

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA
DEPARTMENT OF COMPUTER SYSTEMS

INTELIGENTNÍ SYSTÉM PRO ÚČAST V SOUTĚŽI
SSCAI
INTELLIGENT SYSTEM FOR THE SSCAI TOURNAMENT

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

VÁCLAV HORÁZNÝ

VEDOUCÍ PRÁCE
SUPERVISOR

PAVEL SMRŽ, doc. RNDr., Ph.D.

BRNO 2015

Abstrakt

V této práci se zabývám vytvořením umělé inteligence tak, aby splňovala podmínky pro uvedení do soutěže SSCAI Tournament. Obsahuje charakteristiku a herní mechaniky hry Starcraft: Brood War, popis pravidel zmíněné soutěže a běžně hrané druhy strategií se zaměřením na civilizaci Zerg a to konkrétně strategie *Pool Rush*. V soutěži jsem se umístil na sdíleném 10. a 11. místě ze 42 zúčastněných. Umělou inteligenci jsem vytvořil pro hru Starcraft: Brood War (1.16.1). Práce obsahuje implementaci dvou agentů `ScoutManager` a `DefenseManager`. K těmto agentům jsou připojeni již vytvoření agenti z projektu BWSAL. K vytvoření jsem využil vývojového prostředí Microsoft Visual Studio 2008 (9.0). Pro spuštění umělé inteligence je nutné využít programu Chaoslauncher a využít knihoven BWAPI a BWTA. Práce obsahuje porovnání mnou vytvořeného programu s ostatními zúčastněnými, jak z pohledu způsobu implementace, tak i podle konkrétně volených strategií.

Abstract

In this work I focus on the creation of artificial intelligence to meet the conditions for the competition SSCAI Tournament. It contains the characteristics and gameplay mechanics of the game Starcraft: Brood War, a description of the rules of that competition, usually played kinds of strategies focusing on the Zerg civilization, namely strategy *Pool Rush*. In the competition I was on a shared 10th and 11th place out of 42 participants. I have created the artificial intelligence for the game Starcraft: Brood War (1.16.1). The work includes the implementation of two agents `ScoutManager` and `DefenseManager`. These agents are already connected to created agents of the project BWSAL. I used the development environment Microsoft Visual Studio 2008 (9.0). To launch artificial intelligence it is required to use the program Chaoslauncher and to use libraries BWAPI and BWTA. Work includes comparing my created program with other stakeholders, both in terms of the method of implementation, as well as by specifically chosen strategies.

Klíčová slova

Umělá inteligence, StarCraft, SSCAI Tournament, Rush

Keywords

Artificial intelligence, StarCraft, SSCAI Tournament, Rush

Citace

Václav Horázný: Inteligentní systém pro účast v soutěži SSCAI, bakalářská práce, Brno, FIT VUT v Brně, 2015

Inteligentní systém pro účast v soutěži SSCAI

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Pavla Smrže, doc. RNDr.,Ph.D.

Další informace mi poskytl Ing. Ondřej Soukup.

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Poděkování

Tímto bych chtěl poděkovat svému vedoucímu práce panu Smrži, doc. RNDr.,Ph.D. za udělení obrovské volnosti v práci a panu Ing. Soukupovi za odbornou pomoc při vypracování práce.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

Obsah.....	1
1 Úvod.....	3
2 Starcraft.....	5
2.1 Charakteristika.....	5
2.2 Strategie.....	5
2.2.1 Průzkum.....	6
2.2.2 Rushování.....	7
2.2.3 Partyzánství.....	7
2.2.4 Turtling.....	8
2.2.5 Budovy.....	8
2.2.6 Expanze.....	9
2.2.7 Zásobování.....	9
2.3 Civilizace Zerg.....	10
2.3.1 Průzkum.....	10
2.3.2 Základní mechaniky hry pro Zergy.....	10
2.3.3 Rushovací strategie.....	11
3 Implementace.....	12
3.1 Cíle inteligentních prvků.....	12
3.1.1 Plánování.....	12
3.1.2 Učení.....	12
3.1.3 Neurčitost.....	13
3.1.4 Prostorové a časové uvažování.....	13
3.1.5 Tvorba modelu oponentovi strategie.....	14
3.1.6 Rozložení problému.....	14
3.2 Vývojové prostředí.....	15
3.2.1 The Brood War Application Programming Interface (BWAPI).....	16
3.2.2 The Brood War Terrain Analyzer (BWTA).....	16
3.2.3 The BWAPI Standard Add-on Library (BWSAL).....	16
3.3 Scout Manager.....	17
3.4 Defense Manager.....	18
3.5 ModuleAI.....	19
3.5.1 Počáteční strategie stavební řady.....	19
4 Zhodnocení a výsledky.....	21
4.1 Soutěž SSCAI Tournament.....	21

4.2 Ročník 2014.....	21
4.3 Porovnání počátečních strategií.....	21
5 Závěr a budoucí práce.....	24
Literatura.....	25
Seznam příloh.....	26

1 Úvod

Pojem Umělá inteligence [17] se objevil poprvé v USA v roce 1955 na konferenci v Dartmouthu. Navrhl jej používat John McCarthy, americký informatik, mimo jiné tvůrce jazyka LISP (* 4. září 1927, Boston). Rozdělení umělé inteligence navrhl následovně:

Hraní her – například naučit hrát stroje šachy.

Expertní systémy – rozhodovací a strategické programy ve vojenství, lékařství, na finančních burzách a ve velkých firmách.

Rozpoznávání hlasu – nejen ve smyslu umět hlas přepsat do textu, ale rozpoznat smysl zprávy a umět na ni alespoň do určité míry reagovat.

Neuronové sítě – především simulace činnosti biologických neuronových sítí matematickými algoritmy a jejich využití v praxi.

Robotika – inteligentní stroje, resp. stroje které dokáží reagovat na svoje okolí.

Definice profesora Marvina Minského, amerického vědce zabývajícího se umělou inteligencí:

„Umělá inteligence je věda o vytváření strojů využívajících postupy a činnosti prováděné člověkem, které by vyžadovaly jeho inteligenci.“

Umělá inteligence je tedy obor, který se snaží řešit problémy podobně jako by je řešil člověk, za využití složitých rozhodovacích stavů, optimalizačních výpočtů a heuristických řešení a dosahovat tak co největší shody s nejlepším řešením člověka (experta) bez vlivu chyby lidského faktoru. Využití je možné v široké škále oborů lidské činnosti. Dnes především v expertních systémech, kde umělé inteligence v podstatě nahrazují experta na řešení problémů a napomáhají člověku nalézt správnou cestu k vyřešení konkrétního problému. Například to mohou být navigační systémy letadel, lodí a řešení dopravy vůbec, kosmické řídicí a navigační systémy, řídicí systémy složitých výrobních procesů, manažerské systémy řízení apod. I v oblasti neuronových sítí je celá škála možností. Nabízí se využití ve výrobě a provozu protetických pomůcek handicapovaných, například robotické protézy nebo náhrady zraku, sluchu i hmatu. Umělá inteligence prochází časovými změnami. V průběhu doby se rozdělila do zhruba třiceti disciplín.

Nejznámějších jsou:

- reprezentace znalostí a modelování, znalostní databáze
- rozpoznávání a učení
- plánování a rozhodování
- neuronové sítě
- analýza, syntéza a rozpoznávání mluvené řeči
- zpracování přirozeného jazyka

- počítačové vidění
- znalostní inženýrství
- programovací prostředky umělé inteligence
- multiagentní systémy
- aplikovaná umělá inteligence a expertní systém
- strategie her, herní systémy

V mojí práci jsem se zaměřil na oblast hraní strategických her, a to konkrétně na vytvoření inteligentního systému pro hru Starcraft: Brood War (1.16.1). Po jejím vytvoření jsem se s touto prací přihlásil do soutěže SSCAI Tournament, kde jsem díky možnosti analyzovat chyby, které program obsahoval, docílil standardní strategie *5 Pool Rush*. První část práce obsahuje uvedení do herních mechanik a charakteristik hry. V příloze 2 jsou vysvětleny běžně používané termíny ve hře: například názvy herních strategií, názvy jednotek, budov a výzkumů, které jsou značeny v textu bakalářské práce *kurzívou*. Práce pojednává a popisuje přesněji jednotlivé obecné taktiky jako je *rush*, partyzánství, průzkum a podobně. V kapitole civilizace Zerg se zaměřuji právě na tuto civilizaci a popis jejích strategií, mechanik a možností hry za tuto civilizaci. Po vyhodnocení soutěže jsem se umístil na sdíleném 10. a 11. místě ze 42 zúčastněných. Celý projekt je tvořený ve vývojovém prostředí Microsoft Visual Studio 2008 (9.0) v jazyce C++. Konkrétně jsem implementoval ScoutManager, který se stará o průzkum mapy a přehled o oponentovi, a DefenseManager, který se stará o veškeré bojové situace ve hře. Tyto dva agenty jsem propojil s již vytvořeným systémem agentů BWSAL, kteří pro propojení s hrou využívají knihoven BWAPI a BWTA. V konečné fázi tvorby herní strategie jsem se snažil o co nejrychlejší a zároveň ekonomicky výhodnou rushovací strategii, což se podle výsledků, kterých jsem dosáhl v soutěži podařilo. Nakonec jsem zpracoval porovnání mojí implementace s ostatními účastníky soutěže, konkrétně podle způsobu implementace, volby strategie a změn chování na základě událostí ve hře.

2 Starcraft

2.1 Charakteristika

Starcraft je real-time strategická (RTS) počítačová hra [10, 11] vyvinutá společností Blizzard Entertainment v roce 1999 [1]. Zařadila se mezi nejprodávanější hry v historii. Hráč si může vybrat ze tří různých civilizací, za které lze hrát. Jsou to Terran, Zerg a Protoss. Tyto civilizace se liší vzhledem, původem a typem bojových jednotek a staveb, rozdílnou taktikou boje a strategií ekonomiky [2]. Ve hře je možné těžit dva typy surovin: minerály a plyn, které jsou nutné pro budování jednotek, budov a jejich vylepšování. Minerály je možné těžit pomocí dělníků – *SCV* za Terrany, *Drone* za Zergy a *Probe* za Protossy, zatímco pro těžbu plynu je nutné postavit budovu na jeho zřídle – *Rafinery* za Terrany, *Extractor* za Zergy a *Assimilator* za Protossy. Pak už do těžebny stačí posílat dělníky, kteří obě suroviny nosí do hlavní budovy. Hráči mohou mít jen omezené množství jednotek s výjimkou Zergských jednotek *Overlord*, což jsou v podstatě létající budovy. Každá jednotka má určité požadavky na zásobování, to úzce souvisí se silou jednotky. Hráč tak musí udržovat dostatek podpůrných jednotek, u Terranů pomocí *Supply Depotu*, u Protossů pomocí *Pylonů* a u Zergů pomocí *Overlordů*. Ve hře je také určen maximální limit pro množství zásob (200), který nelze překročit. To hráče nutí útočit a ne jen hromadit armádu a stále čekat. Pro vytváření bojových jednotek a jejich vylepšování je nutné stavět řadu rozličných budov. U mnohých jednotek je nutné postavit konkrétní budovu, provést výzkum či budovu vylepšit, aby bylo možné si je pořídit.

2.2 Strategie

Základem každé strategie je zajištění dostatečného množství surovin pro vytvoření jednotek. Strategie se tak liší v tom, jaké jednotky, v jakém množství a jak rychle je hráč vyrábí před prvním útokem. Některé strategie jsou silně defenzivní, to znamená, že se zaměřují na obranu a postupné naplnění celého limitu jednotek, a až poté vysílají první útok. Opačným extrémem je takzvaná rushovací strategie, kdy jde o to, v co nejkratším čase překvapit oponenta. U většiny těchto strategií je však problém s obranou nebo s expanzí a pokrokem základny a vylepšováním jednotek, proto první útok musí oponenta dostatečně poškodit, jinak je hra v podstatě prohraná a jen s malou pravděpodobností je možné prohru zvrátit. Následující strategie jsou odvozovány od těchto dvou strategií a jsou jedné nebo druhé blíže. Dalším prvkem strategie je například pohyb jednotek po bojišti (mikro kontrola), kdy se jednotky pohybují po bojišti tak, aby utřily co nejmenší poškození, ale zároveň maximalizovaly poškození oponentovi. Pro takovéto řešení se nejčastěji používá

algoritmus alfa-beta. Pro algoritmus α - β je důležité, jaké budovy a jednotky mají být preferovány. Velkou výhodou je mít i přehled o pohybech oponenta a jeho strategii, a přizpůsobovat tak svou strategii. Čím dříve je detekován oponentův útok, tím kvalitnější je i reakce na něj. Důležitou součástí je také pokrok ve vylepšování jednotek výzkumy. Je podstatné vylepšovat bojeschopnost jednotek, které jsou právě vyráběny, nebo už vyrobeny jsou, výjimečně vylepšovat jednotky v předstihu, pokud vím, že jejich vylepšení bude užito. V některých herních mapách úspěch útoku zásadně ovlivňuje jednak rychlost útoku, jeho průběh, ale i rozmístění budov na základně, popřípadě vystavení budovy umožňující výrobu jednotek, co nejbližší oponentově základně. Další možnosti, jak podpořit a urychlit vítězství v útoku, je vystavení obranných budov přímo v bojišti, a tak výrazně podpořit útok svůj (tuto strategii využívá zejména Terran se stavbou *Bunker*). V každé počáteční pozici je jen omezené množství surovin, a tak je v rámci strategie také nutné odhadnout chvíli, kdy je nepostradatelná expanze do vzdálenějších pozic pro těžbu. Čím více je dělníků, tím více a rychleji jsou suroviny získávány. K výrobě dělníků je však potřeba čas a určitá dávka surovin, proto je vhodné najít rovnováhu mezi dostatečným přísunem materiálu a počtem dělníků, aby nedocházelo k zbytečnému nadbytku.

2.2.1 Průzkum

Průzkum je nedílnou a fakticky taktéž nejvýznamnější součástí každé strategie. Znalost akcí, které oponent uskutečňuje, je nutná pro efektivní a správné reakce na jeho strategii. Ale co to vlastně znamená průzkum? Kdy, a jak ho provádět? Předem bychom mohli říci, že náhodné rozesílání jednotek po mapě je neefektivní. Nejvýznamnějším cílem průzkumu v začátku hry je lokalizování pozice oponenta. Po tomto kroku je důležité zůstat co nejdéle v oponentově základně, pro zjištění pořadí výstavby jeho budov. V další části hry je průzkum oponentovy základny důležitý hlavně pro sledování výstavby a vylepšování budov k zdokonalování jednotek. Následujícím důležitým cílem průzkumu je zjištění umístění a složení oponentovy armády. Protože armáda je mobilní, v pokročilejší fázi hry je mnohem obtížnější ubránit všechny těžební pozice na celé mapě. Při přehledu o umístění oponentovy armády je pak jednodušší se připravit na obranu a nebo využít některých nedostatků v jeho obraně a podniknout útok. Posledním cílem průzkumu, a také jedním z nejdůležitějších, je zjištění výstavby nových těžebních základen oponenta a včasné zabránění jejich vzniku. Vzhledem k tomu, že tyto základny jsou nejzranitelnější při jejich výstavbě, je nutné je odhalit v co možná nejkratším čase od počátku jejich stavby. V pozdější části je počet těžebních základen největší výhodou ve hře. Po přihlédnutí ke všem těmto aspektům je zřejmé, že průzkum je důležité provádět nejen na počátku hry, ale i v jejím průběhu.

2.2.2 Rushování

Rushování, nebo-li rychlý útok, se považuje za značně neúčinnou taktiku proti zkušeným hráčům. Je to hlavně z toho důvodu, že hráčova ekonomika je oslabena pro další průběh hry, což v konečném důsledku končí vítězstvím oponenta, pokud rush není úspěšný. Útok nemusí být zcela zničující pro oponenta, ale může ho pouze oslabit, například zničením několika dělníků těžících suroviny, popřípadě jejich zdržením od práce. Nejlepší volbou pro útok je ve většině případů obejít oponentovu obranu a zaútočit nejdříve na jeho dělníky, a tak mu omezit přísun zdrojů. Existují ale případy, kdy se více vyplatí zničit nejdříve obranu, a to hlavně ve chvíli, kdy oponent obranu teprve vytváří nebo je oslabená. Vhodné je také s několika jednotkami se jen pohybovat po oponentově základně a odvádět tak pozornost dělníků od těžby a obranných jednotek od efektivního protiútoků. Pro dosažení úspěchu však musí být rush proveden v co možná v nejkratším čase, a to buď slabými jednotkami na úkor rozvoje technologií a ekonomiky, nebo jednotkami silnými, ale v podstatně v nižším počtu opět na úkor ekonomiky a technologického rozvoje. V tomto případě je také nevýhodou dlouhý časový interval výroby silnějších jednotek, při kterém zůstává základna bez obrany proti případnému oponentovu rushi. Proto je také základním kamenem této strategie průzkum oponentovi základny – jeho budov a množství jednotek. Pro vlastní obranu je důležité mít alespoň nějakou základní obrannou budovu v místě těžby nebo pár základních jednotek. Při detekci útoku je nutné přerušit výrobní posloupnosti a suroviny investovat do výroby jednotek na obranu. Pokud je rush ubráněn, je jedním z nejlepších rozhodnutí udělat rychlý protiútok se zbylými jednotkami. Hodnotit sílu rushe se dá pomocí 3 základních aspektů: rychlosti výroby jednotek, počtu jednotek a vospělosti jednotek. Například 6 jednotek *Zerling* v ranné části hry může mít stejně silný útok jako 3 jednotky *Hydralisk* v pozdější fázi hry.

2.2.3 Partyzánství

Partyzánství nebo-li partyzánská taktika je běžně využívána v průběhu celé hry. Je to spíše řada systematických akcí než přímo strategie. Do těchto taktik patří například multitasking, využívání děr obrany, výsadky, opakované malé útoky a podobně. Ve chvíli nevyrovnané síly armád jsou partyzánské taktiky neocenitelnou výhodou a pomocí nich lze vyhrát i značně nevyrovnané boje. V tomto případě jsou důležité tři body. Vyvarovat se otevřenému pozemnímu boji, získat a udržet terénní výhodu a koordinovat jednotky mezi základnami. Hlavní filozofií těchto taktik, jak výše zmiňuji, je vyhnout se otevřenému pozemnímu boji, proto je lepší využívat menších skupinek jednotek pro boj a útočit na oponenta na několika místech. Zároveň tak donutit jeho armádu k rozdělení a udržovat ho neustále v napětí a pod náporom malých útoků. Díky tomu oponent pomalu, ale jistě, začne ztrácet kontrolu nad územím, zdroji, velikostí armády a počtem budov.

Dalším důležitým aspektem partyzánství je využití terénu a kontroly nad mapou k svému prospěchu. Obrana strategických míst, surovin nebo blokování důležitých cest, to vše dopomáhá ke kontrole nad mapou. Zergská civilizace je pro partyzánské taktiky nejlepší, protože má po provedení příslušného výzkumu velice rychlé a levné pozemní jednotky. Má i výhodu v jednodušší obraně základen proti útočným stavbám, protože oponent nemůže stavět na *Creepu*, tj. na hmotě pokrývající okolí základen. Mohou také využít tunelů pro okamžitý přesun jednotek mezi jednotlivými základnami. Postupem času se partyzánská válka promění ve velkou bitvu na celém území mapy.

2.2.4 Turtling

Turtling nebo-li zaželvení je způsob využití terénu, obrany a speciálních jednotek tak, že se zabrání snadnému průchodu oponenta do životně důležitých částí základny, jako jsou pole nevytěžených surovin nebo technologické budovy. Jde vlastně o běžnou obrannou strategii. Tato strategie je výrazně obranná a není dobré se zcela zaměřovat pouze na ni. Pokud se krunýř obrany někde povede oponentovi prorazit, útok způsobí obrovské škody. Zároveň je složitější rozšiřovat základnu k dalším zdrojům surovin. Při tvoření obrany je nutno se zaměřit na využití terénu. Na vyvýšeném místě je obrana účinnější než v údolích, protože pozemní jednotky z údolí nevidí na vyvýšená místa, ale naopak ano. Dále je třeba při vytváření želvy nezapomenout na vyčerpatelnost surovinového pole a včas začít svoji základnu rozšiřovat. Protože některé z jednotek při obraně budou ztraceny, je nutné je neustále doplňovat, což ale stojí velké množství surovin. Pokud nelze dál těžit, nelze vyhrát hru. Jelikož účelem hry není vybudovat dokonalou základnu, ale porazit oponenta, je nutné kombinovat tuto strategii se strategiemi ostatními.

2.2.5 Budovy

Je mnoho strategií, jak umisťovat budovy. Budovy jsou poměrně levné a staví se krátkou dobu ve srovnání s budováním jednotek. Proto hráči většinou stavějí více, třeba i stejných budov. Důvodem stavby více stejných budov je vytvořit zálohu, aby nedošlo ke zpomalení nebo znemožnění rozvoje, pokud by byla některá z budov zničena. Rozmístění budov lze rovněž využít pro zefektivnění obrany základny. Ať už lepší průchodností uvnitř základny pro rychlejší přemísťování obranných jednotek, nebo uzavírání některých cest budovami a znemožnění průchodu jednotek oponenta. Zergové jako jediná civilizace nemohou stavět jinde než na *Creepu*, jedinou výjimkou je hlavní budova a *Extractor*. Proto je pro Zergy obtížnější se rozšířit do dalších pozic. Výhodnější, pro rozšíření *Creepu*, je stavět další hlavní budovy, aby se rozšířila možnost vyrábět více jednotek najednou, než stavět *Creep Colony*, která rozšiřuje pouze *Creep*.

2.2.6 Expanze

Počet základen se zdroji surovin je jedním z rozhodujících faktorů výsledku hry, proto většina strategií tkví právě v obsazení co nejvíce těžebních pozic a zároveň v znemožnění oponentovi obsazovat nové těžební pozice. Konkrétně u Zergů je důležité obsazovat ložiska plynu. Jednak mohou vystavět *Extractor*, i když poblíž ještě nemají základnu a *Creep*. A dále jsou více závislí na těžbě plynu než ostatní dvě civilizace. Při rushování se expanze odkládá tak dlouho, dokud není podniknut první rychlý útok. Po útoku už ale většinou není dostatek surovin pro okamžitou expanzi, a nebo byl útok natolik úspěšný, že není třeba se nijak rozšiřovat a je dobré pokračovat v útocích. Je mnoho faktorů, které ovlivňují čas první expanze základny. Některými z nich jsou: jakou strategii zvolil oponent, zda se rush vydařil, je-li dostatek surovin pro expanzi, velikost armády pro obranu obou základen, ale třeba také poloha nejbližšího vhodného místa pro rozšíření.

2.2.7 Zásobování

Patrně nejdůležitější částí každé strategie je získávání surovin pro výrobu jednotek, budov a vylepšování. Z počátku každé hry mají všechny civilizace 4 jednotky schopné těžít suroviny a stavět budovy, možnost zásobovat 9 jednotek a mají 50 minerálů, za které je možné pořídit jednu další těžební jednotku. Základem každé strategie je vytěžit co nejvíce surovin a zároveň maximálně znemožnit těžbu oponentovi. Proto je nutno útočit na oponentovi základny, které mají stále velké množství surovin k vytěžení, a ničit jeho těžební jednotky na základnách. Dále je vhodné vyřadit oponentovu těžbu v co možná nejrannější fázi hry, dokud má jen jednu nebo jen několik málo základen. Bude to tak pro něj podstatně větší rána do ekonomiky. Minerály jsou nejdůležitějším zdrojem surovin, a to především v počátku hry, kdy výstavba většiny budov a jednotek stojí nejvíce minerálů a plyn téměř žádný. Nejvíce závislími na těžbě minerálů jsou Protossovové. Naopak v pokročilejší části hry není výhodné útočit na Zergovskou těžbu minerálů, jelikož v té době jsou nejvíce závislí na plynu. Proto útoky na Zergy a jejich těžbu minerálů jsou v další části hry podstatně méně účinné. Při zničení těžební jednotky přichází oponent o suroviny, které výstavba stála a navíc také, které by jednotka vytěžila. Protože hráč nemůže jednotky obnovovat všechny zároveň, škoda se načítá podle vzorce 1.

$$\text{vzorec 1: } ztráta = RN + \frac{1}{2}GN(N + 1) = N\left(R + \frac{1}{2}G(N + 1)\right)$$

kde: R – náklad na vyrobení těžební jednotky

N – počet zničených těžebních jednotek

G – počet surovin co vytěží jednotka za dobu její výroby

Tedy při ztrátě 1 těžební jednotky je škoda 62 minerálů, ale při zničení 5-ti těžebních jednotek je ztráta již 430 minerálů, což je 86 minerálů na jednu zničenou jednotku a při zničení 9 je ztráta 110 minerálů na jednu jednotku. Tento výpočet však platí pouze, pokud oponent vyrábí jednotky pouze v jedné budově a nemůže vyrábět těžebních jednotek více najednou, což se v počáteční fázi hry stává jen zřídka. A později tyto taktiky nejsou účinné. Při útoku je důležité zničit těžbu co nejrychleji, aby nebyla možná reakce oponenta dříve, než se podaří zničit více jak 5 dělníků. Plně naložený výsadek dovede zničit v krátké chvíli 5 a více dělníků a jeho cena je okolo 200 až 400 minerálů včetně výsadkové lodě. Do výsadku není doporučeno naložit jednotky, které jsou určeny k útoku zblízka, z důvodu jejich vzájemného blokování při přecházení od cíle k cíli. Díky této taktice se dá získat cenný čas navíc pro rozvoj vlastních základen a armády.

2.3 Civilizace Zerg

2.3.1 Průzkum

Overlord jsou nejlepší průzkumné jednotky ve hře [3]. Jsou levné, mají hodně zdraví, jsou vzdušné a dokáží odhalit neviditelné jednotky. S rychlostním vylepšením jsou i velmi rychlé a s přepravním výzkumem mohou transportovat ostatní jednotky. Těchto jednotek má hráč též velké množství, protože fungují rovněž jako zdroj zásob pro udržení většího počtu jednotek. Na začátku hry jedním *Overlordem* hráč disponuje, a proto může začít průzkum okamžitě bez ztráty některé z těžebních jednotek (*Drone*). Po nalezení oponentovy základny může na místě zůstat hlídkovat, dokud oponent nepostaví obrannou jednotku, která dovede útočit na vzdušné síly. V pozdější fázi hry je téměř nezbytné provést výzkum pro zrychlení *Overlordů*. Toto vylepšení napomáhá průzkumu nových expanzí oponenta a zároveň umožní *Overlordům* stáhnout se z boje bez jejich ztrát. Na rozdíl od jiných civilizací Zergové mohou díky *Overlordům* prozkoumávat mapu neustále. K průzkumu se mohou využít i jednotky *Zergling*, které jsou nejlevnější jednotkou ve hře a zároveň se mohou s výzkumem *Burrow* zahrabávat a tak se stát nepohyblivými, ale zároveň neviditelnými. Vylepšení hlavní budovy je taktéž důležité udělat co nejdříve, protože díky němu se zpřístupní výzkumy pro vylepšení jednotek *Overlord*.

2.3.2 Základní mechaniky hry pro Zergy

Zergové mají možnost transformovat těžební jednotky do budov a to pouze na *Creep* hmotě, která se tvoří kolem budovy základny, *Creep Colony* a vývojových stádií těchto budov. Na hmotě *Creep* nemohou civilizace Terranů a Protossů stavět žádné budovy, mohou se po nich pouze přesouvat jednotkami. Po zničení budov, které *Creep* rozšiřují, *Creep* pomalu mizí a ostatní civilizace

získávají možnost zde stavět svoje budovy. Veškeré jednotky zergské civilizace se přeměňují z larev, které se generují kolem základních budov každých 15 sekund do počtu 3 na každou základní budovu. Pro udržování (zásobování) jednotek je nutné vyrábět jednotky *Overlord*, které maximum možných jednotek rozšiřují až do počtu 200 zásob. Civilizace Zergů se vyznačuje největší mobilitou jednotek ve hře. Jejich jednotky jsou levné, ale méně silné, takže je nutné oponenta ničit velkou početní převahou. Jedno ze zajímavých vylepšení všech pozemních jednotek je *Burrow*, které umožňuje jednotkám se zahrabat a tak se stát neviditelnými, ale bez možnosti útočit (výjimkou je *Lurker*). Díky jednotce *Overlord*, která má schopnost odhalovat neviditelné jednotky, mají Zergové nejlepší kontrolu nad mapou. Vedlejší, a ne tak podstatná mechanika ve hře, je pomalé samovolné obnovování zdraví budov a jednotek. Naopak nevýhodou může být, že oproti Terranům nemohou budovy opravovat nebo nemají štít, který se postupně dobíjí, jako u Protossů. Velkou mobilitu také dodává stavba *Nydus Canal*, která umožňuje okamžitý přesun z jednoho bodu do druhého [4].

2.3.3 Rushovací strategie

Civilizace Zergů je nejlépe uzpůsobena pro tyto strategie, proto existuje mnoho druhů těchto strategií. Jednou z nejznámějších, v mém projektu také využitou strategií, je *4 Pool Rush*. Je výjimečná právě tím, že se hned od začátku nic nebuduje, ale pouze se čeká na dostatečné množství surovin pro výstavbu *Spawning Pool*. Takže po dostavění *Spawning Pool* má hráč 3 těžební jednotky, které stíhají přesně produkovat minerály pro kontinuální produkci jednotek *Zergling*. Další podobné strategie jsou *5 Pool Rush* a *6 Pool Rush*, kde se analogicky vystaví určitý počet těžebních jednotek a poté budova *Spawning Pool*. Rozdíl je hlavně v nadbytku produkce minerálů a rychlosti prvních útoků, ale nejsou to nikterak velké rozdíly. Mezi pomalejší rushe patří například *Lurker Rush*, kde jde o co nejrychlejší zpřístupnění jednotek, které se dovedou zahrabat a tím zneviditelnit, ale zároveň mohou útočit. Nebo *Mutalisk Rush*, kde jde o co nejrychlejší výrobu armády, která je letecká. Obě dvě strategie jsou účinnější proti civilizaci Protossů, protože mají zpočátku slabou protileteckou obranu a omezenou možnost detekce neviditelných jednotek. Naopak strategie *Pool Rush* jsou více efektivní proti civilizaci Terranů.

3 Implementace

V této části práce se zaměřím přímo na mnou vytvořený program, jeho popis funkčnosti, použité knihovny a části přebrané z jiného projektu.

3.1 Cíle inteligentních prvků

Jak je zmíněno v práci [11], výzkum umělých inteligencí RTS her určil 6 základních cílů: plánování, učení, neurčitost, prostorové a časové uvažování, tvorba modelu oponentovy strategie, rozložení problému.

3.1.1 Plánování

Plánování, je proces určení sekvence herních akcí tak, aby bylo dosaženo předem stanoveného cíle. Obecně v hrách je hlavním cílem vyhrát. Jelikož složitost dosažení tohoto cíle je velká, je potřeba si stanovit menší cíle vedoucí k výhře.

Stavový prostor RTS her je velice rozsáhlý. Počet herních jednotek může být až několiknásobně větší než v běžných hrách jako jsou například šachy. V RTS hře je možné mít až stovky herních jednotek, které mají velké množství možných proveditelných akcí, na rozdíl od šachů, kde každá jednotka má jen několik možností pohybu. Herní pole mají šachy oproti RTS hrám poměrně malé - 64 možných pozic, zatímco v RTS hrách se jedná obvykle o tisíce až statisíce možných pozic. V RTS hře mohou veškeré herní jednotky provádět některou akci zároveň s ostatními a tím se stavový prostor zvětšuje exponenciálně. Komplikace tím nekončí, protože je nutné vše zpracovávat v reálném čase. Konkrétně v RTS hře Starcraft je přibližně 16 tahů za sekundu.

Pro plánování sekvence akcí v RTS hrách je tedy téměř nemožné použít klasické algoritmy pro prohledávání stromů. Dobré je také využít dynamického programování, kde se celý problém zjednoduší na několik menších podproblémů, u nichž není stavový prostor tolik rozsáhlý. Možná optimalizace je uvažovat čas, po který jedna jednotka akci musí vykonávat. V tuto dobu bude jednotka vyřazena z plánování akcí, protože v něm vykonává již naplánovanou akci.

3.1.2 Učení [13]

Úkolem agenta je naučit se chovat ve hře tak, aby dosahoval co nejlepších výsledků proti jakémukoliv protivníkovi. RTS hry jsou velice komplexní a složitý problém s rozsáhlým stavovým prostorem, proto nelze využít všech postupů zpětnovazebního učení [14]. Je možné využít

například Bayseovských sítí [15], algoritmu Monte-Carlo nebo neuronových sítí. Můžeme rozlišit 3 fáze učení, které inteligentní systémy využívají:

a) Před spuštěním

Získávání informací je možné ze záznamů již odehraných her, znalostí struktur herních map, popřípadě zkušeností hráče. V RTS hrách lze hrát obvykle několik různých počátečních strategií jako jsou obranná, útočná, ekonomická nebo rychlá expanze. Pro každou počáteční strategii existuje nějaká účinná protistrategie. Prvním bodem v učení je stanovení tříd, tedy hraných počátečních strategií. Konkrétně ve hře Starcraft jsou počáteční strategie různé i pro jednotlivé civilizace. V mojí práci jsem se snažil využít tohoto typu učení tak, abych vytvořil co nejlepší počáteční strategii.

b) Při běhu programu

Za běhu hry je nutné co nejlépe odhadnout oponentovu strategii, proto je nutné se snažit provádět neustále průzkum mapy hlavně v oblasti jeho základů. Při vytváření modelu oponentovi strategie lze využít neuronových sítí nebo Bayseovských sítí.

c) Po ukončení

Velké množství inteligentních systémů účastnících se soutěže využívá na základě vytvořeného modelu herní strategie oponenta během odehrané hry přizpůsobení strategie ve hrách nadcházejících. Důležité je toto uzpůsobení hlavně v počátku hry, kdy není známa počáteční strategie oponenta.

3.1.3 Neurčitost

Znamená to nejistotu o všem dění na mapě a pohybu a rozložení oponenta, hlavně z důvodu viditelnosti pouze části mapy. Tuto nejistotu snižuje například průzkum (kap. 2.3.1 a 2.2.1). Další nejistotu vytváří neschopnost přesně předpovědět, jaké akce provede oponent. Je totiž možné v jednu chvíli provést obrovské množství akcí, každá jednotka i budova má velké množství možností prováděných akcí.

3.1.4 Prostorové a časové uvažování

Prostorové uvažování se vztahuje jak na strukturu mapy a jejího využívání, tak na zapojení v provádění jednotlivých akcí, jako je výstavba budov, umístění expanzí a tím vytvářet lepší prostředí pro obranu. Další důležitou součástí je prostorové uvažování v taktice boje. Například jednotky mají na vyvýšeném místě výhodu, protože z nízké polohy není vidět na vyvýšená místa.

Podobně v taktice a volbě strategií je klíčové časové uvažování. Přímou v boji se dá využít pro načasování útoku na nejvhodnější chvíli a načasování následného ústupu. Přepínání

jednotlivých strategií se projevuje se zpožděním. Je nutné tudíž zvažovat jaká strategie se vyplatí vzhledem k projeveným změnám až za určitý čas.

3.1.5 Tvorba modelu oponentovi strategie

V této části rozvoje umělých inteligencí se lze ubírat dvěma směry. Buďto to budeme řešit jakou celkovou strategií nasadit proti právě zjištěné strategii oponenta, a nebo se budeme snažit problém řešit celkovým vytvořením strategie za běhu, avšak toto je velice složité z důvodu velkého stavového prostoru. Nejen že je velké množství jednotlivých jednotek, které se každé zvlášť mohou posílat různé příkazy, ale zároveň každá jednotka má velké množství možných akcí, které v jednu chvíli může vykonat. Zároveň může být i problém s rychlostí rozhodování, protože vše musí probíhat v reálném čase.

3.1.6 Rozložení problému

Díky všem výše uvedeným cílům a důvodům je vhodné rozdělit průběh hry StarCraft do menších podproblémů a tyto zjednodušené podproblémy řešit samostatně. Rozdělení využívají i zkušenosti hráči a to na takzvanou „makro“ a „mikro“ kontrolu. Mikro kontrola je schopnost ovládat jednotky samostatně, a tak docílit delší životnosti jednotek. Makro kontrola je schopnost produkce jednotek a rozšíření základny ve správnou chvíli. Docílí se tím podstatně silnější a větší armády. Pro programové řešení a zefektivnění je dobré rozdělení provést ještě detailněji konkrétně na:

a) Strategie

Je nejvyšší úroveň abstrakce a určuje celkový průběh hry. V první řadě jde o určení počáteční strategie (kap. 3.5.1 a 4.4). V průběhu hry pak jejímu přizpůsobování podle strategie oponenta. Konkrétně pro hru Starcraft se počáteční strategie tolik neliší od strategií v průběhu hry. Jde hlavně o určení z jakých jednotek a v jakém poměru se má vytvořená armáda skládat. K tomuto určování se nejčastěji využívá sběr informací o oponentovi a na základě nich odhaduje kolik může právě mít určitých jednotek a v jakém množství. Pak se z těchto informací určí kolik je třeba vytvořit jednotek jakého druhu, aby jejich výroba trvalo co nejkratší dobu, bylo co nejlevnější a porazilo oponenta s co možná nejmenší ztrátou vlastních jednotek. Zároveň do tohoto výpočtu je možné začlenit již vytvořené jednotky a tak proces výroby urychlit.

b) Taktika

Jde hlavně o způsoby pohybu armád po mapě, jejich dělení do menších skupin popřípadě spojování skupin zpět k sobě. I zde je důležité mít přehled o oponentovi a to hlavně o pohybu jeho vojsk i jejich složení. Velice usnadňuje odhad polohy oponentových vojsk i znalost umístění jeho základen. Další důležitou znalostí je znát jaké budovy má vystavěné popřípadě i jak dlouho.

Z čehož je možné též lépe odhadnout složení a početnost vojska. Některé způsoby chování, které se využívají i v jiných RTS hrách mimo Starcraft, je partyzánství, které jsem popsal v kapitole 2.2.3.

c) Relativní ovládní

V taktikách je nutné uvažovat ovládní každé jednotky zvlášť na bitevním poli a tím je právě relativní ovládní. Nejjednodušším provedením je útočení na nejbližší jednotku. Tento algoritmus lze vylepšit s pomocí zadání priorit do jaké jednotky se má útočit nejdříve. Priority jsou nejčastěji založené na typu jednotky, velikosti jejího poškození a vzdálenosti od právě ovládané jednotky.

d) Analýza terénu

Tento problém z velké části řeší knihovna BWTA (kap. 3.2.2), která jednak mapu rozdělí na několik oblastí, což se hodí pro zjednodušení přehledu o poloze nepřátelského vojska, a také pro nalezení nejlepší pozice pro boj či obranu. Také lze díky analýze terénu mapy zjistit, kde by byla nejvhodnější poloha pro postavení obraných budov.

e) Sběr informací

Sběr informací o oponentovi nejvíce komplikuje „fog-of-war“, mlha, zakrývající dění na mapě. Hráč může vidět pouze dění na mapě v dohledu svých jednotek, proto je důležitý průzkum mapy pomocí některých jeho jednotek a to i na úkor zpomalení rozvoje jeho základny nebo oslabení armády. Průzkum pro RTS hru Starcraft popisují podrobněji v kapitole 2.2.1 a 2.3.1. Veškeré informace je nutné ukládat do předem vytvořené struktury. Ukládá se zde poloha naposledy spatřené nepřátelské jednotky, její typ, případně i počet zdraví. V této struktuře může být uložen i odhad počtu jednotek jakými pravděpodobně v tuto chvíli oponent disponuje.

3.2 Vývojové prostředí

Inteligentní systém je možné implementovat v jazyce C++ nebo JAVA. Můj program je implementován v jazyce C++ za pomoci knihoven [BWAPI](#) a [BWTA](#), které popíši v dalších kapitolách. Pro psaní a překlad programu jsem využíval Microsoft Visual Studio 2008 (9.0). Inteligentní systém je nutné kompilovat jako DLL nebo EXE. Jako základ programu jsem použil kostru [BWSAL](#), která byla nestabilní a nefunkční pro implementaci civilizace Zerg. Podařilo se mi opravit několik chyb, které nestabilitu zapříčinily nebo se jim zcela vyhnout díky nevyužití několika modulů, které nebylo nutné využívat a programové výpočty zpomalovaly nebo dokonce zapříčinily spadnutí programu.

3.2.1 The Brood War Application Programming Interface (BWAPI)

Je to zdarma dostupný a open-source C++ framework, který se používá k interakci s populární RTS hrou StarCraft: BroodWar. Díky této knihovně je možné vytvářet umělou inteligenci - agenta hrajícího tuto hru. BWAPI ve výchozím nastavení pouze odhaluje viditelné části herního stavu a tak umožňuje programátorům vytvářet umělé inteligence bez podvádění, jako je například vidění v místech, kde běžný hráč nevidí. V základním nastavení také BWAPI neumožňuje převzetí kontroly hráčem nad hrou. Všechna tato nastavení je možné pozměnit a tak umožnit efektivnější ladění vytvářeného inteligentního systému. Využívám knihovny verze 3.7.4, ale kód je kompatibilní i s novější verzí 4.0.1 Beta. V nedávné době vyšla nejnovější verze 4.1.1 Beta, na které jsem svůj projekt zatím netestoval. Má podstatně lépe a přehledněji vytvořenou dokumentaci na webové stránce [6].

3.2.2 The Brood War Terrain Analyzer (BWTA)

Tato knihovna je rozšířením BWAPI pro analýzu mapy, výpočet regionů, zúžených míst na mapě a nejlepších nebo začátečních pozic základů. BWTA je zaměřena pouze na analýzu běžných map, takže nefunguje správně pro speciální mapy jako je například mapa *Crystallis*.

3.2.3 The BWAPI Standard Add-on Library (BWSAL)

Jedná se o multiagentní systém skládající se z několika základních agentů pro ovládání hry a systému pro zobrazování ladících informací přímo ve hře [7]. Nejdůležitější součástí této knihovny je `Arbitrator`, usnadňuje spolupráci mezi jednotlivými agenty a rozhoduje, který dostane které jednotky k ovládání. Třída `Arbitrator` využívá šablonu `Controller`, z které je děděn každý z agentů v systému BWSAL. Jedinou součástí BWSAL, která není agentem, je `EnhancedUI`. Zpracovává data v StarCraftu a umožňuje zobrazování dodatečných informací do UI hry. Není nijak propojena s BWSAL, takže je ideální pro ladění při vytváření a upravování umělé inteligence. Dalšími součástmi jsou `BuildingPlacer`, který zařizuje vyhledání a určení pozice pro stavbu budovy. Samotné vytváření budov a jednotek a také přeměňování jednotek má na starost `BuildManager`, který podle typu nutné operace předává ovládání jednomu ze tří podagentů: `ConstructionManager` pro stavbu budov a jejich rozšíření, `ProductionManager` pro výrobu jednotek v budovách a `MorphManager` pro přeměňování jednotek. `BuildingOrderManager` agent zprostředkovává prioritní frontu pro výstavbu jednotek, budov či vylepšení. Pokud se s jeho pomocí do fronty zařadí položka, kterou není možné

vykonat nebo chybí nějaký předchozí výzkum, budova nebo jednotka, pak se do fronty zařadí všechny nutné předchozí položky pro umožnění výroby zařazené položky. SupplyManager dohlíží na zásoby a velikost zaplnění zásob. Snaží se o to, aby nedošlo nikdy k nedostatku zásob a úkoluje BuildManager pro výrobu zásobovacích budov, potažmo jednotek. TechManager a UpgradeManager se využívají k vytváření vylepšení a vytváření technických výzkumů. BorderManager a InformationManager v sobě zase uchovávají informace o mapě a oponentovi: jaké má vytvořené výzkumy, vylepšení, budovy, jednotky, která území základny jsou pod jeho kontrolou a jaké jsou hranice vlastního území. Nejdůležitější součástí BWSAL pro moji práci byl ScoutManager a DefenseManager, které se zabývají průzkumem mapy a obranou základny, respektive útokem na oponenta. Součásti, které jsem znovu implementoval nebo upravoval, popíši v následujících kapitolách. Struktura propojení jednotlivých modulů je znázorněna na schématu 1.

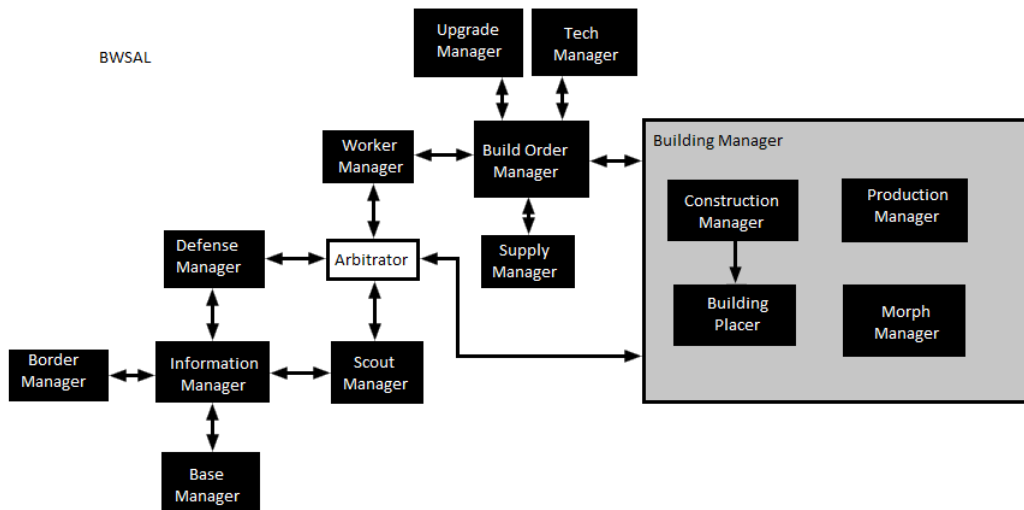


schéma 1: Architektura BWSAL

Černě vybarvené moduly komunikují přímo s hrou pomocí knihovny BWAPI (kap. 4.1.1), plnými šipkami je vyznačeno řízení mezi moduly.

3.3 Scout Manager

Tento agent byl původně vytvořen univerzálně pro všechny tři civilizace, ale vzhledem k tomu, že implementují umělou inteligenci pouze pro civilizaci Zerg, místo univerzální implementace jsem vytvořil tohoto agenta pouze pro tuto civilizaci. Tento agent je programován jako jednoduchý automat, který se snaží o co nejefektivnější průzkum s co možná nejmenšími ztrátami jednotek. Průzkum provádí pouze s jednotkami *Overlord*, takže každá nově vytvořená jednotka je

přiřazována tomuto agentovi k ovládnutí. V agentovi jsou dva stavy, které ovlivňují způsob chování průzkumu. První stav je, pokud není stále objevena oponentova báze. Průzkumníci postupují od nejbližších možných startovních pozic k nejdálším. Seřazení podle vzdálenosti se uskutečňuje pro každého průzkumníka zvlášť a průzkumník s nejkratší vzdáleností se vyšle k této pozici a ze seznamu všech průzkumníků se odstraní, takže nemůže tak dojít k tomu že by dva průzkumníci směřovali ke stejné pozici a jejich průzkum byl neefektivní. Dalším, druhým stavem je, pokud už je oponentova základna objevena. Seřazení všech surovinových základen, výběr a vyslání průzkumníka je stejné jako v prvním případě. Mimo startovní pozice jsou ještě přidány i další možné surovinové základny. Pokud je na průzkumníka započat útok, nejlepší možnou cestou se vzdaluje od útočníků a jeho obsluha agentem je vypnuta a předána agentu `DefenseManager`. Po ukončení útoku na průzkumníka opět začíná průzkum. Při tvorbě tohoto agenta jsem experimentoval se dvěma možnostmi. Při nalezení základny co nejdéle setrvat na oponentově území s prvním průzkumníkem, který ji objeví a odeslat dalšího průzkumníka stejně jako ostatní na průzkum zbytku mapy. Jako lepší se jevila možnost druhá: průzkumníka odeslat na průzkum zbytku mapy, protože u větších map při prozrazení průzkumníka oponent disponoval jednotkou, schopnou průzkumníka velice rychle zničit a tak zmenšit velikost zásob a množství průzkumníků. Vzhledem k tomu, že jsem testoval účinnost a problémy spojené s pohybem, neboli útokem v co nejkratším čase, nebylo tak důležité znát strategii oponenta. Další možností pro rozšíření funkcionality tohoto modulu mohlo být upřednostňování průzkumu těžebních pozic kolem oponentovi základny nebo hlídkování na možných pozemních a vzdušných cestách z území oponenta.

3.4 Defense Manager

V původním provedení tento agent pouze posílal jednotky na hranice vlastního území a tam se hromadili s rozkazem zaútočit na kolemjdoucí oponentovi jednotky. Tento agent je opět implementovaný jako stavový automat s několika stavy. Prvním počátečním stavem je, pokud není objevena oponentova startovní pozice. V tomto případě rozdělí jednotky a odešle je na všechna možná místa na mapě, kam není odeslán průzkumník a kde by se mohl nacházet oponent. V případě, že je objevena oponentova startovní pozice, přechází do druhého stavu. Zde jsou jednotky odesílány na místo, kde by se měla nacházet oponentova základní budova. Pokud je kdekoli na mapě objevena oponentova jednotka nebo budova a je v blízkosti jednotky směřující k útoku, tak tato jednotka přejde do stavu třetího, kdy je udělen rozkaz pro útok na tuto jednotku. Po zničení jednotky, na kterou byl udělen rozkaz útoku, přechází zpět do stavu druhého. Je zde vytvořeno také jednoduché pravidlo upřednostňující útočení na jednotky v největší blízkosti a až

posléze na budovy. Dalším možným rozšířením by mohlo být implementování algoritmu Alfa-Beta, který by využíval lépe informace o jednotkách a jejich vlastnostech. Například by upřednostňoval pro útok více poškozené jednotky nebo jednotky, které by mohl útok více poškodit. Tímto se zabývají například práce [8,9,10]

3.5 ModuleAI

ModuleAI je jádrem inteligentního systému a obsahuje několik metod, které se spouští při určitých událostech, jež nastanou během hry. Mezi základní metody patří `onStart()`, která se realizuje při spuštění hry, `onFrame()`, která se provádí každý snímek hry. Během jedné sekundy ve hře proběhne 16 snímků, pokud ale nastavíme větší rychlost hry počet snímků se přímo úměrně zvětšuje. Při události, kdy se některá jednotka objeví na mapě a je viditelná se volá metoda `onUnitDiscover(Unit)`. Naopak, pokud jednotka zmizí mimo dohled hráče provede se metoda `onUnitEvade(Unit)`. Pokud se jednotka stane neviditelnou (v dohledu hráče) volá se metoda `onUnitHide(Unit)`, naopak, při zviditelnění se provede `onUnitShow(Unit)`. Po zničení jednotky se volá metoda `onUnitDestroy(Unit)`, po vytvoření se volá metoda `onUnitCreate(Unit)`, když se změní jednotka se volá metoda `onUnitMorph(Unit)` a některé další, které naleznete v dokumentaci BWAPI [1]. V metodě `onStart()` jsem nejprve provedl analýzu mapy pomocí knihovny BWTA. Tato analýza, pokud nebyla provedena na stejné mapě v předchozích spuštěních, může trvat i několik minut. Dalším krokem inicializuji počáteční hodnoty proměnných a jednotlivé moduly. Nejdůležitější metodou je `onFrame()`, která provádí aktualizaci všech modulů a agentů, a ve které implementuji počáteční strategii stavební řady. V ostatních metodách, pokud je to potřeba, ošetřuji mizení, vytvoření, ničení, objevení, změnění jednotek a dávám o tom vědět jednotlivým modulům, které využívám.

3.5.1 Počáteční strategie stavební řady

Nejdříve jsem počáteční stavební řadu implementoval pomocí konečného automatu, který sledoval, jaká jednotka, výzkum nebo budova se právě dokončila, ale metoda `onUnitComplete(Unit)` ve starší verzi, ve které bylo nutné implementovat inteligentní systém pro soutěž, nefungovala dobře a občas se provedla několik snímků po vytvoření jednotky. Díky tomuto nepředvídatelnému kritickému zpomalení jsem zvolil strategii implementovat na základě uplynulých snímků od začátku a plněním fronty položkami s prioritou. Jako základní počáteční strategii jsem zvolil *4 Pool Rush*, o které se zmiňuji v předchozí kapitole. Postupně jsem se snažil co nejvíce přiblížit ke strategii *6 Pool Rush*, kde jde jen o malou obměnu, a to, že se výstavba budov pro možnost výroby

armády odsouvá až do doby, kdy je vytvořeno 6 těžebních dělníků. Při takovéto změně je možné i při selhání rushe mít šanci na výhru, pokud byl oponent alespoň částečně oslaben rychlím útokem. Konečná podoba strategie, která byla nejrychlejší a zároveň umožnila nějakého dalšího rozvoje základny po uskutečnění rushe, je nejdříve vyrobit těžební jednotku do počtu 5, natěžit suroviny na stavbu *Spawning Pool*. Během její výstavby vytvořit další dvě těžební jednotky, postavit prvních 6 jednotek *Zergling*, vyrobit další jednotku *Overlord* a pokračovat ve výstavbě dalších jednotek *Zergling*. V tabulce 1 je popsána rychlost stavby 1. vlny útoku pro jednotlivé strategie včetně mojí.

Tabulka 1.

Název strategie	Čas začátku produkce	Počet dělníků	Čas vyprodukování 1. vlny útoku	Jednotky
Moje strategie	2250. snímek	6	2700. snímek	6 Zergling
4 Pool Rush	2250. snímek	3	2700. snímek	6 Zergling
5 Pool Rush	2250. snímek	4	2700. snímek	6 Zergling
6 Pool Rush	2400. snímek	5	2850. snímek	6 Zergling

4 Zhodnocení a výsledky

4.1 Soutěž SSCAI Tournament

SSCAI (Student Starcraft Artificial Intelligence Tournament) je soutěž inteligentních systémů ve hře Starcraft: Brood War. Soutěž pořádá Univerzita Komenského v Bratislavě a České vysoké učení technické v Praze. Soutěže se může účastnit kdokoliv, ale hodnocení je rozdělené na studentskou a veřejnou část. V soutěži se koná zápas každého s každým. Za každý vyhraný zápas se dostanou 3 body a za prohru 0. Vyhrát lze dvěma způsoby - buďto oponentovi zničíme všechny budovy nebo po uplynutí časového limitu máme vyšší bodové hodnocení uplynulé hry než oponent, to je ale hodnoceno pouze 2 body. Zápasy mezi jednotlivými umělými inteligencemi a průběžné hodnocení umělých inteligencí za posledních 20 zápasů je možné sledovat neustále on-line na webové stránce [5].

4.2 Ročník 2014

Tento ročník byl již pátým v řadě. Poprvé nebyl výherce studentské části oceněn finanční odměnou. Soutěžní zápasy započaly na začátku ledna 2015 a vyhodnocení zápasů bylo koncem ledna 2015, do této doby bylo nutné kompletně zpracovat inteligentní systém. Po účasti v této soutěži moje inteligentní systém skončil na sdíleném 10 a 11 místě ze 42 zúčastněných. Za civilizaci Zerg se program umístil těsně na 2. místě. V soutěži nebyl problém některé protivníky pokaždé porazit, naopak některé byl problém porazit vždy, ale nejdůležitější byli právě ti, které můj inteligentní systém někdy porazil a někdy ne. Na těchto zápasech se nejlépe ladila strategie a způsob boje tak, aby nedocházelo k náhodnému porážení mého programu.

4.3 Porovnání počátečních strategií

Ve hře Starcraft existuje velké množství možností jak začít hru, proto je důležitým bodem pro úspěšnou hru i výběr vhodné počáteční strategie. Neexistuje žádná dominantní strategie, která by dovedla porazit většinu ostatních a každá civilizace má různé typy strategií odlišujících je navzájem od sebe. Nelze počáteční strategii vybrat na základě výběru oponenta, protože mapa je zakryta takzvanou „fog-of-war“, viditelnost mapy je omezena jen na blízké okolí jednotek. Výběr počáteční strategie některé inteligentní systémy, účastníci se v soutěži, vybírají na základě předchozích her odehraných proti stejnému oponentovi. Mezi nejznámější počáteční strategie patří:

Pro civilizaci Protoss: „2 Gates“, tato strategie je rush strategie za pomoci pozemních základních jednotek. „Dark Templar Rush“, tato strategie je rush strategie za pomoci technologicky vyspělých jednotek „Dark Templar“, které jsou neviditelné. „High Templar“ je strategie, pro kterou je složitější rychle vyvinout protistrategii, ale zároveň je pomalejší. „Speed Zeal“, velice podobná strategii „2 Gates“, ale je více zaměřená na technologický vývoj jednotek než na velkou a drtivou sílu. „Carrier Push“ zaměřená na produkci velice silných leteckých jednotek, ale velice slabá z počátku hry. „3 Gates“ strategie podobná „2 Gates“, ale produkce je zaměřena na pokročilejší jednotky, které jsou vybaveny děli útočícími na dálku. A poslední běžně hranou strategií za civilizaci Protoss je „Reaver Drop“ zaměřená na produkci přepravních letounů a altilérií velice rychle a efektivně ničí velké skupiny a budovy, jedná se stejně jako u strategie „Corsair“ o velice slabou z počátku hry.

Pro civilizaci Terran: „Marrine+Medic“ strategie s velice rychlou výrobou základních jednotek a jednotek schopných je léčit. „Economic“ strategie snažící se o co nejrychlejší expanzi na další těžební pozici. „2 Factories“ strategie zaměřená na produkci tanků. „Vultures“ strategie zaměřená na produkci rychlých pozemních jednotek výborných pro ničení menších jednotek a lepší kontrolu nad územím, díky možnosti pokládání min. „Air“ strategie zaměřená na produkci leteckých jednotek, slabší z počátku hry. „Drop“ strategie založená na vytvoření přepravních leteckých prostředků a vytváření menších útoků z výsadek přímo do základny oponenta.

Pro civilizaci Zerg: „Pool Rush“ strategie zaměřená na útok v co nejkratším čas za pomoci základních jednotek. „Mutalisk Rush“ snaha o překvapení nepřítele za pomoci leteckých jednotek, řadí se mezi nejrychlejší strategie pro produkci leteckých jednotek. „EcoMutalisk“ strategie zpočátku zaměřená na ekonomiku, později pak produkci leteckých jednotek v podstatně větším množství. „Lurker Rush“ jedná se o strategii zaměřenou na produkci jednotek s možností se v bojišti zahrabat a tím se stát neviditelnými s možností útočit. „Hydra Rush“ snaha překonat oponenta útoky pozemními jednotkami útočícími na dálku, útoky za pomoci této strategie jsou rychlejší než strategii „Lurker Rush“.

Tabulka 2.

Umístění v soutěži	Jméno vývojáře	Architektura	Civilizace	Počáteční strategie
1-2	Tomas Vajda	XIMP	Protoss	Pouze 1 strategie „Carrier Push“
1-2	Martin Rooijackers	UAlbertaBot, Nova	Terran	Volí náhodně počáteční strategii a během hry se přizpůsobuje
3	ICELab	ICEBot	Terran	
4-5	Black White	----	Protoss	Volí náhodně z několika strategií snaží se během hry vylepšovat prostorové a časové uvažování
4-5	Krasimir Krystev	Krasi0bot	Terran	Volí náhodně z několika strategií, během hry se přizpůsobuje
6	Andrew Smith	SkyNet	Protoss	Vybírá ze 3 strategií a během hry se snaží přizpůsobit
7-8	Soeren Klett	W.O.P.R	Terran	Začíná strategií „Marrine+Medic“ nebo „2 Factories“, během hry přizpůsobuje
7-8	Dave Churchill	UAlbertaBot	Protoss	Na základě předchozí hry volí nejlepší strategii, zároveň se během hry přizpůsobuje
9	Marek Suppa	----	Zerg	Pouze 1 strategie „Pool Rush“
10-11	Tomas Cere	----	Protoss	----
10-11	Václav Horázný	BWSAL	Zerg	Pouze 1 strategie „Pool Rush“

5 Závěr a budoucí práce

Inteligentní systém do soutěže se mi podařilo vytvořit tak, že splňovala veškerá pravidla pro její zařazení. V soutěži jsem se umístil na 10-11 místě ze 42 zúčastněných. V části pro studenty byla inteligence umístěna na 5-6 místě a jako civilizace Zerg se umístila na 2. místě. Při ladění počáteční strategie se mi podařilo vytvořit nejrychlejší rush v soutěži a to zároveň díky rychlému průzkumu možných počátečních pozic oponenta. Již přibližně při 2700. snímku hry vyrazí první útok. Průzkum není vždy dostatečně rychlý pro přesné zacílení útoku, proto se může stát, že útok nebude vždy v plné síle. Díky jeho rozdělení na dvě možná umístění oponenta se jeden útok o něco opožďuje. Jako další možné vylepšení tohoto inteligentního systému je možné implementovat algoritmus pro efektivní výběr pro útok a obranu a dotvořit k počáteční strategii heuristiku pro další rozvoj základny, expanzi a výzkumu. Tato práce by se také dala využít jako projekt do předmětu IZU (Základny umělé inteligence), kde by měl student jako úkol vytvořit jednoho z agentů a tak si osvojit lépe dovednosti a znalosti z tohoto předmětu.

Literatura

- [1] *StarCraft* [online]. [cit. 2015-05-10]. Dostupné z: <http://cs.wikipedia.org/wiki/StarCraft>
- [2] *Starcraft: Brood War* [online]. [cit. 2015-03-16]. Dostupné z: <http://us.blizzard.com/en-us/games/sc/>
- [3] *Starcraft/strategies* [online]. [cit. 2015-04-12]. Dostupné z: <http://strategywiki.org/wiki/StarCraft/Strategies>
- [4] *StarCraft Compendium* [online]. [cit. 2015-05-11]. Dostupné z: <http://classic.battle.net/scc/>
- [5] *SSCAI Tournament* [online]. [cit. 2015-03-1]. Dostupné z: www.sscaitournament.com/
- [6] *BWAPI 4.1.1* [online]. [cit. 2015-05-20]. Dostupné z: <http://bwapi.github.io/>
- [7] *BWSAL* [online]. [cit. 2015-05-13]. Dostupné z: <https://code.google.com/p/bwsal/>
- [8] *Optimizing Micro-Management in RTS Games*. Grenoble, 2013. Dostupné z: http://emotion.inrialpes.fr/people/synnaeve/index_files/gameaic.pdf. Článek.
- [9] *Developing game AI for the real-time strategy game Starcraft*. Reykjavik University, 2011. Dostupné z: http://skemman.is/is/stream/get/1946/9882/22934/1/Final_Report.pdf. Výzkumná zpráva.
- [10] Michael Buro. *Call for AI Research in RTS Game*. Department of Computing Science, University of Alberta, Edmonton, AB, T6J 2E8, Canada, 2004. Dostupné z: <https://skatgame.net/mburo/ps/RTS-AAAI04.pdf>. Výzkumná zpráva.
- [11] Michael Buro. *Real-time Strategy Games: A New AI Research Challenges*. Department of Computing Science, University of Alberta, Edmonton, AB, T6J 2E8, Canada, 2003. Dostupné z: <http://ijcai.org/Past%20Proceedings/IJCAI-2003/PDF/265.pdf>. Výzkumná zpráva.
- [12] Santiago Ontanon, Gabriel Synnaeve, Alberto Uriarte, Florian Richoux, David Churchill, et al.. *A Survey of Real-Time Strategy Game AI Research and Competition in StarCraft*. IEEE Transactions on Computational Intelligence and AI in games, IEEE Computational Intelligence Society, 2013, 5 (4), pp.1-19. Dostupné z: <https://hal.inria.fr/file/index/docid/871001/filename/survey.pdf>.
- [13] Frantisek Sailer, Michael Buro a Marc Lanctot. *Adversarial Planning Through Strategy Simulation*. Dept. of Computing Science University of Alberta, Edmonton. Dostupné z: <https://skatgame.net/mburo/ps/rtsmc.pdf>. Výzkumná zpráva.
- [14] Leslie Pack Kaelbling, Michael L Littman. *Reinforcement Learning A Survey*. Computer Science Department, Brown University Providence, USA, 1996. Dostupné z: <http://www.cs.cmu.edu/~gustrin/Class/10701/readings/kaelbling-reinforcement.pdf>. Výzkumná zpráva.
- [15] Gabriel Synnaeve and Pierre Bessière. *A Bayesian Model for Plan Recognition in RTS Games applied to StarCraft*. Collège de France, University of Grenoble, Paris, 2011. Dostupné z: http://emotion.inrialpes.fr/people/synnaeve/index_files/aiide11.pdf. Přednáška.

[17] Bc. Jiří Pazika. *Základní principy umělé inteligence*. Katedra informačních technologií a elektronického obchodování, Bankovní institut vysoká škola Praha, Praha, Česká Republika, 2009. Diplomová práce.

Seznam příloh

Příloha 1. DVD

Obsahuje manuál po instalaci a nastavení všech potřebných součástí projektu, knihovny BWAPI a BWTA, Chaoslauncher

Příloha 2. Tabulka pojmů

Budovy, jednotky a výzkumy ve hře Starcraft: Brood War, použité v textu práce s krátkým popisem.

Příloha 2. Tabulka pojmů [4]

RTS	Real-time strategie
Fog-of-War	Mlha zakrývající herní mapu
Terran, Protoss, Zerg	Civilizace ve hře
SCV	Těžební jednotka civilizace Terran
Drone	Těžební jednotka civilizace Zerg
Probe	Těžební jednotka civilizace Protoss
Rafinery	Těžební stanice plynu civilizace Terran
Extractor	Těžební stanice plynu civilizace Zerg
Assimilator	Těžební stanice plynu civilizace Protoss
Overlord	Letecká jednotka civilizace Zerg, jednotka poskytující zásoby
Supply Depot	Budova poskytující zásoby civilizaci Terran
Pylon	Budova poskytující zásoby civilizaci Protoss
Bunker	Budova civilizace Terran, v které se mohou ukrýt malé jednotky
Rush	Strategie hry, ve které jde o útok v co nejkratším čase
Zerling	Pozemní jednotka civilizace Zerg
Hydralisk	Pozemní jednotka civilizace Zerg
Creep	Hmota tvořící se kolem některých budov civilizace Zerg
Turtling	Strategie hry, ve které jde o vytvoření silné a nepropustné obrany
Creep Colony	Budova civilizace Zerg rozšiřující Creep
Swarm a Overmind	Pojmy z příběhu Starcraftu
Hatchery	Hlavní budova civilizace Zerg
Lair	Vývojové stádium budovy Hatchery
Hive	Vývojové stádium budovy Lair
Sunken Colony	Budova civilizace Zerg zajišťující protipozemní obranu
Spore Colony	Budova civilizace Zerg zajišťující protileteckou obranu
Burrow	Výzkum civilizace Zerg umožňující se pozemním jednotkám zahrabat
Larvea	Pozemní jednotka civilizace Zerg, vyvíjejí se z ní jejich veškeré jednotky
Lurker	Pozemní jednotka civilizace Zerg s možností útočit, když je zahrabaná
Nydus Canal	Dvě budovy civilizace Zerg umožňující přemísťovat pozemní jednotky
Spawning Pool	Budova civilizace Zerg zpřístupňující tvoření jednotek Zergling
Pool Rush	Rushovací strategie se zaměřením na výrobu jednotek Zergling
Lurker Rush	Rushovací strategie se zaměřením na výrobu jednotek Lurker
Mutalisk Rush	Rushovací strategie se zaměřením na výrobu jednotek Mutalisk
Mutalisk	Letecká jednotka civilizace Zerg