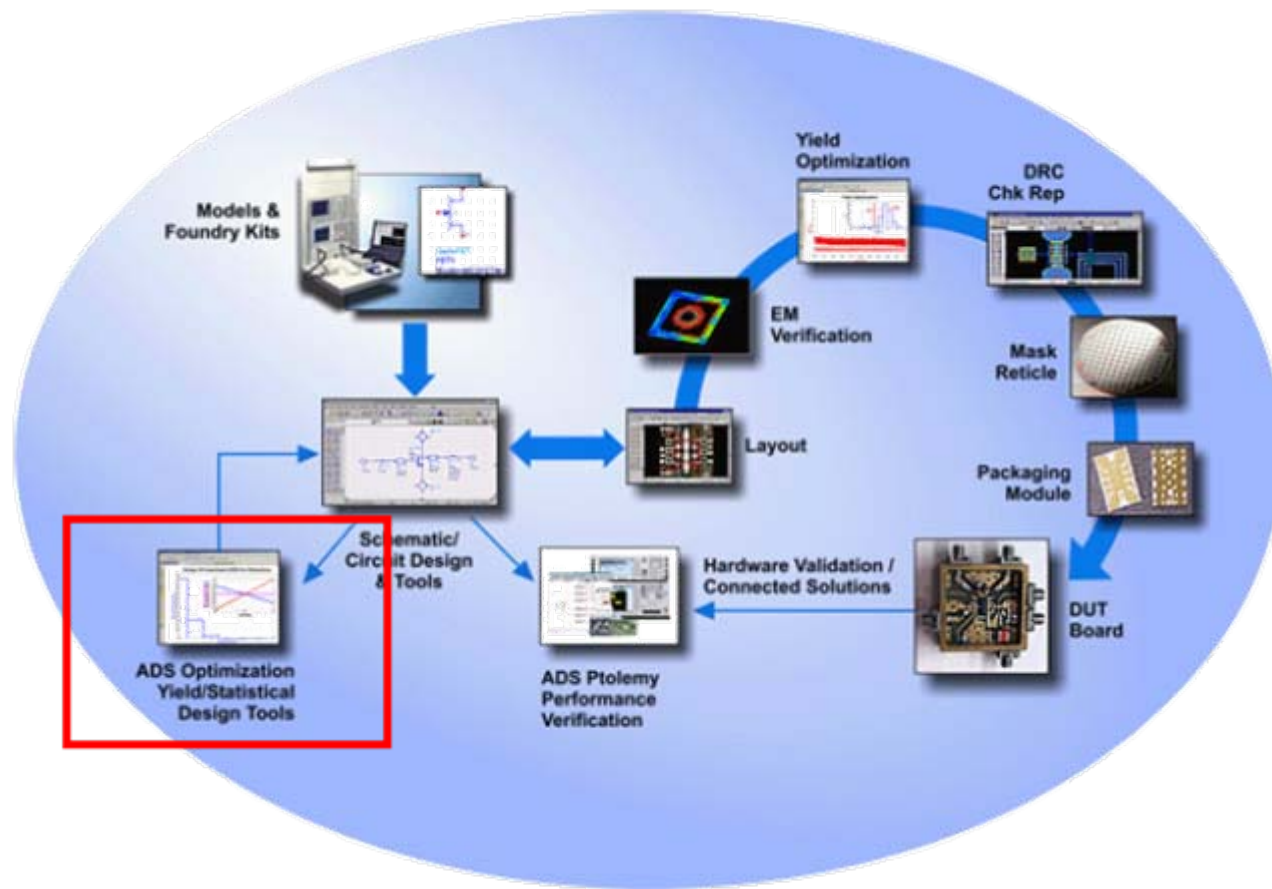
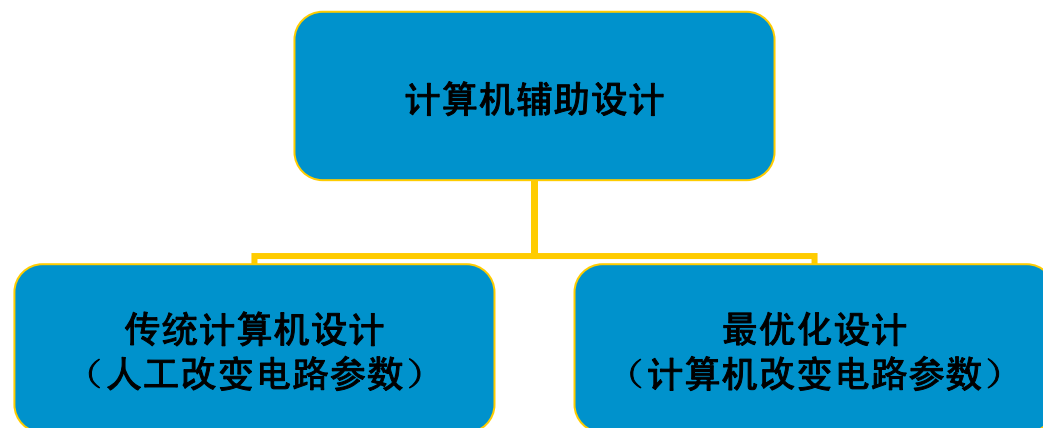


第四章 最优化方法和最优化设计



§ 4.1最优化设计的基本原理



传统计算机设计

- ❖ 人工改变电路参数。
- ❖ 计算机：快速计算。
- ❖ 人：分析判断结果，修改电路。
- ❖ 一种半自动的方法。
- ❖ 主要问题：在设计中，每次电路修改都必须有一次输入和输出，再加上设计者对每次输出结果的分析判断，使整个设计时间延长，机器的使用效率很低，造成很大的机器时间浪费。

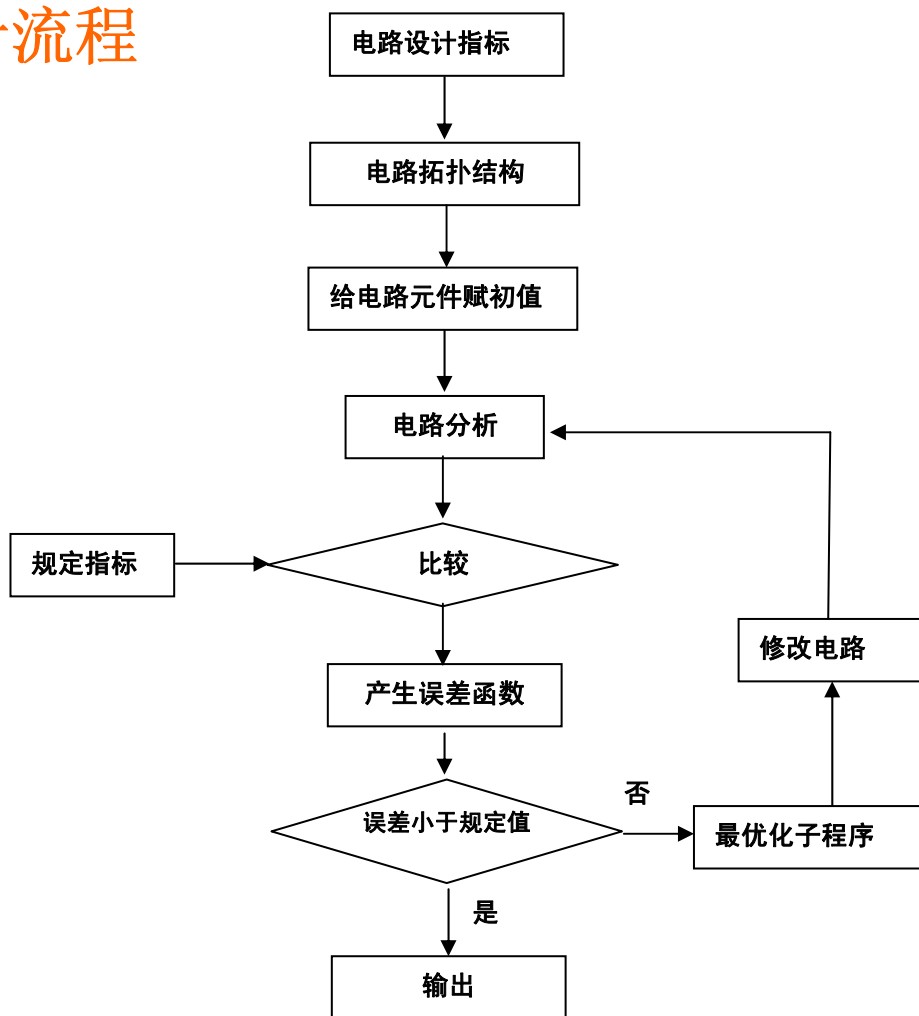
最优化设计

- ❖ 计算机改变电路参数。
- ❖ 在给定电路拓扑情况下优化其元件参数，使电路性能“最优”，或达到预定的目标值。（CAD）
- ❖ 对电路拓扑和元件参数都进行优化，使电路性能达到“最优”。（EDA）



第四章 最优化方法和最优化设计

三、计算机优化设计流程



第四章 最优化方法和最优化设计

1、电路初始拓扑的确定：

a, 滤波器（低通、带通、高通）

b, 功分器

c, 定向耦合器

d, 混频、倍频电路

e, 放大器：

LNA：低噪声放大器；**P.A**:功率放大器

传统上初始电路的选择，主要是依靠经验。现在很多常用的电路形式，在ADS里都可以用design guide，直接给定电路的初始拓扑。



第四章 最优化方法和最优化设计

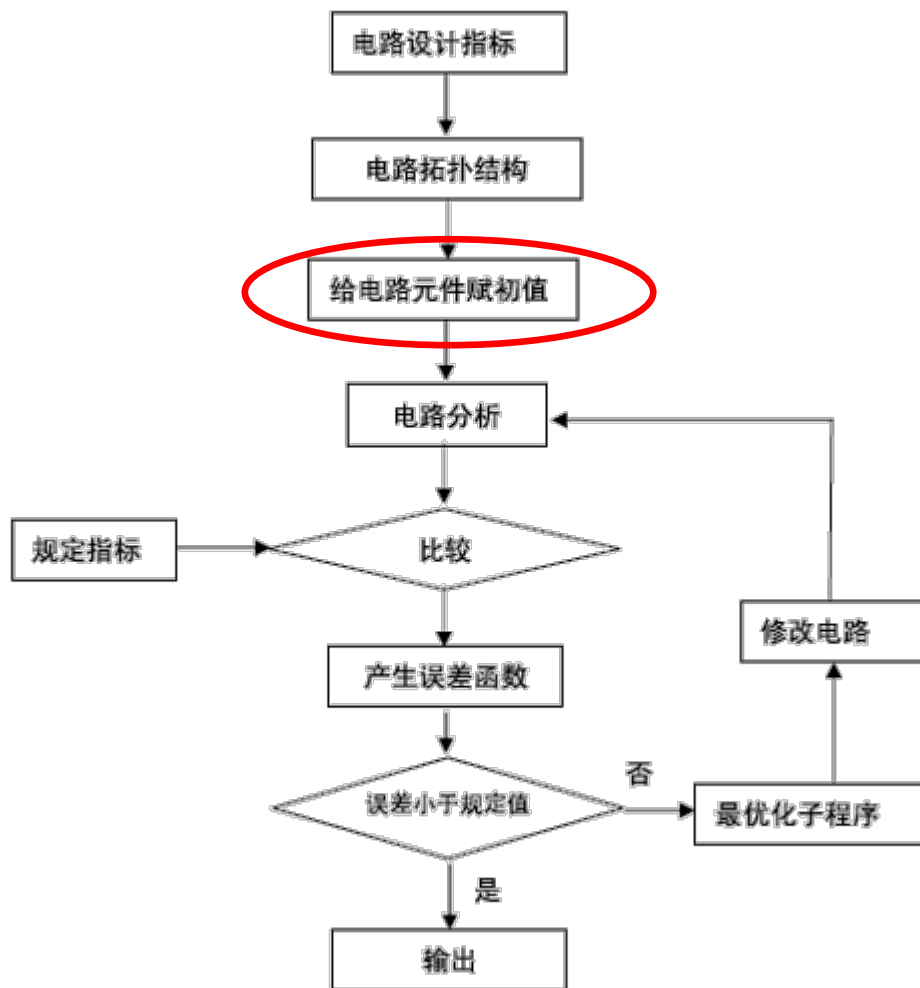
2、初始值的确定：

➤ 经验设计

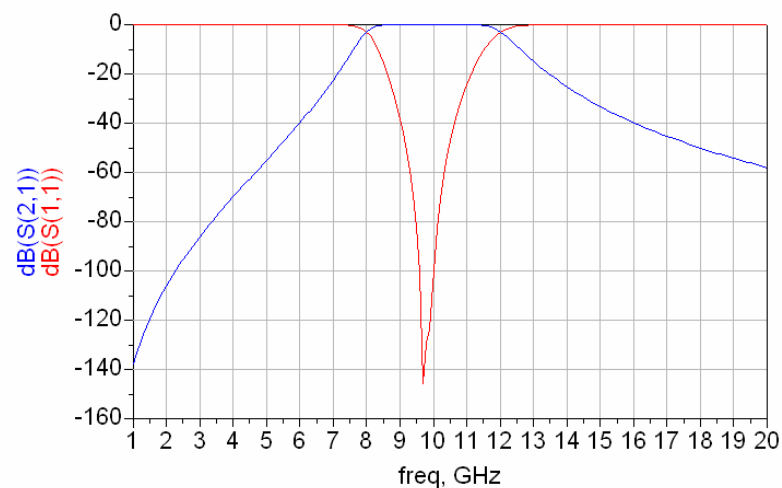
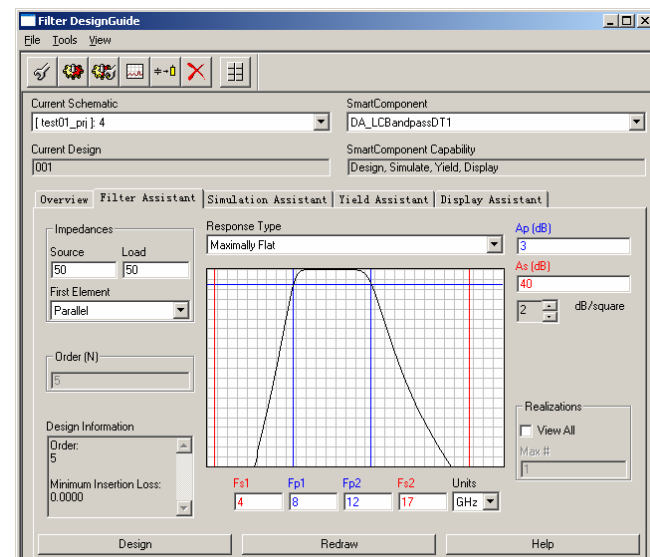
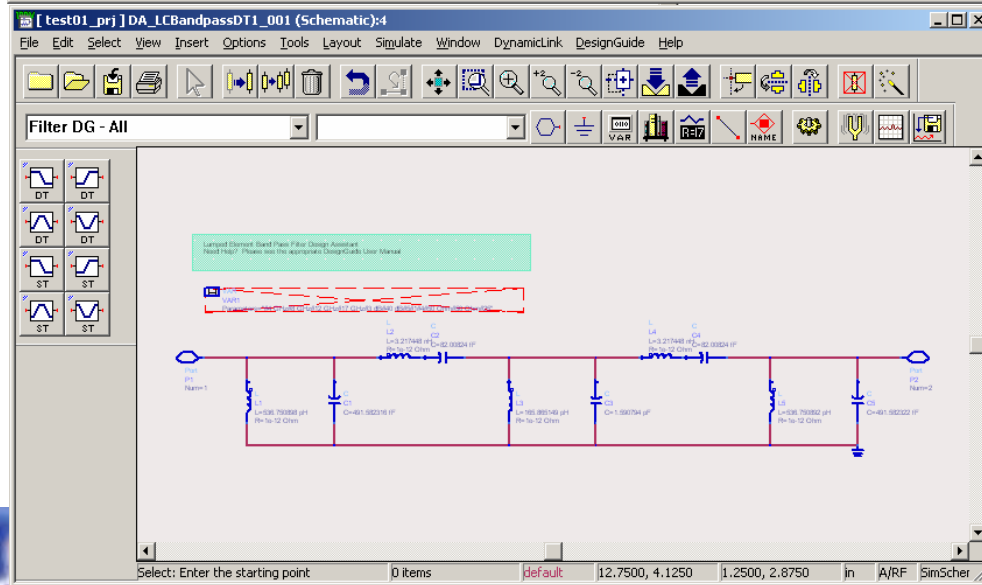
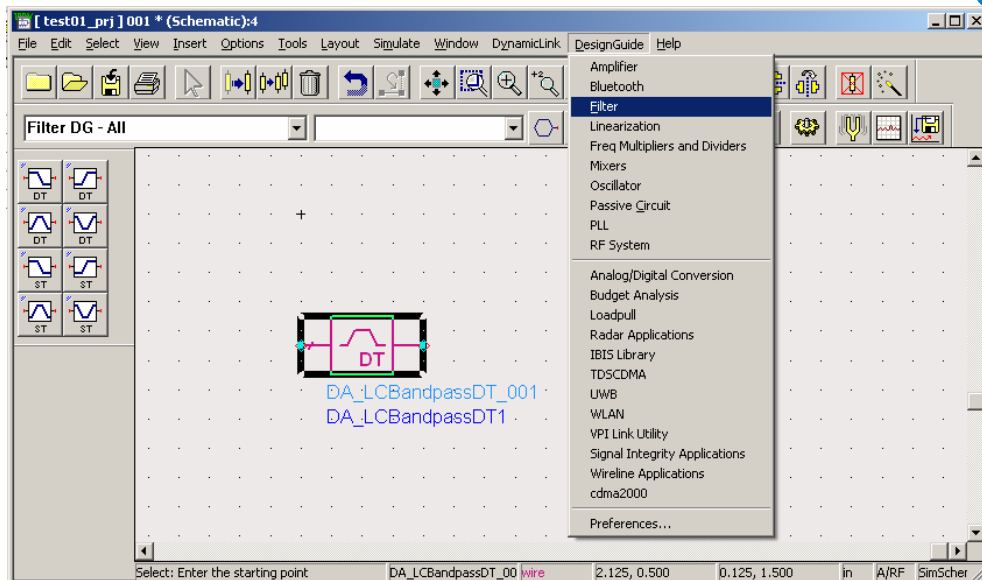
➤ 编程综合的方法

➤ 大型设计软件

如：ADS的design guide



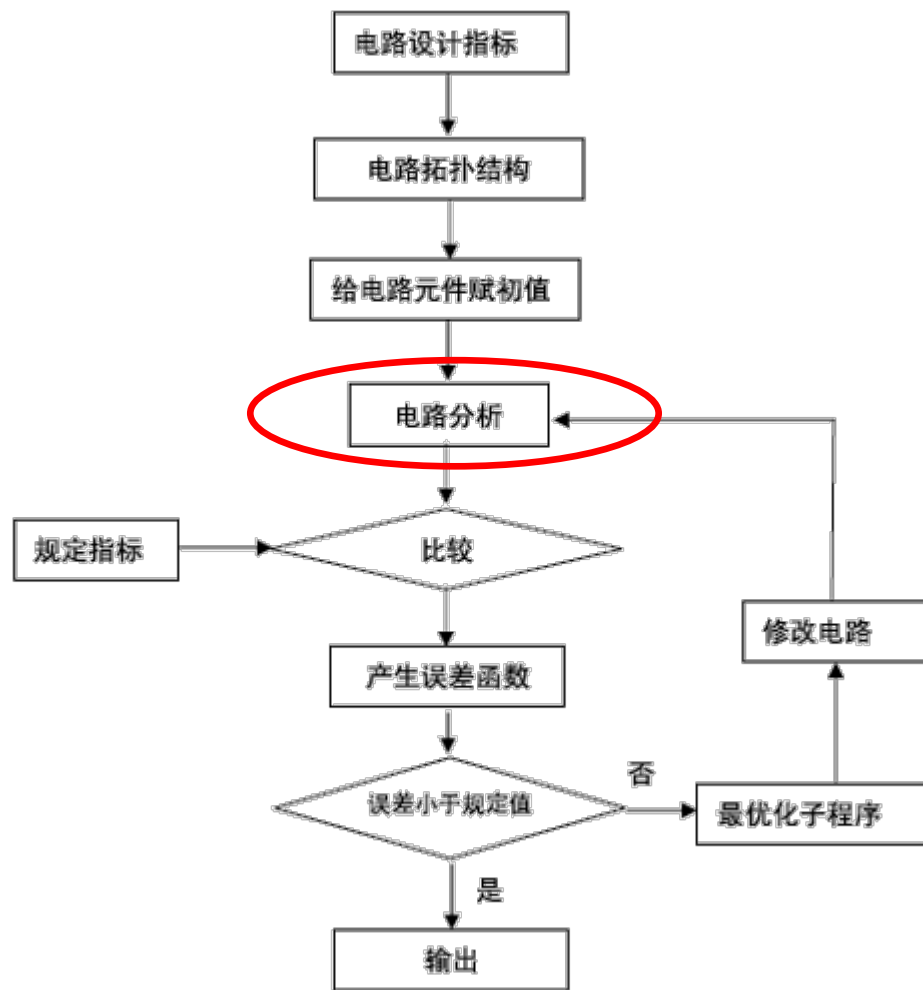
ADS – Design Guide



第四章 最优化方法和最优化设计

3、电路分析：

- 传递矩阵法
- 节点导纳矩阵法
- 散射矩阵法

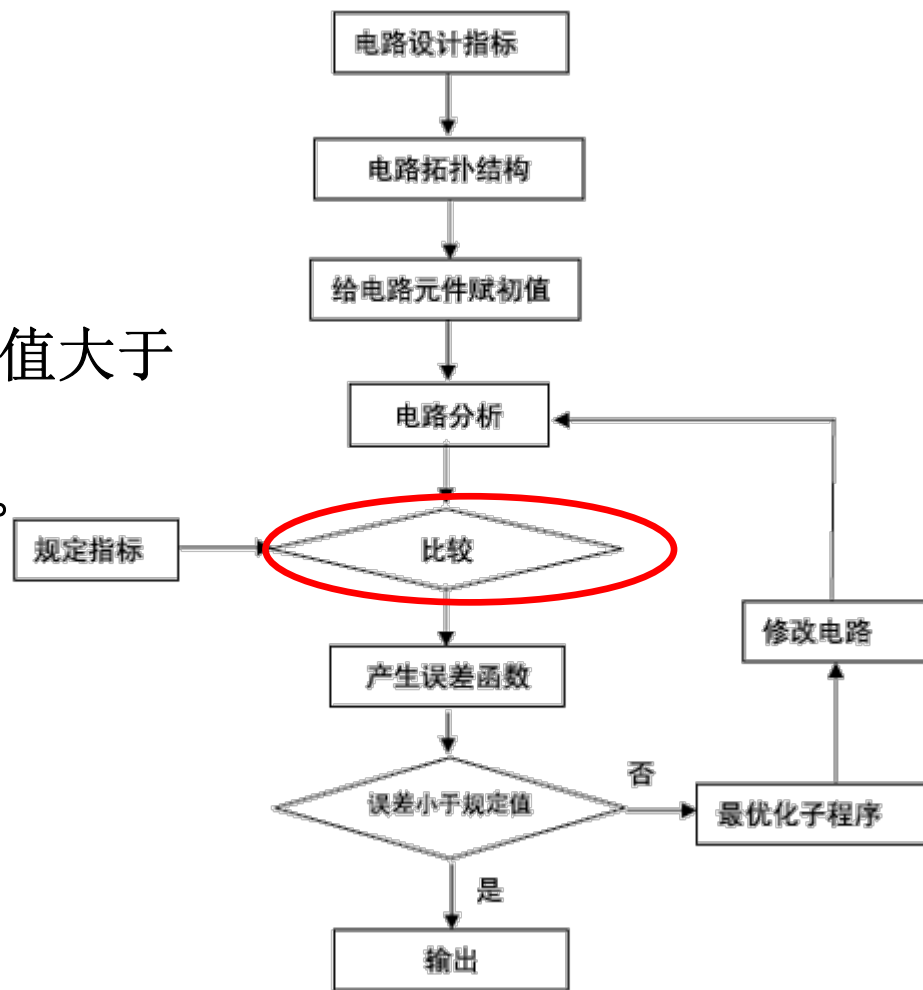


第四章 最优化方法和最优化设计

4、进行比较：

计算出误差函数值，如果误差值大于规定值，则调用最优化子程序。

否则, 设计结束。

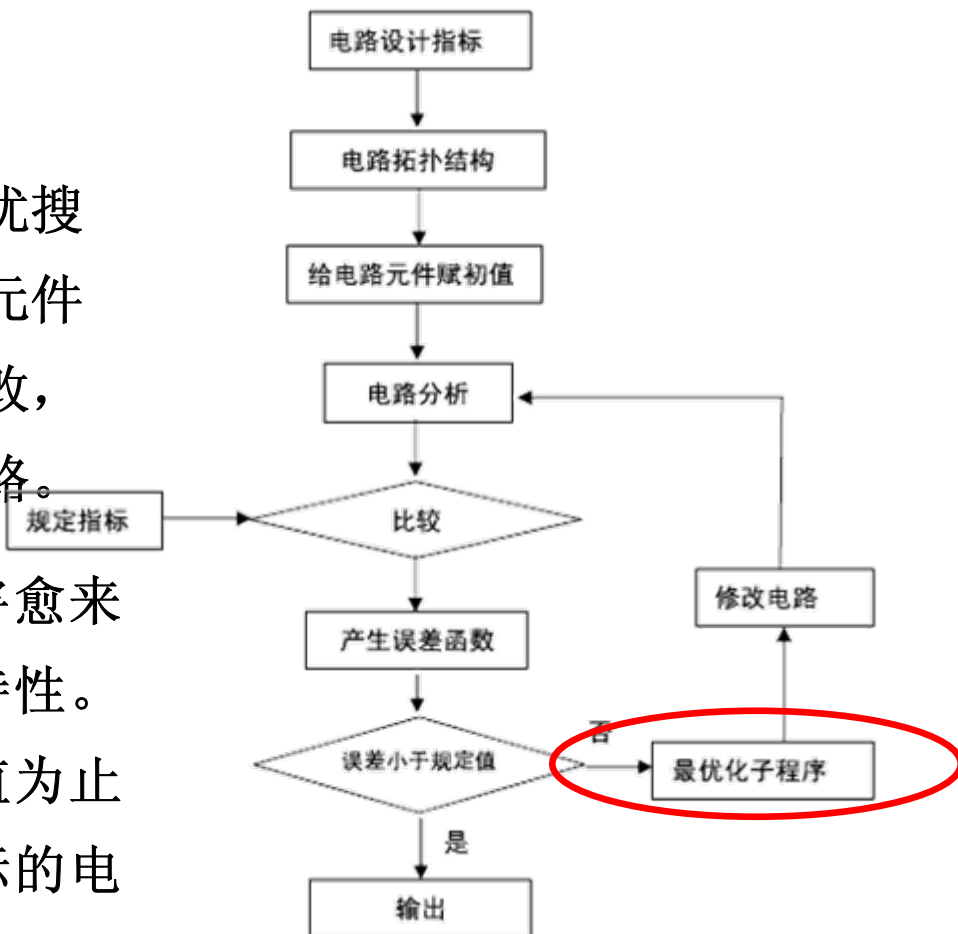


第四章 最优化方法和最优化设计

5、最优化：

最优化子程序按照规定的某种最优搜索方法进行工作，找出使误差减小的元件值，并对原电路元件参数自动进行修改，从而得到比原电路性能有改进的新电路。

如此反复执行上述过程，误差将愈来愈小，电路的特性将不断逼近规定特性。最优搜索一直进行到误差小于规定值为止，此时计算机输出一组满足规定指标的电路元件参数。



§ 4.2 目标函数

一、误差函数

定义：刻画实际电路特性与理想电路特性的差值的函数

$$E(\bar{x}, \omega) = W(\omega) [T(\bar{x}, \omega) - T_i(\omega)]$$

其中：

$\bar{x} = [x_1, x_2, \dots, x_n]^T$: 待优化的参数向量

$T(\bar{x}, \omega)$: 电路特性函数

$T_i(\omega)$: 理想特性函数，即期望的特性值

$W(\omega)$: 频率（或时间）的加权函数



第四章 最优化方法和最优化设计

二、目标函数

目标函数：直接用于进行最优化计算的误差函数

$$\phi(\bar{x}, \omega) = \left[\sum_{k=1}^m W(\omega_k) \left| T(\bar{x}, \omega_k) - T_i(\omega_k) \right|^p \right]^{1/p}$$



最优化方法和最优化设计

目标函数

目标

GOAL

Goal
OptimGoal1
Expr="dB(S(2,1))"
SimInstanceName="SP1"
Min=-2
Max=
Weight=100
RangeVar[1]="freq"
RangeMin[1]=2.99 GHz
RangeMax[1]=3.11 GHz

GOAL

Goal
OptimGoal2
Expr="dB(S(2,1))"
SimInstanceName="SP1"
Min=
Max=-40
Weight=100
RangeVar[1]="freq"
RangeMin[1]=2.7 GHz
RangeMax[1]=2.8 GHz

GOAL

Goal
OptimGoal3
Expr="dB(S(2,1))"
SimInstanceName="SP1"
Min=
Max=-40
Weight=100
RangeVar[1]="freq"
RangeMin[1]=3.3 GHz
RangeMax[1]=3.4 GHz

GOAL

Goal
OptimGoal4
Expr="dB(S(1,1))"
SimInstanceName="SP1"
Min=
Max=-20
Weight=100
RangeVar[1]="freq"
RangeMin[1]=2.99 GHz
RangeMax[1]=3.11 GHz



第四章 最优化方法和最优化设计

二、目标函数

1、 m ：频率采样点数

变化比较平缓的，点数可以少取

变化较剧烈的，就应该多取



二、目标函数

2、p: 目标函数的指数

(1) $p=1$

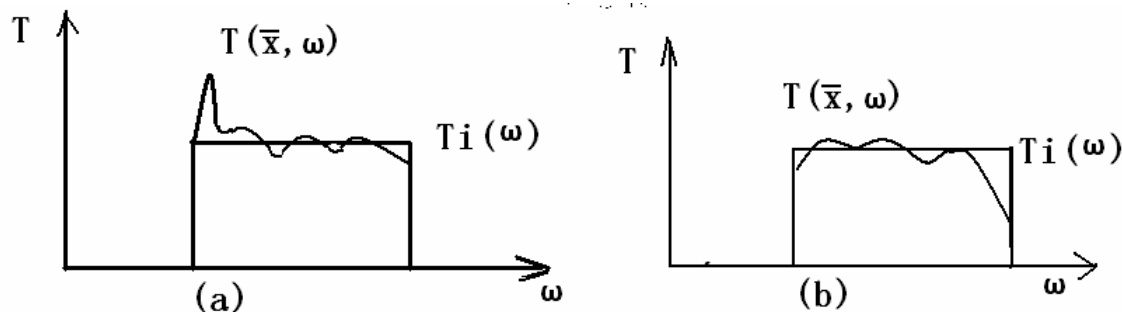
$$\phi(\bar{x}, \omega) = \sum_{k=1}^m W(\omega_k) |T(\bar{x}, \omega_k) - T_i(\omega_k)|$$

优点：优化过程平缓，速度快。

缺点：

a、传输系数可能出现尖峰，难以处理。

b、频带边缘处达不到优化目标。



二、目标函数

2、p: 目标函数的指数

(2) $p \rightarrow \infty$

$$\phi(\bar{x}, \omega) = \max_{k=1, m} \left| T(\bar{x}, \omega_k) - T_i(\omega_k) \right|$$

$\max_{k=1, m}$ 表示在m项中选出最大值的项



二、目标函数

2、p: 目标函数的指数

(3) p适当增大，如p=2，目标函数通常采用平方和的形式

$$\phi(\bar{x}, \omega) = \sum_{k=1}^m \left\{ W(\omega_k) \left[T(\bar{x}, \omega_k) - T_i(\omega_k) \right]^2 \right\}$$

最优化运算进行若干次之后，如果仍有部分频域不能满足要求，可适当加大p值。



二、目标函数

3、 $W(\omega_k)$: 加权函数

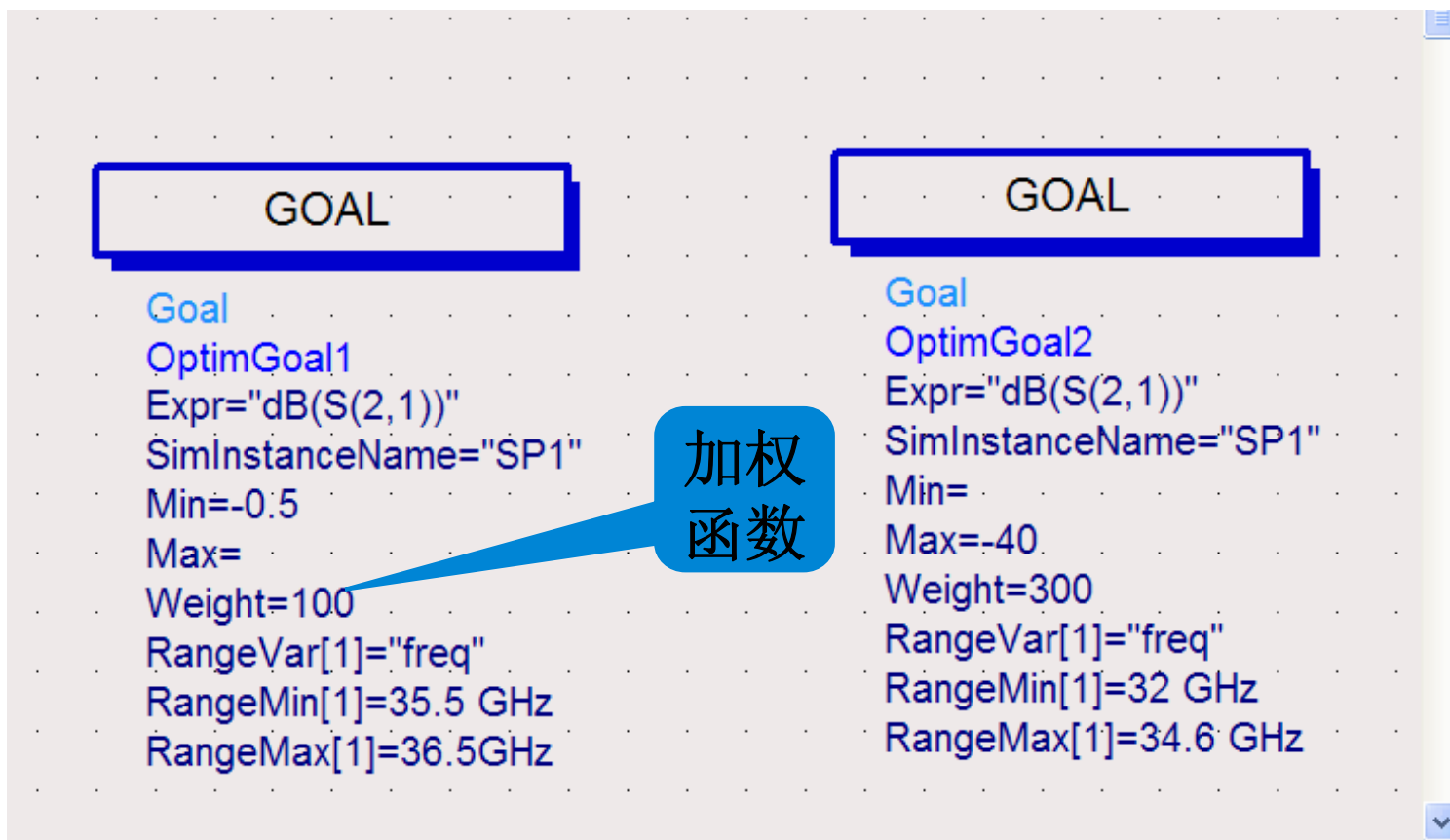
适用情况

- a、用于多个参数进行优化的情况。
- b、同一参数在不同的频段要求不同。



最优化方法和最优化设计

3、加权函数



The screenshot displays two optimization goals in a software interface. A blue callout bubble labeled "加权函数" (Weighted Function) points to the "Weight=100" parameter in the first goal.

Goal 1 (Left):

- Goal
- OptimGoal1
- Expr="dB(S(2,1))"
- SimInstanceName="SP1"
- Min=-0.5
- Max=
- Weight=100
- RangeVar[1]="freq"
- RangeMin[1]=35.5 GHz
- RangeMax[1]=36.5GHz

Goal 2 (Right):

- Goal
- OptimGoal2
- Expr="dB(S(2,1))"
- SimInstanceName="SP1"
- Min=
- Max=-40
- Weight=300
- RangeVar[1]="freq"
- RangeMin[1]=32 GHz
- RangeMax[1]=34.6 GHz



第四章 最优化方法和最优化设计

3、加权函数

选择加权函数的目的是改善设计效果。

在实际的微波电路设计中，因为目标函数所包含的各指标之间有些是互相矛盾或互相制约的，一般很难保证全频域和全部指标都达到最优。借助于选用适当的加权函数，就可以保证重要指标或重要频域内的特性要求。



第四章 最优化方法和最优化设计

三、目标函数极值及全域最小值问题



第四章 最优化方法和最优化设计

三、目标函数极值及全域最小值问题

1、一元函数的极值

充分必要条件：

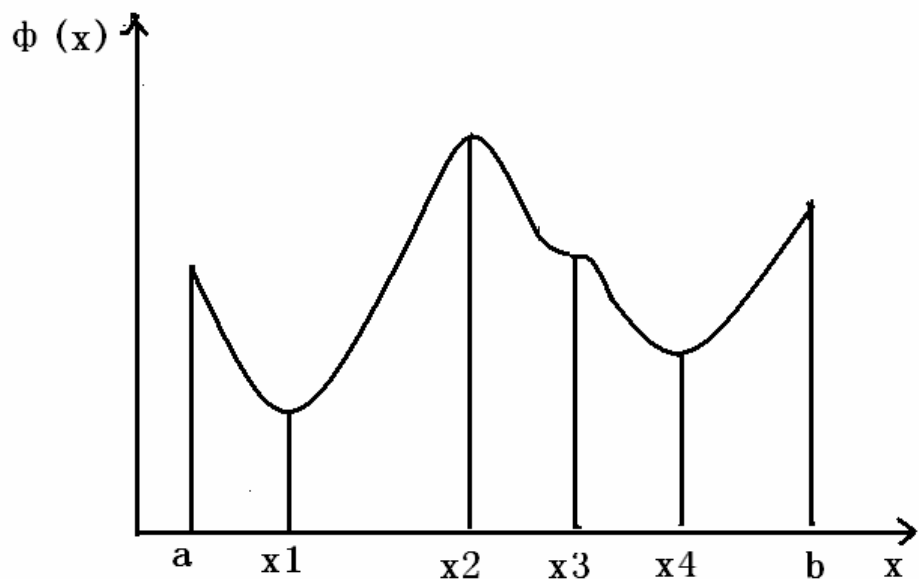
$$\begin{cases} \left. \frac{d\phi(x)}{dx} \right|_{x=x_0} = 0 \\ \left. \frac{d^2\phi(x)}{dx^2} \right|_{x=x_0} > 0 \end{cases}$$



第四章 最优化方法和最优化设计

三、目标函数极值及全域最小值问题

实际函数可能有多个极小值点：



第四章 最优化方法和最优化设计

三、目标函数极值及全域最小值问题

2、多元函数的极值

将多元函数展开成泰勒级数，忽略高次项：

$$\phi(\bar{x}) \approx \phi(\bar{x}_0) + \nabla \phi(\bar{x}_0)^T \Delta \bar{x} + \frac{1}{2} \Delta \bar{x}^T A \Delta \bar{x}$$

其中：

$$\bar{x} = [x_1, x_2, \dots, x_n]^T$$

$$\Delta \bar{x} = \bar{x} - \bar{x}_0$$



第四章 最优化方法和最优化设计

2、多元函数的极值

梯度：

$$\nabla \phi(\bar{x}_0) = \begin{pmatrix} \frac{\partial \phi(\bar{x}_0)}{\partial x_1} \\ \vdots \\ \frac{\partial \phi(\bar{x}_0)}{\partial x_n} \end{pmatrix} = \bar{g}_0$$

汉森矩阵：

$$A = \begin{pmatrix} \frac{\partial^2 \phi(\bar{x}_0)}{\partial x_1^2} & \frac{\partial^2 \phi(\bar{x}_0)}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 \phi(\bar{x}_0)}{\partial x_1 \partial x_n} \\ \frac{\partial^2 \phi(\bar{x}_0)}{\partial x_2 \partial x_1} & \frac{\partial^2 \phi(\bar{x}_0)}{\partial x_2^2} & \cdots & \frac{\partial^2 \phi(\bar{x}_0)}{\partial x_2 \partial x_n} \\ \vdots & \vdots & & \vdots \\ \frac{\partial^2 \phi(\bar{x}_0)}{\partial x_n \partial x_1} & \frac{\partial^2 \phi(\bar{x}_0)}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 \phi(\bar{x}_0)}{\partial x_n^2} \end{pmatrix}$$



第四章 最优化方法和最优化设计

2、多元函数的极值

多元函数极小点的必要条件：

$$\overline{g}_0 = \nabla \phi(\overline{x}_0) = 0$$

多元函数极小点的充分条件：

$$\Delta \overline{x}^T A \Delta \overline{x} > 0$$



包含矩阵**A**的二次型函数



第四章 最优化方法和最优化设计

2、多元函数的极值

多元函数极小点的充分必要条件：

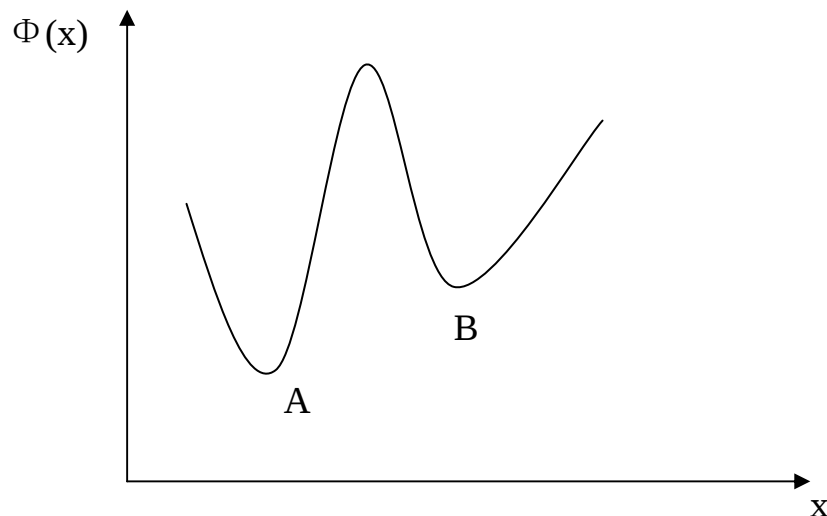
$$\begin{cases} \nabla \phi(\bar{x}_0) = 0 \\ A > 0 \end{cases}$$

即目标函数的梯度为零和二阶偏导数（汉森）矩阵**A**正定



第四章 最优化方法和最优化设计

3、全域最小点



极小值

函数的凸性

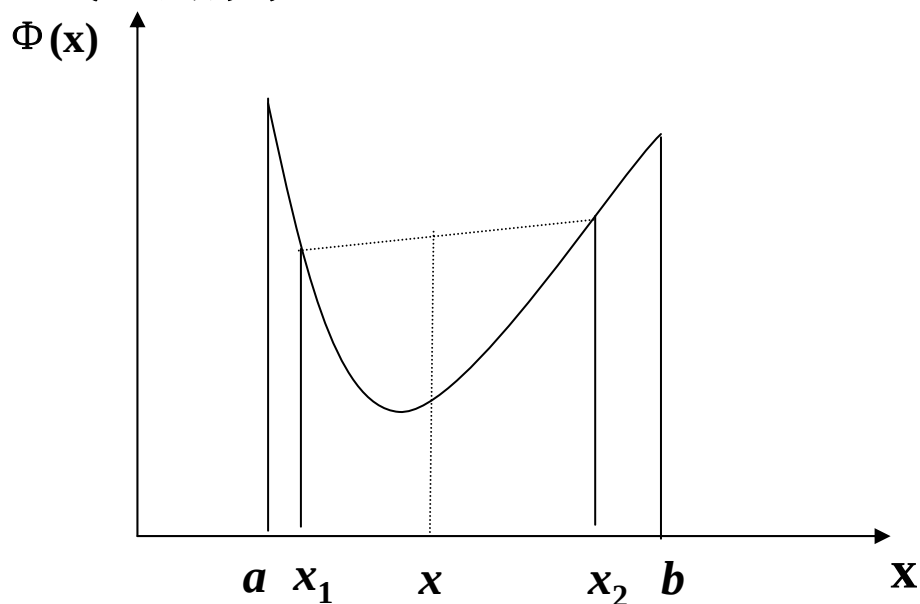
全域
最小值



第四章 最优化方法和最优化设计

凸函数:

函数在取值范围(a,b)内, 任意两个变量点曲线段上的点的函数值小于或等于两点连线上的数值:



$$\phi[ax_1 + (1-a)x_2] \leq a\phi(x_1) + (1-a)\phi(x_2)$$



第四章 最优化方法和最优化设计

(1) 一元凸函数

主要特性：

(a). 函数如果在(a,b)区间为凸函数，则它的极小值就是(a,b)区间最小值

(b). 凸函数判别的充分必要条件是：在(a,b)区间内处处皆存在

$d^2\phi(x)/dx^2 \geq 0$ ，则 $\phi(x)$ 为凸函数。

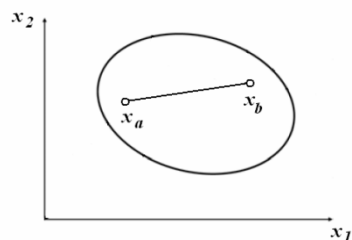


第四章 最优化方法和最优化设计

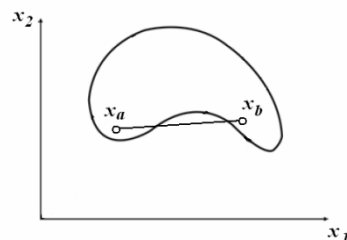
(2) 多元凸函数

可行域： 在多维空间中，变量的取值范围也是个空间，称为可行域。函数凸性的分析仅限于可行域内。

凸集： 设 D 为 n 维欧氏空间的变量可行域，如果 D 内任意两点 x_a 和 x_b 的连线都在可行域内，则此域称为凸集。



凸集



非凸集



第四章 最优化方法和最优化设计

(2) 多元凸函数

主要特性：

(a). 如 $\phi[a\bar{x}_1 + (1-a)\bar{x}_2] \leq a\phi(\bar{x}_1) + (1-a)\phi(\bar{x}_2) \quad (0 \leq a \leq 1)$

则 $\phi(\bar{x})$ 是 $\mathbf{D}$ 上的凸函数

(b). 凸函数的极小值就是全域最小值

(c). 多元凸函数判别方法：若 $\phi(\bar{x})$ 在 $\mathbf{D}$ 上有连续二阶偏导数，而且在

$\mathbf{D}$ 上处处满足： $\bar{x}^T \mathbf{A} \bar{x} \geq 0$ ，即矩阵 $\mathbf{A}$ 处处半正定，则 $\phi(\bar{x})$ 是

凸函数。

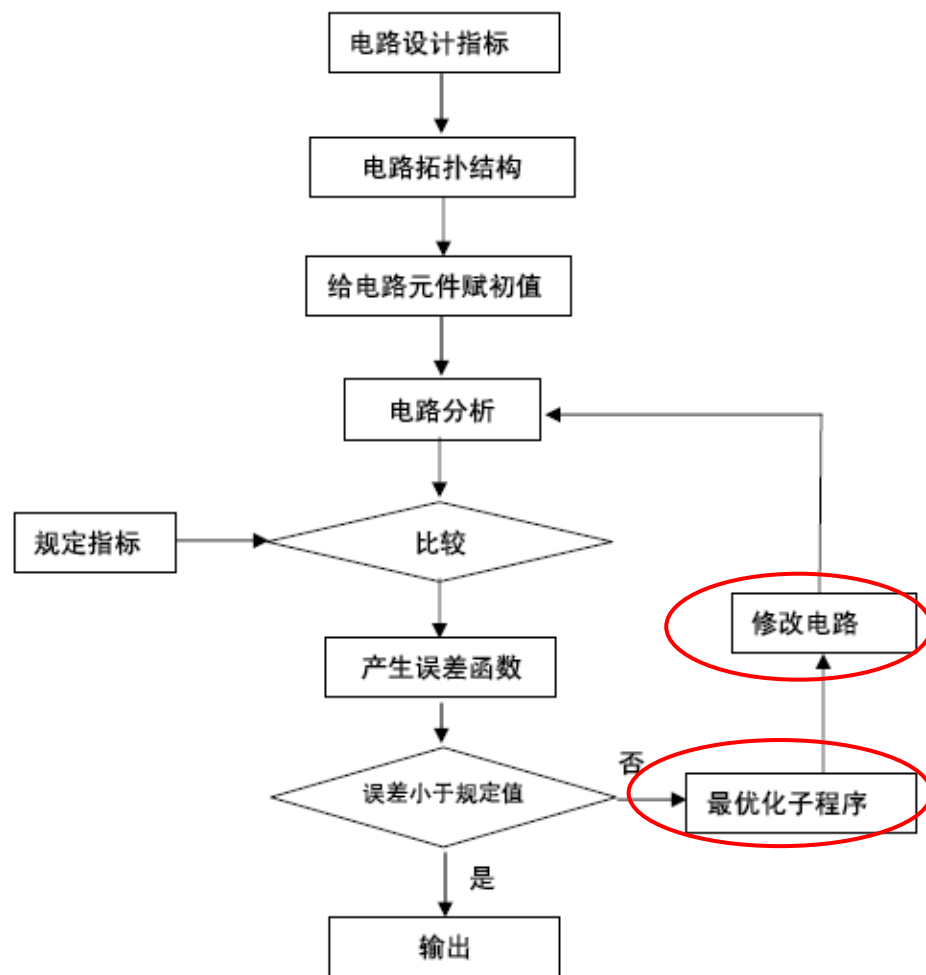


第四章 最优化方法和最优化设计

§ 4.3 最优化方法概述

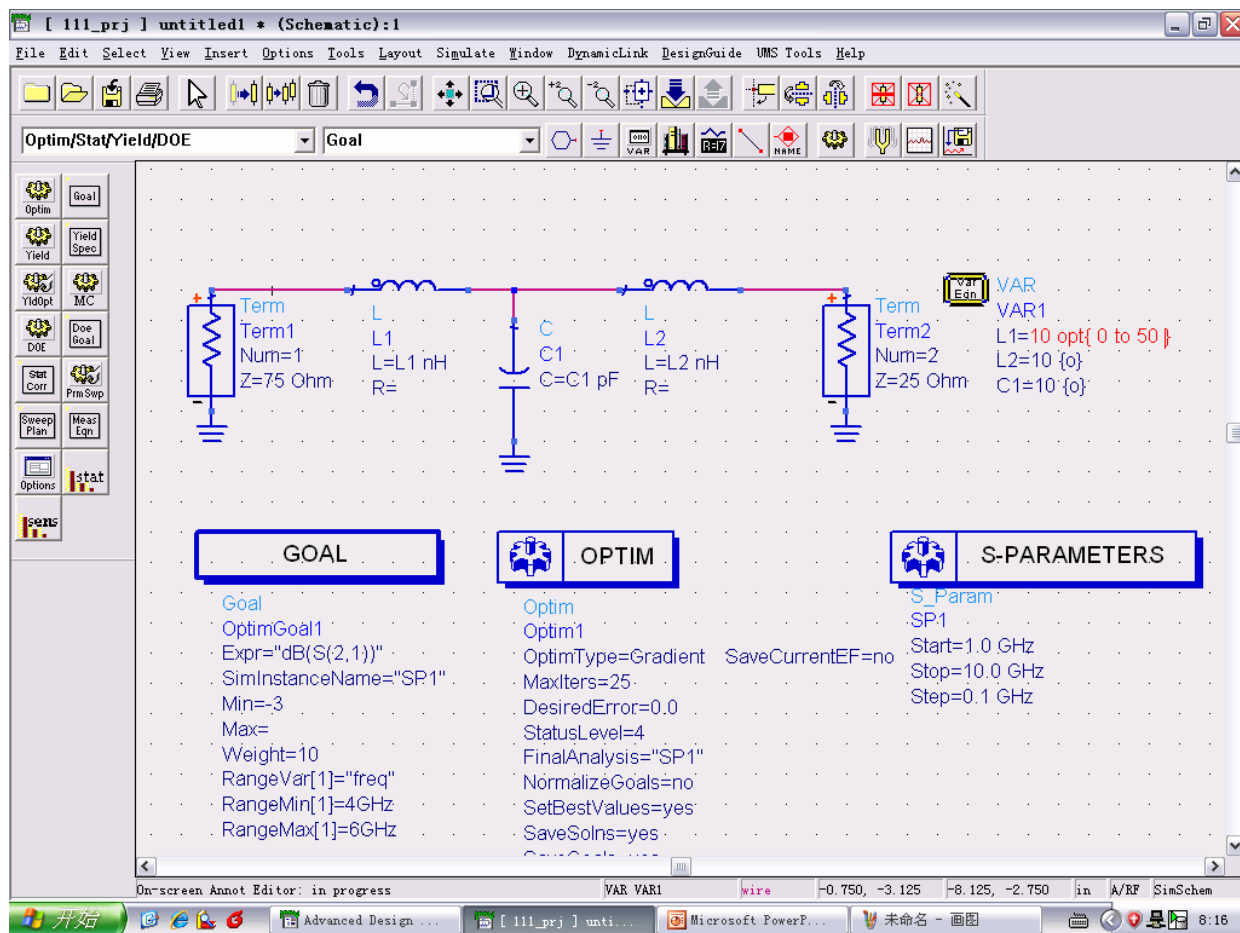
最优化设计过程实质

反复利用最优化方法修改电路使电路特性不断改进，最后达到预定特性的过程。



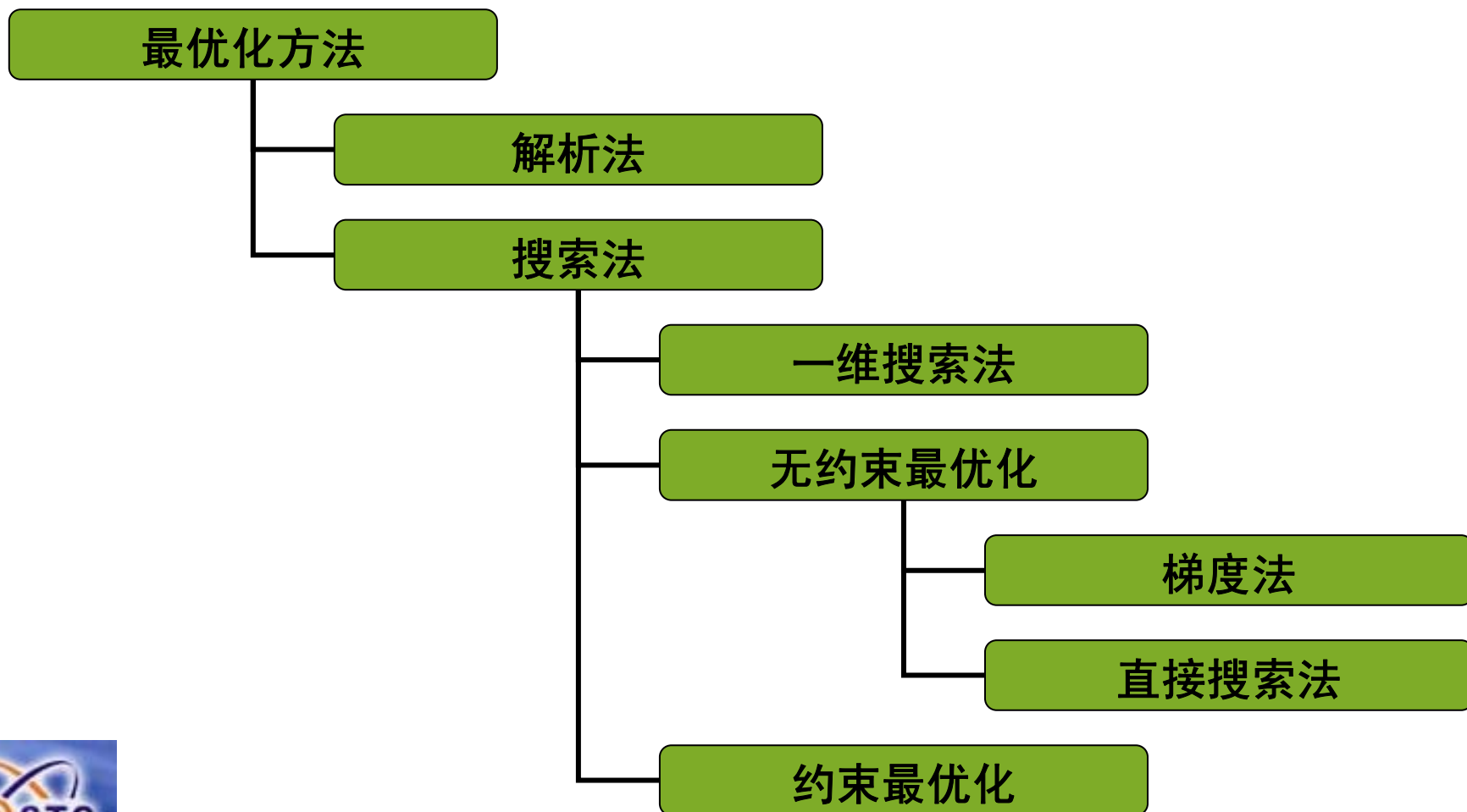
第四章 最优化方法和最优化设计

❖ 微波电路待优化参数：集总元件参数；分布元件的尺寸和阻抗等。



第四章 最优化方法和最优化设计

§ 4.3 最优化方法概述



第四章 最优化方法和最优化设计

§ 4.3 最优化方法概述

解析法：

即导数求极值，它适合于简单函数和具有明确数学表达式的函数。但是复杂电路基本上都没有解析表达式，所以此方法局限性很大。

$$\left. \frac{d\phi(x)}{dx} \right|_{x=x_0} = 0$$

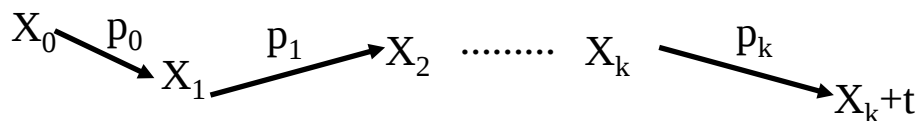


第四章 最优化方法和最优化设计

§ 4.3 最优化方法概述

一维搜索法：

最优化方法通常由一系列一维搜索组成，所以一维搜索法是实现许多最优化方法的一种基本方法。



第四章 最优化方法和最优化设计

§ 4.3 最优化方法概述

一维搜索法：

一般来说，从点 $\mathbf{x}_k$ 出发，沿规定搜索方向 $\mathbf{p}_k$ 求目标函数的极小点 $\mathbf{x}_{k+1}$ 便构成某些最优化方法的一种基本过程。此时目标函数变成了参变量 t 的一元函数 $U(\mathbf{x}_k + t \mathbf{p}_k)$ ，这种求 t 使 $U(\mathbf{x}_k + t \mathbf{p}_k)$ 为极小的过程，称为一维搜索。



第四章 最优化方法和最优化设计

§ 4.3 最优化方法概述

梯度法:

求出电路特性对元件值变化的梯度，以误差函数梯度下降方向修改元件值

直接方法:

少量增减元件值，比较误差值，找出误差下降最有效的方向



第四章 最优化方法和最优化设计

§ 4.3 最优化方法概述

梯度法和直接方法的比较：

梯度方法：

在极小点附近收敛较快(牛顿法)，但梯度计算比较困难，特别是要用数值微分的地方；此外，要用到矩阵代数，程序编写比较复杂。

直接方法：

可用于不存在或不知道梯度的场合，因此适应性较广，且程序编写比较方便。其缺点是在极小点附近收敛较慢。



第四章 最优化方法和最优化设计

§ 4.3 最优化方法概述

约束最优化问题：

实际电路中，待优化变量的范围必须加以限制，因此实际的微波电路优化设计一般总是一个有约束的优化问题。

实际
电路优化
设计问题

对待优化
变量范围
加以限制

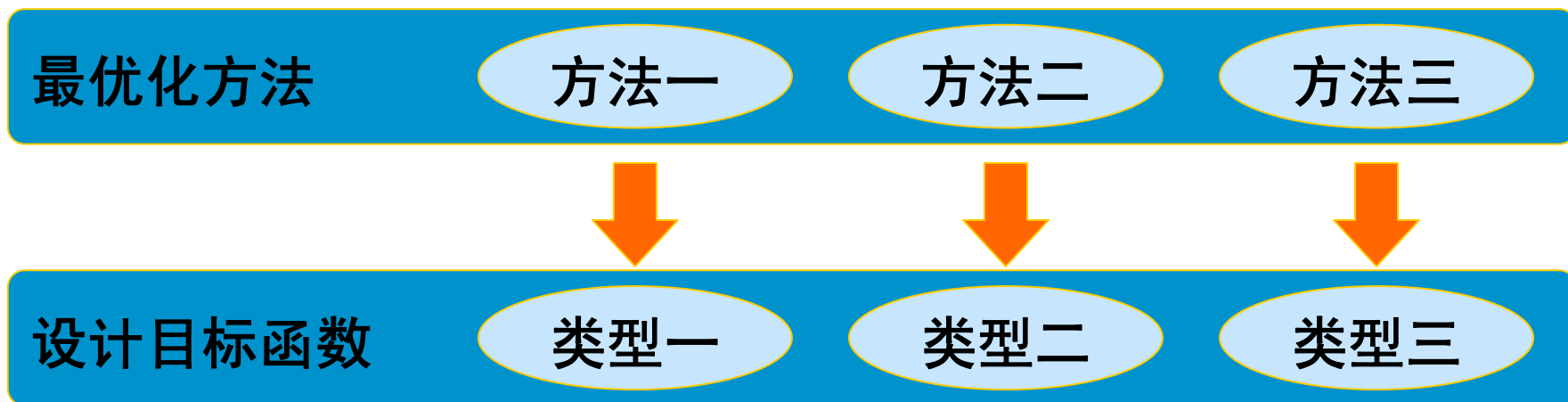
约束
最优化
问题

无约束
最优化
问题



第四章 最优化方法和最优化设计

❖ 最优化方法的选择



有效的设计方法：选用多种优化方法, 建议优化开始阶段选用直接法或随机法，其后选用梯度法。



第四章 最优化方法和最优化设计

一、区间消去法的基本原理

基本思想：

在搜索过程中不断地缩短最小点存在的区间，直至最小点存在的区间达到允许的误差范围为止。



第四章 最优化方法和最优化设计

一、区间消去法的基本原理

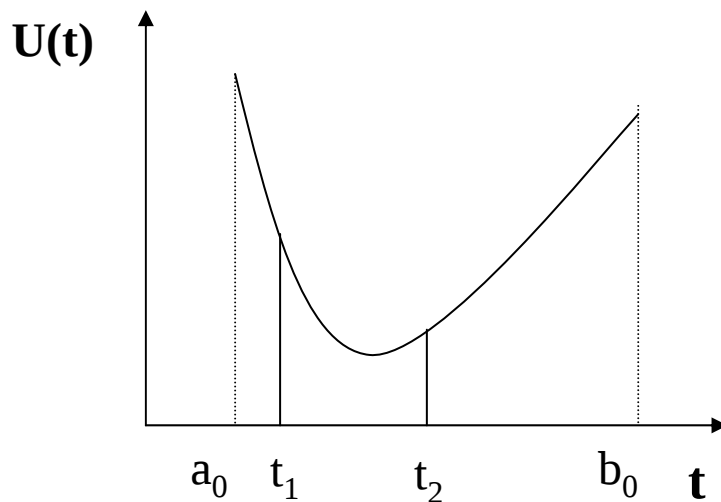
一般做法：

设一元函数 $U(t)$ 在区间 $[a_0, b_0]$ 是一凸函数，先在搜索区间 $[a_0, b_0]$ 内选取两点 t_1 和 t_2 ，并计算其函数值 $U(t_1)$ 和 $U(t_2)$ 。根据函数值的大小，可分为下列三种情况



第四章 最优化方法和最优化设计

一、区间消去法的基本原理



(1) $U(t_1) > U(t_2)$

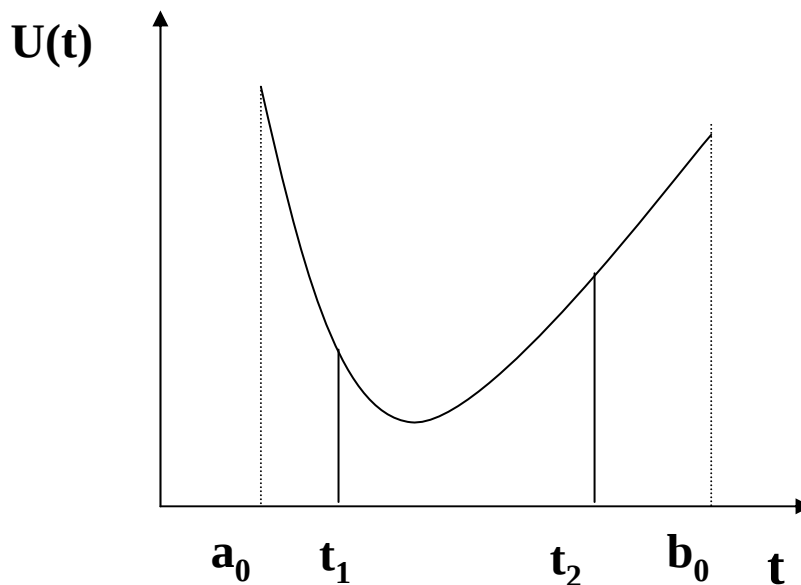
(1)

极小点必然在区间 $[t_1, b_0]$ 内，因此可将区间 $[a_0, t_1]$ 消去，使搜索区间由 $[a_0, b_0]$ 缩短为 $[a_1, b_1]$ ，其中 $a_1 = t_1$ ， $b_1 = b_0$ 。



第四章 最优化方法和最优化设计

一、区间消去法的基本原理



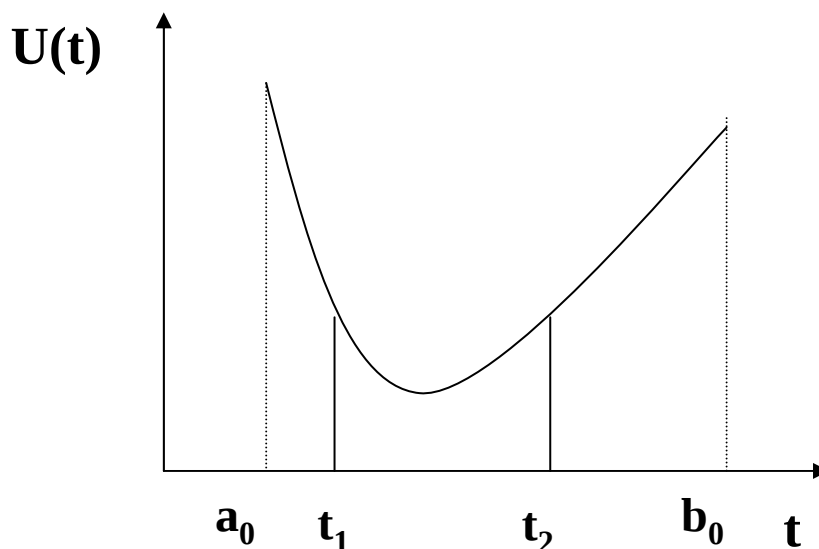
(2) $U(t_1) < U(t_2)$

极小点必然在区间 $[a_0, t_2]$ 内，因此可将区间 $[t_2, b_0]$ 消去，使搜索区间由 $[a_0, b_0]$ 缩短为 $[a_1, b_1]$ ，其中 $a_1 = a_0$ ， $b_1 = t_2$



第四章 最优化方法和最优化设计

一、区间消去法的基本原理：



(3) $U(t_1)=U(t_2)$

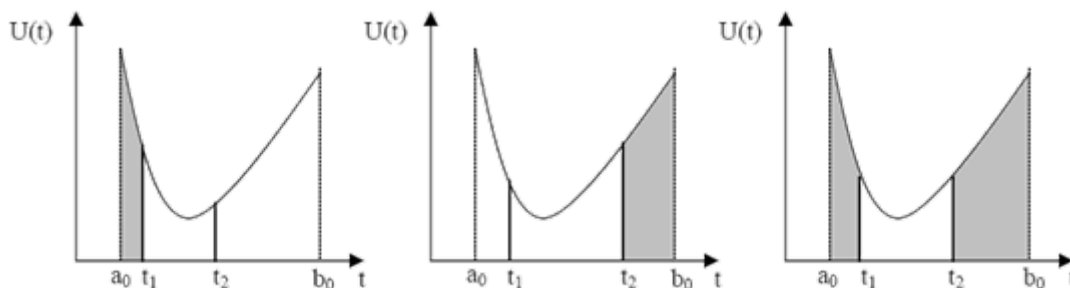
极小点在区间 $[t_1, t_2]$ 内，因此可将两端区间 $[a_0, t_1]$ 和 $[t_2, b_0]$ 消去，使搜索区间由 $[a_0, b_0]$ 缩短为 $[t_1, t_2]$ 。但是，为了在缩短后的区间保留一个选点，以便与下一次选点作比较，通常取 $a_1=a_0$ ， $b_1=t_2$ 或 $a_1=t_1$ ， $b_1=b_0$



第四章 最优化方法和最优化设计

一、区间消去法的基本原理

基本原理：



只要在区间 $[a_0, b_0]$ 内取两点，算出它们的函数值并加以比较，就可将原搜索区间 $[a_0, b_0]$ 缩短为 $[a_1, b_1]$ 。然后再在 $[a_1, b_1]$ 内选取一点，算出其函数值并与该区间的保留值（ t_1 或 t_2 ）的函数值进行比较，按上述方法再消去一段区间，使搜索区间缩短为 $[a_2, b_2]$ 。如此继续下去，经若干次选点和函数值计算，总可找出一个包含极小点的区间，使其长度小于规定范围 δ 。



第四章 最优化方法和最优化设计

一、区间消去法的基本原理

总缩短率：

$$\xi = \frac{(b_{n-1} - a_{n-1})}{(b_0 - a_0)} = \eta_1 \eta_2 \cdots \eta_{n-1}$$

$\eta_1, \eta_2, \cdots, \eta_{n-1}$ ：分别是各次的缩短率

$(b_{n-1} - a_{n-1})$ ：经(n-1)次缩短后的区间长度



第四章 最优化方法和最优化设计

二、菲波那西(Fibonacci)法

菲波那西（**Fibonacci**）数列：

$$\begin{cases} F_0 = F_1 = 1 \\ F_n = F_{n-1} + F_{n-2} \quad n \geq 2 \end{cases}$$



n	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
F _n	1	1	2	3	5	8	13	21	34	55	89	144	233	377	610

在搜索中每次的缩短率等于**Fibonacci**数列中相邻的两数之比。



第四章 最优化方法和最优化设计

二、菲波那西 (Fibonacci) 法

运算步骤:

1、按照下列公式确定 t_1 和 t'_1 , 计算函数值 $U(t_1)$ 和 $U(t'_1)$

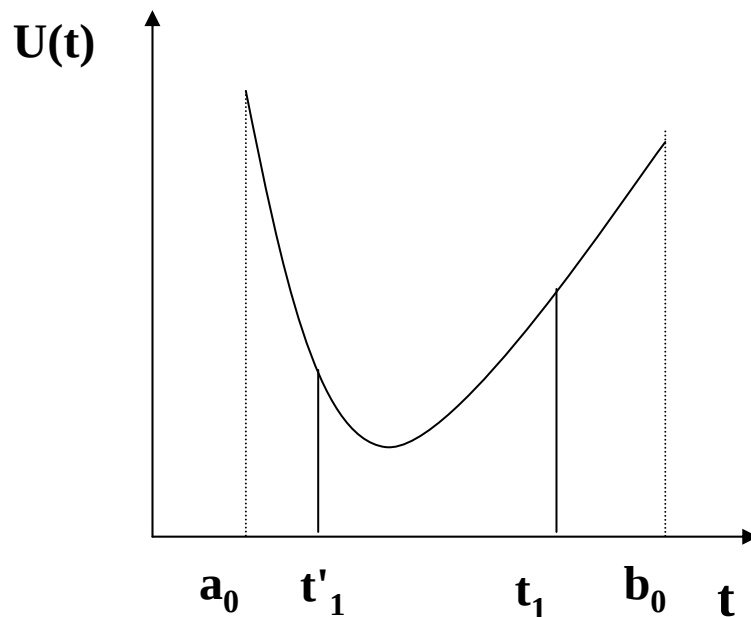
$$\begin{cases} t_1 = a_0 + \frac{F_{n-1}}{F_n}(b_0 - a_0) \\ t'_1 = b_0 - \frac{F_{n-1}}{F_n}(b_0 - a_0) \end{cases}$$



第四章 最优化方法和最优化设计

运算步骤:

2、比较函数值:



如 $U(t_1) \geq U(t'_1)$

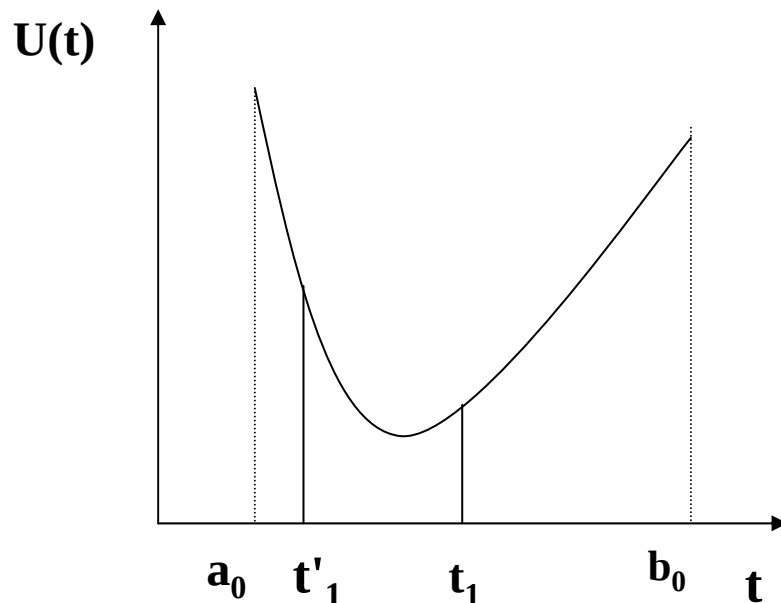
$$a_1 = a_0, b_1 = t_1, t_2 = t'_1, \quad t'_2 = b_1 - \frac{F_{n-2}}{F_{n-1}}(b_1 - a_1)$$



第四章 最优化方法和最优化设计

运算步骤:

2、比较函数值:



如 $U(t_1) \leq U(t'_1)$

$$a_1 = t'_1, b_1 = b_0, t'_2 = t_1, \quad t_2 = a_1 + \frac{F_{n-2}}{F_{n-1}}(b_1 - a_1)$$



第四章 最优化方法和最优化设计

运算步骤:

3、计算新的函数值 $U(t_2)$ 和 $U(t'_2)$ ，然后比较大小，过程同前。一直到(n-1)次得:

$$\begin{cases} t_{n-1} = a_{n-2} + \frac{1}{2}(b_{n-2} - a_{n-2}) \\ t'_{n-1} = b_{n-2} - \frac{1}{2}(b_{n-2} - a_{n-2}) \end{cases} \rightarrow \begin{cases} t_{n-1} = a_{n-2} + \left(\frac{1}{2} + \delta\right)(b_{n-2} - a_{n-2}) \\ t'_{n-1} = b_{n-2} - \frac{1}{2}(b_{n-2} - a_{n-2}) \end{cases}$$



第四章 最优化方法和最优化设计

运算步骤:

4、计算函数值 $U(t_{n-1})$ 和 $U(t'_{n-1})$ ，比较其大小，确定最后区间 $[a_{n-1}, b_{n-1}]$

如 $U(a_{n-1}) < U(b_{n-1})$ ，则 a_{n-1} 为极小点

否则， b_{n-1} 为极小点



第四章 最优化方法和最优化设计

缩短率:

每次的缩短率

$$\eta_1 = \frac{F_{n-1}}{F_n}, \eta_2 = \frac{F_{n-2}}{F_{n-1}}, \dots, \eta_{n-1} = \frac{F_1}{F_2}$$

总的缩短率

$$\xi = \eta_1 \eta_2 \cdots \eta_{n-1} = \frac{1}{F_n}$$

例如, 要求目标函数的精度是**0.001**, 即 $\frac{1}{F_n} = \eta_0 = 0.001$,

由 $\frac{1}{F_{15}} = \frac{1}{987} \approx 0.001$, 得**n=15**



第四章 最优化方法和最优化设计

三、黄金分割(0.618) 法

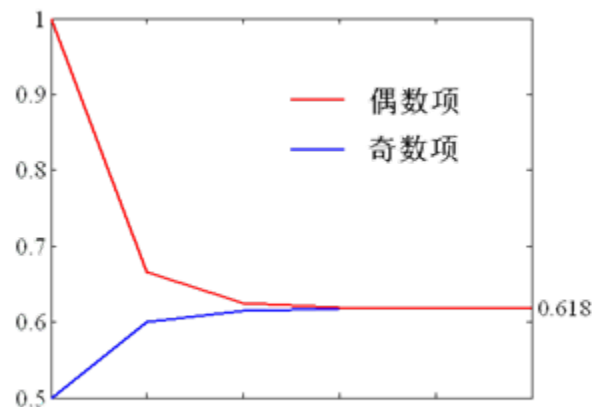
每次缩短率为**0.618**的区间消去法即为黄金分割法

黄金分割率:

当把Fibonacci数列 $\{F_{i-1}/F_i\}$ 分成奇数项及偶数项时, 可形成奇数项数

列 $\{F_{2k-1}/F_{2k}\}$ 和偶数项数列 $\{F_{2k}/F_{2k+1}\}$, 前者单调递增, 后者单调递减,

且收敛于同一极限。这个极限就是**0.618...**, 称为黄金分割率



第四章 最优化方法和最优化设计

三、黄金分割(0.618) 法

每次缩短率为**0.618**

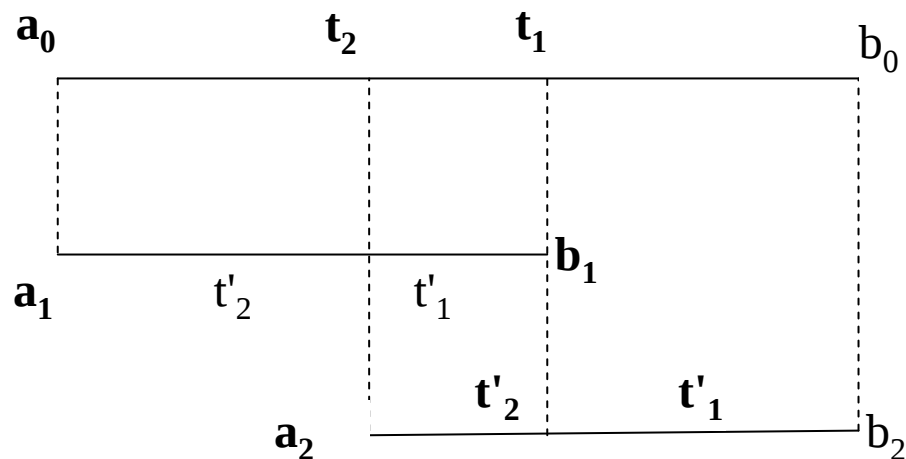
总缩短率:

$$\xi = \eta^{n-1} = 0.618^{n-1}$$



第四章 最优化方法和最优化设计

计算步骤



(1)、先取两个试点 t_1 , t_2 , 分别为

$$t_1 = a_0 + \eta(b_0 - a_0)$$

$$t_2 = b_0 - \eta(b_0 - a_0)$$



第四章 最优化方法和最优化设计

计算步骤

(2)、计算其函数值 $U(t_1)$ 与 $U(t_2)$ ，并进行比较。

如 $U(t_1) \geq U(t_2)$ ，则

$$a_1 = a_0, b_1 = t_1, t'_1 = t_2 \quad t'_2 = b_1 - \eta(b_1 - a_1)$$



第四章 最优化方法和最优化设计

计算步骤

(2)、计算其函数值 $U(t_1)$ 与 $U(t_2)$ ，并进行比较。

如 $U(t_1) \leq U(t_2)$ ，则

$$a_1 = t_2, b_1 = b_0, t'_2 = t_1 \quad t'_1 = a_1 + \eta(b_1 - a_1)$$

采用相似的步骤缩短区间.....



第四章 最优化方法和最优化设计

计算步骤

(3)、每缩短一次区间，判断一下是否满足

$$\left| \frac{b_k - a_k}{b_0 - a_0} \right| \leq \varepsilon$$

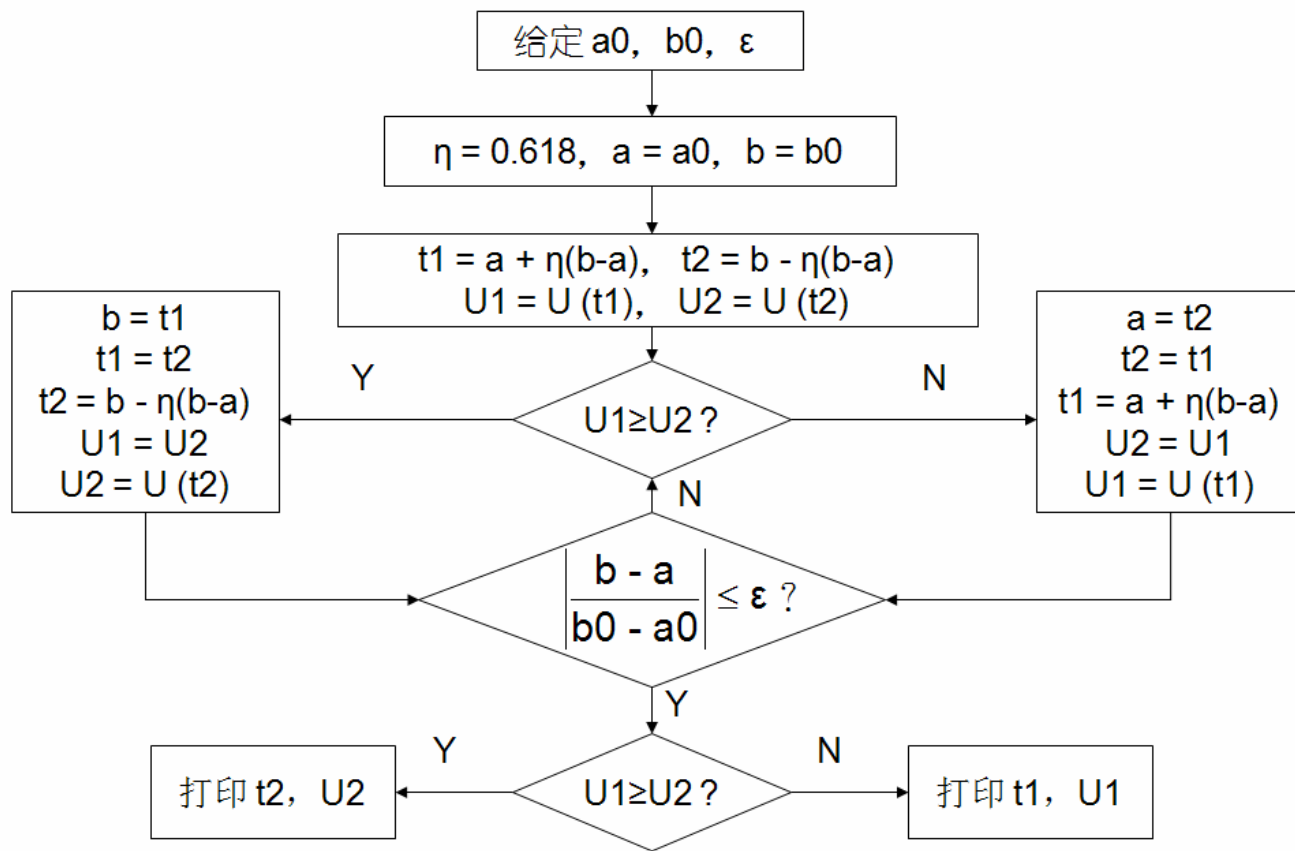
若上式满足，则停止选取试点；若上式不满足，则继续选取试点

(4)、比较最后求得的两函数值，确定最小函数值和极小点。



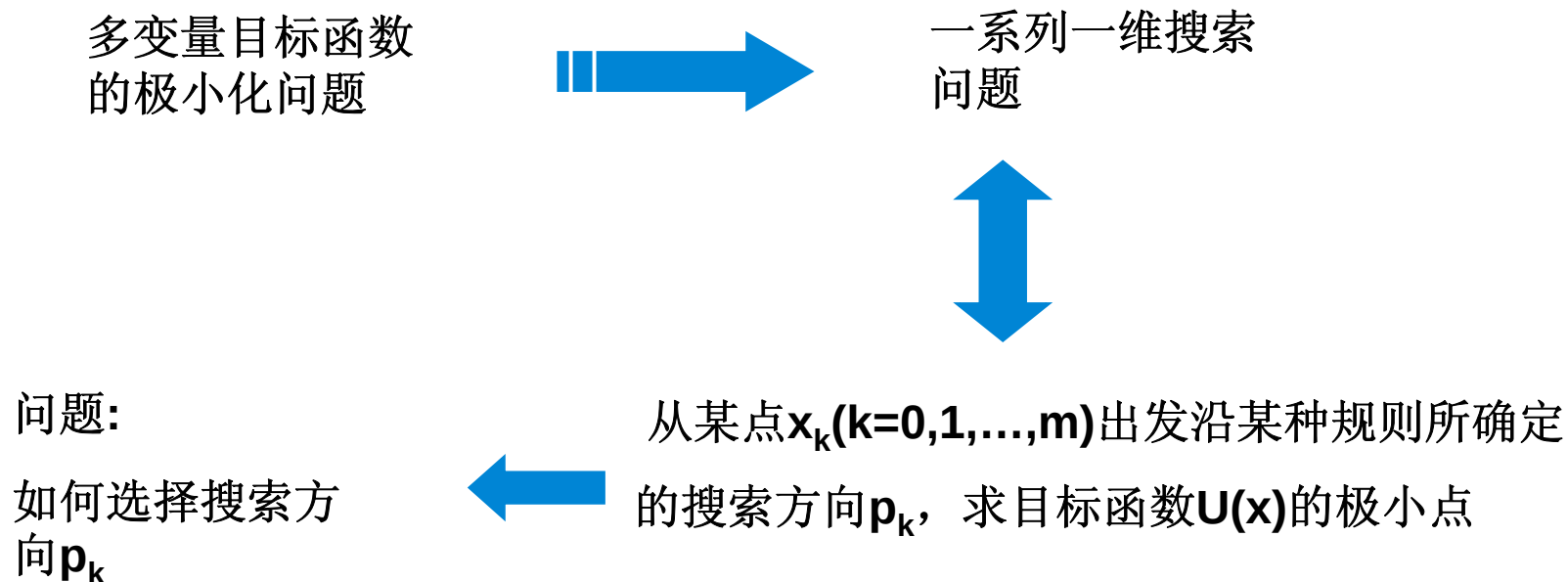
第四章 最优化方法和最优化设计

黄金分割法框图



第四章 最优化方法和最优化设计

4.5 无约束最优化的梯度方法（多维）



第四章 最优化方法和最优化设计

4.5 无约束最优化的梯度方法（多维）

一、最速下降法

函数在 $\mathbf{x}_k$ 的梯度方向垂直于经该点的等值线，是该点函数值的最速上升方向，而与其相反的方向，即负梯度方向，表示函数在该点的最速下降方向。一个很自然的想法是，为了使目标得到较快的下降，搜索应沿着**负梯度方向**进行。



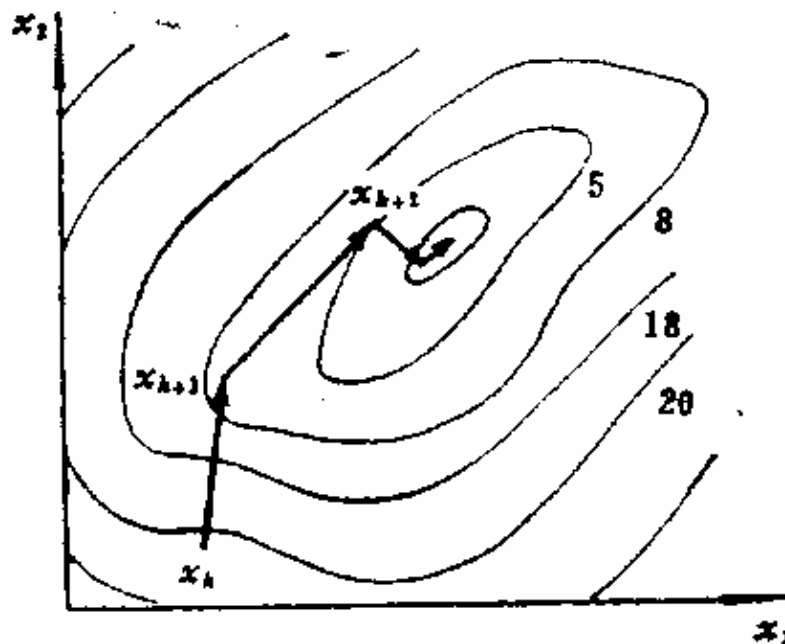
第四章 最优化方法和最优化设计

4.5 无约束最优化的梯度方法（多维）

一、最速下降法

以负梯度方向为搜索方向的方法

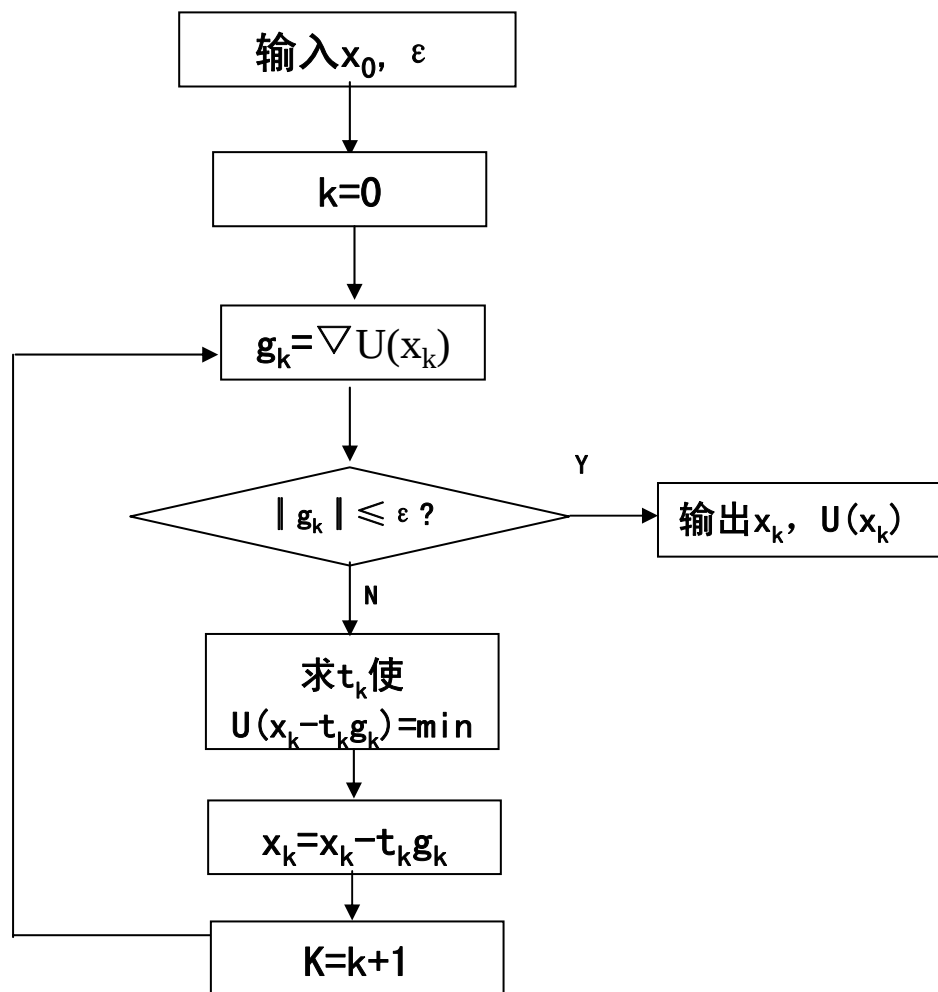
$$p_k = -\nabla U(x_k)$$



第四章 最优化方法和最优化设计

一、最速下降法

计算框图



第四章 最优化方法和最优化设计

一、最速下降法

计算步骤

- 1、选定初值 $\mathbf{x}_0$ 和判敛参数 ε ，并计算目标函数在 $\mathbf{x}_0$ 的梯度 $\nabla U(\mathbf{x}_0)$
- 2、取其负梯度方向为搜索方向，并在该方向用一维搜索法求出极小点 $\mathbf{x}_1$ ；



第四章 最优化方法和最优化设计

一、最速下降法

计算步骤

- 3、再计算目标函数在 $\mathbf{x}_1$ 的梯度，判断其范数（可以理解为函数的函数，一般取模既可）是否小于规定误差 ε 。如果满足 $\|\nabla U(x_1)\| \leq \varepsilon$ ，则 $\mathbf{x}_1$ 为极小点，否则取 $p_1 = -\nabla U(x_1)$ 为新的搜索方向，并在该方向上求目标函数的极小点 $\mathbf{x}_2$ ，再判断该点的梯度范数是否小于 ε ；
- 4、这个过程继续下去，直到满足 $\|\nabla U(x_k)\| \leq \varepsilon$ 为止。



第四章 最优化方法和最优化设计

一、最速下降法

例1:

给定目标函数

$$U(\bar{x}) = U(x_1, x_2) = x_1^2 + (x_2 - 4)^2$$

求 $U(\bar{x})$ 的极小值及极小值变量 $\bar{x}_{\min}$ ，设初值为 $x_0 = [1, 1]^T$



第四章 最优化方法和最优化设计

一、最速下降法

1、计算目标函数在 $\bar{x}_0$ 点的梯度

$$\nabla U(\bar{x}) = \begin{pmatrix} \frac{\partial U(\bar{x})}{\partial x_1} \\ \frac{\partial U(\bar{x})}{\partial x_2} \end{pmatrix} = \begin{pmatrix} 2x_1 \\ 2(x_2 - 4) \end{pmatrix}$$



第四章 最优化方法和最优化设计

一、最速下降法

2、确定寻查方向

$$\bar{p}_0 = -\nabla U(x_{10}, x_{20}) = \begin{pmatrix} -2 \\ 6 \end{pmatrix}$$

3、沿 $\bar{p}_0$ 方向进行寻查。变量：

$$\bar{x}_1 = \bar{x}_0 + t\bar{p}_0 = \begin{pmatrix} 1 \\ 1 \end{pmatrix} + t \begin{pmatrix} -2 \\ 6 \end{pmatrix} = \begin{pmatrix} 1-2t \\ 1+6t \end{pmatrix}$$

目标函数：

$$U(t) = U(\bar{x}_1) = 40t^2 - 40t + 10$$



第四章 最优化方法和最优化设计

一、最速下降法

4、用单变量寻查法求 $u(t)=\min$

$$\text{令: } \frac{\partial U(t)}{\partial t} = 0 \quad \longrightarrow \quad 80t_0 - 40 = 0 \quad \longrightarrow \quad t_0 = 0.5$$

所以 $\bar{p}_0$ 方向的极小点:

$$\bar{x}_1 = \begin{pmatrix} x_{11} \\ x_{21} \end{pmatrix} = \begin{pmatrix} 0 \\ 4 \end{pmatrix}$$



第四章 最优化方法和最优化设计

一、最速下降法

5、收敛判断。

$$\nabla U(\bar{x}_1) = \begin{pmatrix} 2x_1 \\ 2x_2 - 8 \end{pmatrix} = \begin{pmatrix} 2 \times (0) \\ 2 \times 4 - 8 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

梯度的模：

$$|\nabla U(\bar{x}_1)| = \sqrt{x_{11}^2 + x_{22}^2} = 0$$

上例中函数的等高线是一族同心圆，其负梯度方向指向极小点，所以沿负梯度方向搜索只需迭代一次就能找出极小点。



第四章 最优化方法和最优化设计

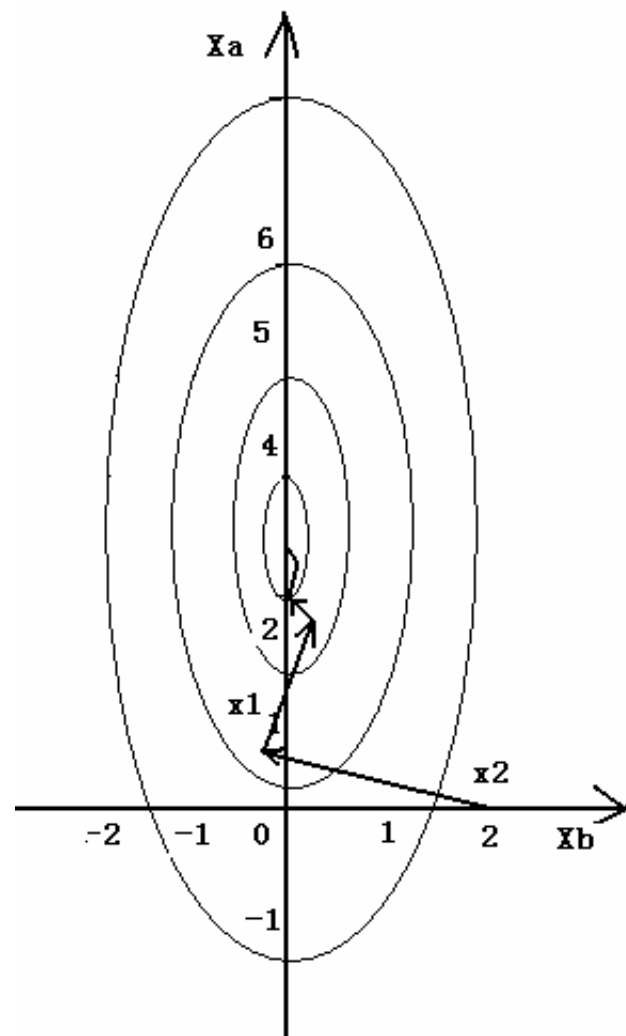
一、最速下降法

例2:

给定目标函数

$$U(\bar{x}) = U(x_1, x_2) = 5x_1^2 + x_2^2 - 6x_2 + 9$$

求 $U(\bar{x})$ 的极小值及极小值变量 $\bar{x}_{\min}$



第四章 最优化方法和最优化设计

一、最速下降法

1、任选初始值 $\bar{x}_0 = [x_{10} \quad x_{20}]^T = [2 \quad 0]^T$

给定判敛误差 $\varepsilon = 0.001$

2、计算目标函数在 $\bar{x}_0$ 点的梯度

$$\nabla U(\bar{x}) = \begin{pmatrix} \frac{\partial U(\bar{x})}{\partial x_1} \\ \frac{\partial U(\bar{x})}{\partial x_2} \end{pmatrix} = \begin{pmatrix} 10x_1 \\ 2x_2 - 6 \end{pmatrix}$$



第四章 最优化方法和最优化设计

一、最速下降法

3、确定寻查方向

$$\bar{p}_0 = -\nabla U(x_{10}, x_{20}) = \begin{pmatrix} -20 \\ 6 \end{pmatrix}$$

4、沿 $\bar{p}_0$ 方向进行寻查。变量：

$$\bar{x}_1 = \bar{x}_0 + t\bar{p}_0 = \begin{pmatrix} 2 \\ 0 \end{pmatrix} + t \begin{pmatrix} -20 \\ 6 \end{pmatrix} = \begin{pmatrix} 2 - 20t \\ 6t \end{pmatrix}$$

目标函数：

$$\begin{aligned} U(t) &= U(\bar{x}_1) = 5(2 - 20t)^2 + (6t)^2 - 6(6t) + 9 \\ &= 2036t^2 - 436t + 29 \end{aligned}$$



第四章 最优化方法和最优化设计

一、最速下降法

5、用单变量寻查法求 $u(t)=\min$

令: $\frac{\partial U(t)}{\partial t} = 0 \quad \longrightarrow \quad 4072t_0 - 436 = 0 \quad \longrightarrow \quad t_0 = 0.1071$

所以 $\bar{p}_0$ 方向的极小点:

$$\bar{x}_1 = \begin{pmatrix} x_{11} \\ x_{21} \end{pmatrix} = \begin{pmatrix} 2 - 20 \times 0.1071 \\ 6 \times 0.1071 \end{pmatrix} = \begin{pmatrix} -0.142 \\ 0.6426 \end{pmatrix}$$



第四章 最优化方法和最优化设计

一、最速下降法

6、收敛判断。

$$\nabla U(\bar{x}_1) = \begin{pmatrix} 10x_1 \\ 2x_2 - 6 \end{pmatrix} = \begin{pmatrix} 10 \times (-0.142) \\ 2 \times 0.6426 - 6 \end{pmatrix} = \begin{pmatrix} -1.42 \\ -4.71 \end{pmatrix}$$

梯度的模：

$$|\nabla U(\bar{x}_1)| = \sqrt{x_{11}^2 + x_{22}^2} = 4.92 > \varepsilon$$

7、以 $\bar{x}_1$ 为起点，以 $\bar{p}_1 = -\nabla U(\bar{x}_1)$ 为方向仿照上述过程继续寻查，直至求出满足误差要求的极小点为止。



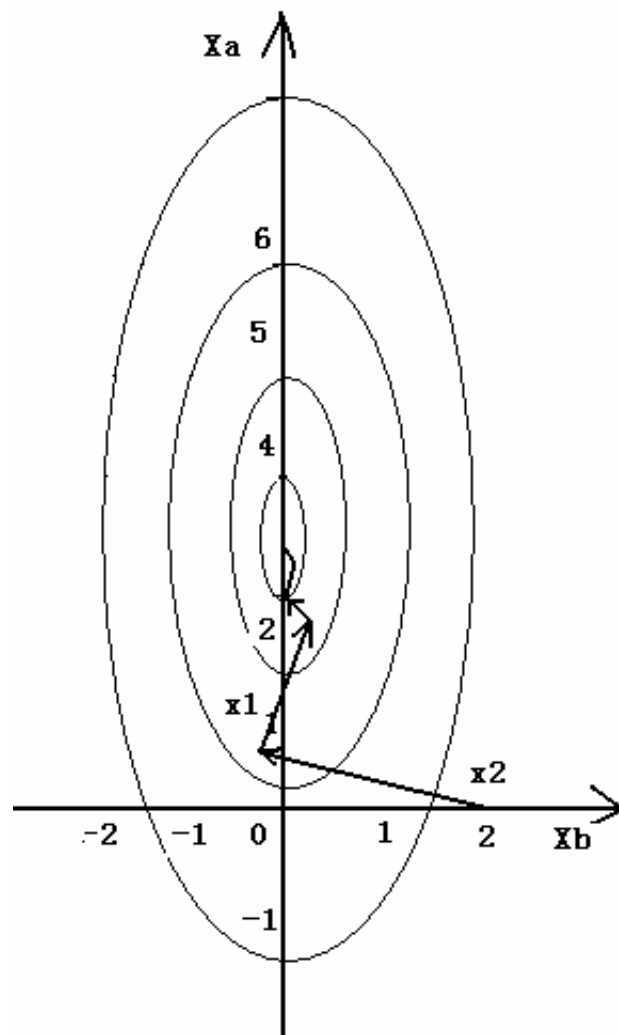
第四章 最优化方法和最优化设计

一、最速下降法

结论:

最初一两次寻查的下降速度较快，而以后各次的下降速度减缓，当目标函数接近极小值时，下降速度则变得很慢。

最速下降法只在初值点远离时具有“最速下降”的性质。而接近极小点时，收敛速度变得比较缓慢。这是因为最速下降法中每相邻两次的搜索方向是互相正交的，故搜索途径呈直角锯齿形。当接近极小点时，齿形越来越小，收敛速度变得相当缓慢。



第四章 最优化方法和最优化设计

一、最速下降法

结论：

为提高收敛速度，在最速下降法发展出牛顿法、变尺度法等许多方法。但是，由于最速下降法在寻查的第一次是简单且有效的，所以各种梯度法大多采用最速下降法作为开始的方法。

