约束非线性问题：

$$\begin{cases} \min f(x) \\ \text{s.t. } g_i(x) \le 0, \quad i = 1, \cdots, l \\ \qquad h_j(x) = 0, \quad j = 1, \cdots, m \end{cases}$$

解法分类：

1、将约束问题转化为一个或一系列无约束问题

基本思想：根据约束条件改造原目标函数，构造同时表达原目标函数和约束条件的无约束新目标函数，然后求解。

对不等式约束和等式约束都适用。

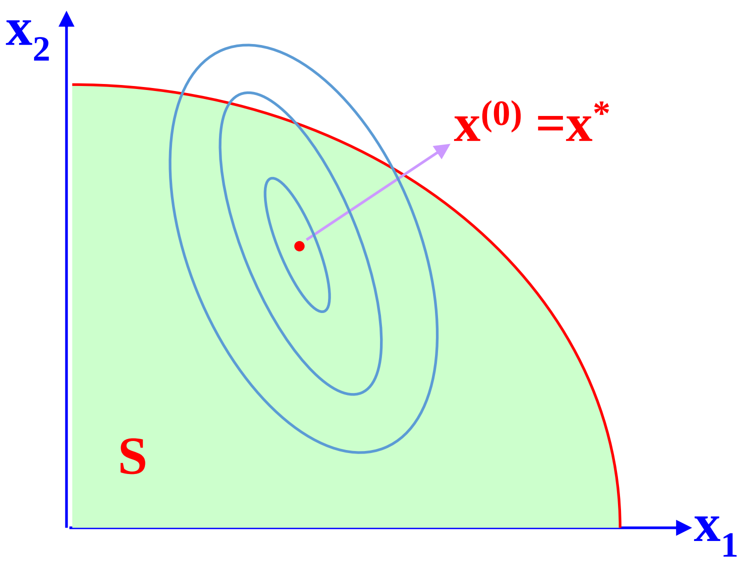
2、直接求解约束问题

基本思想：直接在可行域中寻找约束最优解。

此类方法只适用于不等式约束问题。

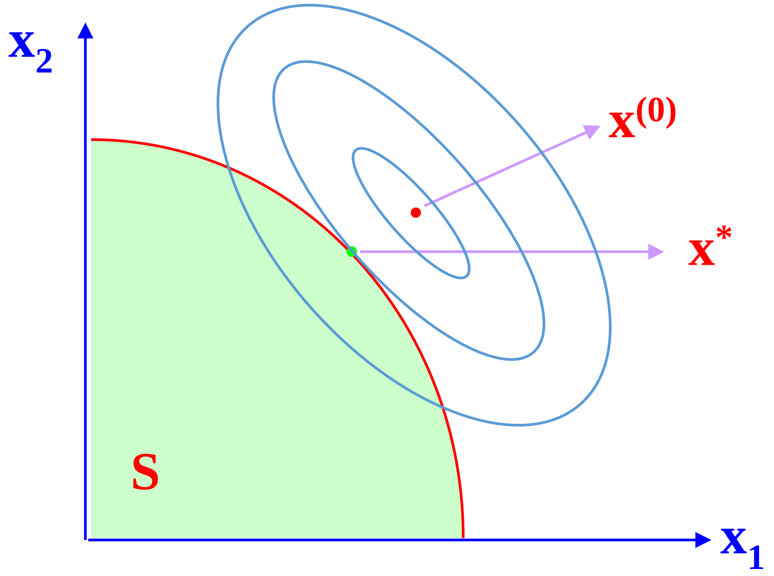
对约束问题，可先作一些尝试或转换处理，以便于求解

1、先不考虑约束条件，求目标函数的无约束最优点



(1) 无约束极值点$x^{(0)} \in S$

即为所求



(2) 无约束极值点$x^{(0)} \notin S$

需使用约束问题解法进一步求解

2、用消元法消去某些比较简单的等式约束

设问题中存在m个等式约束

$$
\begin{cases}
h_1(x_1, x_2, \ldots, x_n)=0 \\
h_2(x_1, x_2, \ldots, x_n)=0 \\
\ldots \ldots \ldots \ldots \ldots \ldots \\
h_m(x_1, x_2, \ldots, x_n)=0
\end{cases}
$$

假定可从这m个方程中解出m个变量来，例如后面的m个变量$x_{n-m+1}, \ldots, x_n$表达为其余的n－m个变量 的显函数：

$$
\begin{cases}
x_{n-m+1}=f_1(x_1, x_2, \ldots, x_{n-m}) \\
x_{n-m+2}=f_2(x_1, x_2, \ldots, x_{n-m}) \\
\ldots \ldots \ldots \ldots \ldots \ldots \\
x_n \quad =f_m(x_1, x_2, \ldots, x_{n-m})
\end{cases}
$$

将上述函数关系代入原目标函数及余下的各约束中，则构成的新问题为仅含 n－m个变量的约束问题。

对简单的等式约束问题，可通过此法简化为无约束问题。

3、用自变量变换法去掉某些简单的不等式约束

如 $\begin{cases} \min f(x) = \min f(x_1, x_2, \ldots, x_n) \\ \text{s.t.} \quad x_i \geq 0 \quad i=1, 2, \ldots, n \end{cases}$

若令 $x_i = y_i^2 \quad (i=1, 2, \ldots, n)$, 则原问题简化为

$$\min f'(y_1, y_2, \ldots, y_n) = \min f(y_1^2, y_2^2, \ldots, y_n^2)$$

求出 y^*, 经反变换 得 x^*

可行域不能变

变换原则:

（1）y_i 取任何实数都要使 $x_i(y_i)$ 满足原约束条件;

（2）对每个 x_i 的可行值必须存在相应的 y_i 值，使 $x_i = y_i(x_i)$ 成立。

问题：1、导致目标函数形式变得更复杂

2、有局限性

Lagrange乘子法

理论基础：约束问题的最优性必要条件，即 KKT条件

原约束问题 —— 乘子×约束＋目标函数 ——> 新的无约束问题

$$
\begin{cases}
\min f(x) \\
\text{s.t.}\ \ g_i(x) \le 0, \quad i = 1, \cdots, l \\
\qquad h_j(x) = 0, \quad j = 1, \cdots, m
\end{cases}
$$

$$
\begin{cases}
\nabla_x L(x^*, \lambda^*, \mu^*) = 0 \\
(\lambda^*) g(x^*)^T = 0 \\
\lambda^* \ge 0
\end{cases}
$$

引入Lagrange函数

$$L(x, \lambda, \mu) = f(x) + \sum_{i=1}^{l} \lambda_i g_i(x) + \sum_{j=1}^{m} \mu_j h_j(x)$$

和向量函数 $g(x) = [g_1(x), g_2(x), \ldots, g_l(x)]^T$

等式约束问题的Lagrange乘子法

$$\begin{cases} \min f(x) \\ \text{s.t. } h_j(x) = 0 \quad j=1,2,\ldots,m \end{cases}$$

K－T条件： $\nabla f(x^*) + \sum_{j=1}^{m} \mu_j^* \nabla h_j(x^*) = 0$ **n个方程**

约束条件： $h_j(x^*) = 0 \quad j=1,2,\ldots,m$ **m个方程**

$\Rightarrow$ **n+m个变量**

引入Lagrange函数 $L(x,\mu) = f(x) + \sum_{j=1}^{m} \mu_j h_j(x)$

$$\nabla L(x,\mu) = 0 \Rightarrow \begin{cases} \partial L/\partial x_i = 0 & i = 1,2,\ldots, n \\ \partial L/\partial \mu_j = 0 & j = 1,2,\ldots m \end{cases}$$

$\Rightarrow$ **Lagrange函数的平稳点**

方程组 $\begin{cases} \partial L/\partial x_i = 0 & i=1,2,\dots,\ n \\ \partial L/\partial \mu_j = 0 & j=1,2,\dots m \end{cases}$ 的求解，可转化为求解下列无约束问题

$$\min\ z(x_1, x_2, \dots, x_n;\ \mu_1, \dots,\ \mu_n) = \sum_{i=1}^{n}\left[\frac{\partial L}{\partial x_i}\right]^2 + \sum_{j=1}^{m}[h_j(x)]^2$$

该问题的极小点就是$L(x, \mu)$的平稳点，其分量$x_1^*, \dots x_n^*$ 是原约束问题的最优点。

例 用Lagrange乘子法求解问题

$$\min f(x) = 60 - 10x_1 - 4x_2 + x_1^2 + x_2^2 - x_1x_2$$

$$\text{s.t.}\ \ h(x) = x_1 + x_2 - 8 = 0$$

解：引入Lagrange函数

$$L = f + \mu h = 60 - 10x_1 - 4x_2 + x_1^2 + x_2^2 - x_1x_2 + \mu(x_1 + x_2 - 8)$$

求$L(x_1, x_2, \mu)$的平稳点，即解方程组

$$\begin{cases} \partial L/\partial x_1 = -10 + 2x_1 - x_2 + \mu = 0 \\ \partial L/\partial x_2 = -4 + 2x_2 - x_1 + \mu = 0 \\ \partial L/\partial \mu = x_1 + x_2 - 8 = 0 \end{cases} \Longrightarrow \begin{cases} x_1^* = 5 \\ x_2^* = 3 \\ \mu^* = 3 \end{cases}$$

原问题的最优点$x^* = [5, 3]^T$，最优解 $f^* = 12$

对等式约束问题：

$$\begin{cases} \min f(x) \\ \text{s.t. } h_j(x) = 0 \quad j=1,2,\ldots,m \end{cases}$$

引入Lagrange函数 $L(x,\mu) = f(x) + \sum_{j=1}^{m} \mu_j h_j(x)$

$$\nabla L(x,\mu) = 0 \Rightarrow \begin{cases} \partial L/\partial x_i = 0 & i = 1,2,\ldots, n \\ \partial L/\partial \mu_j = 0 & j = 1,2,\ldots m \end{cases} \Rightarrow$$

Lagrange函数的平稳点

$$\min z(x_1, x_2, \ldots, x_n;\ \mu_1, \ldots, \mu_n) = \sum_{i=1}^{n} \left[\frac{\partial L}{\partial x_i}\right]^2 + \sum_{j=1}^{m} [h_j(x)]^2$$

该问题的极小点就是L(x, μ)的平稳点。

Lagrange函数L只是μ_j的线性函数，在n+m维空间中，它的平稳点不会是L(x, μ)的极值点，以此不能采用无约束问题优化方法，通过求min L(x, μ)的最小点来求L(x, μ)的平稳点。

不等式约束问题的Lagrange乘子法

$$\begin{cases} \min f(x) \\ \text{s.t.} \quad g_i(x) \le 0, \quad i=1,\cdots,l \\ \qquad h_j(x)=0, \quad j=1,\cdots,m \end{cases}$$

可否直接求解其K－T条件？

$$\begin{cases} \nabla f(x^*) + \sum_{i=1}^{l} \lambda_i^* \nabla g_i(x^*) + \sum_{j=1}^{m} \mu_j^* \nabla h_j(x^*) = 0 \\ \lambda_i^* g_i(x^*) = 0 \qquad i=1,\cdots,l \\ \lambda_i^* \ge 0 \qquad i=1,\cdots,l \end{cases}$$

当然可以！

但一般作法是引入*l*个松弛变量w_i (i=1,…, *l*), 把原问题化为仅有等式约束的问题

$$\begin{cases} \min f(x) \\ \text{s.t.} \quad g_i(x) + w_i^2 = 0, \quad i=1,\cdots,l \\ \qquad h_j(x)=0, \quad j=1,\cdots,m \end{cases}$$

为了不出现新的不等式约束$w_i>0$

引入Lagrange函数 $L(x, \mu, w) = f(x) + \sum_{j=1}^{m} \mu_j h_j(x) + \sum_{k=1}^{l} \mu_{k+m}[g_k(x) + w_k^2]$

求L(x, μ,w)的平稳点, 即解方程组

$$\begin{cases} \partial L/\partial x_i = 0 & i = 1,2,\ldots,n \\ \partial L/\partial \mu_j = 0 & j = 1,2,\ldots m+l \\ \partial L/\partial w_k = 2\mu_{k+w} w_k = 0 & k = 1,2,\ldots,l \end{cases}$$

上述方程组的求解可转化为求解一个最小二乘问题，相应目标函数为

$$z = \sum_{i=1}^{n}\left[\frac{\partial L}{\partial x_i}\right]^2 + \sum_{j=1}^{m}[h_j(x)]^2 + \sum_{k=1}^{l}[g_k(x) + w_k^2]^2 + \sum_{k=1}^{l}(2\mu_{k+m} w_k)^2$$

例 用Lagrange乘子法求解问题

$$
\begin{cases}
\min f(x) = 2x_1^2 - 2x_1x_2 + 2x_2^2 - 6x_1 \\
\text{s.t. } g_1(x) = 3x_1 + 4x_2 - 6 \le 0 \\
\qquad g_2(x) = -x_1 + 4x_2 - 2 \le 0
\end{cases}
$$

解： 引入松弛变量x_3和x_4，把不等式约束变为等式约束

$$G_1(x) = 3x_1 + 4x_2 + x_3^2 - 6 = 0$$

$$G_2(x) = -x_1 + 4x_2 + x_4^2 - 2 = 0$$

构造Lagrange函数

$$L(x, \mu) = (2x_1^2 - 2x_1x_2 + 2x_2^2 - 6x_1) + \mu_1(3x_1 + 4x_2 + x_3^2 - 6) + \mu_2(-x_1 + 4x_2 + x_4^2 - 2)$$

求$L(x, \mu)$的平稳点，即解方程组

$$
\begin{cases}
\partial L/\partial x_1 = 4x_1 - 2x_2 - 6 + 3\mu_1 - \mu_2 = 0 \\
\partial L/\partial x_2 = -2x_1 + 4x_2 + 4\mu_1 + 4\mu_2 = 0 \\
\partial L/\partial x_3 = 2\mu_1 x_3 = 0 \\
\partial L/\partial x_4 = 2\mu_2 x_4 = 0 \\
\partial L/\partial \mu_1 = 3x_1 + 4x_2 + x_3^2 - 6 = 0 \\
\partial L/\partial \mu_2 = -x_1 + 4x_2 + x_4^2 - 2 = 0
\end{cases}
$$

最小二乘法解方程组

由此可见，一个约束问题的一组最优解都对应存在一组确定的Lagrange乘子，也就是说，Lagrange乘子是该约束问题的一组特征值。

Lagrange乘子法可求解所有约束问题，它实质上是一种解析的间接解法，理论上重要。

但其缺点是：维数增加很多，函数形式复杂，最后归结为用最小二乘法解一个非线性方程组，所以不能算是一种很有效的方法。

§ 惩罚函数法

理论和实践价值都大，是求解约束问题的有效方法之一。

出发点：将一个约束问题转换为一个或一系列的无约束问题进行求解 。

§ 外部惩罚函数法（外点法）

（一）基本思想

对约束问题
$$\begin{cases} \min f(x) \\ \text{s.t.} \quad g_i(x) \le 0, \quad i = 1, \cdots, l \\ \qquad h_j(x) = 0, \quad j = 1, \cdots, m \end{cases}$$

想办法构造一个新的目标函数，使得在原问题可行域S内，它的值与原问题目标函数值相同，在可行域S之外，它给出一个正无穷大的“惩罚”值，即

$$F(x) = \begin{cases} f(x), & x \in S \\ +\infty, & x \notin S \end{cases}$$

求解 F(x)的无约束极小值，极小点必定在S中，从而得到原问题极小点。

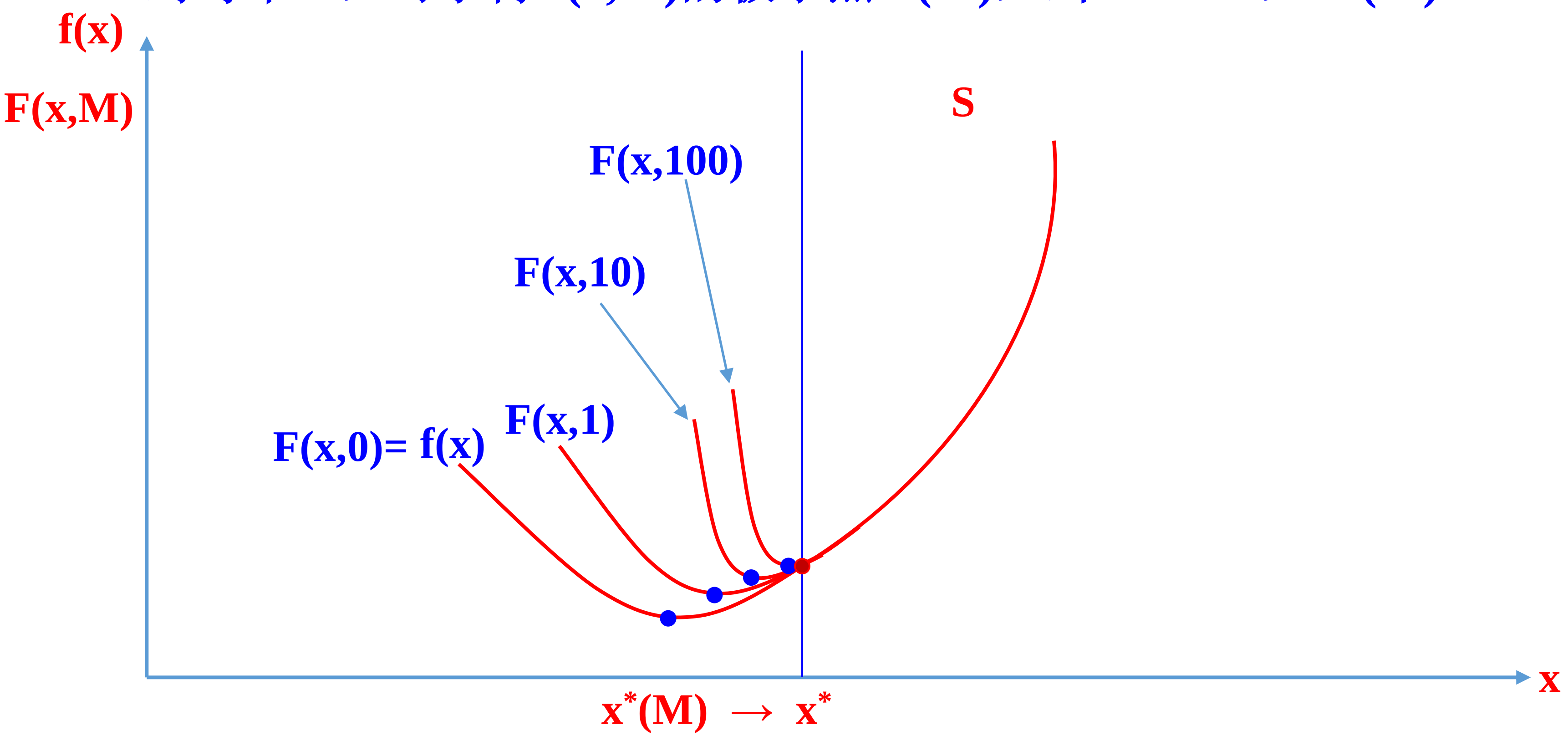
这样将原约束问题转化为一个无约束问题进行求解。

如何表达 F(x) ??? 因为其中出现了+∞

对策：设法构造一系列的函数F(x,M)，不断增大M，使F(x,M)逐步向F(x)逼近。

在S内，F(x,M)的值与f(x)相同，在S之外，给出一个大的“惩罚值”。

对每个M，可求得F(x,M)的极小点$x^*(M)$，当M→ +∞， $x^*(M) \to x^*$



惩罚函数可构造如下

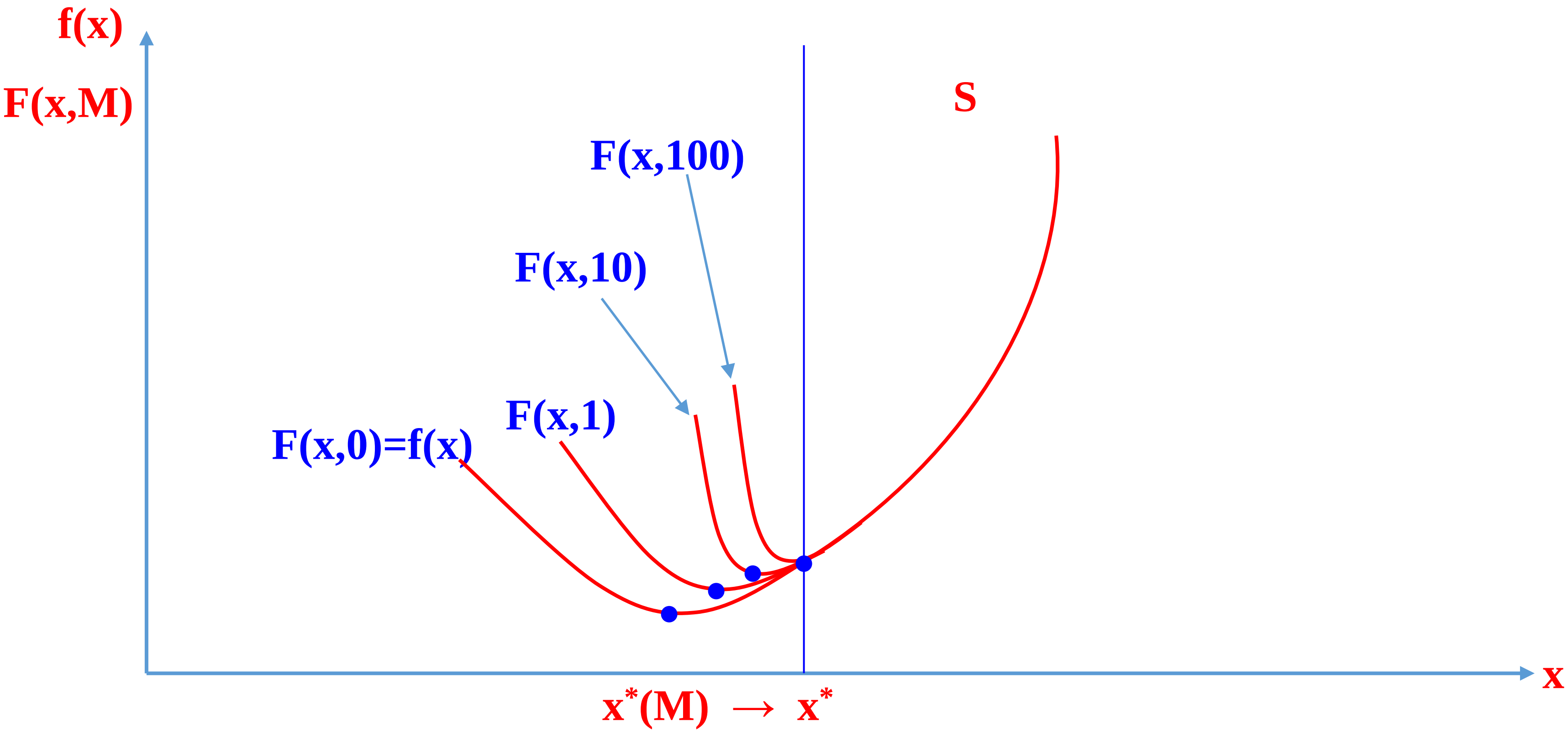
$$F(x,M)=f(x)+M\,P(x)=f(x)+M\left\{\sum_{i=1}^{l}[\max(0,g_i(x))]^2+\sum_{j=1}^{m}[h_j(x)]^2\right\}$$

惩罚因子

惩罚项

当约束不满足时，要么某些$g_i(x)>0$, 要么某些$h_j(x)\neq 0$，当M很大时，F(x,M)非常大，这是对不可行点的惩罚，使该点不会成为约束问题的极小点。

当约束满足时，惩罚项P(x)=0，故F(x,M) = f(x)，没有受到惩罚。



由于$x^*(M)$ 通常是从可行域外部趋于x^*的，所以上式确定的惩罚函数称为外部惩罚函数。这种解法称为外部惩罚函数法或外点法。

例1 用外点法求解问题

$$\begin{cases} \min f(x) = (x_1-4)^2 + (x_2-4)^2 \\ \text{s.t. } h(x) = x_1+x_2-5 = 0 \end{cases}$$

解： 构造惩罚函数 $F(x,M) = (x_1-4)^2 + (x_2-4)^2 + M(x_1+x_2-5)^2$

求F(x,M)的平稳点。对简单问题，可用解析法求解。 即解方程组

$$\begin{cases} \partial F/\partial x_1 = 2(x_1-4) + 2M(x_1+x_2-5) = 0 \\ \partial F/\partial x_2 = 2(x_2-4) + 2M(x_1+x_2-5) = 0 \end{cases}$$

解得 $x_1^*(M)=x_2^*(M) = (5M+4)/(2M+1)$

$M=0,\ x_1(M)=x_2(M)=4$

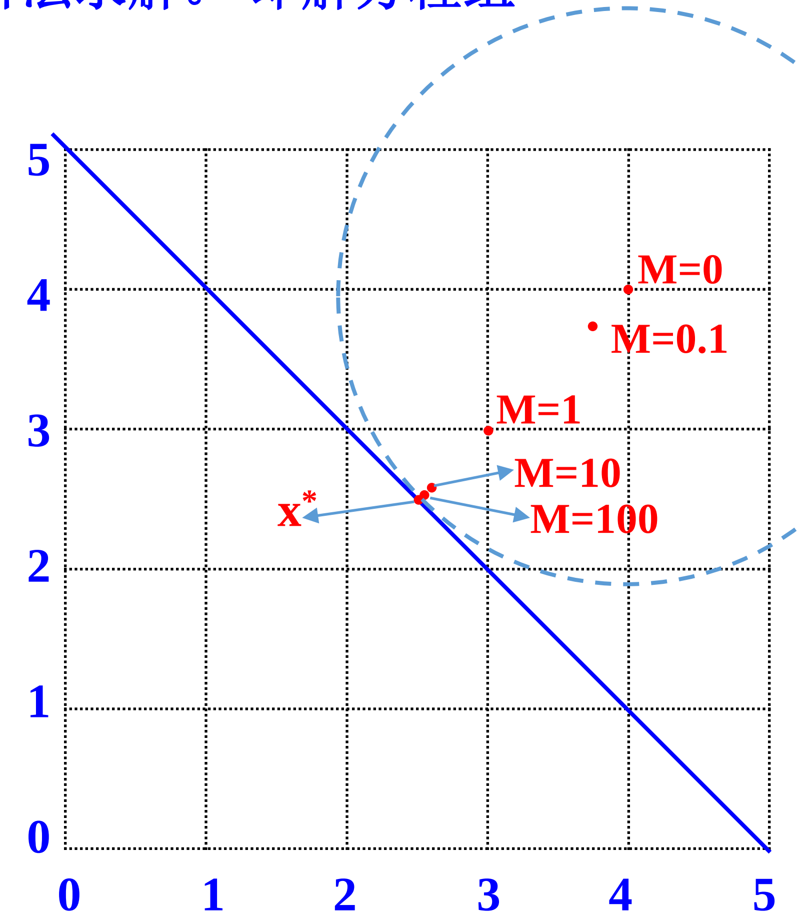
$M=0.1,\ x_1(M)=x_2(M)=3.73$

$M=1,\ x_1(M)=x_2(M)=3$

$M=10,\ x_1(M)=x_2(M)=2.5174$

$M=100,\ x_1(M)=x_2(M)=2.5075$

令$M\to+\infty$，得 $x_1^*(M)=x_2^*(M)\to 2.5$



例2 用外点法解

$$\begin{cases} \min f(x) = (x_1-4)^2 + (x_2-4)^2 \\ \text{s.t. } g(x) = x_1+x_2-5 \le 0 \end{cases}$$

解： 可行域为直线$x_1+x_2-5=0$及其左下方的半平面，记为S

$$F(x,M) = (x_1-4)^2+(x_2-4)^2+M[\max(x_1+x_2-5, 0)]^2$$

$$= \begin{cases} (x_1-4)^2+(x_2-4)^2+M(x_1+x_2-5)^2 & x \notin S \\ (x_1-4)^2+(x_2-4)^2 & x \in S \end{cases}$$

(1) 当$x \notin S$时

$$\begin{cases} \partial F/\partial x_1 = 2(x_1-4)+2M(x_1+x_2-5)=0 \\ \partial F/\partial x_2 = 2(x_2-4)+2M(x_1+x_2-5)=0 \end{cases} \Rightarrow x_1^*(M)=x_2^*(M) = (5M+4)/(2M+1)$$

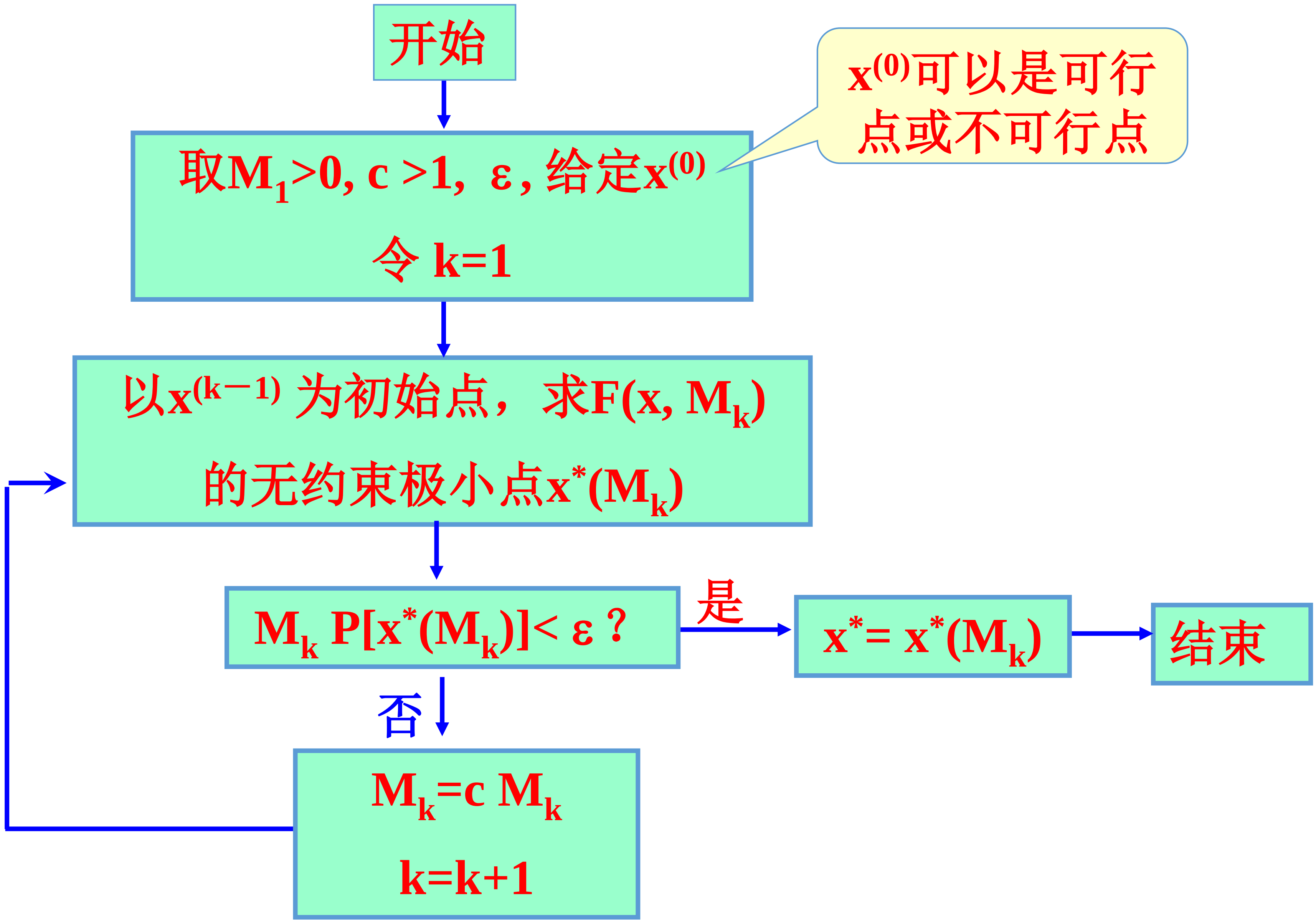
(2) 当$x \in S$时

$$\begin{cases} \partial F/\partial x_1 = 2(x_1-4)=0 \\ \partial F/\partial x_2 = 2(x_2-4)=0 \end{cases} \Rightarrow x_1 = x_2 = 4$$

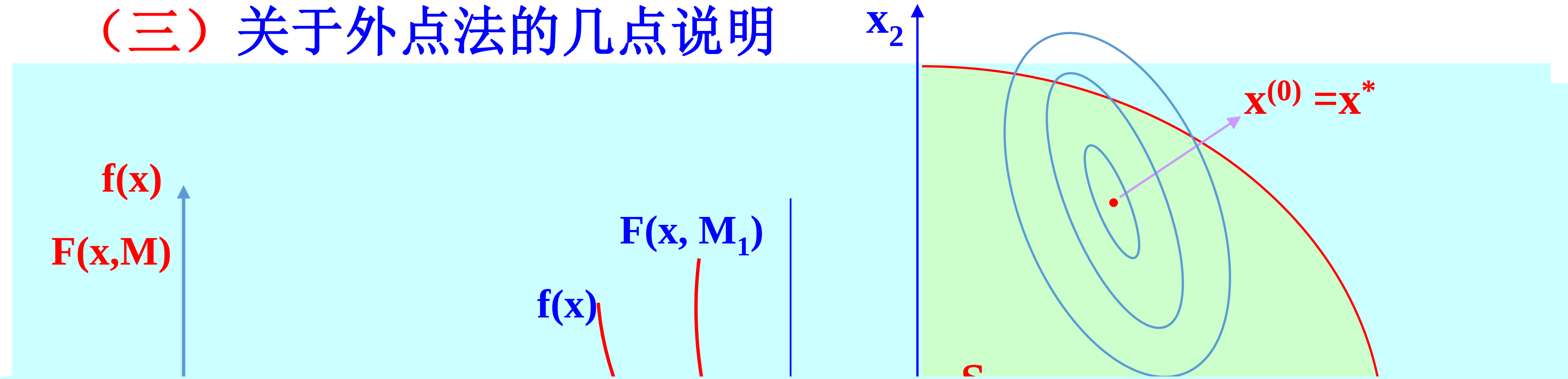
但$g(4,4)=3>0$ 所以$[4,4]^T \notin S$，与所设$x \in S$矛盾，故$\nabla F(x,M)=0$在S中无解。

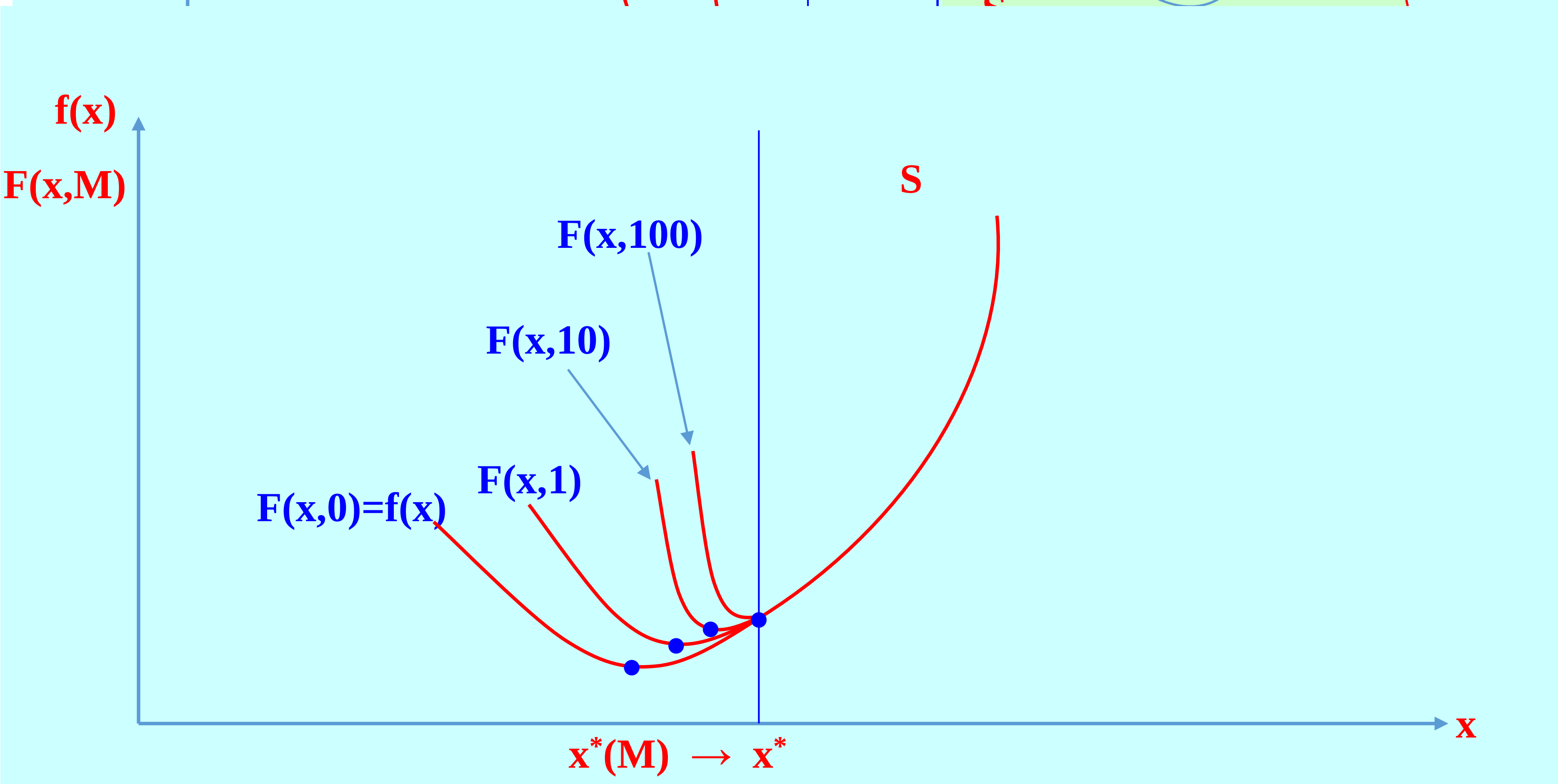
综合上述两种情况，$x_1^*(M)=x_2^*(M) = (5M+4)/(2M+1) \to 2.5$ （当$M \to +\infty$时）

（二）外点法算法


开始
取M_1>0, c >1, ε, 给定$x^{(0)}$
令 k=1
$x^{(0)}$可以是可行点或不可行点
以$x^{(k-1)}$ 为初始点，求$F(x, M_k)$
的无约束极小点$x^*(M_k)$
$M_k\,P[x^*(M_k)] < ε$?
是
$x^* = x^*(M_k)$
结束
否
$M_k = c\,M_k$
k=k+1

（三）关于外点法的几点说明





定理2 设原约束问题中f, g_i, h_j 连续，最优解x^*存在。令$M_{k+1} \geq M_k > 0$ (k=1, 2, …)，

$M_k \rightarrow +\infty$。设对每个M_k，惩罚函数$F(x, M_k)$有极小点$x^*(M_k)$，且序列

$\{x^*(M_k)\}$收敛到某点 $\tilde{x}$，则 $\tilde{x} = x^*$，且成立

$$\lim_{k\to\infty} M_K P[x^*(M_k)] = 0$$

$$\lim_{k\to\infty} F(x^*(M_k),\ M_k) = f(x^*) = f^*$$

3、不能一开始就把M_k取得很大。

当迭代点离极小点较远时，M_k越大，$F(x, M_K)$的Hesse矩阵的条件数越小（矩阵越病态），从而引起相关数值计算的困难。

矩阵A的条件数 $r(A)=\|A\|\times\|A^{-1}\|$，当r(A)很小，称A是病态矩阵

所以只能开始时把M_K取得小些，逐步放大。这样增加了计算量，有时还得不到满意的结果，这是外点法的主要缺点。

4、通过求解一系列无约束问题解约束问题的方法，称为序列无约束极小化技术

§2 内部惩罚函数法（内点法）

此法只能求解不等式约束问题，因为要求此法迭代点始终是内点

$$\begin{cases} \min f(x) \\ s.t.\ g_i(x) \le 0 \quad i=1,2,\ldots,l \end{cases}$$

（一）基本思想

在可行域内选取初始点，构造惩罚函数，当迭代点$x^{(k)}$离边界较远时，给出的惩罚值很小，而当接近$x^{(k)}$边界时，惩罚值陡然增加，有如在边界上筑起一道“墙”，使迭代点不能越过边界，故此法亦称为“围墙法”、“内点法”。

（二）内点法的惩罚函数

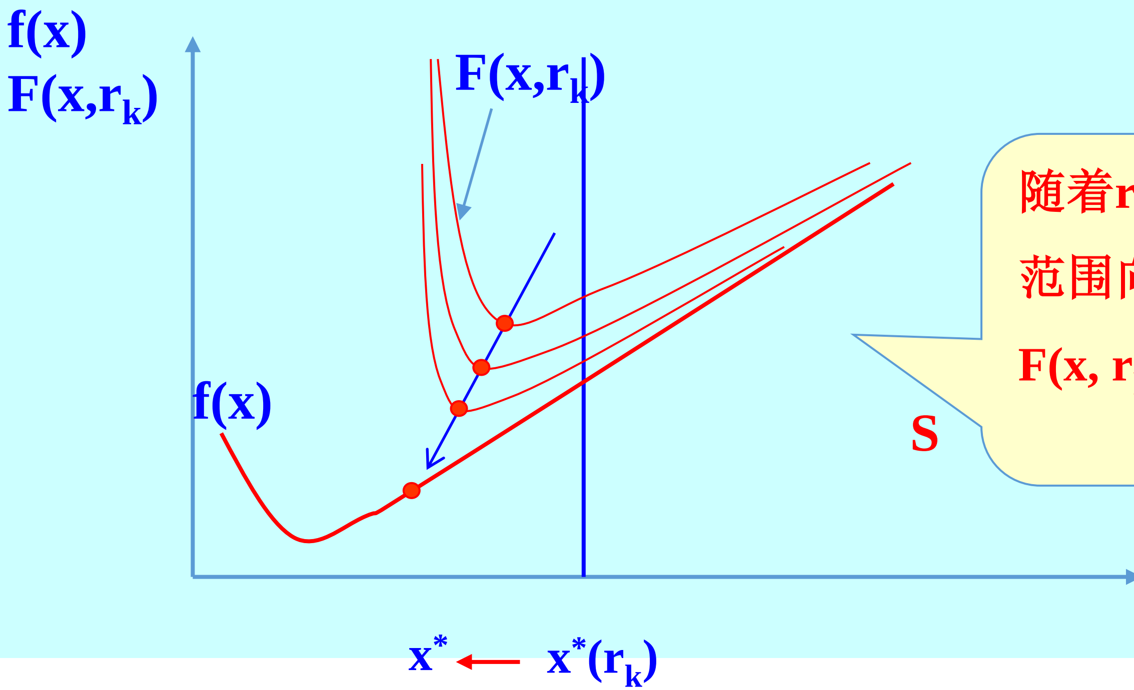
$$F(x,r_k)=f(x)-r_k\sum_{i=1}^{l}\frac{1}{g_i(x)} \quad \text{或} \quad F(x,r_k)=f(x)-r_k\sum_{i=1}^{l}\ln[-g_i(x)]$$

r_k称为惩罚因子，$r_k > r_{k+1} > 0$，且 $\lim\limits_{k\to\infty} r_k = 0$

上式第二项称为惩罚项，记为 $r_kB(x)$，$B(x)$称为“围墙”函数

$$F(x,r_k)=f(x)-r_k\sum_{i=1}^{l}\frac{1}{g_i(x)}$$

对每个r_k，求$F(x, r_k)$的无约束极小点$x^*(r_k)$，$r_k\to 0$，$x^*(r_k)\to x^*$



随着$r_k\to 0$，严厉惩罚的范围向边界靠近，从而使$F(x, r_k)\approx f(x)$的范围扩大。

若原约束问题的最优解x^*存在，令$r_k\geq r_{k+1}>0$ $(k=1, 2, \ldots)$，$r_k\to 0$；设对每个r_k，惩罚函数$F(x, r_k)$有极小点$x^*(r_k)$，且序列$\{x^*(r_k)\}$收敛到某点$\tilde{x}$，则$\tilde{x}=x^*$，且成立

$$\lim_{k\to\infty}F(x^*(r_k), r_k)=f(x^*)=f^*$$

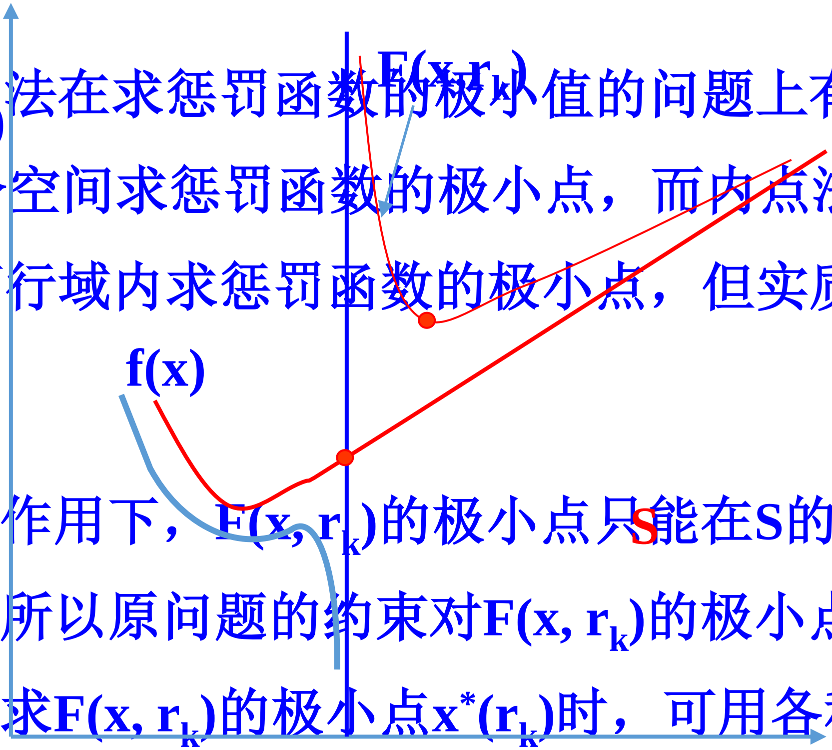
显然，内点法是一种序列无约束极小化方法。

注意点：

(1) 内点法与外点法在求惩罚函数的极小值的问题上有重要区别。

外点法在整个空间求惩罚函数的极小点，而内点法只是在可行域内求极小。

(2) 内点法只在可行域内求惩罚函数的极小点，但实质上还是一个无约束问题。

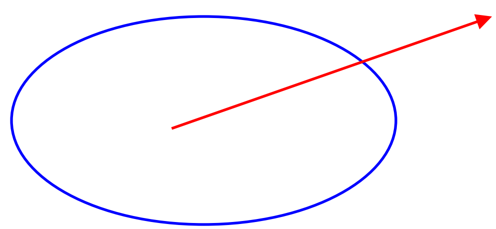


怎么理解？

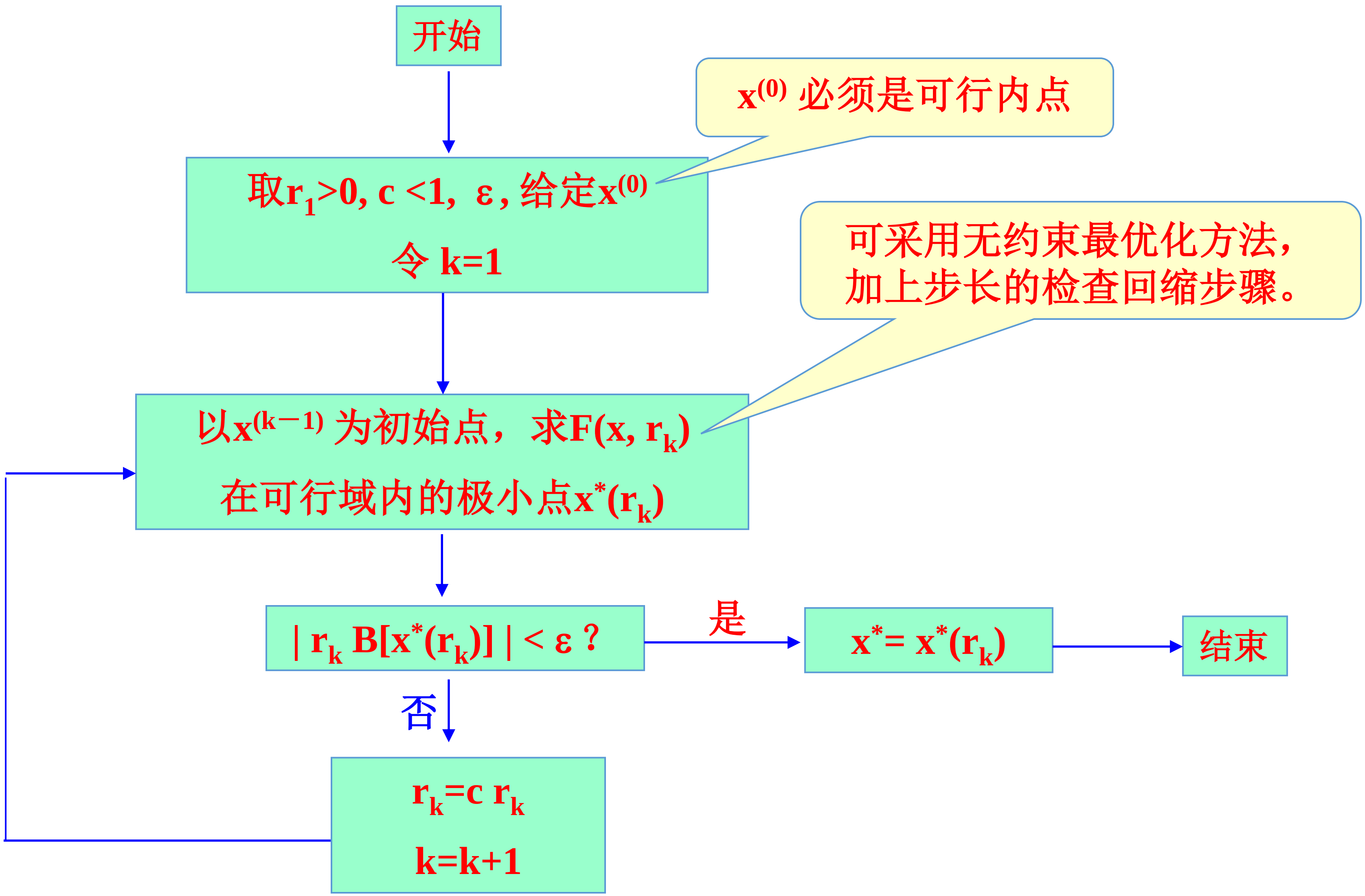
因为在惩罚项的作用下，$F(x, r_k)$的极小点只能在S的内点上达到，迭代点不会跑到边界上去，所以原问题的约束对$F(x, r_k)$的极小点$x^*(r_k)$来说，都是不起作用的约束。所以求$F(x, r_k)$的极小点$x^*(r_k)$时，可用各种无约束问题的解法。

但是在使用搜索法时，如果步长取得太大，迭代点有可能越过边界。

为避免这种情况发生，迭代过程中应检验 $g_i(x)$的符号，一旦某些$g_i(x)>0$，应缩小步长，以保证迭代点为严格的内点。



（三）内点法的算法


开始
$x^{(0)}$ 必须是可行内点
取$r_1>0$, $c<1$, ε, 给定$x^{(0)}$
令 $k=1$
可采用无约束最优化方法，
加上步长的检查回缩步骤。
以$x^{(k-1)}$为初始点，求$F(x, r_k)$
在可行域内的极小点$x^*(r_k)$
$|\, r_k B[x^*(r_k)] \,| < \varepsilon$?
是
$x^* = x^*(r_k)$
结束
否
$r_k = c\, r_k$
$k=k+1$

例4 用内点法解

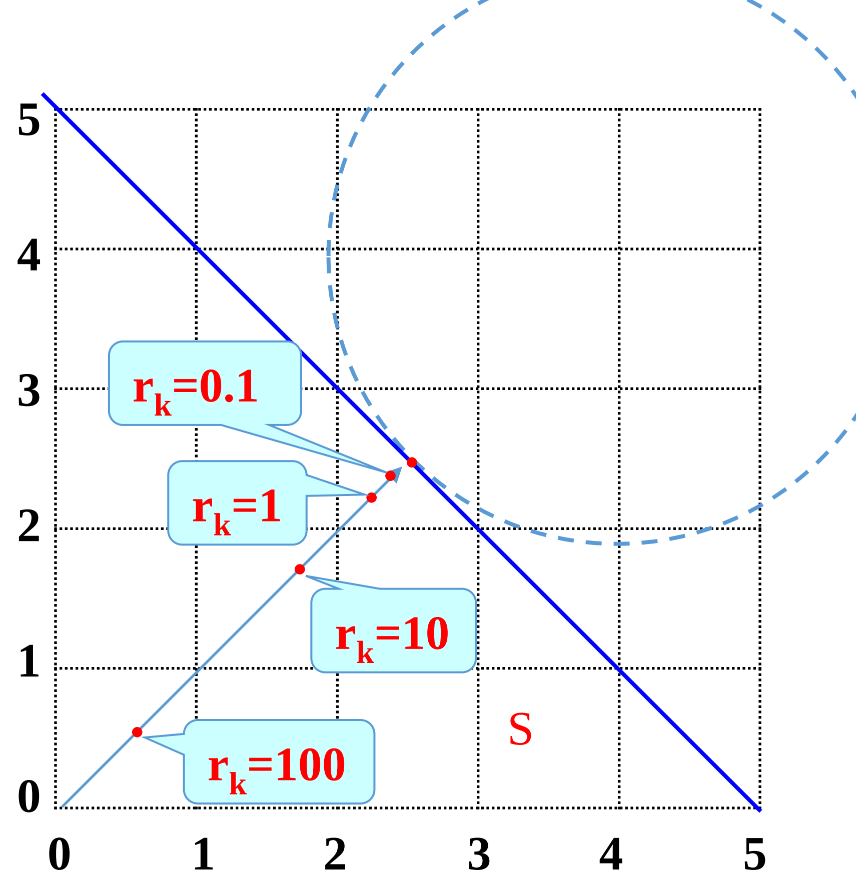
$$\begin{cases} \min f(x) = (x_1 - 4)^2 + (x_2 - 4)^2 \\ \text{s.t.}\ \ g(x) = x_1 + x_2 - 5 \le 0 \end{cases}$$

解：构造惩罚函数

$$F(x, r_k) = (x_1 - 4)^2 + (x_2 - 4)^2 - r_k/(x_1 + x_2 - 5)$$

求$F(x, r_k)$ 的平稳点，即解方程组

$$\begin{cases} \partial F/\partial x_1 = 2(x_1 - 4) + 2r_k(x_1 + x_2 - 5)^{-2} = 0 \\ \partial F/\partial x_2 = 2(x_2 - 4) + 2r_k(x_1 + x_2 - 5)^{-2} = 0 \end{cases}$$



由于 $x_1 = x_2$，所以得

$$4x_1^3 - 36x_1^2 + 105x_1 - 100 + 0.5r_k = 0$$

r_k	100	10	1	0.1	0.01	0.001→0
$x_1^*(r_k) = x_2^*(r_k)$	0.5864	1.7540	2.2340	2.4113	2.4714	2.4909 →2.5= x^*
$f(x^*(r_k))$	23.3053	10.0890	6.2375	5.0479	4.6732	4.5548 →4.5= $f(x^*)$
$F(x^*(r_k), r_k)$	49.4341	16.7914	8.1172	5.6116	4.8480	4.6097 →4.5= $f(x^*)$

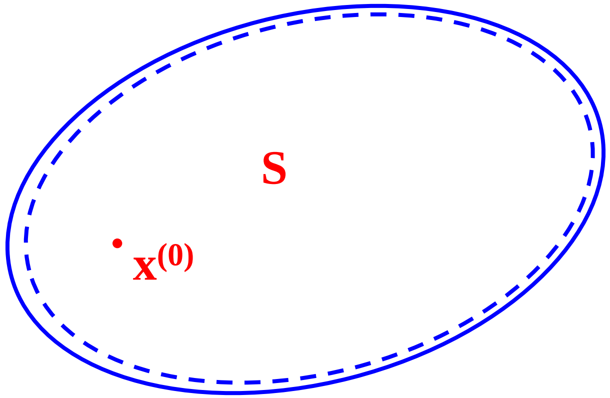
（四）初始可行内点的求法

1、约束不可行性直接最小化法

采用如下的外点惩罚函数型目标函数：

$$f(x)=\sum_{i=1}^{l}[\max(0,\ g_i(x)+\varepsilon)]^2$$

求解无约束问题 **min f(x)**，若 **min f($x^{(0)}$)＝0**，则得到的极小点$x^{(0)}$将是一个严格的内点，即$g_i(x^{(0)})\leq -\varepsilon$



ε称为容差参数，不能取得太大，否则可能造成问题无解。

此法常用，但当约束较多且高度非线性时，目标函数较复杂，计算较困难。

2、随机选择法

随机事件：布朗运动，抛硬币时出现正反面 ...

随机数：在[0,1]均匀分布

计算机产生的随机数为“伪随机数”，按一定的算法或程序产生。如

8520478 $\xrightarrow{x^2}$	72598545348484 $\xrightarrow{x^2}$	35825647615209 $\xrightarrow{x^2}$	68169395938576 $\xrightarrow{x^2}$	2869497093681
0. 8520478	0. 5985453	0. 8256476	0. 1693959	0. 6949709

48298455184681 $\xrightarrow{x^2}$	8907544671601 $\xrightarrow{x^2}$	569220454089 $\xrightarrow{x^2}$	85016771966116 $\xrightarrow{x^2}$	……
0. 2984551	0. 0754467	0. 9220454	0. 0167719	……

随机选择法产生初始可行内点的步骤：

(1) 估计变量x各分量x_i的上下限：

$$a_i \le x_i \le b_i \qquad (i=1,2,\dots,n)$$

(2) 产生试验点x, 各分量取为：

$$x_i = a_i + R_i(b_i - a_i) \qquad (i=1,2,\dots,n)$$

R_i为计算机产生的伪随机数

(3) 验证试验点是否满足约束条件

$$g_i(x) < 0 \qquad (i=1,2,\dots,l)$$

若满足，即得可行内点$x^{(0)}$, 否则回(2)

此法只适用于小规模问题

§ 3 外点法与内点法的比较

（一）优点：适用范围广，对函数$f(x)$, $g_i(x)$, $h_j(x)$无特殊要求；使用方便，只需用 解无约束问题的算法；比较有效，所以常用

（二）缺点：(1) 工作量大，必须求解一系列无约束问题；

(2) 参数M_k, r_k的选取对收敛速度影响大；

(3) $M_k \to +\infty$或$r_k \to 0$，惩罚函数变得越来越病态，使计算困难。

（三）各自特点：

1、内点法要求初始点为可行内点，外点法无此要求；

2、内点法不适用于等式约束问题（老），外点法可用于所有约束问题；

3、当迭代点进入可行域中时，外点法立即得到原问题的最优解，而内点法无此特点；

4、外点法的惩罚函数在边界上可微性差，选无约束方法时受限制

5、内点法的迭代点都是可行点，而外点法的迭代点经常是不可行点，实际问题可能不允许

§ 增广Lagrange乘子法(ALM法)

至此已讨论过的约束问题解法：

- **Lagrange乘子法**：最后化为用最小二乘法求解非线性方程组问题，目标函数复杂
- **惩罚函数法**：转化为求解一系列无约束问题，当$M_k \to +\infty$或$r_k \to 0$时，惩罚函数在边界处很陡，Hesse矩阵病态程度严重，计算可能出现困难

- 1969年Powell等人提出将两者结合起来，成为增广Lagrange乘子法。
- 后来Fletcher等人对此法进行了改进，使其成为一个有效的方法，在收敛速度和数值稳定性上都比简单的惩罚函数法优越。

- 增广Lagrange乘子法的主要思想：在惩罚函数中引入Lagrange乘子，或在Lagrange函数中引入惩罚项， 试图在惩罚因子M_k不太大的情况下，通过调节Lagrange乘子，逐次求解无约束问题，逼近原约束问题的最优解。

§6.3.1 解等式约束问题的ALM法

$$\begin{cases} \min f(x) \\ \text{s.t. } h_j(x) = 0 \quad j=1,2,\ldots,m \end{cases}$$

引入增广Lagrange函数

Lagrange函数L(x,μ)

惩罚项

$$A(x,\mu,M_k) = f(x) + \sum_{j=1}^{m} \mu_j h_j(x) + M_k \sum_{j=1}^{m} [h_j(x)]^2$$

考察一下惩罚项对x的梯度 $\nabla_x \{M_k \sum_{j=1}^{m} [h_j(x)]^2\} = 2M_k \sum_{j=1}^{m} h_j(x) \nabla h_j(x) \xlongequal{\text{当}x\in S\text{时}} 0$

所以，当x为可行点时, 固定M_k，有 $\nabla_x A(x,\mu,M_k) = \nabla_x L(x,\mu)$

若存在x^*及μ^*，使 $\nabla_x A(x^*, \mu^*, M_k) = 0$ ，则有 $\nabla_x L(x^*, \mu^*) = 0$

即对于$\mu=\mu^*$及某个固定的M_k，若x^*为$A(x, \mu, M_k)$的无约束极小点，则x^*是原问题的极小点

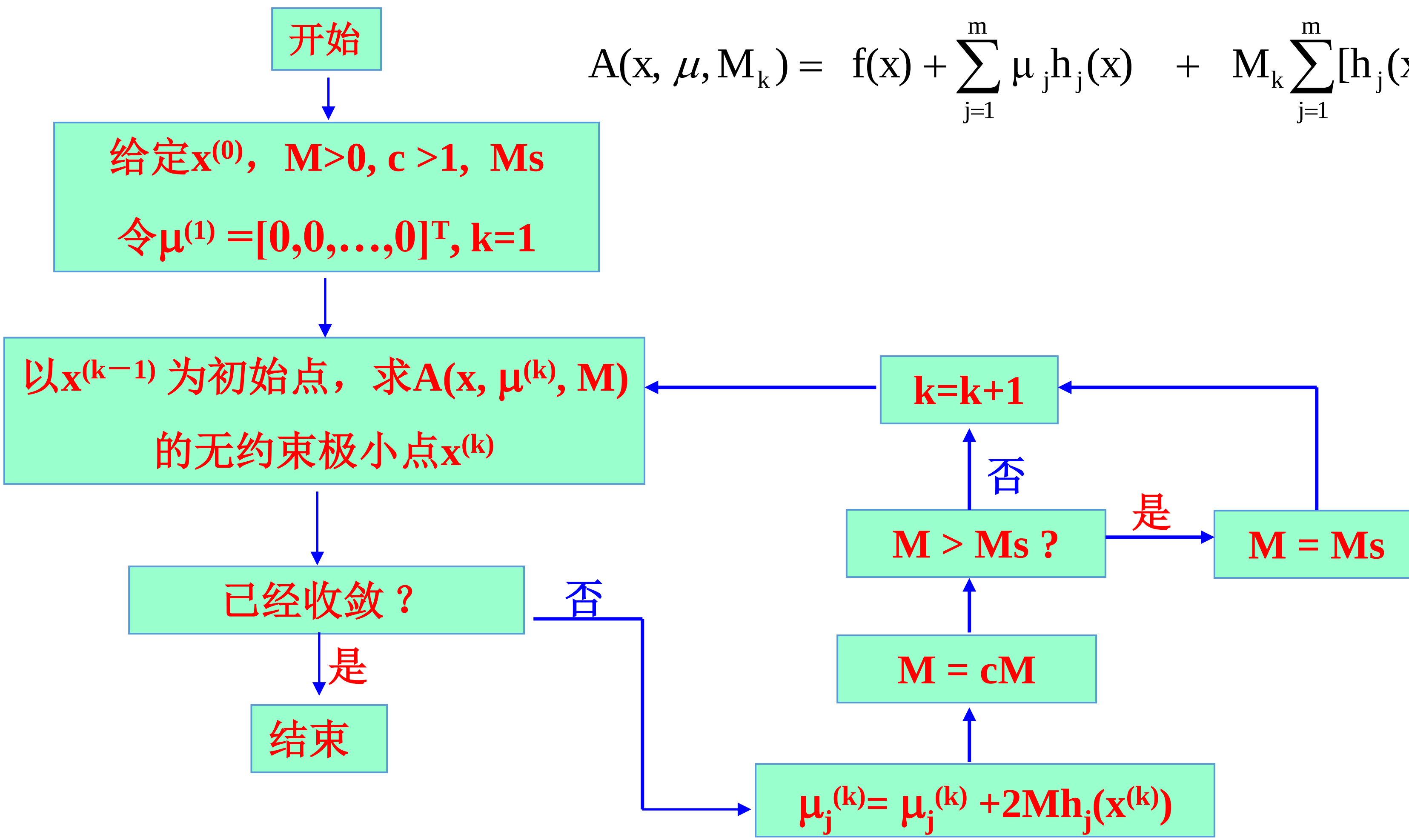
对函数$A(x, \mu^*, M_k)$只要进行一次无约束最优化就可求得原约束问题的极小点x^*

对函数$A(x, \mu^*, M_k)$只要进行一次无约束最优化就可求得原约束问题的极小点x^*

？ 问题是事先并不知道 μ^* ？

对策：用迭代的办法，从任意的μ（通常取为零或单位向量）逼近μ^*

$$A(x, \mu, M_k) = f(x) + \sum_{j=1}^{m} \mu_j h_j(x) + M_k \sum_{j=1}^{m} [h_j(x)]^2$$



例 用增广Lagrange乘子法解问题

$$\begin{cases} \min f(x) = x_1^2 + x_2^2 \\ s.t.\ \ h(x) = x_1 + x_2 - 1 = 0 \end{cases}$$

解：用Lagrange乘子法解得$x^* = [x_1^*, x_2^*]^T = [0.5, 0.5]^T$, $\mu^* = -1$

增广Lagrange函数为 $A(x, \mu, M_k) = x_1^2 + x_2^2 + \mu(x_1 + x_2 - 1) + M_k (x_1 + x_2 - 1)^2$

在下面的三种情况中，保持 $M_k = 1$

1、初始取$\mu^{(1)} = 0$

无约束问题求解次数	μ	$x_1 = x_2$	$h(x)$
1	0.0000	0.33333	−0.33333
2	−0.6667	0.44444	−0.11111
3	−0.8889	0.48148	−0.03704
4	−0.9630	0.49382	−0.01235
5	−0.9877	0.49794	−0.00041

2、初始取$\mu^{(1)} = -2$

无约束问题求解次数	μ	$x_1 = x_2$	$h(x)$
1	−2.0000	0.66667	0.33333
2	−1.3333	0.55556	0.11111
3	−1.1111	0.51852	0.03704
4	−1.0370	0.50618	0.01235
5	−1.0123	0.50205	0.00041

3、初始取$\mu^{(1)} = -1$

求解一次就得

$x_1^* = x_2^* = 0.5$

§ 解不等式约束问题的ALM法

$$\begin{cases} \min f(x) \\ \text{s.t. } g_i(x) \le 0 \quad i=1,2,\dots,l \end{cases}$$

首先引入松弛变量w_i，把不等式约束化为等式约束

$$g_i(x) + w_i^2 = 0 \quad i=1,2,\dots,l$$

$$A(x, \lambda, w, M_k) = f(x) + \sum_{i=1}^{l} \{\lambda_i [g_i(x) + w_i^2] + M_k [g_i(x) + w_i^2]^2\}$$

这样可在M_k不太大的情况下，通过调节Lagrange乘子，逐次求解无约束问题。

§6.3.3 解一般约束问题的ALM法

$$\begin{cases} \min f(x) \\ \text{s.t. } g_i(x) \le 0 \quad i=1,2,\dots,l \\ \qquad h_j(x) = 0 \quad j=1,2,\dots,m \end{cases}$$

首先引入松弛变量w_i，把不等式约束化为等式约束

$$A(x, \mu, \lambda, w, M_k) = f(x) + \sum_{i=1}^{l} \{\lambda_i [g_i(x) + w_i^2] + M_k [g_i(x) + w_i^2]^2\} + \sum_{j=1}^{m} \{\mu_j h_j(x) + M_k [h_j(x)]^2\}$$

这样可在M_k不太大的情况下，通过调节Lagrange乘子，逐次求解无约束问题。

ALM法的特点：

(1) 方法对M_k比较不敏感，不需要将M_k增加到无穷大；

(2) 精确地达到$g_i(x) = 0$ 和$h_j(x) = 0$ 是可能的；

(3) 可通过更新Lagrange乘子的值来实现迭代的加速；

(4) 初始点可以是可行点，也可以是不可行点；

(5) 达到最优点处，由值$\lambda_i^* \neq 0$可自动识别出哪些是起作用约束；

$$\boxed{\lambda_i^* g_i(x^*) = 0}$$

(6) 如果各个无约束子问题能顺利求解，那么 $x^* = \lim x^{(k)}$是一个K－T点；

(7) 如果M_k选得合理，那么 $\min A(x, \lambda,\mu, M_K, w)$ 的难度与$\min f(x)$差不多；

(8) λ,μ, M_K的更新规则是简单的，不需要额外的计算量；

(9) 虽然仍要解一系列无约束问题，但是由于M_k是某个有限值，因而不会出现惩罚函数法中的病态现象，比惩罚函数法有更好的效果。

约束直接搜索法

Lagrange乘子法、惩罚函数法等均是对约束条件进行隐含或间接处理，将原问题化为无约束问题求解。

工程问题经常出现函数的导数不连续或不存在或难求的情况，则只能利用目标函数和约束函数的函数值进行约束直接搜索。

约束直接搜索法的缺点是计算量大，适用于维数低、函数复杂、精度要求不太高的问题。优点是准备时间和准备工作量小，搜索逻辑简单。

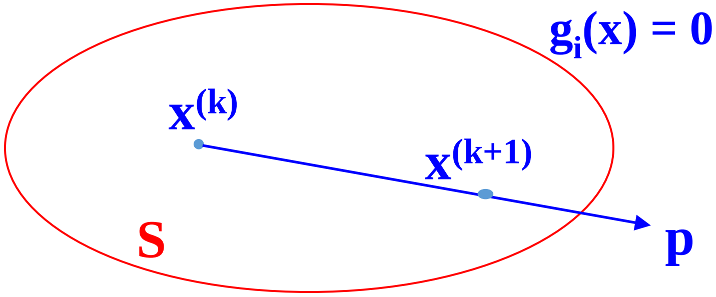
两类约束直接搜索法

- 无约束直接搜索法的改造推广
- 试验点的随机选择为基础

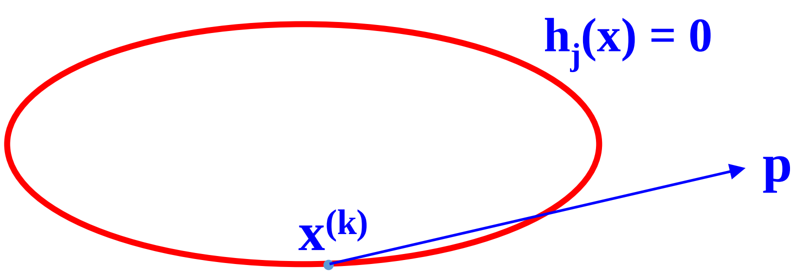
§ 约束直接搜索法的解题准备

（一）将问题中的等式约束消去

对不等式约束问题[$g_i(x) \le 0$]，可使用无约束问题的直接搜索法，加上步长回缩步骤，能保证迭代点的下降且可行。



对等式约束问题[$h_j(x) = 0$]，若使用无约束直接搜索法，则不能保证迭代点一定是可行下降点



所以必须首先采用消元法等消去等式约束

（二）寻找一个初始可行点

a. 惩罚函数最小化法

b. 随机选择法

c. 序列约束最小化方法