

Záverečná správa
projektu IV109

Robot - zbieranie pokladu

Prednášajúci:
doc. Mgr. Radek Pelánek, Ph.D.

Študenti:
Katarína Kejstová 433820
Viktória Vozárová 433334

Jarný semester 2016

1 Zadanie projektu

Variace na príklad zmienený na prednáške. Robot sa pohybuje v čtvercovej mřížce, na niektorých polích jsou poklady, které má posbírat, případně zdi, díry, a pod. Vytvořte genetický algoritmus, který bude vytvářet navigační kód pro robota (tak aby posbíral co nejvíce pokladu).

2 Genetické algoritmy

Genetické algoritmy sú vo všeobecnosti heuristické postupy, aplikujúce princípy z evolučnej biológie. Často sú využívané na riešenia zložitých problémov, pretože sú schopné vytvoriť netriviálne (náhodné) riešenia. Než začneme s popisom nášho riešenia problému, je potrebné definovať niektoré pojmy spájajúce sa s genetickými algoritmami a ich aplikáciou v praxi.

Chromozóm

V aplikácii genetických algoritmov je chromozóm chápaný ako reťazec, resp. postupnosť symbolov, reprezentujúci jedinca s výslednými vlastnosťami. Každá vlastnosť nejako popisuje riešenie a preto reťazec chápeme ako riešenie.

Gén

Je základnou časťou chromozómu. Jeden gén popisuje jednu vlastnosť. Často je reprezentovaný jedným symbolom chromozómu - reťazca, avšak môže byť tvorený aj spojením symbolov - podreťazec.

Populácia

Populáciou rozumieme skupinu reťazcov, a teda potencionálnych riešení. Nad touto skupinou aplikujeme jednu iteráciu genetického algoritmu, výstupom je nová populácia. Veľkosť populácie môže byť premenlivá, v našej aplikácii však pre jeden beh algoritmu udržiavame populáciu konštantnú.

Fitness funkcia

Táto funkcia priradí každému chromozómu, a teda riešeniu hodnotu, vypovedajúcu o jeho úspešnosti. Fitness funkcia pomáha optimalizovať riešenie na základe snahy získať chromozóm s maximálnym ohodnotením. Táto funkcia by preto mala byť vhodne definovaná.

V našej aplikácii sú použité a porovnané rôzne fitness funkcie a ich vplyv na výsledné riešenie (úspešnosť robota). Každá funkcia zároveň potrebuje inak vyladené parametre, by ktorých dáva najlepšie výsledky. Tento fakt tiež zohľadníme v našej analýze.

Kríženie

Kríženie využívame pri procese vzniku novej generácie potomkov. Vďaka kríženiu vznikajú nové, odlišné jedince. V kontexte genetických algoritmov tým rozumieme vytvorenie nového reťazca kombináciou dvoch reťazcov z aktuálnej populácie.

Pri krížení zvolíme náhodný bod na chromozóme, i , a potom skrížením dvoch reťazcov A, B vznikne potomkom reťazec $C = A[1..i] + B[i + 1..n]$, kde n je koniec reťazca B .

Mutácia

Mutácia je proces náhodnej modifikácie určitého génu chromozómu z množiny prístupných vlastností. Zmena génu sa uskutoční s určitou dopredu danou, a často pomerne malou, pravdepodobnosťou. To zaručí objavovanie nových riešení, ku ktorým by sme sa inak nedostali. Zabráňuje uviaznutiu v lokálnom optime. Mutácia dopĺňa operáciu kríženia, a teda vytvárania novej generácie potomkov. Môže byť vynechaná.

3 Analýza problému

Prostredie, v ktorom sa robot pohybuje, je reprezentované ohraňenou mapou, na ktorej je statický počet pokladov a nejaké rozmiestnené prekážky. Uvažujeme dve rôzne mapy líšiace sa obtiažnosťou (množstvom prekážok). Všetky roboty začínajú na dopredu danej počiatočnej pozícii.

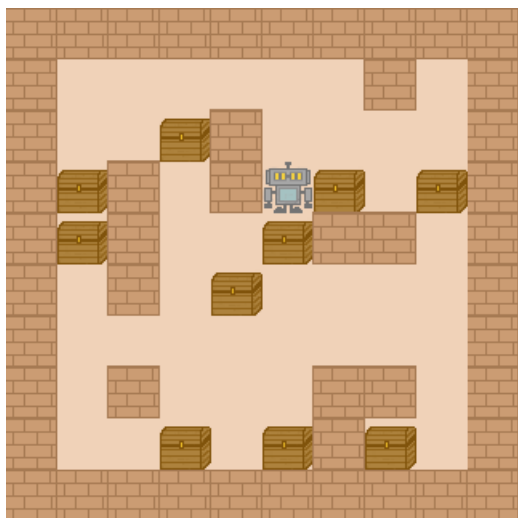
V kontexte nášho problému pod pojmom **gén** chápeme symbol z množiny $\{L, R, U, D\}$, reprezentujúci pohyb po mape (doľava, doprava, hore, dole). **Chromozómom** je reťazec symbolov pohybu, takže určuje celú cestu robota na mape. Dĺžka reťazca závisí od veľkosti mapy. Pri príliš krátkej ceste nemá robot šancu pozbierať dostatočné množstvo pokladu, pri príliš dlhom robí zbytočne veľa krokov navyše a tým stráca na kvalite.

Pri analýze sme uvažovali rôzne **fitness** funkcie líšiace sa v ohodnotení pohybu. Rozlišujeme, kedy robot nájde poklad, narazí na stenu alebo ani jedno z toho. V prípade, že robot narazí na stenu, ostane stáť na mieste.

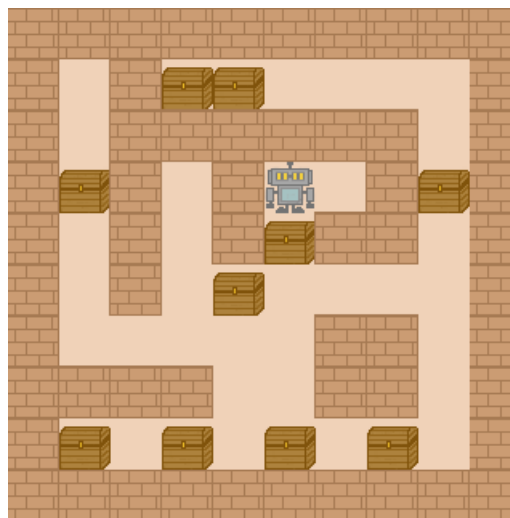
Kríženie aj mutácie prebiehali vždy s dopredu danou pravdepodobnosťou na celý beh genetického algoritmu.

4 Aplikácia genetického algoritmu

Daný problém riešime oddelene vzhľadom na dve rôzne mapy (obrázok 1, obrázok 2), s ktorými budeme pracovať pri ďalších analýzach. Počet pokladov na každej mape je 10, veľkosť každej mapy je 10×10 . Rovnako je fixná počiatočná poloha robota.



Obrázok 1: jednoduchá mapa



Obrázok 2: komplikovaná mapa

Budeme uvažovať zmenu nasledujúcich parametrov:

POPULATION SIZE	určuje veľkosť populácie po celú dobu výpočtu
GENERATIONS	počet generácií
CROSSOVER	pravdepodobnosť kríženia
MUTATION	pravdepodobnosť mutácie génov
MINBOUND & MAXBOUND	minimálna a maximálna dĺžka trasy robota

Pohyb robota na mape je reprezentovaný červenou farbou tak, že opakovaný priechod určitým miestom spôsobí silnejšie červené zafarbenie. Tým ukážeme, kadiaľ robot prešiel, a zároveň kde robil zbytočné kroky navyše.

5 Výsledky

Pre jednoduchosť sme počas analýzy nemenili parameter veľkosti populácie. Ten je vo všetkých behoch rovný 100. Ďalej nemení horné a dolné ohraňenie dĺžky cesty. Všetky cesty sú dlhé 20 – 60

symbolov. Definujeme dve rôzne fitness funkcie a pre každú identifikujeme, pri akých pravdepodobnostiach kríženia a mutácie funguje najlepšie.

5.1 Výsledky na základe fitness funkcií

1.stratégia

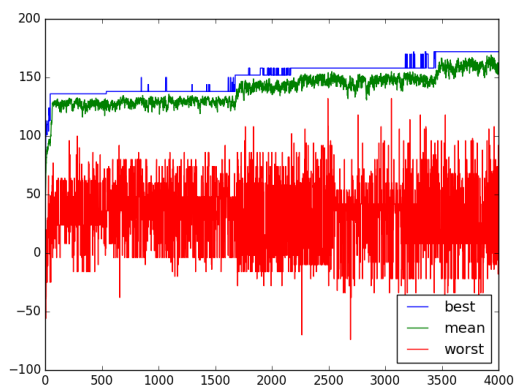
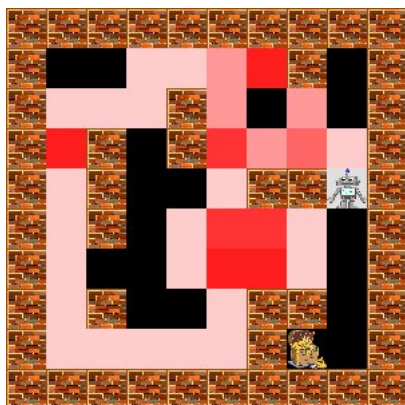
Ako prvú sme otestovali asi najintuitívnejší prístup. Robot dostal nasledujúce ohodnotenia za svoje kroky:

- nenarazíš na stenu ale nenájdeš poklad : 1 bod
- narazíš na stenu : -5 bodov
- nájdeš poklad : 15 bodov

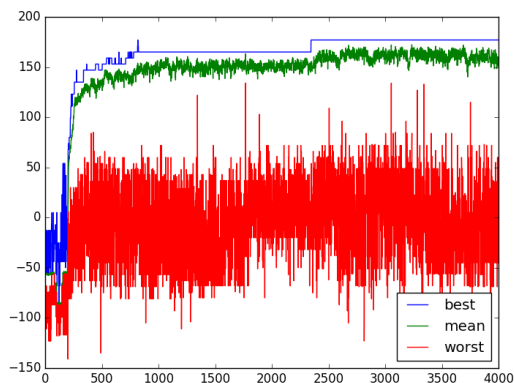
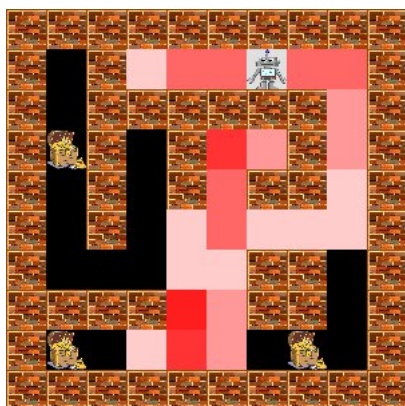
Táto stratégia reprezentuje to, že robota odmeníme ak nájde poklad, a potrestáme, ak narazí do steny. Aby sa po iteráciach neusadil na jednej lokálne dobrej možnosti, odmeníme ho aj za to že chodí. Pri veľmi malom pomere odmeny za poklad a chodenie však táto stratégia spôsobí neustále chodenie dookola, čo nebolo cieľom.

Túto stratégiu sme použili na oba typy máp, kde vstupné parametre boli:

POPULATION SIZE = 100 GENERATIONS = 4000 CROSSOVER = 0.7 MUTATION = 0.3 resp. 0.4



Jednoduché bludisko: najlepšia cesta, proces učenia



Kompikované bludisko: najlepšia cesta, proces učenia

Vyhodnotenie: Výsledky ukazujú, že na jednoduchšej mape sa bol robot schopný pomerne dobre naučiť sa nájsť poklad. Naopak, komplikované bludisko mu robilo väčšie problémy. Na komplikovanom bludisku sme skúsili aj upravené parametre, väčšiu náhodnosť mutácii, aby boli objavené iné možnosti, ako aj zvyšovanie boundu na dĺžku cesty, avšak výsledok dosiahnutý s danými parametrami považujeme za najlepší. Daná stratégia teda pre komplikované bludisko nie je vhodná.

2.stratégia

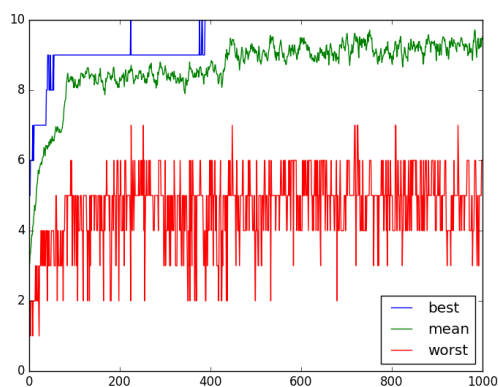
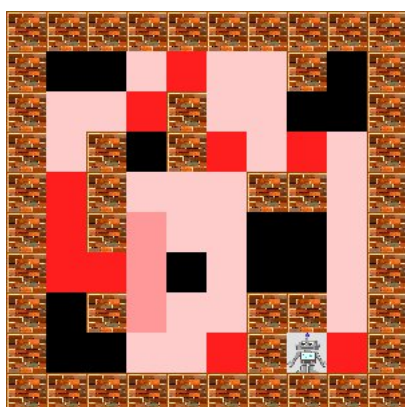
Ako ďalšiu sme otestovali asi nasledovný prístup:

- nenarazíš na stenu ale nenájdeš poklad : 0 bodov
- narazíš na stenu : 0 bodov
- nájdeš poklad : 1 bod

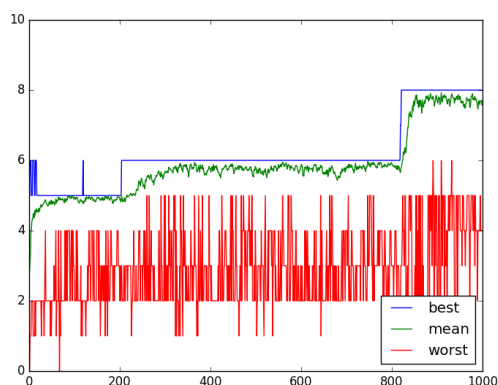
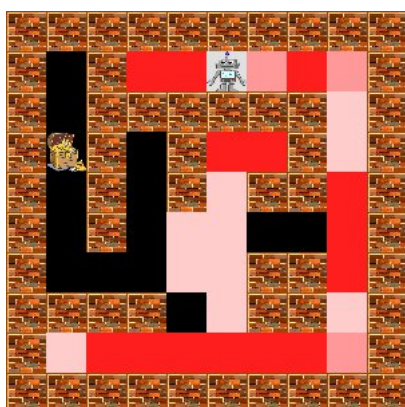
Táto stratégia reprezentuje to, že robota odmeníme len ak nájde poklad, a teda sa predpokladá, že by sa ich mal pokúsiť nájsť čo najviac. Táto stratégia sa považuje za jednoduchšiu, a preto sme ju spustili len na 1000 iteráciach.

Túto stratégiu sme použili na oba typy máp, kde vstupné parametre boli:

POPULATION SIZE = 100 GENERATIONS = 1000 CROSSOVER = 0.6 MUTATION = 0.4



Jednoduché bludisko: najlepšia cesta, proces učenia



Kompikované bludisko: najlepšia cesta, proces učenia

Vyhodnotenie: Maximálny možný počet obdov dosiahnuteľný pri tejto stratégii bol 10. Výsledky ukazujú, že robot má pre danú stratégiu dobré výsledky na oboch typoch máp. Túto stratégiu teda považujeme za veľmi efektívnu v porovnaní s jej jednoduchosťou. Ukázané výsledky predstavujú najlepší beh z 5, kde priemerné sú väčšinou o 1-2 poklady horšie.

5.2 Parametrizácia vstupných parametrov

Pre obe mapy sme nagenerovali rovnaku iniciálnu populáciu a sledovali sme, ako vstupné parametre ovplyvnia dané výsledky. Minimálna a maximálna dĺžka trasy ostala po celú dobu rovnaká, tak isto ako aj veľkosť populácie. Každý beh sme spustili 4 krát a výsledky spriemerovali.

5.2.1 Jednoduché bludisko

1.stratégia

Najprv sme testovali závislosti na kombinácii pravdepodobností kríženia a mutácie.

POPULATION	GENERATIONS	CROSSOVER	MUTATION	best	mean	worst
100	1000	0.8	0.2	164.5	158.82	33.5
100	1000	0.8	0.4	170.25	157.24	66.75
100	1000	0.6	0.2	167.25	155.98	65.25
100	1000	0.6	0.4	206.75	178.25	29.0

Potom sme vybrali najúspešnejšiu stratégiu a skúmali sme ako je ovplyvnená počtom iterácií.

POPULATION	GENERATIONS	CROSSOVER	MUTATION	best	mean	worst
100	100	0.6	0.4	154.25	131.84	37.5
100	500	0.6	0.4	175.25	149.83	22.75
100	1000	0.6	0.4	206.75	178.25	29.0
100	2000	0.6	0.4	200.75	171.31	33.5
100	4000	0.6	0.4	194.25	167.56	53.5

2. stratégia

Najprv sme testovali závislosti na kombinácii pravdepodobností kríženia a mutácie.

POPULATION	GENERATIONS	CROSSOVER	MUTATION	best	mean	worst
100	1000	0.8	0.2	7.5	7.36	4.5
100	1000	0.8	0.4	8.25	8.12	5.75
100	1000	0.6	0.2	9.0	8.56	4.25
100	1000	0.6	0.4	8.75	8.185	4.5

Potom sme vybrali najúspešnejšiu stratégiu a skúmali sme ako je ovplyvnená počtom iterácií.

POPULATION	GENERATIONS	CROSSOVER	MUTATION	best	mean	worst
100	100	0.6	0.2	8.25	7.53	4.75
100	500	0.6	0.2	8.25	7.92	4.75
100	1000	0.6	0.2	9.0	8.56	4.25
100	2000	0.6	0.2	9.25	8.97	4.0
100	4000	0.6	0.2	8.25	7.98	4.25

Vyhodnotenie: blablablabla

5.2.2 Komplikované bludisko

1.stratégia

Najprv sme testovali závislosti na kombinácii pravdepodobností kríženia a mutácie.

POPULATION	GENERATIONS	CROSSOVER	MUTATION	best	mean	worst
100	1000	0.8	0.2	130.5	126.08	40.25
100	1000	0.8	0.4	158.25	147.49	61.5
100	1000	0.6	0.2	147.0	137.4	22.25
100	1000	0.6	0.4	156.75	135.8725	-11.75

Potom sme vybrali najúspešnejšiu stratégiu a skúmali sme ako je ovplyvnená počtom iterácií.

POPULATION	GENERATIONS	CROSSOVER	MUTATION	best	mean	worst
100	100	0.8	0.4	103.5	87.77	31.55
100	500	0.8	0.4	136.5	124.5	18.5
100	1000	0.8	0.4	158.25	147.49	61.5
100	2000	0.8	0.4	147.75	139.68	23.0
100	4000	0.8	0.4	167.25	158.38	68.25

2. stratégia

Najprv sme testovali závislosti na kombinácii pravdepodobností kríženia a mutácie.

POPULATION	GENERATIONS	CROSSOVER	MUTATION	best	mean	worst
100	1000	0.8	0.2	6.5	6.43	5.25
100	1000	0.8	0.4	7.0	6.87	5.0
100	1000	0.6	0.2	6.5	6.3625	3.5
100	1000	0.6	0.4	7.5	7.30	4.5

Potom sme vybrali najúspešnejšiu stratégiu a skúmali sme ako je ovplyvnená počtom iterácií.

POPULATION	GENERATIONS	CROSSOVER	MUTATION	best	mean	worst
100	100	0.6	0.2	6.25	5.95	3.5
100	500	0.6	0.2	7.25	7.02	4.0
100	1000	0.6	0.2	7.5	7.30	4.5
100	2000	0.6	0.2	7.5	7.29	5.0
100	4000	0.6	0.2	7.5	7.23	4.25

Vyhodnotenie: blablablabla

6 Záverečný pokec

Sme krásne a skvele!