

# 浙江大学实验报告

专业: 混合班  
姓名: 徐圣泽  
学号: 3190102721  
日期: 2021.6.27

课程名称: 科学计算      指导老师: 赖俊      实验名称: 方程求解实验

## 第十三次作业

### 第十三次作业

#### 问题

#### 数学原理

- 1、二分法
- 2、迭代法

#### 程序代码

- 1、二分法
- 2、迭代法

#### 结果

- 1、解的存在唯一性
- 2、二分法求解
- 3、迭代法求解
- 4、迭代的等价形式
- 5、“最速”迭代法

#### 思考讨论

## 问题

考虑方程

$$x = e^{-x}$$

- 1、证明该方程在实轴上有且仅有一个根 $\alpha$ ，且该根在区间 $[0, 1]$ 上。
- 2、利用二分法求解该方程，要求计算的精度达到机器精度（*MATLAB*中常数 $eps$ 代表机器精度）。
- 3、利用迭代法 $x_{n+1} = e^{-x_n}$ 求解该方程，初始值 $x_0 = 1$ ，要求计算的精度达到机器精度。
- 4、考虑方程的等价形式

$$x = \frac{we^{-x} + x}{1 + w}$$

其中 $w$ 是不等于0和-1的任意实数。请给出 $w$ 的取值范围，使得在这个范围内，由此构造的迭代法比前一个迭代法局部收敛更快。并在 $w$ 的范围中任取一值，令 $x_0 = 1$ ，验证算法是否比前一个迭代法更快达到机器精度，即迭代次数是否更少。

- 5、承接上题，找到一个最优的 $w$ ，使得迭代次数达到最快，并用程序验证（ $x_0 = 1$ ），这里需要方程的解 $\alpha$ ，可以用上面计算的 $\alpha$ 的近似值代替。

综合以上结果，讨论各迭代法优劣。

# 数学原理

## 1、二分法

给定方程 $f(x) = 0$ ，设 $f(x)$ 在区间 $[a, b]$ 连续，且 $f(a)f(b) < 0$ ，则方程 $f(x)$ 在 $(a, b)$ 内至少有一根。为便于讨论，不妨设方程 $f(x) = 0$ 在 $(a, b)$ 内只有一实根，采取对分区间的方法使有根区间逐步缩小，从而得到满足精度要求的实根的近似值。

在二分法的求解过程中，我们用来重复判断根存在与否并取得新区间的原则就是根的存在定理。

## 2、迭代法

若给定方程 $f(x) = 0$ 以后，把它改写成等价的形式 $x = \varphi(x)$ ，再作迭代 $x_{n+1} = \varphi(x_n)$ 。

当给定 $x_0$ 以后，可以用上述公式计算 $x_1$ ，再把 $x_1$ 代入公式右端计算得到 $x_2$ ，如此等等，这样就计算出一个序列 $\{x_n\}$ ，如果序列 $\{x_n\}$ 收敛，并假设 $\lim_{n \rightarrow \infty} x_n = \xi$ ，则有如下关系，

$$\xi = \lim_{n \rightarrow \infty} x_{n+1} = \lim_{n \rightarrow \infty} \varphi(x_n) = \varphi(\lim_{n \rightarrow \infty} x_n) = \varphi(\xi)$$

也就是说， $\xi$ 是方程 $f(x) = 0$ 的根。 $\xi$ 也叫做 $\varphi(x)$ 的不动点，因此迭代 $x_{n+1} = \varphi(x_n)$ 也称不动点迭代。

# 程序代码

## 1、二分法

首先，根据二分法原理编写程序所用到的函数 `bisection(f,a,b,eps)`：

```
function xc=bisection(f,a,b,eps)
    while (b-a)/2>eps
        c=(a+b)/2;
        if f(c)==0
            break
        end
        if f(a)*f(c)<0
            b=c;
        else
            a=c;
        end
    end
    xc=(a+b)/2;
end
```

接下来，编写二分法的主程序，如下：

```
format long e;
syms x;
f=inline("x-exp(-x)","x");
xc1=bisection(f,0,1,eps);
fprintf('解为%.10f\n',xc1);
```

## 2、迭代法

在本部分，我们直接编写迭代法的主程序：

```
format long e;
x=1;
```

```

N=10000;
eps=eps;
cnt=0;
while cnt<N
    x1=func(x);
    cnt=cnt+1;
    if abs(x1-x)<eps
        break;
    end
    x=x1;
end
fprintf('共迭代%d次,解为%.10f\n',cnt,x1);

```

在上述程序中，我们用到了函数 `func(x)`，用于计算迭代后的值，在此补充定义：

```

function x = func(x)
    x=exp(-x);
    %x=(exp(-x)+x)/2;
    %x=(1/0.5671432904*exp(-x)+x)/(1+1/0.5671432904);
end

```

## 结果

注：以下五个部分的解答分别对应题目中的五个小问题。

### 1、解的存在唯一性

设  $f(x) = x - e^{-x}$ ，则  $f'(x) = 1 + e^{-x}$  在实轴上恒大于0，因此  $f(x)$  在  $R$  上递增。

又由  $f(0) = 0 - 1 = -1 < 0$ ， $f(1) = 1 - e^{-1} > 0$ ，易知  $f(x)$  在  $R$  上有且仅有一个根  $\alpha$ ，且  $\alpha$  在  $[0, 1]$  上。

### 2、二分法求解

将  $f(x) = x - e^{-x}$  代入程序，运行得到以下结果：

解为0.5671432904098

我们可以在程序中加入 `cnt` 变量记录函数循环的次数，可以得到二分法的迭代次数为 53 次。

### 3、迭代法求解

将  $x = e^{-x}$  作为迭代公式代入函数  $func(x)$ ，运行主程序得到以下结果：

共迭代65次,解为0.5671432904098

我们通程序得到了迭代达到要求精度的次数，以此作为衡量迭代快慢的依据，并且需要与后面得到的结果相比较。

### 4、迭代的等价形式

在本部分中，我们改变写法得到了方程的等价形式： $x = \frac{we^{-x}+x}{1+w}$ ，其中  $w \neq 0, -1$ ，为了得比上个部分更加快的迭代法，我们首先需要求解得到  $w$  的范围。

首先令  $\varphi(x) = \frac{we^{-x}+x}{1+w}$ ， $\phi(x) = e^{-x}$ ，为了满足题目要求，需要满足条件： $|\varphi'(\alpha)| < |\phi'(\alpha)|$ 。

代入 $\alpha$ ，解上述不等式得到 $w$ 的取值范围是： $w > \frac{1-e^{-\alpha}}{2e^{-\alpha}} = \frac{1-\alpha}{2\alpha}$ 。

取满足条件的其中一个值 $w = 1$ 代入方程，此时迭代公式为 $x = \frac{e^{-x}+x}{2}$ ，运行程序得到以下结果：

共迭代25次，解为0.5671432904098

可以看出，采用新的迭代公式的算法确实比前一个迭代法需要的迭代次数更少，更快达到机器精度。

## 5、“最速”迭代法

根据上一个部分的讨论我们知道，要寻找到最优的 $w$ 使得迭代次数最快，只需要满足 $|\varphi'(x)| = 0$ ，求解得到 $w = e^{\alpha} = \frac{1}{\alpha}$ 。

将结果代入迭代公式，并运行主程序，得到以下结果：

共迭代5次，解为0.5671432904098

可以发现，迭代次数远远小于前面两个部分得到的结果。

为了保险起见，我们也可以利用 `matlab` 对 $w$ 取值范围内的值进行合理的遍历取值，可以验证这已经是最优算法。

最后 $w$ 的取值为1.763222834351897。

## 思考讨论

根据相关理论和程序编写，我们对方程求解的各种方法进行了一定的探究。在本次求解的过程中，我们主要采用了三种方法——二分法、迭代法、加速优化的迭代法，三种方法各有其优劣之处。

在计算精度上，三种方法相差无几，这也是由于题目中对三种方法的要求精度均为机器精度。由此可见，在方程表达式并不是十分复杂和不考虑程序运行时间的情况下，三种方法我们均可以采取。

在计算时间上，三者在本题运算求解的过程中体现了一定的差异。在本题的情况下，二分法和普通迭代法的计算速度差不多，二分法的迭代次数稍稍小于普通迭代法的迭代次数。对于优化过的迭代法而言，其计算速度有了较大幅度上的提升，从最初的65次提升至25次，最后提升至最优速度5次，可以看出，加速优化对迭代法计算速度的提升是显著的。

虽然第三种方法的性能有显著的优势，但也并非在所有情况下都需要尽量使用优化过的迭代法，在本题求解 $w$ 的范围对迭代法进行优化时，也用到了方程解 $\alpha$ 的估计值，这在一定程度上带来了误差和不必要的麻烦。在方程形式并不是十分繁琐的前提下，计算时间的复杂性不会过大，我们完全可以采用二分法和普通的迭代法进行求解，而在方程较为复杂之时，我们可以采取一定的方法将方程改写成等价的形式，利用优化过的迭代法进行求解。

综上所述，在综合考量计算精度、计算时间、前期处理复杂程度等多种因素后，我们可以采用成本最小的方法对方程进行求解。