《最优化基础——模型与方法》系列教材

现代优化计算方法

邢文训 谢金星 编著

清华大学出版社
http://www.tup.tsinghua.edu.cn

《最优化基础——模型与方法》系列教材

现代优化计算方法

邢文训　谢金星　编著

清华大学出版社

(京)新登字 158 号

内 容 简 介

本书系统介绍了禁忌搜索、模拟退火、遗传算法、人工神经网络和拉格朗日松弛等现代优化计算方法的模型与理论、应用技术和应用案例。

全书共 6 章,第 1 章介绍算法复杂性的基本概念和启发式算法的评价方法,后 5 章分别介绍各个现代优化计算方法。

本书可作为数学、管理科学、计算机科学、工业工程等学科中相关优化专业的研究生教材,也可供相关专业研究人员参考。

图书在版编目(CIP)数据

现代优化计算方法/邢文训,谢金星编著.—北京:清华大学出版社,1999.8

"最优化基础——模型与方法"系列教材

ISBN 7-302-03610-1

Ⅰ.现… Ⅱ.① 邢…,② 谢… Ⅲ.计算方法-最优化-高等学校-教材 Ⅳ.O241

中国版本图书馆 CIP 数据核字(1999)第 24726 号

出版者:清华大学出版社(北京清华大学校内,邮编 100084)
http://www.tup.tsinghua.edu.cn

印刷者:清华大学印刷厂

发行者:新华书店总店北京发行所

开　本:850×1168 1/32　印张:9.75　字数:252 千字

版　次:1999 年 8 月第 1 版　1999 年 8 月第 1 次印刷

书　号:ISBN 7-302-03610-1/O·215

印　数:0001～4000

定　价:13.50 元

《最优化基础——模型与方法》系列教材序言

最优化是人们在工程技术、科学研究和经济管理的诸多领域中经常遇到的问题。结构设计要在满足强度要求等条件下使所用材料的总重量最轻；资源分配要使各用户利用有限资源产生的总效益最大；安排运输方案要在满足物资需求和装载条件下使运输总费用最低；编制生产计划要按照产品工艺流程和顾客需求，尽量降低人力、设备、原材料等成本使总利润最高。可以预料，随着科学技术尤其是计算机技术的不断发展，以及数学理论与方法向各门学科和各个应用领域的更广泛、更深入的渗透，在即将到来的21世纪信息时代，最优化理论和技术必将在社会的诸多方面起着越来越大的作用。

解决实际生活中优化问题的手段大致有以下几种：一是靠经验的积累，凭主观作判断；二是做试验选方案，比优劣定决策；三是建立数学模型，求解最优策略。虽然由于建模时要作适当简化，可能使结果不一定非常完善，但是它基于客观数据，求解问题简便、灵活、经济，而且规模可以很大(将来会越来越大)。人们还可以吸收从经验得到的规则，用实验来不断校正建立的模型。随着数学方法和计算机技术的进步，用建模和数值模拟解决优化问题这一手段，将会越来越显示出它的效能和威力。显然，在决策定量化、科学化的呼声日益高涨的今天，数学建模方法的推广应用是符合时代潮流和形势发展需要的。

最优化理论、模型与方法所包含的内容很多，国内已出版了不少教材和专著介绍其各个分支。但是一方面，近年来发展起来的、有着广泛应用背景的规划模型(如随机规划、模糊规划等)，以及一些已经为许多人采用、受到广泛关注的优化算法(如模拟退火、遗

传算法等），还缺乏详细和系统的介绍；另一方面，一些偏重优化理论和方法的教材，其要求难以与工科学生的数学知识衔接，也缺少对于应用来说十分重要的建模过程和软件介绍，而一些比较通俗的运筹学教材，则在加强理论基础，适应学生将来从事科研工作需要上考虑较少。我们这套教材试图弥补以上两方面的缺陷，力求体现下述特点：

1. 内容既包含传统的线性规划与非线性规划等部分，又纳入有广泛应用前景的随机规划和模糊规划；在传统内容中，既注重典型的数学思想和方法的系统叙述，又引入丰富的建模实例。

2. 数学基础既与工科学生所学知识衔接，又考虑到研究生阅读文献、从事科研工作的需要，适当提高理论基础的起点。

3. 对一般教材介绍的诸多算法进行精选，配合介绍一些应用软件，并引入近年来迅速发展的若干新算法。

本系列教材将陆续出版，首批四册为：《线性与非线性规划》、《网络优化》、《现代优化计算方法》、《随机规划与模糊规划》。

由于水平所限，书中难免有缺陷和错误，诚恳希望读者予以批评指正。

《最优化基础——模型与方法》系列教材编委会

1998 年 5 月

系列教材编委会成员名单

（姓氏笔划为序）

序　言

随着 20 世纪 80 年代初期禁忌搜索、模拟退火、遗传算法和人工神经网络算法的兴起，科学工作者对这些算法的模型、理论和应用技术等一系列问题进行着深入的研究，并将这些算法称为现代优化算法。现代优化算法的主要应用对象是优化问题中的难解问题，也就是优化理论中的 NP-hard 问题。正是因为很多实际优化问题的难解性，和现代优化算法在一些优化问题中的成功应用，使得现代优化算法成为解决优化问题的一种有力工具。科学工作者对现代优化算法抱以极大的热情和期望，几乎在各种难解的优化问题中，他们都尝试这些算法的应用和比较其应用效果。

在国际和国内，每一种现代优化算法都有相应的专著，其中有较为详尽的理论和应用论述。很多科学工作者期望对这些算法有一定的了解，以在实际应用中有目的地采用。基于人们的这些期望和关注，我们将这些算法集于一书，从简单实例的应用，到其理论、应用技术及应用案例的深入分析，由易到难地向相关优化专业的硕士、博士研究生介绍现代优化算法的相关模型、理论、应用技术和一些应用实例，在国内，这可能是首次尝试。本书也可供科学研究人员在这些领域研究时参考。

我们不希望将现代优化算法中的诸多算法简单地堆集在一起，而是设法将它们有机地集于一书。基于这种想法，本书由 6 章组成。第 1 章介绍了现代优化算法要解决的问题及它们中的共同点，并将本书各章衔接在一起。第 2 章、第 3 章、第 4 章和第 5 章分

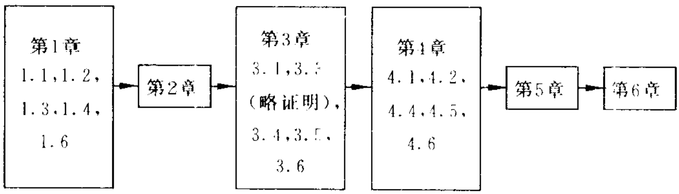
别介绍禁忌搜索算法、模拟退火算法、遗传算法和人工神经网络算法，这些是现代优化算法的组成。第 6 章提供评价算法的一种工具：拉格朗日松弛算法。

第 1 章为概论。首先介绍现代优化算法所要解决的组合优化问题。通过复杂性概念的引入，使得我们知道为什么和在什么情况下将现代优化算法应用到优化问题。通过邻域和算法评价方法的介绍，使我们找出现代优化算法的一些共同点。由于有关复杂性概念的内容不易理解，因此，作者在处理这部分内容时，以多个典型组合优化问题为背景，通过对它们的一步步分析来介绍复杂性的一个个概念。为了适应不同层次的读者，本章将复杂性概念的内容分为 1.2 节和 1.5 节两部分。1.2 节介绍了多项式时间可求得最优解的多项式问题。1.5 节更进一步介绍了 NP、NP-complete 和 NP-hard 概念。对学时要求较少或非运筹学专业学生的教学，可以略去 1.5 节。

将第 2,3,4,5 这四章的内容作为一个整体，从最容易理解的局部搜索算法开始，逐步深入地介绍全局搜索的禁忌搜索算法，带有随机搜索的模拟退火和遗传算法，最后，给出人工神经网络算法。对学时要求较少或非运筹学专业学生的教学，可以略去 3.2 节、3.3 节和 4.3 节中的证明。

第 6 章拉格朗日松弛算法使得本书成为一个整体。我们不仅要学会应用现代优化算法，还应该学会评价这些算法。对于极小化目标函数的优化问题，现代优化算法能给出一个目标值不低于最优目标值的可行解，当评价一个算法的计算效率时，可行解目标值同最优目标值一个下界的差是评价的标准之一。拉格朗日松弛算法则是提供最优目标值下界的工具之一。

对学时要求较少或非运筹学专业学生的教学，可以参考下图教学内容：



本书是在1997年春季我校研究生“现代最优化算法”课程讲稿的基础上，经过1998年和1999年两次讲授后整理和扩充而成的。在此过程中，研究生们给予了非常有益的建议。自90年代初期，姜启源教授和韩继业教授投入了很大的心血，组织了组合优化问题讨论班并得到国家自然科学基金资助，这个讨论班和基金资助使我们积累了大量有价值的材料。在写作的过程中，香港城市大学的黎建强和林国健教授提供的资料大大丰富了书中的内容。谭泽光、胡冠章、林翠琴、刘宝碇教授和张家伟同学等阅读并修改了我们的手稿，给我们提出很多很好的建议。在此，对给予我们帮助的老师、出版社的编辑和同学们表示感谢！对给予我们部分资助的教育部高等学校工科数学教育基地表示感谢！最后，对我们家人的耐心和支持表示由衷的谢意！

由于我们的水平有限，恳请读者对本书的不足之处批评指正。

邢文训　谢金星

1999年春于清华园

目　　录

第1章
概　　论

现代优化算法包括禁忌搜索(tabu search)、模拟退火(simulated annealing)、遗传算法(genetic algorithms)、神经网络(neural networks)和拉格朗日松弛等算法.这些算法涉及生物进化、人工智能、数学和物理科学、神经系统和统计力学等概念,都是以一定的直观基础而构造的算法,我们称之为启发式算法.启发式算法的兴起与计算复杂性理论的形成有密切的联系.当人们不满足常规算法求解复杂问题时,现代优化算法开始体现其作用.现代优化算法自80年代初兴起,至今发展迅速.例如,在1996年,Osman在文[1]中分类罗列了1400篇相关文章.因此,这些算法同人工智能、计算机科学和运筹学相融合就不足为奇了.本章主要介绍组合优化问题、计算复杂性和启发式算法的概念.

1.1　组合最优化问题

组合最优化(combinatorial optimization)是通过对数学方法的研究去寻找离散事件的最优编排、分组、次序或筛选等,是运筹学(operations research)中的一个经典且重要的分支,所研究的问题涉及信息技术、经济管理、工业工程、交通运输、通信网络等诸多领域.该问题可用数学模型描述为:

$$\begin{aligned}&\min f(\boldsymbol{x})\\&\text{s.t. } g(\boldsymbol{x}) \geq 0,\\&\quad\ \ \boldsymbol{x} \in D,\end{aligned}$$

其中，$f(\boldsymbol{x})$ 为目标函数，$g(\boldsymbol{x})$ 为约束函数，$\boldsymbol{x}$ 为决策变量，D 表示有限个点组成的集合.

一个组合最优化问题可用三参数(D,F,f)表示，其中 D 表示决策变量的定义域，F 表示可行解区域 $F=\{\boldsymbol{x}\mid\boldsymbol{x}\in D,\ g(\boldsymbol{x})\geqslant 0\}$，$F$ 中的任何一个元素称为该问题的可行解，f 表示目标函数. 满足 $f(\boldsymbol{x}^*)=\min\{f(\boldsymbol{x})\mid\boldsymbol{x}\in F\}$ 的可行解 $\boldsymbol{x}^*$ 称为该问题的最优解. 组合最优化的特点是可行解集合为有限点集. 由直观可知，只要将 D 中有限个点逐一判别是否满足 $g(\boldsymbol{x})$ 的约束和比较目标值的大小，该问题的最优解一定存在和可以得到. 因为现实生活中的大量优化问题是从有限个状态中选取最好的，所以大量的实际优化问题是组合最优化问题.

例 1.1 0-1 背包问题(knapsack problem)

设有一个容积为 b 的背包，n 个体积分别为 $a_i(i=1,2,\cdots,n)$，价值分别为 $c_i(i=1,2,\cdots,n)$ 的物品，如何以最大的价值装包?这个问题称为 0-1 背包问题. 用数学模型表示为：

$$\max\sum_{i=1}^{n}c_ix_i \tag{1.1}$$

$$\text{s.t.}\ \sum_{i=1}^{n}a_ix_i\leqslant b, \tag{1.2}$$

$$x_i\in\{0,1\},i=1,\cdots,n. \tag{1.3}$$

其中目标(1.1)欲使包内所装物品的价值最大，(1.2)为包的能力限制，(1.3)表示 x_i 为二进制变量，$x_i=1$ 表示装第 i 个物品，$x_i=0$ 表示不装. 此时，$D=\{0,1\}^n$，F 为 D 中满足(1.2)的可行解. f 为目标函数. □

例 1.2 旅行商问题(TSP，traveling salesman problem)

一个商人欲到 n 个城市推销商品，每两个城市 i 和 j 之间的距离为 d_{ij}，如何选择一条道路使得商人每个城市走一遍后回到起点且所走路径最短. TSP 还可以细分为对称和非对称距离两大类问

题. 当 $d_{ij} = d_{ji}, \forall\ i,j$ 时,称为对称距离 TSP,否则称非对称距离 TSP. 对一般的 TSP,它的一种数学模型描述为:

$$\min \sum_{i \neq j} d_{ij} x_{ij} \tag{1.4}$$

$$\text{s.t.} \sum_{j=1}^{n} x_{ij} = 1,\ i = 1,2,\cdots,n, \tag{1.5}$$

$$\sum_{i=1}^{n} x_{ij} = 1,\ j = 1,2,\cdots,n, \tag{1.6}$$

$$\sum_{i,j \in s} x_{ij} \leqslant |s| - 1, 2 \leqslant |s| \leqslant n - 2, s \subset \{1,2,\cdots,n\}, \tag{1.7}$$

$$x_{ij} \in \{0,1\},\ i,j = 1\cdots,n,\ i \neq j. \tag{1.8}$$

以上是基于图论的数学模型. 其中,(1.8) 中的决策变量 $x_{ij} = 1$ 表示商人行走的路线包含从城市 i 到城市 j 路径,$x_{ij} = 0$ 表示商人没有选择走这条路. $i \neq j$ 的约束可以减少变量的个数,使得共有 $n \times (n-1)$ 个决策变量. 目标(1.4)要求距离之和最小. (1.5) 要求商人从城市 i 出来一次,(1.6) 要求商人走入城市 j 只有一次. (1.5) 和(1.6) 表示每个城市经过一次. 仅有(1.5) 和(1.6) 的约束无法避免回路的产生,一条回路是由 $k(1 \leqslant k \leqslant n)$ 个城市和 k 条弧组成,因此,(1.7) 约束旅行商在任何一个城市子集中不形成回路,其中 $|s|$ 表示集合 s 中元素个数. 此时,$D = \{0,1\}^{n \times (n-1)}$,$F$ 为 D 中满足(1.5),(1.6) 和(1.7) 的可行解. f 为目标函数. □

上面两个问题都是组合优化中的经典问题. 它们的共性是读者容易接受这些问题,可行解集合是有限的,在问题的规模较小时,通过枚举很容易得到最优解.

例 1.3 整数线性规划(integer linear programming)

$$\begin{aligned} & && \min \boldsymbol{c}^{\mathrm{T}} \boldsymbol{x} \\ & (\mathrm{IP}) && \text{s.t. } \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \end{aligned}$$

$$x \geqslant 0, x \in \mathbf{Z}^n.$$

其中,$\boldsymbol{c}$ 为 n 维列向量,$\boldsymbol{A}$ 为 $m \times n$ 矩阵,$\boldsymbol{b}$ 为 m 维列向量,$\boldsymbol{x}$ 为 n 维决策变量,$\mathbf{Z}^n$ 表示 n 维整数集合. 模型中的系数 $\boldsymbol{A}$、$\boldsymbol{b}$ 和 $\boldsymbol{c}$ 的元素都是整数. □

例 1.2 和 1.3 的数学模型都具有(IP) 的形式. 一些组合优化问题可以写成整数线性规划问题. 整数线性规划同线性规划形式上非常相似,不同之处是前者的决策变量部分或全部取整数.

在本书中,我们假设线性整数规划的系数是整数. 这主要是因为目前在计算机计算中都是以有限位符号储存和运算. 很容易证明,IP 中的系数是有理数时都可以处理成整数情况.

例 1.4 装箱问题(bin packing)

如何以个数最少的尺寸为 1 的箱子装进 n 个尺寸不超过 1 的物品,该问题为装箱问题. □

例 1.5 约束机器排序问题(参考[2])(capacitated machine scheduling)

n 个加工量为 $\{d_i | i = 1,2,\cdots,n\}$ 的产品在一台机器上加工,机器在第 t 个时段的工作能力为 c_t,求完成所有产品加工的最少时段数. 它的数学模型为

$$\min T \tag{1.9}$$

$$\text{s.t.} \sum_{t=1}^{T} x_{it} = 1,\ i = 1,2,\cdots,n, \tag{1.10}$$

$$\sum_{i=1}^{n} d_i x_{it} \leqslant c_t,\ t = 1,2,\cdots,T, \tag{1.11}$$

$$x_{it} \in \{0,1\},\ i = 1,2,\cdots,n;\ t = 1,2,\cdots,T, \tag{1.12}$$

其中,x_{it},T 为决策变量,$x_{it} = 1$ 表示第 t 时段加工产品 i. (1.9) 要求加工所用的时段数最少,(1.10) 表示产品 i 一定在某一个时段加工,(1.11) 表示每个时段的加工量不超过能力的限制. □

有些优化问题可以用整数规划模型的形式表示,如例 1.1 和

1.2. 有些组合优化问题用整数规划模型表示则比较复杂和不易被理解，不如对问题采用直接叙述更易理解，如例 1.2、1.4 和 1.5. 因此，有大量的组合优化问题是通过文字语言叙述的.

1.2　计算复杂性的概念

由组合最优化问题的定义可知，每一个组合最优化问题都可以通过枚举的方法求得最优解. 枚举是以时间为代价的，有的枚举时间还可以接受，有的则不可能接受. 如例 1.2 的非对称距离 TSP 问题，可以用另一个方法来表示它的可行解：用 n 个城市的一个排列表示商人按这个排列序推销并返回起点. 若固定一个城市为起终点，则需要 $(n-1)!$ 个枚举. 以计算机 1 秒可以完成 24 个城市所有路径枚举为单位，则 25 个城市的计算时间为：以第 1 个城市为起点，第 2 个到达城市有可能是第 2、第 3、…、或第 25 个城市. 决定前两个城市的顺序后，余下是 23 个城市的所有排列，枚举这 23 个城市的排列需要 1 秒，所以，25 个城市的枚举需要 24 秒. 类似地归纳，城市数同计算时间的关系如表 1.1 所示.

表 1.1　枚举时城市数与计算时间的关系

城市数	24	25	26	27	28	29	30	31
计算时间	1s	24s	10min	4.3h	约 4.9d	136.5d	约 10.8a	约 325a

通过表 1.1 可以看出，随着城市数的增多，计算时间增加非常之快，当城市数增加到 30 时，计算时间约 10.8 年，已无法接受.

问题是一个抽象的模型或概念，它通过一些具体的数据表现出来. 问题是需要回答的一般性提问，通常含有若干个满足一定条件的参数. 问题通过下面的描述给定：(1) 描述所有参数的特性，(2) 描述答案所满足的条件. 如 TSP 是一个问题，它由例 1.2 的文

字或模型(1.4)～(1.8)表示，通过给出的城市数和城市间的距离而使这个问题具体化. 目前计算机可以求解的是这些给出城市数和城市间的距离的具体 TSP. 一个算法常常是针对一个问题来设计的，如上面的 TSP 枚举算法可以求解任何一个 TSP 的最优解. 而若用计算机求解，则必须对问题中的参数赋予具体值，如 TSP 中的城市数和城市间的距离，问题中的参数赋予了具体值的例子称为问题的实例(instance). 一个问题通过它的所有实例表现.

衡量一个算法的好坏通常是用算法中的加、减、乘、除和比较等基本运算的总次数同实例的计算机计算时的二进制输入数据的大小关系来度量. 对于一个正整数 x，二进制表示是以

$$x = a_s 2^s + a_{s-1} 2^{s-1} + \cdots + a_1 \times 2 + a_0 \quad (a_s \neq 0)$$

的系数$(a_s a_{s-1} \cdots a_1 a_0)$来表示. 由

$$2^s \leqslant 2^{\log_2 x} = x \leqslant 2^{s+1} \leqslant 2^{\log_2 x + 1},$$

这个二进制数的位数是 $s+1$，也称为数据 x 的输入长度，在 $\log_2 x$ 与 $\log_2 x + 1$ 之间. 于是，x 的二进制数的位数是整数$\lceil \log_2 x \rceil$或$\lceil \log_2 x \rceil + 1$，但不超过$\lceil \log_2 x \rceil + 1$，其中$\lceil x \rceil$表示不小于 x 的最小整数. 虽说 $\log_2 0$ 没有意义和 $\log_2 1 = 0$，特别需要注意的是整数 0 和 1 的二进制位数都是 1. 可以看到，任何一个非负整数的输入长度至少为 1. 在此，在特别定义 $\log_2 0 = 0$ 后，上面的二进制表示方法和输入长度的结论可以推广到非负整数. 常规理解的计算机只能用有限位表示一个实数，因此，我们限定只考虑有理数.

如表 1.1 对应的 TSP，当城市数为 n 且第一个城市为始终点时，计算一条路径$(1, i_2, \cdots, i_n)$长度的基本运算为两两城市间距离的 n 个求和. 因为有$(n-1)!$条路径，枚举所有路径进行$(n-1)!$次比较，可以得到最优路径. 这个枚举算法的基本计算总次数为$\{(n-1)!\}n = n!$. 实例的计算机计算时只需输入城市数和任何两个城市的距离. 于是，输入数据是城市数 n 和城市间的距离 $S = \{d_{ij} | 1 \leqslant i, j \leqslant n, i \neq j\}$，且假设这些数为整数. 对任何一个非零距