

MATLAB Einführung

Numerische Methoden für ITET und MATL 2016

ETH Zürich, Seminar for Applied Mathematics

Dr. S. May, D. Devaud



MATLAB

- Auf ETH Computer vorinstalliert
- Auf Heim PC: von www.ides.ethz.ch herunterladen
- Bemerkung: **MATLAB Node TAH R2015x Stud Version**
- Opensource Alternative: Octave

The screenshot shows the MATLAB R2014a environment. The Command Window contains the following code and output:

```

>> a = 3
a =
    3

>> v1 = [1 ; -2 ; 3]
v1 =
     1
    -2
     3

>> v2 = [1 -2 3]
v2 =
     1    -2     3

>> A = [3 1 2 ; -1 0 5 ; 5 6 7]
A =
     3     1     2
    -1     0     5
     5     6     7

//>>
  
```

The Workspace window on the right shows the following variables:

Name	Value	Min
a	3	3
A	[3 1 2; -1 0 5; 5 6 7]	-1
v1	[1; -2; 3]	-2
v2	[1 -2 3]	-2

Variablen

Zahlen:

```
>> a = 3  
a =  
    3
```

Spaltenvektor:

```
>> v1 = [1 ; -2 ; 3]  
v1 =  
    1  
   -2  
    3
```

Zeilenvektor:

```
>> v2 = [1 -2 3]  
v2 =  
    1    -2    3
```

Matrix:

```
>> A = [3 1 2 ; -1 0 5 ; 5 6 7]  
A =  
    3    1    2  
   -1    0    5  
    5    6    7
```

Operationen

- Intuitive Rechenoperationen $+$, $-$, $*$, $/$
 - ⚠ Beide Variablen müssen die gleiche Grösse haben!
Überprüfen Sie es mit `size`.
- Für eine Matrix A gibt $A(i,j)$ die Zahl in der i -ten Zeile und j -ten Spalte von A zurück

```
>> A(2,3)
```

```
ans =
```

```
5
```

- MATLAB Hilfe: `doc <Befehl>` oder `help <Befehl>`, z.B.

```
>> help size
```

```
size    Size of array.
```

```
D = size(X), for M-by-N matrix X, returns the  
two-element row vector
```

```
D = [M,N] containing the number of rows and  
columns in the matrix.
```

```
.....
```

Lineare Gleichungssysteme Lösen

Um ein lineares Gleichungssystem $Ax = b$ zu lösen kann man verschiedene Funktionen benutzen, z.B.

- Direkt (mit $v1$ statt b):

```
>> x = A\v1
```

```
x =
```

```
    0.3571
```

```
    0.5857
```

```
   -0.3286
```

- QR-Zerlegung: $A = QR$ mit Q orthogonal und R eine rechte Dreiecksmatrix

$$b = Ax = QRx \Rightarrow Rx = Q^T b =: c \Rightarrow x = R^{-1}(Q^T b)$$

```
>> [Q,R] = qr(A);
```

```
>> c = Q'*v1;
```

```
>> x = R\c
```

```
x =
```

```
    0.3571
```

```
    0.5857
```

```
   -0.3286
```

for Schleifen am Bsp. der Fixpunktiteration

Zu einer gegebenen Funktion f suchen wir einen Fixpunkt x von f , d.h. $f(x) = x$. Die Fixpunktiteration ist gegeben durch

$$x_{k+1} = f(x_k), \quad x_0 = 1.$$

```
>> x = 1;
>> for i=1:5
    x = sqrt(x)+1    % f(x) = sqrt(x)+1
end
x =
    2
x =
    2.4142
x =
    2.5538
x =
    2.5981
x =
    2.6118
```

if-then-else Anweisung

$$y = \begin{cases} 2 & \text{falls } x > 1, \\ 7 & \text{sonst.} \end{cases}$$

in MATLAB :

```
>> x = 3;
>> if x > 1                % if-Bedingung
    y = 2                  % then-Anweisung
else                       % else-Anweisung
    y = 7
end
y =
    2
```

 Zahlen vergleichen:

```
>> if(abs(a-b)<1e-10)      % Numerischer Vergleich
und nicht
>> if(a == b)             % Analytischer Vergleich
```

Function Handles

```
>> f = @sin;
>> f(pi)
ans =
    1.2246e-16
>> sin(pi)
ans =
    1.2246e-16
>> f = @(x)(x^2+2);
>> [f(0) f(1) f(2)]
ans =
     2     3     6
```

% Function Handle

MATLAB Hilfe: `doc function_handle`

MATLAB Files - eigenes Skript

Datei: **fixpunkt.m**:

```
% Fixpunktiteration
f = @(x)(sqrt(x)+1);
x0 = 1;
for i=1:100
    x = f(x0);           % Fixpunktiteration
    if(abs(x-x0)<1e-6)  % |x_{k+1} - x_{k}|<1e-6
        break;         % aus der Schleife springen
    else
        x0 = x;        % x_k update
    end
end
x
```

MATLAB Aufruf:

```
>> fixpunkt
x =
    2.6180
```

MATLAB Funktionen

$$f(x) = ax^2 + bx + c$$

$$f'(x) = 2ax + b$$

Datei: **myfunc.m**:

```
function [f ,df] = myfunc(x, a, b, c)    % Gleich Name als  
    f = a*x^2 + b*x + c;                % der .m File  
    df = 2*a*x + b;  
end
```

MATLAB Aufruf:

```
>> [g, dg] = myfunc(1, 2, -1, 3)  
g =  
    4  
dg =  
    3
```

Anwendung zur Numerischen Quadratur

Mittelpunktsregel: (Ordnung 2 → exakt für Polynome vom Grad 1)

$$\int_a^b f(x) dx \approx (b-a) f\left(\frac{a+b}{2}\right)$$

in MATLAB : (**mittelpunkt.m**)

```
function quad_val = mittelpunkt(f,a,b)    % f handle function
    factor = (b-a);
    midpoint = (a+b)/2;
    quad_val = factor*f(midpoint);
end
```

MATLAB Aufruf:

```
>> f = @(x)(4*x-1);
>> mittelpunkt(f,0,1)
ans =
    1
```

Summierte Mittelpunktsregel

$$\int_a^b f(x) dx = \sum_{i=0}^{N-1} \int_{x_i}^{x_{i+1}} f(x) dx \approx \sum_{i=0}^{N-1} Q^M(f; x_i, x_{i+1}),$$

wobei $h = \frac{b-a}{N}$, $x_0 = a$, $x_i = x_0 + ih$, $x_N = b$. Subintervalle in MATLAB

```
>> N = 5;
```

```
>> x = linspace(0,1,N+1)
```

```
x =
```

```
    0    0.2000    0.4000    0.6000    0.8000    1.0000
```

oder

```
>> h = 1/N;
```

```
>> x = 0:h:1
```

```
x =
```

```
    0    0.2000    0.4000    0.6000    0.8000    1.0000
```

Summierte Mittelpunktsregel

Datei: (mittelpunkt_subintervalle.m)

```
function quad_val = mittelpunkt_subintervalle(f,a,b,N)
    x = linspace(a,b,N+1);
    quad_val = 0;
    for i=1:N          % Alternativ: verwende sum
        quad_val = quad_val + mittelpunkt(f,x(i),x(i+1));
    end
end
```

MATLAB Aufruf:

```
>> f = @(x)(4*x-1);
>> N = 5;
>> quad_val = mittelpunkt_subintervalle(f,0,1,N)
quad_val =
    1          % wie mit mittelpunkt(f,0,1)
              % -> exakt für Polynome vom Grad 1
```

Konvergenz der Quadraturformeln

Mittelpunktsregel hat lokale Ordnung 2 $\xRightarrow{\text{Thm.}}$ es gibt ein $C > 0$ so dass

$$E(h) := \left| \int_a^b f(x) dx - \sum_{i=0}^{N-1} Q(f, x_i, x_{i+1}) \right| \leq Ch^2$$

Überprüfen: berechnen für verschiedene $h \rightarrow$ plot!

MATLAB Code: (**convergence.m**)

```
% Fehler Berechnung
```

```
f = @(x)(x^5);
```

```
exact = 1/6;
```

```
N = 4:2:100;
```

```
format long;
```

```
error = zeros(1,length(N));           % Nicht size
```

```
for i=1:length(N)
```

```
    quad_val = mittelpunkt_subintervalle(f,0,1,N(i));
```

```
    error(i) = abs(exact-quad_val);
```

```
end
```

Konvergenz der Quadraturformeln - Plot

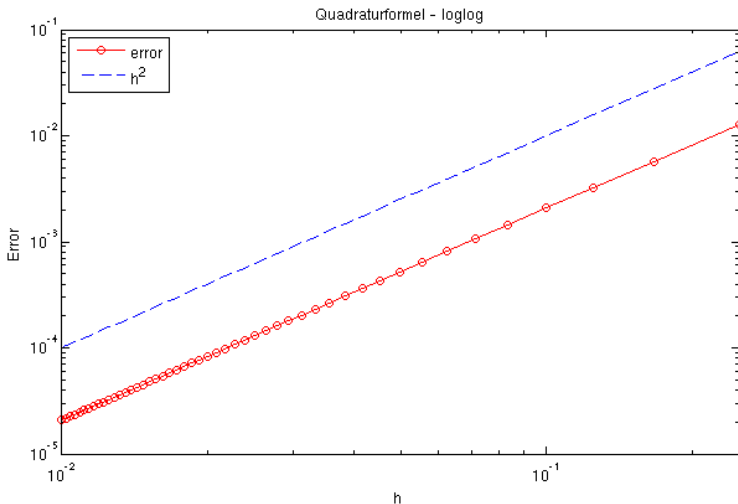
⚠ **log-log plot** → doc loglog ⚠

```
h = 1./N;                % Wichtig Punkt "." um
slope = h.*h;           % komponentweise zu rechnen !!!
loglog(h,error,'-or');
hold on
loglog(h,slope,'--b');
legend('error','h^2','Location','NorthWest'); % doc plot
xlabel('h');
ylabel('Error');
title('Quadraturformel - loglog');
xlim([h(end) h(1)]);
```

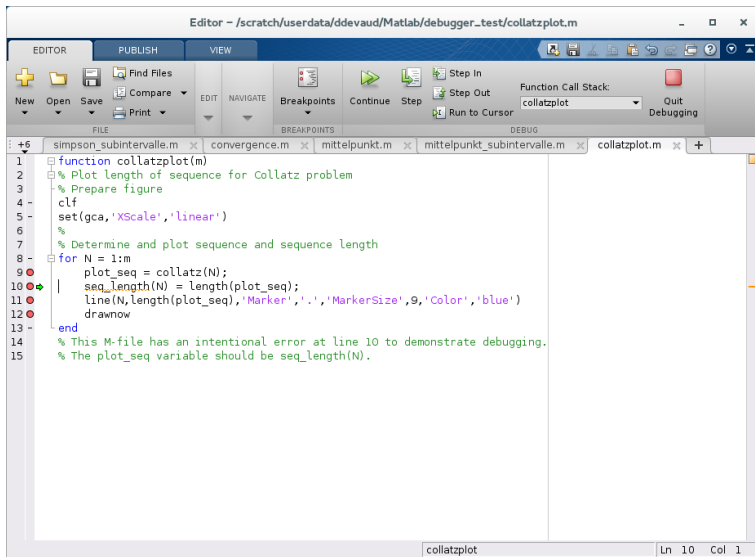
→ Mit log-Skalierung ist es viel einfacher zu vergleichen.

Konvergenz der Quadraturformeln - Plot

⚠ log-log plot → doc loglog ⚠



Debugger



The screenshot displays the MATLAB Editor/Debugger interface. The title bar reads "Editor - /scratch/userdata/ddevaud/Matlab/debugger_test/collatzplot.m". The interface is divided into several sections:

- EDITOR:** Contains standard file operations (New, Open, Save, Compare, Print) and editing tools (EDIT, NAVIGATE).
- PUBLISH:** Contains icons for saving and publishing the script.
- VIEW:** Contains icons for zooming and other view-related actions.
- DEBUG:** Contains debugging controls: Breakpoints, Continue, Step, Step In, Step Out, Run to Cursor, and Quit Debugging. The Function Call Stack shows "collatzplot".

The main editor window shows the following code for `collatzplot.m`:

```
1 function collatzplot(m)
2 % Plot length of sequence for Collatz problem
3 % Prepare figure
4 clf
5 set(gca,'XScale','linear')
6 %
7 % Determine and plot sequence and sequence length
8 for N = 1:m
9     plot_seq = collatz(N);
10    seq_length(N) = length(plot_seq);
11    line(N,length(plot_seq),'Marker','.', 'MarkerSize',9, 'Color','blue')
12    drawnow
13 end
14 % This M-file has an intentional error at line 10 to demonstrate debugging.
15 % The plot_seq variable should be seq_length(N).
```

A red dot indicates a breakpoint is set at line 10. The status bar at the bottom shows "collatzplot" and "Ln 10 Col 1".

Weitere Informationen

- Version **MATLAB Node TAH R2015x Stud** von `www.ides.ethz.ch` herunterladen
- MATLAB-Einführung auf der Webseite
- Fragen → `help` und `doc` Funktionen
- Benutzen Sie den **MATLAB debugger**
- Komplizierte Programme immer in kleinen Schritten schreiben

