

Robotkar irányítása optimális PID szabályzóval

Robot arm control with optimal PID

NAGY Zoltán¹, LENDEK Zsófia², NAGY Levente³

Kolozsvári Műszaki Egyetem, Automatika és számítógép mérnöki kar, Kolozsvár

¹PhD, Zoltan.Nagy@aut.utcluj.ro, +40752538983

²egyetemi docens, Zsofia.Lendek@aut.utcluj.ro

³egyetemi hallgató, levente.nagy960115@gmail.com

ABSTRACT

In this paper we compute an optimal PID controller for a robot arm to track a predefined trajectory. The model of the robot arm incorporates all the physical elements, like gravity and friction. Based on this model, we develop a PID controller. Using the parameters of the PID, further optimization is done to produce better results for trajectory tracking. We test both the original and the optimized controller in simulation and in real-time experiments.

ÖSSZEFOGLALÓ

Ebben a dolgozatban egy optimális PID szabályzót számoltunk egy adott pálya bejárása érdekében egy robotkarra. A létező robotkar alapján határoztuk meg a matematikai modellünket, mely figyelembe veszi a fizikai tényezőket, mint például a gravitációt és a súrlódást is. A modell alapján egy PID szabályzót készítettünk, melynek paramétereit tovább optimalizáltuk, hogy egy adott pálya minél pontosabb lekövetését kapjuk. Dolgozatunkban bemutatjuk a modell alapján kapott PID, és az optimalizált PID vezérlők működését először szimulációban, majd a valós rendszeren is.

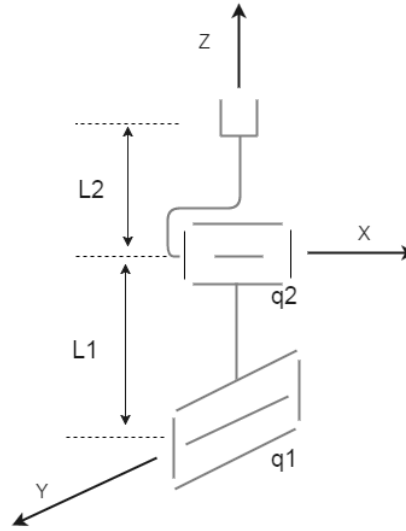
Kulcsszavak: robotkar, matematikai modell, PID szabályzó, optimalizálás

1. BEVEZETŐ

Az emberi munka megkönnyítése érdekében, illetve a hatékonyság növeléséért egyre több iparágban használnak robotkarokat. Ezen robotkarok gyakran ugyanazt a folyamatot kell végig járják több ezerszer minden nap. Egy robotkar pontos vezérléséhez egy matematikai modell szükséges, mely megfelelően leírja ennek működését. Ezen modell alapján tervezhető a szabályzó, mely irányítani fogja az adott robotkart. Dolgozatunkban arra törekedtünk, hogy egy adott feladat elvégzése esetén a szabályzó paramétereit optimalizáljuk, annak érdekében, hogy jobb teljesítményt kapjunk.

Az egyik legelterjedtebb szabályzó a PID (Proportional Integral Derivative). A PID szabályzó paramétereit kiszámításakor többnyire az adott rendszer lineáris modelljét használják, mely csak egy adott tartományban garantálja a rendszer stabilitását. Annak érdekében, hogy a robotkar működése az elvárásoknak megfelelő legyen egy adott pálya lekövetése esetén, további optimalizálás szükséges. Dolgozatunkban erre igyekeztünk megoldást keresni.

Az első lépésben a lineáris matematikai modellt határozzuk meg, majd erre egy alap PID szabályzót számolunk. Ezt a szabályzót a pálya jobb követése érdekében optimalizáljuk. Először a szimulációs eredményeket mutatjuk be, majd a valós rendszeren teszteljük a szabályzót.



Ábra 1 – Két csuklóponos robotkar

2. MATEMATIKAI MODELL

Mivel a cél a robotkar egy bizonyos pályán való minél pontosabb irányítása, a rendszer dinamikus modelljét használjuk, vagyis azt tanulmányozzuk, hogy a nyomaték változása hogyan hat a szögekre, illetve szögsebességekre. Az 1. Ábra bal felén látható a robotkarunk, melyen teszteltük a PID szabályzókat, ugyanakkor a robotkar vázlatos leírása kerül bemutatásra a 1. Ábra jobb felén. A dinamikus modell a két csuklóponos robotkar (1. Ábra) esetében a következőképpen írható fel [1]:

$$Q_m = M(q)\ddot{q} + (D(q, \dot{q}) + Q_f)\dot{q} + G(q) \quad (1)$$

ahol, $Q_m = \tau = [\tau_1 \quad \tau_2]^T$ a nyomaték vektor, $q = [q_1 \quad q_2]^T$ a szög vektor, $\dot{q} = [\dot{q}_1 \quad \dot{q}_2]^T$ a szögsebességek vektora, és $\ddot{q} = [\ddot{q}_1 \quad \ddot{q}_2]^T$ a szöggyorsulási vektor. Ugyanakkor $M(q)$ a tömeg mátrix, $D(q, \dot{q})$ tartalmazza a Coriolis és centrifugális mátrixokat, $Q_f = \begin{pmatrix} b_1 & 0 \\ 0 & b_2 \end{pmatrix}$ a súrlódási együtthatókat tartalmazza, $G(q)$ pedig a gravitációs mátrix.

Táblázat 1

Jelölés	Mértékegység	Érték	Leírás
L_1	[m]	0.095	Távolság az első és a második csukló között
L_2	[m]	0.1	Távolság a második a kar végpontja között
M_1	[kg]	0.095	Az első csukló és a hozzátartozó szegmens tömege
M_2	[kg]	0.37	A második csukló és a hozzátartozó szegmens tömege
I_{1x}	[kgm ²]	0.227e-1	Az első csukló tehetetlenségi nyomatéka az x tengelyre
I_{1y}	[kgm ²]	0.833e-4	Az első csukló tehetetlenségi nyomatéka az y tengelyre
I_{1z}	[kgm ²]	0.181e-4	Az első csukló tehetetlenségi nyomatéka az z tengelyre
I_{2x}	[kgm ²]	0.833e-4	A második csukló tehetetlenségi nyomatéka az x tengelyre
I_{2y}	[kgm ²]	0.227e-1	A második csukló tehetetlenségi nyomatéka az y tengelyre
I_{2z}	[kgm ²]	0.707e-4	A második csukló tehetetlenségi nyomatéka az z tengelyre
b_1	-	0.24	Súrlódási együttható az első csuklóra
b_2	-	0.16	Súrlódási együttható a második csuklóra

A mi robotkarunk esetében a paraméterek leírása és értékei az 1. Táblázatban található. Az (1)-es egyenlet klasszikus állapot tér formában a következőképpen írható fel:

$$\begin{aligned} \dot{q} &= \dot{q} \\ \ddot{q} &= -M(q)^{-1}G(q) - M(q)^{-1}(D(q, \dot{q}) + Q_f)\dot{q} + M(q)^{-1}Q_m \end{aligned} \quad (2)$$

A továbbiakban $x = [q_1 \quad q_2 \quad \dot{q}_1 \quad \dot{q}_2]^T$ jelöli az állapot vektort. A legegyszerűbb vezérlők lineáris állapotleírással vannak kifejlesztve, ezért először is linearizáljuk a rendszert egy operációs pont közelében. Rendszerünknek 2 egyensúly pontja van; a felfele mutató pozíció, amikor mindkét csuklópont szöge nulla, egy instabil egyensúlypont, ahogy ezt láthatjuk az 1. Ábrán is. Amikor viszont az első csuklópont szöge $x_1 = \pi$, és a második csuklópont szöge $x_2 = 0$ akkor egy stabil egyensúlypontot kapunk. A linearizálási pontnak az instabil egyensúlypontot választottuk, és ezáltal a következő lineáris állapotter rendszert kaptuk:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \quad (3)$$

$$A = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 18.566 & 0 & -7.809 & 0 \\ 0 & 7.696 & 0 & -6.785 \end{pmatrix}, \quad B = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 32.540 & 0 \\ 0 & 42.410 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

Mivel a csuklópontok szögei mérhetőek ezért a rendszer kimenete $y = Cx$, ahol C úgy van meghatározva, hogy pont ezt fejezze ki. A linearizálás következtében a két csuklópont teljesen függetlenné vált egymástól, vagyis az egyik csuklópont nyomaték változása nincs hatással a másik csuklópont szögeire. Ez alapján 2 átviteli függvény határozható meg, a két lineáris rendszerre, ahol a bemenet a nyomaték, és a kimenet a csuklópont szöge:

$$H_1(s) = \frac{32.54}{s^2 + 7.8s - 18.56}, \quad H_2(s) = \frac{42.41}{s^2 + 6.78s - 7.69} \quad (4)$$

3. PID SZABÁLYZÓ

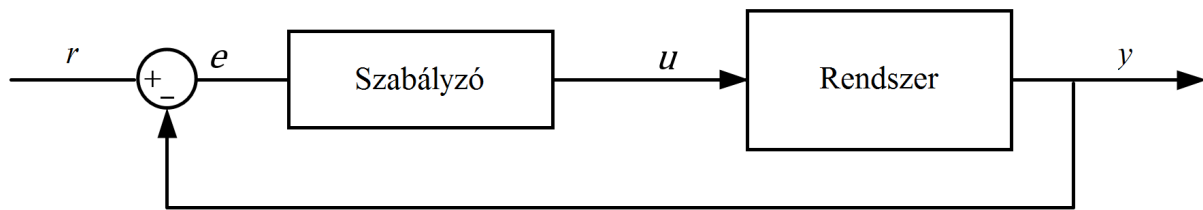
Negatív visszacsatolású szabályzó a 2. Ábrán látható. Egy folytonos idejű PID szabályzó esetén a bemenet:

$$u(t) = K_p e(t) + K_i \int_0^t e dt + K_d \dot{e}(t) \quad (5)$$

Ahol $e(t) = r(t) - y(t)$, vagyis a referencia és a kimenet különbsége, K_p, K_i, K_d pedig a szabályzó paraméterei. Jelen esetben két szabályzót kell meghatározni a (4)-ben említett két rendszerre. A szabályzó paramétereinek meghatározására többféle módszer is létezik, ilyen például a Ziegler-Nichols metódus [2], vagy a Bode diagram segítségével [3] is könnyen meghatározhatók. Jelen dolgozatban a gyökhely görbe módszer (root locus) [3] használtuk.

A mi rendszerünkkel egy pályát szeretnénk bejárni, minél gyorsabban és minél pontosabban, ennek érdekében a következő elvárásokat határoztuk meg: a beállási idő $t_s \leq 0.75s$, illetve a túllendülés $\sigma \leq 6.5\%$. Több szimuláció tesztelése után a következő paraméterek bizonyultak a legjobbnak erre a célra:

$$K_{p1} = 2.707, \quad K_{p2} = 1.450, \quad K_{i1} = 1.202, \quad K_{i2} = 1.142, \quad K_{d1} = 0.102, \quad K_{d2} = 0.102.$$



Ábra 2. Szabályzás

A valós rendszer vezérlése egy digitális vezérlő segítségével valósul meg, tehát ahhoz, hogy irányítani tudjuk egy diszkrét idejű szabályzó szükséges. Az (5) PID szabályt megközelíthetjük a következővel:

$$u(k) = K_p e(k) + K_i T_s \sum_{n=1}^k e(n) + K_d \frac{e(k) - e(k-1)}{T_s} \quad (6)$$

Jelen esetben a (4) alapján a 2 rendszer frekvenciái a következők: $\omega_{n1} = 4.3$, $\omega_{n2} = 2.77$. Mivel a mintavétel periódusa a valós rendszer esetében $T_s = 0.045$, ami közel 10^{-2} nagyságrenddel kisebb, mint a 2. rendszer frekvenciája, tehát a (6) egyenlet egy elég jó megközelítése az (5)-nek.

4. PARAMÉTEREK OPTIMALIZÁLÁSA

A PID paraméterek a linearizált rendszer alapján voltak kiszámítva, ahol a linearizálási pont a felfele mutató, instabil egyensúlypont volt. Mivel a valós rendszerünk több nem-lineáris elemet is tartalmaz, ezért a lineáris megközelítés nem garantálja mindenhol a fent említett elvárásokat. Annak érdekében, hogy ezek minél jobban teljesüljenek, optimalizáltuk a paramétereket figyelembe véve az adott pályát, amit követni akarunk, illetve a lineáris modell helyett a nem-lineáris modellt használtuk.

A pálya szempontjából egy egyszerű mozgást határoztunk meg, mely a szögek mozgását írja le az idő függvényében. A 3. Ábrán látható az alap paraméter beállítással kapott követése. Amint láthatjuk mindkét esetben elfogadható eredményeket kaptunk, ahol lehetne még javítani az az első csuklópontra, ahol megjelenik egy nagyobb túllendülés, illetve oszcillálás.

Mivel a cél a pálya minél jobb követése, ezért az objektív függvényt ez alapján határoztuk meg és a (7) egyenletben látható.

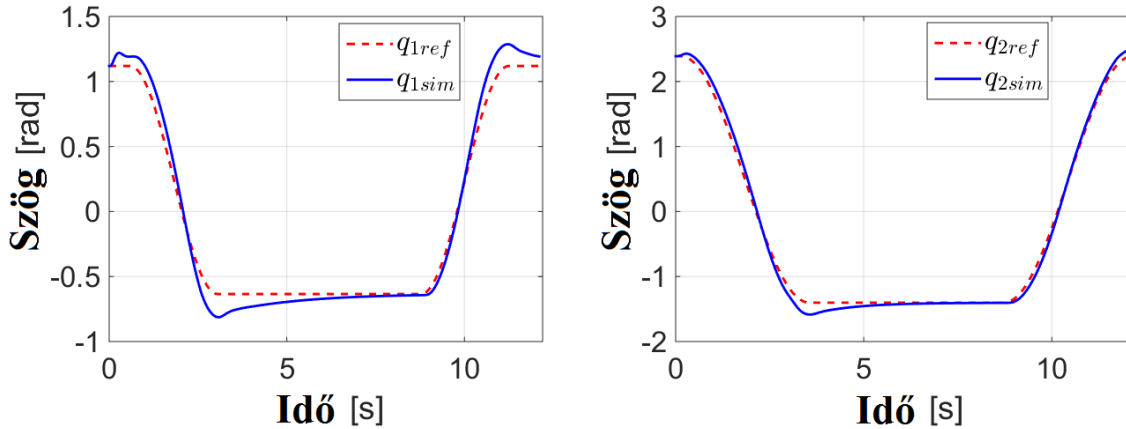
$$f(p) = k_1 \sum_{k=1}^m e_1(p, k)^2 + k_2 \sum_{k=1}^m e_2(p, k)^2 + k_3 (\max(e_1) + \max(e_2)) \quad (7)$$

A (7) egyenletben e_1 , e_2 a csuklópontra szöge és a referencia közötti különbség, p a paraméter vektor, és k_1 , k_2 a különbségek fontosságát jelöli; a túllendülés lecsökkentése érdekében a legnagyobb különbséget is figyelembe vettük, melynek fontosságát a k_3 paraméter határozza meg. Mivel az első csuklópontra terhelése nagyban függ a második csuklópontra pozíciójától, ezért több teszt után a következő súlyozást határoztuk meg az objektív függvényünknek: $k_1 = 7$, $k_2 = 1$, $k_3 = 10$.

Többféle optimalizálási módszer létezik mely megfelel erre a célra, ilyen például a Rosenbrock [4] módszer, a Nelder-Mead [4], vagy akár a genetikus algoritmusok [5]. Mivel a paraméterek amiket kaptunk elfogadható eredményeket adnak, ezért nem érdemes nagyon "eltávolodni" ezektől, vagyis elég ha egy lokális optimumot kapunk. A Nelder-Mead módszert használva a következő paramétereket kaptuk:

$$K_{p1} = 6.004, \quad K_{p2} = 4.642, \quad K_{i1} = 5.894, \quad K_{i2} = 0.678, \quad K_{d1} = 0.141, \quad K_{d2} = 0.087 \quad (8)$$

Amint megfigyeljük az első csuklópontra vonatkozó integrál paraméter jelentősen növekedett (értéke 5.894 vs 1.202), mely érthető, mivel minél pontosabb lekövetést szeretnénk. A kapott eredmények ezen paraméterek (8) segítségével a 4. Ábrán láthatók. Összehasonlítva az előző eredménnyel, jobb követést kapunk.



Ábra 3. Szimuláció PID paraméterekkel a nem-lineáris rendszeren

5. KISÉRLETI EREDMÉNYEK

A matematikai modellünk nem foglal magába minden dinamikát ami a valós rendszerben megjelenik, ezért előfordulhat, hogy az ami nagyon jó eredményt adott szimulációban kevésbé jól fog működni a valós rendszeren. Először a gyökhely görbe módszer alapján kapott paraméterekkel teszteltük a rendszert. A kapott eredmények a 5. Ábrán láthatók. Az objektív függvény értéke $f_{obj} = 10.33$, összehasonlítva ezt a szimulációs eredménnyel megállapíthatjuk, hogy a számított szabályzó az elvártnál jobban működik. Ezután az optimalizált paramétereket használva teszteltük a rendszert, $f_{obj} = 8.299$, az eredmények a 6. Ábrán láthatók. Bár jobb eredményt kaptunk, mint az alap paraméterekkel, a rendszer elkezdett oszcillálni. Ugyanakkor a nem-modellezett dinamika miatt a kapott eredmény jóval nagyobb, mint azt a szimuláció alapján elvártuk volna. Ugyanakkor, fontos megemlíteni, hogy a kapott eredmények erre az adott pályára voltak optimalizálva, vagyis előfordulhat, hogy egy más pályán ez a szabályzás egy instabil rendszert eredményez.

6. KONKLUZIÓK ÉS TOVÁBBI FEJLESZTÉSEK

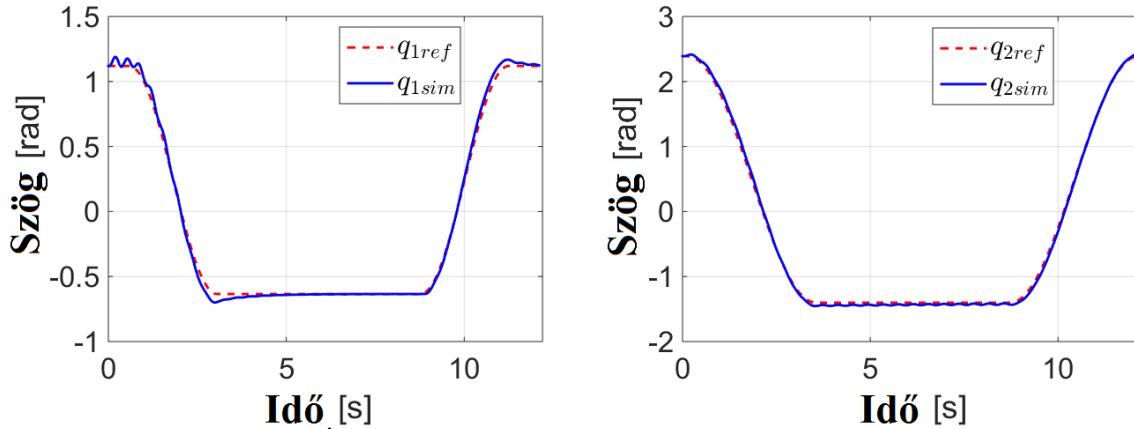
Ezen dolgozat célja egy 2 szabadságfokú robotkar pontos vezérlése volt. Ennek érdekében a lineáris modell alapján meghatároztunk egy-egy PID szabályzót mindkét csuklópontra. A kapott paramétereket tovább optimalizáltuk az adott követési pálya és a nem-lineáris modell segítségével. Az alapbeállítás, illetve az optimalizálás után kapott paramétereket a szimulációban és a valós rendszeren is teszteltük.

A lineáris modell alapján meghatározott PID vezérlés elfogadható eredményt adott a szimulációban. A valós rendszeren való teszteléskor még jobb eredményt kaptunk, mint a szimulációban. Ezzel szemben viszont az optimális paraméter vektor gyengébb eredményt adott, mint ahogy azt elvártuk volna a szimuláció alapján (4. Ábra összehasonlítva a 6. Ábrával), viszont jobb eredményt kaptunk, mint az alapbeállítással.

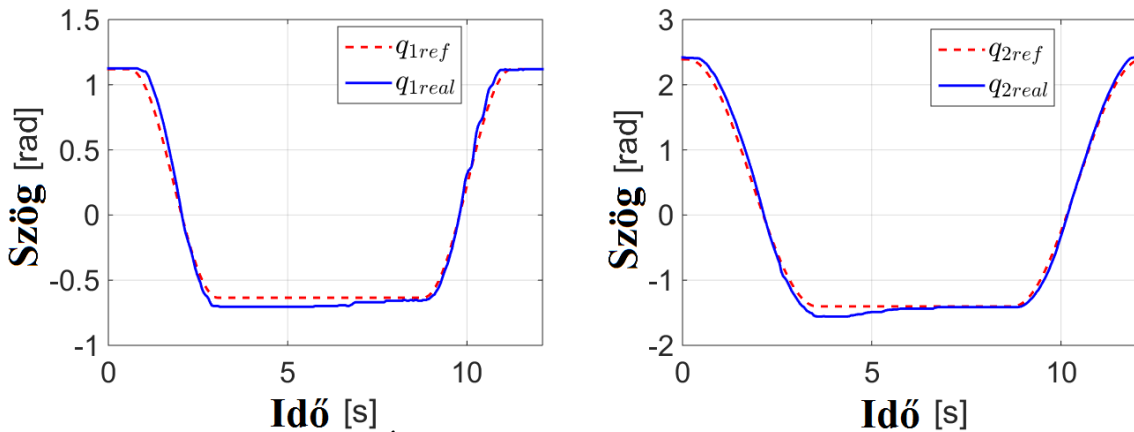
További fejlesztésként szeretnénk tesztelni más optimalizálási algoritmusokat is, melyeket majd akár a valós rendszer segítségével is lehetne tesztelni. A valós rendszer tesztelése szempontjából olyan algoritmust kell válasszunk mely kevés valós futtatást igényel. Egy másik tovább fejlesztési lehetőség a pontosabb szabályzás érdekében egy nem-lineáris szabályzó meghatározása.

7. KÖSZÖNETNYILVÁNÍTÁS

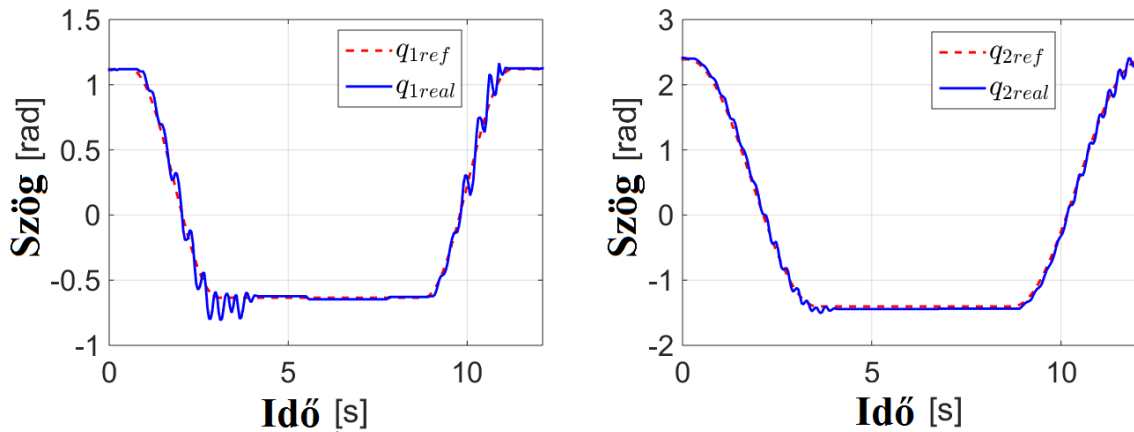
Ez a dolgozat a Román Nemzeti Tudományos Kutatási Hivatal projektjének, CNCS-UEFISCDI, projekt szám PN-II-RU-TE-2014-4-0942, szerződésszám 88/01.10.2015, támogatásával jött létre.



Ábra 4. Szimuláció optimális PID paraméterekkel



Ábra 5. Valós rendszer PID vezérléssel



Ábra 6. Valós rendszer optimális PID vezérléssel

8. HIVATKOZÁSOK

- [1] I Negrean, I Vușcan, N Haiduc, 1997. Robotică. Modelarea cinematică și dinamică, Editura Didactică și Pedagogică
- [2] J.G. Ziegler, N.B. Nichols, 1942. Optimum Settings for Automatic Controllers,
- [3] K. Ogata, 1997. Modern Control Engineering. Prentice Hall
- [4] Zs. Lendek, 2013. Optimization exercises, lecture notes. Cluj Napoca
- [5] Z. Nagy, F. Fodor, Zs. Simon, Zs. Lendek, 2015. Genetikus algoritmusok multiobjektív fakitermelési tervhez, EMT, ENELKO-SzámOkt. Arad