

Hry a herní strategie

Jindřich Matuška

Faculty of Informatics, Masaryk University

24. října 2024

Čas na odpovědníky

Obsah

MINIMAX graf, algoritmus

Jupyter lab

Alfa-beta prořezávání

Neterministické hry

Jupyter lab

Obsah

MINIMAX graf, algoritmus

Jupyter lab

Alfa-beta prořezávání

Neterministické hry

Jupyter lab

MINIMAX graf

Syntax

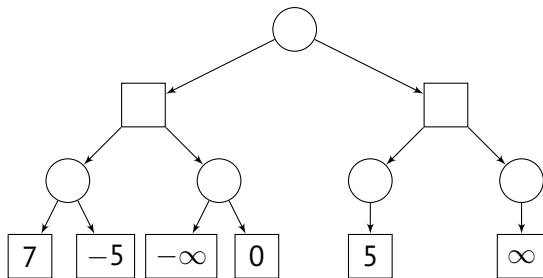
- orientovaný graf
- vnitřní vrcholy typu **MIN** nebo **MAX**
 - značíme kolečkem (MIN) či čtverečkem (MAX)
- koncové vrcholy s přiřazenou hodnotou z $\mathbb{R} \cup \{-\infty, \infty\}$

MINIMAX algoritmus

- Rekurzivní algoritmus
- Hodnota listů daná definicí stromu
- Hodnota vrcholu MIN je nejmenší hodnota z následníků
- Hodnota vrcholu MAX je největší hodnota z následníků
- Odpovídá hře s nulovým součtem 2 hráčů
- Hráči se střídají

$$\text{minimax}(X) = \begin{cases} \max_{n \in X \rightarrow} \text{minimax}(n) & \text{pokud } X \text{ je MAX uzel} \\ \min_{n \in X \rightarrow} \text{minimax}(n) & \text{pokud } X \text{ je MIN uzel} \end{cases}$$

Příklad



MINIMAX pro popis her

	X	
O	X	O
	X	O

-1

X	O	X
O	X	O
O	X	O

0

X	X	O
O	O	O
X	X	O

1

Příklad 4.1.1

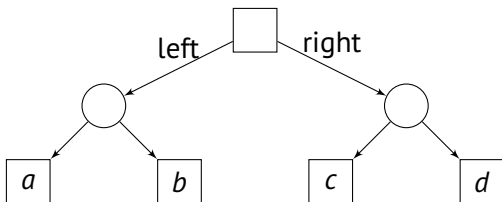
Rozhodněte, zda je následující hry možné modelovat MINIMAX stromy. Pokud ano, zamyslete se nad tím, jak by takové stromy vypadaly.

- a) šachy
- b) sudoku
- c) piškvorky na neomezené hrací ploše
- d) kámen, nůžky, papír
- e) tenis

Příklad 4.1.2

Doplňte nějaké konkrétní ohodnocení koncových stavů her s naznačeným MINIMAX stromem tak, aby začínající hráč dosáhl nejlepšího možného výsledku hraním zadaných strategií, předpokládáme-li, že jeho soupeř nedělá chyby. Jaké vztahy obecně musí v takovém případě platit pro hodnoty listů?

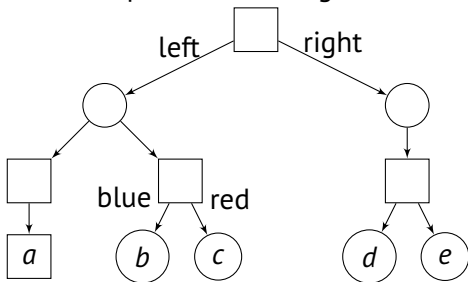
- a) Výherní strategie necht' je *left*



Příklad 4.1.2

Doplňte nějaké konkrétní ohodnocení koncových stavů her s naznačeným MINIMAX stromem tak, aby začínající hráč dosáhl nejlepšího možného výsledku hraním zadaných strategií, předpokládáme-li, že jeho soupeř nedělá chyby. Jaké vztahy obecně musí v takovém případě platit pro hodnoty listů?

- b) Výherní strategie nechť je volit *left* v prvním kole a *blue* ve druhém. Nezapomeňte zařídit, aby nás dokonalý soupeř s jistotou dovedl k volbě potřebné strategie.



Příklad 4.1.3

Dokažte, že MINIMAX stromy rozšiřují AND/OR stromy. Jinými slovy ukažte, že každý AND/OR strom lze chápat jako MINIMAX strom.

Obsah

MINIMAX graf, algoritmus

Jupyter lab

Alfa-beta prořezávání

Neterministické hry

Jupyter lab

Implementace MINIMAX algoritmu

Implementujte MINIMAX algoritmus na příkladu 3×3 piškvorek.

Obsah

MINIMAX graf, algoritmus

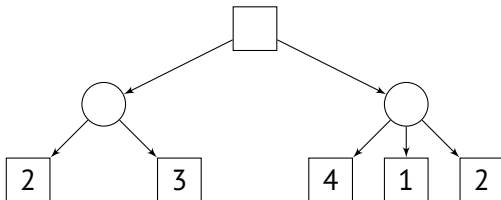
Jupyter lab

Alfa-beta prořezávání

Neterministické hry

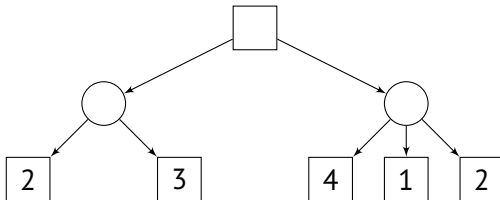
Jupyter lab

Příklad



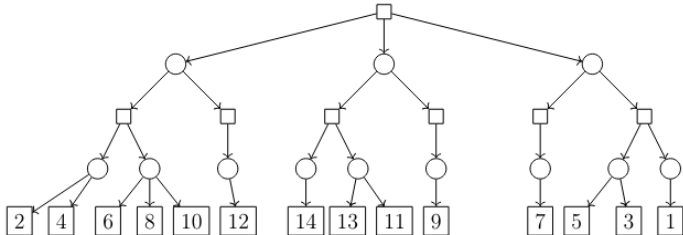
ALFA-BETA prořezávání

- Nepotřebujeme prohledávat všechny uzly
- Pamatujeme si parametry α (alespoň), β (nejvýše)
- V uzlu MAX aktualizujeme α , v MIN β
- Parametry posováme v DFS prohledávání



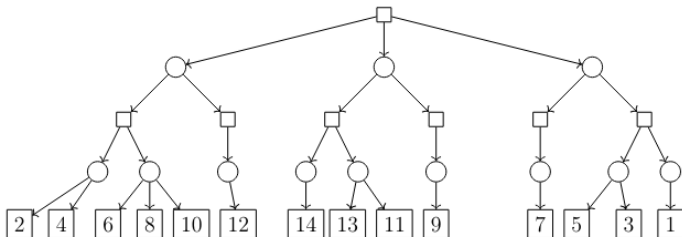
Příklad 4.2.2

Zadaný strom MINIMAX střídá po úrovních typy svých uzlů a každý jeho list je ve stejné hloubce. Takové v literatuře (a hlavně v praxi) budete vídat nejčastěji, neboť přirozeně vznikají při modelování mnoha zajímavých her. Jako v celém zbytku sbírky předpokládejte i zde, že algoritmus prochází vrcholy v pořadí zleva doprava.



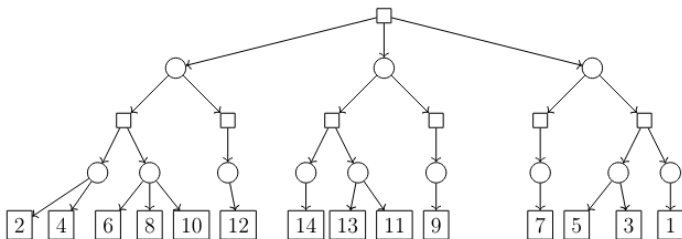
Příklad 4.2.2

- a) Vyřešte strom s pomocí algoritmu ALFA-BETA. Především tedy zjistěte výslednou hodnotu kořene a které podstromy budou uřezány, tj. nebudou vůbec navštíveny. Při výpočtu věnujte zvláštní pozornost tomu, abyste porozuměli, jakým způsobem α odpovídá dolní hranici a β naopak horní.



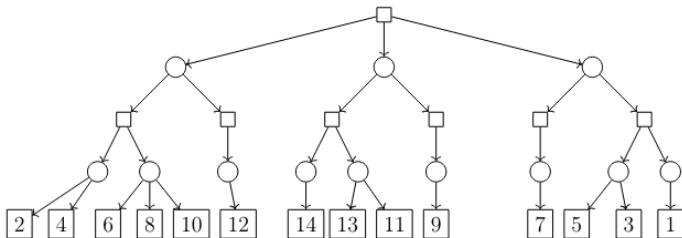
Příklad 4.2.2

- b) Při zachování struktury grafu navrhnete vhodné hodnoty listů tak, aby nedošlo k žádnému prořezávání (a algoritmus tedy navštívil všechny listy).



Příklad 4.2.2

- c) Při zachování struktury grafu navrhnete vhodné hodnoty listů tak, aby došlo k největšímu možnému prořezávání (a algoritmus tedy navštívil co nejméně uzlů).



Příklad 4.2.3

Při kterých z následujících transformací MINIMAX stromu může dojít ke změně nalezené optimální strategie?

- a) Ke všem hodnotám listů přičteme stejnou reálnou konstantu c .
- b) Všechny hodnoty v listech vynásobíme stejnou konstantou c .
- c) Hodnoty ve všech listech se libovolně změní tak, aby mezi nimi zůstalo zachováno jejich původní uspořádání.
- d) Všechny uzly typu MIN změníme na MAX a naopak. Hodnoty v listech pronásobíme -1 .

Obsah

MINIMAX graf, algoritmus

Jupyter lab

Alfa-beta prořezávání

Neterministické hry

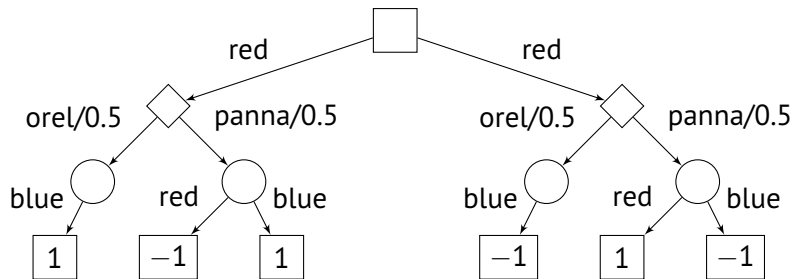
Jupyter lab

Nedeterminismus

- nový typ vrcholu RAND
 - značíme kosočtvercem
 - hrany z něj vycházející jsou ohodnoceny pravděpodobností
- Hodnota RAND vrcholu je váženým součtem

$$minimax(X) = \begin{cases} \max_{n \in X \rightarrow} minimax(n) & \text{pokud } X \text{ je MAX uzel} \\ \min_{n \in X \rightarrow} minimax(n) & \text{pokud } X \text{ je MIN uzel} \\ \sum_{n \in X \rightarrow} minimax(n) \cdot P(n) & \text{pokud } X \text{ je RAND uzel} \end{cases}$$

Příklad



Obsah

MINIMAX graf, algoritmus

Jupyter lab

Alfa-beta prořezávání

Neterministické hry

Jupyter lab

Implementace ALFA-BETA prořezávání

Implementujte ALFA-BETA prořezávání na příkladu 3×3 piškvorek. Můžete s výhodou využít již vytvořený MINIMAX algoritmus