现代数值计算方法

第七章 非线性方程迭代解法

第七章 非线性方程迭代解法

§7.2 简单迭代法及其加速技巧

§7.2.3 迭代法加速技巧

对于一个收敛的迭代过程, 只要迭代足够多次, 就可以使结果达到任意精度, 但有时迭代过程收敛缓慢, 从而使计算过程变得很大, 因此迭代过程的加速是个重要课题.

由定理 7.4 可知, 当 $\varphi'(x^*) \neq 0$ 时, 迭代公式 (7.5) 只有线性收敛速度, 收敛到真解的速度是比较缓慢的. 我们下面来考虑迭代过程的加速问题.

设 x_k 是根 x^* 的某个近似值, 用迭代公式校正一次得

$$y_k = \varphi(x_k).$$

假设 $\varphi'(x)$ 在所考察的范围内变化不大, 其估计值为 q, 则

$$x^* - y_k = \varphi(x^*) - \varphi(x_k) \approx q(x^* - x_k),$$

由此解出 x^* 得

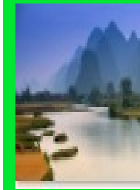
$$x^* \approx \frac{1}{1-q}y_k - \frac{q}{1-q}x_k,$$

这就是说, 如果将迭代值 y_k 与 x_k 加权平均, 可望得到的

$$x_{k+1} = \frac{1}{1-q}y_k - \frac{q}{1-q}x_k$$

是比 y_k 更好的近似根. 这样加工后的计算过程是:

迭代 $y_k = \varphi(x_k)$,



Back
Close

改进 $x_{k+1} = \frac{1}{1-q} y_k - \frac{q}{1-q} x_k.$

这组迭代公式可以合并成

$$x_{k+1} = \frac{1}{1-q}\big[\varphi(x_k) - q x_k\big]. \tag{7.13}$$

例如, 用迭代公式 (7.13) 求解例 7.4, 由于 $\varphi'(x) = -\mathrm{e}^{-x}$, 故可取 $q = -0.6$, 则加速公式 (7.13) 的具体形式为

$$x_{k+1} = \frac{1}{1.6}\big(\mathrm{e}^{-x_k} + 0.6 x_k\big).$$

利用简单迭代法通用程序 maiter.m, 在 MATLAB 命令窗口执行:

```
>> x=maiter(inline('(exp(-x)+0.6*x)/1.6'),0.5,1e-5)
```

可得到计算结果

```
k=3
```

```
x =

   0.56714328557022
```

即只需迭代 3 次, 就可以得到与例 7.4 同样精度的结果 (而在那里迭代了 17 次!), 可见这种加速的效果是明显的.

尽管如此, 加速迭代公式 (7.13) 使用起来却很不方便, 因为常数 q 一般是很难确定的. 鉴于此, Aitken-Steffensen (艾特金-斯蒂文森) 提出了一个实用的加速迭代公式. 其基本思想是:

设当前近似点为 x_k, 令

$$y_k = \varphi(x_k), \tag{7.14}$$

$$z_k = \varphi(y_k). \tag{7.15}$$

仍设 $\varphi'(x) \approx q$, 于是有

$$y_k - x^* \approx q(x_k - x^*), \quad z_k - x^* \approx q(y_k - x^*),$$

将上面两式相除得

$$\frac{y_k - x^*}{z_k - x^*} \approx \frac{x_k - x^*}{y_k - x^*},$$

由上式解出 x^*, 得

$$x^* \approx \frac{x_k z_k - y_k^2}{z_k - 2y_k + x_k} = x_k - \frac{(y_k - x_k)^2}{z_k - 2y_k + x_k}.$$

可将上式的右端作为新的近似值, 即

$$x_{k+1} = x_k - \frac{(y_k - x_k)^2}{z_k - 2y_k + x_k}, \quad k = 0, 1, \cdots \tag{7.16}$$

下面写出具体算法:

算法 7.4 (Aitken-Steffensen **加速方法**)

步 1 取初始点 x_0, 最大迭代次数 N 和精度要求 ε, 置 $k := 0$;

步 2 计算 $y_k = \varphi(x_k)$, $z_k = \varphi(y_k)$, 及

$$x_{k+1} = x_k - \frac{(y_k - x_k)^2}{z_k - 2y_k + x_k};$$

步 3　若 $|x_{k+1} - x_k| < \varepsilon$, 则停算;

步 4　若 $k = N$, 则停算; 否则, 置 $k := k + 1$, 转步 2.

可以证明, Aitken-Steffensen 加速方法具有平方收敛速度, 此结论略去不证.

根据算法 7.4, 编制 MATLAB 通用程序如下:

- Aitken-Steffensen 加速法 MATLAB 程序

```
%maaitken.m
function x=maaitken(phi,x0,ep,N)
%用途: 用Aitken-Steffensen加速方法求f(x)=0的解
%格式: x=maaitken(phi,x0,ep,N)   phi为迭代函数,x0为
%       迭代初值,ep为精度(默认1e-4), N为最大迭代次数
%       (默认500), x返回近似根
```

```
if nargin<4,N=500;end
if nargin<3,ep=1e-4;end
k=0;
while k<N
   y=feval(phi,x0);
   z=feval(phi,y);
   x=x0-(y-x0)^2/(z-2*y+x0);
   if abs(x-x0)<ep, break;  end
   x0=x; k=k+1;
end
if k==N, warning(’已达迭代次数上限’); end
disp([’k=’,num2str(k)])
```

例 7.7 用 Aitken-Steffensen 加速法求方程 $f(x) = xe^x - 1 = 0$ 在

$[0, 1]$ 内的一个实根, 取初始点为 $x_0 = 0.5$, 精度为 10^{-5}.

解 在 MATLAB 命令窗口执行:

```
>> x=maaitken(inline('exp(-x)'),0.5,1e-5)
k= 2
x =
    0.56714329040978
```

即只需 2 次迭代就可以得到满足精度要求的结果.

例 7.8 设函数 $\varphi(x)$ 连续可微, 若迭代公式 $x_{k+1} = \varphi(x_k)$ 局部线性收敛, 对于 $a \in R$, 构造迭代公式

$$x_{k+1} = \psi(x_k), \quad k = 0, 1, \cdots \tag{7.17}$$

其中

$$\psi(x) = \frac{1}{1-a}\varphi(x) - \frac{a}{1-a}x.$$

Back
Close

试选取 a 的值使得迭代公式 (7.17) 具有更高的收敛阶.

解　因 $x_{k+1} = \varphi(x_k)$ 局部线性收敛, 故存在不动点 x^* 使得 $x^* = \varphi(x^*)$. 注意到

$$\psi(x^*) = \frac{1}{1-a}\varphi(x^*) - \frac{a}{1-a}x^* = \frac{1}{1-a}x^* - \frac{a}{1-a}x^* = x^*,$$

即 x^* 也是 $\psi(x)$ 的不动点. 对 $\psi(x)$ 求导数得

$$\psi'(x) = \frac{1}{1-a}\varphi'(x) - \frac{a}{1-a}.$$

要使迭代公式 (7.17) 具有比线性收敛更高的收敛阶, 必须满足条件 $\psi'(x^*) = 0$, 即

$$\frac{1}{1-a}\varphi'(x^*) - \frac{a}{1-a} = 0,$$

解得

$$a = \varphi'(x^*).$$

因此, 如果选取 $a = \varphi'(x^*)$, 则迭代公式 (7.17) 至少是二阶收敛的.

作业：P168: 7.10; P169: 7.19.