

Differentialgleichungen

Themen der Aufgabenstellung:

- Realisierung des Euler-Verfahrens (explizit/implizit)
- Realisierung des Runge-Kutta-Verfahren
- steife Differentialgleichungen (stiff equations)
- nichtlineare Differentialgleichungen (Van-der-Pol-DGL)
- chaotische Systeme (Lorenz-Attraktor)
- Einstieg in MATLAB/Simulink

Aufgabenstellung:

1) Lösung "steifer Differentialgleichungen" mit Euler/Runge-Kutta (RK 2. Ordng.)

Es ist ein C-Programm zu schreiben, welches die folgende Differentialgleichung realisiert (unabh. Variable: x):

$$y' = 10 - 500 \cdot y + 5000 \cdot x \quad y(0) = 1$$

Versuchsvorbereitung:

- Geben Sie das Analogrechner-/Simulink-Schaltbild an.
- Geben Sie die Iterationsgleichungen für das **Euler**-Verfahren an.
- Geben Sie die Iterationsgleichungen für das **RK2**-Verfahren an.
- Geben Sie die Iterationsgleichungen für das **implizite Euler**-Verfahren an.
- Schreiben Sie ein Programm "*Stiff.ch*", welches die DGL mit allen Verfahren löst und zusammen mit der analytischen Lösung (s.u.) in einem Plot anzeigt.
Die analytische Lösung der Differentialgleichung lautet:

$$y = 10 \cdot x + e^{-500x}$$

Geben Sie in einem zweiten Plot die Abweichungen von der richtigen Lösung aus.

Versuchsdurchführung:

- Starten Sie das Programm mit $h=0.001$, $x_{\text{End}}=0.2$.
- Starten Sie das Programm mit $h=0.003$, $x_{\text{End}}=0.2$.
- Starten Sie das Programm mit $h=0.004$, $x_{\text{End}}=0.2$.
- Starten Sie das Programm mit $h=0.005$, $x_{\text{End}}=0.2$.

Wie ist das Ergebnis zu interpretieren?

Differentialgleichungen

2) Lösung einer (nichtlinearen) DGL 2. Ordnung (Van-der-Pol-DGL) mit RK 2

Es ist ein C-Programm zu schreiben, welches die folgende Differentialgleichung realisiert (unabh. Variable: t):

$$\ddot{y} = 6 \cdot (1 - y^2) \cdot \dot{y} - y \quad y(0) = 0, \quad \dot{y}(0) = 1$$

Versuchsvorbereitung:

- Geben Sie das Analogrechner-/Simulink-Schaltbild an.
- Geben Sie die DGL 2. Ordnung als 2 DGLn 1. Ordnung an.
- Geben Sie die Iterationsgleichungen für das **Euler**-Verfahren an.
- Geben Sie die Iterationsgleichungen für das **RK2**-Verfahren an.
- Schreiben Sie ein Programm "*VanDerPol.ch*", welches die DGL mit beiden Verfahren löst und in einem Plot anzeigt.

Versuchsdurchführung:

- Starten Sie das Programm mit $h=0.001$, $t_{\text{End}}=31$.
- Starten Sie das Programm mit $h=0.02$, $t_{\text{End}}=31$.

3) Lösung eines Differentialgleichungssystems (Lorenz-Attraktor) mit RK 2

Es ist ein C-Programm zu schreiben, welches das folgende (chaotische) Differentialgleichungssystem realisiert (unabh. Variable: t):

$$\begin{aligned}\dot{x} &= -10 \cdot (x - y) & x(0) &= 0.01, & y(0) &= 0.01, & z(0) &= 0.0 \\ \dot{y} &= (40 - z) \cdot x - y \\ \dot{z} &= x \cdot y - 2.67 \cdot z\end{aligned}$$

Versuchsvorbereitung:

- Geben Sie die Iterationsgleichungen für das **RK2**-Verfahren an.
- Schreiben Sie ein Programm "*Lorenz.ch*", welches das DGL-System löst.
Geben Sie im 1. Plot die Funktion $x(t)$ aus:
Geben Sie im 2. Plot $z(x)$ aus.
- Realisieren Sie das Differentialgleichungssystem mit **MATLab/Simulink**
(Tipp: Funktionen mit Embedded-Matlab-Funktionsblöcken realisieren,
Ausgabe mit toWorkspace, Plotten in Matlab mit plot3(x,y,z)).

Versuchsdurchführung:

- Starten Sie das Programm mit $h=0.002$, $t_{\text{End}}=120$.

Differentialgleichungen

- b) Ändern Sie jetzt in der 2. Gleichung die 40 auf 40.000000001.
Vergleichen Sie $x(t)$ der beiden Simulationen, indem Sie das Ergebnis in einem Plot darstellen.

Vorzubereiten:

Analogrechner-Schaltbilder
Ableitung der Iterationsgleichungen.

Aufgabenbearbeitung:

Fertigzustellen und abzugeben sind

- Analogrechner/Simulink-Schaltbilder
- Iterationsgleichungen
- CH-Programme
- kommentierte Funktionsausdrucke (Copy-to-Clipboard --> Word-Dokument), d.h.
 - Zweck des Versuchs,
 - Versuchsrandbedingungen,
 - Ergebnisausdrucke,
 - kurze Bewertung des Ergebnisses.

Differentialgleichungen

Verwendung der Ch-Sprachumgebung:

Die zu entwickelnden Programme werden in C geschrieben und sollen in der Ch-Sprachumgebung ablaufen. Die Programme müssen nicht kompiliert werden, da die Ch-Sprachumgebung über einen C-Interpreter verfügt.

Die Programme enden mit der Dateierweiterung `.ch`. Sie können direkt durch Doppelclick im Explorer gestartet werden. Im Fehlerfall ist es günstiger das Programm durch Aufruf in einem CMD-Fenster zu starten, da dann die Fehlermeldungen sichtbar bleiben.

Beispielprogramm:

```
#include <math.h>
#include <chplot.h>

// Versuchsparameter
#define H 0.01
#define T_END 1.1

// ----- Funktionsdeklarationen -----
double Fkt_A(double t, double x, double y){
    return sin(2*M_PI*1*t);
}

double Fkt_B(double t, double x, double y){
    return sin(2*M_PI*3*(t-0.25));
}

// ----- Hauptprogramm -----
int main(){
    int steps = T_END/H;
    array double t[steps], x[steps], y[steps];

    class CPlot plot, lissajous; // 2D-Plotter
    int i;

    // Initialisierung
    x[0]=0;
    y[0]=1;
    t[0]=0;

    // Funktionswerte berechnen
    for(i=0; i<steps-1; i++){
        x[i+1]=Fkt_A(t[i], x[i], y[i]);
        y[i+1]=Fkt_B(t[i], x[i], y[i]);
        t[i+1]=t[i]+H;
    }

    // Funktionen zeichnen
    plot.data2D(t,x);
    plot.data2D(t,y);
    plot.legend("Funktion A", 0);
    plot.legend("Funktion B", 1);
    plot.plotting();

    lissajous.data2D(x, y);
    lissajous.plotting();
}
```

