

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

FACULTY OF ELECTRICAL ENGINEERING AND
COMMUNICATION
DEPARTMENT OF CONTROL AND INSTRUMENTATION

PRŮBĚŽNÁ LOKALIZACE A MAPOVÁNÍ POMOCÍ MOBILNÍHO ROBOTU

SIMULTANEOUS LOCALIZATION AND MAPPING USING MOBILE ROBOT

DOKTORSKÁ PRÁCE
DOCTORAL THESIS

AUTOR PRÁCE
AUTHOR

ING. TOMÁŠ NEUŽIL

VEDOUCÍ PRÁCE
SUPERVISOR

PROF. ING. FRANTIŠEK ŠOLC, CSC.

BRNO 2008

Poděkování

Na tomto místě bych chtěl poděkovat všem, bez kterých by tato práce nemohla vzniknout. Konkrétně bych chtěl poděkovat prof. Ing. Františku Šolcovi, CSc. za jeho podporu a cenné rady. Také bych chtěl poděkovat svým rodičům a přátelům za důvěru, pomoc a podporu během studia a vzniku práce.

Abstrakt

V práci je uveden rozbor současné lokalizace a mapování (SLAM) v oblasti mobilní robotiky. Na základě tohoto rozboru je navržen algoritmus využívající rozšířeného Kalmanova estimátoru stavů pro odhad polohy robotu. Je zde uveden popis matematického modelu robotu a snímače, který je pro úlohu SLAM využíván. Lokalizační metoda využívá informací o pracovním prostředí mobilního robotu, které jsou získávány pomocí laserového snímače. Význačné rysy popisující prostředí a orientační body pro určení hodnoty stavového vektoru (polohy robotu) jsou z naměřených dat získány pomocí Radonovy transformace, jejíž algoritmus je také součástí práce.

Navržená metoda SLAM byla otestována na datech získaných pomocí mobilního robotu UTAR. V závěru práce je uvedeno zhodnocení dosažených výsledků a možnosti rozšíření navržených algoritmů.

Abstract

This work presents an overview of the simultaneous localisation and mapping (SLAM) problem in the mobile robotics. The Extended Kalman filter (EKF) based algorithm for localisation and mapping is proposed. For EKF algorithm the models of the skid steering mobile robot and the laser scanner are presented. The EKF algorithm is feature based algorithm, therefore the method for the landmark position determination was developed. This segmentation method is based on the clustering of the Radon transform space.

Proposed SLAM algorithm was tested with real data measured with UTAR mobile platform. Achievements of the work are summarized in the conclusion of the proposed work and possible improvements of the components are suggested.

Seznam použitých symbolů

x	x-ová souřadnice polohy robotu
y	y-ová souřadnice polohy robotu
φ	natočení robotu
\mathbf{x}_R	stavový vektor robotu
ϑ	šumový vektor robotu
Q	kovarianční matice vektoru ϑ
\mathbf{y}_R	měření získané pomocí laserového snímače
\mathbf{w}	šumový vektor měření
R	kovarianční matice vektoru w
$\tilde{\mathbf{x}}_R$	odhad stavového vektoru robotu
\mathbf{P}	kovarianční matice odhadu stavového vektoru
$\tilde{\mathbf{y}}_R$	odhad měření
$\tilde{\mathbf{x}}_{(k+1 k)}$	odhad stavového vektoru v kroku $k + 1$ určený pomocí měření získaných v kroku k
$\tilde{\mathbf{x}}_{(k+1 k+1)}$	odhad stavového vektoru v kroku $k + 1$ určený pomocí měření získaných v kroku $k + 1$
$\Delta \mathbf{x}$	odchylka mezi stavovým vektorem a jeho odhadem
$\Delta \mathbf{y}$	vektor inovace
\mathbf{L}	Kalmanova váhová matice
$\chi_{\Delta\Delta}$	kovarianční matice inovačního vektoru
$\chi_{x\Delta}$	vzájemná kovarianční matice pro odhad stavu a inovační vektor
f	nelineární funkce popisující model systému (robotu)
h	nelineární funkce popisující model snímače
∇f_x	matice jakobiánu funkce f vyjádřený podle stavového vektoru \mathbf{x}
∇f_ϑ	matice jakobiánu funkce f vyjádřený podle stavového vektoru ϑ
∇h_x	matice jakobiánu funkce h vyjádřený podle stavového vektoru \mathbf{x}
∇h_ϑ	matice jakobiánu funkce h vyjádřený podle stavového vektoru ϑ
M	mapa prostředí
\mathbf{m}	stavový vektor popisující polohu orientačního bodu
b	poloha bodu odpovídajícímu překážce nacházející se v prostoru (získaná měřením pomocí snímače)
\mathcal{H}	Houghův akumulátor

Obsah

1 Úvod	10
2 Cíle disertační práce	12
3 Navigace	13
3.1 Historie navigace	13
3.2 Navigace v robotice	15
3.3 Závěr	15
4 Lokalizace v robotice	16
4.1 Sledování dráhy	18
4.2 Odometrie, inerciální navigační systém	18
4.3 Lokalizace pomocí orientačních bodů	18
4.4 Odhad polohy mobilního robotu	21
4.5 Závěr	23
5 Mapování v robotice	24
5.1 Vývoj robotického mapování, druhy map	24
5.2 Vytváření mapy robotem	27
5.3 Závěr	28
6 Pravděpodobnostní přístup v robotice	29
6.1 Stavové veličiny	29
6.1.1 Stavové veličiny s přímým vztahem k úloze mapování a lokalizace	
30	
6.2 Poloha robotu jako náhodná veličina	30
6.3 Bayesův filtr	31
6.4 Aproximace Bayesova filtru	33
6.4.1 Popis pomocí Gaussova rozdělení pravděpodobnosti	33
6.4.2 Aproximace rozdělení pravděpodobnosti konečným počtem prvků	35
6.5 Kalmanův filtr - KF	35
6.6 Rozšířený Kalmanův filtr - EKF	40
6.7 Závěr	41

<i>OBSAH</i>	6
7 Systémy pro tvorbu map a lokalizaci	43
7.1 Robotické mapování důlních systémů	43
7.2 Projekt PeLoTe	45
7.3 Závěr	46
8 Model robotu a snímače	47
8.1 Model robotu	47
8.1.1 Kinematický model robotu	48
8.1.2 Ověření matematického modelu	50
8.2 Proximitní snímač	51
8.3 Matematický model snímače	52
8.3.1 Uspořádání experimentu	53
8.4 Závěr	57
9 Detekce význačných znaků zkoumané oblasti	58
9.1 Houghova transformace	61
9.2 Radonova transformace	62
9.3 Určení parametrů hledaných přímk	64
9.3.1 Shlukování nalezených přímek	64
9.3.2 Určení vzdálenosti mezi jednotlivými objekty	65
9.3.3 Rozdělení objektů do shluků	66
9.4 Simulátor pracovního prostředí robotu	68
9.5 Závěr	69
10 Současná lokalizace a mapování	70
10.1 Inicializace	72
10.2 Predikce stavového vektoru a hodnot měření	72
10.3 Hledání korespondujících orientačních bodů	73
10.4 Aktualizace stavového vektoru	74
10.5 Aktualizace globální mapy	74
10.6 Experimentální ověření algoritmu SLAM	75
10.7 Zhodnocení kvality globální mapy	77
11 Shrnutí dosažených cílů	80
12 Příloha A	89

Seznam obrázků

3.1	Dead reckoning	14
4.1	Robot Shakey (1972) [27]	16
4.2	Algoritmus lokalizace robotu pomocí orientačních bodů	19
4.3	Geometrické určení polohy robotu v rovině pomocí triangulační metody	20
4.4	Proces odhadu pomocí Kalmanova filtru	21
4.5	Lokalizace na základě mapy prostředí	22
4.6	Střela Tomahawk (1976) [28]	23
5.1	Geometrická reprezentace prostředí	24
5.2	Reprezentace prostředí pomocí mřížky obsazenosti	25
5.3	Topologická reprezentace prostředí	25
5.4	Topologická reprezentace prostředí s	26
5.5	Objektově orientovaná znalostní mapa [35]	26
6.1	Algoritmus Bayesova filtru	32
6.2	Časová posloupnost kroků pro algoritmus KF (EKF)	42
7.1	Robot Groudhog při mapování důlního systému [39]	44
7.2	2D mapa důlního systému [39]	44
7.3	Příklad mapy získané mapovacím systémem PeNa [46]	45
8.1	Schéma mobilního robotu UTAR	48
8.2	Náhrada robotu řízeného smykem robotem řízeným změnou rychlosti kol	49
8.3	Simulační schéma kinematického modelu pro Matlab Simulink	50
8.4	Měření a simulace pohybu těžiště smykem řízeného robotu a jeho matem- atické náhrady	50
8.5	Matematický model snímače	52
8.6	Prvky matice R	53
8.7	Histogram měření vzdálenosti pro 18% středně šedou plochu při vzdálenost 5m	54
8.8	Testované barvy a odstíny šedé	54
8.9	Histogram měření vzdálenosti pro plochu červené barvy, při vzdálenosti 5m	55

8.10	Závislost rozptylu měření na vzdálenosti	57
9.1	Lokální mapa po převedení do kartézského souř. systému	58
9.2	Normálová reprezentace přímký	61
9.3	Binární obraz z proximitního snímače a obraz Houghova akumulátoru	62
9.4	Porovnání algoritmu Houghovy a Radonovy transformace	62
9.5	Akumulátor parametrů přímek pro Radonovu transformaci měření 9.1	63
9.6	Příklad binárního obrazu a jeho akumulátoru $\mathcal{H}(\alpha, r)$	64
9.7	Algoritmus shlukování	66
9.8	Data z laserového proximitního snímače, Houghův akumulátor se zvýrazněnými shluky	67
9.9	Lokální sensorická a geometrická mapa	67
9.10	Zobrazení výsledku simulace	69
10.1	EKF SLAM	71
10.2	Globální mapa prostředí	75
10.3	Lokální sensorické mapy	76
10.4	Globální geometrická mapa prostředí	77
10.5	Kvadratická chyba odhadu polohy robotu v jednotlivých krocích	78
12.1	Měření polohy barevných štítků umístěných na robotu	89
12.2	Měření polohy barevných štítků umístěných na robotu	90
12.3	Měření polohy barevných štítků umístěných na robotu	90
12.4	Měření polohy barevných štítků umístěných na robotu	91
12.5	Měření polohy barevných štítků umístěných na robotu	91
12.6	Tvar rozptylových funkcí pro jednotlivé vzdálenosti, barva černá	92
12.7	Tvar rozptylových funkcí pro jednotlivé vzdálenosti, barva šedá2 (viz obrázek 8.8)	93
12.8	Tvar rozptylových funkcí pro jednotlivé vzdálenosti, barva šedá3 (viz obrázek 8.8)	94
12.9	Tvar rozptylových funkcí pro jednotlivé vzdálenosti, barva - šedá4 (viz obrázek 8.8)	95
12.10	Tvar rozptylových funkcí pro jednotlivé vzdálenosti, barva - bílá	96
12.11	Tvar rozptylových funkcí pro jednotlivé vzdálenosti, barva - červená	97
12.12	Tvar rozptylových funkcí pro jednotlivé vzdálenosti, barva - fialová	98
12.13	Tvar rozptylových funkcí pro jednotlivé vzdálenosti, barva - modrá	99
12.14	Tvar rozptylových funkcí pro jednotlivé vzdálenosti, barva - žlutá	100
12.15	Tvar rozptylových funkcí pro jednotlivé vzdálenosti, barva - zelená	101

Seznam tabulek

4.1	Lokalizace ve známém prostředí	17
4.2	Výrobce udávaná přesnost snímače řady Garmin GPS 25LP	20
5.1	Veličiny využívané při tvorbě mapy prostředí	24
6.1	Popis veličin	37
6.2	Použité substituce	39
8.1	Závislost rozptylu měření na vzdálenosti	56
10.1	Veličiny využívané při tvorbě mapy prostředí metodou SLAM	70

Kapitola 1

Úvod

Robotika je obor lidské činnosti, který naplňuje snahu člověka usnadnit si práci. Přestože slovo robot pochází z pera českého spisovatele Karla Čapka (Rossums Universal Robots 1920), snaha vyrobit stroj vykonávající lidskou práci, případně stroj, který se podobá člověku, je v lidské historii zaznamenaná mnohem dříve. Za první stroj spadající do kategorie robot jsou označovány vodní hodiny, které sestrojil Ctesibius z Alexandrie v roce 270 př. n. l.¹

Také pražský Golem, člověk z hlíny, naplňuje myšlenku vytvořit stroj podobný člověku a za člověka pracující. Karel Čapek, spolu se svým bratrem Josefem, byli při vzniku dramatu R.U.R. inspirováni pověstí o hliněném Golemovi a masově se rozvíjející průmyslovou výrobou, která vyžadovala pracovníky vykonávající těžkou, stále se opakující činnost².

Prvním moderním robotem, který zastával lidskou práci byl v roce 1962 stacionární robot UNIMATE, který pracoval v automobilovém závodě General Motors v New Jersey. V roce 1969 byl na Stanfordské univerzitě představen první autonomní mobilní robot Shakey. Přestože i dnes je většina robotů stacionárních, mobilní roboty našly své uplatnění v celé řadě odvětví, od průzkumu podmořských hlubin, průvodcovství v muzeích, převoz materiálu v továrnách, až po průzkum vesmírných těles. V neposlední řadě je nutné zmínit se o zábavním průmyslu, ve kterém mobilní roboty mají svoji nezastupitelnou pozici. Zpřístupnění mobilní robotiky mimo výzkumné laboratoře umožnilo její velký rozmach. Do skupiny „robotických hraček“ patří například roboty Aibo a Qrio japonské firmy Sony³. Tyto mobilní roboty, ač mohou sloužit jako hračky, umožňují svým uživatelům programování své činnosti a jsou proto využívány jako učební pomůcka na mnoha univerzitách. Mimo jiné jsou roboty Aibo využívány jako hráči robotického fotbalu na mezinárodních soutěžích, ve kterých mají svoji kategorii.

Mobilní robotika v sobě slučuje celou škálu problémů; od návrhu mechanické kon-

¹Tyto hodiny poháněné proudem vody byly také nazývány clepsydra. V překladu toto slovo znamená „zloděj vody“. Předpokládá se, že Ctesibius byl první, který využil energie proudící vody, kterou pomocí ozubených kol a hnacích pásů převedl na pohyb postaviček a zvonu ohlašujícího časový signál.

²Karel Čapek vnímal roboty jako souhrn psychických a fyzických vlastností, které se podobaly vlastnostem lidským. Roboty nevnímal jako stroje a ve svém díle psal slovo robot s velkým R.

³Výroba a vývoj robotů Aibo a Qrio byla firmou Sony ukončena v roce 2006.

strukce a pohonu robotu, přes vybavení vhodnými snímači a následné zpracování signálu, až po mechanismy umožňující jejich samostatné rozhodování se.

Tato práce se zabývá řešením jedné z podúloh, kterou je třeba úspěšně splnit k tomu, aby mohl vzniknout samostatně pracující stroj. Touto podúlohou je úloha lokalizace. S lokalizací, jako součástí navigační úlohy, velmi úzce souvisí také vytváření mapy pracovního prostoru robotu. Kapitoly 3, 4 a 5 jsou věnovány této problematice s ohledem na různé přístupy využívané pro vytváření map, různé druhy robotických map a následnou lokalizaci v těchto mapách.

Pro vytvoření lokalizačně mapovacího algoritmu byl zvolen pravděpodobnostní přístup využívající rozšířeného Kalmanova estimátoru stavů. Popis této problematiky je uveden v kapitole 6. Byl navržen a realizován algoritmus pro určení polohy robotu. Tento algoritmus využívá pouze data získaná měřením pomocí planárního laserového snímače. Kvůli lokalizaci není tedy třeba zásah člověka do pracovního prostředí robotu. Kapitoly 8 a 10 se zabývají návrhu algoritmu pro současné mapování a lokalizaci. Je uveden návrh matematického modelu použitého robotu a laserového snímače SICK. V práci je také uvedena segmentační metoda pro zpracování naměřených dat z laserového snímače založená na Radonově transformaci. Navržená metoda pro současnou lokalizaci a mapování byla otestována na datech získaných pomocí robotu UTAR, který byl vyvinut v Laboratoři teleprezence a robotiky při VUT v Brně.

Kapitola 2

Cíle disertační práce

Cíle disertační práce jsou:

1. Shrnutí současného stavu oblasti mobilní robotiky s ohledem na systémy a algoritmy využívané pro lokalizaci a tvorbu map prostředí, ve kterém robot vykonává svoji činnost. Hlavní pozornost bude věnována algoritmům, které pro lokalizaci a mapování, případně současnou lokalizaci a mapování, využívají matematického modelu robotu a okolního prostředí.
2. Návrh vlastního algoritmu pro současné mapování a lokalizaci mobilního robotu s důrazem na geometrickou reprezentaci prostředí. Předpokladem je, že mobilní robot bude pro získávání informací o pracovním okolí vybaven pouze planárním laserovým dálkoměrem. Součástí návrhu algoritmu bude vytvoření matematického modelu robotu a použitého snímače.
3. Ověření navrženého algoritmu při simulaci a s využitím reálných dat.

Kapitola 3

Navigace

V této kapitole je uveden rozbor navigační úlohy. Je zde uveden historický vývoj navigačních technik v lidské činnosti v souvislosti s vývojem technických prostředků využívaných pro navigaci. Jelikož jednou ze základních schopností robotických systémů provádějících samostatnou činnost je schopnost orientace v prostoru, jsou v poslední části kapitoly uvedeny základní požadavky, které musí samostatně pracující robot splňovat.

3.1 Historie navigace

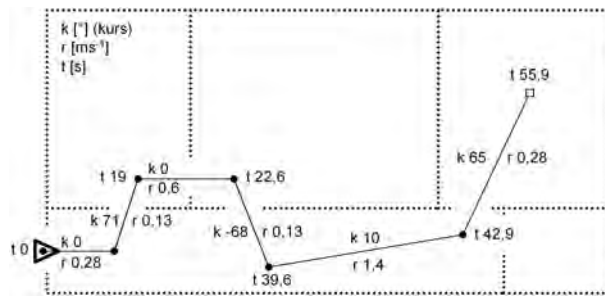
Slovo navigace má základ v latinském slově *navigo*, jehož význam se dá stanovit jako dosažení cíle plavbou (*navis* - loď) [5]. Každý cestovatel, od starověkého mořeplavce až po pilota moderního letounu, potřebuje znát svoji polohu, cíl své cesty a způsob, jakým se do cíle dostane. Tímto způsobem jsou navigace a její podúloha - lokalizace definovány jako:

navigace stanovení přesné polohy cestovatele v prostoru za účelem dosažení požadovaného cíle [1].

lokalizace stanovení přesné polohy cestovatele v prostoru [1].

Nejstarší podoba navigace byla založena na znalosti přesné polohy orientačních bodů v krajině. V místech, kde nebyly viditelné žádné orientační body, byla tato metoda rozšířena o tzv. navigaci výpočtem (DR - *Deduced/ Dead Reckoning*¹).

¹První písemná zmínka o *Dead reckoning* pochází z roku 1613. Slovo *dead* (mrtvý) pravděpodobně označuje plovák lodního rychloměru, jehož poloha vůči vodní hladině je neměnná. Označení *Deduced reckoning* (odvozeno výpočtem) se rozšířilo mezi leteckými navigátory zvláště během 2. světové války.



Obrázek 3.1: Dead reckoning

navigace výpočtem metoda umožňující stanovení polohy na základě známé startovní pozice, rychlosti a směru pohybu (azimutu) [65]. Uvedené veličiny jsou zaznamenávány a s pomocí tohoto záznamu je možné stejnou cestu (trajektorii) kdykoliv opakovat.

Za nejstarší dochovaný záznam cesty se považuje starověký zápis z putování řeckého kupce Pythease z Massalie (350 - 300 př.n.l.). Tehdejší cestovatelé využívali pro navigaci orientačních bodů v krajině. K navigaci byly využívány také hvězdy, avšak o celestiální navigaci v dnešním slova smyslu se hovoří od okamžiku, kdy byl plně pochopen systém pohybu Slunce, hvězd a planet². Pomocí kompasu bylo možno určit kurs plavby, rychlost se určovala pomocí lodního rychloměru a čas byl měřen pomocí přesýpacích hodin. Tento jednoduchý měřicí systém umožnil přesné určení polohy plavidla. Pro záznam námořních plaveb byly používány tzv. lodivodské manuály, ve kterých byly zaznamenány jednotlivé plavby, popsány orientační body, mořské proudy, skaliska apod.

Kolem roku 1730 začal být pro určování výšky hvězdných těles využíván sextant. S pomocí moderního sextantu je možno stanovit polohu plavidla s přesností až 400 m.

Na konci 19. století Nikola Tesla svým objevem rádiových vln vnesl čerstvý vítr do navigačních technik. Přesnou polohu, v této době již nejen lodí, ale také letadel, umožnil stanovit systém majáků tvořených vysílači rádiových stanic se známou polohou.

Kolem roku 1960 byly vyvinuty systémy LORAN a DME pro námořní a leteckou navigaci, které pro navigaci používaly poloautomatické systémy se směrovými anténami pro měření doby letu radiového signálu. V roce 1974 byl jako součást globálního satelitního navigačního systému Námořnictva Spojených států (GPS - Global Positioning System) vypuštěn první satelit. V dnešní době se pro navigaci využívá kombinace rádiových majáků a satelitních navigačních systémů. Navigační systémy našly své uplatnění nejen v oblasti námořní a letecké navigace, ale také v civilním životě v podobě osobních navigačních systémů, kontrolních dopravních systémů, zabezpečovacích systémů apod.

²Základy moderní celestiální navigace stanovil v roce 1837 Thomas H. Summer. Základy celestiální navigace jsou dodnes vyžadovány pro udělení licence v Britském obchodním loďstvu. [1]

3.2 Navigace v robotice

Jedním z cílů mobilní robotiky je vytvoření autonomního mobilního robotu. Aby bylo možno robot považovat za autonomní, musí splňovat následující požadavky:

- schopnost dlouhodobé práce bez nutnosti zásahu člověka.
- schopnost samostatného pohybu v pracovním prostředí.
- schopnost vyhnout se situacím, ve kterých by mohlo dojít k poranění člověka nebo zničení robotu.
- schopnost shromažďovat informace o pracovním prostředí.

Navigace je jedním ze základních prvků mobilní robotiky. Při každém pohybu robotu z místa na místo je nutné si položit otázku „Kde jsem?“ a na tuto otázku získat co nejpřesnější odpověď [3].

V případě systému, který je alespoň částečně autonomní, tzn. bez operátora, který by robotu sděloval, ve kterém místě se nachází, je navíc hledání odpovědi na tuto zdánlivě jednoduchou otázku omezené sensorickým vybavením robotu. Proces navigace lze v oblasti robotiky rozdělit do několika dílčích úloh:

- lokalizace - stanovení polohy robotu (odhad polohy robotu).
- mapování - reprezentace pracovního prostředí robotu
souběžné mapování a lokalizace - (Simultaneous Mapping and Localisation) SLAM.
- plánování trajektorie.

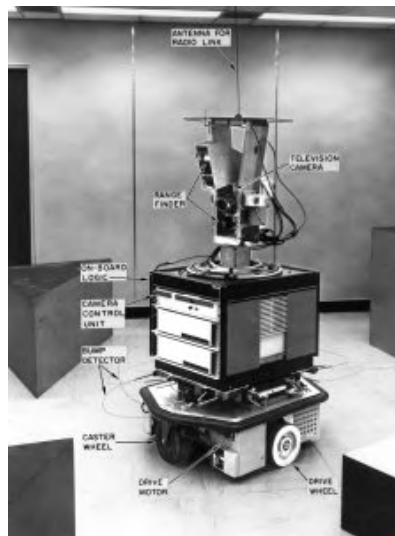
V mobilní robotice lze úlohu navigace rozdělit na dva problémy, které jsou svou složitostí srovnatelné. Jedná se o úlohu lokalizace, tedy stanovení polohy mobilního robotu v globálním souřadnicovém systému a úlohu plánování trajektorie, která využívá dat získaných při mapování. Z výše uvedeného rozdělení je patrné, že veškerá činnost mobilního robotu je závislá na znalosti přesné polohy robotu a na co nejpřesnějším a pro danou činnost nejvhodnějším popisu pracovního prostředí.

3.3 Závěr

V této kapitole jsou uvedeny základní části navigačního systému autonomního mobilního robotu. Lokalizační a mapovací algoritmy budou podrobně rozebrány v následujících kapitolách. Úloha plánování trajektorie zahrnuje širokou škálu otázek, jejichž řešení je postaveno na znalosti polohy robotu. Vzhledem k šíři samotné otázky lokalizace a tvorby map, není plánování trajektorie předmětem této práce.

Kapitola 4

Lokalizace v robotice



Obrázek 4.1: Robot Shakey (1972) [27]

V této kapitole jsou popsány základní principy lokalizace v mobilní robotice tak, jak byly postupně vyvíjeny a využívány při vývoji mobilních robotických systémů s cílem vyrobit alespoň částečně autonomní stroj. Jedním z prvních pokusů o vytvoření autonomního mobilního robotu byl robot Shakey, který byl vyvíjen v letech 1966 - 1972 na Stanfordském výzkumném institutu (SRI) viz obr. 4.1. Tento robot byl vybaven videokamerou, bezdotykovými čidly vzdálenosti a snímači, které mu umožňovaly zpětnou vazbu při jednoduché manipulaci s předměty. Robot byl také schopen, byť v omezené míře, určovat svoji polohu. Robot Shakey byl zároveň schopen vytvořit jednoduchou mapu prostředí.

Na lokalizaci mobilního robotu lze nahlížet jako na problém transformace souřadných systémů. Řešením tohoto problému je nalezení transformace, která umožňuje určení shody mezi částí globální mapy prostředí a lokální mapou získanou měřením robotu.

- lokální mapa - počátek souřadného systému je spjat s mobilním robotem, např. střed laserového snímače,
- globální mapa - počátek souřadného systému je spjat s prostředím, ve kterém se robot pohybuje, např. roh místnosti, vstup do budovy apod.

V tabulce 4.1 jsou uvedeny veličiny, které jsou využívány v souvislosti s lokalizací.

Tabulka 4.1: Lokalizace ve známém prostředí

<i>veličina</i>	<i>hodnota</i>
poloha robotu	neznámá
mapa prostředí poloha orientačních bodů	známá

V závislosti na veličinách, které jsou známy při inicializaci a během lokalizačního algoritmu lze lokalizaci rozdělit do několika skupin [66, 18]:

- lokální - startovní poloha robotu je známá (v okamžiku startu je známý vztah mezi lokálním a globálním souřadným systémem), poloha robotu v jednotlivých krocích je vyhodnocována na základě znalosti polohy v předešlém kroku a měření ze snímačů umístěných na robotu. V zahraniční literatuře je tento druh lokalizace označován jako sledování polohy (*Position Tracking*).
- globální - lokalizační algoritmus robotu odhaduje polohu pouze na základě měření v okamžiku odhadu. K dispozici není žádná znalost o předešlé poloze robotu. V tomto případě nelze předpokládat žádná omezení při stanovení polohy robotu. (V předchozím případě je toto omezení dáno alespoň částečnou znalostí modelu robotu, vlivu šumového signálu snímačů na určení polohy apod.).
- unesený robot - v tomto případě lokalizační algoritmus vychází z předpokladu, že se robot nachází na jiném místě, než je tomu ve skutečnosti. Tento zdánlivě nesmyslný případ je vhodný obzvláště pro testování robustnosti lokalizačních algoritmů a jejich schopnosti vhodně reagovat na kritická selhání při globální lokalizaci. V zahraniční literatuře se tato lokalizační úloha nazývá *kidnapped robot*.

Reálným příkladem tohoto stavu je extrémní hodnota prokluzu kol mobilního robotu. Přestože je pomocí snímačů měřen pohyb kol, který je vyhodnocován jako pohyb daným směrem, poloha robotu zůstává neměnná. Vyhodnocení polohy a přiřazení odpovídajících si orientačních bodů, které probíhá na základě předpokládané polohy robotu, může být příčinou selhání lokalizačního algoritmu.

4.1 Sledování dráhy

Tyto robotické systémy se pohybují po předem dané trajektorii, kterou je senzorický systém mobilního robotu schopen detekovat. Tato navigační technika umožňuje použití poměrně jednoduchého senzorického systému a neklade zvláštní nároky na výpočetní výkon. Tato lokalizační metoda je v zahraniční literatuře označována jako *Line Following*. Nejčastěji využívanými typy drah jsou elektrické vodiče, kterými protéká střídavý elektrický proud, a ten je snímán pomocí cívky umístěné na mobilním robotu. Trajektorie mobilního robotu je předem dána uložením vodičů v podlaze a robot není schopen přesně určit vlastní polohu ve kterémkoliv okamžiku. Tento způsob lokalizace také vyžaduje zásah do prostředí (položení kabelů, namalování barevných pruhů, a pod.), což může být v některých případech nákladné nebo neproveditelné.

Přes uvedené nevýhody jsou tyto systémy využívány pro dopravu materiálu v automatizovaných provozech, nemocnicích a pod. V neposlední řadě se využívají také jako vhodná výuková úloha pro studenty kurzů robotiky a v disciplíně „Path Follower“ jsou pořádány mezinárodní studentské soutěže. Příkladem může být soutěž Istrobot, pořádaná na TU Bratislava, ve které sledování dráhy tvoří jednu z disciplín [6].

V současné době se některá vědecká pracoviště začínají zabývat systémy pro sledování pachové stopy. Pachovou stopu využívají pro svoji orientaci zvířata a hmyz. Výzkum v této oblasti se soustředí na vývoj systému, který by umožňoval lokalizaci a mapování míst úniku jedovatých látek v nebezpečných prostředích [4, 13].

4.2 Odometrie, inerciální navigační systém

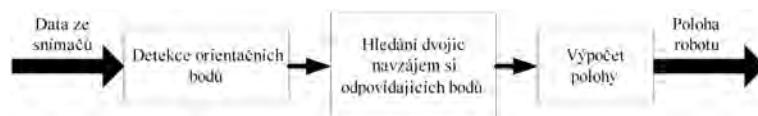
Další možnou metodou lokalizace mobilního robotu je odometrie, metoda využívající měření otáček kol robotu. Pro určení polohy robotu se využívá znalosti rozměrů kol a geometrického uspořádání podvozku. Tato metoda však nezahrnuje vlivy prostředí na robot jako jsou například prokluz kol, smýkání kol při zatáčení apod. a je využitelná pouze u kolových podvozků. Nezávislost na druhu robotického podvozku nabízí systém inerciální navigace, který pro určení polohy využívá měření zrychlení ve všech stupních volnosti a jejich následnou dvojí integraci.

Hlavní nevýhodou inerciálního navigačního systému je skutečnost, že s užitečnou informací, která odpovídá měřenému zrychlení, se integruje také chyba měření. Obě zmíněné metody jsou samostatně nepoužitelné z důvodů rostoucí chyby stanovované polohy robotu.

4.3 Lokalizace pomocí orientačních bodů

Orientační body (v anglické literatuře označované jako *landmarks*) jsou místa s charakteristickými rysy v pracovní oblasti mobilního robotu, která je robot schopen detekovat pomocí vlastních snímačů. Tyto oblasti mohou mít podobu geometrických útvarů, jako jsou čáry, kruhy, mnohoúhelníky apod., popřípadě mohou být vytvořeny uměle, například formou rádiových vysílačů [12]. Základním předpokladem metody

je neměnná poloha orientačních bodů během procesu lokalizace. Lokalizace pomocí orientačních bodů je v zahraniční literatuře označována jako *Beacon Navigation*. Orientační body lze rozdělit do následujících skupin:



Obrázek 4.2: Algoritmus lokalizace robotu pomocí orientačních bodů

přirozené orientační body orientační body, které byly vytvořeny za jiným účelem než je navigace robotu. Typickým příkladem jsou: rohy místností, dlouhé chodby, okna atd.,

umělé orientační body orientační body, které byly navrženy a vytvořeny pouze za účelem navigace robotu. Typickým příkladem jsou černobílé obrazce (kamerové navigační systémy), terče s vysokou odrazivostí (optické laserové systémy), aktivních vysílačů (rádiové systémy),

aktivní orientační body vysílače, které vysílají signál umožňující stanovení polohy robotu (vysílače GPS),

pasivní orientační body odražeče, terče s definovanými rozměry a pod.

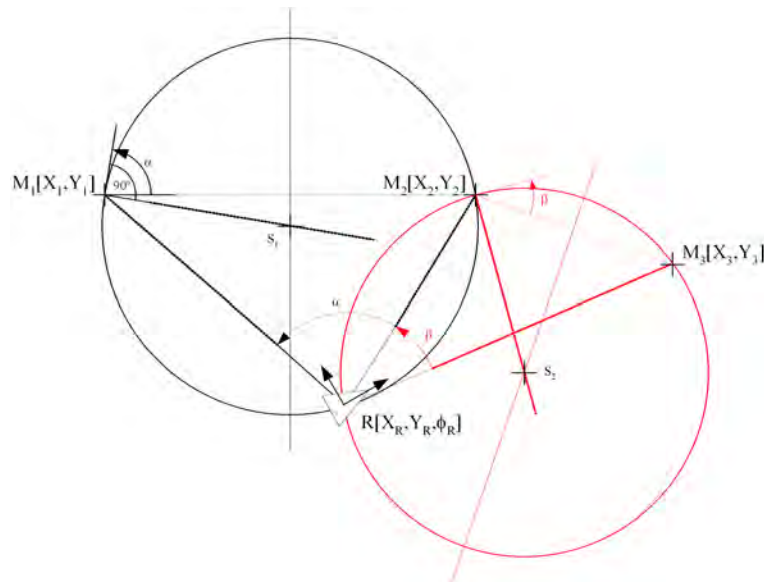
Navigace pomocí umělých orientačních bodů

Naprostá většina lokalizačních systémů využívá pro určení polohy mobilního robotu porovnávání polohy orientačních bodů s předem známou mapou, ve které jsou uloženy polohy těchto orientačních bodů. Za takovou mapu může být považována předem daná statická mapa prostředí nebo mapa získávaná během pohybu robotu pomocí jeho senzorického vybavení [10].

Podle přístupu k měřeným datům lze metody rozdělit do dvou skupin. Jedná se o metody triangulační a metody pro odhad polohy na základě známého modelu robotu, snímačů a řídicích signálů mobilního robotu.

Triangulační metoda

Pokud jsou k dispozici úhly mezi jednotlivými orientačními body a robotem, je možné na základě jednoduchých geometrických pravidel určit přesnou polohu robotu v okamžiku měření. Tato metoda je využívána pro navigaci podle hvězd, kdy měříme úhel mezi hvězdou, pozorovatelem a horizontem a ze známého průběhu polohy hvězd na obloze je možné určit polohu pozorovatele.



Obrázek 4.3: Geometrické určení polohy robotu v rovině pomocí triangulační metody

Pro přesné určení polohy robotu x_R, y_R a jeho natočení φ_R je dostačující znalost o vzájemné poloze nejméně tří orientačních bodů - majáků (M_1, M_2, M_3) viz obr. 4.3.

Global Positioning System (GPS)

GPS systém využívá tzv. trilaterální metodu, která je založena na znalosti vzdáleností mezi přijímačem a jednotlivými satelity (vysílači). Každý ze satelitů vysílá zprávu, ve které je zakódována okamžik jejího vyslání. Vzdálenost satelitu od přijímače je stanovena na základě doby mezi vysláním a přijmutím zprávy a známé rychlosti šíření signálu (každý ze satelitů obsahuje přesný synchronizovaný zdroj hodinového signálu).

Na základě geometrického průniku množin možných poloh přijímače signálu je následně možné určit jeho polohu [7, 8]. Nevýhodou systému GPS je jeho nevhodnost pro použití uvnitř budov, což je způsobeno nízkým výkonem vysílače signálu. Systém GPS také nelze využít v místech, kde není v dosahu dostatečný počet satelitů. Pro využití v mobilní robotice může být omezujícím faktorem také přesnost systému GPS.

Tabulka 4.2: Výrobce udávaná přesnost snímače řady Garmin GPS 25LP

Režim	Přesnost měření polohy
GPS	lepší než 15 m
DGPS (RTCM)	lepší než 5 m

DGPS (Differential Global Positioning System) metoda umožňující zpřesnění měřené polohy. Pro zvýšení přesnosti určované polohy jsou využívána data získána měřením pomocí přístroje, jehož přesná poloha je známá a u nějž se předpokládá viditelnost podobné skupiny satelitů jako u pohyblivého přijímače (určené polohy obsahují stejné chyby). Informace o chybě polohy jsou přenesena do

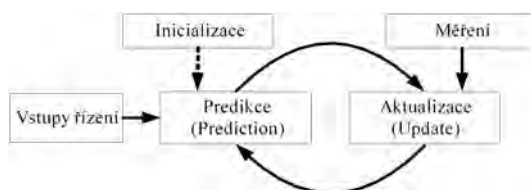
přijímače, jehož přesná (neznámá) poloha je určena na základě vyhodnocení obou měření s větší přesností. Důvodem nepřesností v měření polohy mohou být například atmosférické poruchy, odchylky v synchronizaci času vysílače a přijímače apod.

Příkladem může být systém WAAS/EGNOS (pro využití na území USA/Evropy). Tento systém se skládá z citlivých přijímačů GPS signálu. Tyto přijímače zpracovávají GPS signál, vyhodnocují chyby a informace o těchto chybách poskytují pomocí geostacionárních družic uživatelům. GPS přijímače proto musí být pro tuto možnost vybaveny také přijímačem signálu z družic WAAS/EGNOS. S využitím těchto systémů lze stanovit polohu s přesností cca. 3m.

Dalším omezujícím prvkem při využití GPS může být, zvláště v případě rychle se pohybujících robotů, nízká frekvence (5 Hz) aktualizace polohy satelitů. Toto omezení je dáno šířkou pásma, ve kterém je GPS signál vysílán.

4.4 Odhad polohy mobilního robotu

Polohu mobilního robotu nelze odečíst přímo na výstupu ze senzoru, který je využíván pro navigaci, proto je cílem získat alespoň její pravděpodobný odhad. Pro odhad polohy je využíváno částečné znalosti matematického modelu robotu, používaného snímače (snímačů může být libovolné množství), řídicích vstupů robotu a skutečných měření. Odhad polohy je následně prováděn na principu Bayesovy podmíněné pravděpodobnosti. Za nejznámější zástupce skupiny algoritmů pro odhad polohy mobilního robotu (aproximace Bayesova filtru) lze bezpochyby považovat Kalmanův filtr a jeho modifikace [11], který provádí odhad polohy na základě minimalizace střední kvadratické chyby odhadu polohy robotu. Princip odhadu stavového vektoru pomocí Kalmanova filtru je uveden na obrázku 4.4.

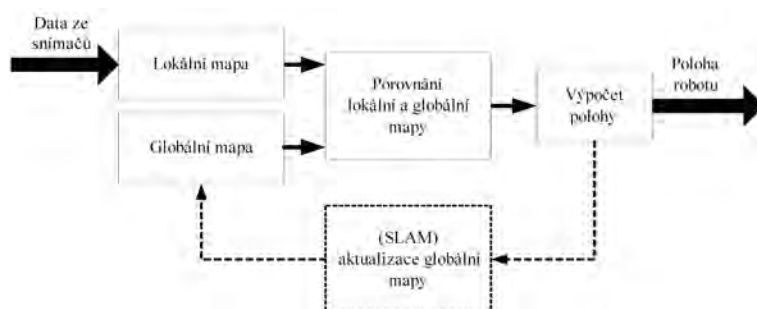


Obrázek 4.4: Proces odhadu pomocí Kalmanova filtru

Současná lokalizace a mapování, SLAM

Současná lokalizace a mapování (*Simultaneous Localisation and Mapping - SLAM*) - proces, při kterém robot vytváří mapu prostředí, kterou zároveň využívá k určení své pozice. Jedná se o situaci, kdy je robot umístěn do zcela neznámého prostředí a v tomto prostředí má být schopen vyvíjet nějakou, alespoň částečně autonomní činnost. Principem je kontinuální tvorba mapy prostředí při využití dat ze snímačů na robotu umístěných. Shodná vstupní data jsou využívána jak pro tvorbu mapy, tak pro lokalizaci v této mapě.

Při této metodě vytváří robot tzv. *lokální mapu* prostředí, kterou následně porovnává s *globální mapou* uloženou v paměti, a určuje svoji polohu. Kvalita odhadu polohy na základě mapy závisí na schopnosti získávat a efektivně využívat informace o svém okolí [14, 15, 16].



Obrázek 4.5: Lokalizace na základě mapy prostředí

Kombinace navigačních metod

Ve většině případů jsou tyto metody kombinovány a využívány k vzájemnému zpřesňování údajů o poloze. Příkladem využívání informací z jednotlivých lokalizačních systémů je střela Tomahawk, jejíž vývoj byl zahájen v roce 1972 (obdobně jako robot Shakey). Tento létající autonomní robot pro určení své polohy využívá kombinaci následujících metod [28]:

- *navigace výpočtem (DR)* - při zahájení mise je střela řízena podle předprogramovaných kroků,
- *data ze snímačů*
data ze snímačů jsou po zpracování porovnávána s předem vytvořenou mapou. Porovnání provádí jednotky:
 - TERCOM - porovnávání geografických vlastností terénu (od roku 1976),
 - DSMAC - porovnávání obrazových informací o prostředí. Data jsou přizpůsobena například denní době mise,
- *inerciální navigační systém*,
- *GPS - DGPS* - (od roku 1993).



Obrázek 4.6: Střela Tomahawk (1976) [28]

Obdobně jako se vyvíjely přístupy pro lokalizaci, vyvíjely se i způsoby pro popis pracovního prostředí robotu. Tyto dvě pod-úlohy, ač jsou uvedeny odděleně, spolu velmi úzce souvisí a většinou jsou řešeny souběžně.

4.5 Závěr

V této kapitole jsou uvedeny základní lokalizační metody, jejich principy, výhody a nevýhody. Ve většině reálných robotických systémů jsou jednotlivé druhy lokalizačních přístupů kombinovány navzájem.

Při volbě lokalizační metody je třeba brát zřetel na prostředí, pro které je mobilní robot určen. Lokalizační metoda pro robot pohybující se v tovární hale např. umožňuje využití umělých orientačních bodů, zatímco pro lokalizaci v neznámém prostředí je nutné, aby robot spoléhal na měření získaná vlastním sensorickým systémem. Návrh SLAM algoritmu, který je uveden v kapitole 10, tento předpoklad zahrnuje, a pro lokalizaci jsou používána pouze data z laserového snímače. Toto prostředí je však strukturované (vnitřní prostředí budov), což usnadňuje nalezení vhodných orientačních bodů.

Kapitola 5

Mapování v robotice

V této kapitole je uvedeno základní rozdělení a popis robotických map. V tabulce 5.1 jsou uvedeny veličiny, které se vztahují k úloze tvorby mapy.

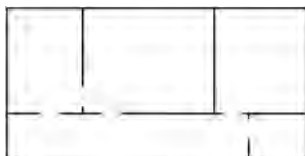
Tabulka 5.1: Veličiny využívané při tvorbě mapy prostředí

<i>veličina</i>	<i>hodnota</i>
poloha robotu	známá
mapa prostředí poloha orientačních bodů	neznámá

5.1 Vývoj robotického mapování, druhy map

V 80. letech 20. století se problematika mapování v mobilní robotice rozdělila na dva základní způsoby [24]:

Geometrický - při tomto způsobu reprezentace je mapa ukládána jako sada geometrických primitiv popisujících prostředí. Na obrázku 5.1 je znázorněno kancelářské

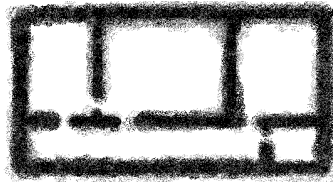


Obrázek 5.1: Geometrická reprezentace prostředí

prostředí. Jednotlivé místnosti jsou ohraničeny úsečkami a průchody mezi místnostmi jsou ponechány volné.

Zásadním přínosem pro oblast geometrických map bylo využití mřížek obsazenosti. Tento přístup popisuje prostředí jako dostatečně jemnou mřížku, jejíž jednotlivé buňky obsahují informaci, zda je buňka obsazená, či nikoliv. Mřížky obsazenosti (Occupancy grids) poprvé pro mapování v mobilní robotice zmiňuje Alberto Elfes v roce 1989 [25]. Tuto techniku rozvinul také Hans Moravec v [26]. Metoda

umožňuje akumulovat informaci o obsazenosti buňky v prostoru na základě Bayesova teorému o podmíněné pravděpodobnosti. Každá buňka této mapy obsahuje hodnotu pravděpodobnosti toho, zda je obsazená překážkou, kterou je možno zjistit za pomoci použitých snímačů. Měření těmito snímači (v době vzniku metody byly využívány ultrazvukové snímače) je popsáno pomocí pravděpodobnostního modelu snímače.

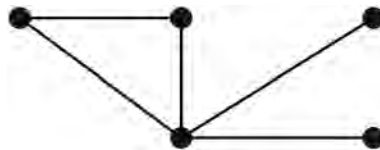


Obrázek 5.2: Repräsentace prostředí pomocí mřížky obsazenosti

Tento způsob zobrazení prostoru je velmi podobný tomu, jak by danou oblast zobrazil člověk. Porozumění tomuto druhu mapy je proto velmi intuitivní a usnadňuje jak využití výsledků robotického mapování, tak zadávání předem stanovené mapy do navigačního systému robotu.

Z tohoto důvodu jsou výsledky robotického mapování velmi často žádány právě v tomto formátu a to nejen v případě geodetických přístrojů, které jsou určeny pro vytváření map.

Topologický - tento přístup popisuje okolní prostředí robotu pomocí vztahů mezi jednotlivými, pro navigaci robotu důležitými, místy. Jednotlivá „význačná místa“ reprezentují uzly grafu. Spojnice mezi těmito uzly pak popisují způsob, kterým se lze mezi body pohybovat. Na obrázku 5.3 uzly reprezentují jednotlivé místnosti, spojnice pak reprezentují průchody mezi těmito místnostmi.

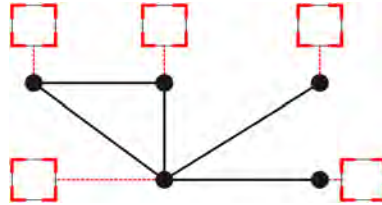


Obrázek 5.3: Topologická reprezentace prostředí

Třetím způsobem, který je částečnou kombinací obou předchozích, doplněným o další informace je přístup:

Znalostní - do mapy je zaznamenávána znalost o prostředí (*cognitive map*). Na rozdíl od geometrických a topologických reprezentací, které jsou primárně určené pro navigaci robotu, jsou tyto mapy pokusem o vysvětlení toho, jaké objekty se nachází v okolí robotu. Zatímco při geometrickém popisu prostředí jsou typicky využívány rohy a stěny místností, při popisu znalostním jsou využívány entity typu: dveře, stůl, skříň, atd. [35].

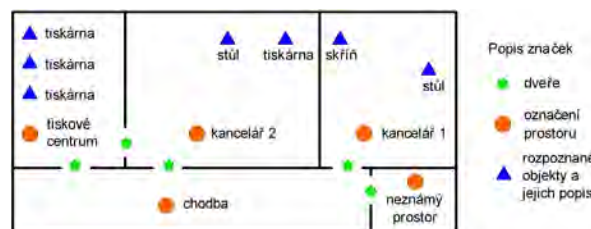
Ve skutečnosti nejsou první dvě skupiny odděleny ostrou hranicí. Většina topologických algoritmů pracuje s informacemi, které jsou využívány pro tvorbu geometrických map. Topologické mapy popisují pracovní prostor robotu a jsou doplněny o geometrický popis význačných oblastí prostoru. Tato místa jsou pak využívána lokalizačním algoritmem pro stanovení polohy, případně její zpřesnění. Těmito orientačními body jsou většinou míněny rohy místnosti [31].



Obrázek 5.4: Topologická reprezentace prostředí s metrickými prvky

Obrázek 5.1 znázorňuje hybridní mapu prostředí, které je shodné jako prostředí na obr. 5.1. Globální mapa prostředí je topologická a je shodná s topologickou mapou popsanou výše. Tato globální topologická mapa je doplněna o lokální geometrické mapy, které jsou znázorněny pomocí rohů (červená barva). Z každého uzlu vede červená čára, která ukazuje, že k uzlu jsou příslušné nejen topologické spojnice, tedy informace jak se dostat do dalšího uzlu, ale i informace o geometrických vlastnostech význačných oblastí v prostoru, v zobrazeném případě se jedná o rohy místnosti. Tyto význačné oblasti jsou spolu taktéž spojeny topologickou vazbou (tenká černá čára), která říká, jakým způsobem jsou rohy místnosti uspořádány.

Dalším příkladem mapy, která v sobě zahrnuje jak geometrické, tak topologické informace, mohou být mapy vytvořené člověkem. Přestože nejsou vytvářeny přímo robotem, jsou k jejich tvorbě využívány stroje (např. pro letecké snímkování) a jsou vytvářeny pomocí technik používaných i v mobilní robotice (triangulace, trilaterace, fotogrammetrie, apod.) [30]. Na těchto mapách jsou pomocí geometrických primitiv zobrazeny uzly - (budovy, volná prostranství, lesy, vodní zdroje, ...) a spojnice - (cesty, vodní toky, ...), které popisují způsob, jakým se lze dostat z jednoho místa k druhému. Tyto informace jsou navíc člověku interpretovány pomocí barevného odlišení jednotlivých druhů uzlů i jejich spojnic.



Obrázek 5.5: Objektově orientovaná znalostní mapa [35]

Znalostní přístup k robotickému mapování se svým výsledkem nejvíce blíží způsobu, jakým při průzkumu prostředí postupují zvířata a člověk [35, 36]. Člověk při průzkumu prostoru vytváří „mapu“ informací o jednotlivých objektech a jejich vzájemných vazbách. Tyto informace využívá k snadnému zapamatování a pochopení zkoumaného prostoru. Příklad znalostní mapy prostředí je uveden na obrázku 5.5.

5.2 Vytváření mapy robotem

K vytvoření mapy prostředí musí být robot vybaven snímači, které mu umožňují získat potřebné informace o prostředí. Nejčastěji používanými senzory pro vytváření robotických map jsou: kamery, laserové dálkoměry (planární nebo 3D), GPS přijímače, radary apod. Výstupy z většiny těchto snímačů jsou omezeny (např. přijímače GPS signálu nefungují uvnitř budov) a výsledky měření jsou zatíženy chybami a šumem. Také pohyb robotu je zatížen šumem, reprezentovaným vlivy okolí robotu, které při řízení jeho pohybu nejsou brány v úvahu, z čehož vyplývá skutečnost, že polohu robotu nelze určit z řídicích signálů, které do robotu vstupují.

V případě, že by oba šumové signály byly na sobě nezávislé, přesnost mapy by bylo možno zvýšit pomocí zvýšení počtu měření a jejich vhodnou statistickou analýzou. V reálných robotických systémech jsou však signály statisticky závislé. Odchytky polohy způsobené řídicím signálem, který je zatížen šumem, rostou v čase a ovlivňují tak polohu, která je určována z měření jednotlivých snímačů. Tyto vypočítané polohy jsou zatíženy systematickou chybou.

Tvorba robotických map je komplikována obtížností určení, zda data naměřená v různých časových okamžicích zobrazují stejná místa v prostoru. Tento problém je v literatuře velmi často zmiňován jako *problém korespondence* měření (correspondence problem), nebo jako problém *sdužování dat* (data association problem) [24, 33]. Hlavním důsledkem problému sdužování dat je skutečnost, že s rostoucím časem a počtem měření může celková odchylka polohy překročit mez, která je pro lokalizační a mapovací algoritmus přijatelná (závisí na typu aplikace). Výpočetní náročnost určení korespondujících bodů měření je proto omezujícím faktorem pro některé algoritmy. Tyto algoritmy dokáží zpracovat pouze omezený počet prvků, čímž je omezena velikost mapy, pro kterou je tato skupina SLAM algoritmů využitelná [34].

Dalším důležitým aspektem při vytváření robotické mapy je dynamika prostředí. Reálné prostředí většinou prochází změnami. Při mapování kancelářského prostředí lze za takovou změnu považovat například: pohyb osob v kanceláři, změnu počtu knih v knihovně, nebo změnu stavu dveří z otevřených na zavřené. Tuto skutečnost pak může lokalizační algoritmus vyhodnotit dvěma způsoby:

- poloha objektů se změnila a mapovací algoritmus může aktualizovat mapu prostředí,
- lokalizační algoritmus ze změněné polohy objektů nesprávně vyhodnotí polohu a dojde k jeho selhání.

Většina současných mapovacích algoritmů předpokládá statické prostředí a všechny dynamické změny v prostředí jsou vyhodnocovány jako šumový signál. Důsledkem

tohoto předpokladu je, že lokalizační a mapovací algoritmy jsou funkční v relativně krátkých časových intervalech, ve kterých je prostředí neměnné.

Také strategie průzkumu prostředí je důležitým prvkem při robotickém mapování. V neznámém prostředí musí robot nejen vytvářet mapu a lokalizovat se v ní, ale zároveň musí v této mapě být schopen naplánovat optimální trajektorii. Tato trajektorie musí být naplánována nejen s ohledem na délku, energetickou a časovou náročnost, ale také s ohledem na její vhodnost pro získání informací pro vytvoření mapy a s uvážením rizika, že robot nebude schopen tuto informaci získat a ztratí se ve zkoumaném prostoru. Z tohoto úhlu pohledu lze lokalizačně mapovací algoritmy (LAM) rozdělit na aktivní a pasivní:

- pasivní algoritmy pro lokalizaci a mapování - data z lokalizačního algoritmu nejsou využívána pro plánování trajektorie. V tomto případě je LAM využíván pouze jako pozorovatel prostředí,
- aktivní algoritmy pro lokalizaci a mapování - pro řízení pohybu mobilního robotu jsou využívána data z LAM. Tento přístup umožňuje minimalizovat ztráty způsobené navedením robotu do riskantních oblastí, umožňuje aktivně plánovat proces navigace tak, aby při průzkumu byla prozkoumána celá oblast, umožňuje částečné řešení lokalizace a mapování symetrických prostředí, apod. [69].

5.3 Závěr

V této kapitole jsou uvedeny základní druhy map, které jsou používány v mobilní robotice spolu s jejich výhodami a nevýhodami. Také je zde uvedeno rozdělení LAM algoritmů. Z uvedených možností reprezentace pracovního prostředí robotu je pro člověka nejpřirozenější reprezentace geometrická. Tento druh mapy usnadňuje komunikaci mezi lidskou obsluhou a robotickým systémem. Geometrickou mapu lze využít nejen jako výstup z robotického systému, ale v případě známého prostředí lze takovou mapu snadno vytvořit (plány budov, geodetické mapy, atp.) a usnadnit tak úlohu lokalizace. Z tohoto důvodu byl tento druh reprezentace zvolen také pro algoritmus navržený v kapitole 10.

Kapitola 6

Pravděpodobnostní přístup v robotice

Při reálném provozu mobilního robotu jsme většinou nuceni se spolehnout na informace o okolním prostředí, které si je robot schopen sám získat pomocí zabudovaných snímačů. Snímače umístěné na mobilním robotu nejsou ve většině případů schopny dodat přesnou informaci o stavu robotu. Vzhledem k tomu, že hlavním tématem práce je úloha lokalizace a mapování, stavem robotu je myšlena poloha robotu v prostoru a poloha překážek, které se v prostoru nacházejí. Tato skutečnost je dána tím, že jednotlivé principy měření mají svá omezení a výsledky měření jsou zatíženy šumem.

6.1 Stavové veličiny

Jedním z možných přístupů ke zpracování hodnot měřených snímači je tzv. *pravděpodobnostní přístup* (probabilistic robotics). Robot a prostředí, ve kterém se pohybuje, je popsán pomocí stavových veličin. Stavové veličiny lze rozdělit do dvou skupin v závislosti na tom, zda jsou v čase konstantní, nebo proměnné:

- statické stavové veličiny - veličiny, které jsou během činnosti robotu konstantní. Do této skupiny patří například: teplota sledovaného objektu, barva objektu, poloha objektů, (zdi, sloupy, okna, . . .) ,
- dynamické stavové veličiny - veličiny, které jsou proměnné během činnosti robotu. Typickým zástupcem této skupiny je rychlost překážky pohybující se v pracovním prostředí robotu.

Mezi hlavní stavové veličiny využívané při lokalizaci a mapování patří:

- poloha robotu - stavová veličina popisující polohu robotu v globálním souřadném systému. V případě, že se jedná o robot pohybující se v rovině, je poloha robotu dána souřadnicemi popisující polohu (x, y) a natočení robotu (φ) ,

- poloha a stav objektů (překážek) v pracovním prostředí robotu. Ve většině případů se jedná o statické veličiny, které popisují polohu a stav význačných oblastí, bodů v okolí robotu. Do této skupiny náleží také poloha a rychlost pohybující se překážky (např. osoby pohybující se v zorném poli robotu),
- význačné oblasti - oblasti v pracovním prostředí robotu, které jsou spolehlivě rozpoznatelné a snadno matematicky popsitelné.

Na tomto místě by bylo možno uvést nespočetné množství možných veličin popisujících stav robotu a jeho prostředí (stav napájecího systému, teplota pohonných jednotek, teplota okolního prostředí, složení ovzduší, apod.). Tyto veličiny však nemají přímý vztah k úloze lokalizace nebo mapování (popřípadě SLAM).

6.1.1 Stavové veličiny s přímým vztahem k úloze mapování a lokalizace

- $\mathbf{x}_k = (x, y, \varphi)$ stavový vektor popisující polohu a orientaci mobilního robotu v prostoru v kroku k . Souřadnice x, y popisují polohu robotu v kartézském souřadném systému a φ je úhel natočení robotu,
- \mathbf{u}_k vektor řídicí veličiny, který způsobí přechod systému do stavu \mathbf{x}_k . Tento vektor může obsahovat požadované hodnoty rychlosti otáčení jednotlivých kol robotu, u kolových podvozků je v praktických realizacích vstup \mathbf{u}_k velmi často nahrazen výstupem z inkrementálních snímačů umístěných na kolech robotu. Tímto opatřením lze eliminovat odchylky regulátoru pro řízení motorů robotu,
- \mathbf{o}_k měření (pozorování - observation) orientačních bodů v prostoru prováděné v časovém kroku k ,
- $M = (m_1, m_2 \dots m_n)$ mapa, množina stavových vektorů příslušejících jednotlivým orientačním bodům,
- m_i vektor popisující polohu orientačního bodu. Poloha jednotlivých orientačních bodů je časově konstantní $m_i = (x_i, y_i)$.

6.2 Poloha robotu jako náhodná veličina

Na vývoj polohy robotu, stejně tak jako na měření prováděná robotem, lze pohlížet jako na náhodný proces. Hodnota stavového vektoru \mathbf{x}_t je závislá na vstupním řídicím signálu \mathbf{u}_t a předchozí hodnotě stavu systému \mathbf{x}_{t-1} a lze ji vyjádřit jako $p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t)$. Tato podmíněná pravděpodobnost je nazývána *pravděpodobností přechodu stavů* a popisuje průběh stavu \mathbf{x}_t v čase [18].

Obdobným způsobem lze popsat také *pravděpodobnost měření*, jako: $p(\mathbf{o}_t | \mathbf{x}_t)$. Z tohoto zápisu je patrné, že hodnota měření je závislá pouze na poloze robotu v prostoru. Tato pravděpodobnost je popisem předpisu, podle něhož jsou generována

měření \mathbf{o}_i na základě polohy robotu \mathbf{x}_i . Zjednodušeně lze o měřeních \mathbf{o}_i říci, že se jedná o projekci stavu \mathbf{x}_i , která je zatížena šumem.

Jelikož polohu robotického systému nelze stanovit přímo, je nutné tuto polohu odvodit z měřených dat a předpokládaného stavu (polohy) robotu.

Předpokládaný stav robotu je aposteriorní pravděpodobnost vyjadřující hodnotu stavu po provedení akčního zásahu do robotického systému. V tomto případě je pro určení stavu robotu využívána hodnota měření \mathbf{o}_t . Obdobně lze stanovit také hodnotu aposteriorní pravděpodobnosti před započítáním hodnoty pozorování \mathbf{o}_t jako $p(\mathbf{x}_t | \mathbf{o}_{1:t-1}, \mathbf{u}_{1:t})$. Tyto dvě aposteriorní pravděpodobnosti jsou v literatuře označovány jako (state of knowledge, information state), případně (belief) a jsou vyjádřením vnitřní znalosti stavu robotu.

- aposteriorní (hledaná) pravděpodobnost zahrnující hodnotu měření:

$$p(\mathbf{x}_t | \mathbf{o}_{1:t}, \mathbf{u}_{1:t}) \quad (6.1)$$

- aposteriorní pravděpodobnost nezahrnující hodnotu měření. Tato pravděpodobnost je nazývána *predikcí stavu* systému:

$$p(\mathbf{x}_t | \mathbf{o}_{1:t-1}, \mathbf{u}_{1:t}) \quad (6.2)$$

Přechod od aposteriorní pravděpodobnosti (6.2) k aposteriorní pravděpodobnosti (6.1) se pak nazývá *aktualizací (korekcí) stavu* systému. Způsob přechodu mezi těmito dvěma pravděpodobnostmi je popsán v následujícím textu.

6.3 Bayesův filtr

Obecným způsobem výpočtu aposteriorních pravděpodobností je Bayesův filtr. Pomocí tohoto algoritmu lze určit hodnotu aposteriorní pravděpodobnosti $p(\mathbf{x}_t | \mathbf{o}_{1:t}, \mathbf{u}_{1:t})$ na základě aposteriorní pravděpodobnosti z předcházejícího kroku $p(\mathbf{x}_{t-1} | \mathbf{o}_{1:t-1}, \mathbf{u}_{1:t-1})$. Podle Bayesovy věty [17] lze hledanou hodnotu vyjádřit jako:

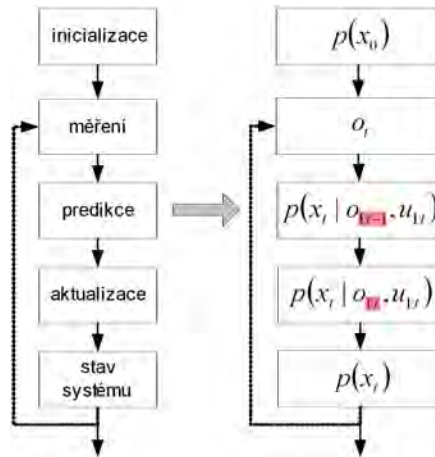
$$p(\mathbf{x}_t | \mathbf{o}_{1:t}, \mathbf{u}_{1:t}) = \frac{p(\mathbf{o}_t | \mathbf{x}_t, \mathbf{o}_{1:t-1}, \mathbf{u}_{1:t}) p(\mathbf{x}_t | \mathbf{o}_{1:t-1}, \mathbf{u}_{1:t})}{p(\mathbf{o}_t | \mathbf{o}_{1:t-1}, \mathbf{u}_{1:t})} \quad (6.3)$$

Na základě předpokladu, že hodnoty předchozích stavů, řídicí vstupy a měření neobsahuje žádnou informaci, která by umožnila zpřesnit predikci stavu, lze pro první část čitatele zlomku napsat:

$$p(\mathbf{o}_t | \mathbf{x}_t, \mathbf{o}_{1:t-1}, \mathbf{u}_{1:t}) = p(\mathbf{o}_t | \mathbf{x}_t) \quad (6.4)$$

po této úpravě je možno psát:

$$p(\mathbf{x}_t | \mathbf{o}_{1:t}, \mathbf{u}_{1:t}) = \frac{p(\mathbf{o}_t | \mathbf{x}_t) p(\mathbf{x}_t | \mathbf{o}_{1:t-1}, \mathbf{u}_{1:t})}{p(\mathbf{o}_t | \mathbf{o}_{1:t-1}, \mathbf{u}_{1:t})} \quad (6.5)$$



Obrázek 6.1: Algoritmus Bayesova filtru

druhou část čitatele (6.3) je pak možné pomocí věty o úplné pravděpodobnosti [17] vyjádřit jako:

$$p(\mathbf{x}_t | \mathbf{o}_{1:t-1}, \mathbf{u}_{1:t}) = \int p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{o}_{1:t-1}, \mathbf{u}_{1:t}) p(\mathbf{x}_{t-1} | \mathbf{o}_{1:t-1}, \mathbf{u}_{1:t}) dx_{t-1} \quad (6.6)$$

za stejného předpokladu, který byl použitý pro zjednodušení rovnice (6.3), lze i tuto rovnici zjednodušit na tvar:

$$p(\mathbf{x}_t | \mathbf{o}_{1:t-1}, \mathbf{u}_{1:t}) = \int p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t) p(\mathbf{x}_{t-1} | \mathbf{o}_{1:t-1}, \mathbf{u}_{1:t-1}) dx_{t-1} \quad (6.7)$$

algoritmizovaný výpočet hledané hodnoty (stavu systému) je zobrazen na obrázku 6.1: Pro zjednodušení 6.4, 6.6 je využíváno předpokladu, že měřená data a stav systému v nadcházejícím časovém okamžiku $t+1$ jsou závislá pouze na stavu systému v časovém okamžiku t . Tato skutečnost říká, že na robotický systém je možno nahlížet jako na Markovův proces [17].

Při využití tohoto algoritmu je však nutné si uvědomit, že podmínka úplné stavové reprezentace systému, která vyplývá z předpokladu, že se jedná o systém splňující vlastnosti Markovova procesu, je málokdy splněna. Nesplnění podmínky vyplývá z faktu, že v reálném případě většinou nejsme schopni zahrnout všechny vlivy, které na systém (mobilní robot) působí.

Bayesův filtr, tak jak byl uveden výše, není prakticky využitelný z důvodu, že nelze implementovat na číslicovém počítači. Hlavní principy jsou však shodné i pro algoritmy, které pro výpočet aposteriorní pravděpodobnosti $p(\mathbf{x}_t | \mathbf{o}_{1:t}, \mathbf{u}_{1:t})$ využívají aproximace.

Základní vlastnosti algoritmu jsou:

- algoritmus je rekurzivní - pro určení stavu systému \mathbf{x}_t využívá hodnotu stavu

\mathbf{x}_{t-1} ,

- hodnota stavu \mathbf{x}_t je počítána jako aposteriorní pravděpodobnost na základě věty o pravděpodobnosti příčin [17]. Pro výpočet jsou využívány pouze hodnoty předchozích stavů a vstupních signálů (řídící signál, vstup měření).

6.4 Aproximace Bayesova filtru

Analytické vyjádření a výpočet aposteriorní pravděpodobnosti $p(\mathbf{x}_t | \mathbf{o}_{1:t}, \mathbf{u}_{1:t})$ jsou možné pouze v omezeném počtu případů. Aby bylo možno aplikovat Bayesovský filtr v reálném robotickém systému, je zapotřebí využít vhodné aproximace této pravděpodobnosti. Pro aproximaci se využívá několik základních druhů popisu: [18, 19, 20].

6.4.1 Popis pomocí Gaussova rozdělení pravděpodobnosti

U tohoto způsobu aproximace se k popisu aposteriorní pravděpodobnosti $p(x_t | o_{1:t}, u_{1:t})$ využívá n -rozměrného Gaussova rozdělení a jeho základních charakteristik:

$$p(x) = \det(2\pi\Sigma)^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}(x - \mu)^T \Sigma^{-1}(x - \mu)\right\} \quad (6.8)$$

- Σ - kovarianční matice. Symetrická, pozitivně definitní matice, jejíž prvky mají hodnotu σ_{ij}
- μ - vektor středních hodnot. Algoritmy, které využívají tohoto zjednodušeného popisu, jsou nazývány gaussovskými filtry. Mezi nejznámější zástupce této skupiny patří:

Kalmanův filtr - *KF* pro parametrizaci Gaussova rozdělení je využíváno jeho prvních dvou momentů - *středních hodnot* a *rozptylů*. Hlavní nevýhodou Kalmanova filtru (přestože splňuje všechny podmínky k tomu, aby ho možno využít jako aproximaci Bayesovského filtračního algoritmu) je, že tento algoritmus je vhodný pouze pro spojité, lineární systémy.

Rozšířený Kalmanův filtr - *EKF* (*Extended Kalman Filter*). Jelikož funkce pro přechody mezi jednotlivými stavy a závislosti měření většinou nejsou lineární, vyvstává potřeba linearizace přechodové funkce. Algoritmus EKF využívá pro aproximaci nelineární funkce Taylorova rozvoje, ze kterého využívá pouze koeficienty pro první řád.

Rozptylový Kalmanův filtr - *UKF* (*Unscented Kalman Filter*). Tento algoritmus aproximuje nelineární přechodovou funkci přímým výpočtem charakteristických bodů. Charakteristické body odpovídají rozptylu rozdělení náhodné veličiny. Hodnoty těchto bodů jsou následně dosazeny do nelineární funkce a výsledné hodnoty použity pro rekonstrukci výsledného Gaussova rozdělení. Velkou výhodou tohoto filtračního algoritmu je, že odpadá nutnost výpočtu matice jakobiánu při linearizaci pomocí Taylorova rozvoje.

Informační filtr - *IF*. Hlavní rozdíl mezi algoritmy vycházejícími z Kalmanova filtru a informačním filtrem je způsob reprezentace aposteriorní pravděpodobnosti $p(\mathbf{x}_t \mid \mathbf{o}_{1:t}, \mathbf{u}_{1:t})$. Zatímco prvně zmíněné algoritmy využívají pro popis momentů rozdělení pravděpodobnosti (střední hodnoty a kovariance), informační filtr využívá kanonické parametrizace vícerozměrného Gaussovského rozdělení. Tato parametrizace je definována pomocí matice Ω , která se nazývá *informační matice* (matice přesnosti) a její hodnotu lze určit pomocí: $\Omega = \Sigma^{-1}$ a informačního vektoru ξ , který je vyjádřen jako: $\xi = \Sigma^{-1}\mu$. Tento algoritmus je duální k algoritmu KF, z čehož vyplývá omezení využití pouze pro lineární systémy.

Rozšířený informační filtr - *EIF*. Tento algoritmus je rozšířením algoritmu IF pro nelineární systémy. Tento algoritmus je snadno využitelný při slučování dat při spolupráci více robotů. Při takovéto spolupráci jsou data z jednotlivých robotů slučována pomocí Bayesova vzorce. Jelikož informační filtr využívá kanonické parametrizace, pravděpodobnost je reprezentována v logaritmické podobě, což umožňuje slučování dat pomocí jednoduché operace sčítání. Mezi hlavní nevýhody však patří skutečnost, že v kroku aktualizace aposteriorního rozdělení pravděpodobnosti, je potřeba určit převrácenou hodnotu informační matice, což je u matic s velkým počtem prvků velmi náročné na výpočetní čas.

Částicový filtr - (*Particle Filter*). Algoritmus částicového filtru pro popis hustoty pravděpodobnosti využívá vícenásobných kopií stavových proměnných $X_t (x_t^1, x_t^2, \dots, x_t^n)$. Při vytváření kopie stavové veličiny (částice) je každé kopii přiřazen váhový koeficient w_t^n , který popisuje pravděpodobnost toho, že pro hodnotu částice x_t^n nastane pozorování \mathbf{o}_t . Jedná se tedy o popis hustoty pravděpodobnosti $p(x_t^n \mid \mathbf{o}_{1:t}, \mathbf{u}_{1:t})$. Následně jsou váhy jednotlivých částic nastaveny s ohledem na model systému, šum systému, apod. Tento krok se nazývá *predikce*. V kroku nazývaném *aktualizace* jsou hodnoty přepočítány v závislosti na hodnotách získaných měření a částice s malými váhovými koeficienty jsou odstraněny (převzorkování). Fáze výpočtu váhových koeficientů jsou shodné s fázemi výpočtu Kalmanova filtru, viz kapitola 6.5. Výhodou tohoto přístupu je možnost aproximace nelineární hustoty pravděpodobnosti stavové veličiny a možnost nastavení počtu částic. Nastavením počtu částic lze velmi snadno ovlivnit rychlost a přesnost výpočtu. V některých případech může dojít k situaci, v níž jsou hodnoty všech váhových koeficientů natolik malé, že jsou všechny částice odstraněny. Tato skutečnost je jednou z hlavních nevýhod částicového filtru [18, 67].

Souhrnný Kalmanův filtr - *EnKF (Ensemble Kalman Filter)*. Algoritmus, který byl navržen pro aplikace, v nichž stavový vektor a jeho kovarianční matice mají vysoký počet prvků. Tento algoritmus je využíván v oblasti zpracování dat z modelů využívaných pro vytváření předpovědi počasí. Stavový vektor tvoří tzv. souhrn (ensemble). S tímto souhrnem je nakládáno jako s náhodnou veličinou, přestože jednotlivé prvky souhrnu nejsou nezávislé veličiny. V případě, že jednotlivé částice představují nejlepší možný odhad stavové veličiny (střední hodnota) a jejich váhové koeficienty jsou nastaveny tak, aby představovaly rozptyl

rozložení náhodné veličiny (viz kapitola 6.4.1), odpadá nutnost výpočtu kovarianční matice. Jedná se v podstatě o částicový filtr, který aproximuje gaussovské rozložení pravděpodobnosti [68].

6.4.2 Aproximace rozdělení pravděpodobnosti konečným počtem prvků

Jedná se o rozdělení aposteriorní pravděpodobnosti konečným počtem prvků. Hlavní výhodou tohoto přístupu je možnost volby počtu prvků, což s sebou přináší možnost volby přesnosti a časové náročnosti výpočtů [18]. Dvě základní metody jsou:

Diskrétní Bayesův filtr v případě, že stav robotu \mathbf{x}_t nabývá pouze konečného počtu hodnot, je spojitý Bayesův filtr nahrazován filtrem diskrétním. Typickou robotickou aplikací je úloha mapování a lokalizace pomocí mřížek obsazenosti.[18] Pro predikci a aktualizaci každé stavové veličiny platí následující vztahy:

$$\tilde{p}(\mathbf{x}_{k,t}) = \sum_i p(X_t = \mathbf{x}_k | \mathbf{u}_k, X_{t-1} = \mathbf{x}_i) p(\mathbf{x}_{i,t-1}) \quad (6.9)$$

$$p(\mathbf{x}_{k,t}) = \frac{p(\mathbf{o}_t | X_t = \mathbf{x}_k | \mathbf{u}_k, X_{t-1} = \mathbf{x}_i) \tilde{p}(\mathbf{x}_{k,t})}{p(\mathbf{o}_t | \mathbf{o}_{1:t-1}, \mathbf{u}_{1:t})} \quad (6.10)$$

Tyto vztahy odpovídají fázi predikce a aktualizace stavového vektoru, obdobně jako u 6.7, 6.5. V případě, kdy se pomocí diskrétního Bayesova filtru nahrazuje spojitá stavová veličina, je takový filtr nazýván *histogramovým*.

Algoritmus histogramového filtru rozděluje stavový prostor na konečný počet oblastí. Pro každou oblast je určena souhrnná hodnota, která reprezentuje rozdělení aposteriorní pravděpodobnosti.

Částicový filtr algoritmus částicového filtru je uveden v sekci 6.4.

6.5 Kalmanův filtr - KF

Pro úlohu současného mapování a lokalizace byl jako vhodný aproximační algoritmus Bayesovského filtru zvolen Kalmanův filtr a jeho modifikace pro nelineární případy - EKF. Tato volba byla provedena na základě skutečnosti, že algoritmy vycházející z KF jsou jednoduché a výpočetně efektivní. Zatímco výpočetní náročnost rozšířeného Kalmanova filtru je kvadraticky závislá na rozměru stavového vektoru, u částicového filtračního algoritmu je tato závislost exponenciální.

Vzhledem k tomu, že EKF tvoří základní prvek lokalizačního algoritmu, je tento estimátor stavu zevrubně popsán v následujících kapitolách. Uvedený popis vychází z [8, 11, 22, 18].

Hlavní myšlenkou algoritmu Kalmanova filtru je spojení algoritmu zpětnovazebního estimátoru pro odhad stavu systému s rekurzivním výpočtem minimalizace odchylky metodou nejmenších čtverců. Algoritmus Kalmanova filtru předpokládá, že model mobilního robotu je popsán stavovými rovnicemi. Tyto rovnice vyjadřují

lineární diskrétní popis systému:

$$\mathbf{x}_{(k+1)} = f[\mathbf{x}_{(k)}, \mathbf{u}_{(k)}, \vartheta_{(k)}] \quad (6.11)$$

z tohoto popisu je patrné, že systém splňuje podmínku bayesovské filtrace a to, že hodnota stavu $\mathbf{x}_{(k+1)}$ je závislá na hodnotě předchozího stavu $\mathbf{x}_{(k)}$ a hodnotě vstupního signálu $\mathbf{u}_{(k)}$ (6.3). Veličiny vstupující do výpočtů jsou zatíženy náhodnými ději, které zatěžují stavovou veličinu, proto je potřeba s těmito jevy uvažovat i při návrhu algoritmu filtru.

Jednotlivé složky vektoru šumu systému $\vartheta_{(k)}$ jsou popsány Gaussovým rozdělením s nulovou střední hodnotou μ_ϑ . Šumovému vektoru také přísluší kovarianční matice $Q_{(k)}$.

$$\vartheta_{(k)} = N(0, Q_{(k)}) \quad (6.12)$$

$$E[\vartheta_{i(k)}\vartheta_{j(k)}^T] = Q_{(k)} \quad (6.13)$$

Vektor výstupu systému je:

$$\mathbf{y}_{(k)} = h[\mathbf{x}_{(k)}, \mathbf{w}_{(k)}] \quad (6.14)$$

Druhým šumovým vektorem je vektor šumu měření w_k . Tento vektor v sobě zahrnuje přesnost snímačů, které jsou využívány pro lokalizaci mobilního robotu, a je popsán rovnicí:

$$\mathbf{w}_k = N(0, R_{(k)}) \quad (6.15)$$

$$E[\mathbf{w}_{i(k)}\mathbf{w}_{j(k)}^T] = R_{(k)} \quad (6.16)$$

Důležitou vlastností obou šumových signálů je, že jsou nezávislé navzájem a také jsou nezávislé na prvcích stavového vektoru systému, z čehož vyplývá následující skutečnost:

$$E[\vartheta_{i(k)}\mathbf{x}_{j(k)}^T] = 0 \forall i, j$$

$$E[\mathbf{w}_{j(k)}\mathbf{x}_{j(k)}^T] = 0 \forall i, j$$

$$E[\vartheta_{j(k)}\mathbf{w}_{j(k)}^T] = 0 \forall i, j$$

kde $E[a_{i(k)}b_{j(k)}]$ je matice vyjadřující hodnotu kovariance obou vektorů [17].

Odhad skutečného stavu systému $\mathbf{x}_{(k+1)}$ je označován jako $\tilde{\mathbf{x}}_{(k+1|k)}$. Tento zápis značí, že odhad je prováděn v časovém okamžiku $k+1$ na základě posloupnosti měření získaných v čase k . Tuto posloupnost měření lze popsat jako: $Y^{k+1} = [y_{(1)}, y_{(2)}, \dots, y_{(k)}]^1$. Pro odchylku mezi skutečnou hodnotou stavu $\mathbf{x}_{(k+1)}$ a jeho odhadem $\tilde{\mathbf{x}}_{(k+1|k)}$ platí:

$$\Delta\mathbf{x}_{(k+1|k)} = \mathbf{x}_{(k+1)} - \tilde{\mathbf{x}}_{(k+1|k)} \quad (6.17)$$

¹Jedná se o stejnou posloupnost výstupů, která je v kapitole 6.2 popsána písmenem O . Změna popisu byla provedena vzhledem ke konvencím používaným pro značení výstupu systému v oblasti simulace, modelování a řízení.

Kovarianční matice (*rozptylová matice* [17]) rozdílu mezi skutečným stavem robotu a jeho odhadem je vyjádřena jako:

$$P_{(k+1|k)} = E \left[\Delta \mathbf{x}_{(k+1|k)} \Delta \mathbf{x}_{(k+1|k)}^T \right] \quad (6.18)$$

Algoritmus Kalmanova filtru umožňuje najít takovou hodnotu odhadu stavového vektoru, pro niž bude tato odchylka minimální.

KF pro lineární systémy

Maticovým zápisem lze mobilní robot popsat jako dynamický systém:

$$\mathbf{x}_{(k+1)} = F_{(k)} \mathbf{x}_{(k)} + G_{(k)} \mathbf{u}_{(k)} + \vartheta_{(k)} \quad (6.19)$$

$$\mathbf{y}_{(k)} = H_{(k)} \mathbf{x}_{(k)} + \mathbf{w}_{(k)} \quad (6.20)$$

rovnice matematického modelu mobilního robotu pro odhad stavového vektoru jeho kovarianční matice jsou:

$$\tilde{\mathbf{x}}_{(k+1|k)} = F_{(k)} \tilde{\mathbf{x}}_{(k|k)} + G_{(k)} \mathbf{u}_{(k)} \quad (6.21)$$

$$P_{(k+1|k)} = F_{(k)} P_{(k|k)} F_{(k)}^T + Q_{(k)} \quad (6.22)$$

Význam jednotlivých prvků, které jsou použity pro popis, je uveden v tabulce 6.1.

Tabulka 6.1: Popis veličin

F	matice soustavy
G	matice buzení
H	matice výstupu
$x_{(k)}$	stav systému
$u_{(k)}$	vstup řízení
$v_{(k)}$	šum systému
$w_{(k)}$	šum měření

V případě, že do rovnice pro odhad stavu systému vhodným způsobem zavedeme další členy, jsme schopni rovnici měnit a ovlivňovat výpočet stavu systému. Člen, vy-povídající o kvalitě odhadu stavu, je obsažen v rovnici pro výstupní hodnotu systému (6.22), který je:

$$\Delta \mathbf{y}_{(k)} = \mathbf{y}_{(k)} - \tilde{\mathbf{y}}_{(k)} \quad (6.23)$$

a nazývá se *vektorem inovace*. Tento rozdíl je zaveden pomocí matice L do rovnice pro odhad stavu (6.21). Vhodná volba matice L ² umožní ovlivnit odhad stavu tak, aby jeho odchylka od stavu skutečného byla minimální. Algoritmus výpočtu matice L je nazýván Kalmanovým filtrem - estimátorem stavu.

²V literatuře je tato matice někdy označována písmenem W jako matice váhová. Označení L vychází z Luenbergerova deterministického estimátoru stavu. Struktura Kalmanova estimátoru je shodná s Luenbergerovým estimátorem, způsob výpočtu váhové matice je však odlišný.

$$\begin{aligned}
\tilde{\mathbf{x}}_{(k+1|k+1)} &= F\tilde{\mathbf{x}}_{(k|k)} + G\mathbf{u}_{(k)} + L_{(k+1)} [\mathbf{y}_{(k|k)} - \tilde{\mathbf{y}}_{(k|k)}] \\
&= F\tilde{\mathbf{x}}_{(k|k)} + G\mathbf{u}_{(k)} + L_{(k+1)} [\mathbf{y}_{(k|k)} - H\tilde{\mathbf{x}}_{(k|k)}] \\
&= \tilde{\mathbf{x}}_{(k+1|k)} + L_{(k+1)} \Delta \mathbf{y}_{(k)}
\end{aligned} \tag{6.24}$$

Návrh matice L je prováděn metodou rekurzivní minimalizace čtverců odchylek. Odchylka odhadu stavové veličiny od skutečné hodnoty je dána rovnicí:

$$\begin{aligned}
\Delta \mathbf{x}_{(k+1|k)} &= \mathbf{x}_{(k+1)} - \tilde{\mathbf{x}}_{(k+1|k)} \\
\Delta \mathbf{x}_{(k+1|k)} &= F\mathbf{x}_{(k)} + G\mathbf{u}_{(k)} + \vartheta_{(k)} - \{F\tilde{\mathbf{x}}_{(k|k)} + G\mathbf{u}_{(k)} + L_{(k+1)} \Delta \mathbf{y}_{(k)}\} \\
&= F\mathbf{x}_{(k)} - F\tilde{\mathbf{x}}_{(k|k)} + \vartheta_{(k)} - L_{(k+1)} \Delta \mathbf{y}_{(k)} \\
\Delta \mathbf{x}_{(k+1|k)} &= \Delta \mathbf{x}_{(k|k)} - L_{(k+1)} \Delta \mathbf{y}_{(k)}
\end{aligned} \tag{6.25}$$

pro kovarianční matici vyjadřující střední hodnotu chyby odhadu hodnoty stavových veličin platí:

$$\begin{aligned}
P_{(k+1|k+1)} &= E \left[\Delta \mathbf{x}_{(k+1|k+1)} \Delta \mathbf{x}_{(k+1|k+1)}^T \right] \\
&= E \left\{ \left[\Delta \mathbf{x}_{(k|k)} - L_{(k+1)} \Delta \mathbf{y}_{(k)} \right] \left[\Delta \mathbf{x}_{(k|k)} - L_{(k+1)} \Delta \mathbf{y}_{(k)} \right]^T \right\} \\
&= E \left[\Delta \mathbf{x}_{(k|k)} \Delta \mathbf{x}_{(k|k)}^T \right] + E \left[L_{(k+1)} \Delta \mathbf{y}_{(k)} \Delta \mathbf{y}_{(k)}^T L_{(k+1)}^T \right] \\
&\quad - E \left[\Delta \mathbf{x}_{(k|k)} \Delta \mathbf{y}_{(k)}^T L_{(k+1)}^T \right] - E \left[L_{(k+1)} \Delta \mathbf{y}_{(k)} \Delta \mathbf{x}_{(k|k)} \right] \\
P_{(k+1|k+1)} &= P_{(k+1|k)} + L_{(k+1)} \left[\Delta \mathbf{y}_{(k)} \Delta \mathbf{y}_{(k)}^T \right] L_{(k+1)}^T \\
&\quad - E \left[\Delta \mathbf{x}_{(k|k)} \Delta \mathbf{y}_{(k)}^T \right] L_{(k+1)}^T - L_{(k+1)} E \left[\Delta \mathbf{y}_{(k)} \Delta \mathbf{x}_{(k|k)} \right]
\end{aligned} \tag{6.26}$$

Pro zpřehlednění zápisu jsou zavedeny substituční matice vyjadřující hodnotu kovarianční matice inovačního vektoru 6.27 a vzájemné kovariance pro chybu predikce stavu a inovačního vektoru 6.28:

$$\chi_{\Delta\Delta(k+1|k)} = E \left[\Delta y_{(k)} \Delta y_{(k)}^T \right] \tag{6.27}$$

$$\begin{aligned}
\chi_{x\Delta(k+1|k)} &= E \left[\Delta x_{(k|k)} \Delta y_{(k)}^T \right] \\
&= E \left[\Delta y_{(k)} \Delta x_{(k|k)}^T \right]
\end{aligned}$$

$$\chi_{x\Delta(k+1|k)} = \chi_{x\Delta(k+1|k)}^T \tag{6.28}$$

Pro hodnotu váhové matice $L_{(k+1)}$ platí:

$$L_{(k+1)} = \chi_{x\Delta(k+1|k)} \chi_{\Delta\Delta(k+1|k)}^{-1} \tag{6.29}$$

Hodnota této matice poskytuje optimální odhad stavového vektoru ve smyslu nejmenších čtverců odchylek. Váhová matice L je určena z hodnoty kovarianční matice

mezi chybou odhadu a matice stavového vektoru a kovariance vektoru inovace. Pro aktualizovanou hodnotu stavového vektoru (po uvážení hodnot získaných měření) a jeho kovarianční matici platí:

$$\tilde{\mathbf{x}}_{(k+1|k+1)} = \tilde{\mathbf{x}}_{(k+1|k)} + L_{(k+1)} \Delta \mathbf{y}_{(k+1)} \quad (6.30)$$

$$P_{(k+1|k+1)} = P_{(k+1|k)} - L_{(k+1)} \chi_{\Delta\Delta(k+1|k)} L_{(k+1)}^T \quad (6.31)$$

Rovnice 6.26 až 6.31 popisují způsob, jakým se projevuje využití modelu systému v algoritmu výpočtu Kalmanova Fistru. Místo toho, aby se využívalo přímé hodnoty výstupu z modelu, je využíván vektor inovace 6.23 a hodnota jeho kovarianční matice 6.27. Hodnoty vektorů využitých pro substituci v aktualizacím kroku jsou:

$$\begin{aligned} \Delta \mathbf{y}_{(k+1)} &= \mathbf{y}_{(k+1)} - \tilde{\mathbf{y}}_{(k+1|k)} \\ &= \mathbf{y}_{(k+1)} - H \Delta \mathbf{x}_{(k+1|k)} \end{aligned} \quad (6.32)$$

$$\begin{aligned} \chi_{\Delta\Delta(k+1|k)} &= H P_{(k+1|k)} H^T + R_{(k+1)} \\ \chi_{x\Delta(k+1|k)} &= P_{(k+1|k)} H^T \end{aligned} \quad (6.33)$$

KF shrnutí

- rovnice popisující lineární systém:

$$\begin{aligned} \mathbf{x}_{(k+1)} &= F \mathbf{x}_{(k)} + G \mathbf{u}_{(k)} + \vartheta_{(k)} \\ \mathbf{y}_{(k)} &= H \mathbf{x}_{(k)} + \mathbf{w}_{(k)} \end{aligned}$$

- odhad stavového vektoru a jeho kovarianční matice:

$$\begin{aligned} \tilde{\mathbf{x}}_{(k+1|k)} &= F \tilde{\mathbf{x}}_{(k|k)} + G \mathbf{u}_{(k)} \\ P_{(k+1|k)} &= F P_{(k|k)} F^T + Q_{(k)} \end{aligned}$$

- aktualizace stavového vektoru a kovarianční matice:

$$\begin{aligned} \tilde{\mathbf{x}}_{(k+1|k+1)} &= \tilde{\mathbf{x}}_{(k+1|k)} + L_{(k+1)} \Delta \mathbf{y}_{(k+1)} \\ P_{(k+1|k+1)} &= P_{(k+1|k)} - L_{(k+1)} \chi_{\Delta\Delta(k+1|k)} L_{(k+1)}^T \end{aligned} \quad (6.34)$$

kde platí:

Tabulka 6.2: Použité substituce

vektor inovace	$\Delta \mathbf{y}_{(k+1)} = \mathbf{y}_{(k+1)} - H \Delta \mathbf{x}_{(k+1 k)}$	
kovariance vektoru inovace	$\chi_{\Delta\Delta(k+1 k)} = H P_{(k+1 k)} H^T + R_{(k+1)}$	
váhová matice	$L_{(k+1)} = P_{(k+1 k)} H^T \chi_{\Delta\Delta(k+1 k)}^{-1}$	(z rovnice 6.33)

Kompletní odvození algoritmu Kalmanova filtru lze nalézt v [8, 22, 18]. Výše uvedený algoritmus Kalmanovy filtrace je platný za několika podmínek, z nichž některé v

mobilitní robotice velmi často nejsou splněny. Jednou z těchto podmínek je požadavek na linearitu systému. V případě, že se nejedná o lineární systém, je nutné aplikovat algoritmus tzv. *rozšířeného Kalmanova filtru (EKF)*, který pro splnění podmínky lineárního systému využívá linearizace pomocí Taylorova rozvoje prvního řádu.

6.6 Rozšířený Kalmanův filtr - EKF

Předpokladem této metody je, že model systému je lokálně lineární, hodnoty procesního $\vartheta_{(k)}$ a měřicího $\mathbf{w}_{(k)}$ šumu jsou dostatečně malé a že lze provést odhad stavu systému na základě předchozích výstupů. Stavové rovnice systému a jeho výstupu jsou:

$$\mathbf{x}_{(k+1)} = f(\mathbf{x}_{(k)}, \mathbf{u}_{(k)}, \vartheta_{(k)}) \quad (6.35)$$

$$\mathbf{y}_{(k)} = h(\mathbf{x}_{(k)}, \mathbf{w}_{(k)}) \quad (6.36)$$

Linearizace modelu pomocí Taylorova rozvoje za předpokladu, že prvky vyššího řádu jsou zanedbatelné (využíváme pouze lineární části), umožňuje využití algoritmu KF. Linearizované modely systému a snímače lze popsat jako:

$$\mathbf{x}_{(k+1)} = f(\tilde{\mathbf{x}}_{(k|k)}, \mathbf{u}_{(k)}, 0) + \nabla f_x \Delta \mathbf{x}_{(k|k)} + \nabla f_\vartheta \vartheta_{(k)} \quad (6.37)$$

$$\mathbf{y}_{(k+1)} = h(\tilde{\mathbf{x}}_{(k+1|k)}, 0) + \nabla h_x \Delta \mathbf{x}_{(k+1|k)} + \nabla h_w \mathbf{w}_{(k+1)} \quad (6.38)$$

kde matice $\nabla f_x, \nabla f_\vartheta, \nabla h_x, \nabla h_w$ jsou matice Jakobiánů funkcí f_x a f_ϑ , jejichž rozvoj je proveden podle \mathbf{x} respektive ϑ a \mathbf{w} a jejich hodnota je vyjádřena pro $\Delta \mathbf{x}_{(k|k)}, \Delta \mathbf{x}_{(k+1|k)}$ respektive pro $\vartheta_{(k)}, \mathbf{w}_{(k+1)}$.

Rovnice umožňující odhad stavu systému je:

$$\tilde{\mathbf{x}}_{(k+1|k)} = f(\tilde{\mathbf{x}}_{(k|k)}, \mathbf{u}_{(k)}, 0) \doteq E[f(\tilde{\mathbf{x}}_{(k|k)}, \mathbf{u}_{(k)}, 0) + \nabla f_x \Delta \mathbf{x}_{(k|k)} + \nabla f_\vartheta \vartheta_{(k)}] \quad (6.39)$$

jestliže platí, že hodnoty $\Delta \mathbf{x}_{(k|k)}$ a $\vartheta_{(k)}$ jsou dostatečně malé a mají nulovou střední hodnotu, lze hodnotu odchylky vyjádřit jako:

$$\begin{aligned} \Delta \mathbf{x}_{(k+1|k)} &= \mathbf{x}_{(k+1)} - \tilde{\mathbf{x}}_{(k+1|k)} \\ &= \nabla f_x \Delta \mathbf{x}_{(k|k)} + \nabla f_\vartheta \vartheta_{(k)} \end{aligned}$$

kovarianční matice odhadu stavového vektoru odpovídá

$$\begin{aligned} P_{(k+1|k)} &= E[\Delta \mathbf{x}_{(k+1|k)} \Delta \mathbf{x}_{(k+1|k)}^T] \\ &\doteq \nabla f_x P_{(k|k)} \nabla f_x^T + \nabla f_\vartheta Q_{(k)} \nabla f_\vartheta^T \end{aligned} \quad (6.40)$$

Obdobným způsobem lze určit také vztah pro hodnotu vektoru odhadů výstupních hodnot a inovačního vektoru:

$$\tilde{\mathbf{y}}_{(k+1)} = h(\tilde{\mathbf{x}}_{(k+1|k)}, 0) \quad (6.41)$$

$$\begin{aligned} \Delta \mathbf{y}_{(k+1)} &= \mathbf{y}_{(k+1)} - \tilde{\mathbf{y}}_{(k+1)} \\ &\doteq \nabla h_x \Delta \mathbf{x}_{(k+1|k)} + \nabla h_w \mathbf{w}_{(k)} \end{aligned} \quad (6.42)$$

pro aktualizaci stavového vektoru a kovarianční matice (stejně jako v 6.34):

$$\begin{aligned} \tilde{\mathbf{x}}_{(k+1|k+1)} &= \tilde{\mathbf{x}}_{(k+1|k)} + L_{(k+1)} \Delta \mathbf{y}_{(k+1)} \\ P_{(k+1|k+1)} &= P_{(k+1|k)} - L_{(k+1)} \chi_{\Delta\Delta(k+1|k)} L_{(k+1)}^T \end{aligned}$$

a hodnoty příslušných kovariančních matic:

$$\chi_{\Delta\Delta(k+1|k)} \doteq \nabla h_x P_{(k+1|k)} \nabla h_x^T + \nabla h_w R_{(k+1)} \nabla h_w^T \quad (6.43)$$

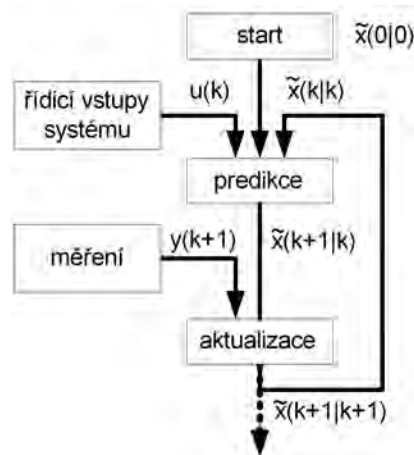
$$\begin{aligned} \chi_{x\Delta(k+1|k)} &\doteq P_{(k+1|k)} \nabla h_x^T \\ L_{(k+1)} &= P_{(k+1|k)} \nabla h_x^T \chi_{\Delta\Delta(k+1|k)} \\ &= \chi_{x\Delta(k+1|k)} \chi_{\Delta\Delta(k+1|k)} \end{aligned} \quad (6.44)$$

Tímto způsobem výpočtu kovarianční matice inovačního vektoru $\chi_{\Delta\Delta}$ a váhové matice L je zaručeno, že odhad stavového vektoru robotu $\tilde{\mathbf{x}}_R$ je neposunutý³. Odhad stavového vektoru $\tilde{\mathbf{x}}_{(i|j)}$ je považován za neposunutý, pokud platí, že matice: $P_{(i|j)} - E[\Delta \mathbf{x}_{(i|j)} \Delta \mathbf{x}_{(i|j)}^T]$ je semidefinitně pozitivní [8].

6.7 Závěr

V této kapitole byly popsány základní vlastnosti Kalmanova filtru a jeho rozšíření pro nelineární robotické systémy - EKF. V algoritmu SLAM je EKF využíván jako estimátor pro stanovení polohy mobilního robotu.

³V případě posunutého odhadu stavového vektoru hodnota $\tilde{\mathbf{x}}_R$ nekonverguje k \mathbf{x}_R .



Obrázek 6.2: Časová posloupnost kroků pro algoritmus KF (EKF)

Předpokládá se, že robot má k dispozici předem uložené body, podle nichž se lokalizuje. Tyto body nemusí být nutně uloženy v paměti robotu před započítím lokalizace. Může se jednat o body, které byly získány při předchozím měření. Tyto orientační body taktéž nemusí mít podobu geometrických vlastností prostředí, ale může se (v obecném případě) jednat o jakýkoliv druh charakteristiky popisující okolní prostředí (lokalizace pomocí pachové stopy). Vysvětlení časové posloupnosti jednotlivých kroků cyklu je uvedeno na obr. 6.2, platí jak pro KF, tak pro EKF [8].

Kapitola 7

Systémy pro tvorbu map a lokalizaci

V této kapitole jsou uvedeny příklady systémů, které jsou vyvíjeny a testovány na předních světových pracovištích. Z vyvíjených lokalizačních systémů byly vybrány systémy, které pro lokalizaci využívají 2D mapy.

7.1 Robotické mapování důlních systémů

Jedním z nejvýznamnějších vědeckých pracovišť zabývajícím se problematikou robotického mapování je laboratoř umělé inteligence (Artificial Intelligence Laboratory) na Stanfordské univerzitě (USA). Jako příklad výzkumu, kterým se tato laboratoř zabývá, byl zvolen robot Groundhog, který je určen pro robotické mapování vnitřních prostřední důlních systémů. Tento robot byl vyvinut ve spolupráci s robotickým institutem na univerzitě Carnegie Mellon (USA) [40].

Přestože výsledná mapa důlního systému je ve vytvářena ve 3D, pro samotnou lokalizaci mobilního robotu Groundhog bylo využito 2D mapy. Tato mapa byla vytvořena metodou porovnávání po sobě jdoucích snímků získaných pomocí laserového proximního snímače (*Scan Matching*) [38]. Robot pro svou lokalizaci využívá pouze informace z laserových snímačů, k dispozici nejsou data získaná odometrií. Na obrázku 7.1 je zobrazen robot Groundhog při mapování důlního systému.



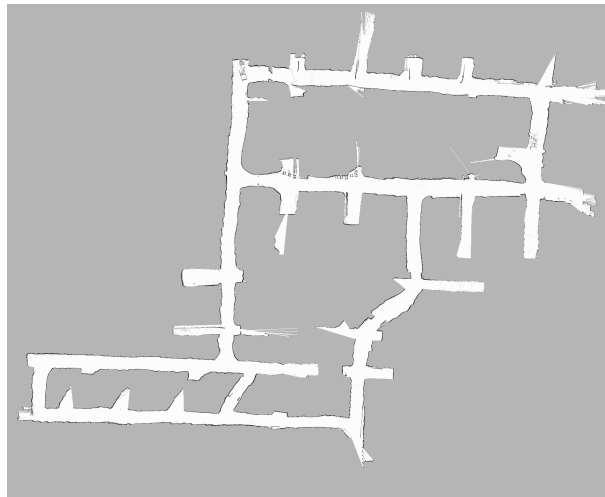
Obrázek 7.1: Robot Groudhog při mapování důlního systému [39]

Mobilní robot není autonomní, je řízen pomocí kabelu, který slouží zároveň jako pojistka v případě selhání pohonného systému robotu.

Vzájemné pootočení a posunutí bylo vyhodnocováno na základě algoritmu ICP (Iterative Closest Point).

$$E(R, t) = \sum_{i=1}^M \sum_{j=1}^D w_{i,j} \|m_i - (Rd_j + t)\|^2 \quad (7.1)$$

Tento iterativní algoritmus minimalizuje hodnotu odchylky mezi modelem (vzorem) $M = (m_1, \dots, m_i)$ a novým snímkem - naměřenými daty $D = (d_1, \dots, d_j)$.



Obrázek 7.2: 2D mapa důlního systému [39]

V jednotlivých krocích algoritmu jsou nastavovány hodnoty translační a rotační matice tak, aby vzájemná odchylka byla minimální - lokální mapa D je transformována tak, aby vyhovovala mapě globální M [41]. Hodnota váhového vektoru $w_{i,j}$ určuje,

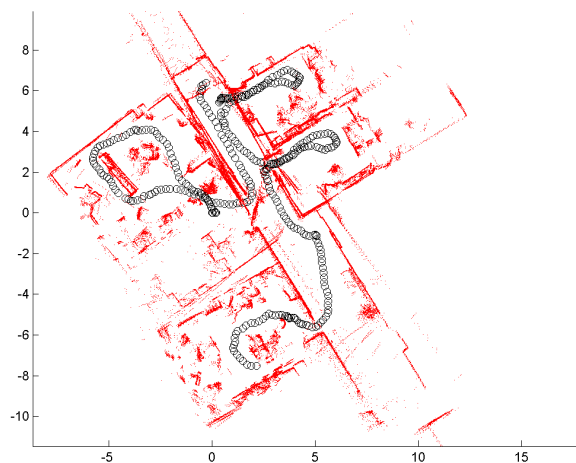
zda se jedná o body mapy, které spolu korespondují (hodnota 1) nebo nekorespondují (hodnota 0). Výsledek mapovacího algoritmu je zobrazen na obrázku 7.2. Na implementaci ICP algoritmu se podílela také skupina robotiky z německého Fraunhofer Institutu [42], která se zabývá 3D mapováním. Tato skupina se specializuje na vývoj metod pro minimalizaci chybové funkce (viz rovnice 7.1) metody ICP pro lokalizaci a mapování (převážně v 3D) [43], [44].

7.2 Projekt PeLoTe

Jako zástupce českých pracovišť zabývajících se lokalizací a mapováním v mobilní robotice lze jistě považovat Gerstnerovu laboratoř [45] při pražském ČVUT. Tato laboratoř se mimo jiné specializuje na spolupráci mezi člověkem a mobilním robotem. Příkladem může být projekt PeLoTe, který si klade za cíl usnadnit práci hasičským záchranným týmům.

Pomocí mobilního robotu je prozkoumávána nebezpečná oblast, v níž se mohou nacházet lidské oběti. Pokud mobilní robot nalezne oběť, předá informaci do řídicího centra, které tuto informaci zpracuje a zašle ji lidskému záchranáři, který je vybaven digitálním zařízením s digitální mapou prostředí. Záchranář na základě této informace může rychle a účinně podniknout kroky nutné pro záchranu.

Pro lokalizaci a vytváření robotických map je robot vybaven planárním laserovým snímačem SICK. Z dat získaných pomocí tohoto laserového snímače jsou na základě algoritmu ICP vytvářeny robotické mapy, v nichž je prováděna lokalizace. Obdobným systémem (PeNa - Personal Navigation) je vybaven také lidský záchranář, jehož poloha je určována z inerciálního lokalizačního systému vybaveného optickým gyroskopem (odometrie), systémem laserových snímačů pro určení délky kroku a planárním laserovým snímačem. Data z těchto jednotek jsou také využívána pro automatickou tvorbu digitální mapy a lokalizaci v této mapě. Příklad digitální mapy získané tímto systémem je uveden na obrázku 7.3.



Obrázek 7.3: Příklad mapy získané mapovacím systémem PeNa [46]

7.3 Závěr

Úlohou robotického mapování a lokalizace se zabývají téměř všechna pracoviště, která řeší problematiku autonomních robotických systémů. V prvním případě se jedná o pracoviště, na kterém v roce 1972 vznikl robot Shakey (viz kapitola 3) a lze tedy říci, že s autonomní mobilní robotikou má nejdelší zkušenosti. Druhé pracoviště bylo zvoleno jako český tvůrce robotických systémů pro vytváření map a lokalizaci pomocí mobilních robotů.

Kapitola 8

Model robotu a snímače

Popsané algoritmy pro lokalizaci mobilního robotu využívají odhadu polohy robotu a odhadu měření získaného ze snímačů, na robotu umístěných. Aby bylo možno provést tyto odhady, je třeba vytvořit matematický model robotického systému a použitého snímače. Algoritmus SLAM byl definován jako algoritmus, který stanovuje polohu robotu a orientačních bodů v prostoru pouze na základě měření, která jsou získávána snímači umístěnými na robotu. Poloha robotu \mathbf{x}_R v prostoru je popsána souřadnicemi v globálním kartézském systému.¹

$$\mathbf{x}_R = [x, y, \varphi]^T \quad (8.1)$$

Souřadnice robotu zároveň reprezentují stavové veličiny, které jsou využity pro popis robotu a jeho modelu.

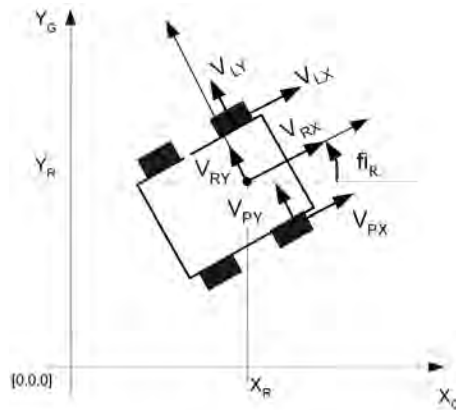
Prostředí, ve kterém se robot pohybuje, je popsáno množinou bodů \mathbf{m}_N , kde $N \in R$. Jednotlivé prvky množiny \mathbf{m}_i jsou popsány souřadnicemi x_i a y_i v globálním kartézském systému. Tyto souřadnice jsou zároveň stavovými veličinami popisujícími stav orientačního bodu. Předpokladem je, že stav orientačních bodů je v čase konstantní.

8.1 Model robotu

Pro úlohu mapování a lokalizace je využíván robot UTAR, který je vyvíjen na Ústavu automatizace a měřicí techniky od roku 1999. Jedná se o čtyřkolový mobilní robot, který je poháněn dvěma stejnosměrnými motory. Kola na každé straně robotu jsou spojena řetězem, robot nemá otočnou nápravu, proto při otáčení robotu dochází ke smyku. Hlavní výhodou této konstrukce mobilního podvozku je jeho jednoduchá mechanická konstrukce a velká robustnost, což umožňuje pohyb nejen ve vnitřních prostorách, ale i ve vnějším prostředí.

¹ x_R označuje vektor stavových veličin. Vzhledem k tomu, že stavové veličiny odpovídají souřadnicím v kart. souř. systému (x, y, φ) , bylo z důvodu větší přehlednosti zvoleno označení R . Jedná se však o proměnnou, která odpovídá proměnné x použité v kap. 6 pro popis stavového vektoru.

Pro účely mapovacího algoritmu byl vytvořen jednoduchý matematický model umožňující stanovení stavových veličin. Matematicky lze robot popsat na základě sil na něj působících, což vede ke vzniku složitějšího dynamického modelu. Zatímco určení polohy robotu pomocí dynamického modelu je dáno řešením soustavy diferenciálních rovnic, u modelu kinematického se jedná o relativně jednoduchý výpočet. Při sestavování kinematického modelu jsou pro stanovení průběhu rychlosti pohybu robotu použity pouze rychlosti jednotlivých kol. V takovém případě matematický model robotu nezohledňuje síly, které na něj při pohybu působí. V případě robotu UTAR se jedná o zvláště smykové síly, které působí na kola robotu během zatáčení. Jedním z předpokladů pro sestavení kinematického modelu robotu je, že kola robotu



Obrázek 8.1: Schéma mobilního robotu UTAR

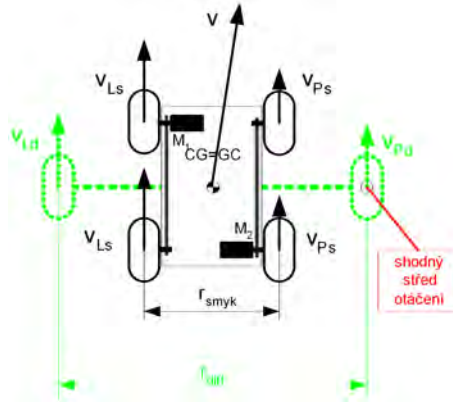
vykonávají čistě rotační pohyb kolem své osy a nedochází k jeho smýkání. Z tohoto předpokladu vyplývá skutečnost, že kinematický model smykem řízeného robotu nelze sestavit [70, 63].

8.1.1 Kinematický model robotu

Idea náhrady jednotlivých robotů je zobrazena na obrázku 8.2. Požadavkem je, aby se střed otáčení obou robotů nacházel ve stejném bodu. Aby bylo možno provést tuto záměnu obou typů robotických podvozků, je nutné, aby byly splněny následující podmínky:

- podklad, po kterém se robot pohybuje, je homogenní, jeho vlastnosti jsou pod všemi koly shodné (v reálném případě je nutné, aby podklad byl shodný alespoň přibližně),
- kola robotu mají shodné vlastnosti (materiál pneumatik, tlak v pneumatikách, míra opotřebení, apod.),
- těžiště smykem řízeného mobilního robotu leží v geometrickém středu robotu.

Pokud jsou splněny uvedené podmínky, lze smykem řízený mobilní robot popsat pomocí kinematického modelu pro robot řízený změnou rychlosti otáčení kol.



Obrázek 8.2: Náhrada robotu řízeného smykem robotem řízeným změnou rychlosti kol

Pro dopřednou a úhlovou rychlost robotu lze psát:

$$\Delta v(k) = \frac{v_L(k) + v_P(k)}{2} \quad (8.2)$$

$$\Delta \omega(k) = \frac{v_L(k) - v_P(k)}{r_{diff}} \quad (8.3)$$

$$r_{diff} = K r_{smyk} \quad (8.4)$$

$$\Delta v_L(k) = r_r \Delta \omega_L(k) \quad (8.5)$$

$$\Delta v_P(k) = r_r \Delta \omega_P(k)$$

jednotlivé stavové veličiny odpovídající poloze robotu lze určit podle:

$$\begin{aligned} \varphi(k+1) &= \varphi(k) + \Delta \omega(k) \\ x(k+1) &= x(k) + \Delta v(k) \sin(\varphi(k+1)) \\ y(k+1) &= y(k+1) + \Delta v(k+1) \cos(\varphi(k+1)) \end{aligned} \quad (8.6)$$

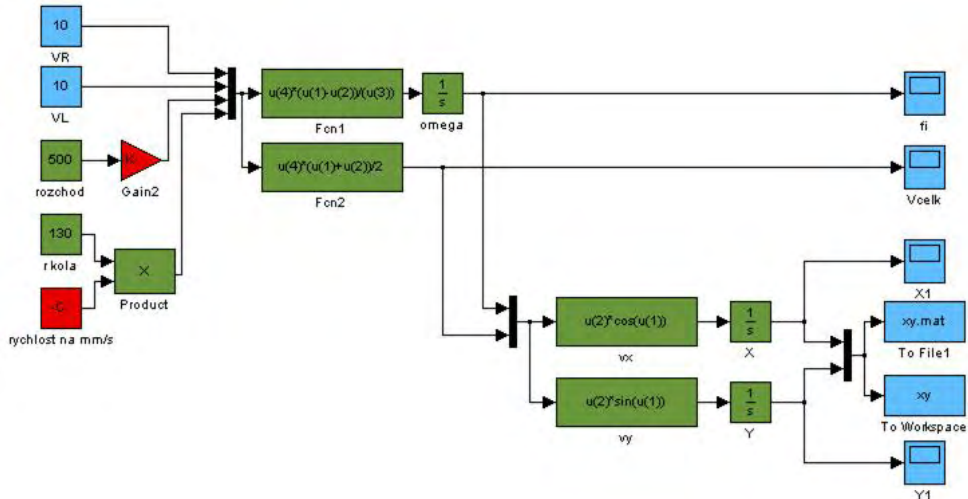
Tento model lze popsat nelineární přenosovou funkcí v maticovém tvaru jako:

$$\mathbf{x}_{R(k+1)} = f(\mathbf{x}_{R(k)}, \mathbf{u}(k), \vartheta(k)) = \begin{pmatrix} x(k|k) + \left(\frac{\Delta v_P(k) + \Delta v_L(k)}{2} + \vartheta_v \right) \cos \left(\varphi(k|k) + \frac{v_L(k) - v_P(k)}{r_{diff}} + \vartheta_\varphi \right) \\ y(k|k) + \left(\frac{\Delta v_P(k) + \Delta v_L(k)}{2} + \vartheta_v \right) \sin \left(\varphi(k|k) + \frac{v_L(k) - v_P(k)}{r_{diff}} + \vartheta_\varphi \right) \\ \varphi(k|k) + \vartheta_\varphi + \left(\frac{v_L(k) - v_P(k)}{r_{diff}} + \vartheta_v \right) \end{pmatrix} \quad (8.7)$$

kde $\vartheta(k)$ je šumový vektor se složkou ϑ_v odpovídající rychlosti robotu a složkou ϑ_φ , která odpovídá šumu vnášenému do systému odchylkami úhlové rychlosti. Předpokladem je, že šumové složky rychlostí jednotlivých kol ϑ_{vL} a ϑ_{vP} jsou shodné.

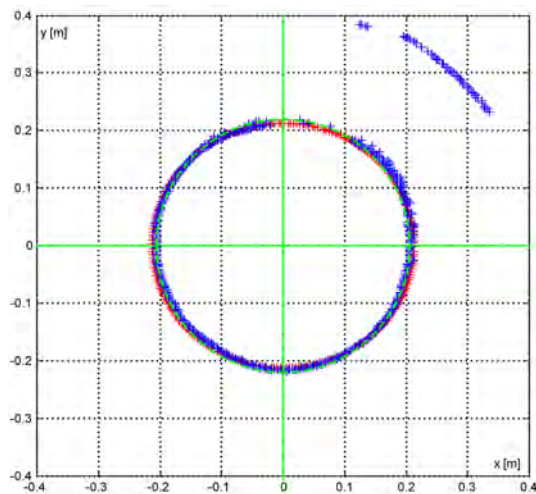
Platnost tohoto jednoduchého matematického modelu byla ověřena pomocí simulace a měření. Schéma modelu pro Matlab Simulink je zobrazeno na obrázku 8.3. Barevně jsou označeny jednotlivé části modelu, modrá reprezentuje vstupně/výstupní část, zelená barva reprezentuje samotný matematický model dvoukolového robotu. Červená

barva označuje konstantu nutnou pro přepoččet vstupních hodnot pro mikroprocesor regulátoru motorů a konstantu pro určení poměru mezi rozchody kol jednotlivých druhů podvozků 8.4.



Obrázek 8.3: Simulační schéma kinematického modelu pro Matlab Simulink

8.1.2 Ověření matematického modelu



Obrázek 8.4: Měření a simulace pohybu těžiště slykem řízeného robotu a jeho matematické náhrady

Cílem experimentu bylo stanovit, za jakých podmínek je možné slykem řízený mobilní robot (*skid steering*) popsat pomocí kinematického modelu dvojkolového robotu, řízeného změnou rychlosti jednotlivých kol (*differential drive*). Při ověřování matematického modelu bylo využito kamery, která byla zavěšená nad plochou, po níž se

pohyboval robot. Na robotu byly umístěny barevné štítky, které umožnily snadnou identifikaci bodů nacházejících se na robotu. Štítky byly umístěny v těžišti robotu a v osách kol. Výstupem měření byla nejen poloha těchto štítků, ale také hodnota odpovídající otočení kol robotu. Tato hodnota spolu s hodnotou řídicího vstupu byla vyčítána přímo z řídicí jednotky robotu UTAR.

Prvním krokem bylo zjištění koeficientu pro přepočet mezi velikostí řídicího signálu regulátoru motoru, hodnotou měřenou na výstupu optických enkodérů a reálnou vzdáleností ujetou robotem. V dalším kroku byl pro daný povrch (betonová podlaha laboratoře) určen koeficient K (rovnice 8.4). Výsledek měření v porovnání s modelem je zobrazen na obrázku 8.4. Modré body představují výsledek měření polohy kamerou. Červené body jsou výstupem ze simulace pomocí navrženého matematického modelu. Zelená čára představuje ideální trajektorii robotu v případě, že by nedocházelo k prokluzům pneumatik.

Vzhledem k tomu, že měřicí aparatura nebyla primárně určena pro stanovení modelu robotu (jedná se o uspořádání pro robotický fotbal), jsou v pravém horním rohu obrázku 8.4 patrné body, které leží mimo předpokládanou kružnici. Tato skutečnost je způsobena nestejnými optickými vlastnostmi podlahy, po které se robot pohyboval. V těchto místech vyhodnocovací algoritmus provedl špatnou interpretaci barev štítků umístěných na robotu, z čehož vyplynulo špatné určení polohy těžiště robotu.

Shrnutí

V předchozí části kapitoly je uveden návrh matematického modelu mobilního robotu. Jedná se o model, který je založen na velikostech rychlostí, které jsou vyvíjeny jednotlivými koly mobilního robotu. Vzhledem k mechanické konstrukci robotu (robot řízený smykem) dochází při zatáčení robotu ke smýkání kol. Tyto smykové síly jsou však v matematickém modelu zanedbány. Pro ověření správnosti navrženého matematického modelu bylo provedeno měření pomocí kamerového systému. Poloha těžiště mobilního robotu odpovídá poloze těžiště získané simulací matematického modelu (viz obrázek 8.4). Na základě těchto experimentů lze říci, že pro dané prostředí lze pro výpočet polohy těžiště robotu řízeného smykem použít navržený kinematický model.

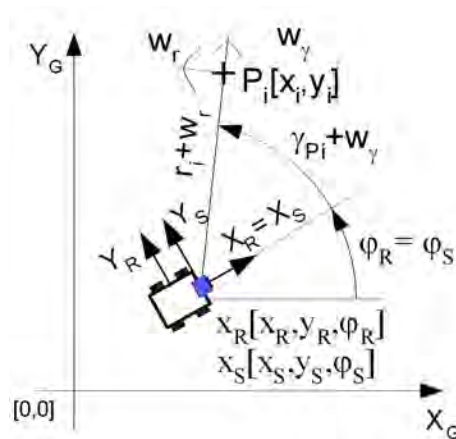
8.2 Proximitní snímač

V současné době se v mobilní robotice rozšiřuje využívání laserových proximitních senzorů. Tyto senzory se vlivem vyvíjející se technologie stávají přístupnější pro robotická pracoviště a svými vlastnostmi doplňují a rozšiřují možnosti stávajících sensorických systémů robotu. Jedná se většinou o senzory, které nejsou primárně určené pro mobilní robotiku, ale pro využití v průmyslových aplikacích, jako jsou zabezpečení nebezpečných provozů, kontrola výroby apod. Mezi významné výrobce této skupiny patří například firma Keyence, Minolta a Sick. Tyto snímače určují vzdálenost na základě známé rychlosti šíření paprsku a doby letu. Proto je skupina těchto snímačů nazývána (*TOF - Time Of Flight*).

S rozvojem mobilní robotiky se objevují také firmy, které vyvíjejí a vyrábějí laserové snímače přizpůsobené potřebám mobilní robotiky. Jedná se zvláště o přizpůsobení rozměrů a hmotnosti snímače robotickým platformám. Cílem je vyrábět co nejmenší a nejlehčí snímače tak, aby šly využít nejen u robotů pozemních, ale například i u robotů létajících. Mezi nejvyužívanější principy měření u těchto snímačů patří aktivní triangulační metoda a TOF. Vývojem a výrobou těchto snímačů se mimo jiné zabývají firmy Hokuyo a Mesa Imaging AG (3D proximní kamera). Hmotnost těchto snímačů je většinou snížena na úkor robustnosti a odolnosti proti nepříznivým vlivům prostředí, a proto nejsou určeny pro použití ve vnějším prostředí. Přesto však jsou tyto snímače využívány pro návrhy a testování navigačních algoritmů pro vnější prostředí.

8.3 Matematický model snímače

Pro snadnou dostupnost byl jako hlavní snímač pro získávání orientačních bodů zvolen laserový proximní snímač *SICK LMS 200*. Tento snímač patří do skupiny snímačů TOF (vlastní měření probíhá pomocí určování fázového rozdílu signálů). Snímač měří vzdálenost překážek, které se nacházejí v kruhové výseči před snímačem. Nastavením snímače lze definovat úhel kruhové výseče (max. 180°), maximální vzdálenost detekovatelné překážky je 50 m. Výstupem ze snímače je sada hodnot odpovídajících měření v jednotlivých krocích. Velikost kroku lze nastavit na $0,25^\circ$, $0,5^\circ$ a 1° [47]. Výsledkem



Obrázek 8.5: Matematický model snímače

měření je množina vzdáleností k překážce $s = (s_1, s_2, \dots, s_j)$.

Hodnota koeficientu j závisí na nastavení parametrů snímače a odpovídá počtu kroků, na které je rozdělen měřený interval kruhové výseče. Každý prvek množiny s je definován souřadnicemi $s_i = (r_i, \varphi_i)$ v polárním souřadném systému s počátkem ve středu snímacího prvku proximního snímače. Pro přepočítání do kartézského souřad-

ného systému s počátkem ve středu snímacího prvku platí:

$$x_j = r_j \cos \varphi_j \quad (8.8)$$

$$y_j = r_j \sin \varphi_j \quad (8.9)$$

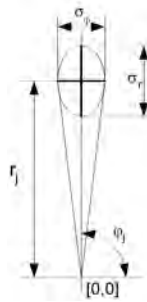
Výstup ze snímače lze zpracovat i v polárním souřadném systému. Transformace souřadných systémů je prováděna z důvodu, že pohyb robotu a globální mapa prostředí je definován v kartézském souřadném systému.

V rovnicích Kalmanova estimátoru stavů se vyskytuje šum měření \mathbf{w} , který má v případě uvedeného laserového snímače dvě složky w_r a w_φ . Hodnota těchto prvků šumového vektoru a jejich kovarianční matice R výrazně ovlivňuje kvalitu algoritmu estimátoru.

$$R = \begin{pmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\varphi^2 \end{pmatrix} \quad (8.10)$$

V technických specifikacích snímače *SICK LMS 200* jsou uvedeny parametry popisující chování snímače v různých podmínkách (odrazivost, závislost na teplotě, ...). Parametry, ze kterých by bylo možno určit hodnoty vektoru w , zde však uvedeny nejsou. Základní parametr pro optické měření vzdálenosti - závislost chyby měření na odrazivosti - je pro tyto materiály v katalogovém listu snímače uveden. Cílem měření bylo zjistit, zda je přesnost měřené vzdálenosti závislá na barvě materiálu při jeho konstantním složení [49].

8.3.1 Uspořádání experimentu

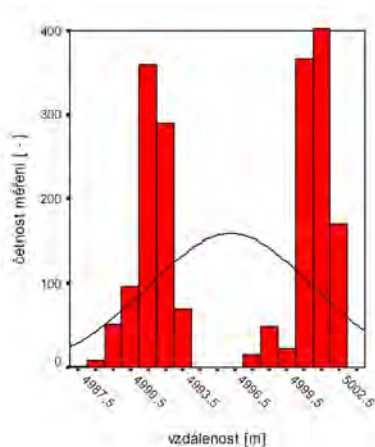


Obrázek 8.6: Prvky matice R

Experiment byl prováděn v tmavé místnosti tak, aby odražený paprsek nebyl ovlivňován okolním světlem. Laserový snímač byl nastaven tak, aby bylo vyhodnocováno měření pouze jedním paprskem s_{90} .

Pro stanovené vzdálenosti byla provedena sada měření pro jednotlivé stupně šedi a barvy odrazové plochy. Měřené vzdálenosti byly stanoveny na základě předpokladu, že důležitost měřených dat (vzdálenost překážky) klesá s rostoucí vzdáleností překážky. Vzdálenosti byly stanoveny na: 1, 2, 3, 4, 5, 7, 9, 11, 13, 15m. Maximální hranice byla stanovena s ohledem na předpoklad, že se robot pohybuje ve vnitřním prostředí a měřené vzdálenosti jsou omezeny velikostí zkoumaných prostor. Každá sada měření

pro jednotlivé barvy materiálu obsahovala 1900 hodnot, které byly podrobeny statistické analýze. V této kapitole jsou uvedeny pouze vybrané sady měření, kompletní přehled naměřených hodnot je uveden v příloze. Na grafu 8.7 je zobrazeno rozložení naměřených hodnot pro 18% střední odstín šedé barvy.²



Obrázek 8.7: Histogram měření vzdálenosti pro 18% středně šedou plochu při vzdálenost 5m

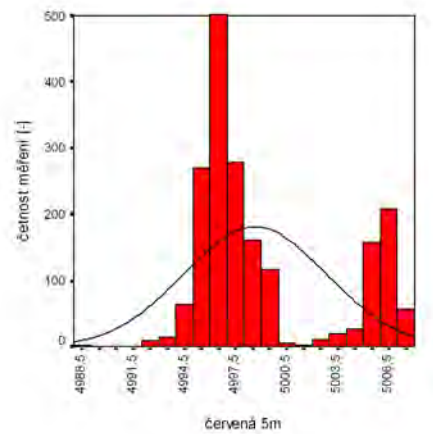
Z grafu 8.7 je patrné, že přestože rozptyl měření neodpovídá Gaussovu rozložení, lze jej aproximovat pomocí rozložení s parametry, které jsou vypsány v grafu. Odrazné plochy byly vyrobeny z matného papíru a odstíny šedi byly zvoleny tak, aby obsahly pokrytí bílého papíru černým inkoustem od 0 do 100%. Odrazivost odstínu č. 3 (obrázek 8.8) odpovídá střední šedé barvě. Tato hodnota je definována v RGB prostoru jako [127, 127, 127] [R, G, B].



Obrázek 8.8: Testované barvy a odstíny šedé

Jako zástupce barevné škály bylo vybráno měření pro červenou barvu odrazné plochy. Vzhledem k tomu, že vlnová délka měřicího laserového paprsku spadá do infračervené oblasti (824 nm), je u červené odrazné plochy pro měření vzdálenosti nejmenší rozptyl.

²18% střední šedá - odstín šedé barvy, který odráží 18% dopadajícího světla, ve fotografii je používán jako standard pro správné nastavení expozice [48].



Obrázek 8.9: Histogram měření vzdálenosti pro plochu červené barvy, při vzdálenosti 5m

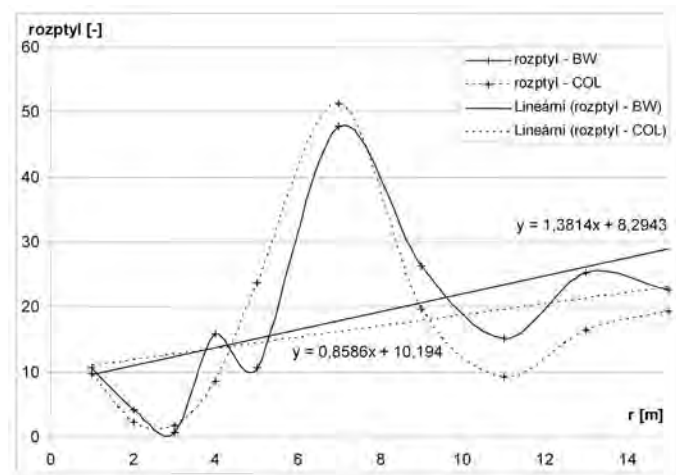
Závislost rozptylu na měřené vzdálenosti lze určit z tabulky 8.1. V této tabulce jsou uvedeny hodnoty rozptylu pro jednotlivé barvy odrazné plochy. V posledním sloupci je uvedena průměrná hodnota rozptylu pro danou vzdálenost.

Tabulka 8.1: Závislost rozptylu měření na vzdálenosti
rozptyl měřené vzdálenosti

r [m]	bílá	šedá3	šedá2	šedá1	černá	BW - prům.
1	13,62	23,01	5,28	1,57	9,63	10,62
2	5,88	11,30	2,59	0,99	0,49	4,25
3	0,37	0,71	0,71	0,79	0,99	0,71
4	3,06	12,62	12,85	25,87	25,28	15,94
5	2,19	6,72	22,75	1,14	20,69	10,70
7	80,41	60,05	10,36	80,56	7,97	47,87
9	24,65	35,27	35,53	25,43	10,82	26,34
11	8,85	13,56	14,14	16,69	22,26	15,16
13	27,87	37,98	24,40	23,25	22,94	25,29
15	21,84	21,48	25,52	24,80	20,16	22,76
	18,87	21,27	15,41	20,14	14,12	
r [m]	červená	žlutá	zelená	modrá	fialová	CL - prům.
1	1,54	3,96	17,76	15,66	9,56	9,70
2	2,44	3,19	1,44	1,75	2,51	2,27
3	1,53	2,31	2,98	0,91	1,24	1,79
4	5,42	16,61	10,03	6,96	3,98	8,60
5	17,40	76,02	3,24	2,84	18,86	23,67
7	51,87	12,87	65,06	63,27	62,98	51,21
9	23,51	5,30	9,98	26,97	32,87	19,73
11	5,76	13,13	6,05	11,44	10,03	9,28
13	14,41	15,17	12,78	21,71	20,00	16,41
15	15,62	20,07	17,59	22,18	21,49	19,39
	13,95	16,67	14,69	17,37	18,35	

Závislost rozptylu měření na vzdálenosti je graficky znázorněna v 8.10. Rovnice závislosti rozptylu na vzdálenosti je: $R_r = 1,12r + 9,24$ ³.

³Výsledná směrnice přímky a hodnota rozptylu pro nulovou vzdálenost byly určeny jako průměr rozptylů všech měření pro jednotlivé vzdálenosti.



Obrázek 8.10: Závislost rozptylu měření na vzdálenosti

Druhým parametrem kovarianční matice R je prvek R_φ . Měření pro tuto hodnotu parametru nebylo provedeno. Hodnota tohoto prvku byla odhadnuta z technické dokumentace poskytované výrobcem snímače [47]. Pro dané rozlišení měření $0,25^\circ$ byla stanovena na 0.1.

8.4 Závěr

V této kapitole jsou popsány dva základní prvky SLAM systému. Matematický model robotu a planárního laserového snímače.

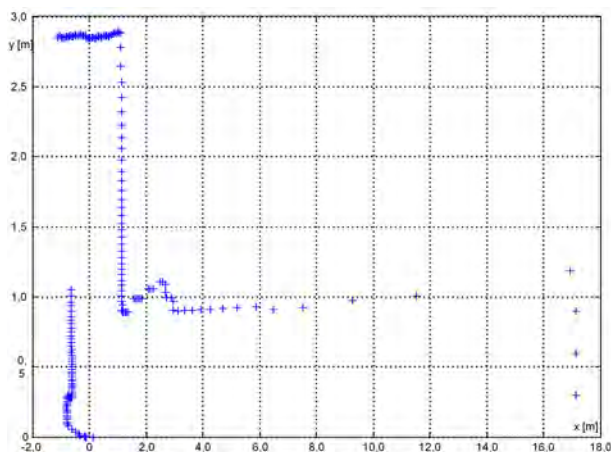
Kinematický model robotu usnadňuje odhad polohy mobilního robotu. Přestože při zatáčení robotu UTAR dochází k smýkání jednotlivých kol, za určitých podmínek je možná náhrada dynamického modelu, v němž jsou řešeny soustavy diferenciálních rovnic, modelem kinematickým, kde je řešení určeno pomocí soustavy jednoduchých trigonometrických funkcí. Pomocí kamerového měřicího systému bylo provedeno ověření kinematického modelu. Tento praktický test také dává informaci o chování mobilního robotu a usnadňuje nastavení hodnoty rozptylové matice Q .

Také experimenty s laserovým snímačem byly provedeny za účelem odhadu rozptylů šumové matice měření R a ověření závislosti rozptylu měření vzdálenosti na barvě plochy.

Kapitola 9

Detekce význačných znaků zkoumané oblasti

Výstupem z proximitního snímače vzdálenosti je sada měření, která reprezentuje vzdálenost překážek, které se nacházejí v polovině před snímačem. Každý prvek měření reprezentuje vzdálenost bodu v prostoru, který je popsán souřadnicemi v polárním souřadnicovém systému. Jednotlivé sady měření reprezentují tzv. *lokální mapy* viz obr. 9.1. Lokální mapa představuje vjem robotu v okamžiku měření bez vztahu k počátku globálního souřadnicového systému. Počátek souřadnicového systému lokální mapy je umístěn v ose rotace zrcátka, které rozmítá měřicí laserový paprsek. Na lokální mapu lze pohlížet také jako na binární obraz $O = (B_1, B_2 \dots B_n)$, jehož každý bod může nabývat hodnoty 1 nebo 0. Buňka binárního obrazu nabývá hodnoty 1 v místě, které odpovídá překážce. V ostatních místech nabývá binární obraz hodnoty 0.



Obrázek 9.1: Lokální mapa po převedení do kartézského souř. systému

Cílem algoritmu SLAM je nalezení *globální mapy* $M_G = (m_1 \dots m_n)$ prostředí, která je vztažena ke globálnímu souřadnicovému systému. V kapitole 6 jsou uve-

deny principy základních estimátorů polohy robotu. Měřená data, která jsou zatížena náhodným šumovým signálem, jsou v těchto algoritmech porovnávána s daty, které jsou získány odhadem z modelu snímače a modelu robotu. Aby odhad polohy robotu (umožňuje vztahování lokální mapy ke globálnímu souřadnicovému systému) byl co nejpřesnější, je třeba zajistit, aby byla porovnávána data, která spolu vzájemně korespondují. Jednou z možností, jak dosáhnout tohoto výsledku, je zpracování lokální mapy tak, aby byly nalezeny *význačné rysy* (features) prostředí $l = (l_1 \dots l_n)$.

- význačný rys prostředí - uspořádání částí naměřených hodnot, které charakterizují zkoumané prostředí. Znakem vhodného rysu je skutečnost, že je snadno matematicky popsatelný a je detekovatelný v posloupnosti po sobě jdoucích měřeních [70].

V důsledku skutečnosti, že lokální mapa prostředí odpovídá binárnímu obrazu, lze pro nalezení význačných rysů oblasti použít algoritmy využívané v oblasti počítačového vidění. Mezi význačné rysy, které lze využít jako orientační body při lokalizaci mobilního robotu patří: [51, 52, 53]:

- geometrická primitiva - přímky, oblouky, části kružnic. Tyto geometrické útvary popisují překážky, které jsou tvořeny rovnými nebo zakřivenými souvislými plochami, odpovídajícími například stěnám místnosti. Hlavním zdrojem chyb při hledání geometrických prvků je šum, kterým jsou zatížena výstupní data ze snímače¹,
- rohy místností - tato místa v obrazové funkci jsou vhodná jako orientační body při lokalizaci robotu z toho důvodu, že jsou většinou detekovatelná v sekvenci několika po sobě jdoucích obrazů. V obrazové funkci jsou definována jako místa, v nichž je velké zakřivení hranic zkoumaného prostoru. Detekci rohů místností lze převést na hledání průsečíků křivek. Pro hledání průsečíků v obraze získaném laserovým proximitním snímačem nejsou vhodné metody, které jsou založeny na hledání rohu pomocí porovnávání obrazu a předem dané šablony. Důvodem je klesající závislost hustoty bodů na vzdálenosti,
- statistický popis obrazu - momentové charakteristiky obrazové funkce, těžiště obrazu, osy symetrie apod. Tyto charakteristiky umožňují získat ucelenější popis prostředí. Pomocí momentových charakteristik je možné charakterizovat velikost prostředí, jeho složitost (počet přímek) apod.

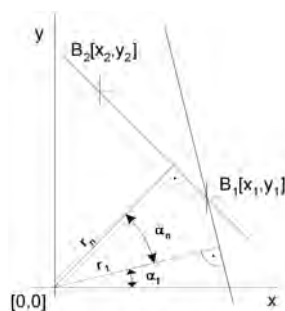
Pro metodu SLAM byl navržen algoritmus pro hledání částí přímek, které odpovídají stěnám. Tento algoritmus byl zvolen na základě skutečnosti, že většinu vnitřních prostředí (haly, chodby, skladiště, apod.) lze charakterizovat pomocí částí přímek, které jsou zároveň jednoduše matematicky popsatelné.

Jako orientační body pro lokalizační algoritmus byly zvoleny rohy, které byly určeny jako průsečíky nalezených přímek. V mobilní robotice jsou pro hledání přímek

¹Tato skutečnost má za následek, že data reprezentující přímku neleží na ideální přímce, z čehož vyplývá nutnost nalezení vhodné aproximace.

(úseček) v obraze získaném pomocí laserového snímače nejvyužívanější následující postupy [53]:

- *rozdělování a slučování* - (split and merge). Přímka je postupně skládána z jednotlivých bodů obrazu. Po přidání každého nového bodu k přímce je určena vzdálenost všech bodů od přímky. V případě, že se některý bod nachází ve větší vzdálenosti, než je předem stanovená hodnota, je poslední přidaný bod z přímky odstraněn. Tento bod tvoří počátek nové hledané přímky. Na konci algoritmu jsou všechny segmenty ležící na jedné přímce sloučeny dohromady,
- *regresní algoritmus* - (line regression algorithm). Tento algoritmus je principem obdobný jako algoritmus Houghovy transformace. Problém hledání přímek v obraze je převeden na hledání objektů v prostoru parametrů přímek pomocí hierarchického shlukování. Prostor parametrů je získán volbou vhodného modelu pro popis přímek,
- *inkrementální algoritmus* - (incremental algorithm, line tracking). K přímce, která je na počátku algoritmu definována prvními dvěma body jsou postupně přidávány další body. Pro takto zvolenou množinu bodů je metodou nejmenších čtverců vypočtena rovnice přímky. Pokud je součet vzdáleností bodů od takto určené přímky menší než zvolená hranice, je k přímce přiřazen další bod. V opačném případě je nový bod odebrán a vytvořena nová přímka.
- *Ransac* - (Random Sample Consensus). Dvěma náhodnými body obrazu je proložena přímka. Z ostatních obrazových bodů jsou na základě vzdálenosti vybrány body, které mohou příslušet přímce (nacházejí se v dostatečné blízkosti přímky). V případě, že je takto nalezených bodů méně než předem určený počet, jsou vybrány nové body. Tento algoritmus lze použít pro jakýkoliv geometrický útvar popisující prostředí, pokud je znám jeho matematický popis.
- *Radonova a Houghova transformace*. Hledání parametrů přímek je převedeno na hledání lokálních maxim v prostoru parametrů přímek. Prostor parametrů přímek je vytvořen na základě normálové rovnice přímek a je popsán parametry $\mathcal{H}(r, \alpha)$ viz obr. 9.2. Tento popis přímky tvoří tzv. model, podle něhož jsou určovány hodnoty prvků akumulátoru. Obecná verze obou metod je vhodná nejen k hledání přímek v obraze, ale umožňuje hledání i jiných geometrických primitiv v závislosti na volbě modelu křivky.
Radonova transformace - pro každý prvek z prostoru parametrů $\mathcal{H}(r, \alpha)$ je hledán počet bodů binárního obrazu O splňujících zvolený model.
Houghova transformace - pro každý bod z binárního obrazu O jsou hledány prvky z prostoru parametrů $\mathcal{H}(r, \alpha)$, které vyhovují zvolenému modelu.
Mezi hlavní výhody Houghovy a Radonovy transformace patří skutečnost, že nejsou citlivé na vzdálenost mezi body obrazu. Další výhodou je, že chybějící body v obrazové funkci O nevedou ke špatné interpretaci nalezených přímek.
Oproti předem uvedeným metodám je však tento algoritmus výpočetně náročný



Obrázek 9.2: Normálová reprezentace přímky

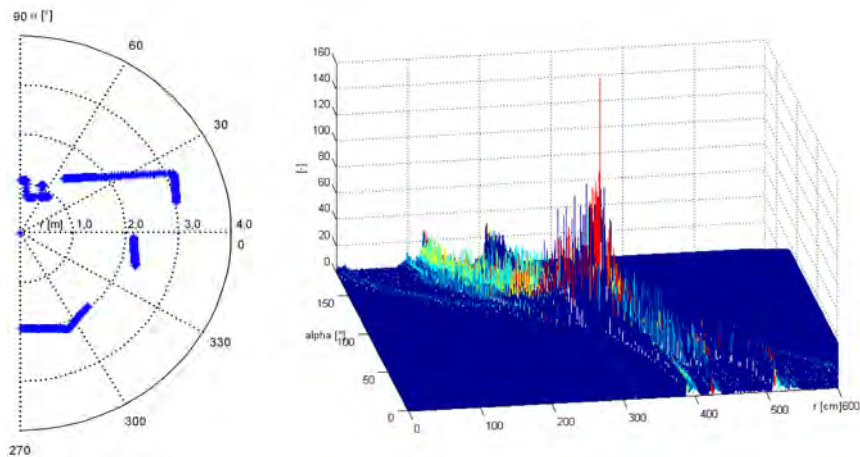
a jeho výsledkem jsou rovnice přímek, nikoliv úsečky, které by přímo odpovídaly zdem ohraničujícím prostor. Je tedy nutné metodu doplnit o algoritmus pro nalezení počátků a konců úseček.

Vzhledem k tomu, že s rostoucí vzdáleností měřených objektů roste vzdálenost mezi jednotlivými body obrazu a díky necitlivosti na chybějící body, byla jako metoda pro získávání geometrického popisu prostředí zvolena metoda Radonovy a Houghovy transformace. Byla navržena modifikace algoritmu, která umožňuje zpracování binárního obrazu z laserového snímače.

9.1 Houghova transformace

Každým bodem binárního obrazu $O = (\mathbf{b}_1, \mathbf{b}_2 \dots \mathbf{b}_n)$ se souřadnicemi $\mathbf{b}_n = (x_n, y_n)$ v kartézském souřadném systému je proložena přímka, která je reprezentována v normálovém tvaru: $r_i = x_n \cos \alpha_j + y_n \sin \alpha_j$. Pro každou přímku je určen počet bodů, které na této přímce leží. Takto určená hodnota je uložena do tzv. Houghova akumulátoru $\mathcal{H}(\alpha, r)$, což je pole hodnot, jehož jednotlivé prvky mají souřadnice (r_i, α_i) . Velikost úhlu natočení normály α_i je v rozsahu $\alpha \in (0, \pi)$.

Lokální maxima v akumulátoru $\mathcal{H}(\alpha, r)$ odpovídají parametrům přímek, na kterých leží největší počet bodů [54, 55, 56, 57].

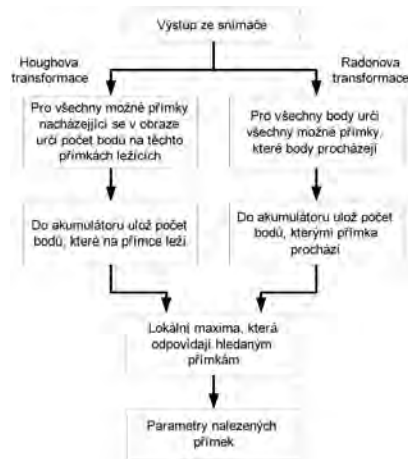


Obrázek 9.3: Binární obraz z proximitního snímače a obraz Houghova akumulátoru

Na obrázku 9.3 je zobrazen výstup ze snímače Hokuyo URG-04LX [37] a Houghův akumulátor pro přímky obsažené v tomto obraze.

9.2 Radonova transformace

Radonova transformace je obdobou předešlého algoritmu [59]. Porovnání obou metod je uvedeno na obrázku 9.4.



Obrázek 9.4: Porovnání algoritmu Houghovy a Radonovy transformace

Pomocí této transformace, která je nejčastěji využívána ke zpracování obrazu v medicínské technice, je možno z binárního obrazu O získaného pomocí laserového proximitního snímače určit skupiny bodů, které náležejí přímkám. Radonova transformace je definována jako:

$$Rf(r, \alpha) = \int \int O(x, y) \delta(r - x \cos(\alpha) + y \sin(\alpha)) dx dy \quad (9.1)$$

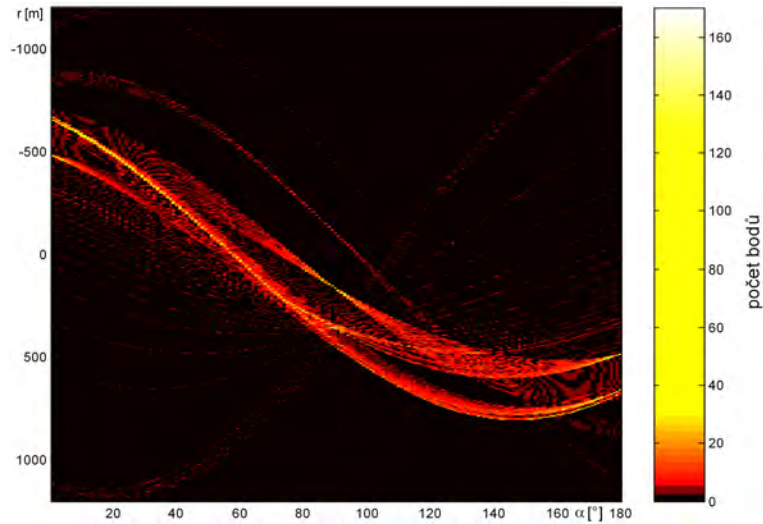
Obrazová funkce $O(x, y)$, kde: $(x, y) \in R^2$, je touto rovnicí transformována na funkci $Rf(r, \alpha)$, kde $r \in R$ a $\alpha \in (0, \pi)$. Rovnice 9.1 je vyjádřením spojité Radonovy transformace. Množina bodů naměřených laserovým snímačem je však diskrétní. Pro použitý laserový snímač SICK LMS 200 má krok mezi jednotlivými měřeními (v polárních souřadnicích snímače) velikost $0,25^\circ, 0,5^\circ, 1^\circ$ [47]. Proto bylo využito diskrétní Radonovy transformace. Tento algoritmus je implementován v programu Matlab 6.5 (verze 13) ve funkci `radon.m`. V následujícím textu je uveden jeho princip [60, 50].

V každém kroku je binární obraz pootočen kolem svého středu a je určen počet bodů v každém řádku obrazu. Toto číslo je uloženo do akumulátoru $\mathcal{H}(\alpha, r)$, který má stejnou strukturu jako akumulátor u Houghovy transformace. Pootočení obrazu je prováděno v rozmezí $\alpha \in (0, \pi)$. Velikost pootočení k_α obrazu je nastavitelná, což umožňuje ovlivnit přesnost hledaného parametru α_i a zároveň ovlivňuje rychlost algoritmu.

Při otáčení obrazu O jsou souřadnice každého bodu ($\mathbf{b}_1, \mathbf{b}_2 \dots \mathbf{b}_n$) přepočítány pomocí rotační matice 9.2.

$$\begin{pmatrix} x_n' \\ y_n' \end{pmatrix} = \begin{pmatrix} \cos k_\alpha & -\sin k_\alpha \\ \sin k_\alpha & \cos k_\alpha \end{pmatrix} \begin{pmatrix} x_n \\ y_n \end{pmatrix} \quad (9.2)$$

Po provedení rotace je určen počet bodů v každém řádku obrazu. Toto číslo je uloženo do akumulátoru $\mathcal{H}(\alpha, r)$. Parametry normálových rovnic hledaných přímk jsou v tomto akumulátoru opět reprezentovány jako lokální maxima [58]. Obrázek 9.5

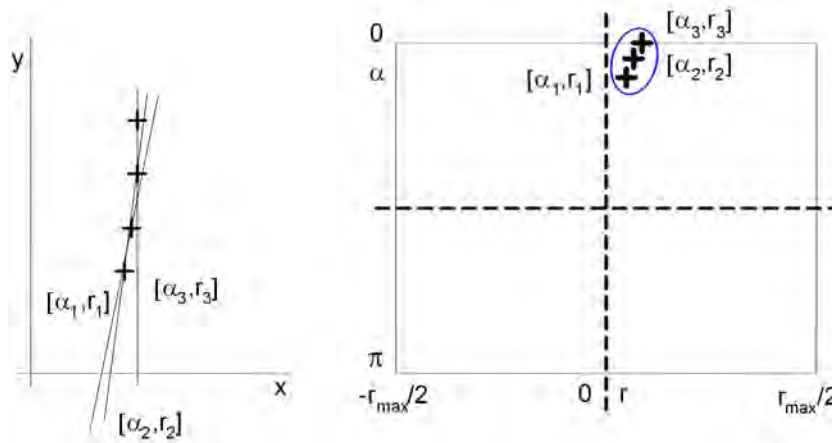


Obrázek 9.5: Akumulátor parametrů přímk pro Radonovu transformaci měření 9.1

zobrazuje Houghův akumulátor pro měření, které je zobrazeno na obrázku 9.1.

9.3 Určení parametrů hledaných přímek

Vzhledem k tomu, že v reálném případě neleží všechny body na jedné přímce, je třeba hodnoty v akumulátoru $\mathcal{H}(\alpha, r)$ zpracovat takovým způsobem, aby nalezené přímky byly nejvhodnější reprezentací pro změřené body. Na obrázku 9.6 je zobrazen příklad binárního obrazu se zobrazenými přímkami a výsledným Houghovým akumulátorem. Z tohoto obrázku je patrné, že všechny body v Houghově prostoru, které popisují výstup Houghovy (Radonovy) transformace, reprezentují jednu přímku, která se nachází v obraze. Z tohoto příkladu plyne nutnost návrhu algoritmu pro určení přímky, která nejlépe reprezentuje část obrazové funkce.



Obrázek 9.6: Příklad binárního obrazu a jeho akumulátoru $\mathcal{H}(\alpha, r)$

Při robotickém průzkumu většinou není předem známá struktura prostředí, což znamená, že dopředu nelze určit kolik a jak dlouhé jsou přímky, které prostředí definují. Z tohoto důvodu není vhodné použít jednoduchou metodu, při níž by se určila prahová hodnota pro hledanou přímku. V takovém případě by byly ze sady nalezených přímek odstraněny přímky odpovídající malému počtu bodů. Pro hledání přímek byl proto navržen algoritmus na základě shlukové analýzy parametrů přímek v Houghově akumulátoru $\mathcal{H}(\alpha, r)$.

9.3.1 Shlukování nalezených přímek

Základním principem shlukové analýzy je rozdělení objektů (nalezené přímky) do shluků v závislosti na jejich podobnosti, resp. odlišnosti. Objekt je reprezentován svými parametry: $\mathbf{o} = (p_1, p_2)$ kde $p_1 \approx \alpha$ a $p_2 \approx r$. Pro všechny objekty $(\mathbf{o}_1, \mathbf{o}_2, \dots, \mathbf{o}_n)$ je určena jejich vzájemná vzdálenost d . Na základě této vzdálenosti a nastavené hodnoty prahové T pro rozdělení objektů do jednotlivých shluků jsou potom parametry přímek rozděleny do skupin, ze kterých je vybrána přímka s největší četností bodů.

9.3.2 Určení vzdálenosti mezi jednotlivými objekty

Pro výpočet vzájemných vzdáleností objektů jsou vhodné tyto metody určení vzdálenosti mezi objekty:

Mahalanobisova vzdálenost - tento způsob určení vzdálenosti mezi dvěma objekty je vhodný zejména z toho důvodu, že pomocí váhové matice A v rovnici 9.3 lze nastavit důležitost koeficientů pro přiřazení do shluku. Vzdálenost mezi dvěma objekty je určena podle:

$$d_M(o_k, o_l) = \sqrt{(o_k - o_l)^T A S^{-1} A (o_k - o_l)} \quad (9.3)$$

kde matice A je symetrická matice váhových koeficientů pro jednotlivé parametry a S je matice vzájemných kovariancí mezi jednotlivými parametry.

Euklidovská vzdálenost - v případě, že váhová a rozptylová matice jsou matice jednotkové, Mahalanobisova vzdálenost přechází ve výpočetně jednodušší euklidovskou vzdálenost. Tento způsob však neumožňuje váhování koeficientů. Pro výpočet euklidovské vzdálenosti platí:

$$d_E(o_k, o_l) = \sqrt{\sum_{i=1}^n (p_{ik} - p_{il})^2} \quad (9.4)$$

Vzhledem k tomu, že Houghův akumulátor je dvourozměrné pole, $n = 2$. V tomto dvourozměrném případě lze nastavení váhy důležitosti jednotlivých parametrů provést změnou měřítka jedné z os v $\mathcal{H}(\alpha, r)$. Tohoto způsobu nastavení citlivosti bylo využito v navrženém a otestovaném algoritmu.

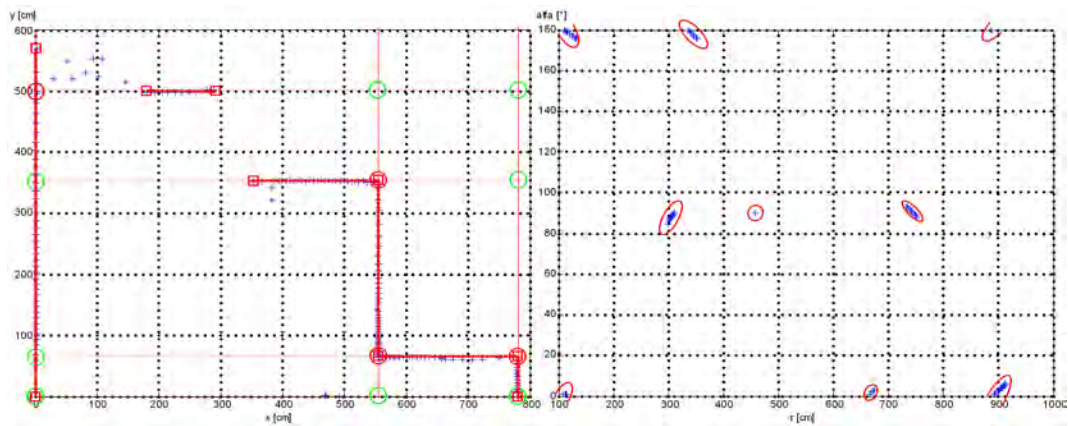
9.3.3 Rozdělení objektů do shluků



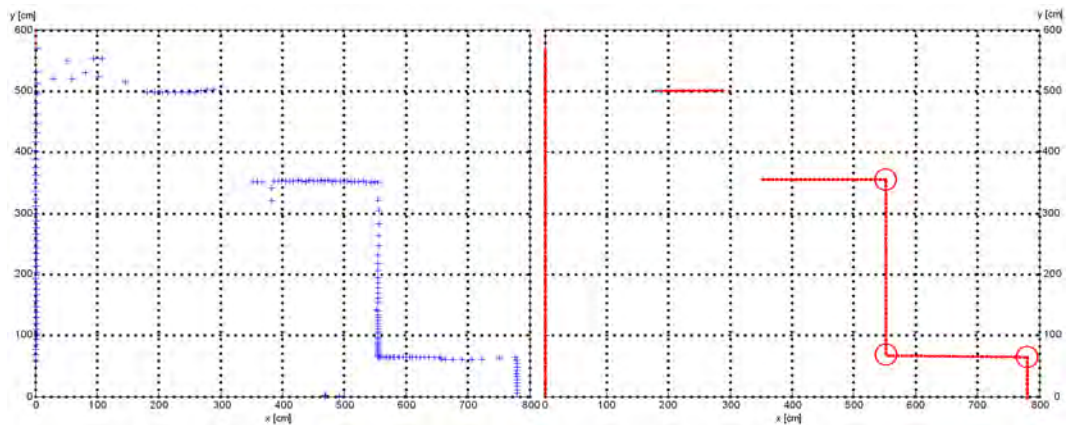
Obrázek 9.7: Algoritmus shlukování

Postup při vytváření shluků, z nichž každý reprezentuje jednu z přímek v binárním obraze, je uveden na obrázku 9.7. Pro rozdělení objektů do shluků je třeba vytvořit matici vzdáleností mezi všemi objekty nalezenými v Houghově akumulátoru $\mathcal{H}(\alpha, r)$. Nastavením prahové hodnoty T pro rozdělení parametrů přímek do jednotlivých shluků je možno nastavit citlivost algoritmu na vzájemnou vzdálenost úseček a rozptyl úhlu jednotlivých úseček reprezentujících stěny a jejich natočení. Při testování navrženého algoritmu se hodnoty parametrů pohybovaly v rozmezí $\alpha \in (0, \pi)$ a hodnota $r_{max} \approx 10m$ (viz obr.9.6). Aby byla citlivost na oba parametry (α, r) nastavena shodně, byly hodnoty koeficientů $(\alpha_1 \dots \alpha_n)$ vynásobeny hodnotou $z = \frac{r_{max}}{\pi}$. Pro výpočet matice vzdáleností byla využita základní, euklidovská vzdálenost mezi objekty. Pro přiřazení nebo vyloučení objektu ke shluku bylo použito vzájemné vzdálenosti nejbližších prvků shluků. Tímto nastavením bylo dosaženo výsledku zobrazeného na obrázku 9.8.

V levé části obrázku 9.8 je znázorněn výstup z laserového snímače, který byl získán při reálném měření. Také jsou zde zobrazeny nalezené přímky, průsečíky přímek a počátky a konce přímek. Zelenou barvou jsou vyznačeny průsečíky přímek, které neodpovídají rohům místnosti. Platné přímky jsou zobrazeny čárkovaně. Konce a počátky úseček jsou označeny čtvercovou značkou, zatímco průsečíky jsou zobrazeny jako kružnice.



Obrázek 9.8: Data z laserového proximitního snímače, Houghův akumulátor se zvýrazněnými shluky



Obrázek 9.9: Lokální sensorická a geometrická mapa

V pravé části obrázku je zobrazen Houghův akumulátor $\mathcal{H}(\alpha, r)$, pro toto měření (pro větší názornost není měřítko upraveno koeficientem z). V $\mathcal{H}(\alpha, r)$ jsou barevně ohraničeny shluky bodů. Každý shluk reprezentuje jednu z přímek nalezených v binárním obraze měření. V každém shluku je následně nalezen bod reprezentující přímku, na níž leží největší počet bodů (lokální maximum hledané v oblasti shluku). Souřadnice tohoto bodu (α, r) jsou použity pro aproximaci měření.

Nalezené parametry mohou být využity přímo jako jeden z hledaných znaků popisující zkoumanou oblast nebo jako vstup do další části algoritmu, která body ležící v okolí nalezené přímky aproximuje pomocí metody nejmenších čtverců. Zároveň jsou určeny všechny průsečky přímek.

Jestliže se průsečík nachází v blízkém okolí bodů náležejícím oběma přímkám (vzdálenost je stanovena podle rovnice 9.4), jedná se o roh místnosti. V opačném případě průsečík přímek neodpovídá reálnému orientačnímu bodu. Jsou stanoveny hranice úseček (počátek, konec), které reprezentují zdi zkoumané oblasti.

Na obrázku 9.9 jsou znázorněny lokální mapy stejného prostředí, levá část ukazuje lokální mapu tak, jak byla získána měřením. Pravá část zobrazuje geometrickou lokální mapu prostředí, která byla získána pomocí Radonovy transformace a následného zpracování shlukovacím algoritmem. V geometrické mapě jsou také zobrazeny orientační body $\mathbf{x}_{Li} = (x_{Li}, y_{Li})$, odpovídající rohům místnosti. Pro model snímače, jehož výstupem je poloha orientačních bodů platí:

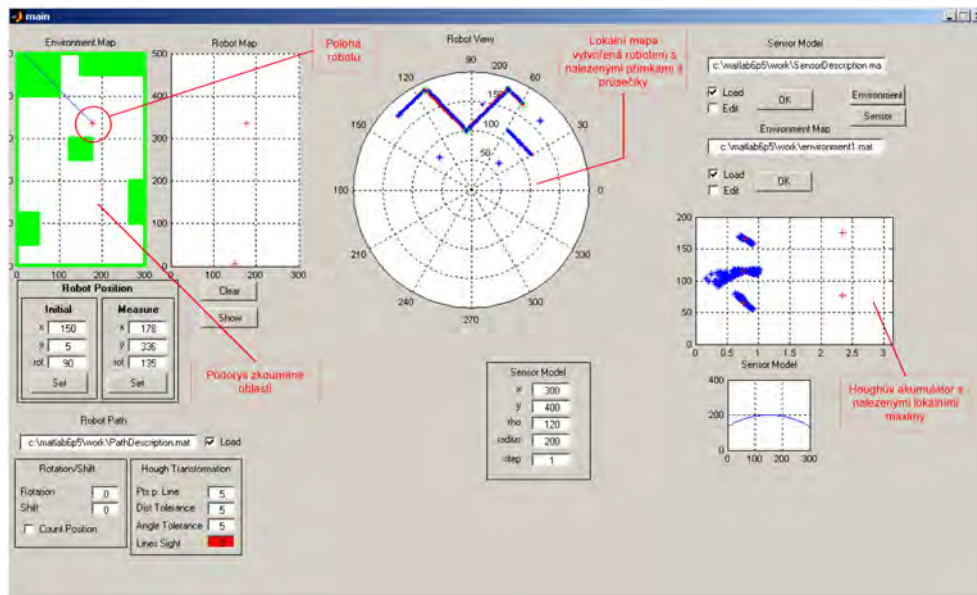
$$h(\mathbf{x}_{(k)}, \mathbf{w}_{(k)}) = \begin{pmatrix} r_i \\ \varphi_i \end{pmatrix} = \begin{pmatrix} \sqrt{(x_{R(k)} - x_{Li})^2 + (y_{R(k)} - y_{Li})^2} \\ \tan\left(\frac{y_{R(k)} - y_{Li}}{x_{R(k)} - x_{Li}}\right) - \varphi_R \end{pmatrix} + \begin{pmatrix} w_{r(k)} \\ w_{\varphi(k)} \end{pmatrix} \quad (9.5)$$

Jedná se o model snímače ², který definuje polohu orientačních bodů v polárních souřadnicích. Tento model byl zvolen z důvodu snazšího zohlednění šumového signálu $\mathbf{w}_{(k)}$ (viz kapitola 8.2). V této práci jsou považovány složky šumového signálu za nezávislé na čase. Označení časové závislosti je uvedeno pro úplnost definice matematického modelu snímače. Funkce $h(\mathbf{x}_{(k)}, \mathbf{w}_{(k)})$ je v podstatě převodním vztahem mezi souřadným systémem globální mapy m a lokálním souřadným systémem robotu, který je použit pro reprezentaci nalezených orientačních bodů. Vzhledem k tomu, že snímač je nedílnou součástí robotu, lze střed souřadného systému snímače snadno odvodit ze známých rozměrů robotu a snímače.

9.4 Simulátor pracovního prostředí robotu

Pro ověření funkčnosti navržených algoritmů pro detekci orientačních bodů byl v programu Matlab navržen simulátor. Jednoduchým nastavením lze změnit matematický model použitého snímače. Je možné zde definovat prostředí, ve kterém se robot pohybuje, zvolit trasu a body v nichž robot provádí měření. Přestože lze vytvářet i globální mapy prostředí, byl simulátor využit pouze pro testování detekčních algoritmů. Hlavní obrazovka simulátoru je zobrazena na obrázku 9.10.

²Vytvořený model snímače popisuje převodní vztah mezi lokálním souřadnicovým systémem robotu a globálním souřadnicovým systémem, ve kterém je tvořena globální mapa prostředí.



Obrázek 9.10: Zobrazení výsledku simulace

9.5 Závěr

V kapitole je uveden výčet způsobů pro segmentaci obrazové funkce O získané pomocí laserového snímače. Z těchto způsobů byl pro svoji univerzálnost a necitlivost na rostoucí vzdálenost mezi body měření zvolen algoritmus Radonovy (Houghovy transformace). Oba tyto algoritmy převádí problém hledání geometrických prvků v obraze na hledání lokálních maxim v Houghově akumulátoru \mathcal{H} . V případě, že se jedná o aproximaci pomocí přímek, je Houghův akumulátor tvořen maticí buněk popsaných parametry α, r , které jsou definovány normálovou rovnicí přímky viz obr. 9.2. V této matici jsou pak shlukovou metodou hledány shluky bodů, jsou popisem jednotlivých přímek. V závěru kapitoly je uveden matematický model laserového snímače $h(\mathbf{x}_{(k)}, \mathbf{w}_{(k)})$. Tento matematický model byl vytvořen za účelem výpočtu inovačního vektoru při kroku aktualizace stavového vektoru (viz kapitola 6).

Kapitola 10

Současná lokalizace a mapování

Výhodou tohoto přístupu k mapování a lokalizaci v mobilní robotice je skutečnost, že mobilní robot je schopen pohybu v předem neznámém prostředí. Pro orientaci v prostředí nepotřebuje zásahu člověka a vytvoření umělých orientačních bodů. Pro určení polohy jsou využívána měření ze snímačů, tato data jsou současně využívána i pro vytváření mapy, ve které se robot lokalizuje, což činí mobilní robot autonomním (ve smyslu schopnosti samostatného pohybu prostorem) [64]. V tabulce 10.1 jsou uvedeny veličiny a jejich hodnoty využívané pro algoritmus SLAM.

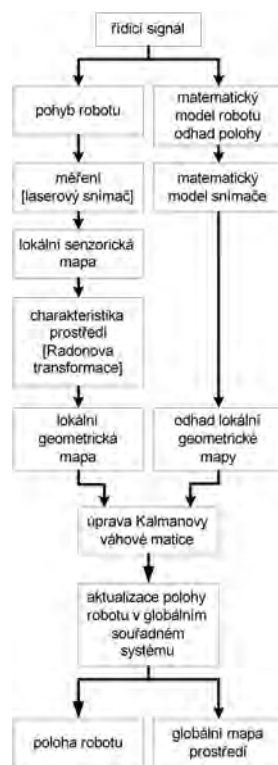
Tabulka 10.1: Veličiny využívané při tvorbě mapy prostředí metodou SLAM

<i>veličina</i>	<i>hodnota</i>
poloha robotu	neznámá
mapa prostředí poloha orientačních bodů	neznámá

Z pravděpodobnostního úhlu pohledu na metodu SLAM (kap. 6) existují dva základní druhy tohoto algoritmu:

- *Online SLAM* - algoritmus, při němž jsou pro určení polohy robotu a orientačních bodů využívány pouze proměnné (poloha, poloha překážek, ...), které jsou aktuální v prováděném kroku k . Po té, co jsou tyto proměnné zpracovány, nejsou ukládány v paměti robotu. Tento algoritmus je inkrementální a umožňuje vytvoření mapy prostředí o téměř libovolné velikosti. Data mohou být přidávána do mapy v libovolném počtu kroků. Jelikož je výpočetní náročnost online SLAMu kvadraticky závislá na počtu orientačních bodů, je velikost globální mapy omezena výpočetním výkonem. Nevýhodou tohoto algoritmu je potřeba velmi robustního procesu pro rozhodování o tom, kterému orientačnímu bodu mapy $\mathbf{m} = \{\mathbf{m}_1, \mathbf{m}_2, \dots, \mathbf{m}_n\}$ odpovídá orientační bod \mathbf{m}_{ik} získaný v kroku k . Při řešení online SLAMu se jedná o hledání aposteriorní pravděpodobnosti: $p(\mathbf{x}_k, \mathbf{m} \mid \mathbf{o}_{1:k}, \mathbf{u}_{1:k})$ (viz kapitola 6.3),
- *Celkový SLAM* - pro lokalizaci robotu a vytvoření mapy je využívána posloupnost všech stavů robotu a orientačních bodů získaných během robotického průzkumu.

Tento algoritmus je vhodný pro zpracování sady dat (stav robotu, stav orientačních bodů) o konstantní velikosti. Není vhodný pro přidávání nových měření v libovolném počtu kroků. Řešení tohoto problému vede na vytvoření orientovaného grafu. Tento algoritmus je proto nazýván jako *GraphSLAM*. Vzhledem k tomu, že všechna data získaná během průzkumu jsou ukládána v paměti robotu a vyhodnocení probíhá až po skončení průzkumu, lze pro hledání korepondujících orientačních bodů v tomto případě využít iterativních metod. V porovnání s metodou online SLAMu je pomocí GraphSLAMu možno vytvořit mapu prostředí s mnohem větší přesností [18]. V případě GraphSLAMu je hledána a posteriori pravděpodobnost: $p(\mathbf{x}_{1:k}, \mathbf{m} \mid \mathbf{o}_{1:k}, \mathbf{u}_{1:k})$ (viz kapitola 6.3).



Obrázek 10.1: EKF SLAM

Navržený algoritmus náleží k online algoritmům pro SLAM. Pro odhad polohy robotu a vytváření mapy je využito rozšířeného Kalmanova estimátoru stavů. Matematický model robotu a laserového snímače jsou uvedené v předchozích kapitolách. Popis těchto prvků byl uveden s ohledem na jejich využití pro algoritmus SLAM. Na obrázku 10.1 je zobrazeno schéma online SLAMu s využitím EKF. Jsou zde naznačeny jednotlivé fáze jednoho kroku algoritmu.

10.1 Inicializace

Střed globálního souřadnicového systému je nastaven do bodu, kde se nachází robot $\mathbf{x}_0 = [0, 0, 0]^T$. Díky tomuto nastavení lze na globální lokalizační úlohu pohlížet jako na úlohu lokální, viz kapitola 4. V tomto okamžiku jsou také nastaveny hodnoty všech kovariančních matic. Hodnota matice Q vyplývá z odhadu vlivu šumových signálů vstupujících do systému a ovlivňujících tak polohu mobilního robotu při pohybu. Matice R pak zohledňuje šum v měření. Hodnoty těchto matic byly nastaveny na základě experimentů, které jsou popsány v kapitole 8.

$$P = \begin{pmatrix} 0,1 & 0 & 0 \\ 0 & 0,1 & 0 \\ 0 & 0 & 0,1 \end{pmatrix} \quad (10.1)$$

$$Q = \begin{pmatrix} 0,1 & 0 & 0 \\ 0 & 0,1 & 0 \\ 0 & 0 & 0,1 \end{pmatrix} \quad (10.2)$$

$$R = \begin{pmatrix} 0,1 & 0 \\ 0 & 0,1 \end{pmatrix} \quad (10.3)$$

V této poloze je provedeno první měření prostředí a segmentace naměřených dat pomocí přímků a jejich průsečíků. Tyto význačné rysy jsou uloženy do matice $\mathbf{m}_G = (\mathbf{m}_1, \dots, \mathbf{m}_n)$, čímž je vytvořena globální mapa pro $k = 1$.

10.2 Predikce stavového vektoru a hodnot měření

Na základě řídicího signálu $\mathbf{u}_k = (v_{Lk}, v_{Pk})$ se robot přemístí do polohy $\mathbf{x}_{R(k+1)} = (x_{(k+1)}, y_{(k+1)}, \varphi_{(k+1)})$. Odhad polohy robotu $\tilde{\mathbf{x}}_{R(k+1)}$ je určen z matematického modelu robotu, který byl definován v kapitole 8.1. Rovnice 10.4 popisuje přenosovou funkci $f(\mathbf{x}_{(k)}, \mathbf{u}_{(k)}, 0)$

$$\tilde{\mathbf{x}}_{R(k+1|k)} = \begin{pmatrix} \tilde{x}_{(k+1|k)} \\ \tilde{y}_{(k+1|k)} \\ \tilde{\varphi}_{(k+1|k)} \end{pmatrix} = \begin{pmatrix} \tilde{x}_{(k|k)} \\ \tilde{y}_{(k|k)} \\ \tilde{\varphi}_{(k|k)} \end{pmatrix} + \begin{pmatrix} \left(\frac{\Delta v_{Pk} + \Delta v_{Lk}}{2} \right) \cos(\tilde{\varphi}_{(k+1|k)}) \\ \left(\frac{\Delta v_{Pk} + \Delta v_{Lk}}{2} \right) \sin(\tilde{\varphi}_{(k+1|k)}) \\ \left(\frac{v_{Lk} - v_{Pk}}{r_{diff}} \right) \end{pmatrix} \quad (10.4)$$

Vzhledem k tomu, že se jedná o nelineární přechodovou funkci, je pro výpočet kovariančních matic a Kalmanovy váhové matice potřeba určit jakobiány této funkce a vyjádřit jejich příslušné hodnoty. Jakobián ∇f_x funkce f vyjádřený pro stav systému \mathbf{x} je definován rovnicí.

$$\nabla f_x = \begin{pmatrix} 1 & 0 & -\left(\frac{v_{Pk} + v_{Lk}}{2} \right) \sin(\varphi_k) \\ 0 & 1 & \left(\frac{v_{Pk} + v_{Lk}}{2} \right) \cos(\varphi_k) \\ 0 & 0 & 1 \end{pmatrix}_{\tilde{\mathbf{x}}_{R(k+1|k)}} \quad (10.5)$$

Pro jakobián ∇f_ϑ funkce f vyjádřený podle šumového vektoru ϑ platí:

$$\nabla f_\vartheta = \begin{pmatrix} \cos(\varphi_k) - \left(\frac{v_{Pk} + v_{Lk}}{2} \right) \sin(\varphi_k) & -\left(\frac{v_{Pk} + v_{Lk}}{2} \right) \sin(\varphi_k) \\ \sin(\varphi_k) + \left(\frac{v_{Pk} + v_{Lk}}{2} \right) \cos(\varphi_k) & \left(\frac{v_{Pk} + v_{Lk}}{2} \right) \cos(\varphi_k) \\ 1 & 1 \end{pmatrix}_{\mathbf{u}_{(k)}} \quad (10.6)$$

Na základě odhadu polohy robotu $\tilde{\mathbf{x}}_{R(k+1|k)}$ je z globální mapy \mathbf{m}_G určen odhad

měření pro krok $k + 1$. Odhad polohy vychází z modelu definovaného v kapitole 8.2. Převod souřadnic je prováděn podle rovnice 9.5.

V poloze $\mathbf{x}_{R(k+1)}$ je laserovým snímačem získáno měření \mathbf{y}_R . Metodou Radonovy transformace jsou z obrazu extrahovány přímky a orientační body. Tyto orientační body tvoří lokální mapu $\mathbf{m}_{L(k+1)} = (\mathbf{x}_{L1(k+1)}, \dots, \mathbf{x}_{Lm(k+1)})$. Počet prvků m lokální mapy nemusí být nutně shodný s počtem prvků n globální mapy.

10.3 Hledání korespondujících orientačních bodů

V okamžiku, kdy je na základě modelu systému a snímače určena předpokládaná poloha orientačních bodů v prostoru, je nutné nalézt dvojice vzájemně si odpovídajících orientačních bodů z globální a lokální mapy. Pro každou kombinaci prvků \mathbf{x}_{Lj} a \mathbf{x}_{Gk} je určena jejich vzájemná euklidovská vzdálenost (viz rovnice 9.4). Jako odpovídající si je vybrána dvojice bodů, pro jejichž vzájemnou vzdálenost platí:

$$d_{tE} > \min_{j \in \{1 \dots m\}, k \in \{1 \dots n\}} \left| \sqrt{(x_{Lj} - x_{Gk})^2 + (y_{Lj} - y_{Gk})^2} \right| \quad (10.7)$$

Kde hodnota d_{tE} je prahová hodnota pro stanovení příslušnosti bodů. Tato hodnota je odhadnuta na základě kovariančních matic systému a snímače.

V případě, že orientační body nepředstavují reálné body v prostoru, je možné navzájem si odpovídající body nalézt pomocí inovačního vektoru 6.42a jeho kovarianční matice 6.43. Jako odpovídající si je vybrána dvojice bodů, pro jejichž vzájemnou vzdálenost platí:

$$d_{tM} > \min \left| \sqrt{\Delta y_{(k+1)} \chi_{\Delta \Delta(k+1|k)} \Delta y_{(k+1)}^T} \right| \quad (10.8)$$

Kde hodnota d_{tM} je prahová hodnota pro stanovení příslušnosti bodů. Tento způsob vyhodnocení shody orientačních bodů nebere v úvahu pouze hodnotu stavu (polohu) orientačního bodu, ale pravděpodobnost odhadu polohy orientačního bodu. Při vyhodnocení shody je tedy brán zřetel i na rozptyl hustoty pravděpodobnosti odhadu stavu orientačního bodu. Výpočet jakobiánů nutných pro určení hodnoty inovačního vektoru je uveden v následující podkapitole.

Při vyhodnocování shody může nastat stav, ve kterém si některé z bodů globální mapy m a lokální mapy l , nejsou navzájem přiřazeny. Tento stav může nastat ve dvou případech:

- orientační bod globální mapy \mathbf{m}_G se nenachází v zorném poli snímače. Tento bod není ve stávajícím kroku využit pro lokalizaci,

- v lokální mapě \mathbf{m}_L byl nalezen nový orientační bod, který není ve stávajícím kroku využit pro lokalizaci. V kroku mapování může být tento bod zanesen do globální mapy \mathbf{m}_G jako nový orientační bod.

Pro vyhodnocování vzájemné příslušnosti bodů byl při realizaci algoritmu zvolen první způsob, tzn. byla vyhodnocována pouze geometrická vzdálenost bodů.

10.4 Aktualizace stavového vektoru

Dvojice navzájem si příslušejících bodů jsou v kroku aktualizace stavu využity pro vytvoření inovačního vektoru $\Delta y_{(k+1)}$ (viz rovnice 6.42). Pro vytvoření inovačního vektoru je využito příslušných jakobiánů výstupní přenosové funkce $h(x_{(k)}, w_{(k)})$ (viz rovnice 9.5) vyjádřených podle stavu systému \mathbf{x} a šumového signálu w .

$$\nabla h_x = \begin{pmatrix} 2(x_R - x_L) [(x_R - x_L)^2 - (y_R - y_L)^2]^{-\frac{1}{2}} & 2(x_R - x_L) [(x_R - x_L)^2 - (y_R - y_L)^2]^{-\frac{1}{2}} & 0 \\ \frac{1}{\cos^2 \frac{y_R - y_L}{x_R - x_L}} \frac{-(y_R - y_L)}{(x_R - x_L)} & \frac{1}{\cos^2 \frac{y_R - y_L}{x_R - x_L}} \frac{(x_R - x_L)}{(y_R - y_L)} & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{matrix} \\ \\ \end{matrix} \mathbf{x}_{R(k+1|k)} \quad (10.9)$$

$$\nabla h_w = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad (10.10)$$

Pomocí těchto jakobiánů je z rovnice 6.44 určena aktualizovaná hodnota kovarianční matice stavového vektoru $P_{(k+1|k+1)}$ a následně i aktualizovaný stavový vektor $\mathbf{x}_{R(k+1|k+1)}$. Popsaný krok aktualizace zpracovává polohy jednotlivých orientačních bodů postupně. V každém kroku je pro určení polohy robotu využit pouze jeden orientační bod. Tento způsob se nazývá *sekvenčním krokem aktualizace* (sequential update step). Výhodou tohoto přístupu je jeho snadnější implementace a možnost nastavit váhy jednotlivých orientačních bodů.

Další možností je tzv. *aktualizace po dávkách* (batch update step), při níž jsou jakobiány pro všechny orientační body sloučeny do jedné matice. Této matici je pak vhodně přizpůsobena také kovarianční matice stavu robotu a krok aktualizace je proveden ze všech poloh orientačních bodů naráz. Tento způsob aktualizacího kroku umožňuje využití stavových veličin pro rozšíření globální mapy. V tomto případě by se globální mapa robotu skládala nejen z orientačních bodů, případně rysů charakterizujících prostředí, ale mohla by obsahovat také hodnoty stavového vektoru v jednotlivých krocích. Využití rozšířené globální mapy umožňuje například slučování jednotlivých map v systémech s větším počtem robotů [71].

10.5 Aktualizace globální mapy

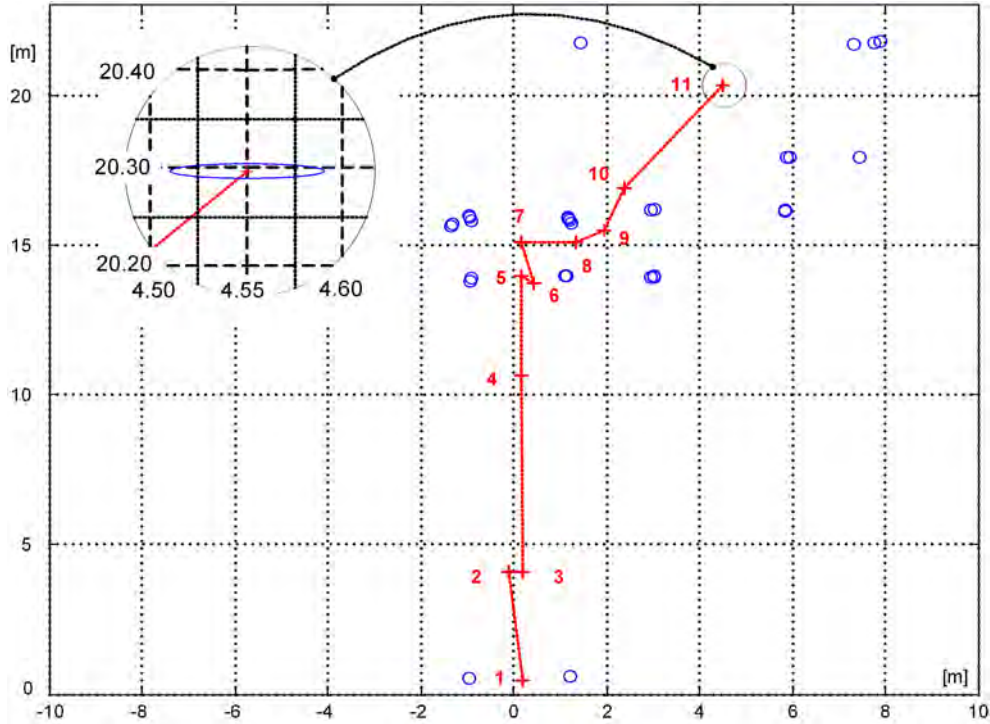
V tomto kroku je poloha orientačních bodů z lokální mapy \mathbf{m}_L , které nebyly využity pro úlohu lokalizace (viz kapitola 10.3) přepočítána podle rovnic 10.11 a 10.12 na základě aktualizovaného odhadu polohy robotu do globálních souřadnic. Tyto body

jsou pak přidány do globální mapy \mathbf{m}_G .

$$x_{iG} = r_i \cos(\varphi_i - \varphi_{R(k+1|k+1)}) + x_{R(k+1|k+1)} \quad (10.11)$$

$$y_{iG} = r_i \sin(\varphi_i - \varphi_{R(k+1|k+1)}) + y_{R(k+1|k+1)} \quad (10.12)$$

Krokem aktualizace globální mapy je smyčka algoritmu SLAM uzavřena a lze pokračovat zpracováním měření v nové poloze robotu. Globální mapa prostředí je tvořena polohami jednotlivých orientačních bodů.



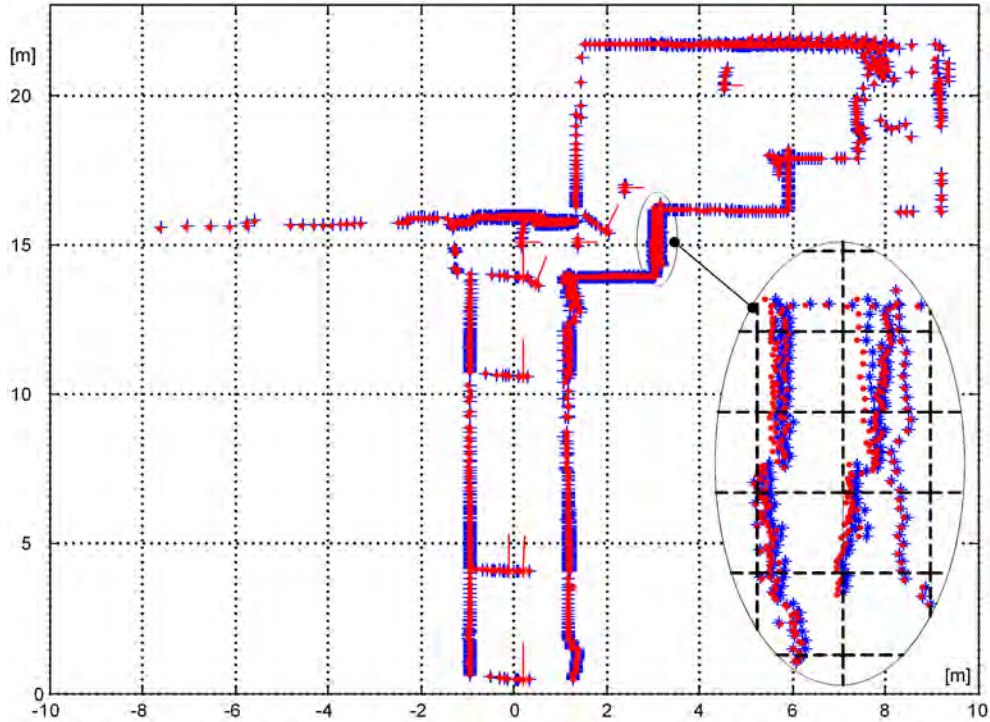
Obrázek 10.2: Globální mapa prostředí

Na obrázku 10.2 je zobrazena globální mapa prostředí, která je výsledkem realizovaného algoritmu SLAM. Modrou barvou jsou v mapě zobrazeny orientační body. Červenou čarou je vynesena trajektorie robotu a body, ve kterých bylo prováděno měření. V levém horním rohu obrázku je zobrazena elipsa, která odpovídá rozptylu hustoty pravděpodobnosti odhadnuté polohy mobilního robotu na konci mapování.

10.6 Experimentální ověření algoritmu SLAM

Navržený algoritmus byl experimentálně otestován v prostředí budovy VUT v Brně. Pro dané prostředí byl vytvořen příslušný model robotu. Materiál podlahy byl tvořen keramickými dlaždicemi. Vzhledem k tomu, že se jednalo o vnitřní prostory budovy, byl splněn i předpoklad rovných zdí, které bylo možno nahradit přímkami. Prostedí bylo snímáno z různých poloh pod různými úhly. Na obrázku 10.3 jsou naznačena

místa a natočení robotu, při kterém byla prováděna měření. Na základě matematického modelu robotu byl proveden odhad polohy robotu, který byl následně aktualizován informací získané Radonovou transformací z měření.



Obrázek 10.3: Lokální sensorické mapy

Na obrázku 10.3 jsou zobrazeny lokální sensorické mapy získané při jednotlivých měřeních. Poloha jednotlivých bodů měření je přepočítána do globálního souřadného systému. Modrou barvou je vyneseno měření v polohách $\mathbf{x}_{R(1...k)}$, které byly zjištěny přesným měřením (pomocí pásma a úhloměru). Tyto polohy nebyly využity pro konstruování mapy, ale pouze pro kontrolu přesnosti algoritmu SLAM. Červené body jsou lokální sensorické mapy přepočítané na základě odhadu polohy robotu $\tilde{\mathbf{x}}_{R(1...k)}$. V levém horním rohu obrázku 10.2 je zobrazena elipsa, která odpovídá rozptylu hustoty pravděpodobnosti odhadu polohy mobilního robotu na konci mapování. Rozptylu odhadu polohy $\tilde{\mathbf{x}}_{R(1...k)}$ odpovídá přibližně 1% z rozměru globální mapy.

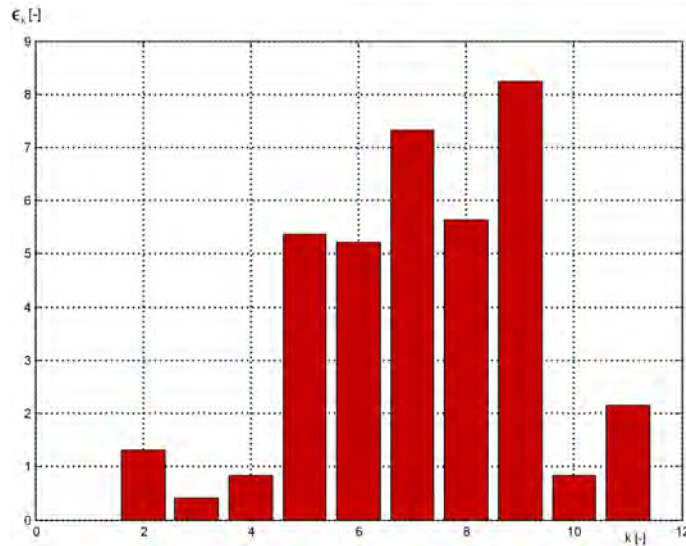
Kvalitu odhadu lze posoudit pomocí kritéria NEES (normalizovaný kvadrát odchylek odhadu), které je definované jako:

$$\epsilon_k = (\mathbf{x}_{Rk} - \tilde{\mathbf{x}}_{R(k|k)}) P_{(k|k)}^{-1} (\mathbf{x}_{Rk} - \tilde{\mathbf{x}}_{R(k|k)})^T \quad (10.14)$$

a její průměrné hodnoty pro všech N kroků algoritmu SLAM:

$$\bar{\epsilon} = \frac{1}{N} \sum_{i=1}^N \epsilon_i \quad (10.15)$$

Odhad stavového vektoru je neposunutý, pokud se hodnota $\bar{\epsilon}_k$ pro N jdoucí k nekonečnu blíží rozměru stavového vektoru [73]. V grafu 10.5 je zobrazen vývoj kvadratické chyby pro jednotlivé kroky měření. Kroky měření jsou označeny na obrázku 10.2.



Obrázek 10.5: Kvadratická chyba odhadu polohy robotu v jednotlivých krocích

Z grafu je patrná klesající tendence hodnoty chyby v případě opakovaného pozorování orientačních bodů (kroky 2-4). Krok 5 odpovídá přidání nových orientačních bodů do globální mapy.

Hodnota koeficientu $\bar{\epsilon}$ pro $N = 11$ je: $\bar{\epsilon} = 3,76$, což odpovídá 92. percentilu. Tato hodnota vyjadřuje, že navržený Kalmanův estimátor je optimistický (nesplňuje podmínku definovanou 10.13) [73]. Důsledek této skutečnosti může při mapování rozlehlých prostor vést k nedostatečné přesnosti odhadu polohy a selhání algoritmu.

Nárůstu kvadratické chyby odhadu lze zabránit zvýšením počtu orientačních bodů, čehož lze docílit vhodným nastavením segmentačního algoritmu viz kapitola 9 tak, aby byly detekovány i krátké úsečky, které se nacházejí v těsné blízkosti (odpovídající např. zárubním dveřím). Druhou možností je provádět aktualizaci odhadu stavového vektoru pomocí dávkového algoritmu, který je popsán v 10.4. Nevýhodou tohoto postupu je však nutnost výpočtu inverzních matic s mnoha prvky, což při globálních mapách

rozlehlých prostředí může být neúměrně výpočetně náročné.

Kapitola 11

Shrnutí dosažených cílů

V práci je řešena problematika současné lokalizace a mapování v oblasti mobilní robotiky. První část práce je věnována popisu možných způsobů lokalizace a různých druhů map využívaných pro navigaci mobilních robotů. Přestože jsou tyto dvě úlohy popisovány odděleně, ve většině případů je v mobilní robotice využívána jejich vzájemná kombinace. Jedním z možných případů je současná lokalizace a mapování (SLAM), při níž jsou shodná data využívána jak pro tvorbu robotické mapy, tak i pro lokalizaci v této mapě.

Omezujícím faktorem pro návrh algoritmu byl předpoklad, že lze využívat pouze data získaná ze snímačů umístěných na robotu. Z tohoto důvodu byl zvolen pravděpodobnostní přístup zpracování informací ze sensorů a matematického modelu robotu. Tento způsob, založený na principu podmíněné pravděpodobnosti, umožňuje sloučení těchto dvou zdrojů informace. V kapitole 6 je uveden popis pravděpodobnostního přístupu lokalizace a tvorby map. Jsou zde také uvedené způsoby možného řešení úlohy s ohledem na popis polohy robotu a orientačních bodů pomocí Gaussova rozložení pravděpodobnosti.

Pro řešení algoritmu SLAM byl zvolen rozšířený Kalmanův filtr (EKF). Pomocí tohoto algoritmu je na základě matematického modelu robotu a snímače, které jsou popsány v kapitole 8, určen odhad polohy robotu. Pro to, aby bylo možno stanovit odhad polohy mobilního robotu, byl vytvořen matematický model umožňující záměnu robotického podvozku řízeného smykem za podvozek řízený změnou rychlosti jednotlivých kol. Byly stanoveny podmínky, za kterých je možné tuto záměnu provést. Pro ověření správnosti předpokladu bylo provedeno experimentální ověření. Výsledky experimentů byly využity nejen pro ověření vhodnosti navrženého modelu, ale také pro stanovení parametrů potřebných pro algoritmus EKF.

Vzhledem k tomu, že navržený algoritmus SLAM patří do skupiny algoritmů, které pro svoji funkci využívají zjišťování polohy orientačních bodů, je v kapitole 9 navržen vlastní způsob segmentace naměřených dat využívající Radonovy transformace, která umožňuje získání parametrů geometrických prvků využívaných pro popis prostředí. Pracovní prostředí robotu je snímáno pomocí laserového planárního snímače. Výstupem segmentačního algoritmu je poloha orientačních bodů v lokálním souřadnicovém

systému robotu. Polohy těchto bodů jsou v mapovací části navrženého algoritmu interpretovány jako informace o zkoumaném prostředí.

Podrobný popis jednoho kroku navrženého algoritmu, ve kterém je určen odhad polohy a doplněny nové orientační body do globální mapy, je uveden v kapitole 10. V této kapitole je také popsán experiment, který byl proveden za účelem ověření navrženého způsobu lokalizace a tvorby mapy. Výsledkem experimentu je globální mapa zkoumaného prostředí. V závěru této kapitoly je také uvedeno zhodnocení dosažené kvality globální mapy.

Všechny navržené algoritmy byly otestovány s využitím reálných dat, která byla získána pomocí laserového snímače SICK LMS 100 a robotu UTAR. Algoritmy byly implementovány v prostředí programu Matlab verze 6.5. Pro otestování funkčnosti segmentačních algoritmů založených na Houghově a Radonově transformaci byl v témže prostředí navržen a realizován simulátor, který umožňuje nastavení parametrů prostředí a simulaci různých modelů snímače. Využití simulátoru není omezeno na laserové snímače (triangulační, TOF), lze simulovat také modely jiných druhů snímačů, například radaru nebo ultrazvukových snímačů.

Možná rozšíření navržené metody

Přestože byla otestována funkčnost všech navržených částí algoritmu SLAM a cíle disertační práce definované v kapitole 2 byly splněny, pro další výzkum v oblasti lokalizace a mapování je množství dílčích úloh, které s řešením problému souvisí a přesto nebyly v práci řešeny.

Prvním krokem k získání k autonomního mobilního robotu je implementace navrženého algoritmu v řídicím systému využívaném mobilním robotem UTAR. Tímto by bylo zaručené zpracování naměřených dat v reálném čase a lokalizační data by bylo možno využít například pro plánování trajektorie mobilního robotu.

Přesnost a robustnost navrženého algoritmu by bylo možné zvýšit zvláště využitím všech dosažitelných parametrů popisujících zkoumané prostředí (viz kapitola 9). Implementace těchto informací získaných z měření by umožnila využití algoritmu pro mapování rozlehlých prostor, kde počet měření nabývá řádově větších hodnot. V testovaném případě bylo za účelem vytvoření globální mapy prostředí provedeno jedenáct měření.

Vzhledem k tomu, že použitý laserový snímač nedokáže rozpoznat opticky transparentní prostředí, je pro reálné využití nutné tento snímač doplnit snímačem pracujícím na jiném principu (např. ultrazvukový snímač vzdálenosti) a měření z těchto snímačů vhodně kombinovat.

Pro využití algoritmu ve venkovním prostředí je nutné robotický systém doplnit o možnost měření nerovnosti povrchu, po kterém se robot pohybuje. Také segmentační metoda založená na Radonově transformaci není vhodná pro nestrukturované vnější prostředí a musela by být nahrazena (doplněna) metodou jinou (například orientační body definované pomocí GPS souřadnic).

Literatura

- [1] <wikipedia.org> [cit. 19.2.2007]
- [2] *American Practical Navigator*, <www.irbs.com/bowditch> [cit. 19.2.2007]
- [3] NEWMAN, P., TARASSENKO, L., *Robotics and Cognition*, Research Review, <www.tech.plym.ac.uk/soc/staff/GuiBugm/pub/foresightroboticsfinal.pdf> [cit. 21.2.2007]
- [4] GRASSO, F. W., *Invertebrate-Inspired Sensory-Motor Systems and Autonomous, Olfactory-Guided Exploration*, Biological Bulletins, Volume 200, p.160-168 (2001), <www.biolbull.org/cgi/reprint/200/2/160.pdf> [cit. 21.2.2007]
- [5] ŠPAŇÁR, J., *Latinsko-slovenský slovník*, Slovenské pedagogické nakladateľstvo Bratislava, 1983, ISBN 67-475-83
- [6] <www.robotika.sk> [cit. 10.1.2008]
- [7] VLACHÝ, D., *Využití GPS snímače v robotice*, Diplomová práce, VUT Brno FSI, 2006
- [8] CSORBA, M., *Simultaneous Localisation and Mapping*, Ph.D. thesis, Oxford, 1997
- [9] *Technical documentation Garmin GPS 25LP*, <www.garmin.com> [cit. 24.2.2007]
- [10] WANG, L. Ch., YONG, L. S., ANG, M. H., *Mobile Localisation For Indoor Environment*, SIMTech Technical Report, 2002 <www.simtech.a-star.edu.sg> [cit. 24.2.2007]
- [11] MODRLÁK, O., *Základy analýzy a syntézy ve stavovém prostoru*, studijní materiály, Katedra řídicí techniky Technická Univerzita v Liberci, 2004, <www.fm.vslib.cz/~krt/krt_cz/krt_om.htm> [cit. 29.8.2007]
- [12] BORENSTEIN, J., EVERETT, H. R., FENG, L., „Where am I?“ *Sensors and Methods for Mobile robot Positioning*, University of Michigan, 1996
- [13] ACHIM, L., TUCKET, T., *Building gas concentration gridmaps with a mobile robot*, Robotics and Autonomous System 48, 2004 <www.sciencedirect.com> [cit. 26.2.2007]

- [14] TALLURI, R., AGGARWAL, J.K., *Position Estimation for an Autonomous Mobile Robot in an Outdoor Environment*, pp. 573 - 584, IEEE Transactions on Robotics and Automation, vol. 8, No. 5, October 1992, ISSN: 1042-296X
- [15] DURRANT-WHYTE, H., BAILEY, T., *Simultaneous Localisation and Mapping: Part I*, IEEE Robotics & Automation Magazine, June 2006
- [16] DURRANT-WHYTE, H., BAILEY, T., *Simultaneous Localisation and Mapping: Part II*, IEEE Robotics & Automation Magazine, September 2006
- [17] REKTORYS, K. a spol., *Přehled užité matematiky I, II*, Edice odborné literatury, Praha, 2000, ISBN 80-7196-179-5
- [18] THRUN, S., BURGARD, W., FOX, D., *Probabilistic robotics*, The MIT Press, 2005, ISBN 0-262-20162-3
- [19] JULIER, S.S., UHLMANN, J.K., *A New Extension of the Kalman Filter to Nonlinear Systems*, SPIE AeroSense Symposium, April 21-24, 1997. <www.cs.unc.edu/~welch/kalman> [cit. 18.6.2007]
- [20] NING, X., FANG, J., *An autonomous celestial navigation for LEO satellite base on unscented Kalman filter and information fusion*, <www.sciencedirect.com> [cit. 18.6.2007]
- [21] FOX, D., HIGHTOWER, J., LIAO, L., SCHULZ, D., BORRIELLO, G., *Bayesian Filters for Location Estimation*, IEEE Pervasive Computing, July-September 2003, ISSN: 1536-1268
- [22] FRANKLIN, G., F., POWELL, J., D., WORKMAN, M., L., *Digital Control of Dynamic Systems*, Addison-Wesley Publishing Company, Inc., 1990, ISBN: 0-201-51884-8
- [23] WELCH, G., BISHOP, G., *An Introduction to the Kalman Filter*, Department of Computer Science, University of North Carolina at Chapel Hill, <www.cs.unc.edu/~welch/kalman> [cit. 3.8.2007]
- [24] THRUN, S., *Robotic Mapping: A Survey, Exploring artificial intelligence in the new millenium*, Morgan Kaufman Publishers Inc., 2003, ISBN: 1-55860-811-7
- [25] ELFES, A., *Using Occupancy Grids for Mobile Robot Perception and Navigation*, 1989, IEEE Computer, Vol. 22, Issue 6, ISSN 0018-9162
- [26] MARTIN, C. M., MORAVEC, P. H., *Robot Evidence Grids*, Carnegie Mellon University Pittsburgh, CMU-RI-TR-96-06
- [27] <www.sri.com/about/timeline/shakey.html> [cit. 28.8.2007]
- [28] <www.raytheon.com> [cit. 28.8.2007]
- [29] <www.fas.org/irp/agency/dod/uspacom/cmsa/index.html> [cit. 28.8.2007]

- [30] <<http://oldmaps.geolab.cz>> [cit. 28.8.2007]
- [31] TOMATIS, N., NOURBAKHSI, I., SIEGWART, R., *Hybrid simultaneous localisation and map building: natural integration of topological and metric*, Robotics and Autonomous Systems 44, 2003, <www.sciencedirect.com> [cit. 1.9.2007]
- [32] NIETO, J., GUIVANT, J., NEBOT, E., THRUN, S., *Real Time Data Association for FastSlam*, Robotics and Automation, 2003, ICRA-IEEE, pp: 412 - 418,
- [33] NIETO, J., *Detailed Environment Representation for the SLAM Problem*, Ph.D. Thesis, University of Sydney, 2005
- [34] LOSADA, R., D., MATIA, F., GALAN, R., *Building geometric feature based maps for indoor service robots*, Robotics and Autonomous Systems 54, 2006, <www.sciencedirect.com> [cit. 1.9.2007]
- [35] VASUDEVAN, S., GÄCHTER, S., BERGER, M., SIEGWART, R., *Cognitive Maps for Mobile Robots - An Object Based Approach*, Robotics and Autonomous Systems 55, 2007, <www.sciencedirect.com> [1.9.2007]
- [36] TOLMAN, E., C., *Cognitive Maps in Rats and Men*, The Psychological Review 55 (4), 1948, <psychclassics.yorku.ca/Tolma/Maps/maps.htm> [cit. 14.9.2007]
- [37] <www.hokuyo-aut.jp> [cit. 10.1.2008]
- [38] THRUN, S., HAHNELY, D., FERGUSON, D., MONTEMERLO, M., TRIEBEL, R., BURGARD, W., BAKER, Ch., OMOHUNDRO, Z., THAYER, S., WHITTAKER, W., *A System for Volumetric Robotic Mapping of Abandoned Mines*, ICRA 2003, <www.cs.cmu.edu/~thrun> [cit. 12.1.2008]
- [39] <www.cs.cmu.edu/~thrun/mines> [cit. 12.1.2008]
- [40] <www.ri.cmu.edu> [cit. 12.1.2008]
- [41] FITZGIBBON, A. W., *Robust Registration of 2D and 3D Point Sets*, 2001, <www.robots.ox.ac.uk/~vgg> [cit. 12.1.2008]
- [42] <www.ais.fraunhofer.de/ARC/kurt3D> [cit. 12.1.2008]
- [43] NUCHTER, A., LINGEMANN, K., HERTZBERG, J., SURMANN, H., *6D SLAM - Mapping Outdoor Environments*, Gaithersburg, 2006, <www.ais.fraunhofer.de/ARC/kurt3D/>, [cit. 12.1.2008]
- [44] LINGEMANN, K.; NÜCHTER, A.; HERTZBERG, J.; SURMANN, H., *High-speed laser localization for mobile robots*, Robotics and autonomous systems 51 (2005), ISSN: 0921-8890, <www.sciencedirect.com> [cit. 13.1.2008]
- [45] <gerstner.felk.cvut.cz/mobile-robotics> [cit. 13.1.2008]
- [46] <<http://labe.felk.cvut.cz/~pelote/>> [cit. 13.1.2008]

- [47] <www.sick.com> [cit. 14.9.2007]
- [48] <www.fotoroman.cz> [cit. 1.6.2007]
- [49] NEUŽIL, T., JEŽ O., *Data processing for mapping in mobile robotics*, IETA 2007, Springer – Verlag, 2007, (Currently in press)
- [50] NEUŽIL, T., *Simultaneous Mapping and Navigation for Skid Steered Mobile Robot*, Advanced Topics on Signal Processing, Robotics and Automation, WSEAS Press, 2008, ISBN 978-960-6766-442
- [51] LEI, B. J., HENDRIKS, E. A., REINDERS, M. J. T., *On Feature Extracion from Images*, Research Review of MCCWS project, Information and Communication Theory Group, TU Delft, 1999, <<https://doc.telin.nl/dsweb/Get/Document-10236/>> [cit. 10.12.2007]
- [52] ALEMPIJEVIC, A., *High Speed Feature Extraction in Sensor Coordinates for Laser Rangefinders*, Australasian Conference on Robotics & Automation, 2004, <www.araa.asn.au/acra/acra2004> [cit. 1.12.2007]
- [53] NGUYE, V. MARTINELLI, A., TOMATIS, N., SIEGWART, R., *A Comparison of Line Extraction Algorithms using 2D Laser Rangefinder for Idoor Mobile Robotics*, IROS'2005, ISBN 0-7803-891-3, 2005
- [54] HANSEN, K., ANDERSEN, J. D., *Understanding the Hough Transform: Hough cell support and its utilisation*, Image and Vision Computing 15, pp. 205-218
- [55] OLSON, C. F., *Constrained Hough Transforms for Curve Detection*, Computer Vision Understanding 58, pp.329-345
- [56] MEJDANI, E.S., DUBEAU, R. E., *Champ de hauteurs de la transformée de Hough standard*, CIRO'05, pp.133-144
- [57] CENSI, A., LUCA, I., GRISETTI, G., *Scan Matching in the Hough Domain*, ICRA05, <www.informatik.uni-freiburg.de/~grisetti/> [cit. 5.12.2007]
- [58] AVERBUCH, A., COIFMAN, R. R., DONOHO, D. L., ISRAELI, M., WALDÉN, J., *Fast Slant Stack: A notion of Radon Transform in a Cartesian Grid which is Rapidly Computible, Algebraically Exact, Geometrically Faithfull and Invertible*, <www.stat.stanford.edu/~donoho/Reports> [cit. 1.12.2007]
- [59] GINKEL, M., HENDRIKS, L., VLIET, L. J., *A Short Introduction to the Radon and Hough transforms and how they relate to each other*, Quantitative Imaging Group, Report QI-2004-01, Deft University of Technology, <www.ph.tn.tudelft.nl/~michael/> [cit. 15.12.2007]
- [60] <www.mathworks.com> [cit. 15.12.2007]
- [61] LEWIS, F. L., SHUZI, S. S. *Autonomous Mobile Robots*, CRC Press, 2006, ISBN: 0-8493-3748-8

- [62] <www.karlin.mff.cuni.cz/~zichova> [cit. 15.12.2007]
- [63] PAZDERSKI, D., KOZLOWSKI, K., DIXON, W. E., *Tracking and Regulation Control of a Skid Steering Vehicle*, ANS International Conference on Robotics and Remote Systems for Hazardous Environments, 2004, pp. 369-376
- [64] DISSANAYAKE, G. M. W. M., NEWMAN, P., CLARK, S., DURRANT-WHYTE, H. F., CSORBA, M., *A Solution to the Simultaneous Localisation and Map Building (SLAM) Problem*, IEEE transactions on Robotics and Automation, vol. 17, No. 3, 2001, <www.robots.ox.ac.uk/~pnewman> [cit. 15.12.2007]
- [65] CHOSET, H., LYNCH, M. K., HUTCHKINSON, S., KAVRAKI, L. E., BURGARD, W., THRUN, S., *Principles of Robot Motion*, A Bradford Book, MIT Press, 2005, ISBN 0-262-03327-5
- [66] TAMIMI, H., ZELL, A., *Global Robot Localization Using Iterative Scale Invariant Feature Transform*, 36th International Symposium Robotics (ISR 2005), Tokyo, Japan, Nov. 29-Dec. 1, 2005, <www-ra.informatik.uni-tuebingen.de/publikationen/> [cit. 20.12.2007]
- [67] REKLEITIS, J. M., *A Particle Filter Tutorial for Mobile Robot Localization*, Mc.Gill University, Québec, <www.cim.mcgill.ca/~yiannis/ParticleTutorial.html> [cit. 20.12.2007]
- [68] EVENSEN, G., *The Ensemble Kalman Filter: theoretical formulation and practical implementation*, Ocean Dynamics (2003) 53: pp. 343–367, Springer-Verlag, 2003 <www.nersc.no/~geir/Evensen.titles.html> [cit. 20.12.2007]
- [69] BURGARD, W., FOX, D., THRUN, S., *Active Mobile Robot Localization*, IJCAI, 1997 <<http://lims.mech.northwestern.edu/students/solberg/papers/>> [cit. 20.12.2007]
- [70] SIEGWART, R., NOURBAKSH, I. R., *Introduction to Autonomous Mobile Robots*, A Bradford Book, MIT Press, 2004, ISBN 0-262-19502-X
- [71] RIKOSKI, R. J., LEONARD, J. J., NEWMAN, P. M., *Stochastic Mapping Frameworks*, Proceedings of the 2002 IEEE International Conference on Robotics and Automation, May 2002, pp. 426-433, <www.robots.ox.ac.uk/~pnewman> [cit. 1.1.2008]
- [72] CASTELLANOS, J. A., NEIRA, J., TARDÓS, J. D., *Limits to the Consistency of EKF Based SLAM*, IFAC Symposium on Intelligent Autonomous Vehicles, 2004, <webdiis.unizar.es/~jdtardos> [cit. 10.1.2008]
- [73] BAILEY, T., NIETO, J., GUIVANT, J., STEVENS, M., NEBOT, E., *Consistency of the EKF-SLAM Algorithm*, IEEE/RSJ, 2006, <www-personal.acfr.usyd.edu.au/tbailey> [cit. 10.1.2008]

Publikace autora

NEUŽIL, T., *Simultaneous Mapping and Navigation for Skid Steered Mobile Robot*, Advanced Topics on Signal Processing, Robotics and Automation, WSEAS Press, 2008, ISBN 978-960-6766-442

NEUŽIL, T., *Simultaneous Mapping and Navigation for Skid Steered Mobile Robot*, WSEAS Applied Informatics & Communications, 2008, ISSN 1790-5117

NEUŽIL, T., JEŽ, O., *Data processing for mapping in mobile robotics*, IETA 2007

NEUŽIL, T., *Scanner data processing for mapping in mobile robotics*, Modern Trends in Control, 2006, ISBN 80-969224-6-7

NEUŽIL, T., *3D kamerový scanner*, AT&P journal PLUS 5 2004, ISSN 1336-5010

NEUŽIL, T., *Camera Based Laser Proximity Sensor*, ISBN 80-86059-41-3

NEUŽIL, T., *3D Camera Based Laser Proximity Sensor*, EEICT 2004, ISBN 80-214-2636-5

NEUŽIL, T., *CCD Camera Proximity Scanner*, The 6th International Scientific Conference, ISBN 80-7194-662-1

NEUŽIL, T., *CCD Laser Proximity Scanner*, RAAD 2004, , ISBN 80-7204-341-2

NEUŽIL, T., *Laser Proximity Scanner for Mobile Robotics*, DAAAM 2004, ISBN 3-901509-42-9

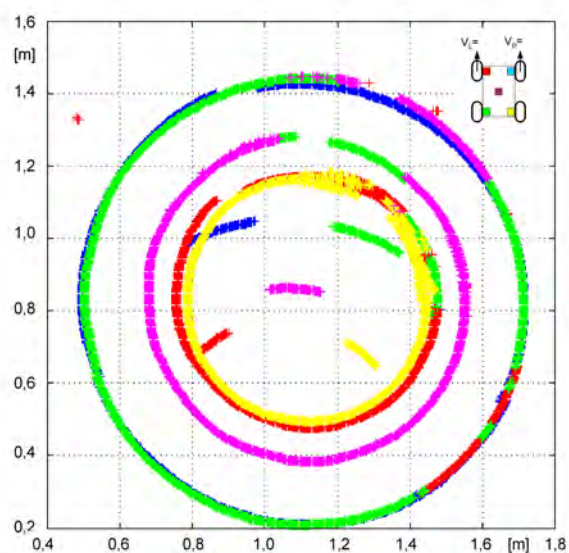
NEUŽIL, T., *Kamerový proximní skener*, Radešín 2003, ISBN 80-214-2479-6

NEUŽIL, T., KOPEČNÝ, L., ŽALUD, L., *Vision System of Universal Telepresence and Autonomous System*, DAAAM 2001, ISBN 3-901509-19-4

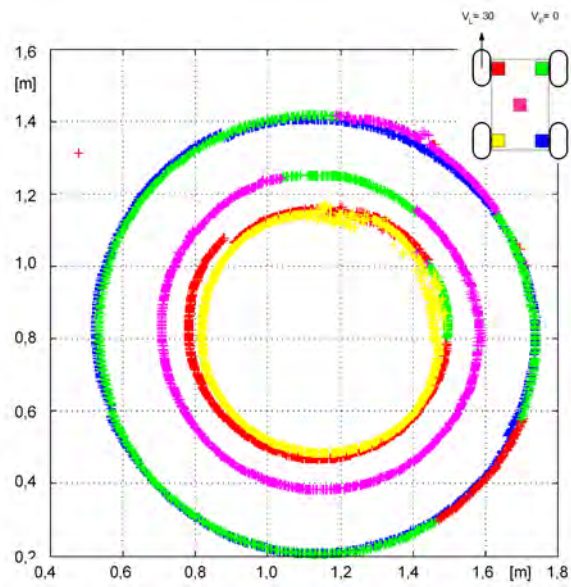
NEUŽIL, T., HONZÍK, B., KOPEČNÝ, L., *Robotic Soccer System*, DAAAM 2001, ISBN 3-901509-19-4

Kapitola 12

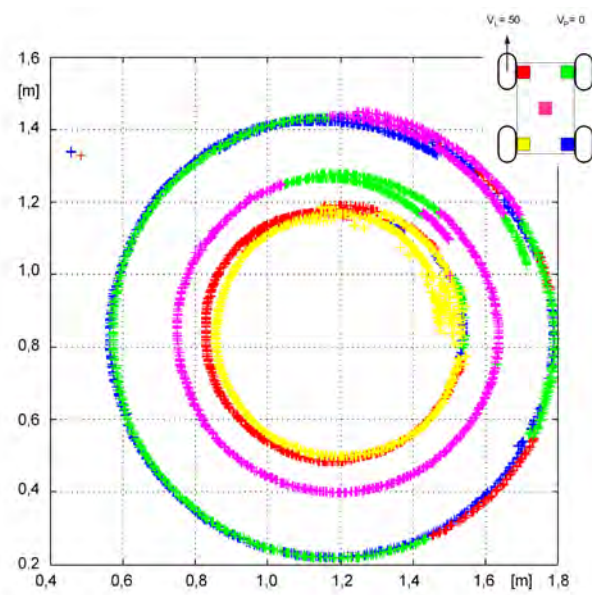
Příloha A



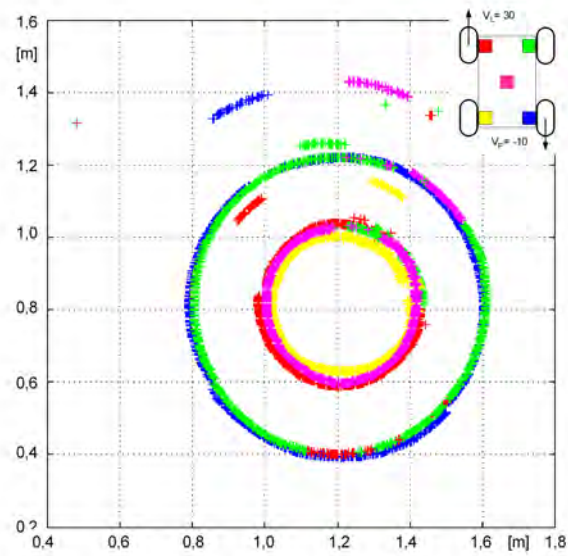
Obrázek 12.1: Měření polohy barevných štítků umístěných na robotu



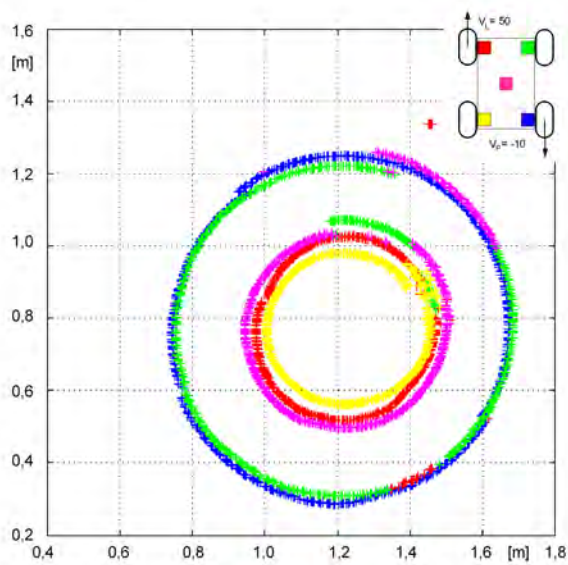
Obrázek 12.2: Měření polohy barevných štítků umístěných na robotu



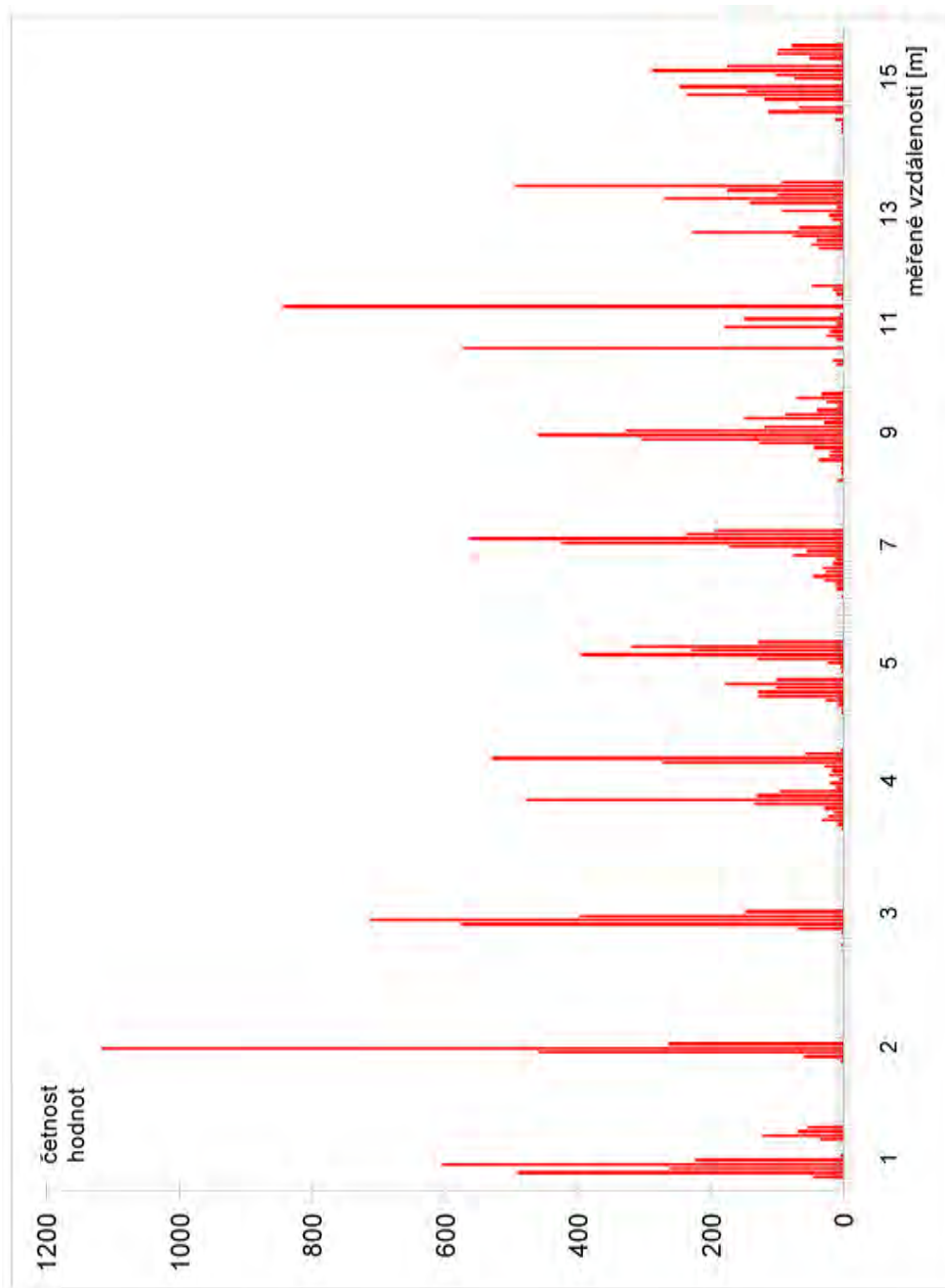
Obrázek 12.3: Měření polohy barevných štítků umístěných na robotu



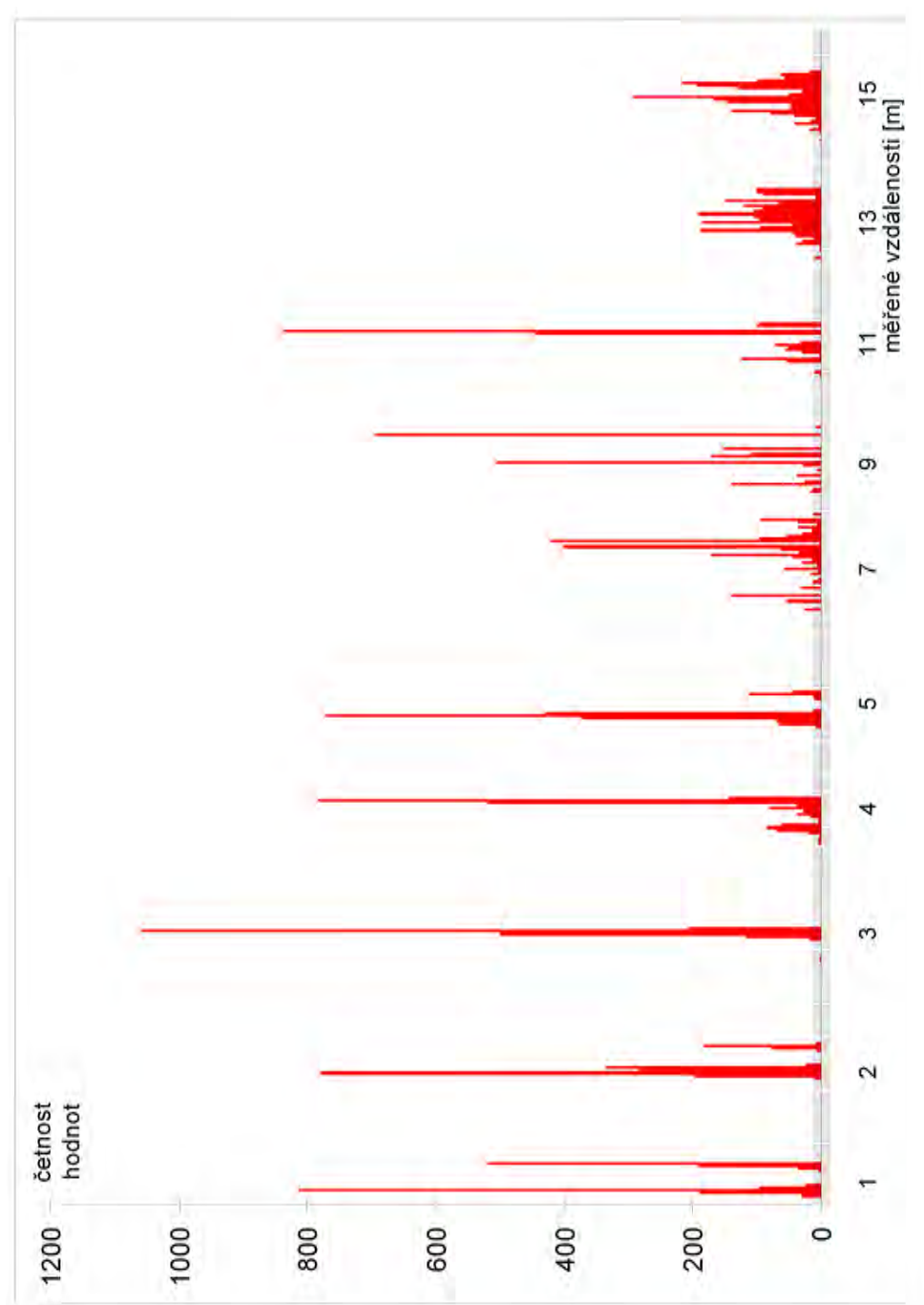
Obrázek 12.4: Měření polohy barevných štítků umístěných na robotu



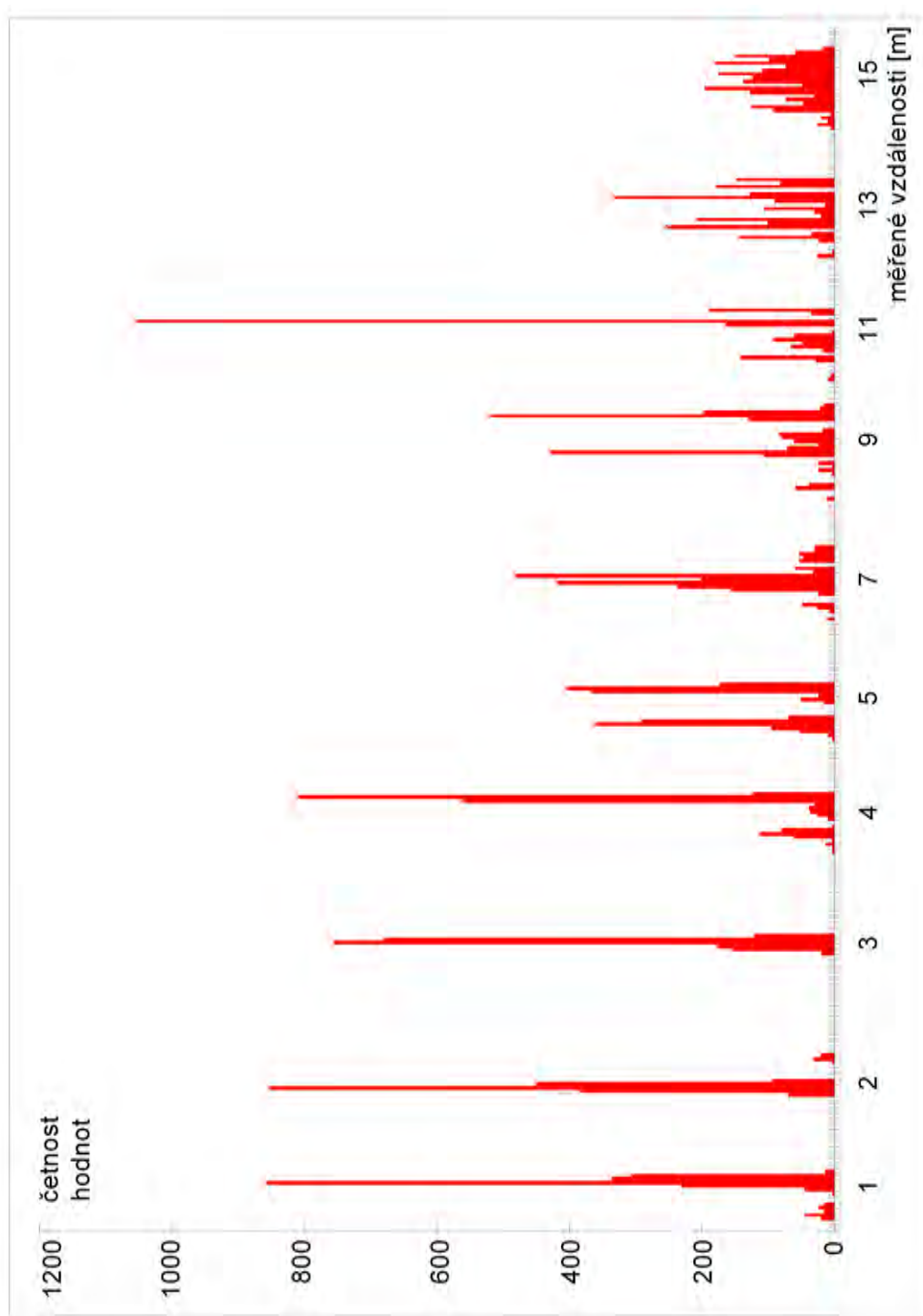
Obrázek 12.5: Měření polohy barevných štítků umístěných na robotu



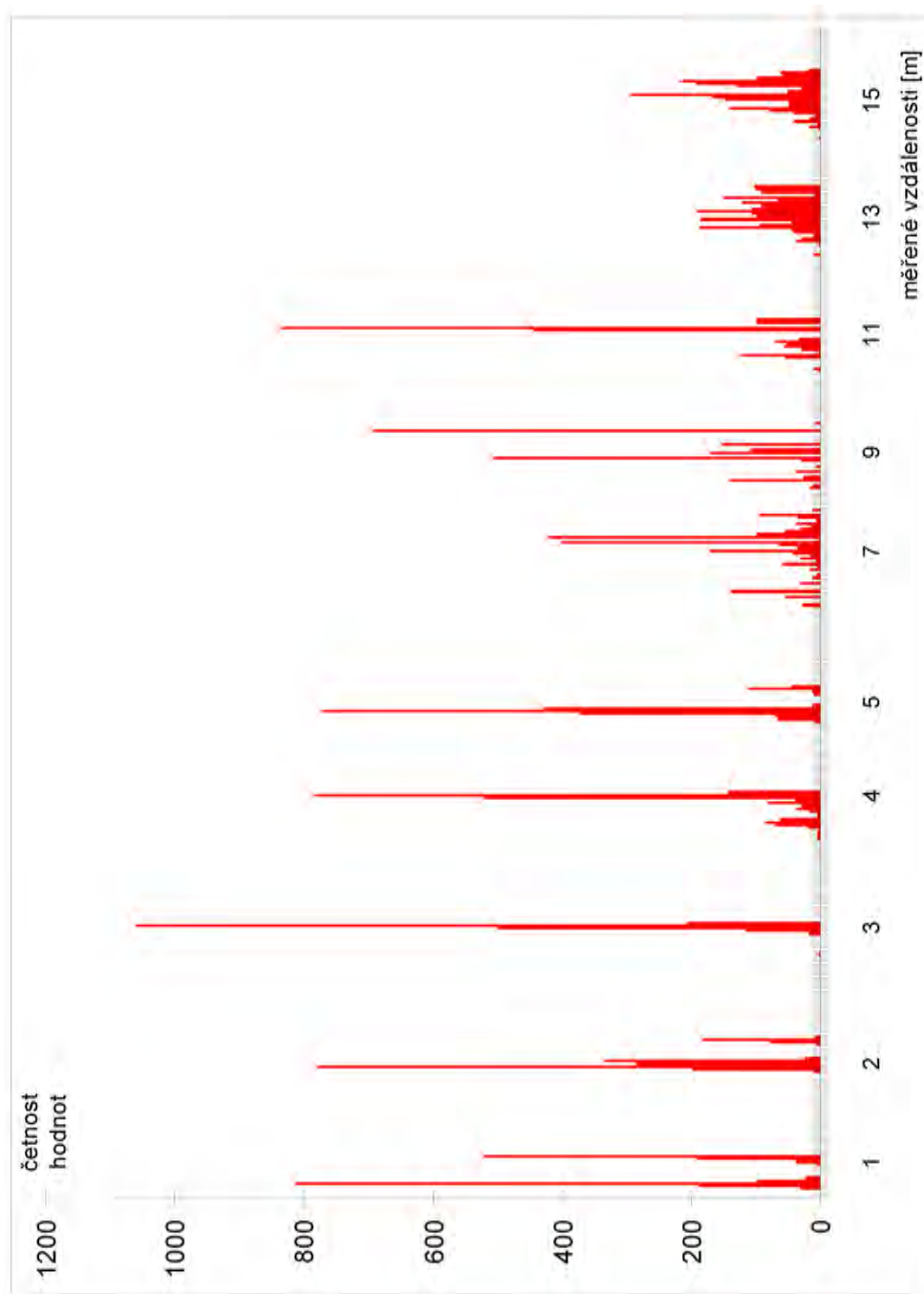
Obrázek 12.6: Tvar rozptylových funkcí pro jednotlivé vzdálenosti, barva černá



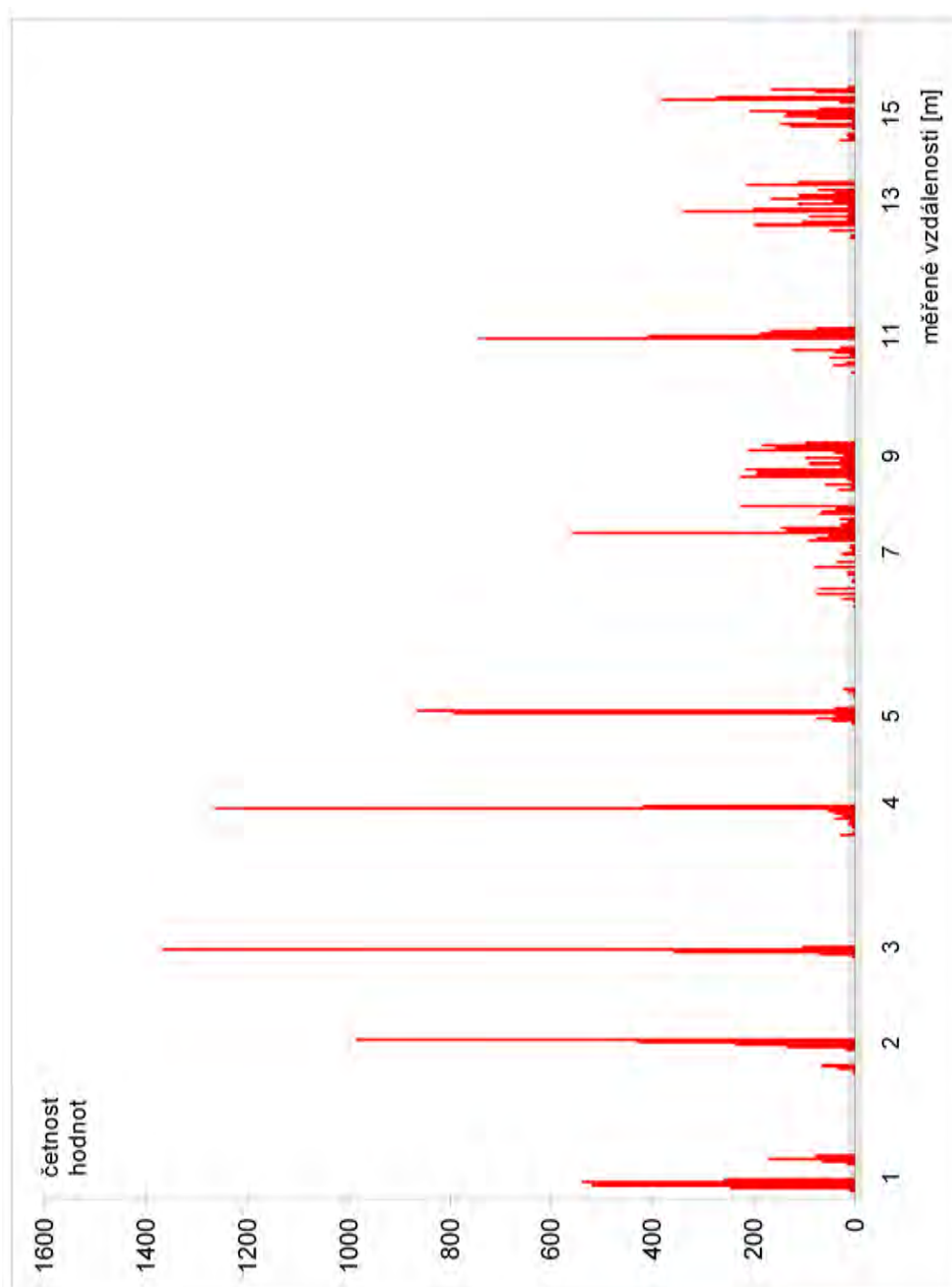
Obrázek 12.7: Tvar rozptylových funkcí pro jednotlivé vzdálenosti, barva šedá2 (viz obrázek 8.8)



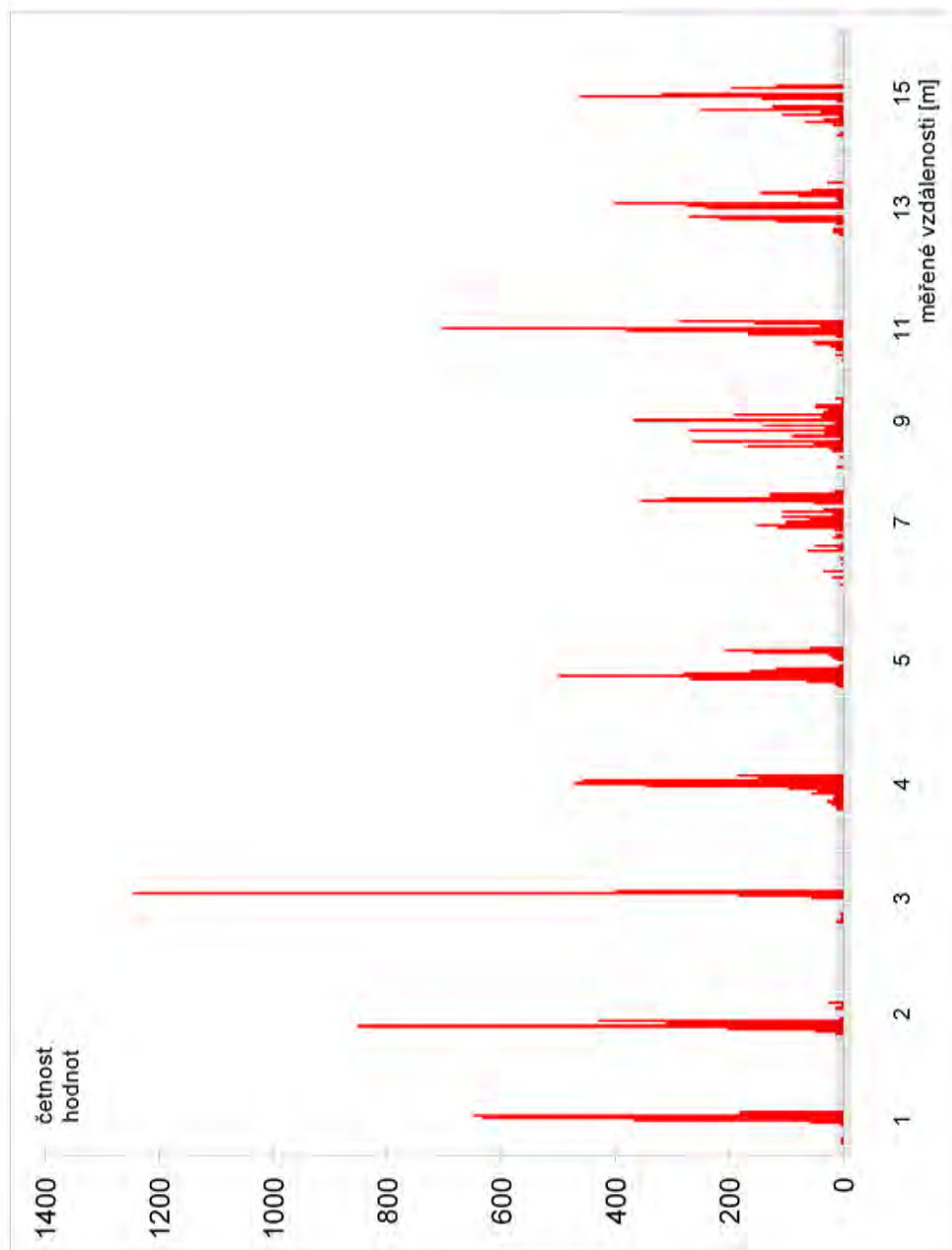
Obrázek 12.8: Tvar rozptylových funkcí pro jednotlivé vzdálenosti, barva šedá3 (viz obrázek 8.8)



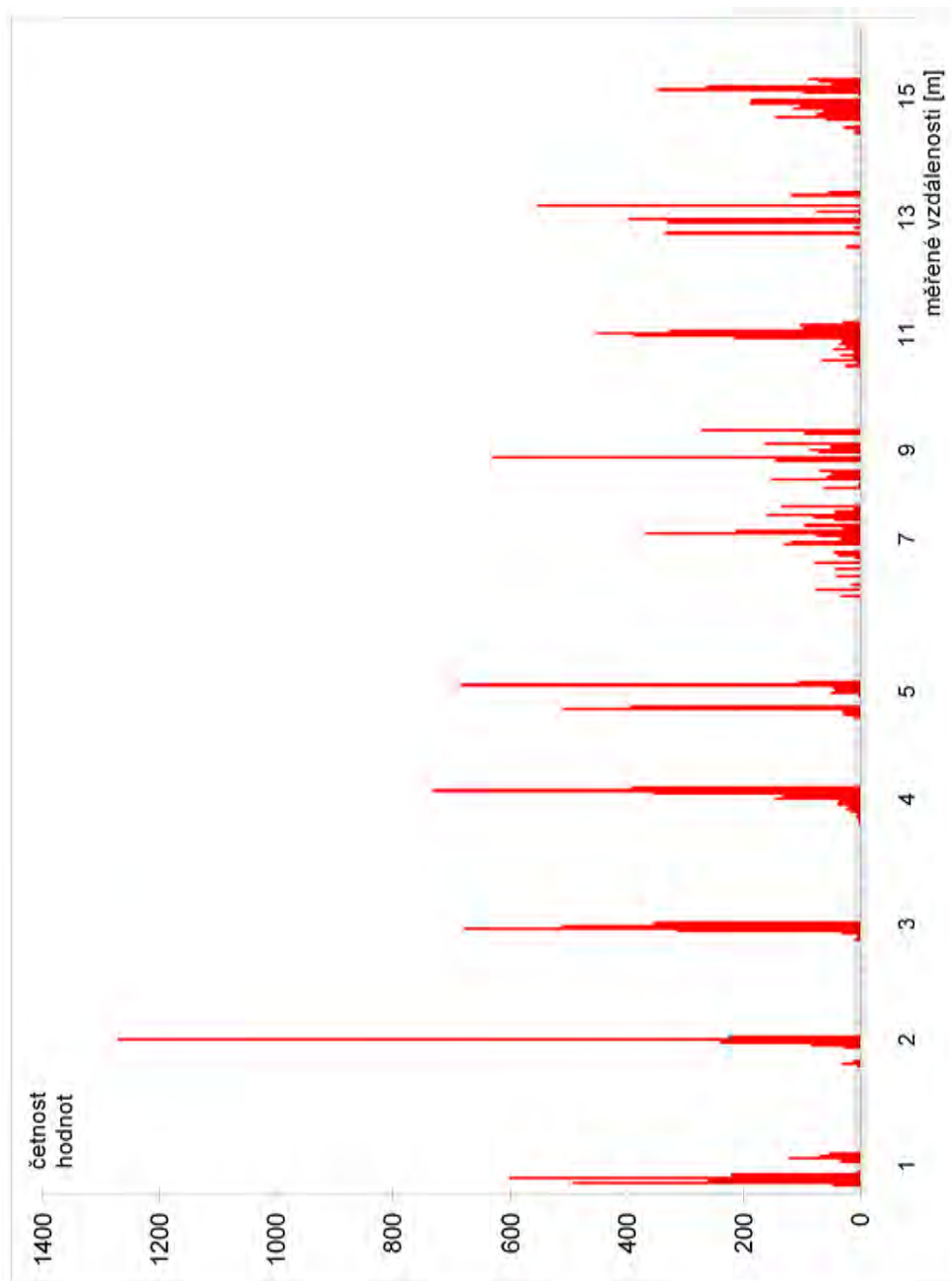
Obrázek 12.9: Tvar rozptylových funkcí pro jednotlivé vzdálenosti, barva - šedá (viz obrázek 8.8)



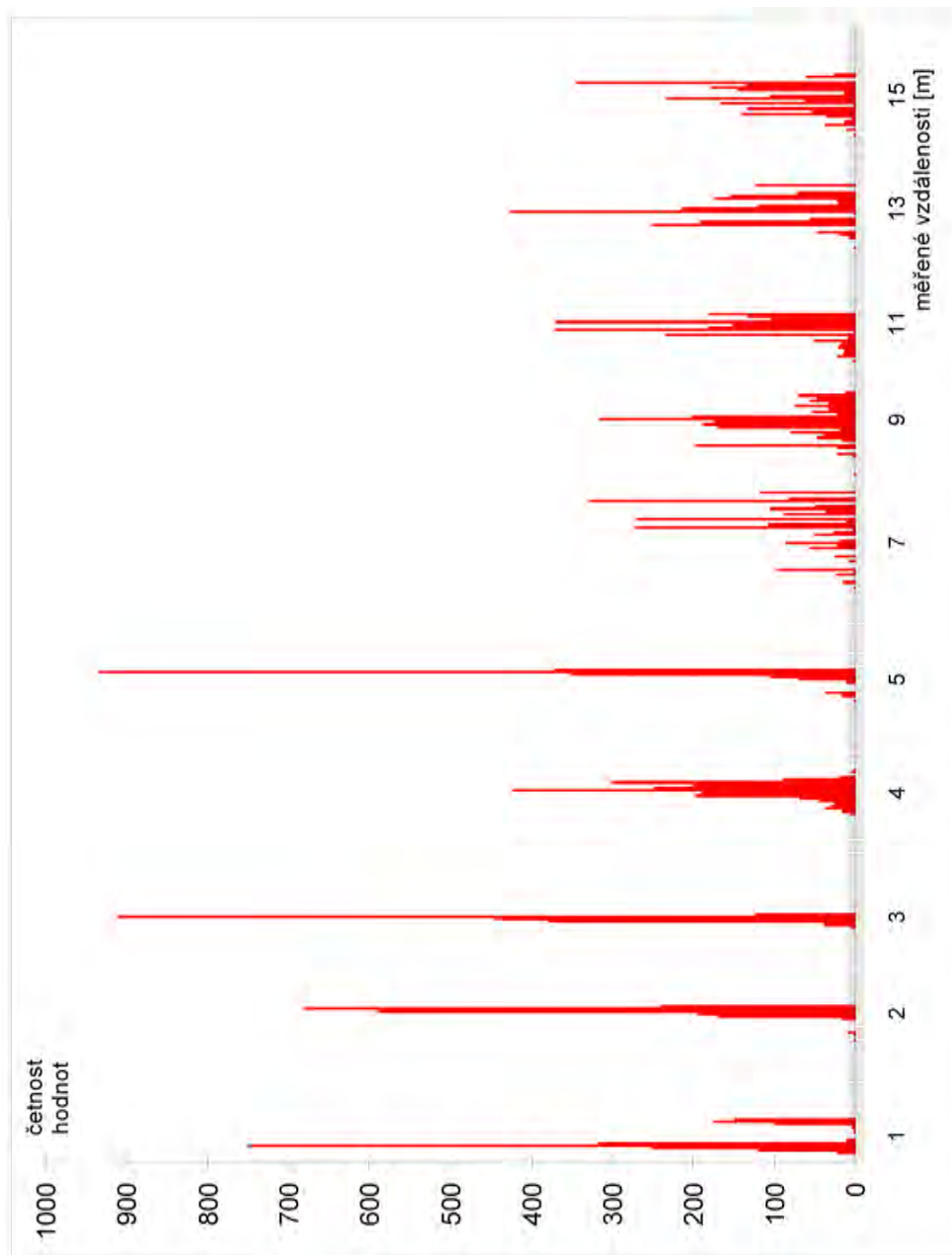
Obrázek 12.10: Tvar rozptylových funkcí pro jednotlivé vzdálenosti, barva - bílá



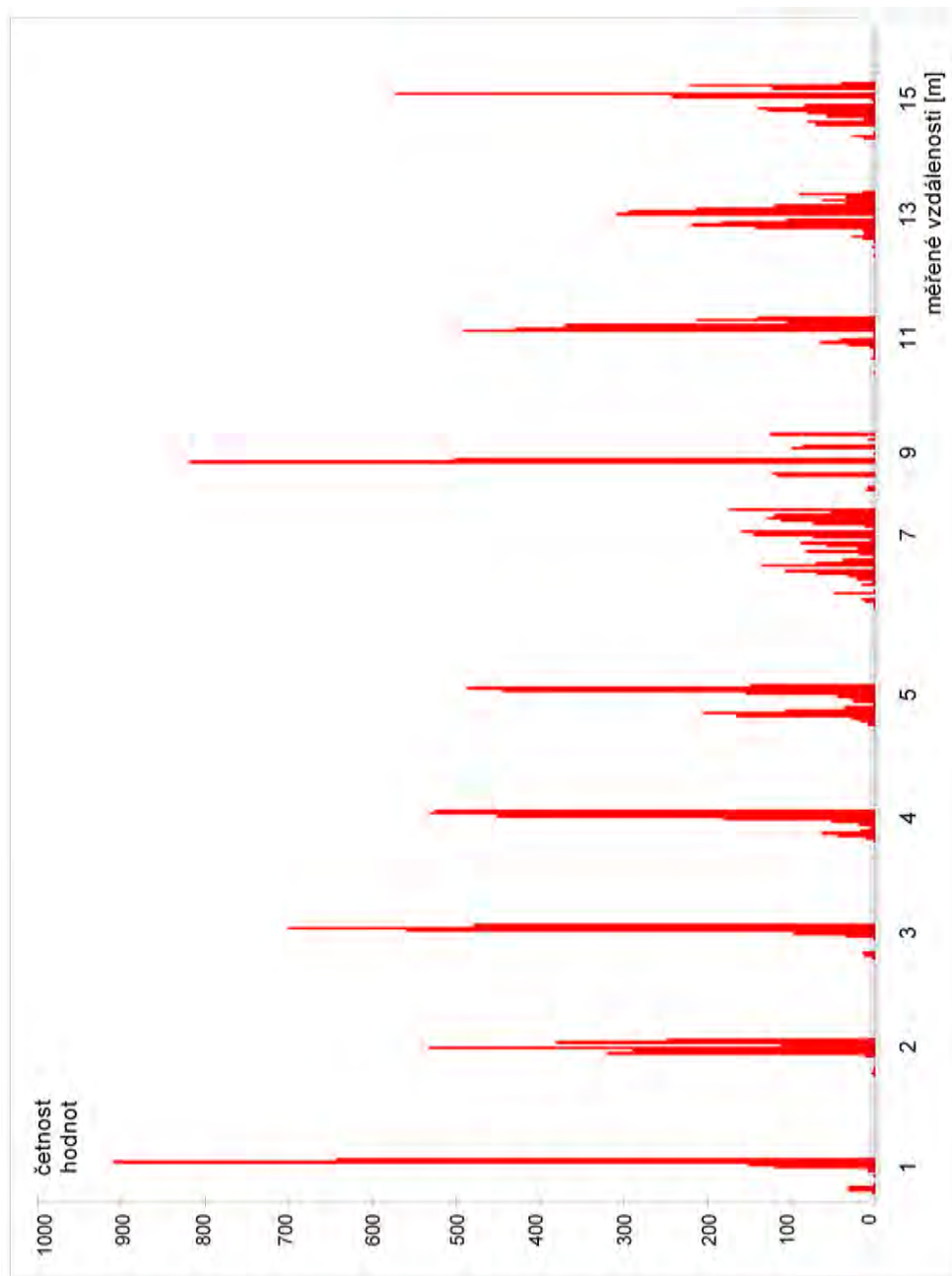
Obrázek 12.11: Tvar rozptylových funkcí pro jednotlivé vzdálenosti, barva - červená



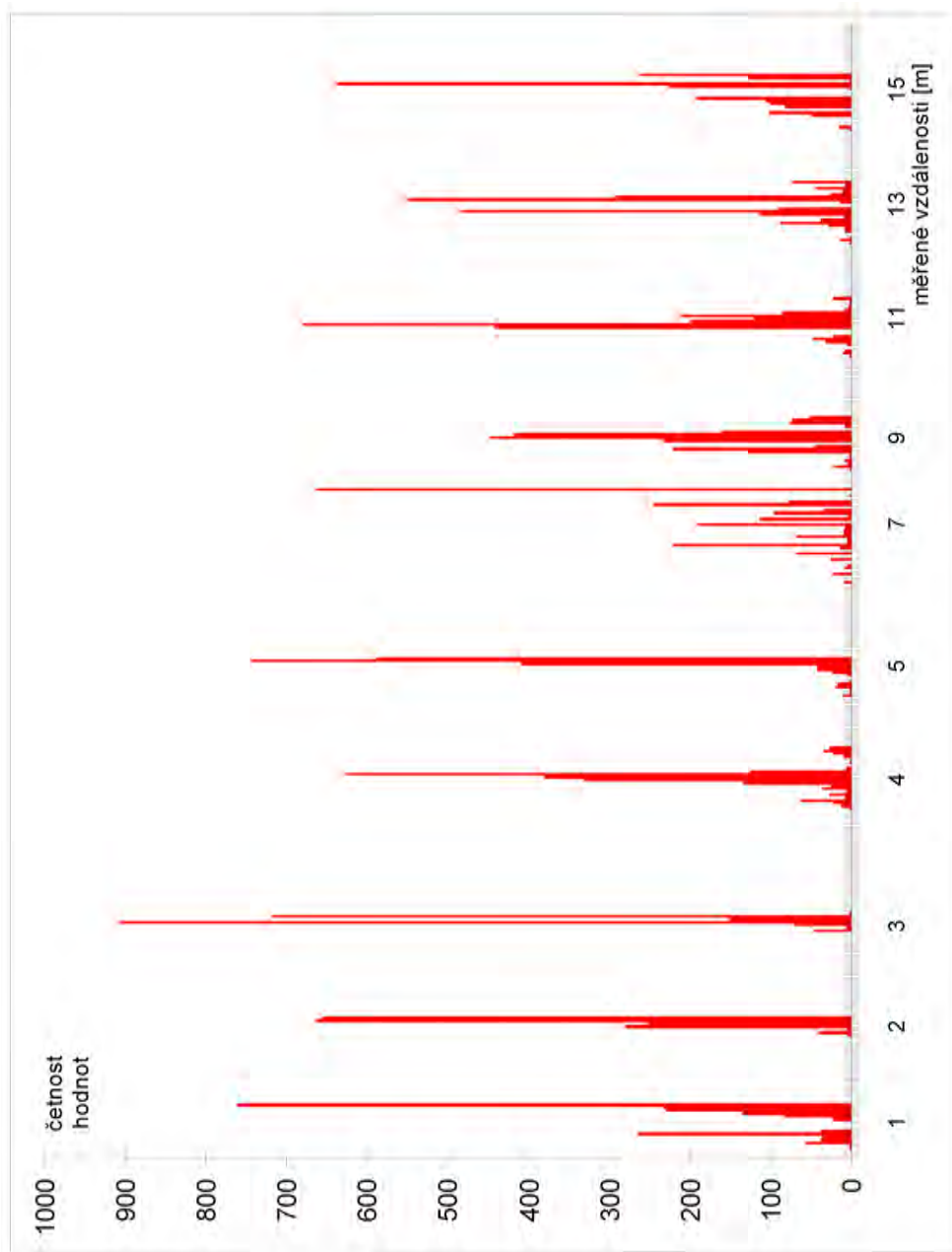
Obrázek 12.12: Tvar rozptylových funkcí pro jednotlivé vzdálenosti, barva - fialová



Obrázek 12.13: Tvar rozptylových funkcí pro jednotlivé vzdálenosti, barva - modrá



Obrázek 12.14: Tvar rozptylových funkcí pro jednotlivé vzdálenosti, barva - žlutá



Obrázek 12.15: Tvar rozptylových funkcí pro jednotlivé vzdálenosti, barva - zelená