

Herní algoritmy, předn.1

Jan Hric, Petr Baudiš

ZS: 0/2 Z, NAIL103

26. října 2012

Obsah

- 1 Standardní algoritmy: Minimax
- 2 Monte Carlo Tree Search
- 3 Proof-number search
- 4 Prohledávání založené na hrozbách
- 5 ... a další

Outline

- 1 Standardní algoritmy: Minimax
- 2 Monte Carlo Tree Search
- 3 Proof-number search
- 4 Prohledávání založené na hrozbách
- 5 ... a další

Hry dvou hráčů s úplnou informací ...

Zakladní část: Hry dvou hráčů s úplnou informací a nulovým součtem

- Zahrňuje: šachy, dáma, piškvorky, **go**, arimaa, hex, ... GIPF projekt
- Nezahrňuje: bridge, poker, Stratego (neúplná informace/nejistota); vrhcáby/backgammon (náhoda), věžňovo dilema (nenulový součet), Risk (n hráčů)
- na pomezí: optimalizační problémy \approx hry 1 hráče \approx hlavolamy, heuristické evaluační funkce

Techniky

- minimax, alfa-beta, ... a heuristiky
- Monte Carlo Tree Search
- Exaktní prohledávání - Proof Number Search: **koncovky**, lokální cíle
- Prohledávání založené na hrozbách: **λ -search**, null move search
- Implementační techniky: herní strom, Transpoziční Tabulky, paralelizace, bitboard

- Další: hry 1 hráče, hraní za nejistoty a s neúplnou informací
- (Hraní obecných her, GGP: General Game Playing; učení ohodnocovacích funkcí)

Minimax

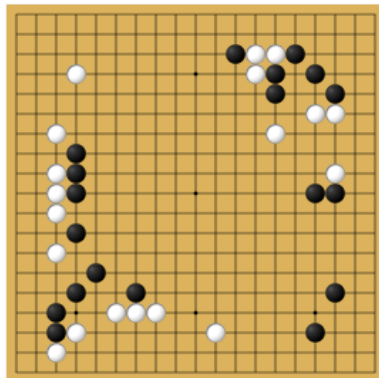
- Základy: Minimax, alfa-beta, ...
- Varianty: Negamax, Principiální varianta, (Negascout), Iterativní prohlubování, MTD(f), (SSS*) ...
- Impl: Transpoziční tabulky, okno, history heuristika, killery, null-move, ...
- Související pojmy: heuristická ohodnocovací funkce, efekt horizontu, ...

Outline

- 1 Standardní algoritmy: Minimax
- 2 Monte Carlo Tree Search**
- 3 Proof-number search
- 4 Prohledávání založené na hrozbách
- 5 ... a další

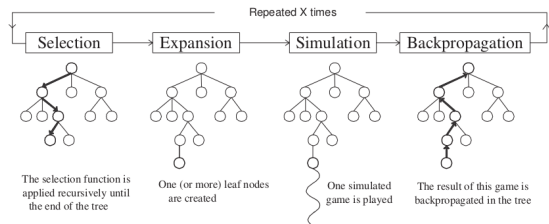
Monte Carlo Tree Search (idea)

- Desková hra Go: široký stavový prostor, žádná dobrá ohodnocovací funkce



Monte Carlo Tree Search (idea)

- Desková hra Go: široký stavový prostor, žádná dobrá ohodnocovací funkce
- Ohodnocujeme pomocí *náhodných simulací* — výsledku sehrávek do koncové pozice
- Pravděpodobnostní minimaxový strom, každý uzel má očekávanou pravděpodobnost výhry μ



Monte Carlo Tree Search (idea)

- Desková hra Go: široký stavový prostor, žádná dobrá ohodnocovací funkce
- Ohodnocujeme pomocí *náhodných simulací* — výsledku sehrávek do koncové pozice
- Pravděpodobnostní minimaxový strom, každý uzel má očekávanou pravděpodobnost výhry μ
- Výběr synů při sestupu: *Multi-armed Bandit* (exploration vs. exploitation)
- Strom roste dynamicky podle směru prohledávání

Monte Carlo Tree Search (praxe)

- Na loupežníkovi záleží — UCT, RAVE
- Simulace vlastně nemusí být tolik náhodné: doménové heuristiky, vyvažování
- Co s čerstvými listy? Inicializace na základě doménových heuristik
- Paralelizace (výhody stochastického algoritmu)

Monte Carlo Tree Search (praxe)

- Na loupežníkovi záleží — UCT, RAVE
- Simulace vlastně nemusí být tolik náhodné: doménové heuristiky, vyvažování
- Co s čerstvými listy? Inicializace na základě doménových heuristik
- Paralelizace (výhody stochastického algoritmu)

- Doménové heuristiky — ale jaké a jak je efektivně počítat? Efektivní reprezentace, pattern matching, strojové učení
- Otevřené problémy — sdílení informací, taktické oblasti

Outline

- 1 Standardní algoritmy: Minimax
- 2 Monte Carlo Tree Search
- 3 Proof-number search**
- 4 Prohledávání založené na hrozbách
- 5 ... a další

Proof-number search (PNS)

Pracuje nad And/Or stromem (implicitně zadaným)

PNS určí splnitelnost cíle: ano/ne/nevím (tj. přesnou minimax. hodn.)

Určen pro koncovky a lokální cíle (vzetí figury, spojení v go)

- Počítá proof-number a disproof-number (PN a DN): minimální počty uzlů, které potřebuju pro dokázání resp. vyvrácení cíle
- Sestup do listu k rozvinutí se vybírá podle nejmenších PN/DN synů pro MAX/MIN vrchol (tj. nejnadějnější); ... a propagace hodnot nahoru
- Základní alg. je „do šířky“ (velká spotřeba paměti), jsou varianty počítající (iterovaně) „do hloubky“

Proof-number search: techniky

PNS neformálně: preferuje úzké stromy, staví nerovnoměrné stromy, zvládá pohyblivé figurky (shogi - jap. šachy)

Techniky

- *chytré rozvíjení*
- *simulace*

Outline

- 1 Standardní algoritmy: Minimax
- 2 Monte Carlo Tree Search
- 3 Proof-number search
- 4 Prohledávání založené na hrozbách**
- 5 ... a další

λ -search

- Hrozba: tah, na který musí soupeř (lokálně) odpovědět, jinak prohraje (resp. zhorší si situaci); tempo/sente
- Řád hrozby n : kolik tahů může soupeř vynechat
- Příklad: piškvorky
- Základní alg. buduje λ_n -stromy - útočné a obranné - z hlediska jednoho hráče
- Vhodné pro koncovky a taktické varianty, buduje nerovnoměrné stromy, používá se pro "dořešení úloh": piškvorky, hex

Outline

- 1 Standardní algoritmy: Minimax
- 2 Monte Carlo Tree Search
- 3 Proof-number search
- 4 Prohledávání založené na hrozbách
- 5 ... a další**

Možnosti

- Historie: PNS 1994, λ 2000, UCB 2002, UCT/MCTS 2006, (GGP 2005)
- Řešení her: piškvorky (1994), hex 9x9 (část) 2010, go 5x5 2005, checkers/dáma 2007

Vlastní práce:

- Kombinace algoritmů, nové **efektivní** strategie (a implementační triky)
- Varianty, odstraňování nevýhod (šetření paměti, go: open space solver)
- Aplikace na jiné hry (Arimaa, Quoridor, Samegame): jiné techniky jsou vhodné, „implicitní“ vlastnosti
- Doménové znalosti a jejich integrace

Jiné hry

Hry 1 hráče

- neúplné prohledávání velkých stavových prostorů
- mnoho různých přístupů
- **MCTS** je aplikovatelné
- („No free lunch theorem“ in search and optimization)

Hry s neurčitostí

- Minimax lze upravit na ExpectiMax

Různé

- Drobnosti: článek o optimální přepisovací strategii v TT: 2big
- GHI: Graph History Interaction
- Doménové znalosti, odstraňování nevýhod (go: nakade, criticality; Arimaa: zajetí velblouda)
- Učení heuristických ohodnocovacích funkcí (Backgammon: Tesauro 1992, reinforcement learning), self-play, ladění parametrů (velký prostor, velký rozptyl, citlivé)
- Učení Opening book, analýza pozice
- Hry těžké pro počítače: Arimaa, (Game of the) Amazons, GIPF project (Dvonn, Gipf, Zertz, Yinsh, Punct, Tzaar, (Tamsk))
- SAT, QBF: Quantified Boolean Formula - And/Or strom
- CGT: Combinatorial Game Theory (viz)