

Dokumentácia

Umelá Inteligencia

Zenová záhrada (Zadanie 2A)

Jakub Martinák

Faculty of Informatics and Information Technology, Slovak Technical
University

Obsah

1	Úvod	2
2	Prehľad genetického algoritmu	2
2.1	Inicializácia populácie	2
2.2	Hodnotenie vhodnosti	2
2.3	Mechanizmy výberu	2
2.4	Kríženie	2
2.5	Mutácia	2
2.6	Vytvorenie novej generácie	2
2.7	Iterácia cez generácie	2
2.8	Ukončenie	3
3	Funkcia ‘move‘	3
3.1	Vstupy	3
3.2	Výstupy	3
3.3	Proces	3
3.4	Vedľajšie Efekty	3
3.5	Ošetrovanie Chýb	3
3.6	Príklady Použitia	4
3.6.1	Poznámky	4
4	Graf	5
4.1	Data pre graf	5
4.2	Fitness hodnoty	5

Abstrakt

Úlohou je vývoj evolučného algoritmu na simuláciu presného procesu hrabania Zenovej záhrady, priestoru vyplneného hrubým pieskom alebo malými kamienkami a presiatého nepohyblivými väčšími objektmi, ako sú skaly, sochy a prirodzený rast. Cieľom je vytvoriť rad priamych, horizontálnych alebo vertikálnych brazd v piesku mníchom, ktorý začína na okraji záhrady a ťahá priamu čiaru až na protiľahlý okraj alebo prekážku. Mních môže voľne chodiť mimo záhrady, ale pri stretnutí s prekážkou vnútri, ako je kameň alebo už upravený piesok, sa musí otočiť. Smer otočenia – doľava, doprava alebo celý obrat – je určený dostupným voľným priestorom. Hra sa končí, keď nie sú možné ďalšie otáčky. Úspešný priechod vyústi do úplne upravenej záhrady alebo maximálneho počtu pokrytých štvorcov podľa daných pravidiel

1 Úvod

Tento dokument popisuje genetický algoritmus (GA) použitý na optimalizáciu cesty mnícha pri hrabaní Zenovej záhrady. Záhrada je reprezentovaná ako matica, pričom niektoré bunky sú obsadené prekážkami (kameňmi) a cieľom je určiť cestu, ktorá mníchovi umožní pohrabať čo najväčšiu plochu záhrady.

2 Prehľad genetického algoritmu

2.1 Inicializácia populácie

GA začína vytvorením počiatkovej populácie objektov *Individual*, pričom každý predstavuje potenciálne riešenie s unikátnym genomom, ktorý určuje cestu hrabania cez záhradu.

2.2 Hodnotenie vhodnosti

Vhodnosť každého *Individuálneho* jedinca je hodnotená na základe plochy záhrady, ktorú úspešne pohrabal. Vyššia vhodnosť zodpovedá väčšej pohrabanej ploche.

2.3 Mechanizmy výberu

Používajú sa dva mechanizmy výberu:

- **Turnajový výber:** Skupina jedincov je vybraná náhodne, pričom dvaja s najvyššou vhodnosťou sú vybraní ako rodičia.
- **Ruletový výber:** Jedinci sú vybraní s pravdepodobnosťou proporcionálnou k ich vhodnosti.

2.4 Kríženie

Funkcia *crossover* v triede *Individual* kombinuje genómy dvoch rodičovských jedincov na vytvorenie nového potomka, pričom sa používajú buď metódy jednopodmiestného alebo uniformného kríženia.

2.5 Mutácia

Mutácia zavádza genetickú rozmanitosť náhodnou zmenou častí genomu potomka.

2.6 Vytvorenie novej generácie

Nová generácia je formovaná z najlepšieho jedinca predchádzajúcej generácie a potomkov z kríženia a mutácie.

2.7 Iterácia cez generácie

GA iteruje cez preddefinovaný počet generácií, pričom sleduje najlepšieho jedinca v každej z nich.

2.8 Ukončenie

GA končí, keď *Individual* prejde celú záhradu alebo keď je dosiahnutý maximálny počet generácií.

3 Funkcia ‘move’

Funkcia ‘move’ simuluje proces hrabania Zen záhrady mníchom, ktorý je reprezentovaný objektom *Individual*. Iteruje genom jedinca na vykonanie série pohybov na mriežke záhrady, začínajúc od preddefinovanej hraničnej pozície. Funkcia aktualizuje mriežku záhrady na odraz pohrabaných chodníčkov a vypočíta vhodnosť jedinca na základe počtu pohrabaných štvorcov.

3.1 Vstupy

Funkcia ‘move’ neberie žiadne parametre, pretože operuje na atribútoch objektu *Individual*.

3.2 Výstupy

- Návratová hodnota: Celkový počet úspešne pohrabaných štvorcov mníchom (vhodnosť jedinca).

3.3 Proces

1. Inicializácia: Funkcia začína nastavením počiatočných premenných na sledovanie pohybu mnícha, vrátane počítadiel chodníčkov a vlajok na detekciu prekážok.
2. Východzia pozícia: Určuje sa východisková pozícia na základe atribútu `start_pos`, ktorý je odvodený z počiatočnej konfigurácie jedinca.
3. Simulácia pohybu: Funkcia vstupuje do slučky, ktorá pokračuje, kým mních neprevedie počet pohybov rovnajúci sa maximálnej dĺžke genomu alebo kým nie sú možné žiadne ďalšie pohyby.
 - (a) Kontrola platného pohybu: Na každom kroku sa kontroluje, či je ďalší pohyb platný.
 - (b) Aktualizácia chodníčka: Ak je pohyb platný, mriežka záhrady sa aktualizuje na označenie pohrabaného chodníčka.
 - (c) Výpočet nasledujúcej pozície: Určuje sa ďalšia pozícia mnícha na základe súčasného smeru pohybu.
 - (d) Riešenie prekážok: Pri stretnutí s prekážkou sa rozhodne o novom smere na základe genomu jedinca.
 - (e) Riešenie hraničných prípadov: Ak mních dosiahne okraj záhrady a nemôže sa ďalej pohybovať, hľadá sa nová východisková pozícia.
4. Priradenie vhodnosti: Po vykonaní všetkých pohybov alebo ak nie sú možné žiadne ďalšie pohyby, vhodnosť jedinca sa určí na základe celkového počtu pohrabaných štvorcov.
5. Inicializácia génu: Ak sa začne nová cesta, funkcia inicializuje génovú sekvenciu pre novú cestu na základe vykonaných pohybov.

3.4 Vedľajšie Efekty

- Mriežka `garden` je modifikovaná na odraz cesty pohrabané mníchom.
- Atribút `fitness` objektu *Individual* je aktualizovaný na odraz počtu pohrabaných štvorcov.
- Atribút `genes` môže byť aktualizovaný s novými sekvenciami génov, ak sa počas procesu pohybu začnú nové cesty.

3.5 Ošetrenie Chýb

Funkcia obsahuje kontroly pre hranicové podmienky, aby sa uistila, že mních sa nepohne mimo mriežku záhrady. Tiež zvláda prípady, kde mních narazí na prekážku alebo predtým pohrabaný chodníček.

3.6 Príklady Použitia

```
individual = Individual() # Predpokladá sa, že je to platný objekt Individual

individual.move() # Simuluje proces hrabania mnícha
print(individual.get_fitness()) # Vypíše skóre vhodnosti, t.j. počet pohrabaných štvorcov
```

3.6.1 Poznámky

- Funkcia sa spolieha na modul `gardenGenerator` na poskytnutie počiatočnej mriežky záhrady s prekážkami.

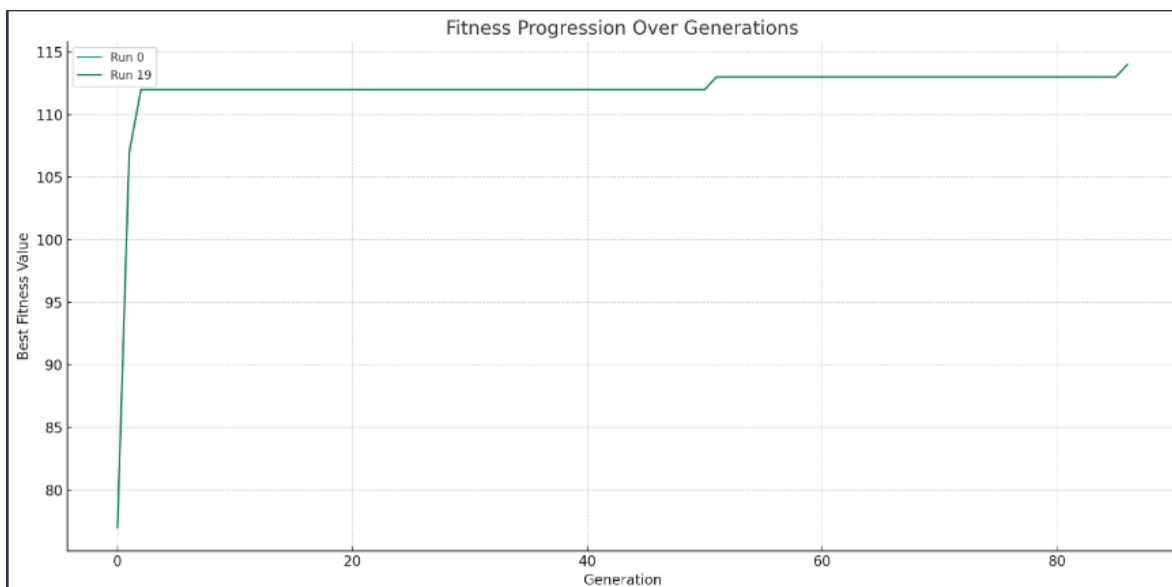
4 Graf

4.1 Data pre graf

Run	Generation	Final	Best
0	1	114	90
0	2	114	110
0	3	114	110
...
0	9	114	114
1	1	114	102
1	2	114	102
...
1	11	114	114
...
19	1	114	77
19	2	114	107
...
19	80	114	113
19	81	114	113
...
19	90	114	114

Tabuľka 1: Fitness values from multiple runs

4.2 Fitness hodnoty



Obr. 1: Graf priebehu fitness hodnoty podľa generácie v 20 behoch

Záver

V tejto dokumentácii sme sa zaoberali rôznymi aspektmi použitia genetického algoritmu (GA) na optimalizáciu cesty hrabania v simulácii Zen záhrady. GA, založený na princípoch prirodzeného výberu vrátane kríženia a mutácie, evoluje riešenia v priebehu generácií. Trieda *Individual* zahŕňa správanie mnícha a funkcia *move* je kľúčová pri simulácii cesty mnícha a určovaní vhodnosti každého riešenia.

Analýza výkonu

Vizualizácia progresie vhodnosti cez viaceré behy ukazuje konvergenciu GA k optimálnym riešeniam. Každý beh ukazuje zlepšenie vo vhodnosti, čo naznačuje schopnosť algoritmu učiť sa a prispôbiť svoju stratégiu hrabania záhrady.

Potenciál pre zlepšenie

1. **Ladenie parametrov:** Úprava parametrov GA môže viesť k rýchlejšej konvergencii a lepším riešeniam.
2. **Stratégia výberu:** Experimentovanie s inými metódami výberu by mohlo priniesť lepšie výsledky.
3. **Funkcia vhodnosti:** Vylepšenie funkcie vhodnosti by mohlo viesť k nuancovanejším riešeniam.
4. **Reprezentácia genomu:** Rozšírenie binárnej reprezentácie môže zahrnúť zložitejšie správanie.
5. **Adaptívne mechanizmy:** Zavedenie adaptívnych mechanizmov by mohlo zabrániť predčasnej konvergencii.
6. **Paralelné behy a diverzita:** Udržiavanie diverzity môže zabrániť lokálnym optimám.
7. **Hybridné prístupy:** Kombinácia GA s inými technikami optimalizácie by mohla rafinovať riešenia.

Optimalizácia výsledkov

Na dosiahnutie optimálnych výsledkov je nevyhnutné nájsť rovnováhu medzi prieskumom a využitím. Úprava miery mutácie môže ovplyvniť túto rovnováhu. Väčšia populácia ponúka viac diverzity, ale vyžaduje viac výpočtových zdrojov.

Záverečné myšlienky

Hoci súčasný systém je robustným východiskovým bodom, existuje dostatok priestoru pre experimentovanie a zlepšenie. Úpravou parametrov a stratégií GA môžeme potenciálne zlepšiť výkon systému a dosiahnuť konzistentnejšie optimálne výsledky. Získané poznatky z vizualizácie údajov pomáhajú porozumieť správaniu GA a usmerňujú smer týchto vylepšení.