

Základy umělé inteligence

Automatické řešení úloh

Formalizace úlohy

- UI chápe řešení úloh jako proces hledání řešení v podstatě metodou zkoušek a omylů.
- Je-li dán počáteční stav úlohy s_0 a požadovaný koncový (cílový) stav úlohy s_c , je úkolem systémů UI hledat vhodnou posloupnost akcí, jejichž aplikací lze přejít od počátečního stavu k cílovému. Tato posloupnost se nazývá **plán**.

Formální označení

- s_0 ... počáteční stav
- s_1, s_2, s_3 , atd. ... jednotlivé stavy, kterými úloha v průběhu řešení prochází
- s_c ... koncový (cílový) stav – může jich být několik
- $\varphi_1, \varphi_2, \varphi_3$, atd. ... jednotlivé operátory, které reprezentují přechod úlohy mezi jednotlivými stavy
- η ... plán úlohy, tj. posloupnost operátorů, které převedou úlohu z počátečního stavu s_0 do cílového stavu s_c
- Příklad: Například plán $\eta = \{\varphi_2, \varphi_4, \varphi_1\}$ znamená, že úlohu lze z počátečního stavu do cílového převést postupným provedením operací $\varphi_2, \varphi_4, \varphi_1$.

Znázornění průběhu řešení úlohy

- Re prezentaci (průběh řešení) řešené úlohy lze znázornit různým způsobem.
- Nejčastější způsob je znázornění **orientovaným stromem** s uzly, které představují jednotlivé stavy s , a orientovanými hranami, které představují přechody mezi stavy φ .



Znázornění průběhu řešení úlohy

- U stromu lze pro každý uzel definovat tzv. **hloubku uzlu**:
 - počáteční uzel (nazývaný též **kořen**) má hloubku 0;
 - jestliže má konkrétní uzel určitou hloubku, má každý jeho bezprostřední následník hloubku o jednu větší (**bezprostřední následník** uzlu s je takový uzel, do kterého se lze z uzlu s dostat použitím jednoho operátoru).



Princip řešení

- Při automatickém řešení úloh je třeba pro nalezení řešení použít vhodnou **prohledávací strategii**.
- V úlohách UI jsou využívány dva základní typy prohledávacích strategií:
 - metoda zpětného navracení (backtracking)
 - metoda prohledávání stromů
 - prohledávání do šířky
 - prohledávání do hloubky
 - prohledávání s minimální cenou
 - prohledávání s heuristikami

Metoda zpětného navracení (backtracking) – vlastnosti

- Pamatuje si vždy jen jedinou cestu od počátečního k poslednímu vygenerovanému stavu.
- Její výhodou je, že je snadno implementovatelná na počítači a má malé nároky na paměť počítače.
- Nevýhodou je, že se musí znovu procházet a rozvíjet stavy, ve kterých již algoritmus byl, ale zapomněl je.

Metoda zpětného navracení (backtracking) – princip

- Na stav, ve kterém se úloha právě nachází, postupně zkouším použít jednotlivé operátory.
- Jestliže nějaký operátor použít lze, použiji ho a na stav, který tak vznikne, se opět pokouším aplikovat jednotlivé operátory.
- Tento proces se opakuje tak dlouho, dokud nově vzniklý stav není cílový, nebo dokud nenastala „chyba“.
- Jestliže při řešení nastala „chyba“, právě zkoumaný stav se zruší a algoritmus se vrátí zpět k bezprostředně předcházejícímu stavu (pokud existuje).

Metoda zpětného navracení (backtracking) – princip

- „Chyba“ nastane tehdy, jestliže
 - nově vzniklý stav se již vyskytuje v zapamatované cestě řešení, nebo
 - nebyl nalezen vhodný operátor, nebo
 - byla dosažena maximální hloubka prohledávání bez nalezení řešení.



Metoda zpětného navracení (backtracking) – s omezenou hloubkou

- Nevýhodou backtrackingu s neomezenou hloubkou prohledávání je možnost „vjetí“ algoritmu ve stromu řešení na „nekonečnou“ cestu, která nemusí obsahovat řešení.
- Modifikací obecné metody zpětného navracení je proto nastavení maximální přípustné hloubky prohledávání.



Metody prohledávání stromů

- Pamatují si celou postupně generovanou strukturu stromu řešení.
- V důsledku toho se znovu negenerují stavy, s nimiž již algoritmus pracoval, avšak za tu cenu, že algoritmus má velké paměťové nároky.
- Princip metod spočívá v tom, že na stav, ve kterém se úloha právě nachází, se najednou aplikují všechny použitelné operátory (provádí se tzv. **expanze uzlu**)



Metody prohledávání stromů

- Podle toho, jak se vybírá stav pro expanzi, dělíme metody prohledávání stromů na:
 - prohledávání do šířky
 - prohledávání do hloubky
 - prohledávání s minimální cenou
 - prohledávání s heuristikami

Prohledávání do šířky – princip

- 1) Prove se expanze počátečního stavu, tj. na počáteční stav se v jedné chvíli aplikují všechny operátory, a najdou se všichni následníci tohoto stavu.
- 2) Najde se stav s nejmenší hloubkou, který zatím nebyl expandován, a prove se expanze tohoto stavu. Při expanzi se okamžitě zruší ty stavy, které se již kdekoli ve stromu řešení vyskytují. Tento krok se opakuje tak dlouho, dokud mezi vygenerovanými stavy není cílový stav.

Prohledávání do šířky – vlastnosti

- Algoritmus prohledávání do šířky najde vždy nejkratší cestu z počátečního do cílového stavu.