

# lab4 实验报告

---

计63 黄冰鉴 2016011296

2019/5/26

---

## 第四章上机题2

这道上机题主要考察的是用matlab实现三种经典的迭代解法，并用来解决实际常微分方程求解问题。

### 解题过程

- 首先在matlab中用矩阵的形式表示A，它的维度是 $n-1 \times n-1$ 。
- 然后将A分解为迭代求解适用的D L U三个矩阵。
- 接着在三个不同的文件中实现雅可比，G-S和SOR三个算法，并在主脚本中调用三个算法得到迭代解。
- 与精确解比较，得到三种算法的误差。

### 代码介绍

初始化&获得精确解

```
eps = 1;
a = 0.5;
n = 100;
h = 1/n;
A = zeros(n-1);
for i = 1:n-1
    A(i, i) = -(2*eps+h);
    if i ~= 1
        A(i-1, i) = eps+h;
        A(i, i-1) = eps;
    end
end
b = ones(n-1, 1);
b = b .* (a*h*h);
y_prec = A\b;
```

计算通用的DLU三矩阵

```

D = zeros(n-1);
L = zeros(n-1);
U = zeros(n-1);
for i = 1:n-1
    D(i, i) = A(i, i);
    if i ~= 1
        L(i, i-1) = -A(i, i-1);
        U(i-1, i) = -A(i-1, i);
    end
end
end

```

调用三个函数得到迭代解，并计算误差。

```

y_jacobi = jacobi(D, L, U, b);
y_gaussseidel = gaussseidel(D, L, U, b);
w = 0.9;
y_sor = sor(D, L, U, b, w);
y_res1 = (y_jacobi - y_prec) ./ y_prec;
y_res2 = (y_gaussseidel - y_prec) ./ y_prec;
y_res3 = (y_sor - y_prec) ./ y_prec;
norm(y_res1, Inf)
norm(y_res2, Inf)
norm(y_res3, Inf)

```

在三个函数的实现上，我使用norm(Inf)作为有效数字判断标准，并用DLU的形式表示迭代步。

```

%判断是否达到4位有效数字
while norm((x_prev - x_next) ./ x_prev, Inf) >= 1e-4
%三个迭代算法的步骤
%jacobi
x_next = D \ (L + U) * x_prev + D \ b;
%gaussseidel
x_next = (D - L) \ U * x_prev + (D - L) \ b;
%sor
x_next = (D .* w_inv - L) \ ((w_inv - 1) .* D + U) * x_prev + (D .* w_inv - L) \ b;

```

## 结论

不同 $\varepsilon$ 条件下使用三个算法的相对误差如下，

$\varepsilon$	jacobi	gaussseidel	sor
1	0.0919	0.0898	0.1076
0.1	0.0319	0.0299	0.0363
0.01	0.0011	0.0010	0.0013
0.0001	3.31e-05	4.31e-06	5.25e-05

$\varepsilon$  越小，相对误差也就越小。