

Univerzitet u Novom Sadu  
Fakultet tehničkih nauka  
Departman za računarstvo i automatiku

# Optimizacija agenta u igri Flappy Bird primenom genetskog algoritma

Projekat urađen iz predmeta:Operaciona istraživanja

Irina Radanović (IN 5/2022)  
Nikolina Brkljač(IN 2/2022)  
Marko Stojković(IN 54/2022)  
Studijski program:Informacioni inženjering  
Februar, 2026.

Sadržaj:

1. Uvod
2. Parametri
3. Neuronska mreža
4. Genetski algoritam
5. Rezultati
6. Zaključak

## **1. Uvod**

Cilj ovog projekta je razvoj autonomnog intelligentnog agenta sposobnog da samostalno igra popularnu igru *Flappy Bird*. Za razliku od tradicionalnog programiranja gde se pravila kretanja definišu fiksnim uslovima, ovaj projekat koristi **neuroevoluciju** – kombinaciju veštačkih neuronskih mreža i genetskih algoritama.

Agent započinje proces bez ikakvog predznanja o mehanici igre. Kroz proces evolucije, populacija agenata prolazi kroz selekciju, ukrštanje i mutaciju, postepeno "učeći" optimalne strategije za izbegavanje prepreka i maksimizaciju rezultata.

Za implementaciju su korišćene PyGame(za iscrtavanje i simulaciju igre) i PyTorch(za implementaciju neuronske mreže) biblioteke.

## **2. Parametri**

Uspeh učenja direktno zavisi od pažljivo podešenih parametara okruženja i samog algoritma. U nastavku su prikazani ključni parametri korišćeni u simulaciji:

Fiksne konstante:

SCREEN\_WIDTH = 400

SCREEN\_HEIGHT = 600

Ova dva parametra se odnose na dimenzije prozora igre, 400px sirine i 600px duzine

PIPE\_SPEED = 4

Brzina kretanja prepreka(cevi) s desna na levo. Broj 4 nam govori da se cev pomeri za 4 piksela ka levo u svakom frejmu.

GRAVITY = 0.5

Vertikalno ubrzanje koje simulira gravitaciju u svakom frejmu.

BIRD\_JUMP = -8

Snaga impulsa koji ptica dobija pri skoku (negativna vrednost zbog Pygame koordinatnog sistema).

**POPULATION\_SIZE = 100**

Broj agenata koji se istovremeno testira u jednoj generaciji.

**MUTATION\_RATE = 0.15**

Verovatnoća i intenzitet nasumične promene težina u neuronskoj mreži.

**MUTATION\_STRENGTH = 0.02**

Definiše intenzitet promene težine.

Pored osnovnih vrednosti, u procesu mutacije koristimo dvostepeni intenzitet:

Fino podešavanje (Fine-tuning): U 90% slučajeva mutacije, koristi se **MUTATION\_STRENGTH = 0.02**. Ovo omogućava agentima da zadrže većinu naučenog ponašanja uz minimalne korekcije težina.

Radikalno istraživanje (Exploration): U 10% slučajeva, snaga mutacije se povećava na 0.1 (pet puta jače). Ovo služi da se populacija "izvuče" iz lokalnih optimuma ukoliko dođe do zastoja u učenju, uvodeći potpuno nove strategije kretanja.

Broj frejmova u sekundi: 60

Promenljive:

Gap size - odnosi se na otvor izmedju gornje i donje cevi kroz koji agent treba da prođe.

Nasumična vrednost između 120 i 180px.

Next pipe destination - odnosi se na udaljenost između dva para cevi. Nasumična vrednost između 200 i 350px.

### **3. Neuronska mreža**

Svaka ptica(agent) u sebi nosi "mozak", odnosno poseduje sopstvenu neuronsku mrežu koja donosi odluku da li treba da skoči(zamahne, *flap*) ili ne.

Mreža se sastoji od tri sloja:

1. Ulazni sloj sa 6 neurona koja prima sledeće podatke:

- Apsolutna visina ptice (y): Omogućava agentu svest o granicama ekrana (plafon i pod).
- Trenutna vertikalna brzina (velocity): Ključna za predviđanje putanje i inercije pada.
- Y pozicija gornje ivice cevi: Definiše gornju granicu prepreke.
- Y pozicija donje ivice cevi: Definiše donju granicu prepreke (razlika između ove dve vrednosti implicitno definiše i promenljivu veličinu otvora).
- Horizontalna udaljenost od cevi: Određuje tajming skoka.

- Relativna pozicija ptice u odnosu na centar otvora: Direktna informacija o odstupanju od idealne putanje, što služi kao primarni signal za korekciju leta.
2. Skriveni sloj sa 16 neurona
    - Koristi ReLU aktivacionu funkciju za obradu nelinearnih odnosa između podataka.
  3. Izlazni sloj koristi Tanh aktivacionu funkciju koja vraća vrednost u opsegu od -1 do 1. Ako je izlazna vrednost veća od 0.5, agent donosi odluku da skoči. Tanh se pokazao stabilnijim za gradijentni prostor u ovom problemu.

## 4. Genetski algoritam

Genetski algoritam upravlja procesom učenja kroz cikluse generacija. On nam služi za optimizaciju agenta tokom igre kako bi se kroz više generacija razvile neuronske mreže koje donose bolje odluke i time utiču na bolje preživljavanje ptica (*survival rate*).

### Fitness funkcija (Ocena uspešnosti)

Fitness funkcija je najvažniji deo algoritma jer definiše šta je "poželjno" ponašanje.

Izračunava se na sledeći način:

- Dugovečnost: +0.1 poena za svaki preživljeni frejm (nagrađivanje stabilnosti).
- Uspeh: +1000 poena čim nos ptice (pozicija X=50) pređe početak cevi (nagrađivanje preciznog ulaska u rupu).
- Stroža kazna: -10 poena za udarac u ivice ekrana (brza eliminacija loših gena).
- Kolizija: -50 poena za udarac u cev.
- Kazna za skok: -0.5 poena za svaki zamah krilima (podsticanje energetske efikasnosti i mirnijeg leta).

### Evolucijski proces

Kada svi agenti u jednoj generaciji završe igru (uginu), nastupa proces evolucije:

1. Selekcija (*Selection*): Agenti se rangiraju prema fitness rezultatu. Top 15 agenata postaju "roditelji". Njihov genetski materijal (težine unutar neuronske mreže) služi kao osnova za sledeću generaciju
2. Elitizam (*Elitism*): Kako bi se osiguralo da najbolja rešenja nikada ne budu izgubljena zbog slučajnosti u ukrštanju ili mutaciji, primenjuje se strategija elitizma. Top 3 agenta iz trenutne generacije se direktno, bez ikakvih promena, kopiraju u novu populaciju. Ovo garantuje da maksimalni fitnes generacije nikada ne opada.
3. Ukrštanje (*Crossover*): Ostatak populacije (do 100 jedinki) popunjava se "decom". Stvaraju se novi agenti mešanjem težina izabranih roditelja (nasumična šansa 50% za svaki gen).

4. Mutacija (*Mutation*): Ključna za održavanje genetskog diverziteta. Na težine potomaka primenjuje se Gausov šum (nasumične vrednosti). Ovo omogućava agentima da "istražuju" pokrete koje njihovi roditelji nisu probali.

Kriterijum zaustavljanja:

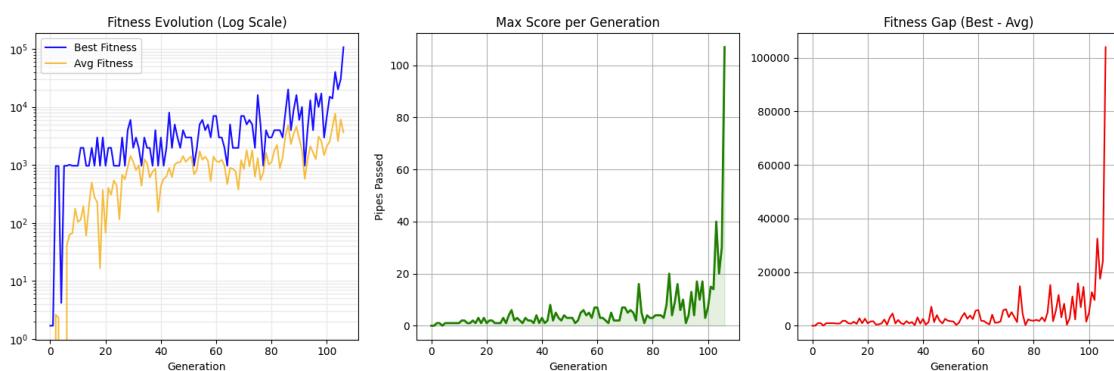
Simulacija se automatski prekida kada najbolji agent u generaciji uspešno savlada 100 uzastopnih prepreka (cevi) bez sudara, što se smatra dokazom potpune optimizacije strategije.

## 5. Rezultati

U početnim fazama razvoja, primetili smo da je maksimalni rezultat (Max Score) dugo ostajao na nuli, uprkos rastu fitnesa. Razlog za to je bila previše striktna logika nagrađivanja: agenti su dobijali poene tek nakon što bi *potpuno* prošli cev. Kako nijedna ptica u ranoj fazi nije imala razvijenu motoriku da prezivi ceo prolazak kroz niz cevi, evolucija nije imala jasan pravac. Problem smo rešili uvođenjem "Checkpoint" sistema:

- **Rana nagrada:** Pomerili smo trenutak bodovanja na sam ulazak u cev. Ovo je pružilo neophodan podsticaj agentima koji su "shvatili" gde treba da nišane, čak i ako bi poginuli delić sekunde kasnije.
- **Normalizacija ulaza:** Inicijalno korišćenje sirovih piksela zamenili smo normalizovanim vrednostima (0-1). Ovo je omogućilo neuronskoj mreži da lakše generalizuje pravila.

Analiza napretka kroz generacije:



Prikazani grafikoni ilustruju evolutivni put populacije agenata od nasumičnog ponašanja do postizanja statusa 'eksperta'. Na osnovu prikupljenih podataka iz 107 generacija, možemo identifikovati tri ključne faze u procesu učenja:

- **Faza inicijalne stagnacije (Gen 1 – 10):** U prvih deset generacija, maksimalni score je stabilno na nuli ili jedinici. Prosečan fitnes je često negativan zbog kazni za koliziju, što ukazuje na to da agenti još uvek nemaju razvijenu kontrolu visine.

Međutim, blagi rast 'Best Fitness' vrednosti sugerije da algoritam uspešno identificuje jedinke koje preživljavaju makar sekundu duže.

- **Faza stabilizacije i motoričkog razvoja (Gen 11 – 80):** Ovde primećujemo prve značajne skokove. Agent u 44. generaciji dostiže score 8, što predstavlja prvu veliku prekretnicu. U ovom periodu, 'Best Fitness' osciluje, ali se 'Avg Fitness' postepeno podiže, što je direktna posledica elitizma i selekcije — loši geni se eliminišu, a populacija postaje ujednačenija u kvalitetu.
- **Faza eksponencijalne optimizacije (Gen 81 – 107):** Finalna faza pokazuje moć akumuliranog znanja. Nakon 100. generacije, rezultati naglo rastu: sa skora 30 u 106. generaciji, najbolji agent u 107. generaciji dostiže rekordnih 107 pređenih prepreka. Ovakav nagli skok (tzv. *elbow point*) tipičan je za neuroevoluciju, gde neuronska mreža konačno 'zaključava' optimalne težine za sve moguće scenarije u igri.

## 6. Zaključak

Ovaj projekat je uspešno demonstrirao moć neuroevolucije u rešavanju problema sa dinamičkim preprekama. Ključni zaključci izvedeni tokom rada su:

1. **Dizajn fitnes funkcije je presudan:** Suptilne promene, poput uvođenja male kazne za nepotrebno skakanje, transformisale su let agenta iz haotičnog u miran i elegantan, koji troši manje resursa.
2. **Balans između elitizma i mutacije:** Prevelik elitizam dovodi do stagnacije (svi postaju isti), dok prevelika mutacija uništava naučeno. Postavljanjem Top 3 elite i mutacije od 15%, pronađen je optimum koji omogućava napredak bez gubitka najboljih rešenja.
3. **Generalizacija znanja:** Agent koji je prošao 356 cevi nije samo naučio stazu, već je razvio razumevanje fizike igre. Ovakav model bi se verovatno uspešno snašao i u verzijama igre sa različitom gravitacijom ili brzinom cevi.

Primenjeni model neuronske mreže sa samo jednim skrivenim slojem pokazao je da za probleme ove složenosti nije potrebna duboka arhitektura (Deep Learning), već pametno koncipiran proces učenja kroz genetski algoritam.