HAL open science

Localisation en intérieur par SLAM magnéto-visuel-inertiel

Jade Coulin

To cite this version:

Jade Coulin. Localisation en intérieur par SLAM magnéto-visuel-inertiel. Algorithme et structure de données [cs.DS]. Université Paris sciences et lettres, 2023. Français. NNT : 2023UPSLM012 . tel-04250524

HAL Id: tel-04250524

https://pastel.hal.science/tel-04250524v1

Submitted on 19 Oct 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

PSL UNIVERSITÉ PARIS

THÈSE DE DOCTORAT
DE L'UNIVERSITÉ PSL

Préparée à MINES Paris-PSL

Localisation en intérieur par SLAM magnéto-visuel-inertiel

Soutenue par

Jade Coulin

Le 21 mars 2023

École doctorale n°621

Ingénierie des Systèmes, Matériaux, Mécanique, Energétique

Spécialité

Informatique temps réel, robotique et automatique

Composition du jury :

Silvère BONNABEL Professeur, Mines Paris PSL	*Président*
David FILLIAT Professeur, ENSTA Paris, Institut Polytechnique de Paris	*Rapporteur*
Christophe PRIEUR Directeur de recherche, Gipsa-lab, CNRS	*Rapporteur*
Patrick RIVES Directeur de recherche, INRIA	*Examinateur*
Ni ZHU Chargée de Recherche, GEOLOC, Université Gustave Eiffel	*Examinateur*
Richard GUILLEMARD Ingénieur-Chercheur, Find & Order	*Examinateur*
Cyril JOLY Chargé d'Enseignement Recherche, Mines Paris PSL	*Examinateur*
Arnaud DE LA FORTELLE CTO, Heex Technologies	*Directeur de thèse*

MINES PARIS | PSL

Table des matières

1 Introduction **11**
1.1 Contexte . . . 11
1.2 Solutions de localisation en intérieur existantes . . . 12
1.2.1 Approches visuelles-inertielles . . . 12
1.2.2 Autres capteurs envisageables pour se localiser . . . 13
1.2.3 Utilisation du champ magnétique en intérieur . . . 14
1.3 Fusion magnéto-visuelle-inertielle et ses objectifs . . . 15
1.4 Contributions . . . 15
1.5 Plan . . . 16

2 Préliminaires : algorithmes de localisation visuels-inertiels **17**
2.1 Présentation du système d'acquisition . . . 18
2.1.1 Capteurs . . . 18
2.1.2 Notations et référentiels . . . 18
2.1.3 Synchronisation . . . 21
2.1.4 Calibration . . . 24
2.2 Un filtre de Kalman visuel-inertiel : le MSCKF . . . 30
2.2.1 Vecteur d'état . . . 31
2.2.2 Propagation . . . 32
2.2.3 Mesure . . . 35
2.3 SLAM visuel-inertiel par ajustement de faisceaux et fermeture de boucles . . . 36
2.3.1 Optimisation de graphe visuel-inertiel . . . 37
2.3.2 Réutilisation de cartes visuelles et fermeture de boucles . . . 38

3 Le champ magnétique : modélisation et cartographie **41**
3.1 Modélisation usuelle du champ magnétique . . . 42
3.1.1 Le champ magnétique en intérieur . . . 42
3.1.2 Caractéristiques physiques du champ magnétique . . . 43
3.1.3 Interpolation du champ par processus gaussiens . . . 44
3.1.4 Création de carte magnétique . . . 47
3.1.5 Adaptation à la création de carte en ligne . . . 51
3.2 Impact des erreurs de localisation dans l'estimation du champ . . . 52
3.3 Processus Gaussiens avec entrées et sorties simultanément bruitées . . . 54
3.3.1 Méthodes de l'état de l'art . . . 54
3.3.2 Introduction du problème . . . 55
3.3.3 Étude de la loi de probabilité d'une fonction avec entrées et sorties simultanément bruitées . . . 56

3.3.4 Adaptation à un Processus Gaussien à rang réduit 65
3.3.5 Adaptation pour un Processus Gaussien itératif à rang réduit : RRONIG (Reduced-Rank Online Noisy Input GP) 68
3.4 Etude du comportement de différents modèles de GP avec entrées et sorties bruitées sur des données simulées 70
3.5 Carte obtenue à partir de mesures avec positions et orientations bruitées 74
3.5.1 Insertion de l'erreur de rotation 74
3.5.2 Processus Gaussien à rang réduit avec positions et orientations bruitées . 81

4 Un SLAM magnéto-visuel-inertiel par fusion serrée 85
4.1 État de l'art sur la localisation magnétique 86
4.1.1 Relocalisation magnétique dans une carte pré-existante 86
4.1.2 SLAM magnétiques 88
4.2 Le MSCKF magnéto-visuel-inertiel 89
4.2.1 Vecteur d'état 89
4.2.2 Propagation 89
4.2.3 Mesures magnétiques du MSCKF 90
4.3 MSCKF magnéto-visuel-inertiel exploitant une carte magnétique créée hors ligne 91
4.3.1 Protocole expérimental 91
4.3.2 Création de la carte magnétique 92
4.3.3 Résultats de localication du MSCKF magnéto-visuel-inertiel 93
4.3.4 Comparaison avec un algorithme de SLAM visuel-inertiel réutilisant une carte visuelle 96
4.4 MSCKF magnéto-visuel-inertiel avec création de carte en ligne 100
4.4.1 Résultats avec le modèle classique de création de carte 100
4.4.2 Ajout de l'incertitude de localisation dans le modèle 101

5 Calibration en ligne du magnétomètre en intérieur 107
5.1 Calibration en extérieur et en intérieur : avantages et inconvénients 108
5.1.1 La méthode classique : calibration en extérieur dans un champ magnétique constant 108
5.1.2 Difficultés de la calibration en intérieur 108
5.2 Une nouvelle méthode de calibration en intérieur 109
5.2.1 Description générale 109
5.2.2 Etape de calibration grossière 110
5.2.3 Etape de raffinement : optimisation de la calibration à l'aide de la carte magnétique 113
5.3 Evaluation de la précision de la calibration 113
5.4 Apport d'une calibration précise pour une localisation en ligne basée sur le magnétomètre 115

6 Conclusion 120
6.1 Conclusion 120
6.2 Perspectives 121
6.2.1 Nouveaux algorithmes de localisation intégrant le magnétomètre 121
6.2.2 Relocalisation 121
6.2.3 Prise en compte des ascenseurs et autres perturbations magnétiques . . . 122

6.2.4 Applications du modèle de création de carte à d'autres variables que le champ magnétique 123

A Dérivées des fonctions propres (processus gaussien à rang réduit) 124

A.1 Dérivée première 124

A.2 Dérivée seconde 124

A.3 Généralisation 125

Table des figures

2.1 Casque utilisé pour les acquisitions. Il est équipé de quatre caméras FLIR Blackfly S sur chacun de ses côtés et d'un capteur SBG-Ellipse-N sur le sommet, qui contient une centrale inertielle, un GNSS et un magnétomètre. 18
2.2 Schéma de la triangulation avec le même point 3D vu par deux caméras différentes. Les deux projections de son image par chacune des deux caméras permettent d'estimer sa position en 3D, $\tilde{x}$. 21
2.3 Schéma des mesures physiques effectuées par les différents capteurs au niveau du système (en abscisse, le temps en millisecondes). La longueur des barres indiquent le temps nécessaire pour l'acquisition d'une donnée. Les mesures de l'IMU en orange correspondent aux signaux qui sont utilisés pour déclencher des mesures caméras. 22
2.4 Schéma des mesures physiques effectuées par les différents capteurs et leur temps de transmission au système (l'ordinateur), indiqués en violet. 23
2.5 Variance d'Allan de l'accéléromètre (acquisition sur 50 h). 26
2.6 Exemple d'image obtenue avec une caméra de type "fish-eye". 27
2.7 Exemple d'aprilgrid pour la calibration des caméras. 27
2.8 Mesures acquises lors d'une calibration en extérieur, loin de toute perturbation magnétique. 30
2.9 Variance d'Allan du magnétomètre (acquisition sur 50 h). 31
2.10 Schéma du SLAM visuel par ajustement de faisceaux. La fenêtre figée correspond aux poses caméras qui visualisent les mêmes points 3D que dans la pose actuelle. La dernière pose figée est la pose directement antérieure à la première de la fenêtre locale. 38
2.11 Schéma du SLAM visuel-inertiel (par ajustement de faisceaux contraints par les mesures inertielles). 39

3.1 Processus gaussien à une dimension. La courbe bleue indique la fonction réelle et la rouge le processus gaussien obtenu à partir des mesures bruitées (points noirs). Les deux courbes en pointillés représentent l'intervalle de confiance à 2σ. 46
3.2 Premières fonctions constituant la base de fonctions permettant de modéliser le champ magnétique pour un carré (2 dimensions). 48
3.3 Exemple de cube de données et de cube de modèle élargi obtenus à partir du cube étudié. En points bleus figurent toutes les données acquises pour le cube de données, représenté en pointillés réguliers bleus. Ce modèle permet d'obtenir la carte sur l'aire du carré bleu (ligne continue). En dehors des limites en bleu (continu), les modèles d'autres cubes sont utilisés. De même pour les cubes rouges. Les cubes de modèle élargi sont représentés en couleur plus claire en pointillés discontinus et correspondent aux cubes les plus grands. 51

3.4 Valeurs magnétiques acquises par le magnétomètre (+) comparées aux valeurs prédites par le modèle (ligne continue, les pointillés correspondent à l'intervalle d'incertitude 3σ). Le carré en haut à droite représente le chemin parcouru, auquel correspondent ces prédictions. La trajectoire rouge représente la trajectoire des données acquises pour cette comparaison. Les prédictions du modèle sont obtenues à partir des données acquises par la trajectoire bleue du cube tracé en haut à droite (données d'apprentissage). 52

3.5 Comparaison de deux profils magnétiques à z = 0.3 m, pour deux valeurs différentes de σ_m. À gauche, $\sigma_m = 0.28\mu T$. À droite, $\sigma_m = \sqrt{2}\mu T$. Les trajectoires rouges correspondent aux séquences d'acquisition des mesures pour créer la carte. 53

3.6 Exemple illustrant les résultats obtenus par SONIG ([BSvV16]), en haut, et notre méthode (processus Gaussien à rang réduit avec données d'entrée bruitées), en bas. $\sigma_x = 0.4$ et $\sigma_y = 0.1$. Les courbes de prédiction sont en noir et les enveloppes grises correspondent à l'intervalle à 95%. En bleu, la fonction à approximer. Les cercles rouges correspondent aux mesures. Les cercles noirs indiquent les nœuds pour SONIG. 71

3.7 Exemple de résultats obtenus par SONIG qui n'interpolent pas correctement la fonction vérité-terrain ([BSvV16]). $\sigma_x = 0.4$ et $\sigma_y = 0.1$. Les courbes de prédiction sont en noir et les enveloppes grises correspondent à l'intervalle à 95%. En bleu, la fonction à approximer et les cercles correspondent aux mesures. 73

4.1 Schéma du réseau de neurones combinant la vision et le magnétomètre. V_g contient un descripteur de taille 4096 pour les images. V_m correspond aux quatre valeurs du champ magnétique $(m_x, m_y, m_z, ||m||)$. FC signifie "fully connected layer". 87

4.2 Principe général du MSCKF magnéto-visuel-inertiel avec carte magnétique construite hors ligne. La carte est obtenue à partir de données magnétiques et de positions vérité-terrain. En ligne, la carte magnétique est réutilisée par le MSCKF pour une localisation sans dérive et robuste au cours du temps. 91

4.3 Norme du champ magnétique prédit (en μT). Seules les prédictions qui sont situées à moins d'un mètre de la trajectoire d'apprentissage (en noir) sont représentées. 93

4.4 Localisation obtenue par le MSCKF magnéto-visuel-inertiel (Section 4.2) et le MSCKF visuel-inertiel [MR07] sur une séquence obtenue en juillet 2021. La carte magnétique utilisée date de plus d'un an (29 mai 2020). Les données sont plus récentes (juillet 2021). En haut : vue Google Earth. En bas : vue 3D. 94

4.5 Estimation du biais pour la séquence de la Figure 4.4 (juillet 2021). Les valeurs de la calibration vérité-terrain sont en pointillés. Elles ont été obtenues en réalisant une calibration en extérieur quelques minutes avant le début de la séquence. . . . 95

4.6 Comparaison du champ magnétique prédit et des mesures brutes du champ magnétique (après application de la calibration) selon les axes x, y et z pour la séquence de la Figure 4.4 (juillet 2021). En dehors de la carte magnétique les prédictions sont égales à 0. La séquence commence en extérieur pendant les premières 60 secondes. Ensuite, entre la 120ème et la 130ème secondes, la trajectoire passe par un corridor qui est en dehors du domaine cartographié magnétiquement. 96

4.7 Comparaisons du MSCKF magnéto-visuel-inertiel et de l'algorithme VISLAM [CER+21] pour une utilisation avec une caméra, une IMU et un magnétomètre (non utilisé par le VISLAM (Séquence 1 du Tableau 4.1). Les résultats sont obtenus quelques minutes après l'initialisation, seule la trajectoire parcourue pendant les 30 dernières secondes avant la position courante est représentée. 99
4.8 Principe général du MSCKF magnéto-visuel-inertiel avec carte magnétique construite en ligne. La carte est obtenue à partir de données magnétiques et des positions calculées par notre algorithme. La carte magnétique permet de corriger la dérive principalement lorsque le système repasse par des endroits qu'il a déjà explorés. Contrairement à la vision, la correction est possible même si l'angle de vue est très différent. 100
4.9 Utilisations du MSCKF magnéto-visuel-inertiel avec une caméra et une IMU (avec un modèle de carte classique, sans prise en compte des incertitudes de position). La trajectoire vérité-terrain est en bleu et les trajectoires calculées par le MSCKF en rouge. La deuxième trajectoire dérive rapidement. 101
4.10 Utilisations du MSCKF magnéto-visuel-inertiel avec une caméra et une IMU avec un modèle de carte prenant en compte ou non les incertitudes de position (même séquence que sur la Figure 4.9). La trajectoire vérité-terrain est en bleu, les trajectoires calculées par le MSCKF sont en rouge. On voit que lorsque les incertitudes de position sont incluses dans le modèle, la dérive entre chaque tour est plus faible. 103
4.11 Comparaisons de l'algorithme VISLAM et du MSCKF magnéto-visuel-inertiel pour un système équipé d'une caméra, d'une IMU et d'un magnétomètre pour plusieurs boucles effectuées avec des angles de vue différents. La trajectoire vérité-terrain est représentée en bleu, les trajectoires estimées par les algorithmes de SLAM sont en rouge. 105
4.12 Comparaisons de l'algorithme VISLAM et du MSCKF magnéto-visuel-inertiel pour un système équipé d'une caméra, d'une IMU et d'un magnétomètre. Le système parcourt plusieurs couloirs dans les deux sens. La trajectoire vérité-terrain est représentée en bleu, les trajectoires estimées par les algorithmes de SLAM sont en rouge. 106

5.1 Schéma global de l'algorithme de calibration proposé. 110
5.2 Zone étudiée correspondant à la Séquence 3 dans le Tableau 5.1 et le champ magnétique vérité-terrain le long de la trajectoire de calibration utilisée. Le champ magnétique est représenté selon chacune de ses trois composantes, x, y et z. . . . 115
5.3 Données magnétiques obtenues durant la Séquence 2. 116
5.4 Zone où la calibration est effectuée pour la Séquence 1 dans le Tableau 4.1. . . . 117
5.5 Exécutions du MSCKF magnéto-visuel-inertiel avec différentes méthodes de calibration du magnétomètre (Séquence 1 dans le Tableau 5.2). La trajectoire vérité-terrain est représentée en bleu et les trajectoires SLAM en rouge. En haut à gauche : les mesures magnétiques ne sont pas utilisées et l'algorithme est semblable à une odométrie visuelle-inertielle. 119

6.1 Norme du champ magnétique des trois étages d'un bâtiment et du terrain l'entourant (prédictions jusqu'à 1 m de la trajectoire). 121
6.2 Exemple de champ magnétique avec ses orientations différentes (données simulées). Les cercles rouges indiquent plusieurs candidats possibles pour la relocalisation : le champ magnétique à ces trois endroits a une norme similaire. 122

Liste des tableaux

1.1 Caractéristiques de différents capteurs pour la localisation en intérieur. 14

3.1 Précision moyenne de notre algorithme et de SONIG ([BSvV16]) pour 500 fonctions différentes générées. n est l'index de la donnée étudiée et N le nombre total de mesures. 72

4.1 ATE, erreurs de déclinaison et d'azimut du MSCKF [MR07], du MSCKF avec réutilisation de carte magnétique (Section 4.2) et du VISLAM avec réutilisation de carte visuelle [CER+21] sur trois séquences différentes. Ces séquences sont exécutées avec des cartes de plus en plus anciennes (carte visuelle pour [CER+21] et carte magnétique pour l'algorithme proposé). Pour le MSCKF [MR07] qui n'utilise aucune carte, la même erreur de localisation est indiquée pour chaque date associée à la création de cartes. 97

4.2 Temps d'exécution et erreur de localisation du MSCKF avec et sans données magnétiques avec réduction du nombre d'images utilisées. 98

4.3 ATE, erreurs de déclinaison et d'azimut du MSCKF magnéto-visuel-inertiel avec création de carte magnétique en ligne (une caméra) avec différents modèles de mise à jour de la carte magnétique. Les résultats de l'odométrie visuel-inertielle sont indiqués. 102

4.4 ATE, erreurs de déclinaison et d'azimut du MSCKF magnéto-visuel-inertiel avec création de carte magnétique en ligne (une caméra) avec l'insertion de l'erreur de position, avec ou sans gonflement du bruit capteur pour prendre en compte la trop grande confiance du MSCKF dans ses prédictions. 103

5.1 Comparaison de la précision de la calibration obtenu avec différentes méthodes. . 116

5.2 Erreurs de position (ATE), de déclinaison et d'azimut du MSCKF magnéto-visuel-inertiel en ligne, pour différentes calibrations magnétiques. Les résultats d'une odométrie visuelle-inertielle sont aussi indiqués. 117

Notations

$\mathbb{R}$ Espace des réels

B Fonction représentant le champ magnétique en tout point (ou toute autre variable dans un cas plus général)

x Valeur scalaire

$\mathbf{x}$ Vecteur

$\mathbf{x}^j$ Vecteur numéro j dans une liste

$\mathbf{x}_i$ i^{me} coordonnée du vecteur $\mathbf{x}$

M Matrice

$*$ Opérateur pour la multiplication entre deux matrices

$.^\top$ Transposée

$\otimes$ Composition de deux poses pour en former une nouvelle

$SO(3)$ Groupe des matrices de rotation

$\log_{SO(3)}$ Opérateur sur le groupe $SO(3)$ permettant d'obtenir un vecteur à partir d'une matrice de rotation à l'aide de l'algèbre de Lie

$\exp_{SO(3)}$ Opérateur sur le groupe $SO(3)$ permettant d'obtenir une matrice de rotation à partir d'un vecteur à l'aide de l'algèbre de Lie

$\hat{x}$ Valeur estimée de x

$\tilde{x}$ Erreur entre la valeur réelle de x et son estimation

$\mathbb{E}[.]$ Espérance

$\mathrm{Var}(.)$ Variance

$\mathrm{Cov}(.\,,.)$ Covariance

$\kappa(.\,,.)$ Covariance de la variable étudiée (ici B) : $\mathrm{Cov}(B(.)\,,B(.))$

ξ Bruit blanc gaussien sur les données d'entrée

Σ^{t_α} Variance sur les données d'entrée (les positions dans le cas du champ magnétique) au temps t_α

ε Bruit blanc gaussien sur les données de sortie

Σ_b Variance sur les données capteur

$.\wedge.$ Produit vectoriel

$\lfloor\cdot\rfloor_\times$ Opérateur associant à un vecteur la matrice qui correspond à son produit vectoriel

Ω Opérateur calculant l'espérance élément par élément d'une matrice

$\langle . , . \rangle$ Type de produit scalaire pour une matrice P et un tenseur Q. P est de dimension $m \times n$ et Q est de dimension $l \times m \times n$. On obtient un vecteur de taille l, dont la i-ème composante a pour expression : $(\langle P, Q \rangle)_i = \sum_{j=1}^{m} \sum_{k=1}^{n} P_{j,k} Q_{i,j,k}$

$H_1(\kappa)(\mathbf{x}, \mathbf{x}')$ Dérivée seconde de κ selon la première composante de $\mathbf{x}$ uniquement

$H_2(\kappa)(\mathbf{x}, \mathbf{x}')$ Dérivée seconde de κ selon la deuxième composante :

$$\begin{cases} H_1(\kappa)(\mathbf{x}, \mathbf{x}') = \dfrac{\partial^2 \kappa(\mathbf{x}, \mathbf{x}')}{\partial \mathbf{x} \partial \mathbf{x}} \\ H_2(\kappa)(\mathbf{x}, \mathbf{x}') = \dfrac{\partial^2 \kappa(\mathbf{x}, \mathbf{x}')}{\partial \mathbf{x}' \partial \mathbf{x}'} \end{cases}.$$

$\lfloor \cdot \rfloor_D$ Opérateur transformant un vecteur en une matrice diagonale :

$$\begin{aligned} &\lfloor \cdot \rfloor_D : \mathbb{R}^3 \to \mathbb{R}^{3 \times 3} \\ &\left\lfloor \begin{bmatrix} a \\ b \\ c \end{bmatrix} \right\rfloor_D = \begin{bmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{bmatrix} \end{aligned} \tag{1}$$

$\lfloor \cdot \rfloor_S$ Opérateur transformant un vecteur en une matrice symétrique à diagonale nulle :

$$\begin{aligned} &\lfloor \cdot \rfloor_S : \mathbb{R}^3 \to \mathbb{R}^{3 \times 3} \\ &\left\lfloor \begin{bmatrix} a \\ b \\ c \end{bmatrix} \right\rfloor_S = \begin{bmatrix} 0 & a & b \\ a & 0 & c \\ b & c & 0 \end{bmatrix} \end{aligned} \tag{2}$$

Remerciements

Je remercie mes encadrants pour leur aide et leurs remarques sur mon travail qui m'ont permis de beaucoup progresser au cours de ces trois ans. Je suis très heureuse que Vincent Gay-Bellile et Richard Guillemard m'aient offert la possibilité de continuer de travailler en thèse après mon stage de fin d'études dans le laboratoire LVML (Laboratoire pour la Vision, la Modélisation et la Localisation) du CEA-List (Laboratory for Integration of Systems and Technology). J'ai beaucoup apprécié la confiance de Richard dans le projet d'utilisation du magnétomètre et son enthousiasme à chaque fois que nous parvenions à obtenir de bons résultats au terme de nombreux efforts. Vincent m'a permis de voir à de nombreuses reprises le problème sous des angles différents et d'enrichir ainsi fortement mon travail. Merci beaucoup à Cyril Joly d'avoir montré dès le début beaucoup d'intérêt dans le montage d'une collaboration avec le Centre de Robotique de Mines Paris PSL et pour son soutien, malgré les difficultés liées au suivi à distance à cause de la crise sanitaire. Je suis très reconnaissante à Arnaud de La Fortelle d'avoir accepté d'être mon directeur de thèse et de m'avoir initiée en profondeur au formalisme des processus Gaussiens.

Je tiens à remercier également mes rapporteurs David Filliat et Christophe Prieur, ainsi que Patrick Rives, Ni Zhu et Silvère Bonnabel pour avoir accepté de relire ma thèse et d'évaluer mon travail. Leurs commentaires avisés m'ont permis d'améliorer plusieurs aspects du manuscrit.

Je souhaite aussi remercier Régis Vinciguerra, le directeur du laboratoire LVML, d'avoir permis que s'installe une collaboration entre le LVML et le Centre de Robotique.

Je tiens à adresser ma gratitude aux différents membres du LVML que j'ai appris à connaître durant ces trois ans. Je suis très heureuse de savoir que ce travail continue d'être utilisé, en particulier il a permis le développement d'un autre projet de thèse avec Iad Abdul Raouf.

Enfin, merci à ma famille et à mes amis pour leur soutien. Même s'ils n'ont pas contribué directement à cette thèse, ils savent tous désormais que la localisation en intérieur est un problème complexe !

Chapitre 1

Introduction

1.1 Contexte 11
1.2 Solutions de localisation en intérieur existantes 12
1.2.1 Approches visuelles-inertielles 12
1.2.2 Autres capteurs envisageables pour se localiser 13
1.2.3 Utilisation du champ magnétique en intérieur 14
1.3 Fusion magnéto-visuelle-inertielle et ses objectifs 15
1.4 Contributions 15
1.5 Plan 16

1.1 Contexte

La problématique de la localisation en intérieur est un des défis principaux de la robotique mobile actuelle [RSS11, CCC+16]. Dans la vie courante, les robots aspirateurs se déplacent aussi en utilisant des algorithmes de SLAM (Simultaneous Localization And Mapping) [VHKR11] : ils réalisent à la fois une carte de l'environnement dans lequel ils évoluent, tout en s'y localisant. De nombreuses autres applications sont possibles mais elles requièrent une localisation de meilleure qualité que celle de ces robots : en cas d'incendie, une localisation précise des pompiers en intervention dans les bâtiments permettrait de faciliter le travail des forces de soutien restées à l'extérieur. En cas de danger, elles pourraient plus facilement intervenir dans les opérations de sauvetage. Dans le domaine de la logistique, des aides pour que l'opérateur dans l'entrepôt se rende plus facilement à l'endroit où se trouve le produit commandé se développent [HGJL17, DU22]. Certaines méthodes utilisent des robots ou bien des humains qui se déplacent en étant aidés par un logiciel spécifique. De nombreuses tâches de transports d'objets peuvent être exécutées par des robots à condition qu'ils se repèrent avec exactitude, comme pour aider les personnes à mobilité réduite ou transporter de lourds colis. Connaître la position d'une personne permettrait aussi aux entreprises de publicité de créer des annonces avec une visibilité et un impact plus grands [LAO04].

Dans le cadre de cette thèse, on s'intéresse particulièrement à la problématique de la localisation embarquée : on cherche à connaître la position d'une personne équipée d'un ensemble de capteurs dans un bâtiment inconnu. Les capteurs utilisés doivent être légers et peu coûteux. Il est nécessaire que les déplacements de l'utilisateur et des personnes qu'il peut croiser soient gênés

aussi peu que possible par la présence de l'équipement, tout en obtenant une localisation très précise. De plus, comme l'environnement n'a pas été exploré au préalable, aucune instrumentation de celui-ci n'est disponible.

En intérieur, les signaux provenant des systèmes GNSS (Global Navigation Satellite System, Système Global de Navigation par Satellite) sont trop affaiblis par le passage des cloisons pour être exploitables. Les solutions pour la navigation en extérieur ne sont pas directement utilisables. De nombreux travaux portant sur la localisation en intérieur ont été développés ces dernières années. La plupart traitent simultanément la problématique de la localisation et de la cartographie (SLAM) car l'environnement n'est pas toujours connu *a priori*. La méthode la plus performante actuellement consiste à utiliser plusieurs caméras et une centrale inertielle (IMU) pour se repérer [CER+21]. Néanmoins d'autres possibilités, comme le calcul de distances à partir des signaux de bornes WiFi [NJD+17, HABD19, KKZA22] ou d'UWB (Ultra-Wide-Band, [TW19, FWM+21]) ou de LIDAR [NYC+21, SEM+20], ont été développées. Les premières solutions nécessitent que l'environnement soit instrumenté et des phénomènes de réflexions multiples des signaux rendent parfois difficiles l'estimation des distances réelles. Ces signaux peuvent aussi être utilisés autrement. Si l'on sait de quelles bornes on reçoit un signal, on peut estimer grossièrement la zone géographique du système et l'utiliser pour désambigüiser la localisation avec d'autres capteurs (par exemple pour des environnements répétitifs). On peut aussi créer des cartes à partir des signaux reçus : à chaque point de l'espace on associe l'intensité des bornes qui y a été mesurée. Cela permet de caractériser chaque emplacement (méthode de fingerprinting, récupération d'empreintes UWB ou WiFi). Le LIDAR, de son côté, reste un capteur coûteux et ses mesures sont affectées par les chocs et vibrations. D'autres capteurs ont été aussi étudiés dont le baromètre et le magnétomètre. Le baromètre est très sensible aux variations de pression, qui ne sont pas uniquement dues à l'altitude mais aussi à la météo, ce qui rend ces mesures très variables. Il ne permet pas une estimation simple de l'altitude du système en cas de variations de hauteur de quelques mètres seulement et il est surtout utilisé dans l'aviation.

Le magnétomètre permet aussi de caractériser l'environnement en intérieur. Le magnétomètre mesure le champ magnétique et, en extérieur, il permet de connaître l'orientation du système en cap grâce au Nord magnétique [LZZ+19]. En intérieur, le champ magnétique a une apparence spatiale chaotique en raison des éléments métalliques dans la structure du bâtiment. Cependant, c'est justement cette variabilité qui peut servir à se localiser. Le champ est variable spatialement mais temporellement stable dans des environnements tels que des bureaux ou des pièces d'une maison [LGDR12, CDS+11]. On peut donc établir des cartes magnétiques qui caractérisent chaque point de l'espace étudié et en tirer parti pour se localiser. Le magnétomètre est un capteur à bas coût que l'on trouve dans tous les smartphones. Des SLAM magnétiques obtenus par fusion lâche avec des algorithmes d'odométrie déjà existants ont montré la possibilité d'utiliser le magnétomètre et son intérêt pour réduire la dérive des trajectoires estimées [VHKR11, KS18].

1.2 Solutions de localisation en intérieur existantes

1.2.1 Approches visuelles-inertielles

Les capteurs permettant la localisation la plus précise dans le cadre de la navigation en intérieur, tout en étant peu coûteux, sont la centrale inertielle (ou IMU, Inertial Measurement Unit) composée de trois accéléromètres et de trois gyromètres, un selon chaque axe, ainsi que des caméras. Ces capteurs sont très répandus.

L'accéléromètre mesure la force spécifique, liée à l'accélération du système : il est nécessaire

d'intégrer deux fois pour obtenir la position. Le gyromètre renvoie la vitesse de rotation : une intégration permet d'obtenir l'orientation. Or chaque intégration augmente l'incertitude sur la mesure : la variance augmente linéairement avec le temps dans le cadre d'une simple intégration, et cubiquement dans le cadre d'une double intégration. Les centrales inertielles MEMS (Micro-ElectroMechanical Systems), même hauts de gamme, dérivent au bout de quelques secondes si aucun renseignement complémentaire sur l'odométrie réelle du système ne leur est fournie. La caméra permet de limiter cette dérive : il est possible de calculer le déplacement (à une échelle près) entre chaque image grâce au calcul du déplacement des points d'intérêt via la théorie de la géométrie épipolaire. Néanmoins, dans les environnements peu texturés, le capteur visuel ne dispose pas de renseignements suffisants pour évaluer la position. Dans ce cas, le risque de divergence est élevé et l'IMU permet de compenser les erreurs liées à la vision. De nombreux algorithmes fusionnant ces deux capteurs ont été proposés [DRMS07, LCT07, LS13, MR07, CER+21, SC22]. Ils peuvent être classés en deux types : des filtres de Kalman [DRMS07, LCT07, LS13, MR07] et des approches par optimisation de graphes [CER+21, SC22].

Cependant, l'IMU est un capteur de localisation relative c'est à dire qu'il estime le déplacement entre deux postions. Il en est de même pour la caméra si son utilisation consiste à comparer le déplacement des pixels d'une image à l'autre. La localisation globale se fait alors en accumulant les localisations relatives, ce qui cause une accumulation d'erreurs au cours du temps et donc de la dérive. Un capteur de position absolue ne présente pas ce problème car chacune de ses mesures de position est indépendante des précédentes. Pour réduire la dérive accumulée au cours du temps, des fermetures de boucles visuelles peuvent êtres appliquées lorsqu'un même lieu est visité plusieurs fois. Néanmoins cela ne fonctionne que si l'on repasse par le même lieu avec un point de vue similaire : si l'angle de vue est trop différent, la fermeture de boucle risque d'échouer [RMA14]. De plus, cette méthode est peu robuste dans le cadre d'environnements à l'apparence répétitive, tels que des couloirs où les portes sont identiques et régulièrement espacées.

Une autre solution pour réduire la dérive des algorithmes de SLAM visuel-inertiel est d'utiliser une carte visuelle de l'environnement reconstruite hors ligne après une première visite des lieux [CER+21]. Des algorithmes de reconnaissance de points de vue sont alors utilisés pour mettre en correspondance l'image courante à localiser et l'image la plus ressemblante dans la carte. Mais ces algorithmes de relocalisation ne sont pas robustes aux variations de luminosité et aux déplacements des objets au cours du temps. Une manière de robustifier ces approches est de réaliser plusieurs acquisitions des mêmes scènes à des instants différents, aussi variées que possible [LM22, CK18]. Une telle acquisition peut être longue et fastidieuse et il est parfois difficile d'envisager toutes les possibilités. L'ajout d'un capteur supplémentaire complémentaire de l'IMU et des caméras peut résoudre ces problèmes de manque de robustesse de la localisation.

1.2.2 Autres capteurs envisageables pour se localiser

De nombreux capteurs aux propriétés très différentes ont été étudiés dans le contexte de la localisation. Certains nécessitent une instrumentation du bâtiment, comme les UWB (Ultra Wide-Band) [TW19, FWM+21] et le signal WiFi [NJD+17, HABD19]. Les UWB nécessitent de poser en amont dans la zone étudiée plusieurs émetteurs, dont la position doit être connue, et ne sont donc pas adaptés à l'exploration d'un environnement inconnu. Comme de plus en plus de lieux sont équipés du WiFi, la deuxième solution semble la plus facile à utiliser pour se repérer dans de grandes surfaces. Cependant, le signal WiFi peut rebondir et être affaibli par la traversée des murs ce qui peut causer parfois de mauvaises estimations des distances aux bornes [KKZA22]. Pour l'instant, ce problème n'est que partiellement résolu, il s'agit d'un domaine de recherche actif, à la fois pour le WiFi et les UWB [KKZA22, MR22].

Le LIDAR est un autre des capteurs étudiés [NYC+21, SEM+20]. Il permet d'obtenir une visualisation en 3D de scènes. Il est utilisé principalement dans le cadre du véhicule autonome et aussi parfois pour la navigation en intérieur mais son prix reste élevé, ce qui limite actuellement son usage. Le LIDAR est aussi assez sensible aux chocs et les surfaces réfléchissantes, telles que les miroirs, perturbent la réception de son signal. Le magnétomètre ne présente pas ces inconvénients. Néanmoins, l'information qu'il fournit est bien moins riche que celle d'un LIDAR : le champ magnétique est un vecteur en 3 dimensions. Les informations sur ces différents capteurs sont résumées dans le Tableau 1.1.

Le magnétomètre est le capteur qui répond le mieux aux différentes contraintes que nous avons posées. Son usage est récent dans le contexte de la localisation en intérieur [CDS+11, FAL+13, VHKR11, KS18], alors qu'on le retrouve fréquemment pour des applications en extérieur afin d'estimer le cap [SDS13, LZZ+19].

Capteur	Capteur relatif/absolu (pour la localisation)	Instrumentation de l'environnement	Limitations	Coût
Centrale inertielle (IMU)	Relatif	Non	Chocs et vibrations, bruits importants sur les mesures	Moyen (pour une IMU de milieu de gamme)
Caméra	Relatif / Absolu si carte visuelle	Non	Luminosité, flou de bouger, peu de textures, miroirs	Faible
UWB	Absolu	Oui	Rebonds du signal, signal faible à cause de murs épais	Moyen
WiFi	Absolu	Oui (mais Wifi souvent disponible)	Rebonds du signal, signal faible à cause de murs épais	Faible
LIDAR	Relatif / Absolu si carte 3D	Non	Rebonds du signal, chocs et vibrations, miroirs	Elevé
Magnétomètre	Absolu si carte magnétique préexistante / Relatif si utilisation du gradient	Non	Forts courants électriques variables, ascenseurs	Faible

TABLE 1.1: Caractéristiques de différents capteurs pour la localisation en intérieur.

1.2.3 Utilisation du champ magnétique en intérieur

On retrouve dans la littérature deux méthodologies principales d'utilisation des propriétés du champ magnétique : l'une est basée sur le calcul du gradient magnétique [CES+17], l'autre sur le repérage par création de carte magnétique [CDS+11, VHKR11, KS18]. Dans tous les cas, les données du magnétomètre sont combinées avec un algorithme d'odométrie utilisant d'autres capteurs : centrale inertielle et/ou caméras.

La première solution repose sur l'utilisation de plusieurs magnétomètres et s'appuie sur le gradient du champ. Elle consiste à modéliser l'évolution du champ magnétique, permettant d'estimer la vitesse. Cette équation supplémentaire est ajoutée à l'état du système dans un algorithme de Kalman contraint multi-états : le MSCKF, qui est détaillé en Section 2.2. La localisation est améliorée par rapport au MSCKF classique car, au lieu de mesurer l'accélération, il est possible d'obtenir directement la vitesse du système. Mais en conditions réelles, cet algorithme est difficile d'application. La calibration de plusieurs magnétomètres est une tâche ardue car ils se perturbent mutuellement à cause des éléments magnétiques qui les composent et cette méthode est brevetée [DVP08]. De plus, si la robustesse de l'algorithme est améliorée par rapport à un MSCKF visuel-inertiel, le phénomène de dérive au cours du temps reste présent.

La deuxième approche consiste à créer une carte magnétique en caractérisant chaque point de l'espace étudié par sa valeur magnétique [FAL+13, VHKR11, KS18]. Cela nous permet de corriger les erreurs si le système se trouve dans la carte. La position est estimée en comparant les valeurs magnétiques mesurées avec les prédictions de la carte. Cependant cette méthode fonctionne bien si on a déjà une approximation correcte de la localisation car de nombreux endroits sur une carte complète ont des champs magnétiques similaires. Le même problème existe avec les cartes visuelles quand plusieurs endroits sont visuellement similaires (comme des couloirs identiques). Se relocaliser dans une carte magnétique sans aucun a priori sur sa position reste une tâche ardue. Des méthodes de relocalisation utilisant uniquement le magnétomètre, ou en couplant le magnétomètre avec d'autres capteurs, existent depuis quelques années [LZL+17, LAH18, MTSB16].Pour la localisation le long d'une trajectoire, plusieurs SLAM magnétiques couplés par fusion lâche à des algorithmes d'odométrie en 2D [VHKR11] et en 3D [KS18], sans carte magnétique connue *a priori*, ont été proposés. Leurs résultats démontrent que le magnétomètre peut être utilisé de façon efficace pour corriger la dérive des algorithmes de type SLAM.

1.3 Fusion magnéto-visuelle-inertielle et ses objectifs

Le magnétomètre est un capteur utile pour corriger les problèmes de dérive des algorithmes de localisation visuelle-inertielle qui constituent l'état de l'art. Pour combiner au mieux les avantages du magnétomètre et des méthodes visuelles-inertielles, l'intégration par fusion serrée des données magnétiques dans les algorithmes visuels-inertiels déjà existants semble une solution naturelle. Pour cela, il est nécessaire d'être capable de créer à partir d'un nombre limité de mesures magnétiques une carte aussi précise que possible de l'environnement exploré, en amont de la localisation si l'environnement est déjà connu, ou bien en ligne quand il n'a pas pu encore être exploré. Avec une carte magnétique de bonne qualité, il est possible de comparer la position du système avec la valeur du champ magnétique qu'il mesure, permettant ainsi de corriger sa trajectoire. Pour les méthodes de SLAM en ligne, un temps d'initialisation relativement court est important. Comme pour les caméras et la centrale inertielle, il est crucial que la calibration du magnétomètre puisse être effectuée simplement et rapidement, en développant une méthode efficace pour estimer les facteurs de calibration du magnétomètre en intérieur. Cependant, la variabilité du champ rend leur estimation plus complexe.

Ainsi, l'objectif principal de la thèse est d'améliorer la localisation grâce au capteur magnétique, en étudiant son intégration dans la structure plus globale d'un algorithme de localisation.

1.4 Contributions

Les contributions de la thèse se divisent en trois axes principaux :

— Une nouvelle méthode d'interpolation par processus Gaussien des données du champ magnétique, en incluant l'incertitude sur les positions des mesures. Cette méthode est à rang réduit et adaptée pour une utilisation en temps réel.
— Un algorithme magnéto-visuel-inertiel par fusion serrée. Il permet d'obtenir une localisation plus robuste et réduit fortement le problème de dérive au cours du temps quand une carte magnétique préalable de l'environnement a été réalisée. Des cartes magnétiques datant de plus d'un an peuvent être réutilisées car elles sont sujettes à beaucoup moins de variations que les cartes visuelles. La carte magnétique peut aussi être construite en ligne sans connaissances préalables de l'environnement.
— Un procédé de calibration en intérieur du magnétomètre presque aussi précis qu'une calibration en extérieur du magnétomètre. Cette contribution permet de faciliter le déploiement des solutions de localisation basées sur le champ magnétique puisqu'il n'est plus nécessaire de devoir sortir du bâtiment pour calibrer le magnétomètre avant de s'y localiser.

1.5 Plan

Dans le Chapitre 2, nous allons présenter les éléments de base qui ont servi de support pour cette thèse : le système d'acquisition utilisé et ses capteurs, ainsi que les deux types principaux d'algorithmes de localisation visuelle-inertielle par fusion serrée. Les principes d'une approche basée sur le filtre de Kalman puis par graphes avec ajustement de faisceaux sont décrits.

Les caractéristiques du champ magnétique en intérieur ainsi que sa modélisation sont étudiées dans le Chapitre 3. Le champ magnétique est la grandeur mesurée par le magnétomètre et plus les cartes magnétiques sont de bonne qualité, plus la localisation devrait être bien corrigée par l'ajout du magnétomètre dans les algorithmes existants. À partir d'un processus Gaussien classique adapté au champ magnétique, un modèle plus complet est proposé, prenant en compte à la fois un bruit sur les données d'entrée (les positions) et de sortie (bruit du capteur).

Une fois que la modélisation du champ magnétique et les algorithmes de fusion visuelle-inertielle ont été détaillés, on propose un algorithme de fusion serrée magnéto-visuelle-inertielle dans le Chapitre 4, ayant pour objectif de combiner les informations des trois types de capteur pour réduire la dérive de la localisation. Les résultats obtenus avec des cartes reconstruites hors ligne (lieu déjà connu) et en ligne (pas d'information sur le lieu exploré) sont présentés.

Pour l'utilisation en ligne du magnetomètre dans un algorithme de localisation, une calibration du magnétomètre requise régulièrement car elle varie de façon importante au bout de quelques jours. Actuellement, les méthodes de calibration très précises sont fondées sur l'hypothèse d'un champ magnétique constant qui n'est généralement valide qu'en extérieur. Or c'est un inconvénient pour parvenir à initialiser rapidement un algorithme de localisation magnétique avec création de carte en ligne. Le Chapitre 5 présente un algorithme de calibration en intérieur permettant d'estimer précisément les paramètres de calibration du magnétomètre.

Chapitre 2

Préliminaires : algorithmes de localisation visuels-inertiels

2.1 Présentation du système d'acquisition 18
2.1.1 Capteurs 18
2.1.2 Notations et référentiels 18
2.1.3 Synchronisation 21
2.1.4 Calibration 24
2.2 Un filtre de Kalman visuel-inertiel : le MSCKF 30
2.2.1 Vecteur d'état 31
2.2.2 Propagation 32
2.2.3 Mesure 35
2.3 SLAM visuel-inertiel par ajustement de faisceaux et fermeture de boucles 36
2.3.1 Optimisation de graphe visuel-inertiel 37
2.3.2 Réutilisation de cartes visuelles et fermeture de boucles 38

Introduction

Dans cette partie, les éléments servant de base aux travaux développés au cours de la thèse sont présentés. Dans un premier temps, nous présentons le système d'acquisition composé de 4 caméras et d'une centrale inertielle qui a servi à acquérir l'ensemble des données expérimentales présentées dans ce mémoire (Section 2.1). Les caractéristiques des capteurs utilisés, leur synchronisation et leur calibration sont décrites car ces paramètres ont une forte influence sur les résultats de toute expérience de localisation en conditions réelles. Puis les deux principaux types d'algorithmes de localisation visuels-inertiels sont détaillés. Le premier est une extension du filtre de Kalman, particulièrement adapté pour la fusion serrée de plusieurs capteurs (Section 2.2) : il sert de base à l'algorithme magnéto-visuel-inertiel proposé dans le Chapitre 4. Le deuxième est un SLAM visuel-inertiel par ajustement de faisceaux et fermeture de boucles (Section 2.3) : c'est la méthode la plus précise de l'état de l'art actuellement. Nous l'utilisons pour obtenir les vérités-terrain pour nos expériences : il est exécuté avec quatre caméras et suivi d'une optimisation globale de l'ensemble de la reconstruction.

2.1 Présentation du système d'acquisition

Dans cette Section, on présente chacun des capteurs utilisés sur le système utilisé pour se localiser (Section 2.1.1). Puis la manière dont les données sont reçues et traitées pour être correctement synchronisées est brièvement expliquée dans la Section 2.1.3. Enfin, la calibration des différents capteurs, élément indispensable pour obtenir des mesures de bonne qualité, est abordée dans la Section 2.1.4.

2.1.1 Capteurs

Les données utilisées tout au long de ce manuscrit ont été acquises par un casque porté par un piéton. Ce casque est équipé de différents capteurs : il est composé de quatre caméras FLIR Blackfly S, deux alignées à l'avant et deux à l'arrière et d'un capteur SBG-Ellipse-N, incluant une centrale inertielle, un GNSS et un magnétomètre (Figure 2.1). Les caméras sont à obturateur global et les images sont en noir et blanc. Le système GNSS est utilisé uniquement dans le cadre de la localisation en extérieur. Son utilisation ne va pas être détaillée. Tous les capteurs sont montés rigidement et leurs données sont synchronisées (voir Section 2.1.3). La personne portant le casque n'a pas toujours été la même en fonction des expériences. La seule différence significative observée est l'altitude moyenne du casque, qui dépend de la taille de l'utilisateur.

FIGURE 2.1: Casque utilisé pour les acquisitions. Il est équipé de quatre caméras FLIR Blackfly S sur chacun de ses côtés et d'un capteur SBG-Ellipse-N sur le sommet, qui contient une centrale inertielle, un GNSS et un magnétomètre.

2.1.2 Notations et référentiels

Pour des transformations 3D du référentiel E vers le référentiel F, les notations suivantes sont utilisées :

$$X_F = \mathrm{P}_{FE} X_E \tag{2.1}$$

avec P_{FE} la matrice de passage du référentiel E vers le référentiel F. $\mathbf{X}_F$ contient les coordonnées dans le référentiel d'arrivée F et $\mathbf{X}_E$ les coordonnées dans le référentiel d'origine E.

$$\mathbf{X}_E = \begin{bmatrix} x_E & y_E & z_E & 1 \end{bmatrix}^\top, \tag{2.2}$$

$$\mathbf{X}_F = \begin{bmatrix} x_F & y_F & z_F & 1 \end{bmatrix}^\top. \tag{2.3}$$

$$\mathrm{P}_{FE} = \begin{bmatrix} \mathrm{R}_{FE} & \mathbf{t}_{FE} \\ 0_{1\times 3} & 1 \end{bmatrix} \in SE(3) \tag{2.4}$$

où $\mathrm{R}_{FE} \in SO(3)$ est la matrice de rotation du référentiel E au référentiel F, $\mathbf{t}_{FE} \in \mathbb{R}^3$ les coordonnées 3D du référentiel F dans E. Dans la suite, les référentiels principaux qui vont être utilisés sont le référentiel de l'IMU, I, les référentiels liés aux caméras C_i avec i l'indice de la caméra étudiée et le référentiel lié au magnétomètre M. Un référentiel global G est aussi utilisé : son axe Oz est aligné avec le vecteur gravité $\mathbf{g}$. C'est un référentiel local dans lequel la trajectoire du casque est étudiée.

$\lfloor \cdot \rfloor_\times$ est l'opérateur permettant d'associer à un vecteur la matrice qui correspond à son produit vectoriel : $\mathbf{a} \wedge \mathbf{b} = \lfloor \mathbf{a} \rfloor_\times \mathbf{b}$.

2.1.2.1 Centrale inertielle (IMU)

La centrale inertielle mesure la force spécifique et la vitesse de rotation du système. Ses mesures peuvent être erronées en cas de chocs et de vibrations, ce qui arrive rarement lorsqu'elle est portée par un piéton. L'accéléromètre mesure la force spécifique, c'est à dire toutes les forces sauf la force de gravité :

$$\mathbf{F}_{spe} = \mathrm{R}_{IG}(\mathbf{a}_G - \mathbf{g}_G), \tag{2.5}$$

où I est le référentiel de la centrale inertielle, G le repère gravité, $\mathbf{F}_{spe}$ la force spécifique, $\mathbf{a}_G$ l'accélération réelle du système dans le référentiel G et $\mathbf{g}_G$ la force de gravité.

Une centrale inertielle est soumise à la force de gravité et à la force de Coriolis. Pour une centrale inertielle avec des performances moyennes (ce qui est le cas ici), la qualité des mesures étant assez mauvaise, on suppose négligeables la force de Coriolis et la force de rotation de la Terre : leurs valeurs sont très faibles en comparaison du bruit du capteur.

L'accélération est la dérivée seconde de la position. En regardant l'effet d'une double dérivation sur le bruit capteur, on constate que la variance de l'erreur de prédiction évolue de façon cubique. Le gyromètre mesure la vitesse de rotation ω_I. L'orientation du système s'obtient en intégrant une fois ω_W, le recalage de ω_I dans le repère monde W, avec une variance de l'erreur augmentant de façon linéaire. Comme la centrale inertielle utilisée est peu précise, la position diverge au bout de quelques secondes, ce qui rend indispensable son couplage à des caméras ou à un autre capteur pour obtenir une trajectoire robuste sur le long terme.

2.1.2.2 Caméra

La caméra est le capteur du casque qui fournit l'information la plus riche sur l'environnement. Elle peut être mise en défaut pour la localisation lorsque l'environnement observé manque de texture, qu'un trop grand nombre d'objets mobiles occupent l'image ou si la luminosité est insuffisante.

Pour le modèle de projection des caméras, le modèle sténopé (pinhole model) est très fréquemment utilisé. Il permet de modéliser la reprojection des points 3D sur le plan de la caméra,

en 2D. Les paramètres intrinsèques de la caméra ont une influence sur l'image obtenue : il s'agit des distances focales f_x et f_y et des coordonnées de la projection du centre optique dans l'image c_x et c_y. Ce sont les principaux paramètres de la caméra, d'autres sont présents comme le facteur de distortion mais ici nous nous plaçons dans le cadre le plus simple pour expliquer de façon globale le fonctionnement du capteur. Nous pouvons synthétiser leurs valeurs dans la matrice K associée :

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}. \tag{2.6}$$

La fonction de reprojection s'obtient directement à partir des paramètres intrinsèques, pour un point $\mathbf{x}$:

$$\Pi : \begin{bmatrix} x \\ y \\ z \end{bmatrix} \mapsto \begin{bmatrix} f_x \frac{x}{z} + c_x \\ f_y \frac{y}{z} + c_y \end{bmatrix}. \tag{2.7}$$

La caméra permet de connaître la position du système en calculant le déplacement entre deux images. Celui-ci peut s'obtenir de plusieurs manières, une fois que les points d'intérêts identiques entre les deux images sont associés à l'aide d'algorithmes d'appariements [BTG06, Low04, RRKB11] ou de méthodes directes, qui utilisent directement les intensités des pixels et le gradient de l'image pour réaliser l'estimation des poses relatives des images caméras [MFMR16, ACCM21]. Dans le cas des algorithmes d'appariements, le déplacement moyen des pixels permet d'estimer le déplacement entre les deux images. Les points d'intérêt visuels sont des endroits facilement reconnaissables d'une image à l'autre, comme les bords d'un objet. Nous regardons chaque pixel indépendamment pour savoir s'il peut être considéré comme un point d'intérêt : on étudie sa valeur ainsi que celle des pixels voisins, dans un carré de taille 8×8, pour savoir s'il se trouve dans une zone suffisamment contrastée. À chaque point d'intérêt est associé un descripteur unique qui permet de caractériser à quel point il est saillant dans l'image. Pour éviter que certaines zones de l'image concentrent presque tous les points d'intérêt et ne déséquilibrent les calculs, l'image est subdivisée en plusieurs zones avec un nombre minimal de points d'intérêt à trouver dans chacune d'elles. Il existe plusieurs méthodes permettant de détecter les points d'intérêt : les plus utilisées sont basées sur les détecteurs SIFT et SURF, ainsi que les points ORB [BTG06, Low04, RRKB11].

Les formules de la géométrie épipolaire permettent d'estimer le déplacement entre deux images à partir de leurs points 2D, par correspondances 2D-2D, à l'aide de l'algorithme des 5 points [Nis04]. Une autre méthode, le calcul de poses, utilise la projection en 3D des points caméras pour estimer la position. Les positions des points 3D sont obtenus grâce à toutes les images caméras disponibles qui les observent : à partir des coordonnées 2D, les coordonnées 3D des points sont estimées par triangulation. Puis, une fois que les positions 3D sont supposées connues, le calcul de pose de la caméra peut être effectué. L'image du point d'intérêt sur l'image caméra se trouve sur un faisceau provenant du point 3D et passant par le centre optique de la caméra (voir la Figure 2.2). De manière générale, le calcul de poses est moins coûteux en calculs et les positions caméra obtenues sont généralement mieux estimées qu'avec des correspondances 2D-2D [JPTD08]. Ce modèle permet de retrouver l'inter-déplacement entre 2 images caméras tout en reconstituant la géométrie 3D de l'environnement observé à une échelle près si la vision est monoculaire.

2.1.2.3 Magnétomètre

Un magnétomètre peut être mono ou tri-axial. Le magnétomètre tri-axial permet d'obtenir un vecteur 3D qui est une mesure du champ magnétique dans le référentiel lié au magnétomètre,

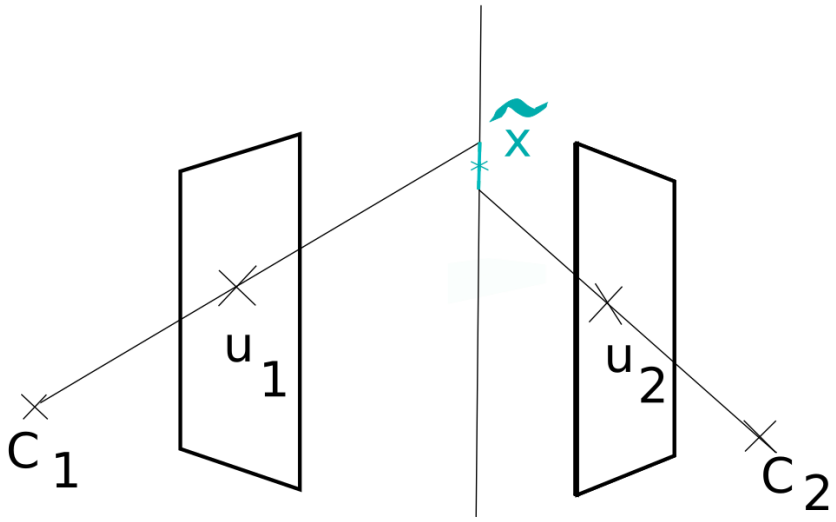


FIGURE 2.2: Schéma de la triangulation avec le même point 3D vu par deux caméras différentes. Les deux projections de son image par chacune des deux caméras permettent d'estimer sa position en 3D, $\tilde{x}$.

que l'on note M. Le champ magnétique dans un bâtiment non exploré est *a priori* inconnu. Il est nécessaire d'y collecter des données magnétiques, couplées à des positions et orientations, pour réaliser une cartographie magnétique. Si l'on repasse au même endroit, le magnétomètre peut aussi permettre d'améliorer l'estimation de la trajectoire, à condition que la position n'ait pas trop dérivé. Toutefois, la carte magnétique étant peu discriminante (seulement un vecteur 3D par point de l'espace) la localisation divergerait très vite. Couplé à d'autres capteurs en intérieur, il peut néanmoins être utilisé pour réduire la dérive des algorithmes de SLAM. Les mesures du magnétomètre présentent l'avantage de n'être que peu affectées par les chocs et vibration qui peuvent perturber la centrale inertielle. De plus, le magnétomètre est moins sensible à l'orientation que la caméra et des endroits visuellement similaires peuvent être discriminés grâce à lui [LZL+17].

2.1.3 Synchronisation

Dans la Section précédente, les capteurs ont été présentés indépendamment. Or ils sont utilisés ensemble pour le calcul de la localisation. Il est fondamental que leurs données soient synchronisées temporellement : les valeurs d'accélération et de rotation de la centrale inertielle permettent d'obtenir une estimation grossière du déplacement des caméras entre deux images, qui est ensuite affinée par l'optimisation des poses caméras. Si les mesures inertielles sont décalées par rapport aux mesures caméras, elles sont inutiles car elles ne retranscrivent pas le mouvement étudié par les caméras.

Au niveau du système, les caméras et la centrale inertielle n'ont pas la même horloge interne car ces capteurs sont distincts. Pour les mesures du magnétomètre, comme nous utilisons une centrale inertielle dans laquelle est aussi incluse un magnétomètre, il n'y a pas de problème de synchronisation. Les mesures du magnétomètre sont synchronisées par le constructeur avec l'horloge de l'IMU. Pour que les mesures des caméras et de l'IMU aient lieu aux mêmes instants, on choisit d'utiliser l'horloge de l'IMU comme référence et on la paramètre pour qu'elle envoie en

sortie un signal électrique à une fréquence donnée f_{ac} (par exemple 20 Hz). Ce signal électrique est transmis aux caméras par des câbles et passe par leur GPIO (General Purpose Input-Output), qui permet aux caméras de communiquer avec d'autres systèmes. Le signal arrive au niveau du port qui sert à déterminer quand une acquisition d'image doit commencer. Le temps d'arrivée du signal peut être considéré comme négligeable par sa vitesse, le peu de distance à parcourir et la vitesse d'analyse de l'information par le GPIO (moins de 1 ms). Ainsi, le début d'acquisition des images a lieu au même instant pour chaque caméra et est synchronisé avec les données IMU. Néanmoins, le temps d'acquisition des images n'est pas toujours le même car il dépend de la luminosité minimale souhaitée sur l'image et donc de ce que la caméra observe. Le fonctionnement général est représenté sur la Figure 2.3.

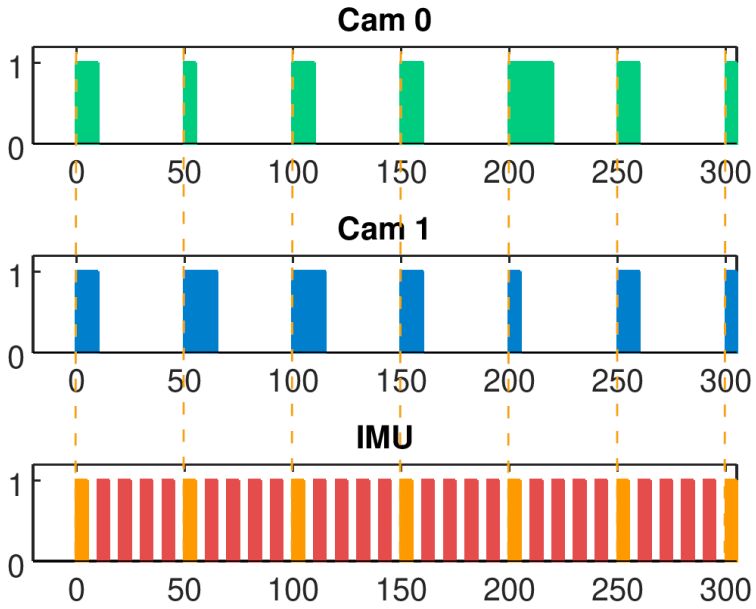


FIGURE 2.3: Schéma des mesures physiques effectuées par les différents capteurs au niveau du système (en abscisse, le temps en millisecondes). La longueur des barres indiquent le temps nécessaire pour l'acquisition d'une donnée. Les mesures de l'IMU en orange correspondent aux signaux qui sont utilisés pour déclencher des mesures caméras.

Cependant, la réception des données acquises par les caméras et l'IMU doit aussi être prise en compte : elles passent par des câbles USB avant d'être traitées par un ordinateur. Le temps de traitement, bien que faible, est suffisamment important et aléatoire pour que les données IMU et caméras avec les temps d'arrivée les plus proches ne correspondent pas toujours au même instant d'acquisition. IL correspond à un retard. Quand les mesures arrivent sur l'ordinateur, le temps d'horloge qui leur est associé est désormais celui de l'ordinateur. Il est aussi possible d'avoir accès à leur numéro de mesure pour le capteur qui les a effectuées. Ces nombres sont différents pour l'IMU et pour chaque caméra. Ils ne permettent pas de retrouver le temps réel mais, entre chaque numéro, si on connaît la fréquence du capteur, nous pouvons estimer le temps qui s'est écoulé.

Pour chaque capteur, on utilise les temps d'arrivée sur l'ordinateur comme référence. Comme les différents paquets d'information ne mettent pas tous le même temps à arriver, ces temps sont

peu réguliers. Par exemple, le temps d'arrivée d'une image peut être plus long que d'habitude, 80 ms au lieu de 50 ms en moyenne si $f_{ac} = 20$ Hz, et l'image suivante peut arriver très vite, 25 ms plus tard parce qu'elle était quasiment finalisée alors que les informations de la précédente étaient encore en train d'être transmises. On note les délais de réception $t_{rec,cam0}$, $t_{rec,cam1}$ et $t_{rec,IMU}$ pour respectivement les deux caméras (ou plus) et l'IMU. Les seuls temps que nous connaissons sont ceux fournis par l'ordinateur, t_{PC}. Les temps d'arrivée système sont erronés par rapport aux temps d'acquisition réels, comme nous pouvons le voir sur la Figure 2.4. Pour supprimer ce défaut, les temps système sont modifiés : ils sont lissés pour être plus proches de la réalité.

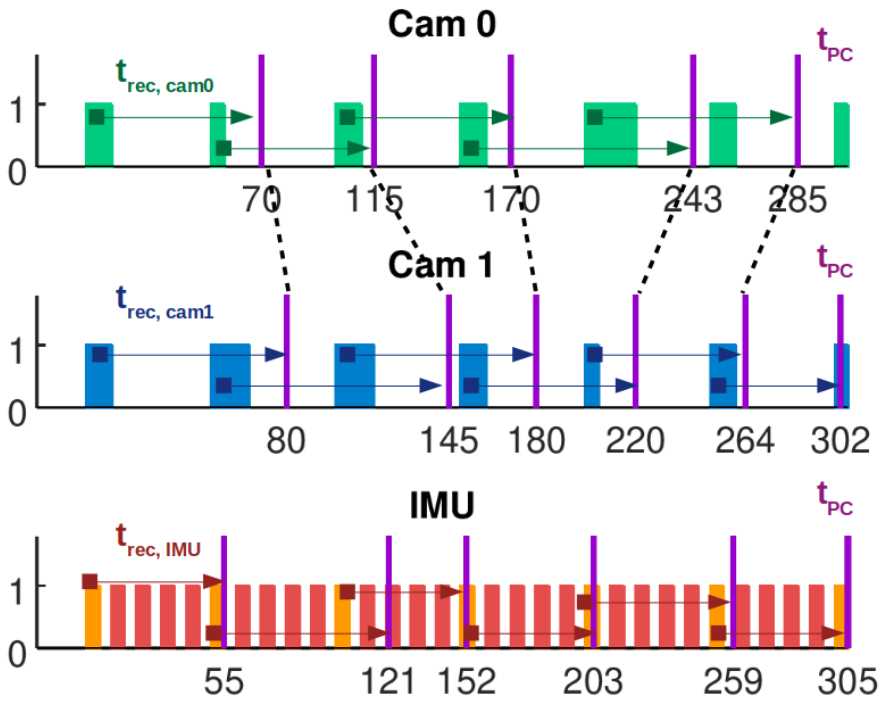


FIGURE 2.4: Schéma des mesures physiques effectuées par les différents capteurs et leur temps de transmission au système (l'ordinateur), indiqués en violet.

À ce stade, chaque suite de données capteur est cohérente mais il reste à les synchroniser. Comme la fréquence d'acquisition des caméras est relativement faible, les images pour chaque caméra arrivant dans le système avec les temps les plus proches sont considérées comme obtenues au même instant. Cependant, les données IMU sont réglées à une fréquence beaucoup plus élevée (que l'utilisateur peut choisir ; généralement, pour nos expériences, elle est de 100 ou 200 Hz). Sur la Figure 2.4, seuls les temps pour certaines données IMU sont indiqués : ils correspondent aux acquisitions des caméras. Mais toutes les valeurs intermédiaires sont aussi transmises par l'IMU et il faut retrouver laquelle est la valeur correcte parmi ce flot de nombreuses données. Il est impossible de sélectionner arbitrairement la donnée IMU la plus proche de l'arrivée de la dernière donnée caméra : les temps de réception de l'information ne sont pas comparables, une donnée IMU est beaucoup plus légère qu'une donnée caméra (uniquement 6 flottants). Ce décalage temps IMU-temps caméra est constant. Pour le déterminer, le logiciel *kalibr* est utilisé [RNS+16, LYZM14]. Il permet aussi d'évaluer les paramètres de calibration intrinsèques et extrinsèques des caméras et de l'IMU, comme décrit dans la Section 2.1.4.3. La différence de temps

t_d entre l'IMU et la caméra fait partie des paramètres à optimiser : en comparant les résultats d'un calcul de poses visuel et les données inertielles, il est possible d'estimer quelle valeur de t_d permet de minimiser les différences entre la trajectoire estimée par chaque capteur, malgré la présence de bruit sur les mesures [PGGB20]. L'estimation de t_d permet d'éviter les incohérences entre capteurs, par exemple, la détection d'un virage à 90° dans un couloir par la vision et l'IMU (via le gyromètre) doit être simultanée. Le cas contraire est souvent source de divergence des algorithmes de localisation visuelle-inertielle dans cet exemple précis.

2.1.4 Calibration

2.1.4.1 Centrale inertielle

La centrale inertielle est composée d'un accéléromètre et d'un gyromètre. Les paramètres de calibration de l'IMU sont les paramètres intrinsèques des deux capteurs, ainsi que leur écart mutuel de position et d'alignement. De plus, l'accélération a une influence sur les valeurs obtenues par le gyromètre.

Pour l'accéléromètre (ainsi que le gyromètre), les mesures sont modifiées par des facteurs d'échelle, un par axe de l'accéléromètre. Cela se traduit mathématiquement par une matrice diagonale de taille 3×3 :

$$S_a = \begin{bmatrix} s_x^a & 0 & 0 \\ 0 & s_y^a & 0 \\ 0 & 0 & s_z^a \end{bmatrix}. \tag{2.8}$$

Les axes de l'accéléromètre ne sont pas parfaitement alignés. Une matrice triangulaire inférieure modélise ces écarts :

$$T_a = \begin{bmatrix} 0 & 0 & 0 \\ t_{xy}^a & 0 & 0 \\ t_{xz}^a & t_{yz}^a & 0 \end{bmatrix}. \tag{2.9}$$

La somme de ces deux matrices permet d'obtenir une unique matrice d'échelle et d'alignement $A_a = S_a + T_a$.

La mesure de l'accélération $\hat{\mathbf{a}}_I$ correspond à l'accélération du référentiel I, qui correspond au référentiel de l'IMU (considéré comme confondu avec celui de l'accéléromètre) :

$$\hat{\mathbf{a}}_I = A_a R_{IG}(\mathbf{a}_G - \mathbf{g}_G) + \mathbf{b}_a + \boldsymbol{\eta}_a \tag{2.10}$$

avec G le référentiel terrestre, $\mathbf{g}_G$ le vecteur gravité, $\mathbf{a}_G$ la valeur réelle de l'accélération, A_a la matrice d'échelle et d'alignement, $\mathbf{b}_a$ le biais de l'accéléromètre et $\boldsymbol{\eta}_a \sim \mathcal{N}(0_3, \sigma_a I_3)$ un bruit blanc gaussien représentant le bruit du capteur. R_{IG} est la matrice de rotation pour passer du référentiel G au référentiel I.

Le biais $\mathbf{b}_a$ change au cours du temps, il suit une marche aléatoire : sa dérivée suit une loi gaussienne $\dot{b}_a = \mathcal{N}(0_3, \sigma_{wa} I_3)$.

La mesure de la vitesse de rotation ω_I présente des bruits capteurs similaires et les résultats du gyromètre peuvent être écrits comme suit :

$$\hat{\omega}_{Gyr} = A_g \omega_{Gyr} + \boldsymbol{\eta}_g + \mathbf{b}_g + A_s \mathbf{a}_G \tag{2.11}$$

avec Gyr le référentiel du gyromètre, A_g la matrice d'échelle et d'alignement, $\boldsymbol{\eta}_g$ un bruit blanc gaussien de loi $\mathcal{N}(0_3, \sigma_g I_3)$ et $\mathbf{b}_g$ une marche aléatoire telle que $\dot{\mathbf{b}}_g \sim \mathcal{N}(0_3, \sigma_{wg} I_3)$. Le terme supplémentaire $A_s \mathbf{a}_G$ traduit l'influence de l'accélération sur le gyromètre. Les valeurs de

A_s sont le plus souvent faibles (inférieures à $10^{-3}s^2.m^{-1}$). Ce terme peut être supposé négligeable si les vitesses du système restent faibles.

Les positions et orientations du gyromètre et de l'accéléromètre ne sont pas les mêmes, bien que très proches. La différence de position est estimée par la calibration en usine. Comme les deux capteurs sont fixés rigidement et ne bougent pas, il n'est pas nécessaire de la réestimer. Cependant la différence d'orientation modifie les résultats si la vitesse de rotation n'est pas associée au bon repère :

$$\hat{\omega}_I = R_{IGyr}\hat{\omega}_{Gyr}. \tag{2.12}$$

R_{IGyr} est proche de la matrice identité et correspond à la rotation du référentiel I dans le référentiel Gyr.

La majorité de ces valeurs sont supposées constantes car elles changent très peu au cours du temps. Il n'est pas nécessaire de les recalculer à chaque exécution du système : les matrices d'échelles et d'alignement, les facteurs A_s et R_{IGyr}, ainsi que les paramètres des bruits des capteurs $\boldsymbol{\eta}_a$ et $\boldsymbol{\eta}_g$, ne varient presque pas. Néanmoins, les biais $\mathbf{b}_a$ et $\mathbf{b}_g$ suivent une marche aléatoire et ne peuvent être considérés comme constants que si σ_{wa} et σ_{wg} sont faibles par rapport à l'échelle de temps étudiée. Moins les centrales intertielles sont coûteuses, plus ces valeurs sont élevées. σ_{wa} (ainsi que $\boldsymbol{\eta}_a$) s'obtient en réalisant une acquisition de longue durée (de 15 à 48 heures ou plus en fonction de la centrale) et en calculant la variance d'Allan associée, et de même pour le gyromètre. Le calcul de la variance d'Allan et la lecture du graphique obtenu sont expliqués dans [VBH12] : c'est sur ses résultats que s'appuie notre analyse. Pour l'accéléromètre utilisé dans les expériences, les valeurs obtenues par une acquisition de 50 h environ sont illustrées sur la Figure 2.5. L'intensité du bruit blanc gaussien correspond à la valeur de la tangente descendante avec une pente de coefficient $-\frac{1}{2}$ à $t = 1$ s multipliée par la racine carrée de la fréquence du capteur : $\sigma_a = \sqrt{f_{IMU}} \times 6.0 \times 10^{-4}\ \mu\text{T.s}^{1/2} = 6.0 \times 10^{-3} m.s^{-2}$. La deuxième partie de la variance d'Allan, après qu'elle a atteint son minimum et commence à augmenter régulièrement, permet d'évaluer la marche aléatoire du biais. σ_{wa} s'obtient en lisant la valeur en ordonnée sur sa ligne asymptotique (de pente $+\frac{1}{2}$) à $t = 3$ s : $\sigma_{wa} = 5.5 \times 10^{-6}\ \mu m.s^{-5/2}$. Les valeurs moyennes renvoyées par l'accéléromètre sont autour de $10 m.s^{-2}$, ce qui correspond approximativement à la force de réaction du support (le sol). L'accélération d'un piéton est inférieure à $3 m.s^{-2}$ environ pour une marche à rythme normal [DLY+16]. Le biais doit être considéré comme variable car il évolue très fortement après quelques heures (au bout d'un jour, nous pouvons observer jusqu'à 5% de différence sur les valeurs habituellement mesurées).

Pour le gyromètre, $\sigma_g = 4.0 \times 10^{-4} rad.s^{-1} = 0.023°.s^{-1}$ et $\sigma_{wg} = 8.0 \times 10^{-7} rad.s^{-3/2} = 4.58 \times 10^{-5°}.s^{-3/2}$: la conclusion est identique, il est nécessaire de recalculer la valeur du biais du gyromètre régulièrement : au bout de quelques heures, il peut être modifié de plus d'un degré.

L'algorithme de localisation doit donc recalculer les biais aussi régulièrement que possible.

2.1.4.2 Caméra

À chaque caméra sont associés plusieurs paramètres de calibration. Une partie d'entre eux sont liés aux propriétés de l'image obtenue : souvent, les objets de l'image sont déformés par rapport à leur aspect réel (voir Figure 2.6), en raison de phénomènes de distortion dus à la lentille utilisée (cet effet est particulièrement visible pour des caméras de type "fish-eye"). Les paramètres de distortion de l'image, ainsi que la distance focale et le centre optique en pixels (déjà mentionnés plus haut), sont les paramètres intrinsèques de la caméra [Hem03]. L'objectif est de connaître les différents paramètres de la matrice K définie dans l'Equation 2.6. En connaissant K et la matrice

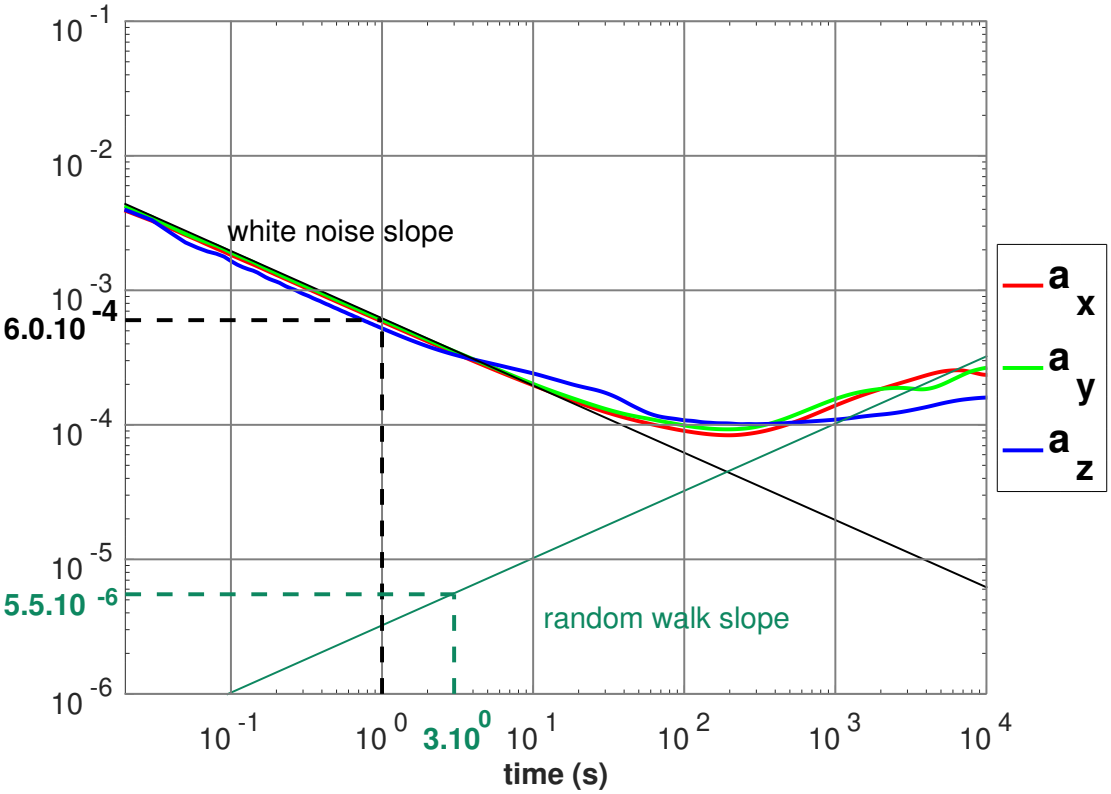


FIGURE 2.5: Variance d'Allan de l'accéléromètre (acquisition sur 50 h).

de pose permettant de passer du repère G au repère de la caméra i, C_i, nous avons les informations nécessaires pour passer des coordonnées du point 3D vers ses coordonnées dans le repère caméra, (x_{cam}, y_{cam}), et réciproquement.

$$\begin{cases} \Pi_i : \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \mapsto \begin{bmatrix} x_{cam} \\ y_{cam} \\ 1 \end{bmatrix} \\ \Pi_i \left(\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \right) = K \begin{bmatrix} R_{C_iG} & t_{C_iG} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \end{cases} \tag{2.13}$$

Quand plusieurs caméras sont utilisées, la calibration nécessite aussi de connaître les translations et rotations pour passer d'une caméra à l'autre : il s'agit des paramètres extrinsèques.

Pour obtenir les paramètres de calibration des caméras, on utilise une mire composée de carrés noir et blanc, ressemblant à un damier ou à une matrice de pixels [ML96, lEA10]. Le bon contraste de la mire permet de réaliser facilement les correspondances entre deux images au niveau des coins. Il existe plusieurs types de mires. La plus classique est un échiquier, avec un contraste noir et blanc prononcé. D'autres sont composées de carrés noirs et/ou blancs de différentes tailles, comme la aprilgrid (Figure 2.7). Pour la calibration extrinsèque, qui détermine la transformation

FIGURE 2.6: Exemple d'image obtenue avec une caméra de type "fish-eye".

entre les différentes caméras, l'estimation de l'écart gagne en précision si les différentes caméras voient simultanément la cible, avec le maximum de recouvrements.

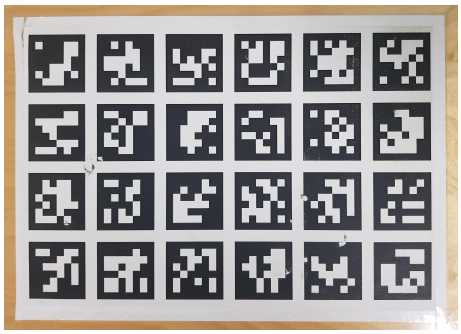


FIGURE 2.7: Exemple d'aprilgrid pour la calibration des caméras.

2.1.4.3 Calibration simultanée des caméras et de la centrale inertielle

Dans notre cas, les paramètres de calibration de la caméra sont estimés en même temps que ceux de la centrale inertielle. Ils sont obtenus par l'optimisation des résultats de calculs de poses sur une courte séquence où un grand nombre d'orientations de la caméra, tout en ayant la mire dans l'angle de vue, sont explorées. Un logiciel existe pour obtenir ces paramètres automatiquement [RNS+16] : son principe général est présenté ici.

Toutes les valeurs de calibration, à l'exception de la variance de la marche aléatoire, peuvent être estimées à l'aide d'algorithmes tels que *kalibr* [RNS+16]. *Kalibr* est adapté à différents types de modèle caméra : le modèle Pinhole, l'OCM (Omnidirectionnal camera model) et l'EUCM (Extended Unified Camera Model) [RNS+16]. Pour les caméras de type "fish-eye", les résultats sont de mauvaise qualité et une autre méthode a été créée pour résoudre ce problème ([SUS+20], logiciel *basalt*) : elle se base sur une implémentation spécifique d'un modèle double-sphère, adapté aux caméras à champ de vision large. Nous décrivons l'algorithme *kalibr* dans la suite car c'est celui que nous avons utilisé.

Le principe général de ces algorithmes repose sur l'utilisation des informations des caméras pour mesurer le déplacement et la rotation réels de la centrale inertielle. Cela permet de calculer les paramètres intrinsèques de la centrale inertielle mais aussi des caméras [FBS12, RNS+16, SUS+20]. L'accélération du système est estimée à partir de la mesure $\hat{\mathbf{a}}_I$ mais peut aussi être

obtenue à partir de la position et de la vitesse de rotation :

$$\mathbf{a}_B = R_{BI}(R_{IG}(\ddot{\mathbf{t}}_G - \mathbf{g}_G) + \lfloor \dot{\omega}_I \rfloor_\times \mathbf{t}_{IB} + \lfloor \dot{\omega}_I \rfloor_\times^2 \mathbf{t}_{IB}) \tag{2.14}$$

avec B le repère du système et $\mathbf{t}_{IB}$ la translation du référentiel B vers le référentiel I.

L'accélération angulaire est estimée à partir du paradigme de l'intégration continue par morceaux (continuous-time batch optimization paradigm, [FBS12]) : elle est supposée continue et dérivable en tout point. Ces conditions ont pour effet un aspect lisse de la courbe de l'accélération obtenue. La position au cours du temps est inconnue au départ et approximée comme une somme de fonctions splines qui ne dépendent que du temps et constituent ainsi une base de fonctions propres :

$$\begin{aligned} \mathbf{t}_G &= \sum_{j=1}^{m} c_j \phi_j \\ \theta_G &= \sum_{j=1}^{m} d_j \phi_j \end{aligned} \tag{2.15}$$

avec c_j et d_j les coefficients de pondération pour chaque fonction ϕ_j. Comme les fonctions ϕ_j sont connues, il est facile d'obtenir les dérivées premières et secondes de la position et de la rotation :

$$\begin{aligned} \ddot{\mathbf{t}}_G &= \sum_{j=1}^{m} c_j \ddot{\phi}_j \\ \dot{\theta}_G &= \sum_{j=1}^{m} d_j \dot{\phi}_j \end{aligned} \tag{2.16}$$

avec $\dot{\phi}_j$ la dérivée première et $\ddot{\phi}_j$ la dérivée seconde de la j-ème fonction spline. Les biais sont aussi modélisés comme une somme de fonctions splines.

Pour prendre en compte les erreurs liées aux mesures, on émet l'hypothèse que chaque mesure est perturbée par un processus gaussien de moyenne nulle et d'amplitude σ_v dépendant du capteur. Pour obtenir les différents paramètres de calibration, correspondant au vecteur χ, on minimise une somme composée de plusieurs éléments :

$$J = J_a + J_\omega + J_{cam} + J_{b_a} + J_{b_g}. \tag{2.17}$$

Les trois premiers éléments correspondent aux erreurs entre la mesure et son approximation. Les valeurs de ces approximations s'obtiennent à partir de la rotation et de la position estimées par l'algorithme de SLAM visuel. Pour la caméra, la mesure correspond à la reprojection des points 3D sur la caméra, à partir de l'Equation 2.7 en prenant en compte les changements de repère.

Pour des mesures de type i (c'est à dire $i = J_a$, ω, cam, b_a ou b_g), on pose :

$$J_i = \sum_{k=0}^{N} \frac{1}{\sigma_i^2} \parallel \hat{\mathbf{m}}_k - h(\chi, t_k) \parallel^2 . \tag{2.18}$$

J_{b_a} et J_{b_g} modélisent les évolutions du biais en intégrant depuis le début de l'acquisition au temps t_0 jusqu'à t_N :

$$J_b = \int_{t_0}^{t_N} \frac{1}{\sigma_b^2} \parallel \dot{\mathbf{b}}(\tau) \parallel^2 d\tau. \tag{2.19}$$

Plus l'estimation du biais est constante le long de la séquence, plus ces termes sont faibles. C'est en adéquation avec les biais réels que l'on souhaite observer : pour une courte séquence, ils sont quasiment constants.

En minimisant la fonction J par un algorithme de Levenberg-Marquadt, les coefficients χ obtenus sont proches des paramètres de calibration réels.

2.1.4.4 Magnétomètre

Les mesures du magnétomètre sont faussées par différentes sources d'erreur [VES+11]. Certaines sont dues au capteur lui-même : la mesure du capteur est différente du champ magnétique réel à cause d'un biais sur les mesures $\mathbf{b}_m$, d'une distortion des mesures selon chaque axe représentée par une matrice d'échelle et d'alignement $\mathbf{A}_m$, ainsi que d'un bruit blanc gaussien $\boldsymbol{\eta}_b$. De plus, des éléments proches du capteur, tels que des matériaux ferromagnétiques, modifient le champ magnétique mesuré. Deux types de perturbations doivent être distingués : le phénomène de *hard iron* et celui de *soft iron*. Le premier est dû à l'aimantation permanente du système dans lequel est placé le magnétomètre, entraînant un décalage $\mathbf{b}_{HI}$ alors que le *soft iron* dépend de l'orientation du système par rapport au champ magnétique et est modélisé par une matrice $\mathbf{A}_{SI}$. De plus l'alignement de l'axe du magnétomètre est imparfait, provoquant un autre biais sur les mesures $\mathbf{b}_{align}$. La mesure du champ magnétique renvoyée par le capteur est donc donnée par :

$$\begin{aligned} \hat{\mathbf{m}}_M &= \mathbf{A}_m(\mathbf{A}_{SI}\mathbf{m}_M + \mathbf{b}_{HI} + \mathbf{b}_{align}) + \mathbf{b}_m + \boldsymbol{\eta}_m \\ \text{avec } \boldsymbol{\eta}_m &\sim \mathcal{N}(\mathbf{0}, \sigma_m^2\mathbf{I}_3). \end{aligned} \tag{2.20}$$

M est le référentiel lié au magnétomètre, $\hat{\mathbf{m}}_M$ la valeur du capteur, $\mathbf{m}_M$ le champ magnétique réel dans le repère du magnétomètre et $\boldsymbol{\eta}_m$ le bruit blanc gaussien associé au capteur [VES+11].

Nous pouvons simplifier l'équation précédente sans perdre en généralité :

$$\begin{aligned} \hat{\mathbf{m}}_I &= \mathbf{A}\mathbf{m}_I + \mathbf{b}_m + \boldsymbol{\eta}_m \\ \text{avec } \dot{\mathbf{b}}_m &\sim \mathcal{N}(0_3, \sigma_{wm}) \\ \boldsymbol{\eta}_m &\sim \mathcal{N}(\mathbf{0}, \sigma_m^2\mathbf{I}_3) \end{aligned} \tag{2.21}$$

avec $\mathbf{A}$ la matrice d'échelle et d'alignement et $\mathbf{b}_m$ le biais du magnétomètre.

Pour estimer $\mathbf{A}_m$ et $\mathbf{b}_m$, la méthode de calibration la plus utilisée consiste à calibrer le magnétomètre en extérieur, à plusieurs mètres des bâtiments et éléments métalliques. Loin de toute perturbation magnétique, le champ magnétique est égal au champ magnétique terrestre. Si l'on mesure le champ dans toutes les directions, on obtient théoriquement une sphère car le champ est constant.

Cependant, le capteur présente un biais et un champ magnétique propre (le casque d'acquisition est en métal et du courant électrique circule dans les câbles) : la sphère obtenue est déformée en ellipsoïde. En connaissant $\mathbf{A}_m$ et $\mathbf{b}_m$ pour passer de l'ellipsoïde à la sphère centrée, nous pouvons corriger les déformations des mesures [RAL10]. Ces paramètres constituent la matrice de transformation et le biais du magnétomètre. Nous avons donc réalisé des calibrations en forêt (à plus de 10 m des habitations), pour être certains de ne mesurer que le champ magnétique terrestre. Nous avons acquis des valeurs du magnétomètre dans le maximum de directions possibles, en le faisant tourner sur lui-même, et le résultat obtenu est une ellipsoïde avec un décalage par rapport à l'origine (Figure 2.8).

Nous avons réalisé d'autres calibrations loin des bâtiments, pour vérifier si cette distance de 10 m est suffisante pour ne mesurer que le champ magnétique terrrestre, sans champs parasites.

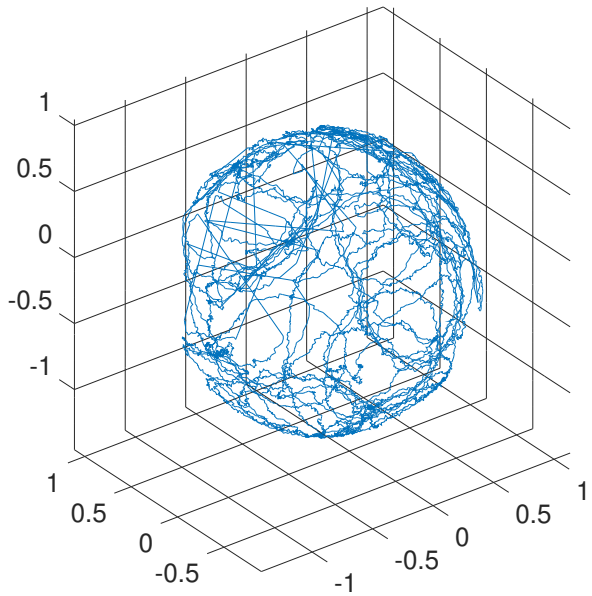


FIGURE 2.8: Mesures acquises lors d'une calibration en extérieur, loin de toute perturbation magnétique.

Certaines acquisitions ont été réalisées deux fois au même endroit. Les biais et les matrices de transformation obtenues sont toutes très proches, presque identiques (pour $\mathbf{b}_m$, la différence est de moins de 1% de l'intensité du champ magnétique terrestre ; pour $\mathbf{A}_m$, proche de l'identité, l'écart entre les estimations est de moins de 10^{-2} pour chaque coefficient).

De la même manière que pour l'accéléromètre, les grandeurs liées au bruit du capteur, σ_m et la loi de la marche aléatoire suivie par $\boldsymbol{\eta}_m$ s'obtiennent en calculant la variance d'Allan du capteur. La Figure 2.9 montre la variance d'Allan obtenue pour le magnétomètre, lors de la même acquisition de 50 h que pour la variance d'Allan de l'accéléromètre (voir la Section 2.1.4.1). Comme précédemment, l'intensité du bruit blanc gaussien est égale à la valeur de la tangente descendante avec une pente de coefficient $-\frac{1}{2}$ à $t = 1$ s multipliée par la racine carrée de la fréquence du capteur : $\sigma_a = \sqrt{f_a} \times 0.047\ \mu\text{T.s}^{1/2} = 0.33\ \mu\text{T}$. La marche aléatoire du biais σ_{wa} s'obtient en lisant la valeur en ordonnée de la ligne asymptotique de la partie droite de la courbe de pente $+\frac{1}{2}$ à $t = 3$ s : $\sigma_{wa} = 1.93 \times 10^{-4}\ \mu\text{T.s}^{-1/2}$. Le biais magnétique change donc significativement au cours du temps (jusqu'à 0.56 μT par jour) et doit être ré-estimé régulièrement.

Il est donc nécessaire de réaliser la calibration régulièrement pour mettre à jour $\mathbf{A}$ et $\mathbf{b}_m$. Par conséquent, le système doit être facilement transportable. Cette calibration doit être effectuée peu de temps avant les séquences réalisées en intérieur, quelques minutes avant quand cela est possible.

2.2 Un filtre de Kalman visuel-inertiel : le filtre de Kalman contraint multi-états (MSCKF)

Nous présentons ici le fonctionnement d'un algorithme de localisation : le filtre de Kalman multi-états (MSCKF). Il sert de fondation pour l'algorithme magnéto-visuel-inertiel que nous pro-

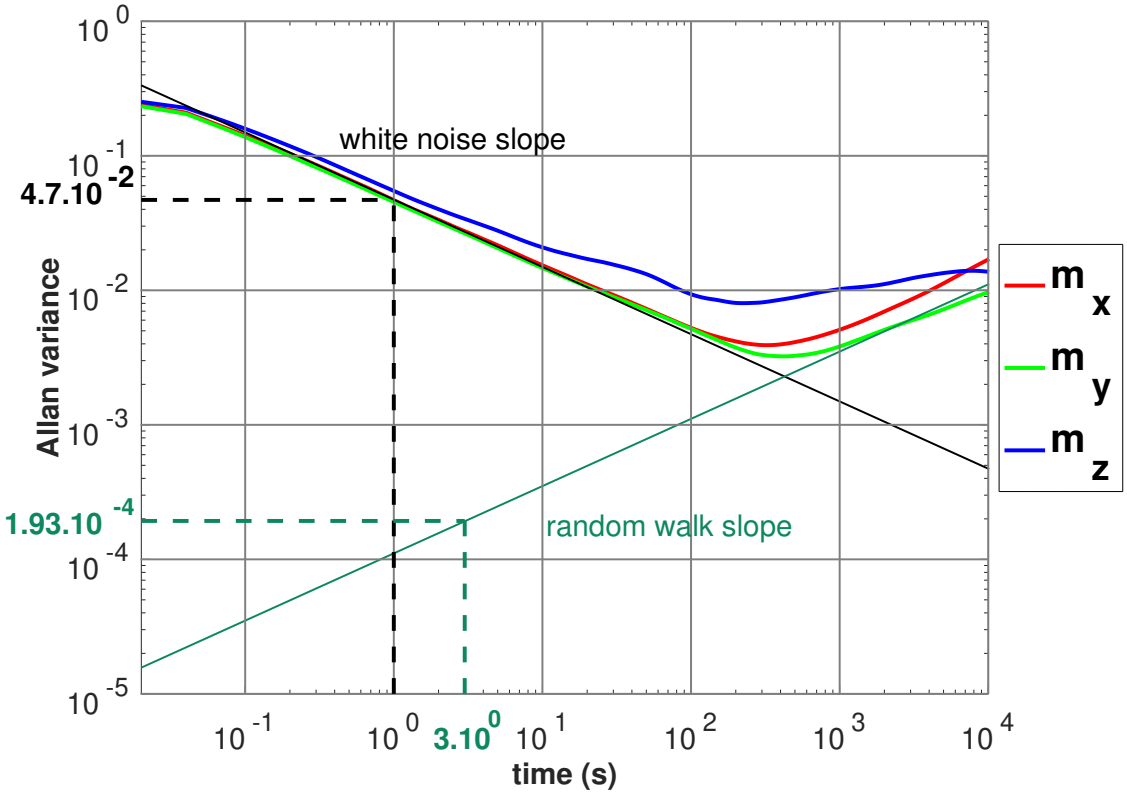


FIGURE 2.9: Variance d'Allan du magnétomètre (acquisition sur 50 h).

posons car la structure du filtre de Kalman est conçue pour permettre une insertion facile de données fournies par différents types de capteur. Son implémentation dans le cas monoculaire est résumée dans cette section car, dans nos cas d'usage, nous utilisons le MSCKF avec une centrale inertielle et une seule caméra. Après avoir introduit les notations nécessaires (Section 2.2.1), les différentes étapes de l'algorithme sont détaillées, de l'utilisation des données inertielles (Section 2.2.2) à la correction de la localisation par les mesures caméras (Section 2.2.3).

2.2.1 Vecteur d'état

Différents filtres de Kalman pour la localisation visuelle et visuelle-inertielle ont été proposés dans la littérature. Actuellement, au sein de ce groupe d'algorithmes, les meilleures performances sont obtenues par le filtre de Kalman contraint multi-états proposé dans [MR07], le MSCKF (Multi-State Constraint Kalman Filter), et ses variantes [She14, LM13]. La bonne précision de ce filtre est due à la structure de son vecteur d'état, qui est différente de celle utilisée par les précédentes méthodes. Dans les approches précédentes, les coordonnées 3D des points estimés sont présentes dans le vecteur d'état, le rendant très volumineux ([Str04]). Le problème principal qui en résulte est que l'optimisation de la pose du système ainsi que celle des points 3D ne prend pas en compte suffisamment les erreurs dues au reprojections 3D. Les poses estimées sur les images précédentes ne sont pas remises en cause puisque seule la pose courante est dans le vecteur d'état : les erreurs dues au facteur d'échelle des premières images, estimé par l'IMU, se répercutent fortement dans l'estimation des points 3D, qui doivent servir à estimer les nouvelles positions. Dans

le même temps le facteur d'échelle est réestimé par l'IMU, entraînant des incohérences dans l'optimisation de la dernière position caméra à l'aide du nuage de points. Les erreurs s'accumulent rapidement.

Pour compenser cette difficulté, le calcul des poses caméra à l'aide de la géométrie épipolaire est proposé, en calculant les contraintes entre deux poses caméra à l'aide des points 2D directement et non de leur reprojection en 3D [BB05, RJPK05]. Cependant, les temps de calcul sont longs et les optimisations ne sont réalisées que pour des paires d'images [MR07].

Le MSCKF, lui, permet d'optimiser un nombre arbitraire de positions caméras (entre 10 et 30 en pratique). À chaque nouvelle image, il utilise la méthode de triangulation, qui consiste à projeter les points 2D dans l'espace 3D. Il minimise ensuite les erreurs de reprojection pour tous les points d'intérêt observés, optimisant les positions des dernières images caméra. Mais il ne conserve pas le résultat des positions 3D obtenues car elles ne sont utiles que pour la minimisation des erreurs de position de la caméra et leur stockage prend une place importante. À chaque nouvelle image une nouvelle triangulation est réalisée. Le vecteur d'état du MSCKF ne contient que la position courante de la caméra et les positions des dernières images la précédant. Même si l'erreur de position augmente au fur et à mesure, il en résulte une localisation plus précise qui derive moins au cours du temps.

Le vecteur d'état complet se décompose en deux parties : une partie contenant l'état courant du système, $\mathbf{x}^C$, et une partie contenant les poses des dernières images caméra, $\mathbf{x}^W$:

$$\begin{aligned} \mathbf{X} &= (\mathbf{x}^C, \mathbf{x}^W), \\ \mathbf{x}^C &= (\mathrm{R}_{GI}, \mathbf{t}_{GI}, \mathbf{v}_{GI}, \mathbf{b}_g, \mathbf{b}_a), \\ \mathbf{x}^W &= (\mathrm{R}^1_{GI}, \mathbf{t}^1_{GI}, \ldots, \mathrm{R}^K_{GI}, \mathbf{t}^K_{GI}), \end{aligned} \tag{2.22}$$

avec $\mathbf{v}_{GI} \in \mathbb{R}^3$ la vitesse du référentiel I dans le référentiel G, $\mathbf{b}_g$ et $\mathbf{b}_a$sont respectivement le biais du gyroscope et de l'accéléromètre. La covariance de l'état $\mathbf{X}$ s'écrit Σ et R_{GI} représente la rotation du système dans le référentiel G. $\mathbf{x}^W$ est composé de la fenêtre des poses de K dernières images caméra. R^i_{GI} correspond à la rotation associée à la pose de la $K-i+1$ dernière image caméra. Le repère du système est considéré confondu avec celui de la centrale inertielle. L'estimation du vecteur d'état s'écrit $\hat{\mathbf{X}}$.

Les erreurs en position et sur les biais peuvent être définies comme la différence entre l'état estimé et l'état réel (vérité-terrain). Pour l'orientation, l'erreur peut être calculée de différentes manières. Pour les rotations, le produit de la rotation estimée et de l'inverse de la rotation réelle correspond à l'erreur d'orientation.

Le vecteur d'erreur est représenté comme $\widetilde{\mathbf{x}}^C = (\delta\theta_{GI}, \delta\mathbf{t}_{GI}, \delta\mathbf{v}_{GI}, \delta\mathbf{b}_g, \delta\mathbf{b}_a)$. $\delta\theta_{GI}$ est défini de façon particulière à partir de la matrice représentant l'orientation réelle du système R et son estimation $\hat{\mathrm{R}}$. Nous représentons l'erreur par un vecteur de taille 3 dans l'algèbre de Lie, $\log_{SO(3)}(\mathrm{R}\hat{\mathrm{R}}^\top)$.

2.2.2 Propagation

L'étape de propagation consiste à utiliser les mesures de la centrale inertielle pour obtenir une prédiction de l'état suivant. Il s'agit de prédire l'état courant à partir de l'état précédent sur un intervalle de temps δt correspondant à la fréquence d'acquisition images. On intègre la dérivée temporelle de l'état sur δt. L'expression de la dérivée de chaque composante de l'état s'obtient facilement :

$$\dot{\mathrm{R}}_{GI}(t) = \lfloor \omega_I(t) \rfloor_\times \mathrm{R}_{GI}(t) \tag{2.23}$$
$$\dot{\mathbf{t}}_{GI}(t) = \mathbf{v}_{GI}(t) \tag{2.24}$$
$$\dot{\mathbf{v}}_{GI}(t) = \mathbf{a}_{GI}(t) \tag{2.25}$$
$$\dot{\mathbf{b}}_g(t) = \boldsymbol{\eta}_{wg} \tag{2.26}$$
$$\dot{\mathbf{b}}_a(t) = \boldsymbol{\eta}_{wa}. \tag{2.27}$$

Cependant, la propagation de la covariance s'applique sur le vecteur d'erreur et non sur le vecteur d'état. Les erreurs liées au bruit capteur y sont présentes et l'expression de $\widetilde{\mathbf{x}}^C$ est différente de celle de $\mathbf{x}^C$.

La dérivée temporelle du vecteur d'erreur est composée de deux termes :

$$\dot{\widetilde{\mathbf{x}}}^C = \dot{\mathbf{x}}^C - \dot{\hat{\mathbf{x}}}^C \tag{2.28}$$
$$= F\hat{\mathbf{x}}^C + G\mathbf{n}_I \tag{2.29}$$

avec $\hat{\mathbf{x}}^C$ l'estimation, $\mathbf{x}^C$ la valeur réelle, $\mathbf{n}_I$ un vecteur contenant les bruits liés au capteur et Q_I la matrice de covariance associée.

$$\mathbf{n}_I = \begin{bmatrix} \boldsymbol{\eta}_g \\ \boldsymbol{\eta}_a \\ \boldsymbol{\eta}_{wg} \\ \boldsymbol{\eta}_{wa} \end{bmatrix} \tag{2.30}$$

et

$$Q_I = \begin{bmatrix} \sigma_g^2 I_3 & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & \sigma_a^2 I_3 & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & \sigma_{wg}^2 I_3 & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & \sigma_{wa}^2 I_3 \end{bmatrix}. \tag{2.31}$$

Nous obtenons F et G en calculant analytiquement la dérivée de $\widetilde{\mathbf{x}}^C$. Les mesures du gyromètre et de l'accéléromètre sont obtenues dans le référentiel I : nous les notons ω_{Gyr} et $\mathbf{a}_{acc}$.

$$\omega_{Gyr}(t) = \omega_G(t) + \lfloor \omega_I(t) \rfloor_\times \mathrm{R}_{GI}(t) + \mathbf{b}_g + \boldsymbol{\eta}_g \tag{2.32}$$

avec ω_G correspondant aux effets de rotation de la planète (et peuvent donc être négligés dans notre cas, où notre IMU présente un bruit capteur important).

Nous obtenons l'estimation de $\omega_G(t)$ à partir de la mesure $\omega_{Gyr}(t)$ et des différentes estimations des autres paramètres, qui sont les informations les plus précises dont nous disposons :

$$\hat{\omega}_G(t) = \omega_{Gyr}(t) - \lfloor \hat{\omega}_I(t) \rfloor_\times \hat{\mathrm{R}}_{GI}(t) - \hat{\mathbf{b}}_g \tag{2.33}$$

.

$\hat{\omega}_G(t)$ correspond à la dérivée de $\mathrm{R}_{GI}(t)$.

L'accéléromètre nous fournit une mesure des forces spécifiques s'exerçant sur le système, c'est à dire de toutes les forces sauf de la gravité :

$$\mathbf{a}_{acc} = \mathrm{R}_{IG}(\mathbf{a}_G - \mathbf{g}_G + 2\lfloor\omega_G\rfloor_\times \mathbf{v}_{GI} + \lfloor\omega_G\rfloor_\times^2 \mathbf{t}_{GI}) + \mathbf{b}_a + \boldsymbol{\eta}_a. \tag{2.34}$$

L'accélération étant la dérivée de la vitesse, nous avons donc la dérivée de $\mathbf{v}_{GI}$ en nous plaçant dans le référentiel G :

$$\dot{\hat{\mathbf{v}}}_{GI} = \mathrm{R}_{GI}\mathbf{a}_{acc} + \mathbf{g}_G - 2\lfloor\omega_G\rfloor_\times \mathbf{v}_{GI} + \lfloor\omega_G\rfloor_\times^2 \mathbf{t}_{GI} - \hat{\mathbf{b}}_a \tag{2.35}$$

Les expressions de F et G s'obtiennent facilement à partir des équations précédentes :

$$F = \begin{bmatrix} -\lfloor\hat{\omega}_I(t)\rfloor_\times & 0_{3\times3} & 0_{3\times3} & -I_3 & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & I_3 & 0_{3\times3} & 0_{3\times3} \\ -\hat{\mathrm{R}}_{GI}\lfloor\hat{\mathbf{a}}_I(t)\rfloor_\times & -\lfloor\omega_G\rfloor_\times^2 & -2\lfloor\omega_G\rfloor_\times & 0_{3\times3} & -\hat{\mathrm{R}}_{GI} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \end{bmatrix} \tag{2.36}$$

et

$$G = \begin{bmatrix} -I_3 & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & -\hat{R}_{IG_l}^\top & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & I_3 & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & I_3 \end{bmatrix}. \tag{2.37}$$

La matrice de covariance peut être écrite sous la forme :

$$\Sigma_{l,l} = \begin{bmatrix} \Sigma_{CC,l,l} & \Sigma_{CK,l,l} \\ \Sigma_{KC,l,l} & \Sigma_{KK,l,l} \end{bmatrix} \tag{2.38}$$

avec $\Sigma_{CC,l+1,l}$ de taille 15×15 ($\mathbf{x}^C$ est un vecteur de taille 15). Cela permet de diviser les éléments correspondant à la covariance de $\mathbf{x}^C$ et $\mathbf{x}^W$, ainsi que leur covariance mutuelle.

La propagation de la matrice de covariance complète s'obtient par intégration temporelle à partir de la matrice de covariance précédemment obtenue, à l'aide de l'algorithme de Runge-Kutta d'ordre 4. Elle peut être écrite comme suit, avec $\Phi(t_k + \delta t, t_k)$ la matrice de transition de l'état au temps t_k vers l'état suivant, à $t_k + \delta t$:

$$\Sigma_{l+1,l} = \begin{bmatrix} \Sigma_{CC,l+1,l} & \Phi(t_k + T, t_k)\Sigma_{CK,l,l} \\ \Sigma_{KC,l+1,l}\Phi(t_k + \delta t, t_k)^T & \Sigma_{KK,l,l} \end{bmatrix}. \tag{2.39}$$

$\Sigma_{KK,l+1,l+1}$ n'est pas modifiée par rapport à $\Sigma_{KK,l,l}$, tant que les images choisies restent les mêmes. Pour les autres composants, on les obtient par intégration de l'équation de Lyapounov :

$$\dot{\Sigma}_{CC,l,l} = F\Sigma_{l,l} + \Sigma_{l,l}F^\top + GQ_IG^\top \tag{2.40}$$

Pour les éléments non diagonaux,

$$\dot{\Phi}(t_l + \delta t, t_l) = F\Phi(t_l + \delta t, t_l) \tag{2.41}$$

avec, comme condition intiale, $\Phi(t_l, t_l) = I_{15}$.

A chaque étape, la plus ancienne position caméra est supprimée du vecteur d'état et la nouvelle est insérée en bas du vecteur $\mathbf{x}^W$. Supprimer une ancienne pose consiste simplement à enlever les lignes qui lui sont associées. Pour l'ajout, la jacobienne associée prend en compte le changement de repère depuis le référentiel de l'IMU vers le référentiel de la caméra C au temps actuel. Le changement d'état se modélise comme suit :

$$\Sigma_{l,l} \leftarrow \begin{bmatrix} I_{15+3(K-1)} \\ J \end{bmatrix} \Sigma_{l,l} \begin{bmatrix} I_{15+3(K-1)} \\ J \end{bmatrix}^\top . \tag{2.42}$$

avec

$$J = \begin{bmatrix} I_3 & 0_{3\times3} & 0_{3\times(9+6K)} \\ 0_{3\times3} & I_3 & 0_{3\times(9+6K)} \end{bmatrix} . \tag{2.43}$$

2.2.3 Mesure

La mise à jour du vecteur via les données images se fait par l'étape de correction du filtre de Kalman. Pour cela le modèle de mesure utilisé doit être défini, avec $\mathbf{z}$ correspondant à la mesure capteur et $\hat{\mathbf{z}}$ à son estimation par le filtre de son état courant. Le résidu se définit de la manière suivante :

$$\mathbf{r} = \mathbf{z} - \hat{\mathbf{z}} \approx H\widetilde{\mathbf{X}} + \boldsymbol{\eta} \tag{2.44}$$

avec H la jacobienne par rapport au vecteur d'état, $\widetilde{\mathbf{X}}$ le vecteur d'erreur et $\boldsymbol{\eta}$ le bruit lié aux mesures. H et $\boldsymbol{\eta}$ dépendent du capteur utilisé.

Cependant, pour la caméra, les mesures sont les positions en 2D des points 3D sur les images caméra et ne dépendent pas uniquement de l'état. $\mathbf{r}$ correspond donc à l'erreur de reprojection des points 3D qui ne sont pas dans le vecteur d'état. Le point d'intérêt 3D $\mathbf{x}^{f_j}$ est observé i fois par les i dernières images caméra. $\mathbf{x}_{C_i}^{f_j}$ correspond à ses coordonnées dans le référentiel I :

$$\mathbf{x}_{C_i}^{f_j} = R_{C_iG}(\mathbf{x}_G^{f_j} - \mathbf{t}_{GC_i}) = \begin{bmatrix} x_{C_i}^{f_j} \\ y_{C_i}^{f_j} \\ z_{C_i}^{f_j} \end{bmatrix} \tag{2.45}$$

La position $\mathbf{x}_{C_i}^{f_j}$ est estimée à partir des mesures caméra associées à ce point d'intérêt par triangulation : la fonction de reprojection lie les deux types de coordonnées. Nous ajoutons un bruit blanc gaussien $\boldsymbol{\eta}_{C_i}$ à la mesure, qui est le même pour tous les points 2D observés :

$$\mathbf{z}_{C_i}^{f_j} = \Pi_i(\mathbf{x}_{C_i}^{f_j}) + \boldsymbol{\eta}_{C_i}. \tag{2.46}$$

Π_i correspond à la fonction de reprojection pour la caméra i (voir l'Equation 2.13). Pour estimer avec autant de précision que possible les poses des caméras, nous optimisons l'erreur de reprojection pour les j points d'intérêt, chacun avec i points de vue différents. Le calcul pour obtenir la meilleure estimation de $\mathbf{x}_G^{f_j}$ est détaillé dans [MR07].

Comme les positions des points 3D influent sur le résultat, l'approximation du résidu est en réalité égale à :

$$\mathbf{r}_j^{(i)} \approx H\widetilde{\mathbf{X}}^{(i)} + H_{(i)}^{f_j}\widetilde{\mathbf{x}}_{G(i)}^{f_j} + \boldsymbol{\eta}^{(i)} \tag{2.47}$$

pour la pose caméra i et chaque point d'intérêt $\mathbf{x}_G^{f_j}$. $\mathbf{r}$ est une matrice colonne concaténant $\mathbf{r}^{(i)}$ pour chaque valeur de i et de j :

$$\mathbf{r}_j \approx \begin{bmatrix} \mathbf{r}_j^{(1)} \\ \dots \\ \mathbf{r}_j^{(n)} \end{bmatrix} = H\widetilde{\mathbf{X}} + H^{f_j}\widetilde{\mathbf{x}}_G^{f_j} + \boldsymbol{\eta} \tag{2.48}$$

avec H^{f_j} la jacobienne par rapport aux points 3D. Pour pouvoir appliquer le filtre de Kalman étendu (Extended Kalman Filter, EKF), il est nécessaire que le résidu soit mis sous la forme de l'Equation 2.44. Pour cela, on applique au résidu une transformation pour supprimer le terme dépendant des points 3D. H^{f_j} est de taille $2n \times 3$ si le nombre de reprojections 2D dans le vecteur d'état est égal à n, avec n le nombre de poses caméras étudiées. Le rang de H^{f_j} est inférieur ou égal à 3 à cause de son nombre de colonnes : son noyau est donc de taille $2n-3$. Grâce à la décomposition en valeurs singulières, nous saons qu'il existe une matrice A vérifiant $A^\top H^{f_j} = 0_{(2n-3)\times 3}$ et $A^\top A = I_{2n}$. Ainsi,

$$\mathbf{r}_j \approx A^\top (H\widetilde{\mathbf{X}}^{(i)} + H^{f_j}\widetilde{\mathbf{p}}^{f_j} + \boldsymbol{\eta}) \tag{2.49}$$

$$A^\top \mathbf{r}_j \approx A^\top H\widetilde{\mathbf{X}} + A^\top \boldsymbol{\eta}. \tag{2.50}$$

La covariance du bruit est inchangée :

$$\mathrm{Cov}(A^\top \boldsymbol{\eta}) = A^\top \mathrm{Cov}(\boldsymbol{\eta})A = A^\top A\,\mathrm{Cov}(\boldsymbol{\eta}) = \mathrm{Cov}(\boldsymbol{\eta}) \tag{2.51}$$

car $\mathrm{Cov}(\boldsymbol{\eta})$ est une constante. Après cette transformation, il devient possible d'utiliser les équations de l'EKF.

Nous pouvons ensuite appliquer les équations de correction du MSCKF :

$$L = \Sigma_{l+1|l} H^\top (H\Sigma_{l+1|l} H^\top + R_{\boldsymbol{\eta}})^{-1} \tag{2.52}$$

$$\Sigma_{l+1|l+1} = (I_{15+6K} - LH)\Sigma_{l+1|l} \tag{2.53}$$

$$\Delta\mathbf{X} = Lr \tag{2.54}$$

$$\mathbf{X} = \mathbf{X} \otimes \Delta\mathbf{X} \tag{2.55}$$

avec $R_{\boldsymbol{\eta}}$ la covariance du bruit du capteur visuel. Une décomposition QR de la matrice H rend le calcul de l'inverse de $H\Sigma_{l+1|l}H^\top + R_{\boldsymbol{\eta}}$ beaucoup plus rapide. Nous pouvons ensuite traiter la mesure suivante tant que de nouvelles données arrivent.

2.3 SLAM visuel-inertiel par ajustement de faisceaux et fermeture de boucles

À part le MSCKF et ses variantes, d'autres algorithmes visuels-inertiels ont été développés et sont actuellement les plus performants de l'état de l'art : ce sont des SLAM visuels-inertiels basés sur l'optimisation de graphes par ajustement de faisceaux. C'est donc ce type d'algorithme, avec 4 caméras, qui nous permet d'obtenir les vérités-terrain pour nos séquences. Ils intègrent aussi des fermetures de boucle visuelles qui permettent de limiter la dérive de la localisation au cours du temps. Le principe global est décrit dans la Section 2.3.1, puis l'algorithme de fermetures de boucle à l'aide de cartes visuelles est expliqué (Section 2.3.2).

2.3.1 Optimisation de graphe visuel-inertiel

Les images fournies par une caméra à différents instants peuvent être utilisées pour estimer le déplacement du système, comme mentionné dans la Section 2.1.2.2. Le principe d'un SLAM visuel [CER+21] consiste à optimiser les K dernières poses de la caméra ainsi que les coordonnées des points 3D en minimisant la somme des erreurs de reprojection, contrairement au MSCKF qui ne conserve pas les positions des points 3D (voir la Section 2.2). Mathématiquement, cela se traduit comme suit :

$$\mathbf{r}_{i,j} = \mathbf{z}_{i,j} - \Pi(P_{C_iG} \oplus \mathbf{x}_G^j) \tag{2.56}$$

avec C_i le référentiel de la caméra d'indice i, $\mathbf{z}_{i,j}$ les coordonnées du point j sur l'image caméra i, $\mathbf{x}_G^j$ ses coordonnées 3D associées.

Les variables qui peuvent être modifiées pour l'optimisation sont les positions et orientations des caméras dans la fenêtre locale, c'est à dire les K dernières images-clés de la caméra jusqu'à la position courante, et les positions des points 3D. Les images-clés sont sélectionnées de manière à être suffisamment distantes entre elles afin de maximiser la qualité de l'étape de triangulation mais sans être trop distantes pour pouvoir assurer leur mise en correspondance. Pour rendre le calcul plus précis, les images-clés antérieures qui observent les mêmes points 3D que la fenêtre locale sont également utilisées mais leurs poses sont fixées au cours de l'optimisation. Ces images passées permettent d'avoir une meilleure approximation de la position réelle des points 3D. Le fonctionnement de l'ajustement de faisceaux visuels est illustré Figure 2.10. Afin de pallier aux faux appariements pouvant entraîner des erreurs de reprojections élevées il est commun d'utiliser un estimateur robuste, par exemple Huber, pour limiter l'impact de ces mesures erronées dites outliers.

Au début, quand l'algorithme s'initialise, aucun point 3D n'étant connu, le SLAM est initialisé en calculant les correspondances 2D-2D à l'aide de la géométrie épipolaire. Après quelques images, il est possible d'utiliser les projections 3D des points pour corriger la localisation : un premier nuage 3D de points s'obtient par triangulation sur les images caméra déjà obtenues.

Pour un SLAM visuel-inertiel, le principe est similaire mais des contraintes supplémentaires sont incluses entre les poses caméras : il s'agit des mesures de la centrale inertielle et de son biais, à partir desquels le déplacement de la caméra entre deux images-clés peut être prédit (Figure 2.11). Le vecteur d'état est modifié : on lui ajoute la vitesse du système, qui correspond à l'intégration de l'accélération, qui est estimée par l'accéléromètre et aussi corrigée par les mesures des caméras, et les biais de la centrale inertielle, qui ne sont pas connus *a priori*. L'étape d'estimation du déplacement de la caméra à partir des données inertielles est appelée l'étape de pré-intégration [FCDS15]. Elle consiste à intégrer l'accélération et la vitesse de rotation du système sur l'intervalle de temps séparant les deux images, obtenant ainsi une estimation de la vitesse du système $\Delta\mathbf{v}_{i-1,i}$, de sa position $\Delta\mathbf{p}_{i-1,i}$ et de son orientation $\Delta R_{i-1,i}$. La valeur des biais de la centrale inertielle dans le graphe est réestimée en fonction de l'algorithme de minimisation : c'est un aspect essentiel pour l'utilisation des données IMU car ses biais évoluent très rapidement et doivent constamment être réestimés. $\mathbf{r}_{i-1,i}^{IMU}$ correspond à l'erreur résiduelle sur la rotation, la position et la vitesse de la caméra ente les images $i-1$ et i, qui doit être minimisée par le graphe. $\Delta t_{i-1,i}$ est la durée séparant l'acquisition des images $i-1$ et i. Pour les équations suivantes, nous simplifions les notations R_{GI}, $\mathbf{p}_{GI}$ et $\mathbf{v}_{GI}$ en R, $\mathbf{p}$ et $\mathbf{v}$.

$$\mathbf{r}^{IMU}_{i-1,i} = \begin{bmatrix} \mathbf{r}^{\Delta R}_{i-1,i} & \mathbf{r}^{\Delta \mathbf{v}}_{i-1,i} & \mathbf{r}^{\Delta \mathbf{p}}_{i-1,i} \end{bmatrix}^\top \tag{2.57}$$

$$\mathbf{r}^{\Delta R}_{i-1,i} = \log_{SO(3)}(\Delta R^\top_{i-1,i} R^\top_{i-1} R_i) \tag{2.58}$$

$$\mathbf{r}^{\Delta \mathbf{v}}_{i-1,i} = R^\top_{i-1}(\mathbf{v}_i - \mathbf{v}_{i-1} - \mathbf{g}\,\Delta t_{i-1,i}) - \Delta \mathbf{v}_{i-1,i} \tag{2.59}$$

$$\mathbf{r}^{\Delta \mathbf{p}}_{i-1,i} = R^\top_{i-1}(\mathbf{p}_i - \mathbf{p}_{i-1} - \mathbf{v}_{i-1}\Delta t_{i-1,i} - \frac{1}{2}\mathbf{g}\delta t^2_{i-1,i}) - \Delta \mathbf{p}_{i-1,i} \tag{2.60}$$

La somme à minimiser pour optimiser les K dernières poses du système est modifiée et intègre les contraintes supplémentaires de la centrale inertielle. Pour l points 3D, elle vaut :

$$\sum_{i=1}^{K} \| \mathbf{r}^{IMU}_{i-1,i} \|_{\Sigma^{-1}_{IMU,i-1,i}} + \sum_{j=1}^{l} \sum_{i \in \mathscr{W}} \rho_{Hub} \| \mathbf{r}_{i,j} \|_{\Sigma^{-1}_{i,j}} \tag{2.61}$$

avec $\mathscr{W}$ l'ensemble des poses caméra de la fenêtre locale, $\Sigma_{IMU,i-1,i}$ la matrice de covariance sur le bruit estimé pour synthétiser l'erreur pour plusieurs mesures de la centrale pour $\mathbf{r}^{IMU}$ et $\Sigma_{i,j}$ la matrice de covariance associée de l'observation du point j pour l'image caméra i. ρ_{Hub} permet de limiter l'influence des appariements erronés entre différentes images.

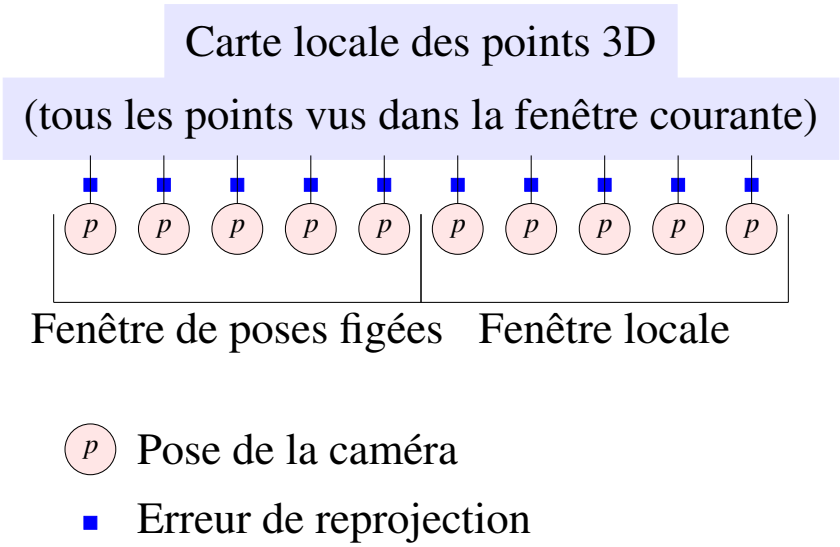


FIGURE 2.10: Schéma du SLAM visuel par ajustement de faisceaux. La fenêtre figée correspond aux poses caméras qui visualisent les mêmes points 3D que dans la pose actuelle. La dernière pose figée est la pose directement antérieure à la première de la fenêtre locale.

2.3.2 Réutilisation de cartes visuelles et fermeture de boucles

Le SLAM visuel-inertiel présenté dans la partie précédente dérive au cours du temps car les erreurs s'accumulent au fur et à mesure. Le calcul des positions caméra dans la fenêtre locale dépend des positions de la fenêtre de poses figées, qui ont été calculées auparavant : les erreurs sur les positions des points 3D augmentent, et donc sur leur reprojection 2D.

Les fermetures de boucle visuelle limitent ce phénomène. Elles peuvent être traitées de façon efficace en fusionnant les informations similaires du graphe visuel-inertiel, obtenues lors de deux passages au même endroit [RMA14].

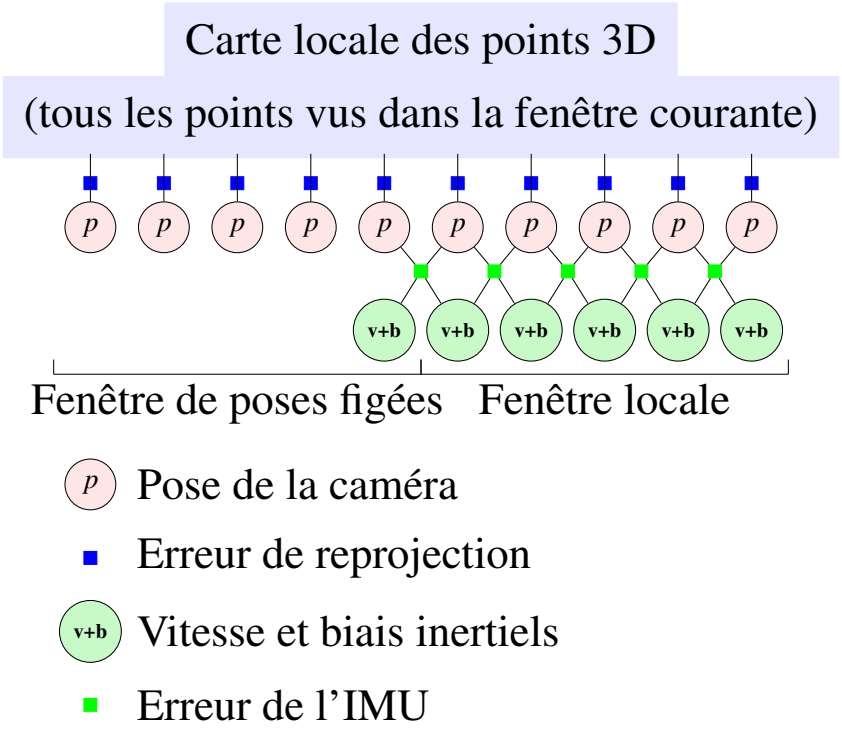


FIGURE 2.11: Schéma du SLAM visuel-inertiel (par ajustement de faisceaux contraints par les mesures inertielles).

L'image courante U_a est comparée à chaque fois avec les images-clés d'une carte visuelle déjà créée qui est stockée et dont les éléments déjà inclus sont non modifiables. Les K dernières images-clés obtenues avant U_a forment une autre carte, une carte active, dont les positions sont acquises plus récemment que les données de la carte stockée et sont susceptibles d'être modifiées. Pour trouver l'image de la carte la plus similaire à U_a, l'algorithme de sacs de mots DBoW2 [GLT12] est souvent utilisé. Les images directement antérieures à U_a ne sont pas incluses : elles ne permettent pas de faire de larges fermetures de boucles. Pour chaque image susceptible de représenter le même endroit que l'image actuelle, que l'on note U_m, nous sélectionnons d'autres images-clés qui ont un fort coefficient de covisibilité avec elles. La covisibilité est d'autant plus élevée que les deux images observent la même zone de l'espace, avec de nombreux points identiques visibles sur les deux images.

Une estimation de la matrice de transformation du référentiel de l'image U_m à U_a, P_{am}, est obtenue par l'algorithme RANSAC ([FB81]) qui aligne au mieux les points 3D observés par les deux images. Si l'erreur d'au moins une des comparaisons U_m-U_a est en-dessous d'une valeur limite, cela signifie que nous avons peut-être trouvé une image candidate pour la fusion de la carte actuelle avec la carte stockée. Nous appliquons la même transformation P_{am} à l'ensemble des points de la fenêtre locale dont U_a est le dernier élément. On optimise ensuite P_{am} pour qu'elle soit la plus précise possible.

Pour vérifier qu'il s'agit effectivement d'un cas où nous pouvons utiliser la fusion de cartes et non d'un faux positif entre U_a et U_m, ce processus est répété sur plusieurs images consécutives. S'il s'agit d'un cas isolé, la relocalisation est considérée comme erronnée et n'est pas prise en compte. Si au contraire la transformation est valable pour plusieurs images, nous pouvons fusionner les deux cartes.

La fusion commence par l'alignement des fenêtres covisibles de U_m dans le repère de U_a. Des liens sont créés entre les fenêtres covisibles de U_a et de U_m et leurs points 3D en commun sont fusionnés pour éviter les doublons. Par ajustement de faisceaux, les dernières positions de la caméra et des images-clés non fixées sont corrigées. Ces corrections ne s'appliquent qu'à un certain nombre d'images et non à toute la boucle quand celle-ci est très longue, à cause du coût élevé de l'optimisation pour un SLAM visuel-inertiel.

Dans le cas de la création de nos vérités-terrain, nous exécutons cet algorithme avec quatre caméras et une centrale inertielle. Le SLAM visuel-inertiel est exécuté sans carte visuelle préconstruite car l'environnement est inconnu. Le résultat obtenu est une trajectoire et une carte visuelle associée. En réalisant une optimisation globale de la carte et des fermetures de boucles, les positions du système sont calculées très précisément.

Conclusion

Dans cette partie, nous avons présenté les deux types actuels de méthodes de localisation en intérieur visuelles-inertielles les plus fréquemment utilisées. Le SLAM visuel-inertiel par ajustement de faisceaux et fermeture de boucles est l'algorithme le plus précis de l'état de l'art et nous sert à créer les vérités-terrain pour nos séquences avec les informations de 4 caméras et d'une centrale inertielle. L'objectif de cette thèse est de rajouter les informations du magnétomètre pour obtenir des trajectoires plus précises. Nous allons donc étudier dans les chapitres suivants la création de cartes du champ magnétique puis la manière d'inclure les données du magnétomètre dans le MSCKF, qui utilise une structure basée sur le filtre de Kalman et est donc particulièrement adapté à l'ajout de nouveaux capteurs.

Chapitre 3

Le champ magnétique : modélisation et cartographie

3.1 Modélisation usuelle du champ magnétique 42
3.1.1 Le champ magnétique en intérieur 42
3.1.2 Caractéristiques physiques du champ magnétique 43
3.1.3 Interpolation du champ par processus gaussiens 44
3.1.4 Création de carte magnétique 47
3.1.5 Adaptation à la création de carte en ligne 51
3.2 Impact des erreurs de localisation dans l'estimation du champ 52
3.3 Processus Gaussiens avec entrées et sorties simultanément bruitées 54
3.3.1 Méthodes de l'état de l'art 54
3.3.2 Introduction du problème 55
3.3.3 Étude de la loi de probabilité d'une fonction avec entrées et sorties simultanément bruitées 56
3.3.4 Adaptation à un Processus Gaussien à rang réduit 65
3.3.5 Adaptation pour un Processus Gaussien itératif à rang réduit : RRONIG (Reduced-Rank Online Noisy Input GP) 68
3.4 Etude du comportement de différents modèles de GP avec entrées et sorties bruitées sur des données simulées 70
3.5 Carte obtenue à partir de mesures avec positions et orientations bruitées 74
3.5.1 Insertion de l'erreur de rotation 74
3.5.2 Processus Gaussien à rang réduit avec positions et orientations bruitées . 81

Introduction

Dans ce chapitre, la modélisation et la cartographie du champ magnétique en intérieur sont étudiées. Dans un premier temps, nous montrons qu'il est possible d'utiliser des cartes du champ magnétique pour se repérer. Les caractéristiques physiques principales du champ magnétique sont présentées, ainsi qu'un modèle d'interpolation par processus Gaussien (Section 3.1). Ce modèle utilise des couples constitués d'une pose associée à une donnée magnétique pour construire une

carte complète du champ : les poses correspondent aux entrées, ou antécédents, et les données magnétiques aux sorties de la fonction associée au champ magnétique, B, que l'on souhaite modéliser. Cependant, la modélisation classique par processus Gaussiens (GP, Gaussian process) utilisée généralement pour créer les cartes présente des défauts, comme la non-prise en compte du fait que les positions sont estimées par le système et ont donc une erreur associée (Section 3.2). C'est pourquoi nous avons travaillé sur une nouvelle approche permettant d'insérer le bruit sur les données d'entrée. Elle est expliquée en Section 3.3. Les résultats sont comparés avec l'état de l'art (Section 3.4) et une version avec l'erreur de rotation en plus de l'erreur de position est décrite en Section 3.5.

3.1 Modélisation usuelle du champ magnétique

Dans cette section, nous montrons que le champ magnétique varie peu temporellement dans des bâtiments tels que des bureaux ou des habitations (Section 3.1.1). Son importante variation spatiale permet de créer des cartes magnétiques robustes dans le temps et utiles pour s'y localiser. Dans un premier temps, les caractéristiques physiques du champ magnétique que nous utilisons sont décrites en 3.1.2. Puis la modélisation du champ magnétique en intérieur par un processus Gaussien est présentée en Section 3.1.3.

3.1.1 Le champ magnétique en intérieur

Le champ magnétique peut être modélisé par un vecteur en trois dimensions. En extérieur, s'il n'y a aucune perturbation environnante, il est égal au champ magnétique terrestre. Les perturbations correspondent à tous les objets dont la présence affecte le champ. Il s'agit essentiellement d'objets métalliques et d'aimants, par exemple les moteurs et carrosseries de voitures, les infrastructures (poutres métalliques) de bâtiments, les ordinateurs et les câbles électriques. Chacun de ces objets perturbe le champ de façon différente.

En extérieur, le champ magnétique étant constant, le magnétomètre nous permet de connaître le cap précisément, de la même manière qu'une boussole. En fonction du lieu géographique, il présente des valeurs différentes mais sur quelques kilomètres ces variations sont négligeables. En France, il vaut en moyenne 48 μT. Il évolue de façon très lente au cours du temps (variation de 1 μT entre 1948 et 1997). On peut donc considérer qu'il est constant dans le temps sur une zone de quelques kilomètres carrés. Le champ magnétique en intérieur, au contraire, possède une grande variabilité spatiale, ce qui ne permet pas de l'utiliser facilement pour trouver le cap. Mais, étudiée sous une autre perspective, cette variabilité permet de caractériser l'environnement, en créant des cartes du champ magnétique, plus pérennes dans le temps que des cartes visuelles. D'après les expériences réalisées dans [LZL+17, CDS+11], au bout de 3 ou 6 mois, le champ magnétique mesuré dans des bâtiments est presque identique à celui qui avait été mesuré par le passé. Il est donc possible de créer des cartes du champ magnétique, qui peuvent être utilisées pour améliorer la localisation. Le détail de la modélisation et l'utilisation des cartes sont détaillés dans la suite.

Dans le cas de bâtiments modernes, dont une grande partie de la structure se compose de métal, les contrastes de champ d'une zone à une autre sont plus importants que pour des bâtiments anciens. Certains objets perturbant le champ magnétique sont mobiles ou peuvent être arrêtés / redémarrés (par exemple des ordinateurs) et donc modifient au cours du temps la carte magnétique. Cependant, la majorité d'entre eux n'ont qu'un impact très limité. En effet, par exemple, à plus de 15 cm d'une montre, la déviation du champ est inférieure à 0.3 μT, sachant que son intensité totale atteint très rarement des valeurs inférieures à 30 μT au sein d'un bâtiment. Cela signifie donc qu'à

15 cm d'une montre, son impact sur le champ magnétique correspond à moins d'1% d'altération. Pour un ordinateur, la perturbation est un peu plus forte mais le plus souvent négligeable : pour un changement de moins de 0.3 μT, la distance caractéristique est de 30 cm ([LGDR12, CDS+11]). Cependant, pour certains objets, les perturbations engendrées ne sont pas négligeables. C'est le cas par exemple des ascenseurs : à une distance de 2 m, l'amplitude du champ magnétique est perturbée de 2%. Deux ascenseurs proches peuvent influencer le champ sur 7 m ([CDS+11]). Dans les zones couvertes par des ascenseurs, l'hypothèse de constance au cours du temps du champ magnétique n'est plus valable.

Ainsi, la majorité des modifications (déplacement d'ordinateurs, téléphones) qui ont lieu dans un bâtiment n'ont qu'un impact restreint et ne nécessitent pas de mettre à jour la carte magnétique de l'environnement. Cependant, les zones proches des ascenseurs présentent une variabilité magnétique significative qui risquent de poser des problèmes importants pour se localiser. Nous en déduisons que dans la majorité des cas de la vie quotidienne nous pouvons utiliser une carte magnétique pour se repérer dans l'espace car le champ magnétique est stable temporellement, sauf au niveau des ascenseurs et en cas de grande proximité avec des éléments générant un champ très localisé (montres, téléphones, ordinateurs, moteurs).

Le champ magnétique apporte des informations différentes de celles que fournissent les autres capteurs. Il permet de distinguer des zones que d'autres capteurs ne peuvent pas discriminer. Par exemple, dans un couloir avec plusieurs portes de bureau à intervalles identiques, il est impossible pour un capteur visuel de déterminer devant quelle porte il se situe. Le champ magnétique, lui, aura souvent une empreinte différente pour chacun de ces emplacements car il est influencé par tous les objets magnétiques comme les structures et les tuyaux se trouvant dans les plafonds. De plus, le magnétomètre ne présente pas de déni de service dans les situations où les caméras et la centrale intertielle sont mis en défaut (pour les caméras : flou, pas de lumière, pas de textures ; pour la centrale inertielle : chocs, vibrations).

Perturbations du champ magnétique

Eléments métalliques :

1. Perturbations sur une faible distance ($\sim$ 30 cm) : montres, téléphones, ordinateurs
2. Perturbations sur une longue distance (quelques mètres) : ascenseurs, charpente métallique

Conclusion : La cartographie magnétique d'un bâtiment composé de bureaux ou d'habitations est stable au cours du temps, sauf pour de rares exceptions (comme les ascenseurs). On peut donc utiliser ces cartes pour se localiser.

3.1.2 Caractéristiques physiques du champ magnétique

De plus, le champ magnétique possède des propriétés physiques particulières : elles peuvent être utilisées pour améliorer la qualité des cartes du champ magnétique. Les équations de Maxwell sont vérifiées, en particulier, dans le cas quasi-statique (pas de variations temporelles) :

$$\begin{aligned} \nabla \times \mathbf{B} &= \mu_0 \mathbf{J} \\ \nabla \cdot \mathbf{B} &= 0 \end{aligned} \tag{3.1}$$

avec $\mathbf{J}$ la densité de courant et $\mathbf{B}$ le champ magnétique, $\nabla\times$ symbolise le rotationnel et $\nabla\cdot$ la divergence.

La densité de courant est composée d'une partie constante, $\mathbf{J}_m$, intrinsèque aux matériaux métalliques, et d'une partie variable, $\mathbf{J}_f$, souvent générée par un courant électrique ([Wah15]). $\mathbf{M}$ est appelée magnétisation et son rotationnel correspond à $\mathbf{J}_m$.

$$\begin{aligned} \mathbf{J} &= \mathbf{J}_m + \mathbf{J}_f \\ \mathbf{J}_m &= \nabla \times \mathbf{M} \end{aligned} \tag{3.2}$$

En pratique, $\mathbf{J}_f$ est négligeable devant $\mathbf{J}_m$: $\mathbf{J}_f$ est généré par les variations de courant électrique, dans les câbles par exemple, et sa valeur est faible devant celle de $\mathbf{J}_m$. $\mathbf{J}_m$ est le courant généré par le déplacement des électrons dans le matériau étudié. Dans tous les matériaux non magnétiques, la magnétisation $\mathbf{M}$ est nulle [WKTG13]. Or nous mesurons le champ magnétique de l'air environnant le magnétomètre : sous ces hypothèses, $\mathbf{M} = \mathbf{0}_3$.

Le rotationnel de $\mathbf{B}$ est alors égal à : $\nabla \times \mathbf{B} = \mathbf{0}_3$.

Maintenant que nous savons que le champ magnétique peut être utilisé pour créer une carte permettant de se localiser, comment créer cette carte quand nous ne disposons que d'un nombre d'échantillons limité et discret ?

3.1.3 Interpolation du champ par processus gaussiens

Pour construire une carte magnétique, on utilise les mesures magnétométriques obtenues dans le lieu étudié. Il s'agit d'acquisitions discrètes et non continues ([FAL+13]) : il est donc nécessaire de réaliser des interpolations pour obtenir une carte complète.

Une première méthode consiste à réaliser une interpolation linéaire : à chaque point, la valeur de son champ magnétique correspond à la somme pondérée des valeurs du champ de ses plus proches voisins ([WWP+19]). Cette méthode s'applique très facilement en 2D, si les valeurs obtenues forment un quadrillage régulier de la zone. Mais cette méthode est bien plus difficile à appliquer dans le cas d'une trajectoire qui n'est pas parallèle aux murs et qui n'est pas rectiligne. De plus elle ne prend pas en compte les bruits du capteur.

Une autre façon de créer la carte s'appuie sur l'utilisation de processus gaussiens ([Val18], [SKW+15]). Un processus gaussien est une fonction restreinte à une dimension dans l'espace d'arrivée. Pour prédire chacune des composantes du champ, nous pouvons donc utiliser trois processus gaussiens, selon l'axe x, l'axe y et l'axe z [VHKR11]. Une partie du champ estimé correspond au champ magnétique terrestre, qui est constant. Or on ne connaît pas *a priori* son orientation par rapport à la carte. On décide donc de supposer que notre processus gaussien a une moyenne nulle au départ, mais une variance de taille importante, κ. Le champ magnétique est un vecteur en trois dimensions avec $\mathbf{B} : \mathbb{R}^3 \to \mathbb{R}^3$. Chaque composante peut donc être considérée comme un processus gaussien indépendant. Pour un point de l'espace $\mathbf{x}_i \in \mathbb{R}^3$,

$$\mathbf{B}_1(\mathbf{x}_i) \sim GP(0, \kappa_1(\mathbf{x}_i, \mathbf{x}_i)), \tag{3.3}$$

$$\mathbf{B}_2(\mathbf{x}_i) \sim GP(0, \kappa_2(\mathbf{x}_i, \mathbf{x}_i)), \tag{3.4}$$

$$\mathbf{B}_3(\mathbf{x}_i) \sim GP(0, \kappa_3(\mathbf{x}_i, \mathbf{x}_i)). \tag{3.5}$$

Cependant, les valeurs $\mathbf{y}_i$ du capteur (avec i le numéro de la mesure) ne sont pas parfaites et un bruit blanc gaussien modélise ces déviations : $\varepsilon_i \sim N(\mathbf{0}_3, \sigma_m^2 I_3)$. σ_m correspond à l'écart-type du bruit sur les données. Il doit être en théorie égal au bruit du capteur, σ_{noise}.

$$\mathbf{y}_i = \mathbf{B}(\mathbf{x}_i) + \varepsilon_i \tag{3.6}$$

$\mathbf{y}_i$ est un vecteur à 3 dimensions. Nous notons $\mathbf{y}_i^d$ sa valeur selon la d-ième composante. $\mathbf{y}_i^d$ est l'addition de deux processus gaussiens et par conséquent est aussi un processus gaussien. Le calcul du modèle s'appuie sur les observations $\mathbf{y}_1^d, \mathbf{y}_2^d ... \mathbf{y}_n^d$ aux emplacements $\mathbf{x}_1, \mathbf{x}_2 ... \mathbf{x}_n$. $\mathbf{x}_i$ correspond au point d'apprentissage i. On cherche à prédire les valeurs du modèle au niveau du point de prédiction $\mathbf{x}_*$, son espérance et sa covariance.

$\mathbf{k}_*$ un vecteur colonne contenant $\kappa(\mathbf{x}_*, \mathbf{x}_i)$ pour tout i ; $vec(\mathbf{y}^d)$ un autre vecteur colonne contenant $\mathbf{y}_1^d$ ... $\mathbf{y}_n^d$. K est une matrice pré-calculée avec les coefficients $K_{i,j} = \kappa(\mathbf{x}_i, \mathbf{x}_j)$ pour tous les points d'apprentissage.

$$\begin{cases} \mathbb{E}[B_d(\mathbf{x}_*)] = \mathbf{k}_*^\top (K + \sigma_m^2 I_n)^{-1} vec(\mathbf{y^d}) \\ \mathbb{V}[B_d(\mathbf{x}_*)] = \kappa(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^\top (K + \sigma_m^2 I_n)^{-1} \mathbf{k}_* \end{cases} \tag{3.7}$$

La forme de la fonction κ choisie pour modéliser le champ magnétique est une exponentielle quadratique (d'autres formules seraient possibles). L'indice SE indique que le modèle utilisé est une exponentielle quadratique (Squared Exponential). Chaque coefficient κ_{SE} est défini ainsi :

$$\kappa_{SE}(\mathbf{x}_i, \mathbf{x}_j) = \sigma_{SE}^2 exp\left(-\frac{\left|\left|\mathbf{x}_i - \mathbf{x}_j\right|\right|^2}{2l_{SE}^2}\right) \tag{3.8}$$

Les coefficients $\kappa_{SE}(\mathbf{x_i}, \mathbf{x_j})$ permettent de pondérer l'impact des valeurs d'apprentissage. Les mesures les plus proches d'un point $\mathbf{x}$ sont celles dont les coefficients $\kappa_{SE}(\mathbf{x_i}, \mathbf{x_j})$ sont les plus grands (proches de 1). Ainsi, plus une mesure est proche d'un point, plus elle influencera la prédiction en ce point. L'espérance en un point sera déterminée de façon pondérée par les valeurs magnétiques obtenues pour les zones voisines. La valeur de l_{SE} détermine le degré d'interaction entre deux points : plus l_{SE} est petit, plus la distance à partir de laquelle les points peuvent être considérés comme non corrélés (avec une variance proche de 0) sera petite. σ_{SE}^2 est la variance du modèle initial pour les points étudiés : la corrélation est maximale pour deux points identiques.

Au niveau des points d'apprentissage, si le capteur est supposé parfait, $\sigma_m = \sigma_{noise} = 0$, il renvoie exactement la valeur mesurée par le capteur et la variance vaut $\mathbf{0}_{3\times3}$. Pour un capteur réel, la valeur du modèle n'est pas toujours égale à la mesure du capteur en ce point, à cause du bruit du capteur : le modèle prend en compte cette déviation et y associe une variance qui permet de connaître l'intervalle dans lequel doit se situer la vraie valeur.

La Figure 3.1 présente le modèle d'un processus gaussien pour une dimension. Plus on s'éloigne des points d'apprentissage, plus l'incertitude du modèle devient grande. On voit que l'épaisseur de l'intervalle de confiance est lié à la distance entre l'interpolation effectuée au point $\mathbf{x}$ et la distance aux points d'apprentissage les plus proches.

Cette approche avec un processus Gaussien par axe restitue mieux les erreurs liées au modèle que la méthode naïve d'interpolation linéaire. Jusqu'à présent, ces observations sont valables pour l'étude de fonctions totalement inconnues mais le champ magnétique est une grandeur physique spécifique qui suit certaines propriétés, que nous avons expliquées dans la Section 3.1.1. Sous l'hypothèse que la densité de courant est négligeable, le potentiel du champ magnétique a un rotationnel égal à $\mathbf{0}_3$. Cela permet de relier les trois différentes dimensions du champ entre elles, car elles dérivent toutes d'un unique potentiel scalaire φ ([SKW+15]). Le processus gaussien est appliqué sur ce potentiel scalaire : il est nécessaire de calculer une fonction seulement au lieu de 3.

$$\nabla \times \mathbf{B} = 0 \Rightarrow \mathbf{B} = -\nabla\varphi. \tag{3.9}$$

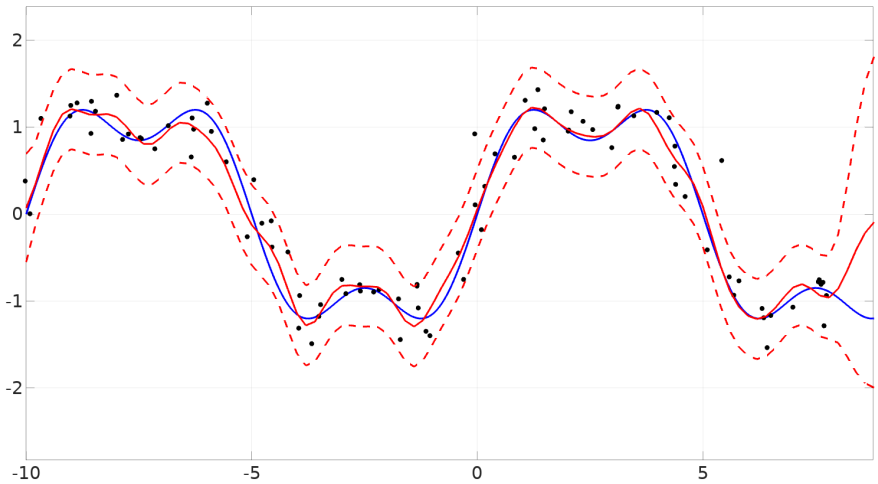


FIGURE 3.1: Processus gaussien à une dimension. La courbe bleue indique la fonction réelle et la rouge le processus gaussien obtenu à partir des mesures bruitées (points noirs). Les deux courbes en pointillés représentent l'intervalle de confiance à 2σ.

La mesure du champ magnétique $\hat{\mathbf{m}}(\mathbf{x})$ au point $\mathbf{x}$ suit donc la loi

$$\hat{\mathbf{m}}(\mathbf{x}) = -\nabla\varphi(\mathbf{x}) + \varepsilon. \tag{3.10}$$

Contrairement au GP classique, la moyenne du champ n'est pas nulle mais nous ignorons son orientation. Il est possible, en gonflant la variance, de permettre au modèle de converger vers cette moyenne. Comme l'opérateur gradient est linéaire, si $\nabla\phi$ suit un processus Gaussien, alors ϕ aussi. La variance choisie doit permettre au modèle de calculer la véritable moyenne du champ dans la zone étudiée (proche du champ magnétique terrestre), selon la bonne orientation. Pour vérifier cette condition, nous avons donc une variance composée de deux parties : une partie quadratique (intégration de la partie constante du champ magnétique), κ_{lin}, et une partie variable (exponentielle quadratique), κ_{SE}. Si on dérive selon chaque composante la partie quadratique de la variance de φ, on obtient une constante, $\sigma_{lin}^2 I_3$, qui correspond à la variance du champ magnétique selon chacune de ses trois composantes (c'est pourquoi son indice s'appelle *lin*). La partie exponentielle quadratique est utilisée car on ne connaît pas les variations du champ et ce modèle est couramment employé parce qu'il permet souvent de bien approximer des fonctions inconnues.

$$\varphi(\mathbf{x}_i) \sim GP(0, \kappa_{lin}(\mathbf{x}_i, \mathbf{x}_j) + \kappa_{SE}(\mathbf{x}_i, \mathbf{x}_j)) \tag{3.11}$$

$$\kappa_{lin}(\mathbf{x}_i, \mathbf{x}_j) = \sigma_{lin}^2 \mathbf{x}_i^\top \mathbf{x}_j \tag{3.12}$$

Une fois que le processus gaussien a été estimé, il suffit de calculer son gradient afin d'obtenir les trois composantes du champ magnétique. Grâce à la prise en compte des dépendances physiques entre les 3 composantes, la qualité de l'estimation obtenue est meilleure que lorsqu'on estime chaque composante du champ de manière indépendante [SKW+15].

En appliquant ce modèle théorique à des mesures réelles, il est possible d'obtenir des cartes d'interpolation complètes d'une zone, avec les incertitudes associées. Nous allons étudier dans la prochaine partie l'implémentation de ce modèle pour de grandes cartes, en divisant l'espace en plusieurs sous-cartes avec chacune un modèle indépendant.

3.1.4 Création de carte magnétique

3.1.4.1 Le champ modélisé par un processus Gaussien à rang réduit

Pour un volume important de données d'entrée, le calcul par processus gaussiens devient rapidement impossible pour la puissance de calcul d'un ordinateur car sa complexité est en $O(n^3)$ avec n le nombre de données (pour calculer l'espérance et la variance, il est nécessaire d'inverser une matrice de taille n).

Pour réduire la complexité du calcul, plusieurs modifications des formules classiques de l'approximation par processus gaussien ont été proposées. Une d'entre elles, plutôt commune, consiste à stocker les valeurs en certains points uniquement et en interpolant à partir de ces points pour obtenir des prédictions. Dans [SKW+15], l'approximation à rang réduit est utilisée [SS20]. La matrice de covariance du processus gaussien peut être approximée de la manière suivante :

$$\kappa(\mathbf{x}_i, \mathbf{x}_k) \approx \sum_{j=1}^{m} S(\lambda_j)\phi_j(\mathbf{x}_i)\phi_j(\mathbf{x}_k) \tag{3.13}$$

avec $S(\cdot)$ la densité spectrale de la fonction $\kappa(\mathbf{x}_i, \mathbf{x}_k)$. $\phi_j(\mathbf{x}_i)$ est la valeur au point $\mathbf{x}_i$ de la j-ième fonction propre négative de Laplace et λ_j^2 la valeur propre associée. Le problème peut être résolu en un temps $O(nm^2)$, avec m fixé.

$$\begin{cases} -\nabla^2\phi_j(\mathbf{x}) = \lambda_j^2\phi_j(\mathbf{x}), & \mathbf{x} \in \Omega \\ \phi_j(\mathbf{x}) = 0, & \mathbf{x} \in \partial\Omega \end{cases} \tag{3.14}$$

avec j l'indice correspondant à la fonction propre étudiée.

Une condition supplémentaire est ajoutée pour pouvoir résoudre le système de façon explicite : les conditions aux limites de Dirichlet qui imposent la nullité aux bords. La nullité aux bords est un artifice de calcul qui permet de résoudre facilement le système d'équations et n'a aucun fondement physique.

Pour une résolution explicite, nous nous plaçons dans le cas où le domaine Ω est un parallélépipède rectangle, de côté $2 \times L_q$ selon chaque dimension q,

$$\lambda_j^2 = \sum_{q=1}^{d} \left(\frac{\pi n_{j,q}}{2L_q}\right)^2 \tag{3.15}$$

avec $n_{j,d}$ l'entier associé à la fréquence correspondante.

L'intérêt de l'approximation à rang réduit est de diminuer significativement la complexité des calculs par rapport à un processus Gaussien classique, dans le cas où un grand nombre de données n servent à établir le modèle : la complexité en $O(nm^2)$ au lieu de $O(n^3)$. L'écriture à rang réduit permet utiliser une base de fonctions pré-calculables $(\phi_j)_{j=1:m}$. À l'aide des valeurs magnétiques obtenues sur le terrain, on trouve le coefficient associé à chaque fonction.

Les équations se résolvent de façon explicite : il s'agit de sinus à différentes fréquences.

$$\phi_j(\mathbf{x}) = \prod_{q=1}^{3} \frac{1}{\sqrt{L_q}} \sin\left(\frac{\pi n_{j,q}(\mathbf{x}_q + L_q)}{2L_q}\right) \tag{3.16}$$

La Figure 3.2 représente les premières fonctions obtenues dans un espace en 2 dimensions. Plus on décide d'utiliser de nombreuses fonctions propres pour représenter l'espace, plus les fréquences des sinus sont grandes et leur combinaison permet de restituer des phénomènes de plus en plus localisés.

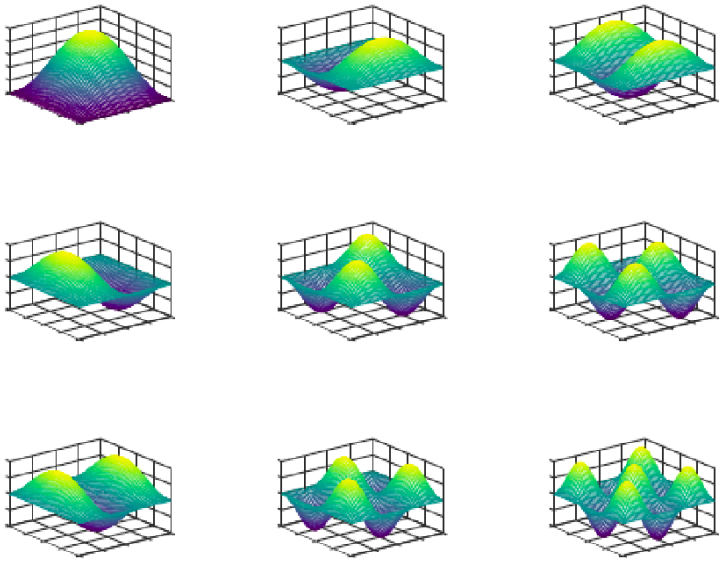


FIGURE 3.2: Premières fonctions constituant la base de fonctions permettant de modéliser le champ magnétique pour un carré (2 dimensions).

L'espace est subdivisé en carrés ou bien en hexagones (en 2D, [RFA+13]), en cubes ou en cylindres à base hexagonale (en 3D, [SKW+15], [KS18]). Cela permet de limiter le nombre de fonctions propres nécessaires pour modéliser l'espace étudié et donc de limiter la complexité des calculs.

Soit, Φ la matrice contenant les valeurs de toutes les fonctions ϕ pour tous les points de données. À chaque point $\mathbf{x}_i$ correspond une ligne de la matrice Φ composée des éléments suivants : $\Phi_i = (\mathbf{x}^\top, \phi_1(\mathbf{x}_i), ..., \phi_m(\mathbf{x}_i))$. Le vecteur $vec(Y) = (\mathbf{y}_1^\top, ..., \mathbf{y}_n^\top)^\top$ contient toutes les mesures associées.

Λ est la matrice des valeurs de la densité de fonction $S_{SE}(\cdot)$:

$$\Lambda = (\sigma_{lin}^2, \sigma_{lin}^2, \sigma_{lin}^2, S_{SE}(\lambda_1), ..., S_{SE}(\lambda_m)) \tag{3.17}$$

avec, pour une valeur scalaire quelconque ω,

$$S_{SE}(\omega) = \sigma_{SE}^2 (2\pi l_{SE}^2)^{\frac{3}{2}} \exp\left(\frac{-\omega^2 l_{SE}^2}{2}\right). \tag{3.18}$$

Le champ $\mathbf{B}$ qui est mesuré correspond au gradient de ϕ. Comme la covariance est une fonction bi-linéaire, on peut obtenir la variance de $\mathbf{B}$ à partir de celle de ϕ : elle est égale à $\nabla\Phi\Lambda\nabla\Phi^\top$. La matrice $\nabla\Phi$, de taille $3n \times (3+m)$, est une concaténation horizontale des différentes dérivées des fonctions ϕ évaluées aux points $\mathbf{x}_i$:

$$\nabla\Phi_i = (I_3, \nabla\phi_1(\mathbf{x}_i), ..., \nabla\phi_m(\mathbf{x}_i)). \tag{3.19}$$

On définit ensuite Φ_*, de façon semblable à Φ mais avec les coordonnées des points pour lesquels on souhaite connaître l'estimation du champ.

On peut écrire sous forme matricielle les calculs du modèle pour tous les points $\mathbf{x}_*$, approximés à partir des données apprises [SKW+15] :

$$\begin{cases} \mathbb{E}[B(\mathbf{x}_*)] \approx \nabla\Phi_*([\nabla\Phi]^\top\nabla\Phi + \sigma_m^2\Lambda^{-1})^{-1}[\nabla\Phi]^\top vec(Y) \\ \mathbb{V}[B(\mathbf{x}_*)] \approx \sigma_m^2\nabla\Phi_*([\nabla\Phi]^\top\nabla\Phi + \sigma_m^2\Lambda^{-1})^{-1}[\nabla\Phi_*]^\top \end{cases} \tag{3.20}$$

avec *vec* l'opérateur qui permet de vectoriser une matrice.

Nous pouvons réécrire ces équations comme :

$$\begin{cases} \mathbb{E}[B(\mathbf{x}^*)] \approx \nabla\Phi_*\boldsymbol{\mu}_l \\ \mathbb{V}[B(\mathbf{x}^*)] \approx \nabla\Phi_*\Sigma_l\nabla\Phi_*^\top \end{cases} \tag{3.21}$$

$$\text{avec } \boldsymbol{\mu}_l = ([\nabla\Phi]^\top\nabla\Phi + \sigma_m^2\Lambda^{-1})^{-1}[\nabla\Phi]^\top \mathrm{vec}(\mathbf{m}),$$
$$\Sigma_l = \sigma_m^2([\nabla\Phi]^\top\nabla\Phi + \sigma_m^2\Lambda^{-1})^{-1}.$$

$\boldsymbol{\mu}_l$ et Σ_l peuvent être précalculés une fois que les mesures sont collectées. On peut créer une carte magnétique listant les cubes explorés et associant à chacun d'entre eux les paramètres $\boldsymbol{\mu}_l$ et Σ_l. En utilisant les valeurs sotckées, on peut prédire rapidement le champ en tout point de la carte.

3.1.4.2 Calcul des hyperparamètres du modèle

Les équations ci-dessus sont utilisables à condition de connaître a priori 4 hyperparamètres, qui peuvent être obtenus en minimisant une fonction de probabilité. Ces 4 hyperparamètres sont l'écart-type du bruit sur les mesures en sortie σ_m, et les paramètres de la partie variable de la covariance, σ_{lin}, qui correspond à l'impact sur le processus gaussien de la partie constante du signal (égale au champ magnétique terrestre en théorie), σ_{SE} et l_{SE} (qui sont nécessaires pour connaître sa densité de probabilité). σ_m correspond en théorie au bruit du capteur, σ_{noise}, mais l'incertitude sur les positions impose d'augmenter la valeur de σ_m par rapport à σ_{noise} : les effets du choix de σ_m seront détaillés dans la Section 3.2. Ces hyperparamètres peuvent être les mêmes pour tous les cubes constituant la carte.

Les paramètres sont le résultat de l'optimisation d'une fonction de vraisemblance $\mathscr{L}(\theta)$, avec $\theta = (\sigma_{lin}, \sigma_{SE}, l_{SE})$ [SKW+15]. Cette fonction se calcule à partir des couples de positions et champs magnétiques mesurés :

$$\mathscr{L}(\theta) = \frac{1}{2}\log|K_\theta + \sigma_m^2 I_{3n}| + \frac{1}{2}vec(Y)^\top\log(K_\theta + \sigma_m^2 I_{3n})^{-1}vec(Y) + \frac{3n}{2}\log(2\pi) \tag{3.22}$$

avec $K_\theta = \nabla\Phi_\theta\Lambda_\theta\nabla\Phi_\theta^\top$.

Il est intéressant d'obtenir les hyperparamètres les plus génériques possibles, qui soient valables dans un grand nombre d'environnements différents. Pour les estimer efficacement, il faut scanner plusieurs cubes de $5m \times 5m \times 2m$ en collectant le plus de données possibles, en choisissant des cubes ayant des profils magnétiques relativement exhaustifs dans le bâtiment (couloirs, salles, escaliers, hall). Les valeurs magnétiques et les positions vérité-terrain dans ces cubes seront utilisées pour optimiser la fonction de vraisemblance, $\mathscr{L}_\theta$.

Les paramètres qui donnaient les meilleurs résultats en moyenne pour nos expériences sont proches de ceux déterminés dans [KS18], à part σ_m que nous avons modifié en fonction des cas d'usage (quand la localisation est plus précise, σ_m peut être réduit, comme expliqué dans la Section 3.2 qui suit). Les valeurs respectives sont données par $\sigma_{lin} = 25\ \mu$T, $\sigma_{SE} = 15\ \mu$T.m^{-1},

$l_{SE} = 1.3$ m et $\sigma_m^2 = 2\ \mu$T, sauf si une autre valeur est indiquée (souvent, c'est σ_m que nous modifions : les autres valeurs n'ont pas une influence aussi sensible sur le modèle et sont bien choisies pour nos cas d'usage). Il est à noter que nous avons estimé ces hyperparamètres sur nos propres acquisitions qui diffèrent de celles utilisées dans [KS18]. Le fait de trouver des résultats similaires demontre que ces valeurs sont assez génériques.

3.1.4.3 Pavage de l'espace

Si nous modélisons tout l'espace que nous souhaitons cartographier à l'aide d'un unique cube, le nombre de valeurs propres nécessaires est très élevé et même la formulation par processus Gaussien à rang réduit devient très coûteuse en termes de calculs (pour un m très élevé). Nous décidons donc de subdiviser l'espace en plusieurs zones de tailles limitées. Chaque zone est étudiée de façon indépendante, en n'utilisant que les informations des mesures qui lui appartiennent et sans prendre en compte les autres. Cependant, les conditions aux limites de Dirichlet créent des incohérences : comme les fonctions propres sont toutes égales à 0 aux frontières du cube, le modèle de ϕ converge vers 0 et le champ magnétique vers une constante. Ceci est d'autant plus problématique que la covariance est faible au voisinage des bords, à cause des conditions aux limites choisies. Pour supprimer cet effet, qui est non négligeable et non souhaité, il est nécessaire d'appliquer le modèle de carte magnétique sur un cube plus grand. La zone entre le cube plus grand et le cube initial est utilisée comme une zone tampon : les valeurs dans cette zone sont considérées comme non valides. Mais les données renvoyées par la modélisation avec le grand cube permettent d'obtenir une bonne modélisation, sans effets de bord, sur toute la surface couverte par le cube dans lequel on souhaite connaître le profil du champ magnétique. Avec les valeurs mentionnées précédemment et pour 8 fonctions propres par axe, le cube de modèle élargi adéquat a une taille de $(L_1 + (2\sqrt{2}+1)l_{SE}) \times (L_2 + (2\sqrt{2}+1)l_{SE}) \times (L_3 + (2\sqrt{2}+1)l_{SE})$ mais sa valeur peut varier en fonction de la taille du cube de base.

Un autre type de problème est aussi présent. Il est dû au découpage discret de l'espace : des problèmes de cohérence au niveau des bords des cubes peuvent exister. Pour un même emplacement, deux modèles différents avec des résultats distincts sont obtenus. Pour atténuer ces écarts, il est possible d'agrandir les cubes initiaux pour la collecte de données. Ceux-ci vont se superposer et partager la même information à leurs frontières : ces nouveaux cubes sont appelés les cubes de données (Figure 3.3). En pratique, un cube de données a une taille égale à $(L_1 + l_{SE}) \times (L_2 + l_{SE}) \times (L_3 + l_{SE})$ dans nos expériences mais sa valeur peut être différente en fonction du contexte, comme pour le cube de modèle élargi : cela dépend du compromis établi pour des transitions lisses entre les pararallélépipèdes et un temps de calcul plus long à cause d'un plus grand nombre de données à traiter pour chaque cube. Le cube de données doit être plus petit que le cube de modèle élargi, une nouvelle fois pour éviter les problèmes de conditions aux limites.

En pratique, le modèle obtenu estime plutôt bien le champ magnétique mais lorsque le système est loin des données d'appentissage, la qualité de l'apprentissage est détériorée et la variance associée aux prédictions est élevée. La Figure 3.4 montre l'évolution de l'enveloppe d'incertitude dans ce cas-là, avec une courbe de prédiction qui s'éloigne de plus en plus : cela se traduit par un gonflement progressif de l'intervalle de confiance. Au niveau du bord, la variance semble diminuer légèrement, en raison des effets des conditions aux limites de Dirichlet : à 1 mètre se trouve la limite du cube agrandi, le modèle considère que toutes les valeurs valent 0 ce qui correspond pour le modèle à l'équivalent de points d'apprentissage, alors qu'il s'agit d'un procédé artificiel. Cela montre l'importance de recourir à un modèle de cube plus grand que le cube étudié.

Nous pouvons utiliser directement les Equations 3.21 pour chaque cube indépendamment, afin

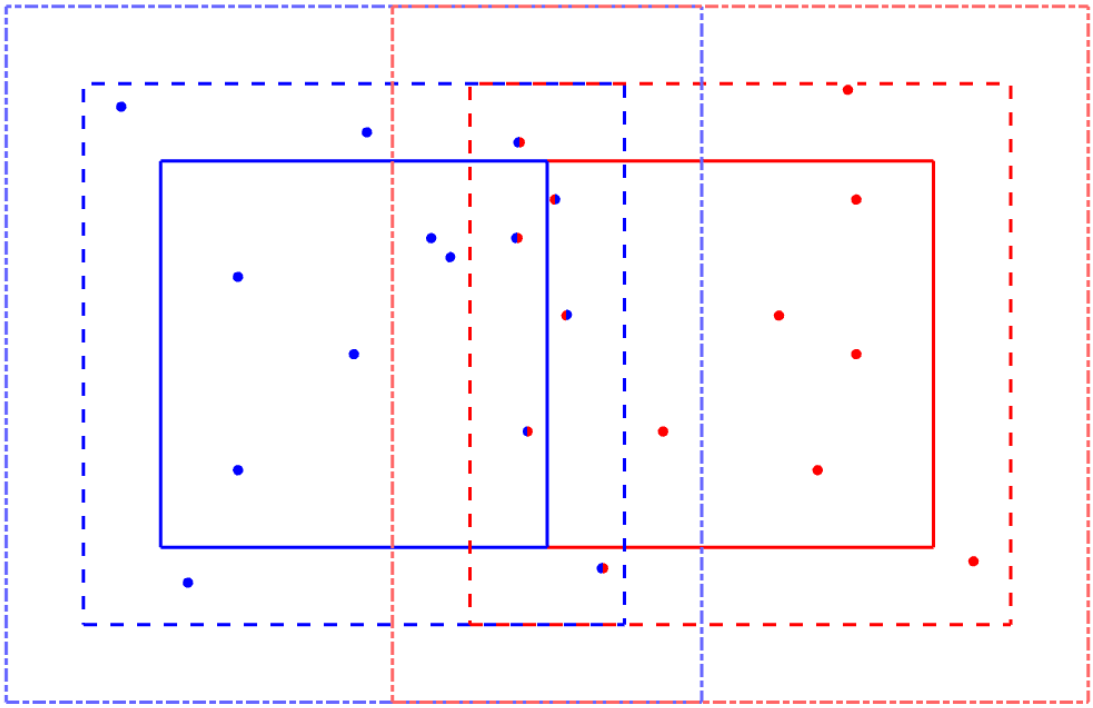


FIGURE 3.3: Exemple de cube de données et de cube de modèle élargi obtenus à partir du cube étudié. En points bleus figurent toutes les données acquises pour le cube de données, représenté en pointillés réguliers bleus. Ce modèle permet d'obtenir la carte sur l'aire du carré bleu (ligne continue). En dehors des limites en bleu (continu), les modèles d'autres cubes sont utilisés. De même pour les cubes rouges. Les cubes de modèle élargi sont représentés en couleur plus claire en pointillés discontinus et correspondent aux cubes les plus grands.

d'obtenir un modèle de carte complet, capable de prédire le champ magnétique en tout point.

Cette méthode est celle qui est utilisée comme référence pour la cartographie dans cette thèse car elle est précise tout en ne nécessitant que peu de temps de calcul. Elle sert aussi de base pour les améliorations du modèle présentées dans la suite de cette partie. Elle s'appuie sur les propriétés du champ magnétique et elle permet d'extrapoler les valeurs pour des zones non explorées mais proches des trajectoires d'apprentissage (approximativement l_{SE}), de manière relativement précise. Elle est aussi facilement applicable en trois dimensions.

3.1.5 Adaptation à la création de carte en ligne

Dans la partie précédente, le modèle est obtenu à partir de toutes les mesures en une seule étape. Cependant, si la carte magnétique est créée en ligne, de nouvelles mesures permettant d'améliorer la carte arrivent au fur et à mesure. Ici, une autre version de l'algorithme, cette fois-ci permettant une estimation itérative, est présentée. Elle est basée sur le formalisme de [S. 13] et correspond à un filtre de Kalman linéaire, avec $\boldsymbol{\mu}_l$ le vecteur d'état, Σ_l sa covariance et $\nabla\Phi$ joue le rôle de H. Elle donne les mêmes résultats que la méthode précédente mais est capable de traiter les données une par une.

On appelle $\boldsymbol{\mu}_l^i$ et Σ_l^i les valeurs stockées par la carte à l'étape i, dans le cube où se trouve la

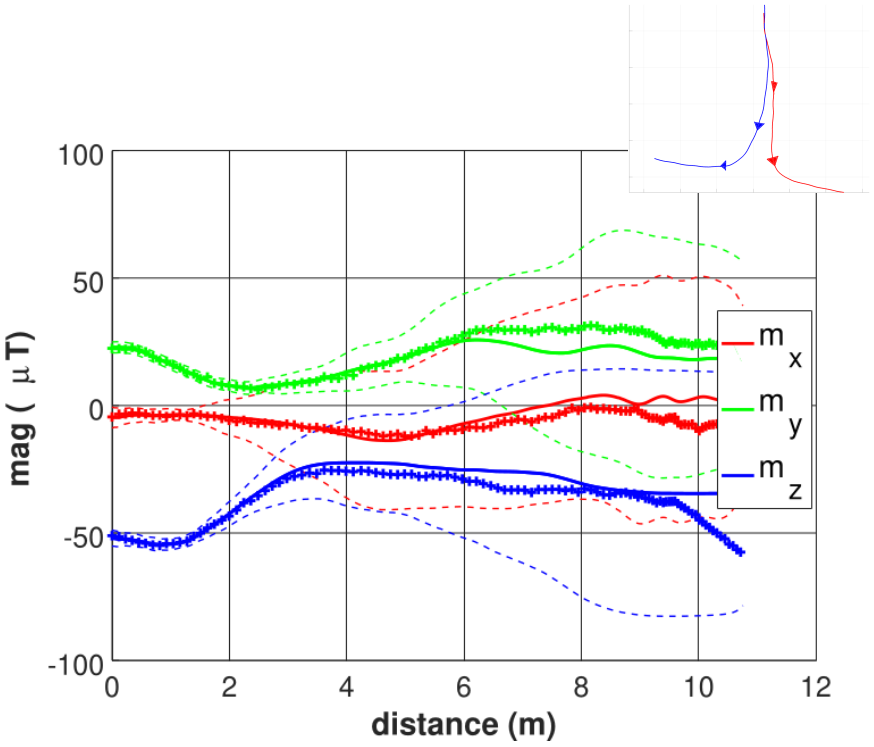


FIGURE 3.4: Valeurs magnétiques acquises par le magnétomètre (+) comparées aux valeurs prédites par le modèle (ligne continue, les pointillés correspondent à l'intervalle d'incertitude 3σ). Le carré en haut à droite représente le chemin parcouru, auquel correspondent ces prédictions. La trajectoire rouge représente la trajectoire des données acquises pour cette comparaison. Les prédictions du modèle sont obtenues à partir des données acquises par la trajectoire bleue du cube tracé en haut à droite (données d'apprentissage).

donnée $\mathbf{x}_i$. Pour chaque nouvelle donnée $(\mathbf{x}_i, \mathbf{y}_i)$:

$$\begin{aligned}
S_i &= \nabla\Phi_i \Sigma_{i-1} [\nabla\Phi_i]^\top + \sigma_m^2 I_3 \\
K_i &= \Sigma_l^{i-1} [\nabla\Phi_i]^\top S_i^{-1} \\
\boldsymbol{\mu}_l^i &= \boldsymbol{\mu}_l^{i-1} + K_i(\mathbf{y}_i - \nabla\Phi_i \boldsymbol{\mu}_l^{i-1}) \\
\Sigma_l^i &= \Sigma_l^{i-1} - K_i S_i K_i^\top .
\end{aligned} \tag{3.23}$$

Cette version itérative permet de travailler avec des matrices de petite taille ($3 \times m$ et $m \times m$).

3.2 Impact des erreurs de localisation dans l'estimation du champ

Nous avons vu dans la section précédente le modèle de l'état de l'art le plus précis et le plus utilisé jusqu'à maintenant pour la modélisation du champ magnétique. Cependant, ce modèle reste incomplet puisqu'il suppose que les positions $\mathbf{x}_i$ soient parfaitement connues et ne prend en compte que le bruit du capteur. Or la position du système est estimée par un algorithme de localisation, qui retourne aussi la variance associée à son estimation de la pose. Comme les positions associées aux données magnétiques ne peuvent être parfaites en pratique, leur incertitude doit être prise en compte dans le modèle.

Dans l'article utilisant la modélisation sans bruit sur les positions [KS18], le bruit du capteur dans le modèle, σ_m, est artificiellement gonflé par rapport au bruit du capteur réel, pour prendre en compte les erreurs de localisation. Au cours de nos expériences, nous avons aussi pu constater que l'utilisation du bruit réel du capteur engendrait des cartes magnétiques peu réalistes. En effet, lorsque deux mesures magnétiques sont acquises proches l'une de l'autre, un positionnement relatif peu précis engendre une surestimation du gradient de la carte à cet endroit comme l'illustre la Figure 3.5. L'expérience réalisée consiste à parcourir plusieurs fois le même endroit dans des directions différentes. Pour $\sigma_m = 0.28\mu T$, des valeurs de plus de $100\mu T$, très localisées, sont prédites par le modèle : de telles amplitudes sont aberrantes, alors qu'avec $\sigma_m = \sqrt{2}\mu T$, les variations du champ sont plus faibles et restent dans une plage de valeurs plus en accord avec la réalité (partie droite de la Figure 3.5).

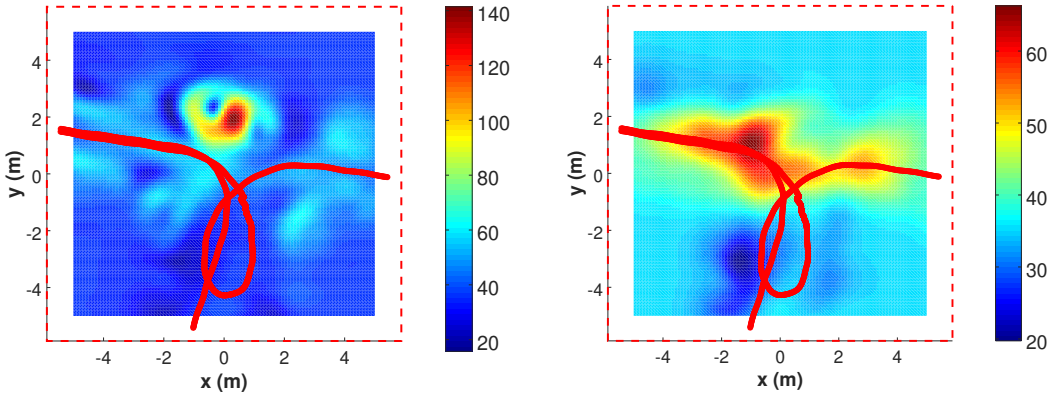


FIGURE 3.5: Comparaison de deux profils magnétiques à z = 0.3 m, pour deux valeurs différentes de σ_m. À gauche, $\sigma_m = 0.28\mu T$. À droite, $\sigma_m = \sqrt{2}\mu T$. Les trajectoires rouges correspondent aux séquences d'acquisition des mesures pour créer la carte.

Pour le modèle de mesure du magnétomètre, $\sigma_{noise} \neq \sigma_m$, nous conservons toujours $\sigma_{noise} = 0.28\mu T$ comme la vraie valeur associée au bruit blanc gaussien du capteur, égale pour les trois différentes dimensions du capteur : la valeur réelle magnétique se trouve, avec $\mathbf{m}$ la valeur réelle, entre $\mathbf{m} - 3\sigma_{noise}\mathbf{1}_3$ et $\mathbf{m} + 3\sigma_{noise}\mathbf{1}_3$ à 99.7 %, pour chaque dimension. En diminuant la confiance du modèle dans le capteur, c'est à dire en augmentant artificiellement σ_m à $\sqrt{2}\mu T$, non seulement l'intervalle de confiance s'élargit, mais le modèle lui-même est modifié : les oscillations locales sont presque effacées, la courbe du modèle est plus proche de celle des valeurs réelles.

Cependant, même si l'augmentation du bruit du capteur permet de limiter les erreurs de modélisation, l'estimation de σ_m s'obtient uniquement de manière empirique et suppose que l'erreur de localisation est semblable sur toute la trajectoire. De plus, la valeur de l'incertitude de la position varie au cours du temps, elle n'est pas fixe : certaines valeurs devraient avoir plus d'importance dans le modèle car elles ont une incertitude associée moins grande. C'est pourquoi le travail réalisé dans les prochaines sections a pour objectif de faire apparaître l'erreur de localisation dans les équations. Cela permet de réaliser une cartographie avec des valeurs de variance et des prédictions plus correctes d'un point de vue mathématique mais aussi expérimental.

3.3 Processus Gaussiens avec entrées et sorties simultanément bruitées

Nous avons vu dans la section précédente que la modélisation par processus gaussiens classiques présente des défauts : l'augmentation artificielle du bruit capteur permet de lisser les prédictions et d'éviter l'apparition de gradients magnétiques aberrants, mais ne permet pas de prendre plus ou moins en compte une donnée magnétique en fonction de la précision de la position qui lui est associée. Dans cette section, nous nous intéressons donc à un nouveau type de processus gaussien, prenant en compte les bruits sur les sorties mais aussi les entrées. Dans la Section 3.3.1, les solutions déjà existantes sont présentées. Les nouvelles variables et notations sont introduites en Section 3.3.2. Puis notre raisonnement et les calculs de notre nouvelle approche sont développés en Section 3.3.3.

3.3.1 Méthodes de l'état de l'art

Plusieurs appoches ont déjà été développées pour prendre en compte l'incertitude des données d'entrée pour un processus Gaussien (GP). Certaines sont basées sur une hypothèse de non-homogénéité du bruit sur les sorties : elles estiment le bruit du capteur en fonction des données collectées. Celui-ci est supposé non-constant et est considéré comme une variable supplémentaire à optimiser par le modèle [GWB11, KPPB07]. Ces méthodes hétéroscédastiques peuvent être utilisées pour décrire des séquences de données sujettes à un bruit sur les données d'entrée. Toutefois des effets de sur-apprentissage sont difficiles à éviter et il n'est pas possible d'introduire directement la vraie valeur de la covariance sur les données d'entrée.

Plusieurs méthodes ont été proposées pour modifier le modèle GP dans l'objectif d'inclure le bruit sur les données d'entrée. Elles ont démontré expérimentalement que les estimations gagnent en précision quand ce bruit est pris en compte, en plus du bruit sur les sorties [CC03, MR11]. Dans [CC03, CP15, DBCd09], un bruit blanc Gaussien est ajouté aux entrées et les covariances sont modifiées en respectant ces changements. L'espérance avec incertitude sur les entrées $Z_{pu}(\cdot)$ est modélisée comme l'intégrale de l'espérance du processus Gaussien sans incertitude $Z(\cdot)$ pour toutes les valeurs d'entrée possibles, pondérées par la densité de probabilité des données d'entrée. Le même raisonnement est appliqué au calcul des covariances. L'inconvénient principal de cette méthode est que la plupart du temps, ces intégrales ne peuvent être résolues analytiquement. Pour obtenir des valeurs numériques, une estimation par la méthode de Monte-Carlo est proposée. Cette méthode devient précise si un grand nombre de tirages de Monte Carlo sont exécutés, ce qui demande des ressources de calcul importantes. D'autres approches utilisent la décomposition de Taylor [MR11, BSvV16] : les estimations sont bien meilleures qu'avec un GP classique pour un temps de calcul réduit.

Cependant, les matrices obtenues avec ces modèles ont des tailles de plus en plus grandes au fur et à mesure que le nombre de données croît. Les calculs numériques deviennent intractables : dans [SS20], une approximation d'un processus Gaussien avec un nombre fixe de fonctions est proposé. L'algorithme proposé dans [BSvV16], SONIG (Sparse Online Noisy Input GP), résout ce problème à l'aide d'un processus Gaussien discret : le principe est de calculer et de stocker la moyenne et la covariance au niveau de certains points, appelés nœuds. Ces nœuds sont généralement répartis de façon uniforme sur tout l'espace étudié. La fonction étudiée peut être prédite en tout point à partir des valeurs en ces nœuds ([BvSV15]). Il s'agit d'une approximation par rapport à un processus Gaussien classique : ses prédictions sont un peu moins précises mais le temps de calcul est largement diminué. Cependant, le ratio de l'erreur moyenne sur la covariance est sou-

vent trop élevé, comme les auteurs le montrent dans leurs expériences : ce ratio est bien meilleur qu'avec les autres méthodes, mais il est tout de même égal à 2.7 dans l'expérience présentée. Le modèle a un intervalle de confiance trop faible par rapport à la réalité. De plus, pour certains jeux de données, l'algorithme diverge et ses résultats deviennent inutilisables alors que les autres méthodes d'estimation parviennent à approximer la fonction, bien que souvent avec des erreurs plus importantes qu'en moyenne. D'autres types d'approximation existent et ont une complexité similaire, comme les processus Gaussiens à rang réduit ([SS20]), que nous utilisons déjà pour la création de la carte magnétique. Cependant, à notre connaissance, cette méthode n'a pour l'instant pas encore été adaptée pour inclure l'incertitude sur les données d'entrée. Nous avons donc décidé de développer les équations du processus Gaussien à rang réduit avec entrées bruitées et de comparer le nouvel algorithme avec les approches existantes. Comme mentionné dans la partie expérimentale, notre modèle permet d'obtenir de meilleures estimations que [BSvV16] dans certains cas d'usage, en particulier quand la covariance est élevée (voir la Section 3.4). De plus, notre méthode est beaucoup plus simple à implémenter que [BSvV16].

Ici, nous présentons le raisonnement permettant d'abord d'obtenir l'expression exacte des espérances et covariances approximées au second ordre. Puis une méthode de GP à rang réduit est détaillée, précise, robuste et utilisable quel que soit le nombre de données.

3.3.2 Introduction du problème

Jusqu'à la fin de ce chapitre, nous nous intéressons à un processus gaussien dont les données d'entrée et de sortie sont bruitées. L'objectif est d'obtenir un modèle plus juste mathématiquement, permettant d'éviter d'avoir à gonfler le bruit du capteur artificiellement. Les équations sont d'abord développées dans le cas général puis appliquées au cas particulier du champ magnétique à la toute fin de la Section 3.5. Dans un premier temps, les différentes notations sont introduites et le problème est expliqué en termes probabilistes.

On cherche à évaluer un champ $B(\mathbf{x})$ en tout point $\mathbf{x}$ de l'espace. Il est supposé ne pas varier en fonction du temps. En un point donné, le champ est décrit par un vecteur dans un repère donné.

Nous souhaitons modéliser l'aspect aléatoire de B afin de l'estimer le plus précisément possible au travers de mesures. On considère B comme un champ aléatoire (dépendant de $\mathbf{x} \in \mathbb{R}^d$) à valeurs vectorielles ($B(\mathbf{x}) \in \mathbb{R}^c$). Plus précisément, nous utilisons la théorie des processus gaussiens, et décrivons B comme un processus gaussien, ce qui signifie qu'il est entièrement défini par son espérance $\mathbb{E}[B(\mathbf{x})]$ et sa covariance :

$$\kappa(\mathbf{x},\mathbf{x}') = \mathrm{Cov}\left(B(\mathbf{x}), B(\mathbf{x}')\right) = \mathbb{E}\left[\left(B(\mathbf{x}) - \mathbb{E}[B(\mathbf{x})]\right)\left(B(\mathbf{x}') - \mathbb{E}\left[B(\mathbf{x}')\right]\right)\right]. \tag{3.24}$$

On choisit initialement $\mathbb{E}[B(\mathbf{x})] = 0$ puisque nous n'avons aucun a priori sur l'orientation du champ ; on verra plus loin comment choisir une bonne fonction de covariance.

La question devient : quelle information apportent les mesures ?

On modélise une mesure de la manière suivante. La mesure a lieu en un point de l'espace $\mathbf{x} \in \mathbb{R}^d$, pour lequel on lit une valeur $\mathbf{b} \in \mathbb{R}^c$. Par la suite nous ferons des séries de mesures, que nous indicerons par i, i étant le numéro de la mesure, et nous aurons donc une série de lieux de mesures $\mathbf{x}_i$ auxquels sont associées des valeurs $\mathbf{b}_i$ de champ mesuré selon les c directions de l'espace.

Idéalement, on voudrait écrire $\mathbf{b} = B(\mathbf{x})$, ce qui serait le cas d'un capteur parfait. Cependant, on constate que le capteur possède un bruit de mesure $\varepsilon \in \mathbb{R}^c$, qu'on modélise comme une gaussienne isotrope de variance σ_b^2, i.e. ε est de moyenne nulle et de matrice de covariance diagonale dont

tous les termes valent σ_b^2 ; ce qui est équivalent à dire que chacune des c composantes de ε suit la loi $\mathcal{N}(0,\sigma_b^2)$. Ainsi modélisée, la variable aléatoire associée à la mesure s'écrit :

$$\hat{b} = B(\mathbf{x}) + \varepsilon \tag{3.25}$$

On note $\mathbf{b}$ une réalisation de la variable aléatoire $\hat{b}$ (soit une mesure du champ B à une certaine position).

Ce cas de mesure correspond à une localisation parfaite. En réalité, il est nécessaire d'inclure une incertitude sur la position $\mathbf{x}$, qui fait aussi l'objet d'une mesure. Pour modéliser l'imperfection de cette mesure, on introduit l'erreur de localisation $\xi \in \mathbb{R}^d$, qui est un vecteur aléatoire de taille d. Contrairement à l'erreur sur le champ ε, l'erreur de localisation peut varier au cours du temps et le modèle de variation diffère en fonction de la méthode d'acquisition : on utilisera un vecteur gaussien, ξ, de moyenne nulle et de variance Σ, dépendant du temps. La véritable position $\hat{\mathbf{x}}$ de la mesure s'écrit :

$$\hat{\mathbf{x}} = \mathbf{x} + \xi \tag{3.26}$$

En tenant compte de cette erreur, la variable aléatoire associée à la mesure magnétique peut être modélisée ainsi :

$$B_{LE} = B(\hat{\mathbf{x}}) + \varepsilon = B(\mathbf{x} + \xi) + \varepsilon \tag{3.27}$$

En notant $B_{LE} = B(\mathbf{x} + \xi) + \varepsilon$ la variable aléatoire pré-mesure (LE = Localization Error), la question qui se pose désormais est : quelle information la mesure apporte-t-elle ? Ceci devient, en termes probabilistes, comment le champ B est-il modifié lorsqu'on le conditionne à la variable $B_{LE} = b$? Comment ceci se généralise-t-il à une série de mesures, donc un conditionnement à de multiples $B_{LE}(\alpha) = b_\alpha$ aux environs des points $\mathbf{x}^\alpha$?

Cette fois-ci, cette loi n'est pas un processus gaussien à cause du terme $B(\mathbf{x} + \xi)$. Cependant, pour pouvoir l'étudier, on décide de l'approximer par une loi gaussienne : cela nécessite de connaître la moyenne et la covariance de B_{LE}.

$B(\mathbf{x})$ est une variable aléatoire gaussienne : on note sa moyenne $\mathbb{E}[B(\mathbf{x})]$ et sa covariance $\kappa(\mathbf{x},\mathbf{x})$. Pour deux points distincts $\mathbf{x}$ et $\mathbf{x}'$,

$$\begin{bmatrix} B(\mathbf{x}) \\ B(\mathbf{x}') \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu_{B(\mathbf{x})} \\ \mu_{B(\mathbf{x}')} \end{bmatrix}, \begin{bmatrix} \kappa(\mathbf{x},\mathbf{x}) & \kappa(\mathbf{x},\mathbf{x}') \\ \kappa(\mathbf{x},\mathbf{x}')^\top & \kappa(\mathbf{x}',\mathbf{x}') \end{bmatrix} \right) \tag{3.28}$$

3.3.3 Étude de la loi de probabilité d'une fonction avec entrées et sorties simultanément bruitées

Cette Section a pour objectif d'étudier les équations du conditionnement de B à une réalisation de la variable aléatoire B_{LE}, en adoptant une approche similaire au conditionnement d'un processus gaussien classique. Nous pouvons ainsi voir les différences induites dans le modèle par l'apparition du bruit ξ sur les entrées.

Les équations de conditionnement d'un processus gaussien s'obtiennent à partir de la loi jointe de B et B_{LE}, avec $\mathbf{b}$ une réalisation de B_{LE} [RW06]. On souhaite prédire le champ magnétique en deux points quelconques, $\mathbf{x}$ et $\mathbf{x}'$. On choisit deux points distincts pour obtenir une expression générale de la covariance et pas uniquement de la variance en un point.

$$\begin{bmatrix} B(\mathbf{x}) \\ B(\mathbf{x}') \\ B_{LE}(\mathbf{x}^\alpha) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu_{B(\mathbf{x})} \\ \mu_{B(\mathbf{x}')} \\ \mu_{B_{LE}(\mathbf{x}^\alpha)} \end{bmatrix}, \begin{bmatrix} K_1 & K_2 \\ K_2^\top & K_3 \end{bmatrix} \right) \tag{3.29}$$

$\mu_{B(\mathbf{x})}$ est l'espérance du champ au point $\mathbf{x}$. K_1, K_2 et K_3 sont différentes matrices de covariance, dont l'expression explicite est développée dans l'encadré ci-dessous. K_1 est de taille $2c \times 2c$, K_2 de taille $2c \times c$ et K_3 de taille $c \times c$. À partir de la loi jointe, il est possible d'obtenir directement l'espérance et la covariance aux points étudiés :

$$\boxed{\mathbb{E}\left[\begin{bmatrix} B(\mathbf{x}) \\ B(\mathbf{x}') \end{bmatrix} |\mathbf{b}\right] = \begin{bmatrix} \mu_{B(\mathbf{x})} \\ \mu_{B(\mathbf{x}')} \end{bmatrix} + K_2 K_3^{-1}(\mathbf{b} - \mu_{B_{LE}(\mathbf{x}^\alpha)})} \tag{3.30}$$

$$\boxed{\mathrm{Cov}\left(\begin{bmatrix} B(\mathbf{x}) \\ B(\mathbf{x}') \end{bmatrix}, \begin{bmatrix} B(\mathbf{x}) \\ B(\mathbf{x}') \end{bmatrix} |\mathbf{b}\right) = \mathrm{Cov}\left(\begin{bmatrix} B(\mathbf{x}) \\ B(\mathbf{x}') \end{bmatrix}, \begin{bmatrix} B(\mathbf{x}) \\ B(\mathbf{x}') \end{bmatrix}\right) - K_2 K_3^{-1} K_2^\top} \tag{3.31}$$

Processus Gaussien avec incertitude sur les données d'entrée et de sortie

Processus Gaussien classique : covariances initiales et bruit sur les sorties

Notre méthode : ajout du bruit sur les entrées

Mêmes équations, mais des termes supplémentaires dans les matrices de covariance utilisées pour les calculs.

Variance du modèle :

$$K_1 = \begin{bmatrix} \kappa(\mathbf{x},\mathbf{x}) & \kappa(\mathbf{x},\mathbf{x}') \\ \kappa(\mathbf{x},\mathbf{x}')^\top & \kappa(\mathbf{x}',\mathbf{x}') \end{bmatrix}$$

Covariance modèle / mesures :

$$K_2 \simeq \begin{bmatrix} \kappa(\mathbf{x},\mathbf{x}^\alpha) + \frac{1}{2}\left\langle \Sigma^{t^\alpha}, H_2(\kappa)(\mathbf{x},\mathbf{x}^\alpha) \right\rangle \\ \kappa(\mathbf{x}',\mathbf{x}^\alpha) + \frac{1}{2}\left\langle \Sigma^{t^\alpha}, H_2(\kappa)(\mathbf{x}',\mathbf{x}^\alpha) \right\rangle \end{bmatrix}$$

Variance des mesures :

$$K_3 \simeq \left[\kappa(\mathbf{x}^\alpha,\mathbf{x}^\alpha) + \Sigma_b + \left\langle \Sigma^{t^\alpha}, H_1(\kappa)(\mathbf{x}^\alpha,\mathbf{x}^\alpha) \right\rangle + \mathrm{Var}\left(\nabla B(\mathbf{x}^\alpha)\xi^\alpha\right)\right]$$

Σ_b est la matrice de covariance du bruit du capteur et Σ^{t^α} la matrice de covariance sur la position au temps t_α.

$$\langle P, Q \rangle = \sum_{j=1}^{d} \sum_{k=1}^{d} P_{j,k} Q_{1:d,j,k}$$

$H_1(\kappa)(\mathbf{x},\mathbf{x}')$ correspond à la dérivée seconde de κ selon la première composante de $\mathbf{x}$ uniquement, et $H_2(\kappa)(\mathbf{x},\mathbf{x}')$ selon la deuxième composante :

$$\begin{cases} H_1(\kappa)(\mathbf{x},\mathbf{x}') = \dfrac{\partial^2 \kappa(\mathbf{x},\mathbf{x}')}{\partial \mathbf{x} \partial \mathbf{x}} \\ H_2(\kappa)(\mathbf{x},\mathbf{x}') = \dfrac{\partial^2 \kappa(\mathbf{x},\mathbf{x}')}{\partial \mathbf{x}' \partial \mathbf{x}'} \end{cases}.$$

Dans les pages suivantes, on démontre le résultat présent dans l'encadré.

3.3.3.1 Développement de la formule de Taylor

On commence d'abord par étudier $B(\mathbf{x}+\xi)$, à $\mathbf{x}$ fixé. On étudie les différences par rapport à un processus gaussien classique, où seules les sorties sont bruitées.

Il est possible soit de l'étudier directement en calculant sa densité de probabilité soit en l'approximant. La densité de probabilité dépend directement du type de modèle de covariance choisi et elle est souvent difficile à résoudre analytiquement. Or on souhaite se placer dans le cas le plus général possible. Les méthodes actuelles de calcul direct, quand une solution analytique n'existe pas, sont basées sur des méthodes de Monte-Carlo [CP15, CC03] mais nécessitent un temps d'exécution long. Pour simplifier l'expression de $B(\mathbf{x}+\xi)$ et pouvoir la calculer approximativement, on décide d'approximer $B(\mathbf{x}+\xi)$ par sa formule de Taylor.

Pour obtenir l'expression explicite de la moyenne et de la covariance de B, on commence d'abord par développer son expression au voisinage du point $\mathbf{x}$ à l'aide de la formule de Taylor, mentionnée précédemment :

$$B(\mathbf{x}+\xi) = B(\mathbf{x}) + \nabla(B(\mathbf{x}))\xi + \frac{1}{2}h(B(\mathbf{x}),\xi) + O(\| \xi \|^3) \tag{3.32}$$

avec (sachant que $\mathbf{x}$ est un vecteur de taille d, les indices i varient de 1 à d pour chaque composante du vecteur qu'ils représentent) :

$$\Big(h(B(\mathbf{x}),\xi)\Big)_i = \sum_{j=1}^{d}\sum_{k=1}^{d} \xi_j \xi_k \Big(\partial_j \partial_k B_i(\mathbf{x})\Big) \tag{3.33}$$

L'expression vectorielle complète de $h(B(\mathbf{x}),\xi)$ s'écrit :

$$h(B(\mathbf{x}),\xi) = \sum_{j=1}^{d}\sum_{k=1}^{d} \xi_j \xi_k \frac{\partial^2 B(\mathbf{x})}{\partial \mathbf{x_j} \partial \mathbf{x_k}} \tag{3.34}$$

$h(B(\mathbf{x}),\xi)$ est un vecteur de taille c. Pour simplifier l'écriture, on note $B(\mathbf{x}) = B$. Ici et dans la suite du calcul, $\mathbf{x}$ est une valeur fixée ; il n'est donc pas nécessaire d'indiquer la dépendance en $\mathbf{x}$.

3.3.3.2 Espérance de $B(\mathbf{x}+\xi)$

B et ξ sont deux variables indépendantes donc l'espérance de leur produit est le produit de leurs espérances, et de même pour la dérivée et la hessienne de B, avec $\mathbf{1}_c$ un vecteur composé de

1 et de taille c :

$$\begin{aligned}\mathbb{E}\left[B(\mathbf{x}+\xi)\right] &= \mathbb{E}\left[B+\nabla B\xi+\frac{1}{2}h(B(\mathbf{x}),\xi)+O(\parallel \xi \parallel^3 \mathbf{1}_c^\top)\right] \\ &\simeq \mathbb{E}\left[B\right]+\mathbb{E}\left[\nabla B\right]\mathbb{E}\left[\xi\right]+\frac{1}{2}\mathbb{E}\left[h(B(\mathbf{x}),\xi)\right] \\ &\simeq \mathbb{E}\left[B\right]+\frac{1}{2}\mathbb{E}\left[h(B(\mathbf{x}),\xi)\right]\end{aligned} \tag{3.35}$$

car ξ est un vecteur aléatoire (de taille d) associé à un bruit blanc. Il est donc de moyenne nulle.

Maintenant, en calculant l'espérance et en utilisant son expression développée donnée en 3.34 :

$$\mathbb{E}\left[h(B(\mathbf{x}),\xi)\right] = \sum_{j=1}^{d}\sum_{k=1}^{d}\mathbb{E}\left[\xi_j\xi_k\frac{\partial^2 B}{\partial \mathbf{x_j}\partial \mathbf{x_k}}\right] \tag{3.36}$$

Comme, pour tous j,k, $\xi_j\xi_k$ et $\frac{\partial^2 B(\mathbf{x})}{\partial \mathbf{x_j}\partial \mathbf{x_k}}$ sont indépendants :

$$\begin{aligned}\mathbb{E}\left[h(B(\mathbf{x}),\xi)\right] &= \sum_{j=1}^{d}\sum_{k=1}^{d}\mathbb{E}\left[\xi_j\xi_k\right]\mathbb{E}\left[\frac{\partial^2 B}{\partial \mathbf{x_j}\partial \mathbf{x_k}}\right] \\ &= \sum_{j=1}^{d}\sum_{k=1}^{d}\mathrm{Cov}(\xi_j,\xi_k)\,\mathbb{E}\left[\frac{\partial^2 B}{\partial \mathbf{x_j}\partial \mathbf{x_k}}\right]\end{aligned} \tag{3.37}$$

$$\begin{aligned}\mathbb{E}\left[B(\mathbf{x}+\xi)\right] &= \mathbb{E}\left[B\right]+\frac{1}{2}\sum_{j=1}^{d}\sum_{k=1}^{d}\mathrm{Cov}(\xi_j,\xi_k)\,\mathbb{E}\left[\frac{\partial^2 B}{\partial \mathbf{x_j}\partial \mathbf{x_k}}\right] \\ &= \mu_{B(\mathbf{x})}+\frac{1}{2}\sum_{j=1}^{d}\sum_{k=1}^{d}\mathrm{Cov}(\xi_j,\xi_k)\frac{\partial^2\,\mathbb{E}\left[B\right]}{\partial \mathbf{x_j}\partial \mathbf{x_k}} \\ &= \mu_{B(\mathbf{x})}+\frac{1}{2}\left\langle \Sigma, H(B)\right\rangle\end{aligned} \tag{3.38}$$

Ici, $\left\langle \Sigma, H(B)\right\rangle$ est de taille c car $H(B)$ est un tenseur de taille $c\times d\times d$ (il s'agit du hessien de B). $\langle\cdot,\cdot\rangle$ peut aussi être considéré comme la trace du produit de deux matrices, généralisé aux tenseurs. Σ est la matrice de covariance de l'incertitude de la position : elle varie en fonction du temps car la précision de la localisation n'est pas constante. Ici, Σ correspond à l'incertitude de localisation au moment où la mesure a été réalisée par le capteur.

3.3.3.3 Covariance de $B(\mathbf{x}+\xi)$

Pour la covariance :

$$\begin{aligned}\mathrm{Cov}\left(B(\mathbf{x}+\xi),B(\mathbf{x}'+\xi')\right) \simeq \mathrm{Cov}\Big(&B(\mathbf{x})+\nabla B(\mathbf{x})\xi+\frac{1}{2}h(B(\mathbf{x}),\xi)), \\ &B(\mathbf{x}')+\nabla B(\mathbf{x}')\xi'+\frac{1}{2}h(B(\mathbf{x}'),\xi')\Big)\end{aligned}$$

Nous développons l'expression obtenue, en ne conservant que les termes d'ordre inférieur ou égal à 2, notre développement limité de B étant restreint à l'ordre 2 :

$$\begin{aligned}\mathrm{Cov}\Big(B(\mathbf{x}+\xi),B(\mathbf{x}'+\xi')\Big) \simeq\ & \mathrm{Cov}(B(\mathbf{x}),B(\mathbf{x}')) \\ & +\mathrm{Cov}(\nabla B(\mathbf{x})\xi,B(\mathbf{x}'))+\mathrm{Cov}(B(\mathbf{x}),\nabla B(\mathbf{x}')\xi') \\ & +\frac{1}{2}\mathrm{Cov}(h(B(\mathbf{x}),\xi),B(\mathbf{x}')) \\ & +\frac{1}{2}\mathrm{Cov}(B(\mathbf{x}),h(B(\mathbf{x}'),\xi')) \\ & +\mathrm{Cov}\Big(\nabla B(\mathbf{x})\xi,\nabla B(\mathbf{x}')\xi'\Big)\end{aligned} \tag{3.39}$$

Nous développons chaque terme séparément car ils sont nombreux. On commence par la première covariance :

$$\begin{aligned}\mathrm{Cov}\Big(\nabla B(\mathbf{x})\xi,B(\mathbf{x}')\Big) &= \mathbb{E}\Big[(\nabla B(\mathbf{x})\xi)B(\mathbf{x}')^\top\Big]-\mathbb{E}\left[(\nabla B(\mathbf{x})\xi)\right]\mathbb{E}\Big[B(\mathbf{x}')^\top\Big] \\ &= \mathbb{E}\Big[\nabla B(\mathbf{x})\xi B(\mathbf{x}')^\top\Big]-\mathbb{E}\left[\nabla B(\mathbf{x})\right]\mathbb{E}\left[\xi\right]\mathbb{E}\Big[B(\mathbf{x}')\Big]\end{aligned} \tag{3.40}$$

car ξ et $\nabla B(\mathbf{x})$ sont indépendantes.

$$\begin{aligned}\mathrm{Cov}\Big(\nabla B(\mathbf{x})\xi,B(\mathbf{x}')\Big) &= \mathbb{E}\Big[\nabla B(\mathbf{x})\xi B(\mathbf{x}')^\top\Big]-\mathbb{E}\left[\nabla B(\mathbf{x})\right]\mathbb{E}\left[\xi\right]\mathbb{E}\Big[B(\mathbf{x}')\Big] \\ &= \mathbf{0}_{c\times c}\end{aligned} \tag{3.41}$$

car $\mathbb{E}\left[\xi\right]=\mathbf{0}_{1\times c}$. Pour $\mathbb{E}\Big[\nabla B(\mathbf{x})\xi B(\mathbf{x}')^\top\Big]$, la démonstration peut être faite terme à terme.

Le même raisonnement s'applique de façon symétrique à $\mathrm{Cov}\Big(\nabla B(\mathbf{x}),B(\mathbf{x}')\xi'\Big)$. La nouvelle expresion de la covariance devient :

$$\begin{aligned}\mathrm{Cov}\Big(B(\mathbf{x}+\xi),B(\mathbf{x}'+\xi')\Big) \simeq\ & \mathrm{Cov}(B(\mathbf{x}),B(\mathbf{x}'))+\mathrm{Cov}(\nabla B(\mathbf{x})\xi,\nabla B(\mathbf{x}')\xi') \\ & +\frac{1}{2}\mathrm{Cov}(h(B(\mathbf{x}),\xi),B(\mathbf{x}'))+\frac{1}{2}\mathrm{Cov}(B(\mathbf{x}),h(B(\mathbf{x}'),\xi'))\end{aligned}$$

Nous calculons maintenant les termes non nuls :

$$\begin{aligned}\mathrm{Cov}\Big(h(B(\mathbf{x}),\xi),B(\mathbf{x}')\Big) &= \mathrm{Cov}\Big(\sum_{j=1}^{d}\sum_{k=1}^{d}\xi_j\xi_k\frac{\partial^2 B(\mathbf{x})}{\partial\mathbf{x_j}\partial\mathbf{x_k}},B(\mathbf{x}')\Big) \\ &= \sum_{j=1}^{d}\sum_{k=1}^{d}\mathrm{Cov}\Big(\xi_j\xi_k\frac{\partial^2 B(\mathbf{x})}{\partial\mathbf{x_j}\partial\mathbf{x_k}},B(\mathbf{x}')\Big) \\ &= \sum_{j=1}^{d}\sum_{k=1}^{d}\mathbb{E}\Big[\xi_j\xi_k\frac{\partial^2 B(\mathbf{x})}{\partial\mathbf{x_j}\partial\mathbf{x_k}}B(\mathbf{x}')^\top\Big]-\mathbb{E}\Big[\xi_j\xi_k\frac{\partial^2 B(\mathbf{x})}{\partial\mathbf{x_j}\partial\mathbf{x_k}}\Big]\mathbb{E}\Big[B(\mathbf{x}')^\top\Big]\end{aligned}$$

ξ est un bruit blanc Gaussien, indépendant des valeurs de $B(\mathbf{x})$:

$$\begin{aligned}\mathrm{Cov}\left(h(B(\mathbf{x}),\xi),B(\mathbf{x}')\right) &= \sum_{j=1}^{d}\sum_{k=1}^{d}\mathbb{E}\left[\xi_j\xi_k\right]\mathbb{E}\left[\frac{\partial^2 B(\mathbf{x})}{\partial \mathbf{x_j}\partial \mathbf{x_k}}B(\mathbf{x}')^\top\right]-\mathbb{E}\left[\xi_j\xi_k\right]\mathbb{E}\left[\frac{\partial^2 B(\mathbf{x})}{\partial \mathbf{x_j}\partial \mathbf{x_k}}\right]\mathbb{E}\left[B(\mathbf{x}')^\top\right]\\ &= \sum_{j=1}^{d}\sum_{k=1}^{d}\mathbb{E}\left[\xi_j\xi_k\right]\left(\mathbb{E}\left[\frac{\partial^2 B(\mathbf{x})}{\partial \mathbf{x_j}\partial \mathbf{x_k}}B(\mathbf{x}')^\top\right]-\mathbb{E}\left[\frac{\partial^2 B(\mathbf{x})}{\partial \mathbf{x_j}\partial \mathbf{x_k}}\right]\mathbb{E}\left[B(\mathbf{x}')^\top\right]\right)\end{aligned}$$

Comme ξ est de plus de moyenne nulle, $\mathbb{E}\left[\xi_j\xi_k\right]=\mathrm{Cov}(\xi_j,\xi_k)$, nous pouvons écrire :

$$\begin{aligned}\mathrm{Cov}\left(h(B(\mathbf{x}),\xi),B(\mathbf{x}')\right) &= \sum_{j=1}^{d}\sum_{k=1}^{d}\Sigma_{j,k}\,\mathrm{Cov}\left(\frac{\partial^2 B(\mathbf{x})}{\partial \mathbf{x_j}\partial \mathbf{x_k}},B(\mathbf{x}')\right)\\ &= \sum_{j=1}^{d}\sum_{k=1}^{d}\Sigma_{j,k}\frac{\partial^2}{\partial \mathbf{x_j}\partial \mathbf{x_k}}\left[\mathrm{Cov}\left(B(\mathbf{x}),B(\mathbf{x}')\right)\right]\\ &= \sum_{j=1}^{d}\sum_{k=1}^{d}\Sigma_{j,k}\frac{\partial^2\kappa(\mathbf{x},\mathbf{x}')}{\partial \mathbf{x_j}\partial \mathbf{x_k}} \qquad (3.42)\\ \mathrm{Cov}\left(h(B(\mathbf{x}),\xi),B(\mathbf{x}')\right) &= \left\langle \Sigma, H_1(\kappa)(\mathbf{x},\mathbf{x}')\right\rangle \qquad (3.43)\end{aligned}$$

Nous obtenons le même résultat pour $\mathrm{Cov}\left(B(\mathbf{x}),h(B(\mathbf{x}'),\xi')\right)$, à l'exception que, cette fois-ci, c'est la matrice de covariance de ξ' : on note celle-ci Σ', et on calcule la Hessienne par rapport au deuxième argument : $H_2(\kappa)(\mathbf{x},\mathbf{x}')$:

$$\mathrm{Cov}\left(B(\mathbf{x},h(B(\mathbf{x}'),\xi'))\right)=\left\langle \Sigma', H_2(\kappa)(\mathbf{x},\mathbf{x}')\right\rangle. \qquad (3.44)$$

Pour résumer, à ce stade des calculs, nous avons calculé tous les termes, à l'exception de $\mathrm{Cov}\left(\nabla B(\mathbf{x})\xi,\nabla B(\mathbf{x}')\xi'\right)$. L'expression actuelle de la covariance, avec les simplifications, est :

$$\begin{aligned}\mathrm{Cov}\left(B(\mathbf{x}+\xi),B(\mathbf{x}'+\xi')\right) \simeq \mathrm{Cov}\left(B(\mathbf{x}),B(\mathbf{x}')\right) &+ \frac{1}{2}\left\langle \Sigma, H_1(\kappa)(\mathbf{x},\mathbf{x}')\right\rangle\\ &+ \frac{1}{2}\left\langle \Sigma', H_2(\kappa)(\mathbf{x},\mathbf{x}')\right\rangle\\ &+ \mathrm{Cov}\left(\nabla B(\mathbf{x})\xi,\nabla B(\mathbf{x}')\xi'\right) \qquad (3.45)\\ \simeq \kappa(\mathbf{x},\mathbf{x}') + \frac{1}{2}\left(\left\langle \Sigma, H_1(\kappa)(\mathbf{x},\mathbf{x}')\right\rangle + \left\langle \Sigma', H_2(\kappa)(\mathbf{x},\mathbf{x}')\right\rangle\right)&\\ &+ \mathrm{Cov}\left(\nabla B(\mathbf{x})\xi,\nabla B(\mathbf{x}')\xi'\right) \qquad (3.46)\end{aligned}$$

Nous calculons le dernier terme de la covariance :

$$\begin{aligned}\mathrm{Cov}\left(\nabla B(\mathbf{x})\xi,\nabla B(\mathbf{x}')\xi'\right) &= \mathbb{E}\left[\nabla B(\mathbf{x})\xi(\nabla B(\mathbf{x}')\xi')^\top\right]-\mathbb{E}\left[\nabla B(\mathbf{x})\xi\right]\mathbb{E}\left[\nabla B(\mathbf{x}')\xi'\right]^\top\\ &= \mathbb{E}\left[\nabla B(\mathbf{x})\xi\xi'^\top\nabla B(\mathbf{x}')^\top\right]-\mathbb{E}\left[\nabla B(\mathbf{x})\right]\mathbb{E}\left[\xi\right]\mathbb{E}\left[\xi'\right]^\top\mathbb{E}\left[\nabla B(\mathbf{x}')\right]^\top\\ &= \mathbb{E}\left[\nabla B(\mathbf{x})\xi\xi'^\top\nabla B(\mathbf{x}')^\top\right] \qquad (3.47)\end{aligned}$$

Pour chaque terme, nous obtenons :

$$\begin{aligned}
\mathrm{Cov}\left(\nabla B(\mathbf{x})\xi, \nabla B(\mathbf{x}')\xi'\right)_{i,j} &= \mathbb{E}\left[\sum_{k=1}^{d}\sum_{l=1}^{d} \nabla B_{i,k}(\mathbf{x})(\xi\xi'^{\top})_{k,l}\nabla B_{j,l}(\mathbf{x}')\right] \\
&= \sum_{k=1}^{d}\sum_{l=1}^{d} \mathbb{E}\left[(\xi\xi'^{\top})_{k,l}\right]\mathbb{E}\left[\nabla B_{i,k}(\mathbf{x})\nabla B_{j,l}(\mathbf{x}')\right] \\
&= \sum_{k=1}^{d}\sum_{l=1}^{d} \mathrm{Cov}(\xi_k, \xi'_l)\,\mathbb{E}\left[\nabla B_{i,k}(\mathbf{x})\nabla B_{j,l}(\mathbf{x}')\right] \qquad (3.48)
\end{aligned}$$

Nous calculons maintenant le dernier terme d'ordre 2 de la covariance $\mathrm{Cov}(B(\mathbf{x})\xi, B(\mathbf{x}')\xi')$. Si les deux mesures ont lieu à différents instants, ξ et ξ' étant supposés indépendants, cette covariance est supposée nulle. Ce n'est pas toujours le cas (en particulier pour la localisation, où le calcul de la position dépend de l'estimation de la position antérieure) mais cela permet de simplifier les calculs. Dans le cas où $\xi = \xi'$, nous calculons $\mathrm{Cov}(\nabla B(\mathbf{x})\xi, \nabla B(\mathbf{x})\xi) = \mathrm{Var}(\nabla B(\mathbf{x})\xi)$. Sachant que $\mathbb{E}\left[(\nabla B(\mathbf{x})\xi)\right] = \mathbb{E}\left[\nabla B(\mathbf{x})\right]\mathbb{E}\left[\xi\right] = \mathbf{0}_c$:

$$\begin{aligned}
\left(\mathrm{Var}(\nabla B(\mathbf{x})\xi)\right)_{i,j} &= \sum_{k=1}^{d}\sum_{l=1}^{d} \Sigma_{k,l}\,\mathbb{E}\left[\nabla B_{i,k}(\mathbf{x})\nabla B_{j,l}(\mathbf{x})\right] \\
&= \sum_{k=1}^{d}\sum_{l=1}^{d} \Sigma_{k,l}\left(\mathrm{Cov}(\nabla B_{i,k}(\mathbf{x}), \nabla B_{j,l}(\mathbf{x})) + \mathbb{E}\left[\nabla B_{i,k}(\mathbf{x})\right]\mathbb{E}\left[\nabla B_{j,l}(\mathbf{x})\right]\right) \\
&= \sum_{k=1}^{d}\sum_{l=1}^{d} \Sigma_{k,l}\left(\frac{\partial^2 \kappa_{i,j}(\mathbf{x},\mathbf{x})}{\partial x_k^1 \partial x_l^2} + \mathbb{E}\left[\nabla B_{i,k}(\mathbf{x})\right]\mathbb{E}\left[\nabla B_{j,l}(\mathbf{x})\right]\right) \\
&= \mathbb{E}\left[\nabla B_i(\mathbf{x})\right]\mathrm{Var}(\xi)\,\mathbb{E}\left[\nabla B_j(\mathbf{x})\right]^{\top} + \sum_{k=1}^{d}\sum_{l=1}^{d} \Sigma_{k,l}\frac{\partial^2 \kappa_{i,j}(\mathbf{x},\mathbf{x})}{\partial x_k^1 \partial x_l^2} \qquad (3.49)
\end{aligned}$$

Les exposants indiquent par rapport à laquelle des deux variables (la première ou la deuxième) la dérivée est effectuée.

$$\left(\mathrm{Var}(\nabla B(\mathbf{x})\xi)\right) = \mathbb{E}\left[\nabla B(\mathbf{x})\right]\mathrm{Var}(\xi)\,\mathbb{E}\left[\nabla B(\mathbf{x})\right]^{\top} + \sum_{k=1}^{d}\sum_{l=1}^{d} \Sigma_{k,l}\frac{\partial^2 \kappa(\mathbf{x},\mathbf{x})}{\partial \mathbf{x}_k^1 \partial \mathbf{x}_l^2} \qquad (3.50)$$

3.3.3.4 Calcul des prédictions du modèle à partir des données

Grâce à la partie précédente, nous déduisons l'espérance et la covariance de B_{LE}, qui vaut $B(\mathbf{x}+\xi)+\varepsilon$. ε correspond au bruit du capteur et suit la loi $\mathcal{N}(\mathbf{0}_{c\times 1}, \sigma_b^2 I_c)$. De plus, comme les incertitudes de localisation dépendent du temps, nous indexons les matrices Σ par le temps où la mesure a été effectuée : par la suite, t^{α} et t^{ν}.

$$\mathbb{E}\left[B_{LE}(\alpha)\right] \simeq \mathbb{E}\left[B(\mathbf{x}^\alpha)\right] + \frac{1}{2}\left\langle \Sigma^{t^\alpha}, H(B)(\mathbf{x}^\alpha)\right\rangle \tag{3.51}$$

$$\begin{aligned}\mathrm{Cov}\left(B_{LE}(\alpha), B_{LE}(\nu)\right) \simeq \mathrm{Cov}\left(B(\mathbf{x}^\alpha), B(\mathbf{x}^\nu)\right) &+ \frac{1}{2}\left\langle \Sigma^{t^\alpha}, H_1(\kappa)(\mathbf{x}^\alpha, \mathbf{x}^\nu)\right\rangle \\ &+ \frac{1}{2}\left\langle \Sigma^{t^\nu}, H_2(\kappa)(\mathbf{x}^\alpha, \mathbf{x}^\nu)\right\rangle \\ &+ \delta_{\alpha,\nu}(\Sigma_b + \mathrm{Var}(\nabla B(\mathbf{x}^\alpha)\xi^\alpha))\end{aligned} \tag{3.52}$$

avec $\delta_{\alpha,\nu}$ la notation de Kronecker.

Si deux mesures n'ont pas été effectuées au même instant, nous faisons de nouveau l'hypothèse que les bruits de mesure ξ sont indépendants : si et seulement si $\alpha = \nu$, la variance liée au bruit de mesure apparaît dans la covariance liée à B_{LE}.

Nous calculons la covariance entre $B(\mathbf{x})$ et $B_{LE}(\alpha)$, pour pouvoir ensuite écrire leur loi jointe :

$$\begin{aligned}\mathrm{Cov}\left(B(\mathbf{x}), B_{LE}(\alpha)\right) &\simeq \mathrm{Cov}\left(B(\mathbf{x}), B(\mathbf{x}^\alpha) + \nabla B(\mathbf{x}^\alpha)\xi + \frac{1}{2}h(B(\mathbf{x}^\alpha), \xi)\right) \\ &\simeq \mathrm{Cov}\left(B(\mathbf{x}), B(\mathbf{x}^\alpha)\right) + \mathrm{Cov}\left(B(\mathbf{x}), \nabla B(\mathbf{x}^\alpha)\xi\right) \\ &\quad + \frac{1}{2}\mathrm{Cov}\left(B(\mathbf{x}), h(B(\mathbf{x}^\alpha), \xi)\right) \\ &\simeq \kappa(\mathbf{x}, \mathbf{x}^\alpha) + \frac{1}{2}\left\langle \Sigma^{t^\alpha}, H_2(\kappa)(\mathbf{x}, \mathbf{x}^\alpha)\right\rangle\end{aligned} \tag{3.53}$$

$\mathrm{Cov}\left(B(\mathbf{x}), \nabla B(\mathbf{x}^\alpha)\xi\right) = \mathbf{0}_{c\times c}$ car ξ est indépendant de $B(\mathbf{x}^\alpha)$ et de $B(\mathbf{x})$.

En utilisant les équations de conditionnement d'un processus gaussien [RW06], nous écrivons la loi jointe de B et B_{LE}, avec $\mathbf{b}$ une réalisation de B_{LE} :

$$\begin{bmatrix} B(\mathbf{x}) \\ B(\mathbf{x}') \\ B_{LE}(\mathbf{x}^\alpha) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu_{B(\mathbf{x})} \\ \mu_{B(\mathbf{x}')} \\ \mu_{B_{LE}(\mathbf{x}^\alpha)} \end{bmatrix}, \begin{bmatrix} K_1 & K_2 \\ K_2^\top & K_3 \end{bmatrix}\right) \tag{3.54}$$

K_1 reste identique : il s'agit de la même matrice de covariance :

$$K_1 = \begin{bmatrix} \kappa(\mathbf{x}, \mathbf{x}) & \kappa(\mathbf{x}, \mathbf{x}') \\ \kappa(\mathbf{x}, \mathbf{x}')^\top & \kappa(\mathbf{x}', \mathbf{x}') \end{bmatrix} \tag{3.55}$$

Les expressions de K_2 et K_3 deviennent, avec Σ^{t^α} l'incertitude de localisation de la mesure qui a été réalisée au temps t^α, au niveau du point x^α :

$$K_2 \simeq \begin{bmatrix} \kappa(\mathbf{x}, \mathbf{x}^\alpha) + \frac{1}{2}\left\langle \Sigma^{t^\alpha}, H_2(\kappa)(\mathbf{x}, \mathbf{x}^\alpha)\right\rangle \\ \kappa(\mathbf{x}', \mathbf{x}^\alpha) + \frac{1}{2}\left\langle \Sigma^{t^\alpha}, H_2(\kappa)(\mathbf{x}', \mathbf{x}^\alpha)\right\rangle \end{bmatrix} \tag{3.56}$$

$$K_3 \simeq \left[\kappa(\mathbf{x}^\alpha, \mathbf{x}^\alpha) + \left\langle \Sigma^{t^\alpha}, H_1(\kappa)(\mathbf{x}^\alpha, \mathbf{x}^\alpha)\right\rangle + \sigma_b^2 I_c + \mathrm{Var}\left(\nabla B(\mathbf{x}^\alpha)\xi^\alpha\right)\right] \tag{3.57}$$

L'expression développée du calcul de $H_2(\kappa)(\mathbf{x},\mathbf{x}^\alpha)$ s'appuie sur l'expression trouvée en 3.42 :

$$\begin{aligned}&\frac{1}{2}\left(\left\langle \Sigma, H_1(\kappa)(\mathbf{x},\mathbf{x}')\right\rangle + \left\langle \Sigma', H_2(\kappa)(\mathbf{x},\mathbf{x}')\right\rangle\right)\\ &= \frac{1}{2}\left(\sum_{j=1}^{d}\sum_{k=1}^{d}\Sigma_{j,k}\frac{\partial^2\kappa(\mathbf{x},\mathbf{x}')}{\partial\mathbf{x_j}\partial\mathbf{x_k}} + \sum_{j=1}^{d}\sum_{k=1}^{d}\Sigma'_{j,k}\frac{\partial^2\kappa(\mathbf{x},\mathbf{x}')}{\partial\mathbf{x'_j}\partial\mathbf{x'_k}}\right)\end{aligned} \tag{3.58}$$

avec Σ' la covariance de l'erreur de localisation au point $\mathbf{x}'$ (et dépendant de l'instant où la mesure a été réalisée). Si $\mathbf{x}=\mathbf{x}'$, comme la covariance est symmétrique, $H_1(\kappa)(\mathbf{x},\mathbf{x}) = H_2(\kappa)(\mathbf{x},\mathbf{x})$.

La différence par rapport aux formules sans erreurs de localisation consiste en l'apparition de matrices hessiennes. La variance obtenue est aussi différente de celle obtenue dans [BSvV16] : la matrice symétrique de $\frac{1}{2}\left\langle \Sigma^{t^\alpha}, H_1(\kappa)(\mathbf{x}^\alpha,\mathbf{x}^\alpha)\right\rangle$ apparaît dans l'expression de la variance, contrairement à ce qui est obtenu dans [BSvV16]. Ce terme supplémentaire garantit la symétrie de la variance pour les cas où $\mathbf{x}$ n'est pas mono-dimensionnel.

K_1 est de taille $2c\times 2c$, K_2 de taille $2c\times c$ et K_3 de taille $c\times c$. $\begin{bmatrix}B(\mathbf{x})\\B(\mathbf{x}')\end{bmatrix}$ est de taille $2c\times 1$.

Le système à résoudre est le même que pour un processus Gaussien classique [RW06], mais les expressions des matrices K_2 et K_3 sont différentes (nous les avons calculées plus haut) :

$$\boxed{\mathbb{E}\left[\begin{bmatrix}B(\mathbf{x})\\B(\mathbf{x}')\end{bmatrix}|\mathbf{b}\right] = \begin{bmatrix}\mu_{B(\mathbf{x})}\\\mu_{B(\mathbf{x}')}\end{bmatrix} + K_2K_3^{-1}(\mathbf{b}-\mu_{B_{LE}(\mathbf{x}^\alpha)})} \tag{3.59}$$

$$\boxed{\mathrm{Cov}\left(\begin{bmatrix}B(\mathbf{x})\\B(\mathbf{x}')\end{bmatrix},\begin{bmatrix}B(\mathbf{x})\\B(\mathbf{x}')\end{bmatrix}|\mathbf{b}\right) = \mathrm{Cov}\left(\begin{bmatrix}B(\mathbf{x})\\B(\mathbf{x}')\end{bmatrix},\begin{bmatrix}B(\mathbf{x})\\B(\mathbf{x}')\end{bmatrix}\right) - K_2K_3^{-1}K_2^\top} \tag{3.60}$$

$\mathbb{E}\left[\begin{bmatrix}B(\mathbf{x})\\B(\mathbf{x}')\end{bmatrix}|b\right]$ est de taille $1\times 2d$. Le sous-vecteur de ces troix premières composantes correspond à l'espérance de $B(\mathbf{x})$ conditionné à la valeur $\mathbf{b}$ (et pour $B(\mathbf{x}')$, il s'agit des d dernières).

$\mathrm{Cov}\left(\begin{bmatrix}B(\mathbf{x})\\B(\mathbf{x}')\end{bmatrix},\begin{bmatrix}B(\mathbf{x})\\B(\mathbf{x}')\end{bmatrix}|\mathbf{b}\right)$ est de taille $2c\times 2c$: elle est composée de quatre sous-blocs de taille $c\times c$:

$$\mathrm{Cov}\left(\begin{bmatrix}B(\mathbf{x})\\B(\mathbf{x}')\end{bmatrix},\begin{bmatrix}B(\mathbf{x})\\B(\mathbf{x}')\end{bmatrix}|\mathbf{b}\right) = \begin{bmatrix}\mathrm{Var}\left(B(\mathbf{x})|\mathbf{b}\right) & \mathrm{Cov}\left(B(\mathbf{x}),B(\mathbf{x}')|\mathbf{b}\right)\\ \mathrm{Cov}\left(B(\mathbf{x}),B(\mathbf{x}')|\mathbf{b}\right) & \mathrm{Var}\left(B(\mathbf{x}')|\mathbf{b}\right)\end{bmatrix} \tag{3.61}$$

Il suffit d'extraire un des sous-blocs non-diagonaux pour obtenir la covariance entre $B(\mathbf{x})$ et $B(\mathbf{x}')$: $\mathrm{Cov}\left(B(\mathbf{x}),B(\mathbf{x}')|\mathbf{b}\right)$.

3.3.4 Adaptation à un Processus Gaussien à rang réduit

Le problème est le même dans cette sous-partie que dans la précédente, mais cette fois nous l'adaptons à un processus gaussien à rang réduit, pour que la taille des matrices de covariance soit suffisamment faible pour permettre de résoudre le système, même en présence d'un très grand nombre de données. Cette fois-ci, plusieurs mesures peuvent être incorporées simultanément dans le modèle.

Nous voulons calculer l'espérance et la covariance du champ B au point $\mathbf{x}^\star$, sachant que sa covariance est de la forme suivante, à rang réduit, comme dans l'Equation 3.13 :

$$\kappa(\mathbf{x}^\star, \mathbf{x}) = \phi(\mathbf{x}^\star)\Lambda\phi(\mathbf{x})^\top. \tag{3.62}$$

Le résultat final est indiqué dans l'encadré et démontré dans les pages qui suivent.

Processus Gaussien *à rang réduit* avec incertitude sur les données d'entrée et de sortie

Processus Gaussien à rang réduit : covariances initiales et bruit sur les sorties
Notre méthode : ajout du bruit sur les entrées
Equations similaires, mais ajout de matrices supplémentaires liées aux fonctions propres : Θ et D_{Var}.
La sortie est de dimension 1 ($c = 1$) car la formulation à rang réduit telle qu'utilisée ici n'estime les paramètres que d'un unique processus gaussien.
$B(\mathbf{x}^{\star 1}, \ldots, \mathbf{x}^{\star p}) \sim \mathcal{N}(\mu_{B_\star}, \text{Var}(B_\star))$ et $B(\mathbf{x}^1, \ldots, \mathbf{x}^n) \sim \mathcal{N}(\mu_B, \text{Var}(B))$.
$\mathbf{b}^1, \ldots \mathbf{b}^n$ sont des réalisations de B aux points $\mathbf{x}^1, \ldots, \mathbf{x}^n$.

$$\mathbb{E}\left[B(\mathbf{x}^{\star 1}, \ldots, \mathbf{x}^{\star p}) | \mathbf{b}^1, \ldots \mathbf{b}^n\right] = \mu_{B_\star} + \Phi_\star\Lambda\left(\Phi^\top + \frac{1}{2}\Theta^\top\right)$$
$$* \left(\Phi\Lambda\Phi^\top + D_{\Sigma_b,n} + \frac{1}{2}(\Phi\Lambda\Theta^\top + (\Phi\Lambda\Theta^\top)^\top) + D_{\text{Var}}\right)^{-1} (\mathbf{b} - \mu_{B_{LE}})$$

avec $\mu_{B_{LE}} = \mu_B + \frac{1}{2}\left\langle \Sigma^t, H(B) \right\rangle$.

$$\text{Var}(B(\mathbf{x}^{1,\star}, \ldots, \mathbf{x}^{p,\star}) | \mathbf{b}^1, \ldots \mathbf{b}^n) = \text{Var}(B_\star) - \Phi_\star\Lambda\left(\Phi^\top + \frac{1}{2}\Theta^\top\right)$$
$$* \left(\Phi\Lambda\Phi^\top + D_{\Sigma_b,n}\frac{1}{2}(\Phi\Lambda\Theta^\top + \Theta\Lambda\Phi^\top) + D_{\text{Var}}\right)^{-1}$$
$$* \left(\Phi^\top + \frac{1}{2}\Theta^\top\right)^\top \Lambda^\top \Phi_\star^\top$$

avec

$$\Theta = \begin{bmatrix} \zeta_{\mathbf{x}^1} \\ \ldots \\ \zeta_{\mathbf{x}^n} \end{bmatrix} \quad \text{et} \quad \zeta_{\mathbf{x}^i} = \sum_{j=1}^{d}\sum_{k=1}^{d} \Sigma_{j,k} \frac{\partial^2 \phi^\top}{\partial \mathbf{x}^i{}_j \partial \mathbf{x}^i{}_k}$$

D_{Var} est la matrice diagonale par blocs dont chaque bloc vaut $\text{Var}\left(\nabla B(\mathbf{x_i})\xi_i\right)$, avec i allant de 1 à n. Elle est de taille $n \times n$ (voir l'Equation 3.49 pour son expression développée pour les calculs numériques).

$$D_{\text{Var}} = \begin{bmatrix} \text{Var}\left(\nabla B(\mathbf{x}_1)\xi_1\right) & 0 & \mathbf{0}_{1\times(n-2)} \\ 0 & \text{Var}\left(\nabla B(\mathbf{x_2})\xi_2\right) & \mathbf{0}_{1\times(n-2)} \\ \dots & \dots & \dots \end{bmatrix}. \tag{3.63}$$

$D_{\Sigma_b,n}$ est composée de $n \times n$ valeurs. Seuls les éléments diagonaux sont non-nuls et égaux à Σ_b (Σ_b est un scalaire car B est supposé mono-dimensionnel) :

$$D_{\Sigma_b,n} = \begin{bmatrix} \Sigma_b & 0 & \dots & 0 \\ 0 & \Sigma_b & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \Sigma_b. \end{bmatrix} \tag{3.64}$$

Démonstration

Nous cherchons à obtenir l'espérance et la covariance en tout point, nous posons $\mathbf{x}^\star$ et $\mathbf{x}^\diamond$ deux points quelconques où des mesures sont effectuées.

Le conditionnement s'effectue sur une valeur mesurée au point $\mathbf{x}^\alpha$, avec α l'index de la mesure. La mesure de $\mathbf{x}^\alpha$ n'est pas parfaite et par conséquent il faut introduire une incertitude sur $\mathbf{x}^\alpha$. Il nous faut donc d'abord calculer K_2 et K_3.

Pour plus de simplicité, nous nous plaçons d'abord dans le cas où une unique nouvelle mesure est ajoutée au modèle. $\mathbf{x}^\alpha = \mathbf{x}$, $\Sigma^{t^\alpha} = \Sigma$, $\phi = \phi(\mathbf{x}) = \phi(\mathbf{x}^\alpha)$, $\phi_\star = \phi(\mathbf{x}^\star)$ et $\phi_\diamond = \phi(\mathbf{x}_\diamond)$. Nous calculons $\left\langle \Sigma, H_2(\kappa)(\mathbf{x}^\star, \mathbf{x}) \right\rangle$:

$$\left\langle \Sigma, H_2(\kappa)(\mathbf{x}^\star, \mathbf{x}) \right\rangle = \left\langle \Sigma, H_2(\kappa)(\mathbf{x}^\star, \mathbf{x})^\top \right\rangle \tag{3.65}$$

$$= \sum_{j=1}^{d} \sum_{k=1}^{d} \Sigma_{j,k} \frac{\partial^2 \phi_\star \Lambda \phi^\top}{\partial \mathbf{x}_j \partial \mathbf{x}_k} \tag{3.66}$$

$\phi_\star \Lambda$ ne dépend pas de la variable $\mathbf{x}$ donc :

$$\left\langle \Sigma, H_2(\kappa)(\mathbf{x}^\star, \mathbf{x}) \right\rangle = \sum_{j=1}^{d} \sum_{k=1}^{d} \Sigma_{j,k} \phi_\star \Lambda \frac{\partial^2 \phi^\top}{\partial \mathbf{x}_j \partial \mathbf{x}_k} \tag{3.67}$$

Comme $\Sigma_{j,k}$ est un scalaire :

$$\left\langle \Sigma, H_2(\kappa)(\mathbf{x}^\star, \mathbf{x}) \right\rangle = \phi_\star \Lambda \sum_{j=1}^{d} \sum_{k=1}^{d} \Sigma_{j,k} \frac{\partial^2 \phi^\top}{\partial \mathbf{x}_j \partial \mathbf{x}_k} \tag{3.68}$$

$$= \phi_\star \Lambda \zeta \tag{3.69}$$

Nous pouvons donc maintenant expliciter K_2 et K_3, sachant que Λ est symétrique par construction :

$$K_2 \simeq \begin{bmatrix} \phi_\star \Lambda \phi^\top + \frac{1}{2} \phi_\star \Lambda \zeta \\ \phi_\diamond \Lambda \phi^\top + \frac{1}{2} \phi_\diamond \Lambda \zeta \end{bmatrix} \tag{3.70}$$

$$K_3 \simeq \left[\phi\Lambda\phi^\top + \tfrac{1}{2}(\phi\Lambda\zeta + (\phi\Lambda\zeta)^\top) + \Sigma_b + \mathrm{Var}\left(\nabla B(\mathbf{x}^\alpha)\xi^\alpha\right)\right] \tag{3.71}$$

avec Σ_b la matrice correspondant à la variance du bruit du capteur, de taille $c \times c$.
L'espérance, conditionnée à la donnée $\mathbf{x}^\alpha$, est égale à :

$$\begin{aligned}
\mathbb{E}\left[B(\mathbf{x}^\star)|\mathbf{b}\right] &= \mu_{B(\mathbf{x}^\star)} + K_2 K_3^{-1}(\mathbf{b} - \mu_{B_{LE}(\mathbf{x}^\alpha)}) \\
&= \mu_{B(\mathbf{x}^\star)} + \phi_\star\left(\Lambda\phi^\top + \frac{1}{2}\Lambda\zeta\right) \\
&\qquad * \left(\phi\Lambda\phi^\top + \frac{1}{2}(\phi\Lambda\zeta + (\phi\Lambda\zeta)^\top) + \Sigma_b + \mathrm{Var}\left(\nabla B(\mathbf{x}^\alpha)\xi^\alpha\right)\right)^{-1} \\
&\qquad * (\mathbf{b} - \mu_{B_{LE}(\mathbf{x}^\alpha)})
\end{aligned} \tag{3.72}$$

Dans le cas de plusieurs valeurs, nous notons :

$$\begin{aligned}
vec(\mathbf{b_1}, ...\mathbf{b_n}) &= \mathbf{b} \\
vec(\mu_{B_{LE}(\mathbf{x_1})}, ..., \mu_{B_{LE}(\mathbf{x_n})}) &= \mu_{B_{LE}}.
\end{aligned}$$

Pour K_3 :

$$\begin{aligned}
&\frac{1}{2}\left(\left\langle \Sigma^p, H_1(\kappa)(\mathbf{x}^p, \mathbf{x}^q)\right\rangle + \left\langle \Sigma^q, H_2(\kappa)(\mathbf{x}^p, \mathbf{x}^q)\right\rangle\right) \\
&= \frac{1}{2}\left(\sum_{j=1}^{d}\sum_{k=1}^{d} \Sigma^p_{j,k}\frac{\partial^2\phi_p\Lambda\phi_q^\top}{\partial x_j^p \partial x_k^p} + \sum_{j=1}^{d}\sum_{k=1}^{d} \Sigma^q_{j,k}\frac{\partial^2\phi_p\Lambda\phi_q^\top}{\partial \mathbf{x}_j^q \partial \mathbf{x}_k^q}\right)
\end{aligned} \tag{3.73}$$

Θ est la matrice qui généralise le calcul à tous les points $\mathbf{x^1}, ...\mathbf{x^n}$:

$$\Theta = \begin{bmatrix} \zeta_{\mathbf{x^1}} \\ ... \\ \zeta_{\mathbf{x^n}} \end{bmatrix} \tag{3.74}$$

avec chaque sous matrice $\xi_{\mathbf{x^i}}$ de taille $d \times (d+m)$.

Pour un conditionnement avec plusieurs données, $\mathbf{x}^1$, ... $\mathbf{x}^n$, en définissant $\Phi = \begin{bmatrix} \phi(\mathbf{x}^1) \\ ... \\ \phi(\mathbf{x}^n) \end{bmatrix}$ (et de même pour $\phi_\star$), nous pouvons généraliser le calcul :

$$\begin{aligned}
\mathbb{E}\left[B(\mathbf{x}^{1,\star}, ..., \mathbf{x}^{n,\star})|\mathbf{b}^1, ...\mathbf{b}^n\right] &= \mu_{B_\star} + K_2 K_3^{-1}(\mathbf{b} - \mu_{B_{LE}}) \\
&= \mu_{B_\star} + \left(\Phi_\star\Lambda\Phi^\top + \frac{1}{2}\Phi_\star\Lambda\Theta^\top\right) \\
&\quad * \left(\Phi\Lambda\Phi^\top + \frac{1}{2}(\Phi\Lambda\Theta^\top + \Theta\Lambda\Phi^\top) + D_{\Sigma_b,cn} + D_{\mathrm{Var}}\right)^{-1}(\mathbf{b} - \mu_{B_{LE}}) \\
\mathbb{E}\left[B(\mathbf{x}^{1,\star}, ..., \mathbf{x}^{n,\star})|\mathbf{b}^1, ...\mathbf{b}^n\right] &= \mu_{B_\star} + \Phi_\star\left(\Lambda\Phi^\top + \frac{1}{2}\Lambda\Theta^\top\right) \\
&\quad * \left(\Phi\Lambda\Phi^\top + \frac{1}{2}(\Phi\Lambda\Theta^\top + \Theta\Lambda\Phi^\top) + D_{\Sigma_b,cn} + D_{\mathrm{Var}}\right)^{-1}(\mathbf{b} - \mu_{B_{LE}})
\end{aligned} \tag{3.75}$$

avec :

$$\mu_{B_{LE}} = \mu_B + \frac{1}{2}\left\langle \Sigma^t, H(B) \right\rangle. \tag{3.76}$$

La variance est calculée à partir de la formule générale 3.60 :

$$\mathrm{Var}(B(\mathbf{x}^{1,\star},...,\mathbf{x}^{n,\star})|\mathbf{b}^1,...\mathbf{b}^n) = \mathrm{Var}(B_\star) - K_2 K_3^{-1} K_2^\top \tag{3.77}$$

En remplaçant les expressions du système par leur écriture à rang réduit :

$$\begin{aligned}\mathrm{Var}(B(\mathbf{x}^{1,\star},...,\mathbf{x}^{n,\star})|\mathbf{b}^1,...\mathbf{b}^n) = \mathrm{Var}(B_\star) - &\left(\Phi_\star\Lambda\Phi^\top + \frac{1}{2}\Phi_\star\Lambda\Theta^\top\right)\\ &* \left(\Phi\Lambda\Phi^\top + \frac{1}{2}(\Phi\Lambda\Theta^\top + \Theta\Lambda\Phi^\top) + D_{\Sigma_b,cn} + D_{\mathrm{Var}}\right)^{-1}.\\ &* \left(\Phi_\star\Lambda\Phi^\top + \frac{1}{2}\Phi_\star\Lambda\Theta^\top\right)^\top\end{aligned} \tag{3.78}$$

La formule finale est :

$$\begin{aligned}\mathrm{Var}(B(\mathbf{x}^{1,\star},...,\mathbf{x}^{n,\star})|\mathbf{b}^1,...\mathbf{b}^n) = \mathrm{Var}(B_\star) - &\Phi_\star\Lambda\left(\Phi^\top + \frac{1}{2}\Theta^\top\right)\\ &* \left(\Phi\Lambda\Phi^\top + \frac{1}{2}(\Phi\Lambda\Theta^\top + \Theta\Lambda\Phi^\top) + D_{\Sigma_b,cn} + D_{\mathrm{Var}}\right)^{-1}\\ &* \left(\Phi^\top + \frac{1}{2}\Theta^\top\right)^\top \Lambda^\top\Phi_\star^\top\end{aligned} \tag{3.79}$$

Θ est de taille $c \times (d+m)$, avec m le nombre de fonctions propres ϕ_j. Ses d premières colonnes sont nulles car il s'agit de la dérivation à l'ordre 3 de ses coordonnées. Les matrices Θ et D_{Var} sont composées respectivement des dérivées secondes et premières des fonctions ϕ_j : leurs expressions sont calculées explicitement dans l'Annexe A.

Il est possible de généraliser pour une sortie à plusieurs dimensions si les différentes composantes de B sont indépendantes (d'une manière similaire à celle mentionnée dans [BSvV16], en étudiant chaque composante du champ séparément).

3.3.5 Adaptation pour un Processus Gaussien itératif à rang réduit : RRONIG (Reduced-Rank Online Noisy Input GP)

Les formules précédentes permettent d'obtenir l'espérance et la covariance avec toutes les données simultanément. L'objectif de cette Section est d'adapter les équations à une mise à jour itérative du modèle (comme dans la Section 3.1.5 mais adapté à notre modèle, prenant en compte l'incertitude des entrées). Nous nommons le modèle obtenu RRONIG (Reduced-Rank Online Noisy Input GP) car, comme SONIG, il s'agit d'une régression par processus Gaussiens prenant en compte le bruit sur les données d'entrées, mais cette fois-ci en utilisant l'approximation par rang réduit.

L'espérance et la variance s'écrivent sous la forme suivante, avec l le numéro de la dernière itération ayant servi à mettre à jour le modèle :

$$\begin{cases}\mathbb{E}[B(\mathbf{x}^*)] \approx \Phi_*\mathbf{\Gamma}_l\\ \mathbb{V}[B(\mathbf{x}^*)] \approx \Phi_*\Upsilon_l\Phi_*^\top\end{cases} \tag{3.80}$$

$\mathbf{\Gamma}_l$ est un vecteur de taille m avec des valeurs intiales toutes égales à 0 comme le GP initial est supposé avoir une moyenne nulle. Υ_l est une matrice carrée de taille $n \times n$. Sa valeur initiale doit vérifier $\Phi_* \Upsilon_l \Phi_*^\top = \kappa(\mathbf{x}^*, \mathbf{x}^*)$ avec $\kappa(\mathbf{x}^*, \mathbf{x}^*)$ la covariance initiale choisie (voir la Section 3.1.5 ou bien [SKW+15]).

Nous écrivons, par souci de simplification, $\Phi_* = \Phi(\mathbf{x}^*)$, $\Phi = \Phi(\mathbf{x})$ et $\mathbf{b}$ un vecteur contenant toutes les mesures rangées selon leur indice. Les Equations 3.80 prédisent la valeur de $B(\mathbf{x}^*)$ et son incertitude associée sur le domaine Ω.

Pour la suite, $\Phi_l = \Phi(\mathbf{x}^1, ..., \mathbf{x}^U)$ avec U le nombre de points de données pour la l-ième itération.

Pour les nouvelles données, nous calculons leur variance associée :

$$K_l = \left[\Phi_l \Upsilon_{l-1} \Phi_l^\top + \tfrac{1}{2}(\Phi_l \Upsilon_{l-1} \Theta_l^\top + \Theta_l \Upsilon_{l-1} \Phi_l^\top) + D_{\mathrm{Var},l} + D_{\Sigma_b,U}\right]. \tag{3.81}$$

$D_{\mathrm{Var},l}$ correspond à la matrice diagonale par blocs définie de la même manière que D_{Var} dans la partie précédente, mais pour les données introduites à l'itération l. Après avoir obtenu Φ_l, Θ_l et K_l, les paramètres Υ_{l-1} et $\mathbf{\Gamma}_l$, associés à l'espérance et à la variance, peuvent être actualisés :

$$J_l = \Upsilon_{l-1}(\Phi_l + \frac{1}{2}\Theta_l)^\top K_l^{-1} \tag{3.82}$$

$$\mathbf{\Gamma}_l = \mathbf{\Gamma}_{l-1} + J_l\left(\mathbf{b}_l - (\Phi_l + \frac{1}{2}\Theta_l)\mathbf{\Gamma}_{l-1}\right) \tag{3.83}$$

$$\Upsilon_l = \Upsilon_{l-1} + J_l K_l J_l. \tag{3.84}$$

Processus Gaussien *itératif à rang réduit* avec incertitude sur les données d'entrée et de sortie

Processus Gaussien itératif à rang réduit : covariances initiales et bruit sur les sorties
Notre méthode : ajout du bruit sur les entrées
Equations similaires, mais ajout de matrices supplémentaires liées aux fonctions propres : Θ et D_{Var}.
Initialisation : $\Gamma_0 = \mathbf{0}_m$ et $\Upsilon_0 = (S_{SE}(\lambda_1), ..., S_{SE}(\lambda_m))$
avec, pour une valeur scalaire quelconque ω,

$$S_{SE}(\omega) = \sigma_{SE}^2 (2\pi l_{SE}^2)^{\frac{d}{2}} \exp\left(\frac{-\omega^2 l_{SE}^2}{2}\right). \tag{3.85}$$

$$K_l = \Phi_l \Upsilon_{l-1} \Phi_l^\top + \frac{1}{2}\left(\Phi_l \Upsilon_{l-1} \Theta_l^\top + \Theta_l \Upsilon_{l-1} \Phi_l^\top\right) + D_{\mathrm{Var},l} + D_{\Sigma_b,U} \tag{3.86}$$

$$J_l = \Upsilon_{l-1}(\Phi_l + \frac{1}{2}\Theta_l)^\top K_l^{-1} \tag{3.87}$$

$$\mathbf{\Gamma}_l = \mathbf{\Gamma}_{l-1} + J_l\left(\mathbf{b}_l - (\Phi_l + \frac{1}{2}\Theta_l)\mathbf{\Gamma}_{l-1}\right) \tag{3.88}$$

$$\Upsilon_l = \Upsilon_{l-1} + J_l K_l J_l. \tag{3.89}$$

Dans le cas mono-dimensionnel, $\frac{1}{2}\left(\Phi_l \Upsilon_{l-1} \Theta_l^\top + \Theta_l \Upsilon_{l-1} \Phi_l^\top\right) = \Phi_l \Upsilon_{l-1} \Theta_l^\top$.

Pour une application au champ magnétique en particulier, nous nous plaçons en dimension 3 pour les entrées et les sorties avec $B : \mathbb{R}^3 \to \mathbb{R}^3$. Ici, les trois composantes du champ correspondent au gradient d'un processus Gaussien mono-dimensionnel donc nous remplaçons Φ par $\nabla\Phi$. Trois premières lignes supplémentaires correspondant à κ_{lin} doivent être ajoutées, correspondant à $\nabla\mathbf{x}$ et $\Upsilon_0 = (\sigma_{lin}^2, \sigma_{lin}^2, \sigma_{lin}^2, S_{SE}(\lambda_1), ..., S_{SE}(\lambda_m))$.

3.4 Etude du comportement de différents modèles de GP avec entrées et sorties bruitées sur des données simulées

Pour pouvoir évaluer les performances de notre modèle, que nous avons appelé RRONIG (Reduced-Rank Online Noisy Input Gaussian Process), nous nous sommes appuyés dans un premier temps sur des données synthétiques afin de pouvoir réaliser un grand nombre de tirages et de disposer d'une meilleure compréhension du comportement des différents algorithmes testés. Les résultats expérimentaux, en lien avec la localisation, sont présentés dans la Section 4.4.2 du chapitre suivant.

Les expériences réalisées découlent de celles déjà utilisées dans [BSvV16] pour comparer différents algorithmes de régression par processus Gaussiens, avec ou sans erreur de localisation. Les fonctions simulées sont obtenues à partir des résultats d'un processus Gaussien. Ce processus Gaussien utilise une covariance de la forme exponentielle quadratique :

$$\kappa_f(x^i, x^j) = \sigma_f^2 exp\left(-\frac{\left(x^i - x^j\right)^2}{2l_f^2}\right) \tag{3.90}$$

avec $\sigma_f = 1$ et $l_f = 1$. Les valeurs des mesures pour ce processus Gaussien sont obtenues de manière aléatoire et suivent une loi normale. La fonction ainsi générée possède des valeurs dans l'intervalle $[-5, 5]$.

Puis, pour chaque fonction simulée, un certain nombre de mesures bruitées sont générées, avec un écart-type spécifique pour les entrées et pour les sorties : 200 points de mesure, tirés de façon aléatoire, sont utilisé. Ce sont ces points de mesure qui sont utilisés par chacun des modèles pour approximer la fonction vérité-terrain. Nous étudions ensuite les différences entre les prédictions de chaque algorithme et la vérité-terrain au niveau de 100 points, répartis équitablement sur tout le domaine étudié. Ces points sont distincts des points utilisés pour l'apprentissage.

SONIG (Sparse Online Noisy Input Gaussian Process), [BSvV16]) et RRONIG, notre méthode, sont deux algorithmes conçus pour être exécutés en ligne. Parce que leur fonctionnement consiste à mettre à jour le modèle à chaque nouvelle mesure, ils n'ont pas d'extension permettant de calculer les hyperparamètres du modèle à partir de toutes les mesures à la fois. Pour pouvoir comparer les différents algorithmes avec les mêmes conditions d'exécution, nous utilisons les paramètres calculés par NIGP (Noisy Input GP, [MR11]), une autre méthode incluant l'erreur sur les données d'entrée et qui estime la valeur des hyperparamètres, σ_f, l_f, σ_x (l'écat-type du bruit sur les entrées) et σ_y (l'écat-type du bruit sur les sorties). Cependant, pour les expériences avec un écart-type variable sur les données d'entrée, nous donnons directement au modèle la valeur véritable de σ_x car il n'est pas possible pour le modèle d'apprendre les hyperparamètres pour une seule mesure, sachant que les autres mesures ont une valeur de σ_x différente. Les valeurs de l'écart-type sur les entrées sont supposées connues. Les autres paramètres sont cependant ceux obtenus par NIGP, mais le σ_x obtenu, lui, est considéré comme inutile car il donne une information globale sur l'écart-type des entrées et non pour chacune des entrées individuellement. De façon indicative,

nous incluons aussi les résultats pour un processus Gaussien classique, sans erreur sur les données d'entrée, mais qui calcule ses propres hyperparamètres, σ_x ne faisant pas partie de son modèle.

Pour chaque jeu de données, l'algorithme SONIG n'utilise pas les mesures pour lesquels il prédit une variance négative. Lorsque ce problème apparaît pour 10 mesures, le modèle prédit par SONIG est considéré comme non valide. Lorsque ce type de problème est survenu, nous indiquons dans les notes du Tableau 3.1 combien de séquences ont été concernées. Nos résultats, obtenus avec le même code que celui de la publication de SONIG, sont légèrement différents parce que nous n'avons pas généré les mêmes fonctions aléatoires, même si elles avaient les mêmes paramètres en entrée.

SONIG diverge parfois : les prédictions obtenues sont alors très différentes de la vérité-terrain. Avec $\sigma_x = 0.4$ et $\sigma_y = 0.1$, SONIG est considéré comme invalide dans 18 expériences sur 500 par le modèle et 17 d'entre elles obtiennent des résultats qui sont très différents des valeurs de la fonction réelle.Le calcul des dérivées de SONIG est perturbé par le bruit sur les mesures alors qu'aucune erreur sur les dérivées n'est présente dans le modèle, causant quelquefois l'apparition de très forts gradients au niveau des fonctions prédites. C'est sans doute la raison principale du phénomène de divergence observé. Mais avec RRONIG nous n'avons rencontré cette difficulté pour aucune des séquences.

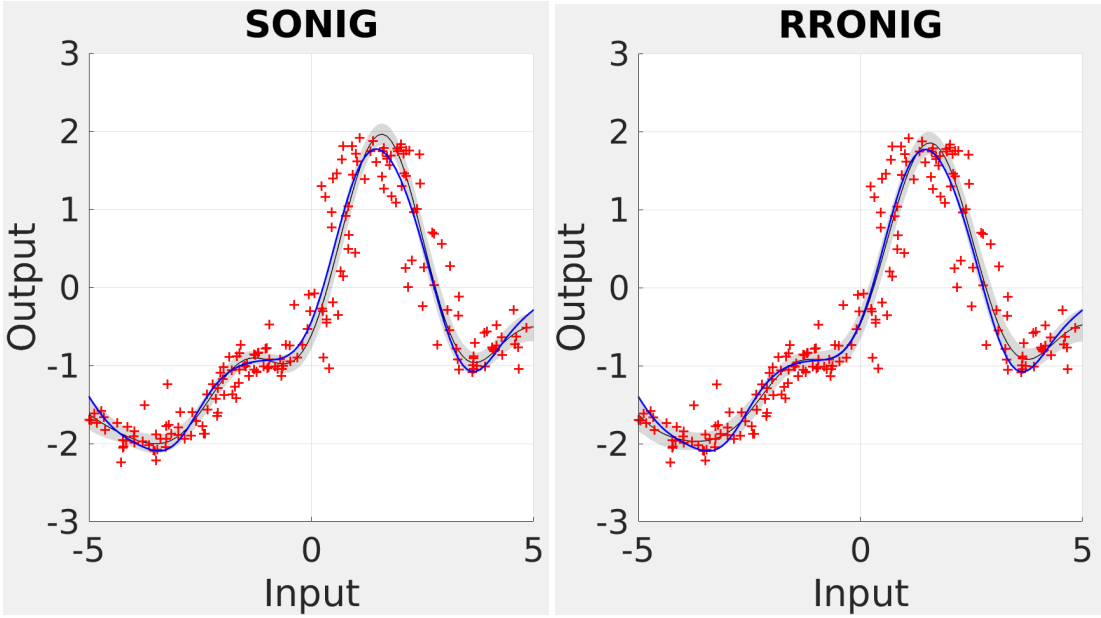


FIGURE 3.6: Exemple illustrant les résultats obtenus par SONIG ([BSvV16]), en haut, et notre méthode (processus Gaussien à rang réduit avec données d'entrée bruitées), en bas. $\sigma_x = 0.4$ et $\sigma_y = 0.1$. Les courbes de prédiction sont en noir et les enveloppes grises correspondent à l'intervalle à 95%. En bleu, la fonction à approximer. Les cercles rouges correspondent aux mesures. Les cercles noirs indiquent les nœuds pour SONIG.

Les résultats des comparaisons sont présentés dans le Tableau 3.1. Quand le bruit sur les mesures de position est faible, nous pouvons remarquer que le processus Gaussien classique a des performances légérement meilleures que les algorithmes intégrant l'erreur sur les entrées. Les résultats coïncident avec la théorie : SONIG et RRONIG s'appuient sur des méthodes utilisant des approximations plutôt que le calcul du processus Gaussien complet et sont donc moins précis, dans le cas où le bruit sur les entrées est presque négligeable. Cependant, quand l'erreur sur les entrées

Ecart-type sur les entrées	Ecart-type sur les sorties		Erreur moyenne ($\times 10^{-3}$)	Variance moyenne ($\times 10^{-3}$)	Ratio moyen
$\sigma_x = 0.1$ (réestimé par NIGP)	$\sigma_y = 0.4$	NIGP	12.122	10.417	1.183
		SONIG[1]	12.337	10.993	**1.135**
		RRONIG	12.152	10.684	1.158
		GP cl.*	**12.094**	10.514	1.160
$\sigma_x = 0.4$ (réestimé par NIGP)	$\sigma_y = 0.1$	NIGP	26.723	5.462	5.186
		SONIG[2]	3654.2	71.2	2.956
		SONIG[3]	23.098	9.360	2.306
		RRONIG	**21.418**	10.601	**2.052**
		GP cl.*	29.292	6.966	4.581
$\sigma_x = 0.4(1-(n-1)/N)$	$\sigma_y = 0.1$	NIGP	6.597	3.095	2.151
		SONIG	**2.969**	2.529	**1.148**
		RRONIG	3.031	2.548	1.166
		GP cl.*	7.299	3.766	1.966
$\sigma_x = 0.4n/N$	$\sigma_y = 0.1$	NIGP	6.819	3.101	2.165
		SONIG	3.359	2.687	1.240
		RRONIG	**2.984**	2.543	**1.150**
		GP cl.*	7.512	3.753	2.017

* : après estimation des hyperparamètres par le GP classique (σ_x n'est pas utilisé dans ce modèle, les incertitudes sont entièrement incluses dans l'estimation de σ_y).

[1] : les fonctions pour lesquelles SONIG est invalide ont été supprimées (il y en a 3 sur 500).

[2] : les fonctions pour lesquelles SONIG diverge ont été supprimées. Il y en a 18.

[3] : les 17 fonctions pour lesquelles SONIG est invalide selon l'algorithme ont été supprimées. Le critère pour les sélectionner est une erreur moyenne supérieure à 0.3. Les 18 fonctions SONIG invalides ne sont pas incluses non plus dans les calculs des moyennes.

TABLE 3.1: Précision moyenne de notre algorithme et de SONIG ([BSvV16]) pour 500 fonctions différentes générées. n est l'index de la donnée étudiée et N le nombre total de mesures.

est plus importante, nous observons une très nette amélioration des prédictions avec SONIG et RRONIG, en particulier quand la covariance n'est pas la même pour tous les points. Avec SONIG, pour les trois dernières séquences du Tableau, l'erreur a diminué de 45.3% par rapport à un GP classique (en moyennant uniquement sur les résultats suffisamment proches de la vérité-terrain, ce qui n'est que partiellement représentatif) ; avec RRONIG elle a baissé 48.6%.

Parfois, SONIG présente des erreurs d'approximation assez fortes sur certaines fonctions, quand le bruit sur les entrées est fort : pour $\sigma_x = 0.4$ et $\sigma_y = 0.1$, sur les 500 fonctions que nous avons générées, SONIG ne trouve parfois pas la solution et présente de fortes covariances (voir la Figure 3.7 pour un exemple de ce problème). Nous n'avons pas rencontré cette difficulté avec notre méthode, RRONIG, qui est plus robuste.

Les prédictions de RRONIG sont plus précises que celles de SONIG et ses variances aussi sont plus proches des valeurs désirées. La variance représente l'amplitude des erreurs du modèle qui devrait être obtenue : elle devrait être égale à l'erreur moyenne. Si le ratio de l'erreur sur la variance est plus grand que 1, cela signifie que le modèle a trop confiance dans ses prédictions : de nombreuses portions de la fonction étudiée seront en dehors de l'intervalle de confiance à

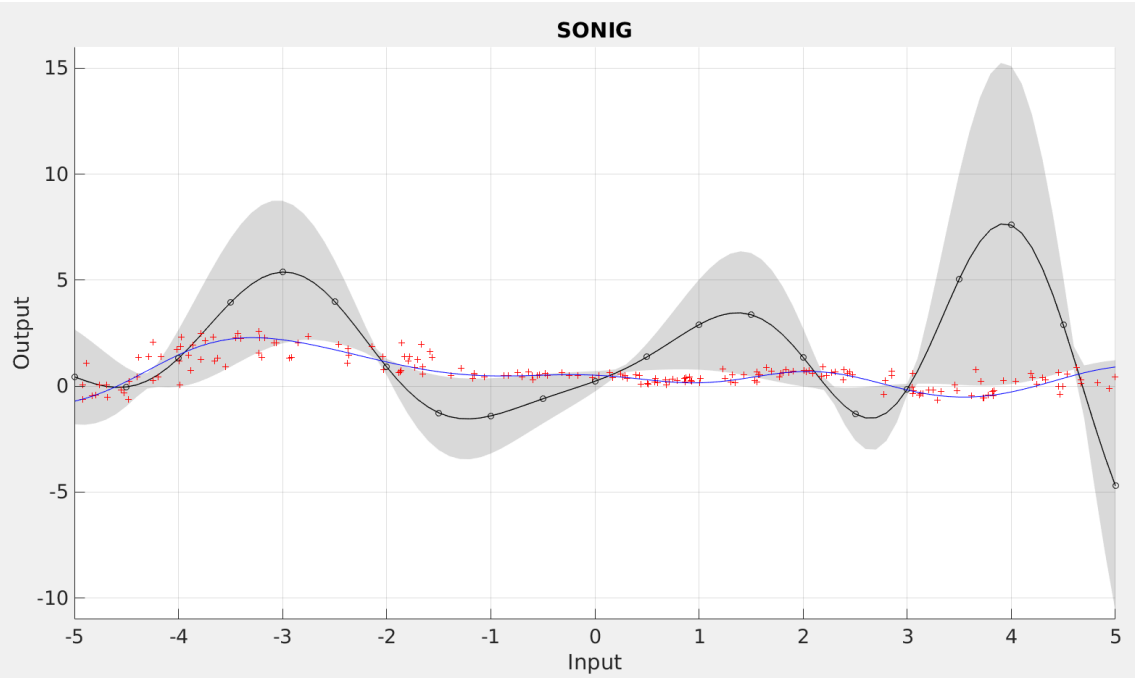


FIGURE 3.7: Exemple de résultats obtenus par SONIG qui n'interpolent pas correctement la fonction vérité-terrain ([BSvV16]). $\sigma_x = 0.4$ et $\sigma_y = 0.1$. Les courbes de prédiction sont en noir et les enveloppes grises correspondent à l'intervalle à 95%. En bleu, la fonction à approximer et les cercles correspondent aux mesures.

95%. Notre méthode, RRONIG, a un plus petit ratio que les autres méthodes, ne dépassant pas 2.1 sur tous les cas étudiés dans le Tableau 3.1, alors que le ratio pour SONIG est égal à 2.3 quand $\sigma_x = 0.4$ et $\sigma_y = 0.1$, en ne sélectionnant que les séquences pour lesquelles SONIG a des prédictions de bonne qualité, proches de la vérité-terrain. Ces valeurs qui restent élevées pour tous les modèles s'expliquent probablement par leur confiance dans les dérivées calculées. Au début, la fonction n'est connue qu'avec peu de points et les dérivées calculées sont très approximatives. Notre modèle ne prend pas cela en considération. Cependant, le ratio de l'erreur sur la covariance de RRONIG reste tout de même plus proche des valeurs désirées que pour les autres modèles, dont les ratios moyens sont plus élevés. La Figure 3.6 montre les prédictions de SONIG et de RRONIG. L'intervalle à 95% est plus large avec RRONIG que pour SONIG : la courbe de la vérité-terrain est donc moins souvent en dehors de l'intervalle de confiance. Dans le cas de SONIG, dans l'intervalle $[1.4, 2.2]$ environ, la vérité-terrain est en dehors de l'intervalle de confiance, alors que pour RRONIG, la vérité-terrain se trouve au bord de l'enveloppe fixée par les covariances.

Nous pouvons conclure que les cartes magnétiques obtenues à partir de notre modèle RRONIG seront plus précises car les résultats de RRONIG sont généralement meilleurs que ceux obtenus avec un processus Gaussien classique et avec les autres méthodes incluant un bruit sur les données d'entrées, sauf si ce bruit est très faible. Comme nous n'avons pas accès à la vérité-terrain de la carte magnétique du bâtiment, la précision de la carte sera évaluée dans le chapitre suivant en mesurant la qualité de la localisation obtenue à l'aide de cette carte.

3.5 Carte obtenue à partir de mesures avec positions et orientations bruitées

3.5.1 Insertion de l'erreur de rotation

Nous savons que les erreurs de la carte sont liées aux erreurs de localisation. Jusqu'à présent, nous avons pris en compte l'erreur de position mais le champ que nous étudions est un champ vectoriel : il est donc orienté. Or des erreurs de localisation sont présentes en position mais aussi en orientation : la localisation calcule à la fois une estimation de la position et la rotation. La prise en compte des erreurs de rotation permet ainsi de compléter le modèle. Les résultats expérimentaux liés à ces formules sont détaillés dans la Section 4.4.2 car les algorithmes de localisation estiment de façon imparfaite la position mais aussi l'orientation du système. Ici nous nous plaçons dans le cas particulier du champ magnétique, qui est considéré comme le gradient d'un potentiel scalaire : il est modélisé par un unique processus Gaussien.

Nous faisons apparaître les erreurs d'estimation de la rotation, en plus des erreurs de position. R_{MG} est la matrice de passage de M (pose actuelle du magnétomètre) à partir de G, le repère global. $\hat{R}_{MG}$ est son estimation (à partir de mesures). L'erreur entre l'estimation et la valeur réelle, $\hat{R}_{MM}^{\top} R_{MG}$ peut être écrite sous la forme d'un vecteur de taille 3 dans le formalisme de l'algèbre de Lie, de moyenne $\mathbf{0}_3$ et sa covariance est calculée par le MSCKF :

$$\Psi = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}. \tag{3.91}$$

$$B_{LER}(\mathbf{x}) = \hat{R}_{GM}(R_{MG}B(\hat{\mathbf{x}}) + \varepsilon) \tag{3.92}$$
$$= \hat{R}_{GM}R_{MG}B(\hat{\mathbf{x}}) + \hat{R}_{GM}\varepsilon \tag{3.93}$$

B_{LER} correspond à la loi suivie par les mesures, sachant que la position et l'orientation sont bruitées (auxquelles il faut ajouter le bruit du capteur, qui est un bruit présent sur les sorties). Soit $\delta R = \hat{R}_{GM}R_{MG}$ la matrice de passage de la rotation estimée à la rotation réelle à effectuer.

3.5.1.1 Synthèse des résultats

Nous présentons ici le résultat final pour le conditionnement du champ magnétique par les valeurs de B_{LER}, bruitées en position et en orientation. Les différentes étapes de la démonstration permettant d'obtenir les expressions de K_2 et de K_3 sont démontrées dans les Sections suivantes.

Processus Gaussien avec incertitude sur les positions et les orientations

Processus Gaussien conditionné à une mesure :
Avec l'erreur de position (Section 3.3.3).
Avec l'erreur d'orientation (avec Ψ le vecteur d'erreur dans le formalisme de l'algèbre de Lie).

Variance du modèle :

$$K_1 = \begin{bmatrix} \kappa(\mathbf{x},\mathbf{x}) & \kappa(\mathbf{x},\mathbf{x}') \\ \kappa(\mathbf{x},\mathbf{x}')^\top & \kappa(\mathbf{x}',\mathbf{x}') \end{bmatrix}$$

Covariance modèle / mesures :

$$K_2 \simeq \begin{bmatrix} K_{2,LE}(\mathbf{x},\mathbf{x}^\alpha) + \varkappa\Big(\mathrm{Cov}(\Psi_\times,\xi),\mathrm{Cov}(B(\mathbf{x}),\nabla B(\mathbf{x}^\alpha))\Big) + \mathrm{Cov}(B(\mathbf{x}),B(\mathbf{x}^\alpha))\Omega(\delta R_2)^\top \\ K_{2,LE}(\mathbf{x}',\mathbf{x}^\alpha) + \varkappa\Big(\mathrm{Cov}(\Psi_\times,\xi),\mathrm{Cov}(B(\mathbf{x}'),\nabla B(\mathbf{x}^\alpha))\Big) + \mathrm{Cov}(B(\mathbf{x}'),B(\mathbf{x}^\alpha))\Omega(\delta R_2)^\top \end{bmatrix}$$

Variance des mesures :

$$\begin{aligned} K_3 \simeq K_{3,LE}(\mathbf{x}^\alpha,\mathbf{x}^\alpha) &+ \mathrm{Cov}(B(\mathbf{x}^\alpha),B(\mathbf{x}^\alpha))\Omega(\delta R_2^\alpha)^\top + \Omega(\delta R_2^\alpha)\mathrm{Cov}(B(\mathbf{x}^\alpha),B(\mathbf{x}^\alpha)) \\ &+ 2\varkappa\Big(\mathrm{Cov}(\Psi_\times^\alpha,\xi_\alpha),\mathrm{Cov}(B(\mathbf{x}^\alpha),\nabla B(\mathbf{x}^\alpha))\Big) + \mathrm{Var}\left(\Psi_\times^\alpha B(\mathbf{x}^\alpha)\right) \end{aligned}$$

avec :

$$\Psi = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} \tag{3.94}$$

et :

$$\delta R_2 = \begin{bmatrix} 1-\frac{\theta^2}{2}-\frac{\psi^2}{2} & -\psi+\frac{\phi\theta}{2} & \theta+\frac{\phi\psi}{2} \\ \psi+\frac{\phi\theta}{2} & 1-\frac{\phi^2}{2}-\frac{\psi^2}{2} & -\phi+\frac{\theta\psi}{2} \\ -\theta+\frac{\phi\psi}{2} & \phi+\frac{\theta\psi}{2} & 1-\frac{\phi^2}{2}-\frac{\theta^2}{2} \end{bmatrix}. \tag{3.95}$$

Pour simplifier les notations, $\Psi_\times = \lfloor\Psi\rfloor_\times$.

L'opérateur Ω calcule l'espérance, élément par élément, d'une matrice. L'image par cet opérateur de δR_2 est :

$$\Omega(\delta R_2) = \frac{1}{2}\begin{bmatrix} -(\mathrm{Var}\,\theta+\mathrm{Var}\,\psi) & \mathrm{Cov}(\phi,\theta) & \mathrm{Cov}(\phi,\psi) \\ \mathrm{Cov}(\phi,\theta) & -(\mathrm{Var}\,\phi+\mathrm{Var}\,\psi) & \mathrm{Cov}(\theta,\psi) \\ \mathrm{Cov}(\phi,\psi) & \mathrm{Cov}(\theta,\psi) & -(\mathrm{Var}\,\phi+\mathrm{Var}\,\theta) \end{bmatrix} \tag{3.96}$$

et nous définissons :

$$\varkappa\Big(\mathrm{Cov}(\Psi_\times,\xi),\mathrm{Cov}(B(\mathbf{x}'),\nabla B(\mathbf{x}))\Big) = \mathrm{Cov}(B(\mathbf{x}'),\Psi_\times\nabla B(\mathbf{x})\xi) \tag{3.97}$$

$$= \sum_{k=1}^{3}\sum_{l=1}^{3}\mathrm{Cov}((\Psi_\times)_{:,k},\xi_l)\frac{\partial}{\partial \mathbf{x}_l}\mathrm{Cov}(B(\mathbf{x}'),B_k(\mathbf{x})) \tag{3.98}$$

3.5.1.2 Développement limité de la matrice de rotation

Nous réutilisons l'expression de B_{LE} (voir l'Equation 3.32) pour obtenir B_{LER} :

$$B_{LER}(\mathbf{x}) = \delta R\Big(B(\mathbf{x})+\nabla(B(\mathbf{x}))\xi+\frac{1}{2}h(B(\mathbf{x}),\xi)+O(\|\ \xi\ \|^3)\Big)+\hat{R}_{GM}\varepsilon. \tag{3.99}$$

L'espérance de $B_{LER}(\mathbf{x})$ se développe ainsi, sachant que $\hat{R}_{GM}$ et ε sont indépendantes :

$$\begin{aligned}\mathbb{E}\left[B_{LER}(\mathbf{x})\right] &= \mathbb{E}\left[\delta R(B(\mathbf{x})+\nabla(B(\mathbf{x}))\xi+\frac{1}{2}h(B(\mathbf{x}),\xi)+O(\|\ \xi\ \|^3))+\hat{R}_{GM}\varepsilon\right]\\ &= \mathbb{E}\left[\delta R\times(B(\mathbf{x})+\nabla(B(\mathbf{x}))\xi+\frac{1}{2}h(B(\mathbf{x}),\xi)+O(\|\ \xi\ \|^3))\right]+0_{3\times1}\end{aligned} \tag{3.100}$$

Maintenant, nous développons la matrice de rotation d'erreur à l'aide de la formule de Taylor, selon un raisonnement similaire à celui utilisé pour le calcul de l'erreur de position. Nous écrivons la matrice de rotation dans le groupe $SO(3)$ car notre algorithme nous donne leur erreur. Nous utilisons donc l'algèbre de Lie. Nous développons jusqu'à l'ordre 2 pour chaque angle dans l'expression de la matrice de rotation :

$$\delta R \simeq I_3+\Psi_\times+\frac{(\Psi_\times)^2}{2} \tag{3.101}$$

Comme nous nous limitons à l'ordre 2 dans la suite des calculs, nous définissons :

$$\delta R_2 = I_3+\Psi_\times+\frac{(\Psi_\times)^2}{2}. \tag{3.102}$$

Nous développons l'expression de δR_2 élément par élément :

$$\delta R_2 = \begin{bmatrix} 1-\frac{\theta^2}{2}-\frac{\psi^2}{2} & -\psi+\frac{\phi\theta}{2} & \theta+\frac{\phi\psi}{2}\\ \psi+\frac{\phi\theta}{2} & 1-\frac{\phi^2}{2}-\frac{\psi^2}{2} & -\phi+\frac{\theta\psi}{2}\\ -\theta+\frac{\phi\psi}{2} & \phi+\frac{\theta\psi}{2} & 1-\frac{\phi^2}{2}-\frac{\theta^2}{2}\end{bmatrix} \tag{3.103}$$

3.5.1.3 Calcul de l'espérance

Si nous supposons que δR est une matrice de rotation pour les trois angles θ, ψ et ϕ, le calcul de $B_{LER}(\mathbf{x})$ devient :

$$B_{LER}(\mathbf{x}) = \delta R\times\Big(B(\mathbf{x})+\nabla(B(\mathbf{x}))\xi+\frac{1}{2}h(B(\mathbf{x}),\xi)+O(\|\ \xi\ \|^3)\Big)+\hat{R}_{GM}\varepsilon \tag{3.104}$$

De nouveau, nous conservons uniquement les termes d'ordre 2 :

$$\begin{aligned} B_{LER}(\mathbf{x}) = & \begin{bmatrix} (1-\frac{\theta^2}{2}-\frac{\psi^2}{2})B_1(\mathbf{x})+(-\psi+\phi\theta)B_2(\mathbf{x})+(\theta+\phi\psi)B_3(\mathbf{x}) \\ \psi B_1(\mathbf{x})+(1-\frac{\phi^2}{2}-\frac{\psi^2}{2})B_2(\mathbf{x})+(-\phi+\theta\psi)B_3(\mathbf{x}) \\ -\theta B_1(\mathbf{x})+\phi B_2(\mathbf{x})+(1-\frac{\phi^2}{2}-\frac{\theta^2}{2})B_3(\mathbf{x}) \end{bmatrix} \\ & + (I_3+\Psi_\times)\nabla B(\mathbf{x})\xi + \frac{1}{2}h(B(\mathbf{x}),\xi) + \hat{R}_{GM}\varepsilon + O(\| \xi \|^3) \end{aligned} \tag{3.105}$$

$\mathbb{E}[I_3\nabla B(\mathbf{x})\xi] = \mathbf{0}_3$ car $B(\mathbf{x})$ et ξ sont indépendantes. Le troisième terme est égal à, pour chaque coordonnée i :

$$(\Psi_\times\nabla B(\mathbf{x})\xi)_i = \sum_{k=1}^{3}\sum_{l=1}^{3}(\Psi_\times)_{i,k}\nabla B(\mathbf{x})_{k,l}\xi_l. \tag{3.106}$$

L'expression complète de l'espérance de B_{LER} est donc :

$$\begin{aligned} \mathbb{E}[B_{LER}(\mathbf{x})] \simeq & \mathbb{E}[B(\mathbf{x})] + \frac{1}{2}\Big\langle \Sigma, H(B)\Big\rangle \\ & + \frac{1}{2}\mathbb{E}\left[\begin{bmatrix} -(\mathrm{Var}\,\theta+\mathrm{Var}\,\psi)B_1(\mathbf{x})+\mathrm{Cov}(\phi,\theta)B_2(\mathbf{x})+\mathrm{Cov}(\phi,\psi)B_3(\mathbf{x}) \\ \mathrm{Cov}(\phi,\theta)B_1(\mathbf{x})-(\mathrm{Var}\,\phi+\mathrm{Var}\,\psi)B_2(\mathbf{x})+\mathrm{Cov}(\theta,\psi)B_3(\mathbf{x}) \\ \mathrm{Cov}(\phi,\psi)B_1(\mathbf{x})+\mathrm{Cov}(\theta,\psi)B_2(\mathbf{x})-(\mathrm{Var}\,\phi+\mathrm{Var}\,\theta)B_3(\mathbf{x}) \end{bmatrix}\right] \\ & + \mathbb{E}\left[\begin{bmatrix} \sum_{k=1}^3\sum_{l=1}^3 \mathrm{Cov}((\Psi_\times)_{1,k}\xi_l)\nabla B(\mathbf{x})_{k,l} \\ \sum_{k=1}^3\sum_{l=1}^3 \mathrm{Cov}((\Psi_\times)_{2,k},\xi_l)\nabla B(\mathbf{x})_{k,l} \\ \sum_{k=1}^3\sum_{l=1}^3 \mathrm{Cov}((\Psi_\times)_{3,k},\xi_l)\nabla B(\mathbf{x})_{k,l} \end{bmatrix}\right] \end{aligned} \tag{3.107}$$

En posant Ω l'espérance élément par élément jusqu'à l'ordre 2 d'une matrice, nous pouvons réécrire l'expression sous la forme d'une somme de produits matriciels :

$$\begin{aligned} \mathbb{E}[B_{LER}(\mathbf{x})] \simeq & \mathbb{E}[B(\mathbf{x})] + \frac{1}{2}\Big\langle \Sigma, H(B)\Big\rangle \\ & + \Omega(\delta R_2)\,\mathbb{E}[B(\mathbf{x})] + \sum_{k=1}^{3}\sum_{l=1}^{3}\mathrm{Cov}((\Psi_\times)_{:,k}\xi_l)\,\mathbb{E}[\nabla B(\mathbf{x})_{k,l}] \end{aligned} \tag{3.108}$$

3.5.1.4 Calcul des covariances : expression de K_2

Désormais, nous devons calculer l'expression des matrices de covariance K_2 et K_3 pour utiliser de nouveau les équations de conditionnement des processus Gaussiens. Nous commençons par le calcul de K_2, dont les éléments sont de la forme $\mathrm{Cov}(B(\mathbf{x}'), B_{LER}(\mathbf{x}))$.

Pour la covariance avec $B(\mathbf{x}')$, $\mathbf{x}'$ étant un autre point pouvant (ou non) être confondu avec $\mathbf{x}$, sachant que $B(\mathbf{x})$ et $\hat{R}_{GM}\varepsilon$ sont indépendantes :

$$\begin{aligned} \mathrm{Cov}(B(\mathbf{x}'), B_{LER}(\mathbf{x})) &= \mathrm{Cov}(B(\mathbf{x}'), \delta R B(\hat{\mathbf{x}})) + \mathrm{Cov}(B(\mathbf{x}'), \hat{R}_{GM}\varepsilon) \\ &= \mathrm{Cov}(B(\mathbf{x}'), \delta R B(\hat{\mathbf{x}})) + \mathbf{0}_3 \\ &= \mathrm{Cov}(B(\mathbf{x}'), \delta R\Big(B(\mathbf{x}) + \nabla(B(\mathbf{x}))\xi + \frac{1}{2}h(B(\mathbf{x}),\xi) + O(\| \xi \|^3)\Big)) \end{aligned} \tag{3.109}$$

$$\begin{aligned}\mathrm{Cov}(B(\mathbf{x}'),B_{LER}(\mathbf{x})) \simeq \mathrm{Cov}(B(\mathbf{x}'),\delta R_2 B(\mathbf{x})) + \mathrm{Cov}(B(\mathbf{x}'),(I_3+\Psi_\times)\nabla B(\mathbf{x})\xi) \\ + \mathrm{Cov}(B(\mathbf{x}'),\frac{1}{2}h(B(\mathbf{x}),\xi))\end{aligned} \tag{3.110}$$

$$\begin{aligned}\mathrm{Cov}(B(\mathbf{x}'),B_{LER}(\mathbf{x})) \simeq \mathrm{Cov}(B(\mathbf{x}'),B_{LE}(\mathbf{x})) + \mathrm{Cov}(B(\mathbf{x}'),(\delta R_2 - I_3)B(\mathbf{x})) \\ + \mathrm{Cov}(B(\mathbf{x}'),\Psi_\times \nabla B(\mathbf{x})\xi)\end{aligned} \tag{3.111}$$

$$\begin{aligned}\mathrm{Cov}(B(\mathbf{x}'),(\delta R_2 - I_3)B(\mathbf{x})) &= \mathbb{E}\left[(B(\mathbf{x}') - \mathbb{E}\left[B(\mathbf{x}')\right])((\delta R_2 - I_3)B(\mathbf{x}) - \mathbb{E}\left[(\delta R_2 - I_3)B(\mathbf{x})\right])^\top\right] \\ &= \mathbb{E}\left[B(\mathbf{x}')B(\mathbf{x})^\top(\delta R_2 - I_3)^\top\right] - \mathbb{E}\left[B(\mathbf{x}')\right]\mathbb{E}\left[(\delta R_2 - I_3)B(\mathbf{x})\right])^\top\end{aligned} \tag{3.112}$$

En raison de l'indépendance de $B(\mathbf{x})$ et δR_2, nous pouvons simplifier certains termes de la covariance :

$$\begin{aligned}\mathrm{Cov}(B(\mathbf{x}'),(\delta R_2 - I_3)B(\mathbf{x})) &= \mathbb{E}\left[B(\mathbf{x}')B(\mathbf{x})^\top(\delta R_2 - I_3)^\top\right] - \mathbb{E}\left[B(\mathbf{x}')\right]\mathbb{E}\left[(\delta R_2 - I_3)B(\mathbf{x})\right])^\top \\ &= \left(\mathbb{E}\left[B(\mathbf{x}')B(\mathbf{x})^\top\right] - \mathbb{E}\left[B(\mathbf{x}')\right]\mathbb{E}\left[B(\mathbf{x})\right]^\top\right)\mathbb{E}\left[(\delta R_2 - I_3)^\top\right] \\ &= \mathrm{Cov}(B(\mathbf{x}'),B(\mathbf{x}))\,\mathbb{E}\left[(\delta R_2 - I_3)^\top\right] \\ &= \mathrm{Cov}(B(\mathbf{x}'),B(\mathbf{x}))\Omega(\delta R_2)^\top\end{aligned} \tag{3.113}$$

Le terme suivant que nous cherchons à calculer est $\mathrm{Cov}(B(\mathbf{x}'),\Psi_\times\nabla B(\mathbf{x})\xi)$. Par linéarité de la covariance, nous pouvons séparer les différents éléments de la somme :

$$\begin{aligned}\mathrm{Cov}(B(\mathbf{x}'),\Psi_\times\nabla B(\mathbf{x})\xi)_{i,j} &= \mathrm{Cov}(B_i(\mathbf{x}'),\sum_{k=1}^{3}\sum_{l=1}^{3}((\Psi_\times)_{j,k}\nabla B_{k,l}(\mathbf{x})\xi_l) \\ &= \mathrm{Cov}(B_i(\mathbf{x}'),\sum_{k=1}^{3}\sum_{l=1}^{3}(\Psi_\times)_{j,k}\nabla B_{k,l}(\mathbf{x})\xi_l) \\ &= \sum_{k=1}^{3}\sum_{l=1}^{3}\mathrm{Cov}(B_i(\mathbf{x}'),(\Psi_\times)_{j,k}\nabla B_{k,l}(\mathbf{x})\xi_l) \\ &= \sum_{k=1}^{3}\sum_{l=1}^{3}\mathbb{E}\left[(B_i(\mathbf{x}') - \mathbb{E}\left[B_i(\mathbf{x}')\right])(\nabla B_{k,l}(\mathbf{x})(\Psi_\times)_{j,k}\xi_l - \mathbb{E}\left[\nabla B_{k,l}(\mathbf{x})(\Psi_\times)_{j,k}\xi_l\right])\right]\end{aligned} \tag{3.114}$$

Finalement, nous obtenons :

$$\begin{aligned}&\mathrm{Cov}(B(\mathbf{x}'),\Psi_\times\nabla B(\mathbf{x})\xi)_{i,j} \\ &\quad= \sum_{k=1}^{3}\sum_{l=1}^{3}\mathbb{E}\left[(B_i(\mathbf{x}')(\nabla B_{k,l}(\mathbf{x})(\Psi_\times)_{j,k}\xi_l)\right] - \mathbb{E}\left[B_i(\mathbf{x}')\right]\mathbb{E}\left[\nabla B_{k,l}(\mathbf{x})(\Psi_\times)_{j,k}\xi_l\right]\end{aligned} \tag{3.115}$$

Par indépendance de $B(\mathbf{x})$ et $(\Psi_\times)_{j,k}\xi_l$, nous pouvons développer la somme :

$$\mathrm{Cov}(B(\mathbf{x}'),\Psi_\times\nabla B(\mathbf{x})\xi)_{i,j} = \sum_{k=1}^{3}\sum_{l=1}^{3}\mathbb{E}\left[(\Psi_\times)_{j,k}\xi_l\right]\Big(\mathbb{E}\left[B_i(\mathbf{x}')(\nabla B_{k,l}(\mathbf{x})\right] - \mathbb{E}\left[B_i(\mathbf{x}')\right]\mathbb{E}\left[\nabla(B(\mathbf{x}))_{k,l}\right]\Big) \tag{3.116}$$

$$\begin{aligned}\mathrm{Cov}(B(\mathbf{x}'),\Psi_\times\nabla B(\mathbf{x})\xi)_{i,j} &= \sum_{k=1}^{3}\sum_{l=1}^{3}\mathbb{E}\left[(\Psi_\times)_{j,k}\xi_l\right]\mathrm{Cov}((B_i(\mathbf{x}'),\nabla B_{k,l}(\mathbf{x}))\\ &= \sum_{k=1}^{3}\sum_{l=1}^{3}\mathrm{Cov}((\Psi_\times)_{j,k},\xi_l)\,\mathrm{Cov}(B_i(\mathbf{x}'),\nabla B_{k,l}(\mathbf{x})\\ &= \sum_{k=1}^{3}\sum_{l=1}^{3}\mathrm{Cov}((\Psi_\times)_{j,k},\xi_l)\frac{\partial}{\partial\mathbf{x}_l}\mathrm{Cov}(B_i(\mathbf{x}'),B_k(\mathbf{x}))\end{aligned} \tag{3.117}$$

$\mathrm{Cov}((\Psi_\times)_{j,k},\xi_l) = \mathbb{E}\left[(\Psi_\times)_{j,k}\xi_l\right]$ car $\mathbb{E}[\Psi_\times] = 0_{3\times3}$ et $\mathbb{E}[\xi] = \mathbf{0}_3$ (et de même avec $\nabla B_{k,l}(\mathbf{x})$ et $(\Psi_\times)_{j,k}$).

Nous introduisons un nouvel opérateur, $\varkappa$:

$$\begin{aligned}\mathrm{Cov}(B(\mathbf{x}'),\Psi_\times\nabla B(\mathbf{x})\xi) &= \sum_{k=1}^{3}\sum_{l=1}^{3}\mathrm{Cov}((\Psi_\times)_{:,k},\xi_l)\frac{\partial}{\partial\mathbf{x}_l}\mathrm{Cov}(B(\mathbf{x}'),B_k(\mathbf{x}))\\ &= \varkappa\Big(\mathrm{Cov}(\Psi_\times,\xi),\mathrm{Cov}(B(\mathbf{x}'),\nabla B(\mathbf{x}))\Big)\end{aligned} \tag{3.118}$$

La covariance $\mathrm{Cov}(B(\mathbf{x}'),B_{LER}(\mathbf{x}))$ s'écrit alors, avec une approximation à l'ordre 2 :

$$\begin{aligned}\mathrm{Cov}(B(\mathbf{x}'),B_{LER}(\mathbf{x})) \simeq\ &\mathrm{Cov}(B(\mathbf{x}'),B_{LE}(\mathbf{x})) + \varkappa\Big(\mathrm{Cov}(\Psi_\times,\xi),\mathrm{Cov}(B(\mathbf{x}'),\nabla B(\mathbf{x}))\Big)\\ &+\mathrm{Cov}(B(\mathbf{x}'),B(\mathbf{x}))\Omega(\delta R_2)^\top\end{aligned} \tag{3.119}$$

Cette expression est celle utilisée pour compléter les différents blocs composant K_2.

3.5.1.5 Calcul des covariances : expression de K_3

Étudions désormais la covariance à deux instants t^α et t^ν, pour connaître l'expression de K_3 :

$$\mathrm{Cov}\Big(B_{LER}(\alpha),B_{LER}(\nu)\Big) \simeq \mathrm{Cov}\Big(\delta R^\alpha B_{LE}(\mathbf{x}^\alpha),\delta R^\nu B_{LE}(\mathbf{x}^\nu)\Big) \tag{3.120}$$

$$\begin{aligned}\mathrm{Cov}\Big(B_{LER}(\alpha),B_{LER}(\nu)\Big) \simeq\ &\mathrm{Cov}(B_{LE}(\alpha),B_{LE}(\nu)) + \mathrm{Cov}(B_{LE}(\mathbf{x}^\alpha),(\delta R_2^\nu - I_3)B_{LE}(\mathbf{x}^\nu))\\ &+\mathrm{Cov}((\delta R_2^\alpha - I_3)B_{LE}(\mathbf{x}^\alpha),B_{LE}(\mathbf{x}^\nu))\end{aligned} \tag{3.121}$$

$$\begin{aligned}\mathrm{Cov}\Big(B_{LER}(\alpha),B_{LER}(\nu)\Big) \simeq \mathrm{Cov}\Big(&B_{LE}(\mathbf{x}^\alpha) + (\delta R_2^\alpha - I_3)B(\mathbf{x}^\alpha) + \Psi_\times^\alpha\nabla B(\mathbf{x}^\alpha)\xi_\alpha,\\ &B_{LE}(\mathbf{x}^\nu) + (\delta R_2^\nu - I_3)B(\mathbf{x}^\nu) + \Psi_\times^\nu\nabla B(\mathbf{x}^\nu)\xi_\nu\Big)\end{aligned} \tag{3.122}$$

Nous développons l'équation en plusieurs membres, que nous étudions séparèment, en se limitant à l'ordre 2.

$$\begin{aligned}
&\mathrm{Cov}\Big(B_{LE}(\mathbf{x}^\alpha),(\delta R_2^\nu - I_3)B(\mathbf{x}^\nu) + \Psi_\times^\nu \nabla B(\mathbf{x}^\nu)\xi_\nu\Big) \\
&\qquad = \mathrm{Cov}\Big(B(\mathbf{x}^\alpha) + \nabla B(\mathbf{x}^\alpha)\xi_\alpha + \frac{1}{2}h(\mathbf{x}^\alpha, B(\mathbf{x}^\alpha)), (\delta R_2^\nu - I_3)B(\mathbf{x}^\nu) + \Psi_\times^\nu \nabla B(\mathbf{x}^\nu)\xi_\nu\Big) \\
&\qquad \simeq \mathrm{Cov}\Big(B(\mathbf{x}^\alpha), (\delta R_2^\nu - I_3)B(\mathbf{x}^\nu)\Big) + \mathrm{Cov}\Big(B(\mathbf{x}^\alpha), \Psi_\times^\nu \nabla B(\mathbf{x}^\nu)\xi_\nu\Big) \\
&\qquad + \mathrm{Cov}\Big(\nabla B(\mathbf{x}^\alpha)\xi_\alpha, \Psi_\times^\nu B(\mathbf{x}^\nu)\Big)
\end{aligned} \tag{3.123}$$

$$\begin{aligned}
&\mathrm{Cov}\Big(B_{LE}(\mathbf{x}^\alpha),(\delta R_2^\nu - I_3)B(\mathbf{x}^\nu) + \Psi_\times^\nu \nabla B(\mathbf{x}^\nu)\xi_\nu\Big) \\
&\qquad \simeq \mathrm{Cov}(B(\mathbf{x}^\alpha), B(\mathbf{x}^\nu))\Omega(\delta R_2^\nu)^\top + \varkappa\Big(\mathrm{Cov}(\Psi_\times^\nu, \xi^\nu), \mathrm{Cov}(B(\mathbf{x}^\alpha), \nabla B(\mathbf{x}^\nu))\Big) \\
&\qquad + \mathrm{Cov}\Big(\nabla B(\mathbf{x}^\alpha)\xi_\alpha, \Psi_\times^\nu B(\mathbf{x}^\nu)\Big)
\end{aligned} \tag{3.124}$$

$$\mathrm{Cov}\Big(\nabla B(\mathbf{x}^\alpha)\xi_\alpha, \Psi_\times^\nu B(\mathbf{x}^\nu)\Big) = \mathbb{E}\left[\Big(\nabla B(\mathbf{x}^\alpha)\xi_\alpha - \mathbb{E}\left[\nabla B(\mathbf{x}^\alpha)\xi_\alpha\right]\Big)\Big(\Psi_\times^\nu B(\mathbf{x}^\nu) - \mathbb{E}\left[\Psi_\times^\nu B(\mathbf{x}^\nu)\right]\Big)^\top\right] \tag{3.125}$$

Par indépendance des variables $\nabla B(\mathbf{x}^\alpha)$ et ξ_α entre elles et $\Psi_\times^\nu$ et $B(\mathbf{x}^\nu)$, sachant aussi que $\mathbb{E}\left[\xi_\alpha\right] = \mathbf{0}_3$ et $\mathbb{E}\left[\Psi_\times^\nu\right] = \mathbf{0}_3$, nous pouvons nous restreindre au calcul suivant :

$$\begin{aligned}
\mathrm{Cov}\Big(\nabla B(\mathbf{x}^\alpha)\xi_\alpha, \Psi_\times^\nu B(\mathbf{x}^\nu)\Big)_{i,j} &= \mathbb{E}\left[\Big(\nabla B(\mathbf{x}^\alpha)\xi_\alpha\Big)\Big(\Psi_\times^\nu B(\mathbf{x}^\nu)\Big)\right]_{i,j} \\
&= \mathbb{E}\left[\Big(\sum_{k=1}^{3}\sum_{l=1}^{3}\nabla B_{i,k}(\mathbf{x}^\alpha)\xi_{\alpha,k}\Big)B(\mathbf{x}^\nu)_l \Psi_{\times,l,j}^{\nu\top}\right] \\
&= \sum_{k=1}^{3}\sum_{l=1}^{3}\mathbb{E}\left[\xi_{\alpha,k}\Psi_{\times,l,j}^{\nu\top}\right]\mathbb{E}\left[\nabla B_{i,k}(\mathbf{x}^\alpha)B(\mathbf{x}^\nu)_l\right] \\
&= \mathbf{0}_{3\times 3}
\end{aligned} \tag{3.126}$$

$$\begin{aligned}
&\mathrm{Cov}\Big(B_{LE}(\mathbf{x}^\alpha),(\delta R_2^\nu - I_3)B(\mathbf{x}^\nu) + \Psi_\times^\nu \nabla B(\mathbf{x}^\nu)\xi_\nu\Big) \\
&\qquad \simeq \mathrm{Cov}(B(\mathbf{x}^\alpha), B(\mathbf{x}^\nu))\Omega(\delta R_2^\nu)^\top + \varkappa\Big(\mathrm{Cov}(\Psi_\times^\nu, \xi^\nu), \mathrm{Cov}(B(\mathbf{x}^\alpha), \nabla B(\mathbf{x}^\nu))\Big)
\end{aligned}$$

Nous avons la même chose par symétrie pour l'autre membre. Pour le dernier morceau, nous nous restreignons à l'ordre 2, ce qui limite le développement à un terme :

$$\begin{aligned}
&\mathrm{Cov}\Big((\delta R_2^\alpha - I_3)B(\mathbf{x}^\alpha) + \Psi_\times^\alpha \nabla B(\mathbf{x}^\alpha)\xi_\alpha, (\delta R_2^\nu - I_3)B(\mathbf{x}^\nu) + \Psi_\times^\nu \nabla B(\mathbf{x}^\nu)\xi_\nu\Big) \\
&\qquad \simeq \mathrm{Cov}\Big(\Psi_\times^\alpha B(\mathbf{x}^\alpha), \Psi_\times^\nu B(\mathbf{x}^\nu))\Big)
\end{aligned} \tag{3.127}$$

$\Psi_\times^\alpha$ et $\Psi_\times^\nu$ sont supposées indépendantes sauf si $\alpha = \nu$.
Le développement complet de la covariance $\mathrm{Cov}\left(B_{LER}(\alpha), B_{LER}(\nu)\right)$ est donc :

$$\begin{aligned}\mathrm{Cov}\left(B_{LER}(\alpha), B_{LER}(\nu)\right) \simeq\ & \mathrm{Cov}\left(B_{LE}(\mathbf{x}^\alpha), B_{LE}(\mathbf{x}^\nu)\right) \\ & + \mathrm{Cov}(B(\mathbf{x}^\alpha), B(\mathbf{x}^\nu))\Omega(\delta R_2^\nu)^\top \\ & + \Omega(\delta R_2^\alpha)\,\mathrm{Cov}(B(\mathbf{x}^\alpha), B(\mathbf{x}^\nu)) \\ & + \varkappa\Big(\mathrm{Cov}(\Psi_\times^\alpha, \xi_\alpha), \mathrm{Cov}(B(\mathbf{x}^\nu), \nabla B(\mathbf{x}^\alpha))\Big) \\ & + \varkappa\Big(\mathrm{Cov}(\Psi_\times^\nu, \xi_\nu), \mathrm{Cov}(B(\mathbf{x}^\alpha), \nabla B(\mathbf{x}^\nu))\Big) \\ & + \delta_{\alpha,\nu}\,\mathrm{Var}\left(\Psi_\times^\alpha B(\mathbf{x}^\alpha)\right)\end{aligned} \tag{3.128}$$

Les autres paramètres sont considérés d'ordre 3 donc négligeables.
Calculons explicitement $\mathrm{Var}\left(\Psi_\times^\alpha B(\mathbf{x}^\alpha)\right)$, sachant que $\mathbb{E}\left[\Psi_\times^\alpha B(\mathbf{x}^\alpha)\right] = \mathbf{0}_3$:

$$\begin{aligned}\mathrm{Var}\left(\Psi_\times^\alpha B(\mathbf{x}^\alpha)\right) &= \mathbb{E}\left[\Psi_\times^\alpha B(\mathbf{x}^\alpha)(\Psi_\times^\alpha B(\mathbf{x}^\alpha))^\top\right] \\ &= \mathbb{E}\left[(-B(\mathbf{x}^\alpha)_\times \Psi^\alpha)(-B(\mathbf{x}^\alpha)_\times \Psi^\alpha)^\top\right]\end{aligned} \tag{3.129}$$

En calculant terme par terme, nous obtenons :

$$\begin{aligned}(\mathrm{Var}\left(\Psi_\times^\alpha B(\mathbf{x}^\alpha)\right))_{i,j} &= \sum_{k=1}^{3}\sum_{l=1}^{3} \mathbb{E}\left[B(\mathbf{x}^\alpha)_{\times,i,k}\Psi_k^\alpha \Psi_l^\alpha B(\mathbf{x}^\alpha)_{\times,j,l}\right] \\ &= \sum_{k=1}^{3}\sum_{l=1}^{3} \mathrm{Cov}(\Psi_k^\alpha, \Psi_l^\alpha)\,\mathbb{E}\left[B(\mathbf{x}^\alpha)_{\times,i,k} B(\mathbf{x}^\alpha)_{\times,j,l}\right]\end{aligned} \tag{3.130}$$

L'expression finale est :

$$(\mathrm{Var}\left(\Psi_\times^\alpha B(\mathbf{x}^\alpha)\right)) = \sum_{k=1}^{3}\sum_{l=1}^{3} \mathrm{Cov}(\Psi_k^\alpha, \Psi_l^\alpha)\,\mathbb{E}\left[B(\mathbf{x}^\alpha)_{\times,:,k} B(\mathbf{x}^\alpha)_{\times,:,l}\right] \tag{3.131}$$

Avec les Equations 3.36 et 3.49, tous les éléments composant l'équation peuvent être calculés. La valeur réelle de R_{GM} est inconnue : on l'approxime par la valeur obtenue lors de la réalisation de la variable aléatoire $\hat{R}_{GM}$.

3.5.2 Processus Gaussien à rang réduit avec positions et orientations bruitées

Nous cherchons à obtenir l'espérance et la covariance en tout point, en utilisant une formulation à rang réduit des covariances, pour les raisons de temps d'exécution déjà mentionnées dans la partie précédente (le raisonnement est le même que pour l'erreur de position, dans la Section 3.3).

Processus Gaussien *à rang réduit* avec incertitude sur les positions et les orientations

Processus Gaussien à rang réduit itératif, mesure par mesure. Ici, la mesure correspond au point $\mathbf{x}^\alpha$.
Ajout du bruit sur les entrées (voir Section 3.3.5)
Avec l'erreur d'orientation (avec Ψ le vecteur d'erreur dans le formalisme de l'algèbre de Lie).

$$K_l = \Phi_l \Upsilon_{l-1} \Phi_l^\top + D_{\Sigma_b,3} + \Phi_l \Upsilon_{l-1} \Theta_l^\top + D_{\mathrm{Var},l} + \kappa(\mathbf{x}^\alpha, \mathbf{x}^\alpha)\Omega(\delta R_2^\alpha)^\top + \Omega(\delta R_2^\alpha)\kappa(\mathbf{x}^\alpha, \mathbf{x}^\alpha)$$
$$+ 2\varkappa\Big(\mathrm{Cov}(\Psi_\times^\alpha, \xi_\alpha), \mathrm{Cov}(B(\mathbf{x}^\alpha), \nabla B(\mathbf{x}^\alpha))\Big) + (\mathrm{Var}\left(\Psi_\times^\alpha B(\mathbf{x}^\alpha)\right)) \tag{3.132}$$

$$J_l = \Upsilon_{l-1}(\Phi_l + \frac{1}{2}\Theta_l + \Omega(\delta R_2^\alpha)\Phi_l + \varkappa\Big(\mathrm{Cov}(\Psi_\times^\alpha, \xi_\alpha), \mathrm{Cov}(B(\mathbf{x}^\alpha), \nabla B(\mathbf{x}^\alpha))\Big)^\top)^\top K_l^{-1} \tag{3.133}$$

$$\mathbf{\Gamma}_l = \mathbf{\Gamma}_{l-1} + J_l\Big(\mathbf{b}_l$$
$$- (\Phi_l + \frac{1}{2}\Theta_l + \Omega(\delta R_2^\alpha)\Phi_l + \varkappa\Big(\mathrm{Cov}(\Psi_\times^\alpha, \xi_\alpha), \mathrm{Cov}(B(\mathbf{x}^\alpha), \nabla B(\mathbf{x}^\alpha))\Big)^\top)\mathbf{\Gamma}_{l-1}\Big) \tag{3.134}$$

$$\Upsilon_l = \Upsilon_{l-1} + J_l K_l J_l. \tag{3.135}$$

Démonstration

Nous posons $\mathbf{x}^\star$ et $\mathbf{x}^\diamond$ deux points quelconques, pour B_{LER} cette fois-ci au lieu de B_{LE} :

$$\mathbb{E}\left[\begin{bmatrix} B(\mathbf{x}^\star) \\ B(\mathbf{x}^\diamond) \end{bmatrix} |\mathbf{b}\right] = \begin{bmatrix} \mu_{B(\mathbf{x}^\star)} \\ \mu_{B(\mathbf{x}^\diamond)} \end{bmatrix} + K_2 K_3^{-1}(\mathbf{b} - \mu_{B_{LE}(\mathbf{x}^\alpha)}) \tag{3.136}$$

$$\mathrm{Cov}\left(\begin{bmatrix} B(\mathbf{x}^\star) \\ B(\mathbf{x}^\diamond) \end{bmatrix}, \begin{bmatrix} B(\mathbf{x}^\star) \\ B(\mathbf{x}^\diamond) \end{bmatrix} |\mathbf{b}\right) = \mathrm{Cov}\left(\begin{bmatrix} B(\mathbf{x}^\star) \\ B(\mathbf{x}^\diamond) \end{bmatrix}, \begin{bmatrix} B(\mathbf{x}^\star) \\ B(\mathbf{x}^\diamond) \end{bmatrix}\right) - K_2 K_3^{-1} K_2^\top \tag{3.137}$$

Le conditionnement s'effectue sur une valeur mesurée au point $\mathbf{x}^\alpha$, avec α l'index de la mesure. La localisation n'est pas parfaite : par conséquent il faut introduire une incertitude sur la position $\mathbf{x}^\alpha$. Nous connaissons les expressions de K_2 et K_3, calculées dans la partie précédente :

$$(K_2)_{1:3,1:3} \simeq \kappa(\mathbf{x}^\star, \mathbf{x}^\alpha)(I_3 + \Omega(\delta R_2^\alpha)^\top) + \frac{1}{2}\left\langle \Sigma^{t^\alpha}, H_2(\kappa)(\mathbf{x}^\star, \mathbf{x}^\alpha)\right\rangle$$
$$+ \varkappa\Big(\mathrm{Cov}(\Psi_\times, \xi), \mathrm{Cov}(B(\mathbf{x}^\star), \nabla B(\mathbf{x}^\alpha))\Big) \tag{3.138}$$

$$
\begin{aligned}
(K_2)_{4:6,1:3} \simeq \kappa(\mathbf{x}^\diamond,\mathbf{x}^\alpha)(I_3+\Omega(\delta R_2^\alpha)^\top)+\frac{1}{2}\Big\langle \Sigma^{t^\alpha}, H_2(\kappa)(\mathbf{x}^\diamond,\mathbf{x}^\alpha)\Big\rangle \\
+\varkappa\Big(\mathrm{Cov}(\Psi_\times,\xi),\mathrm{Cov}(B(\mathbf{x}^\diamond),\nabla B(\mathbf{x}^\alpha))\Big)
\end{aligned}
\tag{3.139}
$$

$$
\begin{aligned}
K_3 \simeq \kappa(\mathbf{x}^\alpha,\mathbf{x}^\alpha)+\Big\langle \Sigma^{t^\alpha}, H_1(\kappa)(\mathbf{x}^\alpha,\mathbf{x}^\alpha)\Big\rangle+\sigma_b^2 I_3+\mathrm{Var}\left(\nabla B(\mathbf{x}^\alpha)\xi^\alpha\right) \\
+\kappa(\mathbf{x}^\alpha,\mathbf{x}^\alpha)\Omega(\delta R_2^\alpha)^\top+\Omega(\delta R_2^\alpha)\kappa(\mathbf{x}^\alpha,\mathbf{x}^\alpha) \\
+2\varkappa\Big(\mathrm{Cov}(\Psi_\times^\alpha,\xi_\alpha),\mathrm{Cov}(B(\mathbf{x}^\alpha),\nabla B(\mathbf{x}^\alpha))\Big)+\mathrm{Var}\left(\Psi^\alpha B(\mathbf{x}^\alpha)\right)
\end{aligned}
\tag{3.140}
$$

Pour plus de simplicité, $\mathbf{x}^\alpha=\mathbf{x}$, $\Sigma^{t^\alpha}=\Sigma$, $\phi=\phi(\mathbf{x})=\phi(\mathbf{x}^\alpha)$, $\phi_\star=\phi(\mathbf{x}^\star)$ et $\phi_\diamond=\phi(\mathbf{x}_\diamond)$. Comme B est le gradient du potentiel scalaire modélisé par processus Gaussien, c'est le gradient de ϕ et non ϕ qui apparaît dans nos équations. Nous souhaitons écrire $\varkappa\Big(\mathrm{Cov}(\Psi_\times,\xi),\mathrm{Cov}(B(\mathbf{x}'),\nabla B(\mathbf{x}))\Big)$ sous une forme à rang réduit (nous connaissons déjà l'expression à rang réduit pour les autres termes) :

$$
\begin{aligned}
\varkappa\Big(\mathrm{Cov}(\Psi_\times,\xi),\mathrm{Cov}(B(\mathbf{x}^\star),\nabla B(\mathbf{x}))\Big) &= \sum_{k=1}^{3}\sum_{l=1}^{3}\mathrm{Cov}((\Psi_\times)_{:,k},\xi_l)\frac{\partial}{\partial \mathbf{x}_l}\mathrm{Cov}(B(\mathbf{x}^\star),B_k(\mathbf{x})) \\
&= \sum_{k=1}^{3}\sum_{l=1}^{3}\mathrm{Cov}((\Psi_\times)_{:,k},\xi_l)\frac{\partial}{\partial x_l}\kappa(\mathbf{x}^\star,\mathbf{x}_k) \\
&= \sum_{k=1}^{3}\sum_{l=1}^{3}\mathrm{Cov}((\Psi_\times)_{:,k},\xi_l)\frac{\partial}{\partial x_l}(\nabla\phi_\star\Lambda\nabla\phi^\top)_{:,k} \\
&= \nabla\phi_\star\Lambda\sum_{k=1}^{3}\sum_{l=1}^{3}\mathrm{Cov}((\Psi_\times)_{:,k},\xi_l)\frac{\partial(\nabla\phi^\top)_{:,k}}{\partial \mathbf{x}_l} \\
&= \nabla\phi_\star\Lambda\omega_{\Psi_\times,\mathbf{x}}
\end{aligned}
\tag{3.141}
$$

Ce résultat est valable pour un couple de points. Pour plusieurs points de données, et sachant que $(\Psi_\times)_{:,k}$ dépend du point étudié. Il s'agit de faire plusieurs fois le même calcul avec les différents points et de concaténer les matrices :

$$
\varkappa_{all}=\nabla\phi_\star\Lambda\begin{bmatrix}\omega_{\Psi_\times^1,\mathbf{x}_1}\\ \omega_{\Psi_\times^2,\mathbf{x}_2}\\ \dots \\ \omega_{\Psi_\times^n,\mathbf{x}_n}\end{bmatrix}
\tag{3.142}
$$

Nous pouvons donc écrire K_2 simplement sous une forme à rang réduit.

Pour K_3, le dernier terme est différent de ceux déjà explicités pour B_{LE}. Nous écrivons son expression de façon explicite :

$$
\begin{aligned}
(\mathrm{Var}\left(\Psi^{\alpha}_{\times} B(\mathbf{x}^{\alpha})\right)) &= \sum_{k=1}^{3}\sum_{l=1}^{3} \mathrm{Cov}(\Psi^{\alpha}_{k}, \Psi^{\alpha}_{l})\, \mathbb{E}\left[B(\mathbf{x}^{\alpha})_{\times,:,k} B(\mathbf{x}^{\alpha})_{\times,:,l}\right] \\
&= \sum_{k=1}^{3}\sum_{l=1}^{3} \mathrm{Cov}(\Psi^{\alpha}_{k}, \Psi^{\alpha}_{l})\Big(\mathrm{Cov}(B(\mathbf{x}^{\alpha})_{\times,:,k}, B(\mathbf{x}^{\alpha})_{\times,:,l}) + \mathbb{E}\left[B(\mathbf{x}^{\alpha})_{\times,:,k}\right]\mathbb{E}\left[B(\mathbf{x}^{\alpha})_{\times,:,l}\right]\Big)
\end{aligned}
\tag{3.143}
$$

Comme mentionné au début de cette section, les résultats expérimentaux obtenus se trouvent dans la Section 4.4.2 du chapitre suivant car, pour vérifier le fonctionnement de nos équations, nous avons besoin de données réelles de trajectoires. Nous appliquons donc nos résultats théoriques à un algorithme de localisation utilisant le champ magnétique, tel que celui décrit dans le Chapitre 4.

Conclusion

Ces travaux permettent donc de modéliser de façon complète l'erreur sur les poses utilisées pour la création d'une carte du champ magnétique. Ils reposent sur le développement de la formule de Taylor à l'ordre 2. Pour les rotations et les positions, les erreurs sont de types différents et en fonction de leur amplitude, l'erreur de position ou l'erreur de rotation peut avoir une influence plus ou moins forte sur les résultats finaux. Si les erreurs de rotation n'ont qu'un impact très faible par rapport aux erreurs de position, il peut sans doute être judicieux de se limiter à un développement à l'ordre 1. Dans cette partie cependant nous avons voulu nous placer dans le cas le plus général possible.

Une hypothèse forte est faite sur le modèle : l'indépendance entre les différentes mesures, qui n'est pas vérifiée pour les algorithmes de localisation. Elle permet de beaucoup simplifier les calculs mais l'ajout des corrélations entre les mesures dans les équations pourrait permettre d'obtenir une modélisation plus proche de la réalité.

Maintenant que nous avons étudié le modèle de création d'une carte magnétique, nous allons pouvoir l'utiliser pour améliorer la robustesse d'un algorithme visuel-inertiel, le MSCKF.

Chapitre 4

Un SLAM magnéto-visuel-inertiel par fusion serrée

4.1 État de l'art sur la localisation magnétique 86
4.1.1 Relocalisation magnétique dans une carte pré-existante 86
4.1.2 SLAM magnétiques 88
4.2 Le MSCKF magnéto-visuel-inertiel 89
4.2.1 Vecteur d'état 89
4.2.2 Propagation 89
4.2.3 Mesures magnétiques du MSCKF 90
4.3 MSCKF magnéto-visuel-inertiel exploitant une carte magnétique créée hors ligne 91
4.3.1 Protocole expérimental 91
4.3.2 Création de la carte magnétique 92
4.3.3 Résultats de localication du MSCKF magnéto-visuel-inertiel 93
4.3.4 Comparaison avec un algorithme de SLAM visuel-inertiel réutilisant une carte visuelle 96
4.4 MSCKF magnéto-visuel-inertiel avec création de carte en ligne 100
4.4.1 Résultats avec le modèle classique de création de carte 100
4.4.2 Ajout de l'incertitude de localisation dans le modèle 101

Introduction

Dans ce chapitre, après un bref état de l'art (Section 4.1), nous présentons l'algorithme de fusion serrée des données magnétiques, visuelles et inertielles développé au cours de cette thèse (Section 4.2). La localisation est estimée conjointement par les trois capteurs.

Comme mentionné dans le Chapitre 2, les performances des algorithmes de reconnaissance visuelles se dégradent au cours du temps en raison des changements de luminosité et des modifications de l'apparence de l'environnement (meubles déplacés, nouvelle décoration, etc). Le champ magnétique n'est pas influencé par ces perturbations (sauf s'il s'agit d'objets métalliques très volumineux). L'ajout des informations du magnétomètre à celles de la caméra et de l'IMU permet de limiter le phénomène de dérive de façon robuste sur de longues périodes de temps (plus d'un an

comme démontré dans la Section 4.3). Contrairement aux cartes visuelles, les cartes magnétiques varient très peu au cours du temps au sein de bâtiments tels que des bureaux.

Les résultats expérimentaux permettant de vérifier ces assertions sont présentés après la description du MSCKF magnéto-visuel-inertiel (Section 4.3 avec création de carte hors ligne et Section 4.4 sans carte préalable).

4.1 État de l'art sur la localisation magnétique

Dans cette section, les différentes approches permettant de se localiser à partir du champ magnétique en intérieur sont présentées. On peut distinguer deux types d'applications. Le magnétomètre peut être utilisé pour désambiguïser la relocalisation (i. e. pour savoir où le système se trouve dans une carte, Section 4.1.1). Il peut aussi être utilisé pour se localiser au cours du temps, afin de corriger des trajectoires en ligne : le magnétomètre est alors couplé à un algorithme d'odométrie indépendant (Section 4.1.2). Ces approches utilisent une carte magétique pré-existante ou bien créée en ligne.

4.1.1 Relocalisation magnétique dans une carte pré-existante

Lorsque l'on possède une carte magnétique complète d'un lieu, on peut utiliser un magnétomètre pour s'y relocaliser. Le champ magnétique mesuré est un champ vectoriel et il peut avoir la même valeur à plusieurs endroits différents, créant des ambigüités. Il est nécessaire de coupler le magnétomètre à un autre capteur ou utiliser des séquences de données magnétiques pour obtenir des résultats corrects. La relocalisation peut être magnéto-visuelle par exemple, la caméra et le magnétomètre apportant des informations complémentaires. Dans la littérature, deux types de relocalisation utilisant un magnétomètre ont été abordées : LIDAR-magnétomètre ([WWP+19]) et vision-magnétomètre ([LZL+17]).

4.1.1.1 Relocalisation LIDAR-magnétomètre

Dans [WWP+19] est présentée une méthode de relocalisation s'appuyant sur la méthode des k-plus proches voisins pour sélectionner les zones de champ magnétique proche de celui observé. Le LIDAR est utilisé ensuite pour sélectionner parmi les k positions candidates laquelle est la plus probable.

La carte utilisée dans [WWP+19] se résume à une carte 2D du champ magnétique. Un modèle d'interpolation linéaire permet de modéliser le champ en tout point, à partir des données collectées, qui sont discrètes. La carte est parcourue en entier dans un sens puis dans l'autre, parallèlement aux murs.

La méthode des k-plus proches voisins permet d'estimer la différence entre les vecteurs magnétiques de la carte et le vecteur mesuré là où se trouve le robot. Le LIDAR permet ensuite, en comparant les profils de la carte qu'il connaît avec l'endroit qu'il vient de scanner, de choisir entre les k candidats restants lequel concorde le mieux avec ses mesures. Avec des hypothèses très restrictives, cette relocalisation a un fort taux de réussite : le robot se déplace toujours parallèlement aux murs et ne peut être dirigé que selon quatre angles : $0, \pi, \frac{\pi}{2}, -\frac{\pi}{2}$. Il n'y a que quatre valeurs possibles que peut renvoyer le magnétomètre pour une même position or, pour un système non contraint en direction (comme la majorité des systèmes utilisés), tous les angles sont possibles selon les 3 axes. Dans ce cas, sans aucune idée de l'orientation, seule l'intensité du champ est utilisable : au lieu d'avoir trois paramètres pour discriminer la position (le champ selon chaque

axe), il n'y en a plus qu'un, la norme du champ. De nombreux endroits possèdent des intensités similaires, ce qui limite beaucoup les informations apportées par le champ magnétique.

Cette méthode apparaît donc comme inapplicable pour un système non contraint en orientation. Elle montre les problèmes que causent la non-connaissance a priori du cap pour pouvoir discriminer plusieurs zones entre elles à l'aide du champ magnétique. Mais pour un algorithme de localisation avec plusieurs capteurs, qui déterminent conjointement la position et l'orientation, ce problème ne se présente pas aussi fortement dans le cas d'une grande trajectoire car l'orientation est estimée avec une bonne précision grâce à la complémentarité des capteurs.

4.1.1.2 Relocalisation vision-magnétomètre

Dans [LZL+17], les auteurs abordent la relocalisation par une approche fusion visuelle-magnétique. La carte est réalisée en découpant l'espace en cellules de taille 60×60 cm. Pour chaque cellule, en dix endroits différents, un couple de mesures est pris : une image et la valeur du champ magnétique. Cet ensemble de dix couples constitue l'empreinte de la cellule. Les emplacements sont choisis de façon aléatoire, avec un smartphone tenu verticalement et suivant la direction du couloir.

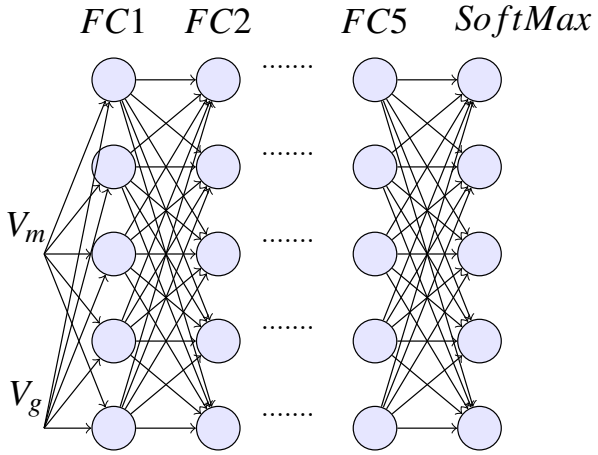


FIGURE 4.1: Schéma du réseau de neurones combinant la vision et le magnétomètre. V_g contient un descripteur de taille 4096 pour les images. V_m correspond aux quatre valeurs du champ magnétique $(m_x, m_y, m_z, ||m||)$. FC signifie "fully connected layer".

Les images sont traitées par un réseau de neurones qui renvoie un descripteur de taille 4096. Un second réseau de neurones (Figure 4.1) prend en entrée ce descripteur et 4 valeurs magnétiques : la valeur du champ selon chaque direction et son amplitude totale. Il retourne un descripteur de taille 4196, qui correspond à la taille du vecteur de description d'une des couches du réseau de neurones : il combine l'image et les signaux magnétiques.

La carte obtenue permet de se relocaliser efficacement, avec un taux de succès de localisation de 91% à 0.81 m près (diagonale de la cellule), dans deux des environnements testés, et à 1.25 m près pour deux autres (au total, quatre environnements ont été explorés). Mais la procédure d'acquisition de données pour cette méthode est très fastidieuse : il faut explorer chaque cellule, y prendre dix mesures en tenant un téléphone de façon adéquate pour les acquisitions. Les images et les mesures magnétiques sont toutes acquises selon la même direction : cette méthode est donc totalement dépendante du cap.

Ainsi, on observe qu'un des problèmes majeurs présentés par la relocalisation magnétique est de parvenir à connaître l'orientation du système. Sans une connaissance du cap, seule l'amplitude du champ magnétique est une empreinte utilisable de la zone. Cependant, même en connaissant son orientation, le champ magnétique ne se compose que de trois valeurs et cela est insuffisant pour caractériser un endroit de façon unique. Des capteurs complémentaires doivent donc être utilisés.

4.1.2 SLAM magnétiques

Actuellement, les méthodes de SLAM exploitant le champ magnétique combinent le magnétomètre avec une centrale inertielle ([VHKR11], [KS18])) ou un système d'odométrie quelconque ([RFA+13]).

Dans la majorité des publications consultées, la fusion avec le magnétomètre se fait par filtre particulaire. La différence réside dans le modèle de prédiction du champ utilisé (qui sont présentés plus haut, appliqués à la création de cartes : Section 3.1.4). Les trois méthodes employées pour la création de la carte sont le découpage de la carte en une grille, avec pour chaque cellule la valeur mesurée correspondante si cette case a déjà été explorée ([RFA+13]), l'utilisation de trois processus gaussiens pour modéliser les trois directions du champ ([VHKR11]) et la modélisation du champ comme un potentiel scalaire ([KS18]).

Pour cette dernière modélisation, l'équation fondamentale utilisée pour prédire la nouvelle valeur magnétique s'appuie directement sur la prédiction du champ magnétique au point que l'on étudie, à laquelle on applique la matrice de rotation du système [KS18] :

$$\hat{\mathbf{m}}_t^M = R_{MfromG}\hat{\mathbf{m}}(\mathbf{x}_t^G) + \varepsilon_t \tag{4.1}$$

avec $\mathbf{x}_t^G$ la position dans le repère global et $\hat{\mathbf{m}}(\mathbf{x}_t^G)$ la valeur magnétique prédite par la carte au point $\mathbf{x}_t^G$.

Le SLAM implémenté utilise une centrale inertielle, une caméra et un magnétomètre. Les acquisitions sont réalisées grâce à un téléphone. Une trajectoire effectuée avec ce SLAM consiste à tourner cinq fois dans les mêmes couloirs, en passant par le même chemin. Avec les résultats odométriques seuls renvoyés par le logiciel ARKIT utilisant la caméra et la centrale inertielle, la déviation de la localisation entre le premier et le dernier passage est de 5 m. En revanche, en utilisant aussi les données du magnétomètre en fusion lâche pour corriger la trajectoire calculée par ARKIT, l'écart n'est plus que de quelques dizaines de centimètres. Donc l'utilisation du magnétomètre améliore nettement la localisation.

Pour une autre trajectoire sur trois étages, aller puis retour, le SLAM magnétique améliore aussi la localisation : l'itinéraire de retour est proche de l'itinéraire d'arrivée, l'ajout du magnétomètre permet de mieux le restituer qu'avec l'odométrie calculée par ARKIT seul. Néanmoins, pour le début de la trajectoire, tant qu'il n'y a pas fermeture de boucle ou que la même zone est explorée une deuxième fois, des problèmes de dérive liés à la fusion visuelle-inertielle peuvent apparaître et le magnétomètre aide très peu à les compenser. À l'aller, le magnétomètre construit une carte magnétique en s'appuyant sur les informations de l'odométrie et la fusion lâche du magnétomètre ne contribue pas à corriger le biais de l'IMU, bien que les informations du magnétomètre soient utiles pour le déterminer plus précisément. C'est le seul exemple de SLAM magnétique réalisé en 3D. La fusion des données est uniquement lâche alors qu'une fusion serrée pourrait permettre de corriger la trajectoire encore plus efficacement.

Aujourd'hui, ce SLAM utilisant les données magnétiques peut être utilisé pour se localiser sur des trajectoires simples mais il est encore difficile de le mettre en place sur de grandes distances,

plus complexes.

D'autres algorithmes dérivés de cette méthode et combinant la centrale inertielle et le magnétomètre seuls ont été proposés récemment [OVK22, VHK22] mais ne sont eux aussi utilisables que dans des zones de taille réduite. Il s'agit de deux méthodes de fusion serrée combinant la centrale inertielle et le magnétomètre, dans le premier cas avec un filtre particulaire et dans le deuxième avec un filtre de Kalman étendu. Mais la localisation n'est possible que dans un lieu déjà connu et cartographié en utilisant des instruments de capture de mouvement car il n'est pas possible de créer une carte du champ magnétique de façon précise seulement à l'aide d'une IMU. La vision, qui est absente dans ces deux cas, permettrait de robustifier le système : les trajectoires magnéto-inertielles présentées sont limitées à de petites zones de la taille d'une pièce.

4.2 Le MSCKF magnéto-visuel-inertiel

Le MSCKF magnéto-visuel-inertiel est présenté dans cette section. L'objectif de ce chapitre est d'intégrer les mesures magnétiques dans le MSCKF visuel-inertiel déjà existant. En se basant sur le Chapitre 2, les modifications apportées au vecteur d'état sont présentées (Section 4.2.1), avec les changements qu'elles induisent pour l'étape de propagation (Section 4.2.2). L'étape de mesure magnétique est aussi décrite (Section 4.2.3).

4.2.1 Vecteur d'état

Dans toute la suite, le référentiel du magnétomètre est supposé confondu avec I (le repère de la centrale inertielle). Cette hypothèse est vérifiée ici car l'IMU que nous utilisons contient un magnétomètre avec ses axes alignés sur ceux de l'IMU.

La carte magnétique est exprimée dans le référentiel global G.

Le vecteur d'état du MSCKF magnéto-visuel-inertiel est très proche d'un MSCKF classique, mais on y ajoute le biais du magnétomètre car, comme celui de l'accéléromètre et du gyromètre, il est important de le réévaluer souvent (voir la Section 2.1.4.4 pour la démonstration) :

$$\begin{aligned} \mathbf{X} &= (\mathbf{x}^C, \mathbf{x}^W), \\ \mathbf{x}^C &= (\mathrm{R}_{GI}, \mathbf{t}_{GI}, \mathbf{v}_{GI}, \mathbf{b}_g, \mathbf{b}_a, \mathbf{b}_m), \\ \mathbf{x}^W &= (\mathrm{R}^1_{GI}, \mathbf{t}^1_{GI}, \ldots, \mathrm{R}^W_{GI}, \mathbf{t}^W_{GI}), \end{aligned} \tag{4.2}$$

avec les mêmes notations que dans la Définition 2.22 et $\mathbf{b}_m$ le biais du magnétomètre. Pour simplifier l'écriture de la jacobienne magnétique, on préfère ici la notation avec la matrice de rotation plutôt que les quaternions, avec R_{GI} la rotation du repère G dans le repère I. La conversion de rotation à quaternion et inversement s'obtient simplement en appliquant les relations d'équivalence. La covariance de l'état $\mathbf{X}$ s'écrit aussi P.

4.2.2 Propagation

L'accéléromètre et le gyroscope fournissent des mesures de l'accélération et de la vitesse angulaire qui sont intégrées pour prédire la position et l'orientation, ainsi que leur covariance. Les équations de propagation IMU ont été détaillées dans la Section 2.2.2 et plus de détails se trouvent dans la publication de référence [MR07]. Avec les données magnétiques, une équation de propagation sur le biais du magnétomètre est ajoutée dans le MSCKF :

$$\begin{aligned} \dot{\mathbf{b}}_m &= \boldsymbol{\eta}_{wm} \\ \text{avec } \boldsymbol{\eta}_{wm} &\sim \mathcal{N}(\mathbf{0}_3, \sigma^2_{wm} I_3). \end{aligned} \tag{4.3}$$

La propagation de $\mathbf{b}_m$ pendant Δt est obtenue en mettant à jour sa covariance :

$$\mathrm{P}_{\mathbf{b}_m} \leftarrow \mathrm{P}_{\mathbf{b}_m} + \Delta t\, \sigma_{wm}^2 I_3, \tag{4.4}$$

avec $\mathrm{P}_{\mathbf{b}_m}$ la sous-matrice de taille 3×3 de P correspondant à l'emplacement de la covariance du biais magnétique.

4.2.3 Mesures magnétiques du MSCKF

Pour appliquer la mise à jour par le filtre de Kalman étendu (EKF), telle qu'adaptée au MSCKF, la fonction de mesure $\mathbf{h}$, le jacobien H et la covariance Σ des capteurs doivent être définis. Pour les capteurs visuels, la fonction de mesure $\mathbf{h}_{vision}$ est la même que dans le MSCKF original. C'est la reprojection 2D des points 3D sur la pose actuelle et sur les K dernières poses. Comme dans [MR07], la dépendance de $\mathbf{h}_{vision}$ sur les points 3D est marginalisée avec la transformation left-nullspace.

Pour le magnétomètre, $\mathbf{h}_{mag}$ est la prédiction des valeurs magnétiques à partir de la carte magnétique à la position actuelle $\mathbf{t}_{GI}$:

$$\begin{aligned} \mathbf{h}_{mag}(\mathbf{x}^C) &= \mathbf{A}\mathrm{R}_{GI}^\top \mathbf{m}_G(\mathbf{t}_{GI}) + \mathbf{b}_m \\ \text{avec } \mathbf{m}_G(\mathbf{t}_{GI}) &= \mathbb{E}[B(\mathbf{t}_{GI})] = \nabla\Phi(\mathbf{t}_{GI})\boldsymbol{\mu}_l. \end{aligned} \tag{4.5}$$

La matrice jacobienne H_{mag} de $\mathbf{h}_{mag}$ par rapport à la variable $\mathbf{x}^C$ se divise en trois parties : H_t la dérivée par rapport à la translation, H_R par rapport à la rotation et H_b par rapport au biais (égal à I_3),

$$\mathrm{H}_{mag} = \begin{bmatrix} \mathrm{H}_R & \mathrm{H}_t & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathrm{H}_b \end{bmatrix}. \tag{4.6}$$

Pour H_t, la dérivée est simple à calculer car elle dépend seulement de la dérivée de $\nabla\Phi$:

$$\mathrm{H}_t = \mathbf{A}\mathrm{R}_{GI}^\top \nabla^2\Phi(\mathbf{t}_{GI})\boldsymbol{\mu}_l, \tag{4.7}$$

alors que pour H_R, la dérivée est obtenue en se plaçant dans le groupe $SO(3)$ et en utilisant l'approximation $\exp_{SO(3)}(\delta\theta) \simeq (I_3 + \lfloor\delta\theta\rfloor_\times)$:

$$\begin{aligned} \mathbf{h}_{mag}\big(\exp_{SO(3)}(\delta\theta)\mathrm{R}_{GI}\big) - \mathbf{h}_{mag}(\mathrm{R}_{GI}) &\simeq \mathbf{A}\big(\lfloor\delta\theta\rfloor_\times \mathrm{R}_{GI}\big)^\top \mathbf{m}_G \\ &\simeq -\mathbf{A}\mathrm{R}_{GI}^\top \lfloor\delta\theta\rfloor_\times \mathbf{m}_G \\ &\simeq \mathbf{A}\mathrm{R}_{GI}^\top \lfloor\mathbf{m}_G\rfloor_\times \delta\theta. \end{aligned}$$

Donc H_R est égale à :

$$\mathrm{H}_R = \mathbf{A}\mathrm{R}_{GI}^\top \lfloor\mathbf{m}_G\rfloor_\times. \tag{4.8}$$

La covariance Σ_{mag} complète est composée de la covariance du magnétomètre et de la covariance de la carte magnétique, dont les prédictions sont imparfaites (à cause de l'incertitude du processus gaussien associé). L'incertitude associée à $\boldsymbol{\mu}_l$ est ajoutée au bruit du capteur :

$$\Sigma_{mag} = \sigma_m^2 \mathrm{I}_3 + \mathbf{A}\mathrm{R}_{GI}^\top \mathbb{V}[B(\mathbf{t}_{GI})]\mathrm{R}_{GI}\mathbf{A}^\top \tag{4.9}$$

avec $\mathbb{V}[B(\mathbf{t}_{GI})] = \nabla\Phi(\mathbf{t}_{GI})\Sigma_l \nabla\Phi(\mathbf{t}_{GI})^\top$.

4.3 MSCKF magnéto-visuel-inertiel exploitant une carte magnétique créée hors ligne

Nous nous intéressons maintenant aux résultats de localisation obtenus par notre algorithme lorsque la carte magnétique a été créée au préalable hors ligne. Après une présentation du protocole utilisé (Section 4.3.1) et de la création de carte magnétique (Section 4.3.2), nous pouvons observer la trajectoire du système calculé par notre méthode dans la Section 4.3.3. En comparant nos résultats à un algorithme visuel-inertiel considéré comme le plus précis de l'état de l'art (Section 4.3.4), on note une grande robustesse au cours du temps des cartes magnétiques, qui permettent de corriger de façon efficace et robuste la dérive.

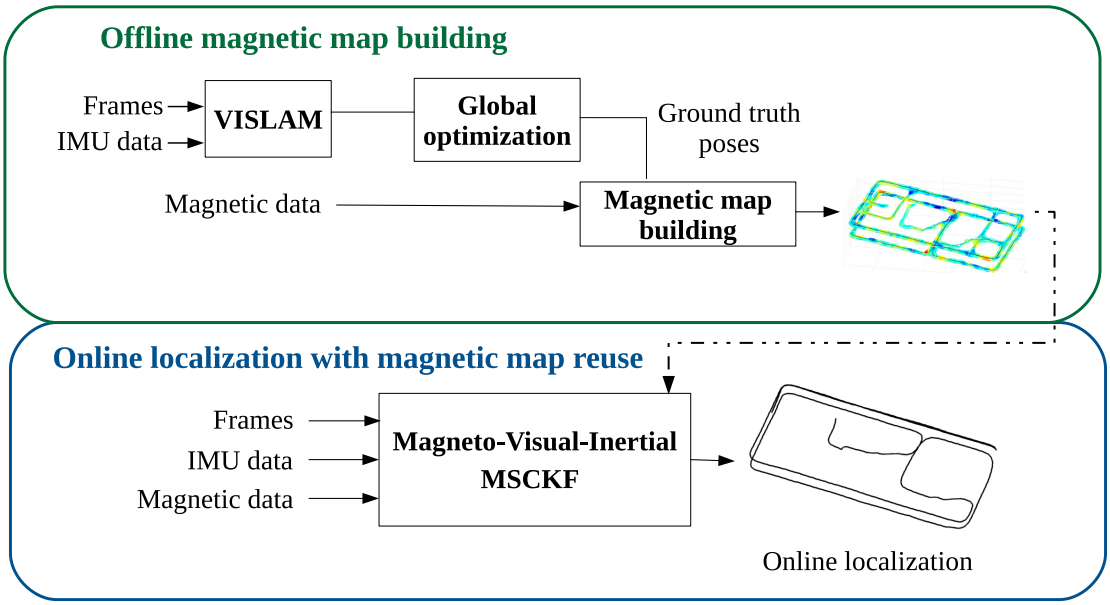


FIGURE 4.2: Principe général du MSCKF magnéto-visuel-inertiel avec carte magnétique construite hors ligne. La carte est obtenue à partir de données magnétiques et de positions vérité-terrain. En ligne, la carte magnétique est réutilisée par le MSCKF pour une localisation sans dérive et robuste au cours du temps.

4.3.1 Protocole expérimental

À notre connaissance, il n'existe pas de jeu de données disponible dans d'anciennes publications fournissant à la fois des données magnétiques, visuelles et inertielles, synchronisées et calibrées, acquises sur de longues périodes de temps (un an ou plus). Par conséquent, nous avons créé notre propre jeu de données. Notre plateforme d'acquisition est la même pour toutes les expériences (Section 2.1). Plusieurs acquisitions de données magnétiques, visuelles et inertielles ont été réalisées entre mai 2020 et août 2021.

Nous comparons le MSCKF magnéto-visuel-inertiel avec une carte magnétique déjà créée avec les deux algorithmes présentés dans le Chapitre 2 : le MSCKF visuel-inertiel [MR07] et l'algorithme de SLAM visuel-inertiel par ajustement de faisceaux [CER+21]. Tous ces algorithmes fonctionnent en utilisant les données d'une seule caméra du casque pour une comparaison équi-

table (la caméra avant gauche). Les quatre flux de caméras ne sont utilisés que pour obtenir la vérité-terrain de la pose du système au cours du temps. Pour cela, un algorithme VISLAM basé sur des images clés avec fermeture de boucle est utilisé, suivi d'un ajustement de faisceaux global. Il est exécuté sur les 4 flux de caméra, les données IMU et les données GNSS. Le GNSS est utilisé pour géoréférencer la vérité-terrain (les premières données sont acquises à l'extérieur). Les positions vérité-terrain sont nécessaires pour la cartographie magnétique (Section 4.3.2) et pour l'évaluation de la précision de localisation par notre MSCKF magnéto-visuel-inertiel (Sections 4.3.3 et 4.3.4). Trois critères sont utilisés pour évaluer la précision de la localisation : l'erreur de translation absolue (Absolute Translation Error, ATE) et deux erreurs de rotation, sur l'azimut et la déclinaison (azimuth et leveling). L'azimut décrit l'erreur d'orientation dans le plan horizontal perpendiculaire à la Terre, et la déclinaison l'erreur d'orientation selon l'axe, lié à l'altitude, perpendiculaire à ce plan.

L'ATE est définie comme $e = \sqrt{\frac{1}{n}\sum_{j=1}^{n} ||\mathbf{t}_{GI}^{j} - \hat{\mathbf{t}}_{GI}^{j}||^2}$ avec n le nombre de données. Les erreurs de rotation sont $e_{azimuth} = \sqrt{\frac{1}{n}\sum_{j=1}^{n} |(R_{GI}^{j} \ominus \hat{R}_{GI}^{j})_z|^2}$ et $e_{leveling} = \sqrt{\frac{1}{n}\sum_{j=1}^{n} ||(R_{GI}^{j} \ominus \hat{R}_{GI}^{j})_{xy}||^2}$, avec $\ominus : SO(3) \times SO(3) \to \mathbb{R}^3$, $R_1 \ominus R_2 = \log_{SO(3)}(R_1 R_2^{\top})$ ce qui résulte en un vecteur de dimension 3 représentant la différence de rotation sur chaque axe. $\mathbf{t}_{GI}$ et $\mathbf{R}_{GI}$ sont les positions et rotations de la vérité-terrain et $\hat{t}_{GI}$ et $\hat{R}_{GI}$ leur estimation par un des algorithmes en ligne.

4.3.2 Création de la carte magnétique

Ici, nous nous intéressons à la localisation dans un environnement déjà connu, pour lequel nous disposons de cartes magnétiques. L'étape préalable à la localisation par le MSCKF magnéto-visuel-inertiel décrit dans la Section 4.2 est la construction hors ligne d'une carte magnétique (voir la partie encadrée en vert de la Figure 4.2, la partie en bleu correspond à la description du MSCKF dans la Section 4.2). Elle nécessite la mesure, à des positions dans l'espace aussi nombreuses que possible, $\mathbf{x_1}, ..., \mathbf{x_n}$, de leur champ magnétique respectif $\mathbf{m_1}, ..., \mathbf{m_n}$. La précision de la carte dépend de la qualité des données d'entrée. Pour des positions précises dans l'espace, nous utilisons les positions de la vérité-terrain obtenues à l'aide du protocole décrit dans la Section 2.3. Pour les mesures magnétiques, la calibration du magnétomètre avant de construire une carte magnétique est cruciale.

La calibration magnétique peut se faire à l'extérieur (voir Section 2.1.4.4 et l'article [RAL10] pour plus de détails sur la calibration en extérieur). C'est la seule et unique fois que le magnétomètre est calibré. Lors de la localisation en ligne, le biais du magnétomètre est initialisé à la valeur utilisée lors de la construction de la carte magnétique puis raffiné par le MSCKF. Il est aussi possible de réaliser la calibration en intérieur mais l'approche decrite dans le Chapitre 5 a été développée plus tard, après la réalisation des expériences présentées ici.

Une fois que les données nécessaires ont été collectées, la première étape de la création de carte magnétique par une régression par processus Gaussien consiste à estimer les hyperparamètres σ_{lin}, σ_{SE}, l_{SE} et σ_m. Ils sont optimisés en minimisant le logarithme de la probabilité marginale (log marginal likelihood, voir la Section 3.1.4.2). La carte magnétique utilisée est décrite en détail dans la Section 3.1.4. Chaque parallélépipède de taille $5 \times 5 \times 2$ mètres est décrit par $m = 8 \times 8 \times 8 = 512$ fonctions propres : selon chaque axe, nous choisissons d'utiliser les 8 premières fréquences possibles des fonctions ϕ_j.

Une carte magnétique construite à partir des données acquises en mai 2020 avec l'approche décrite ci-dessus est illustrée par la Figure 4.3. La longueur de la trajectoire d'apprentissage utilisée pour créer cette carte est de 1150 mètres et elle explore trois étages. La taille de la carte magnétique en terme de mémoire est de 116 Mo. Pour la même trajectoire, la carte visuelle reconstruite

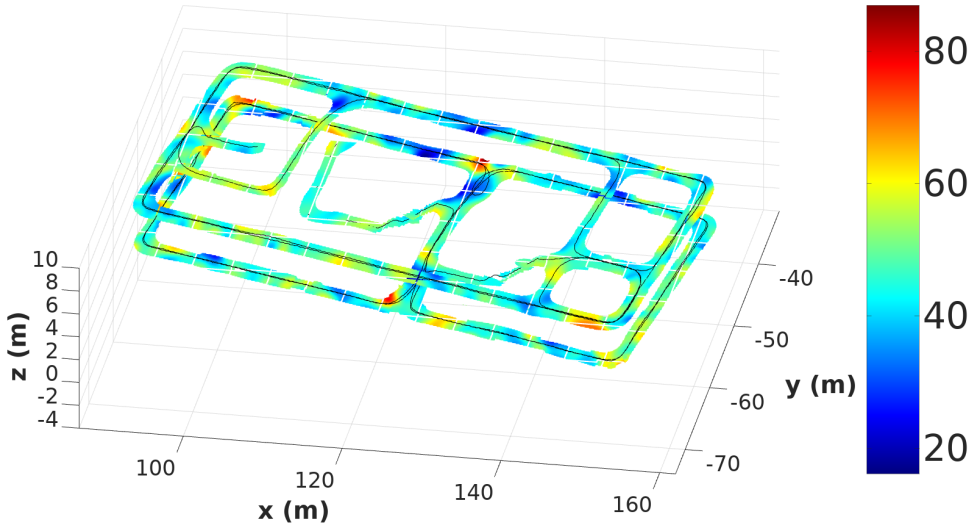


FIGURE 4.3: Norme du champ magnétique prédit (en μT). Seules les prédictions qui sont situées à moins d'un mètre de la trajectoire d'apprentissage (en noir) sont représentées.

par le VISLAM occupe un volume de 240 Mo.

4.3.3 Résultats de localication du MSCKF magnéto-visuel-inertiel

L'intérêt d'ajouter des données magnétiques au MSCKF est évalué dans cette section.

Le MSCKF magnéto-visuel-inertiel est comparé au MSCKF visuel-inertiel qui lui a servi de base [MR07], sur une séquence de 515 mètres acquise en juillet 2021 (soit un an après l'acquisition des données pour construire la carte magnétique). La séquence commence à l'extérieur, les données GNSS sont utilisées dans le MSCKF pour géoréférencer la localisation et ainsi retrouver ses coordonnées dans la carte magnétique pré-construite. Une fois à l'intérieur du bâtiment, lorsque la trajectoire entre dans le domaine de la carte magnétique, le MSCKF proposé ici commence à traiter les données du magnétomètre. Il en résulte une localisation sans dérive comme illustrée sur la Figure 4.4, contrairement à la localisation par MSCKF sans données magnétiques. L'ATE du MSCKF magnéto-visuel-inertiel sur cette séquence est de 32.6 cm alors que pour le MSCKF d'origine, l'ATE est de 2.4 mètres. En termes de précision de la rotation estimée, l'erreur de déclinaison s'aggrave mais pas significativement (0.54° contre 0.38°) alors que l'estimation de l'azimut est grandement améliorée par l'ajout du magnétomètre : l'erreur de déclinaison n'est que de 0.19° avec l'algorithme proposé alors que pour le MSCKF classique il est de 3.51° (ces résultats sont synthétisés dans la troisième colonne pour la séquence du 23 juillet 2021 dans le Tableau 4.1). Nous pouvons observer que le MSCKF visuel-inertiel diverge fortement en altitude, alors que ce n'est pas le cas avec le magnétomètre. Au cours de la séquence, la trajectoire passe par un couloir qui n'appartient pas au domaine de la carte magnétique. Dans ce couloir, la prédiction magnétique (Equation 4.5) est nulle et ignorée automatiquement par le MSCKF : il s'exécute donc comme un MSCKF visuel-inertiel. Les données magnétiques sont à nouveau utilisées lorsque la trajectoire retourne à l'intérieur de la carte magnétique corrigeant les inexactitudes de localisation accumulées entre temps. Toutefois, rester trop longtemps en dehors des zones couvertes par la carte magnétique entraîne une dérive importante et le MSCKF ne sera plus capable d'utiliser la carte magnétique car sa position sera trop différente de celle de la carte. Dans ce cas, pour réutiliser

l'information magnétique dans le MSCKF lorsque nous nous retrouvons à nouveau dans une zone cartographiée magnétiquement, la dérive doit être corrigée au préalable. Une possibilité consiste à utiliser des algorithmes de relocalisation magnétique ([LAH18, MTSB16]). Cependant, ils nécessitent d'accumuler plusieurs secondes de données magnétiques pour reconnaître un lieu. Une autre possibilité est d'utiliser la relocalisation visuelle, alliant un stockage conjoint d'une carte visuelle et d'une carte magnétique, mais comme démontré en Section 4.3.4, cette solution n'est pas robuste aux changements d'apparence dans le temps. Enfin, la relocalisation magnéto-visuelle ([LZL+17]) semble prometteuse pour une relocalisation robuste avec une solution unique et sans ambigüité. Elle permet de disposer de plus d'informations que les approches purement magnétiques. Mais les solutions existantes dépendent fortement de l'angle de vue (s'il est trop différent de celui de la base de données, l'algorithme ne sera pas capable de reconnaître que le système est revenu à un endroit déjà exploré). La gestion des sorties et des retours dans une carte magnétique est un axe de recherche à part entière qui sort du cadre de cette thèse.

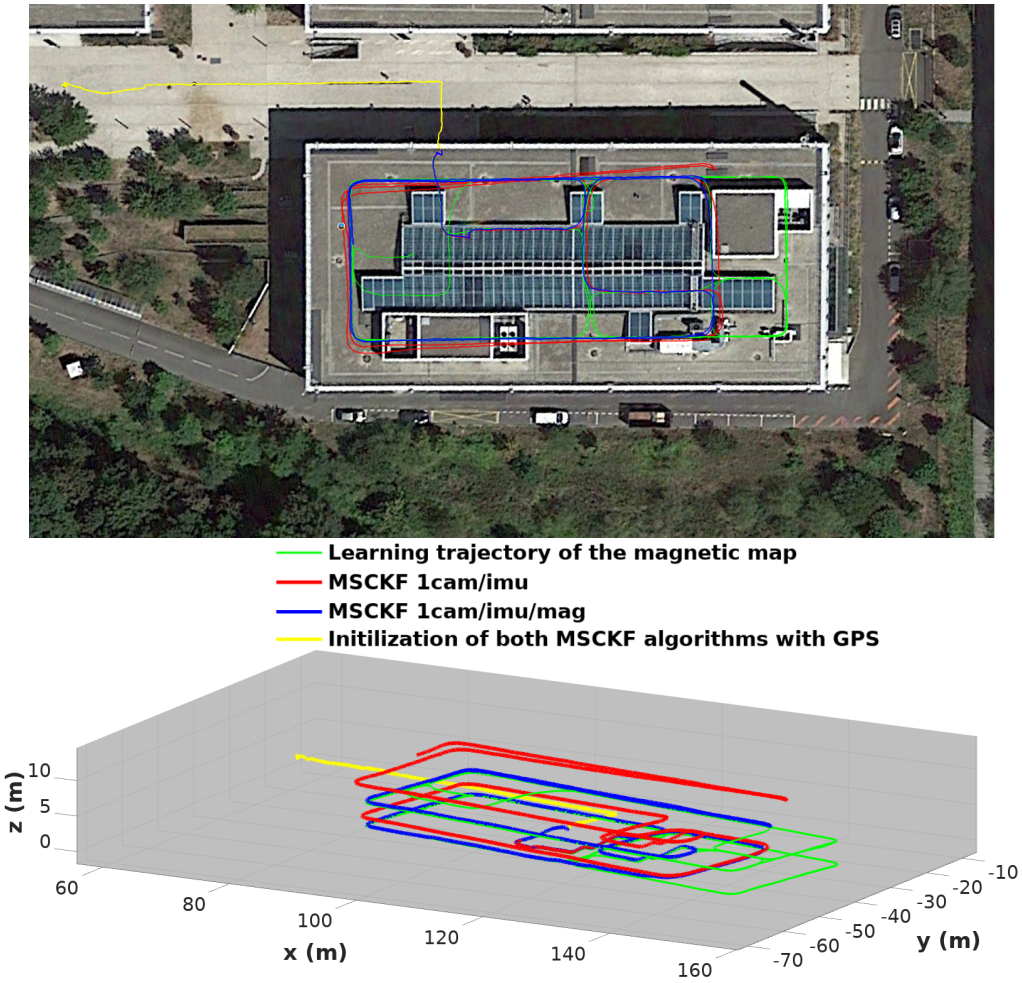


FIGURE 4.4: Localisation obtenue par le MSCKF magnéto-visuel-inertiel (Section 4.2) et le MSCKF visuel-inertiel [MR07] sur une séquence obtenue en juillet 2021. La carte magnétique utilisée date de plus d'un an (29 mai 2020). Les données sont plus récentes (juillet 2021). En haut : vue Google Earth. En bas : vue 3D.

Le MSCKF proposé réestime le biais du magnétomètre. Pour la séquence de la Figure 4.4, le

biais initial est réglé sur sa valeur de mai 2020. Comme le montre la Figure 4.5, le biais du magnétomètre estimé par le MSCKF magnéto-visuel-inertiel converge après quelques secondes à une valeur proche de la vérité terrain (obtenue en recalibrant le magnétomètre juste avant l'acquisition selon la méthode décrite en Section 2.1.4.4). En un an, le biais du magnétomètre a radicalement changé : à partir d'un biais initial de $(-9.07; -10.85; -24.17)\mu$T, les valeurs estimées par le MSCKF proposé se stabilisent autour de $(-2.99; 10.03; 17.30)\mu$T ce qui est proche des résultats de la calibration en extérieur égale à $(-2.57; 10.18; 17.39)\mu$T (à titre de comparaison, la norme du champ magnétique terrestre est d'environ 40μT en France). La covariance initiale du MSCKF doit être soigneusement définie pour que l'estimation du biais soit correcte. Pour les capteurs du système d'acquisition, le réglage des bruits capteurs dépend des incertitudes sur la position initiale : $\sigma^0_{yaw} = 0.05$ rad, $\sigma^0_{xy} = 1$ m et $\sigma^0_{\mathbf{b}_m} = 10$ μT sont des valeurs pertinentes. σ^0_{yaw} et σ^0_{xy} ont des valeurs faibles car ils sont déjà bien estimées au départ grâce au GNSS, qui est disponible au début de la séquence car elle commence en extérieur. Nous souhaitons que la valeur de $\sigma^0_{\mathbf{b}_m}$ soit suffisamment élevée pour permettre à $\mathbf{b}_m$ de converger facilement vers sa valeur réelle : $\mathbf{b}_m$ doit se trouver dans l'intervalle $[\mathbf{b}^0_m - 3\sigma^0_{\mathbf{b}_m}, \mathbf{b}^0_m + 3\sigma^0_{\mathbf{b}_m}]$: c'est l'intervalle de confiance à 99.7% de notre valeur, avec $\mathbf{b}^0_m$ la valeur intiale du biais. D'après nos acquisitions, nous savons que $\mathbf{b}_m$ a été modifié d'environ 41μT en un an : ceci correspond à une différence d'amplitude très forte, bien plus importante que ce que nous avons observé habituellement au bout de plusieurs mois pour différents magnétomètres. Nous estimons donc que cette valeur est un peu plus grande que $3\sigma^0_{\mathbf{b}_m}$ s'il s'agit de réestimer $\mathbf{b}_m$ au bout de quelques mois. L'évolution des prédictions magnétiques (Equation 4.5) pour la séquence de la Figure 4.4 suit de près les variations des données magnétiques brutes comme le montre la Figure 4.6. Cela démontre que la carte magnétique construite un an auparavant est toujours valable, tout comme le modèle du magnétomètre et le procédé d'estimation de biais par le MSCKF magnéto-visuel-inertiel.

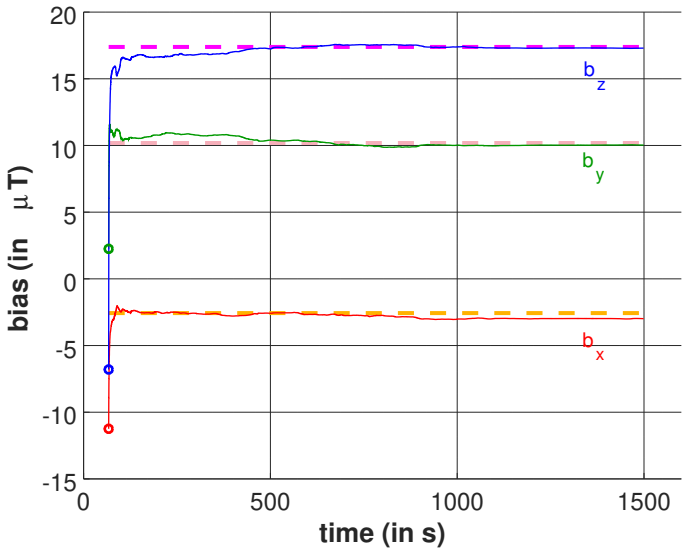


FIGURE 4.5: Estimation du biais pour la séquence de la Figure 4.4 (juillet 2021). Les valeurs de la calibration vérité-terrain sont en pointillés. Elles ont été obtenues en réalisant une calibration en extérieur quelques minutes avant le début de la séquence.

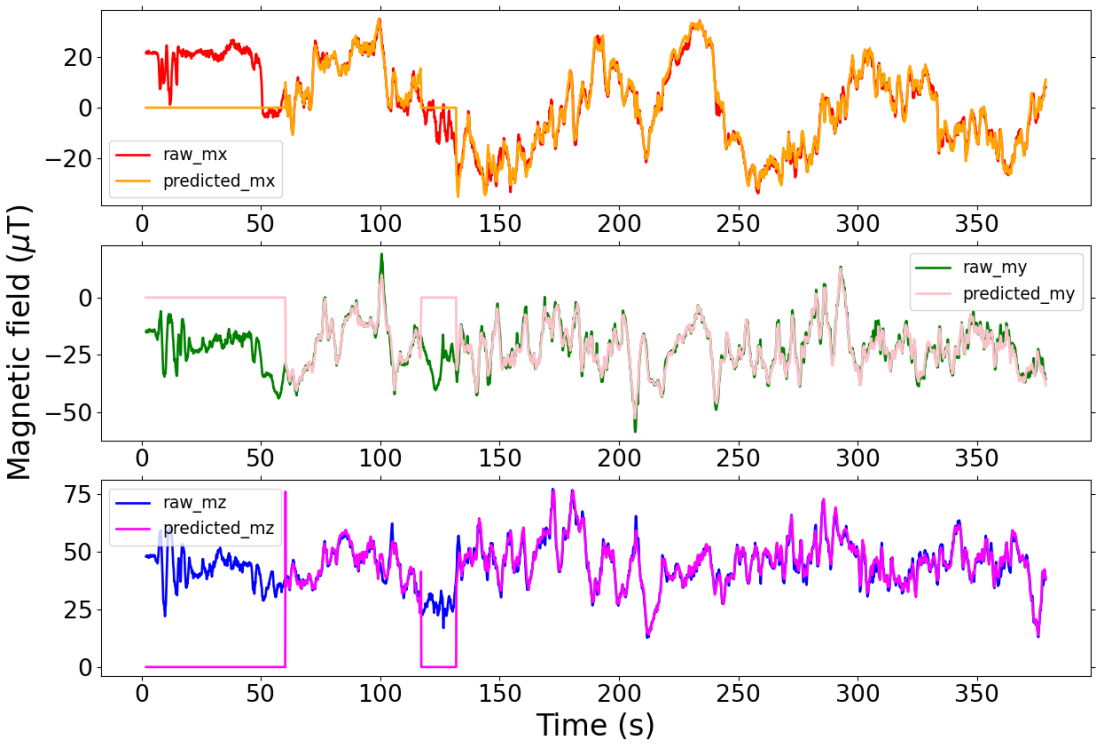


FIGURE 4.6: Comparaison du champ magnétique prédit et des mesures brutes du champ magnétique (après application de la calibration) selon les axes x, y et z pour la séquence de la Figure 4.4 (juillet 2021). En dehors de la carte magnétique les prédictions sont égales à 0. La séquence commence en extérieur pendant les premières 60 secondes. Ensuite, entre la 120ème et la 130ème secondes, la trajectoire passe par un corridor qui est en dehors du domaine cartographié magnétiquement.

4.3.4 Comparaison avec un algorithme de SLAM visuel-inertiel réutilisant une carte visuelle

Dans cette section, le MSCKF magnéto-visuel-inertiel proposé est comparé à un algorithme VISLAM qui réutilise une carte visuelle [CER+21]. Pour cela, les deux algorithmes sont exécutés avec des cartes de plus en plus anciennes sur des séquences correspondant à des trajectoires différentes dans le bâtiment. Les cartes visuelles et magnétiques sont construites simultanément (c'est-à-dire sur les mêmes séquences) à partir des poses vérité-terrain (voir Section 2.3). Le MSCKF visuel-inertiel est également exécuté sur ces séquences. Pour une comparaison équitable, tous les algorithmes sont initialisés avec la même première pose (correspondant à celle de la vérité-terrain).

Les résultats sont répertoriés dans le Tableau 4.1. Le MSCKF visuel-inertiel qui ne réutilise aucune carte est l'algorithme le moins précis. Le VISLAM avec carte visuelle prédéfinie fonctionne légèrement mieux que l'algorithme magnéto-visuel-inertiel lorsque la carte est récente par rapport à la séquence de test. La reconnaissance de point de vue utilisée dans [CER+21] fonctionne bien parce que l'apparence de l'environnement n'a pas beaucoup changé. De plus, les cartes visuelles sont plus discriminantes localement que les cartes magnétiques. D'autre part, la perte moyenne de précision sur la position de l'approche proposée n'est que de 10% sur quelques mois et de 14% après un an.

Date de localisation en ligne		23 juillet 2021 (2 min 23, 212 m)			30 août 2021 (3 min 8, 286 m)			31 août 2021 (1 min 57, 182 m)		
Date de création de carte visuelle ou magnétique		07/23/21	05/18/21	05/29/20	08/30/21	05/18/21	05/29/20	08/31/21	05/18/21	05/29/20
MSCKF 1cam/IMU [MR07]	ATE	4.417 m	4.417 m	4.417 m	1.888 m	1.888 m	1.888 m	3.001 m	3.001 m	3.001 m
	Erreur de déclinaison	**0.38°**	**0.38°**	**0.38°**	0.40°	0.40°	0.40°	0.46°	0.46°	0.46°
	Erreur d'azimut	3.51°	3.51°	3.51°	2.28°	2.28°	2.28°	0.94°	0.94°	0.94°
MSCKF 1cam/IMU/mag avec carte magnétique	ATE	0.318 m	**0.293 m**	**0.321 m**	0.189 m	**0.294** m	**0.255 m**	**0.224 m**	**0.216 m**	**0.257 m**
	Erreur de déclinaison	0.62°	0.51°	0.54°	**0.37°**	0.37°	**0.37°**	**0.46°**	**0.45°**	**0.46°**
	Erreur d'azimut	0.29°	**0.18°**	**0.19°**	0.23°	**0.34°**	**0.35°**	**0.36°**	**0.30°**	**0.49°**
VISLAM 1cam/IMU avec carte visuelle [CER+21]	ATE	**0.262 m**	0.788 m	0.771 m	**0.108 m**	0.582 m	2.317 m	0.435 m	0.775 m	1.576 m
	Erreur de déclinaison	1.04°	1.03°	1.75°	0.41°	**0.32°**	0.99°	1.68°	1.85°	1.43°
	Erreur d'azimut	**0.19°**	0.40°	0.62°	**0.13°**	0.42°	0.89°	1.09°	1.15°	1.32°

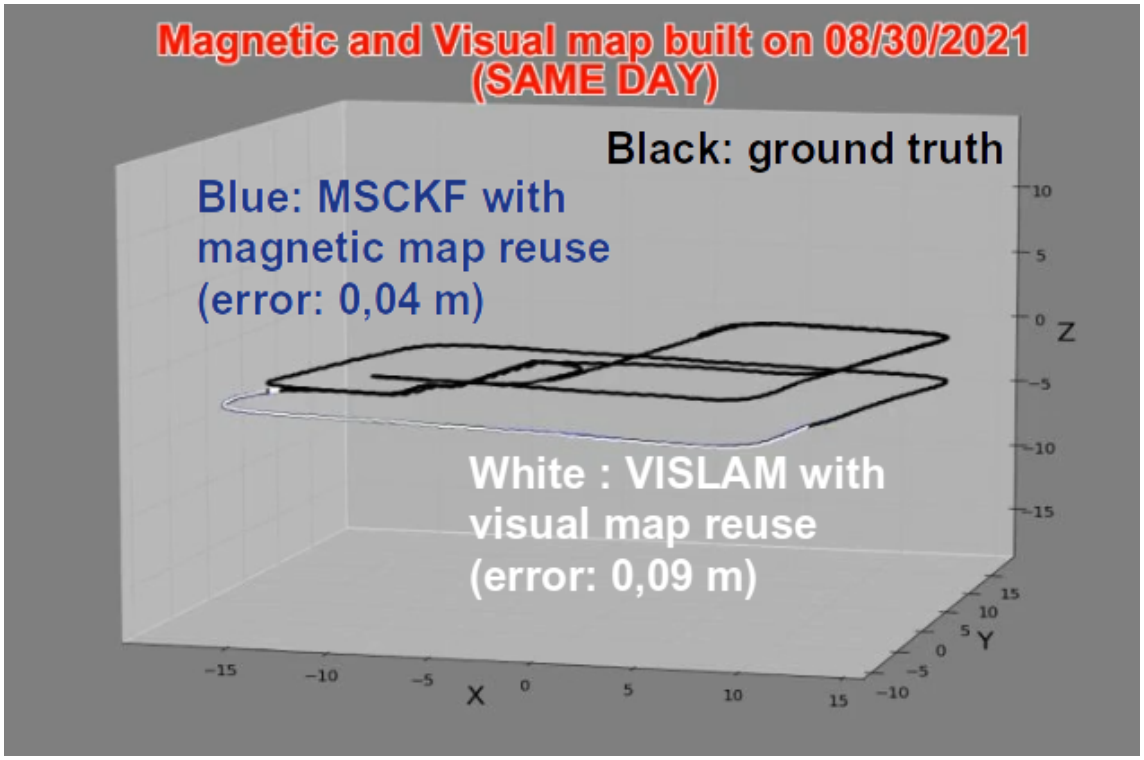
TABLE 4.1: ATE, erreurs de déclinaison et d'azimut du MSCKF [MR07], du MSCKF avec réutilisation de carte magnétique (Section 4.2) et du VISLAM avec réutilisation de carte visuelle [CER+21] sur trois séquences différentes. Ces séquences sont exécutées avec des cartes de plus en plus anciennes (carte visuelle pour [CER+21] et carte magnétique pour l'algorithme proposé). Pour le MSCKF [MR07] qui n'utilise aucune carte, la même erreur de localisation est indiquée pour chaque date associée à la création de cartes.

Pour le VISLAM avec réutilisation de carte visuelle, la précision de la position se dégrade d'un facteur moyen de 2.6 après quelques mois (5.7 après un an). La déclinaison est bien estimée par le MSCKF classique et peu détériorée par l'ajout du magnétomètre (13% en moyenne et pour certaines séquences son estimation est légèrement améliorée). D'autre part, la réutilisation d'une carte visuelle détériore l'estimation de la déclinaison de 180% en moyenne par rapport au MSCKF visuel-inertiel. Mais le MSCKF classique manque de précision pour l'azimut. Réutiliser une carte visuelle pour le VISLAM améliore grandement la qualité de son estimation. Cependant, la précision de l'azimut se dégrade lorsque d'anciennes cartes sont réutilisées (4 fois moins précis lorsqu'une carte visuelle date de plus d'un an est réutilisée). Pour le MSCKF magnéto-visuel-inertiel proposé ici, l'estimation de l'azimut reste précise quel que soit la date de création de la carte magnétique. Les résultats expérimentaux démontrent que la précision de l'algorithme proposé est presque la même sur un an, contrairement aux méthodes VISLAM réutilisant des cartes visuelles dont la précision de localisation se dégrade. La Figure 4.7 montre clairement qu'une carte visuelle datant d'un an ne supprime pas la dérive, alors que la carte magnétique acquise à la même date permet toujours d'obtenir une trajectoire suivant la vérité-terrain.

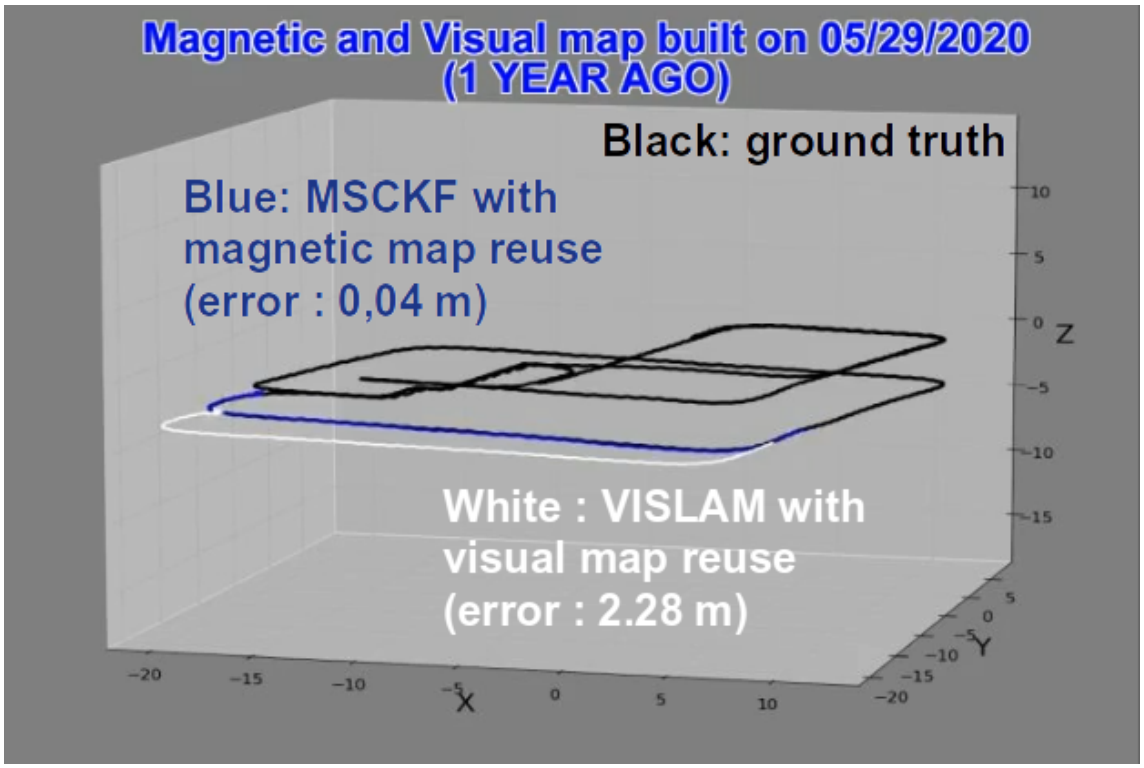
Le temps de traitement des données par le MSCKF magnéto-visuel-inertiel et par le MSCKF sans données magnétiques sont assez proches comme illustré dans le Tableau 4.2. Le temps requis par l'étape magnétique du Kalman est courte puisque $\boldsymbol{\mu}_l$ et Σ_l, les matrices de l'Equation 3.21, ont été pré-calculées. Le traitement d'une donnée magnétique dure environ 1.25 ms, ce qui est 15 fois plus rapide que la cadence d'acquisition des données magnétiques (20 ms par donnée). Le surcoût de calcul sur une séquence entière avec une fréquence de 20 Hz est d'environ 10%. De plus, le MSCKF magnéto-visuel-inertiel est plus robuste à la réduction de fréquence d'image que le MSCKF d'origine comme illustré dans le Tableau 4.2. Le MSCKF magnéto-visuel-inertiel peut traiter moins d'images tout en restant précis, ce qui permet de pouvoir l'utiliser facilement en ligne avec des temps de calcul faibles.

	Temps d'exécution (en s)	ATE	Erreur de déclinaison	Erreur d'azimut
MSCKF [MR07], f_{cam} 20 Hz	108.7	1.888 m	0.40°	2.28°
MSCKF [MR07], f_{cam} 5 Hz	19.5	diverge après 1 min 8 s		
MSCKF [MR07], f_{cam} 2 Hz	5.3	diverge après 50 s		
MSCKF mag, f_{cam} 20 Hz	120.0	0.255 m	0.37°	0.34°
MSCKF mag, f_{cam} 5 Hz	36.6	0.260 m	0.38°	0.53°
MSCKF mag, f_{cam} 2 Hz	18.5	0.351 m	0.40°	0.89°

TABLE 4.2: Temps d'exécution et erreur de localisation du MSCKF avec et sans données magnétiques avec réduction du nombre d'images utilisées.



(a) cartes obtenues le même jour que la séquence de localisation



(b) cartes obtenues un an avant la séquence de localisation

FIGURE 4.7: Comparaisons du MSCKF magnéto-visuel-inertiel et de l'algorithme VISLAM [CER+21] pour une utilisation avec une caméra, une IMU et un magnétomètre (non utilisé par le VISLAM (Séquence 1 du Tableau 4.1). Les résultats sont obtenus quelques minutes après l'initialisation, seule la trajectoire parcourue pendant les 30 dernières secondes avant la position courante est représentée.

4.4 MSCKF magnéto-visuel-inertiel avec création de carte en ligne

Dans cette dernière section, on s'intéresse cette fois-ci à la localisation sans carte magnétique préalable : la carte est créée en ligne uniquement, voir Figure 4.8 pour un schéma du principe de fonctionnement). Les équations sont les mêmes que précédemment pour le MSCKF. La différence principale est que chaque position calculée est aussi utilisée pour mettre à jour la carte magnétique au fur et à mesure. La première Section 4.4.1 étudie les résultats avec le modèle classique de l'état de l'art et la deuxième avec notre modèle prenant en compte les erreurs de position, montrant une amélioration significative des résultats (Section 4.4.2).

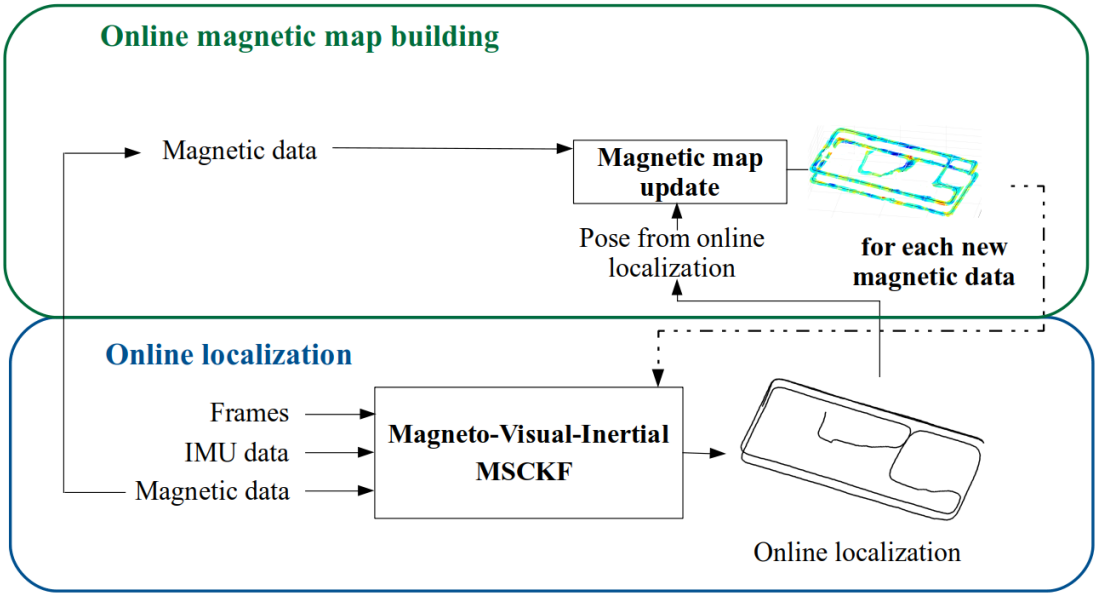
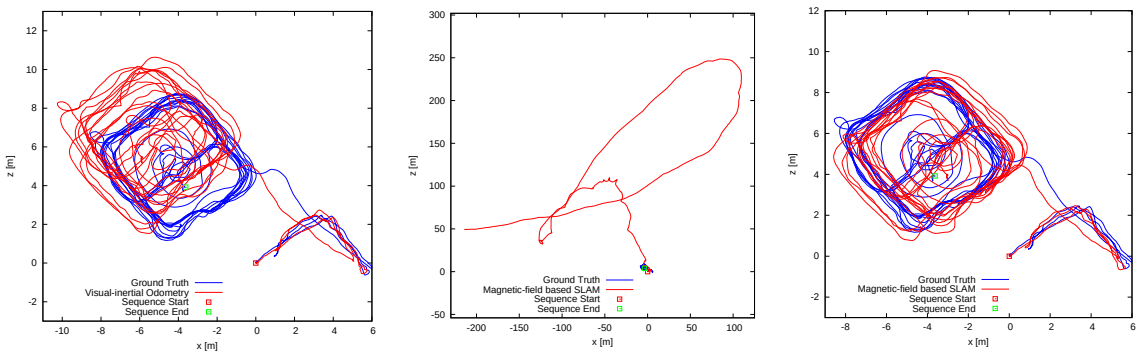


FIGURE 4.8: Principe général du MSCKF magnéto-visuel-inertiel avec carte magnétique construite en ligne. La carte est obtenue à partir de données magnétiques et des positions calculées par notre algorithme. La carte magnétique permet de corriger la dérive principalement lorsque le système repasse par des endroits qu'il a déjà explorés. Contrairement à la vision, la correction est possible même si l'angle de vue est très différent.

4.4.1 Résultats avec le modèle classique de création de carte

Il est aussi possible d'exécuter le MSCKF magnéto-visuel-inertiel sans carte magnétique préalable : la carte peut être construite en ligne à partir des mesures obtenues au cours de la trajectoire. Les étapes du MSCKF sont inchangées avec les Equations 3.23 données dans la Section 3.1.5.

Cependant, les équations de la carte sont les équations classiques mentionnées en Section 3.1.5, c'est à dire sans prise en compte de l'incertitude des positions dans le modèle de processus Gaussien. La création de carte en ligne requiert elle aussi de modifier le bruit du capteur artificiellement pour le modèle. Sinon la trajectoire diverge rapidement, comme montré sur la Figure 4.9 où le système parcourt le même carré dans la même direction plusieurs fois. En augmentant le bruit du capteur, cette fois plus fortement que pour une carte complète, on parvient à obtenir des trajectoires robustes. Avec $\sigma_m = 4.47\mu\mathrm{T}$ (soit le triple du bruit du capteur utilisé pour créer les cartes avec notre VISLAM, la méthode la plus précise ; $\sigma_m^2 = 20\mu\mathrm{T}^2$), le phénomène de divergence



(a) sans magnétomètre (b) $\sigma_m = 0.3\mu\mathrm{T}$ (c) $\sigma_m = 4.47\mu\mathrm{T}$

FIGURE 4.9: Utilisations du MSCKF magnéto-visuel-inertiel avec une caméra et une IMU (avec un modèle de carte classique, sans prise en compte des incertitudes de position). La trajectoire vérité-terrain est en bleu et les trajectoires calculées par le MSCKF en rouge. La deuxième trajectoire dérive rapidement.

est évité et on observe une amélioration de l'estimation de la trajectoire par rapport à un MSCKF uniquement visuel-inertiel. Le décalage progressif de la trajectoire est réduit car la carte magnétique est réutilisée lorsque la trajectoire repasse aux mêmes endroits. La valeur du bruit du modèle est plus forte que pour les cartes construites hors ligne car, expérimentalement, les trajectoires obtenues en ligne sont significativement de moins bonne qualité.

4.4.2 Ajout de l'incertitude de localisation dans le modèle

Si les résultats obtenus grâce au gonflement du bruit sont satisfaisants, des améliorations peuvent être apportées par notre modèle de carte qui prend en compte l'erreur de localisation. Les équations proposées intègrent cette fois-ci le facteur principal de détérioration de la qualité de la carte. La Figure 4.10 représente la même trajectoire qu'auparavant, mais avec le nouveau modèle proposé en Section 3.3.4.

Avec plusieurs expériences, réalisées dans les mêmes conditions que précédemment, nous pouvons observer les changements qu'apporte le nouveau modèle de carte magnétique.

Deux types d'erreur, en plus du type de modèle choisi, peuvent diminuer la qualité de la carte magnétique : les erreurs de position (ou translation) et les erreurs de rotation, comme mentionné dans la Section 3.5.

Le MSCKF magnéto-visuel-inertiel est exécuté avec création de carte en ligne pour toutes les séquences présentées dans cette section : deux d'entre elles consistent à faire des cercles autour d'une table ou de plusieurs chaises (Séquences 1 and 2 dans le Tableau 4.3). Au cours de la troisième séquence, plusieurs couloirs sont parcourus dans les deux directions plusieurs fois.

L'ajout de l'incertitude de position dans le MSCKF magnéto-visuel-inertiel augmente significativement la robustesse de l'algorithme : pour deux des séquences, la trajectoire diverge après une courte période de temps si l'incertitude n'est pas incluse et que $\sigma_m^2 = \sigma_{noise}^2$. Avec la méthode que nous proposons, ce phénomène n'est pas observé et la précision des trajectoires est élevée (moins d'1.10 m pour toutes les Séquences). L'ATE est améliorée de 28% par rapport à la solution grossière consistant à utiliser $\sigma_m^2 = 20\mu\mathrm{T}^2$.

L'hypothèse d'indépendance entre deux valeurs de position consécutives est sans doute fausse car la position précédente a une influence sur le calcul de la position suivante. Cette corrélation peut être intégrée dans le modèle mais demande plus de ressources de calcul. Comme les résultats

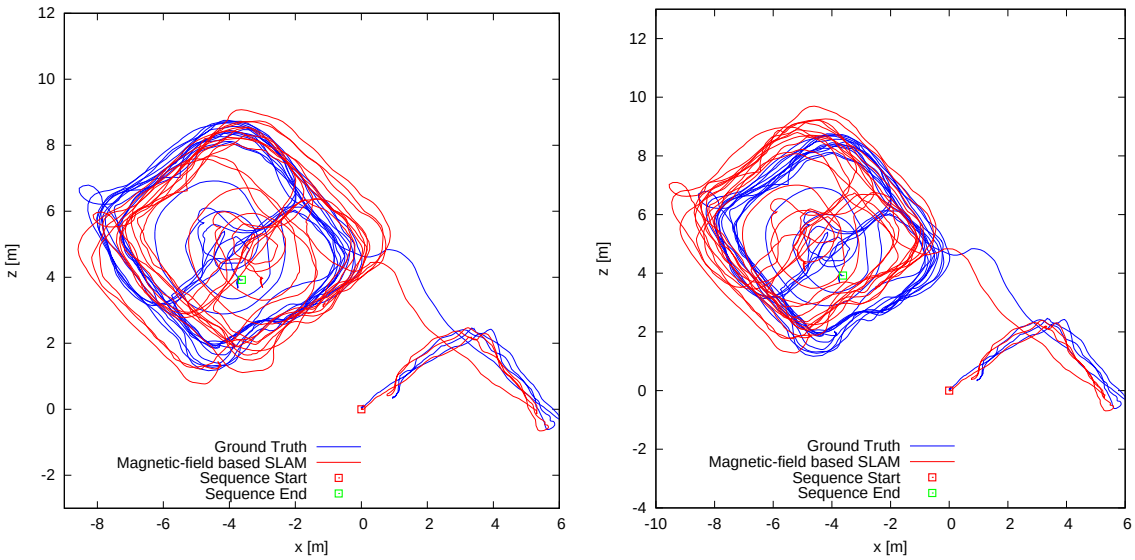
de localisation sont améliorés avec notre modèle, on peut estimer que l'hypothèse permet d'obtenir de bons résultats et est moins forte que celle consistant à multiplier par plus de 10 le bruit du capteur, même s'il serait intéressant de voir si les résultats de localisation peuvent encore être améliorés en intégrant ces corrélations dans le modèle.

Les variances de la rotation obtenue par le MSCKF magnéto-visuel-inertiel sont très faibles, même à la fin de la trajectoire : moins de 0.3° alors que la variance en position est de 6 cm. On peut considérer que l'erreur de rotation a une influence négligeable pour la création de la carte magnétique : en observant les variations spatiales du champ, les variations sur quelques centimètres sont beaucoup plus importantes qu'une petite déviation en orientation de moins d'un degré. Il est utile d'intégrer les erreurs de rotation uniquement pour des séquences longues et/ou difficiles ou pour des algorithmes ayant une localisation moins précise. Les résultats obtenus sur les mêmes séquences, en intégrant l'erreur de rotation en plus de l'erreur de position, sont très similaires : l'ATE est modifiée de moins de 8 cm sur les trois séquences. Généralement l'ajout de l'erreur de rotation permet d'améliorer les résultats mais cette amélioration est peu significative. Cependant, pour d'autres algorithmes où l'erreur sur l'estimation du cap est plus importante, nous pouvons faire l'hypothèse que les différences entre les deux algorithmes seraient plus marquées. C'est une perspective possible pour une application des équations prenant en compte l'erreur de rotation.

Numéro de la séquence		Séquence 1 (5 min 44, 288 m)	Séquence 2 (6 min 26, 346 m)	Séquence 3 (4 min 29, 248 m)
Odométrie visuelle-inertielle	ATE	1.760 m	1.806 m	0.736 m
	Err. déclinaison	0.358°	0.157°	0.691°
	Erreur d'azimut	2.717°	4.630°	2.132°
MSCKF magnéto-visuel-inertiel	ATE			3.709 m
$\sigma_m = 0.3\mu$T,	Err. déclinaison	divergence	divergence	0.325°
sans erreur de position	Erreur d'azimut			**0.373**°
MSCKF magnéto-visuel-inertiel	ATE	1.540 m	1.179 m	0.565 m
$\sigma_m = 4.47\mu$T,	Err. déclinaison	**0.330**°	0.122°	0.200°
sans erreur de position	Erreur d'azimut	**2.674**°	5.974°	2.259°
MSCKF magnéto-visuel-inertiel	ATE	**0.947 m**	1.006 m	0.487 m
$\sigma_m = 0.3\mu$T,	Err. déclinaison	0.390°	**0.105**°	0.544°
avec erreur de position	Erreur d'azimut	2.744°	2.934°	1.252°
MSCKF magnéto-visuel-inertiel	ATE	0.948 m	**0.981 m**	**0.415 m**
$\sigma_m = 0.3\mu$T,	Err. déclinaison	0.392°	**0.105**°	**0.197**°
avec err. de pos. et d'orientation	Erreur d'azimut	2.746°	**2.902**°	1.030°

TABLE 4.3: ATE, erreurs de déclinaison et d'azimut du MSCKF magnéto-visuel-inertiel avec création de carte magnétique en ligne (une caméra) avec différents modèles de mise à jour de la carte magnétique. Les résultats de l'odométrie visuel-inertielle sont indiqués.

Cependant, nous savons que le MSCKF a trop confiance dans ses prédictions. Pour résoudre ce problème, deux possibilités s'offrent à nous : gonfler le bruit du capteur pour la modélisation, mais de façon beaucoup plus légère qu'auparavant, ou bien gonfler l'erreur renvoyée par le MSCKF. Nous avons regardé les évolutions causées sur nos expériences si $\sigma_m = 0.55\mu$T, pour prendre en compte cette trop grande confiance. Cette augmentation du bruit capteur reste très inférieure à



(a) sans incertitude, $\sigma_m = 4.47\mu$T (b) avec incertitude, $\sigma_m = 0.3\mu$T

FIGURE 4.10: Utilisations du MSCKF magnéto-visuel-inertiel avec une caméra et une IMU avec un modèle de carte prenant en compte ou non les incertitudes de position (même séquence que sur la Figure 4.9). La trajectoire vérité-terrain est en bleu, les trajectoires calculées par le MSCKF sont en rouge. On voit que lorsque les incertitudes de position sont incluses dans le modèle, la dérive entre chaque tour est plus faible.

celle qui était nécessaire auparavant pour assurer la convergence de l'algorithme ($\sigma_m = 4.47\mu$T). Les résultats avec $\sigma_m = 0.55\mu$T sont meilleurs qu'avec $\sigma_m = 0.3\mu$T : l'amélioration de l'ATE est de 10.4% en moyenne sur nos trois séquences (les résultats sont indiqués dans le Tableau 4.4). Cette modification a plus de répercussions sur la qualité des résultats que l'insertion de l'erreur de rotation.

Numéro de la séquence		Séquence 1 (5 min 44, 288 m)	Séquence 2 (6 min 26, 346 m)	Séquence 3 (4 min 29, 248 m)
MSCKF magnéto-visuel-inertiel	ATE	0.947 m	1.006 m	0.487 m
$\sigma_m = 0.3\mu$T,	Err. déclinaison	**0.390**°	**0.105**°	0.544°
avec erreur de position	Erreur d'azimut	**2.744**°	2.934°	1.252°
MSCKF magnéto-visuel-inertiel	ATE	**0.913 m**	**0.917 m**	**0.395 m**
$\sigma_m = 0.54\mu$T,	Err. déclinaison	0.503°	**0.105**°	**0.426**°
avec erreur de position	Erreur d'azimut	2.994°	**2.771**°	**0.551**°

TABLE 4.4: ATE, erreurs de déclinaison et d'azimut du MSCKF magnéto-visuel-inertiel avec création de carte magnétique en ligne (une caméra) avec l'insertion de l'erreur de position, avec ou sans gonflement du bruit capteur pour prendre en compte la trop grande confiance du MSCKF dans ses prédictions.

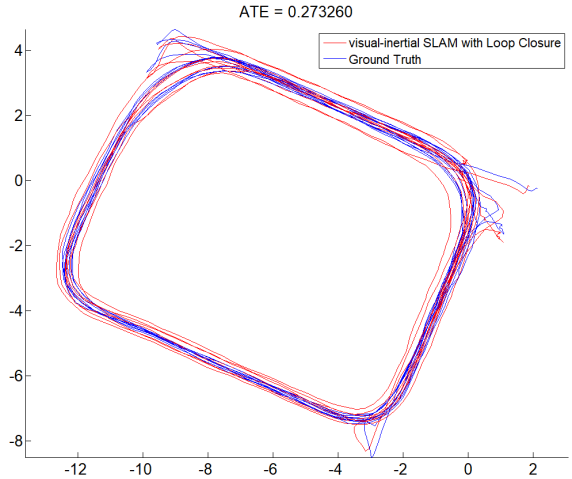
Nous pouvons aussi comparer le MSCKF magnéto-visuel-inertiel avec le modèle de carte amélioré avec l'algorithme VISLAM avec fermeture de boucles et ajustement de faisceaux. Pour

cela, ces algorithmes ont été exécutés sur une nouvelle Séquence similaire à la Séquence 3 et une autre séquence dans laquelle le système fait plusieurs fois le tour de la même zone mais à chaque fois avec des angles de vue différents. Les résultats sont meilleurs qu'avec le VISLAM avec fermeture de boucle pour $\sigma_m = 0.3\mu$T. Nous observons sur les deux Figures 4.11 et 4.12 que notre algorithme permet d'obtenir de meilleurs résultats, réduisant l'ATE de 49% parce que le VISLAM, de son côté, n'est pas capable de réaliser de fermetures de boucle. Cependant, pour les deux premières Séquences du Tableau 4.1, le VISLAM avec fermetures de boucle a les meilleures performances car la caméra est toujours dans la même orientation. Un algorithme de fermetures de boucle magnéto-visuel-inertiel peut être une perspective intéressante pour des travaux futurs, combinant les avantages des deux méthodes.

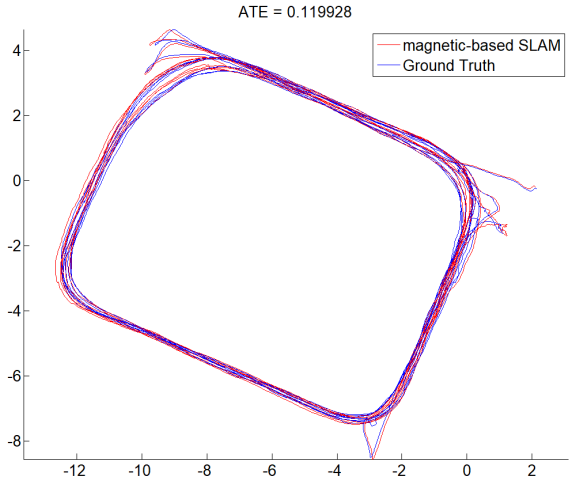
Conclusion

Nous avons vu dans cette partie les résultats obtenus par le MSCKF magnéto-visuel-inertiel. Le magnétomètre permet de corriger efficacement la dérive, dans un lieu déjà exploré ou non. Les cartes magnétiques sont presque inchangées dans le temps (la carte du champ magnétique d'un bâtiment est utilisable pour corriger la dérive même après un an). De plus, nous avons pu observer que le modèle de carte utilisé a une influence directe sur les performances : plus la carte obtenue a des prédictions et des variances proches de la réalité, plus les performances obtenues sont élevées. Dans le cas d'une utilisation en ligne, l'algorithme de localisation étant moins précis, il est d'autant plus important de prendre en compte les incertitudes sur les poses calculées par le système, qui sont utilisées pour créer la carte magnétique.

En ligne et hors ligne, le MSCKF visuel-inertiel permet d'obtenir de meilleurs résultats que les algorithmes VISLAM, en particulier lorsque les cartes visuelles ont été modifiées ou que la relocalisation visuelle est rendue difficile à cause d'angles de vue très différents. Cependant, pour pouvoir utiliser le magnétomètre, il est indispensable qu'il soit correctement calibré : nous avons vu que, dans le cas d'une carte déjà créée, le MSCKF permet d'estimer avec précision le bruit du magnétomètre, si notre position initiale dans la carte est connue. Mais dans le cas d'une utilisation dans un lieu non exploré qui va être cartographié, il est nécessaire de trouver une méthode simple pour calibrer le magnétomètre facilement et avec précision : c'est l'objet du prochain chapitre.



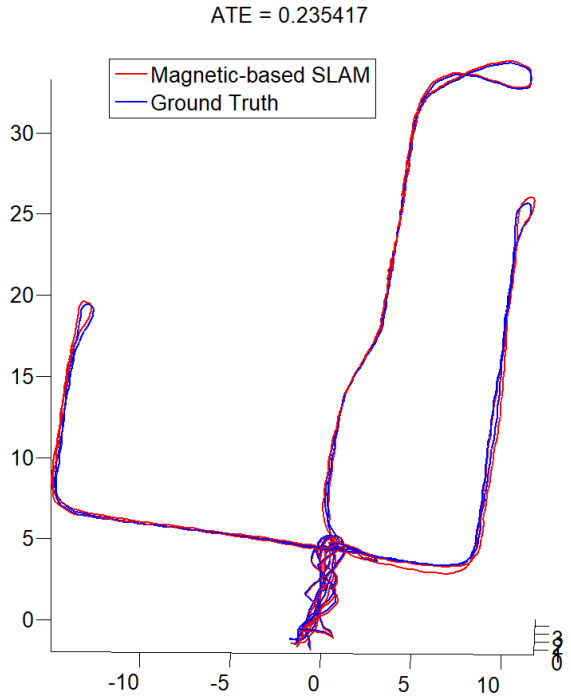
(a) VISLAM avec fermetures de boucle : les fermetures de boucle ne sont pas détectées par l'algorithme à cause de la différence des points de vue.



(b) MSCKF mag avec incertitude, $\sigma_m = 0.55\mu$T. Le magnétomètre permet de corriger la trajectoire car il n'est pas influencé par la différence de point de vue.

FIGURE 4.11: Comparaisons de l'algorithme VISLAM et du MSCKF magnéto-visuel-inertiel pour un système équipé d'une caméra, d'une IMU et d'un magnétomètre pour plusieurs boucles effectuées avec des angles de vue différents. La trajectoire vérité-terrain est représentée en bleu, les trajectoires estimées par les algorithmes de SLAM sont en rouge.

(a) VISLAM avec fermetures de boucle : les fermetures de boucle ne sont pas détectées par l'algorithme à cause de la différence des points de vue



(b) MSCKF mag avec incertitude, $\sigma_m = 0.55\mu$T. Le magnétomètre permet de corriger la trajectoire car il n'est pas influencé par la différence de point de vue.

FIGURE 4.12: Comparaisons de l'algorithme VISLAM et du MSCKF magnéto-visuel-inertiel pour un système équipé d'une caméra, d'une IMU et d'un magnétomètre. Le système parcourt plusieurs couloirs dans les deux sens. La trajectoire vérité-terrain est représentée en bleu, les trajectoires estimées par les algorithmes de SLAM sont en rouge.

Chapitre 5

Calibration en ligne du magnétomètre en intérieur

5.1 Calibration en extérieur et en intérieur : avantages et inconvénients 108
5.1.1 La méthode classique : calibration en extérieur dans un champ magnétique constant 108
5.1.2 Difficultés de la calibration en intérieur 108
5.2 Une nouvelle méthode de calibration en intérieur 109
5.2.1 Description générale 109
5.2.2 Etape de calibration grossière 110
5.2.3 Etape de raffinement : optimisation de la calibration à l'aide de la carte magnétique 113
5.3 Evaluation de la précision de la calibration 113
5.4 Apport d'une calibration précise pour une localisation en ligne basée sur le magnétomètre 115

Introduction

Dans les Chapitres 3 et 4, le magnétomètre est supposé parfaitement calibré. Une calibration précise est indispensable pour la construction de cartes magnétiques précises ou la localisation magnétique avec création de carte en ligne. Pour les expériences mentionnées, la calibration du magnétomètre a été effectuée en extérieur, selon le protocole décrit dans la Section 2.1.4.4. Or la calibration en extérieur est contraignante car elle requiert de transporter le système en extérieur et loin de tout bâtiment (plus de 3 m). De plus, il est nécessaire de réaliser ce protocole régulièrement car le biais évolue au cours du temps : au bout de quelques jours, son évolution n'est plus négligeable. Des méthodes de calibration en intérieur ont été proposées mais leur précision reste significativement inférieure à une calibration en extérieur. Or cet écart a un impact sur la qualité des cartes magnétiques et de la localisation obtenue. Après avoir fait l'inventaire des méthodes existantes (Section 5.1), nous proposons une nouvelle méthode qui prend mieux en compte les perturbations magnétiques et qui est donc mieux adaptée aux environnements intérieurs (Section 5.2). En terme de précision, la calibration obtenue avec notre algorithme est supérieure à celle

obtenue avec les approches existantes de l'état de l'art et très proche de ce que l'on obtient avec une calibration réalisée en extérieur (Section 5.3).

5.1 Calibration en extérieur et en intérieur : avantages et inconvénients

5.1.1 La méthode classique : calibration en extérieur dans un champ magnétique constant

La calibration du magnétomètre revient à estimer son biais et sa matrice d'alignement et d'échelle (le modèle de calibration est celui de l'Equation 2.21, présenté dans la Section 2.1.4.4) : $\hat{\mathbf{m}}_I = \mathbf{A}\mathbf{m}_I + \mathbf{b}_m + \boldsymbol{\eta}_m$). Les méthodes les plus courantes de calibration exigent un champ magnétique constant. Cette condition est vérifiée en extérieur, loin de tout bâtiment ou objet métallique : le champ magnétique mesuré est alors égal au champ magnétique terrestre. Il est facile d'obtenir sa valeur en tout point du globe grâce à des modèles physiques basés sur de nombreuses données, tels que le WMM (World Magnetic Model). C'est pourquoi la calibration du magnétomètre a fait l'objet d'études approfondies en extérieur [FE08, VES+11, TW13, FDD18]. Les algorithmes de calibration en extérieur exigent généralement une rotation du magnétomètre dans le plus grand nombre de directions possibles ([VES+11, TW13, KS16, VES+15]). Les mesures magnétiques non calibrées recueillies forment un ellipsoïde dont le centre est décalé de l'origine (0;0;0). Les valeurs exactes de l'alignement, de l'échelle et du biais du magnétomètre sont estimées en ajustant cet ellipsoïde à une sphère dont le centre est (0;0;0) (voir la Section 2.1.4.4 pour plus de détails).

5.1.2 Difficultés de la calibration en intérieur

En intérieur, la calibration du magnétomètre est plus difficile car le champ n'est pas connu a priori et est soumis à de nombreuses fluctuations spatiales dues aux éléments ferromagnétiques de la structure du bâtiment (piliers en acier, tubes métalliques). La matrice d'alignement et d'échelle $\mathbf{A}$ est une matrice symétrique, avec 6 degrés de liberté. Comme tous les paramètres d'un environnement intérieur sont observables jusqu'à un facteur d'échelle inconnu, un des paramètres de l'échelle doit être fixé. Sa valeur est obtenue une fois pour toutes par une unique calibration en extérieur. Le facteur d'échelle varie très peu par la suite donc il est possible de conserver cette valeur. Par conséquent, il y a un total de 8 paramètres de calibration inconnus pour une calibration en intérieur.

Dans [KS16], des capteurs inertiels sont utilisés pour améliorer la robustesse de la calibration du magnétomètre lorsqu'il est monté sur un système perturbant le champ magnétique. Cette approche peut théoriquement être appliquée à l'intérieur puisqu'elle n'exige pas la norme et la direction du champ magnétique réel pour la calibration. En revanche, elle nécessite un champ magnétique constant, ce qui est rarement le cas à l'intérieur [VHKR11]. Cela limite son utilisation : il est difficile de déterminer quels sont les endroits à faibles variations magnétiques alors que la majorité de ces variations est due aux éléments contenus dans les murs, et donc invisibles. Il est difficile de maintenir toutes les mesures de calibration sur un petit volume de quelques cm^3, car les rotations du système sur lequel il a été monté provoquent habituellement des changements de position du capteur.

Comme nous l'avons montré en Section 4.2, notre algorithme MSCKF magnéto-visuel-inertiel est capable d'estimer le biais du magnétomètre en même temps que sa localisation à partir d'une carte magnétique préexistante (voir Section 4.3.3 pour les résultats expérimentaux). De même, un

filtre particulaire pour la localisation et la calibration est présenté dans [SSH21]. Il estime cette fois-ci les neuf paramètres de la matrice **A**, y compris le facteur d'échelle. Cette manière de calibrer un magnétomètre dépend directement de la carte magnétique déjà obtenue et n'est donc pas adaptée à l'exploration d'un environnement inconnu. De plus, le magnétomètre doit être recalibré avant chaque nouvelle cartographie magnétique. [SSH21] propose une solution consistant à construire une carte magnétique sans calibration précise du magnétomètre pour éviter de devoir le recalibrer dans chaque nouvel environnement non cartographié. Pour cela, les auteurs supposent qu'à chaque position, le champ magnétique réel en chaque point est unique (il n'y a pas d'ambigüités possibles). Dans leur cadre applicatif, où un train miniature se déplace le long d'un rail dans la même direction, le faible nombre de degrés de liberté permet de vérifier cette condition. Mais cette hypothèse est fausse pour les mesures magnétiques acquises au même endroit avec des orientations différentes lorsque la calibration du magnétomètre est inexacte.

Au lieu d'utiliser une carte magnétique pré-construite, d'autres approches proposent de calibrer le magnétomètre en estimant simultanément les variations du champ [MMLC19]. L'idée principale consiste en une estimation simultanée des paramètres de calibration du magnétomètre et d'un modèle de champ magnétique au moyen d'un algorithme d'optimisation non linéaire. Des splines sont utilisées pour modéliser la variation du champ magnétique. Il en résulte une calibration peu précise puisque les propriétés physiques du champ magnétique ne sont pas prises en compte. De plus, pour estimer le déplacement du magnétomètre dans l'espace au cours du processus de calibration, ils utilisent des outils de capture de mouvement, non mobiles. Cela rend cette solution difficile à déployer.

5.2 Une nouvelle méthode de calibration en intérieur

Notre algorithme de calibration du magnétomètre est décrit dans cette Section. Après un bref aperçu de sa structure globale, présenté dans la Section 5.2.1, les deux étapes de notre processus de calibration sont détaillées dans les Sections 5.2.2 et 5.2.3.

5.2.1 Description générale

Notre algorithme de calibration est un processus en deux étapes prenant comme entrée des données magnétiques avec leurs poses associées. Elles sont obtenues à partir de l'algorithme VISLAM en associant le magnétomètre avec un capteur IMU et des caméras. Le système est peu coûteux et transportable facilement d'un endroit à l'autre.

La première étape de notre processus de calibration est une calibration grossière. On suppose que le champ magnétique est localement constant et on vérifie que les données recueillies sont suffisamment variées en termes de poses pour contraindre la calibration et assurer l'unicité de la solution. Cette étape de calibration est basée sur le fait que deux mesures de champ magnétique acquises au même endroit physiquement doivent mesurer le même champ. En raison de l'imperfection des mesures de la position, de l'orientation du magnétomètre et aussi de la calibration, ce n'est pas le cas en pratique. Nous montrons dans la Section 5.2.2 que les erreurs de calibration sont observables à partir de paires de mesures magnétiques prises au même endroit avec des orientations différentes. En pratique, ces paires sont constituées de données acquises à proximité les unes des autres (de l'ordre de quelques centimètres), le champ magnétique étant supposé très constant localement. L'acquisition de mesures magnétiques est arrêtée lorsqu'un nombre suffisant de paires garantissant une calibration précise est atteint, comme décrit dans la Section 5.2.2.3.

La seconde étape affine la calibration obtenue lors de la première étape en levant l'hypothèse

selon laquelle le champ magnétique est localement constant. Pour cela, les variations du champ magnétique modélisées par un procédé gaussien (Section 3.1.3) sont estimées simultanément avec les paramètres de calibration du magnétomètre par un algorithme d'optimisation non linéaire. Nous montrons dans la Section 5.2.3 que ce problème d'optimisation peut être reformulé pour ne dépendre que des paramètres de calibration.

Un aperçu de l'algorithme de calibration proposé est schématisé par la Figure 5.1.

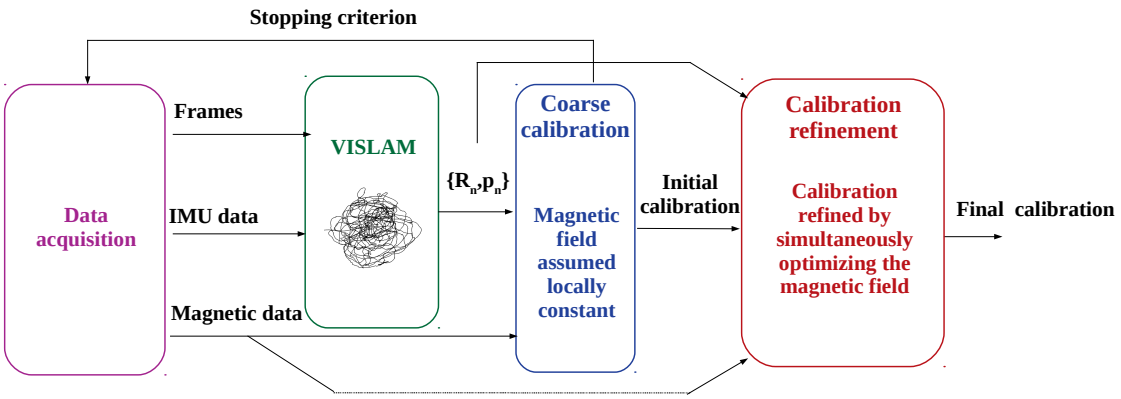


FIGURE 5.1: Schéma global de l'algorithme de calibration proposé.

5.2.2 Etape de calibration grossière

Pour évaluer les paramètres intrinsèques du magnétomètre, nous proposons de comparer des paires de mesures magnétiques obtenues à des positions rapprochées dans l'espace et avec différentes orientations du capteur. Ces paires sont formées à partir d'un algorithme des k-plus proches voisins (kNN) en sélectionnant, parmi les k candidats, le candidat dont la différence de rotation est la plus optimale avec le point étudié, c'est à dire permettant la meilleure observabilité des paramètres de calibration.

L'algorithme proposé pour l'estimation grossière de la calibration du magnétomètre alterne l'estimation du biais (la matrice d'alignement et d'échelle est fixée) et l'estimation de la matrice d'alignement et d'échelle (le biais est fixé) jusqu'à convergence. La valeur initiale de $\mathbf{A}$ est choisie égale à la matrice identité, I_3, multipliée par le facteur d'échelle, fixé préalablement à la valeur trouvée lors de la dernière calibration en extérieur.

5.2.2.1 Estimation du biais

Soit une paire de mesures magnétiques $(\hat{\mathbf{m}}_{B_1}, \hat{\mathbf{m}}_{B_2})$ acquises presque au même endroit, $\mathrm{R}_{B_1B_2} \in SO(3)$ est la matrice de rotation obtenue à partir des orientations du magnétomètre R_{B_1G} et R_{B_2G}. $\mathbf{m}_{B_1}$ et $\mathbf{m}_{B_2}$ sont les valeurs réelles du champ magnétique qui auraient été mesurées par un capteur idéal. La relation entre ces deux valeurs, en supposant que le champ magnétique est localement constant, est donnée par :

$$\mathbf{m}_{G_1} = \mathbf{m}_{G_2} \Rightarrow \mathbf{m}_{B_1} = \mathrm{R}_{B_1B_2}\mathbf{m}_{B_2} \tag{5.1}$$

En utilisant l'Equation 2.21 cette égalité devient :

$$\mathbf{A}^{-1}(\hat{\mathbf{m}}_{B_1} - \mathbf{b}_m - \boldsymbol{\eta}_1) = \mathrm{R}_{B_1B_2}\mathbf{A}^{-1}(\hat{\mathbf{m}}_{B_2} - \mathbf{b}_m - \boldsymbol{\eta}_2) \tag{5.2}$$

Nous négligeons l'impact des bruits η_1 et η_2. Comme $\mathrm{R}_{B_1B_2}\mathbf{A}^{-1}$ est proche d'une matrice de rotation, les deux composantes de bruit de l'Equation 5.2 peuvent être considérées comme égales. L'Equation 5.2 est alors simplifiée comme suit et forme un système linéaire :

$$\begin{aligned}(\mathrm{R}_{B_1B_2} - I_3)\mathbf{A}^{-1}\mathbf{b}_m &= \mathrm{R}_{B_1B_2}\mathbf{A}^{-1}\hat{\mathbf{m}}_{B_2} - \mathbf{A}^{-1}\hat{\mathbf{m}}_{B_1} \\ \mathbf{E}\mathbf{b}_m &= \mathbf{F}.\end{aligned} \tag{5.3}$$

En concaténant les Equations pour toutes les paires de données magnétiques recueillies et en supposant que $\mathbf{A}$ est connu, on obtient le biais du magnétomètre en résolvant le système.

On peut observer que si les deux mesures magnétiques sont acquises dans des directions opposées, le côté gauche de l'Equation 5.3 est égal à $-2\mathbf{A}^{-1}\mathbf{b}_m$, alors que si la rotation est faible, elle est proche de 0. Dans ce dernier cas, l'hypothèse selon laquelle le bruit peut être négligé n'est pas vérifiée.

Une pondération est alors appliquée à l'Equation 5.3 : elle est de la forme $\exp(-\frac{(\theta_b^{opt} - \theta_{\mathrm{R}_{B_1B_2}})^2}{\Theta^2})$ avec θ_b^{opt} l'angle permettant la meilleure observabilité du biais. $\theta_b^{opt} = \pi$ car c'est l'angle entre deux directions opposées. $\theta_{\mathrm{R}_{B_1B_2}}$ est la valeur absolue de l'angle de rotation, entre 0 et π :

$$\theta_{\mathrm{R}_{B_1B_2}} = \arccos\left(\frac{(tr(\mathrm{R}_{B_1B_2}) - 1)}{2}\right) \tag{5.4}$$

avec $tr(\mathrm{R}_{B_1B_2})$ la trace de la matrice. Θ est un paramètre qui définit le degré d'influence d'une Equation en fonction de l'angle entre deux données magnétiques. Cela permet de donner des poids plus forts aux équations ayant une très bonne observabilité du biais.

5.2.2.2 Estimation de la matrice d'alignement et d'échelle

L'estimation de la matrice d'alignement et d'échelle peut être divisée en deux parties : l'estimation de son échelle (c'est-à-dire sa diagonale) d'abord et les coefficients d'alignement. $\mathbf{A}$ est habituellement proche de la matrice d'identité : les coefficients d'échelle sont proches de 1 et l'alignement devrait être proche de 0 pour un capteur correctement fabriqué (axes parfaitement alignés). Les trois paramètres diagonaux sont représentés par $1+s_1$, $1+s_2$ et $1+s_3$ et les paramètres d'alignement par t_1, t_2 et t_3 tels que :

$$\mathbf{A} = \begin{bmatrix} 1+s_1 & t_1 & t_2 \\ t_1 & 1+s_2 & t_3 \\ t_2 & t_3 & 1+s_3 \end{bmatrix} \tag{5.5}$$

avec $s_1, s_2, s_3, t_1, t_2, t_3 \ll 1$. Dans la suite, s_3 est fixé, comme mentionné dans la Section 5.1.2.

On peut écrire $\mathbf{A} = I_3 + \lfloor \mathbf{S} \rfloor_D + \lfloor \mathbf{T} \rfloor_S$ avec $\mathbf{S} = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix}$ et $\mathbf{T} = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$. Les opérateurs $\lfloor \cdot \rfloor_D$ et $\lfloor \cdot \rfloor_S$ sont définis comme :

$\lfloor \cdot \rfloor_D : \mathbb{R}^3 \to \mathbb{R}^{3\times 3}$ et $\lfloor \cdot \rfloor_S : \mathbb{R}^3 \to \mathbb{R}^{3\times 3}$.

$$\begin{aligned} &\lfloor \cdot \rfloor_D : \mathbb{R}^3 \to \mathbb{R}^{3\times 3} \\ &\left\lfloor \begin{bmatrix} a \\ b \\ c \end{bmatrix} \right\rfloor_D = \begin{bmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{bmatrix} \end{aligned} \tag{5.6}$$

et

$$\lfloor \cdot \rfloor_S : \mathbb{R}^3 \to \mathbb{R}^{3\times 3}$$
$$\left\lfloor \begin{bmatrix} a \\ b \\ c \end{bmatrix} \right\rfloor_S = \begin{bmatrix} 0 & a & b \\ a & 0 & c \\ b & c & 0 \end{bmatrix}. \tag{5.7}$$

Pour estimer les paramètres de la matrice **A**, un algorithme alternant l'estimation de **S** et de **T** est utilisé. Ainsi, en supposant que le biais et les coefficients d'alignement sont connus et en appliquant la formule de Taylor pour **A** pour l'Equation 5.2, les paramètres d'échelle **S** sont estimés à partir de :

$$(\Gamma - \lfloor \mathbf{S} \rfloor_D)(\hat{\mathbf{m}}_{B_1} - \mathbf{b}_m) = \mathrm{R}_{B_1B_2}(\Gamma - \lfloor \mathbf{S} \rfloor_D)(\hat{\mathbf{m}}_{B_2} - \mathbf{b}_m) + o(\| \mathbf{S} \|) + o(\| \mathbf{T} \|). \tag{5.8}$$

avec Γ égal à $I_3 - \lfloor \mathbf{T} \rfloor_S$.

Comme $\lfloor \mathbf{S} \rfloor_D \mathbf{K} = \lfloor \mathbf{K} \rfloor_D \mathbf{S}$ pour tout vecteur **K** de taille 3 et que $o(\| \mathbf{S} \|)$ et $o(\| \mathbf{T} \|)$ sont négligeables, l'Equation 5.8 est simplifiée comme suit :

$$(\mathrm{R}_{B_1B_2} \lfloor (\hat{\mathbf{m}}_{B_2} - \mathbf{b}_m) \rfloor_D - \lfloor (\hat{\mathbf{m}}_{B_1} - \mathbf{b}_m) \rfloor_D)\mathbf{S} = \mathrm{R}_{B_1B_2}\Gamma(\hat{\mathbf{m}}_{B_2} - \mathbf{b}_m) - \Gamma(\hat{\mathbf{m}}_{B_1} - \mathbf{b}_m) \tag{5.9}$$

Comme pour le biais, **S** peut être estimé en résolvant un système linéaire obtenu en concaténant l'Equation 5.9 pour toutes les paires de données magnétiques recueillies. L'angle de rotation entre deux mesures magnétiques qui permet d'estimer de façon optimale l'échelle est $\theta_s^{opt} = \frac{\pi}{2}$. En effet, pour cette valeur angulaire (en supposant que le biais et l'alignement sont parfaitement connus), le rapport entre la norme des deux mesures magnétiques est approximativement égal au rapport des échelles. Comme pour l'estimation du biais, l'Equation 5.9 est pondérée par $\exp\left(-\frac{(\theta_s^{opt} - \theta_{\mathrm{R}_{B_1B_2}})^2}{\Theta^2}\right)$.

En suivant la même procédure que pour les paramètres de l'échelle, les coefficients d'alignement **T** sont obtenus par :

$$(\mathrm{R}_{B_1B_2} \lfloor (\hat{\mathbf{m}}_{B_2} - \mathbf{b}_m) \rfloor_S - \lfloor (\hat{\mathbf{m}}_{B_1} - \mathbf{b}_m) \rfloor_S)\mathbf{T} \simeq \mathrm{R}_{B_1B_2}\Upsilon(\hat{\mathbf{m}}_{B_2} - \mathbf{b}_m) - \Upsilon(\hat{\mathbf{m}}_{B_1} - \mathbf{b}_m) \tag{5.10}$$

Comme l'alignement correspond à l'angle entre l'orientation des axes du capteur par rapport au référentiel B_i, il s'apparente à un biais supplémentaire qui varie en fonction du champ magnétique mesuré. Son observabilité est donc maximale lorsque $\theta_t^{opt} = \pi$. Comme pour l'estimation du biais, l'Equation 5.10 est pondérée par $\exp\left(-\frac{(\theta_t^{opt} - \theta_{\mathrm{R}_{B_1B_2}})^2}{\Theta^2}\right)$.

5.2.2.3 Critère d'arrêt

Pour garantir une estimation précise des paramètres de calibration, il doit y avoir suffisamment de paires de mesures magnétiques pour lesquelles la rotation associée $R_{B_1B_2}$ forme un angle proche de l'angle d'observabilité optimal du biais et de la matrice d'alignement et d'échelle.

Il est nécessaire de définir un critère pour arrêter l'acquisition de données magnétiques pour la calibration. Il est basé sur le nombre d'équations à forte observabilité des paramètres de calibration. Une paire de mesures magnétiques satisfait ce critère si :

$$abs(\theta_{\mathrm{R}_{B_1B_2}} - \theta_j^{opt}) < \delta_j \tag{5.11}$$

avec δ_j une valeur seuil. j est égal à b, s ou t (pour le biais, l'échelle ou l'alignement).

Plus δ_j est petit, plus le bruit est faible dans les équations sélectionnées. Cependant, il est difficile d'obtenir deux valeurs à des endroits proches avec la différence d'angle parfaite. Etant donné qu'un nombre relativement important d'équations est nécessaire pour obtenir une estimation précise, le seuil de tolérance devrait être assez élevé.

5.2.3 Etape de raffinement : optimisation de la calibration à l'aide de la carte magnétique

La seconde étape de notre processus de calibration affine la calibration grossière obtenue lors de la première étape en estimant en parallèle la carte locale du champ magnétique. La modélisation utilisée est la méthode modélisant le champ comme le gradient d'un unique potentiel scalaire, décrite dans la Section 3.1.3, sans insertion de l'incertitude de localisation.

La fonction de coût à minimiser correspond à la somme des erreurs entre les mesures magnétiques recueillies et le vecteur de prédiction $\mathbb{E}[B(\mathbf{p})]$:

$$c = \| \mathbb{E}[B(\mathbf{p})] - \mathrm{vec}\Big(R_{GM}(\mathbf{A}^{-1}(\hat{\mathbf{m}}_\mathbf{n} - \mathbf{b}_m))\Big) \|^2 \tag{5.12}$$

En écrivant l'expression complète de l'espérance obtenue par notre modèle (Equation 3.20), nous pouvons simplifier l'expression, sachant que $\nabla\Phi = \nabla\Phi_*$ puisque les prédictions du modèle sont calculées aux mêmes positions que celles des données collectées :

$$c = \| (\nabla\Phi([\nabla\Phi]^\top \nabla\Phi + \sigma_m^2 \Lambda^{-1})^{-1}[\nabla\Phi]^\top - I_{3D})\mathrm{vec}\Big(R_{GM}(\mathbf{A}^{-1}(\hat{\mathbf{m}}_\mathbf{n} - \mathbf{b}_m))\Big) \|^2 \tag{5.13}$$

En introduisant la variable Ω, nous pouvons obtenir une expression plus synthétique du calcul :

$$\begin{aligned} c &= \| (\nabla\Phi\Omega - I_{3D})\mathrm{vec}(R_{GM}(\mathbf{A}^{-1}(\hat{\mathbf{m}}_\mathbf{n} - \mathbf{b}_m))) \|^2 \\ \text{avec } \Omega &= ([\nabla\Phi]^\top \nabla\Phi + \sigma_m^2 \Lambda^{-1})^{-1}[\nabla\Phi]^\top. \end{aligned} \tag{5.14}$$

$\nabla\Phi\Omega - I_{3D}$ est constant et peut être précalculé une fois pour toutes. Cette Equation ne dépend que des paramètres de calibration $s_1, s_2, t_1, t_2, t_3, b_m$. s_3 est fixé et est égal à la valeur obtenue à l'extérieur, puisqu'à l'intérieur, la matrice d'échelle est observable jusqu'à un facteur d'échelle inconnu. S'il est impossible d'aller en extérieur, cette valeur est fixée à 1 car la matrice d'échelle est proche de la matrice identité. L'Equation 5.14 est optimisée par l'algorithme de Levenberg-Marquadt. Il n'y a aucune garantie que cet algorithme ne converge pas vers des minima locaux. Un des avantages de notre étape d'initialisation est qu'elle permet de commencer l'optimisation assez proche de la valeur cherchée (à 2 ou 3 μT près), diminuant la probabilité que l'algorithme converge vers un minimum local, différent de celui de la solution réelle.

5.3 Evaluation de la précision de la calibration

Les Séquences d'acquisition ont été obtenues avec le même casque que celui décrit en Section 2.1. Pour évaluer la calibration proposée sur un ensemble de données représentatif, nous utilisons 4 Séquences différentes. Pour les Séquences 1 et 2, le magnétomètre est tourné dans tous les sens dans une zone aussi petite que possible (c'est-à-dire avec le moins de mouvements de translation possibles). Pour ces deux Séquences, les variations du champ magnétique sont faibles. Pour les Séquences 3 et 4, le casque est tourné dans une zone plus large, de taille 5 m $\times$ 5 m $\times$ 2 m. La durée de toutes ces séquences est limitée à 30 s, afin de prouver que la calibration du magnétomètre peut

être réalisée en ligne dans un délai raisonnable lors d'une phase d'initialisation d'un algorithme SLAM basé sur le champ magnétique (comme le MSCKF magnéto-visuel-inertiel présenté). La qualité du biais estimé et des matrices d'alignement et d'échelle est évaluée par :

$$\begin{cases} err_{\mathbf{b}} = \| \mathbf{b} - \mathbf{b}_{ext} \| \\ err_{\mathbf{A}} = \left[\sum_{i=1}^{3} \sum_{j=1}^{3} (\mathbf{A}\mathbf{A}_{ext}^{-1} - I_3)_{i,j}^2 \right]^{\frac{1}{2}} \end{cases} \tag{5.15}$$

où $\mathbf{A}_{ext}$ et $\mathbf{b}_{ext}$ sont les matrices de calibration et le biais vérité-terrain obtenus à partir d'une calibration en extérieur [FE08] effectuée juste avant chaque séquence. Pour le réglage des paramètres de notre méthode, nous avons choisi $\Theta = 1 \simeq \frac{\pi}{3}$ radian car les poids doivent être faibles lorsque les valeurs sont loin de la différence d'angle optimale, tout en restant suffisamment élevés pour les paires relativement proches de cette différence optimale. Néanmoins, cette valeur a été obtenue expérimentalement et peut être modifiée en fonction des données d'acquisition. Les paramètres pour le critère d'arrêt sont $\delta_b = \delta_t = \frac{\pi}{2}$ et $\delta_s = \frac{\pi}{4}$.

Afin de s'assurer que notre méthode puisse être utilisée dans différents environnements intérieurs, les zones où les calibrations ont été effectuées ont été remplies de différents objets. La Figure 5.2 est une photo de l'un d'eux. Le champ magnétique vérité-terrain est obtenu en appliquant aux mesures brutes la calibration vérité-terrain (calibration en extérieur) et en leur appliquant à chacune une rotation spécifique pour les obtenir dans le même référentiel. Le bruit du capteur est négligé : son amplitude est d'environ 0,3 μT pour le capteur utilisé dans les expériences. La prise en compte du bruit du capteur ne changerait pas le profil général de la carte. Le champ magnétique est spatialement très contrasté, surtout selon l'axe y : sur une distance de 10 cm, on observe une différence de 15 μT. Selon [LGDR12], l'effet sur le champ magnétique des petits objets métalliques, comme les montres ou les ordinateurs, diminue fortement avec la distance. La portée est d'environ 30 cm pour un ordinateur portable. Un ascenseur, au contraire, a une portée de perturbation de quelques mètres. Par conséquent, dans la plupart des environnements intérieurs, les principales perturbations du champ magnétique sont causées par la structure métallique du bâtiment : les objets de la pièce ont une influence très limitée sur la calibration, tant que le système n'est pas très proche d'eux. Néanmoins, notre méthode perdrait en précision si la calibration est proche d'un ascenseur qui se déplace entre plusieurs étages car les variations temporelles du champ ne sont plus négligeables.

Selon [KS16], le champ magnétique local pour de petits volumes peut être considéré comme constant. Dans nos expériences, cette approximation n'est vraie qu'avec la première séquence pour laquelle un algorithme d'ajustement d'ellipsoïde avec contrainte d'échelle a fonctionné relativement bien ($err_{\mathbf{b}} = 0.795\mu$T et $err_{\mathbf{A}} = 2.733.10^{-2}$). Cependant, cette hypothèse d'un champ magnétique constant n'est pas vérifiée pour toutes les autres séquences. Comme l'illustre la Figure 5.3, les données magnétiques recueillies pour la Séquence 2 forment approximativement un demi-ellipsoïde, mais de nombreuses valeurs aberrantes sont présentes, ce qui donne une mauvaise estimation de la calibration. Le biais estimé sur cette séquence avec [FE08] est aberrant ($err_{\mathbf{b}} > 100\mu$T).

C'est pourquoi nous comparons uniquement la méthode de calibration proposée avec notre implémentation de la méthode la plus récente de calibration en intérieur [MMLC19], qui optimise simultanément les paramètres de calibration et de la carte magnétique, modélisée à l'aide de fonctions splines. Pour cette dernière, nous avons utilisé 27 points de contrôle pour le modèle spline sur chacune des quatre séquences, car les zones explorées sont toutes relativement petites. Pour notre méthode, 27 fonctions propres sont utilisées pour la modélisation par processus gaussiens.

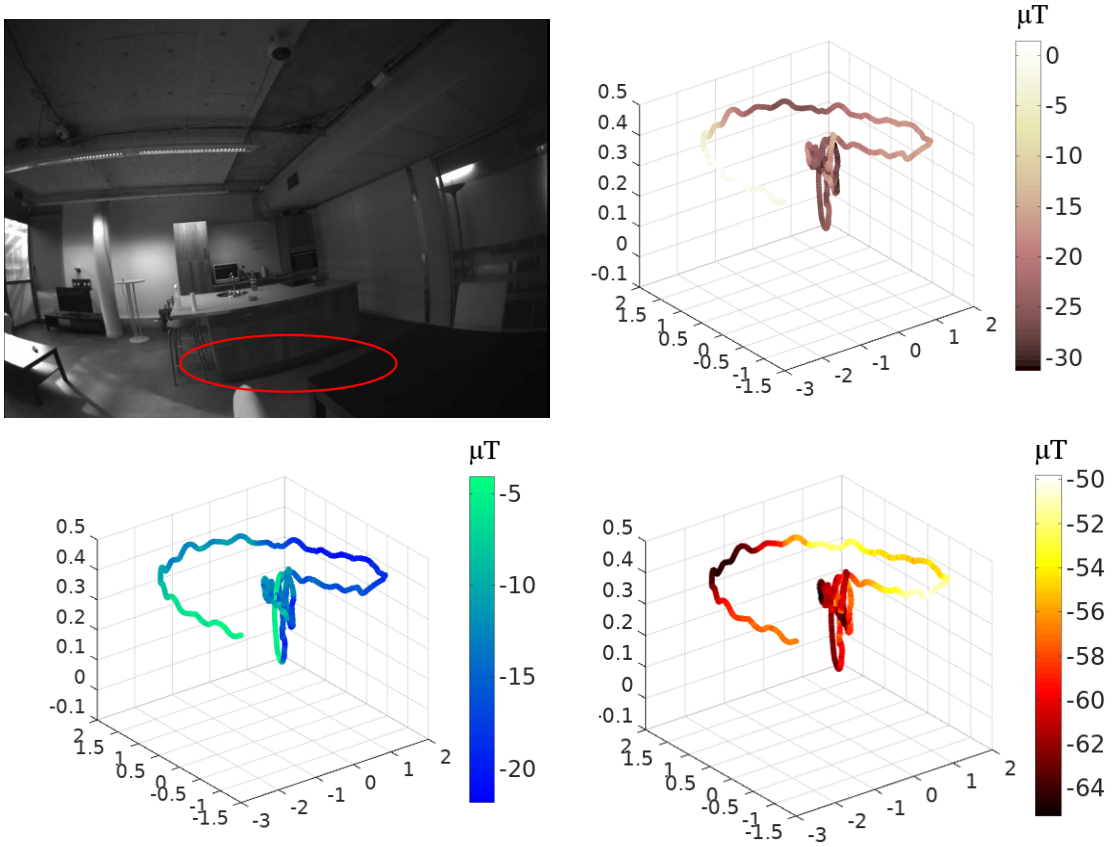


FIGURE 5.2: Zone étudiée correspondant à la Séquence 3 dans le Tableau 5.1 et le champ magnétique vérité-terrain le long de la trajectoire de calibration utilisée. Le champ magnétique est représenté selon chacune de ses trois composantes, x, y et z.

Le Tableau 5.1 présente la comparaison de notre processus de calibration du magnétomètre avec l'algorithme de calibration de l'état de l'art [MMLC19] sur les quatre séquences décrites ci-dessus. Notre méthode est plus précise : sur les quatre séquences, l'amélioration de l'estimation du biais est de 46% en moyenne et de 36% pour la matrice de d'alignement et d'échelle. En particulier pour la Séquence 3, l'estimation du biais de [MMLC19] manque de précision, avec une erreur de 3.657μT. Avec l'approche proposée, l'erreur du biais estimé est inférieure à 1μT. Le temps d'exécution de notre méthode de calibration, pour 30 s d'acquisition de données magnétiques à 50 Hz, est d'environ 600 ms pour notre implémentation en C++. Notre méthode est donc bien adaptée pour calibrer en ligne un magnétomètre pour des algorithmes de type SLAM, utilisant le magnétomètre.

5.4 Apport d'une calibration précise pour une localisation en ligne basée sur le magnétomètre

Dans cette section, nous évaluons l'impact de la calibration du magnétomètre sur la précision d'un algorithme SLAM basé sur le champ magnétique : celui présenté dans le Chapitre 4. Pour cela, différents essais sont effectués sur des séquences d'évaluation avec plusieurs calibrages : une

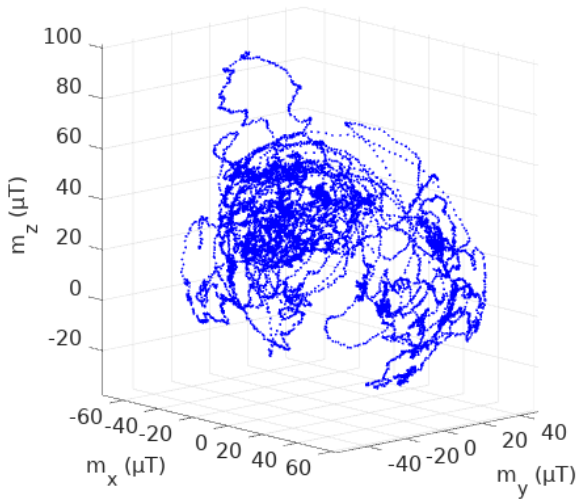


FIGURE 5.3: Données magnétiques obtenues durant la Séquence 2.

		Séquence 1	Séquence 2	Séquence 3	Séquence 4
Etat de l'art [MMLC19]	$err_{\mathbf{b}}$	0.674 μT	1.040 μT	3.657 μT	0.945 μT
	$err_{\mathbf{A}}$	$6.819.10^{-2}$	$4.363.10^{-2}$	$\mathbf{3.689.10^{-2}}$	$\mathbf{6.669.10^{-2}}$
Notre méthode Section 5.2	$err_{\mathbf{b}}$	**0.384** μT	**0.673** μT	**0.862** μT	**0.647** μT
	$err_{\mathbf{A}}$	$\mathbf{1.041.10^{-2}}$	$\mathbf{2.050.10^{-2}}$	$6.090.10^{-2}$	$9.095.10^{-2}$

TABLE 5.1: Comparaison de la précision de la calibration obtenu avec différentes méthodes.

calibration magnétique obtenue par notre approche, par celle de [MMLC19] et une calibration en extérieur. Les résultats sont obtenus avec une seule caméra, une IMU et un magnétomètre. Les deux séquences utilisées pour l'évaluation sont composées de deux parties. Pendant les trente premières secondes, le système d'acquisition est déplacé et pivoté dans tous les sens. La localisation est obtenue par odométrie visuelle-inertielle et le magnétomètre calibré selon l'une des méthodes décrites ci-dessus (à l'exception de la calibration en extérieur). Ensuite, une trajectoire est réalisée en marchant dans plusieurs couloirs dans les deux sens. L'algorithme MSCKF magnéto-visuel-inertiel (Chapitre 4) est utilisé pour la localisation et la cartographie du champ magnétique en ligne. Pour les exécutions utilisant la calibration en extérieur, la modélisation du champ magnétique commence également après les 30 premières secondes d'acquisition pour que la comparaison des trajectoires estimées ne soit pas biaisée. La Figure 5.5 représente les trajectoires SLAM obtenues avec les différentes calibrations du magnétomètre pour l'une des deux séquences. L'environnement exploré lors de la phase de calibration de la première Séquence du tableau 4.1 est un "coin café" à proximité de bureaux, comme le montre la Figure 5.4. L'initialisation de la Séquence 2 a été obtenue dans la même zone que la Séquence 3 de la table 5.1.

Les différentes exécutions sont comparées en termes d'erreurs de position et de rotation. La vérité-terrain est obtenue par l'algorithme VISLAM utilisant quatre caméras, avec fermeture de boucle et ajustement de faisceaux global en post-traitement (voir la Section 2.3 dans laquelle

FIGURE 5.4: Zone où la calibration est effectuée pour la Séquence 1 dans le Tableau 4.1.

Numéro de la Séquence		Séquence 1 (5 min 44, 309 m)	Séquence 2 (5 min 43, 251 m)
Odométrie visuelle-inertielle	ATE	0.905 m	0.326 m
	Erreur de déclinaison	0.758°	0.234°
	Erreur d'azimut	0.862°	1.494°
MSCKF magnéto-visuel-inertiel Calibration [MMLC19]	ATE	0.667 m	0.342 m
	Erreur de déclinaison	0.740°	0.220°
	Erreur d'azimut	0.747°	2.166°
MSCKF magnéto-visuel-inertiel Notre calibration	ATE	0.371 m	**0.181 m**
	Erreur de déclinaison	0.103°	**0.085°**
	Erreur d'azimut	**0.363°**	1.499°
MSCKF magnéto-visuel-inertiel Calibration en extérieur [FE08]	ATE	**0.352 m**	0.182 m
	Erreur de déclinaison	**0.068°**	0.091°
	Erreur d'azimut	0.449°	**1.421°**
MSCKF magnéto-visuel-inertiel Ancienne calibration en extérieur (5 mois auparavant)	ATE	0.640 m	0.708 m
	Erreur de déclinaison	0.644°	0.323°
	Erreur d'azimut	1.116°	5.210°

TABLE 5.2: Erreurs de position (ATE), de déclinaison et d'azimut du MSCKF magnéto-visuel-inertiel en ligne, pour différentes calibrations magnétiques. Les résultats d'une odométrie visuelle-inertielle sont aussi indiqués.

l'algorithme est décrit).

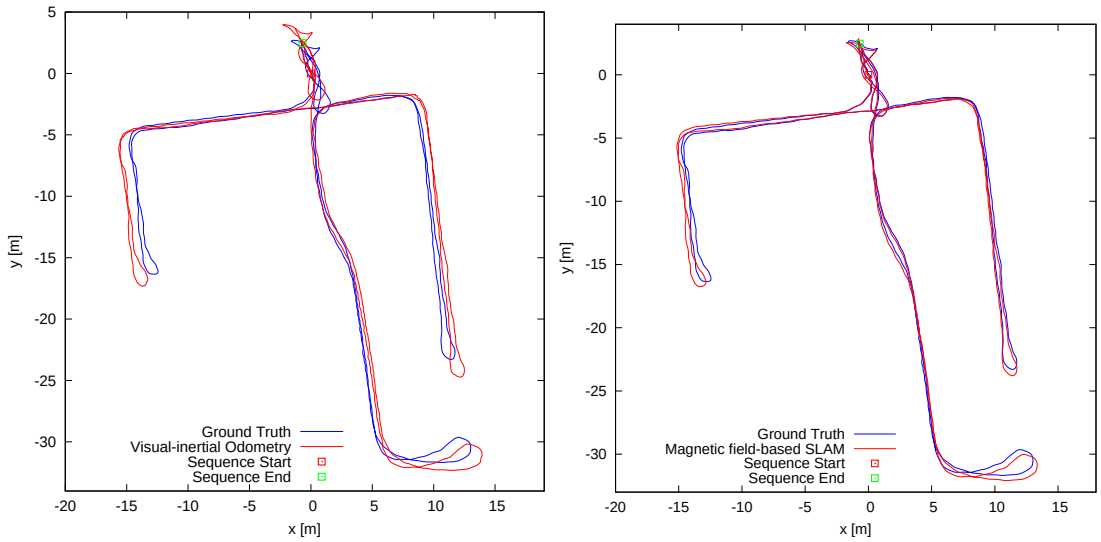
Le Tableau 5.2 présente les erreurs de localisation pour les différentes exécutions. La précision de localisation du MSCKF magnéto-visuel-inertiel en ligne est meilleure que l'odométrie visuelle-inertielle sur toutes les séquences lorsqu'une calibration appropriée est utilisée. L'ATE, en moyenne, lorsque notre calibration ou une calibration en extérieur est utilisée, est 52% plus faible. L'erreur de déclinaison moyenne, par rapport à l'odométrie visuelle-inertielle (VIO), est

inférieure de 77%. Dans toutes ces expériences, nous constatons que notre algorithme a toujours convergé vers le bon calibrage, car le magnétomètre a été déplacé dans de nombreuses directions différentes (voir Section 5.2.2.3). L'imperfection des résultats de calibration est principalement due à une exploration insuffisante des différents angles de rotation du magnétomètre.

Cependant, avec un calibrage moins bon du magnétomètre, comme celui obtenu avec [MMLC19] ou un calibrage obsolète, le gain de précision de localisation par rapport à un algorithme VIO est plus faible et parfois les résultats sont même moins précis que sans le magnétomètre. Par exemple, pour la seconde séquence, l'ATE augmente de 117 % lorsqu'une calibration trop ancienne est utilisée. Cela montre que la calibration doit être ré-effectuée régulièrement. Il est donc important d'avoir une solution facilement déployable pour des recalibrages réguliers : ce sont des critères auxquels répondent notre méthode.

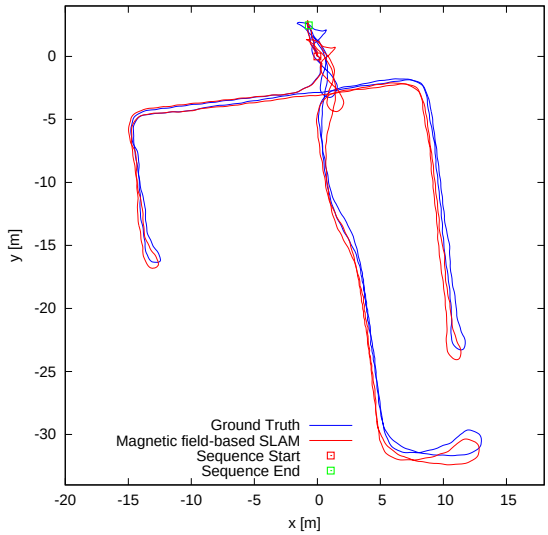
Conclusion

Nous avons présenté ici une méthode permettant de calibrer un magnétomètre en intérieur. Sa précision est suffisante pour obtenir des résultats de localisation presque aussi précis qu'avec une calibration dans un champ magnétique constant. Grâce à cette méthode, il devient plus facile d'utiliser l'algorithme magnéto-visuel-inertiel proposé car l'étape d'initialisation est beaucoup moins complexe à réaliser.

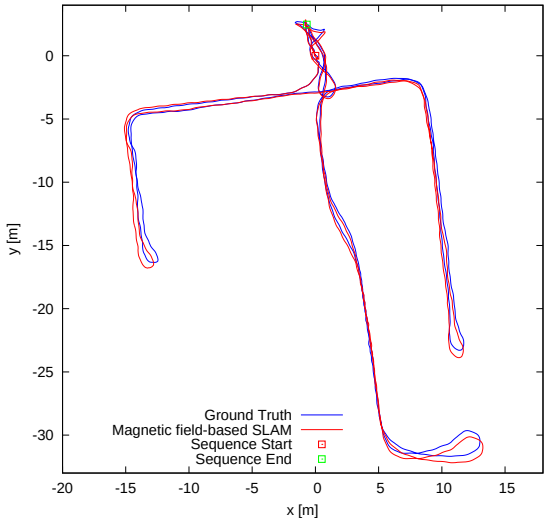


(a) sans magnétomètre

(b) calibration du magnétomètre en extérieur



(c) calibration du magnétomètre [MMLC19]



(d) calibration magnétique proposée ici

FIGURE 5.5: Exécutions du MSCKF magnéto-visuel-inertiel avec différentes méthodes de calibration du magnétomètre (Séquence 1 dans le Tableau 5.2). La trajectoire vérité-terrain est représentée en bleu et les trajectoires SLAM en rouge. En haut à gauche : les mesures magnétiques ne sont pas utilisées et l'algorithme est semblable à une odométrie visuelle-inertielle.

Chapitre 6

Conclusion

6.1 Conclusion

Au cours de cette thèse, de nouvelles avancées pour l'utilisation du magnétomètre pour la localisation en intérieur ont été réalisées. Les variations du champ magnétique étant importantes, on peut utiliser ces valeurs pour caractériser les points de l'espace, comme l'illustre la Figure 6.1. Une carte magnétique, créée en amont ou en ligne, permet de corriger la dérive en position des algorithmes visuels-inertiels de type SLAM quand le système repasse par des zones déjà explorées. Comme la carte magnétique est très stable au cours du temps (pas de sensibilité à la luminosité, aux déplacements d'objets), elle est plus robuste qu'une carte visuelle.

La cohérence des cartes magnétiques obtenues a été améliorée grâce à la prise en compte des différentes sources d'incertitude sur les données : pour obtenir une carte, des valeurs magnétiques sont collectées et associées à la position du système au moment de l'acquisition. Les mesures du magnétomètre mais aussi l'algorithme de localisation ont une incertitude associée. Leur prise en compte simultanée permet d'augmenter la qualité de la modélisation des cartes, qui servent de support ensuite pour la localisation basée sur le magnétomètre.

Un autre apport important est la fusion serrée des données des capteurs magnéto-visuels-inertiels en utilisant la structure d'un filtre de Kalman. Nous avons pu vérifier expérimentalement que l'ajout du magnétomètre permet effectivement d'augmenter la précision de la localisation par rapport aux méthodes visuelles-inertielles. Il est possible de supprimer presque entièrement la dérive en utilisant des cartes magnétiques datant de plus d'un an alors que les performances diminuent significativement avec une ancienne carte visuelle.

Le dernier axe choisi pour l'amélioration de la localisation a été la calibration du magnétomètre : il est crucial de corriger les valeurs du capteur avec exactitude. La calibration en intérieur proposée présente l'avantage d'être précise tout en étant facilement déployable : il n'est pas nécessaire de déplacer le système en extérieur et le temps de calibration requis est court (quelques dizaines de secondes).

La qualité des trajectoires obtenues avec notre algorithme magnéto-visuel-inertiel confirment l'intérêt d'ajouter le magnétomètre comme un capteur supplémentaire pour la localisation embarquée, d'autant plus que c'est un capteur peu encombrant et très répandu. Son application pour la localisation en intérieur est encore récente et de nombreuses pistes d'utilisation se dessinent, avec d'autres capteurs ou d'autres algorithmes. De plus, d'autres usages de la carte magnétique, comme dans le cadre de la relocalisation dans un envrironnement déjà cartographié, sont possibles.

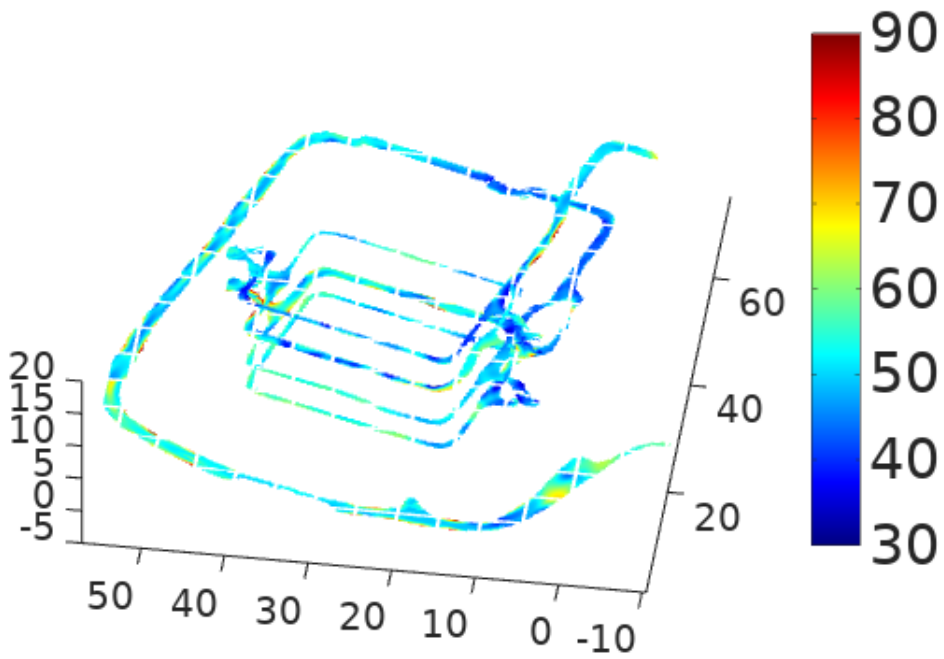


FIGURE 6.1: Norme du champ magnétique des trois étages d'un bâtiment et du terrain l'entourant (prédictions jusqu'à 1 m de la trajectoire).

6.2 Perspectives

6.2.1 Nouveaux algorithmes de localisation intégrant le magnétomètre

Le filtre de Kalman est adapté à la fusion de capteurs et l'intégration du magnétomètre dans le filtre de Kalman multi-états est la solution la plus naturelle pour une fusion serrée magnéto-visuelle-inertielle. Cependant, si on se place dans le cadre de la localisation visuelle-inertielle, l'algorithme VISLAM par optimisation de graphes [CER+21] est le plus précis. Intégrer le magnétomètre d'une manière similaire dans [CER+21] ou ses dérivés [UDS+19, SC22] permettrait sûrement d'améliorer encore la précision de la localisation.

Notre algorithme MSCKF magnéto-visuel-inertiel peut aussi être amélioré : nous savons que les incertitudes de position fournies par le filtre de Kalman sont sous-estimées. Cela a un impact sur la qualité des prédictions du filtre mais aussi sur la carte magnétique associée, qui prend en compte ces incertitudes. Le filtre de Kalman invariant permet de corriger la trop grande confiance dans ses prédictions du filtre de Kalman classique [BB17] : il serait intéressant d'étudier les résultats obtenus avec des covariances plus réalistes que celles que nous avons utilisées jusqu'à présent.

6.2.2 Relocalisation

Comme évoqué dans les parties précédentes, certains travaux portant sur la relocalisation dans une carte déjà connue ont déjà intégré les données magnétiques comme facteur discriminant [LAH18, MTSB16, LZL+17, WWP+19]. Mais ces méthodes manquent encore de précision et n'ont donc pas été intégrées dans les travaux présentés ici. La localisation absolue a été obtenue à partir de mesures GNSS en extérieur au début de la séquence dans les cas de réutilisation d'une

ancienne carte magnétique. Une perspective intéressante serait d'appliquer d'autres méthodes de sélection du meilleur candidat pour la relocalisation. Une difficulté reste la méconnaissance de l'orientation du système. Comme de nombreux endroits ont un champ magnétique d'intensité similaire, il est impératif de coupler la relocalisation magnétique avec un autre capteur : la Figure 6.2 illustre ce problème. Cela a été proposé dans [LZL+17, WWP+19] mais dans des conditions très limitées (espace divisé en carrés de 60 m de côté pour [LZL+17], uniquement deux directions possibles pour [WWP+19]).

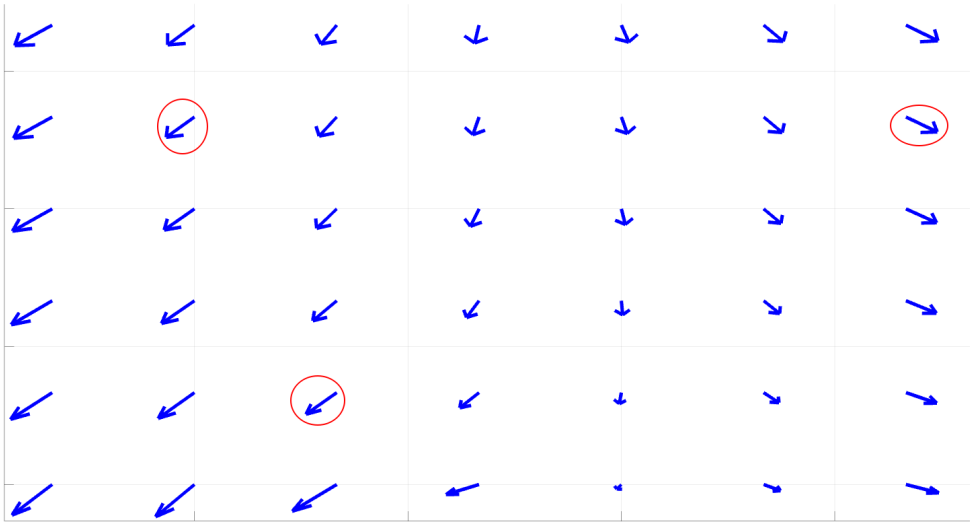


FIGURE 6.2: Exemple de champ magnétique avec ses orientations différentes (données simulées). Les cercles rouges indiquent plusieurs candidats possibles pour la relocalisation : le champ magnétique à ces trois endroits a une norme similaire.

6.2.3 Prise en compte des ascenseurs et autres perturbations magnétiques

Le magnétomètre, bien qu'il ne soit pas perturbé par l'aspect visuel de l'environnement, peut tout de même renvoyer des mesures ne correspondant pas à ce que l'on croit mesurer. Il est important d'étudier comment ne pas les prendre en compte, pour éviter des corrections erronées de la trajectoire : cela est un axe de recherche important pour pouvoir assurer l'usage du magnétomètre dans un nombre de contextes aussi variés que possible. Par exemple, si un aimant est placé très près du magnétomètre, le capteur mesurera le champ lié à l'aimant en plus du champ magnétique généré par le bâtiment et sa structure. S'il est proche de roues dont l'alimentation électrique varie, ses mesures vont aussi être perturbées et ces modifications seront négligeables ou non en fonction de la puissance des variations du courant transmis. Il est fondamental de prendre en compte ces problèmes lors de l'installation du magnétomètre sur un système : la calibration ne prend en compte que les modifications constantes du champ.

Pour les déplacements dans le contexte d'une usine ou proche d'un ascenseur, les mesures du magnétomètre doivent être ignorées par la carte magnétique, car elles ne sont pas pérennes : il suffit que l'ascenseur bouge pour que le magnétomètre renvoie des valeurs très différentes de celles obtenues précédemment. Des solutions permettant de supprimer ces valeurs aberrantes sont nécessaires si l'on veut pouvoir obtenir un système de localisation robuste en toutes circonstances.

6.2.4 Applications du modèle de création de carte à d'autres variables que le champ magnétique

La méthode de modélisation de carte prenant en compte les incertitudes de localisation est applicable à d'autres types de données que le champ magnétique. Elle est généralisable à des variables de plus grandes ou plus petites dimensions. Les méthodes prenant en compte l'incertitude sur les données d'entrée, dans le cadre de la modélisation par processus gaussiens, sont appliquées dans des contextes très différents. Dans [KPPB07], la méthode proposée, qui consiste à optimiser le bruit blanc pour chaque mesure, est appliquée à des données LIDAR réelles. Une carte de la hauteur de la couche d'ozone sur Terre dans [CC03] et de la température dans [CP15] est obtenue à partir de données satellites. Les applications sont donc nombreuses, en particulier pour la cartographie en ligne, pour différents types de données. Cela permet de pouvoir constituer une carte avec des incertitudes plus précises.

Annexe A

Dérivées des fonctions propres (processus gaussien à rang réduit)

Les fonctions propres ϕ utilisées dans la décomposition de la covariance à rang réduit (voir Chapitre 3 et Equation 3.13 pour la dimension 3 ; ici, on se place en dimension finie d). Le calcul de l'espérance et de la covariance dépend de leurs dérivées de l'ordre 1 à 3. On détaille ici le calcul des différentes dérivées.

On se restreint à ϕ_j pour les calculs, qui se généralisent ensuite à toutes ces fonctions.

$$\phi_j(\mathbf{x}) = \prod_{k=1}^{d} \frac{1}{\sqrt{L_k}} \sin\left(\frac{\pi n_{j,k}(\mathbf{x}_k + L_k)}{2L_k}\right) \tag{A.1}$$

L_k et $n_{j,k}$ sont des constantes définies dans la Sous-Section 3.1.4. $\mathbf{x}$ correspond aux coordonnées du point étudié et est de dimension 3.

A.1 Dérivée première

On commence d'abord par calculer la dérivée première, dont le calcul est très simple. On obtient un vecteur de taille 3 dont le coefficient à la ligne p est égal à :

$$\left(\nabla\phi_j(\mathbf{x})\right)_p = \frac{\pi n_{j,p}}{2L_p\sqrt{L_p}} \cos\left(\frac{\pi n_{j,p}(\mathbf{x}_p + L_p)}{2L_p}\right) \prod_{k=1,k\neq p}^{d} \frac{1}{\sqrt{L_k}} \sin\left(\frac{\pi n_{j,k}(\mathbf{x}_k + L_k)}{2L_k}\right). \tag{A.2}$$

A.2 Dérivée seconde

La dérivée seconde est une matrice. Elle s'obtient en dérivant l'expression de la dérivée première. Pour obtenir l'expression de chaque élément de coordonnées (p,q), on a deux cas distincts.

Si $p \neq q$:

$$\left(H(\phi_j)(\mathbf{x})\right)_{p,q} = \prod_{r=1,r\neq p,r\neq q}^{d} \frac{1}{\sqrt{L_r}} \sin\left(\frac{\pi n_{j,r}(\mathbf{x}_r + L_r)}{2L_r}\right) \prod_{k=p,q} \frac{\pi n_{j,k}}{2L_k\sqrt{L_k}} \cos\left(\frac{\pi n_{j,k}(\mathbf{x}_k + L_k)}{2L_k}\right) \tag{A.3}$$

Si $p = q$:

$$\Big(H(\phi_j)(\mathbf{x})\Big)_{p,q} = -\left(\frac{\pi n_{j,p}}{2L_p}\right)^2 \prod_{k=1}^{d} \frac{1}{\sqrt{L_k}} \sin\left(\frac{\pi n_{j,k}(\mathbf{x}_k + L_k)}{2L_k}\right) \tag{A.4}$$

A.3 Généralisation

À partir des expressions déjà obtenues des dérivées, on peut formuler une formule générale, que l'on prouve par récurrence.

L'hypothèse est que chaque dérivée de la fonction $\phi_j(\mathbf{x})$ peut s'écrire, avec (a_i), $i \in \{1, ..., d\}$ une suite d'entiers :

$$\boxed{\frac{\partial^{a_1+a_2+...+a_d}\phi_j(\mathbf{x})}{(\partial \mathbf{x}_1)^{a_1}(\partial \mathbf{x}_2)^{a_2}...(\partial \mathbf{x}_d)^{a_d}} = \prod_{k=1}^{3} \frac{1}{\sqrt{L_k}} \left(\frac{\pi n_{j,k}}{2L_k}\right)^{a_k} (-1)^{\lfloor \frac{a_k}{2} \rfloor} f_{a_k}\left(\frac{\pi n_{j,k}(\mathbf{x}_k + L_k)}{2L_k}\right)} \tag{A.5}$$

avec $f_{a_k} = \sin$ si a_k est pair et $f_{a_k} = \cos$ si a_k est impair.

Pour l'initialisation, à l'ordre 0 ($a_i = 0$ pour tout $i \in \{1, ..., d\}$), on a bien :

$$\phi_j(\mathbf{x}) = \prod_{k=1}^{d} \frac{1}{\sqrt{L_k}} \left(\frac{\pi n_{j,k}}{2L_k}\right)^{0} (-1)^0 f_0\left(\frac{\pi n_{j,k}(\mathbf{x}_k + L_k)}{2L_k}\right) \tag{A.6}$$

$$= \prod_{k=1}^{d} \frac{1}{\sqrt{L_k}} \sin\left(\frac{\pi n_{j,k}(\mathbf{x}_k + L_k)}{2L_k}\right). \tag{A.7}$$

Pour la récurrence, on dérive l'Equation A.5 selon la première dimension de $\mathbf{x}$:

$$\frac{\partial^{a_1+a_2+...+a_d}\phi_j(\mathbf{x})}{(\partial \mathbf{x}_1)^{a_1+1}(\partial \mathbf{x}_2)^{a_2}(\partial \mathbf{x}_3)^{a_3}} = \frac{\partial}{\partial \mathbf{x}_1}\left(\prod_{k=1}^{3} \frac{1}{\sqrt{L_k}} \left(\frac{\pi n_{j,k}}{2L_k}\right)^{a_k} (-1)^{\lfloor \frac{a_k}{2} \rfloor} f_{a_k}\left(\frac{\pi n_{j,k}(\mathbf{x}_k + L_k)}{2L_k}\right)\right) \tag{A.8}$$

$$\tag{A.9}$$

Seuls les termes dépendant de $\mathbf{x}_1$ sont modifiés par la dérivée : les autres coefficients du produit restent identiques.

$$\begin{aligned}\frac{\partial^{a_1+1+a_2+...+a_d}\phi_j(\mathbf{x})}{(\partial \mathbf{x}_1)^{a_1+1}(\partial \mathbf{x}_2)^{a_2}...(\partial \mathbf{x}_d)^{a_d}} = \frac{\partial}{\partial \mathbf{x}_1}&\left(\frac{1}{\sqrt{L_1}} \left(\frac{\pi n_{j,1}}{2L_1}\right)^{a_1} (-1)^{\lfloor \frac{a_1}{2} \rfloor} f_{a_1}\left(\frac{\pi n_{j,1}(\mathbf{x}_1 + L_1)}{2L_1}\right)\right) \\ &* \left(\prod_{k=2}^{d} \frac{1}{\sqrt{L_k}} \left(\frac{\pi n_{j,k}}{2L_k}\right)^{a_k} (-1)^{\lfloor \frac{a_k}{2} \rfloor} f_{a_k}\left(\frac{\pi n_{j,k}(\mathbf{x}_k + L_k)}{2L_k}\right)\right)\end{aligned} \tag{A.10}$$

Si a_1 est pair, $f_{a_1} = \sin$ et la première partie de l'équation est de la forme suivante :

$$\begin{aligned}&\frac{\partial}{\partial \mathbf{x}_1}\left(\frac{1}{\sqrt{L_1}} \left(\frac{\pi n_{j,1}}{2L_1}\right)^{a_1} (-1)^{\lfloor \frac{a_1}{2} \rfloor} \sin\left(\frac{\pi n_{j,1}(\mathbf{x}_1 + L_1)}{2L_1}\right)\right) \\ &= \left(\frac{1}{\sqrt{L_1}} \left(\frac{\pi n_{j,1}}{2L_1}\right)^{a_1+1} (-1)^{\lfloor \frac{a_1}{2} \rfloor} \cos\left(\frac{\pi n_{j,1}(\mathbf{x}_1 + L_1)}{2L_1}\right)\right)\end{aligned} \tag{A.11}$$

$\lfloor \frac{a_1}{2} \rfloor = \lfloor \frac{a_1+1}{2} \rfloor$ car a_1 est pair :

$$\begin{aligned}
&\frac{\partial}{\partial \mathbf{x}_1}\left(\frac{1}{\sqrt{L_1}}\left(\frac{\pi n_{j,1}}{2L_1}\right)^{a_1}(-1)^{\lfloor \frac{a_1}{2} \rfloor}\sin\left(\frac{\pi n_{j,1}(\mathbf{x}_1+L_1)}{2L_1}\right)\right)\\
&\quad=\left(\frac{1}{\sqrt{L_1}}\left(\frac{\pi n_{j,1}}{2L_1}\right)^{a_1+1}(-1)^{\lfloor \frac{a_1+1}{2} \rfloor} f_{a_1+1}\left(\frac{\pi n_{j,1}(\mathbf{x}_1+L_1)}{2L_1}\right)\right).
\end{aligned} \tag{A.12}$$

Si a_1 est impair, $f_{a_1} = \cos$, $\lfloor \frac{a_1}{2} \rfloor + 1 = \lfloor \frac{a_1+1}{2} \rfloor$ et la première partie de l'équation est de la forme suivante :

$$\begin{aligned}
&\frac{\partial}{\partial \mathbf{x}_1}\left(\frac{1}{\sqrt{L_1}}\left(\frac{\pi n_{j,1}}{2L_1}\right)^{a_1}(-1)^{\lfloor \frac{a_1}{2} \rfloor}\cos\left(\frac{\pi n_{j,1}(\mathbf{x}_1+L_1)}{2L_1}\right)\right)\\
&=\left(\frac{1}{\sqrt{L_1}}\left(\frac{\pi n_{j,1}}{2L_1}\right)^{a_1+1}(-1)^{\lfloor \frac{a_1}{2} \rfloor}(-1)\sin\left(\frac{\pi n_{j,1}(\mathbf{x}_1+L_1)}{2L_1}\right)\right)\\
&=\left(\frac{1}{\sqrt{L_1}}\left(\frac{\pi n_{j,1}}{2L_1}\right)^{a_1+1}(-1)^{\lfloor \frac{a_1+1}{2} \rfloor} f_{a_1+1}\left(\frac{\pi n_{j,1}(\mathbf{x}_1+L_1)}{2L_1}\right)\right).
\end{aligned} \tag{A.13}$$

Dans les deux cas, l'Equation A.5 est vérifiée. Comme toutes les dimensions $\mathbf{x}_1$, $\mathbf{x}_2$ ... $\mathbf{x}_d$ sont interchangeables dans l'expression, l'hypothèse est vérifiée pour n'importe quelle suite d'entiers positifs.

Bibliographie

[ACCM21] Y. Ahmine, G. Caron, F. Chouireb, and E. M. Mouaddib. Continuous scale-space direct image alignment for visual odometry from rgb-d images. *Robotics and Automation Letters*, 6(2) :2264–2271, 2021.

[BB05] D.S. Bayard and P.B. Brugarolas. An estimation algorithm for vision-based exploration of small bodies in space. In *American Control Conference*, volume 7, pages 4589–4595, 2005.

[BB17] A. Barrau and S. Bonnabel. The invariant extended kalman filter as a stable observer. *IEEE Transactions on Automatic Control*, 62(4) :1797–1812, 2017.

[BSvV16] H. Bijl, T. B. Schön, J.-W. van Wingerden, and M. Verhaegen. System identification through online sparse gaussian process regression with input noise. *IFAC Journal of Systems and Control*, 2 :1–11, 2016.

[BTG06] H. Bay, T. Tuytelaars, and L.V. Gool. Surf : Speeded up robust features. In *European Conference on Computer Vision*, volume 3951, pages 404–417, July 2006.

[BvSV15] H. Bijl, J.-W. van Wingerden, T. Schön, and M. Verhaegen. Online sparse gaussian process regression using fitc and pitc approximations. *IFAC Symposium on System Identification*, 48(28) :703–708, 2015.

[CC03] N. A. Cressie and J. Cornac. Spatial statistics in the presence of location error with an application to remote sensing of the environment remote sensing of the environment. *Statistical Science*, 18(4) :436–456, 2003.

[CCC+16] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. Leonard. Past, present, and future of simultaneous localization and mapping : Toward the robust-perception age. *IEEE Transactions on Robotics*, 32(6) :1309–1332, 2016.

[CDS+11] J. Chung, M. Donahoe, C. Schmandt, I. Kim, P. Razavai, and M. Wiseman. Indoor location sensing using geo-magnetism. In *International Conference on Mobile Systems, Applications, and Services*, pages 141–154, June 2011.

[CER+21] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel, and J. D. Tardós pp. 1874-1890. ORB-SLAM3 : An Accurate Open-Source Library for Visual, Visual–Inertial, and Multimap SLAM. *IEEE Transactions on Robotics*, 37(6) :1874–1890, December 2021.

[CES+17] D. Caruso, A. Eudes, M. Sanfourche, D. Vissière, and G. Le Besnerais. A Robust Indoor/Outdoor Navigation Filter Fusing Data from Vision and Magneto-Inertial Measurement Unit. *Sensors*, 17(12) :2795, December 2017.

[CK18] L. Clement and J. Kelly. How to train a cat : Learning canonical appearance transformations for direct visual localization under illumination change. *IEEE Robotics and Automation Letters (RA-L)*, 3 :2447–2454, July 2018.

[CP15] D. Cervone and N. S. Pillai. Gaussian process regression with location errors. *preprint, ArXiv*, 2015.

[DBCd09] P. Dallaire, C. Besse, and B. Chaib-draa. Learning gaussian process models from uncertain data. *Neural Information Processing*, pages 433–440, December 2009.

[DLY+16] T.-N. Do, R. Liu, C. Yuen, M. Zhang, and U-X. Tan. Personal dead reckoning device via imu sensor on upper torso and inverted pendulum model. *IEEE Sensors Journal*, 16 :1–10, November 2016.

[DRMS07] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse. Monoslam : Real-time single camera slam. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6) :1052–1067, June 2007.

[DU22] P. Tripicchio S. D'Avella and M. Unetti. Efficient localization in warehouse logistics : a comparison of lms approaches for 3d multilateration of passive uhf rfid tags. *The International Journal of Advanced Manufacturing Technology volume*, pages 4977–4988, 2022.

[DVP08] A. Martin D. Vissière and N. Petit. Système fournissant la vitesse et la position d'un corps en utilisant les variations du champ magnétique évaluées grâce aux mesures de un ou des magnétomètres et de une ou des centrales inertielles, 2008. Patent FR2914739 (A1).

[FAL+13] M. Frassl, M. Angermann, M. Lichtenstern, P. Robertson, BJ. Julian, and M. Doniec. Magnetic maps of indoor environments for precise localization of legged and non-legged. In *International Conference on Intelligent Robots and Systems*, pages 913–920, November 2013.

[FB81] Martin A. Fischler and Robert C. Bolles. Random sample consensus : A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6) :381–395, June 1981.

[FBS12] P. Furgale, T. D. Barfoot, and G. Sibley. Continuous-time batch estimation using temporal basis functions. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2088–2095, May 2012.

[FCDS15] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza. IMU Preintegration on Manifold for Efficient Visual-Inertial Maximum-a-Posteriori Estimation. In *Robotics : Science and Systems (RSS)*, July 2015.

[FDD18] Giuseppe Fedele, Luigi D'Alfonso, and Gaetano D'Aquila. Magnetometer bias finite-time estimation using gyroscope data. *IEEE Transactions on Aerospace and Electronic Systems*, 54(6) :2926–2936, May 2018.

[FE08] C. C. Foster and G. H. Elkaim. Extension of a two-step calibration methodology to include nonorthogonal sensor axes. *IEEE Transactions on Aerospace and Electronic Systems*, 44(3) :1070–1078, October 2008.

[FWM+21] L. Flueratoru, S. Wehrli, M. Magno, E. S. Lohan, and D. Niculescu. High-accuracy ranging and localization with ultrawideband communications for energy-constrained devices. *IEEE Internet of Things Journal*, 9(10) :7463–7480, 2021.

[GLT12] D. Galvez-Lopez and J. D. Tardos. Bags of Binary Words for Fast Place Recognition in Image Sequences. *IEEE Transactions on Robotics*, 28(5) :1188–1197, May 2012.

[GWB11] P. W. Goldberg, C. K. I. Williams, and C. M. Bishop. Regression with input-dependent noise : A gaussian process treatment (neurips). *Advances in Neural Information Processing Systems*, 10 :493–499, December 2011.

[HABD19] Z. Hashemifar, C. Adhivarahan, A. Balakrishnan, and K. Dantu. Augmenting Visual SLAM with Wi-Fi Sensing For Indoor Applications. *Autonomous Robots*, 43(8) :2245–2260, July 2019.

[Hem03] E. E. Hemayed. A survey of camera self-calibration. In *IEEE Conference on Advanced Video and Signal Based Surveillance, 2003.*, pages 351–357, July 2003.

[HGJL17] S. Huang, O. P. Gan, S. Jose, and M. Li. Localization for industrial warehouse storage rack using passive uhf rfid system. *IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 1–8, September 2017.

[JPTD08] Y. Pavlidis J.-P. Tardif and K. Daniilidis. Monocular visual odometry in urban environments using an omnidirectional camera. *IEEE International Conference on Intelligent Robots and Systems*, 2008.

[KKZA22] K. Kostas, R. Y. Kostas, F. Zampella, and F. Alsehly. Wifi based distance estimation using supervised machine learning. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, September 2022.

[KPPB07] K. Kersting, C. Plagemann, P. Pfaff, and W. Burgard. Most-likely heteroscedastic gaussian process regression. *ACM International Conference Proceeding Series*, 227 :393–400, January 2007.

[KS16] M. Kok and T. Schön. Magnetometer Calibration Using Inertial Sensors. *IEEE Sensors Journal*, 16(14) :5679–5689, May 2016.

[KS18] M. Kok and A. Solin. Scalable Magnetic Field SLAM in 3D Using Gaussian Process Maps. In *FUSION*, July 2018.

[LAH18] N. Lee, S. Ahn, and D. Han. AMID : Accurate Magnetic Indoor Localization Using Deep Learning. *Sensors*, 18(5) :1598–1613, May 2018.

[LAO04] J. Korhonen L. Aalto, N. Göthlin and T. Ojala. Bluetooth and wap push based location-aware mobile advertising system. *MobiSys '04 : Proceedings of the 2nd international conference on Mobile systems, applications, and services*, pages 49–58, June 2004.

[LCT07] I. Reid J. Neira L. Clemente, A. Davison and J. D. Tardos. Mapping large loops with a single hand-held camera. In *Robotics : Science and Systems (RSS)*, June 2007.

[lEA10] A. De la Escalera and J. M. Armingol. Automatic chessboard detection for intrinsic and extrinsic camera parameter calibration. *Sensors*, 10(3) :2027–2044, 2010.

[LGDR12] B. Li, T. Gallagher, A. G. Dempster, and C. Rizos. How feasible is the use of magnetic field alone for indoor positioning ? In *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, November 2012.

[LM13] M. Li and A. I. Mourikis. High-precision, consistent ekf-based visual-inertial odometry. *International Journal of Robotics Research*, 32(6) :690–711, May 2013.

[LM22] M. Labbé and F. Michaud. Multi-Session Visual SLAM for Illumination Invariant Re-Localization in Indoor Environments. *Frontiers in Robotics and AI*, 9(115), March 2022.

[Low04] D. G. Lowe. Distinctive image features from scale-invariant keypoints. In *International Journal of Computer Vision*, pages 1573–1405, November 2004.

[LS13] Furgale P. Rabaud V. Chli M. Konolige K. Leutenegger, S. and R. Siegwart. Keyframe-based visual-inertial slam using nonlinear optimization. *Robotics : Science and Systems (RSS)*, 2013.

[LYZM14] M. Li, H. Yu, X. Zheng, and A. Mourikis. High-fidelity sensor modeling and self-calibration in vision-aided inertial navigation. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 409–416, June 2014.

[LZL+17] Z. Liu, L. Zhang, Q. Liu, Y. Yin, L. Cheng, and R. Zimmermann. Fusion of Magnetic and Visual Sensors for Indoor Localization : Infrastructure-Free and More Effective. *IEEE Transactions on Multimedia*, 19(4) :874–888, April 2017.

[LZZ+19] Y. Li, S. Zahran, Y. Zhuang, Z. Gao, Y. Luo, Z. He, L. Pei, R. Chen, and N. El-Sheimy. Imu/magnetometer/barometer/mass-flow sensor integrated indoor quadrotor uav localization with robust velocity updates. *Remote Sensing*, 11(7), 2019.

[MFMR16] R. Martins, E. Fernandez-Moral, and P. Rives. Adaptive Direct RGB-D Registration and Mapping for Large Motions. In *Asian Conference on Computer Vision, ACCV 2016*, November 2016.

[ML96] L. Mengxiang and J-M. Lavest. Some aspects of zoom lens camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18 :1105–1110, 1996.

[MMLC19] M. Muraccini, A. L. Mangia, M. Lannocca, and A. Cappello. Magnetometer calibration and field mapping through thin plate splines. *Sensors*, 19(2) :280–295, 2019.

[MR07] A. I. Mourikis and S. I. Roumeliotis. A Multi-State Constraint Kalman Filter for Vision-aided Inertial Navigation. In *IEEE International Conference on Robotics and Automation (ICRA)*, April 2007.

[MR11] A. Mchutchon and C. Rasmussen. Advances in neural information processing systems (neurips). In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 24, page 1341–1349. Curran Associates, Inc., 2011.

[MR22] M. Müller and C. Röhrig. Angle of arrival estimation in the presence of phase difference ambiguities. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, September 2022.

[MTSB16] R. Montoliu, J. Torres-Sospedra, and O. Belmonte. Magnetic field based Indoor positioning using the Bag of Words paradigm. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–7. IEEE, October 2016.

[Nis04] D. Nister. An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6) :756–770, 2004.

[NJD+17] M. Nowakowski, C. Joly, S. Dalibard, N. Garcia, and F. Moutarde. Topological localization using wi-fi and vision merged into fabmap framework. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2017.

[NYC+21] T. Nguyen, S. Yuan, M. Cao, T. Nguyen, and L. Xie. VIRAL SLAM : Tightly Coupled Camera-IMU-UWB-LIDAR SLAM. May 2021. early access.

[OVK22] M. Osman, F. Viset, and M. Kok. Indoor slam using a foot-mounted imu and the local magnetic field. pages 1–7, July 2022.

[PGGB20] B. Petit, R. Guillemard, and V. Gay-Bellile. Time shifted imu preintegration for temporal calibration in incremental visual-inertial initialization. In *International Conference on 3D Vision (3DV)*, pages 171–179, November 2020.

[RAL10] V. Renaudin, M. H. Afzal, and G. Lachapelle. Complete triaxis magnetometer calibration in the magnetic domain. *Journal of Sensors*, 17(6) :1223–1232, December 2010.

[RFA+13] P. Robertson, M. Frassl, M. Angermann, M. Doniec, B. J. Julian, M. G. Puyol, M. Khider, M. Lichtenstern, and L. Bruno. Simultaneous Localization and Mapping for pedestrians using distortions of the local magnetic field intensity in large indoor environments. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, October 2013.

[RJPK05] A. Watkins R. J. Prazenica and A. J. Kurdila. Vision-based kalman filtering for aircraft state estimation and structure from motion. In *AIAA Guidance, Navigation, and Control Conference*, number 6003, August 2005.

[RMA14] Juan D. Tardos Raul Mur-Artal. Fast relocalisation and loop closing in keyframe-based slam. pages 846–853, 2014.

[RNS+16] J. Rehder, J. Nikolic, T. Schneider, T. Hinzmann, and R. Siegwart. Extending *kalibr* : Calibrating the Extrinsics of Multiple IMUs and of Individual Axes. *ICRA*, May 2016.

[RRKB11] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. Orb : An efficient alternative to sift or surf. In *International Conference on Computer Vision (ICCV)*, pages 2564–2571, November 2011.

[RSS11] I. R. Nourbakhsh R. Siegwart and D. Scaramuzza. *Introduction to Autonomous Mobile Robots*. MIT Press, 2011.

[RW06] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes For Machine Learning*, chapter 2. MIT Press, 2006.

[S. 13] S. Särkkä. *Bayesian Filtering and Smoothing*. Cambridge University Press, 2013.

[SC22] L. Stumberg and D. Cremers. Dm-vio : Delayed marginalization visual-inertial odometry. *IEEE Robotics and Automation Letters*, PP :1–8, January 2022.

[SDS13] V. S. Sokolovic, G. Dikic, and R. Stancic. Integration of ins, gps, magnetometer and barometer for improving accuracy navigation of the vehicle. *Defence Science Journal*, 63(5) :451–455, 2013.

[SEM+20] Tixiao Shan, Brendan Englot, Drew Meyers, Wei Wang, Carlo Ratti, and Daniela Rus. Lio-sam : Tightly-coupled lidar inertial odometry via smoothing and mapping. In *International Conference on Intelligent Robots and Systems*, pages 5135–5142. IEEE, 2020.

[She14] M. A. Shelley. *Monocular Visual Inertial Odometry on a Mobile Device*. PhD thesis, 2014.

[SKW+15] A. Solin, M. Kok, N. Wahlström, T. Schön, and S. Särkkä. Modeling and Interpolation of the Ambient Magnetic Field by Gaussian Processes. *IEEE Transactions on Robotics*, 34(4) :1112–1127, September 2015.

[SS20] A. Solin and S. Särkkä. Hilbert space methods for reduced-rank gaussian process regression. *Statistics and Computing*, 30(2) :419–446, 2020.

[SSH21] B. Siebler, S. Sand, and U. D. Hanebeck. Localization With Magnetic Field Distortions and Simultaneous Magnetometer Calibration. *IEEE Sensors Journal*, 21(3) :3388–3397, February 2021.

[Str04] D. Strelow. *Motion estimation from image and inertial measurements*. PhD thesis, 2004.

[SUS+20] C. Sommer, V. Usenko, D. Schubert, N. Demmel, and D. Cremers. Efficient derivative computation for cumulative b-splines on lie groups. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11145–11153, June 2020.

[TW13] G. Troni and L. Whitcomb. Adaptive Estimation of Measurement Bias in Three-Dimensional Field Sensors with Angular Rate Sensors : Theory and Comparative Experimental Evaluation. In *Robotics : Science and Systems (RSS)*, June 2013.

[TW19] J. Tiemann and C. Wietfeld. Scalability, real-time capabilities, and energy efficiency in ultra-wideband localization. *IEEE Transactions on Industrial Informatics*, 15(12) :6313–6321, 2019.

[UDS+19] V. Usenko, N. Demmel, D. Schubert, J. Stueckler, and D. Cremers. Visual-Inertial Mapping With Non-Linear Factor Recovery. *IEEE Robotics and Automation Letters*, 5 :422–429, December 2019.

[Val18] I. Vallivaara. *Simultaneous localization and Mapping using the indooor magnetic field*. PhD thesis, 2018.

[VBH12] M. Vágner, P. Beneš, and Z. Havránek. Experience with allan variance method for mems gyroscope performance characterization. *Instrumentation and Measurement Technology Conference*, 2012.

[VES+11] J. F. Vasconcelos, G. Elkaim, C. Silvestre, P. Oliveira, and B. Cardeira. Geometric Approach to Strapdown Magnetometer Calibration in Sensor Frame. *IEEE Transactions on Aerospace and Electronic Systems*, 47(2) :1293–1306, April 2011.

[VES+15] J. F. Vasconcelos, G. Elkaim, C. Silvestre, P. Oliveira, and B. Cardeira. On Calibration of Three-Axis Magnetometer. *IEEE Sensors Journal*, 15(11) :6424–6431, November 2015.

[VHK22] F. Viset, R. Helmons, and M. Kok. An extended kalman filter for magnetic field slam using gaussian process regression. *Sensors*, 22 :2833, April 2022.

[VHKR11] I. Vallivaara, J. Haverinen, A. Kemppainen, and J. Roning. Magnetic field-based SLAM method for solving the localization problem in mobile robot floor-cleaning task. *International conference on Advanced Robotics*, 2011.

[Wah15] N. Wahlström. *Modeling of Magnetic Fields and Extended Objects for Localization Applications*. PhD thesis, 2015.

[WKTG13] N. Wahlström, M. Kok, T. Schön Thomas, and F. Gustafsson. Modeling magnetic fields using gaussian processes. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3522–3526, 2013.

[WWP+19] Z. Wu, M. Wen, G. Peng, X. Tang, and D. Wang. Magnetic-Assisted Initialization for Infrastructure-free Mobile Robot Localization. In *International Conference on Cybernetics and Intelligent Systems and Conference on Robotics, Automation and Mechatronics*, pages 518–523, November 2019.

RÉSUMÉ

Un des enjeux majeurs pour la robotique mobile est de pouvoir localiser précisément un robot dans des environnements intérieurs où le GPS n'est pas disponible. Les solutions les plus précises de l'état de l'art actuel reposent sur l'association des données issues de caméras et d'une centrale inertielle : ce sont les algorithmes VISLAM (Visual Inertial Simultaneous Localization And Mapping). Néanmoins la localisation obtenue avec ce type d'algorithmes dérive au cours du temps. Une carte visuelle créée au préalable peut être utilisée pour limiter cette dérive mais celle-ci devient rapidement obsolète quand la luminosité est modifiée ou que les objets ont changé d'emplacement. Cette thèse porte sur l'amélioration d'un algorithme de type VISLAM en exploitant les données issues d'un magnétomètre. En effet, en intérieur, le champ magnétique varie spatialement à cause des nombreuses perturbations occasionnées par les éléments métalliques constituant la structure des bâtiments. Cette propriété peut être utilisée pour créer des cartes magnétiques de la zone étudiée et les exploiter pour améliorer la précision des algorithmes VISLAM. Nous démontrons notamment au cours de cette thèse que les cartes magnétiques sont beaucoup plus stables que les cartes visuelles au cours du temps.

Deux autres problématiques ayant une influence directe sur la précision de la localisation de notre algorithme de SLAM visuel-inertiel-magnétique ont également été étudiées : il s'agit de la modélisation du champ magnétique et de la calibration du magnétomètre. Pour la modélisation du champ magnétique, nous proposons une solution permettant d'améliorer significativement la qualité des cartes magnétiques reconstruites via la prise en compte de l'incertitude sur la position associée à chaque donnée magnétique. Une méthode de calibration en intérieur donnant des résultats précis est proposée dans cette thèse, permettant ainsi d'éviter le fastidieux processus de calibration en extérieur utilisé jusqu'à présent et ceci sans dégrader la qualité de la calibration.

MOTS CLÉS

Localisation, Fusion de capteurs, SLAM, Champ magnétique en intérieur, MSCKF, Processus Gaussien.

ABSTRACT

One of the major challenges for mobile robotics is to localize precisely a robot in indoor environments where the GPS signal is not available. The current state of the art indoor localization algorithms with the highest accuracy consist in the fusion of the data from visual and inertial sensors: these are called VISLAM (Visual Inertial Simultaneous Localization And Mapping) algorithms. However, the localization obtained drifts rapidly over time. A visual map created beforehand limits this effect but it quickly becomes obsolete if the illumination has changed or if some objects were removed or displaced. This thesis focuses on the improvement of a VISLAM algorithm by using data from a magnetometer. Indoor, the magnetic field varies spatially due to the numerous disturbances caused by the metallic elements constituting the structure of buildings. This property can be used to create magnetic maps of the studied area and exploit them to improve the accuracy of VISLAM algorithms. In particular, we demonstrate in this thesis that magnetic maps are much more stable than visual maps over time.

Two other issues having a direct influence on the localization accuracy of our visual-inertial-magnetic SLAM algorithm have also been studied: magnetic field modeling and magnetometer calibration. For the magnetic field modeling, we propose a solution that significantly improves the reconstruction quality of the magnetic maps by taking into account the uncertainty on the position associated to each magnetic data. A new method for indoor calibration with high precision is proposed in this thesis, enabling to avoid the tedious process of outdoors calibration used until now without degrading the quality of the calibration.

KEYWORDS

Localization, Sensor fusion, SLAM, Indoor magnetic field, MSCKF, Gaussian process.