



Fibot 2006

RossumsUniversalRobot

Prezentace strategie

Kdo stojí za RUR?

- Bc. Miroslav Grabovský (poslední ročník N-AP)
 - konzultace Java
- Mgr. Pavel Moravec (druhý ročník D-IN)
 - výběr/návrh vhodných algoritmů, analýza testů
- Bc. Jaroslav Novotný (poslední ročník N-IN)
 - návrh a implementace strategie
- Tomáš Obšíváč (poslední ročník B-AP)
 - testování, správa verzí

Základní principy naší strategie

■ Rozhodování

- **deterministické** / nedeterministické
- **jednokolové** / s ohledem na historii

■ Plánování

- **jeden krok** / více kroků dopředu

■ Povaha robota

- zbabělý / **opatrný** / utočný / sebevrah

Základní podstrategie

■ Brát poklady

- pro každý poklad vyhodnoť, jaký zisk nám vynese, když si pro něj půjdeme a vezmeme jej

■ Střílet po soupeřích

- pro každého soupeře vyhodnoť, jaký zisk bodů nám vynese, když ho půjdeme střelit

■ Skončit s co největším počtem energie

- vyhodnoť odpočívání a cestu k nejbližší baterii, jak se projeví do energie, která nám na konci zůstane

Nejkratší cesty

- Naivní algoritmus (len)
 - počítá s tím že máme nekonečně energie
- S odpočíváním (len_R)
 - vychází z počáteční energie a při klesnutí energie pod 2 provádí odpočinek
- S odpočíváním a dobíjením (len_{RB})
 - jako předchozí, ale snaží se najít kratší cestu s využitím jedné baterie

Brát poklady

- pro každý poklad vyhodnot' koeficient „zisk na kolo“

$$\text{zisk}_{poklad} = \frac{\text{cena}(poklad)}{\text{len}_{RB}(ja, poklad) + 1}$$

- znevýhodni pro každého soupeře, který může dojít k pokladu dříve než my

$$\text{zisk}_{poklad} = \text{zisk}_{poklad} * \sqrt{\frac{\text{len}_R(souper, poklad)}{\text{len}_{RB}(ja, poklad)}}$$

- tento postup je dále rozšířen na všechny uspořádané k -tice pokladů (pro turnaj zvoleno $k = 4$)
- plus mnoho dalších korekcí a optimalizací

Střílet po soupeřích

- pomocné algorimy
 - jak dlouho vydrží nepřítel na ose x (y)
 - jak rychle se dostanu na nepřítelovu osu x (y)
 - kolik bude mít nepřítel energie za k kol
- pro každého nepřítele vypočtu „zisk na kolo“, který obdržím, pokud ho půjdu střílet na jeho osu x (y)
- znevýhodni varianty, kdy může nepřítel utéct

Skončit s co nejvyšší energií

- Kolik bodů mi přinese, když budu až do konce provádět jen odpočívání
 - znevýhodnit, pokud mě někdo může do konce hry zasáhnout
- Kolik bodů mi přinese, když teď půjdu na baterku a v posledím kole provedu dobíjení
 - znevýhodnit, pokud můžu provést i později

Daší problémy

- Uvážnutí
 - předvídání kolizí
 - detekce kolizí (moudřejší ustoupí)
- 50^2-4 pokladů

Statistika nuda je...

- Naše implementace obsahuje
 - 10 (+2) tříd
 - 45 (+10) metod
 - 1659 (+412) řádků
 - 58,1 kB (+17 kB) zdrojového kódu
- Zlepšení oproti verzi z testovacího turnaje
 - získá na soupeře v průměru o 20 bodů na zápas víc, než předchozí verze