

BAKALÁŘSKÁ PRÁCE

Jméno Příjmení

Název práce

Název katedry nebo ústavu

Vedoucí bakalářské práce: Vedoucí práce

Studijní program: studijní program

Studijní obor: studijní obor

Praha ROK

Prohlašuji, že jsem tuto bakalářskou práci vypracoval(a) samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů. Tato práce nebyla využita k získání jiného nebo stejného titulu.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle §60 odst. 1 autorského zákona.

V dne

Podpis autora

Poděkování.

Název práce: Název práce

Autor: Jméno Příjmení

Katedra: Název katedry nebo ústavu

Vedoucí bakalářské práce: Vedoucí práce, katedra

Abstrakt: Abstrakt.

Klíčová slova: klíčová slova

Title: Name of thesis

Author: Jméno Příjmení

Department: Name of the department

Supervisor: Vedoucí práce, department

Abstract: Abstract.

Keywords: key words

Obsah

Úvod	2
1 Navržená hra - Asteroidy	3
1.1 Herní logika	3
1.2 Cíl hry	3
2 Architektura hry	5
2.1 Vesmírné objekty	5
2.1.1 Asteroidy	5
2.1.2 Střely	5
2.1.3 Vesmírná loď	5
2.2 Prostředí	6
2.2.1 Stav prostředí	8
2.3 Agent	8
2.4 Grafické prostředí	8
3 Akční plán	9
3.1 Motivace	9
3.2 Jednotliv plány	9
3.2.1 Útok	9
3.2.2 Sestřelující obrana	9
3.2.3 Úhybná obrana	9
3.2.4 Zastavení letu	9
4 Genetické programování	10
4.1 Základní princip	10
4.2 Využití	10
4.3 Aplikace	10
5 Hluboké Q-učení	12
5.1 Základní princip	12
5.2 Využití	12
5.3 Aplikace	12
6 NEAT	13
6.1 Základní princip	13
6.2 Využití	13
6.3 Aplikace	13
Seznam obrázků	14

Úvod

Počítačové hry mají kromě zábavného prožitku ze samotného hraní další funkci. Herní prostředí často tvoří samostatný, uzavřený svět se svými vlastními zákonitostmi. Takovéto prostředí je pro nás vhodné pro tvoření umělé inteligence. Agent ve světě dané hry "vidí", v jakém stavu se nachází, má na výběr konečný počet akcí, které může provést a cíl, kterého chce dosáhnout.

1. Navržená hra - Asteroidy

1.1 Herní logika

Jedná se o hru dvou hráčů. Každý z hráčů ovládá svou vesmírnou loď. Prostředí hry má představovat vesmírný prostor, je to ale prostor zjednodušený, proto zde neplatí gravitační ani odporové síly. To má tedy za následek, že když se vesmírná loď rozletí v nějakém směru, tak v tomto směru letí i nadále i bez dalšího akcelarování.

Vesmírný prostor je v této hře jistým způsobem nekonečný, pokud vesmírná loď proletí dolní hranicí herního prostoru, tak nezmizí, ani nenabourá, ale objeví se na stejné pozici jen na horní hranici a obráceně. Analogicky to platí i s bočními hranicemi. Vesmírné lodě sebou mohou proletět a nedojde ke srážce. Nejsou zde žádné statické překážky, kterým by bylo třeba se vyhnout. Co se ale může srazit s vesmírnou lodí jsou asteroidy.

Asteroidy vznikají v průběhu hry na náhodných místech a letí náhodným směrem. Nové asteroidy se generují častěji, čím déle hra trvá. V boji s asteroidy má hráč v zásadě dvě možnosti. Buď se může pokusit danému asteroidu vyhnout, tím že s lodí pohne mimo trajektorii asteroidu, anebo může asteroid sestřelit. Každý hráč má omezený počet životů a každá srážka lodě s asteroidem ubere hráči část jeho životů.

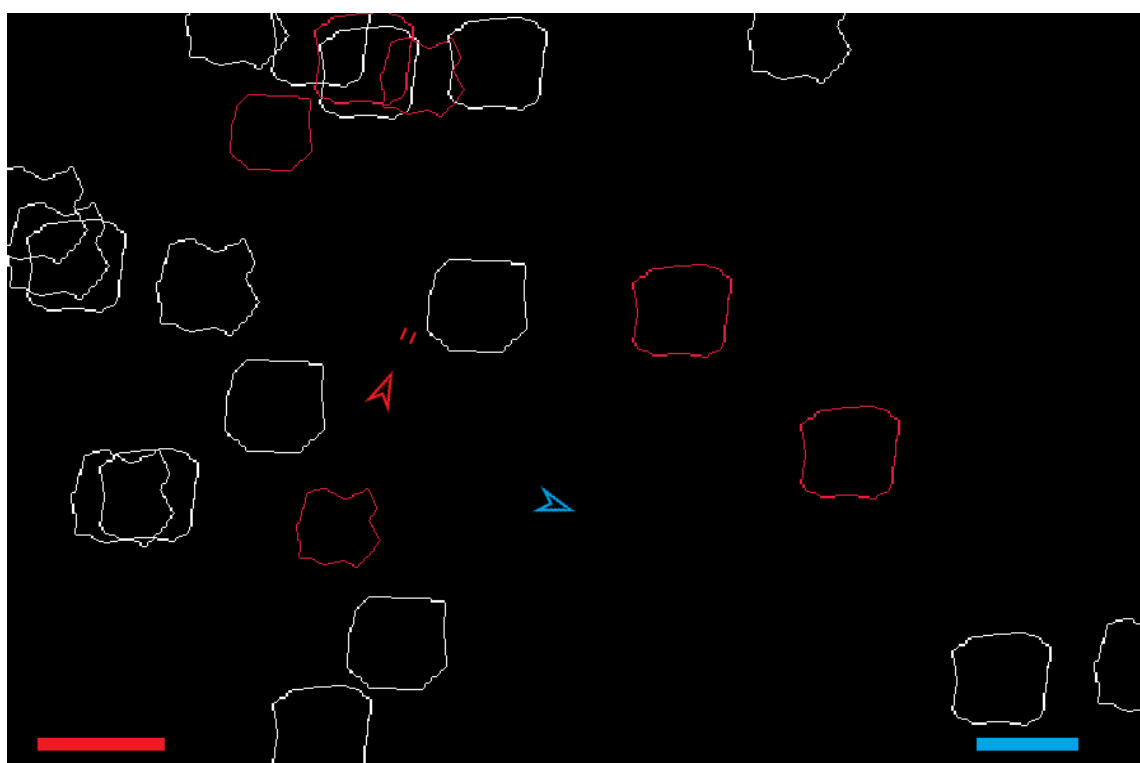
Hráč má k dispozici dva typy střel, obyčejnou a rozdvojovací. Vystřelená střela má značně vyšší rychlost než vesmírné lodě i než kolem letící asteroidy. Střely nejsou určeny k přímému zasažení lodě protihráče, vesmírné lodě jsou k nepřátelké střele imunní. Střely jsou určeny k sestřelování letících asteroidů. Asteroidy mohou mít tři velikosti. Náhodně vytvořený asteroid je vždy největší. Každým rozstřelením daného asteroidu vznikají asteroidy o stupeň menší velikosti. Nově vytvořené asteroidy vznikají na místě původního asteroidu. Asteroid se může rozstřelit různými způsoby. Zde záleží na tom, jakou střelou byl asteroid sestřelen.

V případě střely obyčejné vznikne namísto původního asteroidu jeden menší, který letí stejným směrem jako střela, která ho zasáhla. V případě střely rozdvojovací se původní asteroid rozstřelí na dva menší, kde každý z nich je oproti směru střely vychýlen o 15° po a proti směru hodinových ručiček. Pokud je zasažen asteroid nejmenší velikosti, tak již žádné další asteroidy nevznikají. Rychlost asteroidů je nepřímo závislá na jejich velikosti, čím je asteroid menší, tím vyšší rychlost má.

Asteroidy vzniklé rozstřelením se stávají projektily daného hráče. Hráč nemůže být zasažen asteroidem, který sám vytvořil

1.2 Cíl hry

Během hry vzniká postupně více a více asteroidů, čímž je postupně stále obtížnější se všem asteroidům vyhnout, nebo je sestřelit. Hráč nemůže zranit nepřítele střelou přímo, může se ale snažit rozstřelit nějaký z kolem letících asteroidů tak, aby pomocí nově vzniklých asteroidů trefil nepřítele. Cílem hráče je ovládat svou loď takovým způsobem, aby vydržel ve hře déle. Hra končí a hráč vítězí, když nepříteli nezbydou žádné další životy.



Obrázek 1.1: Screenshot ze hry

2. Architektura hry

2.1 Vesmírné objekty

Všechny vesmírné objekty mají některá data společná. Každý vesmírný objekt má souřadnice své současné polohy a také vektor rychlosti.

2.1.1 Asteroidy

Asteroidy nesou navíc informace o tom, jaké jsou velikosti a zda-li byly vytvořené nějakým z hráčů. Na základě těchto dvou informací je asteroidu při vytvoření přiřazen obrázek, pomocí kterého je po dobu své existence vykreslován.

2.1.2 Střely

Vystřelené střely neletí věčně, ale mají omezenou životnost kolik kroků hry budou existovat. Tato hodnota se nastavuje z konfiguračního souboru z položky *BULLET_LIFE_COUNT*. V každém kroku hry se střele její životnost sníží o jedna a pokud se dostane na nulu, tak střela bude zničena. Střele se při vytvoření nastaví úhel, pod kterým poletí. Tento úhel je roven úhlu natočení vesmírné lodi, který měla při vystřelení. Samozřejmě také u střely musíme evidovat, kterému z hráčů patří, toto je řešeno odkazem na objekt vesmírné lodi, která střelu vystřelila. Jak již bylo zmíněno v předchozí kapitole, střely jsou dvojího druhu. Příznakem *split* se určuje zda se jedná o střelu obyčnou nebo rozdělovací

2.1.3 Vesmírná loď

Vesmírná loď má základní polohové informace rozšířené o úhel. Ten se s každou rotací lodě zvětší nebo zmenší o 12° . Akcelerace funguje vektorovým sčítáním. K současnému vektoru rychlosti se přičte vektor odpovídající současnému úhlu lodi. Maximální rychlost vesmírné lodi je omezená, v případě že akcelerací vznikne vektor rychlosti, jehož délka je větší než hodnota maximální rychlosti, se směr vektoru zachová, ale požadovaně se zkrátí.

2.2 Prostředí

Hra běží v cyklu diskretních kroků, které dohromady simulují plynulý pohyb hry. Herní prostředí je inspirováno projektem `open ai gym` od google (viz <https://gym.openai.com/>). Jedním rozdílem je však přístup k vykreslování hry. V případě `gym.openai` se prostředí vykresluje zavoláním metody `render()` na instanci prostředí zvenku. Já jsem zvolil přístup jiný. V případě, že chceme hru graficky zobrazovat, předáváme v konstruktoru prostředí grafický modul, který implementuje vykreslování jednotlivých typů vesmírných objektů. A prostředí už poté objekty graficky vyresluje interně samo. Rozhodnutí, že se má grafický modul volitelně injektovat v konstruktoru a nemá být natvrdo svázen s prostředím, jsem učinil pro větší nezávislost modulů. Později se při práci s různými knihovnami pro evoluční algoritmy ukázalo být problematické, že bylo herní prostředí svázáno s grafickým modulem.

Herní prostředí se stará o manipulaci všech vesmírných objektů a akcí s nimi spojenými. V každém kroku dostává od hráčů akce, které chtějí provést, a prostředí na to odpovídajícím způsobem reaguje. Akce každého hráče z hráčů jsou pole, které obsahuje elementární možné akce:

- Rotace vlevo
- Rotace vpravo
- Akcelerace
- Obyčejná střela
- Rozdvojovací střela

Hráč může provádět více akcí najednou. Na základě přítomných elementárních akcí se provádí dané reakce. Prostředí se stará o vesmírné objekty přímo. V případě elementárních akcí, které mění rychlost nebo orientaci vesmírné lodi, prostředí zavolá funkce, které požadované změny na vesmírné lodi provede. A v případě elementárních akcí střel se na základě polohy a orientace dané vesmírné lodi vytvoří nová střela, kterou opět bude spravovat právě prostředí.

Jedna instance prostředí odpovídá jedné hře. Herní prostředí má dvě základní metody pro řízení hry.

Metoda `reset()` inicializuje hru do počátečního stavu a tento stav vrátí. Tato metoda se musí zavolat před začátkem hry.

A druhá metoda `next_step(actions_one, actions_two)`, která na základě akcí hráčů, převede hru do následného stavu. V právě této metodě je schovaná celá logika manipulace s vesmírnými objekty.

```

def next_step(self, actions_one, actions_two):
    self.step_count = self.step_count + 1
    self.reward_one = 0
    self.reward_two = 0

    self.handle_actions(actions_one, actions_two)
    self.generate_asteroid()
    self.check_collisions()
    self.move_objects()
    if self.draw_modul is not None:
        self.render()

    (game_over, player_one_won) = self.check_end()
    if not game_over:
        self.reward_one += 1
        self.reward_two += 1

    current_state = State(self.asteroids_neutral,
                           self.rocket_one,
                           self.asteroids_one,
                           self.bullets_one,
                           self.rocket_two,
                           self.asteroids_two,
                           self.bullets_two)

    return self.step_count, \
           (game_over, player_one_won), \
           current_state, \
           (self.reward_one, self.reward_two)

```

2.2.1 Stav prostředí

Úplná informace o všech objektech v prostředí.

2.3 Agent

Agent se na základě informace o současném stavu prostředí rozhodne o své následné akci.

2.4 Grafické prostředí

Mezi nejvíce citované statistické články patří práce Kaplana a Meiera a Coxe (??). ? napsal článek o t-testu.
projektu ACCEPT jsou uvedeny v práci ?.

3. Akční plán

3.1 Motivace

Abstrakce nad akcemi nízké úrovně

3.2 Jednotliv plány

3.2.1 Útok

3.2.2 Sestřelující obrana

3.2.3 Úhybná obrana

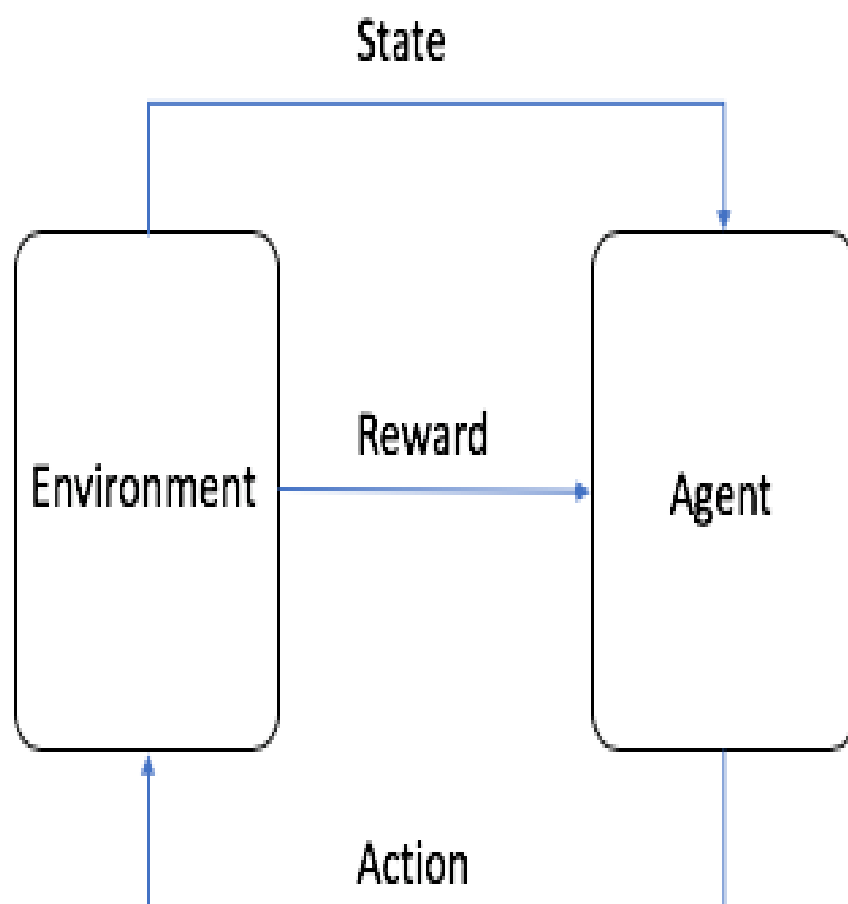
3.2.4 Zastavení letu

4. Genetické programování

4.1 Základní princip

4.2 Využití

4.3 Aplikace



Obrázek 4.1: Herní cyklus

5. Hluboké Q-učení

5.1 Základní princip

5.2 Využití

Kde se používá v praxi.

5.3 Aplikace

Jak jsem to použil já a jakých výsledků jsem dosáhl.

6. NEAT

6.1 Základní princip

6.2 Využití

Kde se používá v praxi.

6.3 Aplikace

Jak jsem to použil já a jakých výsledků jsem dosáhl.

Seznam obrázků

1.1	Screenshot ze hry	4
4.1	Herní cyklus	11