

GENERAL GAME PLAYING

Marika Ivanová

Seminář Herní Algoritmy, 12. 12. 2012

Obsah

- Koncept GGP
- Herní model
- Reprezentace znalostí
- Hráči (General Game Players)
- Řídící systém
- Závěr

Koncept GGP

- Snaha o vytvoření AI systému schopného „porozumět“ pravidlům libovolné hry a naučit jej hrát bez zásahu člověka
- Oproti specializovaným programům nezávisí na algoritmech navržených pro konkrétní hry
- AAAI National Conference 2005: GGP Competition
- Samotná idea je starší: Jacques Petrat: Realization of a general game playing program (1968)

Koncept GGP

- AAI Competition (<http://games.stanford.edu>)
 - 2005, Stanford University
 - 2 fáze (qualification, runoff)
 - Hry s úplnou informací
 - Hry jednoho hráče (bludiště)
 - Kompetitivní hry (Tic-Tac-Toe, varianty šachu)
 - Kooperativní prvky
 - Později i hry s částečnou informací
 - Odměna \$10,000 pro vítěze

Koncept GGP

- Základní oblasti výzkumu GGP
 - Reprezentace znalostí
 - Prohledávání
 - Plánování
 - Učení
- Různé typy her
 - částečná/úplná informace
 - s/bez komunikace mezi hráči
 - Obecně n hráčů

Herní model

- Uvažujeme konečné synchronní hry
- Končený a neměnný počet hráčů
- Formálně vyjádříme hru jako stavový automat:

S – množina stavů

$\{r_1 \dots r_n\}$ – role ve hře n hráčů

$A_1 \dots A_n$ – n množin akcí, každá množina pro roli

$l_1 \dots l_n$ – legální akce ve stavu, kde $l_i \subseteq A_i \times S$

$u: S \times A_1 \times \dots \times A_n \rightarrow S$ – přechodová funkce

$s_1 \in S$ – počáteční stav

$t \subseteq S$ – koncové stavy

g_1, \dots, g_n – kde $g_i \subseteq S \times \mathbb{N}$, cílová relace

Reprezentace znalostí

- Potřeba formálního symbolického jazyka pro popis pravidel libovolné hry
- Game Description Language (GDL) vyvinut za tímto účelem
 - ▣ Stav hry jako databáze
 - ▣ Pravidla pro legálnost, aktualizace stavů, atd.

Reprezentace znalostí

- GDL je založen na jazyku Datalog
- Obvykle prefixová notace
- Proměnné začínají symbolem ?
- Rozlišuje 9 relací:

role

does

goal

init

next

terminal

true

legal

distinct

Reprezentace znalostí

□ GDL – příklad: Tic-Tac-Toe

▣ Konstanty:

`xplayer, oplayer` ... hráči

`x, o, b` ... značky

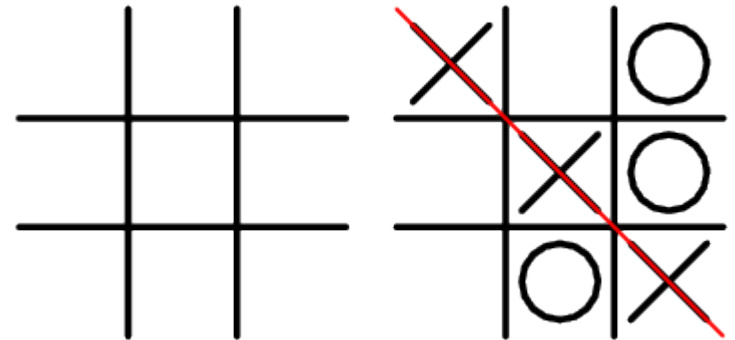
`noop` ... tah

▣ Funkce

`cell(number, number, mark)` ... buňka

`control(player)` ... hráč na tahu

`mark(number, number)` ... tah



Reprezentace znalostí

□ GDL – příklad: Tic-Tac-Toe

□ Hráči (*role*)

`(role xplayer)`

`(role oplayer)`

□ Stav hry (*true*)

■ Situace na obrázku, na tahu je hráč x.

`(true (control xplayer))`

`(true (cell 1 1 x))`

`(true (cell 1 2 b))`

`(true (cell 1 3 b))`

`(true (cell 2 1 b))`

`(true (cell 2 2 o))`

`(true (cell 2 3 b))`

...

X		
	O	

Reprezentace znalostí

□ GDL – příklad: Tic-Tac-Toe

□ Počáteční stav (`init`)

```
(init (cell 1 1 b))
```

...

```
(init (cell 3 3 b))
```

```
(init (control xplayer))
```

□ Aktualizace stavu hry (`next`)

■ Např. střídání hráče na tahu:

```
(<= (next (control xplayer))
```

```
    (true (control oplayer)))
```

```
(<= (next (control oplayer))
```

```
    (true (control xplayer)))
```

Reprezentace znalostí

□ GDL – příklad: Tic-Tac-Toe

▣ Přípustné tahy (`legal`)

- Hráč `?player` může hrát na pozici `[?x, ?y]`, pokud je na tahu a daná pozice zatím není obsazena

```
(<= (legal ?player (mark ?x ?y))
     (true (cell ?x ?y b))
     (true (control ?player)))
(<= (legal xplayer noop)
     (true (control oplayer)))
(<= (legal oplayer noop)
     (true (control xplayer)))
```

Reprezentace znalostí

□ GDL – příklad: Tic-Tac-Toe

□ Tahy (does)

- Označuje tah zahráný hráčem v určitém kroku
- V pravidlech popisujících změnu stavu hry
- Pokud hráč `?player` zahraje na `[?x, ?y]`, v příštím tahu bude tato buňka označena symbolem hráče

```
(<= (next (cell ?x ?y x))  
     (does xplayer (mark ?x ?y)))
```

- Buňky, na které hráč nezahrál, zůstávají nezměněny

```
(<= (next (cell ?x ?y b))  
     (does ?player (mark ?m ?n))  
     (true (cell ?x ?y b))  
     (distinctCell ?x ?y ?m ?n))
```

```
(<= (distinctCell ?x ?y ?m ?n)  
     (distinct ?x ?m))
```

```
(<= (distinctCell ?x ?y ?m ?n)  
     (distinct ?y ?n))
```

Reprezentace znalostí

□ GDL – příklad: Tic-Tac-Toe

▣ Cílové ohodnocení (`goal`)

- Cíle hráčů mohou být opačné, asymetrické, i doplňkové
- Určují ohodnocení jednotlivých hráčů

```
(<= (goal ?player 100)  
      (line ?player))
```

▣ Koncové stavy (`terminal`)

■ Nulární relace

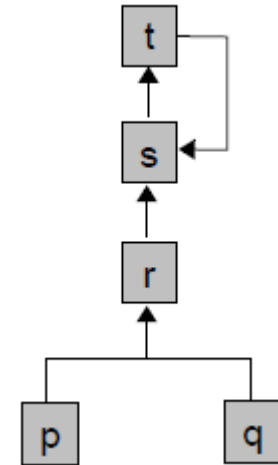
```
(<= terminal  
      (role ?player)  
      (line ?player))
```

```
(<= terminal  
      (not open))
```

Reprezentace znalostí

- Závislostní graf
 - ▣ uzly jsou relace v hlavách a tělech klauzulí
 - ▣ Hrana vede z p do q, pokud p se vyskytuje v těle a q v hlavě klauzule
- GDL restrikce

Relace	Výskyt	závislost
role	Pouze v hlavě pravidla bez těla	
init	Hlava klauzule	Nezávisí na <code>true</code> , <code>legal</code> , <code>does</code> , <code>next</code> , <code>terminal</code> , <code>goal</code>
true	Tělo klauzule	
does	Tělo klauzule	<code>legal</code> , <code>terminal</code> ani <code>goal</code> nezávisí na <code>does</code>
next	Hlava klauzule	



$$\begin{aligned}
 r(X, Y) &\leq p(X, Y) \wedge q(X, Y) \\
 s(X, Y) &\leq r(X, Y) \\
 s(X, Z) &\leq r(X, Y) \wedge t(Y, Z) \\
 t(X, Z) &\leq s(X, Y) \wedge s(Y, X)
 \end{aligned}$$

Reprezentace znalostí

- Ukončení: nekonečná sekvence tahů dosáhne koncový stav v konečném počtu kroků
- Hratelnost: Každá role má v neterminálním stavu aspoň 1 přípustný tah
- Monotonie: každá role má právě jedno ohodnocení v dosažitelném stavu, ohodnocení neklesá
- Vyhratelnost: pro každou roli existuje posloupnost tahů vedoucí do koncového stavu s největším ohodnocením dané role



**Dobře
zformovaná
hra**

Hráči (General Game Players)

- Máme nástroj pro reprezentaci stavu hry
- Můžeme využít tradiční prohledávací techniky (minimax, alfa-beta,...)
- Nabízí se doménově nezávislé heuristiky (history, transpoziční tabulky...)
- Problém: jakou zvolit ohodnocovací funkci?
- Vytvoření co nejlepšího hráče je předmětem GGP Competition

Hráči (General Game Players)

□ Heuristika

- Např. čím více možných tahů, tím lépe
- V obecném případě je těžké najít úspěšnou heuristiku
- Cluneplayer (vítěz 2005)
- Goblin
- Fluxplayer (vítěz 2006)

□ Simulace (Monte Carlo)

- Cadiaplayer (vítěz 2007, 2008)
- Ary (vítěz 2009, 2010)
- HyperPlay (též pro hry s neúplnou informací)

Hráči (General Game Players)

- Cluneplayer
 - ▣ Snaha vytvořit heuristickou ohodnocovací funkci
 - ▣ Využití klíčových aspektů hry
 - ▣ Hledání důležitých výrazů, snaha o jejich interpretaci

Hráči (General Game Players)

- Goblin
 - 2. místo na GGP Competition 2005
 - Alfa-beta
 - Iterativní prohlubování
 - Transpoziční tabulky
 - Paranoid Adversarial Search
 - Feature Extraction

Hráči (General Game Players)

□ Goblin

▣ Paranoid Adversarial Search

- Minimax předpokládá 2 hráče
- GGP umožňuje libovolný počet hráčů
- Očekáváme, že se proti nám všichni spikli
- Generují se všechny možné kombinace tahů oponentů
- Varianta minimaxu proti „velkému nepříteli“

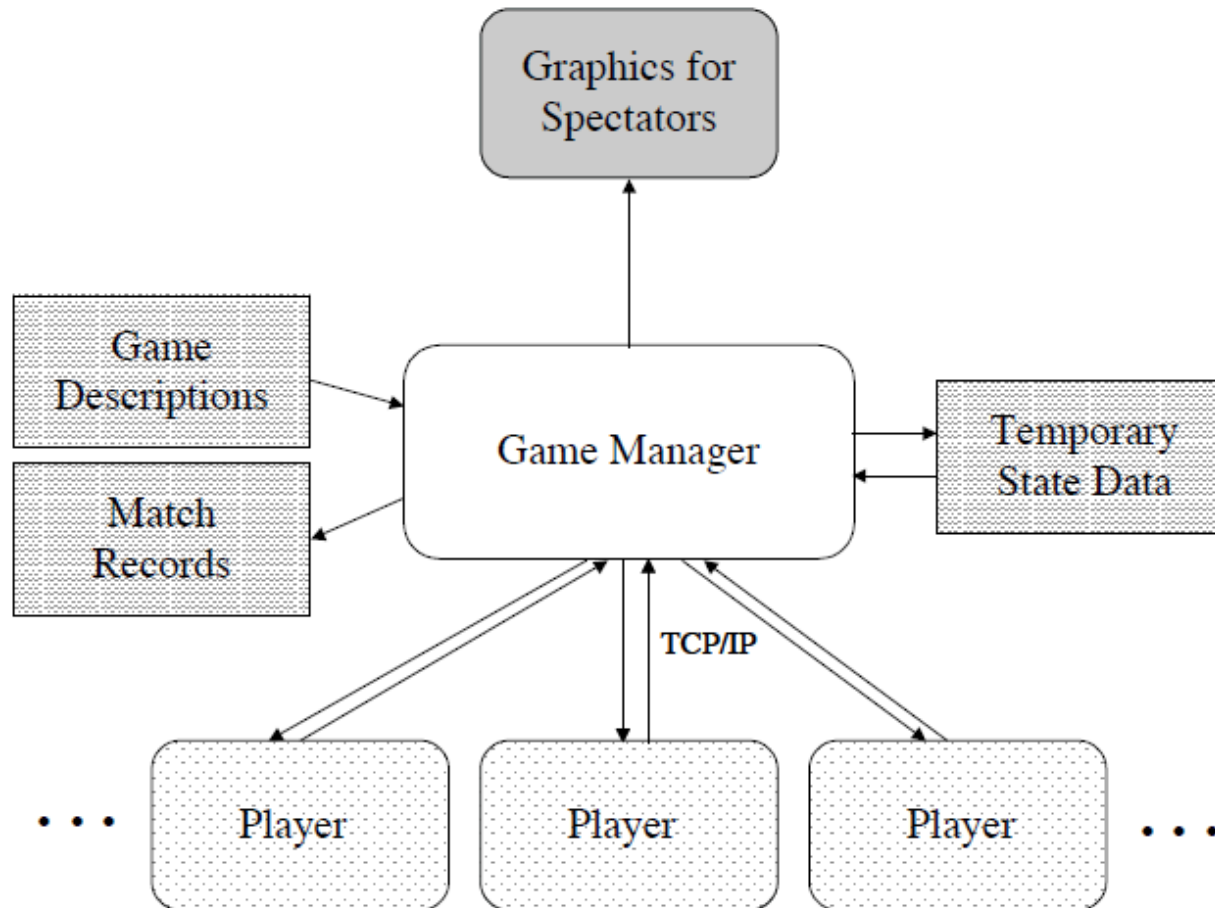
Hráči (General Game Players)

- Cadiaplayer
 - ▣ Pro hry jednoho hráče používá variantu IDA*
 - ▣ UCT simulace pro hry dvou a více hráčů (kooperativní i kompetitivní)
 - ▣ Absence nutnosti doménově specifických znalostí je vhodná v GGP
 - ▣ Překládá GDL do Prologu, samotné prohledávání v C++

Řídící systém

- Arcade
 - ▣ Databáze informací o hrách, hráčích a zápasech
- Game editor
- **Game manager**
 - ▣ Odpovědný za průběh jednotlivých her a zápasů
 - ▣ Komunikace s hráči prostřednictvím HTTP
 - ▣ Distribuce axiomů
 - ▣ Udržování stavu hry a jeho aktualizace
 - ▣ Kontrola přípustnosti tahů
 - ▣ Určení vítězů
 - ▣ Časová synchronizace (startclock, playclock)

Řídící systém



Řídící systém

- ▣ Před začátkem zápasu musí být specifikováno
 - Hra známá pro systém
 - Počet hráčů
 - Startclock – doba před zahájením hry
 - Playclock – čas na tah
- ▣ Dále přebírá odpovědnost **Game Manager**
 - **Start:** (START <MATCHID> <ROLE> <DESCRIPTION>
<STARTCLOCK> <PLAYCLOCK>)
 - **Play:** (PLAY <MATCHID> (<A1> <A2> . . . <An>))
 - **Stop:** (STOP <MATCHID> (<A1> <A2> . . . <An>))

Řídící systém

- Příklad komunikace mezi GM a hráči při hře Tic-Tac-Toe

Game Manager Message	Game Player Response
(START MATCH.435 WHITE <i>description</i> 90 30)	READY
(PLAY MATCH.435 (NIL NIL))	(MARK 2 2)
(PLAY MATCH.435 ((MARK 2 2) NOOP))	NOOP
(PLAY MATCH.435 (NOOP (MARK 1 3))	(MARK 1 2)
(PLAY MATCH.435 ((MARK 1 2) NOOP))	NOOP
⋮	⋮
(STOP MATCH.435 ((MARK 3 3) NOOP))	DONE

Řídící systém

- <http://130.208.241.192/ggpserver/index.jsp>

Závěr

- Mladé téma
- Široké možnosti dalšího výzkumu
- Význam (zatím) spíše teoretický
- Užitečné v didaktice

Zdroje

- Genesereth, M., and Love, N. 2005. General game playing: Overview of the AAIL competition. *AI Magazine* 26(2)
- Genesereth, M., and Love, N. 2005. General game playing: Game description language specification. Technical report, Computer Science Department, Stanford University, Stanford, CA, USA. http://games.stanford.edu/gdl_spec.pdf)
- Jim Clune. Heuristic evaluation functions for general game playing. In Proceedings of the AAIL Conference on Artificial Intelligence, pages 1134{1139, 2007.
- Peter Kissmann and Stefan Edelkamp. Gamer, a general game playing agent. *KI Linstliche Intelligenz*, 25:49{52, 2011.