

Robocode bot evolválása genetikus programozással

PETŐ ZOLTÁN

BMMNO9

MÉRNÖK-INFORMATIKUS 3. ÉVFOLYAM, INTELLIGENS RENDSZEREK ÁGAZAT

PEZO1919@GMAIL.COM

KONZULENS: KOVÁCS DÁNIEL LÁSZLÓ

BME-VIK-MIT

DKOVACS@MIT.BME.HU

2014.05.16

Tartalom

2/11

- ▶ A Robocode
- ▶ Evolúciós számítási módszerek
- ▶ Rendszerterv
- ▶ Megvalósítás, felhasznált eszközök
- ▶ Kiértékelés, futási eredmények
- ▶ Demo
- ▶ Összefoglalás, kiértékelés

A Robocode robotok – mint intelligens ágensek

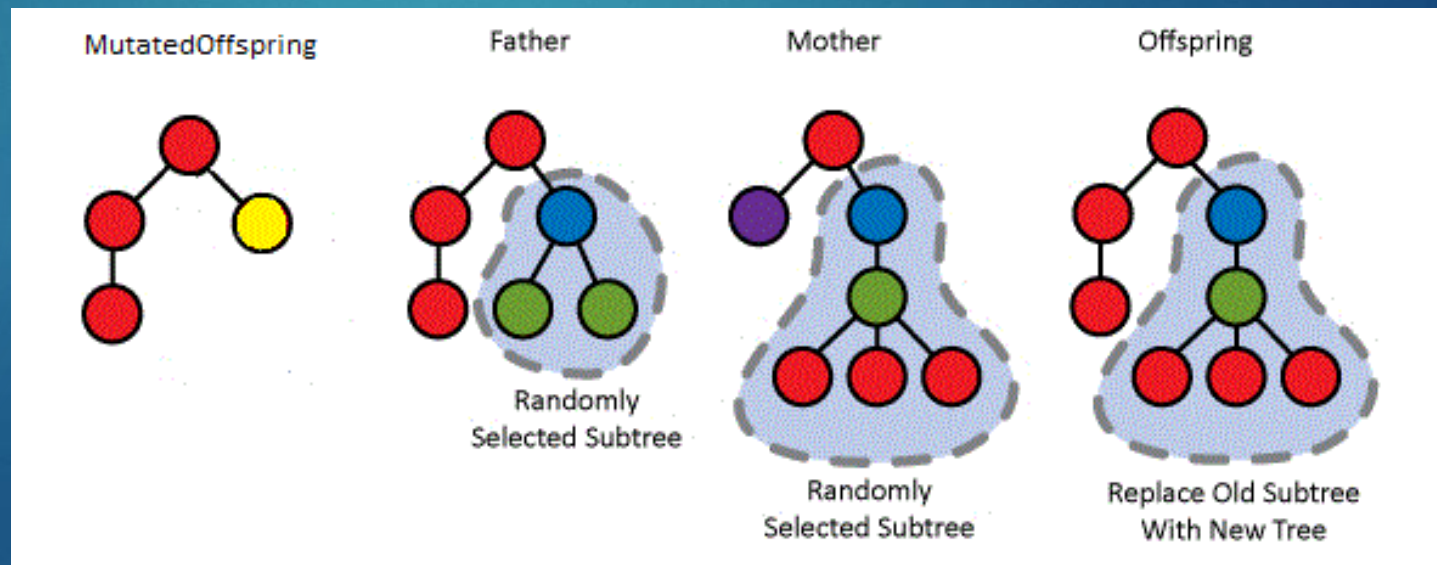
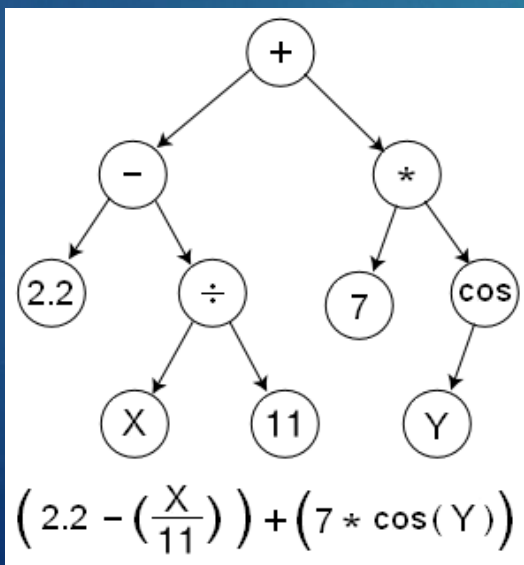
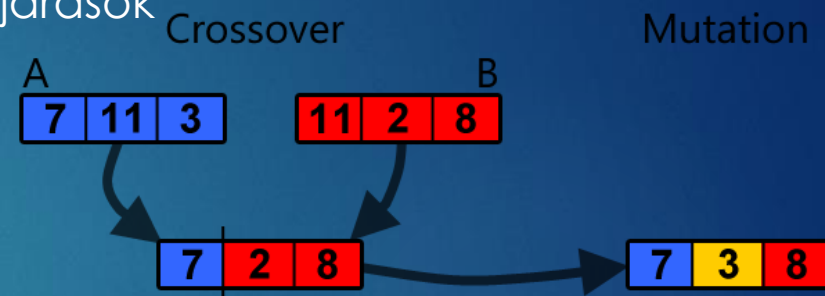
- ▶ A Robocode alapjai, szabályok, célok
 - ▶ Mit tud egy robot?
- ▶ Online versenyek, értékelés, pontozás
 - ▶ <http://literumble.appspot.com/>
 - ▶ Versenyszámok
 - ▶ **RoboRumble (1v1)**
 - ▶ MeeleRumble (10 bot, FFA)
 - ▶ TeamRumble (5v5)
 - ▶ TwinDuel (2v2)
 - ▶ Kategóriák méret szerint
 - ▶ Korábbi próbálkozások



Az evolúciós optimalizálási módszerek

4/11

- ▶ Globális optimumot meghatározó sztochasztikus keresőeljárások
 - ▶ Fitneszt, azaz „jósági mércét” használnak az eljárás során.
- ▶ Genetikus Algoritmusok – az egyedek bitfüzéreik
Genetikus Programozás – az egyedek program-fák (bottom-up kiértékelés)
- ▶ Az evolúció lépései
 - ▶ Szelekció – a populációból kiválasztódnak a köv. populáció egyedei – általában a fitneszt figyelembe véve
 - ▶ Mutáció – egy egyed (programja, vagy bitfüzére) részben megváltozik
 - ▶ Keresztezés – két (vagy több) ősz egyedből létrejön(nek) utód egyedek



Rendszerterv I.

5/11

▶ Érzékelések:

- ▶ Ellenfél sebessége, saját sebesség, ellenféltől való távolság, ellenfél energiája, saját energia
- ▶ Ellenfél iránya a tank törzséhez képest
- ▶ Saját pozíció

▶ Tudásbázis

- ▶ Ellenfel sebessege / tavolsag
- ▶ Ellenfel atlagsebessege, Saját atlagsebesseg
- ▶ Ellenfél közeledése/távolodása
- ▶ Saját energia, ellenfel energiája
- ▶ Falaktól való távolság (x,y)

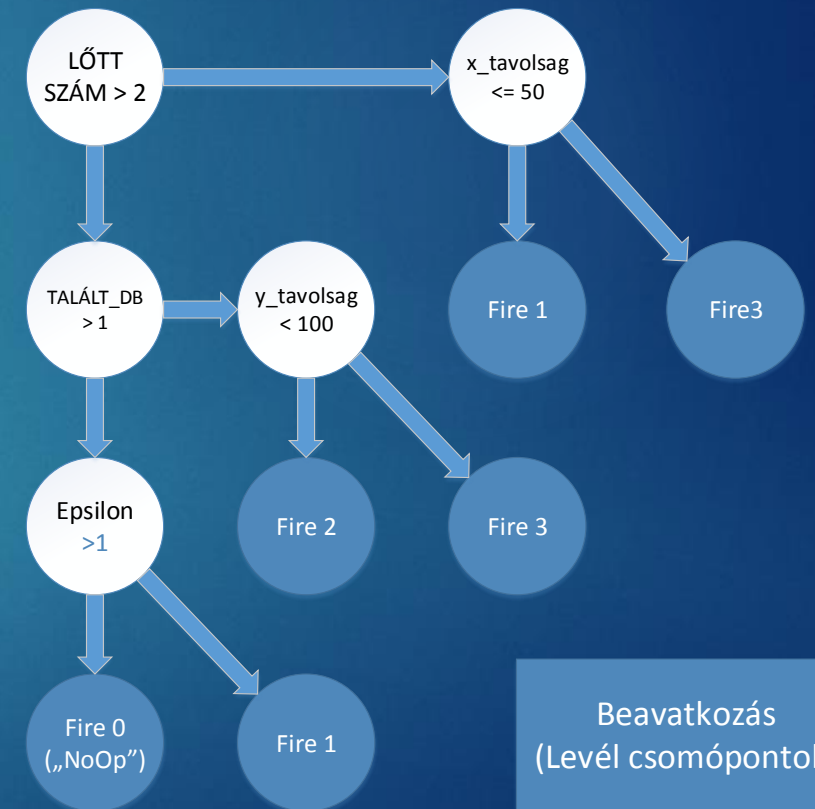
▶ Döntéshozás

- ▶ „Tavolabb van-e a fal „e” egységnel?”
- ▶ „Nagyobb-e a sebességem „s”-nél?”

▶ Beavatkozás

- ▶ 3 fa: Lövés erőssége, Fordulás iránya és mértéke, Gyorsítás iránya és mértéke

Döntéshozás
(Belső csomópontok)



Beavatkozás
(Levél csomópontok)

Rendszerterv II.

6/11

- ▶ Fitness – az ellenfél megmaradt energiája alapján
- ▶ Az evolúciós keretrendszer egy kezdeti populációt és egy kezdeti fitness függvényt alapul véve újabb és újabb populációt hoz létre az evolúciós operátorokkal:
 - ▶ rulett alapú szelekció, (opcionális) elitizmus, mutáció (egy adott részfa helyén új részfa születik), kereszteződés (két részfa kicserélődik a szülők megfelelő fáiban)



A megvalósítás, eszközök – miért éppen ezek?

7/11

Platformválasztás – Java – a Robocode Java alapú

- ▶ IDE: IntelliJ Idea 13.1.1 Community Edition
- ▶ Windows 8, 64bit
- ▶ Intel Core 2 Duo 2.26 Ghz, 4GB ram
- ▶ A Watchmaker (v.0.7.1) keretrendszer
 - ▶ Szelekció, mutáció, kereszteződés – és paramétereik
 - ▶ További evolúciós paraméterek
 - ▶ Elitizmus, populáció méret, stb.
 - ▶ 100 turnre „emlékszünk”
- ▶ Beépített botok – Walls, SpinBot, VelociRobot, Fire
- ▶ Versenyző bot - DrussGT

Kiértékelés, futási eredmények

8/11

- ▶ A beépített botok ellen könnyen győzedelmesedik a robot, de ez betudható a „Head-On” targeting mechanizmusnak.
- ▶ Bár a DrussGT ellen nagyon kis esély van győzedelmeskedni – ez még sem lehetetlen. Ehhez szükség van az ellenfél óriási hibáját kihasználni. (A demóban ez látható lesz.)
- ▶ A célzási mechanizmusból és a fitness függvényből következik, hogy azok a robotok fognak jól szerepelni, akik sikeresen kikerülnek az ellenfél lövedékeit, a bevitt találatok csupán elenyészően számítanak. Látható, hogy egészen jó mozgáskultúra fejlődött ki DrussGT ellen.

pm	Max mélység	Belső Node esély	Körök száma/Meccs	Populáció méret	Elitek	Generáció	Futási idő (min)	Normált fitness
0.1	4	0.75	10	20	10	25	25	51.9
0.2	6	0.82	10	20	10	31	22	68.7
0.3	3	0.65	10	10	3	26	21	73.6
0.2	6	0.65	10	20	5	28	22	12.4 !
0.2	10	0.85	5	20	5	80	20	64.2

Demo

9/11

Összefoglalás, kiértékelés

10/11

- ▶ Az önálló laboratórium során:
 - ▶ Megterveztem egy Robocode bot-ok GP-alapú evolúciós optimalizációját biztosító rendszert
 - ▶ Érzékelés, Tudásbázis, Döntéshozás, Beavatkozás
 - ▶ Implementáltam a rendszert a Watchmaker framework felhasználásával
 - ▶ Többféleképp teszteltem az így elkészített rendszert
- ▶ Továbbfejlesztési lehetőségek:
 - ▶ A 4. fa optimalizálása: további érzékelések és a tudásbázis bővítésével
 - ▶ Head-On targeting helyett bizonyos szöggel az ellenfél elé célzás.
 - ▶ Ellenfélmodellezés
 - ▶ GP kiegészítése Neurális Hálózatokkal – koevolúció
 - ▶ Az párhuzamosan futó evolúciók felváltva billenthetik ki egymást egy lokális minimumból.
 - ▶ Többi Robocode versenyszámban való részvétel (több vs több)
 - ▶ Evobot saját maga ellen játszotása (öntanulás)

Köszönöm a figyelmet!

Kérdések?