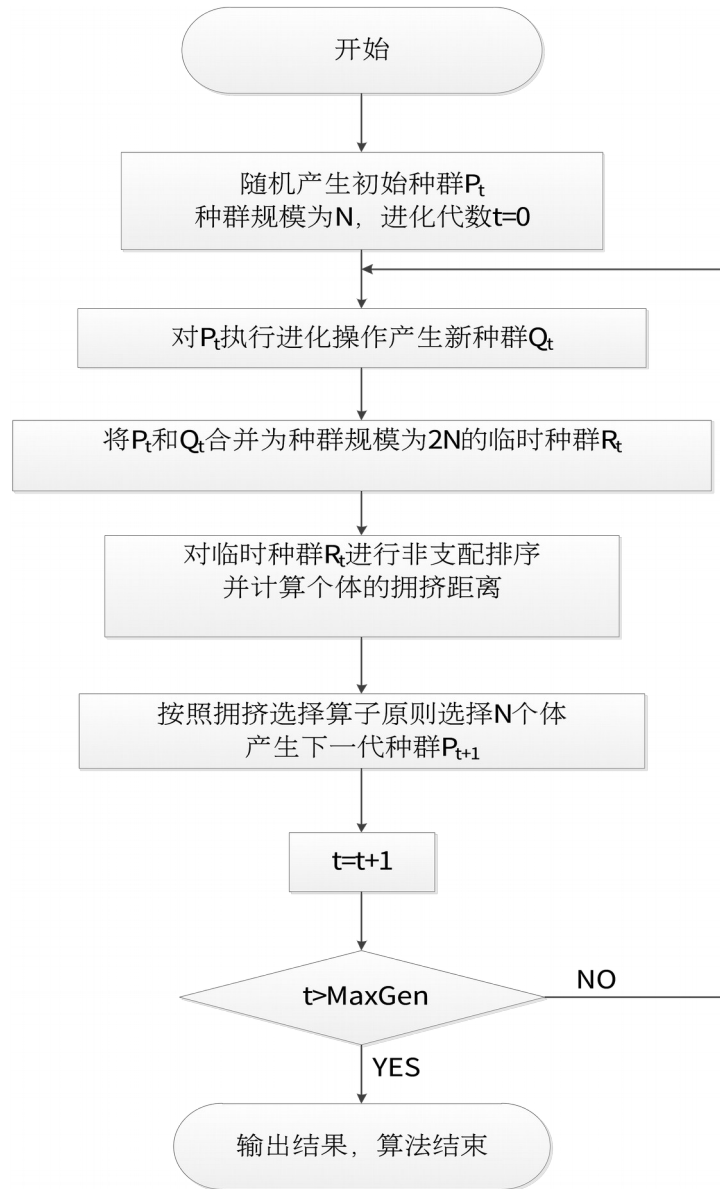


图2- NSGA-II 流程图

2.4 本章小结

本章是全文的理论基础部分，为后续研究工作做理论铺垫。首先介绍了多目标优化问题的相关定义，为后续研究工作奠定坚实的理论基础，并对传统多目标优化解法进行了介绍；然后介绍了多目标优化算法的基本框架、常用策略和性能评价标准；最后，对NSGA-II的关键因素和具体流程进行了介绍。

第3章 基于非支配排序分层的适应性策略

在前一章，给出了NSGA-II的关键技术和基本流程，本章针对其进化算子不能根据外界环境自适应调整运行参数，从而造成搜索效率低的问题，给出了一种基于非支配排序分层的适应性策略。该策略通过提高进化算子的自适应能力从而提升算法的搜索效率，起到抑制算法早熟、加快算法收敛的作用，有效提高算法的全局搜索能力。

3.1 传统的适应性策略

NSGA-II被广泛应用于各种多目标优化问题中，但是它仍存在一定的不足，其中之一就是算法在运行过程中不能够根据外界环境变化动态调整运行参数，导致算法搜索效率较低。该问题也存在于传统的单目标遗传算法中，在传统的单目标遗传算法中往往采用基于交叉率及变异率的适应性策略对其进行克服，其中以基于进化阶段和基于个体的适应性策略为代表。

3.1.1 基于进化阶段的适应性策略

在遗传算法不同进化阶段内由于种群多样性和个体平均适应度的不同，对交叉率及变异率有着不同的需求，运行初期阶段是其在编码空间上主要探索阶段，此时个体平均适应度较低、种群多样性较高，应采用较大的交叉率及变异率，促进新模式基因产生；在运行后期阶段由全局搜索逐渐转换为局部搜索，此时个体平均适应度较高、种群多样性较低，应采用较小的交叉率及变异率，保护优良基因不被破坏。在以上思想的基础上，文献[43]提出了基于进化阶段的适应性策略，并给出了具体的进化阶段划分方法，将算法运行阶段划分为三部分，即进化初期阶段： $[0, T_1]$ ， $T_1 = \alpha T$ ；进化中期阶段： $[T_1, T_2]$ ，

$T_2 = (1 - \alpha)T$ ；进化后期阶段： $[T_2, T]$ 。其中， T 为最大进化代数，参数

$\alpha \in [0, 0.5]$ ，一般设置 $\alpha = 0.382$ ，则 $T_1 = 0.382T$ ， $T_2 = 0.618T$ ；也可设置

$\alpha = 0.258$ ，则 $T_1 = 0.258T$ ， $T_2 = 0.742T$ 。在此基础上，文献[44]为基于进化阶段的适应性策略给出了具体的交叉率及变异率调整函数，如式（3.1）及式

(3.2) 所示, 其中 L 为个体的二进制编码长度。

$$p_c = \begin{cases} 0.00, & t \in [0, T_1] \\ 0.75, & t \in (T_1, T_2] \\ 0.50, & t \in (T_2, T] \end{cases} \quad (3.1)$$

$$p_m = \begin{cases} \min\{0.1, 10/L\}, & t \in [0, T_1] \\ 0/L, & t \in (T_1, T_2] \\ 0.1/L, & t \in (T_2, T] \end{cases} \quad (3.2)$$

由公式 (3.1) 、 (3.2) 可知基于进化阶段的适应性策略对同一进化阶段内的所有进化代数下的个体均设置相同的交叉率及变异率, 并没有针对进化代数及个体的适应度进行动态调整, 同时不同进化阶段之间的交叉率及变异率出现较大间断性。经过实验结果计算, 文献 [44] 给出了以下结论:

(1) 基于进化阶段的适应性策略可以有效延长算法的探索阶段, 对于存在模式欺骗性的函数可以有效提高算法的搜索效果。

(2) 在不同的二进制编码长度及进化阶段划分下, 基于进化阶段的适应性策略对遗传算法具有不同的优化效果。

(3) 基于进化阶段的适应性策略虽然能够提高算法的求解效果, 但是在很多情况下仍然难以达到理想的求解状态。

3.1.2 基于个体的适应性策略

有文献指出提升个体根据外界环境变化自适应调整进化表现的能力才是影响遗传算法性能和效率的根本原因^[45]。基于此种思想, Srinivas 等人提出基于个体的适应性策略^[46], 在该策略认为当个体适应度小于种群平均适应度时, 可以判定该个体性能不佳, 应对其赋予较大交叉率及变异率, 促进具有新模式的个体产生; 当个体适应度大于等于平均适应度时, 可以判定该个体所具有的模式基因较为优秀, 应对其赋予较小交叉率及变异率, 从而保证种群中较优模式基因不被破坏。以下给出其相应模型, 公式 (3.3) 为交叉率调整函数, 公式 (3.4) 为变异率调整函数。

$$P_c = \begin{cases} k_1 \frac{f_{\max} - f'}{f_{\max} - f_{avg}} & f' \geq f_{avg} \\ k_2 & f' < f_{avg} \end{cases} \quad (3.3)$$

$$P_m = \begin{cases} k_3 \frac{f_{\max} - f}{f_{\max} - f_{avg}} & f \geq f_{avg} \\ k_4 & f < f_{avg} \end{cases} \quad (3.4)$$

其中， P_c 为待交叉个体交叉率， P_m 为待变异个体变异率， f_{\max} 为种群个体适应度中最大值， f_{avg} 为种群个体平均适应度， f' 为两个待交叉个体中最大适应度， f 为待变异个体适应度， k_1 、 k_2 为交叉率调整函数的参数， k_3 、 k_4 为交叉率调整函数的参数。一般情况下， $k_1 = k_2$ ， $k_3 = k_4$ 。

由公式（3.3）、（3.4）可以得到交叉率和变异率的变化趋势图，如图 3-1 所示：

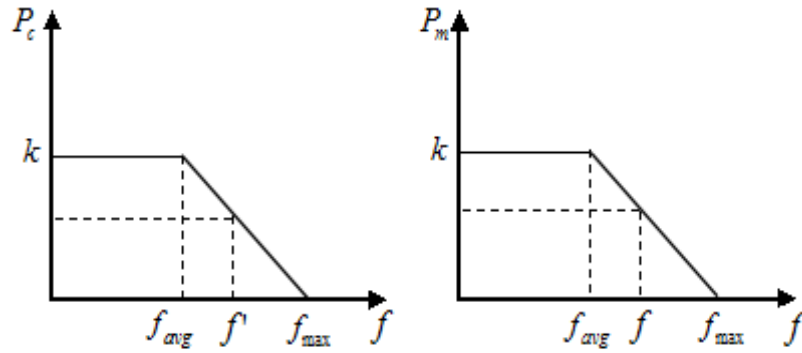


图 3- 自适应交叉率与变异率变化趋势图

由图 3-1 可以知道：

- （1）当个体适应度小于种群平均适应度时，对其使用较大的交叉率及变

异率，从而使其具有较大概率获得新模式基因。

(2) 当个体适应度不低于种群平均适应度时，对其使用较小交叉率及变异率，从而使该个体所具备的优秀模式基因以较大概率不被破坏，增加种群优秀基因模式的数量。

(3) 种群中最优个体的交叉率及变异率一直为零，也就是说在每一代进化中种群的最优个体都是不参与该代进化操作的。

从第一、二点可以知道该策略可以使个体在进化过程中更符合交叉率及变异率对外界环境的需求，提高进化种群中个体的自适应能力；但是从第三点也可以发现一定的不足之处，因为在算法运行的初期阶段和中期阶段是对解空间的全局搜索，在这两部分运行阶段算法难以搜索到问题的真正最优解，在算法运行的后期阶段是算法对解空间的局部搜索，在这部分运行阶段内种群的最优解往往仍然和问题的最优解存在一定的差距，而在进化算法中当前种群最优个体一般被认为具有更大概率进化到全局极值解，因此在算法运行的整体阶段内进化种群中的最优解仍然应该以一定的较小概率参与进化，从而引导种群向着更优的方向进化。

在公式 (3.3) 和 (3.4) 基础上，任子武等人提出了一种改进策略，改进策略引入精英个体保留策略并对基于个体的自适应交叉率及变异率调整函数设置不为 0 的固定初始值^[47]。任子武等人所提改进策略的交叉率及变异率的变化趋势图，如图 3-2 所示：

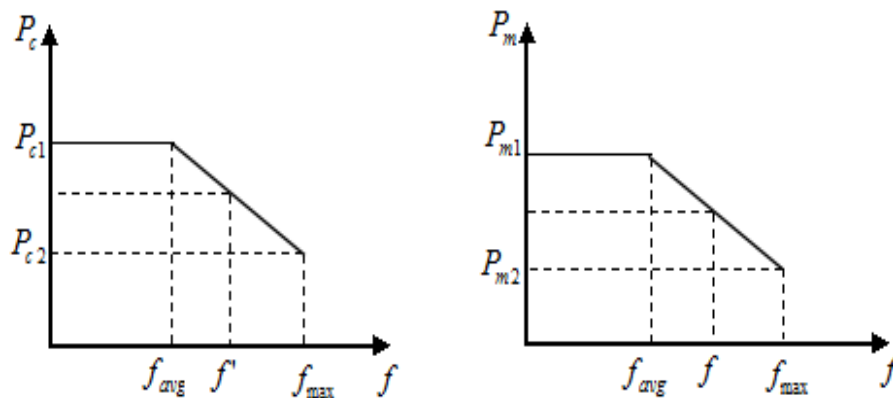


图 3- 改进型自适应交叉率与变异率变化趋势图

由图 3-2 可以看到任子武等人所提改进策略将交叉率及变异率调整函数的下限值上调，使算法在运行过程中种群最优个体的交叉率及变异率不为 0，使种群中最优个体可以参与进化，同时将较优个体的交叉率及变异率整体进行了提升，使它们可以继续进化。经过改进，对应于图 3-3 的交叉算子和变异算子的改进公式为 (3.5)、(3.6)：

$$P_c = \begin{cases} P_{c1} - (P_{c1} - P_{c2}) \frac{f' - f_{avg}}{f_{max} - f_{avg}} & f' \geq f_{avg} \\ P_{c1} & f' < f_{avg} \end{cases} \quad (3.5)$$

$$P_m = \begin{cases} P_{m1} - (P_{m1} - P_{m2}) \frac{f - f_{avg}}{f_{max} - f_{avg}} & f \geq f_{avg} \\ P_{m1} & f < f_{avg} \end{cases} \quad (3.6)$$

其中, P_c 为待交叉个体交叉率, P_m 为待变异个体变异率, f_{max} 为种群中最大适应度, f_{avg} 为种群个体平均适应度, f' 为待交叉的两个个体的最大适应度, f 为待变异个体适应度, P_{c1} 为交叉率调整函数的上限值, P_{c2} 为交叉率调整函数的下限值, P_{m1} 为交叉率调整函数的上限值, P_{m2} 为交叉率调整函数的下限值。

Srinivas 采用一组测试函数进行实验, 表明基于个体的适应性策略在多数情况下对算法的性能有显著提高, 可以有效抑制算法的早熟收敛并促进算法收敛, 但是对于完全欺骗性问题优化效果没有得到明显改善, 同时 Srinivas 也指出随着测试函数的复杂性和模式欺骗程度的不同, 基于个体的适应性策略对算法优化性能改善效果也不相同。

3.1.3 传统适应性策略的局限性分析

(1) 基于进化阶段的适应性策略只考虑了算法在不同进化阶段内对交叉率及变异率需求的不同, 而没有考虑在同一进化代内个体的进化操作与种群其他个体的关系。该策略中每一进化代内的个体均以同一交叉率和变异率参与进化, 只实现了种群对环境适应能力的模拟, 并没有实现个体对环境适应能力的模拟。

(2) 基于个体的适应性策略考虑了进化种群中个体与其他个体之间的关

系，能根据外界环境很好的调节个体的进化操作，但是并没有从进化的整体过程考虑个体的进化表现。该策略只实现了个体对环境适应能力的模拟，其中不同进化代内的个体均以同一交叉率和变异率调整函数对其进化参数进行调整，并没有实现种群对环境适应能力的模拟。

（3）基于进化阶段的适应性策略能够有效延长搜索阶段，通过减缓算法收敛速度提高其对全局最优解所在邻域的发现概率，抑制早熟收敛，从而提高算法的收敛性；基于个体的适应性策略能够有效提高算法全局搜索能力，加快收敛速度，抑制早熟收敛，从而提高算法的收敛性。由此可见，基于进化阶段的适应性策略和基于个体的适应性策略各有其优秀的地方，二者之间存在一定的互补作用。

3.2 基于非支配排序分层的适应性策略

本文针对NSGA-II在运行过程中不能够根据外界环境变化动态调节运行参数，导致搜索效率较低的问题，提出了一种基于非支配排序分层的适应性策略。本文所提策略根据算法运行阶段、运行代数及当前种群非支配个体数对交叉率、变异率、交叉分布指数及变异分布指数进行动态调整。所提策略包括交叉率及变异率调整模型、交叉分布指数及变异分布指数调整模型两部分。

3.2.1 交叉率及变异率调整模型

NSGA-II是一种基于非支配排序分层的算法，种群中个体根据支配关系可以划分为非支配解集和支配解集两部分，其中非支配解个体均优于支配解个体，因此本文所提模型分为两部分，第一部分子模型为非支配集个体的交叉率及变异率调整模型，此部分子模型在基于进化阶段的自适应策略基础上进行改进，根据算法的运行阶段对运行中每一代非支配个体的交叉率及变异率进行赋值；第二部分子模型为支配集个体的交叉率及变异率调整模型，此部分子模型在基于个体的自适应策略基础上进行改进，首先设定全局最大交叉率及变异率，并将其作为每一代种群中性能最劣个体运行参数值，然后根据NSGA-II中非支配排序后个体分层序号与最小分层序号和最大分层序号的关系线性的为支配集中个体赋予交叉率及变异率。

在进化算法中交叉率最大为1，变异率最大一般不能超过0.1，有文献指出如果变异率过大，将会导致算法的搜索过程变为随机搜索，因此本文所提模型中全局最大交叉率为1.0，全局最大变异率为0.1。

（1）非支配集个体交叉率及变异率调整模型

首先，对进化过程进行划分，不同进化阶段的起始交叉率及变异率与基于进化阶段的适应性策略相同，但是个体在同一进化阶段内交叉率及变异率并不

是保持固定不变的，而是随着进化代数的增加线性下降直至下一阶段的起始交叉率及变异率，此时下一阶段的进化开始进行；其次，为了使种群中非支配个体仍然可以以一定的概率参与进化，因此使非支配个体的交叉率及变异率随着进化代数的增加趋近于不为 0 的值。以下给出非支配集个体的交叉率调整模型，如式（3.7）所示。

$$p_c = \begin{cases} 0.25(T_1 - t)/T_1 + 0.75, & t \in [0, T_1] \\ 0.25(T_2 - t)/(T_2 - T_1) + 0.5, & t \in (T_1, T_2] \\ 0.5(T - t)/(T - T_2)(1 - \beta) + 0.5\beta, & t \in (T_2, T] \end{cases} \quad (3.7)$$

由于 NSGA-II 中的拥挤选择算子在生成下一代新种群时优先选择非支配排序分层序号较小的个体，也就是说每一次新种群生成时如果非支配层个体的数量小于种群规模则直接将其复制到新种群中。文献 [29] 中提出 NSGA-II 在一定运行代数后，新生成的种群中所有个体均为临时种群的非支配解，此时临时种群的支配集个体无法参与到下一代的进化操作中，新种群中所有个体互为非支配关系，彼此间优劣程度相同，从而导致种群多样性降低，使算法搜索过程陷入停滞状态，难以搜索到问题的最优解集。下面给出 NSGA-II 产生新种群过程的示意图，如图 3-3 所示。

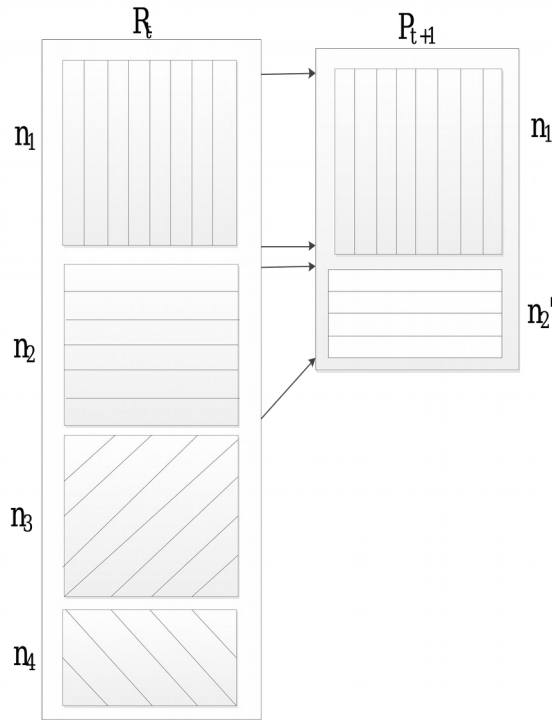


图 3- NSGA-II 新种群产生的过程示意图

其中，种群规模为 N ，临时种群 R_t 为第 t 代父代种群 P_t 和第 t 代子代种群

Q_t 合并而成, 由第 t 代临时种群 R_t 生成第 $t+1$ 代父代种群 P_{t+1} 。图 3-3 左半部分为临时种群 R_t 进行非支配排序分层后的情况, $n_1 + n_2 + n_3 + n_4 = 2N$; 图 3-3 右半部分为父代种群 P_{t+1} 中个体的非支配排序分层情况, $n_1 + n'_2 = N$ 。

本文所提策略中交叉率、变异率、交叉分布指数及变异分布指数调整模型通过提高个体的进化适应性, 促使算法以较快速度收敛, 同时也将促使算法的种群多样性降低得更加迅速。在遗传算法中一般采用高变异率的方式提高种群多样性, 所提非支配解个体变异率调整模型正是基于此种思想。该模型将临时种群中非支配个体数作为衡量算法种群多样性的一个评价指标, 将其划分为两个子部分。

①当临时种群中非支配个体数 r 小于种群规模 N 时。此时, 父代种群中个体存在支配关系, 其非支配排序分层数不为 1, 不需要增强种群多样性。

在遗传算法中当编码采用二进制时, 为保证算法所要求的求解精度往往会采用较长编码长度, 而在 NSGA-II 中由于采用实数编码可能会出现个体基因长度较短的情况, 在此情况下传统基于进化阶段的适应性策略中的变异模型将会出现变异率大于全局最大变异率的无意义现象, 对此本文将非支配集个体变异率模型分为两种情况: 变量个数大于等于 10 的情况和变量个数小于 10 的情况。以下给出具体模型, 变量个数大于 10 时变异率模型如式 (3.8) 所示, 变量个数小于 10 时变异率模型如式 (3.9) 所示。

$$p_m = \begin{cases} \min\{0.1, 10/L\} - 1/L)(T_1 - t)/T_1 + 1/L, & t \in [0, T_1] \\ 1/L - 0.1/L)(T_2 - t)/(T_2 - T_1) + 0.1/L, & t \in (T_1, T_2] \\ 0.1/L)(T - t)/(T - T_2)(1 - \beta) + (0.1/L)\beta, & t \in (T_2, T] \end{cases} \quad (3.8)$$

$$p_m = \begin{cases} 0.1, & t \in [0, T_1] \\ 0.1, & t \in (T_1, T_2] \\ 0.1 - 0.1/L)(T - t)/(T - T_2)(1 - \beta) + (0.1/L)\beta, & t \in (T_2, T] \end{cases} \quad (3.9)$$

其中, 变量的个数为 L , 算法的最大进化代数 T , 进化初期阶段为: $[0, T_1]$, $T_1 = \alpha T$; 进化中期阶段: $[T_1, T_2]$, $T_2 = (1 - \alpha)T$; 进化后期阶段: $[T_2, T]$ 。一般取 $\alpha = 0.382$, 则 $T_1 = 0.382T$, $T_2 = 0.618T$; 也可取 $\alpha = 0.258$, 则 $T_1 = 0.258T$, $T_2 = 0.742T$ 。 β 为进化后期阶段交叉率及变异率的调节系数, 该系数的设置可以保证非支配集个体的交叉率及变异率在进

化后期阶段渐近于后期阶段起始值的 β 倍, $\beta \in (0,1]$, 本文推荐设置为 0.4。

②当临时种群中非支配个体数 r 不小于种群规模 N 时。此时, 父代种群中个体不存在支配关系, 其非支配排序分层数为 1, 需要增强种群多样性, 提高种群中个体的变异率, 以下给出具体模型, 如公式 (3.10) 所示。

$$\begin{aligned} p'_m &= \frac{p_m + p_{mMax}}{2} & N < \alpha N \\ p'_m &= p_{mMax} & \alpha N \leq 2N \end{aligned} \quad (3.10)$$

其中, 种群规模为 N , 临时种群规模为 $2N$, 临时种群中非支配个体数为 r , 全局预设最大变异率为 p_{mMax} , 非支配个体变异率为 p_m , 由式 (3.8) 或者式 (3.9) 所确定, 阈值 α 为种群多样性的评判系数, $\alpha \in (1,2]$, 本文推荐设置为 1.5。 p'_m 为种群中所有个体的变异率, 即多样性提升后非支配个体的新变异率。

(2) 支配集个体交叉率及变异率调整模型

本文认为种群中最劣个体应该一直保持较大的交叉率及变异率, 从而使最劣个体可以一直以较大概率对解空间进行搜索, 加快更优模式的产生, 因此本模型中种群最劣个体的交叉率及变异率等于全局预设最大值。支配集个体根据分层序号与最大分层序号之间的关系线性调整自身的交叉率及变异率。种群中个体的交叉率及变异率模型如式 (3.11)、(3.12) 所示。

$$P'_c = \frac{(P_{cMax} - P_c)(i_{rank} - 1)}{(R-1)} + P_c \quad (3.11)$$

$$P'_m = \frac{(P_{mMax} - P_m)(i_{rank} - 1)}{(R-1)} + P_m \quad (3.12)$$

其中, P'_c 为种群个体的交叉率, P'_m 为种群个体的变异率, P_{cMax} 为全局最大交叉率, P_{mMax} 为全局最大变异率, i_{rank} 为个体的非支配排序分层序号, R 为种群非支配排序的最大分层序号, P_c 为当前种群非支配个体的交叉率,

由式（3.7）所决定， P_m 为当前种群非支配个体的变异率，由式（3.8）或式（3.9）所决定。一般情况下， $P_{cMax}=1.0$ 、 $P_{mMax}=0.1$ 。

3.2.2 交叉分布指数及变异分布指数调整模型

在模拟二进制交叉操作中，子代个体的间距与父代个体的间距成比例，这个间距比例是随交叉分布指数 η_c 变化的递减函数。 η_c 取值越大，所生成子代个体远离父代个体概率越小，反之亦然。在多项式变异操作中，变异分布指数 η_m 取值越大，所生成子代个体远离父代个体概率越小，反之亦然。因此，在算法运行初期阶段应使用较小的 η_c 和 η_m ，使算法具备更大概率搜索到较远空间，提高算法对未知空间的探索能力；随着进化代数的增加，算法从全局搜索转向局部搜索，解个体趋于收敛，此时应逐渐增大 η_c 和 η_m ，实现较小范围的集中搜索，提高算法局部搜索能力。本文认为在考虑进化代数对搜索性能需求的同时，也需要考虑个体与种群中其他个体的关系，在本模型中种群最劣个体一直保持较大搜索空间，种群中最优个体随着进化代数的增加逐渐减少搜索空间，其它个体根据分层序号与最大分层序号之间的关系线性调整自身的 η_c 和 η_m 。与交叉率及变异率调整模型相似，本模型也分为两部分，即非支配集个体交叉分布指数及变异分布指数调整模型、支配集个体交叉分布指数及变异分布指数调整模型。

（1）非支配集个体的交叉分布指数及变异分布指数调整模型

该模型中首先设置非支配集个体在算法初始和终止时的 η_c 和 η_m 值，在算法运行过程中非支配集个体根据进化代数线性调整自身的 η_c 和 η_m 值。以下给出种群中非支配个体 η_c 和 η_m 的模型，如式（3.13）、（3.14）所示。

$$\eta_c = \eta_{cMin} + (\eta_{cMax} - \eta_{cMin}) t / T \quad (3.13)$$

$$\eta_m = \eta_{mMin} + (\eta_{mMax} - \eta_{mMin}) t / T \quad (3.14)$$

其中， η_c 为非支配个体的交叉分布指数， η_m 为非支配个体的变异分布指

数, η_{cMin} 为算法初始时的交叉分布指数, η_{cMax} 为算法结束时的交叉分布指数, η_{mMin} 为算法初始时的变异分布指数, η_{mMax} 为算法结束的变异分布指数, t 为种群的当前进化代, T 为算法的最大进化代数。为使算法在运行初期具有较大搜索空间, 运行后期具有较小搜索空间, 一般情况下推荐设置为 $\eta_{cMin} = \eta_{mMin} = 1$, $\eta_{cMax} = \eta_{mMax} = 30$ 。

(2) 支配集个体的交叉分布指数及变异分布指数调整模型

为使种群最劣个体一直具有较大搜索空间, 对其交叉分布指数及变异分布指数赋值为全局最小值, 即非支配集个体在算法初始时的数值。支配解集个体根据分层序号与最大分层序号之间的关系线性调整自身的交叉分布指数及变异分布指数, 具体模型如式 (3.15)、(3.16) 所示:

$$\eta'_c = \eta_c - \frac{(\eta_c - \eta_{cMin})(i_{rank} - 1)}{(R - 1)} \quad (3.15)$$

$$\eta'_m = \eta_m - \frac{(\eta_m - \eta_{mMin})(i_{rank} - 1)}{(R - 1)} \quad (3.16)$$

其中, η'_c 为种群个体的交叉分布指数, η'_m 为种群个体的变异分布指数, η_{cMin} 为全局最小交叉分布指数, η_{mMin} 为全局最小变异分布指数, i_{rank} 为个体的非支配排序分层序号, R 为种群非支配排序的最大分层序号, η_c 为当前种群非支配个体的交叉分布指数, 由式 (3.13) 所决定, η_m 为当前种群非支配个体的变异分布指数, 由式 (3.14) 所决定。一般情况下, $\eta_{cMin} = 1$ 、 $\eta_{mMin} = 1$ 。

3.3 本章小结

NSGA-II 中进化算子不能根据外界环境自适应调整运行参数, 而造成搜索效率低的问题, 对此本章提出了一种基于非支配排序分层的适应性策略。所提策略在传统的基于进化阶段的适应性策略和基于个体的适应性策略基础之上进行改进和融合, 并将其引入到 NSGA-II 中, 该策略包括交叉率及变异率调整模型、交叉分布指数及变异分布指数调整模型两部分, 通过提高进化算子的自

适应能力提高了算法的搜索效率，起到抑制算法早熟、加快算法收敛的作用，有效提高算法的全局搜索能力。

第 4 章 加入局部搜索的自适应非支配排序遗传算法

在 NSGA-II 运行的前期阶段主要是对解空间的全局搜索，由于其较强的全局搜索能力，能较快定位到最优解集的大致位置；在运行后期阶段主要是对解空间的局部搜索，由于原算法局部搜索能力的不足，往往进化迟缓，难以搜索到问题的真实解集。本章首先针对 NSGA-II 局部搜索能力较弱的问题，对传统模拟退火算法进行适当调整，并将其引入到 NSGA-II 中，给出了一种加入局部搜索的非支配排序遗传算法；然后将其与第三章所提策略相结合，给出了一种加入局部搜索的自适应非支配排序遗传算法，在全局优化和局部优化两方面提高 NSGA-II 求解性能。

4.1 局部搜索算法与传统遗传算法的组合

由模式理论可知，遗传算法在运行的前期阶段，种群中个体大多以低阶模式存在，种群多样性较高，此时算法主要是在解空间中进行全局搜索；在算法运行的后期阶段，种群中个体大多以高阶模式存在，种群多样性较低，此时算法主要是在解空间中进行局部搜索^[48]。由此可知，遗传算法在解决某些较难问题时，在运行后期阶段由于种群多样性的降低及高阶模式的产生，往往局部优化能力不佳，难以获得全局最优极值或求解精度过低^[49, 50]。针对此问题，传统解决方案中往往将局部搜索算法引入到其中，形成混合遗传算法，从而通过提升算法局部优化能力达到提高全局寻优能力的目的^[51]。

4.1.1 爬山算法

爬山算法是一种基于贪心策略的局部搜索算法^[52]，并不具备全局搜索能力，影响其取搜索到全局最优解的因素为初始解的分布情况以及所求问题的峰值数量^[53]。爬山算法实现简单，求解速度快，但是最主要的缺点是无法保证取得的点为全局最优解值，处理多峰问题时往往会陷入局部最优解^[53]。以下给出爬山算法的示意图，如图 4-1 所示。

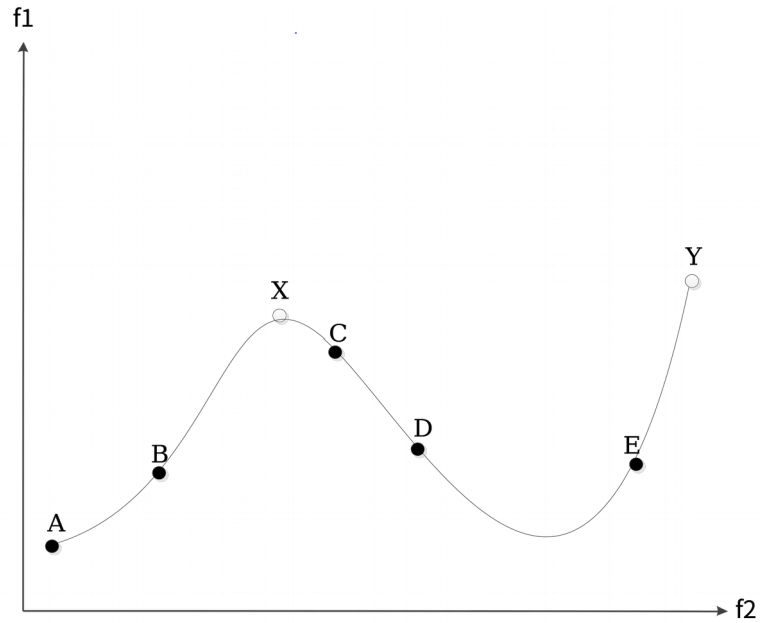


图4- 爬山算法的示意图

其中，点 A 、 B 、 C 、 D 、 E 为爬山算法中的个体，其中点 X 为目标函数的局部极值点，点 Y 为目标函数的全局极值点。由图 4-1 可知，点 A 、 B 、 C 、 D 最终搜索到的极值点为点 X ，也就是说最终陷入局部极值点；点 E 最终到达全局最优值点 Y 。

在爬山算法中，如果个体的编码形式为二进制，则称为位爬山算法。位爬山算法在个体邻域中选择新个体的过程与二进制编码的遗传算法中个体变异过程相似，有文献指出对于基于二进制编码的函数优化和组合问题，与通用的局部搜索算法相比，采用位爬山算法具有很好的解决能力。以下给出位爬山算法中单一个体的搜索过程，其中为便于描述，假设个体中选择不同二进制位进行变异的概率相同，个体的二进制编码长度为 L 。

Step1. 随机初始化个体： $a = (a_1, a_2, \dots, a_L)$ ；

Step2. 计算个体的适应值： $f^* = f(a)$ ；

Step3. 随机生成变异位选择序列： $Bit_list = \{j_1, j_2, \dots, j_L\}$ ， $l = 1$ ；

Step4. 顺序选择变异位 $j_l \in Bit_list$ ；

Step5. 对变异位上基因值进行变异: $a'_{jl} = 1 - a_{jl}$, 生成新个体 a' ;

Step6. 计算产生的新个体的适应值: $f' = f(a')$;

Step7. 比较原个体和新个体适应值大小关系: *if* $f' > f^*$ *then* $f' = f^*, a = a'$;

Step8. 循环: $l = l + 1$, *if* $l < L$, 则返回到 Step4;

Step9. 循环: *if* $l = L$, 返回到 Step3;

Step10. 满足终止条件, 终止运行。

由于位爬山算法与实数编码的爬山算法除了在编码形式上不同以外, 在基本步骤上大致相同, 两者之间最主要不同之处在于具体爬山变异操作不同, 也就是说由于编码不同其具体变异操作也不同。以下对实数编码的爬山算法中单一个体的搜索过程给出具体描述, 其中个体变量的个数为 L , 第 l 个变量的上限为 V_{up}^l , 下限为 V_{low}^l , r 为搜索距离的调节因子, α 为 r 的上限值, $r \in (0, \alpha)$, $0 < \alpha < 1$ 。

Step1. 随机初始化个体: $a = (a_1, a_2, \dots, a_L)$;

Step2. 计算个体的适应值: $f^* = f(a)$;

Step3. 随机生成变异变量选择序列: $Bit_list = \{j_1, j_2, \dots, j_L\}$, $l = 1$;

Step4. 顺序选择变异变量位置 $j_l \in Bit_list$;

Step5. 对所选择的第 l 个变量进行变异, 首先在该变量增长的方向上进行变异, 即 $a_{jl}^x = a_{jl} + r(V_{up}^l - a_{jl})$, 计算产生的新个体的适应值 $f^x = f(a^x)$; 然后在该变量减少的方向上进行变异, 即 $a_{jl}^y = a_{jl} - r(a_{jl} - V_{low}^l)$, 计算产生的新个体的适应值 $f^y = f(a^y)$; 最后对产生的新个体适应值进行比较, *if* $f^x > f^y$, 则 $a_{jl}^x = a_{jl}^x$, *if* $f^x < f^y$, 则 $a_{jl}^y = a_{jl}^y$, *if* $f^x = f^y$, 则在 a_{jl}^x 和 a_{jl}^y 中随机选择一个作为 a_{jl}' , a_{jl}' 为该次爬山变异过程所获得的新个体;

Step6. 计算产生的新个体的适应值: $f' = f(a')$;

Step7. 比较原个体和新个体适应值大小关系: *if* $f' > f^*$ *then* $f' = f^*, a = a'$;

Step8. 循环: $l = l + 1$, *if* $l < L$, 则返回到 Step4 ;

Step9. 循环: *if* $l = L$, 返回到 Step3 ;

Step10. 满足终止条件, 终止运行。

由上述对二进制及实数编码的爬山算法所进行的分析, 可以知道不同编码形式下的爬山算法基本流程大致相同, 因此可以采用一个基本框架来表示。以下给出单一个体爬山搜索的流程图, 如图 4-2 所示。

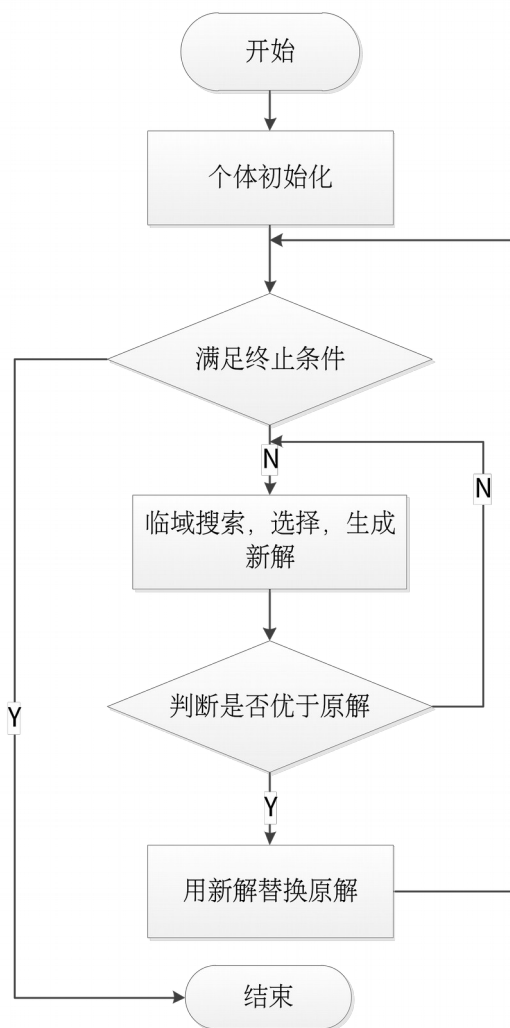


图 4- 单一个体爬山搜索的流程图

4.1.2 模拟退火算法

模拟退火算法^[54] (Simulated Annealing, SA) 是一种基于随机寻优方式的目标优化算法, 具备较强的局部搜索能力, 它在运行本质上其实也是一种贪心算法, 能够有效利用已有的求解信息进行“爬山”寻优操作, 但其与爬山算法并不完全相同: 爬山算法是一种完完全全的贪心算法, 并不具备全局寻优能力, 每次搜索过程中都从当前解的临近解空间中随机选择一个优于当前解的新解作为下一次搜索的起始点; 而模拟退火算法采用随机的方式对解空间进行搜索, 并以一定概率接受搜索过程中得到的劣于原始解的新解, 使其在进入局部极值陷阱的时候可以有一定概率跳出, 并以一定概率搜索到全局最优值^[55]。由此可知, 模拟退火算法具备一定的全局寻优能力, 从理论上说, 在无限次迭代计算的情况下其必然收敛于全局最优值。

模拟退火算法是对自然界中固态物体热度变化过程的模拟, 当固态物体热度较高时, 其内部微粒运动较为剧烈、动能较大, 当其热度较低时, 其内部微粒运动趋于平缓、动能较小, 因此固态物体热度变化的过程, 可以视为其内部微粒在两种不同能量状态间变迁的总体体现^[56, 57]。根据 Metropolis 准则, 固态物体在热度为 T 时其内部微粒从现有能量状态变迁到常温时稳定能量状态的概率为 $e^{(-\Delta E/kT)}$, 其中 E 为固态物体在热度为 T 时其内部微粒的动能, ΔE 为内部微粒的动能变化量, k 为 Boltzmann 常数。模拟退火算法根据 Metropolis 准则来判断是否不拒绝劣于原始解的新解成为下一次运算的起始点, 其新解接受概率如公式 (4.1) 所示。

$$p = \begin{cases} 1 & f(x_{new}) < f(x_{old}) \\ e^{-\frac{f(x_{new}) - f(x_{old})}{kT}} & f(x_{new}) \geq f(x_{old}) \end{cases} \quad (4.1)$$

其中, p 为劣解接受概率, k 为调节参数, 当前进化代对应的热度状态为 T , f 为目标函数, $f(x_{old})$ 为原始解的目标函数值, $f(x_{new})$ 为新解的目标函数值, 目标函数的优化模式为最小化。

由公式 (4.1) 可知, 模拟退火算法中劣解接受概率不仅与新解的劣化程度相关, 同时也与当前热度状态 T 相关, 当前热度越高, 对于劣解的接受概率越大, 当前热度越低, 对于劣解的接受概率越小。由此可见, 对当前热度, 即

当前状态所对应的温度进行控制对算法寻优性能有重要影响，其中常用的温度降低方式有：经典降温方式，如公式（4.2）所示；快速降温方式，如公式（4.3）所示；按比例降温方式，如公式（4.4）所示，其中 α 为降温比例系数， $0 < \alpha < 1$ 。

$$T(t) = \frac{T_0}{\log(1+t)} \quad (4.2)$$

$$T(t) = \frac{T_0}{1+t} \quad (4.3)$$

$$T(t) = T_0 \alpha^t \quad (4.4)$$

由公式（4.2）、（4.3）、（4.4）可知，模拟退火算法中热度冷却采用的是逐渐下降的方式，在运行初期阶段热度值较高，使其以较大概率接受劣于当前解的新解，增加了全局搜索性能，避免其陷入局部极值；在运行后期阶段热度值逐渐由高转低，使其以较小概率接受劣于当前解的新解，避免了当前较优解的丢失，使其以较高的爬山性能进行局部搜索。由此可见，该种方式既符合自然界中固态物体真实冷却过程，同时也符合算法运行过程中对外界环境变化的适应性模拟，从而使其在搜索过程中具备一定的适应性。

以下给出模拟退火算法中单一个体进行搜索的基本步骤，其中以最小化问题为例。

Step1. 设置运行初始参数，包括退火搜索起始状态下的当前温度 T 、调节系数 k 、当前迭代次数 n 、运行迭代次数上限值 N 、内循环当前搜索次数 w 及其上限值 W ，其中 $n=0$ ， $w=0$ ；

Step2. 在当前解 S 可行解空间中的某一邻近范围内采用随机搜索的方式得到一个新解 S' ；

Step3. 计算从解 S 变化到解 S' 的能量增量 $\Delta E = f' - f$ ，其中， f 为解 S 所对应的能量值，即其所对应的目标函数值； f' 为解 S' 所对应的能量值，即其所对应的目标函数值；

Step4. 根据接受准则判断是否接受搜索到的新解，即如果 $\Delta E < 0$ ，则接受新解 S' ，并用其替代当前解，跳转到 Step7；如果 $\Delta E \geq 0$ ，随机生成概率值 r ，跳转到 Step5，其中 $0 < r < 1$ ；

Step5. 如果 $r < e^{(-\Delta E/kT)}$ 则接受新解 S' ，用其替代当前解， $w=0$ ，跳转到 Step7;

Step6. $w = w+1$ ，如果 $w < W$ ，则跳转到 Step2; 否则 $w=0$ ，跳转到 Step7;

Step7. 根据温度控制方式更新当前温度 T ;

Step8. 当前迭代次数 n 自增加 1，即 $n = n+1$ ，判断当前迭代次数 n 是否到达所设定的上限值 N ，到达则终止程序，否则跳转到 Step2。

以下给出单一个体模拟退火搜索的流程图，如图 4-3 所示。

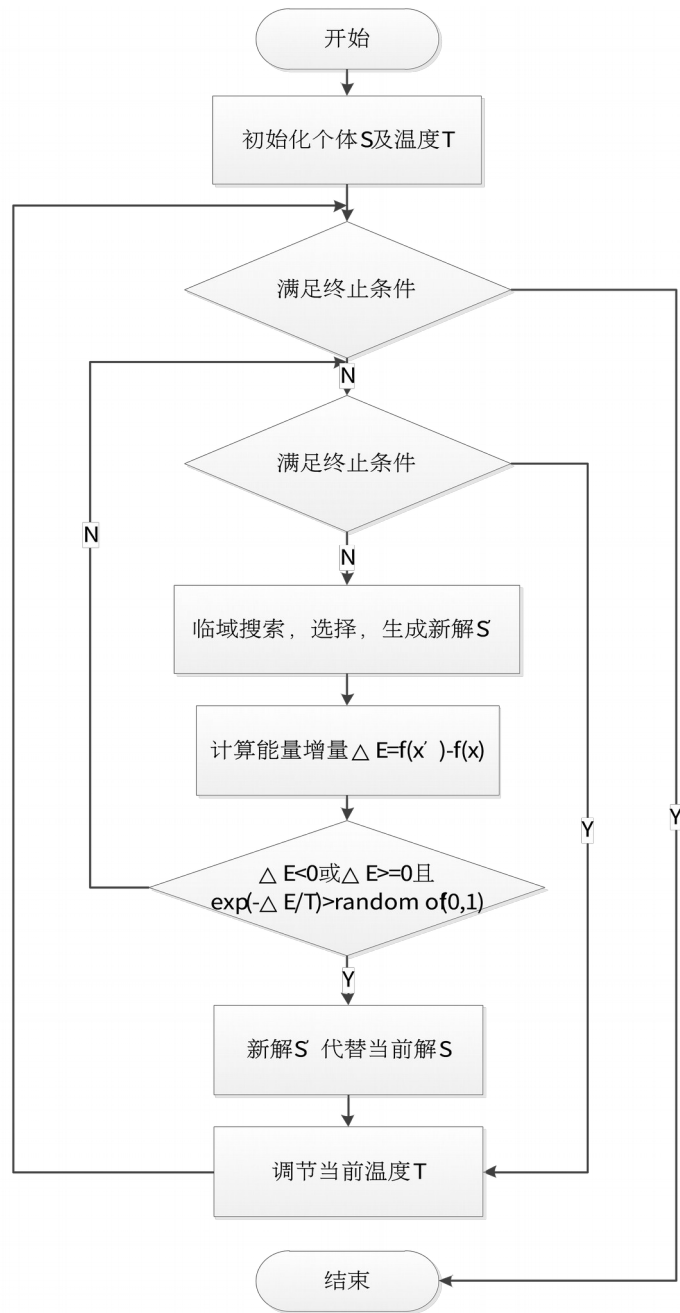


图4- 单一个体模拟退火搜索的流程图

4.1.3 加入局部搜索的遗传算法

爬山算法虽然具备较强的局部搜索能力，但不具备全局寻优能力，是否能

够搜索到全局极值点取决于初始个体的分布及所求问题的峰值数量。模拟退火算法虽然具备较强的局部搜索能力，同时也具备一定的全局寻优及适应性进化能力，但是其缺点也是显而易见的，其最主要的不足之处就是全局搜索能力不足，难以在较短时间内求得高质量的近似全局最优解。通过对局部搜索算法的分析可知在它们运行过程中个体都是独立对解空间进行搜索的，个体之间并不交换所求解到的信息，难以对解空间进行高效率的全局搜索，而遗传算法是一种群体智能算法，具备较强的全局搜索能力，但局部搜索能力不足，为提高对全局最优解的求解效率常将局部搜索算法与遗传算法相结合形成混合遗传算法，由于其运行机制基本一致，因此可以对加入局部搜索的遗传算法用一个基本框架来表示。

加入局部搜索的遗传算法基本思想为：在每次进化过程中，分别对种群个体进行局部搜索操作和遗传进化操作，也就是说将每次局部搜索操作得到的新种群作为遗传进化操作的初始种群，将每次遗传进化操作得到的新种群作为局部搜索操作的初始种群，如此循环操作直到满足终止条件。其中局部搜索操作就是对种群个体依次进行一定次数的局部寻优，如果寻找到更优个体则替换原个体并结束对原个体的寻优，否则继续寻找直到满足终止条件；其遗传进化操作就是对局部搜索生成的新种群进行选择、交叉、变异操作，从而生成下一代进化种群。

以下给出加入局部搜索的遗传算法具体流程描述：

Step1. 初始化种群规模、交叉率、变异率等参数，在解空间中利用随机法生成初始种群；

Step2. 判断当前遗传进化操作是否满足终止条件，如果不满足则跳转到 **Step4**；如果满足则继续向下执行；

Step3. 结束计算，选择出当前种群中适应度值最大个体，并将其作为最优解输出；

Step4. 计算种群个体的适应度；

Step5. 将当前进化种群作为局部搜索法开始的初始种群，遍历其中所有个体，并对每个个体进行一定次数的局部搜索操作；

Step6. 将局部搜索操作结束后所生成的种群作为遗传进化操作开始的初始种群，并对其执行选择、交叉、变异操作；

Step7. 将进化操作后形成的种群作为下一代种群，并跳转到 **Step2**。

以下给出加入局部搜索的遗传算法流程图，如图 4-4 所示。

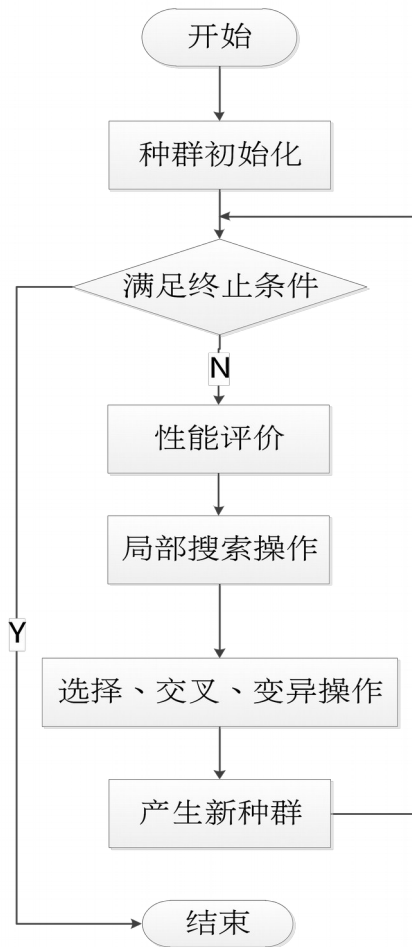


图 4- 加入局部搜索的遗传算法流程图

4.2 加入局部搜索的多目标优化算法

局部搜索算法可以利用已经搜索到的信息对原始解周围进行充分的搜索，在多目标优化算法中加入该方法可以有效提升寻优性能，但由于个体之间的优劣关系并不能由目标函数值大小直接判断得到，而是要依照个体之间支配关系进行判断，因此不能将局部搜索直接加入其中，而是需要做一定适当的改动。以下给出多目标优化算法中进行局部搜索所需依照的具体准则：

（1）选择标准

选择标准是指对多目标优化算法的种群个体进行局部搜索时如何选定进行搜索的个体。选择标准一般有两种：

第一种，只选定当前种群中的非支配解，根据进化理论，精英个体比其它个体具有更大概率进化成全局最优解，因此对种群中精英个体进行局部搜索可以指引种群的进化方向，以较大概率搜索到最优解；

第二种，对种群中所有解均进行局部搜索，该种方式在单目标优化算法中比较常见，该方法认为非精英解周围比较稀疏，具有较大概率取得比原始解更优的新解，通过提高种群整体水平提升运行性能。

通常认为第二种选择方式优于第一种方式，但是第二种选择方式往往对运行造成较大的开销，在实际应用中具体采用何种选择方式一般根据需求决定。

（2）邻域搜索标准

邻域搜索标准是指对于选定的个体如何在其邻域中选定生成的新个体，该标准一般分为两种：

第一种，将所选定个体的某范围值上所有个体均作为生成的新个体，该种方式计算复杂度随着问题变量维数的增长呈指数增加，对于有10个自变量的实数编码算法，每个个体邻域搜索得到的新个体为1024个，由于多目标优化算法中自变量维数一般较多，所以该种方式并不实用；

第二种，设定搜索次数上限值，在该搜索次数上限值内对选定个体进行邻域搜索，即在其临近可行解空间内随机生成新个体，如果新生成个体被接受，则终止对该个体的搜索，否则继续搜索直到搜索次数超出所设定的搜索次数上限值。

（3）接受标准

接受标准是指对于新生成的个体如何判断其是否优于原始个体，常用的接受标准一般分为两种：

第一种，判断所生成新个体是否被原个体支配，如果被支配则直接拒绝生成的新个体，否则接受其为当前解；

第二种，首先判断新个体是否被原个体所支配，如果被支配则直接拒绝生成的新个体，否则的话对其进行下一步判断，比如根据新个体在种群中所处的非支配排序分层序号、拥挤度、适应度等标准判断其是否劣于原个体，如果劣于原个体则拒绝，如果优于原个体则接受，如果两者间并不存在优劣关系则将新个体加入到进化种群，即扩大种群规模。

由以上对多目标优化算法中加入局部搜索的关键准则分析可知，其中单个个体进行局部搜索详细过程为：对于一个给定的种群 P ，按照设定的选择标准从待探索的个体集合中选择出进行局部搜索的个体 a ；按照设定的邻域搜索规则对选定个体进行局部搜索，将搜索到的邻域解作为候选解集 $\sigma(a)$ ，并从 $\sigma(a)$ 中选择随机选择出新个体 a' ；按照设定的接受标准判断新个体 a' 是

否被接受，接受则将新个体 a' 加入到未探索个体集合中；根据邻域搜索规则判断是否结束对个体 a 的局部搜索，如果结束对 a 的局部搜索则将其加入到已探索个体集合中，否则再次从候选解集 $\sigma(a)$ 中随机选择其他个体进行判断，如此往复直到未探索个体集合为空。

4.3 加入局部搜索的非支配排序遗传算法

在单目标优化问题中模拟退火算法和具有全局寻优能力的遗传算法能够很好结合，并且该种结合方式已经在很多具体应用领域中得到了验证。由于多目标优化算法中的局部搜索一般需要对选择标准、邻域搜索标准、接受标准有很好的限定，因此本文对具有随机寻优能力的模拟退火算法进行适当调整，并将其引入到 NSGA-II 中，提出了一种加入局部搜索的非支配排序遗传算法。

4.3.1 对模拟退火算法的调整

本文所提加入局部搜索的非支配排序遗传算法在 NSGA-II 基础之上，利用调整后的模拟退火算法对每一代进化生成的父代种群和子代种群个体进行退火搜索，从而提高 NSGA-II 的局部搜索能力。以下给出对模拟退火算法的具体调整：

首先以支配关系为个体优劣的评判标准，在搜索过程中只接受支配于原个体的新个体，也就是说在调整后的模拟退火搜索中并不以任何概率接受被原始解支配或与原始解非支配关系的新解；其次，根据当前进化代的退火温度动态调整退火搜索的范围，即采用一种基于退火温度的渐变调整方式动态调整每一代的搜索范围。以下给出调整后单个个体在一次进化代内进行局部搜索的基本步骤。

Step1. 设置运行初始参数，包括退火搜索起始温度 T_0 、调节系数 K 、当前搜索次数 w 及其上限值 W ，其中 $n=0$ ， $w=0$ ；

Step2. 在当前解 X 可行解空间中的某一邻近范围内采用随机搜索的方式得到一个新解 X' ， $X' = \max[\min[X + R * e, X_{\max}], X_{\min}]$ ， R 为维度为 V 的随机向量， V 为优化问题中的自变量个数， R 中每一列随机数取值范围为该列所对应的自变量取值范围的正负区间； X_{\max} 、 X_{\min} 为维度为 V 的向量，其每一列

取值分别为该列所对应的自变量的上、下限； e 为搜索半径调节系数， $e = T/K$ ， T 为当前进化代所对应的温度；

Step3. 计算新解 X' 各子目标函数值，并判断新解 X' 与原始解 X 之间的支配关系，若解 X' 支配于解 X ，则接受新解并用其替代当前解，即 $X = X'$ ；

Step4. $w = w + 1$ ，如果 $w < W$ ，则跳转到 Step2；否则，跳转到 Step5；

Step5. 根据传统的温度控制方式更新当前温度 T ，并结束此次进化代中对该个体的局部搜索。

由上可知，本文对传统模拟退火搜索算法的调整体现在 Step2 和 Step3 中。在 Step2 中采用了基于退火温度的渐变调整搜索范围的方式，由于退火温度是关于进化代数的变量，也就是说在进化过程初始时退火温度为最高值，此时退火搜索的范围最大，随着进化代数的增加退火温度逐渐下降，在进化过程结束时退火温度为最低值，此时退火搜索的范围最小，由此可见该种调整方式在本质上也是基于进化代数的。在 Step3 中只接受优于原个体的新解，由于在多目标优化问题中解的优劣是以支配关系作为评判标准的，难以对新解的劣化程度进行量化，并且遗传算法本身也是一种随机搜索算法，在搜索过程中能够接受劣于原始解的新解，因此在此步的调整中退火搜索并不以任何概率接受不优于原始解的新解。

根据上述对个体退火搜索过程的描述可知，个体进行退火搜索操作的时间复杂度为 $O(m * N)$ ，而 NSGA-II 的时间复杂度为 $O(m * N^2)$ ，其中 m 为待优化的多目标个数， N 为整体种群的个体数，所以加入退火搜索后时间复杂度为 $O(N) + O(m * N^2)$ ，即 $O(m * N^2)$ ，因此时间复杂度保持不变，在相同迭代次数下所提算法运行时间和 NSGA-II 相比不会有显著的增加。

4.3.2 加入局部搜索的非支配排序遗传算法

本文所提加入局部搜索的非支配排序遗传算法采用最大搜索次数的邻域搜索标准，利用调整后的模拟退火算法实现对 NSGA-II 进化过程中每一进化代的父代及子代种群个体进行退火搜索，从而实现对 NSGA-II 局部搜索能力增强的目的。以下给出所提算法的流程图，如图 4-5 所示。

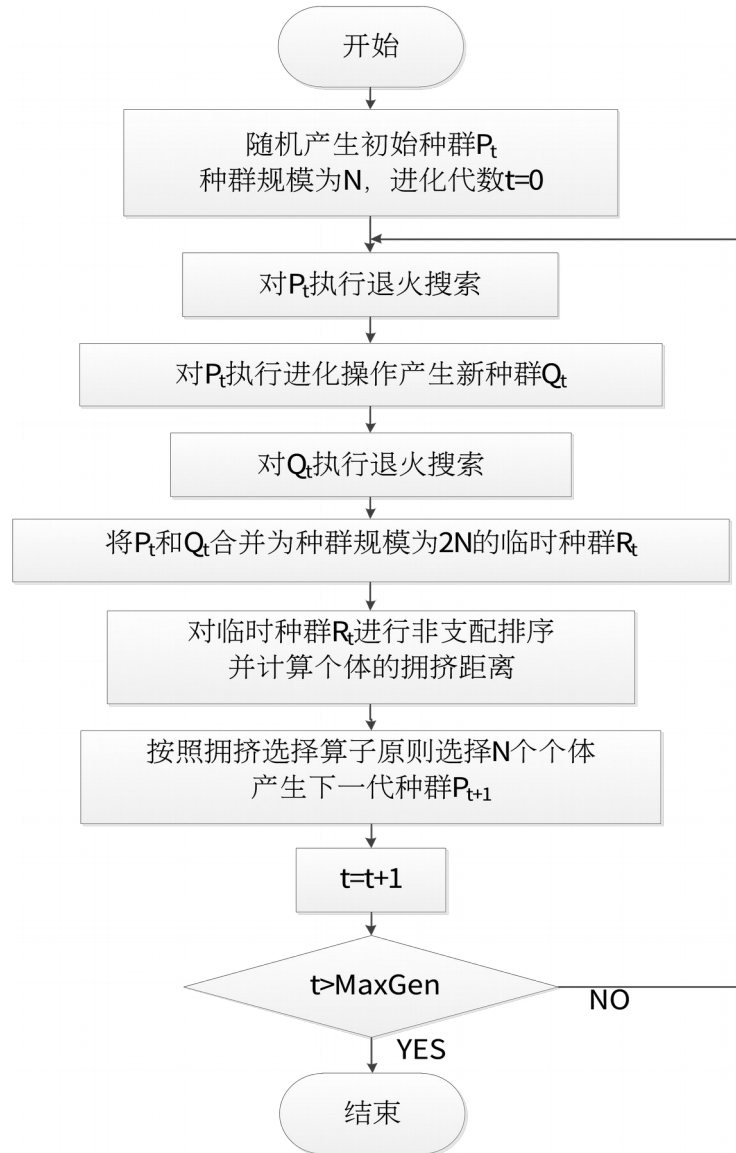


图 4- 加入局部搜索的非支配排序遗传算法流程图

4.4 加入局部搜索的自适应非支配排序遗传算法

当优化问题的解空间中不存在局部极值点或具有较少局部极值点, 比如单峰函数优化问题等, 该种情况下问题不具备模式欺骗性或具有较弱模式欺骗性, 随机采样误差是影响算法搜索到最优解的重要因素^[58, 59], 此时 NSGA-II 在全局求优和局部求优两方面上能够得到较好权衡; 当解空间中存在多个局部极值点,

比如多峰优化问题, 该种情况下所求解问题具有较强模式欺骗性^[60, 61], 此时 NSGA-II 难以在全局求优和局部求优两方面上得到很好权衡。为提高 NSGA-II 全局及局部寻优能力, 进一步提升其求解效率, 本文将第三章所提基于非支配排序分层的适应性策略和本章所提加入局部搜索的非支配排序遗传算法相融合, 给出了一种加入局部搜索的自适应非支配排序遗传算法。该算法可以在运行过程中根据外界环境自适应调整运行参数, 并采用模拟退火的方式对每一代进化生成的父代及子代种群进行局部搜索, 使 NSGA-II 在全局寻优和局部寻优两方面得到较好权衡, 有效提升其求解全局最优值的能力。

本文所提加入局部搜索的自适应非支配排序遗传算法基本流程如图 4-6 所示, 其具体描述为:

Step1. 对运行参数, 交叉率、变异率、交叉分布指数、变异分布指数、退火搜索初始温度、局部搜索的最大内循环次数及调节系数进行初始化;

Step2. 在解空间内随机生成个体数为 N 的初始种群 P_t , $t=0$, 将其作为父代种群, 并对 P_t 进行非支配排序, 根据分层结果计算出每个个体所对应的交叉率、变异率、交叉分布指数、变异分布指数;

Step3. 遍历父代种群 P_t 中所有个体, 对其进行退火搜索, 并根据接收准则更新种群 P_t ;

Step4. 对父代种群 P_t 进行二元竞赛选择、自适应模拟二进制交叉、自适应多项式变异操作, 产生种群 Q_t , 并将 Q_t 作为子代种群;

Step5. 遍历子代种群 Q_t 中所有个体, 对其进行退火搜索, 并根据接收准则更新种群 Q_t ;

Step6. 将 P_t 和 Q_t 融合在一起, 作为规模为 $2N$ 的临时种群 R_t , 对其进行快速非支配排序和拥挤度计算, 根据个体的非支配排序分层序号和个体的拥挤距离选择出最优的 N 个个体作为下一代进化操作中的父代种群 P_t , $t=t+1$;

Step7. 根据非支配排序分层结果计算出 P_t 中每个个体所对应的交叉率、变异率、交叉分布指数、变异分布指数;

Step8. 根据温度控制方式更新当前温度;

Step9. 判断迭代次数是否等于预设上限值, 如果等于则结束运行; 否则, 跳转到 Step3。

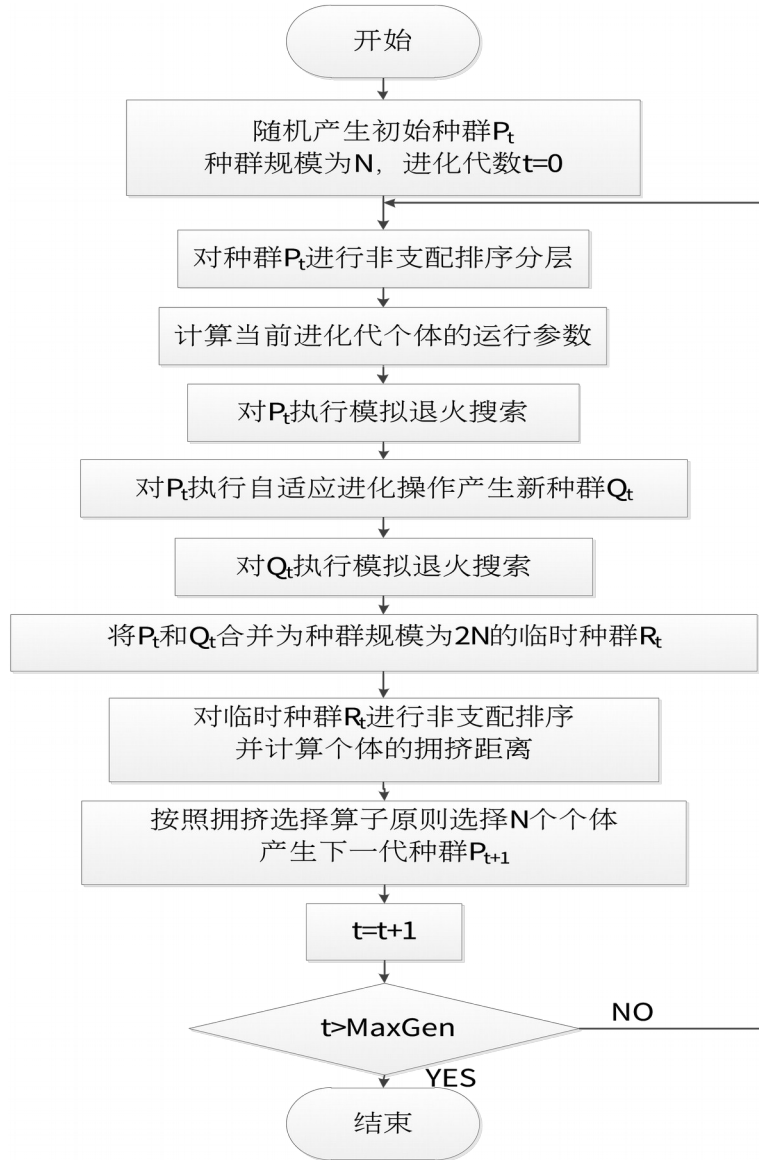


图4- 加入局部搜索的自适应非支配排序遗传算法流程图

4.5 本章小结

NSGA-II 全局最优值求解性能由其全局寻优能力及局部寻优能力所决定, 在其运行的前期阶段主要是对解空间的全局搜索, 由于其较强的全局搜索能力, 能较快定位到最优解集的大致位置; 在运行后期阶段主要是对解空间的局部搜索, 由于其局部搜索能力的不足, 往往进化迟缓, 难以搜索到问题的真实解集。

对于模式欺骗性较大的问题，NSGA-II 难以在全局寻优和局部寻优两方面上进行很好的权衡，而其最重要的原因就是局部寻优能力较弱，从而使其难以在较短时间内搜索到全局最优解集。本章首先针对 NSGA-II 局部搜索能力较弱的问题，对传统模拟退火算法进行适当调整，并将其引入到 NSGA-II 中，给出了一种加入局部搜索的非支配排序遗传算法；然后将其与第三章所提策略相结合，给出了一种加入局部搜索的自适应非支配排序遗传算法，通过这种“强强联合”的方式在全局优化和局部优化两方面提高 NSGA-II 求解性能。

第 5 章 实验与分析

前文首先针对 NSGA-II 在运行时不能根据外界环境变化自适应调整运行参数，从而造成搜索效率较低的问题，提出了一种基于非支配排序分层的适应性策略，该策略通过在 NSGA-II 运行过程中自动调整运行参数，从而影响交叉算子及变异算子的进化表现，提高其搜索效率；其次，针对 NSGA-II 局部寻优能力不足的问题，对传统模拟退火算法进行适当调整，并将其引入到 NSGA-II 中，给出了一种加入局部搜索的非支配排序遗传算法，通过对每一进化代内的父代种群及子代种群进行局部搜索，从而提高 NSGA-II 的局部搜索能力；最后将加入局部搜索的非支配排序遗传算法与第一阶段所提适应性策略相结合，给出了一种加入局部搜索的自适应非支配排序遗传算法，在全局优化和局部优化两方面提高了原算法的搜索效率。

5.1 实验准备

Python 语言是 1989 年被研发出来的一种融合面向过程、面向对象、函数式编程思想的高级语言，它具备语法简单、易于理解、快速编程等优点。Numpy 是 Python 的一种开源扩展框架，其以高效的运行效率而著称，适用于处理大型矩阵运算任务，在某种意义上来说 Python 与 Numpy 的结合构建成了一个更加强化的 Matlab 系统^[62]。R 语言诞生于 1980 年前后，属于 S 语言的一个分支，是统计领域所广泛使用的一门可编程语言，它作为一种统计分析语言集统计分析与图形显示于一体，常被用来进行数据探索、统计分析及作图。本文所做实验采用 Python2.7 和 Numpy 进行编程，可视化作图采用 R 语言，操作系统为 Linux 中的 Ubuntu14.04 64 位桌面操作系统，编辑器采用 Vim，其硬件配置为装备有 Intel I5 4250 型号 CPU 和 12G 内存的 Lenovo 品牌微型计算机。

5.2 模拟仿真测试函数

本文实验采用 ZDT 测试函数集^[41]中关于连续型优化问题的 ZDT1-ZDT4、ZDT6 测试函数，以下给出这五个多目标测试函数的数学形式、变量取值范围，如表 5-1 所示，以及这五个测试函数的真实前沿面示意图^[63]，如图 5-1 所示。

表 5- ZDT 测试函数集

函数名称	目标函数	变量范围
ZDT1	$f_1(x) = x_1$ $f_2(x) = g(x) \sqrt{\frac{f_1}{g(x)}}$ $g(x) = 1 + \frac{9}{n-1} \sum_{i=2}^n x_i$	$x_i \in [0, 1],$ $i = 1, 2, \dots, n,$ $n = 30.$
ZDT2	$f_1(x) = x_1$ $f_2(x) = g(x) \sqrt{\frac{f_1}{g(x)}}$ $g(x) = 1 + \frac{9}{n-1} \sum_{i=2}^n x_i$	$x_i \in [0, 1],$ $i = 1, 2, \dots, n,$ $n = 30.$
ZDT3	$f_1(x) = x_1$ $f_2(x) = g(x) \sqrt{\frac{f_1}{g(x)}} - \frac{f_1}{g(x)} \sin(10\pi x_1)$ $g(x) = 1 + \frac{9}{n-1} \sum_{i=2}^n x_i$	$x_i \in [0, 1],$ $i = 1, 2, \dots, n,$ $n = 30.$
ZDT4	$f_1(x) = x_1$ $f_2(x) = g(x) \sqrt{\frac{f_1}{g(x)}}$ $g(x) = 1 + 10(n-1) + \sum_{i=2}^n (x_i^2 - 10 \cos(4\pi x_i))$	$x_1 \in [0, 1]$ $x_i \in [-5, 5],$ $i = 2, 3, \dots, n,$ $n = 10.$
ZDT6	$f_1(x) = 1 - \exp(-4x_1) \sin^6(6\pi x_1)$ $f_2(x) = g(x) \sqrt{\frac{f_1}{g(x)}}$ $g(x) = 1 + 9 \frac{\left(\sum_{i=2}^n x_i \right)^{0.25}}{n-1}$	$x_i \in [0, 1],$ $i = 1, 2, \dots, n,$ $n = 10.$

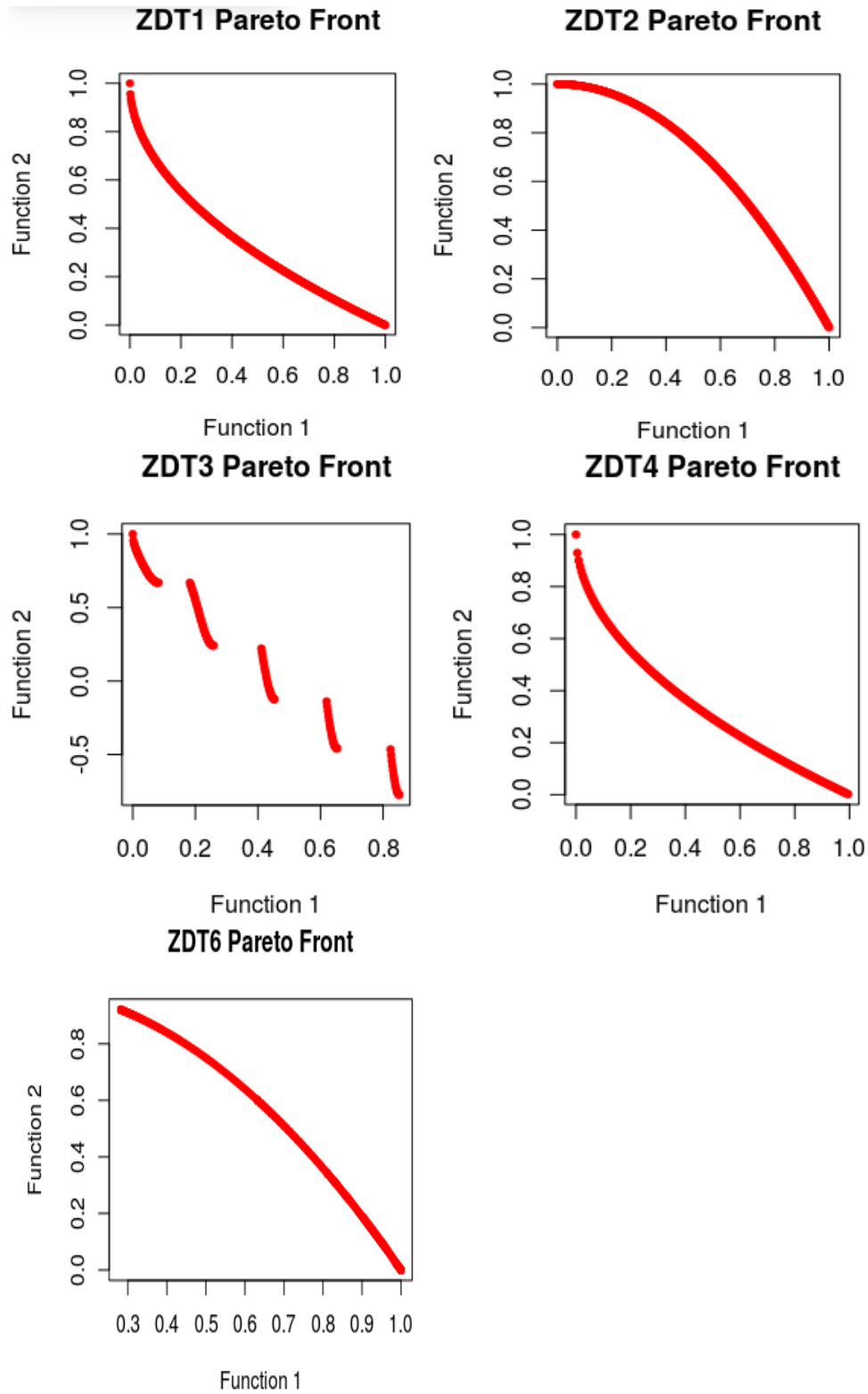


图5- ZDT1-ZDT4、ZDT6 Pareto 前沿面示意图

其中，测试函数 ZDT1 的 Pareto 最优前沿面是凸型的；测试函数 ZDT2 的 Pareto 最优前沿面是非凸型的；测试函数 ZDT3 的 Pareto 最优前沿面是离散型的，由若干个非连续的凸区域所构成；ZDT4 测试函数具有 794280046581 个局部最优前沿面，越靠近其局部最优前沿面个体分布越密集，常被用来测试算法处理多峰问题的能力；ZDT6 测试函数最优前沿面为非凸型、非均匀的，越靠近 Pareto 最优前沿面个体分布越稀疏，越远离 Pareto 最优前沿面个体分布越密集，常被用来测试算法维持种群多样性的能力。ZDT 系列的多目标测试函数都是关于两目标优化问题的，其变量个数能够以任意规模进行扩展，而且它们的真实最优解集位置及形态都是已知的，大量多目标优化算法采用它们作为性能优化的测试函数，所以使用它们作为本文优化改进算法的测试函数可使实验结果具有较好的可对比性。

5.3 性能对比实验

本文通过模拟仿真实验对所提改进进行验证。实验分为三部分：第一部分通过实验对本文所提基于非支配排序分层的适应性策略进行性能验证；第二部分通过实验对本文所提加入局部搜索的非支配排序遗传算法进行性能验证；第三部分通过实验对融合本文所提基于非支配排序分层的适应性策略和加入局部搜索的非支配排序遗传算法的综合算法，即加入局部搜索的自适应非支配排序遗传算法进行性能验证。

5.3.1 基于非支配排序分层的适应性策略性能对比实验

本部分实验对 NSGA-II、基于进化阶段的适应性 NSGA-II、基于进化代数的线性渐变 NSGA-II^[33]以及基于本文所提适应性策略的 NSGA-II 进行优化性能比较。其中，NSGA-II 中交叉率为 0.8，变异率为 $1/n$ ， n 为变量个数，变异分布指数为 20，交叉分布指数为 20；基于进化阶段的适应性 NSGA-II 交叉率及变异率与文献[43]中所给出的设置相同，交叉分布指数及变异分布指数均为 20，在下文中简称其为对比算法 1；基于进化代数的线性渐变 NSGA-II 中全局最小及最大交叉分布指数分别为 1 和 30，变异分布指数为 20，交叉率为 0.8，变异率为 $1/n$ ， n 为变量个数，在下文中简称其为对比算法 2；基于本文所提适应性策略的 NSGA-II 中全局最大交叉分布指数及变异分布指数均为 30，全

局最小交叉分布指数及变异分布指数均为 1，其交叉率及变异率的相关设置采用第三章所给出的推荐值，在下文中简称其为改进算法 1。本部分实验的进化代数数为 500，种群规模为 200，对每个算法均进行 100 次重复实验，ZDT1-ZDT4 临时种群中非支配个体数变化趋势如图 5-2 所示，ZDT6 临时种群中非支配个体数变化趋势如图 5-3 所示，收敛性准则 γ 的值如表 5-2 所示，多样性准则 Δ 的值如表 5-3 所示。

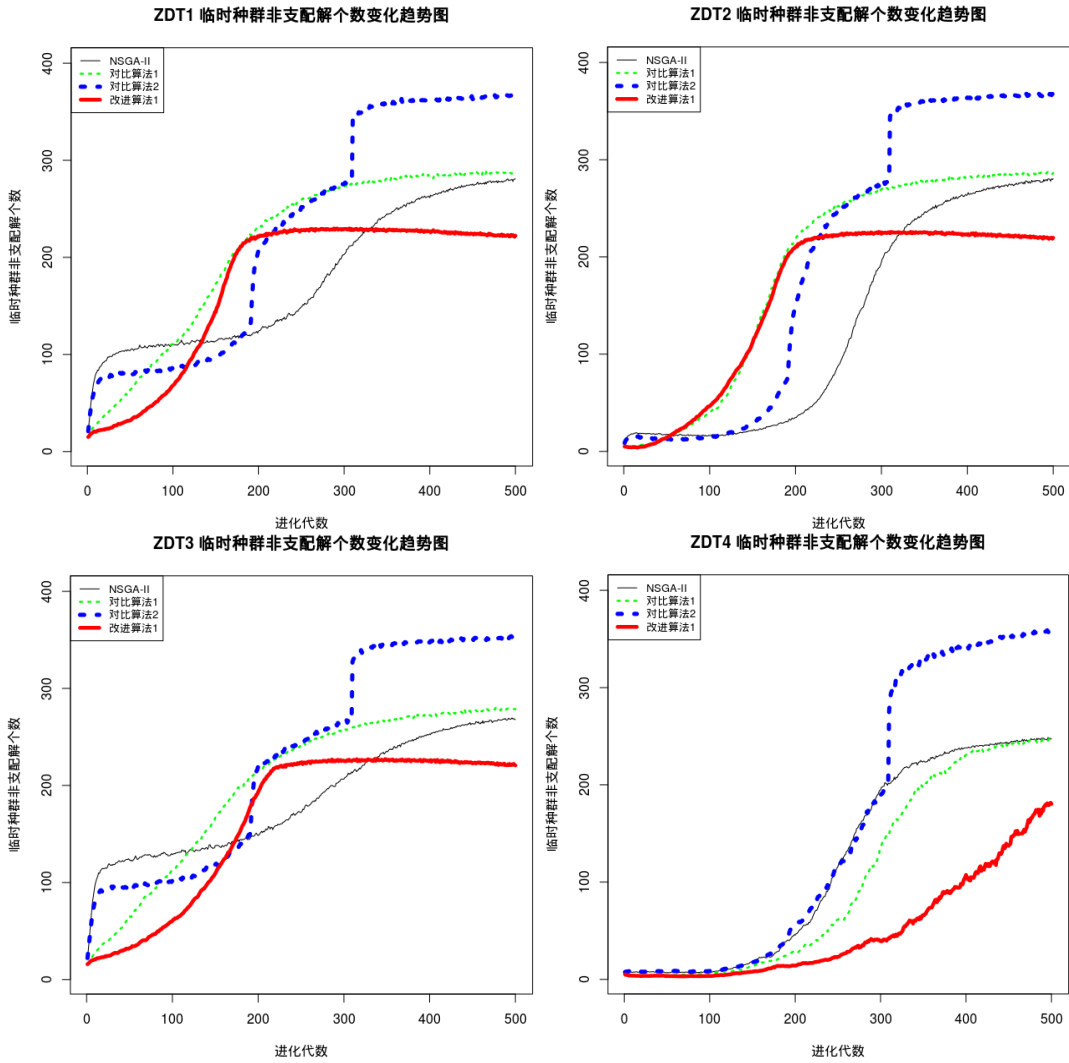


图 5- ZDT1-ZDT4临时种群非支配个体数变化趋势图

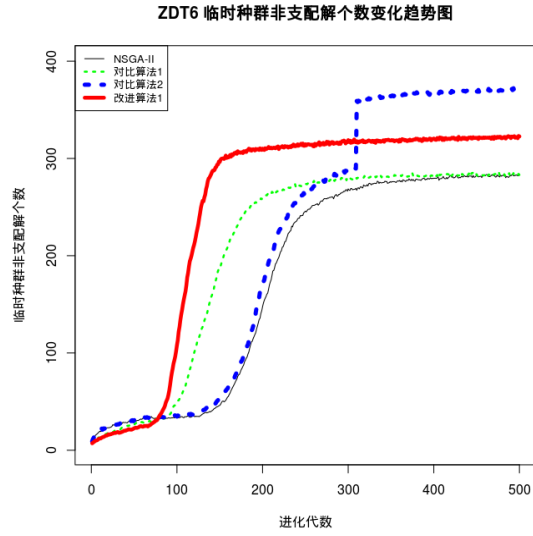


图 5- ZDT6 临时种群非支配个体数变化趋势图

由图 5-2 及图 5-3 可知，本文所提基于非支配排序分层的适应性策略能有效提高算法的收敛速度，基于所提策略的 NSGA-II，即改进算法 1，在 ZDT1、ZDT2、ZDT3 运行前期临时种群非支配个体数增长较为缓慢，在运行中期增长较为迅速，与 NSGA-II、对比算法 1、对比算法 2 相比能更快到达稳定的水平值，也就是说改进算法 1 在运行前期收敛较慢，有效延长探索阶段，能更好的对全局进行搜索，在运行中期能迅速的从全局搜索转为局部搜索，整体收敛速度更快，在运行过程中具有更好的适应性。在 ZDT6 中改进算法 1 收敛速度在运行到 100 代左右时便迅速成为所有对比算法中最快的，并最早到达稳定值，但这同时也使其种群多样性过早降低到最低水平，影响其后期进化。在 ZDT1、ZDT2、ZDT3 中改进算法 1 临时种群非支配个体数的稳定值为对比算法中最小值并略大于种群规模，也就是说改进算法 1 在这三个测试函数的进化后期比其它对比算法具备更强的进化能力。在 ZDT4 中，改进算法 1 收敛速度较慢，在整个进化过程中其临时种群的非支配个体数都在逐渐增加并小于其它对比算法，在算法结束时其最终个数仍是对比算法中最小值，并且小于种群规模，由此可知，在完全欺骗性的多峰优化问题中改进算法 1 更加容易掉入到局部最优解集的陷阱中，其进化能力在整个进化过程中都为所有对比算法中最差的，在算法结束时其进化能力仍在继续攀升，并没有趋于平稳值。

由表 5-2 可知，本文所提基于非支配排序分层的适应性策略能有效提高 NSGA-II 的收敛性，其收敛性均值在 ZDT1、ZDT2、ZDT3、ZDT6 上均为最优，其收敛性方差在 ZDT1、ZDT2、ZDT3 均为最优，在 ZDT6 上略低于 NSGA-II。由实验结果可知本文所提策略不仅能有效提升算法收敛性、抑制早熟，并且鲁棒性更强，在 ZDT4 上优化结果没有得到改进反而得到了劣化，这一点表明基

于适应性策略的 NSGA-II 与单目标优化中适应性策略在完全欺骗性的多峰问题上使算法优化结果劣化的特点相同，通过此实验可以充分表明基于交叉算子及变异算子的适应性策略同样适应于 NSGA-II。

表 5-收敛性准则 γ

函数名称	算法名称	均值	方差
ZDT1	NSGA-II	0.005509	0.000774
	对比算法 1	0.002710	0.000263
	对比算法 2	0.008079	0.001609
	改进算法 1	0.002250	0.000143
ZDT2	NSGA-II	0.004246	0.001206
	对比算法 1	0.002104	0.000303
	对比算法 2	0.007136	0.002053
	改进算法 1	0.001767	0.000168
ZDT3	NSGA-II	0.006160	0.000481
	对比算法 1	0.005153	0.000168
	对比算法 2	0.007335	0.000922
	改进算法 1	0.004960	0.000138
ZDT4	NSGA-II	0.007830	0.002921
	对比算法 1	0.010174	0.004166
	对比算法 2	0.016000	0.018065
	改进算法 1	0.369046	0.266799
ZDT6	NSGA-II	0.029176	0.013015
	对比算法 1	0.030542	0.017539
	对比算法 2	0.038937	0.015774
	改进算法 1	0.017943	0.013988

由表 5-3 可知，本文所提基于非支配排序分层的适应性策略能有效提高 NSGA-II 的多样性，改进算法 1 多样性均值在 ZDT1、ZDT2、ZDT3、ZDT6 上均为最优，多样性方差在 ZDT1、ZDT2、ZDT3 均为最优，在 ZDT6 上略低于

NSGA-II。由实验结果可知本文所提策略不仅能有效提升算法的种群多样性，并且鲁棒性更强，在 ZDT4 上优化结果没有得到改进反而得到了劣化，这一点与该策略在收敛性上表现所一致，也就是说基于适应性策略的 NSGA-II 与单目标优化中基于适应性策略的遗传算法在完全欺骗性的多峰问题上使算法优化结果劣化的特点相同，通过此实验可以充分表明基于交叉算子及变异算子的适应性进化策略能够有效改进 NSGA-II 的种群多样性。

表 5- 多样性准则 Δ

函数名称	算法名称	均值	方差
ZDT1	NSGA-II	0.008439	0.001730
	对比算法 1	0.002630	0.000644
	对比算法 2	0.014084	0.003459
	改进算法 1	0.001303	0.000418
ZDT2	NSGA-II	0.006953	0.002496
	对比算法 1	0.002379	0.000672
	对比算法 2	0.013607	0.004415
	改进算法 1	0.001401	0.000459
ZDT3	NSGA-II	0.020961	0.031298
	对比算法 1	0.004875	0.001399
	对比算法 2	0.037769	0.041326
	改进算法 1	0.002336	0.000664
ZDT4	NSGA-II	0.014611	0.005809
	对比算法 1	0.021191	0.013193
	对比算法 2	0.029337	0.035741
	改进算法 1	0.316590	0.144342
ZDT6	NSGA-II	0.132053	0.062484
	对比算法 1	0.138085	0.071950
	对比算法 2	0.142931	0.066603
	改进算法 1	0.101250	0.065495

5.3.2 加入局部搜索的非支配排序遗传算法性能对比实验

本部分实验对 NSGA-II 和本文所提引入模拟退火算法的 NSGA-II，即加入局部搜索的非支配排序遗传算法进行性能对比。其中，NSGA-II 中交叉率为 0.8，变异率为 $1/n$ ， n 为变量个数，变异分布指数为 20，交叉分布指数为 20；本文所提加入局部搜索的非支配排序遗传算法中交叉率为 0.8，变异率为 $1/n$ ， n 为变量个数，变异分布指数为 20，交叉分布指数为 20，每代进化中单个个体的退火搜索次数为 5，降温方式为按比例降温，初始温度为 1000，调节系数为 3000，在下文中简称其为改进算法 2。本部分实验种群规模为 200，对每个算法均进行进化代数为 100 和 250 的 100 次重复实验，在进化代数为 100 的实验中改进算法 2 的降温比例为 0.95，在进化代数为 250 的实验中改进算法 2 的降温比例为 0.98，进化代数为 250 的实验中 ZDT1、ZDT2 临时种群非支配个体数变化趋势如图 5-4 所示，进化代数为 250 的实验中 ZDT3、ZDT4、ZDT6 临时种群非支配个体数变化趋势如图 5-5 所示，收敛性准则 γ 的值如表 5-4 所示，多样性准则 Δ 的值如表 5-5 所示。

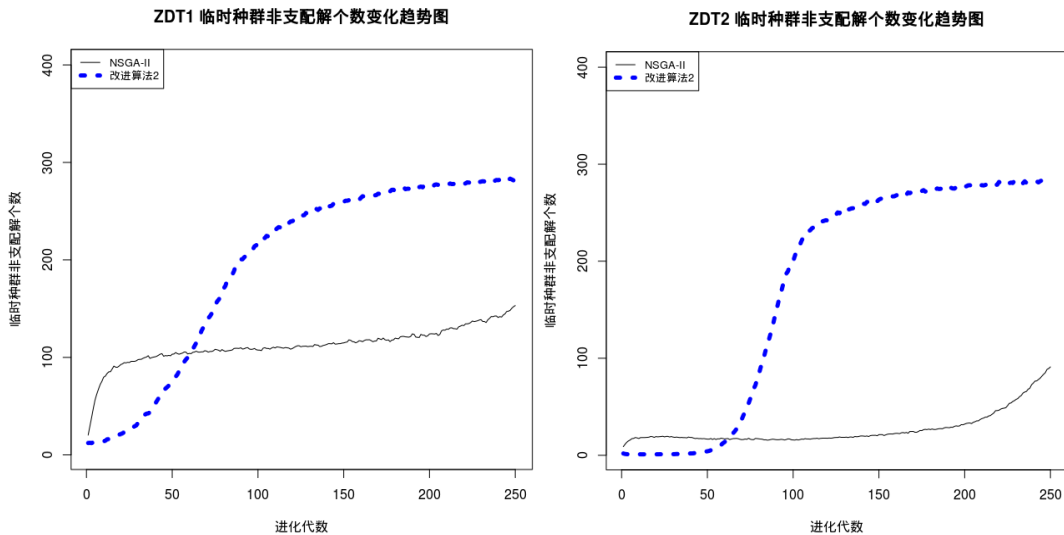


图 5- ZDT1 、 ZDT2 临时种群非支配个体数变化趋势图

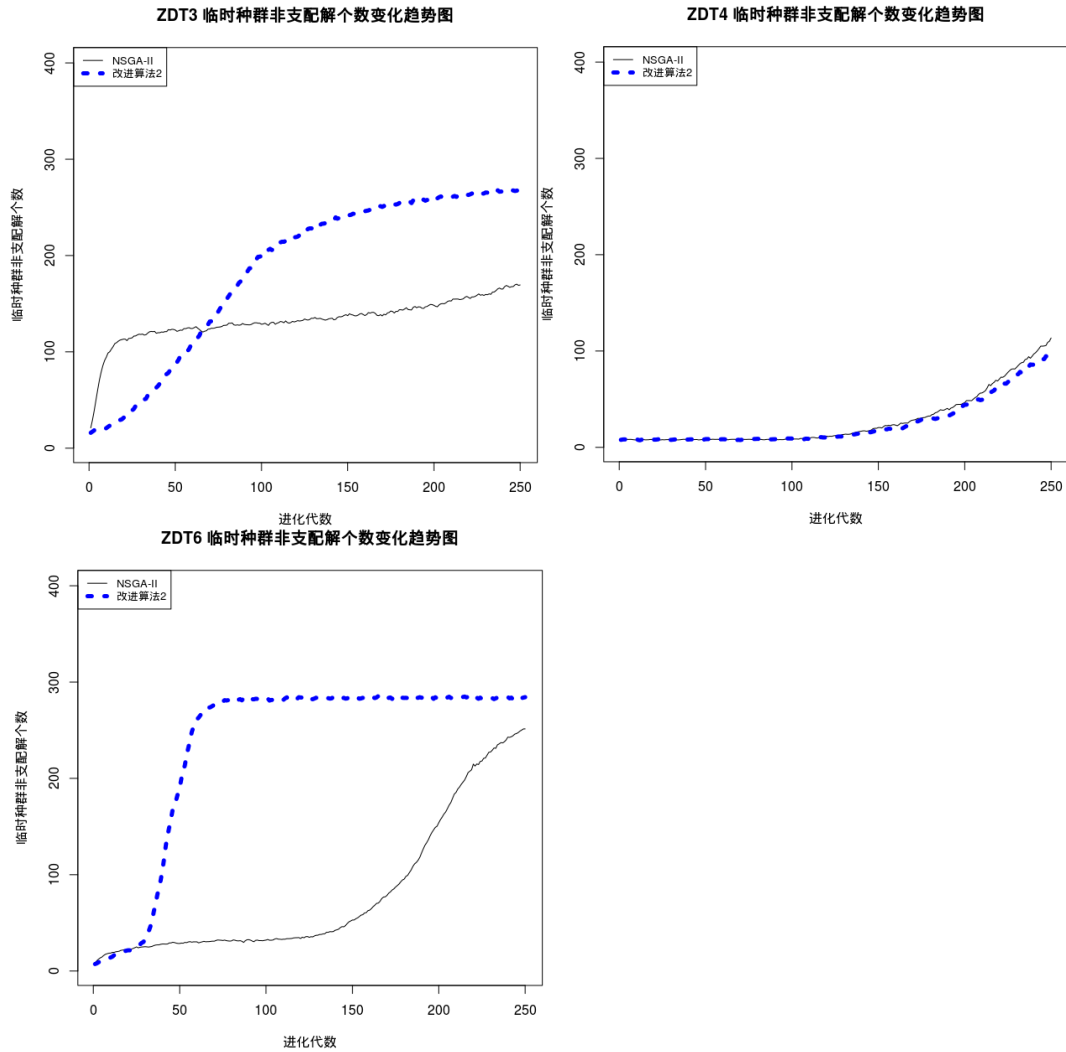


图5- ZDT3 、 ZDT4 、 ZDT6 临时种群非支配个体数变化趋势图

由图 5-4 及图 5-5 可知, 本文所提加入局部搜索的非支配排序遗传算法, 在 ZDT1-ZDT3 运行初期临时种群非支配个体数增长速度慢于原算法, 但之后迅速增加并超过原算法, 最终趋于的稳定值均高于原始算法并超过种群规模, 也就是说所提算法通过提高局部搜索能力能有效提高原始算法的收敛速度。在 ZDT4 中, 所提算法临时种群非支配个体数与原始算法相当, 并且在后期阶段略低于原始算法, 收敛速度并没有明显提升, 甚至略低于原始算法。在 ZDT6 中所提算法运行初期临时种群非支配个体数与原始算法相当, 但大致在30代之后便迅速增长, 并远高于原始算法, 虽然所提算法在该测试函数中收敛速度远高于原始算法, 但同时也导致其种群多样性过早降低, 影响其后期进化。

由表 5-4 可知, 本文所提加入局部搜索的非支配排序遗传算法, 即改进算法 2, 在进化代数为 100 的实验中收敛性均优于原算法, 在进化代数为 250 的

实验中其收敛性均值在 ZDT1-ZDT3 上均为最优，由此可见本文所提改进算法的确能有效抑制算法早熟并提高收敛性。在进化代数为 250 的实验中改进算法收敛性均值在 ZDT4、ZDT6 上低于原始算法，由其在 ZDT4 上的表现可知局部搜索能力的增强对完全欺骗型问题改进效果并不明显，甚至由于局部搜索能力的增强使其种群多样性迅速降低导致其在 ZDT6 上优化能力远低于原始算法。

表 5- 收敛性准则 γ

函数名称	进化代数	算法名称	均值	方差
ZDT1	100	NSGA-II	1.144095	0.117542
		改进算法 2	0.031288	0.010718
	250	NSGA-II	0.212495	0.046981
		改进算法 2	0.003635	0.000391
ZDT2	100	NSGA-II	1.631297	0.180385
		改进算法 2	0.014993	0.005286
	250	NSGA-II	0.189311	0.098311
		改进算法 2	0.002556	0.000502
ZDT3	100	NSGA-II	1.041614	0.104914
		改进算法 2	0.042802	0.028804
	250	NSGA-II	0.183265	0.045652
		改进算法 2	0.005731	0.000567
ZDT4	100	NSGA-II	5.693505	2.696327
		改进算法 2	4.995896	2.473097
	250	NSGA-II	0.207390	0.194829
		改进算法 2	0.233509	0.208237
ZDT6	100	NSGA-II	3.036837	0.599531
		改进算法 2	0.158669	0.061855
	250	NSGA-II	0.065221	0.025410
		改进算法 2	0.114944	0.063397

由表 5-5 可知，本文所提加入局部搜索的非支配排序遗传算法，即改进算

法 2，在进化代数为 100 的实验中其多样性均值在 ZDT1、ZDT2、ZDT3、ZDT6 上均高于原算法，在进化代数为 250 的实验中其多样性均值在 ZDT1、ZDT2、ZDT3 上均高于原算法，由此可见本文所提改进算法的确能有效提高原算法种群多样性。所提改进算法在 ZDT4、ZDT6 上多样性并没有得到显著提升甚至严重劣化，这一点与其在收敛性上的表现相同，也就是说在种群多样性方面局部搜索能力的增强对完全欺骗型问题改进效果并不明显，甚至由于局部搜索能力的增强使其种群多样性迅速降低，导致其在 ZDT6 上优化能力远低于原始算法。

表 5- 多样性准则 Δ

函数名称	进化代数	算法名称	均值	方差
ZDT1	100	NSGA-II	0.582560	0.021717
		改进算法 2	0.119911	0.072959
	250	NSGA-II	0.268153	0.036708
		改进算法 2	0.004652	0.000909
ZDT2	100	NSGA-II	1.631297	0.180387
		改进算法 2	0.299267	0.167493
	250	NSGA-II	0.410788	0.170272
		改进算法 2	0.003816	0.001290
ZDT3	100	NSGA-II	0.563590	0.032728
		改进算法 2	0.283417	0.082654
	250	NSGA-II	0.299531	0.057283
		改进算法 2	0.031249	0.049004
ZDT4	100	NSGA-II	0.827306	0.098405
		改进算法 2	0.841368	0.100769
	250	NSGA-II	0.451607	0.148960
		改进算法 2	0.483313	0.138815
ZDT6	100	NSGA-II	0.509922	0.076939
		改进算法 2	0.307920	0.089915

	250	NSGA-II	0.164245	0.058236
		改进算法 2	0.267513	0.092900

5.3.3 加入局部搜索的自适应非支配排序遗传算法性能对比实验

本部分实验对 NSGA-II 与融合本文所提基于非支配排序分层的适应性策略和加入局部搜索的非支配遗传算法的综合算法，即加入局部搜索的自适应非支配排序遗传算法进行性能对比。其中，NSGA-II 中交叉率为 0.8，变异率为 $1/n$ ， n 为变量个数，变异分布指数为 20，交叉分布指数为 20；本文所提加入局部搜索的自适应非支配排序遗传算法中交叉率、变异率、交叉分布指数、变异分布指数的预设值采用第三章中的推荐值，每代进化中单个个体的退火搜索次数为 10，降温方式为按比例降温，降温比例为 0.99，初始温度为 1000，调节系数为 3000，在下文中简称其为改进算法 3。本部分实验种群规模为 200，进化代数数为 500，对每个算法均进行 100 次重复实验，ZDT1、ZDT2 临时种群非支配个体数变化趋势如图 5-6 所示，ZDT3、ZDT4、ZDT6 临时种群非支配个体数变化趋势如图 5-7 所示，收敛性准则 γ 的值如表 5-6 所示，多样性准则 Δ 的值如表 5-7 所示。

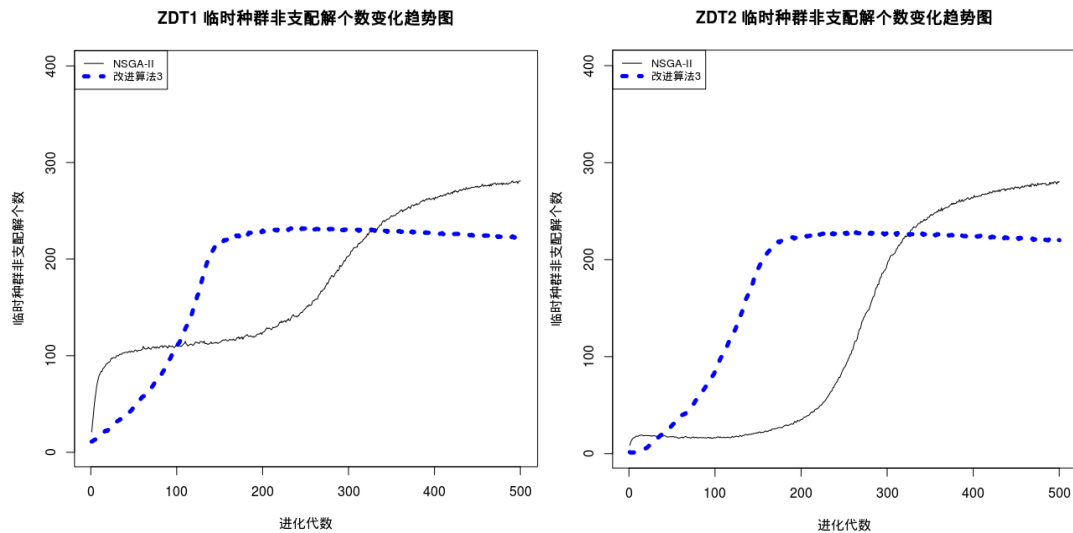


图 5- ZDT1 、 ZDT2 临时种群非支配个体数变化趋势图

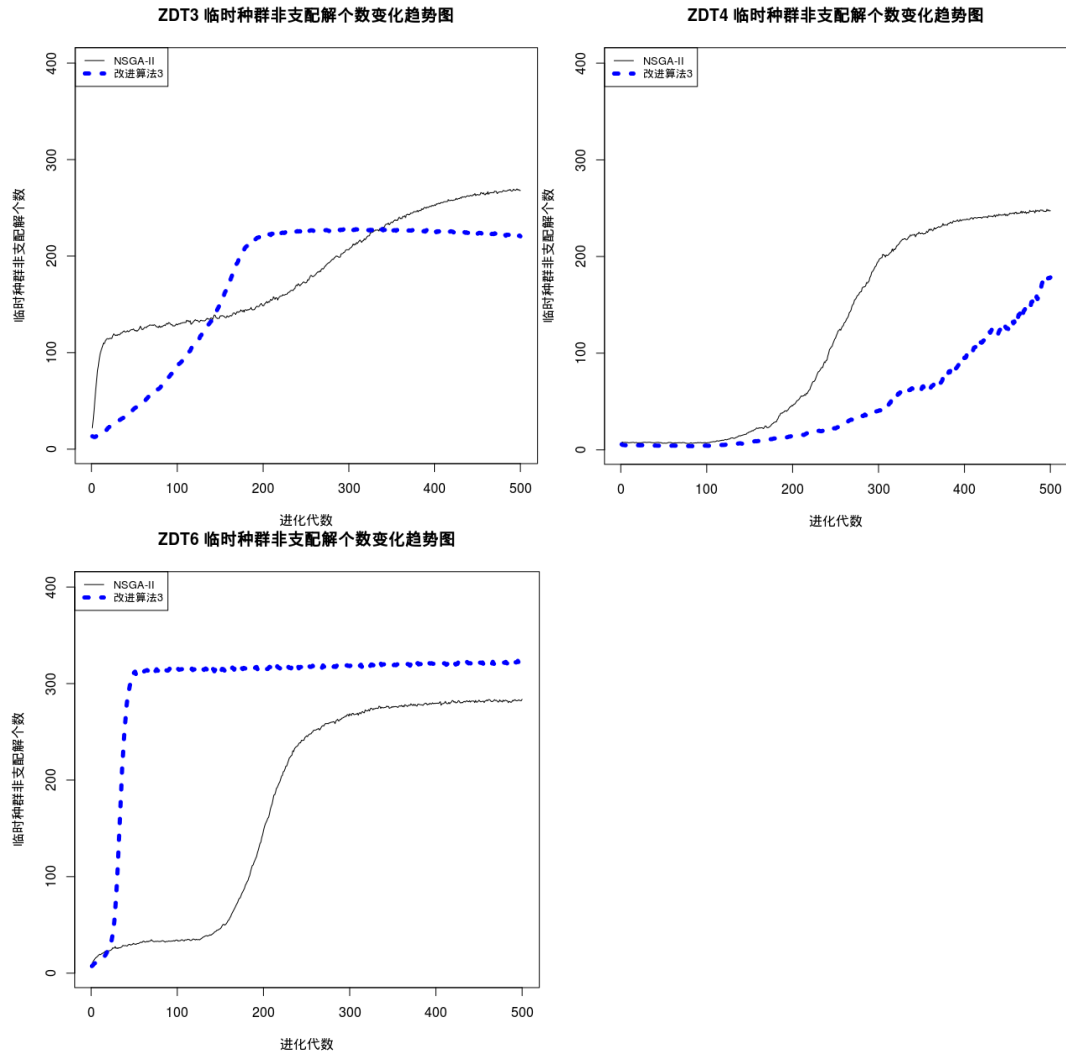


图5- ZDT3 、 ZDT4 、 ZDT6 临时种群非支配个体数变化趋势图

由图 5-6 及图 5-7 可知，本文所提加入局部搜索的自适应非支配排序遗传算法，即改进算法 3，在 ZDT1-ZDT3 运行初期临时种群非支配个体数增长速度慢于原算法，但之后迅速增加并超过原算法，最终趋于的稳定值均高于原始算法并超过种群规模，也就是说所提算法能运行初期能够延长探索阶段，并在运行中期从全局搜索迅速转为局部搜索，在进化的整体过程中能有效提高原始算法的收敛速度。在 ZDT4 中，所提算法临时种群非支配个体数一直低于原始算法及种群规模，收敛速度慢于原始算法，进化能力得到劣化。在 ZDT6 中所提算法运行初期临时种群非支配个体数迅速上升达到稳定值并远高于种群规模，虽然所提算法在该测试函数中收敛速度远高于原始算法，但同时也导致其种群多样性过早降低，影响其后期进化。

由表 5-6 可知，本文所提加入局部搜索的自适应非支配排序遗传算法，即

改进算法 3，其收敛性均值在 ZDT1-ZDT3 上均为最优，并高于表 5-2 中所给出的改进算法 1，由此可见本文所提改进算法 3 很好融合了本文所提基于非支配排序分层的适应性策略及加入局部搜索的非支配排序遗传算法，通过提高算法进化适应性及局部优化能力有效抑制算法早熟并提高收敛性，从而提升原算法求解全局最优解的能力。改进算法 3 收敛性均值在 ZDT4、ZDT6 上低于原始算法及改进算法 1，由此可知改进算法 3 通过将适应性策略和加入局部搜索的改进 NSGA-II 相结合，放大了两者优点的同时也放到了其缺点，即对完全欺骗型问题和对多样性敏感的问题得到了劣化效果。

表 5- 收敛性准则 γ

函数名称	算法名称	均值	方差
ZDT1	NSGA-II	0.005509	0.000774
	改进算法 3	0.002061	0.000133
ZDT2	NSGA-II	0.004246	0.001206
	改进算法 3	0.001538	0.000128
ZDT3	NSGA-II	0.006160	0.000481
	改进算法 3	0.004871	0.000146
ZDT4	NSGA-II	0.007830	0.002921
	改进算法 3	0.413936	0.319085
ZDT6	NSGA-II	0.029176	0.013015
	改进算法 3	0.040160	0.025827

由表 5-7 可知，本文所提加入局部搜索的自适应非支配排序遗传算法，即改进算法 3，其多样性均值在 ZDT1-ZDT3 上均为最优，并高于表 5-3 中所给出的改进算法 1，由此可见本文所提改进算法通过提高算法进化适应性及局部优化能力可以有效增加种群多样性。改进算法 3 多样性均值在 ZDT4、ZDT6 上低于原始算法及表 5-3 中所给出的改进算法 1，该点与其在收敛性上的表现所一致，也就是说改进算法 3 通过将适应性策略和加入局部搜索的改进 NSGA-II 相结合，放大了两者优点的同时也放大了其缺点，即对于完全欺骗性问题由于其收敛性的降低导致其多样性指标也得到了降低，对多样性敏感的问题由于其收敛速度的急剧增加导致种群多样性的迅速下降，使其进化能力不足，从而导

致其多样性优化效果劣化。

表 5- 多样性准则 Δ

函数名称	算法名称	均值	方差
ZDT1	NSGA-II	0.008439	0.001730
	改进算法 3	0.000816	0.000282
ZDT2	NSGA-II	0.006953	0.002496
	改进算法 3	0.000860	0.000333
ZDT3	NSGA-II	0.020961	0.031298
	改进算法 3	0.001526	0.000471
ZDT4	NSGA-II	0.014611	0.005809
	改进算法 3	0.327318	0.148027
ZDT6	NSGA-II	0.132053	0.062484
	改进算法 3	0.276983	0.125747

5.4 本章小结

本章通过使用仿真测试函数对本文所提改进进行了验证，即首先对本文所提基于非支配排序分层的适应性策略进行测试，由实验结果可知本文所提改进可以有效提升 NSGA-II 的优化性能；然后，对本文所提加入局部搜索的非支配排序遗传算法进行性能验证，由实验结果表明本文所提改进能有效提高原算法局部优化能力，从而提高其全局优化性能；最后，对融合本文所提基于非支配排序分层的适应性策略和加入局部搜索的非支配排序遗传算法的综合算法，即加入局部搜索的自适应非支配排序遗传算法进行性能验证，表明本文所提算法在全局搜索和局部搜索两方面提高了原算法求解性能，证明了本文对 NSGA-II 所做优化研究工作的有效性。

第6章 总结与展望

6.1 总结

随着人类文明的进步和社会的发展,越来越多的优化问题呈现在人们眼前,如何能够高效的求解此类问题具有很重要的实际意义及价值,为了更好的解决现实生活中的多目标优化问题,本文在对经典多目标优化算法NSGA-II的研究基础上,首先提出了一种基于非支配排序分层的适应性策略和加入局部搜索的非支配排序遗传算法,然后对两者进行融合提出了一种加入局部搜索的自适应非支配排序遗传算法,最后通过实验验证了所提策略及算法的有效性。

本文主要研究内容可以总结为下几部分:

首先,本文通过对相关文献的研究及学习,充分了解了多目标优化问题的发展历史、背景及意义,深入分析多目标优化问题传统解法的不足,在此基础上对多目标优化算法中常用策略及性能评价指标进行总结,给出了基本框架,并介绍了NSGA-II的关键技术及基本流程。

其次,针对NSGA-II在运行过程中运行参数保持不变,不能根据环境的变化对运行参数进行自适应调整,从而导致求解效率较低的问题给出了一种基于非支配排序分层的自适应进化策略。所提策略根据算法的运行阶段、运行代数及当前种群非支配层个体数对不同个体赋予相应的交叉率、变异率、交叉分布指数、变异分布指数,通过提高算法的自适应性提升了算法的性能。

再次,针对NSGA-II局部搜索能力较弱,从而导致难以搜索到全局最优解集的问题,给出了一种基于模拟退火算法的NSGA-II优化改进。本文所提改进首先对传统模拟退火算法进行调整,即以支配关系作为个体优劣的评价标准,在搜索过程中只接受优于原始解的新解,并根据当前进化代所对应的退火温度对搜索范围进行动态调节;然后将调整后的模拟退火算法引入到NSGA-II中,形成综合算法,即加入局部搜索的非支配排序遗传算法,通过对每一代进化生成的父代及子代种群个体进行退火搜索,从而起到提高算法局部优化能力的作用;最后将加入局部搜索的非支配排序遗传算法与第一阶段所给出的适应性策略相结合,给出了一种加入局部搜索的自适应非支配排序遗传算法,通过提高算法进化适应性及局部优化能力以提升原算法求解全局最优解的能力,从而起到抑制早熟、加快收敛速度、增加种群多样性的作用,实现对NSGA-II的优化改进。

最后,通过模拟仿真实验对所提改进进行验证。实验分为三部分:第一部分通过实验对比表明本文所提基于非支配排序分层的适应性策略能有效提高算

法的全局搜索能力；第二部分通过实验对比表明本文所提加入局部搜索的非支配排序遗传算法能有效提高原算法的局部搜索能力；第三部分通过实验对融合本文所提基于非支配排序分层的适应性策略和加入局部搜索的非支配排序遗传算法的综合算法，即加入局部搜索的自适应非支配排序遗传算法进行性能验证，表明本文对 NSGA-II 进行的两点优化能有效结合，在收敛性及多样性两方面对算法的性能确有提升，从而证明了本文对 NSGA-II 优化研究工作的有效性。

6.2 未来工作展望

本文在多目标优化问题相关理论及传统解法研究基础上，针对 NSGA-II 现存问题提出了相应改进，有效改进了其求优性能，但是仍存在较多的不足，以下给出后续研究的方向：

（1）本文所提基于非支配排序分层的自适应进化策略虽然可以表明基于交叉算子和变异算子的自适应策略对 NSGA-II 的适用性，但是如何使交叉率、变异率、交叉分布指数、变异分布指数的变化过程更加符合 NSGA-II 运行情况仍需进一步的深入探讨，同时适应性策略对完全欺骗型问题的优化能力仍待进一步提升。

（2）本文所提加入局部搜索的非支配排序遗传算法中对于退火搜索参数设定问题没有解决，现有设定仍为依靠个人经验，如何更好地提高 NSGA-II 局部搜索性能仍具有较大提升空间，同时如何解决由局部搜索能力提升所导致的种群多样性下降问题仍需进一步的深入探讨，以使其可以更好地防止 NSGA-II 早熟收敛。

致 谢

时光飞逝，三年的研究生生活即将结束，回忆过往，老师、同学、朋友和家人给予了我巨大的理解、支持和帮助，正是有了你们对我的关心和爱护，才让我有足够的勇气克服重重困难，迈过一个又一个坎坷，顺利的完成了三年的研究生生活。在辽宁大学的这三年学习经历是我人生当中一笔无比珍贵的财富，它将使我终生受益，值得我用一生的时间来感恩。正是我身边的老师、同学、朋友和家人给予了我这无价的财富，在此，衷心地感谢一路上陪伴我和帮助过我的人们，没有你们就没有我今天的成绩。

首先我要感谢我的导师冯勇教授对我的辛勤指导，在攻读硕士研究生期间，冯老师于我亦师亦友，不仅在学术研究中给予我严谨耐心的教导，而且在生活中给予我悉心的关怀，教会我为人处世之道。正是冯老师组织的一周一次的学术汇报，提高了我的科研水平，夯实了我的理论基础，扩宽了我的学术视野，使我能顺利的完成毕业论文的撰写；正是冯老师组织的集体活动使我在枯燥的学习中体会到回家的温暖；正是冯老师对我的激励和肯定使我能一直以饱满的精神态度投入到学习和研究中。冯老师诚恳待人的作风、严谨的科研态度和无私的敬业精神深深的感染了我，冯老师是我一生学习的榜样，激励着我继续努力奋斗。

感谢徐红艳老师在过去的三年中对我的培养和关心，在学术研究上为我提

供了巨大的帮助和支持。还要感谢我的授课老师：陈廷伟、宫元九、范铁生、石祥斌等老师对我的教育和指导。感谢辅导员陈谦老师在学习和生活中对我的帮助。感谢辽宁大学信息学院所提供的宽松学习环境和高性能实验器材，正是学院所提供的实验室才使我能更专注的投入到学术研究中，也正是学院组织的一次次课外活动丰富了我的日常生活。

感谢导师小组的伙伴们、室友兄弟们和实验室的学弟学妹们，感谢你们的帮助，通过和你们的合作交流、问题探讨，让我取得学业上的进步。

感谢我的父母，你们永远是我温暖的港湾，是我一路上坚持不懈的后盾和精神支柱，感谢你们对我的理解与鼓励，你们给予的全身心付出是我终身难以回报的。

最后感谢各位答辩老师和评审专家，谢谢你们花费宝贵的时间给予我的批评和建议，谢谢你们对我的关心和指导！

郭军

二〇一七年四月

参考文献

- [1] 卢海 , 鄢烈祥 , 史彬 等 . 并行多家族遗传算法解多目标优化问题 [J]. 化工学报 , 2012, 63(12): 3985-3990.
- [2] 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 , 2010.
- [3] 崔逊学 . 多目标进化算法及其应用 [M]. 北京 : 国防工业出版社 , 2006.
- [4] T. Back, Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms[M]. Oxford: Oxford Univ. Press, 1996.
- [5] 丁永生 . 计算智能 : 理论、技术与应用 [M]. 北京 : 科学出版社 , 2004.
- [6] Fonseca C. M., Fleming P. J. An overview of evolutionary algorithms in multi-objective optimization[J]. Evolutionary Computation, 1995, 3(1): 1-16.
- [7] Rendon V., Charre U. A non-generational genetic algorithm for multi-objective optimization[A]. In Proceedings of the Seventh International Conference on Genetic Algorithms[C]. San Francisco: Morgan Kaufmann, 1997: 658-665.
- [8] Schaffer J. D. Multiple objective optimization with vector evaluated genetic algorithms[A]. Proceedings of 1st International Conference on Genetic Algorithms and Their Application[C]. Hillsdale: Lawrence Erlbaum Associates, Inc., 1985: 93-100.
- [9] Fourtman M. P. Compaction of symbolic layout using genetic algorithms[A]. Proceedings of the 1st International Conference on Genetic Algorithms[C]. Hillsdale: L. Erlbaum Associates Inc., 1985: 141-153.
- [10] Kursawe F. A Variant of Evolution Strategies for Vector Optimization[A]. In H. P. Schwefel and R. Manner (Eds.), Parallel Problem Solving from Nature-Proceedings of the first workshop PPSN[C]. Berlin: Springer, 1991: 193-197.
- [11] Andersson J. A Survey of Multi-objective Optimization in Engineering Design[R]. Technical Report No. LiTH-IKP-R-1097, Department of Mechanical Engineering, Linköping University, Sweden, 2000.
- [12] Goldberg D. E. Genetic Algorithms for Search, Optimization and Machine Learning[M]. Boston: Addison-Wesley Longman Publishing Co. Inc., 1989.
- [13] Fonscea C. M., Fleming P. J. Genetic Algorithm for multi-objective optimization: Formulation, discussion and generation[A]. In Proceedings of the 5th International Conference on

- Genetic Algorithms[C]. San Mateo, California: MIT Press, 1993: 416-423.
- [14]
尚荣华 , 胡朝旭 , 焦李成 等 . 多目标优化算法在多分类中的应用研究 [J]. 电子学报 , 2012, 40(11): 2264-2269.
- [15]
刘爱军 , 杨育 , 程文明 等 . 复杂制造环境下的改进非支配排序遗传算法 [J]. 计算机集成制造系统 , 2012, 18(11): 2446-2458.
- [16]
刘江 . 改进的遗传算法求解多目标优化问题 [D]. 西安 : 西安电子科技大学 , 2015.
- [17]
张丽科 . 改进蜜蜂进化型遗传算法引导的 NSGA2 两阶段优化算法及应用 [D]. 长沙 : 湖南大学 , 2016.
- [18]
宋晓鹏 , 韩印 , 姚佼 . 基于 NSGA 算法的公交车调度优化模型 [J]. 上海理工大学学报 , 2014, 36(4): 357-361.
- [19]
陶文华 , 刘洪涛 . 基于差分进化与 NSGA-II 的多目标优化算法 [J]. 计算机工程 , 2016, 42(11): 219-224.
- [20]
王聪 , 赵文玲 . 基于自适应多目标指数罚函数的 NSGA-II 算法 [J]. 山东理工大学学报 (自然科学版) , 2016, 30(3): 11-14.
- [21]
耿焕同 , 李辉健 , 赵亚光 等 . 基于自适应混合非支配个体排序策略的改进型 NSGA-II 算法 [J]. 计算机应用 , 2016, 36(5): 1319-1324.
- [22]
边毅 , 袁方 , 郭俊霞 等 . 面向 CPU+GPU 异构计算的多目标测试用例优先排序 [J]. 软件学报 , 2016, 27(4): 943-953.
- [23]
陈志旺 , 陈林 , 白铎 等 . 求解约束多目标区间优化的交互多属性决策 NSGA-II 算法 [J]. 控制与决策 , 2015, 30(5): 865-870.
- [24]
谢承旺 , 李凯 , 廖国勇 . 一种带差分局部搜索的改进型 NSGA2 算法 [J]. 计算机科学 , 2013, 40(10): 235-238.
- [25]
王超学 , 田利波 . 一种改进的多目标合作型协同进化遗传算法 [J]. 计算机工程与应用 , 2016, 52(2): 18-23.
- [26]
熊盛武 , 李锋 . 并行 Pareto 多目标演化算法 [J]. 武汉理工大学学报 (理学版) ,

2003, 49(3): 318-322.

[27]

郑强 . 带精英策略的非支配排序遗传算法的研究与应用 [D]. 杭州 : 浙江大学 , 2006.

[28]

李莉 . 基于遗传算法的多目标寻优策略的应用 [D]. 无锡 : 江南大学 , 2008.

[29]

王鲁 . 基于遗传算法的多目标优化算法研究 [D]. 武汉 : 武汉理工大学 , 2006.

[30]

李中林 . 求解多目标优化问题的混合遗传算法的研究与应用 [D]. 沈阳 : 东北大学 , 2012.

[31]

刘旭红 , 刘玉树 , 张国英 等 . 多目标优化算法 NSGA-II 的改进 [J]. 计算机工程与应用 , 2005, 41(15): 73-75.

[32]

程刚 . 基于非支配个体自适应划分策略的进化多目标优化及应用 [D]. 西安 : 西安电子科技大学 , 2010.

[33]

赵森 . 基于精英级选择与扩展策略的多目标智能算法研究 [D]. 广州 : 华南理工大学 , 2013.

[34]

Cohon, J. L. Multi-objective programming and planning[M]. New York: Academic Press, 1978.

[35]

Jeffrey A. J., Deepak G., Mahmut A. G., etc. Supply chain multi-objective simulation optimization[C]. Proceedings of the 2002 Winter Simulation Conference, San Diego, California: Association for Computing Machinery, 2002: 1306-1314.

[36]

宣家骥 . 多目标决策 [M]. 湖南 : 湖南科学技术出版社 , 1989.

[37]

Deb K., Pratab A., Agrawal S., etc. A fast and elitist multi-objective genetic algorithm: NSGA-II[J]. IEEE transactions on Evolutionary Computation, 2002, 6(2): 182-197.

[38]

Goldberg D. E., Richardson J. Genetic Algorithms with Sharing for Multimodal Function Optimization[A]. Genetic Algorithms and Their Applications: Proceedings of the Second International Conference on Genetic Algorithms[C]. Cambridge: Massachusetts Institute of Technology, 1987: 41-49.

[39]

Holland J. H. Adaptation in natural and artificial systems[M]. Ann Arbor: The University of Michigan Press, MI, 1975.

- [40]
Fogel L. J., Owens A. J., Walsh M. J. Artificial intelligence through simulated evolution[M]. New York: Wiley, 1966.
- [41]
Zitzler E., Deb K., Thiele L. Comparison of multi-objective evolutionary algorithms: Empirical results[J]. Evolutionary Computation, 2000, 8(2): 173-195.
- [42]
Srinivas N., Deb K. Multi-objective optimization using non-dominated sorting in genetic algorithms[J]. Evolutionary Computation, 1994, 2(4): 221-248.
- [43]
Holland J. H. Outline for a logical theory of adaptive systems[J]. Journal of the Association for Computation Machinery, 1962, 9(3):297-314.
- [44]
李敏强 , 窦纪淞 , 林丹 等 . 遗传算法的基本理论与应用 [M]. 北京 : 科学出版社 , 2002.
- [45]
黄友锐 . 智能优化算法及其应用 [M]. 北京 : 国防工业出版社 , 2008.
- [46]
Srinivas M., Patnaik L. M. Adaptive Probabilities of Crossover and Mutation in Genetic Algorithm[J]. IEEE Trans. on SMC, 1994, 24(4): 656 – 667.
- [47]
任子武 , 伞冶 . 自适应遗传算法的改进及在系统辨识中应用研究 [J]. 系统仿真学报 , 2006, 18(1): 41-66.
- [48]
张文修 . 遗传算法的数学基础 [M]. 西安 : 西安交通大学出版社 . 2003.
- [49]
Whitley L. D., Vose M. D. Foundations of genetic algorithms 3[M]. San Mateo, CA: Morgan Kaufmann, 1995.
- [50]
Davis L. Handbook of genetic algorithms[M]. New York: Van Nostrand Reinhold Press, 1991.
- [51]
Davis L., Belew R. K., Booker L. B. etc. Bit-climbing, representational bias, and test suite design. In Proceedings of Fourth International Conference on Genetic Algorithms(ICGA 4) [C]. San Mateo, CA: Morgan Kaufmann Publishers, 1991: 18-23.
- [52]
Michalewicz Z., Fogel D. B. 如何求解问题—现代启发式方法 [M]. 曹宏庆 , 译 , 北京 : 中国水利水电出版社 , 2003.
- [53]

- 罗彪 , 郑金华 , 杨平 . 基于定向爬山的遗传算法 [J]. 计算机工程与应用 , 2008, 44(6): 92-95.
- [54]
Metropolis N., Rosenbluth A., Rosenbluth M. Equation of state calculations by fast computing machines[J]. Journal of Chemical Physics, 1953, 56(21): 1087-1092 .
- [55]
刘怀亮 , 刘淼 . 一种混合遗传模拟退火算法及其应用 [J]. 广州大学学报 , 2005, 4(2), 141-144.
- [56]
Dowsland K. A., Thompson J. M. Simulated annealing[J]. Handbook of Natural Computing, 2012, 43(1) : 1623-1655.
- [57]
博文渊 , 凌朝东 . 布朗运动模拟退火算法 [J]. 计算机学报 , 2014, 37(6): 1031-1037.
- [58]
Vose, M. D., Liepins G. E. Schema Disruption[C]. In Proceedings of the Fourth International Conference on Genetic Algorithms(ICGA 4), San Francisco: Morgan Kaufmann Publishers, Inc., 1991: 237-242.
- [59]
Wilson S. W. GA-easy does not imply steepest-ascent optimisable[C]. In Proceedings of the Fourth International Conference on Genetic Algorithms(ICGA 4), San Francisco: Morgan Kaufmann Publishers, Inc., 1991: 85-89.
- [60]
Goldberg D. E. Genetic Algorithm and Walsh Functions, Part I & II[J]. Complex Systems, 1989, 3(3): 129-171.
- [61]
Deb K., Goldberg D. E. Sufficient conditions for deceptive and easy binary functions[J]. Annals of Mathematics and Artificial Intelligence, 1994, 10(4): 385-408.
- [62]
Mark L. Learning Python 3rd Edition[M]. USA: O'Reilly Media Inc, 2009.
- [63]
Carlos C. C., Gary B. L., David A. V. V. Evolutionary Algorithms for Solving Multi-objective Problems 2nd Edition[M], New York: Springer, 2007.

攻读学位期间发表的学术论文以及参加科研情况

发表的论文

- 1、冯勇，郭军，徐红艳，付潇莹．一种基于解空间分割的并行遗传算法[J]. 计算机数字与工程，2017， 45(2)： 226-229.