

České vysoké učení technické
v Praze

Fakulta elektrotechniky

Katedra kybernetiky



Kooperace skupiny mobotů při
řešení úloh

Bc. Jiří Pavlíček

28. ledna 2000

Anotace

Tato práce se zabývá problematikou kooperace skupiny mobilních robotů (robotů). Cílem práce je navrhnout řídicí systém, umožňující využít výhod vzájemné komunikace a kooperace, které jsou založeny na etologických principech.

Řídicí systém je založen na bázi vrstvené architektury. Výhoda tohoto systému spočívá v relativně rychlé odezvě na podněty z okolního prostředí přepnutím na vhodné chování pro danou situaci.

Navržené řídicí algoritmy byly testovány na skupině čtyř mobilních robotů pomocí simulačního programu vytvořeného právě k tomuto účelu. Výsledky provedených experimentů jsou shrnuty v závěru této práce.

Annotation

This thesis deals with a topic of cooperation in a group of mobile robots (robots). The aim of this thesis is to design a control system that make possible use of advantage of mutual communication and cooperation based on ethological principles.

Control system is based on subsumption architecture of which core is a set of behaviours. The advantage of such system is relatively quick response on sensory stimulus from the environment by switching the appropriate behaviour for given situation.

Proposed control algorithms were tested in a group of four mobile robots by using simulation program that was created for this purpose. Results of performed experiments are summarized in conclusion of this thesis.

Poděkování

Na tomto místě bych chtěl poděkovat všem, kteří jakýmkoli způsobem přispěli ke zdárnému dokončení této práce.

Zvláštní dík patří rodičům, kteří mi umožnili studium na vysoké škole a po celou dobu studia mě finančně a morálně podporovali.

Prohlášení

Prohlašuji, že jsem celou diplomovou práci včetně příloh vypracoval samostatně a uvedl jsem všechny použité podklady a literaturu.

Dále prohlašuji, že souhlasím s využitím této práce a jejích výsledků katedrou.

V Praze, dne 28. ledna 2000

.....

Obsah

Úvod	5
1 Analýza problematiky	7
1.1 Mobilní robotika (stručný přehled)	7
1.1.1 Základní dělení mobilních robotů	7
1.1.2 Oblasti využití	7
1.1.3 Specifické aplikace	8
1.2 Kooperace z hlediska etologie	9
1.2.1 Kooperace v říši živočichů	9
1.2.2 Princip oboustranné výhody	9
1.2.3 Příbuzenská selekce	10
1.2.4 Manipulace	11
1.3 Kooperace v pojetí umělé inteligence	12
1.3.1 Skupinové chování versus DUI	12
1.3.2 Co je to DUI?	13
1.3.3 Významné charakteristiky agenta	15
1.3.4 Charakteristiky společenství agentů	17
1.3.5 Metody komunikace	18
1.3.6 Architektura tabule	20
1.4 Kooperující mobilní roboti a současné projekty	23
1.4.1 Udržování formace v týmu robotů	23
1.4.2 Více rukou - lehčí práce? Výzkum chováním řízených kooperujících autonomních mobilních robotů	27
2 Návrh aplikační úlohy	35
2.1 Výchozí podmínky návrhu	35
2.2 Podmínky nutné pro kooperaci	36
2.3 Požadavky na robota	37
2.4 Důležité aspekty pro návrh aplikační úlohy	39
2.4.1 Orientace v okolním prostoru	40
2.4.2 Identifikace jiného mobota	40
2.4.3 Komunikace mezi roboty	40

2.4.4	Předmět kooperace	40
2.5	Návrh jednoduché aplikační úlohy	41
2.5.1	Formulace kooperativní úlohy.	41
2.5.2	Omezující faktory	41
3	Realizace úlohy	43
3.1	Návrh řídicího systému mobota	43
3.1.1	Implementace řídicího systému	43
3.1.2	Algoritmy řízení mobotů	44
3.2	Simulace	49
3.2.1	Simulační program	49
3.2.2	Vizuální prostředí	49
4	Provedené experimenty	52
4.1	Popis experimentů	52
4.2	Zjištěné výsledky	52
	Závěr	55
	Literatura	57
	Příloha	59

Seznam obrázků

1.1	Formace pro čtyři roboty (zleva doprava: rojnice, zástup, diamant, klín)	24
1.2	Určení pozice ve formaci pomocí různých referenčních technik(zleva doprava: střed formce, vůdce, soused)	26
1.3	Zóny pro určení velikosti směrového vektoru	27
1.4	Čtyři roboti ve formacích <i>diamant</i> , <i>klín</i> , <i>rojnice</i> a <i>zástup</i> s použitím referenční techniky „vůdce“	27
1.5	Porovnání referenční techniky „vůdce“ (vlevo) a „střed formace“ (vpravo)	28
1.6	Architektura systému chování (BSA)	30
1.7	Příklad vzoru chování	31
1.8	Generování výsledné užitečnosti a pohybové odezvy ze dvou vektorů užitečnosti	32
1.9	Typický průběh simulace páru kooperujících robotů	34
3.1	Blokové schéma řídicího systému mobota	44
3.2	Vazby mezi jednotlivými chováními	45
3.3	Vývojové diagramy chování typu <i>hledej-cíl</i> a <i>sleduj cíl</i>	47
3.4	Vývojové diagramy chování typu <i>vyhni-se-překážce</i> a <i>komunikuj</i>	48
3.5	Simulační prostředí	50
4.1	Experiment č.1 - Žádný mobot nekooperuje	53
4.2	Experiment č.2 - Všichni mobot kooperují	54

Seznam tabulek

1.1 Paramtry pohybových schémat	25
-------------------------------------------	----

Úvod

Vývoj multirobotických systémů v posledních letech postupuje kupředu mílovými kroky. Předpokládá se jejich masové nasazení v průmyslové, komerční i vědecké sféře. Počet robotů používaných v současné době v průmyslové oblasti se významně zvyšuje. Vědecký a průmyslový vývoj připravil cestu pro použití robotických systémů i v mnoha dalších oblastech.

Metody pro řízení a komunikaci v multirobotických systémech se vyvíjí již dlouhou dobu. Dokud se používaly multirobotické systémy převážně fixní, tedy bez možnosti autonomního pohybu vlastního robota, řešily se problémy jako koordinace a plánování pohybu pomocí centrálního (hierarchického) řídicího systému. Takovéto multirobotické systémy byly obvykle tvořeny několika typy robotů nebo manipulátorů.

S postupným vývojem se přišlo na myšlenku autonomních mobilních robotů, tedy robotů, kteří se umí v okolním prostředí orientovat a také pohybovat. A díky tomu se také objevily nové problémy jako orientace robota v okolním prostředí, plánování cesty, vyhýbání se překážkám apod. Výzkum autonomních mobilních robotů se začal samozřejmě provádět na jediném robotu interagujícím s okolním prostředím.

V současné době se výzkum zaměřuje na celé skupiny autonomních mobilních robotů díky jejich využitelnosti v úlohách jako jsou vesmírné mise, práce v životu nebezpečných prostředích, armádní operace apod. Takovéto skupinové systémy přinášejí další problémy jako je kooperace, koordinace a autonomní navigace. A právě touto oblastí mobilní robotiky, která mě tolik zaujala, bych se chtěl zabývat ve své diplomové práci.

Kapitola 1 se zbývá analýzou problematiky kooperace a jejím využitím v oblasti mobilní robotiky. Jsou zde uvedeny etologické principy na kterých je kooperace založena a pohled na kooperaci z hlediska umělé inteligence. Dále jsou zde uvedeny projekty ze současné doby zabývající se aplikací kooperativního jednání v rámci skupin mobilních robotů.

Kapitola 2 se zabývá návrhem úlohy pro demonstraci kooperativního jednání ve skupině mobilních robotů, jsou zde uvedeny výchozí podmínky návrhu, různé omezující faktory a vlastní návrh úlohy.

Kapitola 3 popisuje návrh řídicího systému a algoritmy použité pro řízení robotů realizující kooperativní jednání a popis simulačního programu vytvořeného za účelem testování navržených algoritmů.

Kapitola 4 popisuje experimenty provedené k ověření účelnosti kooperativního jednání a jejich výsledky.

Kapitola 1

Analýza problematiky

1.1 Mobilní robotika (stručný přehled)

Mobilní robotika se zabývá vyvíjením mobilních robotů (mobotů), kteří jsou schopni pohybovat se v prostředí a zasahovat do něj. Svým jednáním mohou vykazovat určitý stupeň inteligence. Vlastnosti těchto robotů mohou být různé podle úlohy, pro kterou jsou určeni. Nyní uvedu některé z možností pohledu na dělení mobilních robotů:

1.1.1 Základní dělení mobilních robotů

- řídicí systém (reaktivní, neuronové sítě, neuro-fuzzy, genetické alg.),
- způsob navigace (lokální okolí, globální mapa),
- senzorický systém (kontaktní čidla, IR čidla, sonar, kamera),
- stupeň inteligence (manuální řízení, inteligentní řízení s plánováním pohybu),
- způsob pohybu (nohy, kola, pásy, vrtule),
- morfologie (tvar, velikost),
- prostředí (vnější - vnitřní, vzduch - země - voda),
- řešení úlohy (samostatné, skupinové (multiagentní systém)) ...

1.1.2 Oblasti využití

- autonomní dopravní prostředky (auta, lodě, vlaky, letadla),
- v průmyslu (zemědělství, výroba, doprava, zábava),

- nebezpečná prostředí nebo těžko přístupná prostředí (manipulace s radioaktivním materiálem, chemikáliemi, výbušninami, průzkum potrubí) ...

1.1.3 Specifické aplikace

- úklidové práce (letišť, nádraží, nemocnice),
- doručovatelské služby (zdravotní materiál, jídlo, pošta),
- zemědělství (setí, sklizení, plení, hnojení),
- manipulace s nebezpečnými materiály (výbušniny, chemikálie),
- stavby (domy, silnice),
- armáda (manipulace s výbušninami, boj, špionáž),
- zábava (inteligentní hračky) ...

Ve výše uvedených aplikacích lze jistě najít mnoho úloh, kde se uplatní celá skupina mobilních robotů. Avšak pokud budou roboti pracovat samostatně, efektivita jejich konání zřejmě nebude vysoká. Pokud ale budou roboti moci spolu komunikovat a koordinovat tak svoji činnost lze předpokládat několikanásobné zefektivnění jejich činnosti. Skutečnost, že vzájemná kooperace přináší mnoho výhod pro všechny jedince, ochotné spolupracovat, byla ověřena dlouhodobým evolučním procesem na živých tvorech. Chceme-li tedy proniknout do problematiky kooperace mezi jedinci, a využít tyto principy v oblasti mobilní robotiky, je nutné nejdříve nahlédnout do světa živých tvorů a jejich chování. K tomuto účelu nám dobře poslouží vědní obor zvaný *Etologie*, který se právě chováním živých tvorů zabývá.

1.2 Kooperace z hlediska etologie

1.2.1 Kooperace v říši živočichů

Zkusme se nyní podívat jak souvisí pojem *kooperace* s říší živočichů v etologickém pojetí podle [1]. Mnoho živočišných druhů vytváří buď na určitou dobu, nebo trvale sociální svazky, v nichž si jedinci téhož druhu nejen vzájemně *konkurují*, ale také *kooperují*. Zatímco konkurující si jedinci zkoušejí zvýšit svoji způsobilost na úkor jiných jedinců, aktivity kooperujících jedinců přinášejí prospěch nejen jim, ale i jejich partnerům.

V mnoha případech je zřejmé, že jedinci určité sociální skupiny kooperují. Např.:

- rodiče ptáků odchovávají společně mláďata ,
- ve hmyzím státě pomáhají sterilní jedinci pohlavně zralým jedincům odchovávat potomstvo,
- sociálně výše postavení samečkové primátů se podporují navzájem proti níže postaveným samečkům,
- gazely Thomsonovy se vzájemně varují před plížícím se lvem tak, že provádějí vysoké signální skoky se strnulými končetinami.

Proč ale jedinec spolupracuje, když by vlastně měl maximalizovat vlastní způsobilost a minimalizovat způsobilost konkurentů? Na to jsou možné tři odpovědi:

1. Zvířata spolupracují, protože jejich zájmy nejsou krátkodobé, ale dlouhodobě totožné (*princip oboustranné výhody*).
2. Zvířata spolupracují s příbuznými jedinci a podporují tím přímo i předávání jejich genů na další generaci (*příbuzenská selekce*).
3. Zvířata spolupracují, protože jsou kooperujícím partnerem klamána (*manipulace*)

1.2.2 Princip oboustranné výhody

Partneři jednoho páru se geneticky podílí na společném potomkovi z 50 %. Oba proto musí mít zájem na tom, aby jejich mláďata přežila a rozmnožila se, proto často spolupracují při odchovávání mláďat. Kooperativní agrese nebo společný lov může přinést [spěch všem zúčastněným jedincům.

Jestliže si pomáhají jedinci nespříznění, popisujeme to jako reciproční altruismus. Jako altruismus je kvalifikováno chování, které je pro altruistu nevýhodné a pro jedince, jemuž je určeno, prospěšné. Reciproký altruismus spočívá na výměně

vztahu mezi altruistou a uživatelem. Může-li totiž altruista s dostatečnou pravděpodobností „očekávat“, že uživatel mu altruismus později oplatí, lze evoluci takových altruistických strategií vysvětlit přirozenou selekcí. Výdaje by pro altruistu v každém případě neměly být příliš vysoké, jinak by mohly ohrozit jeho přežití.

Reciproký altruismus předpokládá, že „podvodníci“, kteří využijí altruistické chování, aniž by je oplatili, budou odhaleni. Dále by měli příslušníci jedné sociální skupiny žít pohromadě dostatečně dlouho, aby uživateli poskytnutá altruistická výhoda mohla být později oplacena. Reciproké altruistické strategie lze proto očekávat jen ve stabilních individualizovaných skupinách, disponujících dostatečnými kognitivními schopnostmi v odhalení podvádějících jedinců. Evoluční teoretikové často kladou otázku, proč reciproké altruistické strategie nejsou obcházeny strategiemi podvodnými. Tento problém je formulován jako „věžňovo dilema“ (angl. *prisoner's dilemma*).

Je možné si představit dva zločince, kteří společně vykonali loupežnou vraždu a byli odděleně vzati do vazby. Oba jsou vyslýcháni odděleně a každému je učiněna nabídka, že bude propuštěn i s malou odměnou, jestliže zradí druhého. V tomto případě bude zrazený zvláště přísně potrestán. Jestliže se oba zločinci vzájemně nezradí, kooperují, budou oba propuštěni. Zradí-li se navzájem, oba se tedy podvedou a zůstanou ve vazbě. Dilema spočívá v tom, že kooperující aktér neví, jestli druhý kooperuje také, nebo jestli zradí.

1.2.3 Příbuzenská selekce

Kooperace může často vést i k tomu, že živočichové svůj úspěch - v extrémním případě dokonce i o přežití - doživotně přenechávají partnerům. V tomto případě jde o pravé altruistické chování, které je nejnápadnější u hmyzích států. Potomci královny jsou odchováváni sterilní kastou a ochraňováni členy sterilních kast, jejichž obětavost jde tak daleko, že uskutečňují altruistickou sebevraždu.

Evoluce pravých altruistických strategií chování zůstávala dlouho nevyřešeným evolučně biologickým problémem. Jak je možné, že jsou vyvíjeny strategie, které nezvyšují vlastní životní způsobilost? Teprve teorie příbuzenské selekce přispěla k objasnění výkladu. Příbuzenská selekce (v protikladu k individuální selekci) spočívá v tom, že altruistické chování zvýhodňuje výlučně příbuzné. Hmyzí státy jsou velké rodiny, ve kterých se dělníci a vojáci obětují pro vlastní rodiče a sourozence. Protože altruisté mají s jejich příbuznými na základě jejich společného původu i část společných genů, má jejich chování, oklikou přes příbuzné, příznivý vliv na dále předávané vlastní geny.

Jestliže živočich ochraňuje vlastní potomstvo, je to prospěšné, neboť má s každým potomkem společnou polovinu vlastních genů. Stejný příbuzenský stupeň panuje i mezi ptačími sourozenci, takže se evolučně vyplácí, jestliže si navzájem pomáhají. Nevlastní sourozenci nebo vnuci mají na základě svých příbuzenských vztahů již jen jednu čtvrtinu společných genů a bratrance již jen jednu osminu.

1.2.4 Manipulace

V mezidruhové oblasti jsou rozšířené šalivé strategie jako předpoklad pro sociální parazitismus, např. u kukačky. Samičky pačmeláka rodu *Psithyrus* se opatrně přibližují ke čmeláčím hnízdům, vyhýbají se kontaktu s dělnicemi a zdržují se konečně jeden až dva dny na spodní straně voštin, kde třením o hnízdo a třením o přilétající samečky získávají pach hnízda. Pak usmrtí královnu a kladou vlastní vajíčka o která dělnice pečují, jako by pečovaly o vajíčka vlastních sourozenců.

Je pravděpodobné, že manipulace vedla i k evoluci sterilních kast. Jestliže se samička blanokřídlého hmyzu dělí o hnízdo se svou dospělou dcerou, bylo by pro matku výhodné, kdyby vajíčka dcery nahradila vlastními vajíčky. Dcera by naproti tomu v závislosti na tom, s kolika samečkami matka kopulovala, měla buď nepatrné, nebo vůbec žádné ztráty, poněvadž by místo vlastního potomstva pečovala buď o vlastní, nebo nevlastní sourozence. Zatímco se o příbuzenské selekci jako o hnací síle v evoluci hmyzích států nepochybuje, je vliv manipulace na vývoj hmyzích států dosud neprokázaný.

1.3 Kooperace v pojetí umělé inteligence

Pro řízení skupiny mobilních robotů lze využít teoretických základů z oblasti distribuované umělé inteligence a multiagentních systémů, jak je uvedeno v [4].

1.3.1 Skupinové chování versus DUI

Jak je známo, některé vědní obory se snaží využít pro své účely principy převzaté z živé i neživé přírody, jejichž funkčnost byla otestována staletími vývoje. Ani u (Distribuované Umělé Inteligence - DUI) tomu není jinak.

Nahlédneme-li do světa živých organismů, zjistíme, že jejich základní vlastností je sdružování do skupin. Proč je tomu tak? Výzkumy bylo prokázáno, že v sdružování do skupin poskytuje mnoho výhod nejen jednotlivcům, ale celé skupině. Schopnosti vzniklé skupiny totiž výrazně převyšují prostý součet všeho, čeho je schopen dosáhnout kterýkoli z jednotlivců. Skupina totiž poskytuje kupříkladu větší šanci na přežití, větší úspěšnost při obraně i při lovu. Těchto principů a jejich aplikovatelnosti v oblasti techniky se zabývá právě DUI.

Zcela obecně se DUI zabývá součinností několika totožných či různorodých systémů při řešení společného problému. DUI se soustřeďuje především na problematiku reprezentace, zpracování a využití znalostí v těchto systémech a hledá vhodné modely způsobu rozhodování. Veškeré úvahy týkající se chování a součinnosti agentů v multiagentovém systému vycházejí ze schémat a představ o kooperaci a koordinaci činnosti cílevědomých biologických jedinců, zejména lidí jako inteligentních bytostí v uzavřené komunitě. Obdobně jako v každé komunitě autonomně jednajících jedinců jsou i při návrhu technických řešení multiagentových systémů v centru pozornosti otázky kompetence agentů, způsob a formalismus komunikace mezi agenty, metody a prostředky koordinace aktivit, na vyšší úrovni pak možné způsoby kooperace apod. I vnitřní organizace komunity agentů, která může sdružovat rovnoprávně komunikující zcela rovnocenné agenty či agenty hierarchicky organizované, popřípadě být komunitou s centrálním plánovačem, bývá obvykle inspirována vnitřní strukturou komunity jistého typu (např. komunitou typu „golfový klub“, „vojenský útvar“ nebo „centrálně plánované hospodářství“).

Distribuované řešení úloh reprezentuje tu třídu problémů DUI, která se zabývá hledáním možností jak pro pevně zvolenou úlohu vhodně rozdělit práci na jejím řešení a potřebné znalosti o úloze mezi více modulů (uzlů) a jak mohou jednotlivé moduly sdílet informace o postupu vzniku cílového řešení. V současné době je věnována rostoucí pozornost studiu chování skupin volně propojených autonomních systémů, které spolupracují v zájmu nějakého společného cíle. V tomto kontextu kontextu se hovoří o jednotlivých systémech jako o **agentech**, čímž se zdůrazňuje jejich částečná samostatnost. Celý tento podobor se nazývá **multiagentové systémy**. Využívá sice výsledků a zkušeností získaných při distribuovaném řešení úloh, ale soustřeďuje se především na specifické problémy související se součinností agentů. Agentem zde může být počítačový systém, senzor, robot nebo i člověk.

Obě větve DUI, distribuované řešení problémů a multiagentové systémy, se od sebe liší především problematikou, na kterou každá z nich klade důraz.

Při distribuovaném řešení problémů jde především o to vytvořit dobrý a efektivní systém z hlediska jedné, pevně zvolené řešené úlohy. Dílčí systémy (agenti) řeší přesně specifikované úlohy, přičemž jejich součinnost bývá často přesně (mnohdy centrálně) předdefinována. Důraz se klade spíše na decentralizaci řešení než na autonomnost agentů. Pokud se například podaří ukázat, že v dané úloze je jistý typ chování agenta prospěšný pro celý systém, pak se pozornost distribuovaného řešení problémů soustředí právě na agenty vykazující zmíněný typ chování.

Naopak pro multiagentové systémy takový přístup nepřipadá v úvahu, neboť zde jsou především hledány mechanismy, které zajistí účinnou spolupráci mezi libovolnými samostatnými agenty. Multiagentový přístup může nanejvýš navrhnout takové vlastnosti prostředí systému, které motivují všechny zúčastněné (třeba i sobecké) agenty, aby jednali požadovaným způsobem. Oblast multiagentových systémů klade důraz především na autonomnost agentů a jejich spolupráci v rámci řešení širších, obecnějších tříd úloh. Vazby mezi agenty jsou volnější, flexibilita systému vyšší a vnitřní život komunity bohatší.

Je jistě zbytečné připomínat, že s pojmem agent se můžeme setkat i v některých zneklidňujících a nepříjemných souvislostech. Naštěstí jediným zdrojem nebezpečí, které v rámci studia DUI hrozí, může být rozpor mezi chápáním pojmu agent v DUI a v softwarovém inženýrství. V oblasti softwarového inženýrství je totiž pojem agent chápán jako jakýkoliv výpočetní proces spuštěný na dálku na vzdáleném počítači.

1.3.2 Co je to DUI?

Použití multiagentového systému jak technického řešení nabízí podobné výhody, jakých si ceníme na týmové práci:

- *zkrácení doby řešení* vzhledem k možnostem paralelního postupu,
- *snížení nároků na komunikaci*, pokud se každý specializovaný člen kolektivu zabývá jen svou relevantní částí údajů a předává ostatním především své záměry o nich,
- *zvýšení operativnosti a spolehlivosti* vlivem toho, že tým může podle potřeby přibírat další specialisty, popř. se někteří členové týmu mohou vzájemně zastupovat.

Metodika návrhu multiagentového řešení musí vycházet z důkladného porozumění problematice součinnosti agentů. Tok informací a vývoj znalostí se v multiagentovém systému vztahuje k těmto čtyřem skupinám otázek:

1. Jak jsou problémy formulovány, popsány, dekomponovány a přidělovány agentům? Zda jde o dynamický proces, neboť problémy se mohou objevovat postupně v průběhu řešení výchozí úlohy. S dekompozicí samozřejmě souvisí i způsob syntézy získaných dílčích výsledků.
2. Jaké metody dorozumívání mezi sebou agenti používají? Otázkou není jen to, jaké jazyky a protokoly mají být použity, ale i co, kdy a komu má být sděleno.
3. Jaké metody slouží k odhalení nestejných názorů či rozporuplných záměrů agentů? Jak je zajištěno, aby se rozhodnutí či akce jednotlivých agentů vzájemně nerušily nebo dokonce aby se doplňovaly?
4. Jsou k dispozici prostředky, které dovolí agentovi uvažovat o akcích, plánech a znalostech ostatních agentů?

DUI navrhuje a studuje různé přístupy k řešení těchto otázek, hledá klíčové pojmy pro součinnost složitých skupin. Mezi takové pojmy patří i dotaz, které typické stavy řešení problému je třeba rozlišit, jde li o zajištění účinné koordinace práce celé skupiny agentů. DUI se snaží identifikovat i možné souvislosti mezi použitím jednotlivých metod příslušných každému z naznačených čtyř okruhů otázek. Gasser upozorňuje, že „... různé procedury komunikace a interakce agentů přímo ovlivňují nutnou míru koordinace činnosti agentů, která zajišťuje soulad ve výsledném chování. Různé metody dekompozice úlohy a odtud odvozené vznikající podproblémy kladou rozdílné požadavky na druh interakce mezi agenty či na jejich schopnost uvažovat o chování ostatních agentů. Koordinace výsledného chování záleží na tom, jak se rozhoduje o neshodách v síti agentů, kteří agenti se na těchto rozhodnutích podílejí, ...“.

Technickým cílem DUI je navrhnout praktickou metodologii řešení problémů pomocí distribuovaných systémů. Vlastnosti výsledného řešení záleží jak na schopnostech zúčastněných agentů, tak na organizaci jejich spolupráce. Nabízejí se dvě perspektivy pro pozorování světa DUI a to:

- zdola, z hlediska jednotlivého agenta, a
- shora, z hlediska vztahů ve skupině.

Samozřejmě záleží na tom, jaký stupeň různorodosti (heterogenity) agentů vykazuje budovaný systém. Tento ukazatel se odráží především v rozhodnutích, která souvisejí s organizací celé skupiny. Ovšem pro posouzení míry vzájemné podobnosti mezi agenty musíme poukázat na ty vlastnosti agenta, které jsou podstatné z hlediska DUI.

1.3.3 Významné charakteristiky agenta

Individuální agent může být charakterizován podle toho, zda a na jaké úrovni je schopen zvažovat rozličné varianty svého cíle. V tomto smyslu se rozlišují tři základní kategorie agentů - **reaktivní, intencionální a sociální** (Wooldridge a Jennings, 1994):

- **Reaktivní agent** vždy pouze reaguje na podněty z vnějšího světa. Má k dispozici předem známou množinu akcí, které reprezentují například nějaké stereotypní plány. Vedle toho může reaktivní agent upřesňovat svůj obraz světa tak, že jej upravuje na základě vnějších podnětů. Reaktivní agent může samozřejmě sdělovat své výsledky ostatním. Výběr každé z možných akcí závisí na tom, zda aktuální svět tak, jak je vnímán agentem, splňuje právě její podmínky pro zahájení. Proto reaktivnímu agentovi hrozí zacyklení, pokud se dostane do situace, ve které je jeho vjem totožný s vjemem, se kterým se už dříve setkal. Schopnost řešit úlohy je v případě reaktivního agenta silně závislá na použitém souboru vjemů. Jako typický příklad agenta tohoto typu může sloužit například klasický expertní systém 1 generace. Ovšem modelujeme-li chování **společenství některých nižších živočichů**, pracujeme i s daleko primitivnějšími agenty.
- **Intencionální** nebo **deliberativní** agent zvažuje své možnosti dosáhnout nějakého cíle. Ve volbě cíle se odráží motivace agenta, který vytváří k jeho dosažení plán svých akcí. Koordinace činnosti skupiny agentů probíhá tak, že se agenti vzájemně informují, o tom čemu momentálně věří, i o tom jaké akce plánují.
- **Sociální agent** pracuje dokonce s explicitními modely chování ostatních agentů, kteří spolu s ním tvoří multiagentový systém. Musí být schopen tyto modely upravovat a aktualizovat, neboť je užívá při rozhodování, jak své plány přizpůsobit plánované činnosti jiných agentů. Terminologie v této oblasti není ještě úplně ustálena, Mnohdy se vedle reaktivních agentů uvádějí **modelově a účelově orientovaní agenti** - tyto třídy zhruba odpovídají kategoriím sociální a intencionální agent. v reálném systému může agent v různých situacích vykazovat typy chování odpovídající různým kategoriím. V takovém případě se hovoří o agentu s **hybridním chováním**.

Existuje celá řada definic a podrobných specifikací vlastností agenta. Pro ilustraci uvedme nejcharakterističtější z nich tak, jak je podávají různí autoři:

1. **Bond, Gasser, 1988** - agent je výpočetní proces s jedním centrem řízení a s určitým (lokálním) cílem. Agenti koordinují hlavně své znalosti, cíle, schopnosti a plány tak, aby je mohli využívat společně při akcích a řešení úloh. Agenti v multiagentovém systému mohou ve své činnosti směřovat k jednomu

globálnímu cíli nebo k jednotlivým odděleným cílům, které spolu mohou, ale nemusejí souviset. Agenti sdílejí znalosti o úlohách a řešeních.

2. **Farhoodi, Graham, 1996** - inteligentní agent je entita, která je zodpovědná za rozhodování, zda a jak reagovat na externí podněty. O agentovi obecně platí, že:

- může posílat a přijímat informace od jiných agentů za použití vhodných protokolů,
- může zpracovávat přijaté informace a uvažovat o nich (tj. provádět odvozování, syntézu i analýzu),
- má soubor schopností provádět akce, které se mohou i dynamicky měnit. Akce charakterizují úkoly, které dokáže agent provádět.

Inteligentní agent navíc dovede:

- uvažovat o schopnostech svých i o schopnostech ostatních agentů,
- generovat cíle nebo plány pro sebe i jiné agenty,
- účastnit se složitých interakcí s ostatními (např. za účelem vyjednávání a delegování úkolů), dynamicky se zapojovat do skupin či organizací (které mohou např. sdružovat agenty s podobnou funkcí) nebo takové skupiny opouštět,
- získávat informace a používat jejich zdroje a
- udržovat explicitní modely důvěry pro sebe a ostatní agenty.

3. **Wooldridge a Jennings, 1995** - Wooldridge a Jennings používají dokonce hned dvě definice agenta. První, *obecnější* označuje agenta jako hardwarový nebo častěji softwarový systém, který je:

- autonomní, neboť agent pracuje bez přímého zásahu člověka a do jisté míry řídí své akce a vnitřní stav,
- sociální, protože agenti interagují s ostatními agenty (popř. s lidmi) prostřednictvím jazyka pro komunikaci mezi agenty,
- reaktivní s ohledem na to, že agenti vnímají svoje okolí (fyzický svět, uživatele za grafickým a uživatelským rozhraním, soubor jiných agentů, Internet atd.) a reagují včas na změny v okolí,
- proaktivní, protože reakci agenta na podnět z okolí neurčuje pouze momentální stav, ale jeho chování je řízené cíli v tom smyslu, že agent může sám převzít iniciativu při řešení.

4. **Wooldridge a Jennings, 1995** - druhá *specifičtější definice* dle Wooldridge a Jennigse doplňuje právě uvedené vlastnosti o další požadavky, pro jejichž formulaci využívá antropomorfní termíny (např. důvěra, záměr, závazek). Za agenta je považován autonomní reaktivní a proaktivní počítačový systém, jehož jednání vychází z předpokladů, že jde o systém, který je také:

- mobilní, tj. má schopnost pohybovat se elektronickou sítí,
- korektní, tj. nebude vědomě předávat nepravdivé informace,
- benevolentní, tj. snaží se v rámci svých možností splnit to, o co je žádán,
- racionální, tj. jedná tak, aby dosáhl svých cílů, nikoliv tak, aby v nich selhal (přinejmenším do té míry, jak mu to dovoluje jeho závazek).

Pro realizaci požadovaných schopností jsou podstatné některé základní vlastnosti, které musí mít ve větší či menší míře každý z uvedených typů agentů. Jistě není žádným překvapením, že jde o úlohy, kterými se umělá inteligence už izolovaně zabývala, a je tedy možné využít řady metod vyvinutých v rámci „klasické“ umělé inteligence:

- Agent musí mít určitou **znalost o prostředí**, jehož je součástí. Tuto znalost postupně dotváří a reviduje na základě nově získaných informací.
- Intencionální a sociální agent potřebuje **uvažovat** o světě, o druhých agentech i o hypotetických situacích. Pro takové úvahy jsou vhodné například některé neklasické logiky.
- Adaptace či **učení** jsou vlastní každému živému organismu, proto i každý agent by měl mít možnost těžit z předchozích provozních zkušeností systému. Pro dosažení takového cíle se DUI inspirované jak regulačními principy používanými v lidských společenstvích, které nabízí například trh, tak čistě biologickými principy evoluce.

1.3.4 Charakteristiky společenství agentů

Struktura multiagentového systému je dána vlastnostmi jednotlivých individuálních agentů a komunikační sítí, která je spojuje. Má-li takový systém zajistit řešení určité třídy úloh, pak je nezbytné, aby splňoval uvedené tři podmínky:

- Měl by zaručit **pokrytí** v tom smyslu, že pro každou významnou podúlohu výchozí úlohy existuje skupina agentů (sestavující nejméně z jednoho agenta), která ji dovede řešit
- Stejně důležité je, aby agent měl možnost předávat své závěry agentům, kteří na ně čekají a budou je dále využívat - to závisí na existenci **odpovídajícího spojení**

- Aby bylo možné plně využít možností, které nabízí pokrytí a spojení, měl by být k dispozici dostatečný výpočetní **potenciál** (capability) i odpovídající komunikační cesty, které zajistí vznik řešení (v rozumném čase).

Úkolem organizace multiagentového společenství je pak zajistit **koordinaci, kooperaci a komunikaci** mezi agenty:

- Věcí **koordinace** je především přidělovat nedostatkové zdroje a předávat mezivýsledky. Význam koordinace je zvláště patrný například tehdy, potřebuje-li jeden agent dílčí výsledky od několika dalších. V takovém případě je třeba jejich práci vzájemně synchronizovat. Podobně, má-li více činností využívat týž zdroj, je nutné pečlivě rozprostřít jejich požadavky v čase. Metody koordinace mají svůj vzor v organizaci lidských kolektivů. Patří mezi ně vzájemné přizpůsobení, direktivní velení, nebo i metody standardizace některých aktivit.
- Cílem **kooperace** je zkvalitnění práce multiagentového systému. Durfee (Durfee a kol., 1987) rozlišuje čtyři hlavní cíle, kterých lze dosáhnout vlivem kooperace:
 1. zrychlení řešení využitím paralelního zpracování,
 2. rozšíření třídy řešitelných úloh vyplývajících z výhod sdílení zdrojů,
 3. zvýšení počtu úspěšně završených úloh vzhledem k tomu, že tutéž úlohu zkouší řešit paralelně několik rozličných metod a
 4. snížení počtu kolizí mezi vzájemně soutěžícími agenty.

Stupeň kooperace mezi agenty v systému se může pohybovat od vztahů plně kooperativních až po antagonistické. Mezi metody kooperace patří *vyjednávání* (negotiation), lidem dobře známé třeba z tvorby politických koalic, popř. multiagentní plánování - současné vytváření plánů pro více agentů najednou.

- Možnost **komunikace** je nutnou podmínkou pro kooperaci i koordinaci, v jejich zájmu se budují komunikační kanály sloužící pro předávání informací. Systémy tvořené reaktivními roboty se omezují na primitivní komunikaci, která užívá jen velmi malý soubor signálů, jež mají především zabránit nebezpečným střetům agentů. Při složitější struktuře předávaných zpráv se uplatňuje celá škála metod od přímé výměny zpráv mezi agenty po **metodu tabule**, kdy více agentů zasílá ke zpracování a vyhodnocení své zprávy do místa společného všem agentům.

1.3.5 Metody komunikace

Komunikace je označení pro dorozumívání, tedy přijímání a vysílání informací. Způsob komunikace výrazně určuje vlastnosti distribuovaného systému, protože jeho

prostřednictvím se odehrává koordinace činnosti jednotlivých řešitelů úloh. Základní typy komunikace se liší podle cíle, kam směřují zprávy ve společenství agentů. Komunikaci lze dělit ze dvou hledisek na:

- **komunikaci přímou** - kdy je možné zprávy zasílat pouze přímo dalším agentům a
- **komunikaci nepřímou** - kdy se zprávy soustřeďují v předem zvolené struktuře, která odpovídá sdílené paměti. Tento typ komunikace je obvykle realizován pomocí *tabule*.

Metody **přímé komunikace** lze přesněji charakterizovat podle počtu potenciálních příjemců výchozí zprávy:

- **Adresné posílání zpráv** - znamená předávat zprávy na specifickou adresu příjemce, která musí být odesílateli známa. Motivací pro tento přístup je výrazné omezení počtu vyměňovaných zpráv. K výhodám se počítá i snadná kontrola a zajištění bezpečnosti, protože vysílající agent může zaručit, aby nesprávné informace nikdy nebyly zaslány nesprávným agentům. Jeden z důvodů toho, že adresná způsob předávání zpráv je často používaný, je jistě i jeho snadná a efektivní implementace.

Ale právě adresnost uvedeného postupu je i hlavním zdrojem jeho nevýhod. Jak se zajistí, aby agent věděl, kam má zprávu poslat? Tento problém se stává zvláště aktuální při budování otevřených systémů, kde se počet agentů může dynamicky měnit. Problém nastane například v okamžiku, kdy zmizí adresát zprávy.

- **Všesměrové vysílání zpráv** (broadcasting) - je to přirozený mechanismus, který se stále častěji uplatňuje v distribuovaných systémech. Jeho podstatou je posílání zpráv nikoliv na specifickou adresu, ale všem agentům najednou. k jeho výhodám patří tyto vlastnosti:
 - určitý agent v systému může být nahrazen jiným agentem, který má z vnějšího pohledu ekvivalentní chování, přičemž chování systému jako celku, zůstane nezměněné,
 - je ideální pro systémy, kde jsou úkoly oznamovány a agenti buď soutěží, nebo spolupracují při získávání kontraktů,
 - je široce používané v adaptivních systémech a systémech odolných vůči poruchám, např. v distribuovaných operačních systémech. Agenti, kteří „odumřou“, mohou být nahrazeni jinými, což umožní systému pokračovat v činnosti, i když se některé procesy porouchají.

Všesměrové vysílání zpráv má ale také své nevýhody:

- každý agent má přístup k obsahu všech zpráv, „důvěrné“ informace jsou tedy nedostatečně chráněny,
 - v distribuovaném systému, kde je předáváno značné množství zpráv, může toto neadresné vysílání nadměrně zatěžovat nejen komunikační síť, ale i každého agenta v systému,
 - z programátorského hlediska je považováno za složité
- **Selektivní vysílání zpráv** - nabízí kombinaci adresné a všesměrové komunikace založené na rozdělení prostoru agentů do skupin. Každý agent je členem nejméně jedné ze skupin. Jestliže agent vyšle zprávu, bude zaslána všem členům jedné nebo více skupin agentů. Tak je úplné všesměrové vysílání zpráv vlastně nahrazeno výběrovým vysíláním zpráv.

1.3.6 Architektura tabule

Nepřímé zasílání zpráv vychází z myšlenky sdílené datové struktury (obvykle nazývané *tabule*), do níž jednotliví agenti zasílají výsledky, popř. mezivýsledky, své činnosti, které považují za významné a pro ostatní agenty potřebné. Sdílená datová struktura je pak dostupná všem agentům, kteří z ní mohou kdykoliv načítat informace, které je zajímají.

Komunikaci prostřednictvím jednoduché tabule lze chápat jako nepřímou komunikaci, kdy všechny důležité zprávy od všech agentů jsou zasílány *směřovaně* na jednu adresu (do tabule). Na tuto adresu se současně všichni agenti obracejí jako na (často jediný) zdroj informací o výsledcích činnosti ostatních. Postup řešení problému v tomto prostředí odpovídá živé diskusi skupiny lidí - expertů, kteří mají znalosti z různých oblastí a používají pro výměnu informací křídlo a tabuli. Tabule zajišťuje „kooperativní interakci komunikujících zdrojů znalostí“.

Studujeme-li multiagentový systém s tabulí z pohledu distribuovaného zpracování znalostí (tedy nejenom ze zúženého pohledu nepřímé komunikace zpráv), lze rozlišit tři hlavní komponenty:

1. **Agenti (zdroje znalostí)**. Znalosti, potřebné pro řešení globální úlohy, jsou zachyceny v oddělených, nezávislých souborech, které lze chápat jako agenty. Z pohledu tabule jde tedy vlastně o zdroje znalostí. Každý zdroj znalostí (agent) musí znát své podmínky aplikace, které říkají, jaký stav má být na tabuli před tím, než je zdroj znalostí aktivován. Zdroje znalostí mohou být reprezentovány jako procedury, soubory pravidel nebo logických výroků. Zdroje znalostí modifikují vlastní obsah tabule. Tyto modifikace jsou explicitní a viditelné.
2. **Datová struktura tabule** informuje o aktuálním stavu řešení úlohy, jejím prostřednictvím se odehrává veškerá komunikace a interakce mezi zdroji znalostí. Do tabule se zapisují objekty charakterizující vstupní data, dílčí řešení,

alternativy a konečné řešení (v některých případech i data). Zdroje znalostí mění obsah tabule, což umožňuje budovat řešení úlohy doplňováním údajů.

3. **Řídící část tabule.** Zdroje znalostí reagují oportunisticky na změny na tabuli. Není vyloučeno, že na jistou konstelaci zpráv na tabuli má zájem současně reagovat více zdrojů znalostí. Řízení zodpovídá za zaměřování pozornosti při řešení, tj. indikuje následující položku pro zpracování. Může aktivovat buď zdroje znalostí, nebo objekty na tabuli - v tomto případě upozorní na „ostrůvky“ řešení, které mají být dále sledovány - nebo navrhnout kombinaci obou právě uvedených přístupů, tj. vybrat zdroje znalostí, které mají být použity pro jednotlivá dílčí řešení. Řešení je vytvářeno postupně krok za krokem. V každém stadiu tvorby řešení může být použit libovolný typ odvozování (řízený daty, řízený cíli, řízený očekáváním, atd.). Výsledkem je, že aktivace zdrojů znalostí je dynamická oportunistická, nikoliv pevná a předem naprogramovaná.

Základní podmínkou pro použití architektury tabule je rozdělení výchozí úlohy na volně vázané podúlohy. Tyto podúlohy zhruba odpovídají oblastem specializace v rámci úlohy. Pro konkrétní aplikaci tvůrce organizuje prostor řešení a znalosti potřebné pro nalezení řešení. Tento přístup k dekompozici úlohy a využití znalostí je velmi flexibilní a funguje dobře v různých aplikačních oblastech. Způsob rozdělení úlohy na podúlohy může rozhodovat o jasnosti přístupu, rychlosti, s jakou je nalezeno řešení, s jakou jsou požadovány zdroje a dokonce i schopnosti vůbec vyřešit úlohu jako takovou. Neexistuje ovšem obecně platný předpis, jak při dělení postupovat.

Jeden z možných přístupů pro účinné řízení postupu řešení úlohy je možné popsat podle následujícího algoritmu:

1. Zdroj znalostí provede změnu (změny) na objektu (objektech) na tabuli.
2. Každý zdroj znalostí indikuje svůj možný příspěvek k novému stavu řešení.
3. Na základě informací z bodů 1 a 2 je vybrán směr pro zaměření pozornosti a příslušný řídicí modul se připraví pro vykonání akce podle schématu:
 - (a) Je-li pozornost zaměřena na zdroj znalostí, je vybrán objekt na tabuli (někdy soubor objektů) jako kontext (plánování zaměřené na znalosti).
 - (b) Je-li pozornost zaměřena na objekt na tabuli, je vybrán zdroj znalostí pro zpracování objektu a tento zdroj znalostí je využit s objektem jako kontextem (plánování zaměřené na události).
 - (c) Je-li pozornost zaměřena současně na zdroj znalostí a na objekt, je připravena instance zdroje znalostí pro zpracování s objektem jako kontextem

Vždy musí být k dispozici kritéria, která rozhodnou o tom, kdy ukončit proces řešení úlohy. Obvykle jeden ze zdrojů znalostí indikuje, kdy je proces řešení

úlohy ukončen, buď protože bylo nalezeno přijatelné řešení, nebo protože systém nemůže pokračovat dále z důvodu nedostatku znalostí nebo dat. Základní nevýhodou systémů s tabulí je značná obecnost, která svým způsobem překáží specializaci a omezuje výkonnost.

Bližší informace o problematice multiagentových systémů a metodách kooperace a koordinace lze nalézt v [4].

1.4 Kooperující mobilní roboti a současné projekty

V tomto oddílu předkládám několik projektů na kterých se pracuje v posledních letech. Všechny uvedené projekty se týkají kooperativního chování ve skupinách mobilních robotů popř. projektů se skupinovým chováním související.

1.4.1 Udržování formace v týmu robotů

Cíl projektu

Tento projekt se zabývá vývojem řídicího algoritmu (chování) pro udržování formace v heterogenní skupině mobilních robotů. Výzkum je postaven na platformě čtyř robotických vozidel, sloužících jako průzkumná jednotka v U.S.Army v rámci programu ARPA UGV Demo II. Výhoda udržování formace spočívá v tom, že dovoluje efektivnější využití senzorického systému celého týmu, než při náhodném rozmístění vozidel.

Řešená úloha

Roboti jsou heterogenní v tom smyslu, že každý z nich má unikátní identifikační číslo (ID). Pozice robota ve formaci je určena právě tímto identifikačním číslem. Jiné rozdíly v chování robotů nejsou.

Řízení formace je pouze jedna část mnohem komplexnějšího chování. Roboti musí být schopni nejen udržovat formaci ve skupině, ale v této formaci musí také být schopni dorazit do cílové pozice a vyhybat se překážkám.

Chování umožňující udržování formace je založeno na „pohybových schématech“ (*motor schemas*), což je způsob reaktivní navigační strategie (Arkin 1989). Pohybová schémata pracují jako souběžné asynchronní procesy, kde každý z nich představuje chování na vyšší úrovni. Vjemy jsou transformovány na reakční vektory, které určují aktivní typ chování.

Chování živočichů vedoucí k seskupování do stád nebo hejn je v přírodě dobře známo. Dává jim větší šance na přežití při obraně před predátory nebo větší úspěch při lovu a to díky využití senzorických systémů všech jedinců dohromady. Při zkoumání chování vedoucího na seskupování do stád nebo hejn bylo zjištěno, že toto chování vzniklo jako kombinace požadavku zůstat ve skupině, zatímco bude udržována bezpečná vzdálenost od ostatních jedinců ve skupině. Takovéto chování, které vyplynulo z několika požadavků, které vedly na vytvoření skupinových vztahů, se nazývá *emergentní chování*.

Chování skupiny čtyř mobilních robotů

Jedna z prvních aplikací uměle vytvořeného chování pro seskupování byla použita v počítačové grafice při simulaci hejna ptáků a ryb. Významných výsledků v této oblasti dosáhl Craig Reynolds (Reynolds 1987). Podobných principů využívá i princip pohybových schémat. Tato studie je však zaměřena chování pouze malé skupiny čtyř mobilních robotů.

V rámci projektu byly testovány čtyři typy chování:

- *rojnice*,
- *zástup*,
- *diamant*,
- *klín*.

Tyto formace byly použity z důvodu jejich využití v průzkumných akcích na bitevním poli. V každé formaci má, každý robot pevně určenou pozici podle svého identifikačního čísla (ID), jak je uvedeno na obr.1.1.



Obrázek 1.1: Formace pro čtyři roboty (zleva doprava: rojnice, zástup, diamant, klín)

V tomto výzkumu je úkolem každého z robotů současně dosáhnout cílové pozice, vyhýbat se překážkám, nenarážet do ostatních robotů ve skupině a udržovat formaci. Na základě těchto požadavků budou tedy pohybová schémata takováto:

- **jed-k-cíli**,
- **vyhni-se-pevné-překážce**,
- **vyhni-se-robotu**,
- **udržuj-formaci** a
- **porucha** - schéma řešící problémy způsobené reaktivním řízením.

Každé ze schémat generuje vektor reprezentující požadovanou odezvu chování (směr a vzdálenost pohybu) na podněty z okolního prostředí. Hodnota „zisku“ je použita k určení relativní důležitosti daného chování. Chování na vyšší úrovni je generováno jako násobek každého primitivního chování jeho ziskem, následně pak součtem těchto násobků a znormováním výsledku. Příklad hodnot zisků a dalších parametrů pro prezentované výsledky jsou v tab.1.1.

Parametr	Hodnota	Jednotky
vyhni-se-pevné-překážce		
zisk	1.5	
poloměr dosahu	50	metrů
minimální dosah	5	metrů
vyhni-se-robotu		
zisk	2.0	
poloměr dosahu	20	metrů
minimální dosah	5	metrů
jeď-k-cíli		
zisk	0.8	
porucha		
zisk	0.1	
stálost	6	kroků
udržuj-formaci		
zisk	1.0	
požadovaná vzdálenost	50	metrů
poloměr řídicí zóny	25	metrů
poloměr mrtvé zóny	0	metrů

Tabulka 1.1: Parametry pohybových schémat

Udržování formace je prováděno ve dvou krocích:

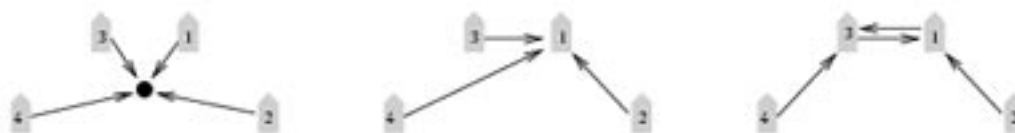
1. proces vnímání okolí, **detekuj-pozici-ve-formaci**, který určuje správnou pozici robota na základě senzorických dat z prostředí,
2. schema **udržuj-formaci**, generující směrový vektor ke správné pozici robota.

Každý robot musí počítat svoji správnou pozici ve formaci v každém kroku pohybu. K určení správné pozice robota byly vyvinuty tři techniky:

1. **Reference ke středu formace** - Střed formace je určen průměrováním x-ových a y-ových souřadnic všech robotů ve formaci. Každý z robotů určuje svoji pozici vzhledem k tomuto středu.

2. **Reference k vůdci** - Každý robot určuje svoji pozici vzhledem k vůdci (Robot 1). Vůdce nemusí udržovat formaci, za udržování formace odpovídají ostatní roboti
3. **Reference k sousedovi** - Každý z robotů udržuje pozici vzhledem k jednomu, předem určenému, robotu.

Vztahy mezi roboty pro jednotlivé referenční techniky jsou zobrazeny na obr. 1.2. Šipky ukazují, jak jsou určovány pozice ve formaci. Každá šipka ukazuje směr od robota k referenčnímu bodu. Schéma **detekuj-pozici-ve-formaci** používá právě jednu z těchto referenčních technik k určení správné pozice robota. Vzdálenost mezi roboty je určena parametrem *požadovaná vzdálenost* ve schématu **detekuj-pozici-ve-formaci**.



Obrázek 1.2: Určení pozice ve formaci pomocí různých referenčních technik (zleva doprava: střed formace, vůdce, soused)

Jakmile je znám požadovaný typ formace, schéma **udržuj-formaci** vygeneruje směrový vektor. Tento vektor směřuje vždy k požadované pozici ve formaci, ale jeho velikost závisí na aktuální vzdálenosti robota od požadované pozice. Obrázek obr.1.3 znázorňuje tři zóny, určené vzdáleností od požadované pozice, použité pro určení velikosti směrového vektoru. Poloměry těchto zón jsou určeny parametry schématu **udržuj formaci**. Velikost směrového vektoru je určena následovně:

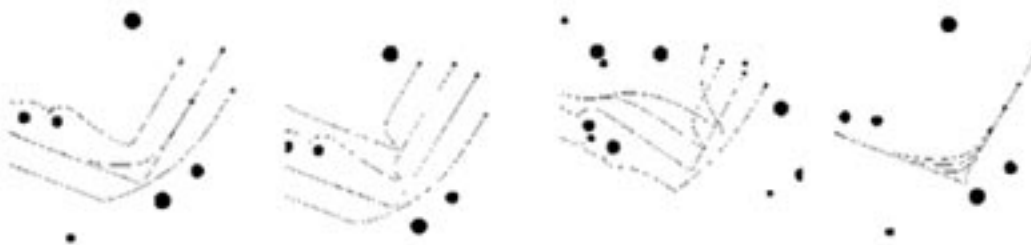
- **Balistická zóna** - velikost je maximální, tj. rovna parametru „zisk“,
- **Řídící zóna** - velikost se mění lineárně, z maxima na vnější hranici na minimum na vnitřní hranici,
- **Mrtvá zóna** - v této zóně je velikost vždy nulová.

Simulace

Simulace je provedena v prostředí MissionLab běžícím na stanici SunSPARC pod SunOS a grafickým prostředím X11. Simulační prostředí má velikost 1000 x 1000 m. Do prostředí mohou být umístěny kruhové překážky různých velikostí. Každý



Obrázek 1.3: Zóny pro určení velikosti směrového vektoru

Obrázek 1.4: Čtyři roboti ve formacích *diamant*, *klín*, *rojnice* a *zástup* s použitím referenční techniky „vůdce“.

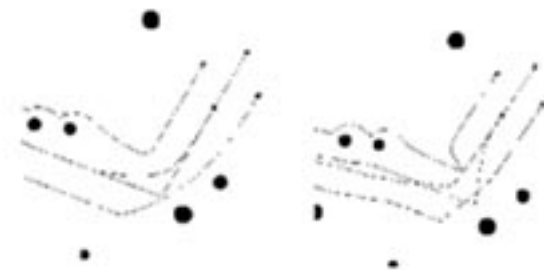
simulovaný robot je reprezentován separátně běžícím programem v jazyce C, interagujícím se simulačním prostředím.

Na obr. 1.4 a obr. 1.5 jsou výsledky simulací pro různé typy formací a referenční techniky. Na obr. 1.4 jsou vidět výsledky simulace pro čtyři roboty, pohybující se ve čtyřech typech formací *diamant*, *klín*, *rojnice* a *zástup* s použitím referenční techniky „vůdce“. Na obr. 1.5 je vidět srovnání dvou referenčních technik „vůdce“ (vlevo) a „střed formace“ (vpravo) ve formaci *diamant*. Podrobnější informace o tomto projektu lze nalézt v [6].

1.4.2 Více rukou - lehčí práce? Výzkum chováním řízených kooperujících autonomních mobilních robotů

Cíl projektu

UK Robotics Ltd. ve spolupráci s University of Salford zkoumají oblast kooperujících mobilních robotů. V této práci započal D.P. Barnes již v roce 1993 a díky jeho aktivitám mohl vzniknout projekt: *Multiple Automata for Complex Task Achievement (MACTA)* (Barnes & Aylet 1994). Uvedený projekt je zaměřen na aplikační



Obrázek 1.5: Porovnání referenční techniky „vůdce“ (vlevo) a „střed formace“ (vpravo).

oblast manipulace s materiálem využívající dvou mobilních robotů, známých jako Ginger a Fred. Tento výzkum, jehož výsledkem byl návrh a implementace řídicího systému robotů (Barnes 1996) umožňující autonomní kooperativní řízení dvou mobilních robotů. Aplikace této řídicí architektury nazvaná jako „Behaviour synthesis architecture“ (BSA) byla s úspěchem demonstrována v jejich výzkumné laboratoři.

Řešená úloha

Úkolem robotů je kooperativně přesunout daný objekt do cílové pozice a vyhybat se překážkám na cestě k cíli. Výzkum ukázal, že chováním řízený systém je pro tuto úlohu vhodnou volbou.

Současný výzkum, se zaměřuje problém vyřazování jaderných elektráren z provozu za pomoci skupiny kooperujících mobilních robotů. Na světě je totiž v provozu mnoho jaderných reaktorů zprovozněných ještě před rokem 1980, což znamená, že jejich provoz by měl být ukončen krátce po začátku nového století. I se současnou technologií je demontování reaktorů velmi náročná práce vyžadující speciální vybavení, jako dálkově řízené manipulátory. Problémem je více manuálně řízených robotických zařízení, např. při přesunu materiálu dvěma mobilními roboty, vizuálně řízenými operátorem a a za pomoci několika dalších mobilních robotů. Toto je problém řízení více robotů s několika stupni volnosti, vyžadující značnou pozornost a dovednost operátora. Vše je samozřejmě komplikováno tím, že se pracuje v radioaktivním prostředí, kde je kladen velký důraz na bezpečnost a veškerá činnost je mnohem pomalejší než v normálním prostředí. V rámci výzkumu byli zhotoveni roboti Ginger a Fred, jejichž úkoly jsou definovány následně:

1. Fred provádí navigaci skrz prostředí laboratoře k nákladní plošině označené majákem. Ginger následuje Freda.
2. Fred zajede na určené místo nákladní plošiny a čeká na náklad.

3. Po naložení nákladu Fred provádí navigaci prostředím i s nákladem. Ginger se přesune co nejbližší k Fredovi.
4. Fred přenesení náklad na Ginger, zatímco se oba pohybují. Po obdržení nákladu konečně Ginger provádí navigaci do cílové oblasti, kde je náklad vyložen.

To je komplexní úkol, který si od robota žádá mnoho schopností: *navigaci, objíždění překážek, vjezd do nákladního prostoru, sledování, naložení a složení nákladu a kooperativní přesun nákladu.*

Nyní bude stručně popsána architektura řídicího systému BSA viz obr. 1.4.2, která je založena na řídicích systémech vyvinutých Arkinem a Brooksem. Řídicí systém obsahuje čtyři úrovně chování:

1. **self** level - zahrnuje typy chování, které maximalizují a doplňují interní zdroje, např. setrvávání v klidu šetří energii baterií.
2. **environment** level - zahrnuje chování spojeného s aktivitami zahrnujícími další objekty uvnitř prostředí, např. zabránění kolizi.
3. **species** level - zahrnuje chování týkající se kooperace, např. udržování správné pozice a orientace s ohledem na objekt, který je kooperativně přesunován.
4. **task** level - zahrnuje typy chování související s částmi úkolu, např. navigace na počáteční souřadnice objektu, který má být převezen a poté navigace do cílové pozice.

Senzorický systém poskytuje vnitřní a vnější stavové informace potřebné pro různé úrovně chování. Každá úroveň chování obsahuje několik vzorů chování (*behaviour patterns* - **bp**), kde

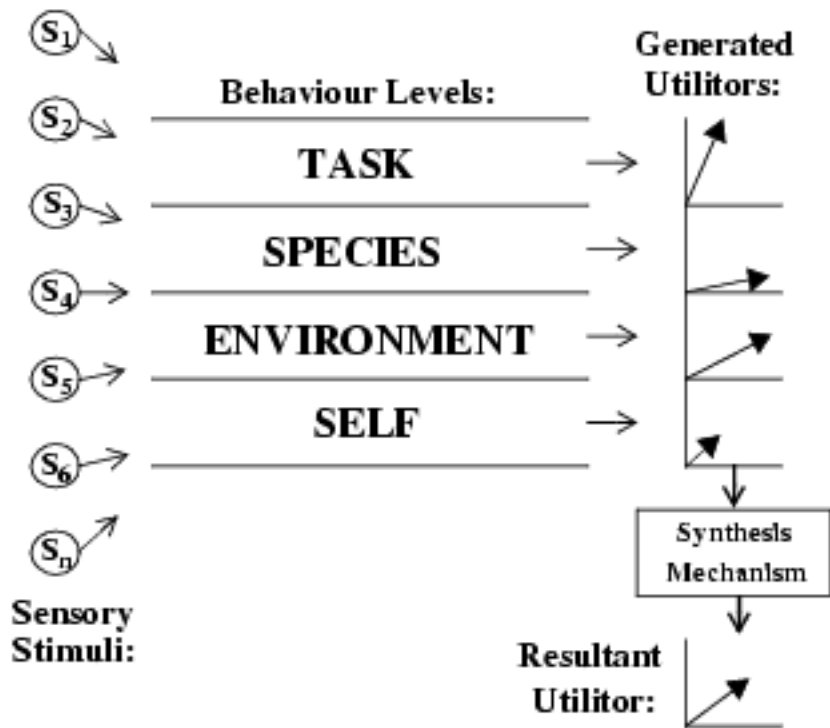
$$\mathbf{bp} = \begin{bmatrix} r \\ u \end{bmatrix}$$

a

$$r = f_r(s)$$

$$u = f_u(s)$$

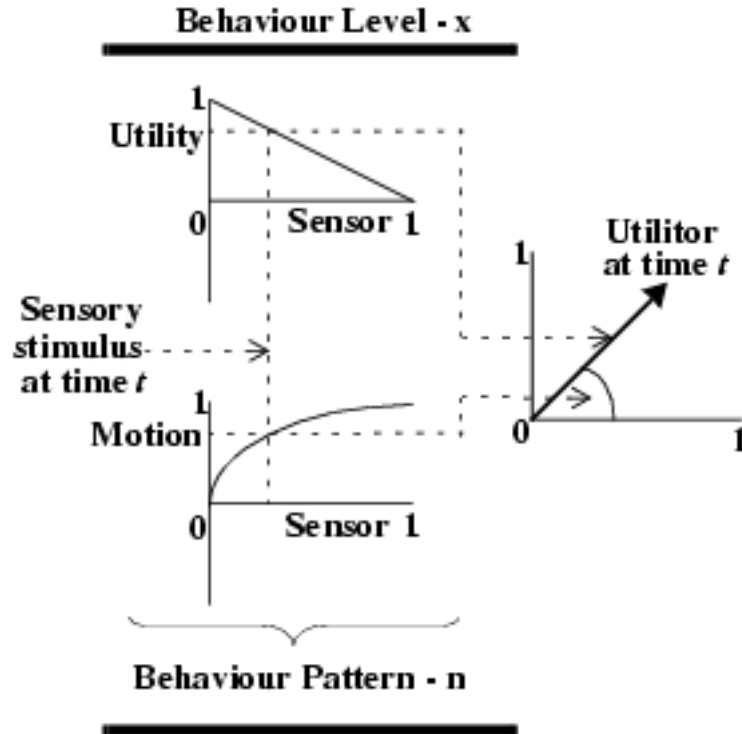
r je požadovaná odezva na podnět ze senzorů a je funkcí f_r , daného stimulu ze senzorů s . Ke každé odezvě je určena míra její užitečnosti nebo důležitosti u . Tato míra je funkcí f_u , téhož senzorického stimulu. Pojem *užitečnost* (**utility**) pochází z předchozího výzkumu a formalizace pojmů pro modelování kooperativního chování. Užitečnost je použita pro velkou efektivitu při výběru vhodné strategie z množiny všech možných strategií a byla koncipována tak, že je možné ji začlenit do uvedené architektury řízení. Proto **bp** definuje nejenom jaká by měla být pohybová odezva



Obrázek 1.6: Architektura systému chování (BSA).

robota na daný vstup senzorů, ale také poskytuje míru, jak se mění relativní důležitost této odezvy vzhledem ke stejnému vstupu senzorů. Hodnoty r a u tvoří vektor definovaný jako *vektor užitečnosti (utilitor)*.

Obr. 1.7 zobrazuje možný případ vzoru chování \mathbf{bp} na dané úrovni. Uvažujme případ, kdy stimulus ze senzorů se vztahuje k senzoru měřícímu čelní vzdálenost robota od překážky a odezva robota se vztahuje k transformaci rychlosti pohybu robota. Na obr. 1.7 je vidět, že jakmile se robot přiblíží k překážce, jeho rychlost pohybu bude redukována na nulu. V ten samý čas se zvýší užitečnost, spojená s danou odezvou. Takže jakmile se robot během cesty přiblíží k překážce, začne robot zpomalovat. V jakémkoli časovém okamžiku se může vyskytnout konflikt několika odezev odezev systému na daný podnět. Například robot může být navigován do cílové pozice, zatímco kooperativně převáží náklad a najednou se v cestě nečekaně objeví překážka a v ten samý čas robot cítí, že je třeba dobít baterie. Co udělá robot v takové situaci? V architektuře BSA je tento konflikt řešen mechanismem syntézy jednotlivých chování, který generuje výslednou odezvu. Soutěžící vektory užitečnosti jsou složeny principem lineární superpozice, a následně je vytvořen výsledný vektor užitečnosti UX_t , kde:



Obrázek 1.7: Příklad vzoru chování.

$$UX_t = \sum_{n=1}^m u_{(t,n)} \cdot e^{j \cdot r_{t,n}}$$

a m je rovno celkovému počtu zúčastněných užitečností generovaných různými úrovněmi chování. Daný výsledný vektor užitečnosti, výsledná užitečnost uX_t a výsledná pohybová odezva rX_t se jednoduše získají ze vztahu:

$$uX_t = \frac{|UX_t|}{m}$$

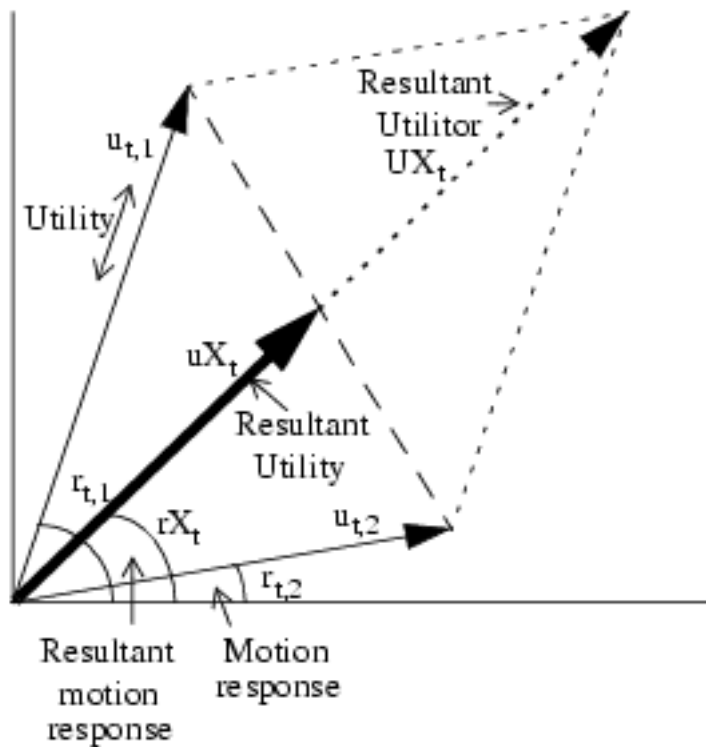
a

$$rX_t = \arg(UX_t)$$

X značí příslušný stupeň volnosti, např. translace nebo rotace. Syntéza jednotlivých chování z různých úrovní je naznačena na obr. 1.8.

Adaptace chování

Přestože architektury založené na chování jako např. vrstvená (subsumption) architektura (Brooks 1986) a BSA umožňují přímé mapování stimulu senzorů na odezvu,



Obrázek 1.8: Generování výsledné užitečnosti a pohybové odezvy ze dvou vektorů užitečnosti.

neumí se vypořádat s adaptací na prostředí, které vyžaduje strukturovanou odezvu, jako např. slepá chodba. K řešení tohoto problému byla přizvána na pomoc fuzzy logika (Zadeh 1965), přesněji řečeno systém fuzzy pravidel, který umožňuje řešit tuto situaci pomocí slovního vyjádření (Surman, Peters & Huser 1995). Fuzzy řídicí systém je velmi robustní a umožňuje tolerovat většinu chyb ve struktuře pravidel (Kosko 1992) a necitlivý na šum nebo neurčitost na řídicích vstupech, hodí se tedy perfektně pro řízení mobilních robotů (Watanabe & Pin 1993). Fuzzy řídicí systém pracuje tak, že umožňuje zakódovat znalosti experta do množiny pravidel, která jsou postupně interpolována a výsledek je potom defuzifikován, abychom získali požadovaný výstup řízení. Každé pravidlo je specifikováno jako trojúhelníková, lichoběžníková nebo jiná funkce např. gausián a přiřazena určitému intervalu hodnot vstupní proměnné. Trojúhelníkové funkce byly vybrány z důvodu malé výpočetní náročnosti, což je pro systém běžící v reálném čase rozhodující. Fuzzy pravidla obvykle mají tvar :

if (x is A) and (y is B) then (z is C),

kde x, y, z jsou slovní proměnné reprezentující vstupy a výstupy fuzzy řídicího

systému a A, B, C jsou termy pro proměnné v prostoru platnosti X, Y, aZ . Fuzzy pravidla mohou být reprezentována pomocí fuzzy asociativní paměťové matice (*Fuzzy Associative memory Matrix - FAM*) (Kosko 1992). V tomto systému reprezentují základní rozměry matice vstupní proměnné a každý vstup FAM reprezentuje výstupní fuzzy množinu. Typicky se používá 3 až 5 výstupních funkcí příslušnosti, např. *negative large (NL)*, *negative small (NS)*, *zero (ZE)*, *positive small (PS)* a *positive large (PL)*. V reprezentaci FAM, je váha pro i -tý vstup FAM určena pomocí minima pravidel

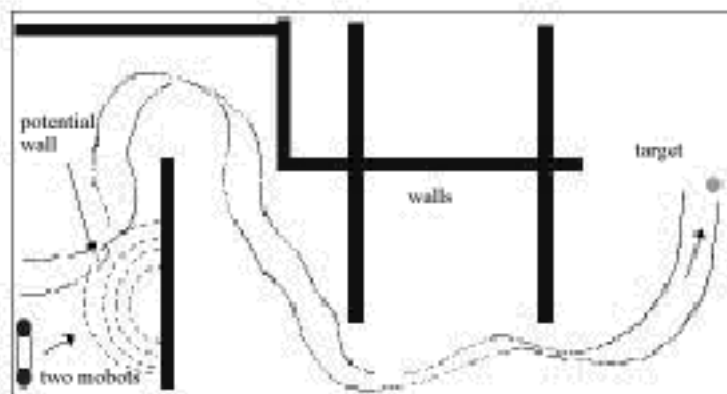
$$w_i = \min\{F_i(x), F_i(y)\},$$

kde x a y určují vstupní dimenzi FAM. Celková defazifikovaná odezva pro n výstupních funkcí příslušnosti je pak

$$F_T = \frac{\sum_i (w_i \cdot B_i)}{\sum_{i=1}^n w_i},$$

kde B_i je výstupní fuzzy množina. Stejně jako BSA používá funkce užitečnosti pro každý vzor chování **bp**, bylo zajištěno, že tyto funkce mají přímo význam adaptace relativní užitečnosti každého z chování ve vyšší vrstvě řízení. Vyvinutá vrstva fuzzy řízení má přímou vazbu na vstup senzorů a má zápornou zpětnou vazbu F_T na užitečnost vybraných chování. Příspěvek každého chování je tedy dynamicky upravován, aby odpovídal podmínkám vnějšího prostředí. Robotovi je tedy umožněno soustředit se na provádění požadované odezvy na současný stav prostředí, zatímco také jistou pozornost věnuje splňování všeobecnějšího cíle. Jako příklad si uveďme situaci, kdy se robot blíží k překážce ve směru k cílové pozici. Důležitost vyhnout se překážce se zvýší díky aktivní funkci objíždění překážky f_u uvnitř BSA, zatímco fuzzy řízení provede snížení užitečnosti pohybu směrem k cíli, tedy $f_u(s)Nav := f_u(s)Nav - F_T$. Vzniklá odezva robota bude tedy taková, že se začne pohybovat po obvodu překážky, dokud se neuvolní cesta směrem k cíli. Pak se robot začne opět pohybovat v cílovém směru, viz. obr. 1.9.

Použití fuzzy systému přináší však také jisté problémy. S rostoucím počtem vstupů a výstupů roste rapidně také počet možných kombinací použitých fuzzy pravidel. To vedlo mnohé vědce k použití všeobecných vyhledávacích algoritmů a umělých neuronových sítí k nalezení optimální množiny pravidel. V tomto projektu bylo použito standardních genetických algoritmů (Holland 1992) pro zpracování výběru vstupů do FAM s ručně vybranými vstupními fuzzy množinami a funkcemi. Schéma kódování pro genetickou optimalizaci je takové, že se vezme každý vstup do FAM a přiřadí se binárnímu vektoru s pevnou délkou. Tyto vektory jsou pak sloučeny dohromady je vytvořen „chromosomový“ vektor. Každá zpracovaná FAM je pak ohodnocena hodnotící funkcí a je aplikován klonovací proces, aby mohla být vybrána nejlépe vyhovující matice. V prováděných experimentech byla zvolena hodnotící funkce jako kombinace vyhýbání se překážkám a minimalizace vzdálenosti od cíle.



Obrázek 1.9: Typický průběh simulace páru kooperujících robotů

Simulace

Simulace byla prováděna na počítači řady 486 a typické časy pro získání vyhovující matice FAM se pohybovaly od 10 do 20 hod., v závislosti na velikosti populace složitosti okolního prostředí. Bližší informace o projektu lze nalézt v [7].

Kapitola 2

Návrh aplikační úlohy

2.1 Výchozí podmínky návrhu

Při návrhu úlohy demonstrující kooperativní jednání mobotů vycházím z několika předpokladů.

Laboratoř mobilní robotiky při Katedře Umělé inteligence a robotiky na ČVUT - FEL v Praze, pod vedením Doc. Pavla Nahodila, CSc., se v posledních letech zabývá vývojem platformy čtyř mobilních robotů (mobotů) za účelem testování skupinového chování. Vlastnosti a vybavení těchto robotů jsou následující:

- mobot má tvar osmihranu, délka hrany 15 cm,
- podvozek se 4 koly, kde přední a zadní kola jsou otočná vodící a postranní kola jsou hnací, poháněná dvěma krokovými motory,
- 16 IR čidel po obvodu mobota, souměrně rozmístěných,
- 24 kontaktních čidel, taktéž souměrně rozmístěných po obvodu mobota,
- 1 otočný sonar, umístěný v geometrickém středu mobota,
- jádro řídicího systému mobota tvoří mikropočítač Intel 8051
- komunikační rozhraní RS232C (pevná linka/radiolinka),

Veškerá "inteligence", tedy hlavní řídicí systém mobota, který rozhoduje o vykonávaných akcích mobota na podněty z vnějšího prostředí, je realizována externím počítačem. Teoreticky by každý mobot mohl mít svůj vlastní, samostatný řídicí systém, ale z finančních důvodů jsou řídicí systémy jednotlivých mobotů emulovány na jednom stroji.

Komunikace mezi nadřazeným systémem a mobotem není přímá, ale obsahuje ještě jeden mezičlánek, kterým je „arbiter“. Arbiter je další počítač, který obstarává přenos a správné směrování informací mezi jednotlivými moboty a jejich nadřazeným

řídícím systémem. Celý systém je navržen tak, že pokud se nadřazený systém nedotáže mobota na jeho stav, nezjistí žádnou změnu v jeho chování. Mobot sám totiž nemůže navázat komunikaci s nadřazeným systémem. Kolizní situaci mobot může řešit tak, že provede odpovídající reakci (zastavení) definovanou jeho vnitřním řídicím systémem, ale vyřešení této situace může provést pouze nadřazený systém a to až po dotazu na stav mobota. Vzájemná komunikace mezi moboty probíhá na úrovni nadřazeného řídicího systému mobotů.

Pokud bude tedy nadřazený řídicí systém používat výpočetně náročné metody pro plánování akcí mobotů, budou moboti nuceni po většinu času stát a čekat na ošetření kolizní situace.

2.2 Podmínky nutné pro kooperaci

V této části se chci zabývat otázkami, které vznikají s požadavkem aplikace společenského chování ve skupině mobilních robotů, konkrétně *kooperace*.

Základním aspektem vzájemné kooperace jedinců, je vzájemná *motivace a komunikace*. Chci-li s někým spolupracovat, musím k tomu mít rozhodně dobrý důvod, který mi přinese prospěch ze vzájemné spolupráce. Hlavním smyslem kooperace je však prospěšnost pro všechny zúčastněné. Někdy dochází i k tomu, že zúčastněný jedinec nemá krátkodobě ze spolupráce žádný prospěch, ale z pohledu dlouhodobějšího časového horizontu se mu spolupráce vyplatí. Zde tedy vyvstává hned první otázka:

1. Co může být motivujícím faktorem pro spolupráci ve společenství robotů, kteří nemají žádné emoce?

K čemu nám ale bude dostatečná míra motivace a umění navázat komunikaci, když nebudeme mít s kým komunikovat? Nejdříve tedy musíme najít někoho, kdo bude ochotný komunikovat. Člověk i ostatní tvorové v přírodě mají *dostatek vstupních informací i rozpoznávacích schopností* k tomu, aby určili, zda objekt, který potkali je živým tvorem, teoreticky schopným komunikace. To že se jedná např. o stejný živočišný druh ještě neříká nic o tom, zda bude komunikace možná. Zde je totiž nutný předpoklad stejného *komunikačního protokolu*. Takže se zde naskýtají další otázky:

2. Jak získá robot dostatek vstupních informací o okolním prostředí? A když tyto informace bude mít, bude schopen tyto informace také zpracovat a vyhodnotit?
3. Jak bude probíhat společná komunikace? Existuje společný komunikační protokol?

Nyní se důkladně zamysleme nad tím, proč vlastně chceme s druhou stranou spolupracovat. Co si představit pod pojmem „motivace“? Celý problém motivace

ke spolupráci bezesporu souvisí s otázkou přežití jedince. Tato pudová vlastnost je každému jedinci dána geneticky, a ten se snaží ze všech sil uzpůsobit své chování a jednání tak, aby uspokojil tento pud. Z přírody jsou známy i případy, kdy se jedinec vzdá práva na život ve prospěch ostatních jedinců z hlediska zachování svého druhu. Otázka přežití a života vůbec je natolik složitá, že se jí zde nehodlám zabývat. Tímto problémem se zabývá vědní obor Etologie, jehož zakladatelem je Konrad Lorenz, viz. [2] a [1]. Ale základní aspekty potřebné pro přežití si zde uvedeme. Začněme např. vhodným okolním prostředím, kde je možné nalézt prostředky k zajištění dodávky *energie* pro živočicha, např. potrava, voda, vzduch apod. Máme-li k dispozici tyto prostředky, musíme je umět také zpracovat a manipulovat s nimi. K tomu potřebujeme vhodnou *tělesnou konstrukci*. Máme zde tedy další otázky:

4. Jak musí být robot konstruován, aby mohl provádět potřebné úkony?
5. Jak bude vyřešena otázka dodávky energie?

Co když se ale ocitneme v situaci, že nemůžeme sami zajistit základní potřeby pro přežití? Odpovědi jsou pouze dvě. Buď narazíme na někoho, kdo nám pomůže nebo zemřeme. Zde se tedy nachází jedna odpověď, nikoli však jediná, na otázku motivace ke spolupráci. Kooperace se nemusí týkat pouze jedinců stejného druhu. Každý živočišný druh má jinou genetickou výbavu a i jedinci stejného druhu mohou mít odlišné schopnosti. Kooperace mezi jedinci tedy může vzniknout i na základě toho, že jeden pro druhého vykonávají činnost kterou ten druhý neumí, a proto tedy spolupracují.

Mají-li být splněny všechny předchozí požadavky potřebujeme ještě element, který oživí celý systém. Tím elementem je samozřejmě řídicí systém, na který jsou kladeny nemalé nároky. Řídicím systémem člověka je samozřejmě mozek. Tajemný orgán, o kterém se dnes stále ještě přesně neví jak funguje. Ale jisté je, že musí ve zlomku vteřiny vyhodnocovat tisíce vjemů a také na ně adekvátně reagovat. V tuto chvíli vyvstává poslední otázka:

6. Co bude mozkiem robota? Jak bude tento mozek fungovat?

2.3 Požadavky na robota

Podívejme se nyní blíže na jednotlivé otázky, týkající se nároků na konstrukci jednotlivých robotů, kteří mají spolu kooperovat, a jejich chování.

Co může být motivujícím faktorem pro spolupráci ve společenství robotů, kteří nemají žádné emoce?

Robot sice nemá emoce, tak jako člověk, ale jisté potřeby přece jen má. Aby mohl fungovat potřebuje zdroj energie. Jedná-li se o mobilního robota, bude zřejmě napájen bateriemi, které se samozřejmě vybíjí a je tedy třeba je dobíjet. Robot může

zjišťovat stav baterií a zjistí-li, že bude energie brzy vyčerpána, můžeme to přirovnat k pocitu hladu. Ví-li robot jak a kde baterie dobít, musí také odhadnout zdali mu zbylá energie postačí, aby dojel k nabíjecí stanici. Pokud neví kde se nabíjecí stanice nachází, nebo už nemá dostatek energie, bude odkázán na pomoc ostatních jedinců. Má tedy motivaci ke kooperaci.

Jak získá robot dostatek vstupních informací o okolním prostředí? A když tyto informace bude mít, bude schopen tyto informace také zpracovat a vyhodnotit?

Tato otázka bude asi poněkud komplikovanější. Aby robot mohl získávat informace z okolního prostředí, potřebuje k tomu sensorický systém, který bude poskytovat dostatečně spolehlivá data. K základnímu sensorickému vybavení patří dotyková čidla, IR čidla, laserová čidla nebo kamera, která umožňuje zpracování obrazu. Tato čidla samozřejmě nejsou dokonalá a ne vždy poskytují pravdivé informace. IR čidla například nebudou reagovat na objekty s malou odrazivostí povrchu. Sonar bude poskytovat falešné údaje díky odrazům v rozích místnosti. Kamera bude fungovat jen při správných světelných podmínkách.

Vzhledem k tomu, že mobilní robot s v prostředí také potřebuje pohybovat, musí se v něm i orientovat, rozpoznat situaci ve které se nachází a plánovat následující akce. Někdy stačí k rozhodování pouze informace z lokálního okolí mobota a není třeba vytvářet globální mapu prostředí, ale ve složitějších aplikacích je vytváření globální mapy nutností, stejně tak jako znalost aktuálních souřadnic robota v této mapě. Jak je vidět, jsou roboti předurčení pro život ve světě geometrických souřadnic. Jak jinak by byli schopni orientace v prostoru?

K tomu, aby mobot mohl navázat komunikaci s jiným mobotem, musí vědět o existenci toho druhého a naopak. Sensorický systém by tedy měl umožnit identifikaci druhého mobota.

Schopnosti vyhodnocení vstupních sensorických dat, rozpoznání situace a plánování akcí jsou dány schopnostmi celého řídicího systému.

Jak bude probíhat společná komunikace? Existuje společný komunikační protokol?

Předpokladem společné komunikace je společný jazyk, v případě robota komunikační protokol. Nejvíce se používá jazyk, který je pro počítač přímo vytvořen, tedy jazyk strojových instrukcí na nižší úrovni a deklarativní nebo procedurální jazyk na vyšší úrovni. Další otázkou je způsob komunikace. Nejjednodušší způsob komunikace je předávání informací po pevné sériové lince, radiovým spojením nebo opticky pomocí infračidel. V poslední době se pracuje i na systémech rozpoznávání řeči nebo vizuálně sdělovaných informací pomocí kamery.

Jak musí být robot konstruován, aby mohl provádět potřebné úkony?

Základním předpokladem u mobilního robota je pohyb v prostoru. Musí tedy být vybaven vhodným lokomočním systémem, jako jsou pásy, kola, nohy, vrtule apod. Od robota se ale neočekává pouze pohyb v prostoru, ale také jeho interakce s prostředím. Měl by tedy být uzpůsoben také k manipulaci s předměty. K tomuto účelu může být robot uzpůsoben tvarově, např. má radlici vhodnou k posouvání předmětů nebo různé robotické paže, umožňující jemnější manipulaci s předměty.

Jak bude vyřešena otázka dodávky energie?

Vzhledem k tomu, že pro řízení robotů v současné době není jiný vhodnější systém než počítač, který potřebuje ke své činnosti elektrickou energii, je dodávka energie u mobilního robota většinou řešena dobíjecími akumulátory. Tím je robotu umožněn snadnější pohyb v prostoru, neboť nemusí za sebou tahat pevný napájecí kabel. Počítač sám spotřebuje poměrně málo energie. Nejvíce energie spotřebují akční členy robota při jeho pohybu, manipulaci s předměty a získávání informací z okolního prostředí senzorickým systémem.

Co bude mozkiem robota? Jak bude tento mozek fungovat?

Jak již bylo zmíněno, mozkiem robota je počítač, který celý systém robota oživuje. Co by ale byl počítač bez programu? Řídící program ovlivňuje činnost celého systému, dává mu jakousi formu inteligence. Programem je určen způsob vyhodnocování vstupních dat, provádění rozhodnutí a akčních zásahů do okolního prostředí.

Program může využívat různé metody pro vyhodnocování stavu systému. Může pracovat deterministicky a používat jednoduchá pravidla. Může využívat neuronové sítě, genetické algoritmy, neuro-fuzzy regulátory atd. Funkčnost celého systému bude také záviset na rychlosti počítače. Není-li odezva systému dostatečně rychlá, celý systém se stává v praxi nepoužitelným. Proto je třeba volit při návrhu řídicího systému různé kompromisy. Výpočetní technika je dnes již na poměrně vysoké úrovni, přesto však na některé aplikace např. neuronové sítě nebo použití genetických algoritmů nestačí. Experimenty prováděné pomocí těchto prostředků mohou trvat několik dní, ba dokonce i týdnů, aniž by bylo dosaženo uspokojivého výsledku. Proto se v praxi stále používají osvědčené metody s jednoduchými řídicími algoritmy.

2.4 Důležité aspekty pro návrh aplikační úlohy

Zohledním-li při návrhu úlohy pro kooperaci robotů skupinu čtyř mobilních robotů vytvořených ve zmíněné Laboratoři mobilní robotiky na naší škole, vyplynou z toho na základě předchozích zjištění následující omezení:

2.4.1 Orientace v okolním prostoru

Podle zadání se moboti mají pohybovat v omezeném, předem neznámém prostoru. Vzhledem k tomu že moboti jsou vybavení 1 otočným sonarem, prstencem IR čidel, prstencem kontaktních čidel a krokovými hnacími motory jsou možnosti orientace v prostoru a určování polohy následující. Díky sonaru může mobot určovat svoji vzdálenost od okolních předmětů. Sonar se otáčí s krokem asi 2° . Otočení sonaru o 360° a tím prozkoumaní celého okolí trvá poměrně dlouhou dobu. Při otáčení sonaru navíc musí robot stát. Sonar tedy není možné používat příliš často, protože by se robot téměř nehnul z místa. IR čidla je možné použít k detekci blízkých předmětů a vyhnout se tak srážce. Pokud ke srážce skutečně dojde, měly by tento fakt oznámit kontaktní čidla. K určování pozice robota v globálních souřadnicích je možné použít odometrii, tedy metody založené na určování ujeté vzdálenosti na základě znalosti velikosti kol a kroku u krokového motoru. Tato metoda je však velmi nepřesná díky prokluzu kol. Polohu je také možné určovat vyhodnocováním sonarických dat pomocí různých metod jako např. mřížka obsazenosti prostoru, zpřesňování polohy pomocí korelace apod. To ale není předmětem mé práce.

2.4.2 Identifikace jiného mobota

Vytvoření moboti nejsou vybavení žádnými čidly umožňující identifikaci druhého robota. Není tedy jiná možnost než řešit znalost existence jiných robotů programově a vzájemnou interakci mobotů odvozovat ze znalosti geometrické polohy robota v globálních souřadnicích nebo robota vybavit čidlem, které umožní identifikaci ostatních robotů.

2.4.3 Komunikace mezi roboty

Moboti nejsou uzpůsobeni k přímé komunikaci mezi sebou. Veškerá komunikace mezi roboty se může odehrávat pouze na úrovni nadřazeného řídicího systému. Způsob sdělovaných informací bude závislý na implementaci objektu mobota a sdělované informace budou závislé na požadavcích aplikační úlohy. Jak je uvedeno v kap. 1.3.5, existuje několik metod jak provádět komunikaci ve větší skupině agentů.

2.4.4 Předmět kooperace

Konstrukčně nejsou moboti uzpůsobeni k jinému účelu než k mapování okolního prostředí nebo přesunu lehkých předmětů posunem po podlaze. Navíc vzhledem k tomu, že moboti nejsou sensoricky vybaveni k rozpoznávání objektů, není opět jiná možnost než rozpoznávat objekty podle předem zadaných údajů o jejich poloze, nebo mobota vybavit potřebnými senzory.

2.5 Návrh jednoduché aplikační úlohy

Na základě předchozích zjištění jsem se pokusil formulovat jednoduchou úlohu, na které by bylo možné alespoň částečně prokázat výhody a smysl kooperace v uvedené skupině mobilních robotů.

2.5.1 Formulace kooperativní úlohy.

V omezeném, předem neznámém prostředí se náhodně pohybují čtyři mobilní moboti hledající určitý objekt. Moboti jsou vybavení čidlem, které umožňuje detekci tohoto objektu. Smyslem úlohy je lokalizovat hledaný objekt a zajistit, aby se všichni moboti dostali do této cílové pozice, pokud možno v co nejkratším čase a s minimální ujetou dráhou.

Pro jednoduchost se předpokládá že všichni moboti jsou ochotni sdělit bezprostředně informaci o nalezení cíle a jeho souřadnice ostatním. Moboti se pak mohou rozhodnout zda tuto informaci použijí ve svůj prospěch nebo ne. Moboti, jsou ochotni kooperovat, po obdržení této informace skončí s náhodným prohledáváním prostoru a okamžitě zamíří k cíli. Při pohybu v prostředí se moboti navzájem vyhýbají sami sobě i překážkám. V případě, že neznají souřadnice cíle, při detekci překážky náhodně změni směr pohybu, v opačném případě se snaží překážku objet a dorazit do cílové pozice.

2.5.2 Omezující faktory

Prostředí

- Moboti se budou pohybovat v omezeném, předem neznámém prostředí. Velikost prostředí je stanovena na maximálně 5×3 m,
- Prostředí je popsáno globálním systémem kartézských souřadnic.

Orientace v prostředí

- Pro orientaci v prostředí budou moboti využívat pouze prstence 16 IR čidel.
- Moboti předem znají svoji pozici a úhel natočení v globálním souřadném systému. Poloha mobota je v vztažena k jeho geometrickému středu, který je také počátkem jeho lokálních souřadnic.

Objekty v prostředí

- Objekty v prostředí mají takový povrch, který dobře odráží infračervené paprsky, z důvodu možnosti jejich detekce IR čidly,

- Měly by mít jednoduchý tvar, pokud možno hranatý. a velikost by řádově měla odpovídat velikosti mobota.

Komunikace mezi moboty

- Moboti předem „vědí“ o své existenci a jsou přesně rozlišeni,
- Vzájemná komunikace probíhá na úrovni nadřazeného systému,
- Moboti používají přímou komunikaci typu ”jeden-všem”.

Detekce cíle

- Moboti jsou vybaveni senzorem pro detekci hledaného objektu, popř. senzory pro detekci jiného mobota.

Řídící systém

- Moboti jsou vybaveni nadřazeným systémem na bázi vrstvené architektury. Jádrem systému je soubor chování, která jsou spouštěna podle rozpoznané situace.

Kapitola 3

Realizace úlohy

3.1 Návrh řídicího systému mobota

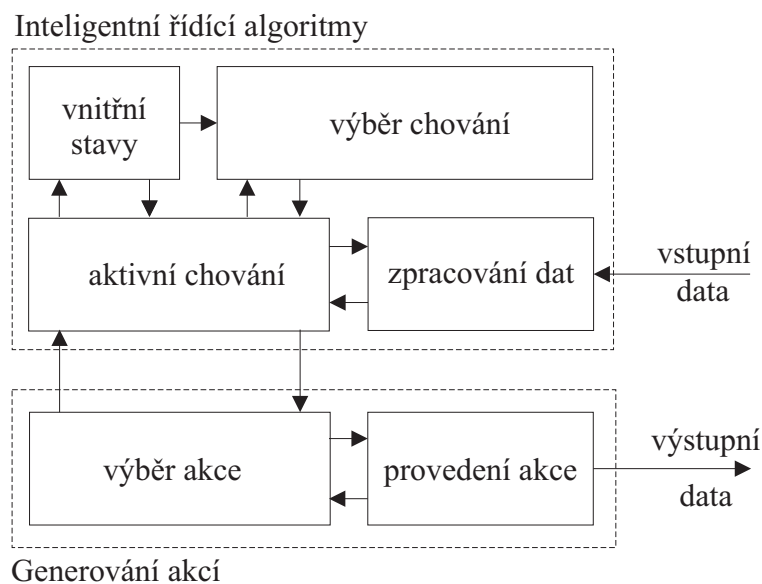
Řídicí systém mobota vychází z principu vrstvené (subsumption) architektury, která byla vyvinuta v roce 1985 na univerzitě MIT pod vedením R.A.Brookse. Přesněji se jedná o tzv. „behaviour based system“, který v sobě zahrnuje několik typů chování. Blokové schéma řídicího systému je znázorněno na obr. 3.1. Kognitivní systém v sobě zahrnuje všechna aplikovatelná chování, vnitřní stavy mobota, vyhodnocení vstupních informací a výběr správného typu chování. Typ chování je vybrán na základě vyhodnocení vnitřního stavu mobota a údajů ze sensorického systému. Jednotlivá chování mají na sebe vzájemné vazby. Je-li aktivní jeden typ chování a nastanou podmínky pro spuštění jiného chování, toto chování se spustí a je tím zajištěna okamžitá reakce na okolní podněty, viz obr. 3.2.

Vzhledem k tomu, že každé z chování využívá ke své činnosti stejné zdroje (vnitřní proměnné, senzory, aktuátory), může být ve stejný časový okamžik aktivní pouze jeden typ chování. Aktivní je tedy vždy nejvhodnější chování pro danou situaci. Každé chování využívá k zajištění předmětu své činnosti pouze nezbytné nutné informace, takže neprovádí žádné zbytečné operace, které by zpomalovaly reakce systému. Bližším popisem vrstvené architektury se zde nebudu zabývat, neboť byl již popsán v rámci diplomových prací předchozích členů naší skupiny mobilní robotiky, viz. [8], [9], [10], [11].

Mobot si nevytváří žádnou globální mapu okolního prostředí, ale pouze velmi jednoduchou reprezentaci. Mobot ví, kde je počátek souřadného systému okolního prostředí a zná svoji polohu vzhledem k tomuto referenčnímu bodu. Před započítím činnosti musí být tato informace mobotu sdělena.

3.1.1 Implementace řídicího systému

Řídicí systém je implementován pomocí objektově orientovaného jazyka C++ (zdrojový kód je součástí přílohy). Skládá se ze dvou hlavních objektů, a sice `TMobot` a



Obrázek 3.1: Blokové schéma řídicího systému mobota

`TDrivingThread`.

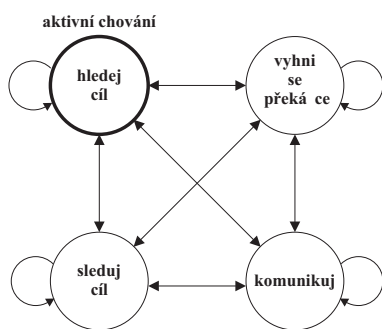
Každý mobot je reprezentován samostatným objektem, odvozeným od objektu `TMobot`. Objekt `TMobot` nese informace o vnitřních stavech mobota a jsou v něm obsaženy i metody pro jeho řízení a vykreslování na obrazovku. Hlavní metoda, která operuje nad množinou chování, má označení `DrivingMobot`.

Tyto hlavní řídicí metody jednotlivých mobotů jsou spouštěny v separátně běžícím vlákně (threadu), které je reprezentováno objektem `TDrivingThread`.

3.1.2 Algoritmy řízení mobotů

Mobot je řízen čtyřmi typy chování:

- 0 `hledej-cíl` - mobot se náhodně pohybuje prostředím. Při nárazu na překážku náhodně zvolí další směr pohybu. Parametr *Random Step* určuje počet kroků, kdy mobot změní směr pohybu, aniž by detekoval překážku. Při detekci překážky se mobot zastaví a přepne na chování `komunikuj`,
- 1 `sleduj-cíl` - mobot již zná souřadnici cíle a směřuje svůj pohyb směrem k němu. Při detekci překážky nebo jiného mobota přepne ba chování `vyhni-se-překážce`,
- 2 `vyhni-se-překážce` - mobot se snaží kopírovat povrch překážky a tímto způsobem ji objet. Jakmile je směr k cíli volný přepne na chování `sleduj-cíl`. Sledování překážky je ovlivněno parametry:



Obrázek 3.2: Vazby mezi jednotlivými chováními

Attraction - ovlivňuje přiklání mobota k překážce,

Repulsion - ovlivňuje odklání mobota od překážky,

NotWallContact - počet kroků, kdy se mobot začne přiklánět k překážce, ztratí-li s ní kontakt.

3 **komunikuj** - toto chování obhospodařuje komunikaci s ostatními moboty, nastane-li podmínka pro zahájení komunikace. Zde je definováno, kterým mobotům se bude posílat zpráva, respektive od kterého se bude přijímat, a jakým způsobem bude mobot na sdělené informace reagovat. V mém případě je komunikace realizována velmi jednoduše tak, že mobot, který našel cíl pošle zprávu všem ostatním o nalezení cíle a do předem určeného místa pošle informaci o souřadnicích cíle.

Tato chování, jsou realizovány položkami `case(0-3)` v příkazu `switch` metody `TDrivingMobot`. Přepínání jednotlivých typů chování se děje na základě nastavování vnitřní proměnné **FBehaviour**, která reprezentuje aktivní chování. Jednotlivé typy chování jsou stručně a přehledně popsány pomocí vývojových diagramů na obr. 3.3 a obr. 3.4.

Každé z chování využívá k zajištění předmětu své činnosti akce typu:

go - popojed o definovaný krok dopředu/dozadu,

zatoč - zatoč o určitý úhel doprava/doleva,

CheckIR - zjistí stav IR čidel,

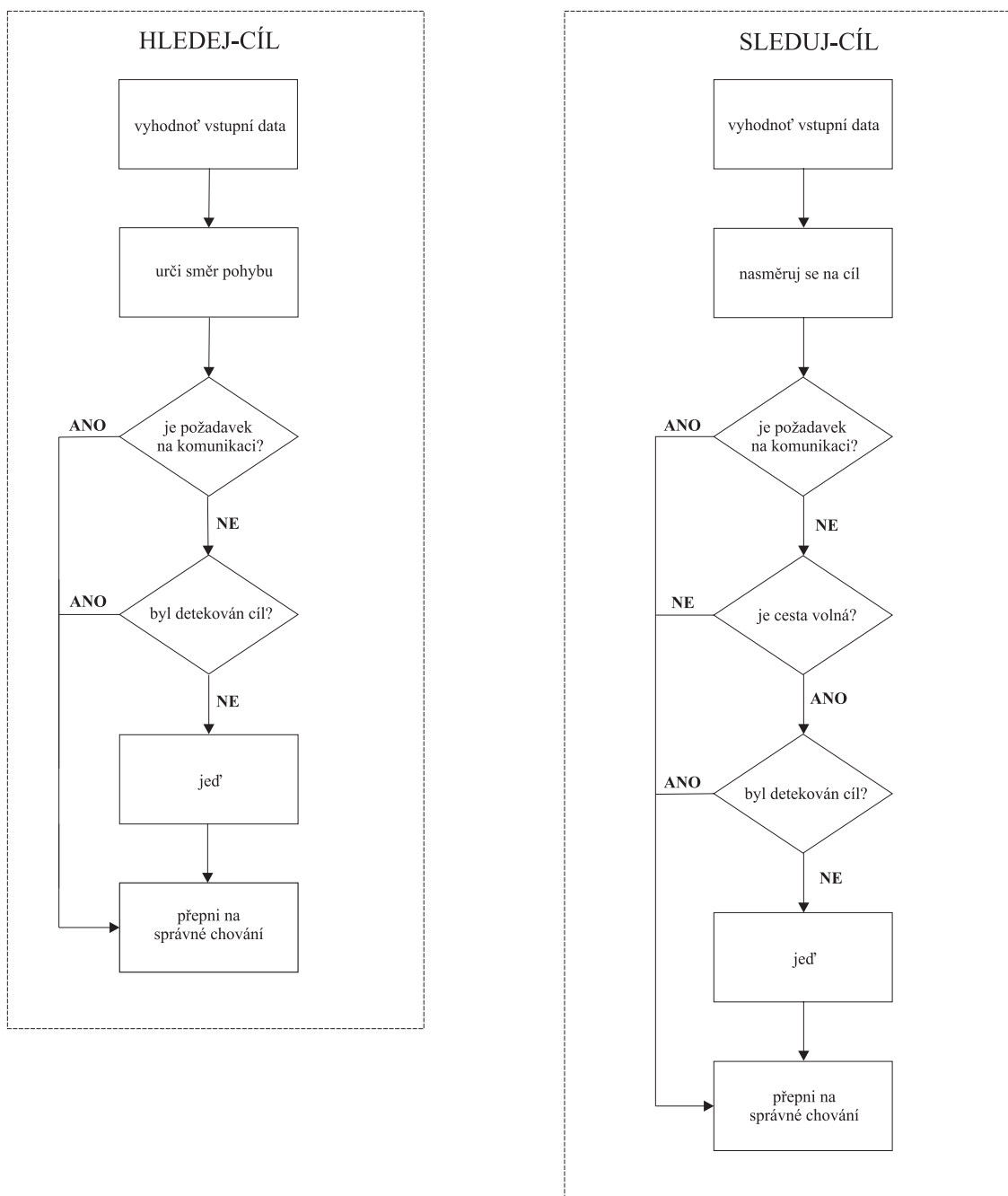
ChooseDirection - určí směr pohybu,

a další metody pro obsluhu grafiky.

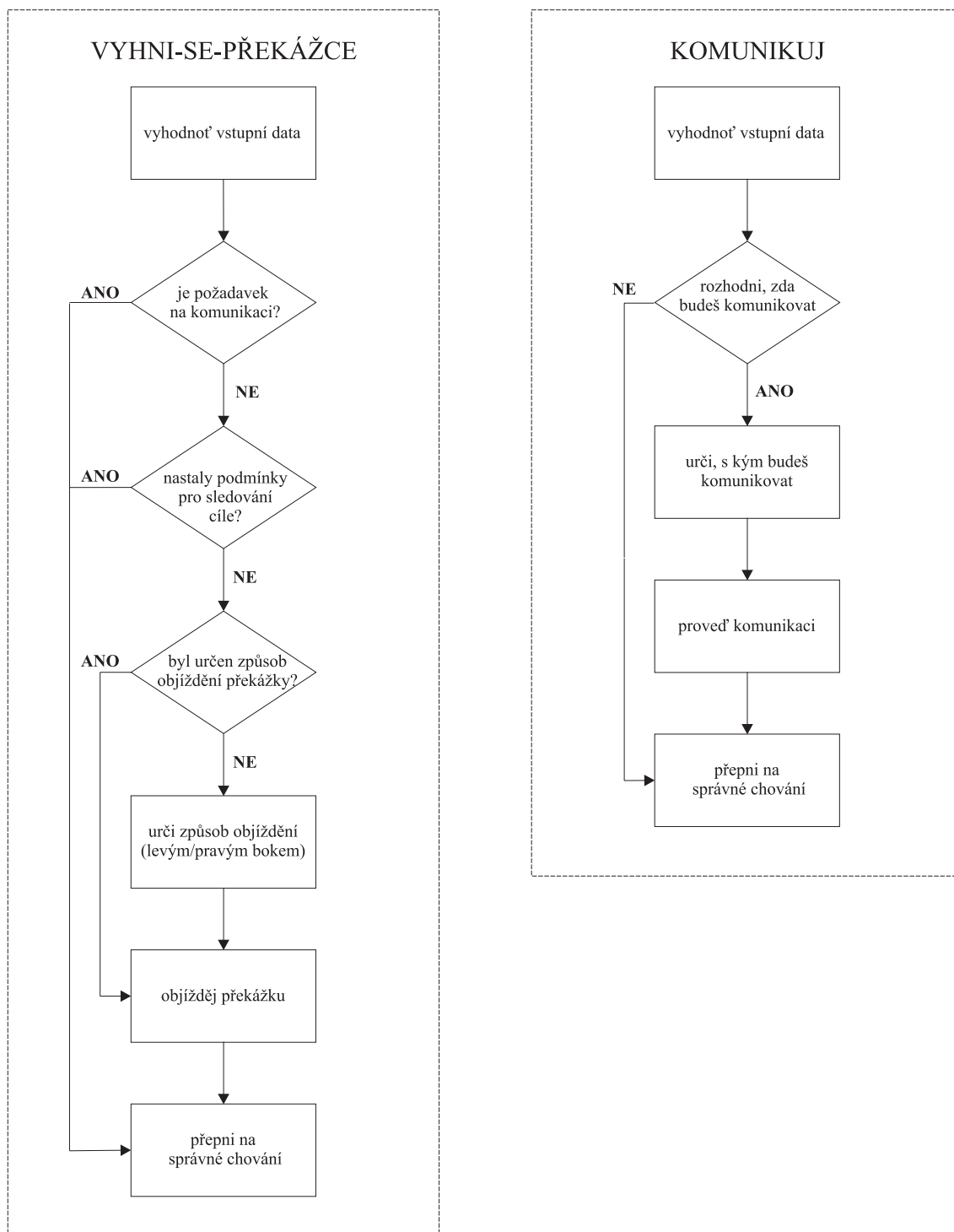
a vnitřní stavové proměnné mobota jako:

- **FOldPosition** - předchozí poloha mobota,
- **FNewPosition** - nová poloha mobota ,
- **FActualAngle** - úhel natočení vzhledem ke globálním souřadnicím,
- **IR_array** - uchování stavu IR čidel,
- **FGoalPosition** - souřadnice cíle,
- **FCooperation** - kooperovat/nekooperovat,
- **FGoalUnknown** - cíl známý/neznámý,
- **FGoalDetected** - cíl nalezen/nenalezen
- atd.

Vyhodnocování IR čidel a detekce cíle je realizováno metodou **CheckIR**, která vyhodnocuje barvu odpovídajících pixelů v bitmapě, která definuje okolní prostředí.



Obrázek 3.3: Vývojové diagramy chování typu hledej-cíl a sleduj cíl



Obrázek 3.4: Vývojové diagramy chování typu vyhni-se-překážce a komunikuj

3.2 Simulace

3.2.1 Simulační program

Simulační program je vytvořen v objektovém prostředí C++ Builder 4.0 v operačním systému Windows 9x/NT. V následujícím textu si popíšeme, jak je simulační program koncipován a co všechno umožňuje. Začneme vizuálním prostředím, protože to je první, co uživatel po spuštění programu uvidí.

3.2.2 Vizuální prostředí

Celé vizuální prostředí simulačního programu je vytvořeno pomocí vizuálních komponent C++ Builderu, viz. obr. 3.5. Po spuštění simulačního programu se zobrazí okno o velikosti 800×600 bodů a není možné měnit jeho velikost. Proto je nutné, aby rozlišení obrazovky bylo minimálně 800×600 bodů.

Simulované prostředí

Největší část obrazovky zabírá rámeček o velikosti 400×300 bodů, kde se bude odehrávat simulace pohybu mobotů. 1 pixel odpovídá 1 cm, takže skutečné prostředí by mělo velikost 5×3 metry. Souřadný systém simulovaného prostředí je stejný jako souřadný systém zobrazování na počítači. Počátek souřadnic je tedy v levém horním rohu. X-ová souřadnice zleva doprava a Y-ová souřadnice odshora dolů. Informace o prostředí se načítají z vnějšího souboru, který je ve formátu BMP a musí mít uvedené rozlišení.

Objekty v prostředí

Volná plocha - černá barva,

Překážky - jakákoli jiná barva, kromě barev mobotů a hledaného objektu,

Moboti - mobot jsou reprezentováni objektem ve tvaru osmiúhelníku v barvách červená, zelená, modrá a fialová. Orientace mobota je označena žlutou čarou vedenou z geometrického středu mobota k jeho okraji,

Hledaný objekt - je označen bílým kolečkem zhruba o velikosti mobota. Objekt se do prostředí umístí kliknutím na plochu prostředí.

Ovládací prvky simulátoru

Initialize - vyvolání dialogového okna pro umístění mobotů do prostředí a jejich úhlu natočení vzhledem k globálním souřadnicím prostředí,

Step - délka kroku mobota

Random Step - počet kroků ovlivňující změnu směru robota při náhodném pohybu,

Not Wall Contact - počet kroků, kdy mobot zareaguje na ztrátu kontaktu s překážkou.

Informační panely mobotů

Jednočíselný údaj - naposledy aktivované chování mobota,

IR čidla - prstenec IR čidel zobrazující stav okolního prostředí. Barvou červená/zelená je signalizováno překážka/volno,

Position x - x-ová souřadnice mobota,

Position y - y-ová souřadnice mobota,

Actual Angle - aktuální úhel natočení mobota,

Goal Distance - vzdálenost k cíli, je-li známa,

Step - prováděný krok.

Kapitola 4

Provedené experimenty

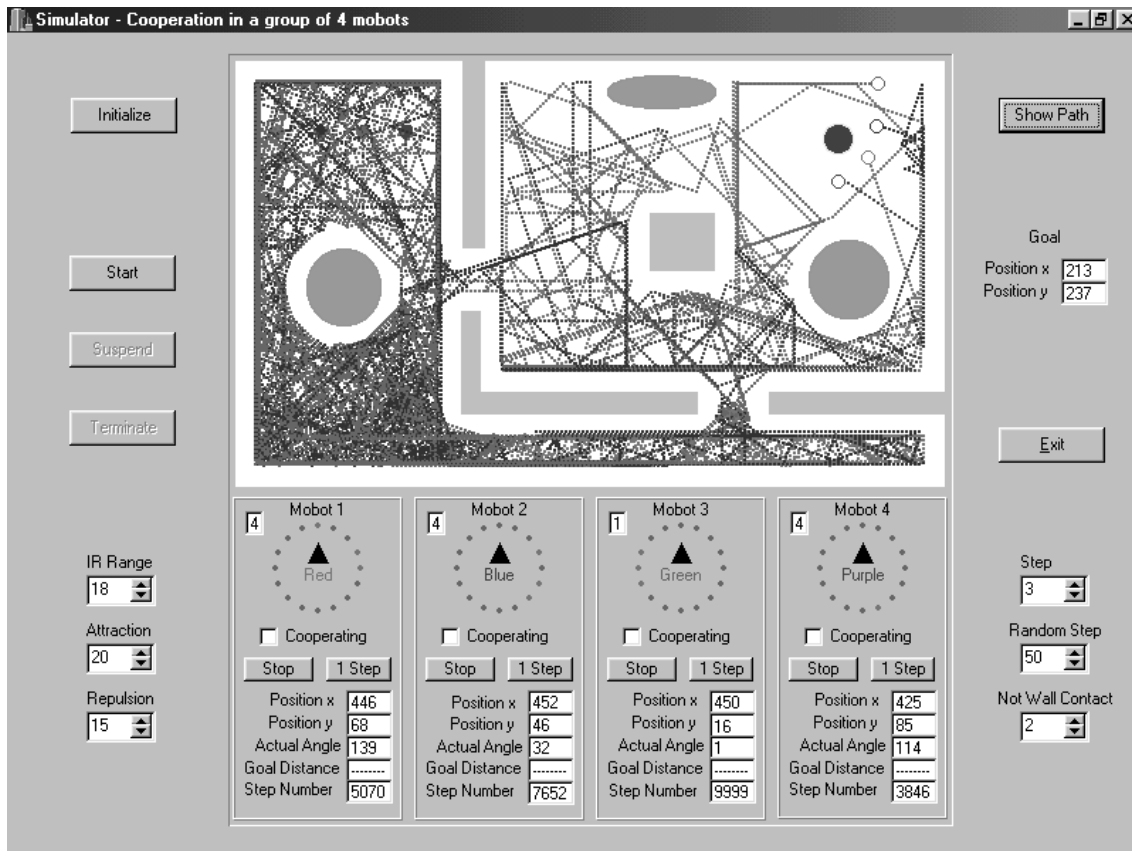
4.1 Popis experimentů

Díky simulačnímu programu, který jsem vytvořil, jsem mohl otestovat navržené algoritmy řízení mobotů a zkoumat jak se moboti budou chovat při různém nastavení parametrů ovlivňujících jejich chování. Jsou to parametry **IR Range**, **Attraction**, **Repulsion**, **Step**, **Random Step**, a **NotWallContact**. Experimentálně jsem určil takové nastavení parametrů, kdy se moboti spolehlivě vyhýbají překážkám a téměř vždy jsou schopni dosáhnout cíle. Jako příklad experimentů uvádím dva extrémní případy, kdy žádný z mobotů nekooperuje a je tedy nucen nalézt cíl vlastními silami, viz. obr. 4.1 a případ, kdy kooperují všichni moboti, každý z nich tedy okamžitě obdrží zprávu o nalezení cíle a jeho souřadnicích, jakmile některý z mobotů cíl lokalizuje, viz. obr. 4.2. Nastavení počátečních parametrů je v obou případech stejné a je vidět na uvedených obrázcích.

4.2 Zjištěné výsledky

Experiment č.1 ukazuje situaci, kdy žádný z mobotů nekooperuje. Jak je vidět z obrázku, cíl objevili pouze 3 moboti. Lze předpokládat, že poslední mobot by cíl po určité době také objevil, ale simulace je omezena na 10 000 kroků a jak je vidět, za tuto dobu je prohledán téměř celý prostor. Účinnost prohledávání prostoru závisí na parametru **Random Step**, který ovlivňuje počet kroků které mobot provede, než změni směr jízdy. Velikost tohoto parametru by měla být dostatečně velká, protože jinak by mobot kmital kolem určitého bodu a nikam by se nedostal, ale neměl by být ani moc velký, protože pak by mobot neustále narážel do zdí.

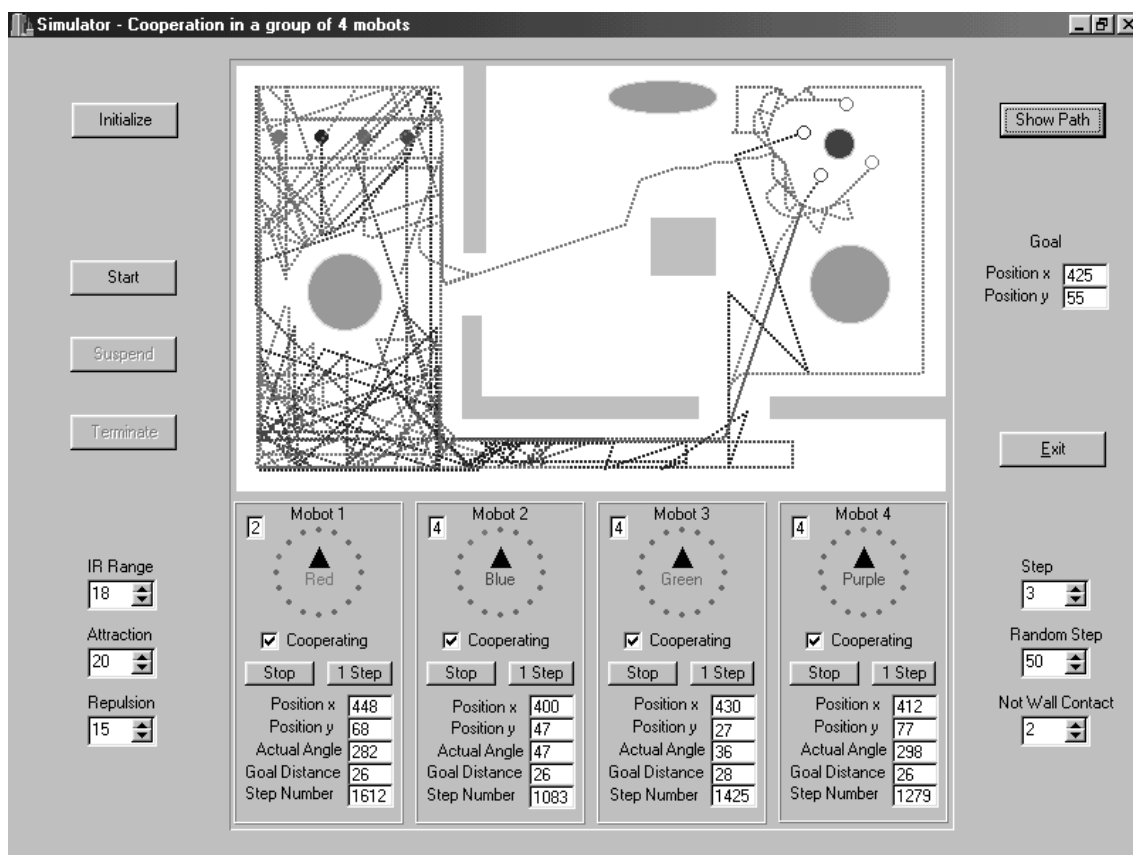
Experiment č.2 ukazuje případ, kdy všichni moboti kooperují. To znamená, že jakmile některý z mobotů objeví cíl, ostatní okamžitě vyrazí za ním a od této chvíle počet kroků nutných k dosažení cíle značně minimalizován. V tomto případě dosáhl poslední mobot cíle již po 1083 krocích. Narazí-li při jízdě na cíl mobot na překážku, je schopen se jí poměrně inteligentně vyhnout, protože zná polohu cíle, ale vzhledem



Obrázek 4.1: Experiment č.1 - Žádný mobot nekooperuje

k tomu, že má informace jen o nejbližším lokálním okolí, není schopen rozhodnout, zda je výhodnější objíždět překážku levým nebo pravým bokem. Proto je při detekci překážky směr objíždění určen náhodně. Úspěšnost mobota při objíždění překážek závisí také na strukturovanosti prostředí. Je-li prostředí příliš strukturované, může se stát, že mobot se zacyklí a cíle již nedosáhne. Vzhledem k tomu že cíl má velikost zhruba odpovídající velikosti mobota, simulace většinou končí tím, že moboti se seskupí dokola kolem cíle.

Porovnáme-li oba obrázky, je vidět, že prozkoumaná plocha a také počet kroků nutných pro dosažení cíle všemi roboty jsou v obou případech značně rozdílné. I když je tedy navržená aplikační úloha poměrně jednoduchá a komunikace i kooperace probíhá na nejnížší možné úrovni, je naprosto zřejmé, že kooperativní jednání mobotů má smysl a efektivita konání mobotů se zvyšuje.



Obrázek 4.2: Experiment č.2 - Všichni mobot kooperují

Závěr

Náplní této diplomové práce bylo seznámení se s problematikou kooperace ve skupinách mobilních robotů s řídicími systémy na bázi etologických principů. Tato část práce mi zabrala poměrně mnoho času protože oblast mobilní robotiky byla pro mě nová a neměl jsem v této oblasti patřičný rozhled. Díky tomu, že škola nám již několik let zprostředkovává připojení k internetu mohl jsem prozkoumat desítky projektů týkající se dané problematiky, které jsou řešeny na různých univerzitách po celém světě. V kapitole 1.4 jsem uvedl alespoň dva z nich, které mě zaujaly.

Výsledkem tohoto snažení měl být návrh a realizace aplikační úlohy demonstrující užitečnost vzájemné kooperace a komunikace ve skupině mobilních robotů. Aplikační úloha byla navržena pro skupinu čtyř mobilních robotů, kteří se v posledních letech vyvíjejí ve skupině mobilní robotiky pod vedením doc. Pavla Nahodila. Vzhledem k tomu že v průběhu řešení úlohy nebyli ještě moboti plně funkční a neměli potřebné senzorické vybavení pro realizaci aplikační úlohy, bylo na pokyn vedoucího diplomové práce od realizace na reálných modelech ustoupeno. Alternativním řešením tedy bylo vytvoření simulačního programu, který umožnil testovat funkčnost navržených algoritmů. Simulační program i se zdrojovými kódy je na přiložené disketě.

Navržený řídicí systém vychází z principu vrstvené architektury. Jeho jádrem je soubor chování, která jsou aktivována v závislosti na podnětech z okolního prostředí. Jedná se o systém reaktivní, který poskytuje bezprostřední reakce na okolní podněty, proto je také vhodný k realizaci úlohy kooperativního jednání.

Navržená aplikační úloha je poměrně jednoduchá. Jedná se o komunikaci a kooperaci na nejnižší možné úrovni, kdy skupina mobilních robotů hledá cílový objekt v předem neznámém omezeném prostředí s překážkami a jejich úkolem je dostat se do cíle v nejkratším možném čase. Moboti si nevytváří žádnou globální mapu prostředí, ale mají informaci o okamžité poloze vzhledem ke globálním souřadnicím. Pokud budou moboti prohledávat prostředí náhodně a nebudou spolupracovat, je velká šance, že cíl najdou, ale rozhodně to nebude optimální řešení z hlediska velikosti prohledaného prostoru. Pokud moboti budou spolupracovat a předávají si informaci o nalezení cíle a jeho souřadnicích, celá situace se tím změní ve prospěch celé skupiny. Díky znalosti souřadnic cílové pozice, mohou moboti koordinovat svůj pohyb vzhledem k cíli a efektivně se tak vyhýbat překážkám, které se na cestě k cíli nachází. Navržený algoritmus tohoto jednání byl simulačně ověřen a bylo dosaženo

uspokojivých výsledků, viz. kapitola 4 o provedených experimentech.

Situace by byla mnohem zajímavější, kdyby moboti byli schopni komunikace a kooperace na vyšší úrovni a mohli činit rozhodnutí na základě nějakých motivačních faktorů jako např. nedostatek energie apod. Tato problematika mě zaujala natolik, že bych se jejímu výzkumu rád věnoval v rámci doktorandského studia na této škole.

Literatura

- [1] Franck, D.: *Etologie*. Karolinum – vydavatelství Univerzity Karlovy, Praha 1996.
- [2] Lorentz, K.: *Základy etologie*. Academia – nakladatelství Akademie věd České republiky, Praha 1993.
- [3] Havel, I.: *Robotika - Úvod do teorie kognitivních robotů*. SNTL, Praha, 1980.
- [4] Mařík, V., Štěpánková, O., Lažanský, J. a kol.: *Umělá Inteligence 2*. Academia – nakladatelství Akademie věd České republiky, Praha 1997, (str. 142 - 177).
- [5] Buchtelová, R., Confortiová, H., Červená, V. a kol.: *Akademický slovník cizích slov*. Academia – nakladatelství Akademie věd České republiky, Praha 1998.
- [6] Balch, T., Arkin, R.C., *Motor Schema-based Formation Control for Multiagent Robot Teams*. IEEE Transactions on Robotics and Automation, 1999.
- [7] Barnes, D.P., Ghanea-Hercock, R.A., Aylett, R.S., Coddington, A.M.: *Many hands make light work? An investigation into behaviourally controlled co-operant autonomus mobile robots*.
- [8] Roubal, M.: *Základní programové vybavení kognitivních robotů*. Diplomová práce ČVUT - FEL, K335, Praha 1996.
- [9] Šafář, M.: *Alternativní řízení mobota MARVINA II*. Diplomová práce ČVUT - FEL, K335, Praha 1994.
- [10] Gruncl, M.: *Rozšiřující SW robotů I*. Diplomová práce ČVUT - FEL, K335, Praha 1995.
- [11] Kurzveil, J.: *Základní SW vybavení chodícího autonomního mobota*. Diplomová práce ČVUT - FEL, K335, Praha 1996.

INTERNET:

- [12] Brock,D., Montana,D., Ceranowicz,A.:*Coordination and Control of Multiple Autonomous Vehicles.*,1992.
<http://www.ai.mit.edu/people/dlb/papers/ieee/www/ieee.html>
- [13] Crombie,D.:The Examination and Exploration of Algorithms and Complex Behaviour to Realistically Control Multiple Mobile Robots,1997.
<http://www.ozemail.com.au/dcrombie/project/>
- [14] Raynolds,C.:*Boids*,1999.
<http://www.red.com/cwr/boids.html>
- [15] Unsal,B.:SELF-ORGANIZATION IN LARGE POPULATIONS OF MOBILE ROBOTS,1993.
<http://armyant.ee.vt.edu/unsalWWW/cemsthesis.html>
- [16] Jung,D.:A Behaviour Architecture for Cooperating Mobile Robots,1997.
http://wwwsyseng.anu.edu.au/rsl/rsl_cooprob.html

Příloha

Disketa

K diplomové práci je přiložena disketa s následujícím obsahem:

- Informační soubor READ_ME.TXT
- Adresář SOURCE se zdrojovými kódy simulačního programu
- Adresář SIMULATOR s vlastním spustitelným simulačním programem