

# 三对角矩阵求逆算法实现

常清俊

2017年11月8日

## Contents

- STEP1 input A
- STEP2 将A分解成 $A = LU$
- 算法2.2的一部分
- STEP3 计算逆矩阵的最后一个元
- STEP4 计算逆矩阵C的主对角线元素
- STEP5 计算第i行位于主对角线元素左边的元素以及位于第i列位于主对角线上方的元素
- STEP6 输出C

```
function [A,C] = cqj_InverseMatrixOfTridiagonalMatrices(a,b,c)
```

```
% 本文实现的是论文《三对角矩阵求逆的算法》论文作者：冉瑞生、黄延祝等人
```

```
% 本文算法的计算复杂度为 $O(n^2)$ 
```

```
% 应用数学和力学，第30卷，第2期 2009年2月15日出版 文章编号：  
1000-0887(2009)02-0238-07
```

```
% Code author:Qingjun Chang qingjun_chang@163.com & qingjun.cn@gmail.com
```

```
% INPUT a: 下对角线  $1*(n-1)$ 
```

```
% b: 对角线  $1*n$ 
```

```
% c: 上对角线  $1*(n-1)$ 
```

```
% OUTPUT A: 原始三对角矩阵
```

```
% C: 逆矩阵
```

```
C = [];
```

```
flag = 0;
```

### STEP1 input A

```
A = diag(a,-1)+diag(b)+diag(c,1);  
a = [0 a];
```

### STEP2 将A分解成A = LU

$$\alpha_1 = b_1, \tau_{i-1} = \frac{c_{i-1}}{\alpha_{i-1}}, \alpha_i = b_i - a_i \tau_{i-1}, \gamma_i = \frac{a_i}{\alpha_{i-1}} (i = 2, \dots, n)$$

```
n = size(A,1);  
alpha(1) = b(1);  
if alpha(1) == 0  
    fprintf('算法2.1失败!正在尝试算法2.2...\n');  
    syms x  
    alpha = x;  
    flag = 1;  
end  
for i = 2:n  
    t(i-1) = c(i-1)/alpha(i-1);  
    alpha(i) = b(i)-a(i)*t(i-1);  
    gamma(i) = a(i)/alpha(i-1);  
    if alpha(i) == 0  
        fprintf('算法2.1失败!正在尝试算法2.2...\n');  
        syms x  
        alpha = [alpha(1:i-1) x];  
        t = [t x];  
        gamma = [gamma x];  
        flag = 1;  
    end  
end  
if alpha(n) == 0  
    fprintf('矩阵是奇异的!\n');  
    return;  
end
```

算法2.1失败!正在尝试算法2.2...

### 算法2.2的一部分

$$P(x) = \prod_{i=1}^n \alpha_i$$

```
if flag
    P = 1;
    for i = 1:n
        P = P*alpha(i);
    end
    P = expand(P);
    P_poly = sym2poly(P); %偶尔会出现错误提示
    是Not a polynomial.以后修复
    if ~P_poly(end)
        fprintf('矩阵是奇异的\n');
        return;
    end
end
```

### STEP3 计算逆矩阵的最后一个元

$$C_{nn} = \frac{1}{\alpha_n}$$

```
C = zeros(n);
if flag
    C = [C(1:end-1) x];
    C = reshape(C,n,n);
end
C(n,n) = 1/alpha(n);
```

### STEP4 计算逆矩阵C的主对角线元素

$$C_{ii} = \frac{1}{\alpha_i} + \tau_i \gamma_{i+1} C_{i+1,i+1} (i = n-1, \dots, 2, 1)$$

```

for i = n-1:-1:1
    C(i,i) = 1/alpha(i)+t(i)*gamma(i+1)*C(i+1,i+1);
end

```

**STEP5** 计算第*i*行位于主对角线元素左边的元素以及位于第*i*列位于主对角线上方的元素

$$C_{ij} = -\gamma_{j+1}C_{i,j+1} (i = n, n-1, \dots, 2; j = i-1, \dots, 1)$$

$$C_{ji} = -\tau_j C_{j+1,i} (i = n, n-1, \dots, 2; j = i-1, \dots, 1)$$

```

for i = n:-1:2
    for j = i-1:-1:1
        C(i,j) = -gamma(j+1)*C(i,j+1);
    end
end
for i = n:-1:2
    for j = i-1:-1:1
        C(j,i) = -t(j)*C(j+1,i);
    end
end

```

**STEP6** 输出C

```

if flag
    C = expand(C);
    str = '@(x) [';
    for i = 1:n
        for j = 1:n
            temp = factor(C(i,j));
            str = [str ' ' char(prod(temp))];
        end
        str = [str ';'];
    end
    str = [str ']'];
end

```

```
Cfun = str2func(str);  
C = Cfun(0);  
return;  
end
```

A =

1	1	0	0
1	3	2	0
0	-1	-1	1
0	0	-1	1

C =

1.0000	0	1.0000	-1.0000
0	0	-1.0000	1.0000
-0.5000	0.5000	1.0000	-1.0000
-0.5000	0.5000	1.0000	0