



Fachgebiet  
Simulation und Optimale Prozesse



TECHNISCHE UNIVERSITÄT  
ILMENAU

Fakultät für Informatik und  
Automatisierung  
Institut für Automatisierungs-  
und Systemtechnik

---

## Praktikum

### Versuch SIM-1

# „Numerische Integrationsverfahren zur Lösung von Simulationsaufgaben“

Verantwortlicher Hochschullehrer: Prof. Dr.-Ing. habil. P. Li  
Versuchsverantwortlicher: Dr.-Ing. S. Hopfgarten

---

<b>Name, Vorname</b>	<b>Matrikel-Nr.</b>
<b>Mitarbeiter</b>	
<b>Datum, Note, Unterschrift</b>	

## 1 Versuchsbeschreibung

Das Kernstück der digitalen Simulation zeitkontinuierlicher und zeitdiskreter Systeme – die numerische Integration – soll neben den Ausführungen in den Vorlesungen und den Aufgaben in den Übungen vertieft werden. Vor allem geht es sowohl um das Verständnis der Funktionsweise numerischer Integrationsverfahren als auch um deren Einsatz zur numerischen Lösung von Differentialgleichungssystemen

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), \mathbf{x}(0) = \mathbf{x}_0, t \in [0, t_f], t_f - \text{Endzeit} \quad (1)$$

und die sich anschließende kritische Bewertung der Ergebnisse.

An den Rechnerarbeitsplätzen steht die Software MATLAB/Simulink<sup>1</sup> zur Verfügung, die als Simulationsmittel eingesetzt wird.

## 2 Versuchsvorbereitung

### 2.1 Einteilung numerischer Integrationsverfahren und deren Einsatz

Kategorisieren Sie numerische Integrationsverfahren und erläutern Sie kurz die Merkmale der Unterteilung! Welche Gesichtspunkte berücksichtigen Sie beim Einsatz numerischer Integrationsverfahren?

### 2.2 Numerische Stabilität von Integrationsverfahren

Geben Sie die numerische Stabilitätsbedingung für Integrationsverfahren an und berechnen Sie die Schrittweite, mit der bei Einsatz des impliziten Euler-Verfahrens eine numerisch stabile Lösung für das System mit folgender Differentialgleichungsbeschreibung erzielt wird!

Systembeschreibung:

$$\dot{x}(t) = -\frac{1}{T}x(t), x(0) = x_0 \quad (2)$$

Implizites Euler-Verfahren:

$$x_{k+1} = x_k + hf(x_{k+1}), \quad k - \text{Iterationszähler}, h - \text{zeitliche Schrittweite} \quad (3)$$

### 2.3 Verfahrensanwendungen

Setzen Sie das explizite Euler-Verfahren, das Adams-Bashfort-Verfahren 2. Ordnung und das Heun-Verfahren (Kombination des Adams-Bashfort-Verfahrens 2. Ordnung (Prädiktorschritt) und des Adams-Moulton-Verfahrens (Korrektorschritt)) in einen Ablaufplan bzw. in MATLAB-Programmcode um! Berechnen Sie manuell unter Nutzung des Euler-Verfahrens die ersten drei Werte für das Beispiel

$$\dot{x}(t) = -0.1x(t) + 0.1u(t), x(0) = 0.5$$

bei einer festen Schrittweite  $h = 0.1$  und einem Eingangssignal  $u(t) = 1$  für  $t \geq 0$ !

<sup>1</sup>MATLAB<sup>®</sup> und Simulink<sup>®</sup> sind eingetragene Warenzeichen der The Mathworks Inc.

Die allgemeine Systembeschreibung stellt sich wie in (1) dar. Die Rechenverfahren der anzuwendenden Verfahren lauten:

Explizites Euler-Verfahren:

$$x_{k+1} = x_k + hf(x_k, u_k) \quad k - \text{Iterationszähler, } h - \text{zeitliche Schrittweite} \quad (4)$$

Adams-Bashfort-Verfahren 2. Ordnung:

$$x_{k+1} = x_k + \frac{h}{2} (3f(x_k, u_k) - f(x_{k-1}, u_{k-1})) \quad (5)$$

Heun-Verfahren 2. Ordnung:

$$x_{k+1}^P = x_k + \frac{h}{2} (3f(x_k, u_k) - f(x_{k-1}, u_{k-1})) \quad (6)$$

$$\dot{x}_{k+1}^P = f(x_{k+1}^P, u_{k+1}) \quad (7)$$

$$x_{k+1}^{(K)} = x_k + \frac{h}{2} (f(x_{k+1}^P, u_{k+1}) + f(x_k, u_k)) \quad (8)$$

## 2.4 Systembeschreibung

Stellen Sie die vereinfachte mathematische Systembeschreibung für das folgende Beispiel der Vertikalbewegung eines Kraftfahrzeugs auf! Die Abb. 1 zeigt ein Zwei-Massen-Modell mit Feder- und Dämpferelement ( $c_A$  und  $d_A$ ) zwischen der Aufbaumasse  $m_A$  und der Radmasse  $m_R$ . Der Reifen hat die Federkonstante  $c_R$ . Die Reifendämpfung wird vernachlässigt. In der Ruhelage ist die Wirkung der Schwerkraft durch die Vorspannung der Federelemente bereits ausgeglichen. Die Bewegung des Systems wird mit den vertikalen Auslenkungen  $x_A$  und  $x_R$  aus der Ruhelage beschrieben. Als Eingangsgröße tritt die Boden- bzw. Fahrbahnwelligkeit  $h(t)$  auf. Benutzen Sie das zweite Newtonsche Gesetz zum Aufstellen der Differentialgleichungen

$$ma(t) = \sum_{i=1}^n F_i \quad (9)$$

mit  $m$  - Masse,  $a(t)$  - Beschleunigung,  $F_i(t)$  -  $i$ -te eingeprägte Kraft.

Hinweise:  $F_C(t) = cx(t)$ ,  $F_D(t) = d\dot{x}(t)$ ,  $F_{C_R}(t) = -F_{C_A}(t)$ ,  $F_{D_R}(t) = -F_{D_A}(t)$ .

Die Konstanten lauten:  $m_A = 365 \text{ kg}$ ,  $m_R = 40 \text{ kg}$ ,  $c_A = 1.5 \cdot 10^4 \text{ kg s}^{-2}$ ,  $c_R = 1.6 \cdot 10^5 \text{ kg s}^{-2}$ ,  $d_A = 2.5 \cdot 10^3 \text{ kg s}^{-1}$ .

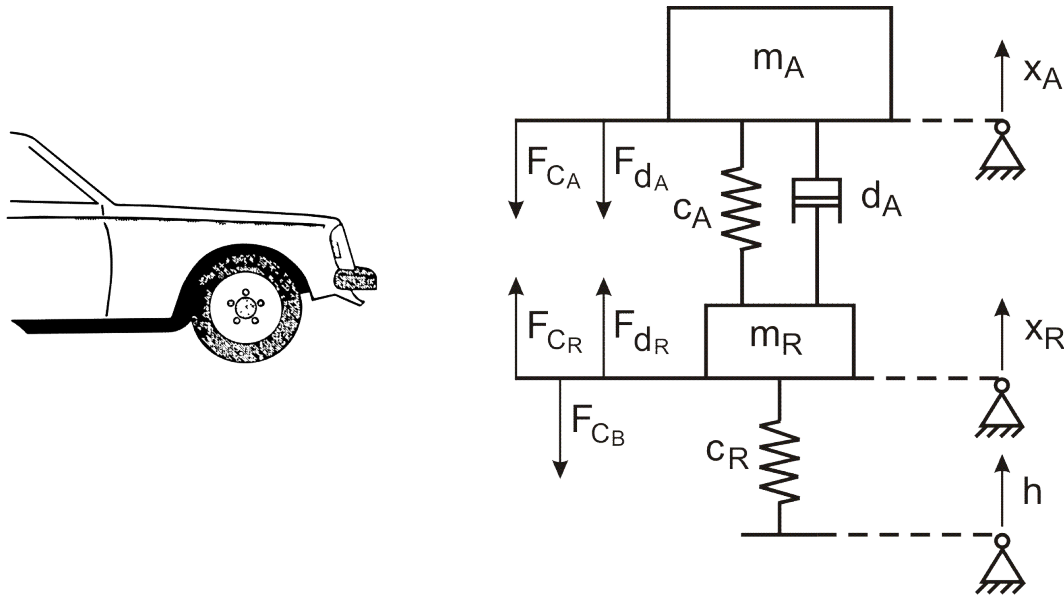


Abbildung 1: Federung und Dämpfung an einem Kraftfahrzeug nach [1] und [8]

### 3 Versuchsdurchführung

#### 3.1 Verfahrensimplementierung, -anwendung und -vergleich

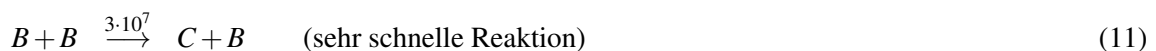
Programmieren Sie die drei Verfahren nach Abschnitt 2.3 mit Hilfe der MATLAB-Kommandosprache! Berechnen Sie jeweils den Zeitverlauf  $x(t)$  für das Beispiel aus Abschnitt 2.3 numerisch und stellen Sie die Ergebnisse tabellarisch und grafisch dar! Vergleichen Sie die Verfahren und Ergebnisse miteinander! Variieren Sie die Schrittweite  $h = \{0.01, 0.1, 0.5, 1\}$  und beurteilen Sie die Ergebnisse der Simulation!

#### 3.2 Einsatz von Verfahren mit Schrittweitensteuerung

- Hinterlegen Sie die Systembeschreibung nach Abschnitt 2.3 in einem M-File! Nutzen Sie die MATLAB-Integrationsroutine `ode45` und vergleichen Sie mit den vorherigen Ergebnissen!
- Wenden Sie anschließend `ode45` auf das Beispiel aus Abschnitt 2.4 an bei einem Eingangssignal  $u(t) = 0.1 \sin(2\frac{\pi}{T}t)$  für  $t \in [0, T]$ ,  $T \in \{3.6, 1.2\}$ , sonst ist  $u(t) = 0$ ! Die Simulation soll über einen Zeitraum von 30 s erfolgen. Erläutern Sie die Ergebnisse!

#### 3.3 Chemische Reaktionskinetik

Folgendes Beispiel einer Reaktionskinetik



führt zu diesem Differentialgleichungssystem:

$$\dot{x}_1(t) = -0.04x_1(t) + 10^4 x_2(t)x_3(t) \quad (13)$$

$$\dot{x}_2(t) = 0.04x_1(t) - 10^4 x_2(t)x_3(t) - 3 \cdot 10^7 x_2^2(t) \quad (14)$$

$$\dot{x}_3(t) = 3 \cdot 10^7 x_2^2(t), \quad \mathbf{x}(0) = \mathbf{x}_0 = [1 \ 0 \ 0]^T$$

Simulieren Sie das Systemverhalten mit geeigneten numerischen Integrationsverfahren und bewerten Sie auftretende Effekte!

### 3.4 Räuber-Beute-Populationsdynamik

Eine Raubtierpopulation (Bestandsgröße  $x_1(t)$ ) existiert mit zwei weiteren, miteinander um die Nahrungsgrundlage konkurrierenden Tierpopulationen  $x_2(t)$  und  $x_3(t)$  in einem gemeinsamen Ökosystem. Dieses System kann vereinfacht durch folgendes Differentialgleichungssystem beschrieben werden:

$$\dot{x}_1(t) = -qx_1(t) + \mu x_1(t)x_2(t) + \nu x_1(t)x_3(t) \quad (15)$$

$$\dot{x}_2(t) = rx_2(t) \left(1 - \frac{x_2(t)}{K}\right) - \alpha x_2(t)x_3(t) - \gamma x_1(t)x_2(t) \quad (16)$$

$$\dot{x}_3(t) = sx_3(t) \left(1 - \frac{x_3(t)}{L}\right) - \beta x_2(t)x_3(t) - \delta x_1(t)x_3(t), \quad \mathbf{x}(0) = \mathbf{x}_0 = [100 \ 100 \ 100]^T$$

mit  $q = r = s = 1$  – Sterberate der Räuberpopulation  $x_1(t)$  bzw. Wachstumsraten der Beutepopulationen  $x_2(t)$  und  $x_3(t)$ ;  $K = L = 1000$  – Sättigung für  $x_2(t)$  und  $x_3(t)$ ;  $\alpha = 0.0015$  – Konkurrenzeffekt auf  $x_2(t)$ ;  $\beta = 0.001$  – Konkurrenzeffekt auf  $x_3(t)$ ;  $\gamma = 0.001$  – Raubbauparameter für  $x_2(t)$ ;  $\delta = 0.015$  – Raubbauparameter für  $x_3(t)$ ;  $\mu = 0.0005$  – Zunahme der Räuber wegen  $x_2(t)$ ;  $\nu = 0.002$  – Zunahme der Räuber wegen  $x_3(t)$

Untersuchen Sie das Verhalten dieser Populationsdynamik über einen Zeithorizont von 1000 Zeiteinheiten unter dem oben angegebenen Anfangszustand! Illustrieren Sie sich das Stabilitätsverhalten! Zu welchen Erkenntnissen kommen Sie hinsichtlich Zeitverhalten und Stabilität?

Variieren Sie nun die Anfangszustände und führen Sie Simulationsrechnungen durch!

Welches Systemverhalten bewirken Parameterwerte  $\nu = \{0.003 \ 0.0035 \ 0.004 \ 0.0045\}$  ?

## Literatur

- [1] C. Becker, L. Litz, G. Siffing. Regelungstechnik Übungsbuch. Hüthig. 1993
- [2] M. J. Canty. Chaos und Systeme. Vieweg. 1995
- [3] F. E. Cellier. Continuous System Simulation. Springer. 2006
- [4] E. Hairer, S.P. Nørsett, G. Wanner. Solving Ordinary Differential Equations I. Springer. 1987
- [5] E. Hairer, G. Wanner. Solving Ordinary Differential Equations II. Springer. 1991
- [6] S. Hopfgarten. Vorlesung Simulation. TU Ilmenau
- [7] J. Kahlert. Simulation technischer Systeme. Vieweg. 2004
- [8] R.L. Woods, K.L. Lawrence. Modeling and Simulation of Dynamic Systems. Prentice Hall. 1997