

Bibliografická citace

DVOŘÁKOVÁ, I. *Plánovací expertní systém pro mobilního robota*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2008. 60 s. Vedoucí bakalářské práce Ing. Jan Valenta

Prohlášení

„Prohlašuji, že svou bakalářskou práci na téma "Plánovací expertní systém pro mobilního robota" jsem vypracovala samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušila autorská práva třetích osob, zejména jsem nezasáhla nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědoma následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.“

V Brně dne : 26.05.2008

Podpis: _____

Poděkování

Za odbornou pomoc děkuji mému vedoucímu bakalářské práce Ing. Janu Valentovi. Dále bych chtěla poděkovat všem mým blízkým, díky nimž jsem si mohla odpočinout a přijít na jiné myšlenky.

2. Obsah

2. Obsah	4
3. Úvod do plánovacích expertních systémů	7
3.1. <i>Historie.....</i>	7
3.2. <i>Použití expertních systémů.....</i>	7
3.3. <i>Definice</i>	8
3.4. <i>Členění</i>	8
3.5. <i>Základní části plánovacího expertního systému.....</i>	9
4. Přístup k tvorbě plánovacích expertních systémů.....	11
4.1. <i>Pojem plánování.....</i>	11
4.2. <i>Tvorba plánovacího expertního systému</i>	11
4.3. <i>Životní cyklus plánovacích expertních systémů.....</i>	12
4.4. <i>Problematika plánovacích expertních systémů</i>	13
4.4.1. <i>Reprezentace báze znalostí.....</i>	14
4.4.2. <i>Tvorba kroků a hledání cílového řešení</i>	14
4.4.3. <i>Složité úlohy a jejich zjednodušení.....</i>	15
4.4.4. <i>Neurčitost</i>	16
4.4.5. <i>Dopředné řetězení</i>	16
4.4.6. <i>Stavový prostor a jeho prohledávání (do šířky)</i>	17
5. Druhy plánovacích expertních systémů	18
5.1. <i>Druhy plánovacích metod a technik.....</i>	18
5.2. <i>Využití plánovacích expertních systémů.....</i>	20
6. Plánovací expertní systémy pro mobilní roboty	20
6.1. <i>Robotika, mobilní roboti</i>	20
6.2. <i>Způsoby plánování cesty-trasy pro robota.....</i>	21

6.3. <i>Optimalizace cesty</i>	26
6.6. <i>Principy řídicích a plánovacích systémů pro mobilní roboty</i>	28
7. Příklad	31
8. Robotický fotbal kategorie MIROSOT	36
8.1. <i>Význam robotického fotbalu</i>	36
8.2. <i>Historie robotického fotbalu</i>	36
8.3. <i>Kategorie robotického fotbalu</i>	37
8.4. <i>Technické parametry kategorie MIROSOT</i>	37
8.5. <i>Základní popis hry a její pravidla</i>	38
8.6. <i>Systém robotického fotbalu</i>	39
8.7. <i>Celkový náhled na hru</i>	41
9. Druhy modulů a systémů pro robotický fotbal MiroSot	42
9.1. <i>Druhy modulů</i>	42
9.1.1. <i>Moduly pro vytváření představy o okolním prostředí</i>	43
9.1.2. <i>Moduly pro vytváření komunikace</i>	44
9.1.3. <i>Moduly pro plánování</i>	45
9.1.3. <i>Moduly pro vnímání a chápání hry</i>	45
9.1.5. <i>Moduly pro ovládání pohybu</i>	46
9.2. <i>Druhy systémů</i>	47
9.3. <i>Zařazení modulů</i>	48
10. Plánovací expertní systém pro robota robotického fotbalu	48
10.1. <i>Herní strategie</i>	48
10.2. <i>Interakce mezi spoluhráči a protihráči</i>	49
10.3. <i>Cíle plánovacího expertního systému pro robotický fotbal</i>	50
10.4. <i>Moduly, kde není plánování potřeba</i>	50
10.5. <i>Nepřímo ovlivňované moduly</i>	50

10.6. Přímou ovlivňované moduly.....	51
10.7. Závěrem.....	51
11. Koncepce plánovacího expertního systému pro robotický fotbal	52
12. Seznam obrázků.	55
13. Seznam použité odborné literatury.	56

3. Úvod do plánovacích expertních systémů

3.1. Historie

Expertní systémy jsou neodmyslitelnou částí součástí umělé inteligence. Jejich počítačový vývoj se datuje již od roku 1965, poté co docházelo k dalšímu vývoji a zdokonalování, byly od roku 1981 nasazeny pro komerční využití.

1965-1970	počáteční fáze (DENDRAL a MACSYMA)
1970-1975	výzkumné prototypy (MYCIN, PROSPECTOR)
1975-1981	experimentální nasazování (PUFF, SACON, SADUCEUS)
Od roku 1981	komerčně dostupné systémy (XCON, XSEL, ADVISOR)

Zatím co v první generační fázi umožňovaly jen jeden způsob reprezentace znalostí, malé omezené schopnosti vysvětlování, přístup ke znalostem pouze od expertů, ve druhé fázi se již projevují modulární a víceúrovňové báze znalostí, hybridní reprezentace znalostí, zlepšení vysvětlovacího mechanismu a prostředky pro automatizované získávání znalostí. [1]

3.2. Použití expertních systémů

Na začátku je důležité si uvědomit, zda se nám používání expertních plánovacích systémů a vůbec jakýchkoliv expertních systémů vyplatí. Vytvoření takového systému, není levné a zabere spousty času. Proto je dobré, než začneme o expertním plánovacím systému vůbec uvažovat, zvážit zda je opravdu potřeba. Je-li systém opravdu potřeba musí splňovat některé z uvedených předpokladů:

- firma je závislá na experech, kteří jsou jen těžce nahraditelní nebo jsou drazí
- úloha je složitá rozsahem či neurčitostí a jiná metoda buď není nebo je pomalá
- expertů je v dané problémové oblasti málo a jsou zaměstnání svými povinnostmi
- daný problém vyžaduje složitější znalosti, avšak musí existovat expert, který tyto znalosti bude mít a bude je schopen předat vhodným způsobem ES
- dokážeme-li říci, že investice do ES bude výhodnější než jiná metoda
- problém není možné vyřešit pomocí běžných počítačových metod
- ve firmách se ekologicky či zdravotně nebezpečnými provozami

- pro řešení vysokém stupni neuspořádanosti problému
- je-li těžko dodržitelný postup i přes odpovědné plnění

3.3. Definice

Expertní systém je počítačový program, který dokáže napodobovat rozhodovací činnost experta při řešení složitých úloh. Dokáže vhodně využívat znalostí přebraných od experta za účelem dosažení kvalitního rozhodnutí na expertově úrovni v dané problémové oblasti. Využívá znalostí přebraných od špičkových odborníků, nedopouští se však lidských omylů. Jeho cílem je dosáhnouti co nejlepší odezvy na reálná data.

Plánovací expertní systém se zaměřuje na typy úloh u kterých známe počáteční stav a cíl řešení u nichž se snaží najít nejlepší posloupnost kroků, které změní počáteční stav na stav cílový. Je schopen vytvořit tuto posloupnost ještě před zásahem do řešeného problému. [1] [3]

3.4. Členění

Expertní systémy lze rozdělit z hlediska : obecnosti, způsobu reprezentace znalostí a charakteru řešených úloh.

Z hlediska obecnosti rozlišujeme expertní systémy:

- Prázdné - neobsahují žádné závislé problémové části tzn. bázi znalostí a bázi dat.
- Řešící konkrétní problém - obsahují bázi znalostí, bázi dat i řídicí mechanismus.

Používá se pro řešení určité problémové oblasti. Jeho architektura, řídicí mechanismus a typ reprezentace znalostí jsou těsně spojeny s danou oblastí problému.

Z hlediska způsobu reprezentace znalostí dále rozlišujeme expertní systémy:

- Založené na pravidlech báze znalostí - tvoří jej množina produkčních pravidel typu situace – akce IF – THEN.
- Založené na rámcích¹

¹ **Rámce** jsou struktury reprezentující pravidelně se opakující situace. Je to prostředek reprezentace znalostí vycházející z toho, že se pro řešení nových a rozbor stávajících řešení používá rámců získaných z předchozích zkušeností.

- Založené na logickém programování - kde se znalosti vyjadřují formou logických formulí.

- a další – např.: Sématické sítě, objekty

Z hlediska charakteru řešených úloh je dále členíme na systémy:

- Diagnostické - porovnávání a vyhodnocování předem stanovené hypotézy a dokázat určit, které z nich budou nejlépe odpovídat reálným datům.

- Plánovací - cíl řešení a počáteční stav. Úkolem je nalezení optimální posloupnosti kroků, díky nimž dosáhneme stanoveného cíle.

- Hybridní - jsou kombinací diagnostických a plánovacích systémů. [1]

3.5. Základní části plánovacího expertního systému

Hlavní části plánovacího exp. systému jsou: báze znalostí, báze dat, řídicí (inferenční) mechanismus, vysvětlovací a komunikační modul, generátor výsledků.

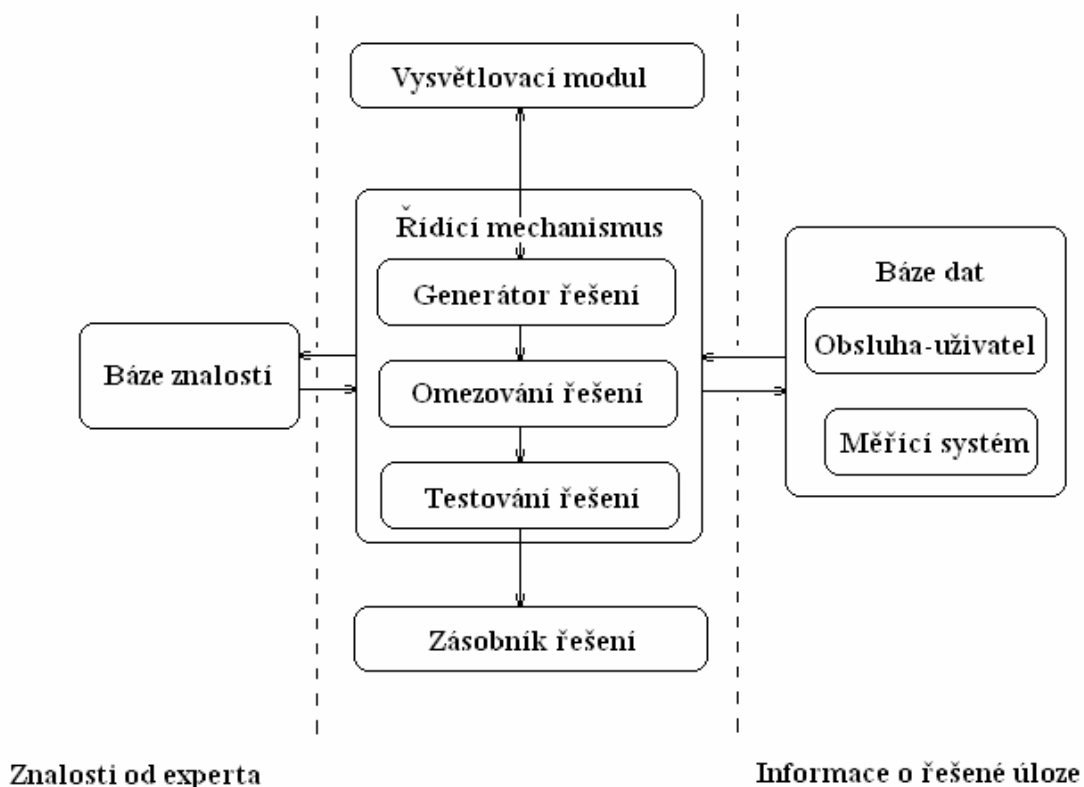
Báze znalostí– bez báze by neměly expertní systémy smysl. Je potřeba této části věnovat vysokou pozornost, protože čím specifičtější a kvalitnější báze znalostí je tím je i lepší expertní systém. Ukládají se zde všechny znalosti experta. Báze znalostí má podobu databáze, obsahuje velké množství obecných i speciálních znalostí, které jsou potřebné k řešení daného problému. Speciální znalosti představují obecný souhrn pravidel většinou ve formě heuristické² informace, které musí být schopen expertní systém využívat.

Báze dat - oblast, kde se shromažďují potřebná data o daném problému, jenž expertní systém potřebuje k řešení úkolu. Tyto data zapisuje buď uživatel expertního systému, pomocí dialogového režimu s počítačem, nebo se jedná o data automaticky měřené (teplota, tlak, aj.). Jedná se o dialog nezkušeného uživatele-odborníka s expertem. Data vznikají na základě řešení zadaného problému.

Řídicí mechanismus - programovatelný modul, předem udávající strategii využití znalostí z báze znalostí, vytváří komunikaci mezi bází datovou a znalostní. Je schopný

² **Heuristické informace.** Nejisté, nedokázané znalosti. Nezaručující nalezení správného řešení, ale pomáhající při řešení určitých problémů.

odvozovat nové znalosti díky těm, které již existují. Vytváří soubor pravidel a zjišťuje jestli je přiložená fakta splňují. Ovlivňuje výběr vhodných operátorů, vytváří zásobník možných řešení testováním shod vygenerovaných řešení s databází dat. Důležitou schopností řídicího mechanismu je zpracovávání neurčitostí.



Obrázek 1: Model plánovacího systému. [13]

Vysvětlovací modul - objasňuje řešení, ke kterému expertní systém dospěl. Poskytuje vysvětlení uživateli pomocí komunikačního modulu, který zprostředkovává a zabezpečuje komunikaci mezi uživatelem a systémem ve všech fázích činnosti.

Generátor řešení – jedna z hlavních částí plánovacího expertního systému. Vytváří kombinace řešení posloupností operátorů. Automaticky je generuje, testuje a kombinuje v závislosti na omezovacích pravidlech databáze znalostí. [1] [2]

4. Přístup k tvorbě plánovacích expertních systémů

4.1. Pojem plánování

Plánování je vytvoření určité posloupnosti činností, aniž by bylo nutné jakýkoliv krok vykonat během plánování. Jak už bylo řečeno výše, plánovací systém má za úkol najít nejvhodnější posloupnost takových činností (kroků), které nás mají dovést ke správnému řešení. Jedná se o jeden z nejobtížnějších úkolů v umělé inteligenci. Plánovací systém je schopen samostatně vytvářet rozhodnutí a nové scénáře. Je schopen uvažovat o velkém množství alternativ a podporuje strategické plánování. [3]

4.2. Tvorba plánovacího expertního systému

Tvorbou expertních systémů se zabývá znalostní inženýrství, což je obor v rámci umělé inteligence, zabývající se tvorbou expertních systémů, jejich aplikací a začleněním do jiných softwarových produktů. Nejvýznamnější částí tohoto oboru je naplňování expertních systémů znalostmi (kódování, testování, třídění, katalogizací dostupných metod a technik reprezentace znalostí, aj.) Plánovací expertní systém musí obsahovat kromě základních částí samotného plánovacího i další části, jako je hardware a software. Také je nutné navrhnout uživatelské rozhraní.

Existuje několik nástrojů pro tvorbu expertních systémů:

- prázdný expertní systém
- speciální programové prostředí - CLIPS, Prolog, Lips, aj....
- obecná programová prostředí - C, C++, Delphi, Pascal, aj...

Znalostní inženýrství se velmi podobá inženýrství softwarovému, rozdíl však spočívá ve znalostech. Znalostní inženýrství se zabývá znalostmi, jejichž vlastnosti a množství nelze na začátku dobře odhadnout. Znalosti jsou zde nepřesné, špatně popsány a mohou se rozšiřovat. Díky tomu vznikají v počátcích tvorby expertních systémů problémy s tvorbou návrhu a s odhadem potřebného úsilí. Abychom je alespoň nějak z počátku zredukovali používáme při tvorbě systémů životního cyklu. [1]

4.3. Životní cyklus plánovacích expertních systémů

Model životního cyklu se skládá z následujících částí:

1. Rozbor řešeného problému.
2. Přesný výpis požadavků.
3. Předběžný návrh.
4. Počáteční vzor a vyhodnocení.
5. Konečný návrh.
6. Získávání znalostí a jejich reprezentace.(opakuje se pro všechny pod-části systému)
7. Prověřování správnosti – testování.(opakuje se pro všechny pod-části systému)
8. Změny.(opakuje se pro všechny pod-části systému)
9. Údržba. [1]

4.3.1. Rozbor řešeného problému

V této fázi se posuzuje jaká metoda či technika plánovacího expertního systému bude pro řešení daného problému nejvhodnější. Posuzování se rozděluje na: vhodnost použití systému a dostupnost zdrojů.

Vhodnost použití systému řeší zda problém skutečně existuje, zda k řešení můžou být použity lidské znalosti, zda jsou to znalosti převážně dokázané či naopak heuristické, jsou-li znalosti dobře chápány, zda jsou vstupní data kompletní a přesná či naopak a jestli náhodou neexistuje lepší metoda než tvoření expertního plánovacího systému a vůbec zda se vytvoření systému vyplatí.

Dostupností zdrojů vzniká další soubor otázek na něž hledáme odpověď např.: zda bude na realizaci systému dost času a financí, zda existuje dostatek expertů, kteří nám dodají znalosti a zda existují i jiné zdroje znalostí. [1]

4.3.2. Přesný výpis požadavků

Uvádí se podstata problému, cíle projektů, profily uživatelů, požadované funkce systémů, požadované omezení (hardwarové, slučitelnost s jinými produkty, spolehlivost, bezpečnost, rychlost, aj.) a ostatní požadavky (ověřování správnosti, dokumentace, aj.). [1]

4.3.3. Předběžný návrh.

Předběžný návrh by měl obsahovat styl reprezentace znalostí, metody usuzování, výběr systému (mezi komerčně dostupnými systémy a vytvářením nového), seznam expertů, znalostních inženýrů a vedoucích projektů a nároky na vývojový tým.

[1]

4.3.4. Počáteční vzor a vyhodnocení

Vytváří se vzorová podoba fungujícího finálního systému. Cílem je rychlé vytvoření tohoto vzoru za použití buď komerčních prázdných systémů, nebo prostředků Prolog³, aj. U tohoto vzoru se provede vyhodnocení a na jeho základě se potvrdí nebo změní všechna rozhodnutí, je dobré když má dostatečnou část znalostí a dobré rozhraní, aby uživatelé mohli posoudit jeho použitelnost. [1]

4.3.5. Získávání znalostí a jejich reprezentace

Je to nejpracnější a nejdéle trvající část expertních systémů. Informace se zadávají do báze znalostí pod určitými organizačními pravidly. [1]

4.4. Problematika plánovacích expertních systémů

Při tvorbě plánovacích expertních systémů se objevují různé komplikace. Zaměříme se nyní především na:

- reprezentaci znalostí
- složité úlohy a jejich zjednodušení
- neurčitost
- řetězení (dopředné)
- prohledávání stavového prostoru (do šířky)

³ **Prolog.** Logický programovací jazyk. Patří mezi deklarativní jazyky, ve kterých programátor popisuje pouze cíl výpočtu, přičemž přesný postup, jakým se k výsledku program dostane, je ponechán na libovůli systému.

4.4.1. Reprezentace báze znalostí

Když už máme vytvořenou bázi znalostí je potřeba ji i účinně reprezentovat. Na reprezentaci báze znalostí závisí efektivita expertního systému. Je důležité abychom mohli bázi znalostí kdykoliv aktualizovat či doplňovat o nově získané a rozšiřující se vědomosti experta nebo z ní naopak odstranit neefektivní a nadbytečné informace – tzv. modularita báze znalostí. Modularita má organizační výhody, jelikož provedené změny se neodráží do celé báze znalostí, zároveň ale komplikuje prohledávání báze protože měnící se informace může být obsažena na více místech báze. Nejnovější expertní systémy již dnes dokáží pracovat s využitím více bází znalostí jež sdílejí stejnou datovou strukturu. Pro reprezentaci znalostí se používá pravidel, rozhodovacích stromů(souvislý neorientovaný graf bez kružnic), objektů, matematické logiky, scénářů a rámců.

4.4.2. Tvorba kroků a hledání cílového řešení

Řešení plánovacích expertních systémů je založeno na vytvořených postupech. Postupy se skládají z kroků (operandů) . Krok se dá nazvat částečným řešením. Na začátku máme informace (z báze znalostí), ty nám pomohou vybrat nejvhodnější pravidla pro první krok. Většinou se toho docílí tak, že se určí množina všech rozdílů mezi aktuálními a cílovými stavy.A pomocí informací se vybere jedno z více možných pravidel (nejvhodnější). Vybrané pravidlo se aplikuje a poté se přepočítá nový stav řešené úlohy. U tohoto řešení je potřeba zjistit, zda již náhodou není řešením cílovým. Úspěšným řešením rozumíme takové řešení, které dokáže změnit původní stav úlohy na cílový. Postup hledání dalších kroků se opakuje dokud se cílové řešení nenalezne. Plánovací expertní systém však musí být schopen poznat i částečně správné řešení (rozeznat částečné řešení od správného) a to následně upravit pro získání správného řešení. Pro tyto úpravy se používá řada speciálních akcí a technik, umožňujících správné řešení naléznout. Způsobů je hned několik.

Nejzákladnější metodou při zjištění částečného řešení, je zapomenout na právě provedené kroky, které nás k takovému řešení dostaly a začít s hledáním řešení

jiného. To je však dost neefektivní způsob a zdaleka ne vždy vede ke správným výsledkům. [3]

Dalším možným řešením je rozebrání a porovnání vzniklého navrhnutého řešení s řešením cílovým. Zde je důležité si uvědomit, že má-li částečné řešení alespoň nějaký smysl, pak bude rozdíl mezi cílovým stavem a stavem částečného řešení menší, než mezi cílovým stavem a stavem předchozím(počátečním).

Třetím způsobem nalezení správného řešení z řešení částečného je ve využití znalostí o tom co je špatně a tuto část podle dostupných informací opravit.

Nejúčinnější metodou však bývá nic neopravovat a ponechat všechna částečná řešení až do posledního možného okamžiku, čímž získáváme mnohem více doplňkových informací , jenž využijeme k dokončení řešení, přičemž zamezíme vzniku sporných situací. [3]

4.4.3. Složitě úlohy a jejich zjednodušení

Ve většině případech plánovacích expertních systémů se jedná o složitější úlohy. Pokud bychom neměli možnost si takovou úlohu zjednodušit může se pro nás stát neřešitelnou, museli bychom totiž brát v úvahu velké množství dat a počítat s nimi v každém kroku při vytváření plánovacího postupu. Existují dva způsoby jak složitou úlohu zjednodušit:nepře počítávat stav úlohy a rozložit úlohy na podskupiny.

Nepře počítávání stavu úlohy. Dobrým krokem ke zjednodušení složitěho úkolu je přestat pře počítávat celý stav úlohy na nový, při každém kroku ve stavovém prostoru. Místo toho se zaměřit jen na zaznamenávání změn stavu úlohy. Změny popisuje tzv. „rámec akce“. Použitím rámců se snažíme věnovat pozornost pouze takovým částem stavů úlohy , které budou příslušnou akcí ovlivněny.

Rozložení na podskupiny úloh. Pokud využijeme této možnosti zjednodušení, pak je potřeba aby vzniklé podskupiny byly minimálně závislé na ostatních (snažíme se o co nejmenší výskyt vazeb mezi podskupinami) viz. Nelineární plánování kap.5.2.

[3]

4.4.4. Neurčitost

Neurčitost je jedna z nejdůležitějších složek expertních systémů. Objevuje se u složitějších systémů a je zapříčiněná nedokonalostí dat v datové i znalostní bázi. Některá data mohou chybět, jiná nemusí být spolehlivá a přesná a nemusí platit ve všech případech. Neurčitost navíc vzniká díky různým druhům chyb, jako jsou například chyby: lidské, systematické, náhodné, usuzovací, při měření, dvojsmyslnost aj. Existuje několik metod jak s neurčitostí pracovat:

Faktory jistoty - Cílem je zredukovat slabiny týkající s pravděpodobnosti. Jedná se o míru důvěry a nedůvěry nějaké informace.

Bayesovský přístup - patří mezi nejstarší a nejlepší metody, využívající k řešení podmíněné pravděpodobnosti⁴.

Využití fuzzy - přístupy ke zpracování neurčitosti využívají tzv. Fuzzy množiny. Fuzzy logika je mnohem vhodnější pro řadu reálných rozhodovacích úloh.

Dempsterova-Shaferova teorie - předpokládá, že existuje tzv. prostředí⁵ obsahující X úplných elementů, jenž se vzájemně nepřekrývají svým významem. Může zde docházet ke změnám věrohodnosti. Na rozdíl od klasické pravděpodobnosti nepředpokládá automatický vztah mezi neznalostí a věrohodností, ale používá případy propojené s podmnožinami prostředí ke kterým jsou věrohodnosti připojené. [1]

4.4.5. Dopředné řetězení

Řetězec je množina násobných inferencí spojující řešenou úlohu s jeho řešením. Inference je odvozování určitých výroků z jiných. Řetězení máme buď dopředné nebo zpětné. V plánovacích expertních systémech se objevuje řetězení dopředné. V dopředném, nebo-li přímém řetězení směřuje inference od známých faktů k závěrům z nich plynoucím. Samotné dopředné řetězení však není příliš efektivní. Systém v každém kroku musí opakovaně porovnávat všechna pravidla s fakty. Proto byl vytvořen algoritmus Rete. Tento algoritmus úspěšně snižuje dobu porovnávání díky své síťové struktuře, kde jsou uloženy informace o ztotožnění pravidel s fakty.

⁴ Podmíněná pravděpodobnost popisuje jevy, které se navzájem ovlivňují.

⁵ Množina objektů které nás zajímají.

Jeho síťová struktura pracuje na principu, že uzly sítě jsou startovací uzel a uzel pro jednu podmínku a její souvislosti, kde je s každým uzlem spojena množina faktů a, se kterými je pravidlo ztotožněno. Hrany sítě jsou označeny pomocí vztahů mezi proměnnými vyskytujícími se v počátečním uzlu hrany. V řetězení se hledají možná řešení s využitím prohledávání stavového prostoru metodou do šířky. [1]

4.4.6. Stavový prostor a jeho prohledávání (do šířky)

Stavový prostor nám pomáhá efektivně řídit hledání možných výsledků. Danou řešenou úlohu jsme schopni rozdělit na počáteční a cílové stavy. Mezi těmito stavy jsme schopni pomocí použití určitých akcí přecházet. Stavový prostor si lze představit jako orientovaný graf (strom). Uzly grafu jsou stavy. Přechody nám udávají akce, které když vykonáme, dostaneme se z jednoho stavu do druhého. Nalezení řešení spočívá v nalezení cesty v grafu mezi počátečním a cílovým uzlem. Musely však být vyvinuty různé metody zjednodušující hledání ve stavovém prostoru, protože ten může být pro velké množství úloh příliš rozsáhlý, někdy i nekonečný. Tyto metody jsou vlastně algoritmy mající určité vlastnosti (zaručení nalezení existujícího řešení, optimalizaci pro zjištění nejlepšího vhodného řešení, schopnost zvládnout danou časovou a paměťovou složitost).

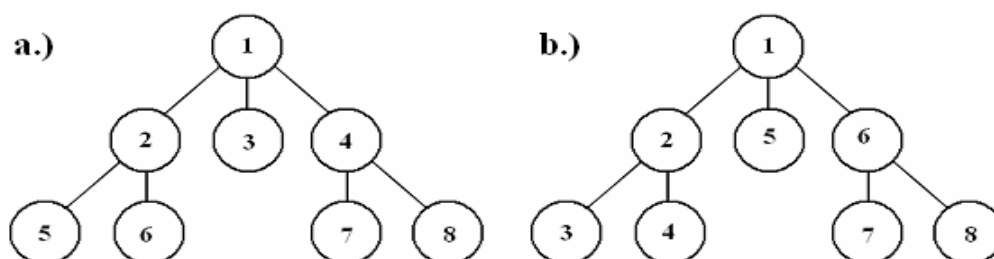
Tyto metody rozdělujeme na:

- informované
- prohledávání do šířky
- lokální prohledávání
- neinformované
- prohledávání do šířky
- prohledávání do hloubky
- obousměrné prohledávání

Informované metody - mají znalosti (heuristické informace) o stavovém prostoru a tak jsou schopni alespoň přibližně odhadnout kde se zrovna nacházejí od cílového stavu.

Neinformované metody - skupina metod, které nemají žádné vhodné znalosti o stavovém prostoru pro umožnění rychlejšího nalezení cesty. Tyto metody systematicky prochází všechny uzly, dokud nenaleznou řešení.

Nyní se zaměříme pouze na metodu prohledávání do šířky, která se týká dopředného řetězení a tudíž i plánovacích expertních systémů. Metoda neinformovaného prohledávání do šířky je grafovým algoritmem⁶ procházejícím postupně všechny vrcholy v dané množině souvislostí. Jak lze vidět na obrázku (4.4.1.) začíná algoritmus prohledávat od vrcholu ke svým sousedním uzlům.



Obrázek 2: Způsob prohledávání stavového prostoru neinformovaně. a.) do šířky
b.) do hloubky.

Tento proces opakuje až projde všechny souvislosti, přičemž každým uzlem projdeme jen jednou. (další -viz.odstavec 6.2.3. Grafové přístupy - grafové algoritmy).

K informovanému prohledání stavového prostoru lze použít prohledávání do šířky také, přičemž se využívá algoritmů, jenž nejprve prohledávají stavy které považují za bližší cíly. jedním z takových algoritmů je tzv. hladový algoritmus (popsaný v kap.6.3.) [3] [6]

5. Druhy plánovacích expertních systémů

5.1.Druhy plánovacích metod a technik

Zásobník cílů

Je jednou z prvních technik, která byla vyvinuta pro řešení složitých úloh s možným působením dvou nebo více činitelů. Celá funkce systému spočívá v tom, že uživatel zadá pouze počáteční a cílovou skupinu pravidel a množinu operátorů. Je-li výsledný cíl obsažen v počáteční skupině je úloha vyřešena. U složitějších úloh se

⁶ Přesný návod nebo postup řešení dané úlohy.

hledají rozdíly mezi skupinou platných a cílových pravidel a zároveň se hledá vhodný operátor k odstranění některých rozdílů. Nalezne-li se takový operátor, jsou jeho pravidla použitelnosti považována jako nový podcíl. A k němu se celý postup opakuje. Tato technika plánování řeší úlohy tak, že zpracovává jeden cíl po druhém. Plán vznikající generováním pomocí této metody obsahuje posloupnost operátorů na vyřešení prvního cíle a poté obsahuje celou posloupnost druhého cíle atd.(systém STRIPS). [3]

Nelineární plánování

Pracuje s podskupinami řešené úlohy. Na podskupinách se pracuje souběžně. Výsledná posloupnost akcí není vytvářena jako lineární řetězení dílčích částí plánu, které jsou určeny k úplnému řešení jednotlivých pod-úloh. Je nutné si uvědomit, že sice pracujeme s podskupinami, ale na konci je nutné jejich výsledky sloučit do konečného řešení. Při nelineárním plánování vycházíme z několika heuristických pravidel převzatých ze systému TWEAK. [3]

Hierarchické plánování

Řešíme-li složitou úlohu, pak pro její vyřešení většinou vytvoří plánovací systém dlouhý plán řešení. Aby byl tento způsob co nejvíce efektivní je potřeba vyloučit určité části úlohy do doby než je nalezeno řešení týkající se hlavních částí úlohy. Budeme-li daný úkol řešit klasickým způsobem bude to znamenat prohledávání obrovského stavového prostoru, naopak využijeme-li hierarchického plánování bude náš postup odpovídat postupu činností, které by volil expert. Hierarchické plánování je v praxi nejvíce požívanou metodou plánování (systém ABSTRIPS, HTN). [3][4][6]

Reakční plánování

Základem reakčních systémů je vyhnout se úplnému a komplikovanému plánování. Systém sleduje pouze aktuální situaci a k ní aktuální reakci. Reakční systém musí obsahovat bázi znalostí, aby díky ní mohl podniknout příslušné akce. Báze znalostí je zde obvykle vedená jako samostatný soubor. Reakční systém se od ostatních plánovacích expertních systémů nejvíce liší tím, že se nesnaží vytvořit

celou posloupnost akcí dříve než udělá první krok. Největší výhodou těchto systémů je jejich schopnost odolně pracovat v oblastech , kde je obtížné úplné modelování.. Dokáží se bez něj obejít a jsou schopny provádět akce na svém vnímání reálného světa. Mají rychlou odezvu, právě proto se hodí pro řešení úloh v reálném čase (využití pro mobilní roboty). [7]

5.2. Využití plánovacích expertních systémů

Expertní systémy se dnes využívají v nejrůznějších oblastech, ať již charakteru diagnostického (různé diagnostiky, identifikace systémů, aj.), plánovacího (technické návrhy, plánování, automatická syntéza programů) a nebo smíšeného (monitorování, aplikace při výuce). Zvláště v posledních letech se objevují stovky nových systémů použitelné v nejrůznějších oblastech k řešení nejrůznějších problémů (chemie, medicíny, genetiky, geologie, práva, výuky, mechaniky, matematiky aj.) Nejvíce z nich je zaměřeno na oblast medicíny.

6. Plánovací expertní systémy pro mobilní roboty

6.1. Robotika, mobilní roboti

Robotika tvoří významnou část využití umělé inteligence.

Mobilní autonomní robot je inteligentní stroj (řízený integrovaným systémem) schopný vykonávat samostatně zadané úkoly bez lidské pomoci v reálném prostředí. Jeho nejdůležitější vlastností je schopnost reagovat na změny okolí. Aby byl schopen tyto změny vnímat musí obsahovat řadu senzorů měřících vlastnosti okolí. Proto, aby naopak mobilní robot mohl ovlivnit své okolí potřebuje tzv. efektory, které mu umožní reagovat. Jsou to zařízení, jejichž stav můžeme ovládat pomocí změny napětí (např.:motory). Kromě senzorů a efektorů má mobilní robot ještě zdroj energie. Mobilní robot musí být dále schopen při řešení úlohy komunikovat s člověkem., musí si na základě znalosti cíle vytvořit plán řešení úlohy a vykonat jej. Tyto plány automaticky generuje pomocí algoritmičeského systému, který se obvykle nazývá plánovacím systémem. Rozlišujeme stupně volnosti robota a to: robot bodový

(reprezentace pouze bodem), robot kruhový (reprezentace kruhem), robot konvexní (přesně popsáný), robot konvexní s otáčením, robot obecný.

Plánovací systémy našly uplatnění ve všech mobilních robotech kde vyžadujeme autonomní chování. Mobilní roboti, jak už uvádí jejich název jsou roboti pohybující se v prostoru. Jejich možnost použití je ohromná. dají se použít všude tam, kde je potřeba částečná nebo úplná autonomní činnost. Objevují se v dopravě, ve vojenství, ale i třeba v dětských hračkách. Je možné je použít při prozkoumávání nebezpečných míst, při těžbě, sběru vzorků (robot Spirit na Marsu), při čištění a uklízení ploch, montáži, aj.). Naleznou řadu možností použití v oblastech lidem nebezpečných (různé kontaminované oblasti, jaderné elektrárny), lidem nedostupných (pod vodou, ve vesmíru), ve stísněných prostorech, pro provádění monotónních činností. [5] [7]

6.2. Způsoby plánování cesty-trasy pro robota

Plánováním cesty pro mobilního robota se snažíme docílit nalezení té nejvhodnější. Cestu hledáme ve vymodelovaném prostředí (pomocí grafických programů určených pro tvorbu obrazu skutečného prostředí). V prostředí se nachází překážky, ty dělíme na:

- překonatelné a nepřekonatelné
- statické a dynamické (statické překážky nemění svou polohu a velikost a dynamické je naopak schopné měnit jsou).

V prostředí se volí startovní a cílové body tratě robota v nichž se nesmí nacházet překážky. Úkolem robota je pak najít cestu (neprocházející překážkami) od startu k cíli. Rozlišujeme dva druhy prostředí a to diskrétní a spojitě.

Prostředí diskrétní je prostředí jenž si můžeme představit jako plochu rozdělenou pomocí mřížky na buňky. Robot se tak může pohybovat pouze v osmi směrech (pravoúhlých a diagonálních). Velikost buněk určuje přesnost s jakou počítáme. Buňky rozdělujeme na volné nebo obsazené překážkou. Pomocí plánovacího algoritmu najdeme posloupnost volných buněk po kterých se může robot pohybovat.

Spojité (složitější) prostředí již není takto rozděleno. Dá se přirovnat k prostředí kolem nás. Robot se může pohybovat jakýmkoliv směrem, ale to je náročnější na prohledávání terénu. Existuje několik metod, jak si poradit s nalezením cíle.

Trasu pro mobilního robota můžeme najít pomocí:

- algoritmů Bug
- přesného plánování
- plánování na mřížce
- pravděpodobnostního plánování

6.2.1 Algoritmy Bug

Je to způsob hledání cesty pro jednoduché automaty (obsahující jen malou paměť) . Datuje se k začátkům robotiky, kdy se řešily hlavně problémy s plánováním cesty ve známém prostředí. Robot je zde popsán bodem ve 2D rovině, je vybaven dotykovými senzory, zná svojí pozici (souřadnice) a pozici cíle. Robot je schopen rychle určovat směr a vzdálenost k cíli díky dotykovým sensorům a znalosti své polohy a k tomu nepotřebuje znát počet a tvar svých překážek.

Definuje se několik druhů Bugů:

- Bug 1 - základní jednoduchý algoritmus - obchází slepě celou překážku – je pomalejší
- Bug 2 - je šikvnější než první – dokáže si zkracovat svojí cestu – jeho nevýhodou je, že obchází překážku zvoleným směrem a může se tak v některých dostat do nekonečné smyčky. Je rychlejší ale občas nesprávný.
- Bug 1+2 - kombinuje algoritmus obou Bugů, aby bylo řešení rychlé a zároveň bez smyčky.
- VisBug - Jeho základem je Bug 2, ale nepoužívá senzor dotykový, ale senzor vzdálenostní. Má za vliv eliminaci zdlouhavých nájezdů. Může se použít v praxi.

Bug algoritmy je možné použít i ve 3D rovině. Překážky je přitom dobré mít přesně popsané. Bug algoritmy se používají pro plánování uklízacích cest nebo náhodné procházení prostoru. [4]

6.2.2. Přesné plánování

Je plánování cesty ve známém prostředí pomocí vektorové mapy kde jsou obsaženy všechny překážky, kam se počítá i hranice terénu. Touto metodou existenci cesty start-cíl, nejbezpečnější a nejkratší cesty pomocí plánovacích algoritmů. Zavádí se zde aproximace pro popis tvaru překážek (většinou pomocí polygonů – n- hranů). Algoritmy se prvně vytvoří pro mobilního robota popsaného jen bodem a poté se upraví pro tvar polygonu či kruhu. Exaktní neboli přesné plánování dokáže pokaždé najít cestu jestli-že opravdu existuje a nebo alespoň ověří že neexistuje. [4]

Způsoby pro bodové a kruhové roboty

Dekompozice – Rozdělení na jednodušší tvary např. lichoběžníky. Je-li přitom start i cíl cesty robota v jednom lichoběžníku je řešení jednoduché a pokud ne, pak se pomocí topologického graf určí skupina lichoběžníků přes které cestu start-cíl absolvujeme.

Graf viditelnosti - slouží k vyhledání nejkratší cesty start-cíl, vrcholem grafu je pozice startu, cíle a vrcholy původních překážek. Vrcholy se spojují hranami pokud mezi nimi není žádná překážka. Graf obsahuje I přebytečné hrany, které můžeme vynechat. Robot je považován za kružnici(proto nevadí, když se robot otáčí) a překážky za body.

Voronoi diagram - překážky jsou řídicí body, které rozdělí oblast na x částí (v každé části je vždy jeden bod), kde platí, že všechny body z dané části mají nejbližší právě ke svému řídicímu bodu a ne k jinému. Aby našel robot správnou (bezpečnou) cestu je třeba, aby s pohyboval po tzv. Voronoiho hranách - jsou to hranice částí ,leží tedy maximální vzdálenosti od překážek. Existuje-li více variant cest musíme vzít v úvahu průměr kružnice definující rozměry robota a vzdálenost řídicích bodů, to znamená brát v úvahu jestli se mezi překážky robot vejde, proto se vždy vybírá cesta, kde je zaručená největší vzdálenost mezi řídicími body – nejbezpečnější cesta. [7]

Způsoby pro přesně popsané - exaktní roboty

Toto plánování lze řešit pomocí Voronoi diagramů, ale ne vždy lze v konvexní oblasti využít kružnice jenž představuje robota (robot se může otáčet – na kružnici to nezaznamenáme). Zde se využívá posunů pomocí konvexních polynomů, kde opět bereme robota jako bod a překážky zvětšíme (ty se ale mohou následně protínat jedna z druhou). Dále lze použít jeden z předchozích algoritmů.

Plánování s otáčením - nejjednodušší je robota brát jako úsečku, zde můžeme využít dekompozice. Budeme-li úsečkou otáčet jen nepatrně pak se topologický graf pro otáčení nezmění. Ale robotem můžeme otáčet o 360° . Tím začne docházet k zánikům některých lichoběžníků a jiné se naopak začnou vytvářet. Vzniklé lichoběžníky můžeme zmenšit o délku úsečky robota. Vznikne nám složitější topologický graf, ale robotem se již může otáčet a posouvat. Některé původní lichoběžníky se nám změní pouze v úsečky se dvěma řídicími body. V tomto momentě vzniká nové patro topologického grafu, kde je obsažen zanikající lichoběžník a jeho sousedi. Původní lichoběžníky změní, ale jejich řídicí vrcholy a strany tvořené původními překážkami zůstanou zachovány.

Chceme-li realizovat pohyb tohoto exaktního robota po cestě nalezené díky topologickému grafu, musíme zajistit že robot nikdy neprotne žádnou překážku. [4][7]

6.2.3. Plánování na mřížce – diskretní prostředí

Plánování na mřížce probíhá pomocí rastrových map. Mapa obsahuje buňky označené jako překážky nebo volný prostor (viz. diskretní prostředí). Tím nám vnikají dvě možnosti reprezentace map. Rastrová mapa – bíločerná mapa obsahující hodnoty 0 a 1 (dle obsazenosti buňky překážkou nebo ne). Pravděpodobnostní mapa - tou lze reprezentovat i nejistoty (víme/nevíme zda daná buňka je volná či ne) tím, že rozšíříme uloženou informaci v buňce na více hodnot. Tato mapa je schopná téměř přímo přijímat data z několika různých senzorů. Využíváme zde následujících metod:

- potenciálové pole
- grafové přístupy
- Pseudo-Voronoi diagramy

Potenciálové pole – patří mezi nejpoužívanější metody plánování na mřížce. Každá buňka dostane na začátku svou hodnotu, tu pak využijeme při rozhodování, kterým směrem se vydat. Kdy start má hodnotu nejvyšší, cíl nejnižší a překážka vždy vyšší než její okolní volné buňky. Existuje několik algoritmů jak hodnoty buňkám přiřadit (záplavové vyplňování), pomocí dosazení euklidovské vzdálenosti cíle.

Grafové přístupy – použití grafového algoritmu procházením do šířky. Vytváří si frontu nezpracovaných vrcholů. Rozděluje buňky a vrcholy na nezpracované, mrtvé (všechny okolní buňky a vrcholy jsou již zpracované) a živé (navštívená, ale okolní ještě ne). Lze zde uplatnit Dijkstrův algoritmus, ten vyhledá nejkratší cestu z grafu tím, že si zvolí prioritní frontu, která bude třídít vrcholy dle vzdálenosti. Tento algoritmus nalezne všechny možné buňky díky, kterým se do cíle dostaneme, ale prochází tím pádem zbytečně mnoho buněk než nalezne cílovou polohou. Proto je zde jeho modifikovaný algoritmus A* který jej zlepšuje díky úpravě třídění vrcholů ve frontě. Vrcholy třídí dle vzdálenosti od polohy startu, ale i od polohy cíle. Pokud naopak budeme pomocí algoritmu B* třídít vrcholy jen podle vzdálenosti od cíle. Vznikne tak agresivní způsob, který ale zaručuje nalezení nejkratšího řešení jen u jednodušších prostředí.

Potenciálové pole a grafový přístup nám vyhledávají nejkratší možné řešení cesty. Jsou ale případy, kdy nehledáme nejkratší cestu. Je možné hledat například cestu nejbezpečnější a v tom momentě se snažíme dodržet bezpečnou vzdálenost od překážek. K tomu lze využít tzv. Pseudo-Voronoi diagramy. Použijeme algoritmy pro procházení do šířky u každé překážky zvlášť. Nadefinujeme Voronoi hrany tam, kde se setkávají vlny ze dvou různých překážek. Po hranách postupujeme do cíle. [4]

6.2.4. Pravděpodobnostní plánování

V oblasti pravděpodobnostního plánování se používá metod: konfiguračního prostoru, pravděpodobnostních plánovačů a algoritmů RRT.

Konfigurační prostor – jednoduchý způsob popisu nejrůznějších plánovacích strategií. Důležité jsou zde parametry popisující uspořádání robota. Opět zde nesmí start a cíl kolidovat s překážkami. Rozeznáváme zde tzv. volný konfigurační prostor což je prostor nekolidující z žádnou překážkou. V momentě, kdy víme kde se volný

prostor nachází, jsme schopni hledat možnou křivku v tomto prostoru. Musíme však brát v úvahu další omezující vlastnosti robota.

Pravděpodobnostní plánovače – slouží k vyhledávání cest. Konfigurační prostor bývá většinou ohromný a není v našich silách jej obsáhnout celý. Proto se neukládají všechny vzorky, ale pouze náhodné. Základní algoritmy vytvoří graf cest a v něm hledají cestu. Graf je pravděpodobnostní, obsahuje náhodně vybrané konfigurace a ověří zda náleží do volného konfiguračního prostoru. Náleží-li tam, vznikne v grafu nový vrchol a snažíme se ho propojit s ostatními. V tom bychom měli pokračovat do doby kdy dosáhneme grafu dobře popisujícího volný konfigurační prostor. Poté připojíme start a cíl a to stejným způsobem jako připojování náhodných konfigurací. Poté hledáme cestu v grafu, když ji najdeme je platná, pokud ji nenajdeme, buď opravdu neexistuje a nebo pokrytí vrcholy v první části nebylo dostatečné a musíme se tedy pokusit o pokrytí lepší.

Algoritmus RRT – v pravděpodobnostním plánovači vzniká problém, že relativně blízké pozice mohou být od sebe dost vzdálené, tento problém řeší RRT (rapidly exploring random trees). K mapování volného konfiguračního prostoru využívá strom jenž má začátek ve startovní pozici. Každý nový vzorek rozvětňuje tento původní strom. Přitom se připojuje k nejbližšímu vrcholu, strom se tak rozrůstá jen mírně. Z připojených vzorků vybereme nejkratší cestu a je-li to cesta bez překážek tak se na konci přiřadí nový vrchol. Tímto způsobem pokryjeme téměř celý volný konfigurační prostor. Občas je nutné vložit (místo náhodných vzorků) cílovou pozici, proto, aby strom k ní směřoval. Algoritmus RRT jde také zdvojit. Zavedeme dva stromy, jeden vede ze startu a druhý z cíle. Výhodou je že oba stromy mají šanci dostat se například ze složité počáteční situace a propojit se až ve volnějším prostoru. [4]

6.3. Optimalizace cesty

Optimalizace cesty přichází na řadu v momentě, kdy se nám podařilo pomocí generátoru možných řešení naléznout více možných cest. Co však máme chápat optimální cestou? Optimalizace hledá maxima a minima dané úlohy, tudíž se za optimální jeví nejkratší, nejrychlejší nebo nejbezpečnější cesta. Není ale vyloučeno, že

výsledná optimální cesta nebude výslednou kombinací těchto možností. Optimalizace se řeší pomocí optimalizačních úloh. Optimalizační úloha obsahuje množinu všech možných řešení, množinu všech omezujících pravidel (podmínek) řešení a účelovou funkci, která přiřadí jednotlivým řešením jejich důležitost (hodnotící číslo) a dle něj pak hledá řešení úlohy (minimum nebo maximum).

Rozlišujeme tři základní druhy úloh týkajících se:

- lineárního programování
- nelineární programování - konvexní a kvadratické

Lineární programování - hledá minimum, nebo maximum lineární funkce v množině popsané lineární nerovností.

Základní tvar lineárního programování obsahuje: maximalizovanou lineární funkci, pravidla omezující problém a kladné proměnné. Jednou ze základních metod lineárního programování je Simplexní metoda.

Simplexní metoda - řeší úlohy lineární optimalizace. Metoda hledá maximum účinkové funkce, jsou dodatečně přidány omezující pravidla, díky nimž vybíráme z množiny možných řešení to neoptimálnější. Narůstá zde počet postupových kroků, zvláště s rostoucím počtem optimalizovaných parametrů, ale nevyžadují speciální přípravy. Začínáme na vrcholu polyedru⁷ možných řešení. Pokud žádný ze sousedních vrcholů nemá lepší hodnotící číslo (danou účelovou funkcí) je tento vrchol považován za optimální řešení. Pokud z vrcholu vede neomezená hrana polyedru a to ve směru lepšího hodnotícího čísla, pak optimální řešení neexistuje. Před použitím simplexní metody je nutné převést nejdříve úlohu do kanonického tvaru – všechny proměnné musí být kladné a přidaná omezení jsou vyjádřena rovnostmi. Simplexní metoda se dále dělí na řadu podskupin, např.: redukovanou, revidovanou, duální, aj.

Nelineární programování – dělí se na dva základní typy – optimalizaci s vazbami a bez nich. Oba typy fungují následujícím způsobem: nejprve opět vytvoří

⁷ **Polyedr** - mnohostěn. Konvexní polyedr je ohraničená konvexní množina mající konečný počet krajních bodů.

množinu všech možných řešení, poté směřují k částem u kterých je nižší hodnotící číslo, pokračuje se až do té doby kdy se přestane účelová funkce měnit. Výsledný bod je uchazečem optimálního řešení. Tento postup je nutné opakovat s různými možnými řešeními, protože tyto metody směřují pouze k lokálnímu minimu.

Rozdělení metod:

- metody s vazbami - metoda přípustných směrů – vybírají vhodný ze směrů neutíkajících z množiny možných řešení – Veinott., Zeuten., aj.
- bariérové algoritmy - zabraňuje porušení omezujících podmínek
- penalizační algoritmy – trestá porušení omezujících podmínek

- metody bez vazeb - DFP
 - metoda největšího spádu (Cauchyho metoda)
 - metoda gradientní (pracuje s lineární aproximací)
- konvexní - má značnou výhodu, že tam kde má lokální minimum má i minimum globální(celkové).
- kvadratická - účelová funkce je kvadratická a její omezující podmínky budou lineární.

Existuje mnoho dalších způsobů nelineárního programování jako např. s či bez použití derivace, hledání volných, vázaných, lokálních a globálních extrémů, aj.

Hladový (greddy) algoritmus je nejjednodušším algoritmem používaným při řešení optimalizace(kombinatorické). Vždy vybírá to nejlepší z toho co mu bylo nabídnuto. Nejvíce se vyplácí tam, kde z daných řešení má vybrat takové, které splňuje předem zvolenou vlastnost a navíc má nejnižší hodnotící číslo. množinu řešení je přitom nutné dobře seřadit , protože to velmi ovlivňuje úspěšnost algoritmu.

6.6. Principy řídicích a plánovacích systémů pro mobilní roboty

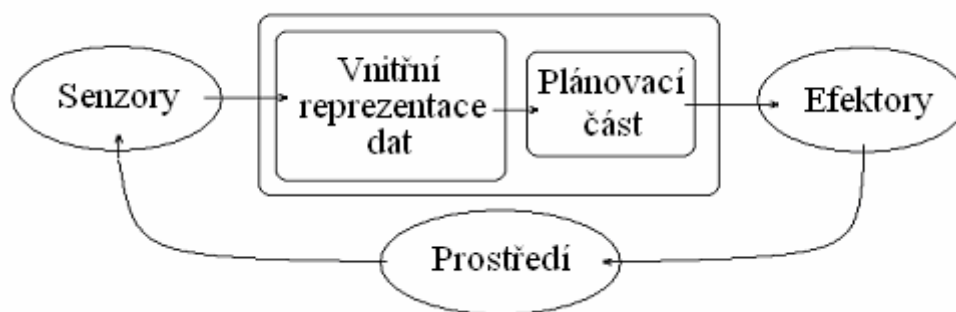
Činnost mobilního robota ovlivňuje řídicí, většinou hybridní systém, kombinující více metod (jehož nedílnou součástí je také plánovací část). Část systému tvoří hardware, další část je samotné řízení skládající se z plánovací a reakční části.

Reakční část je na nižší úrovni než část plánovací. Reakční část je část řízení schopná pracovat v reálném čase, většinou se tak děje při ohrožení robota(objevení překážky, nepředvídatelné události měnící činnost robota, aj.). Plánovací část je oblast obsahující plánovací systém, kde se vytváří posloupnost kroků z výchozího stavu do cílového, pro optimální činnost robota. Jsou zde obsaženy algoritmy, které zpracovávají data přijaté od řídicí části mobilního robota a zpracované je vracejí zpět. Plánovací část se dá nazvat inteligencí robota.

Časem postupně vniklo velké množství řídicích systémů, jejich koncepce můžeme rozdělit na: klasické systémy, reakční systémy, architektury založené na funkčním rozkladu, paralelně řazená architektury a hybridní systémy. [7]

Klasický řídicí systém

Klasický řídicí systém vytváří na základě získaných dat vnitřní model reprezentující vnější prostředí. Pomocí tohoto modelu vytvoří plánovací systém posloupnost dalších kroků. Tyto kroky se pomocí řídicích příkazů zpracují díky



Obrázek 3: Schéma klasického řídicího systému.

efektorům. Mobilní robot, který je vybavený tímto základním řídicím systémem dokáže potřebný plán vytvořit pouze na základě znalosti povolených akcí obecných vlastností okolního prostředí. tento model se však potýká s řadou problémů jako je např.:

- složitost reálného prostředí - je natolik složité , že jej není možné se v něm dostatečně rychle vyznat.
- dlouhé časové prodlevy - mezi vstupem dat, vyhodnocení a reakcí na ně.

(doba odezvy až několik minut při použití plánování).

- drahost - vysoké nároky na rychlost a kvalitu získaných dat ze senzorů. [7]

Reakční řídicí systém

Na základě klasického řídicího systému vzniklo několik požadavků při vytváření nových řídicích systémů. Tyto požadavky se promítly právě v reakčním řídicím systému. Jde o požadované schopnosti mobilního robota, který by měl být schopen rychle reagovat na podměty (pracovat v reálném čase), měl by být odolný vůči změnám okolního prostředí, ale měl by je také brát v ohled. Princip reakčních řídicích systémů je založen na tom, že jednotlivé části systému přímo ovlivňují senzory i efektory.



Obrázek 4: Schéma reakčního řídicího systému.

Výhodou reakčních řídicích systémů je, že jejich cíle nemusí být jednoznačně určeny. Reakční řídicí systémy slouží pro řešení úkolů, které mají klasické plánovací metody, proto zde vzniká několik problémových částí, jako např.:

Plnění více cílů - na rozdíl od klasického plánovacího systému musí být reakční systém schopen řešit několik rozdílných cílů současně.

Velké množství senzorů.

Neomezování - je nutné aby mohl systém pracovat i při ztížených podmínkách (např. chyby některých senzorů) [7]

Architektury založené na funkčním rozkladu

Je to metoda používaná v mnoha oblastech. Řízení je rozděleno do podmnožin akcí a událostí, zodpovědné za určité části běhu systému. Tyto podmnožiny se vzájemně ovlivňují a propojují. Některé podmnožiny řídí pohyb robota, jiné vytvářejí model prostředí a další se starají o práci s daty. Tyto podmnožiny se mohou propojit různými způsoby, z nichž nejjednodušší je posloupné propojení snímání-modelování-plánování-akce. Mezi další přístupy patří např.: rozklad pomocí hierarchického systému – zde je posloupné propojení reprezentováno v každé vrstvě, ze které se hierarchický systém skládá. Dost často používaný způsob je použití tabule. Což je místo, kde si všechny podmnožiny řídicího systému vyměňují informace.

Paralelně řazená architektura

Činnosti řídicího systému se již neřadí sériově, nýbrž paralelně. Tento způsob patří k nejvýhodnějším metodám realizace. Vznikají zde uspořádané vrstvy. Každá z nich reprezentuje určitou dovednost řídicího systému a na ostatních vrstvách není závislá, ale je s nimi propojena. Zároveň jsou eliminovány vstupy z vyšších a nižších vrstev dle důležitosti. Takto provedené řízení má velké množství výhod, systém je odolný, velmi pružný, chování systému je autonomní a dynamické.

7. Příklad

Zadání:

Je dána místnost. V místnosti je pohovka, židle, žárovka, vypínač a robot. Robot má za úkol naléznout vypínač a rozsvítit tak žárovku, aniž by se dostal do kolize s pohovkou. Jelikož robot nedosáhne na vypínač přímo, musí k tomu použít židli, na kterou vyleze a vypínač zapne.

Plánování:

Chceme-li začít vytvářet plánovací systém, musíme si vyjasnit podstatu a problematiku řešené úlohy, její cíl a omezení. Úloha se popíše stavovými veličinami a k něm se přiřadí akce, které budou tyto veličiny upravovat.

Cíl.: Rozsvítit žárovku.

Překážky: Pohovka. Nachází se neznámo kde v místnosti.

Omezení: Výškový rozsah mobilního robota. Rozsvítit žárovku lze jen ze židle.

Nejprve určíme počáteční stavy veličin – (stavy ve kterých se jednotlivé veličiny nacházejí na začátku úlohy):

Robot:

Robot se nachází v místnosti. (Robot_místnost)

Robot je na zemi. (Robot_zem)

Židle: Nachází se někde v místnosti. (Židle_místnost)

Vypínač: Nachází se někde v místnosti. (Vypínač_místnost)

Už máme určené stavové veličiny a pokusíme se k nim přiřadit jejich akce.

(Robot_místnost) – Robot hledá židli a vypínač.

(Robot_místnost, hledej)

(Židle_místnost) – Robot objevil židli a musí ji přemístit pod vypínač.

(Židle_místnost, přenes)

(Robot_zem) – Robot je na zemi a musí vylézt na židli.

(Robot_zem, vylez)

(Vypínač_místnost) – Robot je na židli, přepne vypínač a zapne světlo.

(Vypínač_místnost, rozsviť)

Nyní již můžeme přikročit k plánování. Zde je potřeba určit každému stavu a akci pravidla za kterých můžeme danou činnost provést. Vznikají nám zde však různé stavy, které je potřeba pokrýt omezujícími pravidly, aby nedošlo při plánování ke kolizi.

- hledání židle a vypínače:

Pokud najdeš prvně vypínač, zapamatuj si kde je a pak pod něj dones nalezenou židli.

Pokud prvně najdeš židli, vezmi ji sebou a až najdeš vypínač, tak ji pod něj polož.

Při hledání se vyhni překážkám.

- přenesení židle:

Pokud přenášíš židli nenaraz do pohovky.

- lezení na židli:

Pokud nalezneš židli nelez na ní dříve než ji přenešeš pod vypínač.

Při hledání se vyhni překážkám.

- přepínání vypínače

Bez omezení.

Může nám zde vzniknou ještě jeden problém. Tím je, že se vypínač může nacházet nad pohovkou. V tomto případě, by se dalo zhodnotit zda: může robot pohovku odsunout, nebo zda na ni může vylézt.. Pro zjednodušení řešené úlohy však prozatím tento problém vynecháme a dále budeme předpokládat, že vypínač se nad pohovkou nenachází.

Nyní máme vytvořený plán složený z kroků, který (alespoň teoreticky) dovede robota do cíle.

- | | | | |
|-----------|---------------------------------------|-----------|---|
| 1.A.krok | cíl: nalézt židli | 1.B. krok | cíl: nalézt vypínač |
| 2.A. krok | cíl: (vzít sebou židli) najít vypínač | 2.B. krok | cíl: (zapamatovat si polohu vypínače) najít židli |
| 3. krok | cíl: jít k vypínači (i se židlí) | | |
| 4. krok | cíl: vylézt na židli | | |
| 5. krok | cíl: zapnout vypínač | | |

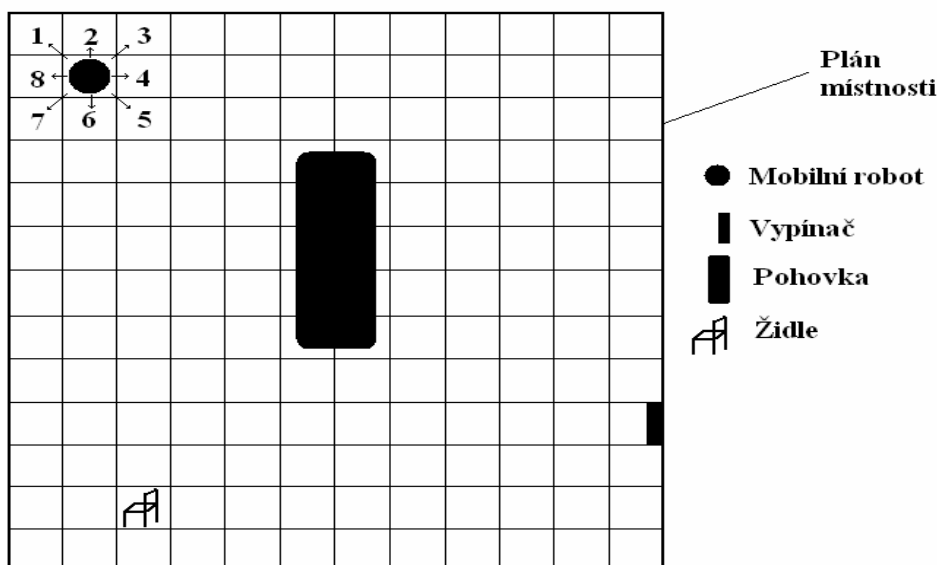
Kroky lze ale ještě zjednodušit na:

- | | |
|---------|---------------------------------------|
| 1. krok | cíl: nalézt židli |
| 2. krok | cíl: nalézt vypínač(židli vzít sebou) |
| 3. krok | cíl : vylézt na židli |
| 4. krok | cíl : zapnout vypínač |

Zde bude přednostnější najít prvně židli a teprve pak vypínač. A plánování se tak mění v hierarchické. Což nám příklad zjednoduší. Posloupnost plánovacích kroků je tedy hotova. Nyní je potřeba zajistit, aby se mohla realizovat.

Plán místnosti, prostředí

Pro zjednodušení příkladu si představíme místnost zakreslenou v mapě pomocí diskretního prostředí a trasu budeme řešit pomocí „plánování na mřížce“. Robot se v diskretním prostředí může pohybovat pouze osmi směry (nepočítáme-li do toho moment lezení na židli). Což nám zjednodušuje prohledávání prostoru mapy. Zvolíme nejběžnější způsob plánování na mřížce, a to potenciálové pole. Mapu pak očíslováme pomocí přiřazovacích algoritmů. Pro připomenutí oblast okolo překážky bude mít vždy vyšší hodnotu než její okolní volné buňky. Právě zde bude lepší postupovat dle hierarchického plánování, které nám umožní lepší přehled na mapě. Nebudeme totiž hledat dva cíle najednou a tak algoritmus lépe přiřadí hodnoty buňkám.



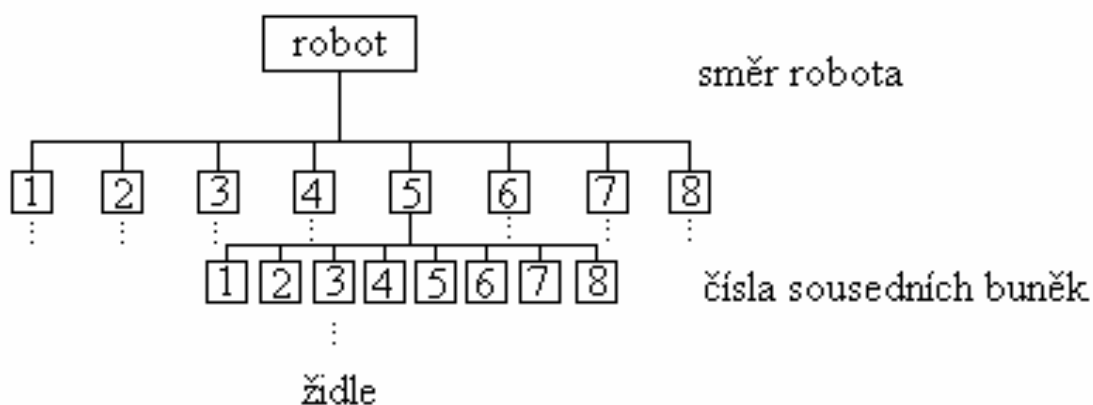
Obrázek 5: Plán místnosti.

Prostor pro reakční plánování nám vznikne v momentě, kdy začneme brát v úvahu předtím zanedbaný problém s vypínačem nad pohovkou. V tento moment nalezneme mobilní robot problém, který nemá naplánovaný, ale čeká se od něj že situaci nějak vyřeší (pokud však úloha řešitelná bude, protože jestli robot nedokáže pohovku ani odsunout ani na ni vylézt, pak světlo nedokáže rozsvítit). Pokud se dokáže, pomocí svých znalostí z báze dovítit toho, že může úlohu vyřešit například vylezením na

překážku (stává se z ní překonatelná), systém svou reakční částí dává podnět do části plánovací, kde dojde k přetvoření kroků.

Prohledávání stavového prostoru

U plánovacích systémů se provádí většinou prohledáváním do šířky. Kdy vrchol stromu bude mobilní robot. Větve od něj jdoucí budou buňky z mapy, označené druhým nejvyšším číslem, takto se vytvoří celý strom, kdy nejspodnější větve budou obsahovat pozici židle a vypínače. Opět zde záleží na tom, který z uvedených plánů kroků si vybereme. Druhý způsob bude opět jednodušší. Aby mohl plánovací systém splňovat svou funkci je potřeba ho „proplést řadou algoritmů“, ať již kvůli prohledávání mapy, stavového prostoru, optimalizace trati aj.



Obrázek 6: Ukázka stromu.

Pokud se budeme řídit hierarchickým plánováním, povede nás první strom od robota k židli, další pak od židle vypínači.

Závěr

Na jednoduchém příkladu jsem se snažila nastínit tvorbu plánu, výběr plánovací metody a následné prohledávání mapy a prostoru při hledání řešení. Zároveň jsem se snažila poukázat na to, jak jediná možnost navíc dokáže ovlivnit (rozdíl mezi klasickým a hierarchickým systémem), popřípadě zkomplikovat (nepředvídatelná

překážka) celou strukturu systému. Systém musí obsahovat souhrn takových pravidel, které nám mimo jiné zabrání v neřešitelnosti úlohy, a ve výskytu nebezpečných situací pro robota a okolí (např.:zamezení možnosti pádu, aj.) Aby se systém mohl realizovat, musí se doplnit řadou různých algoritmů. Je také nutné vybavit zde nezmíněné báze znalostí, zvolit hardware, software, mobilního robota a také danou úlohu naprogramovat buďto pomocí dostupných prázdných systémů a nebo pomocí programovacího jazyka.

8. Robotický fotbal kategorie MIROSOT

8.1.Význam robotického fotbalu

I když tento projekt může vypadat na první pohled jako zábava pro dospělé, nenechme se oklamat. Robotický fotbal se již stal, od doby svého vzniku , celosvětově uznávanou oblastí testování vývoje v odvětví autonomních robotických systémů. Je to jedna z nejlepších možností jak si ověřit nabyté vědomosti z oblasti umělé inteligence a robotiky. Je to oblast kde lze získávat nové znalosti především z oblastí: strategie, dynamičnosti a adaptaci v prostředí, rozhodování v reálném čase, interakci spoluhráčů a autonomních agentů. [14]

8.2.Historie robotického fotbalu

Vznik robotického fotbalu se datuje k roku 1995, kdy s tímto nápadem přišel profesor Jong-Hwan Kim z Korejské univerzity. O rok později již byla stanovena pravidla hry. Koncem roku 1996 se v KAIST (Advanced Institute of Science and Technology) konal první ročník Světového poháru v robotnickém fotbalu.

V roce 1997 pak vzniká organizace FIRA (Federation of International Robot-soccer Association) sdružující zkušené studenty a vědce. Organizace mimo jiné stanovila pravidla turnajů pro třídu fotbalových robotů MiroSot . [8]

8.3. Kategorie robotického fotbalu

Světový pohár se dělí na poháry regionální. Existuje několik kategorií poháru.

Kategorie se liší především rozměry hřiště a parametry robotů. [9]

Jsou to:

Humanoid Robot World Cup Soccer Tournament (HuroSot)

Single Humanoid Robot World Cup Soccer Tournament (S-HuroSot)

Robot World Cup Soccer Tournament (RoboSot)

Single Robot World Cup Soccer Tournament (S-RoboSot)

Micro Robot World Cup Soccer Tournament (MiroSot)

Single Micro Robot World Cup Soccer Tournament (S-MiroSot)

Nano Robot World Cup Soccer Tournament (NaroSot)

Single Nano Robot World Cup Soccer Tournament (S-NaroSot)

Khepera Robot World Cup Soccer Tournament (KheperaSot)

Single Khepera Robot World Cup Soccer Tournament (S-KheperaSot)

Kategorie MiroSot se dále dělí na tři skupiny:

- Small league MiroSot
- Middle league MiroSot
- Large league MiroSot

Hlavní rozdíly mezi těmito třemi skupinami je ve velikosti hřiště a počtu hráčů. [8] [9]

8.4. Technické parametry kategorie MIROSOT

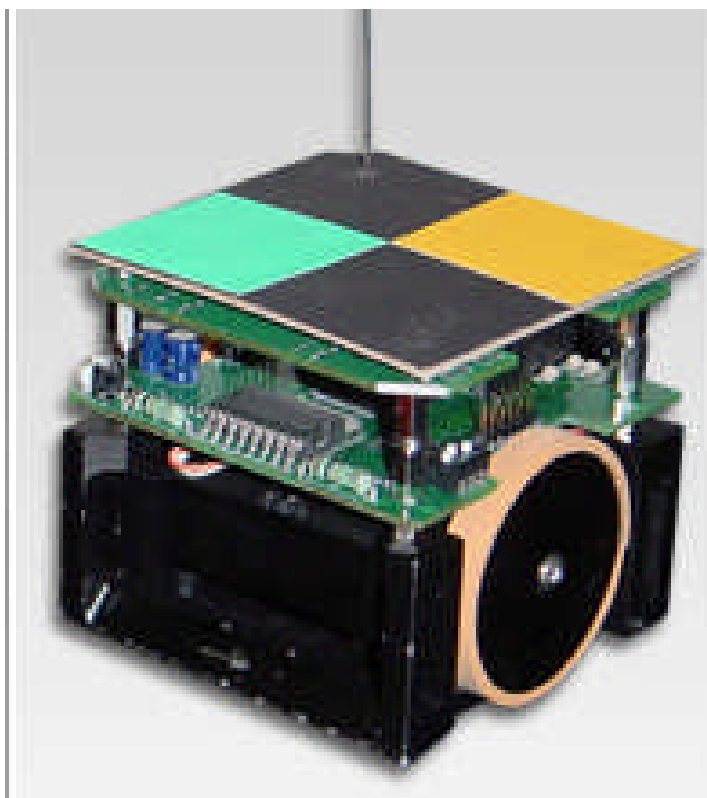
Družstvo: Small league - 3 hráči, Middle league - 5 hráčů, Large league - 11 hráčů
Z toho jeden robot může být brankářem. Brankář je brankářem v momentě kdy se pohybuje v brankovém území. Pokud se v brankovém území objevují současně dva roboti, není za brankáře považován ani jeden v momentě kdy je brankové území prázdné, brankářem se stává první robot, který se v této oblasti vyskytne.

- Robot:** Nesmí přesáhnout stanovený rozměr 7,5 x 7,5 x 7,5 cm a stanovenou váhu 0,6 kg, anténa se do velikosti nezapočítává, robot může být vybaven čímkoliv, např.: nohy – ale nesmí v maximálním rozpětí přesahovat vymezený rozměr.
- Hřiště:** Podklad hrací plochy bývá většinou černý (bez odrazový). Osvětlení 900 – 1000 luxů (doporučená hodnota), minimální hodnota 500 luxů, je důležité aby světlo bylo všude řádně rozptýleno. Hřiště je vyznačeno bílými čarami stejně jako klasické fotbalové hřiště, je obehnáno mantinelem vysokým 5 cm a v rozích je zakulacen, aby se míč nedostal do nedostupné polohy
- Rozměry:** Small league - 150cm x 130cm .
Middle league - 220cm x 180cm, brána 40 cm.
Large league - 400 x 280 cm, brána 60 cm.
- Míč:** Oranžový golfový míček – průměr 42,7 mm, hmotnost 46 g.
- Herní čas:** Dva poločasy o 5 minutách (z důvodu nízké výdrže akumulátorů), organizátoři mohou čas prodloužit (především pro finálové zápasy). Čas se při přerušení hry stopuje
poločas trvá 10 minut, pokud tým není připraven ke hře dostane ještě 5 náhradních minut a pak je ze hry vyloučen. Prodloužení trvá 3 minuty.
[8] [9]

8.5. Základní popis hry a její pravidla

Robotický fotbal se podobá fotbalu klasickému. Hráči jsou však roboti na místo lidí. Roboti jsou řízeni pomocí počítače. Každý tým má jednoho brankáře a několik hráčů. Proti sobě soupeří vždy dva týmy. Lidská obsluha nesmí do hry zasáhnout, pokud to nevyžaduje sama hra – např. při faulu, trestných kopech a patových situacích, kdy se hra přerušuje. Aby se neporušovala pravidla hry, je zde přítomný i lidský rozhodčí. Před zahájením hry se vylosuje herní strana a držení míče. Po losování si týmy rozmístí své roboty na hřišti. Toto je zároveň poslední okamžik, kdy můžou lidští hráči zasáhnout do dění na hřišti. Míček se postaví doprostřed hřiště.

Poté dá rozhodčí signál k zahájení hry. Pravidla jsou velmi podobné klasickému fotbalu, největší rozdíl je v tom, že místo autové čáry je na hřišti mantinel. Gól je uznán v momentě kdy se míček dostane celým svým objemem přes brankovou čáru. Následuje množství pravidel o postavení hráčů-robotů ve hře a o faulování. Mezi fauly se považuje např.: úmyslný střet robotů, tlačení robotů, blokování míče více jak 10 vteřin, postavení více robotů v brankové oblasti aj. Do podrobného znění pravidel lze nahlédnout na stránkách robo-fotbalové organizace FIRA. Robotova poloha se může měnit pouze v případě, že spadl a je potřeba jej zvednout. Až do dalšího přerušení se hráči nesmí dotknout řídicího počítače, jinak mohou být diskvalifikováni. [9]



Obrázek 7: Fotbalový robot skupiny MIROSOT.[9]

8.6. Systém robotického fotbalu

Systém robotického fotbalu z technického hlediska rozdělíme na čtyři části:

- roboty – hráče
- prostředky sloužící k lokalizaci polohy míče a robotů obou družstev

- strategii řízení
- centralizovaný řídicí počítač

Robot – hráč

Celkový systém se skládá z níže uvedených modulů. Do pohybu jej uvádí dvě hnací kolečka spojená s dvěma nezávisle řízenými elektromotorky. Robot se může díky kolečkům pohybovat rychlostí až 2 m/s. Mikroprocesor dekóduje signály vysílané z nadřazeného počítače a řídí elektromotorky pomocí regulačních obvodů se zpětnou vazbou tak, aby dosáhl co nejkratší doby při změně hráčovy polohy na hrací ploše. Pro změnu rychlosti pohybu hráče slouží rychlostní smyčka zpětné vazby. Nejlepším řešením pro zapojení elektromotorků bude diferenciální řízení. Robot je díky tomu schopen otočit se teoreticky na místě. Řízení obsahuje regulátory typu PID a je zde použita pulsní modulace signálu. Každý hráč je částečně autonomním fotbalovým robotem. Částečně proto, že je mechanicky závislý sám na sobě, ale zároveň je závislý na řídicím počítači. Jako zdroj energie slouží NiHM akumulátory. [10]

Prostředky sloužící k lokalizaci polohy míče a robotů obou družstev

Jak bylo zmíněno již dříve, družstvo má k dispozici kameru. Kamera zavěšená ve výši 2,5 - 3 m, je barevná, analogová či digitální a je propojena s řídicím počítačem. Velká liga není omezena pouze na jednu kameru. Rychlost snímaného obrazu bývá zpravidla 30 snímků za sekundu. Jednotliví hráči jsou od sebe rozlišováni barevnými čtverci umístěnými na horní ploše krychle každého hráče. Musí být alespoň dva, kdy jeden identifikuje tým a druhý robota. Ty umožňují od sebe rozeznat vlastní hráče a protihráče a brankáře. Tým nesmí mít nikde na svých robotech stejnou či podobnou barvu jako má soupeř na čtvercích. Kameru má družstvo umístěnou nad hrací plochou. [9] [10]

Strategie řízení

Strategie řízení se dělí do čtyř úrovní. Na nejvyšším stupni strategie program vybírá nejvhodnější postup. To se děje pomocí použití metod umělé inteligence a

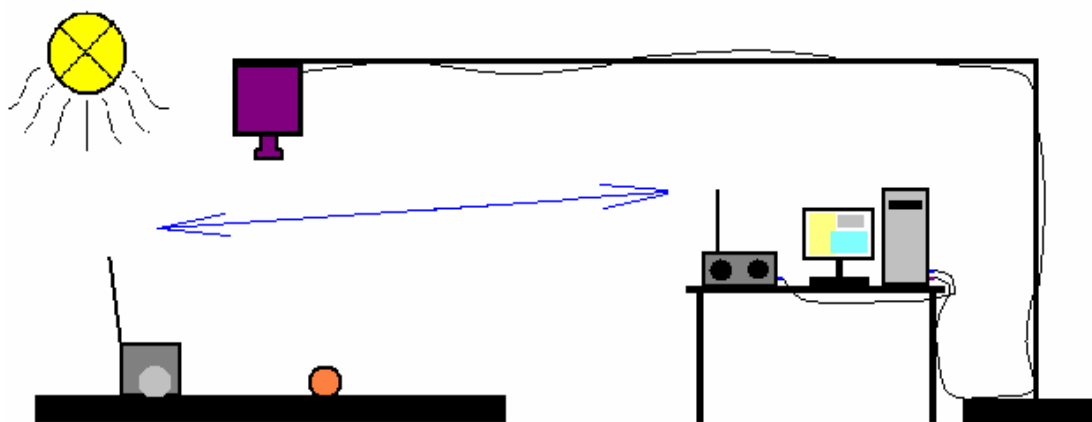
výpočtu pravděpodobnosti. Na tomto základě vznikají rozhodnutí typu: kopni do míče, braň míč, dotkni se míče, utíkej, vrať se, aj. V druhé úrovni se plánuje dráha robota s možností interakce s ostatními. Na třetí úrovni se vytváří odpovídající pohyb kol pro naplánovanou dráhu. V poslední čtvrté úrovni se vytváří potřebné impulsy pro napájení elektromotorků. Strategie řízení bude detailněji popsána v kap. 10.1.[10]

Řídící počítač

Je řídicím mozkiem celého týmu. Ovládá a řídí veškerou komunikaci mezi roboty. Pomocí různých programů, zpracovává obraz kamery, vytváří model prostředí (určuje polohu a identifikaci robotů, soupeřů a míče). Plánuje další postup a strategii. Předává informace do robotů a jejich modulů. Aby mohla lidská obsluha něco upravit musí počkat na přerušeni hry. [9] [10]

8.7.Celkový náhled na hru

Nad hrací plochou má každé družstvo umístěno svojí kameru, která je napojena na řídicí počítač. Kamera snímá barevně označené čtverečky na robotech. Řídící počítač obsahuje program pro identifikaci objektů, modul pro bezdrátovou komunikaci a program pro určování strategie. Jednou z nejdůležitějších vlastností je schopnost rychlého snímání hrací plochy. Řídící počítač zasílá všem robotům jejich data o poloze a data o poloze ostatních robotů a míčku. Na základě polohy míčku a robotů zvolí řídicí počítač nejvhodnější strategii dalšího postupu a vyšle rozkazy pomocí komunikačního modulu. Mechanismus robotů se postará o splnění příslušného rozkazu. Mezitím jsou opět roboti i míček snímány kamerou. Jedná se o velmi rychlou smyčku a díky použití dokonalých programů se jejich pohyb jeví jako téměř plynulý. Celková hra je ovlivňována množstvím jednotlivých modulů a systémů, které ve finále pracují jako jeden celek pomocí komunikačního modulu.



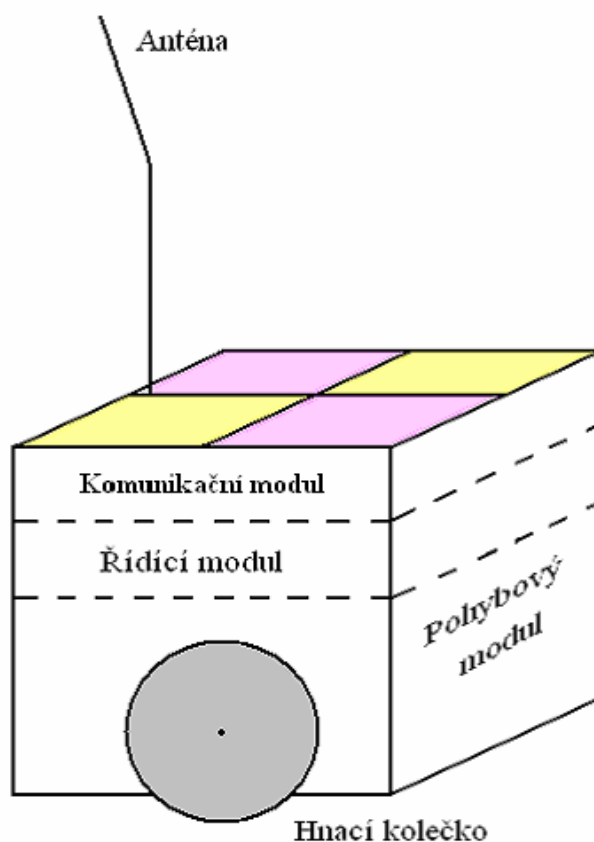
Obrázek 8: Celkový náhled na hru.[14]

9. Druhy modulů a systémů pro robotický fotbal MiroSot

9.1. Druhy modulů

Robot se skládá z modulů. Modulem rozumíme část systému robota řešící nějakou skupinu problémů. Systém se dá rozdělit na libovolný počet modulů. Záleží na tom kolik činnosti chceme každému z nich přiřadit. Některé moduly bývají blízko spojeny s jinými, jiné se částečně či více prolínají. Pro lepší přehled modulů je nyní rozřadíme do následujících podskupin: [11] [12]

- moduly pro vytváření představy o okolním prostředí
- moduly pro vytváření komunikace
- moduly pro plánování
- moduly pro vnímání a chápání hry
- moduly pro ovládání pohybu



Obrázek 9: Schéma fotbalového robota skupiny MIROSOT.[14]

9.1.1. Moduly pro vytváření představy o okolním prostředí

Informace o prostředí (stavu na hrací ploše), se díky neustálému pohybu robotů stále mění. Projevuje se zde reakce v reálném čase a dynamika prostředí. Kamera i senzory snímají prostředí automaticky a jejich funkce není závislá na plánovacím systému. [11] [12]

Modul počítačového vidění

Jedná se o schopnost napodobit lidské vidění, pomocí snímání daného prostředí elektronickými prostředky. Součástí počítačového vidění v robotickém fotbale je barevná kamera, speciální karta pro rozlišování barev a program pro identifikaci snímaného prostředí. Počítačové vidění je jedním z nejdůležitějších modulů, protože

na základě toho co počítač vidí se rozhoduje co udělá dále. Důležitou částí počítačového vidění je rychlé snímkování sledovaného prostředí a následné rychlé zpracování obrazu. Čím větší je rychlost snímkování, tím je celkový systém schopen rychleji reagovat na změny – tzn. zlepšuje se dynamičnost. Toto je také prostor pro rozlišování jednotlivých robotů a jejich soupeřů od sebe, pomocí rozdílných barevných čtverečků na každém z nich. Obraz z kamery je napojen na centrální počítač, kde jsou data o obrazu zpracovávána a posílána do kognitivního modulu. [14]

Modul senzorický

Schopnost vnímat a rozeznávat prostředí. Jsou zde obsaženy senzory, které pomáhají zkvalitnit prostorovou představu robota a které mohou doplňovat informace pro modul počítačového vidění. U robotického fotbalu však většinou plně postačí kamera. Tohoto modulu by se však dalo využít v momentě, kdy je centrální počítač zaneprázdněn a nemůže se danému robotu v daný okamžik věnovat. Pak by se pomocí senzorů dalo zabránit např. nárazu.

Modul kognitivní

Schopnost vytvářet a aktualizovat vnitřní model prostředí. Model prostředí bude zpracováván pomocí informací z kamery a počítače v nějakém vhodném grafickém počítačovém programu na simulaci (identifikaci) prostředí. Data z tohoto modulu přechází do modulů pro plánování.

9.1.2.Moduly pro vytváření komunikace

Modul komunikační

Zde se udržuje komunikace mezi řídicím počítačem robotem a spoluhráči. Realizuje se pomocí bezdrátového připojení. Většinou pomocí rádiového signálu, kdy se před zápasem zadá každému týmu jiná frekvence. Roboti i centrální počítač mají k dispozici přijímač a vysílač. Díky komunikačnímu modulu může centrální počítač předávat jednotlivým robotům jejich úkoly. [11][12]

9.1.3. Moduly pro plánování

Modul dynamický

Schopnost řešit nepředvídatelné události v prostředí. Jedná se o události, které během okamžiku změni náš původní plán, např.: změna směru pohybu protihráče aj. Možnost systému reagovat v reálném čase. Propracování tohoto modulu je zvláště důležité pro plynulost hry. Jak již bylo uvedeno je závislý na modulu počítačového vidění. [11] [12]

Modul plánování a řešení úloh

Schopnost automatického řešení zadané úlohy na základě znalosti prostředí a cílů úlohy. Funguje na předpokladu, že je robot schopen vykonávat nějaké pokyny. Modul obsahuje plánovač, který dokáže vygenerovat posloupnost akcí dalšího nejvhodnějšího postupu a vytvářet tak strategii. Právě tento modul je centrem celého plánování. Generují se zde veškeré plány a úkoly pro všechny moduly, které jsou závislé na plánovacím systému. [11] [12]

9.1.3. Moduly pro vnímání a chápání hry

Tyto moduly by měli patřit spíše do podskupiny modulů pro plánování, jelikož rozhodování v plánovacím systému plně ovlivňují. Jsou však tak důležité pro celkový vzhled a dojem ze hry, že si zaslouží samostatné místo. [11] [12]

Modul strategie

Jedná se o výběr nejvhodnějšího postupu vygenerovaného plánovačem, na základě strategických pokynů. Modul vybírá co je důležitější k provedení realizace na základě strategických úrovní realizace. Právě díky dynamickému prostředí nelze nakonfigurovat strategii přímo již na začátku, ale musí se vlivem dynamičnosti náležitě měnit. Jak bylo již výše uvedeno, dělí se do určitých úrovní, kdy jedna část strategie má přednost před jinou. Strategie se musí měnit podle vzniklé situace, např:

nejvyšším bodem strategie bude „útoč na branku“, ale to bude platit jen do okamžiku kdy náš tým přijde o míč, protože útočit na branku bez něj nemá jaksí smysl. [11] [12]

Modul interakce

Schopnost plánování a řešení úlohy robota na základě vlivu okolních robotů a nejen vnímání sama sebe. Komunikaci bohatě nahradí vizuální informace o pozici ostatních spoluhráčů a protihráčů pomocí kamery, díky níž zná každý robot v daný okamžik přesnou polohu spoluhráčů i protihráčů. [11] [12] [14]

Modul adaptace chování

Schopnost robota přizpůsobit se v daném okamžiku na danou situaci, vnímat a rozpoznávat vlastní situace a situace ostatních robotů v prostředí. Důležité také je aby byl celý systém schopen se pokud možno poučit z nastalých situací. Pokud by došlo k situaci, kdy daný problém vyřeší soupeř lépe, bylo by rozumné aby se náš vlastní systém z dané chyby pokud možno poučil. V takovém případě by bylo možné tuto část doplnit o modul učící se, který by fungoval na principu umělých neuronových sítí⁸. [11] [12]

9.1.5. Moduly pro ovládání pohybu

Modul řídicí

Impulsem pro zahájení činnosti řídicího modulu je vygenerovaný plán z plánovacího modulu. Řídicí modul je omezen pouze na ovládání dvou motorů. Je schopen ovlivnit směr a rychlost podvozku a zároveň je možné ovlivnit točení kol tak, aby došlo ke změně směru. Proto každé kolečko ovládá jiný motor. Jejich řízení probíhá pomocí algoritmů v mikrokontroléru. Zpětná vazba zajišťuje kontrolu nad provedenými změnami. Pokyny z řídicího modulu přechází do modulu pohybového a

⁸ **Neuronová síť** – paralelní distribuovaný systém prvků, které simulují biologické neurony. Systém je schopen zpracovávat požadované informace.

manipulačního. Jsou zde také zpracovávány všechny informace z kamery a senzorů.
[11] [12]

Modul pohybový a manipulační

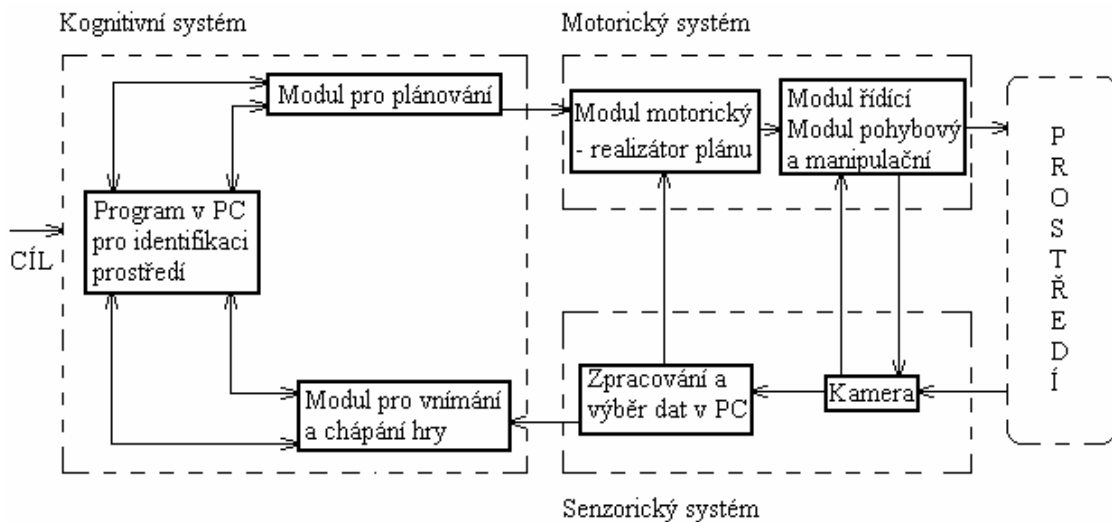
Pro pohyb robotů v robotickém fotbalu kategorie MIROSOT se používá tzv. diferenciálního podvozku. Pohyb robota ovlivňují dvě aktivní kola poháněná dvěma na sobě nezávislými motory. Podvozek má také stabilizační body. Pro změnu jejich rychlosti je připojena zpětnovazební smyčka. Používá se PID regulátoru. Změna pohybu a rychlosti je popsána v kap.8.5. [11] [12]

Modul motorický - realizátor plánu

Schopnost automatického řešení zadané úlohy na základě znalosti prostředí a cílů úlohy. Vydává automatické impulsy podněcující systém k řešení daného problému bez nutnosti zásahu lidské obsluhy. Je obsažen v robotu. [11] [12]

9.2.Druhy systémů

Obecný robot se skládá ze tří systémů. Je to systém kognitivní, motorický a sensorický. Následující odstavec poukazuje na rozdělení struktury robota pro robotický fotbal z hlediska rozložení na systémy. U robota pro robotický fotbal je však sensorický systém nahrazen kamerou a komunikací s řídicím počítačem. Systémy jsou vzájemně propojeny, aby mohly být neustále aktualizovány se změnou stavů. Blokované schéma robota pro robotický fotbal, pak může vypadat následovně.



Obrázek 10: Blokové schéma robota pro robotický fotbal.^[11]

9.3. Zařazení modulů

Každý modul má dané své pevné místo působení. Některé se objevují jen na jednom místě, ostatní musí být přítomny ve více oblastech. Modul pohybový, manipulační a řídicí bude obsahovat pouze robot a modul pro vnímání a chápání hry, kognitivní bude v centrálním počítači, ale komunikační modul je obsažen u obou dvou.

10. Plánovací expertní systém pro robota robotického fotbalu

10.1. Herní strategie

Tak jako v normálním fotbalu je potřeba i v robotickém fotbalu navolit herní strategii (dále jen strategie). Pokud chceme, aby hra vypadala co nejvíce realisticky musíme strategii plynule měnit. Není možné například stále útočit, když už nemáme míč. Strategie se bude dělit na dvě skupiny a to strategií pro tým a pro hráče. Základní týmovou strategií bude vyhrát. V daný moment je pak dle potřeby nutné měnit strategii útočnou na obranou a naopak. Lze docílit i toho, že se bude moci měnit strategie agresivity hry, aby nedocházelo ke zbytečným faulům. Jinak řečeno, týmovým cílem bude útočení a střelení gólů, do momentu než přijde o míč, kdy se cíl změní na bránění a získání míče zpět, pokud možno v rámci pravidel, aby nedocházelo

k znevýhodnění v rámci jejich porušení. Zvláště týmová strategie je pak ovlivňována interakcí se spoluhráči a protihráči.

Je však také důležité udávat strategii každému robotu zvlášť. Tak například pokud je v daném momentě v pozici brankáře, musí se jeho strategie změnit natolik, aby byl schopen ubránit branku. Pokud nastane situace, že brankář opustil brankovou oblast, je potřeba vyhledat robota, který se do brankové oblasti vrátí nejrychleji a zabere tak brankářův post v momentě kdy to bude potřeba.. Dále jsou zde prvky strategie jednotlivých robotů, kdy se musí v daný okamžik rozhodnout zda vykopnout míč, krýt ho „vlastním tělem“, nebo s ním „utíkat“. Obecně platí, že strategie hráče, by měla být taková, aby přinášela prospěch týmu a ne naopak hru komplikovala. Strategie se volí pomocí pravděpodobnostních výpočtů a použití umělé inteligence.

10.2. Interakce mezi spoluhráči a protihráči

Pojmem interakce rozumíme vzájemné působení dvou nebo více činitelů. Popisuje společné chování ve skupině. V interakci rozlišujeme dva druhy chování a to koordinaci a kooperaci. Pomocí koordinace zajišťujeme pozitivní vliv jednoho robota na činnost ostatních, např.: vyvarování se srážek, faulů, všichni se neženou za míčem, apod. K tomu je potřeba komunikovat s celou skupinou, na základě počítačového vidění a realizaci pravidel. Kooperace je nadřazená nad koordinaci. Jde o takovou činnost, kterou nezvládne jediný robot, např.: secvičené signály, různé akce, útok, obrana, aj. kdy chceme aby se do hry zapojilo více robotů hned od začátku a ne až po odehrání jedním. Průběh interakce zajišťuje řídicí počítač. Který neustále vysílá a přijímá data od všech robotů a obraz od kamery. Problém může nastat v momentě kdy počítač přestane fungovat. Například bude zaneprázdněn komunikací s ostatními členy týmu a na jeden robot se nedostane kapacita. V takovém případě je dobré každého robota vybavit multiagentním modulem, kdy každý robot bude mít své přednostní cíle, které bude plnit když nebude v daný moment moci komunikovat s řídicím počítačem. Např.: brankář zůstane v brankové oblasti. Pro správnou interakci mezi roboty musí být dobře naprogramován komunikační protokol, aby se nestalo například to, že se k míči pohnou najednou všichni roboti. [15]

10.3. Cíle plánovacího expertního systému pro robotický fotbal

Plánovací expertní systém pro hráče robotického fotbalu je založen na předpokladu, že robot jako takový je schopen vykonávat určité činnosti (např. pohyb, brzdění, zrychlení, atd.). Na základě těchto činností robota lze v plánovacích modulech vytvářet plán posloupnosti kroků, jejichž hlavním cílem je, aby robo-tým vyhrál. Základní princip plánovacího expertního systému bude udávat strategie řízení. Jak již bylo uvedeno, strategie řízení se dělí do čtyř úrovní. Tyto úrovně jsou hierarchicky uspořádány a jejich pořadí se nemění.

10.4. Moduly, kde není plánování potřeba

Není jich sice mnoho, ale přesto se najdou. Tak například moduly pro vytváření představy o okolním prostředí jej nepotřebují. Tyto moduly pracují automaticky (kamera se zapne a snímá) a nebo na pokyn počítače (při hře je zapnut program pro vytváření modelu prostředí). To že tyto moduly plánování nepotřebují, ovšem neznamená, že tomu není naopak. Pro plánování je vizualizace prostředí základním kamenem, od kterého se vše vyvíjí.

Dalším takovým modulem je modul komunikační. Komunikace probíhá na bázi komunikačního protokolu, který je vytvořen a uložen v řídicím počítači. V případě problému s komunikací tak dochází k fatálnímu selhání.

10.5. Nepřímo ovlivňované moduly

Jedná se o takové typy modulů, jejichž činnost je nepřímo závislá na tvorbě plánovacích kroků, a to, v oblasti nejnižších úrovní strategie řízení. Tím bude třetí (impuls pro ovládání motorků) a čtvrtá úroveň (pohyb kol pro naplánovanou dráhu). Tyto dvě úrovně jsou ovlivňovány pomocí vytvořených algoritmů v mikrokontroléru. Plánování se projeví až v druhé nadřazené úrovni strategie řízení kdy se plánuje dráha robota s možností interakce s ostatními. Mezi nepřímo ovlivňované moduly patří modul řídicí, pohybový a manipulační.

10.6. Přímou ovlivňované moduly

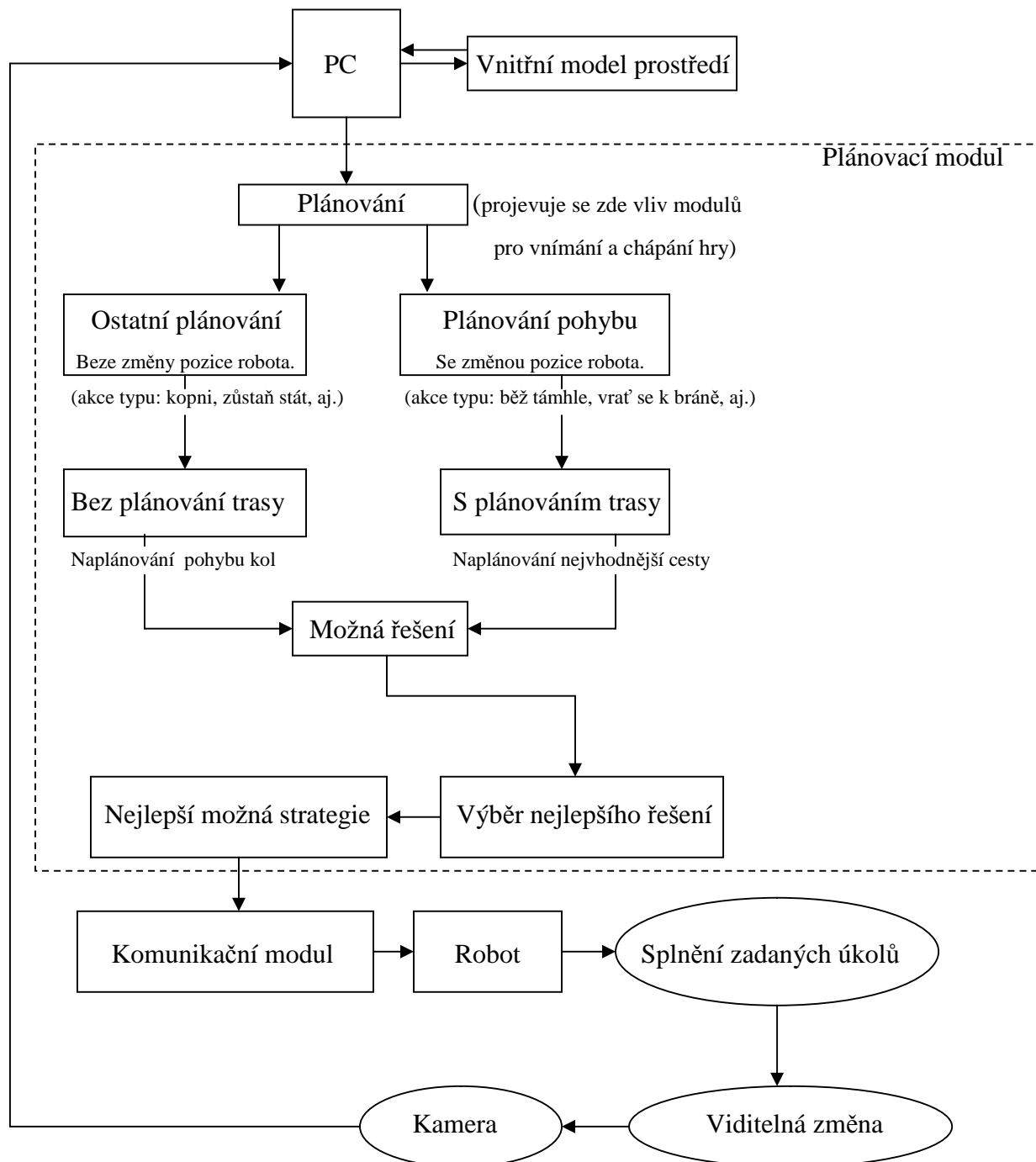
Moduly, jejichž působení je závislé na plánování. V modulu „plánování a řešení úloh“ dochází ke generování postupů a výběru dalších kroků. Je to centrální oblast plánovacího expertního systému. Dále do této skupiny spadá modul: strategie, interakce, adaptivního chování a dynamiky. Tyto moduly jsou úzce spojeny a prolínají se navzájem. Zatím co interakce je spojena s druhou úrovní řídicí strategie, ostatní zde zmíněné moduly zasahují až do první úrovně.

Nastal také vhodný okamžik zmínit se o plánování trasy. Tato část by se dala oddělit do samostatného modulu, ale stejně tak ji můžeme považovat za součást modulu strategie. Plánováním trasy se zabývá druhá úroveň strategie řízení.

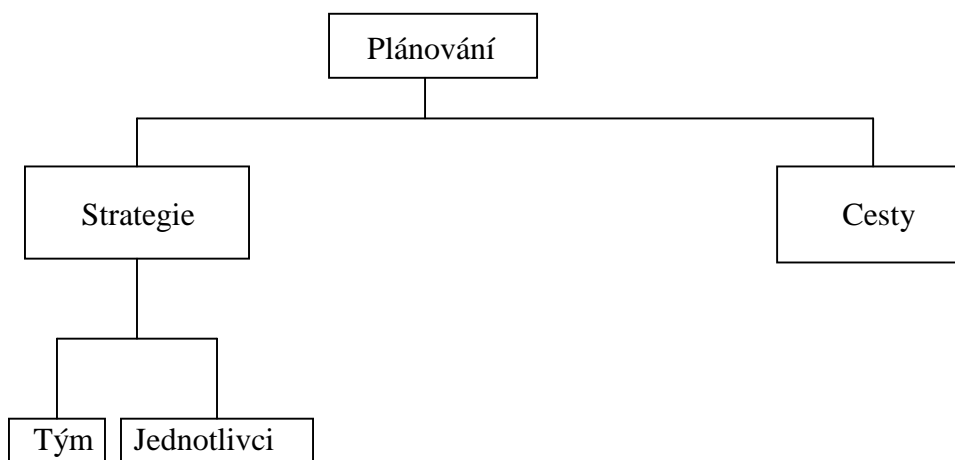
10.7. Závěrem

Aby byl plánovací expertní systém pro fotbalového robota úplný, musí plánovací systém pokrýt veškeré oblasti výše popsaných modulů (všech které to potřebují). Pro vzhled hry jsou nejdůležitějšími oblastmi strategie, interakce, vnímání a chápání hry, které umožňují napodobit celkový dojem ze hry. Celý plánovací systém je potřeba omezit takovou skupinou pravidel, aby se co nejvíce vyloučily záporné situace ve hře. Ať už máme dojem z plánovacího systému jakýkoliv, jeho kvalita se nejlépe projeví v praxi, a tak je potřeba zkusit a hrát si. Jen při hře se zviditelní nedostatky, které z velkou pravděpodobností budou spadat právě do oblasti první a druhé úrovně řídicí strategie.

11. Koncepce plánovacího expertního systému pro robotický fotbal



Obrázek 11: Plánovací modul



Obrázek 12: Blokové schéma - plánování

Báze znalostí

V této části plánovacího systému lze nalézt velké množství znalostí a možných cílů. Čím více a lépe propracované cíle zde budou, tím více bude celková hra vypadat. Ideálním případem pak bude, pokud se plánovací systém bude schopen poučit ze svých chyb při hře a poučit se z lepší taktiky protihráče. Pak lze docílit toho, aby byl systém schopen se některé věci naučit (nové cíle – např. kombinace) a přiřadit si je do své báze a naopak z ní vyhodit takové věci, které vedou ke špatným výsledkům.

Plánování cesty

Za nejvhodnější způsob považuji Voronoi diagramy. Díky obrazu z kamery totiž známe přesnou polohu všech „překážek“ – tudíž se jedná o přesné plánování a pro něj je právě tato metoda vhodná. Metoda je volena tak, aby vybrala nebezpečnější cestu, z hlediska maximální vzdálenosti od všech překážek (tedy robotů a mantinelu). Je ale potřeba vhodným pravidlem zamezit tomu, aby byl považován za překážku i míček. K tomu postačí, pokud budeme přiřazovat robotům jejich „velikosti“ (ve formě kružnice) častěji, než při nálezů více možných řešení. Optimalizaci cesty bych řešila pomocí omezovacích algoritmů. Jedním takovým vhodným algoritmem bude odlišení vlastních spoluhráčů od protihráčů. Pokud nebudeme chtít dodržovat maximální bezpečnou cestu i od svých spoluhráčů, zvýší se nám možnost daných řešení, je ale

potřeba zajistit aby se jejich „velikosti“ nepřekrývali – tudíž nedocházelo ke srážce vlastních spoluhráčů.

Strategie

Je potřeba ji naplánovat ještě před naplánování pohybových akcí pro hráče. Použitím vhodných pravidel, budeme schopni navolit taktiku hry. Vhodnou metodou bude prohledávání stavového prostoru. Bude se jednat o informativní metodu prohledávání do šířky pomocí dopředného řetězení (směrem k cíly), doplněného vhodným algoritmem pro zjednodušení stavového prostoru. Cíly budou druhy herní strategie (útočení, bránění, získání míče, bránění ve hře, aj.). Pomocí daných poznatků, pravidel a provedených akcí se přiblížíme k jednomu z cílů, a ten bude pro daný moment určující strategií. Dalším možným cílem může být možnost nějaké předem definované kombinace, pro kterou se může plánovač rozhodnout na základě dané situace na hřišti (tzv. secvičené signály-at' již plynou ze hry nebo jsou součástí např. trestného kopu). Tuto metodu lze velmi dobře uplatnit jak pro týmovou strategii tak pro strategii jednotlivce, postup bude stejný jen cíle budou jiné. Navíc bude velmi výhodné doplnit prohledávání stavového prostoru tzv. „hladovým algoritmem“, který hledání značně urychlí.

Závěr

Plánovací systém pro robotický fotbal jsem rozdělila na dvě základní části, plánování trasy a strategie. Plánování trasy se udává pro každého robota zvlášť pod vlivem předem dohodnuté strategie. Strategie by se dále dala rozčlenit na podskupiny ohledně úkolu který bude řešit(skupinu cílů). Celkový plánovací expertní systém by se měl ubírat směrem reakčního plánovacího systému, který je vhodný pro činnost v reálném čase a byl by tak dobrým nástrojem především pro dynamický modul. Dále není potřeba vytvořit celý plánovací postup před zahájením činnosti. Z určitého hlediska by to nebylo asi ani možné, jelikož se roboti neustále pohybují. Navíc je schopen řešit několik úkolů najednou, což je potřeba pokud chceme aby byli všichni roboti neustále aktivní.

12. Seznam obrázků.

Obrázek 1: Model plánovacího systému. [13].....	10
Obrázek 2: Způsob prohledávání stavového prostoru neinformovaně. a.)do šířky b.) do hloubky.....	18
Obrázek 3: Schéma klasického řídicího systému.	29
Obrázek 4: Schéma reakčního řídicího systému.	30
Obrázek 5: Plán místnosti.....	34
Obrázek 6: Ukázka stromu.	35
Obrázek 7: Fotbalový robot skupiny MIROSOT.[9]	39
Obrázek 8: Celkový náhled na hru.[14]	42
Obrázek 9: Schéma fotbalového robota skupiny MIROSOT.[14]	43
Obrázek 10: Blokové schéma robota pro robotický fotbal.[11].....	48
Obrázek 11: Plánovací modul	52
Obrázek 12: Blokové schéma - plánování.....	53

13. Seznam použité odborné literatury.

- [1] DVOŘÁK, Jiří. *Expertní systémy*. Vydání: 2004, VUT Brno
- [2] PROVAZNÍK, Ivo. KOZUMPLÍK, Jiří. *Expertní systémy*. Vydání: 1999, VUT Brno
- [3] HABIBALLA, Hasim. *Umělá inteligence*. Vydání: 2004, Ostravská Univerzita
- [4] DLOUHÝ, Martin. *Exaktní plánování, Bug algoritmy* [online] c.07.11.2003, WINKLER, Zbyněk. *Plánování na mřížce* [online] c.03.12.2003. [cit.k 10.5.2008]. Dostupné na: <http://www.robotika.cz/guide/cs>
- [5] BARTÁK, Roman. *Plánování a rozvrhování* [online] c.2004. [cit.k 26.4.2008]. Dostupné na: <http://ktiml.mff.cuni.cz/~bartak/planovani/lectures/lecture09.pdf>
- [6] MATOUŠEK, Václav. MAUTNER, Pavel. OCELÍKOVÁ, Jana. *Umělá inteligence a rozpoznávání*. [cit.k 26.4.2008]. Dostupné na: <http://www.kiv.zcu.cz/studies/predmety/uir/skripta/index.html>
- [7] ANDRŠ, Ondřej. *Řídící systém pro mobilní kolové roboty*. Diplomová práce. Vydání: 2007 VUT Brno
- [8] ROBOHEMIA, *Robotický fotbal* [online] c.2.8.2007. [cit.k 22.4.2008]. Dostupné na: www.robohemia.cz
- [9] FIRA, *Overview and rules for Robo Soccer MIROSOT* [cit.k 22.4.2008]. Dostupné na: <http://www.fira.net/soccer/mirosot/overview.html>
- [10] HÁJEK, Jan. *Robotický fotbal pro zábavu, výuku i vědeckou práci*. c.2005 [cit.k 22.4.2008]. Dostupné na: <http://www.automatizace.cz/article.php?a=684>
- [11] HLAVÁČ, Václav. *Kognitivní robotika*. c.2007. Studijní materiály ČVUT [cit.k 26.4.2008]. Dostupné na: <http://cmp.felk.cvut.cz/~hlavac/TeachPresCz/51Robotika/>
- [12] JELÍNEK, Libor. *Mobilní robot pro výuku navigačních systémů a strojového vnímání*. ZČU Plzeň, [cit.k 26.4.2008].

Dostupné na: <http://www.mechatronicscentre.eu/content/publications/016-Jelinek.pdf>

- [13] KALUŽA, Radovan. *Expertní systémy – úvod*. c.07.05.2006, [cit.k 26.4.2008].

Dostupné na: <http://radovan.bloger.cz/it/expertni-systemy/expertni-systemy---uvod>

- [14] HÁJEK, Jan. *Robotický fotbal pro zábavu, výuku i vědeckou práci*, c.květen 2005 [cit.k 20.5.2008].

Dostupné na: <http://www.automatizace.cz/article.php?a=684>

- [15] KOŠNÁŘ, Karel. *Mobilní robotika*. c.29.03.2007, [cit.k 22.5.2008].

Dostupné na: <http://www.roznovskastredni.cz/dwnl/pel2007/06/Kosnar.pdf>