

Trénování AI pro hraní her, Využití GA pro řešení úloh z prostředí OpenAI Gym

Šimon Sedláček



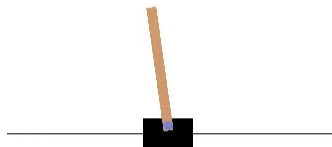
10. května 2022

Trénování AI pro hraní her (M. Piños)

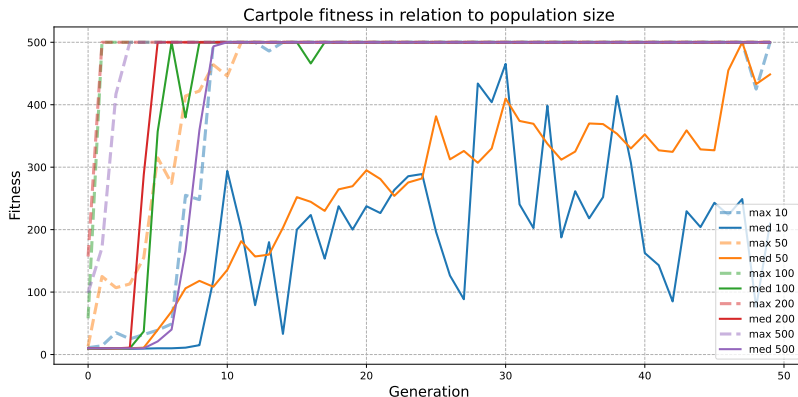
- Seznamte se s frameworkem OpenAI GYM (1). Zvolte si určitý problém (hru) k optimalizaci (hraní) a implementujte evoluční algoritmus, který tento problém (hru) optimalizuje (hraje).
- Proveďte experimenty a vyhodnoťte výsledky.

- Implementace GA pro řešení problémů z prostředí [OpenAI Gym](#)
- Optimalizace vah neuronové sítě
 - Síť mapuje vektor vstupních pozorování na výstupní vektor akcí.
 - Genotypem je tedy zploštělé pole vah a biasů dané sítě
- Jako fitness zde slouží systém odměn každého z prostředí v Gym

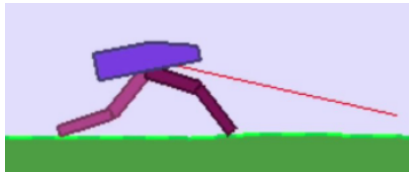
- Jednoduchý control problem
- 4 vstupy, 2 diskrétní výstupní akce
- Tvar sítě: 4-4-2-2
- Budeme testovat vliv velikosti populace na rychlost hledání řešení
 - Velikosti populace 10, 50, 100, 200, 500
 - Uniformní křížení, rychlost mutace 0.07, ruletový výběr



Je zjevné, že velikost populace má značný vliv na rychlost nalézání optimálního řešení. Pro takto jednoduchý problém dostaneme maximální fitness už hned v druhé generaci pro vyšší velikost populace. Větší populace mají také značný vliv na udržování vysokého mediánu fitness.

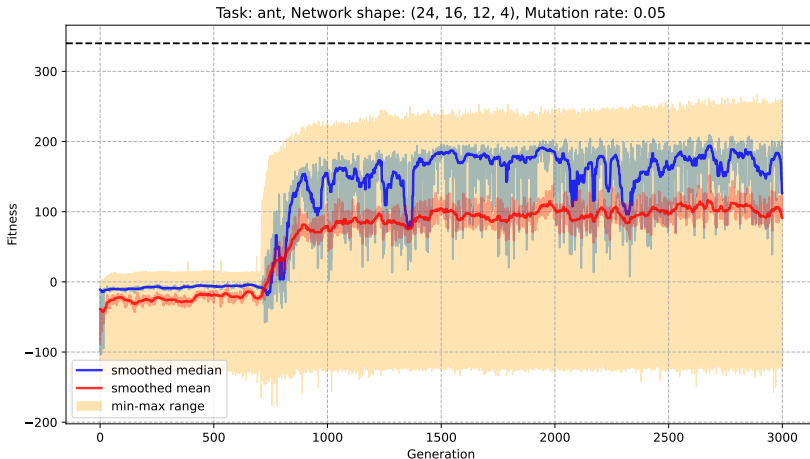


- Klasické “control problems” jsou poměrně jednoduché
 - Diskrétní prostor akcí
 - Malý vektor vstupního pozorování, malý výstupní vektor
- Problémy z prostředí [Box2d](#) představují větší výzvu
 - Mnohem větší stavový prostor
 - Spojité akce
- [Bipedal Walker](#)

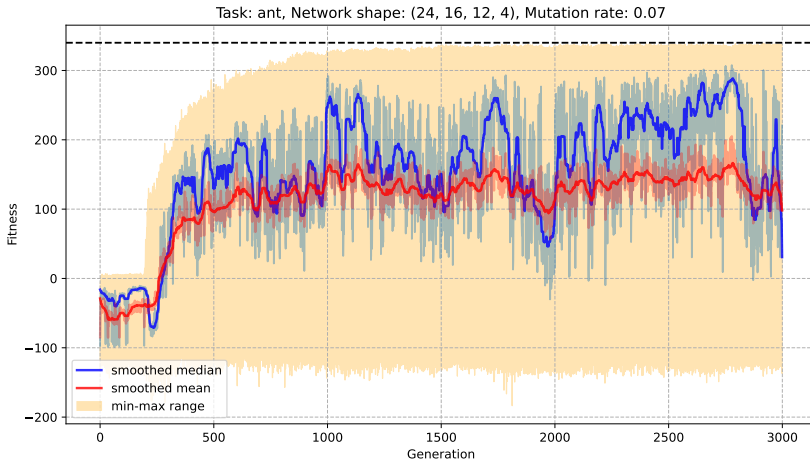


- Velikost populace 200
- Tvar sítě: 24-16-12-4
- Uniformní křížení, ruletový výběr rodičů
- Trénování běží po 3000 generací
- Jeden běh trvá průměrně 8 hodin
 - Výpočty prováděny na metacetrum.cz
 - Paralelizace na 8 procesorech
- Zajímá nás úspěšnost trénování v závislosti na hodnotě parametru *mutace* a velikosti skrytých vrstev sítě.

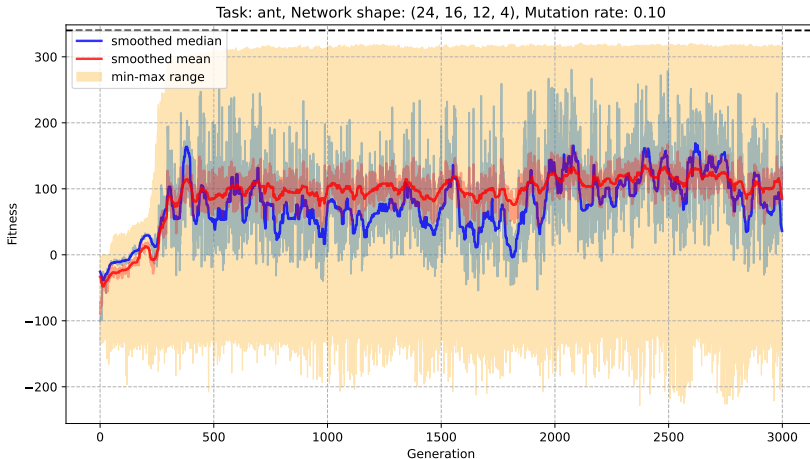
Nedostatečně agresivní mutace – nedosáhneme maximální fitness ani po 3000 generacích.



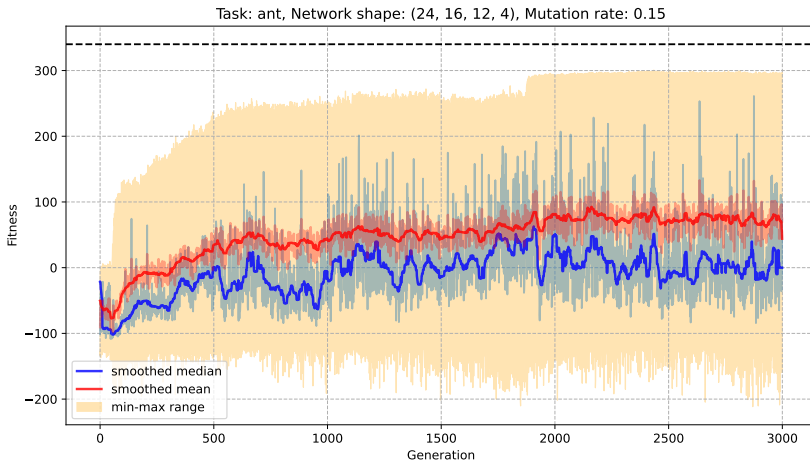
Zdaleka nejlepší výsledek – hodnota 0.07 představuje dobrou rovnováhu mezi agresivitou a stabilitou.



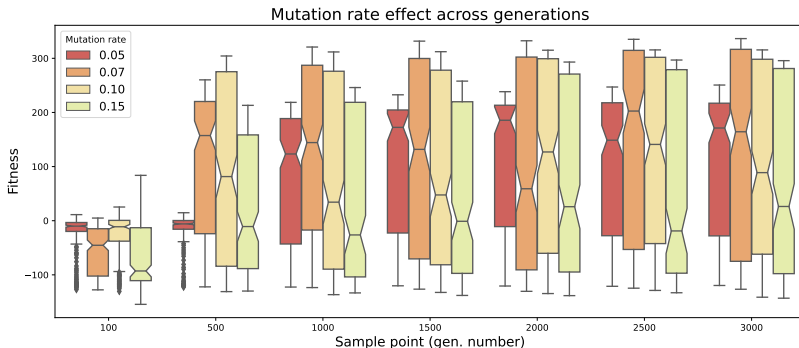
Hodnota 0.1 dosahuje také velice dobrých výsledků, avšak ne tak dobrých jako 0.07.



Hodnota 0.15 je pak již příliš agresivní pro efektivní použití – toto je nejlépe vidět na velice nízko položené mediánové křivce evoluce.

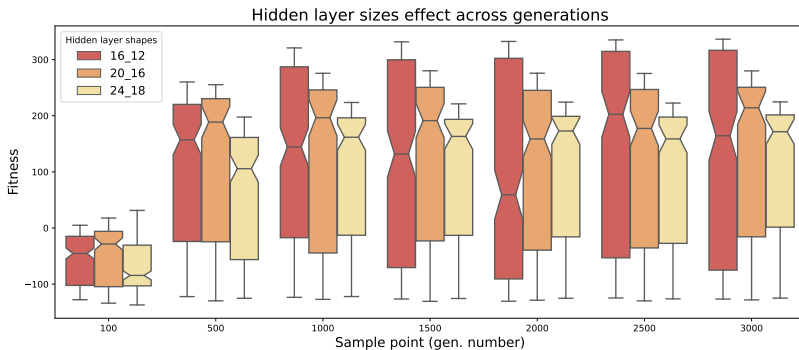


Je zjevné, že nejlepších výsledků dosahuje hodnota rychlosti mutace 0.7. Ostatní hodnoty jsou buď nedostatečně agresivní, nebo příliš nestabilní.



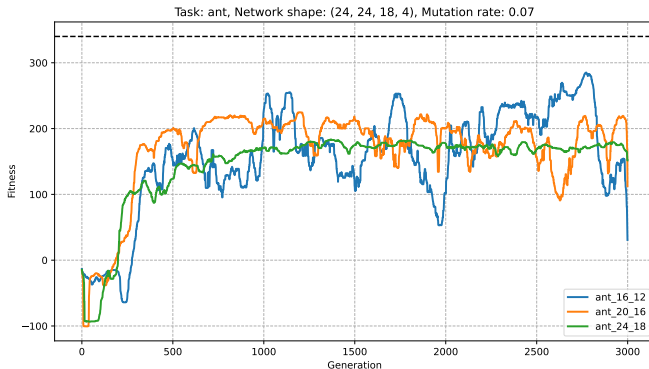
Obrázek: Porovnání fitness pro různé hodnoty rychlosti mutace.

Takto naučený walker však stále do určité míry tahá zadní koleno po zemi. Mohla by složitější síť vést k zajímavějšímu chování? Bohužel, je zjevné, že pro stávající parametry evoluce ani jedna z větších architektur nedosahuje lepších výsledků.



Obrázek: Porovnání fitness populací pro různé tvary dvou vnitřních skrytých vrstev.

Zajímavé však je, že větší architektury stabilněji drží medián hodnoty fitness. Toto je pravděpodobně způsobeno tím, že mutace nějaké z vah menší sítě má pravděpodobně větší vliv na její výstup, než by taková mutace měla na síť s větším množstvím trénovatelných parametrů.



- Byl implementován genetický algoritmus hledající řešení problémů z prostředí OpenAI Gym.
- Toto řešení hledá ve formě hodnot vah a biasů určité neuronové sítě mapující vstupní pozorování agenta na výstupní akce.
- Algoritmus je kromě všech základních "control problems" schopen řešit i náročnější úlohy, jako například úloha Bipedal Walker.
- Vytvořená implementace umožňuje jednoduchou replikaci experimentů, experimentování s různými parametry evoluce a vizualizaci výsledných řešení daných úloh.



Agoston E. Eiben and James E. Smith.
Introduction to evolutionary computing.
In Natural Computing Series, 2003.



Ahmed Gad.
Artificial neural networks optimization using genetic algorithm with python.
(online), rev. 7. březen 2019.
(vid. 2022-05-02).



Satvik Tiwari.
Genetic algorithm: Part 4 -cartpole-v0.
(online), rev. 28. duben 2019.
(vid. 2022-05-02).