工程优化设计系统

龙 连 春

主要内容

1、　线性规划问题求解程序；

2、　二次规划问题求解程序；

3、　对偶二次规划问题求解程序；

4、　通用规划问题求解程序。

工程问题与通用程序

1、　满足工程要求；
2、　考虑多种形式和可能性；
3、 用户使用尽可能简单方便；
4、　方法的可靠性比速度更重要；
5、 充分的考核。

数学规划引论

应用数学 → 运筹学（Operations Research）

→ 数学规划 （**Mathematical Programming**）

数学规划求解的优化问题通常可以表达为下述形式：

$$\begin{cases} \text{求} \quad \boldsymbol{x} = \{x_1, \cdots, x_n\}^T \\ \text{使} \quad f(\boldsymbol{x}) \to \min \\ \text{s.t.} \quad g_j(\boldsymbol{x}) \le 0 \quad (j = 1, \cdots, m) \end{cases}$$

数学规划主要包括：线性规划、非线性规划、对偶规划、几何规划、整数规划、动态规划及多目标规划等。

最常用且比较成熟、也是研究得最多的是线性规划、二次规划和对偶规划（多约束）。

结构优化的三个基础：计算机技术、结构分析、数学规划。

工程优化要求如下特性：

★ 可靠性(在各种情况下都能收敛)；

★ 通用性；

★ 有效性(较少的迭代次数，较少的计算量)；

★ 健壮性(无论初始点在那里，均收敛到最优点)；

★ 准确性(收敛到精确最优点的能力)；

★ 方便性。

线性规划程序

线性规划标准形式：

$$\begin{cases} \min \quad CX \quad X=(x_1\ x_2\ \cdots\ x_n)^T \\ s.t. \quad AX=B \\ \qquad\quad X\geq 0 \end{cases}$$

程序可将其它形式转化为标准形式

1、对求极大值问题，可以将目标函数乘以(-1)，转化为求极小问题。

2、若第i个约束条件为“<=”，则在该约束左边加上一个松弛变量x_{n+i}，（x_{n+i}>=0）

$$\sum_{j=1}^{n} a_{ij}x_j \leq b_i \longrightarrow \sum_{j=1}^{n} a_{ij}x_j + x_{n+i} = b_i$$

3、若某一变量i有上下界约束，则令：$y_i = x_i - \underline{x}_i$

$$\underline{x}_i \le x_i \le \overline{x}_i \longrightarrow \begin{cases} y_i \le \overline{x}_i - \underline{x}_i \\ y_i \ge 0 \end{cases}$$

程序将其它形式自动转化为标准形式。

两相法求标准线性规划问题：

1、求初始基本可行解

构造辅助问题：

$$\begin{cases} \min \quad g = \sum_{i=1}^{m} x_{n+i} \\ s.t. \quad \sum_{j=1}^{n} a_{ij} x_j + x_{n+i} = b_i \qquad (i = 1,2,\ldots,m) \\ \qquad x_j \ge 0 \qquad (j = 1,2,\ldots,n+m) \end{cases}$$

辅助问题的最优解作为初始基本可行解。

2、以辅助问题最优解作为初始基本可行解，单纯形法求问题的最优解。

解的可能性：

1、有最优解；

2、无有限最优解；

3、无可行解。

构造的算例包括：

小于等于约束；

大于等于约束；

等式约束；

变量上、下边界约束；

有最优解；

无有限最优解；

无可行解。

二次规划程序

二次规划的Lemke算法主要针对下列形式的问题：

$$\begin{cases} \min \quad f = \dfrac{1}{2} X^T H X + C X \\ s.t. \quad A X \le B \\ \qquad\quad X \ge 0 \end{cases}$$

一般形式为：

$$\begin{cases} \min \quad f = \dfrac{1}{2} X^T H X + C X \\ s.t. \quad A_1 X \le B_1 \\ \qquad\quad A_2 X = B_2 \\ \qquad\quad \underline{X} \le X \le \overline{X} \end{cases}$$

求解二次规划的方法主要有：直接消去法、主动集方法、Wolfe算法、罚函数法、Lagrange乘子法及Lemke算法等。

二次规划程序

Lemke算法首先引进人工变量，然后通过基底交换的运算沿着相邻的几乎互补基本可行解移动，直到求得互补基本可行解。

计算了6个各种形式的算例。

对偶二次规划程序

对偶方法将设计变量空间的寻优过程转化在对偶变量空间里寻优，它要求目标函数和约束函数是变量可分离的形式。

当原问题为凸规划时，原问题与对偶问题的对偶间隙为零，而可分离性使设计变量和对偶变量成为显式关系，凸性还保证局部最优解也是全局最优解。

由于对偶变量少，且约束只是变量非负限制，不仅问题简单，而且规模比原问题小，而且易于求解。

对偶二次规划程序

一般二次问题转化为对偶问题：

$$\begin{cases} \min \quad f = \frac{1}{2} X^T H X + C X \\ s.t. \quad A_1 X \le B_1 \\ \qquad A_2 X = B_2 \\ \qquad \underline{X} \le X \le \overline{X} \end{cases}$$

标准化处理

$$\begin{cases} \min \quad f = \frac{1}{2} X^T H X + C X \\ s.t. \quad A X \le B \\ \qquad X \ge 0 \end{cases}$$

对偶规划

$$\begin{cases} \max \quad \frac{1}{2} \lambda^T D \lambda + \lambda^T E - \frac{1}{2} C^T H^{-1} C \\ \qquad \lambda \ge 0 \end{cases}$$

通过求出对偶问题的最优解可求得原问题的最优解。另外，若对偶问题无可行解，则原问题目标无界；若对偶问题目标无界，则原问题无可行解。

通用规划程序

一些工程问题的目标函数或约束函数很难表示成某个表达式，或者表达式相对复杂而不宜用一般优化方法来求解，对于这些问题，可以用线性逼近方法来求解。

由于线性问题的求解方法非常成熟稳健，根据某些估计方法或工程经验获得一个初始估计X_0，在X_0点将目标函数和约束函数作泰勒展开，得到一个线性规划问题。

求解上述规划问题得到一个解，然后该解作为初始解，继续迭代，求解一系列线性规划问题，直到收敛为止。

通用规划程序

对一般规划问题，用下列线性规划模型进行序列化近似：

$$\begin{cases} \min \quad C^T X \\ s.t. \quad AX = (\le \text{或} \ge) B \\ \qquad\quad \underline{X} \le X \le \overline{X} \end{cases} \longrightarrow \begin{cases} \min \quad C^T X_1 \\ s.t. \quad A_1 X_1 = B_1 \\ \qquad\quad X_1 \ge 0 \end{cases}$$

首先采用线性规划程序求解，若有可行解，则继续迭代，直到收敛。

若在本步搜索范围内无可行解，则作如下处理：

通用规划程序

等式约束，增加松弛变量，转化为2个不等式：

$$\sum_{j=1}^{n} a_{ij} x_j = b_i \longrightarrow \begin{cases} \sum_{j=1}^{n} a_{ij} x_j + \varepsilon_i \geq b_i \\ \sum_{j=1}^{n} a_{ij} x_j - \varepsilon_i \leq bi \end{cases}$$

小于等于不等式约束，减去一个正的变量，转化为1个等式：

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i \longrightarrow \sum_{j=1}^{n} a_{ij} x_j - \varepsilon_i = b_i$$

大于等于不等式约束，加上一个正的变量，转化为1个等式：

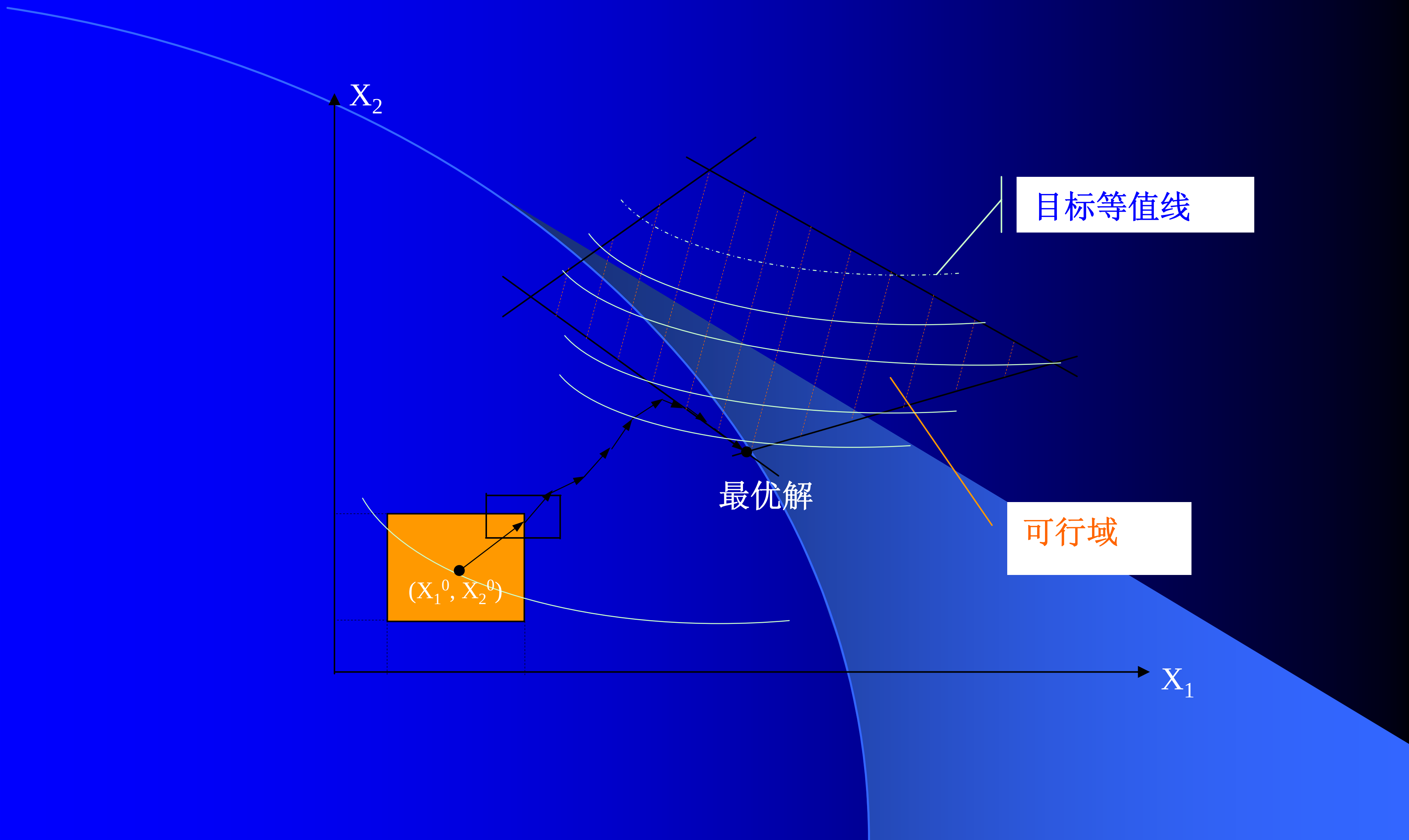
$$\sum_{j=1}^{n} a_{ij} x_j \geq b_i \longrightarrow \sum_{j=1}^{n} a_{ij} x_j + \varepsilon_i = b_i$$

形成新的规划：

$$
\begin{cases}
\mathbf{min} \quad \varepsilon_1+\varepsilon_2+\cdots+\varepsilon_n \\
s.t. \quad A_1X_1+\{\boldsymbol{\varepsilon}\}_{n1}=B_1 \\
\qquad\; A_2X_2+\{\boldsymbol{\varepsilon}\}_{n2}\le(\ge)B_2 \\
\qquad\; X\ge 0,\quad \boldsymbol{\varepsilon}\ge 0
\end{cases}
$$

求解上述模型，使问题在不可行域内找到一个违背约束最小的解，使迭代继续进行。将求解结果作为下一循环的初始值。

收敛过程：



通用规划程序

通用规划程序特点：

1、可求解任意目标形式；

2、初始解可以是不可行的，中间解也可以是不可行的，最终总能回到可行域内；

3、目标和约束的导数可以是解析的，也可以是数值的；

4、多重收敛判断：目标函数和设计变量的变化率（值）及总迭代次数；

5、每步运动极限的大小随迭代步的变化而变化，且与总迭代步相关；

6、对于有些问题，收敛速度可能会比较慢。

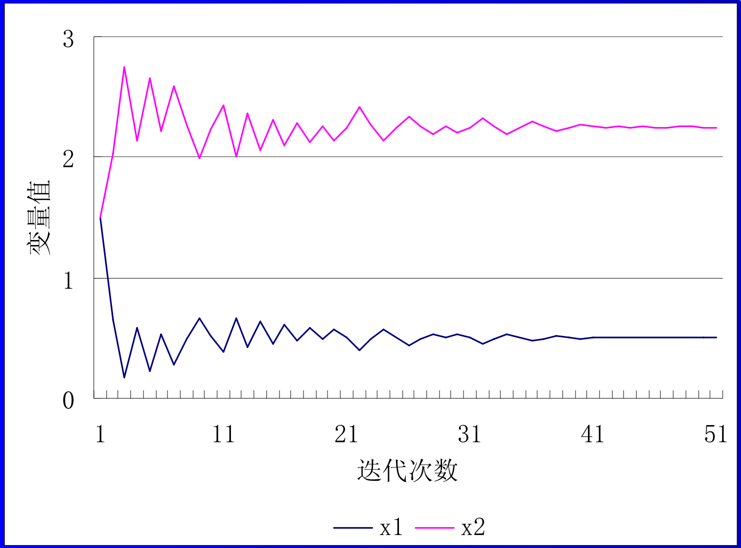
算例1

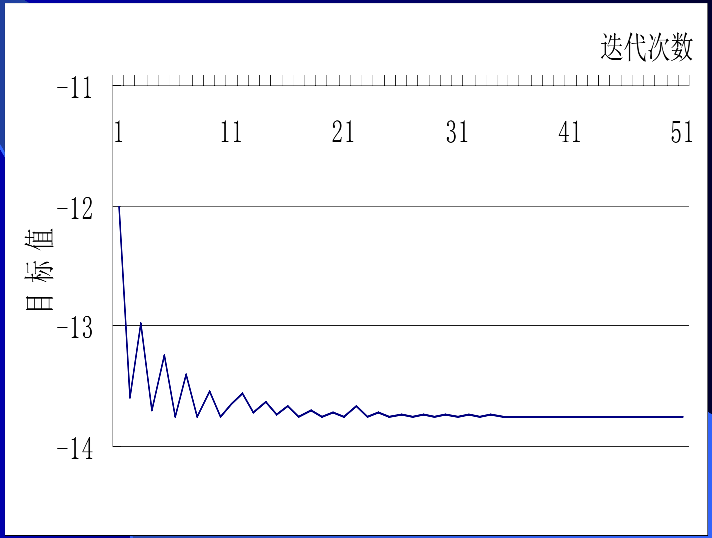
$$\min \quad f = x_1^2 - x_1 x_2 + 2x_2^2 - x_1 - 10x_2$$

$$\text{s.t.} \quad -3x_1 - 2x_2 \ge -6$$

$$x_1 \ge 0, \quad x_2 \ge 0$$

初始值：(1.5,1.5)





通用规划程序

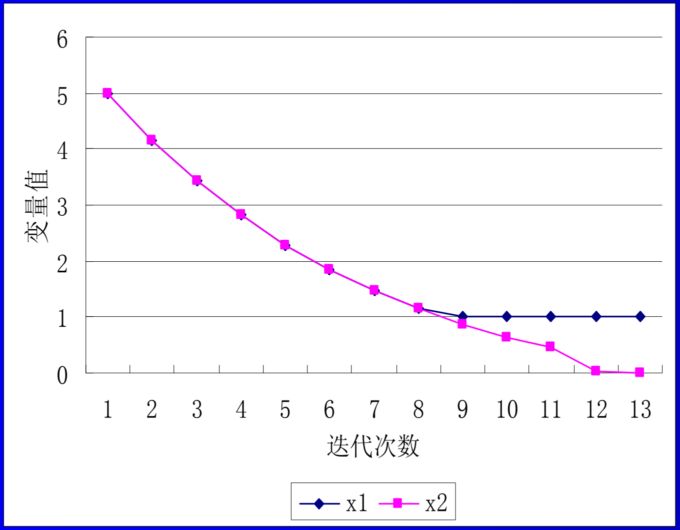
算例2

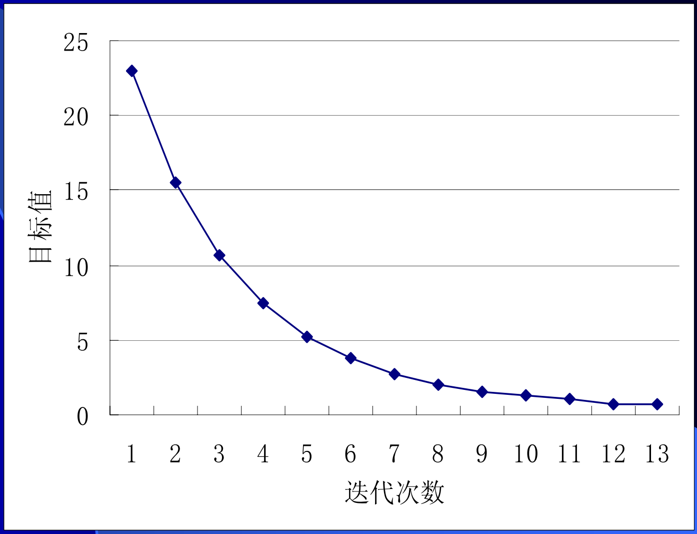
$$\min \quad f = (x_1 + 1)^3/12 + x_2$$

$$\text{s.t.} \quad x_1 - 1 \geq 0$$

$$x_2 \geq 0$$

初始值：(5, 5)





算例3

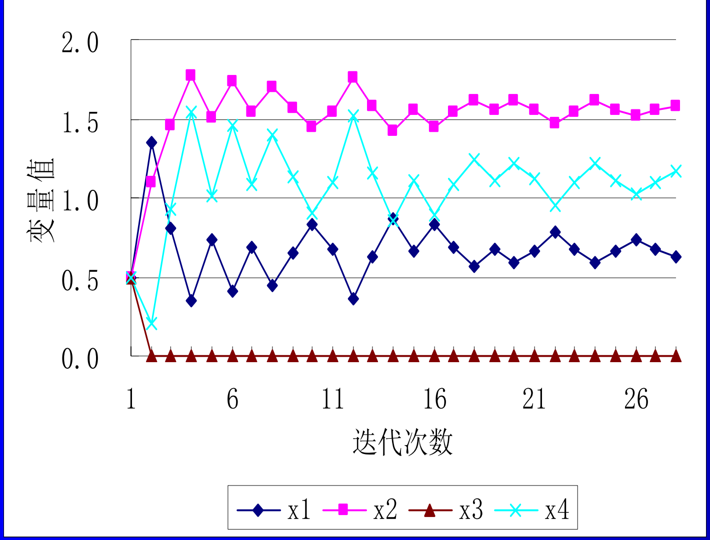
$$\min \quad f = x_1^2 - 2x_1 - x_2$$

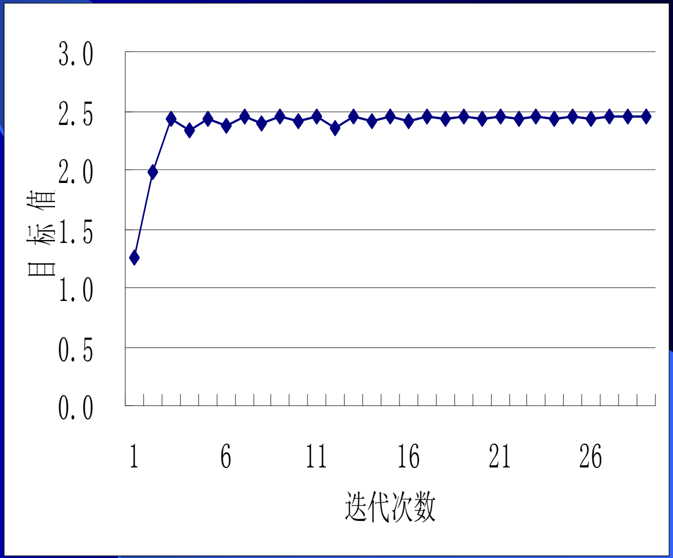
$$\text{s.t.} \quad 2x_1 + 3x_2 + x_3 = 6$$

$$2x_1 + x_2 + x_4 = 4$$

$$x_1 \ge 0, \quad x_2 \ge 0, \quad x_3 \ge 0, \quad x_4 \ge 0$$

初始值：(0.5, 0.5, 0.5, 0.5)





总 结

1、开始的准备工作越详细周到越好；

2、可能的话，手算算例与程序调试同步进行，对检查数据错误非常有用；

3、程序编制的同时附加详细的说明；

4、几个对进度影响比较大的地方。