TECHNISCHE UNIVERSITÄT DRESDEN

Bauingeniuerwesen Institut für Statik und Dynamik

Beleg 1

BIW4-05 SIMULATION DYNAMISCHER SYSTEME

Philipp Göbel

Matrikelnummer: 4607083
Immatrikulationsjahr: 2018

21. Mai 2023

Betreuer

Ines Wollny
Jakob Platen

Betreuender Hochschullehrer

Univ.-Prof. Dr.-Ing. habil. Michael Kaliske

Inhaltsverzeichnis

1 Lösung des Beleges 5

2 Julia-Code 9

Technische Universität Dresden
Fakultät Bauingenieurwesen

Institut für Statik und Dynamik der Tragwerke
Univ.-Prof. Dr.-Ing. habil. M. Kaliske

1. Beleg	**BIW4-05 SIMULATION DYNAMISCHER SYSTEME**	**SS 2023**
Thema	Elastisch gebetteter Balken	
	Bauingenieurwesen	8. Semester

Gegeben ist das als Balken auf elastischer Bettung idealisierte Modell einer Schiene (siehe Abbildung 1) mit den zugehörigen Materialparametern:

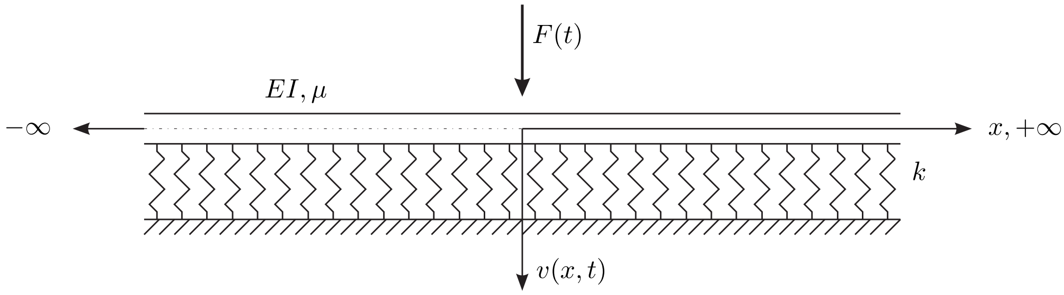

Abbildung 1: Elastisch gebetteter Balken

$$E = 21 \cdot 10^{10}\,\frac{\mathrm{N}}{\mathrm{m}^2}, \quad I = 3055\,\mathrm{cm}^4, \quad \mu = 60\,\frac{\mathrm{kg}}{\mathrm{m}}, \quad k = 3 \cdot 10^6\,\frac{\mathrm{N}}{\mathrm{m}^2}\,.$$

Die dynamische Steifigkeit $K(\eta)$ des Balkens kann analytisch abgeleitet werden:

$$\hat{F}_0 = \overline{K}(\eta)\,\hat{v}_0, \quad \overline{K}(\eta) = 8\,EI\,W^3, \quad W = \frac{\sqrt{2}}{2}\sqrt[4]{\frac{k}{EI}\,(1-\eta^2)}, \quad \eta^2 = \Omega^2\,\frac{\mu}{k}.$$

Für $\eta > 1{,}0$ gilt $\sqrt[4]{-1} = \frac{\sqrt{2}}{2}\,(1+i)$. Die Koeffizienten einer gebrochenrationalen Approximation der Steifigkeit $\tilde{K}(\omega)$,

$$\overline{K}(\Omega) \approx \tilde{K}(\Omega) = \frac{P_0 + i\Omega\,P_1 + (i\Omega)^2\,P_2 + \ldots + (i\Omega)^5\,P_5}{1 + i\Omega\,Q_1 + \ldots + (i\Omega)^4\,Q_4}\,,$$

wurden mit Hilfe der Fehlerquadratmethode bereits ermittelt:

$$\begin{aligned}
P_0 &= 10337706{,}5\ \mathrm{N\,m^{-1}}\\
P_1 &= 48744{,}172\ \mathrm{N\,m^{-1}\,s^{1}}\\
P_2 &= 455{,}205728\ \mathrm{N\,m^{-1}\,s^{2}}\\
P_3 &= 1{,}13950676\ \mathrm{N\,m^{-1}\,s^{3}}\\
P_4 &= 0{,}00483245447\ \mathrm{N\,m^{-1}\,s^{4}}\\
P_5 &= 3{,}11054119 \cdot 10^{-6}\ \mathrm{N\,m^{-1}\,s^{5}}
\end{aligned}$$

$$
\begin{aligned}
Q_1 &= 0{,}00490648418 \ \mathrm{s} \\
Q_2 &= 2{,}71147593 \cdot 10^{-5} \ \mathrm{s}^2 \\
Q_3 &= 5{,}42912842 \cdot 10^{-8} \ \mathrm{s}^3 \\
Q_4 &= 6{,}94958499 \cdot 10^{-12} \ \mathrm{s}^4
\end{aligned}
$$

Aufgabenstellung:

1. Überführen Sie die gebrochenrationale Steifigkeitsbeziehung in ein System von linearen Gleichungen in $(i\Omega)$: $\quad i\Omega\,\mathbf{A}\,\hat{\mathbf{z}} + \mathbf{B}\,\hat{\mathbf{z}} = \mathbf{r}$.

2. Transformieren Sie das System in den Zeitbereich.

3. Zum Zeitpunkt $t = 0$ befindet sich das System in Ruhe und die Durchbiegung v_0 ist gleich Null. Ermitteln Sie numerisch eine Lösung im Zeitbereich für die Durchbiegung $v_0(t)$ infolge der gegebenen Erregung für $0 \leq t \leq 1\,\mathrm{s}$.

$$v_0(t) = v(x = 0, t)$$

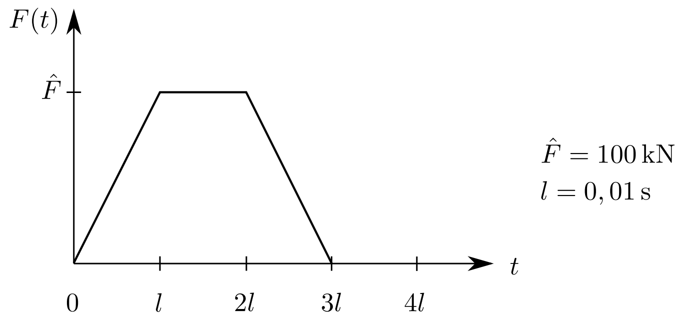

Abbildung 2: Erregung

1 Lösung des Beleges

Mithilfe der bereits gegebenen Koeffizienten einer gebrochenrationalen Approximation der Steifigkeit und des Arbeitsblattes aus Übung 1 kann die Steifigkeitsbeziehung in ein System von linearen Gleichungen $i\Omega\mathbf{A}\hat{\mathbf{z}} + \mathbf{B}\hat{\mathbf{Z}} = \mathbf{r}$ überführt werden.

$$
\begin{aligned}
\lambda^5 :&\quad s_1^{(0)}Q_4^{(0)} &&= P_5^{(0)} \rightarrow s_1^{(0)};\\
\lambda^4 :&\quad s_0^{(0)}Q_4^{(0)} + s_1^{(0)}Q_3^{(0)} &&= P_4^{(0)} \rightarrow s_0^{(0)};\\
\lambda^3 :&\quad s_0^{(0)}Q_3^{(0)} + s_1^{(0)}Q_2^{(0)} + r_3^{(0)} &&= P_3^{(0)} \rightarrow r_3^{(0)};\\
\lambda^2 :&\quad s_0^{(0)}Q_2^{(0)} + s_1^{(0)}Q_1^{(0)} + r_2^{(0)} &&= P_2^{(0)} \rightarrow r_2^{(0)};\\
\lambda^1 :&\quad s_0^{(0)}Q_1^{(0)} + s_1^{(0)} &&= P_1^{(0)} \rightarrow r_1^{(0)};\\
\lambda^0 :&\quad s_0^{(0)} + r_0^{(0)} &&= P_0^{(0)} \rightarrow r_0^{(0)}.
\end{aligned}
$$

$$
\begin{aligned}
\lambda^4 :&\quad s_1^{(1)}r_3^{(0)} &&= Q_4^{(0)} \rightarrow s_1^{(1)};\\
\lambda^3 :&\quad s_0^{(1)}r_3^{(0)} + s_1^{(1)}r_2^{(0)} &&= Q_3^{(0)} \rightarrow s_0^{(1)};\\
\lambda^2 :&\quad s_0^{(1)}r_2^{(0)} + s_1^{(1)}r_1^{(0)} + r_2^{(1)} &&= Q_2^{(0)} \rightarrow r_2^{(1)};\\
\lambda^1 :&\quad s_0^{(1)}r_1^{(0)} + s_1^{(1)}r_0^{(0)} + r_1^{(1)} &&= Q_1^{(0)} \rightarrow r_1^{(1)};\\
\lambda^0 :&\quad s_0^{(1)}r_0^{(0)} \quad + r_0^{(1)} &&= Q_0^{(0)} \rightarrow r_0^{(1)}.
\end{aligned}
$$

$$
\begin{aligned}
\lambda^3 :&\quad s_1^{(2)}r_2^{(1)} &&= r_3^{(0)} \rightarrow s_1^{(2)};\\
\lambda^2 :&\quad s_0^{(2)}r_2^{(1)} + s_1^{(2)}r_1^{(1)} &&= r_2^{(0)} \rightarrow s_0^{(2)};\\
\lambda^1 :&\quad s_0^{(2)}r_1^{(1)} + s_1^{(2)}r_0^{(1)} + r_1^{(2)} &&= r_1^{(0)} \rightarrow r_1^{(2)};\\
\lambda^0 :&\quad s_0^{(2)}r_0^{(1)} + r_0^{(2)} &&= r_0^{(0)} \rightarrow r_0^{(2)}
\end{aligned}
$$

$$
\begin{aligned}
\lambda^2 :&\quad s_1^{(3)}r_1^{(2)} &&= r_2^{(1)} \rightarrow s_1^{(3)};\\
\lambda^1 :&\quad s_0^{(3)}r_1^{(2)} + s_1^{(3)}r_0^{(2)} &&= r_1^{(1)} \rightarrow s_0^{(3)};\\
\lambda^0 :&\quad s_0^{(3)}r_0^{(2)} + r_0^{(3)} &&= r_0^{(1)} \rightarrow r_0^{(3)};
\end{aligned}
$$

$$
\begin{aligned}
s_0^{(4)} &= \frac{r_0^{(2)}}{r_0^{(3)}}\\
s_1^{(4)} &= \frac{r_1^{(2)}}{r_0^{(3)}}
\end{aligned}
$$

Dabei ist $\hat{f}_0 = 0$, weil es keine Anfangsbelastung gibt.

Aus den Beziehungen lassen sich die Matrizen $\mathbf{A}, \mathbf{B}, \mathbf{f}$ wie folgt belegen:

$$\mathbf{A} = \begin{bmatrix} s_1^{(0)} & 0 & 0 & 0 & 0 \\ 0 & -s_1^{(1)} & 0 & 0 & 0 \\ 0 & 0 & s_1^{(2)} & 0 & 0 \\ 0 & 0 & 0 & -s_1^{(3)} & 0 \\ 0 & 0 & 0 & 0 & s_1^{(4)} \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} s_0^{(0)} & 0 & 0 & 0 & 0 \\ 0 & -s_0^{(1)} & 0 & 0 & 0 \\ 0 & 0 & s_0^{(2)} & 0 & 0 \\ 0 & 0 & 0 & -s_0^{(3)} & 0 \\ 0 & 0 & 0 & 0 & s_0^{(4)} \end{bmatrix}$$

$$\hat{\mathbf{f}} = \begin{bmatrix} \hat{f}_0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\mathbf{A} = \begin{bmatrix} 447586.61 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & -4.92 \times 10^{-14} & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & -7.92 \times 10^{8} & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & -2.25 \times 10^{-13} & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 8.68 \times 10^{7} \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} -2.80 \times 10^{9} & 1.0 & 0.0 & 0.0 & 0.0 \\ 1.0 & -3.58 \times 10^{-10} & -1.0 & 0.0 & 0.0 \\ 0.0 & -1.0 & -3.20 \times 10^{11} & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 & 1.75 \times 10^{-13} & -1.0 \\ 0.0 & 0.0 & 0.0 & -1.0 & 1.06 \times 10^{10} \end{bmatrix}$$

$$\hat{\mathbf{f}} = \begin{bmatrix} 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \end{bmatrix}$$

Um die numerische Lösung im Zeitbereich für die Durchbiegung $v_0(t)$ infolge der gegebenen Erregung zu ermitteln, muss die Lösung in Zeitschritten integriert werden.
Zunächst muss das System von dem Frequenzbereich in den Zeitbereich transformiert werden. Für den Frequenzbereich gilt:

$$\lambda\, \mathbf{A}\, \hat{\mathbf{z}} + \mathbf{B}\, \hat{\mathbf{z}} = \hat{\mathbf{F}} \tag{1.1}$$

$$\mathbf{z}(t) = \hat{\mathbf{z}} \cdot e^{i\omega t} = \hat{\mathbf{z}} e^{\lambda t} \tag{1.2}$$

$$\mathbf{F}(t) = \hat{\mathbf{F}} \cdot e^{i\omega t} = \hat{\mathbf{F}} e^{\lambda t} \tag{1.3}$$

$$\tag{1.4}$$

Die Ableitung der Zeit ergibt:

$$\dot{\mathbf{z}}(t) = \lambda \hat{\mathbf{z}} e^{\lambda t} \tag{1.5}$$

Daraus folgt:

$$\mathbf{A}\, \dot{\mathbf{z}}(t) + \mathbf{B}\, \mathbf{z}(t) = \mathbf{F}(t) \tag{1.6}$$

Hierbei sind die Matrizen **A** und **B** zeitlich konstante Matrizen.

Nun zur Lösung durch Integration

$$\int [\mathbf{A}\, \dot{\mathbf{z}}(t) + \mathbf{B}\, \mathbf{z}(t) = \mathbf{F}(t)]\, dt = \int \mathbf{F}(t)\, dt \tag{1.7}$$

Die lineare Ansatzfunktion ergibt sich für den Bereich $t_0 \leq t \leq t_1$ zu:

$$\mathbf{z}(t) \approx \mathbf{z}_0 \left(1 - \frac{\tau}{\Delta t}\right) + \mathbf{z}_1 \left(\frac{\tau}{\Delta t}\right) \tag{1.8}$$

Jetzt wird die Ableitung gebildet und in (1.7) eingesetzt:

$$\hat{\mathbf{z}}(t) \approx \mathbf{z}_0(t)\, \frac{1}{\Delta t} + \mathbf{z}_1(t)\, \frac{1}{\Delta t} \tag{1.9}$$

$$\int_0^{\Delta t} \left[\mathbf{A} \cdot (-\mathbf{z}_0 + \mathbf{z}_1)\frac{1}{\Delta t} + \mathbf{B}\left[\mathbf{z}_0(1 - \frac{\tau}{\Delta t}) + \mathbf{z}_1(\frac{\tau}{\Delta t})\right]\right] d\tau = \int_0^{\Delta t} = \mathbf{F}(t) d\tau \tag{1.10}$$

$$\left(\mathbf{A} + \frac{\Delta t}{2}\mathbf{B}\right) \mathbf{z}_1 + \left(\frac{\Delta t}{2}\mathbf{B} - \mathbf{A}\right) \mathbf{z}_0 = \int_0^{\Delta t} \mathbf{F}(t) d\tau \tag{1.11}$$

nach Umstellen auf **z**, gilt für einen Zeitschritt:

$$\mathbf{z}_1 = (\mathbf{A} + \frac{\Delta t}{2}\mathbf{B})^{-1} \cdot \left[\int_0^{\Delta t} F(t) d\tau - \left(\frac{\Delta t}{2}\mathbf{B} - \mathbf{A}\right) \mathbf{z}_0\right] \tag{1.12}$$

Für die Integration von $\mathbf{F}(t)$ gilt:

$$\int_0^{\Delta t} \mathbf{F}(t) d\tau = \frac{\mathbf{F}_0 + \mathbf{F}_1}{2} \Delta t \tag{1.13}$$

Dabei ist für die Integration zu beachten, dass $F(t)$ auch als Vektor $\mathbf{F}(t)$ darzustellen ist. Die erste Stelle des Vektors wird mit $\hat{F} = 100kN$ befüllt und der Rest ist null.

Für die Berechnung von z_1 habe ich Δt = 0.0001s gewählt.
Als Ergebnis für z_1 komme ich auf folgendes Ergebnis.

$$z_1 = \begin{bmatrix} 0.00044791129649929745 \\ 1.2275350723841512e6 \\ -1.5290166828771846e-6 \\ 19605.72310506311 \\ 8.202975286262283e-6 \end{bmatrix} \tag{1.14}$$

Der erste Wert von z_1 stellt die gesuchte Größe v_0 dar.

Somit ist das Endergebnis:

$$v_0(t) = 0.00044791129649929745\frac{m}{s} \tag{1.15}$$

Der Verlauf von z_1 ist im folgenden Diagramm über den Zeitraum von $0 \leq t \leq 1$ dargestellt.

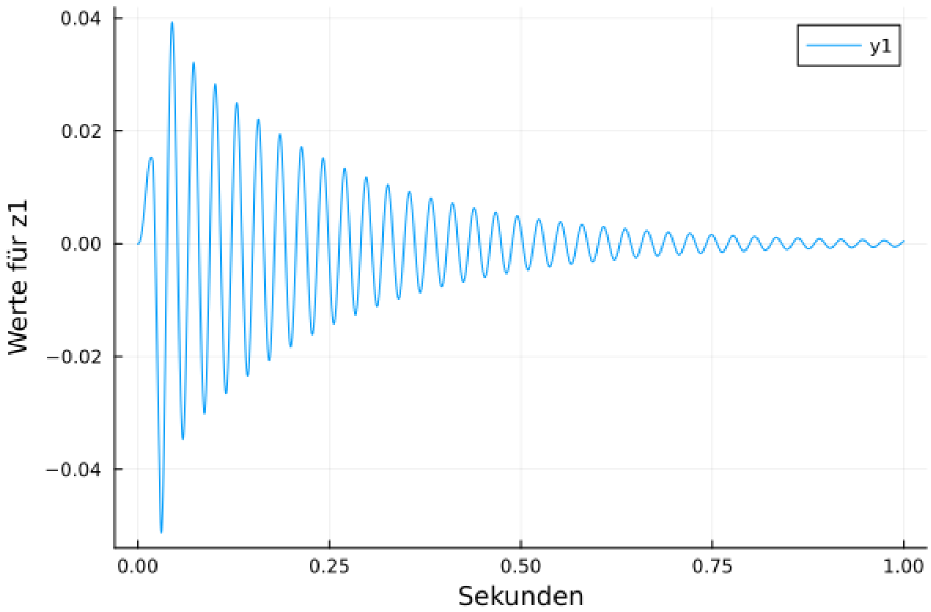

Abbildung 1.1: Verlauf von z_1 über $0 \leq t \leq 1s$

Es ist deutlich die Belastung infolge von $F(t)$ zu sehen, die dann mit der Zeit abklingt.

2 Julia-Code

```julia
using Plots

#  Koeffizienten einer gebrochenrationalen Approximation der Steifigkeit
P0 = 10337706.5
P1 = 48744.172
P2 = 455.205728
P3 = 1.13950676
P4 = 0.00483245447
P5 = 3.11054119e-6

Q0 = 1
Q1 = 0.00490648418
Q2 = 2.71147593e-5
Q3 = 5.42912842e-8
Q4 = 6.94958499e-12

################################################################################



#Abspaltung

s1_0 = P5/Q4
s0_0 = (P4-s1_0*Q3)/Q4
r3_0 = P3-s0_0*Q3-s1_0*Q2
r2_0 = P2-s0_0*Q2-s1_0*Q1
r1_0 = P1-s0_0*Q1-s1_0
r0_0 = P0-s0_0


s1_1 = Q4/r3_0
s0_1 = (Q3-s1_1*r2_0)/r3_0
r2_1 = Q2-s0_1*r2_0-s1_1*r1_0
r1_1 = Q1-s0_1*r1_0-s1_1*r0_0
r0_1 = Q0-s0_1*r0_0

```

```

s1_2 = r3_0/r2_1
s0_2 = (r2_0-s1_2*r1_1)/r2_1
r1_2 = r1_0-s0_2*r1_1-s1_2*r0_1
r0_2 = r0_0-s0_2*r0_1


s1_3 = r2_1/r1_2
s0_3 = (r1_1-s1_3*r0_2)/r1_2
r0_3 = r0_1-s0_3*r0_2


s0_4 = r0_2/r0_3
s1_4 = r1_2/r0_3


f0 = 0


A = [s1_0 0 0 0 0;
     0 -s1_1 0 0 0;
     0 0 s1_2 0 0;
     0 0 0 -s1_3 0;
     0 0 0 0 s1_4]

B = [s0_0 1 0 0 0;
     1 -s0_1 -1 0 0;
     0 -1 s0_2 1 0;
     0 0 1 -s0_3 -1;
     0 0 0 -1 s0_4]



r = [0;0;0;0;0]

println(A)
println(B)
println(r)



################################################################################


#Zeitschrittl nge wird hier im Skript mit t anstatt delta t beschrieben
t = 0.0001 #s

#Anzahl der Iteratiosschritte
# Die Anzahl der Iterationsschritte ergeben sich aus der G r   e von t


```

```
j_ges = Int(1/t+1)
j_Schritt = Int(1/t)
println("Anzahl der Zeitschritte: ",j_Schritt)

#Bef llen der Anfangsarrays
z0 = [0;0;0;0;0]

#Array f r die Darstellung des Plots
z3 = zeros(j_ges)

# Erstellen eines Arrays  ber  die Zeitschritte, f r in die Integration
j = zeros(j_ges, 1)
for i in 1:j_ges
    j[i, 1] = (i - 1) * t
end



# Funktionen von F(t)
# Diese sind in 3 Teilen unterteilt, wie aus der Grafik in der Aufgabenstellung zu er
function F1(k)
     return [1;0;0;0;0]*100000/0.01*k
end

F2 = [1;0;0;0;0]*100000

function F3(k)
     return [1;0;0;0;0]* -100000/0.01*k
end


# Integration  ber  den Gesamten Zeitbereich von 1 Sekunden
# Dabei wird mit der if Anweisung geschaut, in welchem Bereichen des Graphen man sich
for i =1:j_ges

     if i == 1
          z1 = (A+0.001/2*B)^-1 * (F1(j[i])+F1(j[i+1]))/2*t
          global z0 = z1
     end

     if i>1 && i<=0.01*j_Schritt
          z1 = inv((A+t/2*B)) * ((F1(j[i])+F1(j[i+1]))/2*t-(t/2*B-A)*z0)
          global z0 = z1
     end

     if i>0.01*j_Schritt && i<=0.02*j_Schritt
          z1 = inv((A+t/2*B)) * (F2*t-(t/2*B-A)*z0)
          global z0 = z1
     end

     if i>0.02*j_Schritt && i<=0.03*j_Schritt
```

```
        z1 = inv((A+t/2*B)) * ((F3(j[i])+F3(j[i+1]))/2*t-(t/2*B-A)*z0)
        global z0 = z1
    end

    if i>0.03*j_Schritt
        z1 = inv(A+t/2*B) * -(t/2*B-A)*z0
        global z0 = z1
    end

    z3[i] = z1[1]

end

#Ausgabe der Ergebnisse
println("L sung f r z1")
println(z0)

plot(j, z3, xlabel = "Sekunden", ylabel = "Werte f r z1")
savefig("plot.png")
```