

Projekt 4

Lukáš Kuhajda

Akademický rok 2017/2018

Obsah

1	Simultaneous localization and mapping: part I.	2
1.1	Úvod	2
1.2	Historie	2
1.3	Formulace a struktura SLAM	2
1.3.1	Legenda	3
1.3.2	Pravděpodobnostní SLAM	3
1.3.3	Struktura pravděpodobnostního SLAMu	3
1.4	Řešení problému SLAM	4
1.4.1	EKF-SLAM	4
1.5	Rao-Blackweillizedův filtr	5
2	Simultaneous localization and mapping: part II.	6
2.1	Výpočetní složitost	6
2.1.1	Rozšířený stav	6
2.1.2	Rozdělené aktualizace	7
2.1.3	Rozčlenění	7
2.1.4	Globální submapy	8
2.1.5	Submapy vztažné	8
2.2	Sdružování dat	8
2.2.1	Validace várky dat	8
2.2.2	Popis vzhledu	8
2.2.3	Multihypoziční sdružování dat	8
2.3	Reprezentace prostředí	9
2.3.1	Částečná pozorovatelnost a zpožděné mapování	9
2.3.2	3D SLAM	9
2.3.3	Trajektorií orientovaný SLAM	9
2.3.4	Vložené pomocné informace	9
2.3.5	Dynamická prostředí	9

1 Simultaneous localization and mapping: part I.

1.1 Úvod

Simultaneous localization and mapping, v překladu simultánní lokalizace a mapování (SLAM) je problém, při kterém řešíme otázku, zda-li dokáže robot, umístěný do neznámého prostředí a do neznámé lokace, postupně tvořit mapu svého okolí, zatímco určuje, kde se právě nachází. Řešení tohoto problému je nezbytně nutné pro tvorbu zcela autonomně se pohybujících robotů.

1.2 Historie

Jako počátek, kdy se poprvé SLAM objevil, se bere konference Robotics and Automation Conference, která se konala roku 1986 v San Francisco, California, USA, kdy byly pravděpodobnostní metody v začátcích, jak v robotice, tak v umělé inteligenci. Při této příležitosti se uskutečnila debata mezi odborníky v oblasti, z níž vzešlo, že se jedná o problém s mnoha koncepčními a výpočetními otázkami.

K většímu posunu se dostalo o pár let později, kdy vyšla práce pojednávající o vztahu mezi landmarky (orientačními body) a zmenšení geometrické nepřesnosti. Důležitým prvkem zde bylo, že musí být velký stupeň korelace mezi odhady landmarků na mapě, přičemž korelace roste s postupným měřením. O tento posun se zasloužili pánové Smith, Cheesman a Duran-Whyte.

Ve stejném období pracovali Nicholas Ayache a Olivier Faugeras pracovali na základech vizuální navigace a Crowley, Chatila a Laumond na navigaci pracující se sonarem používající Kalmanův filtr. Práce vznikly krátce po té Smithově a byly v základu dosti podobné. Tyto práce ukazovaly, že jak s robot pohybuje v prostoru sbírajíc relativní odhady landmarků, jsou tyto odhady ve vztahu k ostatním kvůli chybě v odhadu pozice robota. Důsledkem bylo zjištění, že pro získání správného řešení, je potřeba vytvářet stav složený z pozice robota a landmarků. Odhadující zařízení tak bude využívat velký stavový vektor s náročností výpočtu rostoucí v kvadrátu. Následovalo pak období, kdy se výzkumníci zaměřili na zlepšení problému mapování, kdy předpokládali minimalizaci, či úplné vymizení korelace mezi landmarky. Teoretická práce na kombinaci lokalizace a mapování se tak pozastavila a vývoj byl zaměřen odděleně pouze na jednu z částí.

Později, když se poprvé formulovala kombinace lokalizace a mapování jako jeden problém, se zjistilo, že snaha minimalizovat korelaci landmarků byla chybná, naopak bylo poté v zájmu korelaci mezi landmarky co nejvíce zvýšit. Struktura SLAMu, a celkově první použití tohoto akronymu, byla prezentována v roce 1995 na International Symposium of Robotics Research (ISRR). Nezbytná teorie konvergence, s řadou počátečních výsledků, byla vytvořena Michaellem Csorbou, zatímco další týmy výzkumníků, například z MIT, Zaragozy nebo z ACFR ze Sydney, již započali práci na využití SLAMu (též nazývaného jako Concurrent mapping and localization = souběžné mapování a lokalizace (CML)) ve vnitřních, venkovních i podvodních systémech. V roce 1999 na ISRR se odehrálo první zasedání pojednávající přímo o SLAM a byla zde představena práce Sebastiana Thruna, ve které bylo dosaženo stupně konvergence mezi SLAMem založeném na Kalmanově filtru a pravděpodobnostním určením pozice a mapováním. Zájem o tento problém, tedy i pokroky v oblasti, od té doby exponenciálně rostly.

1.3 Formulace a struktura SLAM

Jedná se o proces, při kterém robot vytváří mapu prostředí, v němž se pohybuje a na základě mapy určuje svoji pozici v prostoru. Pro určování trajektorie robota a rozložení landmarků není potřeba znalosti lokace robota. Odhad těchto parametrů probíhá současně.

1.3.1 Legenda

k ... čas

x_k ... stavový vektor popisující lokaci a orientaci robota

u_k ... vektor aplikovaný v čase $k - 1$ pro dostání robota v čase k na pozici x_k

m_i ... vektor popisující pozici i -tého landmarku, která je brána jako neměnná se v čase

z_{ik} ... pozorování lokace i -tého landmarku z robota v čase k

$x_{0:k}$... historie umístění robota

$u_{0:k}$... historie vstupních dat pro pohyb robota

m ... množina landmarků

$z_{0:k}$... množina pozorování landmarků

1.3.2 Pravděpodobnostní SLAM

V pravděpodobnostní formě řešení problému SLAM je zapotřebí výpočet rozložení pravděpodobnosti $P(x_k, m | Z_{0:k}, U_{0:k}, x_0)$ pro každý čas k . Toto rozložení popisuje sdruženou posteriorní hustotu landmarků, stavu vozidla s ohledem na zaznamenané pozorování, řídicí vstupy a zahrnuje čas k s počátečním stavem vozidla. Obecně je žádoucí rekurzivní řešení, v tomto případě je pro výpočet využita Bayesův teorém, to má za následek, že je potřeba, aby přechodový stavový model a model pozorování navzájem popisovaly vliv vstupního řízení a pozorování.

Model pozorování popisuje pravděpodobnost tvorby pozorování z_k , když je známá poloha vozidla a landmarků. Tato pravděpodobnost je popsána $P(z_k | x_k, m)$. Z toho lze předpokládat, že pokud je definována lokace vozidla a mapa, tak jsou pozorování, z hlediska mapy a momentálního stavu vozidla, podmíněně nezávislá.

Model pohybu vozidla může být popsán jako stavový přechodový model $P(x_k | x_{k-1}, u_k)$. To znamená, že předpokládáme přechodový stav jako Markovův proces, při kterém následující stav x_k je závislý pouze na předchozím stavu x_{k-1} a aplikovaném řízení u_k a není tak závislý ani na mapě, ani na pozorování.

Nyní je SLAM implementovaný ve dvoustupňové rekurzivní formě korekce předpovědí. Aktualizace času:

$$P(x_k, m | Z_{0:k-1}, U_{0:k}, x_0) = \int P(x_k | x_{k-1}, u_k) \times P(x_{k-1}, m | Z_{0:k-1}, U_{0:k-1}, x_0) dx_{k-1}$$

Aktualizace měření:

$$P(x_k, m | Z_{0:k}, U_{0:k}, x_0) = \frac{P(z_k | x_k, m) P(x_k, m | Z_{0:k-1}, U_{0:k}, x_0)}{P(z_k | Z_{0:k-1}, U_{0:k})}$$

1.3.3 Struktura pravděpodobnostního SLAMu

Model pozorování $P(z_k | x_k, m)$ udává závislost pozorování polohy vozidla a landmarků, což má za následek, že sdružená posteriorní hustota nemůže být klasicky rozdělena stylem

$P(x_k, m | z_k) \neq P(x_k | z_k) P(m | z_k)$, toto rozdělení právě vede k chybným odhadům. Je také dáno, že spousta rozdílných odhadů od skutečné polohy landmarků je způsobena jediným zdrojem, čímž je chyba v odhadu, kde se robot v době pozorování nachází. Z toho vyplývá, že landmarky mezi sebou mají velkou korelaci a vzájemná odhadovaná pozice mezi 2 landmarky je tedy velice přesná i přes ne zcela přesný odhad jednoho z nich. V pravděpodobnostní formě to tedy znamená, že sdružená pravděpodobnostní hustota pro pár landmarků $P(m_i, m_j)$ je téměř zcela přesně dána, přičemž odhad bodu $P(m_i)$ nemusí být úplně přesný.

Nejdůležitějším poznatkem bylo zjištění, že korelace mezi landmarky monotónně vzrůstá s počtem jejich pozorování, toto zjištění však bylo učiněno pouze pro lineární Gausův případ. Znamená to tedy, že odhad pozice všech landmarků $P(m)$ se bude stávat s narůstajícím počtem pozorování

monotónně přesnější. Tento jev nastává díky, v podstatě, skoro nezávislému měření relativních pozic mezi landmarky, které jsou zcela nezávislé na natočení vozidla a úspěšné pozorování z tohoto bodu může nést další nezávislá měření relativních rozložení landmarků.

Jak se robot pohybuje na další pozici x_{k+1} , opět zaznamenává lokaci landmarku m_j , díky které může určit pozici svoji a landmarku vůči své předchozí pozici x_k . Dále je pak aktualizována, díky vysoké korelaci mezi landmarky, pozice landmarku m_i i v případě kdy již není vidět. V případě, že robot v bodě x_{k+1} zpozoruje další dva landmarky, které budou vzhledem k m_j dosud neznámé, jsou ihned spojeny nebo korelovány se zbytkem mapy. Pozdější jejich aktualizace tak budou ovlivňovat landmark m_j , tím pádem i m_i a vytváří se tak síť navzájem propojených bodů, jejichž přesnost se zvyšuje s dalším pozorováním. Tento proces by se dal vizualizovat jako síť, propojující mezi sebou všechny landmarky, které jsou spojeny provázkem, jenž se dle vzájemné vzdálenosti a počtu měření zpevňuje. Opětovným projížděním prostorem se tedy tato síť stává robustnější a přesnější, omezením však stále zůstává kvalita mapy a měřicího senzoru.

1.4 Řešení problému SLAM

Při řešení je potřeba adekvátně obsáhnout jak složku modelace prostředí, tak i tvorbu pohybového modelu. Nejčastěji se setkáváme s reprezentací problému ve formě stavového modelu zatíženého šumem, což vede k použití rozšířeného Kalmanova filtru (v originále *extended Kalman filter* → *EKF*). Jinou možností je rozčlenit pohybový model vozidla na vzorky s obecnějším negausovským rozdělením prtavděpodobnosti, což vede k použití Rao-Blackwellizedova partikulárního filtru.

1.4.1 EKF-SLAM

Popis pohybu vozidla:

$$P(x_k|x_{k-1}, u_k) \leftrightarrow x_k = f(x_{k-1}, u_k) + w_k$$

$f(\cdot)$... funkce modelující pohyb vozidla

w_k ... chyby měření

Q_k ... kovariance

Model pozorování:

$$P(z_k|x_k, m) \leftrightarrow z_k = h(x_k, m) + v_k$$

$h(\cdot)$... geometrické vlastnosti

v_k ... chyby měření

R_k ... kovariance

Tyto dvě definice využijeme v EKF metodě k výpočtu průměru a kovariance sdruženě posteriorního rozložení

průměr:

$$\begin{bmatrix} \hat{x}_{k|k} \\ \hat{m}_k \end{bmatrix} = E \begin{bmatrix} x_k | Z_{0:k} \\ m \end{bmatrix}$$

kovariance:

$$P_{k|k} = \begin{bmatrix} P_{xx} & P_{xm} \\ P_{xm}^T & P_{mm} \end{bmatrix}_{k|k} = E \left[\begin{pmatrix} x_k - \hat{x}_k \\ m - \hat{m}_k \end{pmatrix} \begin{pmatrix} x_k - \hat{x}_k \\ m - \hat{m}_k \end{pmatrix}^T \middle| Z_{0:k} \right]$$

Mezi hlavní problémy této metody se řadí konvergence, výpočetní složitost, sdružování dat a nelinearita. Prvním z nich, konvergence, se projevuje postupným přechodem determinantu mapové kovarianční matice $P_{mm,k}$ a podtříd dvojic landmarků k nule, jednotlivé odchylky landmarků pak konvergují dle původních nepřesností v odhadu pozice robota a jeho pozorování. Výpočetní složitost je brána v tuto

chvíli jako kvadraticky rostoucí s počtem zaznamenaných landmarků, neboť při každém zaznamenaném pozorování se aktualizují již uložené landmarky, tento problém již ale prošel vývojem a existují již metody pracující v reálném čase s tisíci landmarky. Metoda EKF-SLAM je velmi náchylná na chybné spojení pozorování s landmarky. Jedná se zejména o problém s uzavřením smyčky, kdy dochází k opětovnému návratu na místo, ze kterého robot začínal. Nelinearita je posledním z významných problémů, kvůli nelinearitě můžeme dojít k větším nepřesnostem ve výsledku, neboť EKF-SLAM využívá lineárních modelů pro vyjádření nelineárního pohybu a modelu pozorování, konvergence a konzistence modelu je tedy jistá pouze v lineárním případě.

1.5 Rao-Blackwellizedův filtr

Rao-Blackwellizedův filtr, jinak také FastSLAM, je zásadním posunem v návrhu rekuzivního pravděpodobnostního SLAMu. Předchozí pokusy pouze modifikovali EKF-SLAM, FastSLAM, na bázi rekuzivního Monte Carlo modelu, je prvním, který dokáže reprezentovat nelineární stavový model. Je výpočetně nemožné aplikovat na stavový prostor s vysokým počtem dimenzí partikulární filtry, je však možné redukovat velikost vzorků.

Sdružený stav SLAMu může být jako faktor komponentů vozidla a podmíněných komponentů mapy:

$$P(X_{0:k}, m | Z_{0:k}, U_{0:k}, x_0) = P(m | X_{0:k}, Z_{0:k}) P(X_{0:k} | Z_{0:k}, U_{0:k}, x_0)$$

Rozdělení pravděpodobnosti zde není na jednotlivých pozicích x_k , ale na celou trajektorii $X_{0:k}$ a tím se stávají jednotlivé landmarky na sobě nezávislými, mapa je tedy reprezentována jako soubor nezávislých gaussiánů, což znamená lineární složitost oproti kvadratické u čistého EKF-SLAMu. Hlavními ukazateli FastSLAMu je mapa, jež se vypočítává analyticky a vážené vzorky, kterými je reprezentována trajektorie pohybu. Rekuzivní odhad je proveden partikulárním filtrem pro stav pozice a EKF pro stav mapy.

V této metodě se každý landmark zpracovává zvlášť, jeho pozice se aktualizuje stejným způsobem jako v EKF a landmarky, které nebyly zpozorovány, zůstávají na původní pozici a neaktualizují se. Vzájemnou neprovázanost landmarků však vzniká chyba v odhadu, která s časem roste, neboť jak se landmarky neovlivňují, nemohou tak aktualizací sítě landmarků napomoci k odhadu pozice robota a chyba odhadu jeho pozice tak stále roste.

Předpokládáme, pro obecnou formu FastSLAMu, že v čase $k - 1$, je sdružený stav reprezentovaný jako $\{w_{k-1}^{(i)}, X_{0:k-1}^{(i)}, P(m | X_{0:k-1}^{(i)}, Z_{0:k-1})\}_i^V$. V prvním kroce, pro každou částici vypočítáme návrh distribuce, jež je podmíněná svojí specifickou historií a z ní odebereme vzorek x_k , který je poté sdružen k historii částice $X_{0:k}^{(i)}$. Zahrneme zde i rozdíl mezi dvěmi možnými verzemi FastSLAM, jsou to FastSLAM 1.0 a FastSLAM 2.0. V tomto bodě se liší pouze formou navrhování distribuce. Krokem dva, dle funkce důležitosti, stanovíme váhy vzorků, v tomto případě je rozdíl verzí v důležitosti váhy pro další výpočty. Třetím krokem je případné převzorkování, které se provádí různě často dle implementace. Krokem posledním je provést EKF update, na každou, již zpozorovanou částici, která je při aktuálním pozorování zaznamenána.

V dalším porovnání verzí FastSLAMu, verze 2.0 dává pro každou částici nejmenší možnou odchylku váhové důležitosti, vyplývá to z toho, že její návrh distribuce je lokálně optimální. Problém obou verzí je však neschopnost zapomínat minulost, dochází tak ke ztrátě statistické přesnosti, díky převzorkování, které přemazává historii měření, na níž je spolu s pozicí závislá tvorba mapy.

2 Simultaneous localization and mapping: part II.

2.1 Výpočetní složitost

Daný problém SLAM má neobvyklou strukturu, jelikož je odhad sdruženého stavu složený z pozice robota a landmarků, procesní model má tak vliv pouze na stav pozice robota a model pozorování na pár robot-landmark, a to vedlo k vytvoření spousty metod využívajících zmíněnou strukturu, pro redukci výpočetní složitosti. Základním rozdělením metod redukujících výpočetní složitost, je rozlišování optimálních, konzervativních a nekonzistentních metod. První typ, optimální metody, jsou založené na redukci daného výpočtu, výsledkem jsou pak odhady a kovariance, stejně tak, jako je tomu v případě plnohodnotné formy SLAM, rozebírané v předchozích kapitolách. U metod konzervativních dochází k odhadům s vyšší neurčitostí nebo kovariancím, většinou ale, i přes větší nepřesnost, jsou implementovány v reálném použití. Poslední možností jsou nekonzistentní metody. Jsou to algoritmy, které mají nižší neurčitost nebo kovarianci, než algoritmy optimální. Pro řešení SLAM se však nepoužívají.

Prvním přístupem pro redukci výpočetní složitosti, je využití struktury SLAM, pro omezení požadovaného výpočtu rovnicí aktualizace pozorování. Výpočet aktualizace času může být omezen metodami využívající rozšířený stav, výpočet stavu pozorování pak metodami oddělovacími rovnice dané aktualizace a obě tyto omezení vedou k redukci výpočtů, typicky jsou to optimální algoritmy. Další možností, jak snížit složitost procesu, je reformulace stavového prostoru do informační podoby, která umožňuje rozdělení výsledné matice s informacemi pro snížení výpočtů, což obvykle bývají algoritmy konzervativní. Obvykle je v díky nim znatelně redukována výpočetní složitost, stále však má dostatečně dobré odhadovací schopnosti. Dalším přístupem je submapping, který rozděluje mapu na regiony, kdy následné aktualizace můžou nastávat pouze v dané oblasti a s určitou periodou pak v i rámci ostatních oblastí.

2.1.1 Rozšířený stav

Sdružený stavový vektor x_k , v čase k , se skládá z pozice robota a jím zaznamenaných landmarků, kdy model robota ovlivňuje pouze stav pozice, a to vlivem vstupního řízení, stav mapy se tím tedy nemění.

$$x_k = \begin{bmatrix} f_v(x_{vk-1}, u_k) \\ m \end{bmatrix}$$

Pokud při volbě typu SLAM jako EKF, má výpočet předpovědi kovariance kubicky rostoucí složitost s počtem landmarků

$$P_{k|k-1} = \nabla f_x P_{k-1|k-1} \nabla f_x^T + \nabla f_u U_k \nabla f_u^T,$$

to se však dá předělat na formu s pouze lineární složitostí

$$P_{k|k-1} = \begin{bmatrix} \nabla f_{vx} P_{vv} \nabla f_{vx}^T + \nabla f_{vu} U_k \nabla f_{vu}^T & \nabla f_{vx} P_{vm} \\ P_{vm}^T \nabla f_{vx}^T & P_{mm} \end{bmatrix}$$

Přidání nového landmarku má podobný tvar, kdy je nový landmark inicializován jako funkce pozice robota a pozorování a rozšířený stav pak získáme z malého množství existujících stavů

$$x_k^+ = \begin{bmatrix} x_{vk} \\ m \\ g(x_{vk}, z_k) \end{bmatrix},$$

kde $g(x_{vk}, z_k) = m_{new} \rightarrow$ přidání nového landmarku

Rozšíření stavu můžeme aplikovat vždy, když je nový stav funkcí podmnožiny již existujících stavů

$$\begin{bmatrix} x_1 \\ x_2 \\ f(x_2, q) \end{bmatrix},$$

$$\begin{bmatrix} P_{11} & P_{13} & P_{13} \nabla f_{x_2}^T \\ P_{12}^T & P_{23} & P_{23} \nabla f_{x_2}^T \\ \nabla f_{x_2} P_{11}^T & \nabla f_{x_2} P_{32} & \nabla f_{x_2} P_{32} \nabla f_{x_2}^T + \nabla f_q Q \nabla f_q^T \end{bmatrix}$$

2.1.2 Rozdělené aktualizace

Jedná se o metody vytvářející optimální odhady. Při implementaci základní podoby aktualizace pozorování, se při každém novém měření aktualizuje jak stav vozidla, tak i mapy, což vede ke kvadratickému nárůstu složitosti s množstvím landmarků. Při této metodě si však rozdělíme mapu na menší oblasti, které se aktualizují při průjezdu robota danou částí, zatímco aktualizace celé mapy probíhá s výrazně nižší frekvencí.

Rozlišujeme dva způsoby možné implementace. První z nich pracuje na zmenšené oblasti, ale stále si drží globální referenční souřadnice, jedná se například o algoritmus CEKF (compressed EKF). Druhou možností je tvorba menších map s vlastním souřadnicovým rámcem neopouštějícím danou submapu, jedná se o algoritmy CLSF (constrained local submap filter \rightarrow omezený lokální submapový filtr). Pokračovat budeme v rozboru druhé možnosti, neboť je jednodušší a při provádění operací s velkou frekvencí opakování je méně ovlivněna linearizačními chybami, je stabilnější a zabráňuje příliš velkému nárůstu globální kovariance.

Logaritmus submapy se skládá z dvou nezávislých odhadů, které si stále udržuje. Jde o vektory x_G a x_R , kdy x_G je mapa složená z globálně referencovaných landmarků a globálně referencované pozice dané submapy a x_R je lokální submapa s lokálně referencovanou pozicí robota lokálně referencovanými landmarky. Při získání pozorování se aktualizují pouze landmarky náležící aktuální submapě, ve které se robot nachází. Celkový globální odhad pak získáváme periodicky, zaevidováním submapy do mapy celé a použitím aktualizace omezení na společné vlastnosti obou map.

2.1.3 Rozčlenění

Do této doby jsme brali stavový odhad \hat{x}_k a matici kovariance P_k , jako produkty klasicky implementovaného EKF-SLAM, ty pak spolu popisovaly první dva centrální momenty hustoty Gaussovy pravděpodobnosti skutečného stavu x_k . Jinou možností je vyjádřit stavový odhad a matici kovariance v informační formě pomocí matice informací $Y_k = P_k^{-1}$ a vektoru informací $\hat{y}_k = Y_k \hat{x}_k$. Tato změna je výhodná pro mapy s větším měřítkem, kdy spousta nediagonálních prvků bude velmi blízkých nule, což vede k možnosti nastavení těchto prvků na hodnotu nula, přičemž však může vznikat malá ztráta optimality při vzniku map.

Důležitým poznatkem je fakt, že rozšíření stavu je rozčleňovací operace vedoucí k přesnému rozčlenění informační formy SLAMu a má tak ekvivalentní informační formu, kde pro zjednodušení považujeme šum za nulový přídavek $f(x_2, q) = f(x_2) + q$. Podmnožinu stavů x_1 je pak předpokládána jako obsahující většinu stavů mapy a po rozčlenění, dosahuje pouze konstantní složitosti v čase. Můžeme tedy získávat přesné řešení rozšiřováním stavu novým odhadem pozice robota v každém kroce a zachovat všechny předchozí pozice. Díky tomu jsou nenulové nediagonální prvky pouze ty, které jsou spojené napřímo s měřeními daty.

Dále sem musíme zahrnout marginalizaci, jež je nezbytná pro odstranění předchozích stavů pozice. Máme možnost marginalizovat všechny předchozí stavy, což vede na zhuštěnou matici informací, to je stav, kterého dosáhnout nechceme. Správnou vlbou ukotvení pozice můžeme marginalizovat velkou část pozic, aniž bychom vyvolaly nadměrnou hustotu matice informací. Pro reálné využití v praxi se zde stále vyskytuje problém s obnovováním průměru a kovariance v každém kroce, což může být velmi výpočetně náročné, dá se však poměrně efektivně získat konjugacími gradientními metodami.

2.1.4 Globální submapy

U globálních metod rozlišujeme základní dva typy, globálně a lokálně referencované, přičemž oba mají společné, že submapa stanovuje místní souřadnicový rámec a landmarky z jejího okolí jsou odhadovány s ohledem na daný místní rámec.

Lokální metoda získává odhady pomocí optimálního algoritmu SLAM, používajícího pouze lokální landmarky. Tato metoda, i přes výpočetní efektivitu, je při tvorbě struktury submap neoptimální. Globální metoda je mnohem zajímavější. Dokáže z kvadraticky rostoucí složitosti, udělat lineární, či dokonce v čase konstantní složitost. Je to možné díky údržbě a konzervativním odhadům celkové mapy. Metoda stojí na odhadování globální pozice submapy v rámci společného základního rámu, nevede však ke zmírnění problémů s linearizací, způsobenou velkými nepřesnostmi v pozicích.

2.1.5 Submapy vztažené

Základním rozdílem od metody globálních submap, je absence společného základního rámu. V této metodě se submapy zaznamenávají dle sousedství s ostatními a celkovou mapu pak můžeme získat souhrnem vektoru cestou po vytvořené síti submap. Submapy jsou, díky vyhýbání se globálním spojením, velmi zajímavé z hlediska výpočetní složitosti a problémům týkajících se nelinearity. Velkým kladem je například tvorba lokálně optimální mapy s výpočetní složitostí nezávislou na celkové velikosti mapy a dále, díky úpravám pouze na lokální úrovni, je velmi stabilní.

2.2 Sdružování dat

Jedná se velmi důležitý problém, neboť i když během procesu tvorby mapy, dojde k jedné chybné asociaci dat, může to vést k destabilizaci odhadu mapy, často dokonce k pádu celého algoritmu

2.2.1 Validace várky dat

Z prvu se k problému přistupovalo způsobem, kdy každé jednotlivé zachycení landmarku, se porovnávalo se všemi odhady nacházejícími se v blízkém okolí, tento individuální přístup je neproveditelný, pokud je pozice robota velmi nejistá, což znamená selhání ve všech, obzvlášť ve strukturovaných a málo osídlených prostředích. Později se již podařilo násobné asociace zvažovat simultánně.

2.2.2 Popis vzhledu

Jedním ze způsobů snímání okolí je vidění, kdy zaznamenáváme tvar, barvu, strukturu, a tím dokážeme rozlišovat různé balíčky data, což v SLAMu využíváme pro přepověd dané asociace. Nejčastěji tak pro problém s uzavřením smyčky, ke kterému se dostalo z původního využití, a to rozpoznání míst v topologických mapách a indexování databáze obrázků. Pokrok této metody přišel s prací Newmana a spol., kdy se začala počítat metrika podobnosti přes sekvenci obrazů, místo původního jednoho a pro použití v běžném režimu je použita metoda valstních čísel.

2.2.3 Multihypoziční sdružování dat

Jedná se o metodu nezbytně potřebnou pro robustní sběr cílů v zaplněném prostředí, kdy tvorbou oddělených odhadů trasy jízdy pro každou asociaci hypotézu, řeší problém nejednaznačící hypotéz. Tato funkce je však silně limitována dostupným výpočetním výkonem. Dale je metoda vyžívána, zvláště ve velkých prostředích, v implementaci robustního SLAMu, kdy je při uzavírání smyčky vytvořena hypotéza pro smyčku uzavřenou, tak i pro stále neuzavřenou, a tím se bere v potaz, že je prostředí pouze podobné.

2.3 Reprezentace prostředí

Původně se modeloval svět pomocí SLAM pouze jako soubor landmarků, majících svůj určitý tvar, později však, zejména ve venkovním, podvodním a podzemním použití, se ukázala tato metoda jako nevyhovující.

2.3.1 Částečná pozorovatelnost a zpožděné mapování

Základními dvěma typy pro pozorování je vidění, pomocí kamery, nebo nějaký typ dálkového senzoru. Kamera, pokud je na robotovi samostatně, zaznamenává informace neobsahující přesnou vzdálenost objektu. Tento problém měření pomocí senzoru nemá, neboť měření vzdálenosti bývá velmi přesné, za to se pro tento typ vyskytuje problém se šířkou vysílaného paprsku a postraními žlábků. S tímto senzorem nedokážeme jedním pozorováním ani vytvořit přesně položený landmark, jedním měřením totiž získáváme negaussovské rozložení pro pozici landmarku a potřebujeme tak větší počet jeho pozorování pro vytvoření odhadu.

Generalizované rozložení umožňují nezpožděné sledování landmarků, zpožděním inicializace však můžeme získat hned gaussovský odhad pozice landmarku, je při tom ale potřeba zaznamenávat pozici robota v každém okamžiku měření, což provedeme rozšířením stavu o vektor pozic. Nejedná se pouze o částečnou pozorovatelnost, ale shromažďováním informací a zpožděným rozhodováním, zvyšuje robustnost procesu.

2.3.2 3D SLAM

Jedná se v podstatě pouze o rozšíření 2D SLAMu, které má však výrazně větší výpočetní složitost a komplikovanější modelování, způsobené složitějším snímáním. Rozlišujeme základní tři možnosti. První je klasický 2D SLAM s přidanou schopností vytvářet třetí dimenzi, což se užitečné, pokud se robot pohybuje po rovině. Druhý typ vytváří 3D obraz extrakcí diskrétních landmarků a sdruženého odhadu pozice vozidla a mapy. Využití je vhodné v robotech s jedním senzorem, který umožňuje pohyb se šesti stupni volnosti. Poslední možnost se od těch předchozích dost odlišuje. Sdružený stav se totiž skládá z předchozích pozic robota, na každé pozici se udělá 3D sken prostředí a odhad pozice se vyrovná korelaí skenů.

2.3.3 Trajektorií orientovaný SLAM

Základní formulací problému, je odhadovaný stav jako pozice robota a zaznamenané landmarky, jinou, novější, možností je odhadovat místo toho trajektorii vozidla. Mapa tak není součástí stavu, každá pozice robota má ale přidružený sken svého okolí, které se pak srovnají a vytvoří globální mapu. Z toho je možno vidět jeden z problémů, což je neustálý růst stavového prostoru, který se nijak nepromazává. Využití se nachází například při tvorbě topologickým map.

2.3.4 Vložené pomocné informace

Jak je vidět u trajetoriemi orientovaného SLAMu, můžeme ke stavu připojit data, v tomto případě to dále rozšířit například o teplotu, charakteristiku povrchu a mnoho dalších věcí, které pak napomáhají při tvorbě mapy. Tvorba takovéto struktury je ale poměrně komplikovaná. Pohybem robota prostředím, ukládají se pomocná data do datové struktury tak, že každá buňka v této struktuře je přiřazena daným landmarkům v mapě, při aktualizaci jejich pozice se tak s nimi přemísťují i jim přidružené informace.

2.3.5 Dynamická prostředí

Ve světě se setkáváme převážně z dynamicky se vyvíjejícím prostředím, kdy nám do pozorování můžou zasahovat lidé, zvířata, nebo třeba nábytek jako židle, nebo zaparkovaná auta. Musí se tedy nějak určit,

co s takovými objekty bude a jak je za pohyblivé určit. Máme možnost tyto objekty do mapy vůbec nepřidávat, nebo je mít označené a pohyblivé, nesmí se ale stát, že zaznamenáme pohyblivý objekt a uložíme ho jako statický.

Klasická implementace SLAM umí odstranit landmark, i poměrně velké množství landmarků, bez většího vlivu na hodnotu konvergence. To je také využíváno pro úpravu mapy, kdy se odstraňují přebytečné landmarky, které se již na svém místě nenacházejí.