Jak souvisí hledání nejkratší cesty s UI?

Terminologie

- Stav = vrchol: Úplný popis jedné konfigurace
- Akce = hrana: Atomická změna konfigurace
- Cena akce = váha (délka) hrany
- Počáteční stav = počáteční vrchol
- Cíl = množina koncových vrcholů
- Stavový prostor = množina vrcholů
- Transition model = funkce (stav,akce) $\rightarrow$ stav

Proč měnit terminologii?

Příklady hledání cest v UI

- Loydova patnáctka
- Sokoban
- Rubikova kostka
- Další hlavolamy

Prohledávání grafu

Input: Graf G, počáteční vrchol s a cílový t

Všechny vrcholy označ za nenavštívené
Počáteční vrchol označ za navštívený
while *existuje navštívený vrchol a cílový vrchol není prozkoumaný* **do**
    Zvol u libovolný navštívený vrchol
    for v *soused* u **do**
        if v *je nenavštívený* **then**
            Označ v za navštívený
    Označ u za prozkoumaný

Poznámky

- Jestliže s a t leží ve stejné komponentě, tak skončíme prozkoumáním t, jinak projdeme celou komponentu obsahující s
- Průchod do hloubky: vybíráme poslední navštívený vrchol
- Průchod do šířky: vybíráme první navštívený vrchol
- Existuje řada variant: více počátečních i koncových vrcholů, nalezení cest do všech vrcholů, . . .

Dijkstrův algoritmus

Input: Graf G s nezápornou délkou hran c, počáteční vrchol s a cílový t

Všechny vrcholy označ za nenavštívené
Počáteční vrchol označ za navštívený
Délka nejkratší zatím nalezené cesty z s do u je $d[u] := \infty$ kromě $d[s] := 0$
while *existuje navštívený vrchol a cílový vrchol není prozkoumaný* **do**
    $u :=$ navštívený vrchol s nejmenší hodnotou $d[u]$
    for v *soused* u **do**
        if $d[v] > d[u] + c(u, v)$ **then**
            $d[v] := d[u] + c(u, v)$
            Označ v za navštívený
    Označ u za prozkoumaný

Poznámky

- Pro prozkoumané vrcholy u je $d[u]$ délka nejkratší cesty z s to u
- Algoritmus označuje vrcholy za prozkoumané v neklesající vzdálenosti od počátku
- Prozkoumány jsou všechny vrcholy ve vzdálenosti menší než je vzdálenost do cíle
- Graf může být příliš velký, takže d si pamatujeme jen pro navštívené vrcholy
- Která města navštívíme při hledání cesty z Prahy do Brna?
- Vizualizace: `https://qiao.github.io/PathFinding.js/visual/`

A* algoritmus

Input: Graf G s nezápornou délkou hran c, počáteční vrchol s

Všechny vrcholy označ za nenavštívené
Počáteční vrchol označ za navštívený
Délka nejkratší zatím nalezené cesty z s do u je $d[u] := \infty$ kromě $d[s] := 0$
while *existuje navštívený vrchol a libovolný cíl není prozkoumaný* **do**
    $u :=$ navštívený vrchol s nejmenší hodnotou $d[u] + h(u)$
    for *v soused u* **do**
        if $d[v] > d[u] + c(u, v)$ **then**
            $d[v] := d[u] + c(u, v)$
            Označ v za navštívený
    Označ u za prozkoumaný

Poznámky

- Heuristická funkce $h(u)$ dává odhad vzdálenosti z u nejbližšího cíle
- Pokud $h(u) = 0$ pro všechny vrcholy, pak se A* chová stejně jako Dijkstra
- Heuristiku musíme rychle spočítat, ideálně v $O(1)$, ale přesnou délku nejkratší cesty obvykle nedokážeme rychle určit

Vymyslete heuristiku pro Loydovu 15

Definice

Heuristika h je

- přípustná (admissible), jestliže $0 \leq h(u) \leq c^*(u)$
- monotónní (monotonous, consistent), jestliže $0 \leq h(u) \leq h(v) + c(u, v)$

pro všechny vrcholy u a hrany uv a cílové vrcholy t, kde $c^*(u)$ je délka nejkratší cesty z u do nejbližšího cíle.

Cvičení

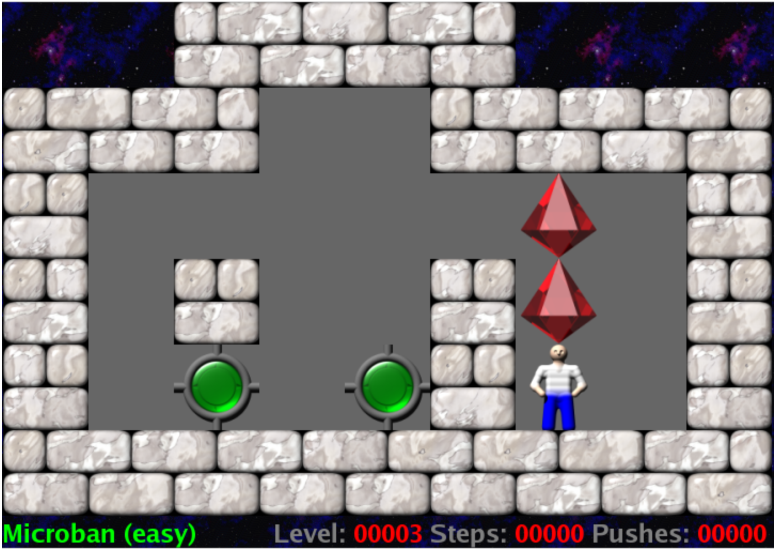
Rozhodněte, zda pro silniční síť jsou následující heuristiky přípustné a monotónní.

- Euklidovská vzdálenost: $h_2(a, b) = \sqrt{(b_1 - a_1)^2 + (b_2 - a_2)^2}$
- Manhattanská metrika: $h_1(a, b) = |b_1 - a_1| + |b_2 - a_2|$
- Maximová metrika: $h_\infty(a, b) = \max\{|b_1 - a_1|, |b_2 - a_2|\}$

Otázky

- Je každá monotónní heuristika přípustná?
- Je každá přípustná heuristika monotónní?
- Proč potřebujeme, aby heuristika byla monotónní?

Vymyslete heuristiku pro Sokoban



- Animovaná verze
- Přehled postupů

Vlastnosti A*

Značení

- $h(u)$: heuristika z u do cíle
- $g(u)$: vzdálenost ze startu do cíle
- $f(u) = h(u) + g(u)$
- $d[u]$: proměnná v A* udávající délku nejkratší nalezené cesty

Pozorování

Předpokládejme, že máme monotónní heuristiku.

- Hodnoty $f(u)$ jsou neklesající na všech nejkratších cestách ze startu.
- A* prozkoumává (uzavírá) stavy v pořadí, ve kterém hodnoty $f(u)$ neklesají. Základní verze A* neurčuje pořadí prozkoumávání stavů se stejnou hodnotou $f(u)$.
- Při prozkoumávání stavu u je hodnota $d[u]$ rovna $g(u)$.
- A* vždy najde optimální plán (nejkratší cestu).
- A* prozkoumá všechny vrcholu u splňující $f(u) < C^*$ a některé vrcholy s $f(u) = C^*$, kde C^* je délka nejkratší cesty.

Vlastnosti heuristik

Pozorování

Dokažte pro přípustné/monotónní heuristiky h_1 a h_2 jsou též

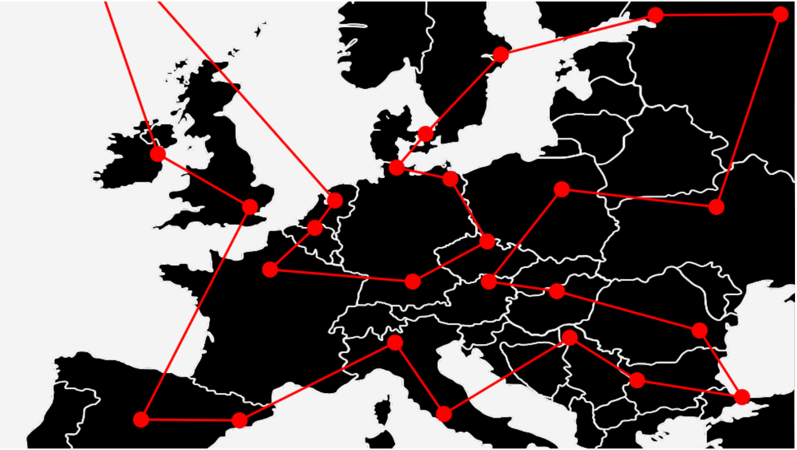
- $\alpha h_1 + (1 - \alpha) h_2$, kde $0 \leq \alpha \leq 1$
- $\max \{h_1, h_2\}$

přípustné/monotónní heuristiky.

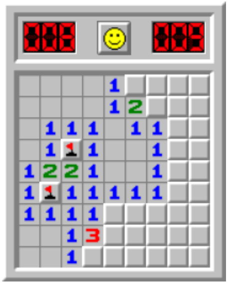
Otázka

Která z výše uvedených kombinací je lepší?

Vymyslete heuristiku pro Obchodního cestujícího

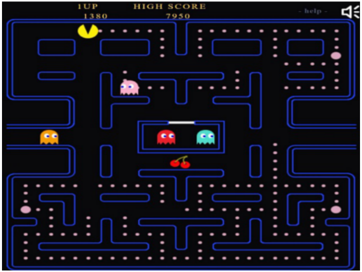


Dokážete vymyslet heuristiky pro tyto hry?



Minesweeper

Šachy



Packman

Kulečník

1. domácí úkol: A* heuristiky

Zadání (zkráceno)

Implementujte **monotónní** heuristiky pro **A*** algoritmus spuštěný na **podgrafy** následujících nekonečných mřížek.

- Klasická dvourozměrná mřížka
- Klasická třírozměrná mřížka
- Dvourozměrná mřížka obsahující i úhlopříčky
- Třírozměrná mřížka obsahující stěnové i prostorové úhlopříčky
- Třírozměrná mřížka obsahující stěnové úhlopříčky ale nikoliv prostorové
- Hrany odpovídají právě pohybům věže po šachovnici
- Skokan se pohybuje o 3 políčka v jedné souřadnici a o 2 políčka v druhé souřadnici
- Král v sedmimílových botách, který může až o 8 políček horizontálně i vertikálně

Zadání: `https://gitlab.mff.cuni.cz/finkj1am/introai/-/blob/master/01-a_star_heuristic/task.md`

Rady

- Úkolem je najít heuristiku, nikoliv přesnou vzdálenost
- Zkuste vymyslet dolní odhad na minimální počet kroků
- Vycházejte z heuristik diskutovaných na cvičení, uzpůsobte je danému grafu a kombinujte je
- Na většinu mřížek stačí 1-2 řádková heuristika, v jednom případě zhruba 5 řádků
- Jestliže celé číslo je větší než 5.5, pak je větší nebo rovno 6
- Nepište nic bez přemýšlení