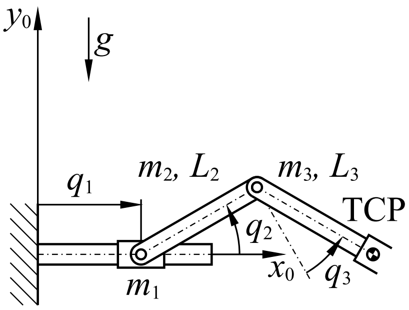
BME Gépészmérnöki Kar	ROBOTMECHANIZMUSOK DINAMIKÁJA	Név: Kovács Hunor
Műszaki Mechanikai Tanszék	2. HÁZI FELADAT	Neptun kód: P953MO
2023/24 II.	Határidő: 2024. május 16.	Aláírás:

Téma: 3 szabadsági fokú síkbeli robot pozíciószabályozása

Az 1-es ábrán látható manipulátort homogén, priztmatikus rudak és lineárisan megvezetett csúszkák alkotják. A prizmatikus rudak m_i tömegűek és L_i hosszúak; továbbá a csúszkák tömege m_j és a tehetetlenségi nyomatékuk $\Theta_{\mathrm{CoM}j}$. A beavatkozó erők/nyomatékok csuklónként u_1, u_2 és u_3. A cél a robot egy állandó $\mathbf{q}^{\mathrm{d}}$ előírt konfigurációban tartása. Kövesse az alábbi lépéseket!



1. ábra. 3DoF robot

1. táblázat. Numerikus adatok

$m_1 = 4.5$ kg	$L_2 = 1.2$ m
$m_2 = 4.25$ kg	$L_3 = 0.8$ m
$m_3 = 3.3$ kg	$g = 9.81$ m/s^2
$h = 0.003$ s	
$\mathbf{q}_0 = [0.55\,\mathrm{m}, 0^\circ, 0^\circ]^\mathsf{T}$	
$\mathbf{q}^{\mathrm{d}} = [0.2\,\mathrm{m}, 20^\circ, 20^\circ]^\mathsf{T}$	

Feladatok:

1. Építse fel a robot kinematikai modelljét, mely alapján a TCP pozíció kiszámítható a csukló-koordináták függvényeként. A kinematikai modell a karok helyzetét, sebességállapotát és a súlyponti sebességeket is megadja tetszőleges csuklókoordináták és csuklósebességek esetén.

   Ellenőrző pont: a TCP pozíció globális x koordinátája és a második kar tömegközéppontjának helyzete az előírt $\mathbf{q}^{\mathrm{d}}$ konfigurációban ellenőrizhető a `www.mm.bme.hu/hwchk` oldalon.

2. Vezesse le a rendszer **mozgásegyenletét** az ábrán megadott csuklókoordináták felhasználásával. A mozgásegyenletet a következő általános alakban adja meg: $\mathbf{M}\ddot{\mathbf{q}} + \mathbf{C} = \mathbf{H}\mathbf{u}$.

   Ellenőrző pont: A tömegmátrix néhány eleme ellenőrizhető.

3. Készítse el a szabad mozgás numerikus szimulációját, amikor a beavatkozó erők/nyomatékok: $\mathbf{u} = \mathbf{0}$! A kezdeti feltételek: $\mathbf{q}(0) = \mathbf{q}_0$ és $\dot{\mathbf{q}}(0) = \mathbf{0}$.

   Ellenőrző pont: a teljes mechanikai energia $T(t) + U(t) =$ konstans esetén nagy eséllyel jók az egyenletek.

4. Inverz dinamikai szabályozót feltételezve számolja ki a leggyorsabb csillapodáshoz tartozó K_P és K_D skalár **erősítési tényezőket** figyelembe véve a megadott h időlépést! Az inverz dinamikára épülő szabályozás a következő: $\mathbf{u} = \mathbf{H}^{-1}(\mathbf{M}\mathbf{v} + \mathbf{C})$ ahol $\mathbf{v} = \ddot{\mathbf{q}}^{\mathrm{d}} - \mathbf{K}_\mathrm{P}(\mathbf{q} - \mathbf{q}^{\mathrm{d}}) - \mathbf{K}_\mathrm{D}(\dot{\mathbf{q}} - \dot{\mathbf{q}}^{\mathrm{d}})$, ahol $\mathbf{K}_\mathrm{P} = K_\mathrm{P}\,\mathbf{I}$, $\mathbf{K}_\mathrm{D} = K_\mathrm{D}\,\mathbf{I}$ és $\mathbf{I}$ a 3×3-as egység mátrix.

5. Készítsen numerikus szimulációt elosztott PD szabályozó felhasználásával inverz dinamikai tagok nélkül ($\mathbf{u} = -\mathbf{K}_\mathrm{P}(\mathbf{q} - \mathbf{q}^\mathrm{d}) - \mathbf{K}_\mathrm{D}(\dot{\mathbf{q}} - \dot{\mathbf{q}}^\mathrm{d})$)! Használja a korábban kiszámolt erősítési tényezőket. Kezdeti feltételek: $\mathbf{q}(0) = \mathbf{q}_0$ és $\dot{\mathbf{q}}(0) = \mathbf{0}$. Határozza meg a pozícionálási hibát $t \mapsto \infty$ esetén a csukló- és a munkatérben egyaránt (t-t kellően nagyra válassza a szimulációban)! Rajzolja meg a beavatkozó erők $\mathbf{u}$ és a csuklókoordináták $\mathbf{q}$ idődiagramját!

6. Készítsen numerikus szimulációt a fent felhasznált inverz dinamikai szabályozó felhasználásával. Használja ismét a korábbi erősítési tényezőket. Kezdeti feltételek: $\mathbf{q}(0) = \mathbf{q}_0$ és $\dot{\mathbf{q}}(0) = \mathbf{0}$. Határozza meg a pozícionálási hibát $t \mapsto \infty$ esetén a csukló- és munkatérben egyaránt! Rajzolja meg a beavatkozó erők $\mathbf{u}$ és a csuklókoordináták $\mathbf{q}$ idődiagramját!

7. **Bónusz I**: Készítsen animációkat a robot mozgásáról, a szimulációs eredmények szemléltetésére! *Megjegyzés: a bónuszfeladatok jutalma további ismeretek és készségek szerzése a témában. Extra pontokat ez nem jelent.*

8. **Bónusz II**: Készítsen szimulációkat, ahol egy $\mathbf{q}^\mathrm{d}(t)$ időtől függő (nem állandó) pálya van előírva (például egy kör követése a munkatérben)!

Kérjük, töltse fel a házi feladatot a Moodle rendszerbe! Két fájlt kell feltölteni `RMF_HF1_NEPTUN.pdf` és `RMF_HF1_NEPTUN.zip` néven. A PDF fájlnak tartalmaznia kell

a) ezt a teljes dokumentumot az aláírásokkal és a kitöltött eredménytáblával,

b) 2–3 oldalas jegyzetet a megoldás főbb lépéseiről és a fontosabb egyenletekről, ábrákról és idődiagramokról,

c) végül a szimulációhoz használt programkódokat.

A ZIP fájlnak a futtatható szimulációs programkódokat és a PDF fájlt is tartalmaznia kell. Ha a bónusz feladatokra is van megoldása, azokat illessze a ZIP fájlba (pl az animációk GIF vagy videó fájljai). A házi feladat megoldásához tetszőleges szoftver felhasználható, figyelembe véve az egyetemi szoftverfelhasználási szabályokat, legcélszerűbb az ingyenes és nyílt forráskódú Python, Julia, Octave vagy Maxima használata, de hallgatói licensszel például a Matlab vagy a Mathematica is használható.

Eredmények:

Név: Kovács Hunor   Aláírás: Hunor [signature]

Neptun kód: P953MO   HF kód: 34112

TCP pozíció az előírt $\mathbf{q}^{\mathrm{d}}$ konfigurációban:

$$\mathbf{r}_{\mathrm{TCP}} = \left[\; 1{,}8419\ \mathrm{m},\quad -0{,}2024\ \mathrm{m},\quad 0\ \mathrm{m} \;\right]^{\mathrm{T}}$$

CoM3 pozíció az előírt $\mathbf{q}^{\mathrm{d}}$ konfigurációban:

$$\mathbf{r}_{\mathrm{CoM3}} = \left[\; 1{,}5847\ \mathrm{m},\quad 0{,}104\ \mathrm{m},\quad 0\ \mathrm{m} \;\right]^{\mathrm{T}}$$

Tömegmátrix $\mathbf{M}(\mathbf{q}^{\mathrm{d}})$ az előírt konfigurációban:

$$\mathbf{M}(\mathbf{q}^{\mathrm{d}}) = \begin{bmatrix} 12{,}05 & -1{,}2154 & 1{,}0112 \\ -1{,}2154 & 8{,}5795 & 1{,}2458 \\ 1{,}0112 & 1{,}2458 & 0{,}704 \end{bmatrix}$$

Optimális erősítési tényezők a leggyorsabb csillapodáshoz:

$$K_{\mathrm{P}} = 4115{,}2263, \qquad K_{\mathrm{D}} = 104{,}9383$$

Csuklótérbeli pozícionálási hiba $\mathbf{e}_q = \mathbf{q} - \mathbf{q}^{\mathrm{d}}$ inverz dinamika nélküli PD szabályozóval:

$$\mathbf{e}_q = \left[\; 0\ \mathrm{m},\quad -0{,}0166\ \mathrm{rad},\quad -0{,}002\ \mathrm{rad} \;\right]^{\mathrm{T}}$$

Munkatérbeli pozícionálási hiba $\mathbf{e}_{\mathrm{TCP}} = \mathbf{r}_{\mathrm{TCP}} - \mathbf{r}^{\mathrm{d}}_{\mathrm{TCP}}$ inverz dinamika nélküli PD szabályozóval:

$$\mathbf{e}_{\mathrm{TCP}} = \left[\; -0{,}0048\ \mathrm{m},\quad -0{,}0283\ \mathrm{m},\quad 0\ \mathrm{m} \;\right]^{\mathrm{T}}$$

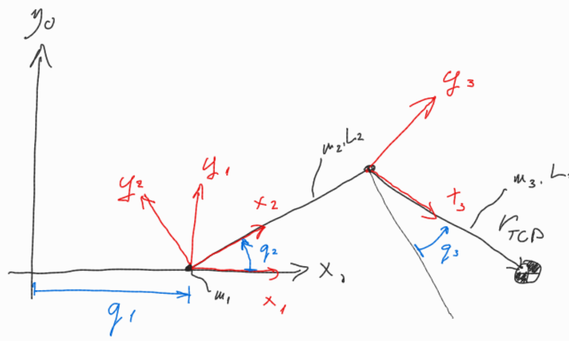
Csuklótérbeli pozícionálási hiba $\mathbf{e}^*_q = \mathbf{q} - \mathbf{q}^{\mathrm{d}}$ inverz dinamikai szabályozó esetén:

$$\mathbf{e}^*_q = \left[\; -0{,}347\cdot 10^{-12}\ \mathrm{m},\quad 0{,}3461\cdot 10^{-12}\ \mathrm{rad},\quad 0{,}3461\cdot 10^{-12}\ \mathrm{rad} \;\right]^{\mathrm{T}}$$

Kovács Ármor
P9S3MO

1) Robot kinematikai modellje

Az összeállítást a következő módon modelleztem:



ahol a csuklókoordináták: $q = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} [SI]$

A karokon meghatározott súlyponti homogén vektorok:

$$ {}^1\underline{\tilde{r}}_{s_1} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \hline 1 \end{bmatrix} \qquad {}^2\underline{\tilde{r}}_{s_2} = \begin{bmatrix} \frac{1}{2}L_2 \\ 0 \\ 0 \\ \hline 1 \end{bmatrix} \qquad {}^3\underline{\tilde{r}}_{s_3} = \begin{bmatrix} \frac{1}{2}L_3 \\ 0 \\ 0 \\ \hline 1 \end{bmatrix} $$

D-H táblázat:

	a_i	ϑ_i
1	q_1	0
2	0	q_2
3	L_2	$q_3 - \frac{\pi}{2}$

A D-H táblázat alapján a transzformációs mátrixokat is előállítom:

$$ {}^0\underline{\underline{T}}_1 = \left[\begin{array}{ccc|c} 1 & 0 & 0 & q_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right] \qquad {}^1\underline{\underline{T}}_2 = \left[\begin{array}{ccc|c} C_2 & -S_2 & 0 & 0 \\ S_2 & C_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right] \qquad {}^2\underline{\underline{T}}_3 = \left[\begin{array}{ccc|c} S_3 & C_3 & 0 & L_2 \\ -C_3 & S_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right] $$

jelölések: $C_i = \cos(q_i)$

$S_i = \sin(q_i)$

↳ a következő azonosságokat felhasználva:

$\sin(-\frac{\pi}{2} + q_3) = -\cos q_3$

$\cos(-\frac{\pi}{2} + q_3) = \sin q_3$

Az egyes koordinátarendszerek közti transzformációs mátrixokból előállítom a globális koordinátarendszerbe transzformáló mátrixokat:

$$ {}^0\underline{\underline{T}}_2 = {}^0\underline{\underline{T}}_1 \cdot {}^1\underline{\underline{T}}_2 = \left[\begin{array}{ccc|c} S_2 & C_2 & 0 & q_1 \\ C_2 & -S_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right] $$

$$ {}^0\underline{\underline{T}}_3 = {}^0\underline{\underline{T}}_2 \cdot {}^2\underline{\underline{T}}_3 = \left[\begin{array}{ccc|c} S_2 \cdot S_3 - C_2 \cdot C_3 & C_2 \cdot S_3 + C_3 \cdot S_2 & 0 & q_1 + L_2 S_2 \\ C_2 S_3 + C_3 \cdot S_2 & C_2 C_3 - S_2 S_3 & 0 & L_2 C_2 \\ 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right] $$

A TCP koordinátái a 3. koordinátarendszerben: ${}^3\underline{r}_{TCP} = \begin{bmatrix} L_3 \\ 0 \\ 0 \end{bmatrix}$

Ezt homogenizálva tudjuk a globális koordinátarendszerbe transzformálni:

$$ {}^0\underline{\tilde{r}}_{TCP} = {}^0\underline{\underline{T}}_3 \cdot {}^3\underline{\tilde{r}}_{TCP} = \left[\begin{array}{c} q_1 - L_3 \cdot (C_2 \cdot C_3 - S_2 \cdot S_3) + L_2 \cdot S_2 \\ L_3 (C_2 S_3 + C_3 \cdot S_2) + L_2 \cdot C_2 \\ 0 \\ \hline 1 \end{array}\right] \quad \left.\right\} {}^0\underline{r}_{TCP} $$

Kovács Hunor
P9S3MO

A kinematikai modellhez a karok helyzetét felírjuk a csuklókoordináták függvényeiben:

Szöghelyzet: $\underline{\alpha}_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$ $\underline{\alpha}_2 = \begin{bmatrix} 0 \\ 0 \\ q_2 \end{bmatrix}$ $\underline{\alpha}_3 = \begin{bmatrix} 0 \\ 0 \\ q_2 + q_3 - \frac{\pi}{2} \end{bmatrix}$

deriválással megkapjuk a szögsebességeket: $\underline{\omega}_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$ $\underline{\omega}_2 = \begin{bmatrix} 0 \\ 0 \\ \dot{q}_2 \end{bmatrix}$ $\underline{\omega}_3 = \begin{bmatrix} 0 \\ 0 \\ \dot{q}_2 + \dot{q}_3 \end{bmatrix}$

A súlypontok koordinátáit a globális koordinátarendszerbe transzformáljuk, majd deriváljuk, így megkapjuk a sebességeket:

Helyvektorok: ${}^0\underline{r}_{S_1} = \begin{bmatrix} q_1 \\ 0 \\ 0 \end{bmatrix}$ ${}^0\underline{r}_{S_2} = \begin{bmatrix} q_1 + \frac{L_2}{2} S_2 \\ \frac{L_2}{2} C_2 \\ 0 \end{bmatrix}$ ${}^0\underline{r}_{S_3} = \begin{bmatrix} q_1 + L_2 S_2 + \frac{L_3}{2}(S_2 S_3 - C_2 \cdot C_3) \\ L_2 \cdot C_2 + \frac{L}{2}(C_2 S_3 + C_3 S_2) \\ 0 \end{bmatrix}$

Sebességek: ${}^0\underline{v}_{S_1} = \begin{bmatrix} \dot{q}_1 \\ 0 \\ 0 \end{bmatrix}$ ${}^0\underline{v}_{S_2} = \begin{bmatrix} \dot{q}_1 - \frac{L_2}{2} S_2 \dot{q}_2 \\ \frac{L_2}{2} \cdot C_2 \cdot \dot{q}_2 \\ 0 \end{bmatrix}$ ${}^0\underline{v}_{S_3} = \begin{bmatrix} \dot{q}_1 - L_2 S_2 \cdot \dot{q}_2 + \frac{L_3}{2} \cdot \cos(q_2+q_3)(\dot{q}_2+\dot{q}_3) \\ L_2 C_2 \dot{q}_2 + \frac{L_3}{2} \cdot \sin(q_2+q_3)(\dot{q}_2+\dot{q}_3) \\ 0 \end{bmatrix}$

Az ellenőrzést elvégeztem, a numerikus értékeket a MATLAB program jeleníti meg.

2 Mozgásegyenlet

A kiindulási alap a Lagrange-egyenlet: $\frac{d}{dt} \cdot \frac{\partial T}{\partial \dot{q}} + \frac{\partial T}{\partial q} + \frac{\partial U}{\partial q} = Q^*$

ahol:

$T = \frac{1}{2} m_1 v_{S_1}^2 + \frac{1}{2} m_2 {}^0v_{S_2}^2 + \frac{1}{2} \cdot \Theta_{S_2} \cdot {}^0\omega_2^2 + \frac{1}{2} m_3 {}^0v_3^2 + \frac{1}{2} \cdot \Theta_{S_3} \cdot {}^0\omega_3^2$; a kinetikus energia

$U = m_2 g \, {}^0y_{S_2} + m_3 g \, {}^0y_{S_3}$; a potenciális energia

$\underline{Q}^* = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}$; az általánosított erő

A célunk a mozgásegyenlet következő alakjának előállítása: $\underline{\underline{M}} \cdot \ddot{\underline{q}} + \underline{C} = \underline{\underline{H}} \cdot \underline{u}$

A tömegmátrix előáll a $\frac{d}{dt}\frac{\partial T}{\partial \dot{q}}$ tag vektorának $\ddot{q}$ együtthatóiból. A beavatkozó erő vektora az adott u_1, u_2, u_3 tagokból áll, H pedig jelen esetben egységmátrix, mivel a beavatkozó erők csak a megfelelő csuklókoordinátákra hatnak.

C vektort megkapjuk, ha a Lagrange-egyenlet baloldalát egyenlővé tesszük a mozgásegyenlet baloldalával.

A kívánt pozícióban (q^d) a mozgásegyenlet a következő értékeket veszi fel:

$$\begin{bmatrix} 12,05 & -1,2154 & 1,0112 \\ -1,2154 & 8,5795 & 1,2458 \\ 1,0112 & 1,2458 & 0,704 \end{bmatrix} \cdot \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 68,3353 \\ 8,3236 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}$$

A paraméteres egyenlet a MATLAB kódban megtalálható.

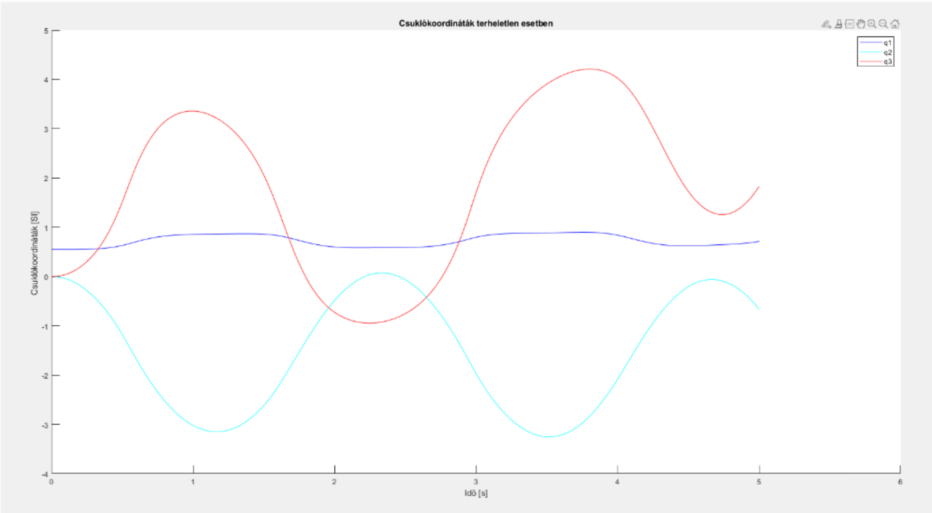
Kovács Hunor
P953MO

③ Terheletlen szimuláció

Amennyiben a beavatkozó erők nullák, a mozgásegyenletbe behelyettesítve $\ddot{q}$-ra tudunk egy egyenletet felállítani. A kezdeti feltételekkel az első pillanatbeli gyorsulást kiszámoljuk, majd ebből explicit euler formulával kiszámolhatjuk a következő pillanatbeli sebességet majd pozíciót. Ezt iterálva lefuttatjuk a szimulációt, melynek eredményeképp a pozíció változását ábrázolom. (Az ellenőrzés MATLAB-ban tekinthető meg)

KF: $\underline{q}_0 = \begin{bmatrix} 0{,}55 \text{ m} \\ 0 \text{ rad} \\ 0 \text{ rad} \end{bmatrix}$

$\underline{\dot{q}}_0 = \underline{0}$



a mozgásegyenletből:

$$\underline{\underline{M}} \cdot \underline{\ddot{q}} + \underline{C} = \underline{0}$$

$$\underline{\ddot{q}} = -\underline{\underline{M}}^{-1} \underline{C}$$

④ Inverz dinamikai szabályozó erősítési tényezői

A leggyorsabb beálláshoz tartozó p és d együtthatókat, valamint azok felhasználásával K_p és K_d meghatározását az előadás során levezettük, ezért a kidolgozásomban ettől eltekintek. Az időlépés h, adott értékű $(0{,}003)$.

Előadás alapján: $p^* = \frac{1}{27}$ és $d^* = \frac{17}{54}$

valamint: $K_p = \frac{p}{h^2} = \underline{4115{,}2263}$

$K_d = \frac{d}{h} = \underline{104{,}9383}$

⑤ PD szabályozó szimulációja inverz tagok nélkül

A szabályozó erők megadott vektora: $\underline{u} = -\underline{\underline{K}}_P(\underline{q} - \underline{q}^d) - \underline{\underline{K}}_D(\underline{\dot{q}} - \underline{\dot{q}}^d)$

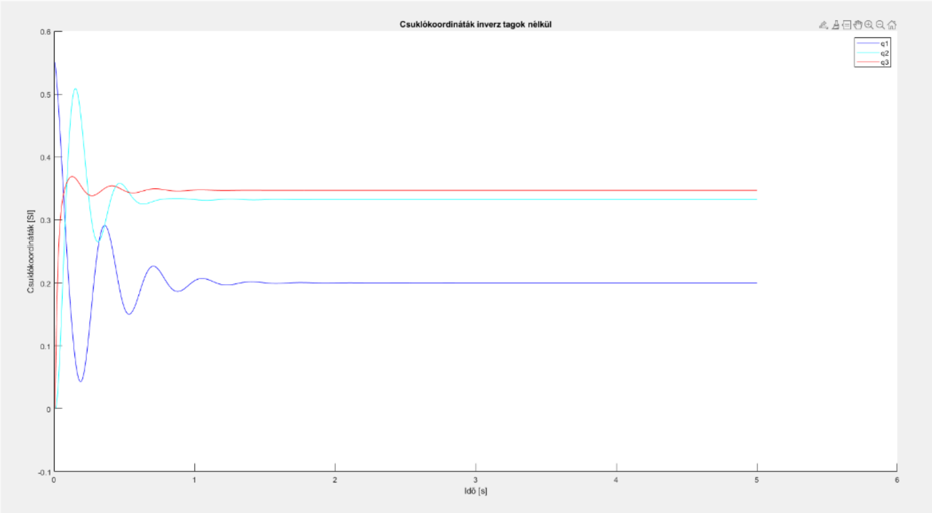
ahol $\underline{q}^d$ és $\underline{q}_0$ meg van adva, $\underline{\dot{q}}^d = \underline{0}$ és $\underline{\underline{K}}_P = K_p \cdot \underline{\underline{E}}$; $\underline{\underline{K}}_d = K_d \cdot \underline{\underline{E}}$

A szimulációt a 3. feladathoz hasonlóan végezzük, azonban a mozgásegyenletből $\underline{\ddot{q}}$ eltérő lesz.

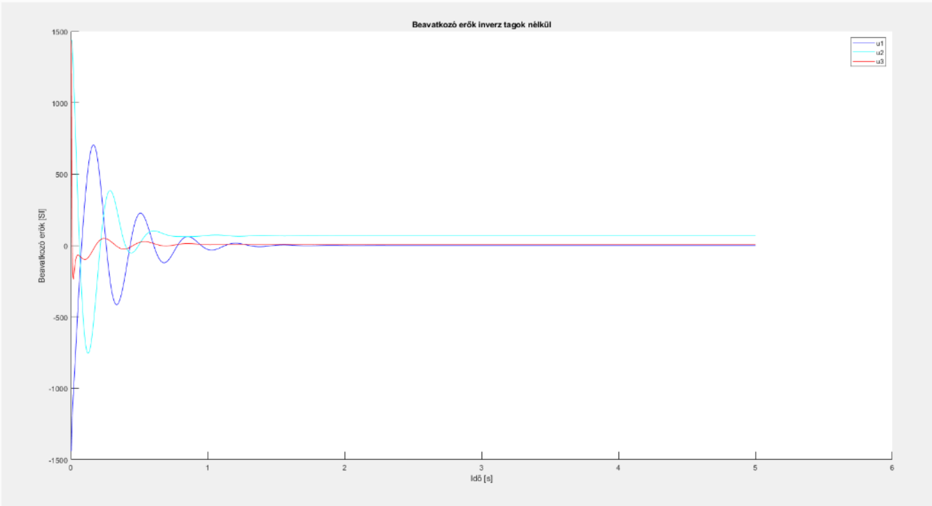
$$\underline{\ddot{q}} = \underline{\underline{M}}^{-1}(\underline{\underline{H}} \cdot \underline{u} - \underline{C}),$$ ahol $\underline{\underline{H}} = \underline{\underline{E}}$ és $\underline{u}$ a fent kifejtett egyenlet eredménye.

Kovács [illegible]
P953MO

A szimuláció során ábrázoljuk a csuklókoordináták értékeit:



Valamint a beavatkozó erőket:



Az ábrákról látszik hogy a szimuláció végén kapott értékeket tekinthetjük állandósultaknak, így a maradó hibát számolhatjuk belőlük.

A csuklókoordináták eltérése a kívánt értékektől:

$$\underline{e}_q = \underline{q}_\infty - \underline{q}^d = \begin{bmatrix} 0 \\ -0{,}0166 \\ -0{,}002 \end{bmatrix} [SI]$$

A csuklókoordinátákkal kiszámolt TCP pozíciójának hibái:

$$\underline{e}_{TCP} = {}^0\underline{r}_{TCP}(\underline{q}_\infty) - {}^0\underline{r}_{TCP}(\underline{q}^d) = \begin{bmatrix} -0{,}0038 \\ -0{,}0283 \\ 0 \end{bmatrix} [m]$$

Kovács Ármin
P353MO

6) Inverz dinamikai szabályozó szimulációja

A szimuláció ugyanarra az elvre épül, mint a korábbi feladatokban, azonban a beavatkozó erő képlete megváltozik.

$$\underline{u} = \underline{\underline{H}}^{-1}\left(\underline{\underline{M}} \cdot \underline{v} + \underline{C}\right) \qquad \rightarrow \underline{\underline{H}}^{-1} = \underline{\underline{E}}$$

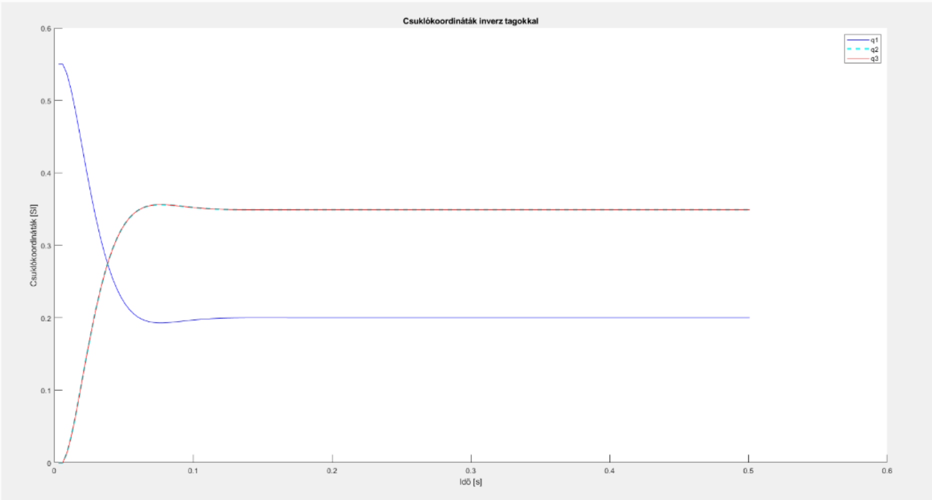
ahol: $\underline{v} = \ddot{\underline{q}}^d - \underline{\underline{K}}_P \cdot (\underline{q} - \underline{q}^d) - \underline{\underline{K}}_d (\dot{\underline{q}} - \dot{\underline{q}}^d)$ ahol $\ddot{\underline{q}}^d$ és $\dot{\underline{q}}^d = \underline{0}$

Ezt a mozgásegyenletbe behelyettesítve:

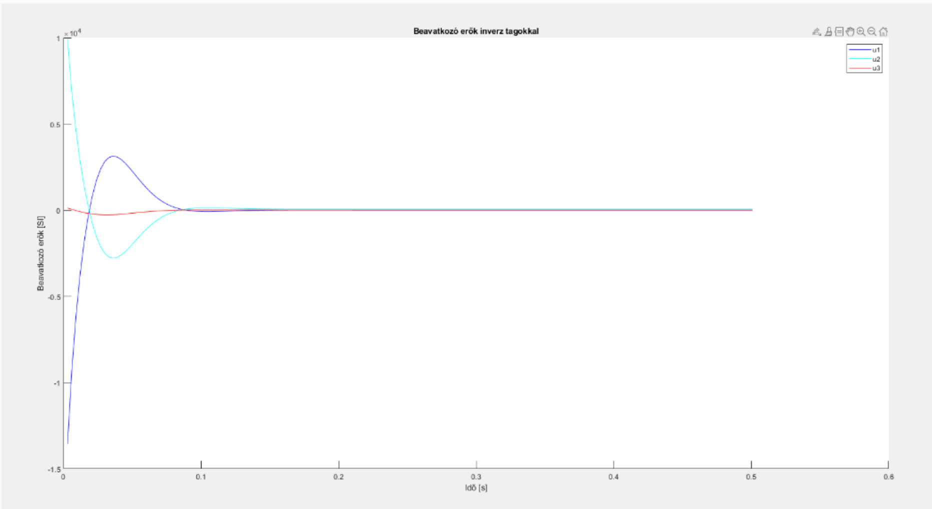
$$\underline{\underline{M}} \cdot \ddot{\underline{q}} + \underline{C} = \underline{\underline{H}} \cdot \underline{u} \longrightarrow \underline{\underline{M}} \cdot \ddot{\underline{q}} + \underline{C} = \underline{\underline{M}} \cdot \underline{v} + \underline{C}$$

$$\ddot{\underline{q}} = \underline{v} = -\underline{\underline{K}}_P (\underline{q} - \underline{q}^d) - \underline{\underline{K}}_d \cdot \dot{\underline{q}}$$

Az így lefuttatott szimuláció csuklókoordinátái:



A beavatkozó erők:



Az előző feladathoz hasonlóan a csuklókoordináták maradó hibái:

$$\underline{e}_q^* = \underline{q}_\infty - \underline{q}^d = \begin{bmatrix} -0,357 \\ 0,356 \\ 0,356 \end{bmatrix} \cdot 10^{-12} \, [SI]$$

És a TCP koordinátáira számolt maradó hibák:

$$\underline{e}_{TCP} = {}^0\underline{r}_{TCP}(\underline{q}_\infty) - {}^0\underline{r}_{TCP}(\underline{q}^d) = \begin{bmatrix} -0,0649 \\ 0,7561 \\ 0 \end{bmatrix} \cdot 10^{-12} \, [m]$$

```
%adatok
M1 = 4.5;
M2 = 4.25;
M3 = 3.3;
h = 0.003;
L2 = 1.2;
L3 = 0.8;
G = 9.81;
q0 = [0.55; 0; 0];
qd = [0.2; 20*pi/180; 20*pi/180];
%%
%1. feladat
%változók
syms q1(t) q2(t) q3(t) m1 m2 m3 l2 l3 g
q = [q1(t); q2(t); q3(t)];
%Transzformációs mátrixok
T01=[1 0 0 q1(t); 0 1 0 0; 0 0 1 0; 0 0 0 1];
T12=[cos(q2(t)) -sin(q2(t)) 0 0; sin(q2(t)) cos(q2(t)) 0 0; 0 0 1 0; 0 0 0 1];
T23=[sin(q3(t)) cos(q3(t)) 0 l2; -cos(q3(t)) sin(q3(t)) 0 0; 0 0 1 0; 0 0 0 1];
T02=T01*T12;
T03=T02*T23;

%TCP pozíciója
r3tcp = [l3;0;0;1];
r0tcp = T03*r3tcp;
%sebessége
v0tcp = diff(r0tcp, t);
%gyorsulása
a0tcp = simplify(diff(v0tcp, t));

disp(r0tcp(1:3));
% disp(v0tcp(1:3));
% disp(a0tcp(1:3));

%Karok szöghelyzete
alfa01 = [0; 0; 0];
alfa02 = [0; 0; q2(t)];
alfa03 = [0; 0; q2(t) + q3(t) - pi/2];
%szögsebességállapota
w01 = diff(alfa01, t);
w02 = diff(alfa02, t);
w03 = diff(alfa03, t);
%helyzete

%súlypontok koordinátái
r1s1 = [0;0;0;1];
r2s2 = [0.5*l2;0;0;1];
r3s3 = [0.5*l3;0;0;1];
r0s1 = T01*r1s1;
r0s2 = T02*r2s2;
r0s3 = T03*r3s3;
% disp(r0s1(1:3));
% disp(r0s2(1:3));
% disp(r0s3(1:3));
%sebességeik
```

```
v0s1 = diff(r0s1, t);
v0s2 = simplify(diff(r0s2, t));
v0s3 = simplify(diff(r0s3, t));
% disp(v0s1(1:3));
% disp(v0s2(1:3));
% disp(v0s3(1:3));

%Ellenőrző pont
TCP = double(subs(r0tcp, [q1(t) q2(t) q3(t) l2 l3], [qd(1) qd(2) qd(3) L2 L3]));
CoM2 = double(subs(r0s2(1), [q1(t) q2(t) q3(t) l2 l3], [qd(1) qd(2) qd(3) L2 L3]));
CoM3 = double(subs(r0s3, [q1(t) q2(t) q3(t) l2 l3], [qd(1) qd(2) qd(3) L2 L3]));
disp(TCP);
disp(CoM2);
disp(CoM3);
%%
%2. feladat

%Sulyponti tehetetlenségi nyomaték 2D-ben csak Z komponens
theta2 = 1/12*m2*l2^2;
theta3 = 1/12*m3*l3^2;

%Lagrange-egyenletek
%T
T = 0.5*m1*(v0s1(1)^2+v0s1(2)^2) + 0.5*m2*(v0s2(1)^2+v0s2(2)^2) + 0.5*theta2*w02(3)^2↙
+ 0.5*m3*(v0s3(1)^2+v0s3(2)^2) + 0.5*theta3*w03(3)^2;
%T q szerinti deriváltja
Tdq = [diff(T, q1(t)); diff(T, q2(t)); diff(T, q3(t))];
%T q derivált szerinti deriváltja
Tdqp = [diff(T, diff(q1(t), t)); diff(T, diff(q2(t), t)); diff(T, diff(q3(t), t))];
%T q derivált és idő szerinti deriváltja
Td = diff(Tdqp, t);

%Tömeg mátrix
M = sym('M', [3, 3]);
for i = 1:3
    for j = 1:3
        [coeff, pol] = coeffs(Td(i), diff(q(j), t, t));
        M(i, j) = simplify(coeff(1));
    end
end

%Ellenőrző pont
disp(double(subs(M, [q1(t) q2(t) q3(t) l2 l3 m1 m2 m3 g], [qd(1) qd(2) qd(3) L2 L3 M1↙
M2 M3 G])));

%Potenciál fv
U = simplify(m2*g*r0s2(2) + m3*g*r0s3(2));

%Lagrange egyenlet baloldala
Lagr = Td - Tdq + [diff(U, q1(t)); diff(U, q2(t)); diff(U, q3(t))];
%C vektor, mivel a Lagrange egyenlet jobboldala 0
qpp2 = [diff(q1(t), t, t); diff(q2(t), t, t); diff(q1(t), t, t)];
C = Lagr - M*qpp2;

disp(double(subs(C, [q1(t) q2(t) q3(t) l2 l3 m1 m2 m3 g], [qd(1) qd(2) qd(3) L2 L3 M1↙
```

```
M2 M3 G])));
%%
%3. feladat

%Állapottér modell
veg = int32(5/h); %vége
%KF és üres vektorok megadása
q1_ = zeros(veg, 1);
q1_(1) = 0.55;
q2_ = zeros(veg, 1);
q3_ = zeros(veg, 1);
qp1 = zeros(veg, 1);
qp2 = zeros(veg, 1);
qp3 = zeros(veg, 1);

%Q kétszeres deriváltjának kiszámítása
Mq = subs(M, [l2 l3 m1 m2 m3 g], [L2 L3 M1 M2 M3 G]);
Cq = subs(C, [l2 l3 m1 m2 m3 g], [L2 L3 M1 M2 M3 G]);
qpp = -1*Mq\Cq;
qpp1 = qpp(1);
qpp2 = qpp(2);
qpp3 = qpp(3);
%%

for i = 1:veg-1
    %Explicit euler formulával kiszámítom a deriváltakat majd a q-kat
    qp1(i+1) = qp1(i) + h*subs(qpp1, [q1(t) q2(t) q3(t)], [q1_(i) q2_(i) q3_(i)]);
    qp2(i+1) = qp2(i) + h*subs(qpp2, [q1(t) q2(t) q3(t)], [q1_(i) q2_(i) q3_(i)]);
    qp3(i+1) = qp3(i) + h*subs(qpp3, [q1(t) q2(t) q3(t)], [q1_(i) q2_(i) q3_(i)]);
    q1_(i+1) = q1_(i) + h*qp1(i);
    q2_(i+1) = q2_(i) + h*qp2(i);
    q3_(i+1) = q3_(i) + h*qp3(i);

end
%%

%Ellenőrzéshez U és T
Tq = subs(T, [l2 l3 m1 m2 m3 g], [L2 L3 M1 M2 M3 G]);%numerikus értékek
Uq = subs(U, [l2 l3 m1 m2 m3 g], [L2 L3 M1 M2 M3 G]);

Tt = subs(Tq, {diff(q1(t), t) diff(q2(t), t) diff(q3(t), t) q1(t) q2(t) q3(t)}, {qp1↙
(:) qp2(:) qp3(:) q1_(:) q2_(:) q3_(:)});%szimulált változók
Ut = subs(Uq, {q2(t) q3(t)}, {q2_(:) q3_(:)});

time3 = h:h:double(veg)*h;
figure();
hold on
plot(time3, q1_, 'b');
plot(time3, q2_, 'c');
plot(time3, q3_, 'r');
title('Csuklókoordináták terheletlen esetben');
legend('q1', 'q2', 'q3');
xlabel('Idő [s]');
ylabel('Csuklókoordináták [m]/[rad]');
hold off
```

```
figure();
hold on
title('T és C összege');
plot(time3, Tt, 'b');
plot(time3, Ut, 'c');
plot(time3, Tt + Ut, 'r');
legend('T', 'U', 'T + U');
xlabel('Idő [s]');
ylabel('T(t), U(t)');
hold off
%%
%4. feladat

p = 1/27;
d = 17/54;
kd = d/h;
kp = p/h^2;
Kd = kd*eye(3);
Kp = kp*eye(3);
disp(kp);
disp(kd);
%%
%5. feladat

%Állapottér modell
ts = int32(5/h);

%KF és üres vektorok megadása
q51 = zeros(ts, 1);
q51(1) = 0.55;
q52 = zeros(ts, 1);
q53 = zeros(ts, 1);
qp51 = zeros(ts, 1);
qp52 = zeros(ts, 1);
qp53 = zeros(ts, 1);

%u vektor számolása
qp = diff(q);
u = - Kp*(q-qd) - Kd*qp; %sebesség elvárt értéke 0


%Q kétszeres deriváltjának kiszámítása
qpp5 = Mq\(u - Cq); %mivel H = 1
qpp51 = qpp5(1);
qpp52 = qpp5(2);
qpp53 = qpp5(3);
%%

for i = 1:ts-1
    qp51(i+1) = qp51(i) + h*subs(qpp51, [diff(q1(t), t) diff(q2(t), t) diff(q3(t), t)↙
q1(t) q2(t) q3(t)], [qp51(i) qp52(i) qp53(i) q51(i) q52(i) q53(i)]);
    qp52(i+1) = qp52(i) + h*subs(qpp52, [diff(q1(t), t) diff(q2(t), t) diff(q3(t), t)↙
q1(t) q2(t) q3(t)], [qp51(i) qp52(i) qp53(i) q51(i) q52(i) q53(i)]);
    qp53(i+1) = qp53(i) + h*subs(qpp53, [diff(q1(t), t) diff(q2(t), t) diff(q3(t), t)↙
```

```
q1(t) q2(t) q3(t)], [qp51(i) qp52(i) qp53(i) q51(i) q52(i) q53(i)]);
    q51(i+1) = q51(i) + h*qp51(i);
    q52(i+1) = q52(i) + h*qp52(i);
    q53(i+1) = q53(i) + h*qp53(i);

end

time5 = h:h:double(ts)*h;
%csuklókoordináták
figure();
hold on
plot(time5, q51, 'b');
plot(time5, q52, 'c');
plot(time5, q53, 'r');
title('Csuklókoordináták inverz tagok nélkül');
legend('q1', 'q2', 'q3');
xlabel('Idő [s]');
ylabel('Csuklókoordináták [m]/[rad]');
hold off
%beavatkozó erők
figure()
u1 = subs(u(1), {diff(q1(t), t) diff(q2(t), t) diff(q3(t), t) q1(t) q2(t) q3(t)},↙
{qp51(:) qp52(:) qp53(:) q51(:) q52(:) q53(:)});
u2 = subs(u(2), {diff(q1(t), t) diff(q2(t), t) diff(q3(t), t) q1(t) q2(t) q3(t)},↙
{qp51(:) qp52(:) qp53(:) q51(:) q52(:) q53(:)});
u3 = subs(u(3), {diff(q1(t), t) diff(q2(t), t) diff(q3(t), t) q1(t) q2(t) q3(t)},↙
{qp51(:) qp52(:) qp53(:) q51(:) q52(:) q53(:)});
hold on
plot(time5, u1, 'b');
plot(time5, u2, 'c');
plot(time5, u3, 'r');
title('Beavatkozó erők inverz tagok nélkül');
legend('u1', 'u2', 'u3');
xlabel('Idő [s]');
ylabel('Beavatkozó erők [SI]');

%Maradó hiba (jelenleg a szimuláció vége)
e = [q51(end) - qd(1); q52(end) - qd(2); q53(end) - qd(3)]; %karok hibája
etcp = subs(r0tcp, [q1(t) q2(t) q3(t) l2 l3], [q51(end) q52(end) q53(end) L2 L3]) -↙
subs(r0tcp, [q1(t) q2(t) q3(t) l2 l3], [qd(1) qd(2) qd(3) L2 L3]);
disp(e);
disp(double(etcp(1:3)));

%%
%6. feladat

%Állapottér modell
ts6 = int32(0.5/h);

%KF és üres vektorok megadása
q61 = zeros(ts6, 1);
q61(1) = 0.55;
q62 = zeros(ts6, 1);
q63 = zeros(ts6, 1);
qp61 = zeros(ts6, 1);
```

```
qp62 = zeros(ts6, 1);
qp63 = zeros(ts6, 1);

v6 = - Kp*(q-qd) - Kd*qp;
u6 = Mq*v6 + Cq; %H = 1
%inverz dinamikai szabályozás
qpp6 = - Kp*(q-qd) - Kd*qp; %Gyorsulás és sebesség elvárt értéke 0
qpp61 = qpp6(1);
qpp62 = qpp6(2);
qpp63 = qpp6(3);
%%

for i = 1:ts6-1
    qp61(i+1) = qp61(i) + h*subs(qpp61, [diff(q1(t), t) diff(q2(t), t) diff(q3(t), t)↙
q1(t) q2(t) q3(t)], [qp61(i) qp62(i) qp63(i) q61(i) q62(i) q63(i)]);
    qp62(i+1) = qp62(i) + h*subs(qpp62, [diff(q1(t), t) diff(q2(t), t) diff(q3(t), t)↙
q1(t) q2(t) q3(t)], [qp61(i) qp62(i) qp63(i) q61(i) q62(i) q63(i)]);
    qp63(i+1) = qp63(i) + h*subs(qpp63, [diff(q1(t), t) diff(q2(t), t) diff(q3(t), t)↙
q1(t) q2(t) q3(t)], [qp61(i) qp62(i) qp63(i) q61(i) q62(i) q63(i)]);
    q61(i+1) = q61(i) + h*qp61(i);
    q62(i+1) = q62(i) + h*qp62(i);
    q63(i+1) = q63(i) + h*qp63(i);

end
%%

time6 = h:h:double(ts6)*h;
%csuklókoordináták
figure();
hold on
plot(time6, q61, 'b');
plot(time6, q62, 'c--', LineWidth=2);
plot(time6, q63, 'r');
title('Csuklókoordináták inverz tagokkal');
legend('q1', 'q2', 'q3');
xlabel('Idő [s]');
ylabel('Csuklókoordináták [m]/[rad]');
hold off
%beavatkozó erők
figure()
u61 = subs(u6(1), {diff(q1(t), t) diff(q2(t), t) diff(q3(t), t) q1(t) q2(t) q3(t)},↙
{qp61(:) qp62(:) qp63(:) q61(:) q62(:) q63(:)});
u62 = subs(u6(2), {diff(q1(t), t) diff(q2(t), t) diff(q3(t), t) q1(t) q2(t) q3(t)},↙
{qp61(:) qp62(:) qp63(:) q61(:) q62(:) q63(:)});
u63 = subs(u6(3), {diff(q1(t), t) diff(q2(t), t) diff(q3(t), t) q1(t) q2(t) q3(t)},↙
{qp61(:) qp62(:) qp63(:) q61(:) q62(:) q63(:)});
hold on
plot(time6, u61, 'b');
plot(time6, u62, 'c');
plot(time6, u63, 'r');
title('Beavatkozó erők inverz tagokkal');
legend('u1', 'u2', 'u3');
xlabel('Idő [s]');
ylabel('Beavatkozó erők [SI]');
%%
```

```
%Maradó hiba (jelenleg a szimuláció vége)
e6 = [q61(end) - qd(1); q62(end) - qd(2); q63(end) - qd(3)]; %karok hibája
etcp6 = subs(r0tcp, [q1(t) q2(t) q3(t) l2 l3], [q61(end) q62(end) q63(end) L2 L3]) -↙
subs(r0tcp, [q1(t) q2(t) q3(t) l2 l3], [qd(1) qd(2) qd(3) L2 L3]);
disp(e6);
disp(double(etcp6(1:3)));
```