

# Algoritmy pro práci s neúplnou informací

Michal Krkavec

Herní algoritmy, NAIL103

23. listopadu 2011

# Obsah

## Náhoda

Expectimax

## Neúplné informace

Monte Carlo Tree Search

Perfect Information Monte Carlo

## Realtime plánování

Plánování v RTS

Monte Carlo Plánování

# Expectimax

- ▶ úprava Minimaxu pro hry s úplnou ale nedokonalou informací
- ▶ některé uzly odpovídají stavům hry
- ▶ speciální uzly pro náhodu
  - ▶ všichni synové mají svou pravděpodobnost, že nastanou
  - ▶ hodnotou uzlu náhody je střední hodnota synů (vážený průměr)
- ▶  $\alpha\beta$ -prořezávání při omezené ohodnocovací funkci

# MCTS

- ▶ best-first search algoritmus
- ▶ rozvinutí nalezeného listu
- ▶ pseudonáhodné simulace až do koncových stavů
- ▶ propagace výsledku simulace ke kořeni
- ▶ nepotřebuje expertní znalosti o hře (heuristické funkce)
- ▶ tudíž je dostatečně univerzální

# Perfect Information Monte Carlo

- ▶ neúplnou informaci obchází vzorkováním herního světa
- ▶ následně je daný stav světa vyhodnocen (přímo / heuristicky) pomocí MCTS
- ▶ úspěchy u karetních her (Bridge, Skat, Hearts)
- ▶ problémy?

## Potíže PIMC

PIMC selhává:

- ▶ *strategy fusion* (vzorkování → přestože dva světy splývají, PIMC si myslí, že je dokáže rozeznat)
- ▶ *non-locality* (závislost na hodnotách **mimo** svůj podstrom)
- ▶ jak často?

Měřitelné vlastnosti stromu:

- ▶ *leaf correlation* (závislost listů)
- ▶ *bias* (odchylka, nadržování)
- ▶ *disambiguation factor* (zmenšování neúplné informace, počet možných světů)

## Umělé stavové stromy s neúplnou informací

Zjednodušení:

- ▶ stupeň uzlů hráče je 2
- ▶ veškerá náhoda (rozdělení neúplných informací) se odehraje v kořeni stromu
- ▶ každý hráč může začínat v  $K$  možných pozicích → stupeň kořene je  $K^2$
- ▶ hodnoty koncových stavů  $\in \{-1, 1\}$

Parametry:

- ▶ leaf correlation, bias, disambiguation factor
- ▶ konkrétně  $K = 8$ , hloubka stromu také 8, 10000 vygenerovaných stromů, na každém se odehrály 2 partie
- ▶ soupeřové: PIMC, random player, CFR (Counter-factual Regret alg. – téměř optimální alg. založený na klasickém prohledávání stavového prostoru)

## Výsledky partií

Na umělých stromech:

- ▶ nejhorší výsledky při nízké leaf correlation (závislosti listů)
- ▶ při nižší závislosti uzlů hraje roli i bias (nadržování)
- ▶ disambiguation factor (zmenšování neúplné informace)  
nejednoznačný vliv

Analýza lidmi sehraných partií hry Skat:

- ▶ zmenšování neúplné informace vždy blízko 0.6
- ▶ závislost listů vysoká
- ▶ odchylka (nadržování, bias) spíše rovnoměrný
- ▶ → dobré výsledky pro PIMC proti random a srovnatelné s CFR



## Výsledky partií

Poker (PIMC vs random vs CFR):

- ▶ nulové odkrývání informací
- ▶ závislost listů 0,5
- ▶ odchylka 0,5
- ▶ → slabší výsledky

## Umělá inteligence v RTS

- ▶ *skriptování* – nejčastější řešení, pevně daná pravidla, málo přizpůsobivý předvídatelný soupeř, klidně statisíce řádků kódu
- ▶ *plánování* – práce se zdroji, stavba jednotek, plánování soubojů
- ▶ různé úrovně plánování
  - ▶ kontrola jednotek – většinou defaultní (reaktivní) chování
  - ▶ taktické (střednědobé) plánování – jakým způsobem vést útok, hledat nepřítel, stavět obranu, ...
  - ▶ strategické (dlouhodobé) plánování – kdy stavět jednotky a jaké, kdy zaútočit, jakým směrem se soustředit
- ▶ další plánování na úrovni jednotek, např. pathfinding

# Monte Carlo Plánování

- ▶ pro plánování ve hrách s neúplnou informací, náhodou a současným hraním „tahů“
- ▶ abstrahuje stavový prostor
- ▶ vzorkuje stavy s neúplnou informací
- ▶ různé úrovně plánování?
- ▶ parametrizovatelné základní plány (útok, stavba budov, stavba jednotek, pohyb)

# MCPlan

Monte Carlo přístup:

- ▶ pro daný (abstraktní) stav hry vygeneruj  $P$  plánů
- ▶ simuluj každý plán celkem  $S$ -krát
- ▶ do maximální hloubky  $K$  kroků
- ▶ ohodnot' závěrečný stav
- ▶ sluč jednotlivá ohodnocení z  $S$  simulací
- ▶ vyber nejlepší plán

# MCPlan

- ▶ pohybujeme se po abstraktních stavech
- ▶ obecný univerzální algoritmus
- ▶ podle konkrétní hry musíme umět:
  - ▶ definovat plány
  - ▶ ohodnotit abstraktní stavy

# MCPlan ve hře Capture the Flag

## Herní svět:

- ▶ symetrická mapa se zdmi
- ▶ soupeři mají základny proti sobě
- ▶ cílem je dovézt soupeřovu vlajku do své základny
  - ▶ polohu vlajky známe
  - ▶ vlajku může nést jen soupeřova jednotka
- ▶ pouze bojové jednotky
  - ▶ počet životů
  - ▶ dostřel, palebná síla

Neřeší zásoby zdrojů, stavbu jednotek.

Soustředí se na průzkum mapy, obranu a krádež vlajky.

# Realizace hry

- ▶ abstrakce stavu
  - ▶ mapa rozdělena na větší políčka
  - ▶ životy jednotek a jejich poloha na políčkách mapy
  - ▶ polohy vlajek a objevených zdí
- ▶ ohodnocovací funkce
  - ▶ zdraví jednotek
  - ▶ prozkoumání mapy + aktuální viditelnost
  - ▶ vzdálenost vlajky od základny
- ▶ simulace
  - ▶ pohyb jednotek na náhodné políčko ve svém okolí
  - ▶ odebrání životů nejbližšímu nepříteli v dostřelu
- ▶ přeplánování
  - ▶ při událostech objevení, zničení nepřítele, zvednutí vlajky

# Výsledky

Sledovat lze různé parametry:

- ▶ počet plánů (vs random)
- ▶ počet jednotek (vs random)
- ▶ složitost mapy (vs rush-the-flag)
- ▶ abstrakce stavu
  - ▶ mapa rozdělena na větší políčka
  - ▶ životy jednotek a jejich poloha na políčkách mapy
  - ▶ polohy vlajek a objevených zdí
- ▶ při událostech objevení, zničení nepřítele, zvednutí vlajky



## MCTS v dalších hrách s neúplnou informací

- ▶ deskové hry – Fantom staré Prahy, Osadníci, Scrabble, Vrhcáby (Backgamon)
- ▶ další – Pacman, Magic: The Gathering

# Literatura

- ▶ Chaslot, Bakkes, Szita, Spronck:  
*Monte-Carlo Tree Search: A New Framework for Game AI*
- ▶ Long, Sturtevant, Buro, Furtak:  
*Understanding the Success of Perfect Information Monte Carlo Sampling in Game Tree Search*
- ▶ Chung, Buro, Schaeffer:  
*Monte Carlo Planning in RTS Games*
- ▶ Ward, Cowling:  
*Monte Carlo Search Applied to Card Selection in Magic: The Gathering*