Algorithmische Methoden in der Numerik - Uebung3

Felix Dreßler (k12105003)
Elisabeth Köberle (k12110408)

30. Juni 2022

1 Aufgabe 4

1.1 Teilaufgabe a

```
function [K_h] = Stiff1(N)

n = (N-1)^2;
h = 1/N;

b1 = -ones(n-1,1);

b = -ones(n-N+1,1);

b1(N-1:N-1:n-1) = 0;

i = [1:n,          1:n-1, 2:n,   1:n-N+1, N:n    ];
j = [1:n,          2:n,   1:n-1, N:n,     1:n-N+1];
s = [4*ones(n,1)', b1',   b1',   b',      b'     ];

K_h = (1/(h*h)) * sparse(i,j,s,n,n);

end
```

```
function [K_h] = Stiff2(N)

n = (N-1)^2;
h = 1/N;
e = ones(n,1);

b1 = -ones(n-1,1);
b1(N-1:N-1:n-1) = 0;


B = [-e,[b1',0]',4*e,[0,b1']',-e];
d = [-N+1,-1,0,1,N-1];

K_h = (1/h^2) * spdiags(B,d,n,n);

end
```

1.2 Teilaufgabe b

```
function [f_h] = RHS(f,N)

n = (N-1)^2;

f_h = zeros(n,1);

h = 1/N;

for i = 1:N-1
    for j = 1:N-1
        f_h(i+(j-1)*(N-1)) = f(j*h,i*h);
    end
end
```

1.3 Teilaufgabe c

Zur Lösung des GLS haben wir zuerst zwei funktionen *f.m* und eine Funktion *ub.m* zur einfacheren Eingabe erstellt.

```
function [a] = f(x,y)

a = 2*(x*(2-3*y)+(-1+y)^2*y+x^2*(-2+3*y)+113*pi^2*sin(15*pi*x)*sin(pi*y));

end
```

```
function [a] = ub(x, y)

a = x*(1-x)*y*(1-y)^2+sin(15*pi*x)*sin(pi*y);

end
```

Im Folgenden nun die Lösungen des GLS: (die Matlab Workspace Dateien waren für die Abgabe zu groß, zur Übersichtlichkeit wurden die Ergebnisse der einzelnen Rechnungen großteils unterdrückt) Zur besseren Übersicht wurde ein kurzes Skript zur Fehlerberechnung erstellt.

```
for N =[10 ,100 ,1000]
disp ('N');
disp (N) ;

K_h = Stiff1 (N);

f_h = RHS (@f ,N);

u_h_exact = RHS (@ub ,N); % exakt

disp ('direkte Berechnung :')
tic
u_h_direct = K_h \ f_h ;
toc

disp ('Fehler direkte Lsg ( Maximumsnorm ):')
max ( abs ( u_h_direct - u_h_exact ) )

disp ('iterative Berechnung ')
tic
u_h_pcg = pcg ( K_h , f_h );
toc

disp ('Fehler iterative Lsg mit ( Maximumsnorm ):')
max ( abs ( u_h_pcg - u_h_exact ))

end
```

Die Outputs waren wie folgt:

```
        N
        10

        direkte Berechnung :
        Elapsed time is 0.000320 seconds.
        Fehler direkte Lsg ( Maximumsnorm ):

        ans =
```

```

    9.6323

    iterative Berechnung
    pcg converged at iteration 12 to a solution with relative residual 9.4e-07.
    Elapsed time is 0.022338 seconds.
    Fehler iterative Lsg mit ( Maximumsnorm ):

    ans =

    9.6323

    N
    100

    direkte Berechnung :
    Elapsed time is 0.019078 seconds.
    Fehler direkte Lsg ( Maximumsnorm ):

    ans =

    0.0186

    iterative Berechnung
    pcg stopped at iteration 20 without converging to the desired tolerance 1e-06
    because the maximum number of iterations was reached.
    The iterate returned (number 1) has relative residual 0.00052.
    Elapsed time is 0.013484 seconds.
    Fehler iterative Lsg mit ( Maximumsnorm ):

    ans =

    0.0535

    N
    1000

    direkte Berechnung :
    Elapsed time is 2.263114 seconds.
    Fehler direkte Lsg ( Maximumsnorm ):

    ans =

    1.8426e-04

    iterative Berechnung
    pcg stopped at iteration 20 without converging to the desired tolerance 1e-06
    because the maximum number of iterations was reached.
    The iterate returned (number 10) has relative residual 0.0016.
    Elapsed time is 0.375518 seconds.
    Fehler iterative Lsg mit ( Maximumsnorm ):

    ans =

    0.0368
```

2 Aufgabe 5

```
%A ... sparse Matrix
%löst GLS A*x = b mit Startvektor x0
function [x,exitflag] = Jacobi(A,b,x0,itmax,myeps)

exitflag = 0;
[n,m] = size(A);

if nargin < 5
    myeps = 1.e-8;
end
if nargin < 4
    itmax = n*n;
end
if nargin < 3
    x0 = zeros(n);
end

if n ~= m
    return;
end

x = x0;
k = 0;
r0 = b-A*x0;

nr0 = norm(r0)*myeps;

rk = r0;

W = diag(A);

if W(1:n) == 0
    return;
end

while nr0 <= norm(rk) && k < itmax
    pk = W .\ rk;
    x = x + pk;
    rk = b-A*x;

    k = k+1;
end

if(nr0 >= norm(rk))
    exitflag = 1;
end

end
```

2.1 Testberechnung

Zur Durchführung dieser Tests wurde wieder ein kurzes Skript erstellt.:

```
for N = [10 ,100 ,1000]
disp ('N:')
```

```
disp (N)

K_h = Stiff1(N);
f_h = RHS(@f,N);

tic ;
[ u_h_Jacobi , err ] = Jacobi ( K_h , f_h );
toc ;

if( err == 0 )
disp ('Did not converge ');
end
u_h_exact = RHS(@ub,N) ;
u_h_pcg = pcg ( K_h , f_h );

max(max ( abs ( u_h_Jacobi - u_h_exact ) ))
max(max ( abs ( u_h_Jacobi - u_h_pcg )))

end
```

das die folgenden Outputs lieferte:

```
        N:
        10

        Elapsed time is 0.161501 seconds.
        pcg converged at iteration 12 to a solution with relative residual 9.4e-07.

        ans =

        9.6323


        ans =

        1.3104e-05
```

Ab N = 100 dauerte die Berechnung länger als 2 Minuten, deshalb wurde an diesem Punkt abgebrochen.