非线性规划理论与算法

FEIXIANXING GUIHUA LILUN YU SUANFA

王宜举 修乃华

陕西科学技术出版社

非线性规划理论与算法

（第二版）

王宜举　修乃华

陕西科学技术出版社

内 容 提 要

本书系统地介绍了非线性规划问题的有关最优化理论与算法，主要包括一些传统理论与经典算法，如无约束优化问题和带约束优化问题的最优性条件，无约束优化问题的下降算法、共轭梯度算法、拟牛顿算法，约束优化问题的可行算法、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、GLP 投影算法.

本书思路清晰，内容丰富，重点突出，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。本书的最大特点是书中内容跳跃度小，系统性强，适合作为运筹学专业的研究生和数学专业高年级本科生从事非线性规划研究的入门教材或参考书，也可作为应用部门工程技术人员的工具书。

图书在版编目(CIP)数据

非线性规划理论与算法/王宜举，修乃华主编. —2 版. 西安：陕西科学技术出版社，2008.2

ISBN 978-7-5369-3825-0

Ⅰ.非... Ⅱ.①王... ②修... Ⅲ.非线性规划—研究生—教材 Ⅳ.0221.2

中国版本图书馆 CIP 数据核字(2007)第 193348 号

出版者 陕西科学技术出版社
西安北大街 131 号 邮编 710003
电话(029)87211894 传真(029)87218236
http://www.snstp.com
发行者 陕西科学技术出版社
电话(029)87212206 87260001
印 刷 陕西丰源印务有限公司
规 格 787mm×1092mm 1/16 开本
印 张 14
字 数 310 千字
印 数 1~1000 册
版 次 2004 年 5 月第 1 版 2004 年 5 月第 1 次印刷
2008 年 2 月第 2 版 2008 年 2 月第 1 次印刷
定 价 22.00 元

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。在20世纪60年代,最优化方法发展成为一门新兴的运筹学基础学科。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,并在此基础上,运筹学逐渐发展出了新的更细的研究分支。

非线性规划作为运筹学的一个重要研究分支在近20年得到了快速发展,新的理论和算法不断出现。为及时吸收新近发展成熟并得到广泛应用的成果,并应用于我们的教学科研中,我们参阅国内外有关最优化理论与方法的许多专著、教材和研究文献,并结合近几年的教学实践编写了这本书。在此,我们对所引用文献的作者表示衷心的谢意。

非线性规划问题的最优化理论与算法内容十分丰富,几乎其每一个研究分支都有相应的专著。由于篇幅所限,本书主要对非线性规划问题的传统理论与经典梯度型数值算法及其新的研究进展做了比较详尽的论述。一方面使有志于从事该领域研究工作的数学工作者打下一个良好的基础,同时也使将非线性规划作为应用工具的工程技术人员能够领略这方面的基本内容。本书力图做到内容丰富,结构合理,使用方便。为此,对书中涉及的一些结论,我们尽量给出比较详细的证明过程,以降低所列内容的跳跃度,使其能够引导对运筹学感兴趣的数学工作者比较轻松地进入非线性规划研究领域,并到达研究前沿。在内容编排上,我们采用流水账式的书写方式,以尽可能减少阅读过程中的回头率。对于课时安排,由于书中的部分内容可留作自学,因此,本书做为教材建议在72学时内讲授完。

该书在2004年首次出版。这次再版,我们除了修正原版中的一些错误,同时还尽可能多地增添对算法的一些评述,使读者能更好地体会一些概念和算法引入的意图及其实现的策略,增强可读性。在此,我们对一些同行提出的宝贵建议表示衷心的感谢。本书第一作者特别感谢重庆师范大学的杨新民教授在2007年秋给本人提供为该校运筹学专业的研究生集中讲授这门课程的机会,使本书内容得以丰富和改进。

我们真诚地期待您的批评和建议,来信请发至:

wyiju@ hotmail. com 或 nhxiu@ bjtu. edu. cn

最后,感谢山东省“十一五”重点学科“运筹学与控制论”和国家自然科学基金(10771120,10671010)经费资助。

王宜举　修乃华

2008年1月

目　录

第 1 章 引言

本章主要介绍在非线性规划问题的最优化理论与算法中涉及到的一些基本概念和基础知识. 最后，我们讨论了无约束优化问题的最优性条件.

1.1 问题的提出

在现实生活中，我们会遇到这样一类实际问题，要求在众多的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，怎样分配有限资源使得分配方案既满足各方面的基本要求又能获得好的经济效益；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；农田规划中，如何安排农作物的布局，才能使农田高产稳产，发挥地区优势；军事作战时，如何确定作战方案，才能以最小的代价夺取最大的胜利. 这类寄希望于“低投入，高产出”的问题在数学上称为最优化问题.

为建立最优化问题的数学模型，我们给出一个具体的例子.

例 1.1.1 (曲线拟合问题) 设有某个物理现象可以通过如下函数描述

$$R(t) = x_1 + x_2 e^{-x_3 t}.$$

用实验测得 $t = t_i$ 时的 R 的值为 $R_i, i = 1, 2, \cdots, m$. 现在要确定参数 x_1, x_2, x_3, 使得这些实验点与理论曲线 $R(t)$ 拟合得最好，即在最小二乘意义下求参数 x_1, x_2, x_3. 这里，我们还要求参数满足 $x_1 + x_2 = 1, x_3 \geqslant 0$.

对上述问题，我们可以建立如下的最优化模型：

$$\begin{aligned} \min f(x) &= \sum_{i=1}^{m} \left[R_i - \left(x_1 + x_2 e^{-x_3 t_i}\right)\right]^2 \\ \text{s.t. } & x_1 + x_2 = 1 \\ & x_3 \geqslant 0 \end{aligned}$$

这里 s.t. 是 subject to 的缩写. 它是指在极小化目标函数时，变量必须满足的约束条件.

基于此，我们可以给出最优化问题的数学模型

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

其中，函数 $f: R^n \to R$ 称为目标函数， $\Omega \subseteq R^n$ 称为可行域. 可行域是在对目标函数求极值过程中对所涉及变量的取值范围的界定，它对应于实际问题的可决策集合.

可行域有多种表述形式，一般用等式和不等式来定义，如

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

对 $i \in \mathcal{E}$, $c_i(x) = 0$ 称为等式约束， $\mathcal{E}$ 称为等式约束指标集；对 $i \in \mathcal{I}$, $c_i(x) \geqslant 0$ 称为不等式约束， $\mathcal{I}$ 称为不等式约束指标集. 简单地讲，最优化问题就是在某个界定区域内或整个欧氏空间中求一个数值函数的极值问题.

1.2 最优化问题分类

最优化问题形形色色，具体的数学模型多种多样. 我们可以从不同的角度对最优化问题进行分类.

1. *根据有无约束划分* 若 $\Omega = R^n$, 即 x 是自由变量，则称 (1.1.1) 为无约束最优化问题；若 $\Omega \subsetneqq R^n$, 称 (1.1.1) 为约束最优化问题.

2. *根据约束函数和目标函数的线性与否划分* 若目标函数及约束函数都是线性的，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函数是非线性的，那么称 (1.1.1) 为非线性规划问题. 进一步，若目标函数是二次函数，约束是线性的，称 (1.1.1) 为二次规划问题；若目标函数是凸函数，等式约束是线性的，不等式约束是凹函数 (此时，可行域为闭凸集)，称 (1.1.1) 为凸规划问题. 有时将目标函数是凸函数，可行域为非空闭凸集的约束优化问题也称为凸规划问题. 凸规划问题的最大特点是它的局部最优值点都是其全局最优值点.

3. *根据函数的可微性质划分* 若目标函数及约束函数都是连续可微的，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函数是不可微的，那么称 (1.1.1) 为非光滑优化问题.

4. *根据可行域内含有可行点的个数划分* 若可行域内的点是有限的，即约束优化问题 (1.1.1) 是在由有限个点组成的可行域中寻求最优解，称 (1.1.1) 为离散优化问题，又称组合优化问题；若可行域内含有无穷多个不可数的点，而且可行域内的点可以连续变化，则称 (1.1.1) 为连续优化问题.

对于约束优化问题 (1.1.1)，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. 实际上，它属于离散优化问题的范畴. 如果部分变量要求取整数，而其余变量在可行域中可以连续变化，那么就称其为混合整数规划问题. 在整数规划问题中，若变量仅取 0 和 1，这样的规划问题称为 0–1 规划问题.

连续优化问题，特别是光滑的连续优化问题在求解过程中可以充分利用目标函数与约束函数的连续性质，而离散优化问题则不然，因为此时在可行域中邻近的两个点对应的目标函数值差别可能很大. 因此，一般来讲，连续优化问题远比离散优化问题求解容易. 若将整数规划问题连续化，即在求解过程中，将变量取整的约束条件放松为取实数，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规划问题的最优解. 也就是说，离散优化问题一般只能用离散优化问题的办法解决.

5. *根据涉及变量的确定性划分* 若问题中含有随机变量，这样的规划问题称为随机规划问题；相应地，不含随机变量的规划问题称为确定型规划问题.

另外还有其他一些分类. 本书主要讨论目标函数和约束函数均连续可微的确定型非线性最优化问题，并简单地称之为非线性规划问题.

1.3 非线性规划问题解的基本概念

对非线性规划问题，我们给出解的定义.

1. Ω 中的点称为最优化问题 (1.1.1) 的可行解或可行点.

2. 设 $x^* \in \Omega$. 若对任意 $x \in \Omega$, 有 $f(x^*) \leqslant f(x)$, 则称 x^* 为 (1.1.1) 的全局最优解或全局最小值点，其对应的目标函数值称为全局最优值或全局最小值. 此时，我们记 $x^* = \arg\min\limits_{x\in\Omega} f(x)$. 若 x^* 还满足对任意 $x \in \Omega$, $x \neq x^*$ 有 $f(x^*) < f(x)$, 则称 x^* 为 (1.1.1) 的严格全局最优解.

3. 设 $x^* \in \Omega$. 若存在 x^* 点的邻域 $N(x^*, \delta), \delta > 0$, 使得对任意 $x \in N(x^*, \delta) \cap \Omega$, 有 $f(x^*) \leqslant f(x)$, 则称 x^* 为 (1.1.1) 的局部最优解. 若 x^* 还满足对任意 $x \in N(x^*, \delta) \cap \Omega$, $x \neq x^*$, 有 $f(x^*) < f(x)$, 则称 $x^* \in \Omega$ 是 (1.1.1) 的严格局部最优解.

由于求目标函数的最大值可以通过给目标函数前面加负号而转化为求目标函数的最小值，因此，除非特别说明，最优化问题的 (全局或局部) 最优值解均是指其 (全局或局部) 最小值解. 需要说明的是，即使是任意阶连续可微的函数，其最小值和最小值解的存在性并不总是一致的. 也就是说，存在任意阶连续可微的函数有最小值但不一定有最小值解. 如二元多项式函数 $x_1^2+(1-x_1x_2)^2$ 的最小值为 0, 但只能在 $x_2 \to \infty$, 并且 $x_1 = \dfrac{1}{x_2} \to 0$ 时达到. 因此，人们有时把非线性规划问题的模型写成下面的形式：

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

对于一般的非线性规划问题，除凸规划问题外，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，我们寄希望于求得问题的局部最优解. 对于局部最优解，我们常借助目标函数和约束函数在最优值点的梯度信息和 Hessian 阵信息来刻画最优值点所满足的性质，进而建立起非线性规划问题的一阶与二阶最优性条件. 在介绍这些最优性条件之前，我们给出非线性规划问题 (1.1.1) 的一个具体的表述形式

$$\begin{aligned} \min\ & f(x) \\ \text{s.t.}\ & c_i(x) = 0, \quad i \in \mathcal{E} \\ & c_i(x) \geqslant 0, \quad i \in \mathcal{I} \end{aligned} \tag{1.3.1}$$

其中， $f: R^n \to R$, $c_i: R^n \to R$, $i \in \mathcal{E} \cup \mathcal{I}$, 均为连续可微函数.

下面是非线性规划问题的一些常见的一阶最优性条件.

1. 稳定点条件 $x^* \in \Omega$ 称为 (1.3.1) 或 (1.1.1) 的稳定点，若成立

$$\langle \nabla f(x^*), x - x^* \rangle \geqslant 0, \quad \forall\, x \in \Omega.$$

这里，$\langle\ \cdot\ \rangle$ 表示欧氏空间中两个向量的内积.

显然，若 $\Omega = R^n$, 则稳定点条件变为 $\nabla f(x^*) = 0$.

对于无约束优化问题，稳定点可能是 (全局或局部) 极大值点也可能是 (全局或局部) 极小值点，也可能二者都不是. 最后一种情况对应的稳定点称为目标函数的鞍点，即对于目标函数，鞍点在从该点出发的一个方向上是极大值点，在另一个方向上是极小值点. 单元函数 $f(x)=x^3$ 的稳定点就是鞍点，而最典型的是马鞍面上的鞍点.

2. FJ 条件　设 x^* 是约束优化问题 (1.3.1) 的 (全局或局部) 最优解，则当该规划问题在 x^* 点满足一定条件时，存在 $\lambda_0^*\in R$ 和 $\lambda^*\in R^{|\mathcal{E}\cup\mathcal{I}|}$ 满足

$$\begin{cases}\lambda_0^*\nabla f(x^*)=\sum\limits_{i\in\mathcal{E}\cup\mathcal{I}}\lambda_i^*\nabla c_i(x^*)\\ c_i(x^*)=0, \qquad i\in\mathcal{E}\\ c_i(x^*)\geqslant 0, \quad \lambda_i^*\geqslant 0, \quad \lambda_i^*c_i(x^*)=0, \qquad i\in\mathcal{I}\\ \lambda_0^*\geqslant 0, \quad (\lambda_0^*,\lambda^*)\neq 0\end{cases}$$

上述系统称为 FJ 条件. 满足 FJ 条件的点称为原规划问题的 FJ 点. 该条件最早由 Fritz John (1948) 提出. 对 FJ 条件, 若 $\lambda_0^*=0$, 该最优性条件对于原问题没有意义; 若 $\lambda_0^*\neq 0$, FJ 条件就是下面的 KKT 条件.

3. KKT 条件　设 x^* 是约束优化问题 (1.3.1) 的 (全局或局部) 最优解，则当约束函数在 x^* 点满足一定条件时，存在非零向量 λ^* 满足

$$\begin{cases}\nabla f(x^*)=\sum\limits_{i\in\mathcal{E}\cup\mathcal{I}}\lambda_i^*\nabla c_i(x^*)\\ c_i(x^*)=0, \qquad i\in\mathcal{E}\\ c_i(x^*)\geqslant 0, \quad \lambda_i^*\geqslant 0, \quad \lambda_i^*c_i(x^*)=0, \qquad i\in\mathcal{I}\end{cases}$$

上述系统称为 KKT 条件，有时也称为 K-T 条件. 满足 KKT 条件的点 x^* 称为约束优化问题的 K-T 点， λ^* 称为约束优化问题在 x^* 点的最优 Lagrange 乘子. 该条件由 Kuhn 和 Tucker (1951) 提出后被广泛接受. 后来人们发现 Karush 早在 1939 年就提出了类似的条件，这就是为什么有的文献中称之为 KKT 条件，而有的文献称之为 K-T 条件.

虽然这些一阶最优性条件与最优解之间有密切的关系，但我们并不奢望通过计算满足最优性条件的点来得到原问题的解. 人们引入最优性条件的目的主要有两个：一是将其应用在算法设计 (如停机准则的设置) 和算法分析中；另外就是将其用于最优化问题的灵敏度分析，即分析问题中某些数据的变化对问题的最优解带来的影响.

1.4　非线性规划问题的算法分类与算法框架

对于一般的非线性规划问题，一个直接的想法是借助微分学和变分法、Lagrange 乘子法等数学工具通过逻辑推导和分析计算直接得到问题的解或解的表达式，这就是所谓的解析方法. 这种方法仅对一些具有特殊结构的非线性规划问题有效. 非线性规划问题的第二类方法是图解法和实验法. 它们虽然操作简单，但仅能处理低维的情况. 鉴于前述方

法的局限性，对于一些实际问题，人们尝试利用目标函数和约束函数在某局部区域上的函数值信息或导数信息，构建迭代型数值解法，即从当前的一个近似解点，逐步调优来产生新的更好的近似解点，直到不能再改进为止.

对于工程技术中的非线性规划问题，它们的求解方法以迭代形式的数值解法最为典型和常见. 这种类型的算法大多与计算机结合，它们不仅能够计算比较复杂的非线性规划问题，而且求解速度快、效率高. 根据迭代过程中搜索策略的确定性与否，这类方法分为两类：随机搜索型方法和确定型方法.

随机搜索型方法是人们受自然界规律的启迪，根据其原理模仿求解问题的方法. 这类方法采用随机搜索技术从概率意义上寻求问题的解，其搜索过程主要利用函数值信息而不依赖于函数的梯度信息. 我们可以借助马尔柯夫链的遍历理论等概率论和随机过程的知识来给它以数学上的描述，并能在概率意义下找到问题的全局最优解，从而保证算法的收敛性. 严格意义上讲，这类算法属于智能算法的范畴，是启发式算法. 它更适用于求解组合优化问题. 如果用来求解非线性连续优化问题，计算速度会比较慢. 目前应用比较广泛的随机搜索型方法主要有遗传算法、模拟退火算法和蚁群算法.

确定型方法根据利用函数信息的程度分为直接搜索型方法和梯度型方法. 直接搜索型方法在迭代过程中主要利用函数值信息，它主要适用于变量较少、目标函数结构比较复杂且梯度不易计算的情形. 常见的主要有坐标轮换法、模式搜索法和单纯形调优法等. 梯度型算法在迭代过程中除利用函数值信息外还需要函数在当前点或已产生点的梯度信息和 Hessian 阵信息. 因此，与直接搜索型方法相比，梯度型算法一般具有快的收敛速度，而且更容易建立算法的理论性质.

本书主要介绍连续可微的非线性规划问题的梯度型数值解法. 无约束优化问题的这类算法大都基于目标函数的线性或二阶近似而构筑起来的. 需要说明的是，后来陆续发展起来的这类算法不再保证算法产生的迭代点列对应的目标函数值数列的单调性，同时对于约束优化问题，包括初始点在内，算法产生的迭代点未必都在可行域内.

在具体的梯度型迭代算法中，人们一般采用两种策略来由当前迭代点产生下一迭代点：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. 它的基本思想是：在每次迭代中，利用当前迭代点 x_k 或已产生迭代点的信息，先产生一个搜索方向 d_k，然后在 x_k 点沿 d_k 方向寻求一个“好”的点，以使目标函数值有某种满意程度的下降. 由于这种过程执行一次之后并不能得到目标函数的最小值点，所以要重复执行，直到满足某种条件为止. 当前迭代点与新点之间的“距离”称为步长. 线搜索方法的核心就是搜索方向的建立和步长的选取.

对于无约束优化问题，线搜索方法的大体框架如下.

算法 1.4.1

步一 取初始点 x_0，令 $k=0$.

步二 验证停机准则.

步三 求 x_k 点的搜索方向 d_k.

步四 求迭代步长 α_k, 使得 $f(x_k+\alpha_k d_k)<f(x_k)$.

步五 产生下一迭代点 $x_{k+1}=x_k+\alpha_k d_k$, $k=k+1$, 转步二.

对于初始点，虽然没有特殊的要求，但如果能根据某些信息选取比较靠近问题最优解的点做为初始点，无疑会有好的数值效果.

对于停机准则而言，常用的有最优性条件准则，点距准则和函数下降量准则三种，也就是所谓的目标函数的梯度范数充分小，相邻两迭代点之间的距离充分小，相邻迭代点对应的目标函数值的 (相对) 下降量充分小. 这里，充分小的参考依据是计算精度.

对于上述无约束优化问题的求解算法，搜索方向 d_k 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. 由于一个数值函数在一点的梯度指向该点函数值增长最快的方向，因此，算法中搜索方向的选择与目标函数在当前迭代点的梯度密切相关. 下面是下降方向的定义.

定义 1.4.1 若向量 $d_k\in R^n$ 满足 $d_k^\top\nabla f(x_k)<0$, 那么 d_k 就称为 $f(x)$ 在 x_k 点的下降方向.

设 d_k 为 $f(x)$ 在 x_k 点的下降方向，则对充分小的正数 α 有

$$f(x_k+\alpha d_k)=f(x_k)+\alpha\nabla f(x_k)^\top d_k+o(\alpha)<f(x_k).$$

所以沿下降方向搜索，我们总可以找到一个点 x_{k+1} 使得 $f(x_{k+1})<f(x_k)$ 成立.

对于无约束优化问题的线搜索方法，不同的搜索方向构成不同的下降算法. 显然，负梯度方向是最简单的下降方向，实际上，它是利用目标函数的线性近似得到的. 该方向称为最速下降方向，对应的算法称为最速下降算法. 该方法在迭代过程中计算量和存储量都很小. 但由于迭代点列的行程容易形成 “Z” 字形，所以收敛速度很慢. 如果利用目标函数的二阶近似构造搜索方向，就得到牛顿方向，对应的算法称为牛顿算法. 该方法收敛速度快，缺点是迭代过程中需要计算目标函数的 Hessian 阵，不适于进行大规模计算. 后来人们建立起来的共轭梯度算法和拟牛顿算法基本克服了上述缺点，成为目前被广泛接受的两类算法.

搜索方向确定以后，步长一般通过对下降方向进行精确或非精确线搜索 (即所谓的一维线搜索) 得到. 如果算法产生的迭代点列对应的目标函数值数列是单调不增的，则称这类算法为下降算法. 对于约束优化问题而言，搜索方向确定以后，步长的选取就不能单纯通过目标函数值的变化来决定，因为有时需要将可行域的因素考虑进去.

与线搜索方法不同，信赖域方法是利用目标函数 $f(x)$ 在 x_k 点的信息构造一个二次模型 $m_k(d)$ 使其在 x_k 点附近 (通过信赖域大小来控制) 与 $f(x)$ 有好的近似，然后将该二次模型的最小值点做为算法的下一迭代点，并视其近似程度来调整信赖域半径的大小.

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

$$\min\ \{m_k(d) \mid d \in R^n,\ \|d\| \leqslant \Delta\} \tag{1.4.1}$$

其中， $\Delta > 0$ 称为信赖域半径. 如果试探点 $\hat{x}_{k+1} = x_k + d_k$ 能使目标函数值有“充分”的下降，那么就取 $x_{k+1} = \hat{x}_{k+1}$, 如果近似效果特好，在下一步还可以扩大信赖域半径；否则，就压缩信赖域半径，重新求解 (1.4.1).

一般地，二次模型 $m_k(d)$ 取如下形式

$$m_k(d) = f(x_k) + d^{\mathrm{T}}\nabla f(x_k) + \frac{1}{2}d^{\mathrm{T}}B_k d,$$

其中， B_k 取为 $\nabla^2 f(x_k)$ 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，Davidon(1980) 还在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，人们对它的研究会越来越多. 它和线搜索方法的区别在于后者先确定搜索方向，然后确定两个迭代点之间的距离；而信赖域方法先确定两个迭代点之间的最大距离，然后确定搜索方向和迭代步长.

最后，我们对梯度型数值解法做一小结. 由于该类方法在迭代过程中过多地依赖于约束函数和目标函数在已产生点的函数值和梯度信息，而这些信息只能反映函数值的局部变化情况，因而梯度型算法大多只能得到问题的局部最优解. 若需要求问题的全局最优解，则需要用多个初始点进行运算，然后在所得到的局部最优解中取其最优者当作全局最优解. 还有就是借助隧道法和填充函数法等由局部最优解向全局最优解一步步靠近. 正因如此，在本书以后的叙述中，除非特别说明，我们对非线性规划问题的全局最优值解和局部最优值解不再严格区分，而泛泛地称之为非线性规划问题的最优值解.

1.5 算法的收敛性分析与收敛速度

对于一般的非线性规划问题，一个求解方法要被认可，既要有理论做保障，又要有满意的数值效果. 具体地，一个好的求解算法应在如下指标具有好的特性.

1. *全局收敛与局部收敛* 对于算法许可范围内的一类问题，选择合理的初始点后，一个好的算法不但要在计算机运行机时许可的范围内得到满足一定精度要求的问题的最优值解，而且在理论上也要保证算法产生的迭代点有越来越靠近问题最优值解或满足某最优性条件的点的趋势. 这就是说，我们需要建立算法的收敛性分析，以给算法的有效性提供理论依据.

如果从任意的初始点出发，算法产生的迭代点列收敛到满足某最优性条件的点，称该算法具有全局收敛性. 若算法只有在初始点和问题的最优值点具有某种程度的接近时才能保证迭代点列的收敛性，则称该算法具有局部收敛性.

在算法分析中，我们还经常碰到如下的收敛定义，即算法产生的迭代点列的某一聚点满足某最优性条件，我们称这种算法具有弱收敛性.

2. 收敛速度与二次终止性　从理论上讲，非线性规划问题的一个数值算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”中的最低点，也就是说算法应具有快的局部收敛速度.

收敛速度主要考虑迭代点列 $\{x_k\}$ 与最优值点 x^* 的差范数所决定的序列 $\{\|x_k - x^*\|\}$ 趋于零的速度，所以讨论收敛速度的前提是迭代点列 $\{x_k\}$ 收敛到 x^*. 显然，序列 $\{\|x_k - x^*\|\}$ 趋于零的速度越快，相应的算法的效率就越高. 对此，有两种常见的衡量尺度： Q- 收敛和 R- 收敛 (Ortega & Rheinboldt, 1970).

设点列 $\{x_k\}$ 收敛到 x^*, 且存在 $q \geqslant 0$ 满足

$$\limsup_{k\to\infty} \frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|} \leqslant q.$$

若 $0 < q < 1$, 称 $\{x_k\}$ Q- 线性收敛到 x^*. 若 $q = 0$, 称 $\{x_k\}$ Q- 超线性收敛到 x^*. 容易证明：如果序列 $\{x_k\}$ Q- 超线性收敛到 x^*, 则

$$\lim_{k\to\infty} \frac{\|x_{k+1} - x_k\|}{\|x_k - x^*\|} = 1. \tag{1.5.1}$$

Q- 超线性收敛是介乎于线性收敛和下面介绍的二阶收敛之间的收敛，常常被看成接近二阶收敛的非常快的收敛.

对上述收敛点列，若存在 $0 \leqslant p < \infty$ 和 $r \geqslant 1$ 使得

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

则称 $\{x_k\}$ Q-r 阶收敛到 x^*, 有时简单地称 $\{x_k\}$ r- 阶收敛到 x^*. 其中最常见的是 2- 阶收敛. 若 $r > 1$, r- 阶收敛必为超线性收敛，但反之不一定成立.

下面定义的 R- 收敛是借助当前迭代点的误差界估计给出的.

设点列 $\{x_k\}$ 收敛到最优值点 x^*. 若存在 $\kappa \in (0, \infty)$, $q \in (0,1)$ 使得

$$\|x_k - x^*\| \leqslant \kappa q^k,$$

我们称 $\{x_k\}$ R- 线性收敛到 x^*.

对上述序列，若存在 $\kappa \in (0, \infty)$ 和收敛于零的正数列 $\{q_k\}$ 使得

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

我们称 $\{x_k\}$ R- 超线性收敛到 x^*.

这里， Q 和 R 分别取自英文单词 "Quotient" 和 "Root" 的第一个字母. 在这些收敛速度中，超线性收敛比线性收敛速度快. 另外，若一个序列 Q-(超) 线性收敛，则它必 R-(超) 线性收敛.

收敛速度主要用来描述迭代点列比较靠近问题最优值解时的趋向特征. 一般地，具有超线性收敛性或二阶收敛速度的算法是比较快的. 但由于计算误差和计算机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该算法在实际计算过程中有同样性质的数值效果.

另外，算法的二次终止性也是判断算法优劣的一个重要指标. 它指的是算法对于任意的严格凸二次函数，从任意的初始点出发，都能经过有限步达到其最小值点.

由于严格凸二次函数是非线性函数中形式最简单条件最强的函数，所以一个好的算法理应在有限步内得到其最优点. 其次，对于一般的目标函数，它在最优值点附近与一个严格凸二次函数拟合得非常好. 因此可以猜想，对于严格凸二次函数数值效果好的算法，对于一般的目标函数也应具有好的数值效果. 无约束优化问题的共轭梯度算法和拟牛顿算法之所以这么有吸引力，在一定程度上与它们的二次终止性有密切关系.

3. *稳定性* 算法的稳定性，实际上是算法数值效果的可靠性. 对于一个算法，如果输入数据有扰动 (即舍入误差)，而在计算过程中，舍入误差不增长，则称该算法是稳定的. 对舍入误差呈恶性增长的算法一般是不推荐使用的.

一般的数值稳定性是对算法而言，但有时也与数学问题本身有关. 对所谓的病态问题，如果输入数据有微小扰动，就会引起输出数据的极大扰动. 这种情况是由数学问题本身的性质决定的，与算法无关.

最简单的例子是线性方程组的求解. 如果系数矩阵条件数很大，那么计算过程中，数据存储引起的误差可能会引起计算结果的大的偏差. 这就是说，对于病态问题，用任何算法直接计算都会产生不稳定性. 实际计算时可以借助调比策略对问题做一转换来克服.

4. *计算复杂性和存储消耗* 我们设计的算法迟终都要在计算机上实现，有快的收敛速度是保证算法高效的一个因素，但算法每一迭代步中的计算量和存储量也是必须考虑的因素. 因为若一个算法在每一迭代步的计算量或存储量比较大，那么对于大规模非线性规划问题，算法每一迭代步的运行机时会比较长，从而影响算法的整体效率.

上述指标主要侧重算法的理论分析，这些理论分析往往能使我们对算法有更深入的理解：一方面，它使我们明白算法为什么适用于某一类问题而不适合于另一类问题，为什么要 "这样" 取初始点，为什么要 "那样" 选取参数. 另一方面，它帮助我们发现算法中的缺陷，进而改进之. 只是人们在借助数学分析等工具对一个算法进行理论分析的时候，一般要对问题本身或其解点做些假设，而这些假设往往是难于验证的.

对于非线性规划问题的求解算法，进行数值试验是非常重要也是非常必要的. 首先，算法本身就是为问题求解设计的；其次，数值试验虽不能给算法的理论分析提供什么保证，但却常常会很可靠地显露出某些可能的理论结果. 只是我们在进行数值分析的时候，需要考虑到算法中参数和初始点的选取对数值效果的影响，因为算法程序对算法而言具有一

定的灵活性，而计算程序中的任何一个微小细节的改动都会对数值结果产生大的影响.

需要指出的是，算法的理论分析和数值分析所得到的结果并不总是一致的. 就象求解线性规划问题的单纯形方法和椭球算法，对于非线性规划问题也存在这样的算法. 里面的原因很复杂，既有计算过程中数据舍入误差和参数取值的影响，也有理论分析过程中所加条件在实际问题中得不到满足的因素.

经过半个多世纪的努力，优化工作者已经设计了求解非线性规划问题的多种算法. 随着新算法的不断出现，一些效率低下的算法渐渐淡出人们的视线，而一些性能好的算法被进一步改进和完善，并被推广应用. 对同一个算法，其性能指标与具体的问题有直接关系，对上面列出的性能指标没有一个统一的量化标准. 至今，人们还没有找到一个算法对上述指标比别的算法都优秀，也没有找到一个令人满意的寻求和检验一般非线性规划问题的全局最优解的有效算法. 所以人们一般只能宣称某种方法对某一类问题比较有效，这也是在非线性规划问题的算法研究中多种算法并存的主要原因.

目前，人们对数值优化算法的改进主要在两个方面：计算效率的提高和特殊问题的求解. 从某种意义上来讲，随着计算机技术的不断发展和计算机性能的提高，后者无疑更具吸引力，除非算法在计算效率方面有质的提高. 特别地，如果一个算法能够“治愈”别的算法都无能为力的“疑难杂症”，即便是计算效率稍低一些那也是相当诱人的.

1.6 凸函数与几个常用不等式

对于一般的非线性规划问题，无论是无约束的还是带约束的，我们很难保证其局部最优值点是全局最优值点，主要原因在于目标函数是“多峰”函数. 为此，我们需要给出连续可微的“单峰”函数，也就是凸函数的定义和一些基本性质.

称函数 $f: R^n \to R$ 是凸函数，若满足

$$f(\lambda x+(1-\lambda)y) \leqslant \lambda f(x)+(1-\lambda)f(y), \quad \forall\, x,y\in R^n,\ \lambda\in[0,1].$$

若凸函数 $f(x)$ 连续可微，则上述条件等价于下述条件之一：

$$f(y)-f(x) \geqslant \nabla f(x)^\top(y-x), \quad \forall\, x,y\in R^n;$$

$$(\nabla f(y)-\nabla f(x))^\top(y-x) \geqslant 0, \quad \forall\, x,y\in R^n.$$

进一步，若凸函数 f 二阶连续可微，则上述条件等价于

$$h^\top\nabla^2 f(x)h \geqslant 0, \quad \forall\, x,h\in R^n.$$

若上述不等式取严格不等号，则称函数 $f(x)$ 为严格凸的. 将上述条件进一步加强就得到一致凸函数的定义.

定义 1.6.1 称函数 $f: R^n \to R$ 是一致凸函数，若存在 $\eta > 0$, 使得对任意 $x, y \in R^n$ 和 $\lambda \in [0,1]$, 成立

$$f(\lambda x + (1-\lambda)y) \leqslant \lambda f(x) + (1-\lambda)f(y) - \frac{1}{2}\lambda(1-\lambda)\eta\|x-y\|^2. \tag{1.6.1}$$

若一致凸函数 f 连续可微，则 (1.6.1) 等价于

$$(y-x)^\top(\nabla f(y) - \nabla f(x)) \geqslant \eta\|y-x\|^2, \quad \forall\, x,\ y \in R^n. \tag{1.6.2}$$

同样，若一致凸函数 f 二阶连续可微，则上述条件等价于

$$h^\top \nabla^2 f(x) h \geqslant \eta\|h\|^2, \quad \forall\, x,\ h \in R^n. \tag{1.6.3}$$

下面，我们给出一致凸函数的上述条件的等价性证明.

(1.6.1)⇒(1.6.2): 由 (1.6.1), 对任意 $x, y \in R^n$ 和 $\lambda \in (0,1)$,

$$\frac{f(\lambda x + (1-\lambda)y) - f(x)}{1-\lambda} + \frac{f(\lambda x + (1-\lambda)y) - f(y)}{\lambda} + \frac{1}{2}\eta\|x-y\|^2 \leqslant 0.$$

也就是

$$\frac{f\big(x + (1-\lambda)(y-x)\big) - f(x)}{1-\lambda} + \frac{f\big(y + \lambda(x-y)\big) - f(y)}{\lambda} + \frac{1}{2}\eta\|x-y\|^2 \leqslant 0.$$

分别令 $\lambda \uparrow 1$ 和 $\lambda \downarrow 0$ 得

$$\begin{aligned} -\nabla f(x)(x-y) + f(x) - f(y) + \frac{1}{2}\eta\|x-y\|^2 \leqslant 0, \\ \nabla f(y)(x-y) + f(y) - f(x) + \frac{1}{2}\eta\|x-y\|^2 \leqslant 0. \end{aligned}$$

两式相加得 (1.6.2).

(1.6.2)⇒(1.6.3): 对 (1.6.2), 利用微积分学中的有关结论，对任意 $x, y \in R^n$,

$$\begin{aligned} \eta\|y-x\|^2 &\leqslant (y-x)^\top(\nabla f(y) - \nabla f(x)) \\ &= (y-x)^\top \textstyle\int_0^1 \nabla^2 f(x + t(y-x))(y-x)\mathrm{d}t. \end{aligned}$$

不等式两边除以 $\|y-x\|^2$, 令 $y \to x$, 并设 $\dfrac{y-x}{\|y-x\|} \to h$. 利用 x, y 的任意性得到 (1.6.3).

(1.6.3) ⇒ (1.6.1): 由 Taylor 展开式，对任意 $x, y \in R^n$ 和 $\lambda \in [0,1]$, 存在

$$\xi_1 \in (x, \lambda x + (1-\lambda)y), \quad \xi_2 \in (\lambda x + (1-\lambda)y), y)$$