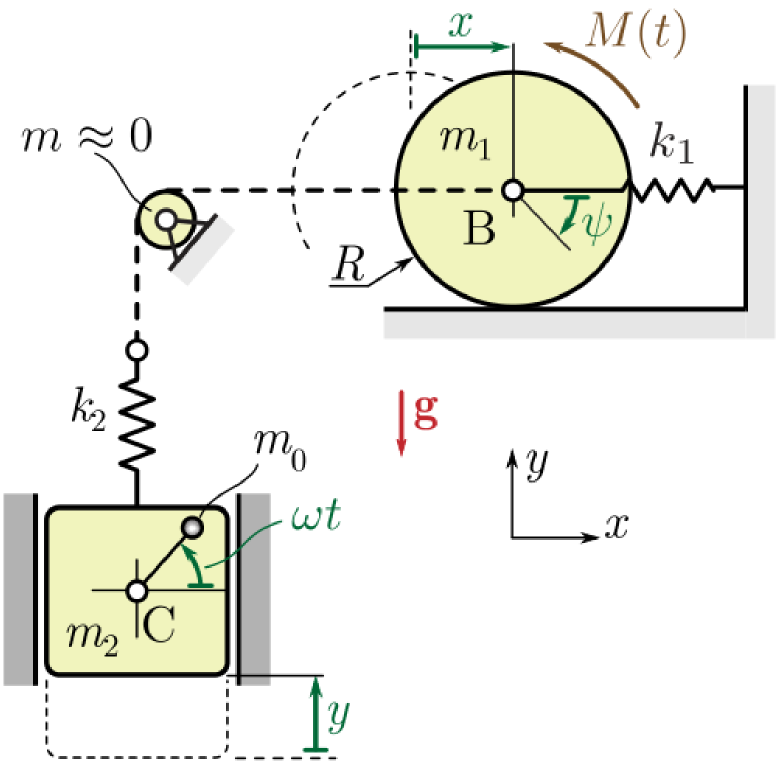
13. Gyakorlat - Rezgésgerjesztő

2021.05.05.

Feladat:



A mellékelt ábrán egy két szabadságfokú rendszer látható, melyet két merev test alkot: egy m_1 tömegű, R sugarú tárcsa és egy m_2 tömegű test. A tárcsa vízszintes talajon gördül és a tömegközéppontja egy k_1 merevségű rugóval van a környezethez rögzítve. A másik test a gravitációs térben van és függőlegesen mozog egy súrlódás mentes megveztés mentén, miközben a k_2 merevségű rugóhoz van rögzítve. A k_2 rugó másik vége egy idális kötélhez csatlakozik, ami egy idális (súrlódásmentes/tömeg nélküli) csigán keresztül a tárcsa tömegközéppontjához van rögzítve. A kötél végig feszített állapotban van.

Adatok:

m_0 = 0.1 kg	R = 0.3 m
m_1 = 1 kg	e = 0.01 m
m_2 = 3 kg	M_0 = 3 Nm
k_1 = 100 N/m	ω = 30 rad/s
k_2 = 200 N/m	ε = π/6 rad/s^2

Részfeladatok:

1. Írja fel a lineáris mátrix együtthatós mozgásegyenletet!

2. Határozza meg a mozgástörvény állandósult állapotbeli tagját!
3. Mekkora a k_2 merevségű rugóban ébredő erő legnagyobb értéke az állandósult állapotban?
4. Határozza meg a sajátkörfrekvenciákat és a hozzátartozó sajátvektorokat!

Megoldás:

1. Feladat:

Kis elmozdulások esetén a lineáris mozgásegyenlet mátrixegyütthatós alakja a következő egyenlettel adható meg

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} = \mathbf{Q}^*,$$

ahol $\mathbf{q}$ az általános koordináták vektora, $\mathbf{M}$ a tömegmátrix, $\mathbf{C}$ a csillapítási mátrix, $\mathbf{K}$ a merevségi mátrix, a $\mathbf{Q}^*$ pedig az általános erők vektora. (Disszipatív energia nincs a rendszerben ezért a csillapítási mátrix zérus lesz.) Első lépésként az általános kkordinátákat kell meghatározni. A rendszer 2 szabadsági fokú, tehát két általános koordinátát kell definiálin, melyből az egyik az ábra alapján legyen a merev test y irányú elmozdulása a másik pedig a tárcsa ψ szögelfordulása

$$\mathbf{q} = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} y \\ \psi \end{bmatrix}.$$

In [1]:

```python
import sympy as sp
from IPython.display import display, Math

sp.init_printing()
```

In [2]:

```python
## Függvények, szimbólumok definiálása

m0, m1, m2, R, e, k1, k2, M0, ω, ε, g = sp.symbols("m0, m1, m2, R, e, k1, k2, M0, ω, ε,

# Készítsünk behelyettesítési listát az adatok alapján, SI-ben
adatok = [(m0, 0.1), (m1, 1), (m2, 3), (R, 0.2),
          (e, 0.01), (k1, 100), (k2, 200), (M0, 3),
          (ω, 30), (ε, sp.pi/6), (g, 9.81)]

# általános koordináták
t = sp.symbols("t", real=True, positive=True)
y = sp.Function('y')(t)
ψ = sp.Function('ψ')(t)

# gerjesztés
M_t = M0*sp.cos(ω*t+ε)
```

In [3]:

```
### Kinetikus energia, potenciális energia, disszipatív energia

### Először fejezzük ki a mennyiségeket az általános koordinátákkal
# B pont sebessége
vB = R*ψ.diff(t)

# 1-es test szögsebessége
ω1 = ψ.diff(t)

# C pont sebessége
vC = y.diff(t)

# Tárcsa tehetetlenségi nyomatéka a B pontra
ΘB = sp.Rational(1,2)*m1*R**2

# m0 tömeg sebessége (helyvektor deriváltja)
konst = sp.symbols("konst") # konstans tag (deriválás után kiesik a kifejezésből)
r0 = sp.Matrix([[e*sp.cos(ω*t)+konst],[y + e*sp.sin(ω*t)+konst]])
v0 = r0.diff(t)

# tárcsa x irányú elmozdulása
x = R*ψ

## Kinetikus energia
T = (sp.Rational(1,2)*m1*vB**2 + sp.Rational(1,2)*ΘB*ω1**2 +
     sp.Rational(1,2)*m2*vC**2 + sp.Rational(1,2)*m0*v0.dot(v0)).expand().trigsimp().si

display(Math('T = {}'.format(sp.latex(T))))

## Potenciális energia
U = sp.Rational(1,2)*k1*(x)**2 + sp.Rational(1,2)*k2*(x-y)**2+m0*g*e*sp.sin(ω*t)

display(Math('U = {}'.format(sp.latex(U))))

## Disszipatív energia most nincs!
```

$$T = \frac{3R^2 m_1 \left(\frac{d}{dt}\psi(t)\right)^2}{4} + \frac{e^2 m_0 \omega^2}{2} + e m_0 \omega \cos(t\omega)\frac{d}{dt}y(t) + \frac{m_0 \left(\frac{d}{dt}y(t)\right)^2}{2} + \frac{m_2 \left(\frac{d}{dt}y(t\right.}{2}$$

$$U = \frac{R^2 k_1 \psi^2(t)}{2} + e g m_0 \sin(t\omega) + \frac{k_2 (R\psi(t) - y(t))^2}{2}$$

In [4]:

```
### Mátrix együtthatók legenerálása
""" A tömegmátrix most nem számítható közvetlenül a kinetikus energiából,
mert az excentrikus tag forgása egy álatlános erő tagot is eredményez,
ami a parciális deriválásnál kiesne az egyenletből.
Ilyen esetben a másodfajú Lagrange-egyenletet kell használni
"""
# Állítsuk elő a Lagrange-egyenletben szereplő deriváltakat
# Ehhez rendezzük listába az általános koordinátákat
q = [y, ψ]
# Majd hozzunk létre egy 2 dimenziós nullvektort a 2 Lagrange egyenlet első két tagján
Mat = sp.zeros(2,1)
for i in range(2):
    Mat[i] = (T.diff((q[i]).diff(t))).diff(t)-T.diff(q[i])
display(Mat)
```

$$\begin{bmatrix} -em_0\omega^2\sin(t\omega) + m_0\frac{d^2}{dt^2}y(t) + m_2\frac{d^2}{dt^2}y(t) \\ \frac{3R^2m_1\frac{d^2}{dt^2}\psi(t)}{2} \end{bmatrix}$$

Ebből a kétdimenziós rendszerből már könnyen kifejezhető a tömegmátrix és az általános erővektor tagja is, mivel erre az kifejezésre az alábbi írható fel (Lagrange alapján)

$$\begin{bmatrix} -em_0\omega^2\sin(t\omega) + m_0\frac{d^2}{dt^2}y(t) + m_2\frac{d^2}{dt^2}y(t) \\ \frac{3R^2m_1\frac{d^2}{dt^2}\psi(t)}{2} \end{bmatrix} = \mathbf{M}\ddot{\mathbf{q}} - \mathbf{Q}^{m_0}(t)$$

Tehát a tömegmátrix az általános erővektor második időszerinti deiváltjának az együttható mátrixa, míg az excentrikus forgómozgásból származó általános erő tag az inhomogenitást okozó tag.

In [5]:

```
# nullmátrix létrehozása a tömegmátrixnak és az erővektornak
M = sp.zeros(2)
Q = sp.zeros(2,1)

# általános koordináták második deriváltja
ddq = sp.Matrix([y.diff(t,2), ψ.diff(t,2)])

for i in range(2):
    for j in range(2):
        M[i,j] = Mat[i].expand().coeff(ddq[j])
Q_m0 = (M*ddq).expand()-Mat.expand()

display(Math('Q^{{m_0}} = {}'.format(sp.latex(Q_m0))))
display(Math('M = {}'.format(sp.latex(M))))
```

$$Q^{m_0} = \begin{bmatrix} em_0\omega^2\sin(t\omega) \\ 0 \end{bmatrix}$$

$$M = \begin{bmatrix} m_0 + m_2 & 0 \\ 0 & \frac{3R^2m_1}{2} \end{bmatrix}$$

In [6]:

```
## Merevségi mátrix már közvetlenül kapható a potenciális energiából
# nullmátrix létrehozása a merevségi mátrixnak
K = sp.zeros(2,2)
# nullmátrix feltöltése a megfelelő parciális derivált értékekkel
for i in range(2):
    for j in range(2):
        K[i,j] = U.expand().diff(q[i]).diff(q[j])

display(Math('K = {}'.format(sp.latex(K))))
```

$$K = \left[\begin{matrix}k_{2} & -Rk_{2}\\-Rk_{2} & R^{2} k_{1} + R^{2} k_{2}\end{matrix}\right]$$

In [7]:

```
### Az általános erővektor másik tagja a külső erők teljesítményéből számítható
# Ebben a feladatban csak az M(t) nyomaték működik külső erőként, ennek teljesítménye p
P = -M_t*ψ.diff(t)

"""Ebből a külső erők vektora kapható ha vesszük az általános koordináták
deriváltajinak az együtthatóit a megfelelő helyen"""
Q_M = sp.zeros(2,1)
for i in range(2):
    Q_M[i] = P.expand().coeff(q[i].diff(t))
Q_M
```

Out[7]:

$$\left[\begin{matrix}0\\- M_{0} \cos{\left(t\omega + \varepsilon \right)}\end{matrix}\right]$$

In [8]:

```
## Az általános erő a két erő tag összegéből kapható
Q = Q_M+Q_m0
display(Math('Q = {}'.format(sp.latex(Q))))

"""Az általános erő szétszedhető sin-os és cos-os tagokra,
(ez a sajátkörfrekvencia számolásnál egy fontos lépés lesz).
Ehhez először használjuk a trig_expand() parancsot, hogy kibontsuk a cos-os tagot"""
Q[1] = sp.expand_trig(Q[1])
display(Math('Q = {}'.format(sp.latex(Q))))

# Majd szedjuk ki a sin(tω) és cos(tω) együtthatóit
Fc = sp.zeros(2,1)
Fs = sp.zeros(2,1)
for i in range(2):
    Fc[i] = Q[i].expand().coeff(sp.cos(ω*t))
    Fs[i] = Q[i].expand().coeff(sp.sin(ω*t))

display(Math('F_s = {}'.format(sp.latex(Fs))))
display(Math('F_c = {}'.format(sp.latex(Fc))))
```

$$Q = \left[\begin{matrix}e m_{0} \omega^{2} \sin{\left(t \omega \right)}\\- M_{0} \cos{\left(t \omega + \varepsilon \right)}\end{matrix}\right]$$

$$Q = \left[\begin{matrix}e m_{0} \omega^{2} \sin{\left(t \omega \right)}\\- M_{0} \left(- \sin{\left(\varepsilon \right)} \sin{\left(t \omega \right)} + \cos{\left(\varepsilon \right)} \cos{\left(t \omega \right)}\right)\end{matrix}\right]$$

$$F_s = \left[\begin{matrix}e m_{0} \omega^{2}\\M_{0} \sin{\left(\varepsilon \right)}\end{matrix}\right]$$

$$F_c = \left[\begin{matrix}0\\- M_{0} \cos{\left(\varepsilon \right)}\end{matrix}\right]$$

Ezzel a mozgásegyenlet

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{K}\mathbf{q} = F_s\sin(\omega t) + F_c\cos(\omega t),$$

2. Feladat

A harmonikus gerjesztés miatt a pertikuláris megoldást harmonikus próbafüggvény segaítségével keressük

$$\mathbf{q}(t) = \mathbf{L}\cos(\omega t) + \mathbf{N}\sin(\omega t)$$

Ennek a deriváltjai:

$$\dot{\mathbf{q}}(t) = -\omega\mathbf{L}\sin(\omega t) + \omega\mathbf{N}\cos(\omega t)$$

$$\ddot{\mathbf{q}}(t) = -\omega^2\mathbf{L}\cos(\omega t) - \omega^2\mathbf{N}\sin(\omega t)$$

Visszaírva a próbafüggvényt és a deriváltjait a mozgásegyenletbe, majd a $\sin(\omega t)$ és $\cos(\omega t)$ együtthatókat összegyűjtve adódik az egyenletrendszer $\mathbf{L}$-re és $\mathbf{N}$ -re

$$\begin{bmatrix}-\omega^2\mathbf{M}+\mathbf{K} & \mathbf{0}\\ \mathbf{0} & -\omega^2\mathbf{M}+\mathbf{K}\end{bmatrix}\begin{bmatrix}\mathbf{L}\\ \mathbf{N}\end{bmatrix} = \begin{bmatrix}\mathbf{F}_c\\ \mathbf{F}_s\end{bmatrix}$$

In [9]:

```python
### Oldjuk meg az egyenletrendszert
# Hozzunk létre szimbolikusan vektorokat a megoldásnak
L1, L2, N1, N2 = sp.symbols("L1, L2, N1, N2")
L = sp.Matrix([[L1],[L2]])
N = sp.Matrix([[N1],[N2]])

# Megoldás
L_sol = sp.solve(((-ω**2*M+K)*L-Fc).subs(adatok))
N_sol = sp.solve(((-ω**2*M+K)*N-Fs).subs(adatok))
L[0] = L_sol[L1].evalf(4)
L[1] = L_sol[L2].evalf(4)
N[0] = N_sol[N1].evalf(4)
N[1] = N_sol[N2].evalf(4)

# írjuk be a partikuláris megoldásba az eredményeket
q_p = (L*sp.cos(ω*t)+N*sp.sin(ω*t)).expand().subs(adatok)
display(Math('\mathbf{{q}}_p = {}'.format(sp.latex(q_p))))
```

$$\mathbf{q}_p = \begin{bmatrix} 0.0002071 \sin(30t) - 0.0009696 \cos(30t) \\ -0.03591 \sin(30t) + 0.06278 \cos(30t) \end{bmatrix}$$

3. Feladat

In [10]:

```python
## A rugerő maximumánál figyelembe kell venni a statikus és dinamikus részt is
# Statikus deformációból adód rész:
Fk2_st =  ((m0+m2)*g).subs(adatok).evalf(4)
display(Math('F{{k2,st}} = {}\\ \mathrm{{N}}'.format(sp.latex(Fk2_st))))

# A dinamikus részhez numerikusan könnyen számítható
import numpy as np
t_val = np.linspace(0,0.5,1000) # lista létrehozása a [0 ; 0,5] intervallum 1000 részre
Fk2_din = np.zeros(len(t_val)) # nulla lista létrehozása (ugyanannyi elemszámmal)
# dinamikus tag számítása adott időpillanatban
for i in range(len(t_val)):
    Fk2_din[i] = (k2*(R*q_p[1]-q_p[0])).subs(adatok).subs(t,t_val[i]).evalf()
Fk2_din_max = max(Fk2_din).round(2)

# Dinamikus tag
display(Math('F{{k2,din,max}} = {}\\ \mathrm{{N}}'.format(sp.latex(Fk2_din_max))))

# Az erő maximuma
Fk2_max = (Fk2_din_max + Fk2_st).evalf(4)
display(Math('F{{k2,max}} = {}\\ \mathrm{{N}}'.format(sp.latex(Fk2_max))))
```

$$Fk2, st = 30.41\ \mathrm{N}$$

$$Fk2, din, max = 3.08\ \mathrm{N}$$

$$Fk2, max = 33.49\ \mathrm{N}$$

4. Feladat

In [11]:

```
## A sajátfrekvenciák a frekvencia egyenletből kaphatók
ω_n2, ω_n = sp.symbols("ω_n2, ω_n")
# oldjuk meg az egyenletet ω_n^2-re, majd vonjunk gyököt
ω_n2_val = sp.solve((-ω_n2*M+K).subs(adatok).det())
ω_n = [(sp.sqrt(i)) for i in ω_n2_val]


display(Math('ω_{{n,1}} = {}\\ \mathrm{{rad/s}}'.format(sp.latex(ω_n[0].evalf(3)))))
display(Math('ω_{{n,2}} = {}\\ \mathrm{{rad/s}}'.format(sp.latex(ω_n[1].evalf(4)))))
```

$$\omega_{n,1} = 4.17\ \mathrm{rad/s}$$

$$\omega_{n,2} = 15.72\ \mathrm{rad/s}$$

In [12]:

```
## Lengéskép vektorok meghatározása
# Hozzunk létre a lengésképvektoroknak egy üres listát, majd töltsük fel 2 lengésképvek
A = []
A2 = sp.symbols("A2")
for i in range(2):
    A.append(sp.Matrix([[1],[A2]]))

    # oldjuk meg az egyenletet a lengésképekre és írjuk be a megoldásokat a lengésképve
    A[i][1] = sp.solve((((-ω_n[i]**2*M+K)*A[i]).subs(adatok))[0])[0]

display(Math('A_{{1}} = {}\\ \mathrm{{m}}'.format(sp.latex(A[0].evalf(3)))))
display(Math('A_{{2}} = {}\\ \mathrm{{m}}'.format(sp.latex(A[1].evalf(4)))))
```

$$A_1 = \begin{bmatrix} 1.0 \\ 3.65 \end{bmatrix}\ \mathrm{m}$$

$$A_2 = \begin{bmatrix} 1.0 \\ -14.15 \end{bmatrix}\ \mathrm{m}$$

Készítette:


```
Juhos-Kiss Álmos (Alkalmazott Mechanika Szakosztály)
Bachrathy Dániel (BME MM) kidolgozása alapján.

 Hibák, javaslatok:
 amsz.bme@gmail.com
 csuzdi02@gmail.com
 almosjuhoskiss@gmail.com

     2021.05.05.
```