科学版

研究生教学丛书

非线性最优化理论与方法

（第二版）

王宜举　修乃华　编著

科学出版社

研究生教学丛书

非线性最优化理论与方法

（第二版）

王宜举　修乃华　编著

科学出版社

北京

内 容 简 介

本书系统地介绍了非线性最优化问题的有关理论与方法，主要包括一些传统理论与经典算法，如优化问题的最优性理论，无约束优化问题的线搜索方法、共轭梯度法、拟牛顿方法，约束优化问题的可行方法、罚函数方法和 SQP 方法等，同时也吸收了新近发展成熟并得到广泛应用的成果，如信赖域方法、投影方法等.

本书在编写过程中既注重基础理论的严谨性和方法的实用性，又保持内容的新颖性. 本书内容丰富、系统性强，可作为运筹学专业的研究生和数学专业高年级本科生教材或参考书，也可作为相关专业科研人员的工具参考书.

图书在版编目(CIP)数据

非线性最优化理论与方法/王宜举，修乃华编著. —2 版. —北京：科学出版社，2015.11

(研究生教学丛书)

ISBN 978-7-03-046275-6

Ⅰ. ①非… Ⅱ. ①王… ②修… Ⅲ. ①非线性—最优化算法—研究生—教材 Ⅳ. ①O224

中国版本图书馆 CIP 数据核字(2015) 第 267927 号

责任编辑：李 欣 / 责任校对：张怡君
责任印制：肖 兴 / 封面设计：陈 敬

科学出版社 出版
北京东黄城根北街 16 号
邮政编码：100717
http://www.sciencep.com

北京通州皇家印刷厂 印刷
科学出版社发行 各地新华书店经销
*
2012 年 1 月第 一 版 开本：720 × 1000 1/16
2016 年 1 月第 二 版 印张:16
2016 年 1 月第三次印刷 字数：323 000

定价：58.00 元

第二版前言

本书 2012 年在科学出版社首次出版. 经过几年的教学实践，我们发现了书中的一些问题和不当之处，同时也积累了一定的教学经验. 在此基础上，我们通过吸收国内外同行和广大读者的意见和建议，对本书进行了全面修订. 在修订过程中，我们保留了原书的结构和风貌. 为提高教材质量，我们在选材和叙述上尽量联系理工科专业的实际，力图将概念和问题交代得通俗易懂. 与第一版相比，新版本结构更严谨，逻辑更清晰，更通俗易懂，便于自学.

本书再版之时，我们要向在本书编写过程中为我们提供指导和帮助的王长钰教授、邓乃扬教授、夏尊铨教授、祁力群教授、张国礼教授、孙文瑜教授、倪勤教授、杨新民教授、张立卫教授、汪嵚教授和李声杰教授表示衷心的感谢, 也向给我们提供建议的广大同行和读者表示诚挚的谢意! 感谢国家自然科学基金重点项目 (11431002) 和山东省高校优秀科研创新团队计划经费资助.

新版中存在的问题，恳请广大同行和读者不吝赐教，继续给予指导和指正.

王宜举　修乃华

2016 年 1 月

第一版前言

第二次世界大战期间, 运筹学伴随军事上的需要而产生. 战后, 运筹学开始转向民用工业的运用, 并不断取得进展. 20 世纪 60 年代, 最优化方法发展成为运筹学的一门新兴学科. 而后, 近代科学技术的发展, 特别是电子计算机技术的飞速发展促进了最优化方法的迅速发展. 很快, 这门新兴的基础学科便渗透到各个技术领域, 形成了最优化方法与技术这门应用学科, 并发展出了新的更细的研究分支.

作为运筹学的一个重要研究分支, 非线性最优化问题的研究在近三十年得到快速发展, 新的理论和方法不断出现. 为及时吸收新近发展成熟并在实际中得到广泛应用的成果以应用于教学科研中, 作者参阅国内外关于最优化理论与方法的许多专著、教材和研究文献, 并结合自己的教学实践编写了这本书.

非线性最优化问题的研究内容十分丰富. 但由于篇幅所限, 本书主要对这类问题的传统理论与经典梯度型数值方法及其新的研究进展做了比较详尽的论述, 使读者不但能掌握非线性最优化问题的基础理论和梯度型数值方法的操作流程, 而且能清楚这些方法的设计思想和性能, 从而为设计更有效的优化方法提供理论依据和帮助.

法国数学家拉格朗日说过, 一个数学家, 只有当他走出去, 对他在大街上遇到的第一个人清楚地解释自己的工作时, 他才算完全理解了自己的工作. 实际上, 他定义的这种境界也是每一个数学工作者, 特别是数学教育工作者追求的目标. 因此, 作者在编写本书时力求把对先修课程的要求放得最低, 使读者只需具备多元微积分和线性代数的基础知识即可阅读此书. 同时, 为增强可读性, 作者尽可能多地介绍一些方法和技术的引入背景、思想及其发展历程, 并对书中结论给出了比较详细的证明过程.

在本书十余年的锤炼过程中, 作者得到曲阜师范大学的王长钰教授、中国农业大学的邓乃扬教授、大连理工大学的夏尊铨教授和张立卫教授、香港理工大学的祁力群教授、澳大利亚科廷大学的张国礼教授、南京师范大学的孙文瑜教授、南京航空航天大学的倪勤教授、重庆师范大学的杨新民教授和重庆大学的李声杰教授的悉心指导和帮助, 国内的很多同行也提出了许多宝贵的指导性建议, 在此一并向他们表示诚挚的谢意! 最后, 感谢国家 973 项目 (2010CB732501) 和国家自然科学基金 (11171180) 经费资助.

恳请读者不吝赐教, 来信请发至: wang-yiju@163.com 或 nhxiu@bjtu. edu.cn.

王宜举　修乃华

2012 年 1 月

符 号 表

$\mathbb{R}^n$	n 维欧氏空间
$\langle \boldsymbol{x}, \boldsymbol{y} \rangle$	向量 $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$ 的内积
$\|\boldsymbol{x}\|$	向量 $\boldsymbol{x}$ 的 2- 范数
$\mathbf{0}$	零向量与零矩阵
$\boldsymbol{e}$	分量全为 1 的向量
$\boldsymbol{e}_i$	第 i 个分量为 1 其余分量为 0 的向量
$(x_1; x_2; \cdots; x_n)$	向量 $\boldsymbol{x} = (x_1, x_2, \cdots, x_n)^{\mathrm{T}}$
$\|\boldsymbol{A}\|$	矩阵 $\boldsymbol{A}$ 的谱范数 (2- 范数)
$\|\boldsymbol{A}\|_{\mathrm{F}}$	矩阵 $\boldsymbol{A}$ 的 Frobenius 范数
$\boldsymbol{I}$	单位矩阵
$\boldsymbol{A}^{\mathrm{T}}$	矩阵 $\boldsymbol{A}$ 的转置
$\mathrm{tr}(\boldsymbol{A})$	矩阵 $\boldsymbol{A}$ 的迹
$\mathrm{rank}(\boldsymbol{A})$	矩阵 $\boldsymbol{A}$ 的秩
$\det(\boldsymbol{A})$	矩阵 $\boldsymbol{A}$ 的行列式
$\boldsymbol{A}^{+}$	矩阵 $\boldsymbol{A}$ 的广义逆 (伪逆)
$\kappa(\boldsymbol{A})$	矩阵 $\boldsymbol{A}$ 的条件数
$\mathcal{R}(\boldsymbol{A})$	矩阵 $\boldsymbol{A}$ 的值空间
$\mathcal{N}(\boldsymbol{A})$	矩阵 $\boldsymbol{A}$ 的核空间
$[\boldsymbol{a}_i]_{i \in \mathcal{E}}$	以 $\boldsymbol{a}_i, i \in \mathcal{E}$ 为列构成的矩阵
$\mathrm{span}[\boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_s]$	向量 $\boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_s$ 生成的线性子空间
$\mathrm{diag}(d_1, d_2, \cdots, d_n)$	以 $d_1, d_2, \cdots, d_n$ 为对角元的对角阵
$\nabla f(\boldsymbol{x})$	函数 $f: \mathbb{R}^n \to \mathbb{R}$ 在 $\boldsymbol{x}$ 点的梯度
$\nabla^2 f(\boldsymbol{x}), \nabla_{\boldsymbol{x}\boldsymbol{x}} f(\boldsymbol{x})$	函数 $f(\boldsymbol{x})$ 的 Hesse 阵
$D\boldsymbol{F}(\boldsymbol{x}), D_{\boldsymbol{x}}\boldsymbol{F}(\boldsymbol{x})$	向量值函数 $\boldsymbol{F}: \mathbb{R}^n \to \mathbb{R}^m$ 在 $\boldsymbol{x}$ 点的 Jacobi 矩阵
$\|\mathcal{E}\|$	指标集 $\mathcal{E}$ 中元素的个数
$\mathrm{bd}(\mathcal{S})$	集合 $\mathcal{S}$ 的边界集
$\mathrm{Aff}(\mathcal{S})$	集合 $\mathcal{S}$ 的仿射包
$\mathrm{int}(\mathcal{S})$	集合 $\mathcal{S}$ 的内点集

$\mathrm{cl}(\mathcal{S})$	集合 $\mathcal{S}$ 的闭包
$\mathrm{ri}(\mathcal{S})$	集合 $\mathcal{S}$ 的相对内点集
$N(\boldsymbol{x},\delta)$	$\boldsymbol{x}$ 的 δ 邻域
$\mathcal{K}^{\perp}$	子空间 $\mathcal{K}$ 的正交补空间
$\mathcal{K}^{\circ}$	锥 $\mathcal{K}$ 的极锥
$\mathcal{L}(\boldsymbol{x}_0)$	水平集 $\{\boldsymbol{x}\in\mathbb{R}^n \mid f(\boldsymbol{x})\leqslant f(\boldsymbol{x}_0)\}$
$\varnothing$	空集

目 录

第 1 章　引　论

本章首先给出了非线性最优化问题的有关概念和基础知识, 然后介绍了一些常见的求解方法, 给出了凸集和凸函数的概念及有关性质, 最后讨论了无约束优化问题的最优性条件.

1.1　最优化问题

在现实生活中, 经常会遇到这样一类实际问题, 要求在众多方案中选择一个最优方案. 例如, 在工程设计中, 如何选择参数使设计方案既满足设计要求, 又能降低成本; 资源分配时, 怎样分配现有资源才能使分配方案既满足要求又能获得好的经济效益; 产品设计中, 如何搭配各种原料的比例才能既降低成本又能提高产品的质量; 金融投资中, 如何进行投资组合才能在可接受的风险范围内获取最大的收益. 这类基于现有资源使效益极大化或为实现某目标使成本最低化的问题称为最优化问题.

上述问题可表述成如下数学问题

$$\begin{aligned} \min \quad & f(\boldsymbol{x}) \\ \text{s.t.} \quad & \boldsymbol{x} \in \Omega \end{aligned} \tag{1.1.1}$$

或

$$\min\{f(\boldsymbol{x}) \mid \boldsymbol{x} \in \Omega\}.$$

我们称其为最优化问题. 对极大化目标函数的情形, 可通过在目标函数前添加负号等价地转化为极小化问题. 本书只考虑极小化目标函数的情形.

在 (1.1.1) 中, s.t. 是英文 subject to 的缩写; 数值函数 $f: \mathbb{R}^n \to \mathbb{R}$ 称为目标函数, 又称费用函数或效益函数; $\Omega \subseteq \mathbb{R}^n$ 称为可行域, 又称决策集, 它是在极小化目标函数过程中对决策变量 $\boldsymbol{x}$ 取值范围的界定.

可行域有多种表述形式, 一般用等式和不等式定义, 即

$$\Omega = \{\boldsymbol{x} \in \mathbb{R}^n \mid c_i(\boldsymbol{x}) = 0,\ i \in \mathcal{E};\quad c_i(\boldsymbol{x}) \geqslant 0,\ i \in \mathcal{I}\}.$$

对 $i \in \mathcal{E}$, $c_i(\boldsymbol{x}) = 0$ 称为等式约束, $\mathcal{E}$ 称为等式约束指标集; 对 $i \in \mathcal{I}$, $c_i(\boldsymbol{x}) \geqslant 0$ 称为不等式约束, $\mathcal{I}$ 称为不等式约束指标集.

最优化问题形形色色, 最优化模型多种多样, 人们从不同角度对其进行分类.

(1) 根据有无约束划分 若 $\Omega=\mathbb{R}^n$, 即决策变量 $\boldsymbol{x}$ 是自由变量, 则称 (1.1.1) 为无约束优化问题; 若 $\Omega \subsetneqq \mathbb{R}^n$, 则称 (1.1.1) 为约束优化问题.

约束优化问题和无约束优化问题在某些情形可以相互转化. 如对 n 阶实对称阵 A, 下述两最优化问题等价:

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \frac{\boldsymbol{x}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x}}{\boldsymbol{x}^{\mathrm{T}}\boldsymbol{x}}, \quad \min_{\boldsymbol{x}\in\mathbb{R}^n}\{\boldsymbol{x}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x} \mid \boldsymbol{x}^{\mathrm{T}}\boldsymbol{x}=1\}.$$

同时, 无约束优化问题 (1.1.1) 也可化成约束优化问题

$$\min_{\boldsymbol{x},t}\{t \mid f(\boldsymbol{x})+t \geqslant 0\}.$$

约束优化问题和无约束优化问题在理论分析和算法设计方面有很大不同. 由于约束优化问题的求解算法在极小化目标函数的同时要顾及解的可行性, 所以, 约束优化问题一般比无约束优化问题难解. 而约束优化问题的一种求解策略就是将其用一个无约束优化问题近似.

(2) 根据约束函数和目标函数的线性程度划分 若目标函数及约束函数都是线性的, 则称 (1.1.1) 为线性规划问题; 若目标函数与约束函数中至少有一个是非线性的, 则称 (1.1.1) 为非线性最优化问题. 特别地, 若目标函数是二次的, 约束函数是线性的, 则称 (1.1.1) 为二次规划问题. 线性规划和二次规划问题是最简单的两类最优化问题. 目前, 它们已有比较完善的理论和有效的算法.

(3) 根据目标函数和可行域的凸性划分 若目标函数为凸函数且可行域为闭凸集, 则称 (1.1.1) 为凸规划问题, 否则称之为非凸优化问题. 凸规划问题的局部最优值点都是全局最优值点.

(4) 根据函数的解析性质划分 若目标函数及约束函数都是连续可微的, 则称 (1.1.1) 为光滑优化问题; 若目标函数与约束函数中至少有一个是不可微的, 称 (1.1.1) 为非光滑优化问题. 对光滑优化问题, 可利用目标函数和约束函数的梯度信息来估计其邻域内点的函数值信息从而建立起梯度型数值方法, 而对非光滑优化问题要建立类似的求解方法, 则需要借助次梯度或光滑化等技术.

(5) 根据可行域中可行点的个数划分 若可行域中含有无穷多个不可数的点且可行域中的点连续变化, 则称 (1.1.1) 为连续优化问题; 若可行域中含有有限个或可数个点, 即该优化问题在由有限个点或可数个点组成的可行域中寻求最优解, 则称 (1.1.1) 为离散优化问题. 在很多情况下, 离散优化问题可行域中的点是通过某些元素的排列组合产生的, 因此, 又称其为组合优化问题.

对离散优化问题, 根据变量的取值, 又分离出整数规划问题, 即变量只能取整数的规划问题. 在整数规划问题中, 若变量只能取 0 和 1, 则称其为 0-1 规划问题. 在一个优化问题中, 如果部分变量为整数变量, 而其余变量为连续变量, 则称这样的优化问题为混合整数规划问题. 类似地, 有 0-1 混合规划问题.

对连续优化问题,特别是光滑的连续优化问题可以利用目标函数与约束函数的连续性质建立求解方法,而离散优化问题则不然,原因是可行域中邻近两点的目标函数值差别可能很大.对整数规划问题,若通过松弛技术将离散变量连续化,即将离散优化问题的整数变量放松为实数变量,其他约束条件不变,那么求解后者得到的最优解无论通过什么方式取整都不能保证它是原问题的最优解.这就是说,离散优化问题一般只能用离散优化问题的方法解决.尽管如此,这两类优化问题还是密切相关的,因为有些离散优化问题,如 0-1 规划,可以通过约束条件 $x(x-1)=0$ 将其化为连续优化问题,其次,连续优化问题的一些方法和技术,如对偶,可移植到离散优化问题的研究中.

(6) *根据模型参数的确定性划分* 若优化问题 (1.1.1) 的所有参数都是确定的,则称其为确定型规划问题;若优化问题 (1.1.1) 的某些参数具有某种不确定性,则称其为不确定规划问题.对不确定规划问题,若不确定参数的取值服从某种概率分布,则称其为随机规划问题.另一种常见的不确定规划问题是模糊规划.

最优化问题还有其他一些分类.从 1947 年线性规划的产生至今,人们对最优化问题的研究先后经历了从线性到非线性、从连续到离散、从确定到动态,再到随机和模糊的发展过程.本书主要讨论目标函数和约束函数均连续可微的确定型规划问题,并简单地称之为非线性最优化问题,有时也称非线性规划问题.

下面给出非线性最优化问题解的定义.

(1) 对约束优化问题 (1.1.1),可行域 Ω 中的点称为可行解或可行点.

(2) 设 $\boldsymbol{x}^*\in\Omega$. 若对任意的 $\boldsymbol{x}\in\Omega$, 有 $f(\boldsymbol{x}^*)\leqslant f(\boldsymbol{x})$, 则称 $\boldsymbol{x}^*$ 为 (1.1.1) 的全局最优解或全局最优值点,对应的目标函数值称为全局最优值或全局最小值,并记

$$\boldsymbol{x}^*=\arg\min_{\boldsymbol{x}\in\Omega} f(\boldsymbol{x}),$$

其中, arg min 取自英文 the argument of the minimum. 若 $\boldsymbol{x}^*$ 还满足对任意的 $\boldsymbol{x}\in\Omega$, $\boldsymbol{x}\neq\boldsymbol{x}^*$ 有 $f(\boldsymbol{x}^*)<f(\boldsymbol{x})$, 则称 $\boldsymbol{x}^*$ 为 (1.1.1) 的严格全局最优解.

有些优化问题在可行域上有下界,但没有最优解.这时目标函数在可行域上的下确界称为该优化问题的最优值.如二元函数 $f(\boldsymbol{x})=x_1^2+(1-x_1x_2)^2$ 在 $\mathbb{R}^2$ 上的最优值为零,但只在 $x_1=\dfrac{1}{x_2}$ 且 $x_2\to\infty$ 时才能达到.基于此,有时把 (1.1.1) 写成

$$\inf\{f(\boldsymbol{x})\mid \boldsymbol{x}\in\Omega\}.$$

(3) 设 $\boldsymbol{x}^*\in\Omega$. 若存在邻域 $N(\boldsymbol{x}^*,\delta)$, 使对任意的 $\boldsymbol{x}\in N(\boldsymbol{x}^*,\delta)\cap\Omega$, 有 $f(\boldsymbol{x}^*)\leqslant f(\boldsymbol{x})$, 则称 $\boldsymbol{x}^*$ 为 (1.1.1) 的局部最优解或局部最优值点. 若 $\boldsymbol{x}^*$ 还满足对任意的 $\boldsymbol{x}\in N(\boldsymbol{x}^*,\delta)\cap\Omega$, $\boldsymbol{x}\neq\boldsymbol{x}^*$, 有 $f(\boldsymbol{x}^*)<f(\boldsymbol{x})$, 则称 $\boldsymbol{x}^*$ 是 (1.1.1) 的严格局部最优解.

非线性最优化问题的研究核心是最优解的存在性及其结构性质、求解算法及性能分析.对一般的非线性最优化问题,求解和验证其全局最优解是一件非常棘手

甚至是不可能的事情. 因此, 人们寄希望于求得问题的局部最优解. 即便如此, 由于计算误差等因素, 几乎所有的数值算法只能给出近似解.

1.2 方法概述

如同一元二次方程的求根公式, 对非线性最优化问题, 一个直接的想法是借助微分学、变分法等数学工具通过逻辑推理和分析运算给出最优解的解析式, 这就是所谓的解析法. 该方法得到的解称为解析解. 解析解精确、简洁、直观, 适于问题的理论分析. 但它仅适用于特殊形式的优化问题, 而且有时不实用. 如对下述二次规划问题

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x}-\boldsymbol{b}^{\mathrm{T}}\boldsymbol{x},$$

当矩阵 $\boldsymbol{A}$ 对称正定时, 其最优解为 $\boldsymbol{x}=\boldsymbol{A}^{-1}\boldsymbol{b}$. 该解析解在实际应用时不但计算量大而且稳定性差. 所以在实际中, 人们选择 Gauss 消元法或三角分解法求解.

非线性最优化问题的第二类求解方法是图解法和实验法. 这类"手工作坊"式的方法操作简单、通俗易懂, 但效率较低, 且仅适用于变量个数很少的情况. 尽管如此, 我国运筹学的奠基人华罗庚在 20 世纪 60 年代提出的"优选法"在我国的工农业生产中发挥了巨大作用.

非线性最优化问题的第三类求解方法是形式转化法. 该方法主要是利用非线性最优化问题的结构性质或最优性条件将其转化成有别于原问题的另一类数学问题, 然后对后者套用现有的方法求解. 不过, 形式转化法只是提供了解决问题的一种途径, 它并非完全有效, 因为转化一般是需要条件的, 而且转化后的问题也并不总存在有效算法.

非线性最优化问题的第四类求解方法是智能算法. 它是人们受自然界规律的启迪, 根据其原理来模拟自然界的某些自然现象而建立的一种随机搜索算法. 该算法终止时可得到问题的一个近似解. 智能算法在计算过程中主要利用目标函数的函数值信息, 其有效性可以借助马尔可夫链的遍历理论和随机过程的知识来给它以数学上的描述, 并在概率意义下得到问题的全局最优解. 它适用于组合优化问题和规模较小的连续优化问题. 目前应用比较广泛的主要有遗传算法、模拟退火算法、蚁群算法和神经网络方法.

非线性最优化问题的第五类求解方法是数值迭代法. 该方法主要利用问题的函数值信息或梯度信息由当前迭代点产生一个更好的迭代点, 直到不能改进为止. 该数值方法得到的解称为数值解, 它是近似解. 根据迭代过程中函数信息的利用程度, 数值迭代法分为模式搜索法和梯度法. 模式搜索法主要根据函数值的变化规律探测目标函数的下降方向并沿该方向寻求更优的点. 该方法简单、直观, 且无需计

算目标函数的梯度, 适用于变量较少、约束简单、目标函数结构比较复杂且梯度不易计算的非线性最优化问题. 常见的主要有坐标轮换法、Hooke-Jeeves 法、Powell 共轭方向法和单纯形调优法等.

与模式搜索法不同, 梯度法在迭代过程中不但需要函数值信息, 而且还需要函数的梯度信息. 因此, 与模式搜索法相比, 梯度法对目标函数和约束函数的解析性质要求较高, 一般有快的收敛速度, 而且更容易建立其理论性质.

梯度法一般通过两种策略产生新的迭代点: 线搜索方法和信赖域方法. 线搜索方法是最常见也是研究最多的一类方法. 在算法的每一迭代步, 首先基于目标函数的梯度信息产生一个搜索方向, 然后沿该方向寻求一个更靠近最优值点的迭代点, 使目标函数值有某种程度的下降. 当前迭代点与新迭代点之间的“距离”称为步长. 由于这种过程执行一次之后并不能得到目标函数的最优解, 所以要重复执行, 直到满足某种条件为止. 具体地, 对无约束优化问题, 线搜索方法的基本框架如下.

算法 1.2.1

步 1. 取初始点 $\boldsymbol{x}_0 \in \mathbb{R}^n$ 及有关参数, 令 $k=0$.

步 2. 验证停机准则.

步 3. 求 $\boldsymbol{x}_k$ 点的搜索方向 $\boldsymbol{d}_k \in \mathbb{R}^n$.

步 4. 计算迭代步长 $\alpha_k > 0$ 使满足 $f(\boldsymbol{x}_k + \alpha_k \boldsymbol{d}_k) < f(\boldsymbol{x}_k)$.

步 5. 产生下一迭代点, 即令 $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k \boldsymbol{d}_k$, $k = k+1$, 转步 2.

下面对该算法框架的有关事项做一说明.

初始点 初始点的选取不但会影响算法的效率, 而且对多峰函数的数值结果也有影响. 显然, 初始点距离最优值点越近算法越有效. 习惯上, 取零点、分量全为 1 的点或随机点为初始点, 而一个理想的初始点取法是通过挖掘问题的结构性质取靠近最优值点的初始点.

算法参数 算法参数的取值会严重影响算法的计算效率. 借助理论分析可得参数合理的取值范围, 而通过大量的数值实验可得其经验值. 如果根据算法进程和迭代状况参数能够自动调整, 无疑会有好的数值效果.

终止条件 一般情况下, 无论设置多么苛刻的条件, 数值算法都很难在有限步内得到问题的精确解. 理论上, 在迭代点充分靠近最优值点时终止算法. 由于最优值点一般是未知的, 故在实际操作中, 一般选择在算法停滞不前时终止计算. 据此, 常见的停机准则主要有最优性条件准则、点距准则和函数下降量准则. 具体来讲, 就是当优化问题一旦近似满足某最优性条件, 或算法产生的点列进展非常缓慢 (相邻两迭代点之间的距离很小), 或目标函数值下降非常缓慢 (相邻两迭代点的目标函数值相差很小) 时算法终止.

搜索方向 搜索方向的选取原则是要保证从当前迭代点沿该方向移动时目标

函数值有所下降. 也就是说, 搜索方向是下降方向.

定义 1.2.1　称 $\boldsymbol{d}\in\mathbb{R}^n$ 为函数 $f(\boldsymbol{x})$ 在 $\boldsymbol{x}\in\mathbb{R}^n$ 点的下降方向, 若存在 $\delta>0$, 使对任意的 $t\in(0,\delta]$,

$$f(\boldsymbol{x}+t\boldsymbol{d})<f(\boldsymbol{x}).$$

对连续可微函数 $f:\mathbb{R}^n\to\mathbb{R}$, 借助梯度可判断一个方向是否为下降方向. 具体地, 设 $\boldsymbol{x}\in\mathbb{R}^n$, 若 $\boldsymbol{d}\in\mathbb{R}^n$ 满足 $\boldsymbol{d}^{\mathrm{T}}\nabla f(\boldsymbol{x})<0$, 则对充分小的 $\alpha>0$,

$$f(\boldsymbol{x}+\alpha\boldsymbol{d})=f(\boldsymbol{x})+\alpha\nabla f(\boldsymbol{x})^{\mathrm{T}}\boldsymbol{d}+o(\alpha)<f(\boldsymbol{x}).$$

因此, $\boldsymbol{d}$ 是目标函数 $f(\boldsymbol{x})$ 在 $\boldsymbol{x}$ 点的下降方向. 特别地, 当搜索方向取负梯度方向时, 该搜索方向为目标函数在该点函数值下降最快的方向, 因此称为最速下降方向. 显然, 一个方向的下降性大小与长度无关.

迭代步长　搜索方向确定后, 需要通过线搜索, 也就是计算函数 $f(\boldsymbol{x}_k+\alpha\boldsymbol{d}_k)$ 关于 $\alpha>0$ 的 (近似) 最小值解求得迭代步长.

一般地, 该算法产生的点列对应的目标函数值数列是单调下降的, 因此线搜索方法又称下降算法.

线搜索方法的核心是搜索方向的选取和迭代步长的计算. 但就它们对算法的影响力而言, 搜索方向大于迭代步长. 也就是说, 对线搜索过程, 方向比速度重要.

与线搜索方法不同, 信赖域方法利用目标函数 $f(\boldsymbol{x})$ 在 $\boldsymbol{x}_k$ 点的信息构造二次模型 $m_k(\boldsymbol{d})$ 使其在 $\boldsymbol{x}_k$ 点附近与 $f(\boldsymbol{x})$ 有好的近似, 然后根据该二次模型的最小值点来产生新的迭代点, 并视二次模型与目标函数的近似度来调整信赖域半径的大小.

具体地, 先求二次模型 $m_k(\boldsymbol{d})$ 在信赖域内的最小值点 $\boldsymbol{d}_k$, 即求解子问题

$$\min\left\{m_k(\boldsymbol{d})\mid\boldsymbol{d}\in\mathbb{R}^n,\ \|\boldsymbol{d}\|\leqslant\Delta_k\right\},$$

其中, $\Delta_k>0$ 为信赖域半径. 如果试探点 $\hat{\boldsymbol{x}}_{k+1}=\boldsymbol{x}_k+\boldsymbol{d}_k$ 能使目标函数值有“充分”的下降, 就取 $\boldsymbol{x}_{k+1}=\hat{\boldsymbol{x}}_{k+1}$. 如果近似效果特好, 在下一步就扩大信赖域半径; 否则, 就压缩信赖域半径.

一般地, 二次模型 $m_k(\boldsymbol{d})$ 取如下形式

$$m_k(\boldsymbol{d})=f(\boldsymbol{x}_k)+\boldsymbol{d}^{\mathrm{T}}\nabla f(\boldsymbol{x}_k)+\frac{1}{2}\boldsymbol{d}^{\mathrm{T}}\boldsymbol{B}_k\boldsymbol{d},$$

其中, $\boldsymbol{B}_k$ 取为 $\nabla^2 f(\boldsymbol{x}_k)$ 或其近似.

无约束优化问题的信赖域方法最早由 Powell(1970) 提出, 而后得到广泛研究. 后来, Davidon(1980) 在上述二次模型的基础上提出了信赖域方法的锥模型. 信赖域方法不如线搜索那样成熟, 应用也没有线搜索那样广泛. 但由于其强的收敛性和

可靠性, 信赖域方法的研究越来越受到重视. 从本质上讲, 它和线搜索方法的区别在于线搜索方法是借助搜索方向将一个多元函数的极值问题转化为一单元函数的极值问题, 而信赖域方法是在一值得 "信赖" 的区域内将复杂的目标函数用一个简单的二次函数近似.

由于梯度型数值方法在迭代过程中过多地依赖约束函数和目标函数的函数值信息和梯度信息, 而这些信息只能反映函数值的局部变化情况, 因而, 对非凸优化问题, 梯度型方法一般只能得到局部最优解. 而该最优解的好坏, 也就是该最优解对应的最优值与全局最优值的近似度完全依赖于初始点的选取. 因此, 若希望求得问题的全局最优解, 则需用多个初始点分别进行计算, 然后在求得的多个最优解中取其最优者当作全局最优解. 另外, 也可采用隧道 (Levy et al.,1985) 和填充 (Ge, 1990) 等技术由局部最优解向全局最优解逐步靠近. 但相对于局部优化数值算法, 全局优化算法还不成熟, 因为人们至今还没有找到一个令人满意的全局最优解的有效算法和检验准则. 正因如此, 在以后的叙述中, 除非特别说明, 我们对全局最优解和局部最优解不再严格区分, 而泛泛地称之最优解.

对非线性最优化问题, 一个数值方法要被认可, 既要有理论保障, 又要有满意的数值效果. 具体地, 一个好的数值方法应对如下指标有好的特性.

(1) *全局收敛与局部收敛* 对梯度型数值方法, 由于很难保证在有限步内得到问题的最优解, 因此, 人们希望算法产生的迭代点列有越来越靠近最优解的趋势, 这便引出了算法收敛性的概念.

如果从任意的初始点出发, 算法产生的迭代点列都收敛到问题的最优值点, 称该算法具有全局收敛性. 若算法只有在初始点和最优值点具有某种程度的靠近时才能保证迭代点列收敛到最优值点, 则称该算法具有局部收敛性. 若迭代点列的某一聚点为优化问题的最优值点, 则称该算法弱收敛.

需要强调的是, 无论是全局收敛还是局部收敛, 这都属于理论分析, 因为在进行实际数值计算时, 算法必须在有限步内终止, 而我们也只能在计算机运行机时的许可范围内得到满足一定精度要求的近似最优解.

(2) *收敛速度与二次终止性* 大量数值实验表明: 一个算法的计算效率在很大程度上依赖于在最优值点附近迭代点靠近最优值点的速度. 也就是说, 一个数值方法高效的基本标志就是一旦迭代点进入目标函数的一个 "狭长的凹谷", 那么以后产生的迭代点应迅速移向该 "凹谷" 的最低点. 下面从收敛速度的角度对此进行分析.

算法的收敛速度主要考虑迭代点列 $\{\boldsymbol{x}_k\}$ 与最优值点 $\boldsymbol{x}^*$ 之间的距离所确定的数列 $\{\|\boldsymbol{x}_k-\boldsymbol{x}^*\|\}$ 趋于零的速度. 所以, 讨论算法收敛速度的前提是算法产生的点列收敛到问题的最优值点. 显然, 数列 $\{\|\boldsymbol{x}_k-\boldsymbol{x}^*\|\}$ 趋于零的速度越快, 相应算法的效率就越高. 对此, 有以下两种衡量尺度: Q- 收敛和 R- 收敛 (Ortega, Rheinboldt, 1970).

Q- 收敛是通过前后两迭代点靠近最优值点的程度进行比较定义的: 设点列 $\{\boldsymbol{x}_k\}$ 收敛到点 $\boldsymbol{x}^*$, 且存在 $q \geqslant 0$ 满足

$$\limsup_{k\to\infty} \frac{\|\boldsymbol{x}_{k+1}-\boldsymbol{x}^*\|}{\|\boldsymbol{x}_k-\boldsymbol{x}^*\|} \leqslant q.$$

若 $0<q<1$, 则称点列 $\{\boldsymbol{x}_k\}$Q- 线性收敛到 $\boldsymbol{x}^*$. 若 $q=0$, 则称点列 $\{\boldsymbol{x}_k\}$ Q- 超线性收敛到 $\boldsymbol{x}^*$.

容易证明: 如果点列 $\{\boldsymbol{x}_k\}$ Q- 超线性收敛到 $\boldsymbol{x}^*$, 则

$$\lim_{k\to\infty} \frac{\|\boldsymbol{x}_{k+1}-\boldsymbol{x}_k\|}{\|\boldsymbol{x}_k-\boldsymbol{x}^*\|} = 1. \tag{1.2.1}$$

对收敛到 $\boldsymbol{x}^*$ 的点列 $\{\boldsymbol{x}_k\}$, 若存在 $0 \leqslant p < \infty$ 和 $r \geqslant 1$ 使

$$\limsup_{k\to\infty} \frac{\|\boldsymbol{x}_{k+1}-\boldsymbol{x}^*\|}{\|\boldsymbol{x}_k-\boldsymbol{x}^*\|^r} \leqslant p,$$

则称点列 $\{\boldsymbol{x}_k\}$ Q-r 阶收敛到 $\boldsymbol{x}^*$, 有时简单地称点列 $\{\boldsymbol{x}_k\}$ r- 阶收敛到 $\boldsymbol{x}^*$. 其中最常见的是 2- 阶收敛. 若 $r>1$, 则 r- 阶收敛必超线性收敛.

与 Q- 收敛不同, R- 收敛是借助一个收敛于零的数列来度量 $\{\|\boldsymbol{x}_k-\boldsymbol{x}^*\|\}$ 趋于零的速度: 设点列 $\{\boldsymbol{x}_k\}$ 收敛到最优值点 $\boldsymbol{x}^*$. 若存在 $\kappa>0$, $q\in(0,1)$ 使

$$\|\boldsymbol{x}_k-\boldsymbol{x}^*\| \leqslant \kappa q^k,$$

则称 $\{\boldsymbol{x}_k\}$ R- 线性收敛到 $\boldsymbol{x}^*$.

对上述点列, 若存在 $\kappa>0$ 和收敛于零的正数列 $\{q_k\}$ 使

$$\|\boldsymbol{x}_k-\boldsymbol{x}^*\| \leqslant \kappa \prod_{i=0}^{k} q_i,$$

则称点列 $\{\boldsymbol{x}_k\}$ R- 超线性收敛到 $\boldsymbol{x}^*$.

两类收敛速度中的 Q 和 R 分别取自英文 “Quotient” 和 “Root”. 在这些收敛速度中, 超线性收敛比线性收敛速度快. 而若一个点列 Q-(超) 线性收敛, 则它必 R-(超) 线性收敛.

收敛速度用来表征迭代点列靠近问题最优解的快慢程度. 一般地, 超线性收敛性算法是比较快的. 但由于计算误差和算法程序本身带来的影响, 算法收敛速度的理论结果并不能保证算法在进行数值计算时具有同样的效果.

另外, 二次终止性也是判断算法优劣的一个重要指标. 它是指对任意的严格凸二次函数, 从任意的初始点出发, 算法经过有限步迭代后均可到达最优值点.