



Eötvös Lóránd Tudományegyetem

Informatika Kar

Programozáselméleti és Szoftvertechnológiai Tanszék

MI kliens 2 dimenziós ügyességi játékhöz

Témavezetők:

--Belső

Bordás Henrik
Szoftverfejlesztő
CAE-Engineering kft.

Szerző:

Vecsernyés Márk
nappali tagozat
programtervező informatikus szak

Budapest, 2018

TARTALOMJEGYZÉK

1. Bevezetés	2
2. Felhasználói dokumentáció	3
2.1. A program által megoldott feladat	3
2.2. Felhasznált módszerek áttekintése	4
2.3. A program használata	4
2.3.1. Minimális rendszerkövetelmények	4
2.3.2. Program indítása	5
2.3.3. Konfigurálás	6
2.3.4. A program által generált eredmények értelmezése (4. ábra)	6
3. Fejlesztői dokumentáció	7
3.1. Probléma részletes specifikációja	7
3.2. Felhasznált módszerek részletes leírása	7
3.3. A program logikai és fizikai szerkezetének leírása	7
3.4. Tesztelés	7
4. Összefoglalás	8
5. Irodalomjegyzék	9

1. Bevezetés

Kevés igazán jó szemléltetőeszköz van a mesterséges intelligencia módszerei interaktív bemutatására, ezért dolgozatomban egy nagyon egyszerű szabályokkal rendelkező játékon keresztül szeretném a felhasználóval megismertetni az egyik kedvelt módszert, az evolúciós algoritmus futását. Sokkal könnyebb észrevenni a módszer előnyeit, illetve hátrányait egy animációs példán keresztül, mint elolvasni ugyanezeket valamilyen szöveges formában. Tetszőlegesen paramétereizhető, kis számítási igényű programot írtam, hogy bárki ki tudja próbálni.

A játék, amin a tanítást végzem a *Flappy Bird*^[1] átirata. Az eredeti 2013. Május 24-én jelent meg okostelefonos platformra. *Dong Nguyen* fejlesztette a *dotGears* cégnél, 2014 elején napi 50.000\$ körüli összeget hozott reklámbevételekből. Két dimenziós, retro játékokra hasonlít, az irányítandó karakter egy sárga madár, aki folyamatosan halad előre balról jobbra, közben a magassága a gravitáció hatására csökken. Egyetlen parancsot lehet neki adni, ez az ugrás, így kell minél messzebb manővereznie az akadályok (oszlopok) között. 2014. Február 10-én *Nguyen* személyes okokra hivatkozva törölte a játékot. Ezután az egyik legtöbbször másolt játék lett az *Apple Store*-ban, megközelítőleg napi 60 másolatot töltöttek fel, de PC-re is számos átirat megjelent.

Dolgozatom célja egy létező algoritmus implementálása egy konkrét alkalmazáson keresztül, és futásának szemléltetése.

Ezúton szeretnék köszönetet mondani a külső konzulensemnek, *Bordás Henriknek*, aki tudása és gyors tanácsai nélkül nem tudtam volna így elkészíteni ezt a dolgozatot.

2. Felhasználói dokumentáció

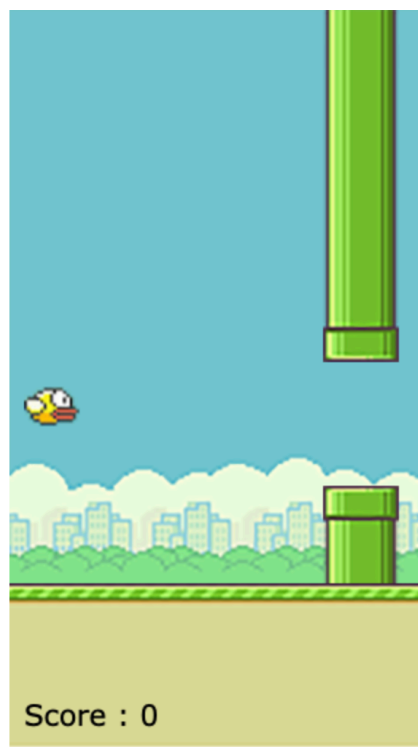
A következőkben ismertetjük *a program által megoldott feladatot*, majd a felhasználó szempontjait szem előtt tartva áttekintjük a *felhasznált módszereket*, végül részletesen bemutatjuk a *program használtát*, paraméterezését és a program által generált eredményeket.

2.1. A program által megoldott feladat

A program a gépi tanulás egy módszerének, az *evolúciós algoritmusnak* a futását, nem pedig magát az algoritmust akarja részletesen megmutatni a felhasználónak.

Bármilyen módszer működése egy adott probléma megoldására akkor szemléltethető a legjobban, ha a modell kellően bonyolult, viszont közérthető. A bonyolultságra azért van szükség, mert az esetleges egyszerűsítések következtében ha túl sok tulajdonságot elhagyunk, akkor a módszer hátrányait nem minden esetben láthatjuk. Közérthetőség nélkül pedig a probléma sokak számára értelmezhetetlenné válik. A bevezetőben ismertetett modellválasztásom és algoritmus kapcsolatát szeretném röviden bemutatni.

A *kiindulási modell* a játék egy változata (1. ábra), melynek forráskódja ingyenesen letölthető az internetről^[2]. Itt a felhasználó manuálisan, egy gomb lenyomásával tudja irányítani a karaktert. Ezt az input interfészt le kell cserélnünk, hogy a számítógép irányítsa madarat, illetve a populációnak megfelelő számú madarat kell elhelyezni a játékban. A minden madarat eltérő mesterséges neurális hálóval kell kódolni, futása (játék indításától a madár haláláig) után pedig fitnessz értékét feljegyezni. Az algoritmus a ezen értékek ismeretében tudja elvégezni a kiértékelést.



1. ábra

A kiértékelési módszer^[3] (2. ábra) egyetlen ciklusból áll. Az evolúció alapeljárásának nevezik. Az implementációmban a populációt a madarak futásának feleltetem meg, az algoritmus további részeit a következő, "Felhasznált módszerek áttekintése" fejezetben részletezem.

Procedure EA

populáció := kezdeti populáció

while terminálási feltétel nem igaz **loop**

szülők := szelekció(populáció)

utódok := rekombináció(szülők)

utódok := mutáció(utódok)

populáció := visszahelyezés(populáció, utódok)

endloop

2. ábra

A program által megoldott feladat tehát a fellebb ismertett modellen történő optimális populáció keresés evolúciós algoritmussal.

2.2. Felhasznált módszerek áttekintése

TO BE DONE, ide jönnék magának az evolucios algoritmusnak a reszletes ismertetese a teljesseg igenye nelkul

2.3. A program használata

A következőkben szeretném felsorolni a *minimális rendszerkövetelményeket* és a futtatáshoz szükséges programokat, csomagokat. Bemutatom a program *indítását* egy adott konfigurációval, amit az *eredmények kiértékelése* követ.

2.3.1. Minimális rendszerkövetelmények

A program jelenleg csak asztali számítógépen futtatható. Operációs rendszertől függetlenül mind *Windows-on*, *Linux-on* és *MacOS-en* lehet futtatni, amennyiben a következő komponensek telepítve vannak:

- *Python3*
- *Flask, psycpg2, configparser csomag Python3-hoz*

- *PostgreSQL* szolgáltatás "postgres" role-lal
- *Böngésző*

2.3.2. Program indítása

Először győződjünk meg róla, hogy a *PostgreSQL* szolgáltatás fut, majd az *app.py* programot kell elindítanunk, amit konzolból a következőképpen tehetünk meg, ha az adott mappában vagyunk (a \$ jel utáni parancsot kell begépelnünk):

```
markvecsernyes@mvecsernyes fbai (master) $ python3 app.py
```

Ekkor az alkalmazás a standard outputra kiír pár log üzenetet. Ha sikerült csatlakoznia az adatbázishoz, és az üzenetek között nincs hiba, akkor elindíthatjuk a böngészőt, majd a <http://127.0.0.1:5000/> címet beírva az URL sávba, megkapjuk a program grafikus felhasználói felületét, amit GUI-nak nevezünk (3. ábra).

A GUI két jól elkülönített részre van osztva. Felül egy bemeneti mező található, ezzel tudjuk paraméterezni a programot, lejjebb pedig egy *vászon*, (mérete 576x512 pixel), ide fogja kirajzolni a program a grafikus elemeket.



3. ábra

2.3.3.Konfigurálás

Első lépésben a felhasználó program paramétereit adhatja meg, ha felül kívánja írni az alapértelmezett értékeket *legördülő listával*, ezek sorban a következők:

- Gravity: *a gravitáció mértékét adja meg 1 és 10 között egyesével. Minél nagyobb ez az érték, annál nagyobb mértékben csökken a madarak magassága egységnyi idő alatt.*
- Population: *a populáció méretét lehet megadni (a madarak számát) 5 és 50 között ötösével.*
- Gap: *egy oszloppár (függőlegesen) közti távolságot lehet megadni 70 és 120 között ötösével.*
- Distance: *két egymást követő oszloppár (vízszintesen) közti távolság méretét lehet megadni 120 és 260 között tizesével.*

A paraméterek megadása után az *Apply* gombbal tudjuk véglegesíteni a választásunk. Ekkor a *Start* gomb aktívvá, míg az *Apply* gomb és a legördülő lista elemei inaktívvá válnak.

A *Start* gomb megnyomásával az algoritmus elindul, további felhasználói input nem adható meg. Bármely állapotban a *Reload* gombbal újra tudjuk tölteni a lapot, akkor minden eddigi eredmény elveszik, és újra lehet konfigurálni a játékot.

2.3.4.A program által generált eredmények értelmezése (4. ábra)

A *Start* gomb megnyomása után a vászonon kirajzolódik a háttér, bal oldalon a madarak és az oszlopok, míg jobb oldalon az aktuális generáció száma, alatta minden madár fitnessz értéke az adott pillanatban. Ha a madár sora fekete szöveggel van írva, akkor a madár életben van, ha piros, akkor meghalt. A program futásának minden további része automatizálva van, a felhasználó nem tud utólag paramétereket módosítani.

3. Fejlesztői dokumentáció

3.1. Probléma részletes specifikációja

3.2. Felhasznált módszerek részletes leírása

3.3. A program logikai és fizikai szerkezetének leírása

3.4. Tesztelés

4. Összefoglalás

5. Irodalomjegyzék

- [1] https://en.wikipedia.org/wiki/Flappy_Bird: 2018.10.03.
- [2] <https://github.com/CodeExplainedRepo/FlappyBird-JavaScript>: 2018. 09. 15.
- [3] Gregorics Tibor: Mesterséges intelligencia fóliasora 2017/18, *tavaszi félév*