**Tehnička škola Čakovec**

**ELABORAT ZAVRŠNOG RADA**

**AI SIMULACIJA**

**Mentor: Učenik:**

**Krešimir Kočiš, prof. Toni Polanec, 4.RT**

**Čakovec, svibanj 2020.**

Prazna stranica

Tekst zadatka

Zahvala ?

Sadržaj

1. Uvod
2. Programiranje funkcionalnosti auta
   1. Kretanje
   2. Interakcija s okolinom
      1. Vidokrug auta
      2. Stanja
3. Programiranje inteligencije
   1. Neuronska mreža
   2. Genetski algoritam
      1. Fitness
4. Kreator staza
5. Zaključak
6. Bibliografija
7. Popis slika
8. Dodaci

Prazna stranica

Uvod

U ovom radu ću pisati o primitivnoj vrsti AI (umjetne inteligencije). Zadatak rada je pokazati kako i na koji način je moguće implementirati algoritam strojnog učenja i genetske evolucije uz koji će subjekti, u ovom slučaju autići, kroz više generacija naučiti kako voziti stazom. Kad populacija dosegne nivo inteligencije kad u svaki generaciji dovoljno autića prođe stazom moguće ih je prebaciti na njima potpuno novu stazu i ukoliko je njihovo učenje bilo efektivno trebali bi moći iz prve proći novom stazom.

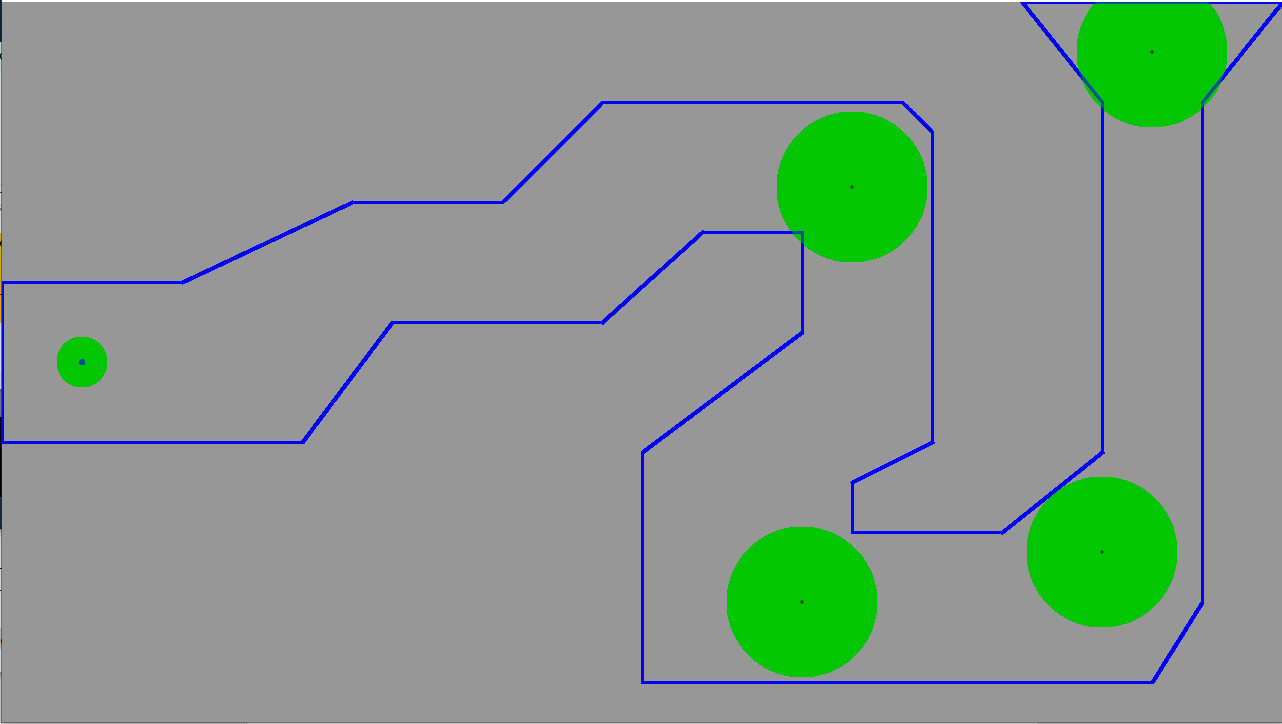
Rad će biti izrađen u dva najbitnija dijela. Programiranje autića da se mogu kretati i interakcija s prostorom. U to ulazi implementiranje osnovnih funkcionalnosti autića (ubrzavanje, kočenje, skretanje), pregled okoline iz njihove perspektive (5 senzora koji gledaju unaprijed) i na kraju njihova interakcija s stazom (dolazak do checkpoint-a, sudaranje s zidom, prolazak cilja).   
Drugi dio je programiranje inteligencije. Svaki auto će se sam voziti. Mozak će biti implementiran sa neuronskom mrežom. Postupak svakog od autića biti će određen s onim što on vidi (senzori, tj. udaljenosti od zidova staze) i stanjem njegovih neurona u mozgu. Učenje ću implementirati genetskim algoritmom. Nakon svake generacije kroz razne algoritme određivat će se roditelji koji će svoje gene pretočiti u njihove potomke koji dalje nastavljaju spomenuto generacijsko učenje.

Umjetna inteligencija je svuda oko nas i ne možemo pobjeći od nje. U svakom mobitelu, u kamerama, u poznatim virtualnim asistentima poput Alexe ili Siri. Smatram da je umjetna inteligencija i njezino implementiranje budućnost tehnologije koja će uvelike pomoći razvoju čovječanstva.

Prva faza – kreiranje mape

Zbog samog funkcioniranja programa Processing način kreiranje staze za autiće morao je biti manualan. Kod pokretanja Processing programa kreira se kanvas za crtanje, te svi entiteti koje vidimo o programu moraju biti „nacrtani“. Tako i granice staze.

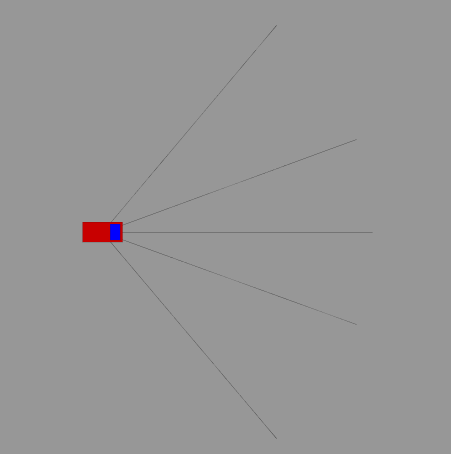
Jedna od bitnijih dijelova mape su takozvani „checkpointevi“ koji će nam služiti u genetskom algoritmu kroz koji ćemo proći kasnije. Oni su postavljeni na relativno bitne dijelove mape, nakon zavoja ili težih prepreka. Kad auto uđe u radijus checkpoint-a tada program prepozna da je auto stigao do određenog dijela staze.



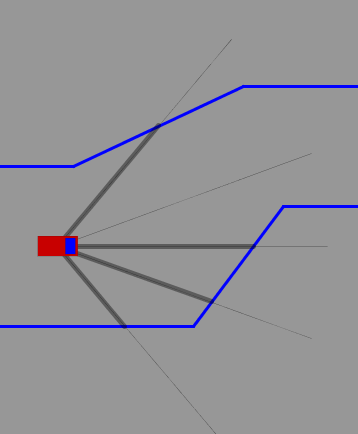
Druga faza – kreiranje autića

Tijelo auta je najobičniji pravokutnik te radi prepoznavanja koji je prednji i zadnji dio auta dodao sam prednje vjetrobransko staklo.

Najbitniji dio auta nije tijelo auta već senzori. Senzori nam prikazuju što i do koje udaljenost autić *vidi*. Senzori su postavljeni da *vide* 300 piksela ispred autića u 5 smjerova. Od kojih je jedan postavljen da gleda ravno naprijed, dva lijevo i desno pod 20 stupnjeva i dva lijevo i desno pod 50 stupnjeva od središnjeg senzora.



Da bi mi uočili i razumjeli što auto vidi, postavio sam uvjet ako senzor nešto prepozna ispred sebe, da se sama crta koja označuje senzor podeblja. S tim svojstvom nama je puno lakše vizualizirati autićevo vidno polje.



Problemi i rješenja

Problem 1: Auto ne detektira zid na vrijeme

Razlog: Funkcija isDead() gleda samo jedan senzor te po njegovoj vrijednosti postavlja dead na true ili false. Ako je taj određeni senzor okrenut lijevo, auto ne detektira zid s desne strane na vrijeme.

Rješenje: For petljom utvrđujem najmanju udaljenost zida svih senzora te istu šaljem isDead() funkciji.

Postupak generiranja nove populacije

1. Provjeravanje da li su svi autići završili u zidu
2. Izračunavanje ukupnog fitnessa svih autića
3. Postavljanje fitnessa svakog od autića po udjelu u ukupnom fitnessu

mojFitness = mojFitness / ukupniFitness;

1. Biranje 2/3 od ukupne populacije autiće koji će nastaviti putovanje pomoću Tournament selection.

Algoritam:

* + 1. Od svih autića biramo 2 autića nasumično.
    2. Od tih 2 uzmemo onaj s boljim fitnessom. Te ga zapisujemo u *winner* polje
    3. Ponavljamo korak 1 i 2 tako dugo dok ne dobijemo 2/3 od ukupnog broja autića u *winner* polje

1. Autići iz *winner* polja su odabrani za produciranje novih autića. (parent – child odnos). Child autić dobijemo tako da uzmemo 2 autića nasumično iz *winner* polja te radimo „crossover“. Pola neural network child dobije od prvog roditelja a drugu polovicu od drugog roditelja.
2. Mutation. Kao i u prirodi kod evolucije se događaju mutacije pa tako i u evolucijskome programu. Stopa mutacije je određena konstantom [0.0001, 0.1]. Stopa mutacije određuje koja je vjerojatnost da se određeni dio neural networka, tj. pojedini weight, promjeni.
3. Takvi novi autići prelaze u novu populaciju (1/3) zajedno sa svojim roditeljima (2/3).

Problem: Generacija 2 odmah mrtva, i neprestano generiranje novih generacija, također mrtvih. Debugiranje trajalo 4 sata. Rješenje: timer koji je mjerio vrijeme od početka programa ako auto ne prijeđe određeni checkpoint na vrijeme, umire se nakon završetka jedne populacije nije resetirao pa svaka sljedeća populacija i svi autići nisi bili u mogućnosti doći do checkpoint-a na vrijeme pa zato i svi bili mrtvi.