

应用数学译丛

Network Models  
and Multiobjective Genetic Algorithm

# 网络模型与多目标遗传算法

[日] 玄光男 林 林 著  
梁承姬 于歆杰 译

清华大学出版社

应用数学译丛

Network Models  
and Multiobjective Genetic Algorithm

# 网络模型与多目标遗传算法



林林 著  
于歆杰 译

清华大学出版社  
北京

## 内 容 简 介

本书首先围绕物流配送计划问题、网络的开放式最短路径优先问题、多阶段供应链管理的网络问题以及双目标网络问题中的网络系统的最小费用最大流量问题这几个可用网络模型一般化的 NP-hard 组合优化问题,介绍如何设计不同的染色体来采用遗传算法解决网络设计问题;然后,在数值实验中通过求解实际问题详细地介绍了遗传算法的使用方法;最后,介绍怎样有效地运用遗传算法求解从基本的网络模型,到通信网络、逻辑系统、先进的生产计划等不同的多目标网络模型。

本书通过使用具体数值实例进行浅显易懂的讲解,而没有涉及难懂的理论讲解,大学低年级学生凭借其现有的数学基础知识就可以完全理解书中介绍的网络数学模型和遗传算法的解法。书中丰富的数值实例能够加深读者对算法的理解,为学习带来便利。

玄光男,林 林

ネットワークモデルと多目的 GA

ISBN: 978-4-320-12206-2

共立出版株式会社 © 2008

本书中文简体字翻译版由共立出版株式会社授权清华大学出版社在中华人民共和国境内(不包括中国香港、澳门特别行政区和中国台湾地区)独家出版发行。未经出版者预先书面许可,不得以任何方式复制或抄袭本书的任何部分。

北京市版权局著作权合同登记号 图字: 01-2010-0282

版权所有,侵权必究。侵权举报电话: 010-62782989 13701121933

### 图书在版编目(CIP)数据

网络模型与多目标遗传算法/(日)玄光男,(日)林林著;梁承姬,于歆杰译.—北京:清华大学出版社,2017

(应用数学译丛)

ISBN 978-7-302-45614-8

I. ①网… II. ①玄… ②林… ③梁… ④于… III. ①计算机网络—网络模型 ②计算机—算法理论 IV. ①TP393.021 ②TP301.6

中国版本图书馆 CIP 数据核字(2016)第 283941 号

责任编辑:刘 颖

封面设计:常雪影

责任校对:王淑云

责任印制:何 芊

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, [c-service@tup.tsinghua.edu.cn](mailto:c-service@tup.tsinghua.edu.cn)

质量反馈:010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

印 刷 者:三河市君旺印务有限公司

装 订 者:三河市新茂装订有限公司

经 销:全国新华书店

开 本:170mm×230mm 印 张:16

字 数:296 千字

版 次:2017 年 3 月第 1 版

印 次:2017 年 3 月第 1 次印刷

印 数:1~2000

定 价:49.00 元

产品编号:030910-01

# 前言

从因特网时代的信息网络系统,到基于 GPS 进行车辆导航的道路信息系统,以及软件开发的项目进度管理系统,均建立在网络模型的基础之上。目前,网络建模已经被灵活地运用到计算机科学、自然科学、运筹学、金融学、工学等诸多领域。网络建模通过点、弧(连接)以及流量来处理网络问题并搜索到最佳的解决方案。

近年来,由于信息通信技术的快速发展,网络技术的飞速进步和普及,以及产业经济全球化,不仅仅是信息通信业,制造业以及物流业也发生着巨大的变革。优化问题的求解过程,如应用大规模网络系统的最优化通信路径,及网络的开放式最短路径优先(Open Shortest Path First, OSPF)问题,以附加快速信息交互能力的企业资源软件包(Enterprise Resource Package, ERP)为基础的生产信息系统的生产物流调度问题,伴随网络环境下物流系统中顾客和供应商的全球化问题的多阶段供应链管理(Supply Chain Management, SCM)网络问题等,因其结构复杂、多伴有很多制约条件,且常为多目标优化问题,被我们定义为 NP-hard 组合优化问题。特别是针对各企业生产物流过程,要求迅速灵活运用准确的信息并给出合理决策,具体指从接受订单到企划,再到生产过程以及密切相关的适时配送计划,即根据供应链管理系统寻求到全局最优化的解。

一般地,大规模组合优化问题用旧有方法求解时存在解决不了的问题,所以在启发式算法里最被广泛灵活应用的遗传算法(Genetic Algorithm, GA)受到了关注。遗传算法是进化计算的一种,在业界作为实用技术之一被广泛地使用。例如,在 SAP、i2、IBM 等世界各地的企业资源软件包中,均标准化地配备了基于遗传算法的最优化工具。近年来,遗传算法被最广泛地应用于求解难以用数学模型定义的问题或者结构复杂的最优化问题等。并且从 SCI 级别的国际刊

物中基于遗传算法的研究论文数量之多可以看出很多学者也对遗传算法的能力表示肯定。

为了灵活运用进化计算之一的遗传算法,本书主要围绕物流配送计划问题、网络的最短路径优先问题、多阶段供应链管理网络问题,以及双目标网络问题中的网络系统的最小费用最大流量问题这几个可用网络模型一般化的 NP-hard 组合优化问题,介绍如何设计不同的染色体来采用遗传算法解决网络设计问题,此外,在数值实验中通过求解实际问题详细地介绍了遗传算法的使用方法。进一步地,怎样有效地运用遗传算法求解从基本的网络模型,到通信网络、逻辑系统、先进的生产计划(Advanced Planning and Scheduling, APS)等不同的多目标网络模型,将在后面的 5 章进行说明。

在第 1 章遗传算法中,对背景和作为基础的染色体的编码、评价函数、遗传操作等进行了说明,通过组合优化问题中的典型模型——配词问题和背包问题来解释应用基础遗传算法的计算过程,并介绍了模糊逻辑和遗传算法组合的混合型遗传算法。第 2 章网络模型基础中,介绍了作为网络模型中最基本的最短路径模型、最大流量模型、最小费用流模型和最小生成树模型。第 3 章物流网络模型中介绍了物流模型、两阶段物流模型、车辆配送模型和工厂—配送中心物流模型。第 4 章多目标遗传算法在简要地说明了多目标最优化模型之后,对多目标遗传算法概要、多目标遗传算法过程、Pareto 最优解的评价,以及多目标遗传算法的数值计算实例进行了介绍。在第 5 章多目标网络模型中,介绍了作为该领域中最新的应用研究用例的最小费用最大流量网络、多目标供应链网络,生产物流系统的网络以及通信系统可靠性网络。

本书充分考虑到实用性,摒弃工具书中难懂的理论讲解,通过使用具体数值实例进行浅显易懂的讲解,保证专科学校学生或者大学低年级学生凭借现有的数学基础知识也可以完全理解书中介绍的网络数学模型和遗传算法的解法。书中丰富的数值实例能够加深读者对算法的理解,为学习带来便利。

本书从 1995 年策划开始,已经受到了很多国内外人士的指导和建议。特别是早稻田大学大学院冈本东博士(岩手县立大学)、棕田实博士(日本工业大学)、访问学者 Fulya Altiparmak (Gazi University),以及软计算研究室的各位博士,特别要感谢刚田几太郎氏、安高真一郎氏,也非常感谢共立出版社(株)的小山透氏、松永智仁氏、国井和郎氏在出版方面给予的帮助。

2008 年 2 月

玄光男 林 林

# 目录

|                      |    |
|----------------------|----|
| 第 1 章 遗传算法           | 1  |
| 1.1 遗传算法基础           | 1  |
| 1.1.1 遗传算法概述         | 1  |
| 1.1.2 编码             | 4  |
| 1.1.3 适值函数           | 4  |
| 1.1.4 遗传操作           | 6  |
| 1.1.5 应用于非线性最优化问题    | 8  |
| 1.2 遗传算法应用于组合优化问题的实例 | 17 |
| 1.2.1 配词问题           | 18 |
| 1.2.2 背包问题           | 29 |
| 1.3 混合遗传算法           | 43 |
| 1.3.1 ls-hGA         | 44 |
| 1.3.2 flc-hGA        | 45 |
| 1.4 参考文献             | 49 |
| 第 2 章 网络模型基础         | 51 |
| 2.1 最短路径模型           | 51 |
| 2.1.1 最短路径问题数学模型     | 52 |
| 2.1.2 基于优先级的遗传算法解法   | 54 |
| 2.1.3 数值计算           | 59 |
| 2.2 最大流量模型           | 63 |

|       |                           |     |
|-------|---------------------------|-----|
| 2.2.1 | 最大流量问题的数学模型 .....         | 64  |
| 2.2.2 | 基于优先级编码的遗传算法 .....        | 66  |
| 2.2.3 | 数值计算 .....                | 70  |
| 2.3   | 最小费用流模型 .....             | 74  |
| 2.3.1 | 最小费用流问题的数学模型 .....        | 75  |
| 2.3.2 | 基于优先级编码的遗传算法 .....        | 77  |
| 2.3.3 | 数值计算 .....                | 79  |
| 2.4   | 最小生成树模型 .....             | 82  |
| 2.4.1 | 最小生成树问题的数学模型 .....        | 83  |
| 2.4.2 | 基于 PrimPred 的遗传算法解法 ..... | 86  |
| 2.4.3 | 数值计算 .....                | 91  |
| 2.5   | 参考文献 .....                | 93  |
| 第 3 章 | 物流网络模型 .....              | 95  |
| 3.1   | 物流模型 .....                | 95  |
| 3.1.1 | 配送计划模型 .....              | 96  |
| 3.1.2 | 基于矩阵的遗传算法解法 .....         | 100 |
| 3.1.3 | 基于生成树的遗传算法解法 .....        | 106 |
| 3.1.4 | 数值计算 .....                | 115 |
| 3.2   | 两阶段物流模型 .....             | 116 |
| 3.2.1 | 两阶段物流模型 .....             | 117 |
| 3.2.2 | 基于优先级的遗传算法解法 .....        | 118 |
| 3.2.3 | 数值计算 .....                | 123 |
| 3.3   | 车辆配送模型 .....              | 126 |
| 3.3.1 | 多配送中心带时间窗的车辆配送模型 .....    | 126 |
| 3.3.2 | 基于遗传算法的解法 .....           | 131 |
| 3.3.3 | 数值计算 .....                | 135 |
| 3.4   | 工厂—配送中心物流模型 .....         | 137 |
| 3.4.1 | P-DC 物流网络数学模型 .....       | 138 |
| 3.4.2 | 基于优先级的遗传算法解法 .....        | 142 |
| 3.4.3 | 数值计算 .....                | 145 |
| 3.5   | 参考文献 .....                | 149 |

|                                                |     |
|------------------------------------------------|-----|
| 第 4 章 多目标遗传算法                                  | 150 |
| 4.1 多目标优化模型概要                                  | 150 |
| 4.1.1 多目标优化问题                                  | 151 |
| 4.1.2 Pareto 最优解                               | 151 |
| 4.2 多目标遗传算法概要                                  | 153 |
| 4.2.1 多目标遗传算法的处理过程                             | 154 |
| 4.2.2 向量评价遗传算法                                 | 155 |
| 4.2.3 评价值共享                                    | 157 |
| 4.3 多目标遗传算法过程                                  | 158 |
| 4.3.1 Pareto 排序评价方法                            | 159 |
| 4.3.2 多目标函数加权和评价方法                             | 161 |
| 4.3.3 多目标函数的加权及保存精英策略的引入                       | 163 |
| 4.4 Pareto 最优解的评价                              | 168 |
| 4.4.1 参照解集 $S^*$                               | 168 |
| 4.4.2 求得的 Pareto 最优解数量 $ S_j $                 | 168 |
| 4.4.3 获得 Pareto 最优解个体数比例 $R_{\text{NDS}}(S_j)$ | 169 |
| 4.4.4 Pareto 最优解集与参照解集间的距离 $D1_R$              | 169 |
| 4.4.5 各目标函数轴的最大值, 最小值, 平均值 $I_{\text{MMA}}$    | 170 |
| 4.5 多目标遗传算法的数值计算                               | 170 |
| 4.5.1 数值计算实例 1                                 | 170 |
| 4.5.2 数值计算实例 2                                 | 176 |
| 4.6 参考文献                                       | 180 |
| 第 5 章 多目标网络模型                                  | 182 |
| 5.1 最小费用最大流量网络模型                               | 182 |
| 5.1.1 最小费用最大流量网络的数学模型                          | 184 |
| 5.1.2 基于优先级的遗传算法解法                             | 185 |
| 5.1.3 数值计算                                     | 192 |
| 5.2 多目标供应链网络模型                                 | 199 |
| 5.2.1 多目标供应链网络数学模型                             | 201 |
| 5.2.2 基于优先级的遗传算法求解                             | 203 |
| 5.2.3 数值计算                                     | 206 |
| 5.3 生产物流系统网络模型                                 | 209 |

|       |                          |     |
|-------|--------------------------|-----|
| 5.3.1 | 生产物流系统的数学模型·····         | 213 |
| 5.3.2 | 基于随机值的多阶段决策遗传算法的解法·····  | 215 |
| 5.3.3 | 数值计算·····                | 225 |
| 5.4   | 通信系统可靠性网络·····           | 233 |
| 5.4.1 | 系统瘫痪率和总成本最小化的数学模型建立····· | 235 |
| 5.4.2 | 基于混合多目标遗传算法的解法·····      | 237 |
| 5.4.3 | 数值计算·····                | 241 |
| 5.5   | 参考文献·····                | 248 |

# 第 1 章

## CHAPTER 1

# 遗传算法

遗传算法(Genetic Algorithms, GA)是 1960 年提出的一种基于生物遗传机理,即生物进化(自然淘汰、交叉、变异等)现象的随机搜索算法,它通过计算机模拟生物进化过程来进行搜索和进化,最后寻求最优解<sup>[1-18]</sup>。一般而言,遗传算法是一种将自然淘汰和遗传现象的机理简单化之后得到的超启发式算法(metaheuristics),其中被称为个体的染色体(即研究对象的候选解)的集合能适应外部环境(研究对象的评价函数),并基于下面的规则在每代中生成新个体集合:

- (1) 越是适应性强的个体,其生存几率越高(自然选择);
- (2) 以原有个体为基础通过遗传操作生成新个体(遗传现象)。

遗传算法用于求解组合优化问题时,规则(1)可看做是概率搜索法,规则(2)可看做是经验搜索法,因此遗传算法是一种将两者有机结合的超启发式算法。

## 1.1 遗传算法基础

### 1.1.1 遗传算法概述

生物体通过周而复始的自然淘汰、交叉和变异以适应环境的变化,从而完成进化。作为元启发式算法(metaheuristics)的一种,遗传算法(Genetic Algorithms, GA)是基于模拟生物的进化过程并将这一过程简单化的算法。它通过数学建

模,采用基于遗传机理的随机搜索来寻找实际问题的最优解。可以说,遗传算法是将生物“对环境的适应”机制应用于求解 NP-hard 组合问题或者数学模型比较复杂的最优化问题<sup>[19,20,22]</sup>,从而得到最佳解。

遗传算法首先随机生成满足实际问题约束的候选解(该候选解形成的集合被称为初始解集),再对候选解反复进行交叉和变异的遗传操作,从而获得最优解或最满意解(对生物而言,即为对环境具有最佳的适应度)。其基本处理流程将在下文中介绍。

### a. 对实际问题的编码

对实际的生物而言,遗传信息的传递载体称为染色体(chromosome),或叫做个体(individual)。在遗传算法中将实际问题的候选解当做染色体,其通常的表现形式为一维或二维的二进制位、字符串或数据列。这就是实际问题的基因表现(representation)或编码(code)。这种编码化的解被称为基因型(gene type),而实际的解被称为表现型(phenotype)。

例如,如图 1.1 所示的旅行商问题(Travelling Salesman Problem, TSP),要求不重复地访问九州地区所有 9 个主要城市,可将图 1.2 中所示的染色体作为一种候选解,其中遗传基因编号表示各城市的访问顺序,遗传基因的值表示各城市的编号。

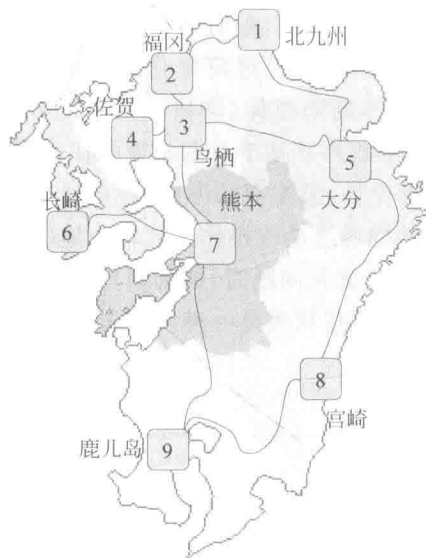


图 1.1 旅行商问题举例

|         |   |   |   |   |   |   |   |   |   |
|---------|---|---|---|---|---|---|---|---|---|
| 遗传基因编号： | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 遗传基因的值： | 1 | 5 | 8 | 9 | 7 | 6 | 4 | 3 | 2 |

图 1.2 染色体举例

### b. 生成初始染色体集合

根据实际问题的约束条件,随机生成一定数量(种群数, population size)的染色体,形成初始的染色体集合(种群, population)作为初始解。这一处理过程称为编码(encoding routine)。与此相应,将染色体还原为实际问题的解的过程称为解码(decoding routine)。

### c. 评价各染色体适应度的适值函数

对于初始集合的各染色体,首先用适值函数(fitness function)来求得各染色体的适应度。适值函数是根据实际问题的目标函数来定义、用来评价染色体对外部环境适应性优劣的函数。适应度越高的染色体被选中存留到下一代的概率越高。

### d. 遗传操作

下一代候选解的生成是通过将生物遗传现象模型化的遗传操作(genetic operation)来实现的。遗传操作有如下三种形式:

**交叉(crossover):**交叉是指双亲(parents)的两个染色体间,部分基因相互交换的操作,即通过双亲中一方的部分基因与另一方的部分基因重新组合,生成新的染色体,即子代(offspring)。

**变异(mutation):**变异是指通过改变父代染色体中的部分遗传基因,从而形成新的子代染色体的操作,这是一种用于保持染色体集合多样性的遗传操作。

**选择(selection):**根据各染色体的适值,选择适应度高的父代、子代染色体,从而形成下一代染色体集合的操作。

### e. 遗传算法中各参数值的设定

- 种群数(population size:  $popSize$ ): 解集(种群)中包含的染色体数目
- 最大代数(maximum generation:  $maxGen$ ): 进化过程的循环次数
- 交叉率(crossover probability:  $p_c$ ): 进行交叉操作的概率
- 变异率(mutation probability:  $p_m$ ): 进行变异操作的概率

各参数的值需要在初始阶段设定。这些值对遗传算法的解的搜索效率和最终解的质量都有很大的影响。因此,可通过预备实验等方法,针对每个实际问题取各参数的经验值。

基本遗传算法(basic GA, bGA)的结构及运算流程如图 1.3 所示。而多数情况下遗传算法会根据实际问题的特点,对其结构进行改变。

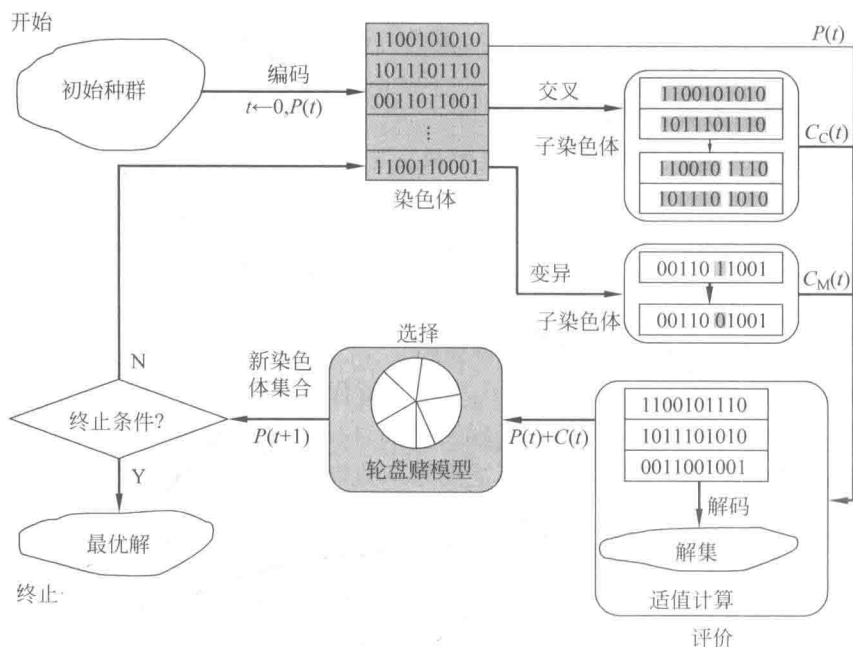


图 1.3 基本遗传算法流程图

下面将对各处理过程做更加详细的介绍。

### 1.1.2 编码

编码和适值函数的设定是应用遗传算法解决实际问题中极为关键的要素。所谓编码就是将求解问题的解空间映射到编码空间(即搜索空间)的过程,它对搜索过程影响很大。编码方式可分为一维或二维的数据列、二进制列或字符串。一般情况下,如果可行解空间(feasible solution space)中的所有点与编码空间中的所有点一一对应,如图 1.4(a)所示,可直接应用基本遗传算法或者稍加改进即可。但是如果编码空间内的一部分点对应到非可行解空间(infeasible solution space),如图 1.4(b)所示,则要对遗传算法进行变更和改进。因此,充分掌握实际问题的构造和性质是编码设计的重要基础。

### 1.1.3 适值函数

适值函数给定了了解的搜索方向,并直接关系到最终解的质量和搜索效率。对于工程问题,一般采用成本或时间等可评价的物理量(例如在调度问题中的最

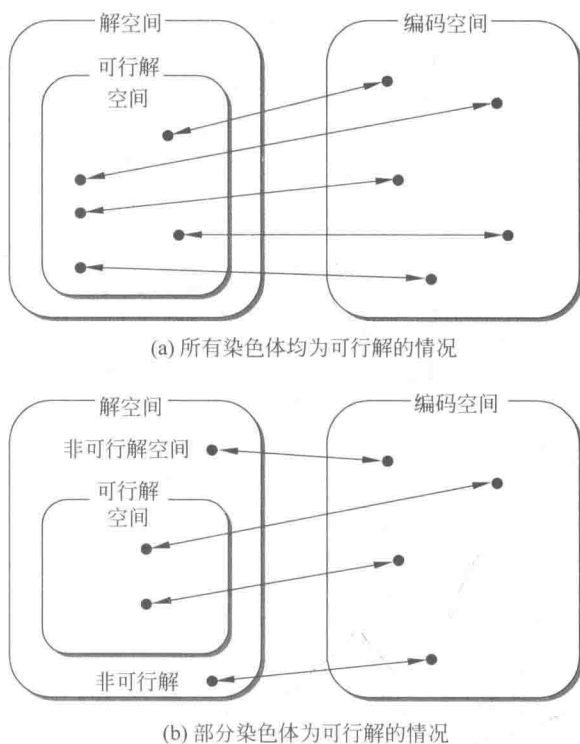


图 1.4 解空间和编码空间的关系

大完成时间)作为适值函数。对于已建立数学模型的问题,则一般直接采用其目标函数作适值函数。但是对于多目标最优化问题,要进行单目标化等变换。同编码一样,适值函数的确定也需要充分了解并准确把握实际问题的构造和性质。

惩罚函数(penalty function)法可解决约束优化问题中产生非可行解(infeasible solution)的问题。它通过对非可行解施加惩罚,以此来降低其在下一代的生存率。然而,在遗传算法中并不是完全否定所有的非可行解,而是在后代中保留少数的候选非可行解。这是因为在非可行解中也有可能包含靠近最优解的元素。从图 1.5 所示的解空间中可看出,当点  $a$  和  $c$  位于可行解域(feasible area),  $b$  和  $d$  位于非可行解域(infeasible area)时,显然非可行解  $b$  与最优解  $a$  的距离比可行解  $c$  与  $a$  的距离近。然而在通常的最优化问题中,没有如此清楚的解空间图,因此很难判断哪一个解更优越。

目前,还没有有效确定惩罚函数的一般性方法,因此要针对每个实际问题确定有效的惩罚函数。

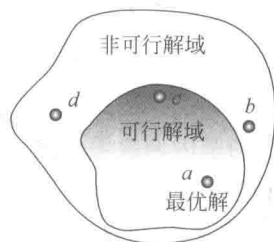


图 1.5 解空间

### 1.1.4 遗传操作

在遗传算法中,遗传操作是提高算法效率的重要组成部分。通常根据实际问题的特点,针对染色体的结构特征,采用不同的遗传操作。在这里将介绍比较常用的几种方法。

#### a. 交叉

交叉(crossover)是改进遗传编码最有效的操作。交叉的基本方法有单点交叉(simple point crossover)、复点交叉(multipoint crossover)、均匀交叉(uniform crossover)等。在很多工程问题中,采用这些交叉方法生成的子染色体往往是非可行解。因此,或者需要重新考虑最适合遗传算法的交叉方法,或者采用修复子染色体为可行解等策略。特别是在行程安排这类最典型的旅行商问题中,提出了多种适用于染色体中基因不允许重复的交叉方法,如部分一致交叉(Partial-Mapped crossover, PMX)、OX(Order crossover)、PBX(Position-Based crossover)等,这些方法的详细说明,请参见文献[3,4]。

下面举个简单例子说明单点交叉的过程。单点交叉是先确定一个交叉的位置,并由双亲中一方染色体的前一部分和另一方染色体的后一部分来构成一个子染色体的操作,也叫做一点交叉。单点交叉的实例如图1.6所示。

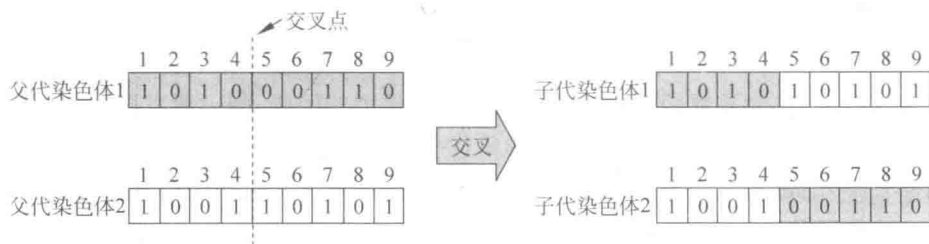


图 1.6 单点交叉的实例

#### b. 变异

变异(mutation)的方法需要根据具体的编码方式来确定。例如,若采用一维的二进制编码,如图1.7所示,可取染色体中的任意一位进行取反。若采用包含数值和文字的编码,则可采用将任意位置的基因,与其前后的基因或其他任意位置的基因调换等方法。然而这样的操作对染色体的可行性影响很大,因此通常结合实际问题考虑其变异方法。例如,针对旅行商问题中的不允许重复这一因素,有逆位(inversion)、插入(insertion)、置换(displacement)、复制(duplication)等变异方法[3]。



图 1.7 二进制染色体的反转变异

### c. 选择

常用的选择(selection)方法有适值比例法(proportional selection method)、期望值选择法(expected value selection method)、排序选择法(ranking selection method)和精英保存法(elite preservation method)等<sup>[3]</sup>。其中适值比例法也叫轮盘赌模型或蒙特卡罗模型,其基本原理是根据每个染色体适值的比例来确定该个体被选择的概率或生存的概率。期望值选择法是通过计算染色体集合中每个个体在下一代中生存的期望值来进行选择,若某个体被选中并要参与交叉、变异,则它在下一代中生存的期望值数减去 0.5。采用此种方法,即使是在最差的情况下,也可以以期望值 0.5 的偏差选择下一代的染色体。排序选择法首先将染色体按照适值大小进行排序,然后根据事先确定的各次序对应的概率选择下一代染色体。精英保存法是在染色体集合中选择适值最高的个体开始依次保存至下一代。但若持续使用该方法,则很容易造成精英染色体在种群中迅速扩大,从而有陷入局部解的可能,因此在一般情况下建议与其他选择方法组合使用。

遗传算法的处理流程一般可用下面的 Pascal 伪代码表示。应用于网络设计问题时,程序需要根据具体情况作相应变更或增加附加条件,但基本上遵循此流程。

遗传算法一般搜索流程

---

```

procedure: basic GA
input: problem data, GA parameters
output: the best solution
begin
     $t \leftarrow 0$ ;                                     //  $t$ : 遗传代数
    initialize  $P(t)$  by encoding routine;           //  $P(t)$ : 第  $t$  代的父代染色体
    evaluate  $eval(P)$  by decoding routine;
    while (not termination condition) do
        create  $C(t)$  from  $P(t)$  by crossover routine; //  $C(t)$ : 第  $t$  代的子代染色体
        create  $C(t)$  from  $P(t)$  by mutation routine;
        evaluate  $eval(C)$  by decoding routine;
        select  $P(t+1)$  from  $P(t)$  and  $C(t)$  by selection routines;
         $t \leftarrow t+1$ ;
    end
    output the best solution;
end

```

---

### 1.1.5 应用于非线性最优化问题

#### 例 多峰非线性最优化问题的实例

下面考虑两个变量的非线性最优化(nonlinear optimization)问题,这个问题具有多个局部解,是多峰问题。目标函数的三维图形如图 1.8 所示,由于多峰最优化问题具有多个局部解的特性,使用传统的方法非常难以获取最优解。

$$\max f(x_1, x_2) = 21.5 + x_1 \sin(4\pi x_1) + x_2 \sin(20\pi x_2) \quad (1.1)$$

$$\text{s. t.} \quad -3.0 \leq x_1 \leq 12.1 \quad (1.2)$$

$$4.1 \leq x_2 \leq 5.8 \quad (1.3)$$

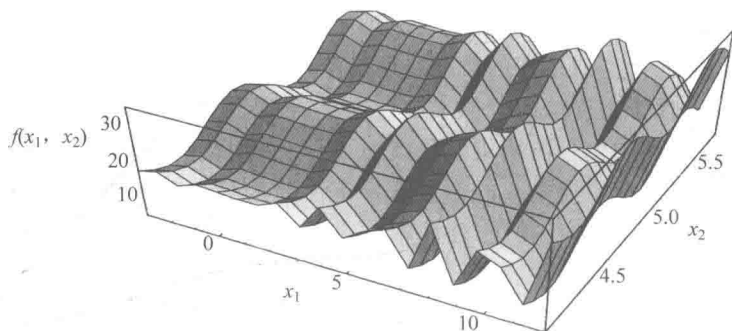


图 1.8 多峰非线性最优化问题的目标函数

下面,我们根据决策变量  $x_1, x_2$  的取值范围,介绍遗传算法求解多峰非线性最优化问题的过程。

#### a. 染色体设计

编码: 这里以二进制字符串来表示决策变量  $x_1, x_2$ 。例如,  $x_j$  的定义域是  $[a_j, b_j]$ ,而要求的精度是小数点后 5 位,这就要求  $x_j$  的定义域至少要划分为  $(b_j - a_j) \times 10^5$  个空间。设变量  $x_j$  所需要的二进制字符串长为  $m_j$ ,则应满足:

$$2^{m_j-1} < (b_j - a_j) \times 10^5 < 2^{m_j} - 1 \quad (1.4)$$

要想将变量  $x_j$  由二进制转为十进制,可根据定义域范围内  $2^{m_j} - 1$  个划分点的位置,按下式计算:

$$x_j = a_j + \text{decimal}(\text{substring}_j) \times \frac{b_j - a_j}{2^{m_j} - 1} \quad (1.5)$$

这里  $\text{decimal}(\text{substring}_j)$  表示变量  $x_j$  的子串  $\text{substring}_j$  的十进制值。

假设变量  $x_1, x_2$  需要的精度都是小数点后 5 位,则两个变量所需要的总