

# Intelligens ágens alapú humanoid robot vezérlés

## Intelligent agent based control of humanoid robot

KOVÁCS Attila<sup>1</sup>, dr. BRASSAI Sándor Tihamér<sup>2</sup>

Sapientia Erdélyi Magyar Tudományegyetem Marosvásárhelyi Kar,

<sup>1</sup>Számítógépes Irányítási Rendszerek Mesteri szak,

attilakovacs508@gmail.com

<sup>2</sup>Villamosmérnöki Tanszék,

Maros megye, Koronka 1c, www.ms.sapientia.ro,

tihai@ms.sapientia.ro

### Abstract

*The subject of the research is to investigate ways to teach an intelligent agent to be able to control a customly designed and built humanoid robot, that has 26 revolute joints. The goal to achieve for the agent, in terms of control is to be able to keep itself balanced, furthermore to learn to develop an efficient walking gait, through sequences of trials. The learning method is based on reinforcement learning, for the actual learning the three dimensional mechanical model was used, paired with a simulation environment specifically designed for this application.*

### Kivonat

*A kutatás témája, annak vizsgálata, hogy milyen módon tudunk egy intelligens ágenszt megtanítani egy saját tervezésű, huszonhat forgásponttal rendelkező humanoid robot vezérlésére. A munka célja annak elérése, hogy az ágens próbálkozások során tanulja meg, hogy milyen módon képes az egyensúlya fenntartására, illetve, hogy milyen mozgási szekvenciák révén tud hatékonyan, egyenesvonalban mozogni. A tanítás, a megerősítő tanulási módszer alapján valósult meg, a vezérelni kívánt robot háromdimenziós mechanikai modellje és egy erre a célra kialakított szimulációs környezet révén.*

**Kulcsszavak** : Intelligens ágens, humanoid robot, vezérlés

## 1. A humanoid robot

### 1.1. A robot tervezése

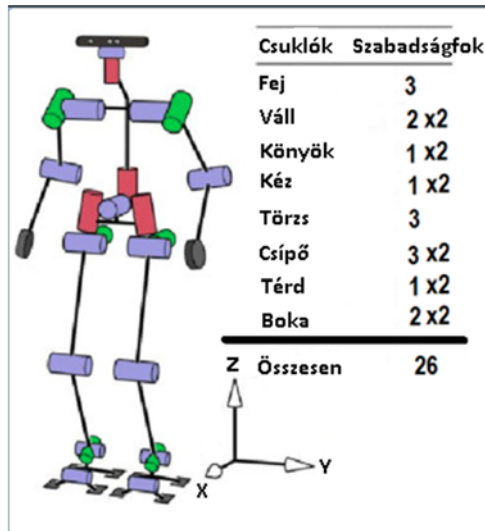
A munka első fázisa a vezérelni kívánt humanoid robot megtervezéséből állt.

Az intelligens ágens általi vezérelhetőség érdekében egy olyan robot megtervezése szükségeltetett, amely rendelkezik a megfelelő szabadságfokokkal ahhoz, hogy emberszerűen viselkedhessen, emellett viszonylag kisméretű és könnyű.

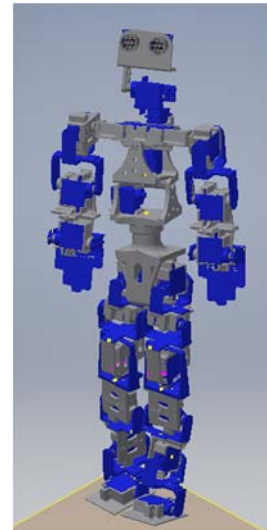
Ennek érdekében a robot tervezésének főbb követelményei a következők voltak:

- Az emberi mozgásformát lehetővé tévő főbb forgáspontok megtartása (1. ábra).
- Megközelítőleges magasság : 40-50 cm
- Megközelítőleges tömeg: 1-1.5 kg
- Moduláris szerkezet
- 3D nyomtatható elemek

A megtervezett és virtuálisan összeszerelt robot a 2. ábrán van szemléltetve.



1. ábra. A robot forgáspontjai



2. ábra. Az elkészült robot terv

## 1.2. A robot megépítése

A tervezési folyamattal párhuzamosan a robot vázának elemei 3D nyomtatással készültek el, amely egy gyors, egyszerű és költséghatékony módot nyújt az elemek legyártására és szükség esetén azoknak helyettesítésére.

Továbbá ilyen módon a legyártott elemek súlya és szerkezeti tartóképesége beállítható a nyomtatási paraméterek módosításával.

A megvalósított és megépített robot a 3. ábrán látható.

## 1.3. A szimulációs modell megalkotása

A megtervezett robot alapján elkészült a robot Simulink környezetben lévő szimulációs modellje, a forgáspontokban vannak elhelyezve a beavatkozók, amelyek mozgatják a robotot a szimulációs környezetben.

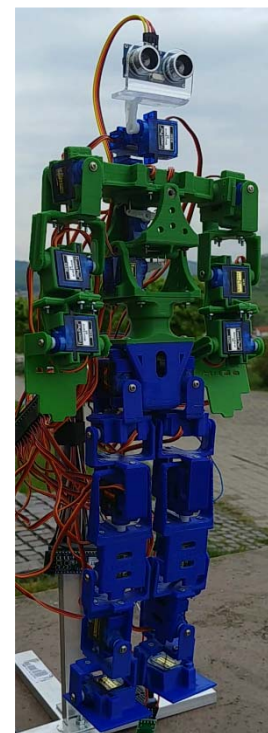
A szimulációs modell a beavatkozók szögszabályozás általi vezérlésére vannak kialakítva, ez három féle módon mehet végbe a kiválasztott beavatkozó szerint:

- I. Szögszabályozás nyomatékkorlát nélkül, ideális szervomotor modell (4. ábra).
- II. Szögszabályozás nyomatékkorlással rendelkező szervomotor modell alapján (5. ábra).
- III. Szögszabályozás nyomatékkorlással rendelkező, valós szervomotor modell alapján a mechanikai és villamos jellemzők figyelembe vételével (6. ábra).

A nyomatékkorlát nélküli szervomodell alkalmazásával a szimuláció futási sebessége megnő, így a tanítás rövidebb időt igényel, viszont nem reprezentálja a rendszert elég részletesen ahhoz, hogy az ágens a szimulációk alatt megtanulja a valós rendszert vezérelni.

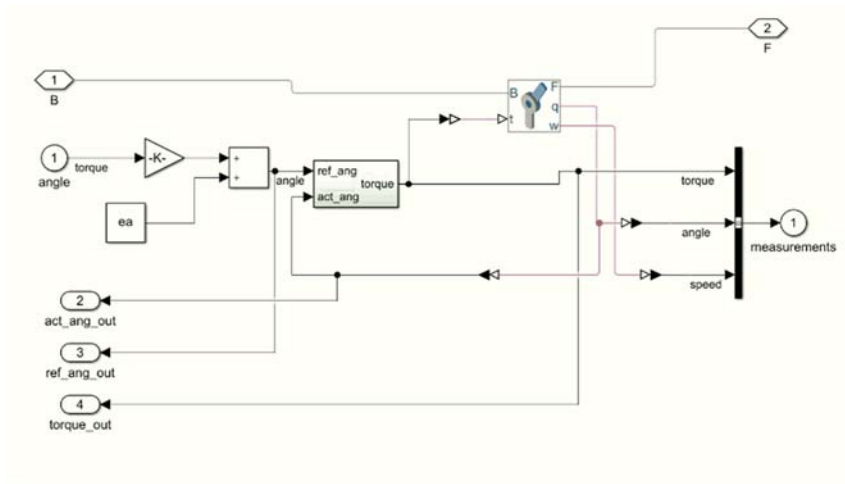
A nyomatékkorlással rendelkező modell esetén a futási idő nem növekszik meg jelentősen, viszont már képes a rendszert a megfelelő módon leírni, ezen okok miatt a tanítás során ez volt használva.

A részletes szervomodell alkalmazásával nagyon pontos leírását érhetjük el a valós rendszernek, viszont a szimuláció futási ideje akár az ötödére is csökkenhet.

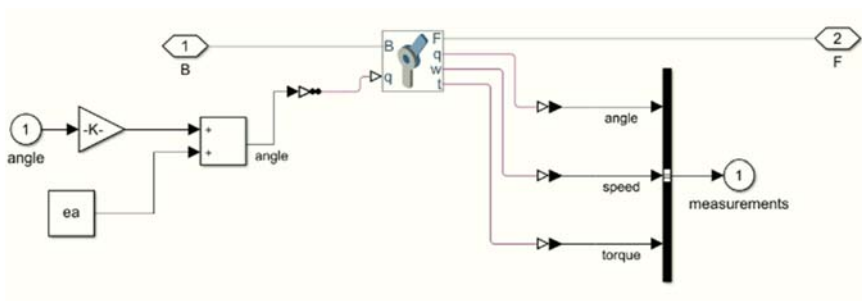


3. ábra

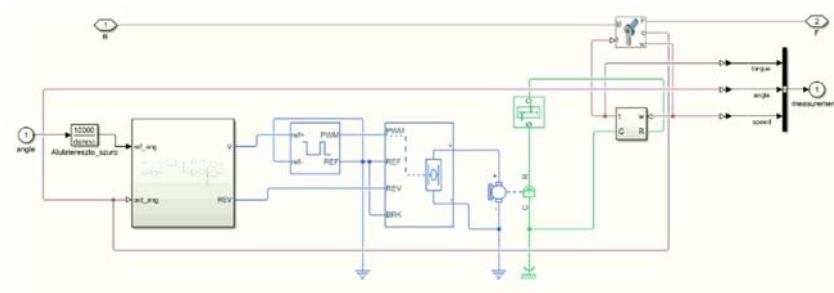
A megépített robot



4. ábra. Az ideális szervomodell

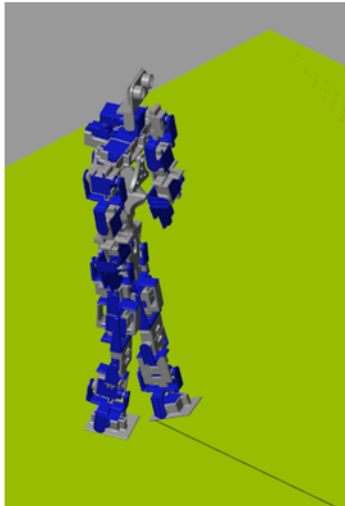


5. ábra. A nyomatékkorláttal rendelkező szervomodell

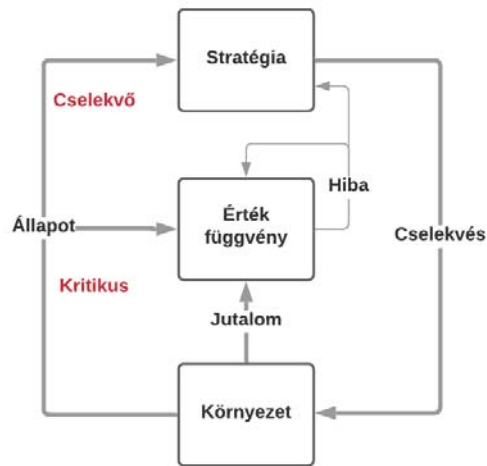


6. ábra. A valós szervomodell

Az elkészült szimulációs modellje a robotnak a 7. ábrán látható, ez megfelel a robot fizikai paramétereinek méret és tömeg szempontjából.



7. ábra. A robot szimulációs modellje



8. ábra. A cselekvő-kritikus módszer

## 2. Az ágens jellemzői

### 2.1. Az ágens jellemzői

Az ágens két fő részből áll (8. ábra):

- I. A stratégia (policy)
- II. A megerősítő tanulási algoritmus (RL algorithm)

#### 2.1.1. A stratégia

A stratégia típusának kiválasztásánál a döntés a cselekvő-kritikus (Actor-critic) módszerre esett mivel ez a tanulás folyamatát gyorsítja a kritikus egység alkalmazásával.

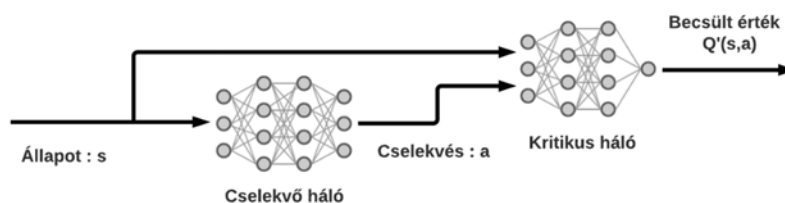
A vezérlésre a cselekvő egység van felhasználva, viszont ha azt szeretnénk, hogy az ágens a valós rendszeren is képes legyen tovább tanulni, akkor ezt megtehetjük a kritikus egység a robotba való ültetése révén.

#### 2.1.2. A megerősítő tanulási algoritmus

Az ágens tanítására a mély determinisztikus stratégia gradiens (Deep deterministic policy gradient) algoritmus van használva, egyik legfontosabb érv emellett a tanulási algoritmus mellett, hogy képes a cselekvő-kritikus típusú ágens tanítását elvégezni. Az algoritmus mély hálók esetében használatos, továbbá, mivel determinisztikus jellegű, emiatt a sztochasztikus módszerekhez képest általánosan gyorsabb konvergenciával rendelkezik. A hálók tanítására a stratégia gradiensét használja fel, emellett viszonylag nagy varianciával rendelkezik.

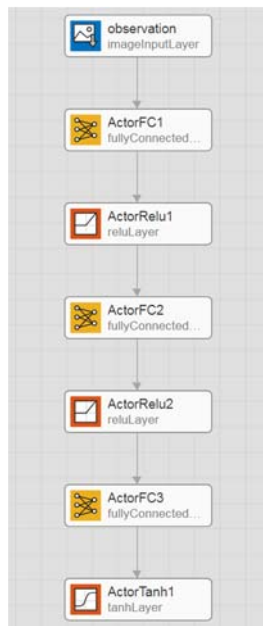
### 2.2. Az ágens megvalósítása

Az ágens stratégia egységének a szerkezete a 9. ábrán látható.



9. ábra. A cselekvő-kritikus architektúra

A cselekvő háló szerkezete a 10. ábrán látható, míg a kritikus háló szerkezete a 11. ábrán.



10. ábra. A cselekvő háló architektúrája



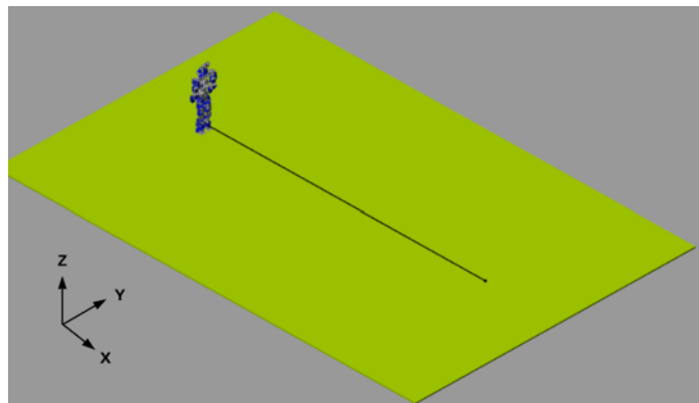
11. ábra. A kritikus háló architektúrája

### 3. Az ágens tanítása

#### 3.1. Megerősítő tanulás alkalmazása

A robot szimulációs modelljét az ágens egy szimulációs környezetben tanulja meg vezérelni, ez a 12. ábrán látható.

Az ágens feladata, hogy a robotot az egyenes vonal mentén vezérelje előre emberszerű mozgásnak megfelelően.



#### 3.2. A jutalom függvény

Az ágens megerősítő tanítással tanulja meg a robot vezérlését, a jutalomfüggvény által szolgáltatott epizód érték alapján tudja, hogy az adott cselekvés sorozat mennyire helyes. A jutalomfüggvényt a 1. egyenlet írja le.

$$J_n = (n * 0,0625) + (5 * chs) + (2 * mt) - ((koh - aoh) * 20)^2 - ((km - am) * 100)^2 + \left| \frac{1}{2} * szgy \right|$$

Egyenlet 1. A jutalomfüggvény

- ehs - előrehaladási sebesség (X tengely)
- mt - megtett távolság (X tengely)
- koh - kezdeti oldal helyzet (Y tengely)
- aoh - aktuális oldal helyzet
- km - kezdeti magasság (Z tengely)
- am - aktuális magasság
- szgy - szöggyorsulás (Y tengely)

### 4. Eredmények

A tanulás összesen 379 órát, azaz közel 16 napot vett igénybe, viszont a megerősítő tanulás tipikusan lassú fejlődést mutat az első időszakokban.

Ez alatt az idő alatt az ágensnek sikerült megtanulnia a robot egyensúlyának megtartását, és megtett 3-4 lépést elesés nélkül. A következő futamok során a járási mozgásforma elsajátítása következhet.

Később, mikor az ágens megfelelő mennyiségű tapasztalattal rendelkezik, akkor a tanulás gyakorta exponenciális mértékben jobb eredménnyel zárul.

## 5. Hivatkozások

- [1] Aström, K. J., Albertos, P., Blanke, M., Isidori, A., Schaufelberger, W., Sanz, R.(Eds.). (2011). Control of complex systems. Springer Science Business Media.
- [2] Kojima, K., Karasawa, T., Kozuki, T., Kuroiwa, E., Yukizaki, S., Iwaishi, S., ...Nozawa, S. (2015, November). Development of life-sized high-power humanoidrobot jaxon for real-world use. In 2015 IEEE-RAS 15th International Conferenceon Humanoid Robots (Humanoids) (pp. 838-843). IEEE
- [3] Bengio, Y. (2012). Practical recommendations for gradient-based training of deeparchitectures. In Neural networks: Tricks of the trade (pp. 437-478). Springer,Berlin, Heidelberg.
- [4] Mathworks, C., Simscape, T. (2016). Multibody (tm) getting started guide, 2016.
- [5] Liebgott, I. (2016). Modeling and Simulation of Multi-Physics Systems with MATLAB and Simulink.
- [7] Van Hasselt, H. (2012). Reinforcement learning in continuous state and actionspaces. In Reinforcement learning (pp. 207-251). Springer, Berlin, Heidelberg.