

Evolutionary Algorithm used in a Chess Game

PETER HOMOLA

Zdroje

2/17

- ▶ Autor práce : Andrej Urminský
- ▶ Vedúci práce : Ing. Zbyšek Gajda
- ▶ Rok publikácie : 2007



Abstrakt

3/17

- ▶ Táto práca sa zaoberá návrhom evolučného algoritmu pre umelú inteligenciu v hre šach.
- ▶ K dosiahnutiu tohto cieľa je použitý tzv. genetický algoritmus.
- ▶ Pre implementáciu algoritmu a grafického rozhrania bol využitý programovací jazyk Java vo vývojovom prostredí Eclipse

História šachových programov

4/17

- ▶ Claude Shannon ->návrh -> prvý počítačový program hrajúci šach
- ▶ Ohodnocovacia funkcia -> materiálová výhoda, formácia pešiakov, pozícia figuriek, mobilita, uväznenie, útoky a pod.

Metódy pre nájdenie najlepšieho ťahu :

1. Prehľadávanie do základnej hĺbky ťahov a potom použitie algoritmu minimax
2. Selektívne vyhľadávať ramená stromu hry do rozličnej hĺbky pomenované alfa-beta

Zaujímavosti

5/17

- ▶ Prvý fungujúci šachový program -> r. 1956
- ▶ Použitý algoritmus alfa-beta -> r. 1958
- ▶ Machack VI. prvá výhra nad klubovým hráčom -> r. 1967
- ▶ Čestný člen Šachovej Federácie Spojených Štátov
- ▶ Ohodnotený na 1640 ELO

ELO -> medzinárodný systém merania výkonu v šachu

Porovnanie algorimov minimax a alfa-beta

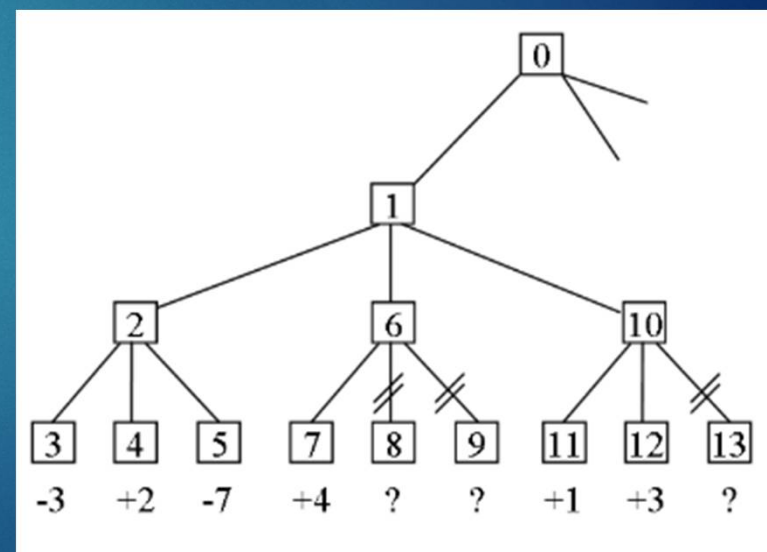
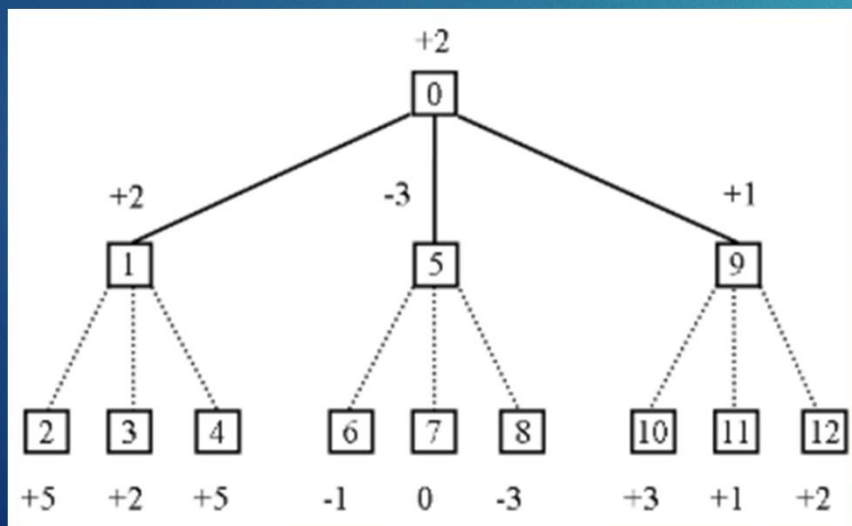
- ▶ Alfa-beta je mierne modifikovaný minimax
- ▶ Zabraňuje prehľadávať nepotrebné ťahy
- ▶ Minimax -> úplné prehľadávanie stavového priestoru
- ▶ Alfa-beta -> čiastočné prehľadávanie stavového priestoru

- ▶ Matematická analýza -> namiesto x -> \sqrt{x} -> úspora $5 * \sqrt{x}$

Minimax a alfa-beta

7/17

- Počítač vychádza z okamžitej pozície na šachovnici (0) a prevedie v pamäti prvý z troch možných ťahov
- Vytvorí pozíciu (1), v ktorej má protivník na výber tri ťahy
- Alfa rezy zabránia zbytočnému vyšetrovaniu ťahov hráča A, bety rezy potom vyšetrovaniu ťahov hráča B
- Na konečnom výsledku to nič nezmení



Evolučné algoritmy

8/17

- ▶ Prirodzený výber -> väčší fitness -> väčšia pravdepodobnosť
- ▶ Fitness -> kvantitatívna miera schopnosti prežiť
- ▶ Náhodný genetický drift -> náhodná mutácia, náhodná smrť

- ▶ Reprodukčný proces -> vytváranie potomkov
- ▶ Tvorba genetickej informácie -> výmenou genetickej informácie rodičov
- ▶ Náhodný výber časti chromozomov -> informácie -> nový potomok
- ▶ Kríženie -> zväčšuje rýchlosť a účinnosť evolúcie

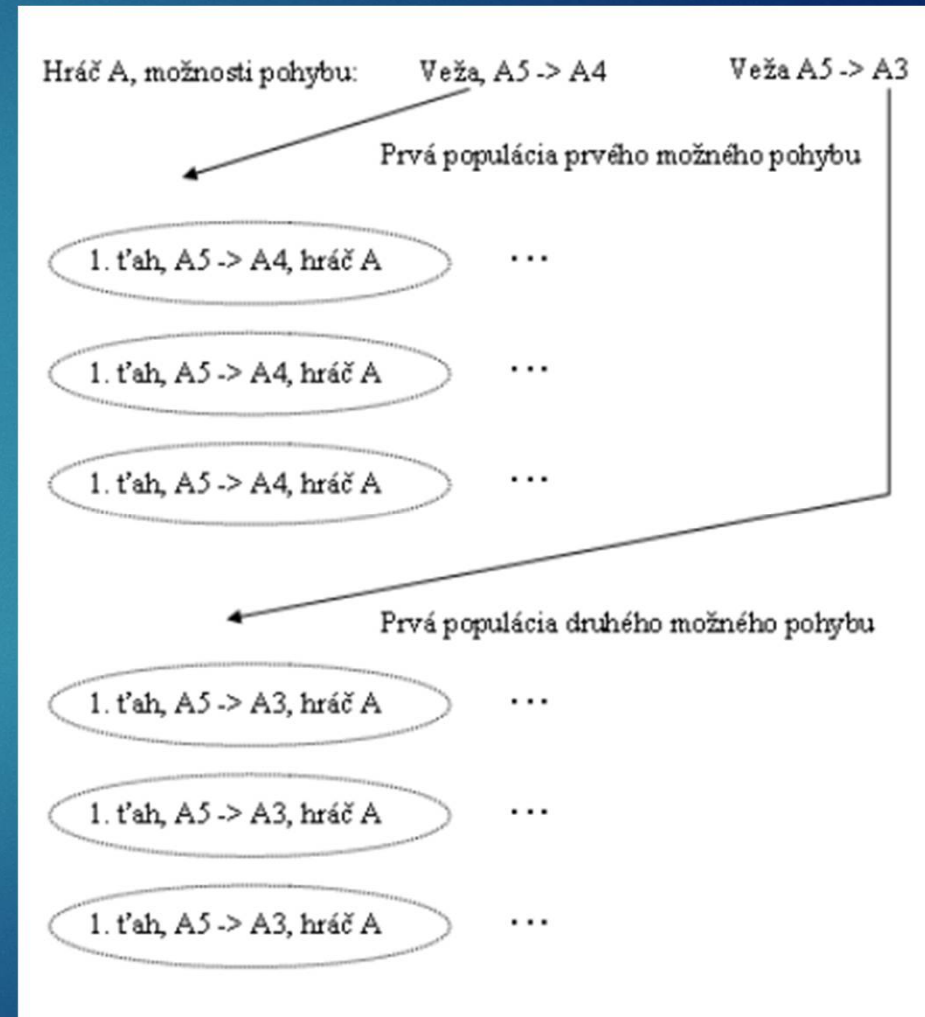
Návrh evolučného algoritmu

9/17

- ▶ Populácia -> chromozómy (jedinci) -> Gény (základná jednotka)
- ▶ Jeden gén -> jeden možný ťah v postupnosti ťahov
- ▶ Jedna postupnosť ťahov -> jeden chromozóm
- ▶ EA pracuje s postupnosťami ťahov -> ohodnocuje
- ▶ Ohodnocovacia funkcia -> proces mutácie -> novovytvorená generácia

Etapy evolučného algoritmu

- ▶ Prvý ťahu -> výber všetky figúrky hráča A
- ▶ Figúrka ma -> jeden alebo viac ťahov
- ▶ Z každého možného ťahu každej figúrky sa vytvorí generácia o veľkosti niekoľkých chromozómov
- ▶ Po vygenerovaní generácie -> súťaž -> najlepší jedinec
- ▶ Chromozóm s najvyššími fitness -> základ novej generácie.
- ▶ Vybraný chromozóm je ponechaný ->
- ▶ Ďalšie chromozómy -> mutácia vybraného chromozómu -> vznikne nová generácia
- ▶ Znova prebieha výber najlepšieho jedinca
- ▶ Toto sa opakuje -> vygenerovania vopred zadaného počtu generácií.



Počiatočná generácia

- ▶ Prvý gén každého chromozómu -> prvý ťah hráča A
 - ▶ Ďalšie gény chromozómu -> generujú náhodne
 - ▶ Hráč B -> všetky možnosti pohybu -> náhodný výber
 - ▶ Koniec -> remíza alebo prehra jedného hráča
-
- ▶ Po vygenerovaní populácie -> súťaž potomkov
 - ▶ Porovnanie fitness hodnôt -> výber jedného chromozómu s najvyššou fitness

Ohodnocovacia funkcia

- ▶ Spočítanie materiálovej hodnoty šachovnice – suma materiálovej hodnoty figúrky k z celkového počtu n figúrok na šachovnici
- ▶ Hodnoty figúrok sú použité štandardné -> pešiak ma hodnotu 1, strelec a kôň majú hodnotu 3, veža 5 a dáma 9
- ▶ V prípade bielych figúrok sú tieto hodnoty kladne, v prípade čiernych záporne
- ▶ Ak hráč dal súperovi šach mat -> s bielymi figúrkami-> fitness vráti 1000000
-> s čiernymi figúrkami-> fitness vráti -1000000
- ▶ Remíza -> fitness vráti 0

Výsledky

13/
17

- Parametre evolučného algoritmu: počet génov v chromozóme je 6 (t.j. 6 poťahov), počet vygenerovaných generácii 200 a počet chromozómov v populácii 3. Program s navrhnutým evolučným algoritmom prehral. Jeden ťah, napríklad e2e4, označuje počiatočnú súradnicu figúrky (E2) a cieľovú (E4).

Poradie ťahu	biely hráč	čierny hráč	poradie ťahu	biely hráč	čierny hráč
1	e2e4	a7a5	10	g2g4	d6e5
2	b1a3	e7e5	11	f1d3	c8g4
3	h2h4	h7h5	12	f2f4	e5f4
4	d1h5	h8h5	13	h4h5	g7g6
5	d2d4	g8f6	14	h5g6	h7h1
6	d4e5	f6g8	15	g1e3	f7g6
7	a1b1	h5h7	16	e2g1	h1g1
8	e1d2	b7b6	17	b1a1	g1d1
9	c2c4	d7d6	18	d2c3	d8d3

Výsledky

- Parametre evolučného algoritmu boli zmenené nasledovne: počet generácii 300, počet chromozómov v populácii 5 a počet génov v chromozóme zostal nezmenený, t.j. 6. Dosiahnutý výsledok je o niečo lepší oproti predchádzajúcemu. Napriek tomu, že internetový šach mal materiálovú prevahu, výsledkom je remíza.

Poradie ťahu	biely hráč	čierny hráč	poradie ťahu	biely hráč	čierny hráč
1	a2a4	e7e5	14	e2g1	a3c1
2	h2h4	d7d6	15	f1c4	b3c4
3	a1a3	h7h5	16	g2g4	c4d5
4	b2b4	f7f5	17	e1f2	d5h1
5	c2c4	d6d5	18	f2g3	h1g1
6	a3h3	d5c4	19	g3f3	g1g4
7	d1b3	c4b3	20	f3e3	g4a4
8	f2f4	d8d4	21	b5b6	a7b6
9	h3b3	d4f4	22	e3f2	a4h4
10	e2e4	f4e4	23	f2e3	a8a2
11	g1e2	e4c4	24	e3d3	a2d2
12	b1a3	c4b3	25	d3c3	c8e6
13	b4b5	f8a3			

Celkový výsledok po 10 hrách

	šachový program Chess	navrhnutý evolučný algoritmus
počet výhier	9	0
počet remíz	1	1

- ▶ Navrhnutý algoritmus nevyberá ťahy veľmi "rozumne". Toto je spôsobené tým, že počas celej evolúcie sa zohľadňujú iba ťahy a materiálové hodnoty chromozómov hráča A. Teda akoby hráč A nepočítal s tým, že hráč B bude uvažovať pri reakciách na ťahy hráča A "rozumne" a že sa bude tiež snažiť vyhrať

- ▶ Vylepšenie algoritmu -> stratégiou typu: mutácia ťahov hráča A, výber najlepšieho chromozómu v prospech hráča A, mutácia ťahov hráča B, výber najlepšieho chromozómu v prospech hráča B.
- ▶ Vylepšiť funkciu ohodnocovania :
- ▶ Zahnúť mobilita, teda počet možných ťahov hráča (čím viac možných ťahov, tým je ťah výhodnejší)
- ▶ Zistenie pozície dvojitého pešiaka (ak sa dvaja pešiaci nachádzajú v stĺpci za sebou, je to väčšinou považované za nevýhodu) a pod.

Ďakujem za pozornosť

[HTTPS://DSPACE.VUT.CZ/SERVER/API/CORE/BITSTREAMS/BDB36871-
FFA8-4DB6-9415-72E5B19B2C0F/CONTENT](https://dspace.vut.cz/server/api/core/bitstreams/bdb36871-ffa8-4db6-9415-72e5b19b2c0f/content)