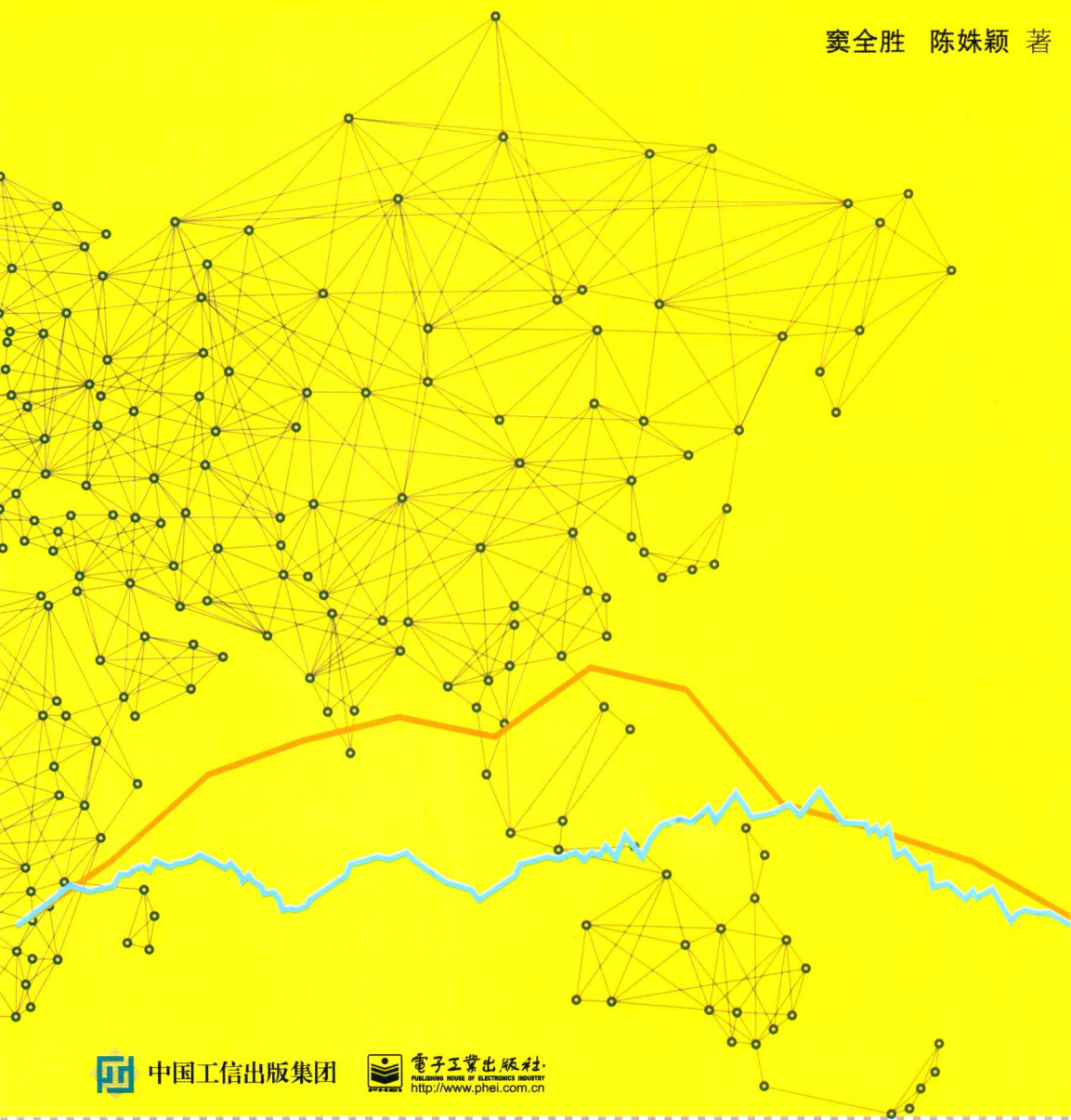


演化计算

方法及应用

窦全胜 陈姝颖 著



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

演化计算方法及应用

窦全胜 陈姝颖 著



电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书全面概括了用演化方法求解优化问题的一些新方法,重点介绍了进化规划、粒子群优化、微分演化、文化算法和蚁群算法,并阐述了几种新的改进算法,例如,群体启发进化规划方法、模拟退火粒子群优化算法及有分工策略的粒子群优化等,同时就所涉及的算法进行了系统的实验和比较,讨论了不同算法对不同环境的适应能力。

本书可作为从事群体智能、演化计算等领域的研究人员的参考书,对于解决优化问题有一定的参考和应用价值。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

演化计算方法及应用 / 窦全胜, 陈姝颖著. —北京: 电子工业出版社, 2016.1
ISBN 978-7-121-26482-5

I. ①演… II. ①窦… ②陈… III. ①算法设计—研究 IV. ①TP301.6

中国版本图书馆 CIP 数据核字 (2015) 第 143937 号

责任编辑: 李 敏

特约编辑: 刘广钦

印 刷: 三河市华成印务有限公司

装 订: 三河市华成印务有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编: 100036

开 本: 720×1000 1/16 印张: 11.25 字数: 178 千字

版 次: 2016 年 1 月第 1 版

印 次: 2016 年 1 月第 1 次印刷

定 价: 38.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

前言

最优化问题是一门既古老又年轻的学科，特别是随着科学技术的发展，许多更加复杂的大规模优化问题不断涌现出来。本书详细阐述了基于演化计算方法的最优化问题求解，主要包括：

- ① 基于遗传算法的优化问题求解；
- ② 进化规划和群体启发进化规划；
- ③ 粒子群优化算法和算法改进；
- ④ 微分演化方法；
- ⑤ 基于文化算法的约束优化问题求解；
- ⑥ 蚁群算法；
- ⑦ 演化算法的应用示例。

本书共 8 章：第 1 章绪论，介绍本书的研究背景，对求解优化问题的数学方法和演化计算方法进行了简要的概括和比较；第 2 章对基于遗传算法的最优化问题求解过程进行了阐述，对以往的一些处理技巧进行了论述；第 3 章对标准进化规划进行了分析，阐述了群体启发进化规划方法，并把它应用于求解高维优化问题；第 4 章介绍了 PSO 算法，并进一步阐述了关于 PSO 方法的两种改进方法；第 5 章介绍了微分演化方法的执行过程；第 6 章概括了用演化计算方法求解约束优化问题的几个策略，着重介绍了几种不可行个体的处理方式，讨论了基于约束与群体的分离策略的文化算法；第 7 章对蚁群算法和蚁群聚类进行介绍；第 8 章列举了几个演化计算方法在实际问题中的应用示例。

本书的工作得到了国家自然科学基金（61272244、60970088、61175053、61173173）等项目和山东省高校智能信息处理重点实验室（山东工商学院）的资助，在此表示感谢。

此外，还要特别感谢吉林大学的周春光教授，山东工商学院的范辉教授、原达教授、谢青松教授、刘培强教授，感谢他们为本书写作提供的帮助与支持。由于水平有限，书中难免有疏漏和不足之处，敬请读者指正。作者的联系方式为：qsdou@yeah.net。

作者

2015 年 2 月

目录

第 1 章 绪论	1
1.1 最优化问题	2
1.2 求解优化问题的数学方法	4
1.3 求解优化问题的演化计算方法	5
第 2 章 遗传算法	9
2.1 标准遗传算法	10
2.2 编码	12
2.2.1 二进制编码	12
2.2.2 值编码 (Value Encoding)	12
2.2.3 互换编码 (Permutation Encoding)	13
2.3 遗传算子	14
2.3.1 交叉	14
2.3.2 变异	15
2.3.3 选择	17
2.4 参数控制	18
2.5 模式定理和隐并行性定理	19
2.6 收缩映射原理	21
2.7 小结	24

第 3 章 进化规划 25

3.1 标准进化规划方法 26

3.2 进化策略 28

3.3 概率分析 29

3.4 群体启发进化规划 33

3.4.1 群体启发进化规划算法 33

3.4.2 PHEP 算法验证 35

3.5 用群体启发进化规划求解高维优化问题 40

3.5.1 高维优化 40

3.5.2 实验结果 41

3.6 小结 44

第 4 章 粒子群优化 45

4.1 标准粒子群优化方法 47

4.2 二进制粒子群优化算法 49

4.3 参数设置 56

4.4 粒子轨迹的确定性分析 59

4.5 粒子的分布特征 62

4.6 粒子的聚度 63

4.7 模拟退火粒子群优化方法 66

4.7.1 模拟退火 67

4.7.2 模拟退火粒子群优化 68

4.8 有分工策略的粒子群优化方法 70

4.9 算法测试 73

4.10 动态优化 76

4.10.1 线性模型 76

4.10.2 环形模型 77

4.10.3 随机模型 77

4.10.4 动态优化仿真 78

4.11 小结 83

第 5 章 微分演化.....	85
5.1 微分演化方法描述	86
5.2 DE 参数的设置	89
5.3 算法仿真	90
5.3.1 低维条件下的仿真结果	90
5.3.2 高维条件下的仿真结果	91
5.4 微分演化粒子群优化.....	92
5.5 用 DE 确定 PSO 的最佳参数.....	95
5.6 小结	98
第 6 章 文化算法.....	99
6.1 约束的处理.....	101
6.1.1 可行解和不可行解	101
6.1.2 可行个体评价函数 $eval_f$ 的设计	102
6.1.3 不可行个体的处理	103
6.2 文化算法简介	108
6.2.1 文化算法框架	108
6.2.2 信仰空间的约束表达和信仰空间的更新	109
6.2.3 群体空间的演化.....	113
6.3 算法测试	113
6.4 小结	114
第 7 章 蚁群优化.....	116
7.1 蚁群优化算法	117
7.2 蚁群聚类	120
7.3 小结	123

第 8 章 应用举例 125

8.1 属性约简 126

8.1.1 信息系统与属性约简 126

8.1.2 常用的属性约简方法 126

8.1.3 基于遗传算法的属性约简 129

8.2 电力负荷关联规则提取 132

8.2.1 问题概述 132

8.2.2 关联规则 133

8.2.3 频繁项集挖掘 136

8.2.4 基于 DPSO 方法负荷规则萃取 138

8.3 神经网络训练 142

8.3.1 神经元模型 143

8.3.2 神经网络 144

8.3.3 神经网络的学习 145

8.3.4 前向神经网络 146

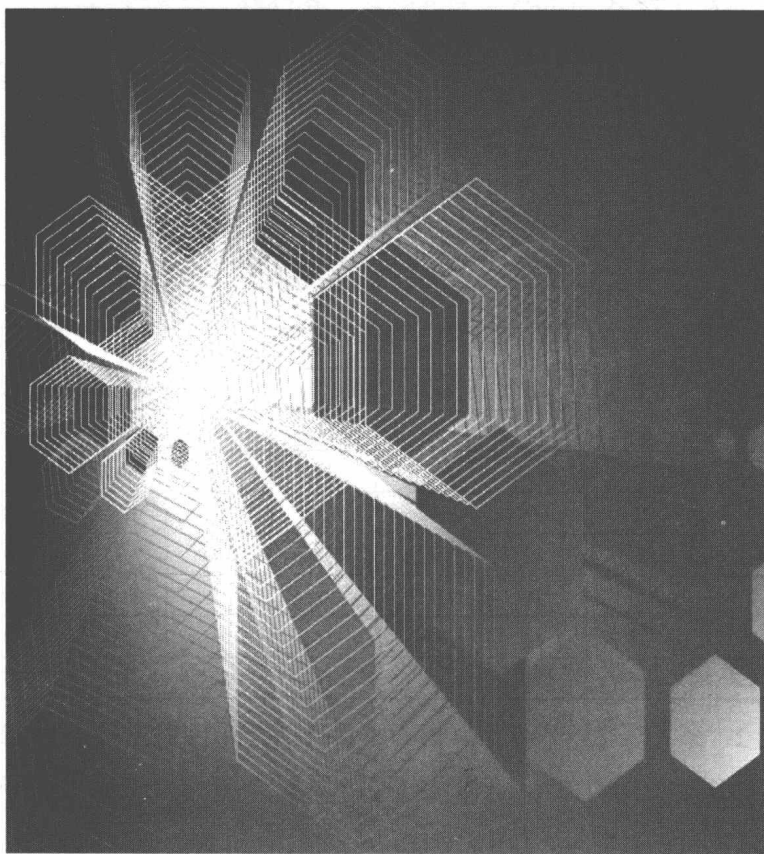
8.4 小结 149

附录 A 无约束优化问题 150

附录 B 约束优化问题 157

参考文献 162

第1章 绪论



1.1 最优化问题

最优化是一个重要的数学分支，也是一门应用相当广泛的学科。最优化的目的是对于给出的实际问题，从众多方案中选择出最优方案。例如，工程设计中怎样选择设计参数，使得设计方案既满足设计要求又能降低成本；资源分配中，怎样分配有限的资源，使得分配方案既能满足各方面的基本要求，又能获得好的经济效益。在人类活动的各个领域，诸如此类，不胜枚举。

应用案例 1：某城市有大型超市 8 个，在“农超对接”对接模式下，8 个超市的某些品种蔬菜全部由该市县级农产品物流中心进行统一配送。其中，县级农产品物流中心位置为 O，8 个超市间距离不同，如表 1-1 所示。配送车辆从 O 出发，选择一个最短的行程对 8 个超市进行农产品配送，最后回到物流中心 O，问如何选择配送路线，使其配送成本最低？

表 1-1 某市 8 个超市间距离 单位 (km)

	O	A	B	C	D	E	F	G	H
O	0	25	16	28	19	10	24	20	22
A	25	0	15	9	7	6	23	12	15
B	16	15	0	9.7	4	15	10	13	5.7
C	28	9	9.7	0	12	5.7	6.2	9	12
D	19	7	4	12	0	18	6	16	10
E	10	6	15	5.7	18	0	12	9	15
F	24	23	10	6.2	6	12	0	21	17
G	20	12	13	9	16	9	21	0	12
H	22	15	5.7	12	10	15	17	12	0

应用案例 2：某工厂生产甲、乙、丙、丁共 4 种产品，需要用到 A、B、C 共 3 种原料，每种产品需要使用的各种原料的数量及其可能获得的利润如表 1-2 所示。另外，A、B 两种原料供应量有限，单位生产周期内只能提供一定的数量，而 C 种原料一经开包使用就必须用足一定量后才可停止使用，且不能单独使用。现有关数据均见表 1-2，问应如何安排生产，才能使该厂所获利润达到最大值？

表 1-2 加工产品所需原料及可能获得的利润

原料	加工每件产品所需原料				单位周期内原料的供应量或 必须使用量
	甲	乙	丙	丁	
A	1.0	1.2	1.4	1.5	≤2100
B	0.5	0.6	0.6	0.8	≤1000
C	0.7	0.7	0.8	0.8	≤1300
每件利润	12	15	8	10	

类似这样的优化问题在生产生活中大量存在，其研究历史可以追溯到十分古老的极值问题。早在 17 世纪，英国伟大科学家牛顿发明微积分的时代，已经提出极值问题，后来又出现 Lagrangian 乘数法。1847 年，法国数学家 Cauchy 研究了函数沿什么方向下降最快的问题，提出了最速下降法。1939 年，前苏联数学家 Л.В.Канторович 提出了解决下料问题和运输问题这两种线性规划问题的求解方法。人们关于最优化问题的研究工作，随着历史的发展不断深入。但是，任何学科的进步，都受到历史条件的限制，直至 20 世纪 30 年代，最优化这个古老的课题还未形成独立的学科。在最近的二三十年中，伴随着计算机技术的高速发展和优化计算方法的进步，各种优化问题的理论研究发展迅速，新方法不断出现，实际应用日益广泛。在计算机的推动下，一些超大规模的优化问题得以实现，最优化理论与方法在经济规划、工程设计、生产管理、交通运输等方面得到了广泛应用，成为一门十分活跃的学科。

最优化问题的一般形式为^[1]

$$\begin{aligned} &\text{Min } f(x) \\ &\text{s.t. } x \in X \end{aligned}$$

(1-1)

其中， $x \in R^n$ 是决策变量， $f(x)$ 为目标函数， $X \subset R^n$ 为约束集或可行域。特别地，如果约束集 $X = R^n$ ，则最优化问题 (1-1) 称为无约束最优化问题。在本书中，若 X 为 R^n 中的超立方体，也近似地把问题 (1-1) 看成无约束最优化问题。

优化问题通常带有一些约束条件，称为约束最优化问题，表述为

$$\begin{aligned} &\text{Min } f(x) \\ &\text{s.t. } c_i(x) = 0, i \in E \\ &\quad c_j(x) \geq 0, j \in I \end{aligned}$$

(1-2)

这里, E 和 I 分别是等式约束指标集和不等式约束指标集, $c_i(x)$ 是约束函数。当目标函数和约束函数均为线性函数时, 问题称为线性规划; 当目标函数和约束函数中至少有一个是变量 x 的非线性函数时, 问题称为非线性规划。此外, 根据决策变量、目标函数和要求的不同, 最优化还可以分为整数规划、动态规划、网络规划、非光滑规划、随机规划、几何规划、多目标规划等若干分支。

1.2 求解优化问题的数学方法

求解最优化问题通常采用迭代方法, 其基本思想如下: 给定一个初值点 $x_0 \in R^n$, 按照某一迭代规则产生一个序列 $\{x_k\}$, 使得当 $\{x_k\}$ 为有穷点列时, 其最后一个点是最优化问题的最优解。当 $\{x_k\}$ 为无穷列时, 它有极限点, 且其极限点是最优化问题的最优解。一个好的算法应该具备如下特征: 迭代点能够稳定地接近局部极小点 x^* 的邻域, 然后迅速收敛于 x^* 。迭代方法的基本结构如下^[1]。

步骤 1 给定初始点 x_0 , 确定搜索方向 d_k 。

步骤 2 确定步长因子 α_k 。

步骤 3 令 $x_{k+1} = x_k + \alpha_k d_k$ 。 (1-3)

步骤 4 若 x_{k+1} 满足某种终止条件, 则停止迭代, 得到近似最优解。否则, 重复上述步骤。

从式 (1-3) 中可以看出, 迭代算法的核心如下: 构造适应各种问题的步长因子 α_k 和搜索方向 d_k , 不同的 α_k 和 d_k 的构造方式, 形成了不同的优化算法。

最速下降法是求解无约束优化问题最经典的方法, 又称梯度下降法, 该方法以函数在 x_k 处的负梯度方向作为算法的下降方向, 最速下降法在最优化中具有重要的理论意义。但是, 最速下降方向仅反映了被优化函数的局部性质, 对于许多问题, 最速下降法并非“最速下降”, 而是下降非常缓慢; 利用函数的二阶导数信息, 即在椭球范数 $\|\cdot\|_2$ 下, 取负梯度方向, 最速下降法就变成了所谓的牛顿法, 牛顿法利用了函数二阶导数信息, 为克服牛顿法中 Hesse 矩阵未必正定的难题, Goldfeld 等人提出了修正牛顿法。类似的

方法还有有限差分牛顿法、负曲率方向法、不精确牛顿法等。牛顿法的基本思想是在迭代点 x_k 附近用二次函数逼近 $f(x)$ ，但这种方法只能保证算法的局部收敛性，信赖域方法是一种能够保证算法总体收敛的方法，并且能够解决 Hesse 矩阵不正定和 x_k 为鞍点等困难，这类方法包括 Levenberg-Marquardt 方法、双折线步法（The Double Dogleg Step Method）等。另一种介于最速下降法和牛顿法之间的方法是共轭方向法，这种方法只需要一阶导数信息，既克服了最速下降法收敛慢的缺点，又避免了存储和计算牛顿法所需的二阶导数信息。共轭方向法是从研究二次函数的极小化问题中产生的，但是可以推广到处理非二次函数极小化问题，典型的共轭方向法包括共轭梯度法和拟牛顿法等。

求解约束优化问题的典型方法有罚函数法、可行方向法、逐步二次规划法等。罚函数法的主要思想是：构造具有“惩罚”性质的函数，对不可行点进行惩罚，借助罚函数把约束问题转化成无约束问题。有许多罚函数被广泛使用，如内点罚函数、乘子罚函数、光滑精确罚函数和非光滑精确罚函数等。典型的可行方向法包括可行点法、广义消去法、广义概约梯度法、投影梯度法等。逐步二次规划法包括 Lagrange-Newton 法、Wilson-Han-Powell 法等。

以上方法的详细描述可以在文献[1]~[3]中找到，用数学方法求解优化问题的历史相对悠久，当前仍然在不断的发展过程中，这些传统方法大多是针对某些特定问题，并且对搜索空间要求相对严格，有些方法还要使用被优化函数的各阶导数信息。但是，随着科学和技术的发展，优化问题也变得异常复杂，有的问题甚至无法用函数关系来表达，对于这类问题，采用上述方法，不可能得到圆满的结果。因此，需要进一步研究和探索新的优化思想和优化方法。

1.3 求解优化问题的演化计算方法

传统演化计算（Evolutionary Computation）是模拟“物竞天择”与“自然遗传”的生物进化过程所产生的随机化计算模型^[4]。它的起源可以追溯到 20 世纪 50 年代，其中有影响的工作是 Bremermann^[5]、Friedberg^{[6][7]}、Box^[8]等。但是，几乎在以后的 30 年时

间里,这一领域的工作对于广大科学工作者来说还是相当陌生,造成这种状况的主要原因是:当时缺少强大的计算机硬件平台和早期演化计算本身的方法缺陷。一直到 20 世纪 70 年代, Holland^[9]、Rechenberg^[10]、Schwefel^[11]、Fogel^[12]等人的奠基性工作才慢慢地改变了这种状况。特别是近十年,以演化计算为主题的学术活动逐年增加,如今面向应用的演化计算研究几乎渗透到各行各业^{[13]~[15]}。演化计算技术在各方面得以广泛应用的原因如下:首先是得益于计算机性能的极大提高;其次演化计算具有自组织性、自学习性、自优化等智能特征,同时又具有内在并行性、原理的简单性、优良的全局性、应用的广泛性等特点。

传统演化计算是由 3 个相互联系、但事实上又彼此独立发展起来的 3 个分支组成,它们是遗传算法 (Genetic Algorithms, GAs)、进化规划 (Evolutionary Programming, EP)、进化策略 (Evolutionary Strategies, ES)。遗传算法最初是由 Holland^{[9][16][17]}等作为研究自适应过程的一般模型提出的,后来经 De Jong^{[18]~[21]}、Goldberg^{[22]~[26]}等人的进一步扩展和完善,现在主要应用于优化领域。进化规划是由 Fogel L.J.^{[12][27]}作为产生人工智能的一种尝试而提出的, Burgin^{[28][29]}、Atmar^[30]等人在这一领域作了深入的工作,其方法是演化一个有限状态机 (Finite State Machines, FSM),使之具有最佳的预测能力。后来, Fogel D.B.^{[31]~[33]}等借助于进化策略方法对进化规划进行了发展,并应用于数值优化等问题中。Rechenberg^{[34][35]}和 Schwefel^{[36][37]}为求解多参数优化问题提出了进化策略 (Evolutionary Strategies, ES)。后来,其他学者在这个领域继续研究,从原始的 $(1+1)$ -ES 进化策略,发展到 $(\mu+1)$ -ES 进化策略,进而发展到 $(u+\lambda)$ -ES 进化策略。从 20 世纪 80 年代开始,由于计算机硬件性能的提高,使得演化计算可应用于求解高度复杂的现实优化问题。

传统的演化计算包括遗传算法、进化策略和进化规划。20 世纪 90 年代以来,遗传程序设计 (Genetic Programming, GP) 作为遗传算法的一个分支独立发展起来,并在自动程序设计方面显示了初步潜力,因此,有些学者把演化计算扩展为包括遗传程序设计在内的 4 个分支。一些研究^[38]把演化计算进一步扩充为“仿生”类演化计算和“拟物”类演化计算,其中“仿生”类演化计算包括上述 4 个分支,“拟物”类演化计算包括模

拟物理系统演化和社会系统演化等所有算法模型，如 Boltzmann 演化策略、文化算法（Culture Algorithms, CAs）、粒子群优化等（Particle Swarm Optimization, PSO）等。有学者将演化计算称为进化计算，本书用演化计算来概括所有用计算机系统模拟大自然演化过程来求解现实问题的一切算法，既包括所有“仿生”类方法，又包括“拟物”类算法。

随着各个学科研究的不断深入，出现了越来越多的难以用传统方法解决的实际问题，人们开始尝试使用演化计算方法。与传统数学方法相比，演化计算方法有如下特点。

（1）演化计算的处理对象可以是参数本身，也可以是经过某种映射形成的特定编码，编码形式可以是矩阵、树、图、集合、串、序列、链和表等。这个特点使演化计算有广泛的应用领域。

（2）演化计算采用群体搜索策略，而传统方法多采用单点搜索策略，这种特点使演化计算具有极好的全局性，减少陷入局部最优的风险；同时，也使演化计算本身易于大规模并行实现，可充分发挥高性能计算机系统的作用。

（3）演化计算基本不依赖搜索空间的知识及其他辅助信息，它采用适应度函数来评价个体，并在此基础上驱动演化过程，而对适应度函数本身无特别严格要求。传统方法则要求函数有诸如连续、可微或空间凸性等条件。这使演化计算有更广泛的应用范围；演化计算用概率的变迁规则来控制搜索的方向，从表面上看好像是在盲目搜索，实际上它遵守某种随机概率，在概率意义上朝最优解方向靠近，因此，它不同于通常采用确定性规则的传统方法，需要构造合适的下降方向。

与求解优化问题的数学方法相比，演化计算方法的优点如表 1-3 所示。通过表 1-3 的比较可以看出，在解决某些复杂优化问题上，演化计算方法有着得天独厚的优势，本书将对演化计算中的几个典型算法进行阐述、比较和分析。

表 1-3 用数学方法和演化方法求解优化问题比较

方 法	数学方法	演化计算方法
应用范围	往往是面向问题的	独立于问题
对初值的依赖性	对初值有很强的依赖性，即初值敏感性	基于群体操作，基本不依赖初值的选取
对搜索空间和优化函数的要求	一般要求优化函数连续、各阶导数等信息，对搜索空间也有严格要求	不用搜索空间等辅助信息
计算复杂性	计算复杂	计算简单