

Schrodinger's Shooter Smart-Bot



Đaković Branko, Kristić Filip, Krčmarević Mladen

Pregled

1. Uvod

- Opis problema.

2. Algoritam i realizacija

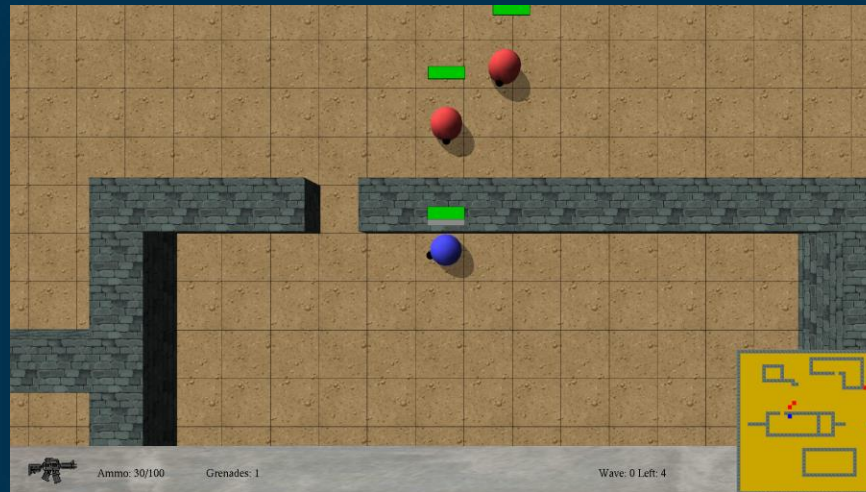
- Struktura igrača
- Genetski algoritam

3. Trening i upoređivanje rešenja

- Prvo rešenje
- Drugo rešenje

Opis problema

- “Schrodinger`s shooter” je mini igra u “top-down shooter” stilu, napravljena kao projekat za kurs Razvoj Softvera.
- Zadatak ovog projekta je implementacija programa koji će igrati igru umesto ljudskog igrača.
- Za realizaciju korišćene su kombinovane tehnike "reinforced learninga", genetskih algoritama i neuronskih mreža.



Struktura igrača

- Igrač je predstavljen neuronskom mrežom koja u svakom intervalu na osnovu unosa koji predstavlja okolinu igrača treba da generiše izlaz tj. odgovarajuću akciju.
- Moguće akcije su kretanje po x i y osi i njihove kombinacije, rotiranje nišana za dati ugao, pucanje i repetiranje.
- Izbačene su kompleksne akcije poput granate, različitog oružja i pancira.

Neuronska mreža

- Za realizaciju mreže korišćena je FANN biblioteka sa omotačem za C++.
- Mreža se sastoji iz tri sloja:
 - Ulaznog – koji se razlikuje u zavisnosti na koji način se predstavlja okolina igrača
 - Skrivenog – veličine 10 u prvoj realizaciji i 8 u drugoj.
 - Izlaznog – veličine 5 čiji izlazi su u intervalu [0-1] i predstavljaju odgovarajuću akciju.

Neuronska mreža

- Za aktivacionu funkciju korišćena je linearna aproksimacija sigmoidne funkcije:

$$y = \frac{1}{1 + e^{-2sx}}$$
$$d = 2sy(1 - y)$$

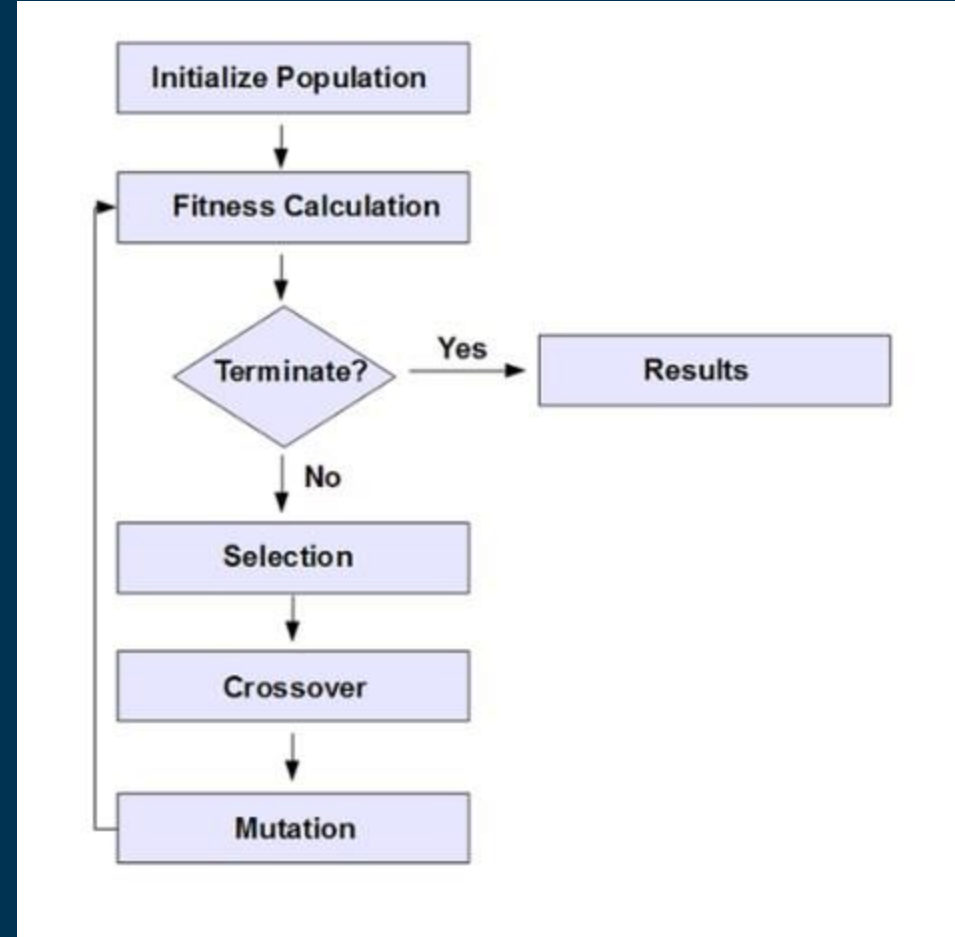
- Gde je x – ulaz, y – izlaz, s – nagib, d – derivacija.

Genetski algoritam

- Učenje se vrši genetskim algoritmom.
- Populaciju čini skup neuronskih mreža (tj. skup nizova težina neuronskih mreža) koje igraju igru sekvencijalno i u zavisnosti od ishoda dobijaju određenu ocenu fitnesa.
- $\text{Fitness} = \text{eliminacije} * f_e + \text{šteta} * f_š$
 - f_e – faktor eliminacije postavljen na 50 da bi eliminacije znatno uticale na fitnes.
 - $f_š$ – faktor štete postavljen na 0.5 da bi nanesena šteta uticala, ali znatno manje, na fitnes.
- Vreme preživljavanja je prvobitno uticalo na fitnes ali je izbačeno zbog prevelikih varijacija.

Genetski algoritam

- Jedinku populacije čini niz realnih vrednosti koje predstavljaju težine konekcija date neuronske mreže.
- Cilj algoritma je da kroz svaku iteraciju, promenom težina neuronskih mreža, stvori sve bolju i bolju populaciju.
- Inicijalna populacija je generisana nasumičnim postavljanjem težinskih vrednosti iz intervala $[-10, 10]$.



Selekcija, ukrštanje, mutacija

- Selekcija – Ruletska selekcija
- Ukrštanje i mutacija su različito implementirani za 1. i 2. rešenje

▪ Za prvo:

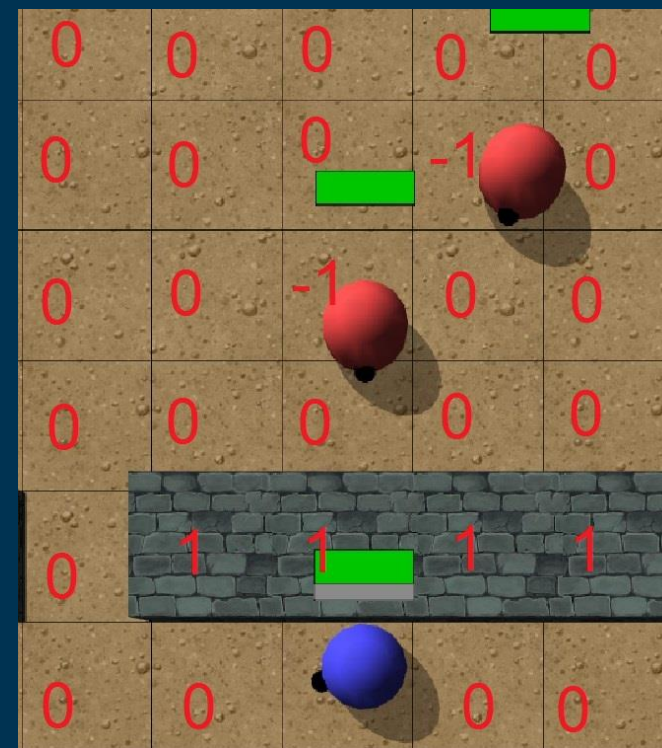
- Ukrštanje – Biraju se dva roditelja iz populacije za reprodukciju, svaka 2 daju 2 deteta:
 - $\text{dete1} = \text{roditelj1 do } i + \text{roditelj2 od } i$
 - $\text{dete2} = \text{roditelj2 do } i + \text{roditelj1 od } i$
 - gde je i nasumično izabran broj.
- Mutacija – nasumično se bira gen za mutaciju i broj t iz $[0-1]$ ako je t veće od stope mutacije promeni gen, u suprotnom ne.

▪ Za drugo:

- Ukrštanje – Biraju se dva roditelja iz populacije za reprodukciju, svaka 2 daju 2 deteta:
 - Za svaki gen bira se sa 50% šanse da li je iz prvog ili iz drugog roditelja za prvo ili drugo dete.
- Mutacija – za svaki gen se vrši mutacija po prethodnom principu.

Trening i upoređivanje rešenja

- Prvo rešenje:
- Ulaz neuronske mreže predstavlja celokupnu okolinu igrača (polja) u matričnoj formi tako da:
 - prazno polje = 0
 - protivnik = -1
 - zid = 1
- Jedno dodatno polje za trenutnu municiju.
- Zbog veličine prostora igre u ovom pristupu bilo je čak 191 ulaznih čvorova.



Trening i upoređivanje rešenja

- Trening i zaključak:

- Veličina populacije za trening: 300
- Broj iteracija: 300
- Vreme trajanja oko 10h

Karakteristike hardvera na kome je vršen trening:

- CPU: intel-i5 2500k
- GPU: AMD Radeon 6850
- RAM: 4GB DDR3
- OS: Ubuntu 16.04

- Dobijena konfiguracija nije davala smislene rezultate te je zaključeno da ovaj pristup nije adekvatan pa je pokušao drugi.

Trening i upoređivanje rešenja

- Drugo rešenje:

- Ulaz je drastično smanjen. Sastoji se iz 10 čvorova.
- `input[0]` = Ugao ka najbližem vidljivom protivniku.
- `input[1]` = Broj raspoložive municije.
- `input[2 - 9]` = Senzori:
 - Linije u 8 pravaca (gore, dole, levo, desno, dijagonale) koje detektuju da li se nešto (zid ili protivnik) nalazi od igrača do kraja ekrana.

Trening i upoređivanje rešenja

- Trening i zaključak:
- Veličina populacije za trening: 100
- Broj iteracija: 300
- Napomena: u ovom treningu dodata je izmena da se za svaku jedinku odigra 5 partija, te da se konačni fitnes računa kao srednja vrednost svih, da bi se smanjio slučajan uticaj.
- Vreme izvršavanja: oko 8h

Karakteristike hardvera na kome je vršen trening:

- CPU: intel-i5 7300hq
- GPU: NVIDIA GeForce 1060
- RAM: 16GB DDR4
- OS: Windows subsystem for Linux – Ubuntu 18.04

Trening i upoređivanje rešenja

- Zaključak:
- Poslednji trening dao je, kao i prvi pristup, nezadovoljavajuće rezultate, te je utvrđeno da ovo nije odgovarajući pristup za dati problem.
- Alternativa i moguće unapređenje: NEAT Algoritam.

The background features a diagonal design. The upper-left portion is a light blue triangle. The lower-right portion is a dark blue triangle. A thin, dark blue diagonal line separates these two areas. A thin, light blue diagonal line runs parallel to the dark blue line, creating a narrow, light blue diagonal band.

Hvala na pažnji.

Literatura:

- Dr Predrag Janičić and Dr Mladen Nikolić . *Veštačka Inteligencija*. Beograd, 2019.
- Kantardzic Mehmed. *Data Mining: Concepts, Models and Algorithms, Second Edition*. 2011.
- Kenneth Stanley O. Clune Jeff Lehman Joel, Risto Miikkulainen. *Designing neural networks through neuroevolution*. *Nature machine Intelligence*, 2019.
<http://www.evolvingai.org/files/s42256-018-0006-z.pdf>
- Kenneth Stanley O. And Risto Miikkulainen. *Evolving Neural Networks through Augmenting Topologies*. The MIT Pressing Journals, 2002. <http://nn.cs.utexas.edu/downloads/papers/stanley.ec02.pdf>